Beyhan Adanur BLOCKCHAIN BASED DATA SHARING PLATFORM FOR BIOINFIRMATICS FIELD chainAGU 2020

BLOCKCHAIN BASED DATA SHARING PLATFORM FOR BIOINFORMATICS FIELD

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF ABDULLAH GUL UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER SCIENCE

By
Beyhan ADANUR
June 2020

BLOCKCHAIN BASED DATA SHARING PLATFORM FOR BIOINFORMATICS FIELD

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE OF ABDULLAH GUL UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER SCIENCE

By
Beyhan ADANUR
June 2020

SCIENTIFIC ETHICS COMPLIANCE

I hereby declare that all information in this document has been obtained in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name-Surname: Beyhan ADANUR
Signature :

REGULATORY COMPLIANCE

M.Sc. thesis title **"Blockchain Based Data Sharing Platform For Bioinformatics Field"** has been prepared in accordance with the Thesis Writing Guidelines of the Abdullah Gül University, Graduate School of Engineering & Science.

Prepared By
Beyhan ADANUR
Signature

Advisor
Assist. Prof. Dr. Burcu BAKIR-GÜNGÖR
Signature

Co-supervisor
Assist. Prof. Dr. Ahmet SORAN
Signature

Head of the Electrical and Computer Engineering Program
Prof. Dr.Vehbi Çağrı GÜNGÖR
Signature

ACCEPTANCE AND APPROVAL

M.Sc. thesis titled **"Blockchain Based Data Sharing Platform For Bioinformatics Field"** and prepared by Beyhan ADANUR has been accepted by the jury in the Electrical and Computer Engineering Graduate Program at Abdullah Gül University, Graduate School of Engineering & Science.

15 / 06 / 2020
(Thesis Defense Exam Date)

JURY: **Signature:**

Advisor : Assist. Prof. Burcu Bakır-Güngör

Co-supervisor : Assist. Prof. Dr. Ahmet Soran

Member : Assist. Prof. Dr. Samet Tonyalı

Member : Assist. Prof. Dr. Fehim Köylü

Member : Assist. Prof. Dr. Özkan Ufuk Nalbantoğlu

APPROVAL:

The acceptance of this M.Sc thesis has been approved by the decision of the Abdullah Gül University, Graduate School of Engineering & Science, Executive Board dated ….. /….. / …….. and numbered ……..…… .

…….. /…….. / ……..

Graduate School Dean
Prof. Dr. İrfan ALAN
Signature

ABSTRACT
BLOCKCHAIN BASED DATA SHARING PLATFORM FOR BIOINFORMATICS FIELD

Beyhan ADANUR
MSc in Electrical and Computer Engineering
Supervisor: Assist. Prof. Dr. Burcu BAKIR-GÜNGÖR
Co-Supervisor: Assist. Prof. Dr. Ahmet SORAN
June 2020

Recently, panomics studies attempt to identify new and actionable biomarkers by combining -omics data with other data types. In this context, there is a need to develop secure platforms that take into account ethical aspects and solve privacy and ownership issues as well as data sharing for an accurate analysis of -omics data. These days, blockchain technology has picked up significant attention in genomics, since it offers a new solution to these problems from a different perspective. In this thesis, we proposed a hybrid platform called GenShare, which is based on blockchain, homomorphic encryption and intel software guard extension (SGX) to provide efficient genomic data sharing, to perform statistical analysis and other similar processes on genomic data. While the proposed model solves security-privacy issues using homomorphic encryption and SGX, it solves other issues by using a combination of Hyperledger Fabric and Ethereum networks. In this study, Hyperledger Fabric network, which is the first phase of the GenShare model, setup is made and the performance of the network is tested with a different number of workloads. At the end of our performance evaluations, we concluded that the GenShare model has a potential to speed up the process of collecting and sharing data and it offers an efficient platform for the participants.

Keywords: Genomic Data Sharing, Hybrid Blockchain, Homomorphic Encryption, Intel Software Guard Extensions (SGX)

ÖZET

BİYOİNFORMATİK ALANI İÇİN BLOKZİNCİR TABANLI VERİ PAYLAŞIM PLATFORMU

Beyhan ADANUR
Elektrik ve Bilgisayar Mühendisliği Bölümü Yüksek Lisans
Tez Yöneticisi: Dr. Öğr. Üyesi Burcu BAKIR-GÜNGÖR
Eş Danışman: Dr. Öğr. Üyesi Ahmet SORAN
Haziran 2020

Son zamanlarda, panomik çalışmalar -omik verileri ile diğer veri türlerini birleştirerek, yeni ve uygulanabilir biyobelirteçleri belirlemeye çalışmaktadır. Bu bağlamda omik verilerinin doğru analizi için veri paylaşımının yanı sıra veri gizliliği ve sahipliği sorunlarını çözen, etik yönleri dikkate alan güvenli platformların geliştirilmesine ihtiyaç vardır. Bugünlerde blokzincir teknolojisi, farklı bir perspektiften bu sorunlara yönelik yeni bir çözüm sunduğu için genomik alanında büyük ilgi görmektedir. Bu tezde, verimli genomik veri paylaşımını sağlamak, genomik veriler üzerinde istatistiksel analiz ve benzeri işlemleri yapmak için blokzinciri, homomorfik şifreleme ve intel yazılım koruması uzantısına (SGX) dayanan, GenShare adlı hibrit bir platform önermekteyiz. Önerilen model, homomorfik şifreleme ve SGX kullanarak güvenlik gizliliği sorunlarını çözerken, diğer sorunları Hyperledger Fabric ve Ethereum ağlarının bir kombinasyonunu kullanarak çözmektedir. Bu çalışmada, GenShare modelinin ilk aşaması olan Hyperledger Fabric ağ kurulumu yapılmış ve farklı sayıda iş yükü ile ağın performansı test edilmiştir. Performans değerlendirmelerimizin sonucunda, GenShare modelinin veri toplama ve paylaşma sürecini hızlandıracağı, ve kullanıcalar için verimli bir platform olacağı sonucuna varılmıştır.

Keywords: Genomik Veri Paylaşımı, Hibrit Blokzincir, Homomorfik Şifreleme, Intel Yazılım Koruma Uzantıları (SGX)

Acknowledgements

I would like to convey my thanks gratefully to my advisors Assist. Prof. Dr. Burcu BAKIR-GÜNGÖR and Assist. Prof. Dr. Ahmet SORAN. Although I proposed crazy ideas to them, during the two years of my studies they always have cared about my ideas, supported me and trusted me. Their diligence and professional behaviors are always going to be a guide for me in the way of being a scientist.

I also would like to express my deepest gratitude to my family. I would like to thank my father Hayrettin and my mother Nurhan for their endless patience, love and labors. I would like to thank my brother Erhan and my sister Güleyhan for their guidance in my life, friendship and supports. Everything I've learned today is thanks to you. I am grateful for everything.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

After the scientific inventions over several centuries, our world has improved at a great pace. The study of DNA, which is the building block of all living creatures, continues unabated for the diagnosis and treatment of human complex diseases with the help of genetics. To illuminate our genome and to uncover the hidden mysteries, new analysis techniques and strategies are proposed every day. Bioinformatics is an interdisciplinary field that tries to solve biological problems by designing algorithms that use primary features of the problem as input and attempt to predict the beneficial outcome [1]. Genomics is a subtopic for research in bioinformatics, and it is a genetic field that relates to the sequencing and analysis of the genome of an organism [2]. As shortly, genomics outputs provide inputs to bioinformatics. Genomics related applications of bioinformatics are taking an important place for the development of life sciences. Many methods are currently being developed in genomics for the discovery and interpretation of confidential information in the genome. These developments are playing a significant role in humans life because, thanks to these analyzes, a disease can be predicted and prevented before it gives a severe hazard or personalized medicine, therapy, and nutrition can be applied for the treatment process [3].

Today, genome sequencing is mostly performed by hospitals for disease research, while people mostly want to learn their gene map, and they apply to a lab for only this

process. So, the sharing of analyzed data is not common. With the permission of the gene owner, hospitals are able to give the genomic data for further analysis, or those who already have their genomic data not tend to share this information with researchers. Possible reasons for that are the limitations of the sharing of genomic data because of the privacy and security issues [4], analysis cost [5], ownership of data, data collection steps [6], and managing huge amount of data [7]. Although there is no proposed model available today developed to solve all of the mentioned problems, privacy-enhancing technologies, like secure multi-party computation (SMC) and homomorphic encryption [8] are widely used in many applications to solve data privacy and security problems. But, to increase the number of beneficial algorithms/works proposed by bioinformatics researchers, all problems should be solved, genomic data should be shared easily, and computations on them should become widespread.

Blockchain technology has recently begun to attract attention in many areas [9], including genomics, due to the solutions it brings from a different perspective to some problems. In this thesis, a hybrid blockchain-based genomic data-sharing platform is designed which meets requirements of genomic data sharing from a different perspective and can perform count queries and statistical analysis on the data. The proposed hybrid GenShare platform includes researchers, data owners, and secure compute nodes for sharing of genomic data easily and performing computation on them, taking advantage of combinations' Ethereum [10] ve Hyperledger Fabric [11]. The GenShare platform provides genomic data privacy and security with homomorphic encryption and intel software guard extensions [12], and solves other problems with blockchain. The organization of thesis is as follows: Chapter 2 explains Background of Genomics; Chapter 3 processes Background of Blockhain; Chapter 4 shows Blockchain Applications in Bioinformatics; Chapter 5 explains Why Blockchain is/is not Suitable for Bioinformatics; Chapter 6 presents Methods used in this study; Chapter 7 explains the Overview of GenShare; Chapter 8 shows Implementation and Results; Chapter 9 presents Evaluation of GenShare; finally Chapter 10 shows Conclusions and Discussion.

Chapter 2

Background of Genomics

2.1 Overview of the Genomics

Understanding the secrets of life has always been a fundamental issue on which different disciplines have been working. In this section, bioinformatics and genomics that examine the secrets of life through biological data will be briefly reviewed. Fundamentally, cells with the same structures and functions come together and form tissues, and different tissues form organs. The organs that perform certain tasks come together to create the systems. The organism is formed as a result of the interconnection of all systems. As shown in Figure 2.1.1, every cell in organism contains hereditary material known as DNA and a long piece of DNA is called a gene. The DNA comes in packages called chromosomes. In line with recent technological advances, the number of biological data waiting to be interpreted has increased tremendously. Bioinformatics is an interdisciplinary field that obtains, stores, examines and interprets to biological data with designed algorithms [1]. Comprehensive analysis of biological systems is expressed in terms of -omics. New -omics technologies and bioinformatics tools have great importance in researching complex relationships. In recent years, a wide variety of -omics subdisciplines have been formed, and each has its own set of techniques, tools and softwares.

Genomics is a subfield of -omics. A genome can be thought of as a complete DNA sequence, which is the code for hereditary material that has passed from generation to generation. The DNA sequence contains all the genes. Therefore genomics involves the analysis and sequencing of all components of DNA. Generally, the type of data processed by bioinformatics is an output of genomics. Genomics allows that; identification of all genes in an organism, researching the interaction of genes with each other and the environment, examining the production and activation of genes. The most important feature of genomics that distinguishes it from other genetic sub branches is that it evaluates genes collectively instead of dealing with them one by one [2].

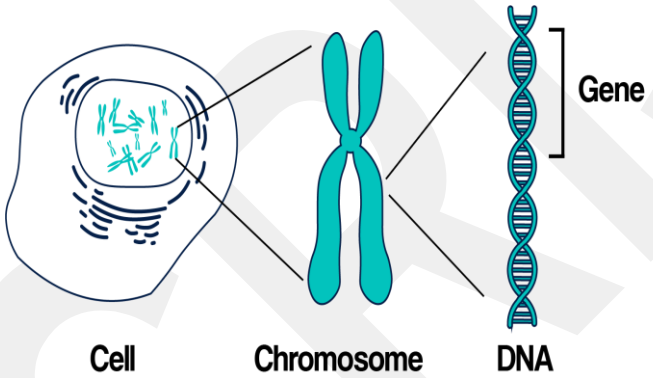

Figure 2.1.1 Structure of a cell [13]

Thanks to next generation sequencing technologies (NGS) sequencing of an organism's entire genome takes a matter of hours [14]. Next Generation Sequencing operates by splitting a strand of DNA into many pieces and decoding all the fragments massively. Many of these fragments' sequences are grouped by length. In order to reach consensus sequences for the genome, several fragments of the same size and sequence values are used. A "puzzle" resolution process patches together the sequence for the whole strand by finding overlaps between fragments [15]. By scanning the entire genome of large samples of individuals with or without a disease, variations can be found that may be associated with a disease or condition. This process is called Genome Wide Association Study (GWAS) [16]. Intensive bioinformatics studies are needed to establish the link between diseases in an organism and the responsible gene or responsible mutations. By applying bioinformatics analysis to -omics or GWAS data, scientists can perform susceptible disease tests, disease predictions, and evolutionary developments of disease.

As a result, a disease can be prevented, and its effects can be reduced before it poses a serious hazard by developing personalized medication, treatment and nutrition [3]. Personalized treatments and preventions are being developed uniquely for each individual based on genetic, environmental and lifestyle factors [17]. Especially in cancer, this method is more preferred treatment. The only thing necessary to take advantage of all these useful methods is the sharing of genomic data easily. Obtaining more efficient studies is directly proportional to the number of collected and used genomic data, but genomics has some problems that need to be solved for this aim.

2.2 Challenges in the Genomics

The cost of analyzing the first human genome was approximately $3 billion and it took 13 years to complete. Although owing to the developments in sequencing technologies cost of analysis was reduced as $1,000, high price of analysis is still complicates participation in a specific sequencing project [5-18]. Because in traditional business model personal genomics companies as shown in Figure 2.2.1, the transactions are carried out with the help of a middleman company vice versa, a direct relation between the data owner and researcher. At the same time, this model prevents the data owner can control its data access permissions and make money as a result of the data sharing [19].

Another problem in genomics is data management, which tackles the four key issues: (i) collection, (ii) integration, (iii) sharing, and (iv) storage. Scientists are spending so much time to data collection and integration. For this aim, there is no perfect solution, but four essential considerations are exist. These considerations include the use of suitable methods, attention to detail, authorization, and recording. Desired data should be checked using correct techniques, eliminating unnecessary ones then integrated. Particular attention should be paid to details; results should be accurately recorded, interpreted, and stated in order to conduct quality research. Before data collection, an individual or an organization responsible for research study must be authorized, take permissions and fulfil the requirements [6].

Figure 2.2.1 Traditional business model of personal genomics companies

The sharing of genomic data heading covers data privacy and security [4]. The privacy of data relates to anonymity [20]. On the other hand, data security is about protecting the data from unauthorized access. Due to the fact that genomic data contains private information about an individual's history, present and future, the sharing of genomic data is a very sensitive topic. The use of synthesized genomic data in crimes could be one potential misuse of genomic data. Also, the development of harmful medicines could be another potential misuse of genomic data. For all these reasons, individuals want to share their data anonymously and ensure that their personal data is kept at high-level protection. Most privacy-enhancing tools are used to fix data privacy-security issues [21-22].

Secure multi-party computing (SMC) and homomorphic encryption (HE) are the most widely used technologies for privacy enhancing [8]. Secure multi-party computation allows two or more parties to jointly perform some computation and get the result without seeing any party's input. Despite its many benefits, SMC is still not a practical method for use in a majority of applications where (near) real-time performance is required [23-24-25]. Homomorphic encryption makes it possible to perform computations on encrypted data without decryption. Three HE schemes are depending on its processing capacity, and each scheme has its own specific advantages and disadvantages [26-27-28].

Usually, governments store genomic data in databases. Public type of databases shows only summary of data or frequency information. It is estimated that the amount of

genomic data will exceed the amount of video and web data in the next decade [7], as sequencing a single human genome would yield around 200 gigabytes of data (may vary depending on the type of sequencing), and approximately 2 billion human genomes will be sequenced by 2025, according to estimations in Table 2.2.1. Investigating massive genomic data requires plenty of disk space for storage, high transfer speed for data sharing, and quick processing power because analysis takes trillions of CPU hours. In summary, although exchanging genomic data provides the unique opportunity to expand our expertise by obtaining new information from the re-analysis of the same datasets and shared datasets, it presents several challenges of ethical, legal, and technical nature [29].

	Genomics
Acquisiton	1 zetta-bases/year
Storage	2–40 EB/year
Analysis	Heterogeneous data and analysis Variant calling, ~2 trillion central processing unit (CPU) hours All-pairs genome alignments, ~10,000 trillion CPU hours
Distribution	Many small (10 MB/s) and fewer massive (10 TB/s) data movement

Table 2.2.1 Four domains of Genomic Big Data in 2025. In each of the four domains, the projected annual storage and computing needs are presented across the data lifecycle [7].

Chapter 3

Background of Blockchain

3.1 Introduction

While the world is changing and developing at a dizzying pace, the contribution of technology to this progress is undeniable. Today, technology plays an active role in every aspect of our lives. Regardless of the field of business, people's work habits are constantly changing with the development of information and communication technologies. Each change is made for a purpose and offers innovation to people. Keeping up with these innovations and incorporating the advantages it brings into business processes is also an extremely important issue. In these days, blockchain technology has picked up significant attention in diverse fields [9] including genomics [30-31-32]. It offers a new solution for problems from a different perspective so the breakdown in traditional business process models has occurred. The basic idea of blockchain is revealed in the late 1980s and early 1990s. However, the first cryptocurrency, Bitcoin, appeared in 2008 with its white paper, which is proposed by unidentified person Satoshi Nakamoto [33]. In today's Internet world, data transfer is performed in many areas (multimedia, communication, web interface, etc.). Blockchain

is the new technology that allows us to transfer also assets that we attribute value [34]. It is ledger of transactions that is based on chain model. This distributed system cannot be destroyed and is managed in a way that allows all participants to make a joint decision without a central manager. Each block contains the cryptographic hash function of the previous block, a timestamp, and process data. The system must approve processes in order to be written to the blocks. The validation mechanism includes system users, and some consensus algorithms provide it. Despite all innovations, blockchain has some technical difficulties and limitations for adapting to the future as in any technology.

3.2 Components

3.2.1 Double-Spending and Single Point of Failure Problems

Fundamental challenges of crypto techonologies are double-spending and single-point-of-failure. The original motivation of the blockchain technology is to work on preventing electronic coins from being spent twice without having a central intermediary. The double-spending problem is illustrated in the Part A of Figure 3.2.1.1. Accordingly, suppose Alice has 10 coins and then sends all 10 coins to Charlie. Charlie and other people using the coin should understand that Alice has not sent the same 10 coins to Bob before, without having a bank to verify transactions. The central intermediary is not used problem-solving because it might cause single-point-of-failure, as shown in Part B (a) of Figure 3.2.1.1. Each computing node in the blockchain network must not only store each transaction in order to enable the distributed verification of the operations but also adopt a distributed timestamp protocol, which is the actual time of a computer-recording case, in order to determine which transactions should be accepted and which should be rejected [35].

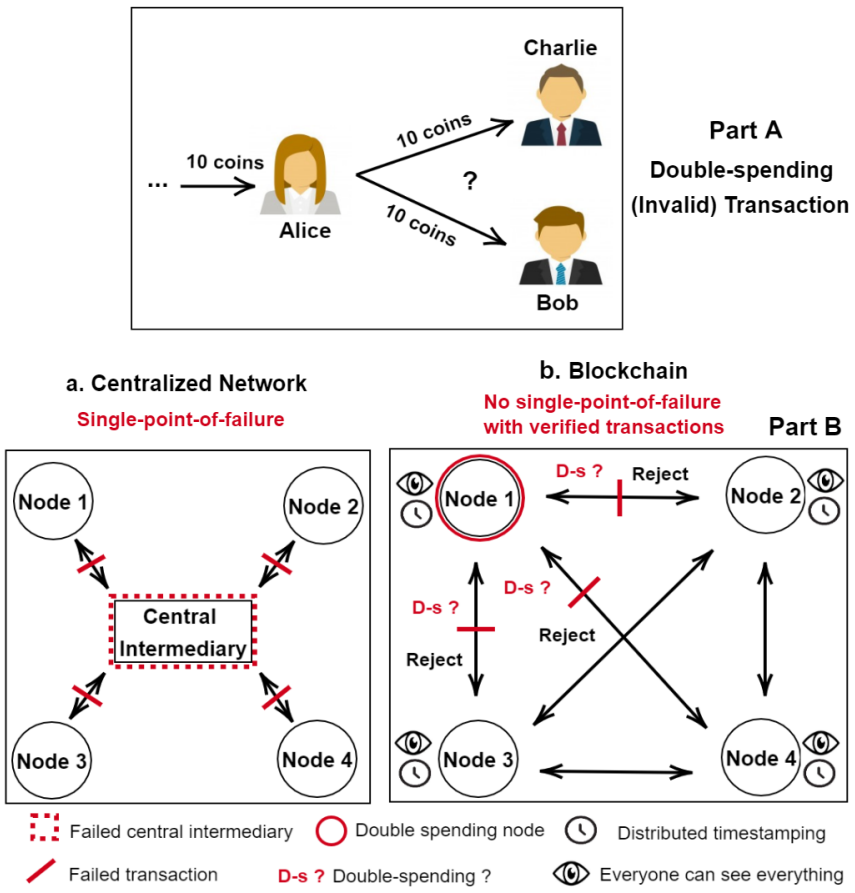

Figure 3.2.1.1 Part (A) double-spending (invalid) transaction, part (B) single-point-failure problem

3.2.2 Chain Model and Components

Blockchain may sound complex; however, it may be simplified by individually analyzing each part. It makes use of well-known processes in computer science and basic cryptography. Main components of blockchain can be listed as follows: cryptographic hash functions, digital signatures, transactions, asymmetric-key cryptography, ledgers, blocks, and how blocks are clustered. Blockchain technology consists of two basic concepts: the blocks and transactions. Any content information that occurs within the blockchain network is called transaction. This information can be values such as money transfer, fixture input, customer records according to the design. For virtual currencies, these records are money transfer information. The records are combined and processed at certain intervals and written into the blocks. Miners who

discover new blocks are rewarded with block rewards for their efforts. The winning miner receives a reward for a block by adding it to the chain as a first transaction. By connecting each newly found block with the previous block, the chain structure forming the name of the technology is obtained as in Figure 3.2.2.1, so the blockchain technology is defined as immutable ledger of transactions. Generally, cryptographic hash algorithms and digital signatures are used during the creation of a block. Instead of the original data, transactions are processed on blocks with their hash values [36].

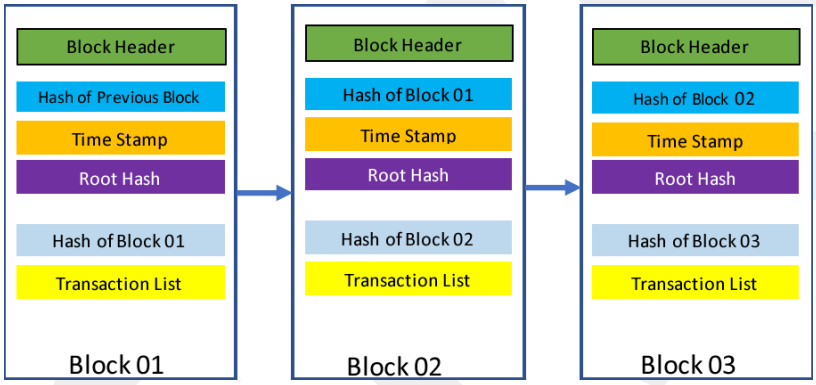

Figure 3.2.2.1 Structure of blocks [37]

The hash function is a process that creates a unique value of a fixed length with mathematical functions of various lengths of data. It is a one-way function, and when viewed, no relationship is established between the original text and the summary value. Also, the original data cannot be obtained from the summary value (Its power against the quantum computing is discussed [38]). In the summarization process, if a change is made in the original data, the summary value also changes. For this reason, hash functions are generally used for data validation and comparison mechanism. Summary functions include MD family, SHA family, RIPEMD etc. algorithms [39]. In addition to these measures, the presence of users' digital signatures in a transaction proves that the relevant transaction has been done by them. Digital signatures vary according to the content of the document. In other words, it changes uniquely according to the signed message because digital signatures consist of a combination of private keys and hashes of the message. Digital signatures are based on asymmetric cryptography. Asymmetric cryptography is an encryption system using two different keys for encryption and

decryption processes. It is used in two ways: i) Encryption with public key and decryption with private key, ii) Signing with private key and verification with public key. The public key is known to everyone on the network, but the private key is known only by the person itself. Unlike asymmetric cryptography, symmetric cryptography uses a single key, which is called secret key, for both encryption and decryption operations. While some algorithms used for asymmetric cryptology are Diffie-Hellman, RSA, ECC, ElGamal and DSA, algorithms for symmetric cryptology are AES, Blowfish, ChaCha, DES, Serpent and Twofish [40].

3.2.3 Blockchain Categorization

Except traditional databases or distributed ledger technology, there are two types of blockchains as permissionless and permissioned [41]. In permissionless blockchains, also called public blockchains, anyone can participate network as users, miners, developers without needing permission from any authority and all transactions are fully transparent so anyone can publish blocks and examine the transaction details. They are developed to obtain a completely decentralized network. Eventually, all permissionless blockchains have a token associated with them, usually designed to encourage and reward network users. Unlike permissionless blockchains, in permissioned blockchain users must be authorized by some authority for publishing blocks. In this way, which users can perform which operations in the network can be regulated. Organization that manages the chain is an important check on the participants and governance structures. Also optionally, networks may be instantiated and maintained using open source or closed source software, but they are more centralized than public blockchains. They are preferred by the companies that want to collaborate and share data but do not want to see their sensitive business data on a public blockchain. Permissioned blockchains are divided into two subcategories as private and consortuim blockchains [42]. In fact, consortium blockchains are subcategory of private blockchains because they have some differences from private blockchains. While a single entity governs private blockchains, consortium blockchains are governed by a group. This collaborative model will be the most appropriate option for businesses that work together but also compete.

3.2.4 Consensus Algorithms

Consensus algorithms are ones that enable a consensus on certain requests in distributed processes or systems in computer science. These systems or processes do not need to be reliable in these algorithms to compromise systems or processes. Therefore, consensus algorithms are used to provide the structure of the blockchain that does not require mutual trust. They play a crucial role in keeping blockchain secure and efficient. The most commonly used algorithms in the blockchain industry are given in Table 3.2.4.1 [43].

Consensus Algorithms	Explanations
PoW	When a user initiates a transaction, miners attempt to solve a cryptographic problem to test that they have worked a lot
PoS	A user encouraged to spend more on building a block until he becomes a validator
PoWeigth	Similar to PoS but the difference is that it depends on several other variables known as weights
PoB	Based on the amount, users submit the coins back into their wallet that they cannot recover from will receive rewards
PoC	Using this protocol, you can use the user's hard drive functionality
DPoS	As with PoS, but users having more coins will be able to vote and nominate witnesses
DBFT	Focuses on a gamified way of block checking among the qualified node checks
PBFT	Byzantine made use of a specific sequence to keep the rouge users at bay

Table 3.2.4.1 Different types of consensus algorithms

Choosing the right consensus algorithm related to a given problem is vital to improving the performance of the system, which might increase the number of blockchain-based applications. Proof of Work (PoW) is used in Bitcoin and Ethereum uses the hybrid version of PoW and Proof of Stake (PoS). PoW is based on an operational problem that is difficult to solve but easy to validate. In projects using PoW algorithms, miners need to solve the problem in order to add blocks on the chain. The person who solves the problem first gets the right to add the block to the chain. The most important detail in this algorithm is the processing power and the total power of miners. Proof of Stake proposes a new algorithm instead of puzzle solving. Ethereum, many projects use this infrastructure, plans to move to PoS completely. It is an algorithm in which the amount of token held by the miner is important. Miners' chances of adding blocks are directly proportional to the number of tokens they have [44].

Each algorithm has its own advantages and disadvantages, depending on the purpose and requirements of the systems but common goals of blockchain consensus models can be listed as follows:

- To reach an agreement
- To cooperate with the participants
- To offer equal rights to each participant
- To ensure that every member of the group is equally active

3.2.5 Key Benefits and Open Issues

Before mentioning the advantages and disadvantages of blockchain technology, it is worth to note that these returns may be more pronounced or less pronounced according to systems' implementation and usage strategies. In general, the advantages and disadvantages of blockchain can be listed as follows.

Key benefits of the blockchain [45];

- Blockchain allows distributed management of the information transfer process without central management. This information management process is recorded in an indestructible way and viewed transparently by everyone.

- Each transaction has to be verified according to the mechanism used by the nodes included in the system, in this case the transactions become more consistent and secure.
- Individuals who are involved in the processing of transaction results on blocks and share the computing power with the system, make money.
- The nodes included in the system perform their transactions anonymously and control each information and process by themselves.
- Thanks to digital signatures and verifications, it is ensured that its stakeholders trust each other easily.
- Certain activities can be automated thanks to smart contracts.

Open issues of the blockchain [46];

- Proof of Work algorithm-based blockchain systems consume a lot of energy.
- The data in the blockchain is kept separately in each node and the consistency of these data is ensured as a result of each completed operation. For this reason, compared to traditional databases, there is a low performance.
- Individuals' privacy can be harmed due to saving data and accessing data content transparently by each node in the network.
- With the increase the number of applications that use blockchain network(s), the extra workload that system needs also have increased. As a result, scalability and performance problems arise. In a large distributed system, as the needs increase, the algorithms running on it will try to perform thousands of operations per second. Thus, system performance might decrease.

As a result, after understanding whether a system needs blockchain technology, it would be a better decision to integrate the blockchain technology into the system [47]. Please note that, blockchain is not always applicable for all kind of distributed problems.

Chapter 4

Blockchain Applications in Bioinformatics

Blockchain technology is integrated into multiple industry areas for facilitating and improving the functions of systems. Considering the health sector, every development in the field actually means goodness to humanity because of all the innovations in this field directly concern the individual. So blockchain technology has picked up significant attention in genomics and healthcare, since it offers a new solution for their problems from a different perspective. Possible use cases for blockchain in genomics and healthcare can be classified as follow [48]:

- **Patient monitoring through the Internet of Things;** Today, one of the vital points for the researches is to collect patient data that may be beneficial for health. Still, it is kept in the cloud system, which is less controlled about the security and access of this information. In contrast, the safer and compatible blockchain system can better protect and record information from objects in the network [49].
- **Detection and Prevention of Fraud in Medicine;** As an activity of Hyperledger, Counterfeit Medicines Project was recently launched [50]. The

project aims to detect illegal, low quality or stolen drugs by marking a date stamp on each drug produced with the blockchain approach.

- **Payment of Insurance Premiums;** It is an example of blockchain applications for making service pricing and payments in health system compatible with each other [51]. Inefficient remuneration system causes both time loss and cost. It is claimed that a blockchain system supported by smart contracts can be much more efficient than current systems.
- **Data Storage and Distribution;** Due to the high costs associated with cloud platforms, distributed data storage enabled by blockchain will also arouse interest. Filecoin can be showed as an example of decentralized network storage. There are also many plans to provide a free private data exchange by giving full power to individuals.
- **Distributed Computation**; For distributed computation, there are already ongoing projects that utilize a blockchain for rewarding as Gridcoin, Curecoin, and FoldingCoin.
- **Identity and Ownership;** Personal identity and data ownership can be checked over blockchain. In this way, individuals can organize access controls of their data while hiding identities.
- **Voting;** Blockchain also offers a way to secure online voting. Genomics works relies heavily on standardization, which is determined by an electoral process. Furthermore, attempts at crowdsourcing, such as treating variations, can be enforced via blockchain, which can also integrate multiuser consensus on curation outcomes.
- **Decentralized Autonomous Organizations (DAOs);** DAOs, such as The Cancer Genome Atlas (TCGA), can be used for operating mediums to predefine laws, regulations, and governance with the powerful sides of smart contracts.
- **Medical Records Management System;** Decentralized medical data recording systems are among the most recommended blockchain applications. Management of medical record systems with blockchain provides to patients transparent, quick access, and authority-corrected errors for their records. On the

other hand, it helps doctors predict the disease before the disease harm the patient.

In this thesis a new blockchain based genomic data sharing platform for bioinformatics researchers has been proposed. For this reason, while reviewing the literature, the existing blockchain based electronic health record (EHR) and genomic data sharing platforms were also examined. These examined projects are Nebula Genomics, Zenome, Genecoin, Gene-Chain, DNATIX, Medrec, IRYO, Coral Health, Open Longevity, Patientory, Medicalchain, GemOS, e-Estonia, Health Nexus, and NeuRoN as in Table 4.1. Among the fifteen projects; Nebula Genomics, Zenome, Genecoin, DNATIX, Medrec, Coral Health, Open Longevity, Patientory, Health Nexus and NeuRoN are based on Ethereum. Gene-Chain is grounded in Hyperledger. IRYO is based on EOS, e-Estonia is based on KSI, Medicalchain, and GemOS is based on a combination of Ethereum and Hyperledger. All of them prefer smart contracts. Also, Nebula Genomics, Zenome, Genecoin, Gene-Chain and DNATIX, have been developed for genomic data sharing, but Medrec, Coral Health, IRYO, Open Longevity, Patientory, Medicalchain, GemOS, e-Estonia and Health Nexus have been developed for EHR sharing, and only the NeuRoN supports both genomic data sharing and EHR sharing. According to these information, Ethereum or Hyperledger technologies are preferred in healthcare and genomics. Generally, Ethereum are used by genomics related projects. In contrast to genomics related projects, Hyperledger are used by healthcare related projects and numbers of blockchain-based EHR sharing platforms are more than blockchain-based genomic data sharing platforms.

Table 4.2 sets out the common advantages and weaknesses of current projects. The benefits of using blockchain in these systems can be itemized as follow: (i) data owner manages data access permissions, (ii) analytical cost can be minimized, (iii) data owner-buyer interactions accelerate and become transparent, (iv) data collection process accelerates and privacy issues are partially solved. On the other hand, (i) it is not possible to provide complete anonymity, (ii) there is no protective mechanism for the attack scenario, (iii) key-related issues are present, (iv) energy consumption and

scalability issues of blockchain technology, and (v) there is no comprehensive documentation.

In Figure 4.1 all projects are classified with the 6 metrics. Nebula Genomics, Genecoin and Gene-Chain support register kit. It means they have their own facilities to sequencing. Data owners can make money at Nebula Genomics, NeuRoN, Open Longevity, Zenome, and IRYO. Zenome and Health Nexus support data concerning human and non-human organisms. Other projects exclusively support only human data. Only Nebula Genomics uses partially homomorphic encryption to share data using encrypted format. So in Nebula Genomics, data privacy is better than others. IRYO, Zenome, Open Longevity and NeuRoN are included in the metric of disease prediction. Because they use some artificial intelligence methods on data and obtain prediction to related diseases. Finally, while Medrec, IRYO, Coral Health, Open Longevity, Patientory, Genecoin, Medicalchain, GemOS e-Estonia and NeuRoN have mobile applications; Medrec, IRYO, Coral Health, Patientory, Medicalchain, e-Estonia, Health Nexus and NeuRoN have patient monitoring system.

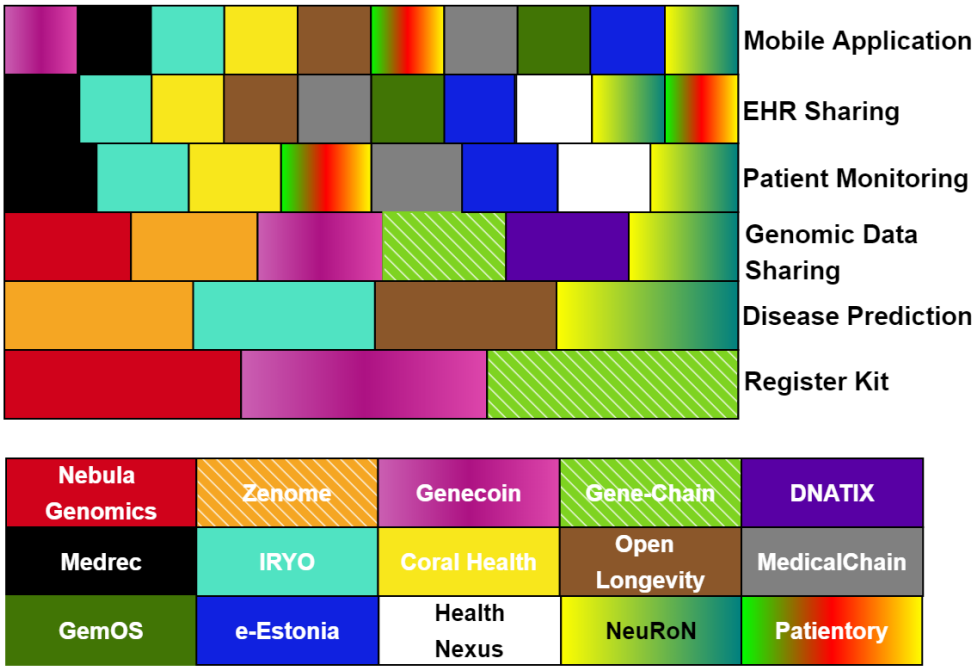

Figure 4.1 Classification of the projects

The General Information about the Projects	Platform	Country	Company	The Focused Area
Nebula Genomics [52]	Ethereum	USA	Nebula Genomics 2016	Genomic and Phenotyping Data Sharing
Zenome (ZNA) [53]	Ethereum	Russia	Zenome 2017	Genomic Data Sharing
Genecoin [54]	Ethereum	Brasil	Genecoin 2017	Genomic Data Sharing
Gene-Chain (DNA) [55]	Hyperledger	USA	EncrypGen 2016	Genomic Data Sharing
DNATIX (DNAtix) [56]	Ethereum	Israel	DNAtix 2014	Genomic Data Sharing
Medrec [57]	Ethereum	USA	MIT Media Lab 2016	EHR Sharing
IRYO (IRYO) [58]	EOS	Slovenia	IRYO 2017	EHR Sharing
Coral Health [59]	Ethereum	USA	Coral Health 2017	EHR and Genetic Test Results Sharing for Personalized Medicine
Open Longevity (YEAR) [60]	Ethereum	Russia	Open Longevity 2016	Biomedical Data Sharing for Development Antiaging Clinical Trials
Patientory (PTOY) [61]	Ethereum	USA	Patientory 2015	EHR Sharing
MedicalChain (MTN) [62]	Hyperledger Ethereum	UK	Medicalchain 2017	EHR Sharing
GemOS [63]	Hyperledger Ethereum	USA	GemOS 2016	EHR Sharing for Personalized Medicine
e-Estonia [64]	KSI	Estonia	Guardtime 2009	EHR Sharing and Electonic Prescription
Health Nexus (HLTH) [65]	Ethereum	USA	SimplyVital Health 2017	EHR Sharing
NeuRoN (NRN) [66]	Ethereum	USA	doc.ai 2016	Genomic Data and EHR Sharing

Table 4.1 General information about the projects

Advantages	• Data owners control data access permissions • Easily and directly communication • Metadata are stored on a block instead of original data • Quick data transmission • Data standardization • The immutable and distributed ledger of transactions • No intermediary companies • Reducing analysis costs • Verification mechanism • Providing interoperability • Pseudonymity
Disadvantages	• No fully homomorphic encryption, so both addition and multiplication operations cannot be performed on the encrypted data. • There is no utterly preventive system towards attacks • No fully anonymity; only pseudo-anonymity • No exact scalability solutions • Key challenges • Energy consumption

Table 4.2 Common features of the projects

Chapter 5

Why/Why not Blockchain is Suitable for Bioinformatics?

The main challenges in genomics are categorized as data sharing, data collection, ownership and analysis cost. Figure 5.1 shows the problems in the genomics field and blockchain based solutions of these problems. Blockchain technology can easily reduce prices of health-based applications and genomic analysis. Because contrary to the functioning of existing systems, data owners can contact data buyers directly without an intermediary company in its network. In this way, the analysis prices decrease, and the data owner makes money. Although this feature seems to be an advantage, it may be dangerous in some cases that contain data that should remain private. The blockchain's verification mechanism is safer compared to existing systems. Because thanks to consensus algorithms, transactions are verified fairly by all system users. At this stage, users may face problems related to energy consumption, performance and scalability. In blockchain technology, transactions are examined by each node in parallel, so the consistency becomes very crucial for the reliability of the network. Therefore, the workload of the nodes increases, and a lot of energy is spent for consensus. Also, as the number of nodes joining the distributed system increases, the system needs increase.

The algorithms that run on it will attempt to do thousands of operations per second. As a consequence, system efficiency is declining [67-68].

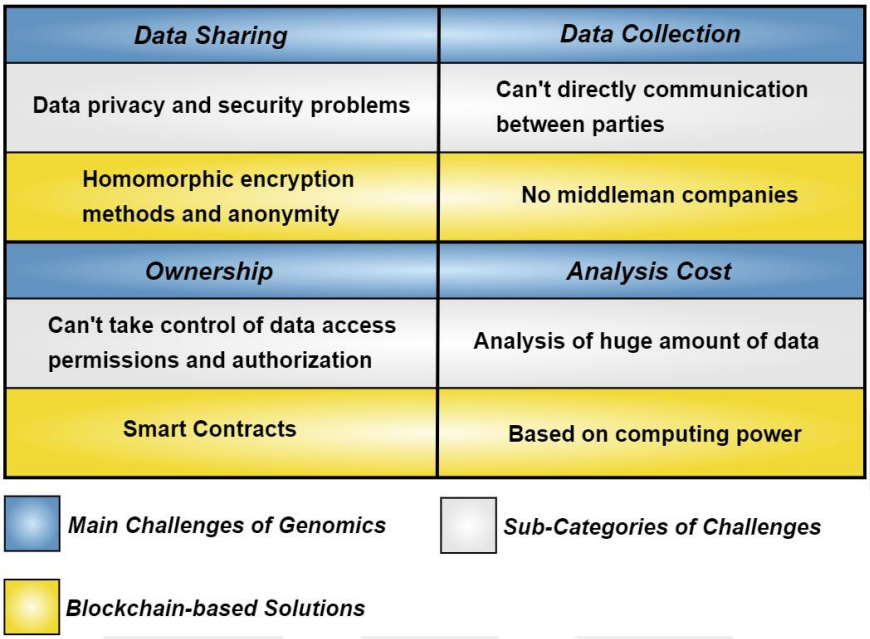

Figure 5.1 Genomics challenges and their blockchain based solutions

The other important problems on genomics are personal data security and privacy. Although there are many privacy preserving techniques, in fact, no fully anonymity can be achieved in any health-based or genomic projects until people develop portable devices that can sequence the genomes without resorting to the laboratory. Only pseudo-anonymity provides by methods including blockchain technology. People generally do not want to share their private data directly. Using blockchain, individuals can share only metadata, which includes general informations about data, with their hash value instead of original data. Likewise, data owners can share their data with encrypted format using any encryption technique such as homomorphic encryption, thereby system security increases [69-70-71]. With the help of smart contracts, unauthorized data access can be prevented. People can easily edit their data access permissions with smart contracts and perform their transactions automatically. As a result, data management is provided entirely by themselves. Processing of large amounts of genomic data requires fast computing power, as it takes trillions of hours of CPU. It can be said that blockchain technology is one of the best technologies that is

suitable for the solution of this problem. Because it's working principle is based on computing power. Based on these innovative solutions [72], in this thesis, the new genomic data sharing and computing platform is designed that is based on hybrid blockchain structure consisting of Hyperledger Fabric and Ethereum, homomorphic encryption, and SGX.

Chapter 6

Methods

In this section, the fundamental technologies used in the platform and the general architecture of the system are explained.

6.1 Ethereum and Hyperledger Fabric

Choosing the most suitable blockchain platform for the design and construction of a blockchain-based project is an important step. There are two types of blockchain as public and private, explained in Chapter 3. The most popular examples of public and private blockchain are Ethereum [10] and Hyperledger Fabric [11]. These two technologies have been developed for different aims [73]. Since the Ethereum is mostly used as a public blockchain in applications, it aimed to be completely transparent and permissionless. So, everyone in the Etherum network can access the transaction ledger, get into the system without permission, and do any operation without restrictions. On the other hand, Hyperledger Fabric is a permissioned blockchain technology that develops to meet the needs of applications where privacy and security are required. A closed network can be easily set up by editing access permissions on the network. It is possible to create multiple channels in the system and only the specified users to use these channels. Thus, unregistered users cannot access the ledger, and private

information can be shared without any notification by the entire network. In addition, Hyperledger Fabric has different types of nodes in the consensus mechanism and these nodes are predefined in the network configuration.

Smart contract consists of a set of rules which is running on the blockchain. It automatically fulfils certain obligations and tasks when appropriate conditions occurs as the software representative of users [74-75]. Purpose of smart contracts are that maintaining of data, managing of contracts and relationships, providing of functions to other contracts and complex authentication. Almost all blockchain-based healthcare and genomics related projects have used smart contracts [76]. Genshare also choose to use smart contracts in proposed project because of its many advantages, such as the automatic fulfillment of obligations and the regulation of data access permissions and relationships feasibly. Ethereum and Hyperledger are suitable environments for smart contracts. As differences, Hyperledger uses also chain-codes instead of Ethereum-type smart contracts. From an application developer's point of view, a smart contract with the ledger forms the heart of the Hyperledger Fabric blockchain system. Whereas a smart contract defines the executable logic that generates new facts that are added to the ledger, a chain-code is typically used by administrators to group related smart contracts for deployment, but can also be used for low level system programming of Fabric.

In the GenShare network, the combined version of Ethereum and Hyperledger networks is proposed. While data owners will share the indices of their data as privately, researchers will be public community that wants to perform computation on shared data. To execute the requirements of data owners and researcher, respectively Hyperledger Fabric and the Ethereum have been selected by the GenShare [77]. Another reason for choosing these two blockchain technologies is that Hyperledger Fabric supports Ethereum smart contracts using the EVM chain-code plugin. Hence, Hyperledger Fabric and Ethereum can interact with each other. It is an important factor because every blockchain project targets a specific area, and applying a combination of different technologies according to the desired characteristics of the project may be necessary. Combining of different blockchain technologies and providing interaction of

them are a current issue which is examined under the blockchain interoperability or cross-chain interaction between all public, private, and consortium blockchains headings [78].

6.2 Partially Homomorphic Encryption

Homomorphic encryption is an encryption method that allows calculation on ciphertexts and generates an encrypted result that matches the result of operations, such as done on plaintext when decrypted. With this technology, data leakage can be effectively prevented, since it works with encrypted text without deciphering it. There are currently three types of homomorphic encryption schemes with respect to the number of allowed operations on the ciphertexts as Partially Homomorphic Encryption (PHE), Somewhat Homomorphic Encryption (SWHE) and Fully Homomorphic Encryption [79]. PHE permits only one type of operation with an unlimited number of times, either addition or multiplication. SWHE permits some types of operations with a limited number of times. FHE permits an unlimited number of operations with unlimited number of times. Nowadays, numerous genomic computations are done with homomorphic encryption [80]. With the help of PHE scheme, statistical analysis over frequency count of genetic data can be performed [81]. SWHE should be preferred for operations that are more complex than statistical analysis, such as pattern matching and searching [82]. Creating an FHE encryption scheme is conceptually straightforward but in computing terms its implementation is too expensive. As the number of operations that can be performed in a scheme increases, the size of the ciphertext expands for each homomorphic encryption operation [83]. Hence, enforcing the fully homomorphic encryption scheme is not favored today. GenShare is preferred to be designed with the Paillier's encryption scheme, kind of PHE, and to perform safe computations on genomic data using an additive homomorphic property of it. The following Figure 6.2.1 illustrates the properties of Paillier’s scheme. Paillier’s encryption scheme is a probabilistic asymmetric algorithm to public key cryptography which is developed by Pascal Paillier in 1999. Due to the difficulty of calculating the n-th root classes, the Paillier encryption

system is based on the decisional composite residuosity assumption. The system shows homomorphic feature according to the addition process; this means that, given only the public key and the encryption of M_1 and M_2, one can compute the encryption of M_1+M_2.

Key Generation	Encryption of m	Decryption of c
1. n=pq, the RSA modulus 2. λ=lcm(p-1, q-1) 3. g ϵ Z /n^2Z s. t. n|or d_n^2(g) 4. Public-key : (n,g), secret key: λ, μ	1. m ϵ {0,1…n-1}, a message 2. h ϵ_R Z /nZ 3. c= $g^m h^n$ mod n^2, a ciphertext	1. m = L (c^{λ} mod n^2) L (g^{λ} mod n^2)$^{-1}$ mod n 2. The constant parameter, L (g^{λ} mod n^2)$^{-1}$ mod n or L (g^{α} mod n^2) $^{-1}$ mod n where g = 1+n mod n^2 can also be recomputed once for all.
Example: Suppose there are two ciphers. If CipherText1 = $g^{m1}x_1^n$ mod n^2 and CipherText2 = $g^{m2}x_2^n$ mod n^2, → CipherText1 · CipherText2 = $g^{m1}x_1^n \cdot g^{m2}x_2^n$ mod n^2 and additive property is: g^{m1+m2} $(x_1x_2)^n$ mod n^2		

Figure 6.2.1 Pailler's cryptosystem

6.3 Intel Software Guard Extensions

The Software Guard Extensions (SGX) is an intel processor architecture security extension that enables private memory regions, called enclaves. By using SGX, private codes and data in enclaves are separated from privileged modules and secured against them. Access control supported by the hardware limits accesses to the enclaves. So SGX can perform protected data computations by untrusted parties on private data [12]. As an alternative to conducting computations while preserving anonymity, SGX, and homomorphic encryption can be combined [84]. Homomorphic encryption provides the data privacy while the SGX achieves the data protection. Figure 6.3.1 illustrates this architecture.

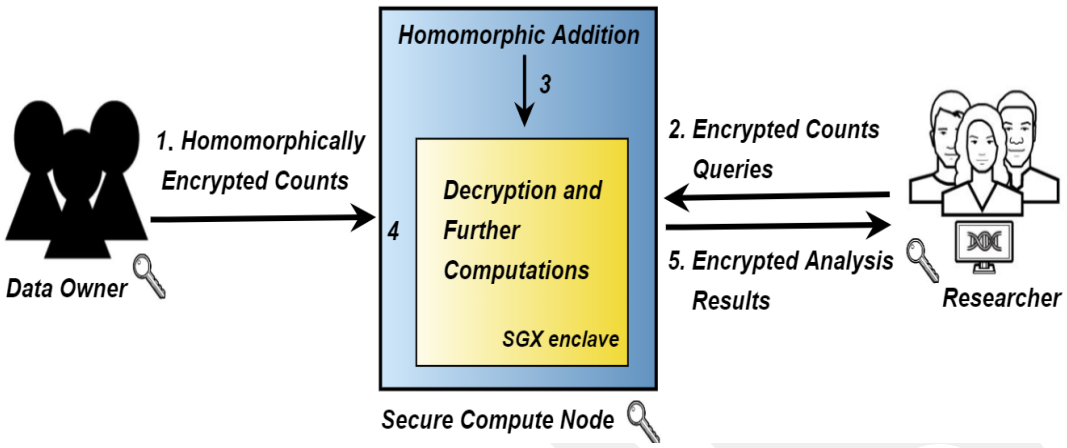

Figure 6.3.1 The combination of SGX and additively homomorphic encryption

Using a combination of the Paillier's scheme and SGX technology, the following computations can be performed on GWAS and NGS data [69-70-71]:

(i) Disease Susceptibility,

(ii) Generation of Contingency Table (counts of observed genomic variants as computing allele frequency and calculating chi-square statistics) and Statistical Analysis of Genomic as Linkage Disequilibrium,

(iii) Hardy-Weinberg Equilibrium,

(iv) Cochran-Armitage Test for Trend (CATT),

(v) Fisher's Exact Test (FET)

(vi) Transmission Disequilibrium Test (TDT).

To reach the privacy aims of GenShare, a combination of Pailler's homomorphic encryption scheme and intel SGX was preferred to perform computations on GWAS an NGS data.

Chapter 7

Overview of GenShare Model

GenShare is a new genomic data sharing and computation platform that is based on blockchain, homomorphic encryption and SGX technologies. When the GenShare model is being developed, the sytem structure is divided into two categories as inclusion of nodes in the GenShare and relations, and obtaining the genomic data. The reason for categorizing the system structure was to evaluate all the solutions that can be applied for each stage and choose the most suitable methods among them for the system requirements of GenShare.

7.1 Inclusion of Nodes in the GenShare and Relations

Primaly, as seen in Figure 7.1.1, the GenShare consist of three different type of nodes as data owner, researcher and secure compute nodes.

- **Data Owner Nodes:** Individuals who want to share their genomic data by organizing the privacy and access permissions of uploaded data. Data owners make money as a result of sharing their data. They utilize the GenShare App and involve in Hyperledger Fabric network to privately share index of their data.

- **Researcher Nodes:** Usually, these nodes are a part of the university or R&D centers. Researcher nodes purchase genomic data from data owners through secure compute nodes and analyze this data in secure compute nodes without seeing the content of data. Researcher nodes involve in the GenShare chain, which is an Ethereum-derived blockchain.

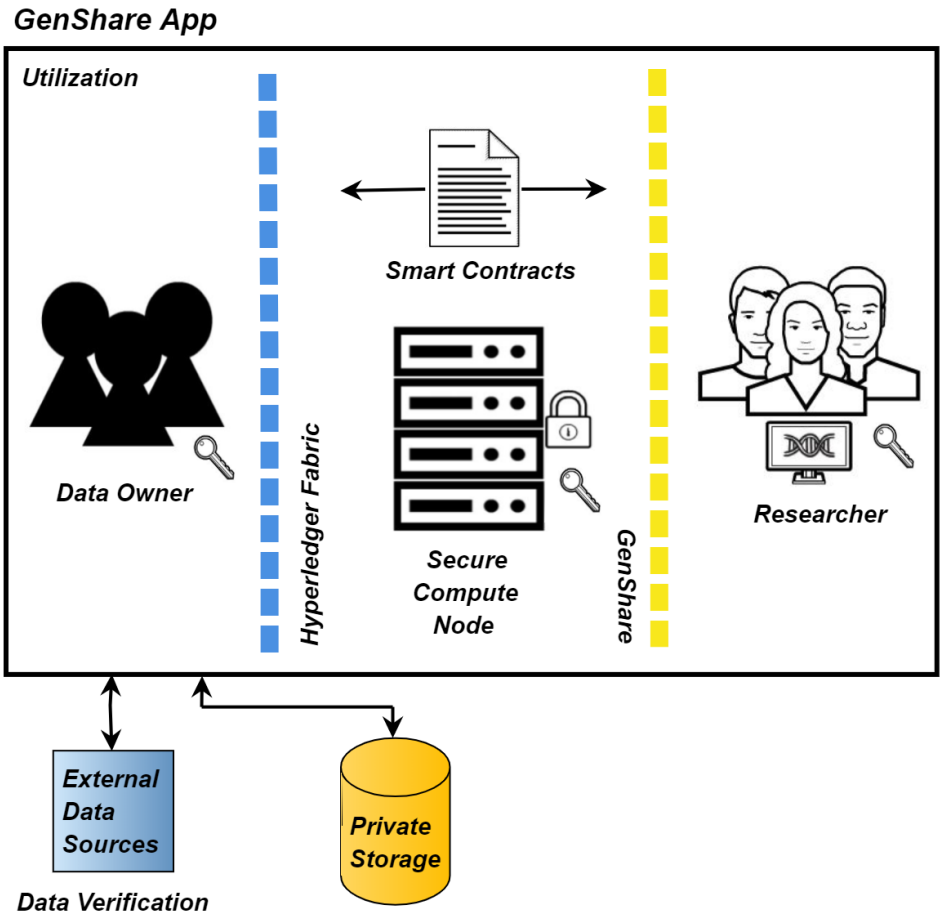

Figure 7.1.1 Overview of GenShare model

- **Secure Compute Nodes:** They are potent servers that support SGX technology and perform computations on homomorphic encrypted data. They are the touchstone of the system and managed by certain universities. They provide communication between the data owners and the researchers by involving in both Hyperledger Fabric and GenShare network. They are responsible from that verifying the data and making the calculations requested by the researcher and

sending the results to the researcher. Secure compute nodes also make money via their computations.

Two different blockchain technologies are used in the GenShare system, as Hyperledger Fabric and Ethereum. Data owners are included in the Hyperledger Fabric network, while researchers are included in the Ethereum-based GenShare chain. Secure compute nodes provide communication between the researcher and the data owner and are included in both chains at the same time. Data types and computations prices in the system can be determined as fixed by the system or as dynamically by the users. In dynamic regulation, users are given the right to choose. However, in order to ensure fair distribution in GenShare system, fixed prices are determined according to data and calculation types.

In the GenShare, data owners store their data wherever they wish, in a homomorphic encrypted form. When they participate in the network, they first specify the properties of their data. In the data sharing phase, they share their data indices instead of data with secure compute nodes. Secure compute nodes obtain the homomorphic encrypted data through sending indexes. After this step, the disease stated by the data owner in the characteristics of the data must be approved by the hospital, and the secure compute nodes must verify the data through hash records in the laboratory. When the data is verified, the fee is transferred to the data owner and demanded computations are made automatically. If the data is not verified, the data owner will be blacklisted and punished for the further calculations to keep the system maintenance safe.

There are two situations to consider the involvement of researchers in the data-sharing platform. Firstly, researchers may want to perform computations on data publicly or privately; secondly, these calculations can be an individual process or group participation. To ensure operation privacy, blockchain systems can offer an off-chain working model to include researchers in the data-sharing platform [85]. Also, using threshold cryptography [86] can push researchers to work altogether by providing data

if and only if at least three researchers request to process it. Thus unnecessary computation requests are prevented, and the system becomes more reliable. After researchers included in the system, on the Ethereum-derived chain, they create a smart contract that includes the solicited data features and desired computation type. The secure compute node reads a contract of researchers and examines the Hyperledger Fabric chain to find suitable data owners. When secure compute node finds suitable data owners, related fee is automatically collected from the researchers. Also, a selection mechanism can be acreated according to the system requirements when selecting a node from multiple suitable data owners with the desired property data. As soon as the data owner's data is verified by the secure compute node, specified calculations are performed, and result is sent to researcher.

In the system, participant universities provide secure compute nodes. These machines must support SGX technology to contribute to the GenShare platform because the shared data is in encrypted form. They are responsible for verifying the data and making the calculations requested by the researcher and sending the results to the researcher. Secure compute nodes gather the requests and find the relevant data according to the applications. After finding relevant data, SCNs receive the corresponding payment from researchers. If a problem occurs in the assigned SCN before the result reaches the researcher, a new computation node can be assigned to the transaction, or the turnaround mechanism refunds the researcher's money. As with any system, there may be trust problems in the GenShare. To prevent this, nodes that in charge to validate the computation results can be included in the system.

7.2 Obtaining Genomic Data

Today, genome sequencing is carried out by the relevant laboratories. In disease research, genome sequencing is mostly performed by hospitals, while sometimes people want to learn their gene map. To obtain the relevant results, firstly individuals' samples should be forwarded to the laboratories. This process is done either by posting the samples to the laboratories or individuals directly go to the laboratories. At this point

anonymity of persons is not entirely assured because the laboratories know the identity details of every person. Briefly, until people develop portable devices that they can sequence their genomes without resorting to laboratory, to speak of full anonymity is not possible. The laboratories conduct research on samples according to the method of sequencing in the final stage of the receiving genomic data and relay the findings to individuals. Laboratories claim that after the findings are passed on to people, they erase the details, but it should be noted that there may be malicious users in every system and organization.

There are three separate concerns to be discussed at the point of sharing the individuals' genomic data; disease confirmation, data verification and data security. Hospitals are mainly using genomic data sequencing for disease research. The data owner presents the data features to the researcher while sharing its data, but it needs to get approval from the hospital to prove that the presence of disease in the data features. Additionally, a data authentication process that determines whether the individual shares his or her original data is needed. If the interaction between the laboratory and the data owner is conducted on blockchain as seen in Figure 7.2.1, part 1, the problem of trust against the laboratory is eliminated, the laboratory from which data is obtained appears transparently, and the selling of data to others by the laboratory is prevented.

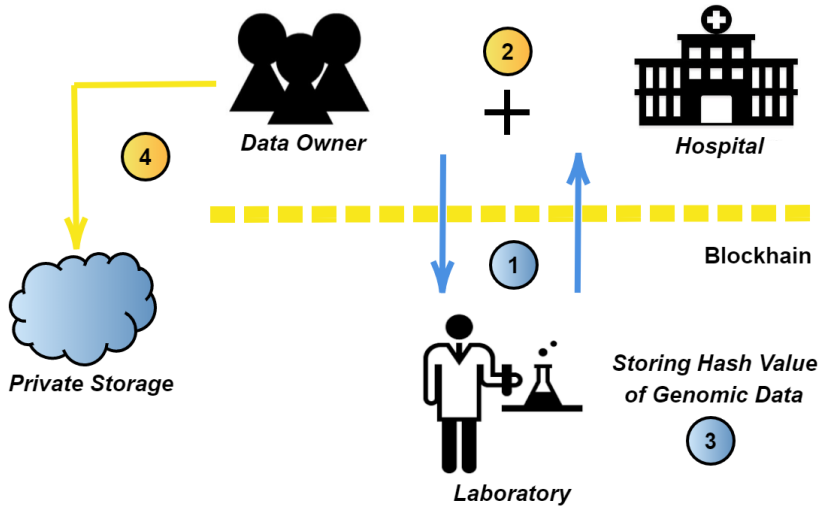

Figure 7.2.1 The process of obtaining genomic data

At the same time, the disease confirmation mechanism can be provided directly with being included in blockchain of hospitals.

In Figure 7.2.1, part 2, the data owner and hospitals can work as a single node by generating partially common keys so that both the data owner and the hospital get approval from each other for the sharing of the data. In the approval mechanism, at least three physicians from a facility can be required for acceptance of illness to deter security breaches. As data verification mechanism, the GenShare recommend keeping the hash values of the data in labs, as shown in Figure 7.2.1, part 3 so, when necessary, secure compute nodes can easily understand whether the data is original by comparing hash values. Even if the genomic data owner's identity is unclear, information about its identity and ancestors can be obtained through its data. Therefore, providing the security of genomic data is also very important. As the most effective methods for this are that data owner can save its data anywhere in an encrypted form with homomorphic encryption and share only index of data as shown in Figure 7.2.1, part 4. In this way, the desired computations are performed on the data without knowing the content of the data and nobody except the data owners can access the content of the data.

Chapter 8

Implementation and Results

Implementation of the GenShare system is divided into three parts as following:

- Hyperledger Fabric network setup and providing communication between data owners and secure compute nodes.
- Secure computation nodes setup and providing secure computations on shared data.
- Ethereum based GenShare network setup and providing communication between secure compute nodes and researchers.

For implementation and results section, the Hyperledger Composer Business Network [87] was set up, tested and the performance of the network was showed with different number of nodes and transactions, since to make the entire system operational will take a long time and providing communication between the data owners and the secure compute nodes is more important for the initial phase.

8.1 Hyperledger Composer Business Network Setup

Hyperledger Composer is a development framework that simplifies the creation of Hyperledger Fabric blockchain applications. Aim of it is helping users to create blockchain applications on Hyperledger Fabric without needing to know the low-level details involved in blockchain networks. As shown in Figure 8.1.1, it permits to define the data model, business logic and access control lists for an application which can be deployed and executed within of a Fabric channel [88]. Hyperledger Composer supports creating web, mobile or native Node.js applications. Using Composer, users' applications do not need to run a local node and if required, they can communicate with a remote node through an RPC or HTTP REST. Besides these, it comes with a web playground which is named Hyperledger Composer Playground [89] for the configuration, deployment and testing of a business network in browser without a local network being set up.

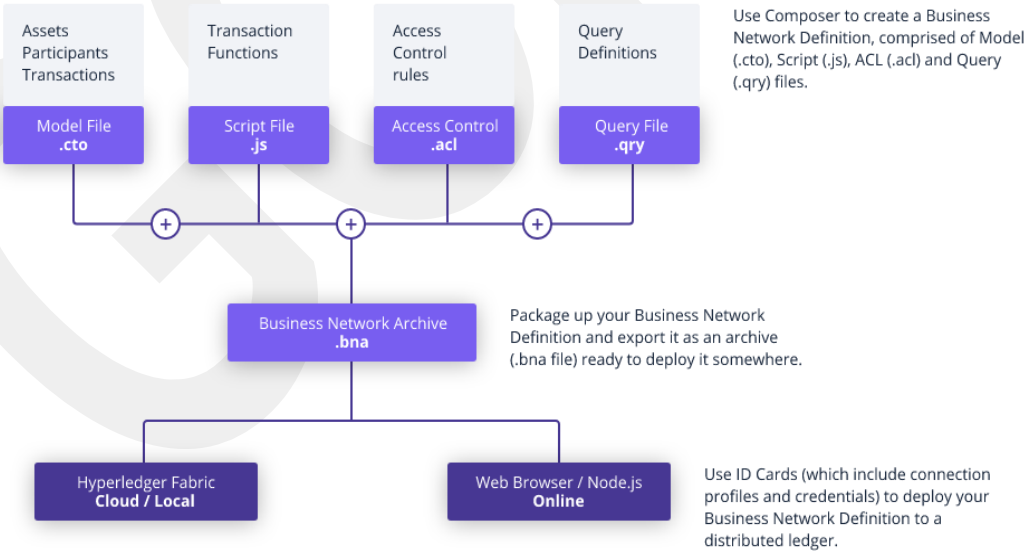

Figure 8.1.1 Hyperledger Composer architecture

Composer has its own object-modeling language. Four types of resources can be defined:

- **Assets:** In the application, items are being monitored.
- **Participants:** Entities that interact with the network. Each of them has its own permits.
- **Transactions:** are sent to update an asset or a participant as well as to execute a custom-defined logic.
- **Events:** can be emitted from the transaction logic and subscribed to by participants.

With take advantage of the mentioned benefits, a Hyperledger Composer Business Network which is named AguWork was created in this thesis. For the configuration, deployment and testing of AguWork Hyperledger Composer Playground was used. AguWork was set up on two different physical nodes, which was communicated on global network. Node 1 run on 4 vCPU, 8 GB memory, 50 GB storage - enabled server. Node 2 run on 4 vCPU, 8 GB memory, 50 GB storage - enabled server. As an OS, Ubuntu 18.04 version was preferred because of it is compatible with Hyperledger Fabric 1.4 version. The reason for the global network preference in communication of physical nodes is that while examining AguWork performance, to achieve realistic results were desired. Figure 8.1.2 shows the playground of the AguWork. It consists of three different files as model, script and access control. The model file contains assests, participants, transactions and events definitions. The script file contains transaction logics as functions, and finally, the access control file contains defined permission on assets, participants, and transactions.

In the AguWork:

- **Participants:** are DataOwner (data owner) and SCN (secure compute node) as in the Figure 8.1.3
- **Assets:** are GenomicData (genomic data) and URL (data index) as in the Figure 8.1.3
- **Transactions:** are SCN Creation, Owner Creation, Assests Creation and Data Request as in the Figure 8.1.4

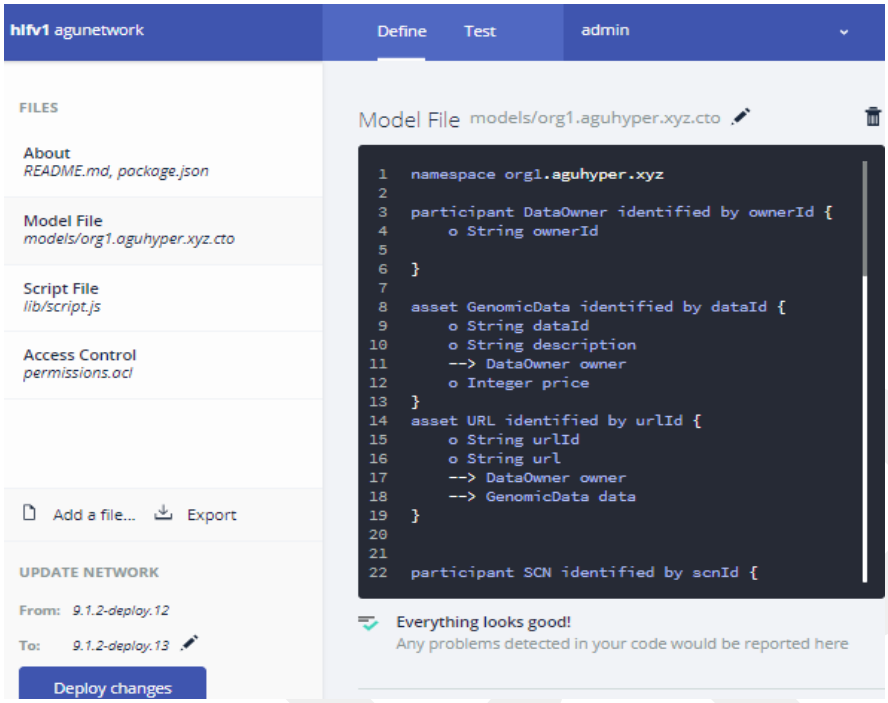

Figure 8.1.2 Hyperledger Composer Playground of AguWork

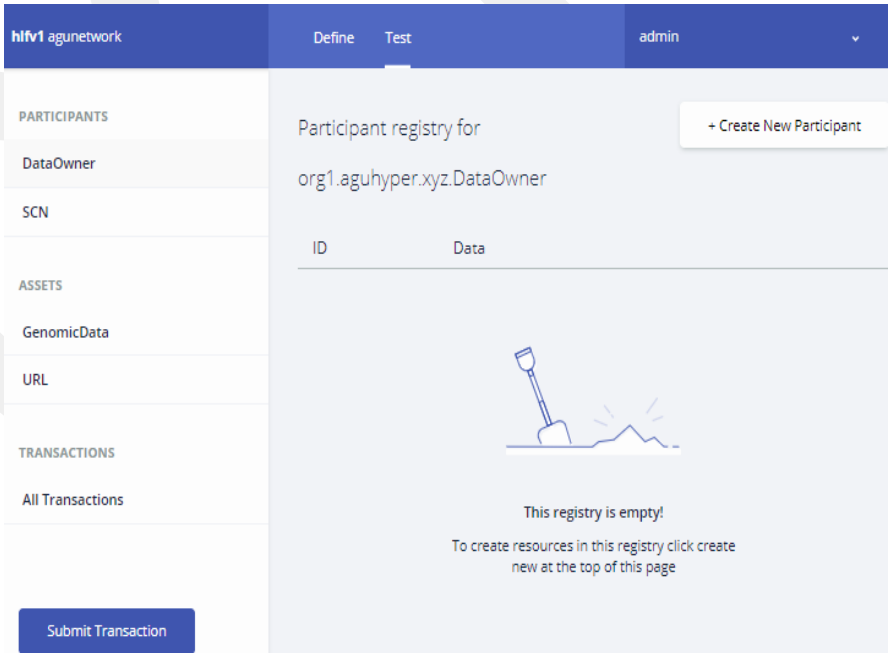

Figure 8.1.3 Participants of AguWork

Data owners join the AguWork with their ownerId as the following.

{ "$class": "org1.aguhyper.xyz.OwnerCreation",
"ownerId": "100" }

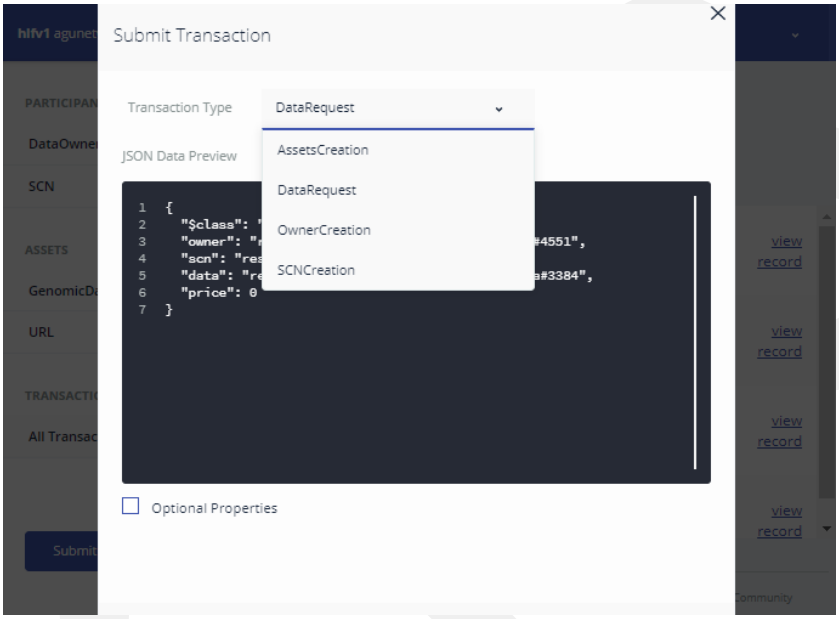

Figure 8.1.4 Transactions of AguWork

At the same time, data owners submit their data information and data index as assets. The following function automatically create two different asset. GenomicData asset contains dataId, description (features of data), owner (pointer for data owner) and price (price of data) parameters. URL asset contains urlId, url (data index) and data (pointer for genomic data) parameters.

{ "$class": "org1.aguhyper.xyz.AssetsCreation",
"dataId": "10",
"description": "GWAS",
"owner": "resource:org1.aguhyper.xyz.DataOwner#100",
"price": 250,
"urlId": "1",
"url": "www.*",
"data": "resource:org1.aguhyper.xyz.GenomicData#10" }

Secure compute nodes join the AguWork with their scnId and affiliations as the following.

```
{  "$class": "org1.aguhyper.xyz.SCNCreation",
  "scnId": "25",
  "affliation": "AGU" }
```

DataRequest is a transaction which the url of the data is requested by SCN, and if the appropriate conditions are provided, access to the data url is granted with SCN. It contains owner (pointer for data owner), scn (pointer for SCN), data (pointer for requested genomic data) and price (the payment amount sent to data owner for its genomic data by SCN) parameters as the following.

```
{  "$class": "org1.aguhyper.xyz.DataRequest",
  "owner": "resource:org1.aguhyper.xyz.DataOwner#100",
  "scn": "resource:org1.aguhyper.xyz.SCN#25",
  "data": "resource:org1.aguhyper.xyz.GenomicData#10",
  "price": 500 }
```

Permission on assets, participants and transaction in the system are as follows:

- Owners can read SCN information
- Owners can read data request transactions
- Owners have full access to their assets
- SCNs can read genomic data and its owner information
- Nobody can access URL information except its data owners
- SCNs can submit DataRequest transactions
- If the appropriate conditions are provided, the SCN gets the permission to read the url information of the relevant genomic data.

For providing data request transaction, there are two types different way as using smart contract and chain-code. While smart contract usage is a user-sided solution method, the usage of chain-codes provides a administrators-sided solution method. The purpose of these two methods is to update the information of the relevant assets or the participant

automatically as determined when appropriate conditions occur. We preferred to use chain-code for data request transaction. In the GenShare, each data owner has a genomic data and URL asset. For data request transaction, if owner of genomic data and URL assets is updated to be secure compute node, the data owner cannot sell the same data for the second time, in order to sell it, data owner must create assets for the same data again. This means an extra load on the AguWork during each data sharing.

For this reason, instead of changing owner of the genomic data and URL assets, GenShare preferred initially blocked access to URL assets everyone except the data owner. Before the DataRequest transaction takes place, SCN searches the genomic data on the AguWork. In the DataRequest transaction, when it finds the genomic data that it is looking for, SCN pays the price of the data. If the amount of money sent is same with the price of the genomic data, SCN automatically has the right to read the relevant URL asset. Otherwise access to URL asset is still restricted. With all these approaches, an extra load on the AguWork was prevented.

8.2 Testing

Apache JMeter is operated to test the performance that using the Hyperledger Composer REST server [90] with several API calls. With these external calls, the system can be run over the global network instead of a local network. The reason for the worldwide network preference in sending requests is that real application of the GenShare cannot work locally, and at the same time, the effects of global network delays to the system were desired to show. Hyperledger Composer Rest Server are used to generate a REST API from a deployed Hyperledger Fabric business network that can be easily consumed by HTTP or REST clients. Figure 8.2.1 shows the generated APIs of GenShare. At runtime, Hyperledger Composer REST server implements Create, Read, Update, and Delete (CRUD) operations to manipulate the state of assets and participants and allow

transactions to be submitted or retrieved with queries. Runtime operations of the GenShare APIs are shown in the Figure 8.2.2. Angular-based web applications can be created to use Hyperledger Composer, but web applications also should make REST API calls to interact with a deployed business network [91-92]. Figure 8.2.3 shows the Angular-based web application of AguWork.

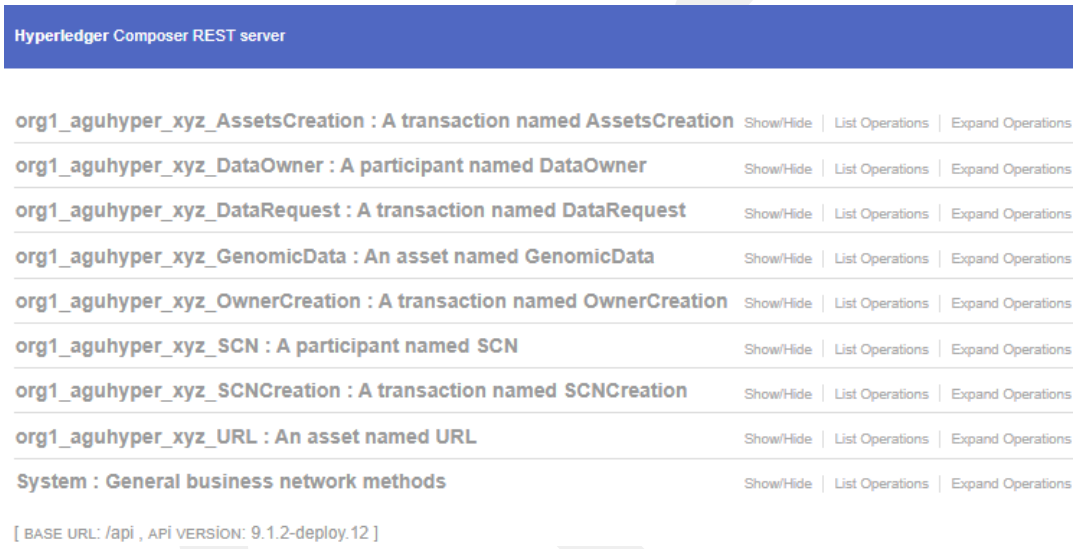

Figure 8.2.1 APIs of GenShare

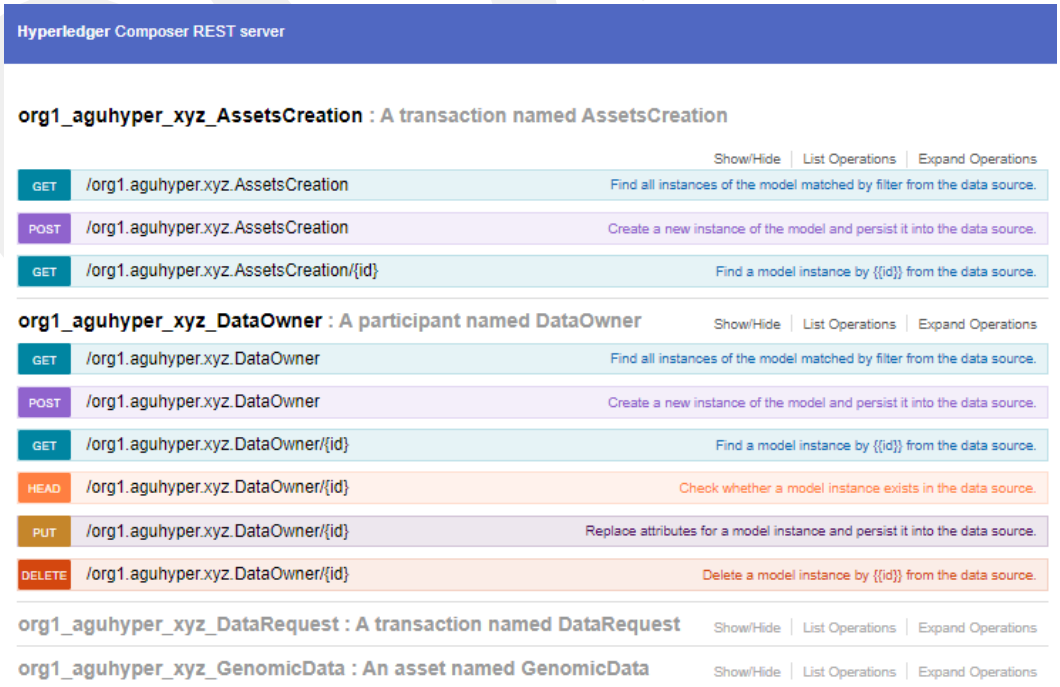

Figure 8.2.2 Runtime operations of GenShare APIs

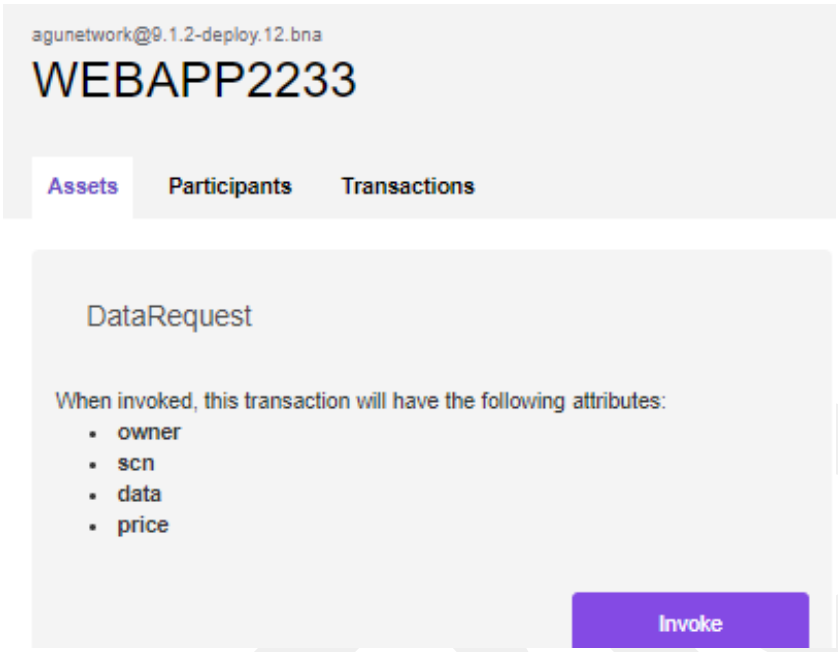

Figure 8.2.3 Angular-based web application of AguWork

The Apache JMeter is a software which is designed to load test functional behavior and measure performance [93]. It can be used to analyze and measure the performance of web application or a variety of services like a functional test, database server test etc. Performance testing means testing a web application against heavy load, multiple and concurrent user traffic. For API calls, the Apache JMeter is preferred in this thesis. Also, Hyperledger Composer Query could be used instead of it. But the Apache JMeter outputs were more suitable for the GenShare system to the calculations made in the result section.

8.3 Results

The performance of the Hyperledger Fabric network in the relevant transactions and workloads is shown in this section. Firstly, the Hyperledger performance metrics, which are used for measurement will be explained before workloads, results, and inferences of test. There are three types of performance metric used in evaluation as following [94]:

- **Average Transaction Latency:** Transaction latency involves the time from the point that it is submitted to the point that the result is widely available in the network. This includes the propagation period and any settling period owing to the developed consensus process. In this thesis, average transaction latency is shown.

  Average Transaction Latency = Total confirmation time – submit time (calculated for each transaction) / number of transactions

- **Transaction Throughput:** is the rate at that valid transactions are committed over a given time period by the blockchain system under test (SUT). At a network scale, this rate is represented as transactions per second (TPS).

  Transaction Throughput = Total committed transactions / total time in seconds

- **Read Latency:** is the time between when the read request is submitted and when the reply is received.

  Read Latency = Time when response received – submit time

Measurements on the AguWork run in two fundamental scenarios: while 100 nodes and 500 nodes in the system. When workload is both 100 nodes and 500 nodes, 10% of nodes are considered as SCN, 90% as data owner. It means that while workload is 100, the system has 10 SCN and 90 data owners. Each owner has one genomic data and url asset. While workload is 500, the system has 50 SCN and 450 data owner. Also, each owner has one genomic data and url asset. While analyzing the results data, it should be remembered that API calls made to the system are sent externally over the global network instead of the local network. As a result of this, Internet speed affected the results certain extent and constant increase rate could not be observed among them.

Firstly, creation of data owners, SCN and assets are examined for both workloads and two figures are obtained as Figure 8.3.1 and Figure 8.3.2. In the Figure 8.3.1, average transaction latencies of the participant and assets creation process are shown. When the workload changed, the number of transactions increased five times, and as expected, when the number of transactions increased, the average transaction latencies increased too. But the point to be emphasized here is that the transaction numbers of data owners and assets are equal. However, the transaction delays of asset creations are approximately twice that of the data owner, because the size of the transmitted message is different. The bytes sent in the creation of data owner, SCN and assets are follows, respectively: 215 bytes, 294 bytes and 489 bytes. From this, it is understood that the increase in the size of the message, such as an increase in the number of transactions, affects the transaction latencies.

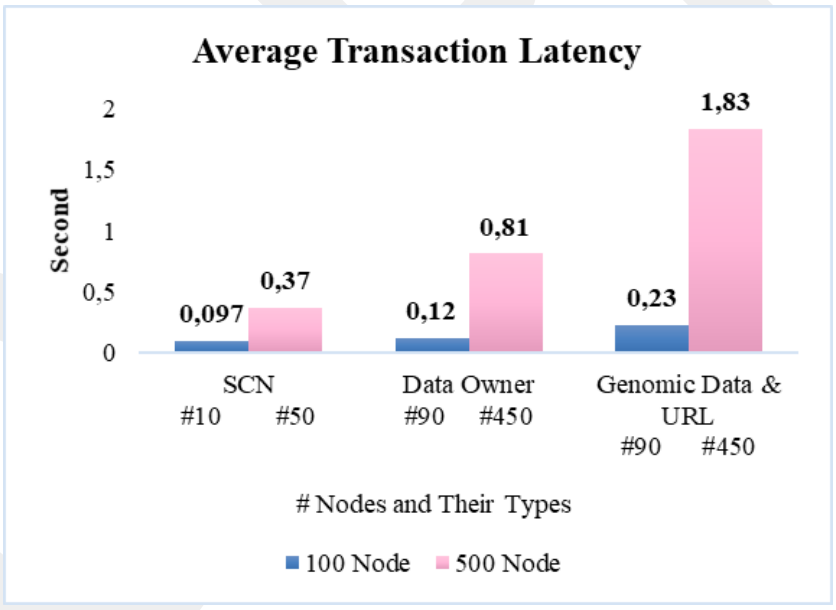

Figure 8.3.1 Average transaction latencies of participant and assets creation

In the Figure 8.3.2, transaction throughput of the participant and assets creation process are shown. When the workload changed, the number of transactions increased and as expected, when the number of transactions increased, the transaction throughput decreased. As mentioned in Figure 8.3.1, although the transaction numbers of data owners and assets are equal, the transaction throughput of asset creations are approximately twice less that of the data owner due to the size of the transmitted

message of them are different. Thus, it is shown that message size affects the transaction throughput performance.

There are some details needed to be explained before showing the results for other measurements. Different workload rates are determined as a parameter for the rest of the tests according to the obtained results. While deciding test scenarios, restrictions imposed by the AguWork and the hardware it works on are analyzed, and the results are given in Table 8.3.1, Table 8.3.2, and Figure 8.3.3. According to our tests AguWork system allows max 900 transactions at the same time because of the limitations. If there are more than that amount of transactions, scheduling techniques will be implemented as a future work. Workload performance can be improved by increasing the power of the hardware where the nodes are installed and by changing the settings made in the Hyperledger core installation. With the hardware resources of AguWork, the specified number of nodes for different roles has the optimum proportion for the comparison of network performance from different angles.

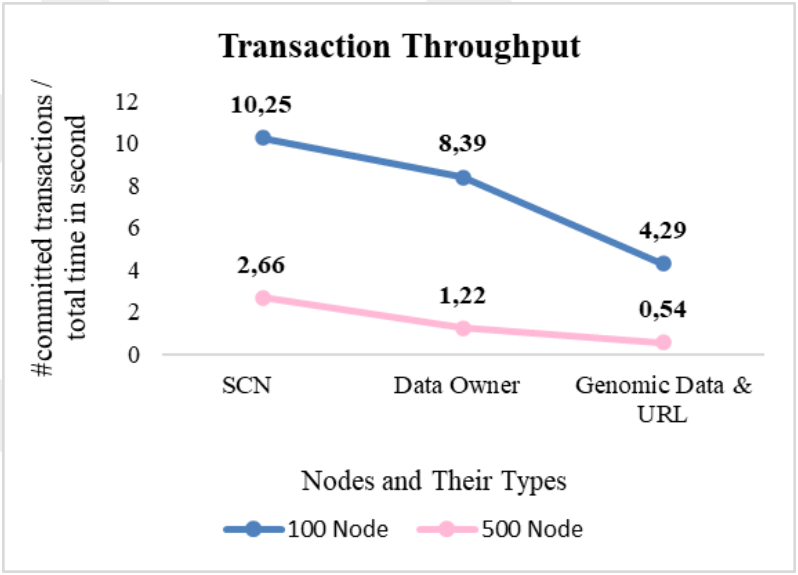

Figure 8.3.2 Transaction throughput of the participant and assets creation

According to different workloads, Table 8.3.1 shows average transaction latencies and average transaction throughput results when data request transactions changed. In one scenario, node distribution in AguWork is designed as 10% SCN and 90% data owner a

as in the node creation process. There are two workloads are applied on 100 nodes and 500 nodes as follows:

1. Simultaneously 10% of the SCNs in the system, request to buy the genomic data from respectively 1, %5 and %10 of the data owners.
2. Simultaneously 20% of the SCNs in the system, request to buy the genomic data from respectively 1, %5 and %10 of the data owners.
    - For example: When there are 100 nodes in the system, 20% means only 2 of them are SCN. When these nodes request to buy genomic data from respectively 1%, 5% and 10% of the data, that means, each of the 2 SCN respectively buy one, five and nine genomic data simultaneously, and separately.

For 100 Node	1 SCN		2 SCN	
	Average Transaction Latency	Transaction Throughput	Average Transaction Latency	Transaction Throughput
1 Owner	0,15	6,62	0,14	6,83
5 Owner	0,14	7,17	0,13	7,95
9 Owner	0,13	7,50	0,12	8,3
For 500 Node	**5 SCN**		**10 SCN**	
	Average Transaction Latency	Transaction Throughput	Average Transaction Latency	Transaction Throughput
1 Owner	0,375	2,66	0,39	2,53
23 Owner	1,57	0,63	1,37	0,73
45 Owner	1,73	0,57	1,9	0,51
Results are expressed in seconds.				

Table 8.3.1 Average transaction latencies and transaction throughput for the data request transaction

When the results obtained for the 100 nodes are examined, it is seen that as the number of transactions increases, the average transaction latency decreases and the transaction throughput increases. In contrast, when the results obtained for the 500 nodes are examined, it is seen that as the number of transactions increases, the average transaction latency increases and the transaction throughput decreases. Normally, the expected results for both scenarios should have followed the same trend as obtained with 500 nodes. It shows that, when there are 100 nodes in the system with a low level of workload, the worker nodes have enough power to handle all the requests since there is no congestion. Therefore, while transaction throughput increases, latency decreases naturally.

In general, when the number of transactions increased, while transaction latency is increasing, throughput is decreasing due to a large volume of requests on the worker nodes. However, the point that should be emphasized here is that although transaction numbers of the parts marked with a red and blue parenthesis in Table 8.3.1 are the same, their results have a different pattern. In other words, making five data sharing requests by one SCN and sending one data sharing request by five different SCN, in total five transactions for both scenarios, does not give the same results. It is shown that as the that as the number of SCN making the request increases in cases where the total number of transactions is the same, average transaction latency increases and transaction throughput values decrease.

According to different workloads, Table 8.3.2 shows average transaction latencies and transaction throughput for the data request transactions when there are several collision rates. Four workloads are applied on 100 nodes and 500 nodes as follows:

1. Simultaneously 20% of the SCNs in the system, request to buy the genomic data from respectively 5% and 10% of the data owners with 10% collision .
2. Simultaneously 20% of the SCNs in the system, request to buy the genomic data from respectively 5% and 10% of the data owners with 25% collision .
3. Simultaneously 20% of the SCNs in the system, request to buy the genomic data from respectively 5% and 10% of the data owners with 50% collision .

4. Simultaneously 20% of the SCNs in the system, request to buy the genomic data from respectively 5% and 10% of the data owners with 100% collision .
    - For example: When there are 100 nodes in the system, 20% means there are 2 SCNs. When these nodes request to buy genomic data from respectively 5% and 10% of the data owners with 100% collision, means from the same set of data owners. It means that simultaneously, each of the 2 SCN respectively buy five and nine genomic data from the same data owner.

When collision-included scenarios applied, Table 8.3.2 shows a similar pattern on latency and throughput analysis, as in Table 8.3.1.

For 100 Node 2 SCN	**10% Collision**		**25% Collision**		**50% Collision**		**100% Collision**	
	ATL	TT	ATL	TT	ATL	TT	ATL	TT
5 Owner	0,61	1,645	0,64	1,56	0,66	1,51	0,63	1,58
9 Owner	0,51	1,96	0,53	1,89	0,54	1,85	0,53	1,92
For 500 Node 10 SCN	**10% Collision**		**25% Collision**		**50% Collision**		**100% Collision**	
	ATL	TT	ATL	TT	ATL	TT	ATL	TT
23 Owner	1,96	0,51	1,98	0,50	2	0,49	2,05	0,48
45 Owner	2,43	0,41	2,573	0,388	2,75	0,362	2,86	0,349

ATL: Average Transaction Latency **Results are expressed in seconds.**
TT: Transaction Throughput

Table 8.3.2 With collisions, average transaction latencies and transaction throughput for the data request transaction

Once again, when the workload for 100 nodes is not high, and the system occupancy rate is low, the system will perform more consistent results as expected. The main goal here is to get an idea of AguWork performance when the same data was bought by more

than one SCN at the same time that shows the load on the data owner node. It is shown that, as the number of collision increases in both workloads, average transaction latency increases and transaction throughput decreases.

According to different workloads, Figure 8.3.3 shows the read latencies for transactions. There are three workloads are applied by 1 node, 5 node and 10 nodes as follows:

1. One participant, respectively request to read one, fifty and hundred transactions.
2. Simultaneously 5 participant, respectively request to read one, fifty and hundred transactions.
3. Simultaneously 10 participant, respectively request to read one, fifty and hundred transactions.

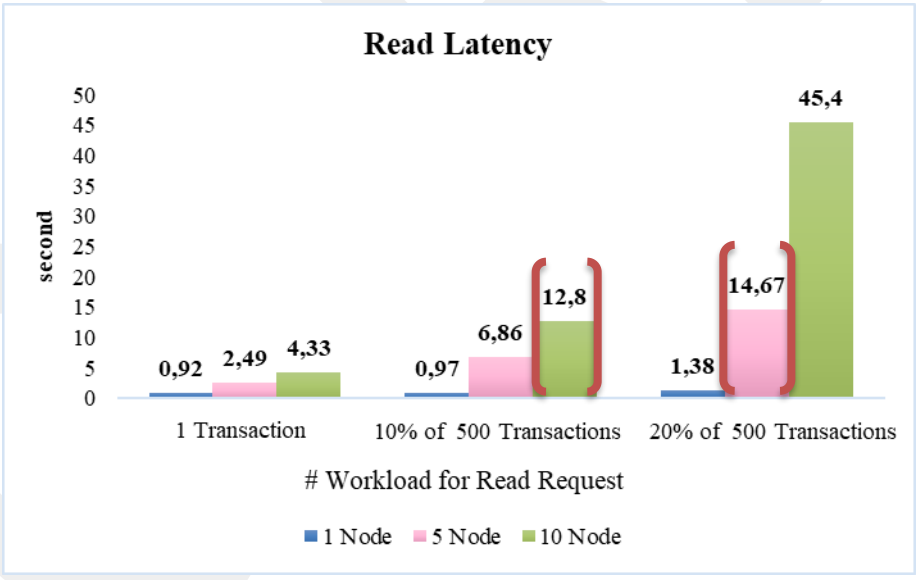

Figure 8.3.3 Read latencies for transactions

When the results are examined, it is seen that as the number of the read request for transactions increases, the read latency increases. Therewithal, although read request numbers of the parts marked with a red parenthesis in Figure 8.3.3 are the same, their results are different. It is shown that as the number of node making the read request decreases, read latency increases. This is the opposite of the situation observed in Table 8.3.1. Based on this, please note that; for POST method in cases where the total number

of transactions is the same, as the number of node making the request increases, average transaction latency increases but for GET method in cases where the total number of transactions is the same, as the number of node making the request decreases, average transaction latency increases .

Chapter 9

Evaluation of the Genshare

9.1 Contributions of the GenShare

The main contributions of proposed hybrid platform are follows:

- When creating the GenShare model, the sytem structure is divided into two categories as obtaining the genomic data, inclusion of nodes in the GenShare and relation between nodes. The reason for categorizing the system structure was to evaluate all the solutions that can be applied for each stage and choose the most suitable methods among them for the system requirements of GenShare. The various solutions presented in this approach will provide insight for many other systems.
- Obtaining genomic data and data verification process have not been considered and explained in any system. This is a subject that is meticulously handled in the GenShare system.
- GenShare is a hybrid platform that is based on blockchain, homomorphic encryption and SGX technologies. There are genomic data computation studies developed with SGX and homomorphic encryption in the literature and genomic

data sharing studies with a single blockchain structure. Combining different blockchain technologies and providing interactions between them is a field that has just been studied and does not have an exemplary representation in bioinformatics. While examining all these studies, the advantages, disadvantages and problems still waiting to be solved are listed and it has been realized that all of them combined with appropriate rules into a novel hybrid structure will improve the advantages and close the number of open issues. In line with this idea, the GenShare model was presented.

- With the help of homomorphic encryption and SGX technology, data privacy and security problems are solved. Thus anyone other than the data owner cannot access the original genomic data. Also through inclusion of powerful SGX supported servers in the system, statistical analysis and count query operation of genomics are performed easily.
- Using blockchain in data sharing process, data owners take control of their data access permissions and authorization. Also they make money from this process. All of positive yields has encouraged data owners to share their data. So a helpful proposal has been created for the problem of not being able to collect the desired number of data in scientific studies.
- For providing data sharing process, there are two types different way as using smart contract and chain-code. We preferred to use chain-code for data request transaction. In the GenShare, if owner of genomic data and URL assets is updated to be secure compute node, the data owner cannot sell the same data for the second time, in order to sell it, data owner must create assets for the same data again. This means an extra load on the AguWork during each data sharing. For this reason, instead of changing owner of the genomic data and URL assets, GenShare preferred initially blocked access to URL assets everyone except the data owner. In this way, when SCN pays the data owner the price of the data, SCN automatically has the right to read the relevant URL asset and an extra load on the AguWork was prevented.

- AguWork was set up on two different physical nodes which was communicated on global network. The reason for the global network preference in communication of physical nodes is that while examining AguWork performance, to achieve realistic results were desired. Because if the nodes provided communication over the local network, the latency results would be lower than it should be in the actual application, and the throughput results would be more.
- In the test section, the API calls made to the system were sent externally over the global network instead of local network. The reason for the global network preference in sending requests is that real application of the GenShare cannot work locally and at the same time, the effects of global network delays to the system were desired to show.
- Many different workload rates are determined for performance measurements made in the system according to the hardware resources of AguWork.

9.2 Security Analysis

When creating the proposed model, the sytem structure is divided into some categories and evaluate all possible solutions that can be applied on each category, and the most suitable methods among solutions are preferred according the system requirements of GenShare. The purpose of this categorization is to minimize any security problems that could occur. In GenShare, genomic data is stored in a private storage with homomorphic encrypted form by owner, and computations are performed on encrypted data safely through the use of SGX, which ensures data privacy and security. The index of encrypted data is shared on the Hyperledger Fabric network. GenShare preferred initially blocked access to index of data everyone except the data owner. In this way, when SCN pays the data owner the price of the data, SCN automatically has the right to read the relevant index of data. Thus unauthorized user can't obtain the access permission are prevented [95]. As data verification mechanism, the GenShare recommend keeping the hash values of the data in labs. Thus through APIs, the SCN easily understand whether the data is original or not by comparing their hash values and

data falsifications are prevented. If a user has malicious actions, the system can blacklist the account and all attacks will be recorded as evidences. For preventing denial-of-service (DDoS) attacks [96], when at least three researchers come together, they can request to process the data thanks to threshold cryptography. Secure compute nodes occur from certain universities in the system but there may be trust problems in each system. To prevent this, nodes that are in charge to validate the computation results of SCNs are included in the system.

Chapter 10

Discussions and Conclusion

In this thesis, a hybrid GenShare platform is proposed for genomic data gathering and sharing with bioinformatics researchers. The system is designed as a combination of Hyperledger Fabric and Ethereum platforms. Also, homomorphic encryption and Intel SGX technologies are applied for sharing genomic data with the privacy-preserving way among all types of users. Thus, all computations on genomics data will be calculated in a secure and safe way.

Developments in genomics play a significant role in human life. In order to increase the efficiency of the genomics based approaches used to protect human health, researchers should collect more genomic data, but there are some difficulties for obtaining and sharing genomic data such as, high analysis costs, an inability on data access permissions, limitations on data privacy and security, and storage of massive amount of data. The proposed model brings new solutions to these problems using combined blockchain, homomorphic encryption and SGX technologies. With homomorphic encryption and SGX, data privacy and security related problems are solved feasibly. In the GenShare model, the genomic data is processed by the owner in a private storage with homomorphically encrypted form and the computations are safely performed on encrypted data using SGX. Other problems have been solved with Hyperledger Fabric

and Ethereum-derived GenShare. Thanks to these two technologies, data owners and researchers can communicate anonymously without an intermediary; data owners and secure compute nodes can make money; researchers can access more data; and data owners can control their data access permissions.

In the proposed system, while the researchers are included in the public side of the GenShare network, the data owners are included only in the permissioned part of GenShare. Communication between these two chains is provided by SCNs, which are linked to both blockchain structures. When a researcher wants to study genomic data, it prepares smart contract with the specified data type and operations on the GenShare. The system assigns an SCN for this process, and related data is searched in the Hyperledger network. We would like to remind that the data owner has two types of an asset: i) genomic data information, and ii) address of data. For the data request processes, GenShare initially prefers blocked access to URL assets for everyone except its owner. As soon as the data is found, SCN pays the price of the genomic data. If the amount of money sent is same with the price of the genomic data, SCN automatically has the right to read the relevant URL asset. If the SCN, who has the right to access the data verifies the data, the paid fee is automatically transferred to the account of the data owner. Then, computations of the researcher are made, and the result is provided to the researcher.

Implementation of the GenShare system is divided into three parts as (i) providing communication between data owners and SCNs, (ii) applying secure computations on shared data, and (iii) establishing communication between SCNs and researchers. Making these three-part operations will take a long time, and providing communication between the data owners and the SCNs is more critical for the initial phase. For these reasons, the Hyperledger Composer Business Network, which is named the AguWork, is set up on two different physical nodes, which communicate via the worldwide network. After AguWork is set up operationally, API calls were performed by Apache JMeter using the Hyperledger Composer REST server externally over the global network. In the testing part, these API calls are performed with both the same and

different workloads, and the performance of the AguWork is evaluated according to well-known metrics like average transaction latency, transaction throughput, and read latency. The performance of AguWork is measured by applying two different scenarios that include 100 nodes and 500 nodes in the system. Different proportions of different roles of users are also tested. Firstly, the creation performance of data owners, SCN, and assets are examined via considering 10% of nodes as SCN, 90% as the data owner. Then according to the hardware resources of AguWork, some parameters were optimized for different workload scenarios. The same scenarios are applied for totally different test cases and collision-included ones. Finally, the performance of transaction reading requests is tested according to specific test cases.

As a result, a steady increase rate should not be expected among the results obtained from the global network test, since the internet speed will affect the results. In test cases, when the number of transaction increases, the average transaction latency increases, and transaction throughput decreases, but the only thing that affects metrics is not the number of transactions. For two different workloads with the same total number of processes, the average transaction latency increases as the transaction throughput decreases when the message size increases, and the number of nodes that send request to buy genomic data increases. On contrary to sending request to buy genomic data, for two different workloads with the same total number of processes, the average transaction latency increases as the transaction throughput decreases when the number of nodes which send request to read transactions decreases. Finally, when the workload and occupancy rate increases at a certain rate, the system will give more consistent results. In conclusion, the proposed GenShare model is suitable for projects where more data collection and sharing are required, such as the Turkish Genome Project [97]. We believe that the proposed model will accelerate the completion time of this type of projects and will be the most efficient platform for its users. It is worth to note that the higher the capacity of the hardware resources used, the higher the performance of the system. As a future work, we aim to develop the remaining parts of the system and make the whole system fully operational.

BIBLIOGRAPHY

[1] NM. Luscombe, D. Greenbaum and M. Gerstein, 'What is bioinformatics? A proposed definition and overview of the field' Methods of information in medicine, 40(04), 346-358, (2001).

[2] JA. Reuter, DV. Spacek and MP. Snyder, 'High-throughput sequencing technologies' Molecular cell, 58(4), 86-597, (2015).

[3] W. Diniz, F. Canduri, 'Bioinformatics: an overview and its applications', Genet Mol Res, 16 (2017).

[4] JP. Hubaux , S. Katzenbeisser , B. Malin, 'Genomic Data Privacy and Security: Where We Stand and Where We Are Heading', IEEE Security-Privacy, 15(5), 10-12, (2017).

[5] A. Sboner, XJ. Mu, D. Greenbaum, RK. Auerbach and MB. Gerstein, 'The real cost of sequencing: higher than you think!' Genome biology, 12(8), 125, (2011).

[6] NH. Steneck, 'Introduction to the responsible conduct of research', Washington, DC: US Government Printing Office, (2007).

[7] ZD. Stephens, SY. Lee, F. Faghri , RH. Campbell, C. Zhai et al. 'Big data: astronomical or genomical?', PloS biology, 13(7), (2015).

[8] JI Choi, KR. Butler, 'Secure Multiparty Computation and Trusted Hardware: Examining Adoption Challenges and Opportunities', Security and Communication Networks, (2019).

[9] J. Yli-Huumo, D. Ko, S. Choi, S. Park, K. Smolander, K, ' Where is current research on blockchain technology?—a systematic review' PloS one, 11(10), e0163477, (2016).

[10] Ethereum Website, https://www.ethereum.org/

[11] Hyperledger Website, https://www.hyperledger.org/

[12] N. Drucker, S. Gueron, ‘Combining Homomorphic Encryption with Trusted Execution Environment: A Demonstration with Paillier Encryption and SGX’, In Proceedings of the 2017 International Workshop on Managing Insider Security Threats, 85-88, (2017).

[13] ‘Basic Genetics’, https://kintalk.org/genetics-101/ .

[14] DC. Koboldt, KM. Steinberg, DE. Larson, RK. Wilson and MR. Mardis, ‘The next-generation sequencing revolution and its impact on genomics’, Cell, 155(1), 27-38, (2013).

[15] JC. Shendure, S. Balasubramanian, GM. Church, W. Gilbert, J. Rogers and et al., ‘DNA sequencing at 40: past, present and future’, Nature, 550(7676), 345, (2017).

[16] RD. Hawkins, GC. Hon and B. Ren, ‘Next-generation genomics: an integrative approach’, Nature Reviews Genetics, 11(7), 476, (2010).

[17] NS. Abul-Husn and EE. Kenny, ‘Personalized medicine and the power of electronic health records’, Cell, 177(1), 58-69, (2019).

[18] K. Schwarze, J. Buchanan, JM. Fermont, H. Dreau, MW. Tilley et al., ‘The complete costs of genome sequencing: a microcosting study in cancer and rare diseases from a single center in the United Kingdom’, Genetics in Medicine, 1-10, (2019).

[19] JH. Tibbetts, ‘Should Individuals Share Their Genomic Profiles? Researchers and patient advocates wrestle with privacy and ethical concerns’, BioScience, 68(9), 633-639, (2018).

[20] A. Gutmann, J. Wagner, Y. Ali, AL. Allen, JD. Arras et al. ‘Privacy and progress in whole genome sequencing’, Presidential Committee for the Study of Bioethical, (2012).

[21] A. Mittos, B. Malin, E. De Cristofaro, ‘Systematizing genome privacy research: a privacy-enhancing technologies perspective’, Proceedings on Privacy Enhancing Technologies, 2019(1), 87-107, (2019).

[22] CA. Azencott, ‘Machine learning and genomics: precision medicine versus patient privacy’, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 376(2128), 20170350, (2018).

[23] W. Xie, M. Kantarcioglu, WS. Bush, D. Crawford D, JC. Denny et al. ‘SecureMA: protecting participant privacy in genetic association meta-analysis’, Bioinformatics: 30(23), 3334-3341, (2014).

[24] P. Baldi, P. Baronio , E. De Cristofaro E, P. Gasti and G. Tsudik, ‘Countering gattaca: efficient and secure testing of fully sequenced human genomes’, In Proceedings

of the 18th ACM conference on Computer and communications security, 691-702, (2011).

[25] XS. Wang, Y. Huang, Y. Zhao, H. Tang, X. Wang et al. 'Efficient genome-wide, privacy-preserving similar patient query based on private edit distance', In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, 492-503, (2015).

[26] B. Hayes, 'Alice and Bob in cipherspace' American Scientist, 100(5), 362-367, (2012).

[27] C. Moore, M. O'Neill, E. O'Sullivan, Y. Doröz, B. Sunar, 'Practical homomorphic encryption: A survey', IEEE International Symposium on Circuits and Systems (ISCAS), 2792-2795, (2014).

[28] PV. Parmar, SB. Padhar, SN. Patel, NI Bhatt, RH. Jhaveri, 'Survey of various homomorphic encryption algorithms and schemes', International Journal of Computer Applications, 91(8), (2014).

[29] M. Naveed , E. Ayday, EW. Clayton , J. Fellay, CA. Gunter et al. ' Privacy in the genomic era', ACM Computing Surveys (CSUR), 48(1), 6, (2015).

[30] A. Maxmen, 'AI researchers embrace Bitcoin technology to share medical data', Nature, 555(7696), (2018).

[31] P. Mamoshina, L. Ojomoko, Y. Yanovich, A. Ostrovski, A. Botezatu et al., 'Converging blockchain and next-generation artificial intelligence technologies to decentralize and accelerate biomedical research and healthcare', Oncotarget, 9(5), 5665, (2018).

[32] S. Angraal, HM. Krumholz, WL. Schulz, 'Blockchain technology: applications in healthcare', Circulation: Cardiovascular quality and outcomes, 10, 1–3, (2017).

[33] S. Nakamoto, 'Bitcoin: A Peer-to-Peer Electronic Cash System', 1-9, (2008), https://bitcoin.org/bitcoin.pdf.

[34] M. Di Pierro, 'What is the blockchain?', Computing in Science & Engineering, 19 (5), 92-95, (2017).

[35] TT. Kuo, HE. Kim, L. Ohno-Machado, 'Blockchain distributed ledger technologies for biomedical and health care applications', J Am Med Informatics Assoc 24**,** 1211–1220, (2017).

[36] D. Yaga, P. Mell, N. Roby, K.Scarfone,' Blockchain technology overview', arXiv preprint arXiv:1906.11078, (2019).

[37] JI. Zahid, A. Ferworn, F. Hussain, 'Blockchain: A technical overview', IEEE Internet Policy Newsl, 1-3, (2018).

[38] V. Mavroeidis, K. Vishi, MD. Zych, A. Jøsang, 'The impact of quantum computing on present cryptography', arXiv preprint arXiv:1804.00200, (2018).

[39] IF. Kaderali, 'Foundations and Applications of Cryptology', (2007).

[40] Y. Kumar, R. Munjal, R, H. Sharma. 'Comparison of symmetric and asymmetric cryptography with existing vulnerabilities and countermeasures', International Journal of Computer Science and Management Studies, 11(03), 60-63, (2011).

[41] K. Sultan, U. Ruhi, R. Lakhani, 'Conceptualizing Blockchains: Characteristics & Applications', arXiv preprint arXiv:1806.03693, (2018).

[42] O. Dib, KL. Brousmiche, A. Durand, E. Thea, EB. Hamida, 'Consortium blockchains: Overview, applications and challenges' International Journal On Advances in Telecommunications, 11(1&2), 2018.

[43] H. Anwar, 'Consensus Algorithms: The Root Of The Blockchain Technology', 101 Blockchains Website, (2018).

[44] D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei, C. Qijun, 'A review on consensus algorithm of blockchain', IEEE Int Conf Syst Man, Cybern SMC 2017, 2567–2572, (2017).

[45] V. Gatteschi, F. Lamberti, C. Demartini, C. Pranteda, V. Santamaria. 'To blockchain or not to blockchain: That is the question', IT Professional, 20(2), 62-74, (2018).

[46] F. Casino, TK. Dasaklis, C. Patsakis, 'A systematic literature review of blockchain-based applications: current status, classification and open issues', Telematics and Informatics, 36, 55-81, (2019).

[47] K. Wüst, A. Gervais, 'Do you need a blockchain?', In 2018 Crypto Valley Conference on Blockchain Technology (CVCBT), 45-54, (2018).

[48] HI. Ozercan, AM. Ileri, E. Ayday, C. Alkan, 'Realizing the potential of blockchain technologies in genomics', Genome research: 28(9), 1255-1263, (2018).

[49] JC Goldwater, 'The Use of a Blockchain to Foster the Development of Patient-Reported Outcome Measures', https://goo.gl/WHVmtT .

[50] P. Taylor, 'Applying blockchain technology to medicine traceability', http://goo.gl/r6cTN3 .

[51] IBM Global Business Services Public Sector Team, ‘Blockchain: The Chain of Trust and its Potential to Transform Healthcare - Our Point of View’, https://goo.gl/oYaC5k .

[52] D. Grishin, K. Obbad, P. Estep, M. Cifric, Y. Zhao et al. ‘White Paper: Nebula Genomics; Blockchain-enabled genomic data sharing and analysis platform’, Harvard University, (2018), https://nebula.org/ .

[53] N. Kulemin, S. Popov, A. Gorbachev, ‘The Zenome Project : Whitepaper blockchain-based genomic ecosystem’, https://zenome.io/ .

[54] B. Schorchit, BA. Monteiro, FC. Gouveia, A. Fischer, M. Rebelo M, ‘Meet Genecoin : the Bioeconomy Currency’, (2018), http://genecoin.science/ .

[55] ‘The Clinical and Investment Potential in the Gene-Chain Project The Unprecedented Growth of Genomic Data’, (2017), https://encrypgen.com/

[56] ‘DNATIX; Genetic Blockchain Ecosystem, Whitepaper’, (2018), https://www.dnatix.com/ .

[57] A. Lipman, B. Ekblaw, A. Johnson, K. Camaron, N. Retzepi, et al. ‘Technical Documentation: Medrec. MIT Media Lab’, (2017), https://medrec.media.mit.edu/ .

[58] ‘White Paper: IRYO; Global participatory healthcare ecosystem’, (2017), https://IRYO.io/ .

[59] A. Park, J. Mullin, A. Mah, M. Gallo, L. Cyca et al. ‘The Blockchain for Personalized Medicine’, (2017), https://mycoralhealth.com/product/ .

[60] ‘ White Paper: Open Longevity’, (2017), http://eng.openlongevity.org/ .

[61] C. Mcfarlane, M. Beer, J. Brown, N. Prendergast. ‘Patientory: A Healthcare Peer-to-Peer EMR Storage Network v1.1.’, https://patientory.com/ .

[62] ‘Whitepaper: Medicalchain 2.1’, https://Medicalchain.com/en/ .

[63] S. Kannan, M. Smith. ‘GemOS Platform Whitepaper’, (2016), https://enterprise.gem.co/ .

[64] ‘e-Estonia’, https://www.guardtime-federal.com/ksi/ .

[65] ‘Health Nexus’, https://www.simplyvitalhealth.com/ .

[66] W. De Brouwer, M. Borda. ‘NeuRoN: Decentralized Artificial Intelligence, Distributing Deep Learning to the Edge of the Network’, (2017), https://www.doc.ai/NeuRoN/ .

[67] T. Kumar, V. Ramani, I. Ahmad, A. Braeken, E. Harjula et al. ‘Blockchain Utilization in Healthcare: Key Requirements and Challenges’, In 2018 IEEE 20th International Conference on e-Health Networking, Applications and Services, (Healthcom); 1-7, (2018).

[68] TK. Mackey, TT. Kuo, B. Gummadi, KA. Clauson, G. Church et al. `Fit-for-purpose?'-challenges and opportunities for applications of blockchain technology in the future of healthcare 2019; BMC medicine: 17(1), 68, (2019).

[69] MN. Sadat, A. Aziz, M. Momin, N. Mohammed, F. Chen et al. ‘Safety: Secure gwas in federated environment through a hybrid solution’, IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 16(1), 93-102, (2019).

[70] W. Chenghong, Y. Jiang, N. Mohammed, F. Chen, X. Jiang X et al. 'SCOTCH: Secure Counting Of encrypTed genomiC data using a Hybrid approach’, In AMIA Annual Symposium Proceedings (Vol. 2017, p.1744): American Medical Informatics Association, (2017).

[71] Chen F, Wang S, Jiang X, Ding S, Lu Y et al. ‘PRINCESS: Privacy-protecting rare disease international network collaboration via encryption through software guard extensions’, Bioinformatics, 33(6), 871-878, (2016).

[72] Deloitte, U. S. ‘Blockchain: Opportunities for Health Care - A new model for health information exchanges’, (2016).

[73] TT. Kuo, H. Zavaleta Rojas, L. Ohno-Machado, ‘Comparison of blockchain platforms: a systematic review and healthcare examples’, Journal of the American Medical Informatics Association, 26(5), 462-478, (2019).

[74] D. Macrinici, C. Cartofeanu, S. Gao. ‘Smart contract applications within blockchain technology: A systematic mapping study’, Telematics and Informatics, (2018).

[75] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han X, ‘Blockchain-Enabled Smart Contracts: Architecture, Applications, and Future Trends’, IEEE Transactions on Systems, Man, and Cybernetics: Systems, (2019).

[76] CC. Agbo, QH. Mahmoud, JM. Eklund, ‘Blockchain technology in healthcare: a systematic review’, In Healthcare: Vol. 7, No. 2, 56, Multidisciplinary Digital Publishing Institute, (2010).

[77] AV. Aswin, B. Kuriakose, ‘An Analogical Study of Hyperledger Fabric and Ethereum’, In Intelligent Communication Technologies and Virtual Mobile Networks, 412-420, (2019).

[78] S. Schulte, M. Sigwart, P. Frauenthaler, M. Borkowski, ‘Towards Blockchain Interoperability’, In International Conference on Business Process Management, 3-10, (2019).

[79] A. Acar, H. Aksu, AS. Uluagac, MA. Conti, ‘A survey on homomorphic encryption schemes: Theory and implementation’, ACM Computing Surveys (CSUR), 51(4), 79, (2018).

[80] K. Lauter, A. Lopez-Alt, M. Naehrig, ‘Private computation on encrypted genomic data’, In International Conference on Cryptology and Information Security in Latin America:, 3-27, (2014).

[81] E. Ayday, JL. Raisaro, JP. Hubaux, J. Rougemont, ‘Protecting and evaluating genomic privacy in medical tests and personalized medicine’, In Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society, 95-106, (2013).

[82] M. Yasuda, T. Shimoyama, J. Kogure, K. Yokoyama, T. Koshiba, ‘Secure pattern matching using somewhat homomorphic encryption’, In Proceedings of the 2013 ACM workshop on Cloud computing security workshop, 65-76, (2013).

[83] Z. Brakerski, ‘Fundamentals of Fully Homomorphic Encryption-A Survey’, In Electronic Colloquium on Computational Complexity 2018 (ECCC), (25), 125, (2018).

[84] F. Chen, C. Wang, W. Dai, X. Jiang, N. Mohammed et al. ‘PRESAGE: privacy-preserving genetic testing via software guard extension’, BMC medical genomics: 10(2), 48, (2017).

[85] J. Eberhardt, S. Tai, ‘On or off the blockchain? insights on off-chaining computation and data’, In European Conference on Service-Oriented and Cloud Computing; Springer: Cham, 3-15, (2017).

[86] C. Stathakopoulous, C. Cachin, ‘Threshold signatures for blockchain systems’, Swiss Federal Institute of Technology, 30, (2017).

[87] Hyperledger Composer, https://www.hyperledger.org/projects/composer , (2019).

[88] Choosing private blockchain tech: Hyperledger Composer, https://hackernoon.com/choosing-private-blockchain-tech-hyperledger-composer-6ed61ea0dbc1 , (2018).

[89] Playground Tutorial, https://hyperledger.github.io/composer/v0.19/tutorials/playground-tutorial.html

[90] Interacting with Hyperledger Composer through RESTful API, https://developer.ibm.com/recipes/tutorials/interacting-with-hyperledger-composer-through-restful-api/ , (2019).

[91] Angular.js Website, https://angular.io/ .

[92] Writing Web Applications, https://hyperledger.github.io/composer/v0.19/applications/web

[93] Apache JMeter Website, https://jmeter.apache.org/

[94] Hyperledger Blockchain Performance Metrics White Paper, https://www.hyperledger.org/resources/publications/blockchain-performance-metrics#definitions

[95] H. Guo, W. Li, M. Nejad, CC. Shen, 'Access Control for Electronic Health Records with Hybrid Blockchain-Edge Architecture', arXiv preprint arXiv:1906.01188, (2019).

[96] J. Dheeraj, S. Gurubharan, 'DDoS mitigation using blockchain', International Journal of Research in Engineering, Science and Management, 1(10), (2018).

[97] C. Alkan, P. Kavak, M. Somel, O. Gokcumen, S. Ugurlu et al. 'Whole genome sequencing of Turkish genomes reveals functional private alleles and impact of genetic interactions with Europe, Asia and Africa', BMC genomics, 15(1), 963, (2014).