
# Capstone Project Phase A

## Blockchain private key security

**Project number** - 23 - 1 - D - 16

Supervisor: Alex Keselman

Karin Shpigelman - karinsh97@gmail.com

Nitzan Shani - nitzan963@gmail.com

# Table Of Contents


**1 Introduction** **3**
1.1 Blockchain Accounts Heist Statistics 4

**2 Background** **6**
2.1 Cryptocurrency and Blockchain 6
2.2 Cryptography 10
2.3 Previous Researches and Related Work 11

**3 Expected Achievements** **11**
3.1 Explore the Problem State 12
3.2 Investigate the Existing Approaches to the Problem Solution 12
3.3 Web Application 12

**4 Engineering process** **12**
4.1 Existing Solutions 13
4.2 The Process 14

**5 Product** **17**
5.1 Proposed Solution 17
5.2 Requirements 21
5.3 Used Technology 22
5.4 Architecture Diagram 23
5.5 Activity Diagram 24
5.6 Web Screens 25

**6 Evaluation/Verification Plan** **26**

**7 References** **27**

# Abstract

In this project, we present a new approach for securing private key in cryptocurrency exchanges and large-scale projects based on blockchain technology. The goal of the project is to reduce the risk of unauthorized access to the private key for the administrator wallet, without the need for human intervention. The proposed algorithm splits the private key into smaller parts and stores them in blocks, which are then distributed across multiple servers. This approach aims to mitigate the risk of unauthorized access to the private key and makes it more difficult for hackers to access and compromise the assets associated with it. The algorithm is demonstrated using a web application, which executes a transaction using the algorithm.


## 1 Introduction

We may be at the beginning of a new revolution. These days we are witnessing plenty of changes in the finance world. When talking about money transfers or investment services, a third party usually confirms and performs the transfer between people or invests the money.

There are some major problems in this system like inflation, slow transfer time, high commission and the main problem is the third party which in most cases is the bank, the government, or a person that controls the money. Blockchain is a peer-to-peer version of electronic cash allowing secure and transparent transactions without the need for a trusted third party, such as a bank. This eliminates the need for costly intermediaries and makes transactions faster and more efficient. Each Blockchain account owner has a private key, so whoever has the key is the account's owner and can perform transfer operations, etc.

The main problems with private keys in the blockchain are that they are vulnerable to being lost or stolen and they can be difficult to manage and secure.

One of the main challenges facing cryptocurrency exchanges, individuals, and large-scale projects based on blockchain is security. Because cryptocurrencies are digital and decentralized, they are vulnerable to hacks and other forms of cyber attacks. This has led to numerous incidents of exchanges being hacked and users losing their funds. For example in December 2021 Hackers withdrew $196 million of cryptocurrency from the crypto exchange BitMart by stealing private keys that opened two wallets.

Many individuals continue to use cryptocurrency exchanges as a means of storing and trading their digital assets due to the convenience and ease of conducting transactions through these platforms.

Cryptocurrency exchanges as a system automating transfers of a lot of valuable assets and having access to the funds locked in all the involved smart contracts, require a security layer solution for minimizing the risk of bad actors obtaining access to the private key to the admin wallet. Cryptocurrency exchanges often implement a combination of technical and operational measures to safeguard private keys, while individuals may be more vulnerable to losing their private key or having it stolen by hackers, due to a lack of resources or expertise. In this study, we will review various methods for securing private keys and propose an algorithm for doing so. By implementing effective technical and operational measures to protect private keys, the risk of hacking and theft of billions of dollars stored in cryptocurrency exchanges can be significantly reduced.

The algorithm we suggest is automated Private key storage, which involves dividing the key into several parts, storing each part on a different server, and using MPC to reconstruct the key when needed. We will examine the requirements and corresponding solutions for private key storage.

## *1.1 Blockchain Accounts Heist Statistics*

Over the past decade, the cryptocurrency market has undergone rapid growth, and as a result, hacking attempts targeting these digital assets have also increased. Three of the most notable hacking events include the Bitfenix exchange hack in 2016, the Coincheck exchange hack in 2018, and the Binance exchange hack in 2019. These incidents have brought attention to the need for improved security measures to protect against unauthorized access and theft of digital assets.

In August 2016, the Hong Kong cryptocurrency exchange Bitfinex experienced a major hack in which approximately 120,000 Bitcoin (BTC) worth around $72 million at the time were stolen from the exchange. This figure represents a substantial amount in today's market, with the current value of the stolen Bitcoin being valued at approximately $2,550,443,532.

The immediate effect on the price of Bitcoin was a sharp drop of about 20% following the news of the hack.

In January 2018, the Japanese cryptocurrency exchange Coincheck experienced a major hack in which approximately 523 million NEM coins worth around $534 million at the time were

stolen from the exchange. This incident was considered one of the largest hacks in the crypto-history. The exchange acknowledged its role in the security incident, stating that it had made the error of storing substantial amounts of user assets on hot wallets rather than cold wallets, which contributed to the vulnerability of the system and ultimately led to the loss of funds. In May 2019, the Binance cryptocurrency exchange experienced a hack in which hackers were able to steal 7,000 Bitcoin (BTC) worth around $40 million at the time. The current value of the stolen Bitcoin being valued at approximately 150,654,978$. The hack occurred when the hackers were able to gain access to the exchange's API keys, 2FA codes, and other sensitive information.

### *1.1.1 Current State*

| DATE | EXCHANGE | CAUSE OF HACK | AMOUNT STOLEN (USD) |
|---|---|---|---|
| 2022, November 12 | FTX | Unauthorized transactions | $600 million |
| 2022, January 17 | Crypto.com | Unknown | $34 million |
| 2021, December 11 | AscendEX | Obtained access to hot wallet | $80 million |
| 2021, December 5 | BitMart | Obtained access to hot wallet | $150 million |
| 2021, August 19 | Liquid | Obtained access to hot wallet | $97 million |
| 2021, April 29 | Hotbit | Obtained access to hot wallet | Nil |

*figure 1: last year's statistics*

It is apparent that in the past two years, the majority of hacks on crypto exchanges have occurred as a result of the perpetrators gaining unauthorized access to the hot wallets of the exchanges.

This trend highlights the importance of implementing security measures to protect these hot wallets and the need for crypto exchanges to continuously update and improve their security protocols to keep pace with the ever-evolving tactics of hackers.

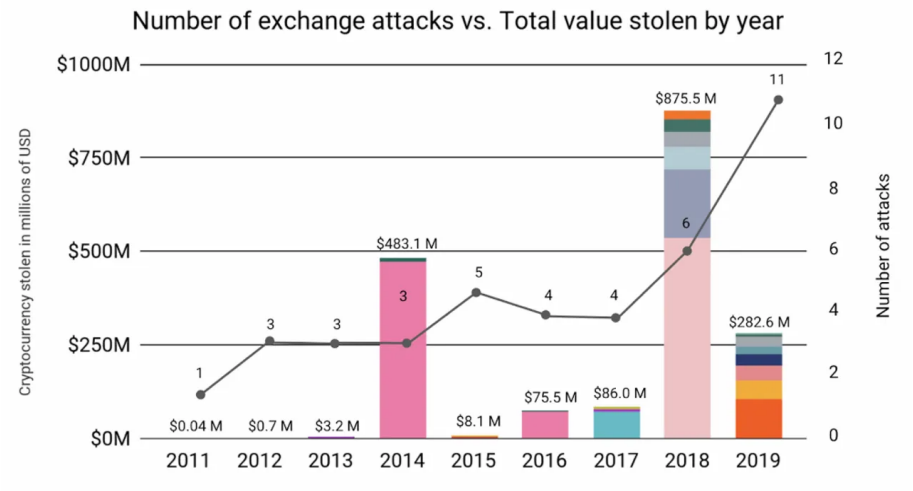



*figure 2: exchanges statistic*

The data shows a trend of an increasing number of attacks occurring annually, indicating a growing vulnerability of digital assets to unauthorized access and theft. This highlights the need for continued efforts to improve security measures to protect against these types of breaches.

# 2 Background

In this section, we will provide an overview of the concepts and technologies that we are using in this project.

## 2.1 Cryptocurrency and Blockchain

### *2.1.1 Cryptocurrency*

Cryptocurrency is a digital currency that uses cryptography for security and is decentralized, meaning it is not controlled by any government or other central authority.

Cryptocurrencies are typically designed to be limited in supply and to operate independently of a central bank, making them immune to government manipulation or interference. They are also often anonymous, allowing users to transact without revealing their identity. Some of the most well-known cryptocurrencies include Bitcoin, Ethereum, and Litecoin.

### *2.1.2 Blockchain*

Blockchain was created by Satoshi Nakamoto in 2008 based on previous work by Stuart Haber, W. Scott Stornetta, and Dave Bayer. It was first introduced as the underlying technology behind the cryptocurrency Bitcoin. It has since been adopted by many other cryptocurrencies and has also been used in a variety of other applications.

Blockchain consists of continuously growing lists of transaction records, called blocks. Blocks are securely linked together using cryptography. Each block contains transaction data, a timestamp, and a cryptographic hash of the previous block. The timestamp proves that the transaction data existed when the block was created. Since each block contains information about the previous block, they form a chain (linked list), with each additional block linking to the ones before it. any attempt to modify a block would require modifying all subsequent blocks in the chain, which would require a significant amount of computing power.
The transparency and immutability of the blockchain allow transactions to be tracked and verified, providing a high level of security and trust.

Blockchains are typically managed by a peer-to-peer (P2P) computer network, whereby two individuals interact directly with each other, without the need for a trusted authority or a central server (third party).

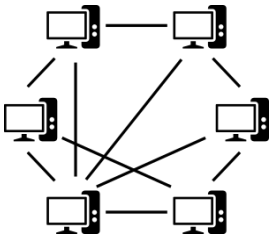


*figure 3: peer to peer*

### *2.1.3 Public Key*

A public key allows you to receive cryptocurrency transactions. While anyone can send transactions to another account, using its public key, you need the private key of your account to prove that you are the owner of the cryptocurrency received in the transaction. You can have any number of public keys connected to a single private key. Unlike the private key, the public key can be shared publicly without compromising the security of the user's funds.

### *2.1.4 Private Key*

A private key gives you the ability to prove ownership or spend the funds associated with your public address. A private key can take many forms: long binary code, hexadecimal code, QR code, mnemonic phrase.

Regardless of its form, a private key is an astronomically large number. While you can generate a public key with a private key, doing the opposite is practically infeasible because of the one-way function. It is important to back up the private key in case it is lost or stolen and never share your private key with anyone. If a private key is lost, stolen, or otherwise compromised, the associated digital assets may be at risk of being accessed or transferred without the owner's consent. It is therefore essential that private keys be kept secure and protected at all times.

### *2.1.5 Transaction*

Transaction is a record of an exchange of value between two or more parties. This can include the transfer of cryptocurrencies, the execution of a smart contract, or the recording of information on the blockchain.

Transactions on a blockchain network are validated and recorded on the distributed ledger using a consensus mechanism, such as proof of work or proof of stake. Once a transaction is added to a block and the block is added to the blockchain, it becomes part of the permanent and tamper-evident record of transactions on the network.

### *2.1.6 Signed Transaction*

In a blockchain, a transaction is signed using a digital signature to ensure its authenticity and integrity. The sender of the transaction creates a digital signature by encrypting a hash of the transaction data with their private key. The transaction, along with the digital signature, is then broadcast to the network. Other nodes on the network can verify the transaction by using the sender's public key to decrypt the digital signature and compare the resulting hash to a hash of the transaction data. If the two hashes match, the transaction is considered to be valid and added to the blockchain. This process helps to ensure that only the owner of a particular private key can create a valid transaction using their associated public key and that the transaction cannot be modified after it has been signed.

### *2.1.7 Submit Transaction to Blockchain*

The process of accepting a transaction by the blockchain involves several steps to ensure that it is properly formatted and follows the rules of the blockchain. First, the transaction is verified to ensure that it is properly formatted and that it follows the rules of the blockchain. Next, it is broadcast to the rest of the network and propagated to other nodes, where it is validated by those nodes to ensure that it is valid and follows the rules of the blockchain. If the transaction is valid, it is then added to a block of transactions, which is broadcast to the network and added to the blockchain by other nodes if it is valid. By following this process, the blockchain can ensure that only valid transactions are added to the blockchain, which helps to maintain the integrity of the network.

### *2.1.8 Cryptocurrency Exchange*

A cryptocurrency exchange is a platform that allows users to buy, sell, and trade digital assets, such as cryptocurrencies. These exchanges operate online and can be accessed through a web browser or a mobile app.

Cryptocurrency exchanges typically support a range of different cryptocurrencies and allow users to exchange them for other cryptocurrencies or fiat currencies, such as the US dollar. Cryptocurrency exchanges use wallets to store the digital assets that are traded on the platform. A wallet is a digital software program that interacts with the blockchain to enable the user to send and receive digital assets.

Exchanges use wallets to securely store the assets of their users and to facilitate the buying, selling, and trading of those assets on the platform. When a user wants to trade an asset, the exchange will use the appropriate wallet to debit the asset from the user's account and credit it to the account of the party they are trading with.

Exchanges may use a single wallet for all assets, or they may use multiple wallets to store different types of assets or to keep the assets of different users separate. Using multiple wallets can provide an additional layer of security and make it easier for the exchange to track and manage the movement of assets on the platform.

## 2.2 Cryptography

### *2.2.1 Public-Key Cryptography*

Public-key Cryptography is a type of cryptography that uses a pair of keys for encryption and decryption. The two keys are known as the public key and the private key. In a public-key encryption system, anyone with a public key can encrypt a message, but only those who know the matching private key can decrypt the ciphertext to receive the original message. This allows for secure communication over an insecure network, such as the internet.

Public key cryptography is essential to many modern technologies, including secure online transactions and secure communication between computers and servers.

### *2.2.2 Digital Signature*

A digital signature is a type of electronic signature that is used to authenticate the identity of the sender of a message, document, or transaction. In the context of cryptocurrency, a digital signature is used to verify the authenticity of a transaction. Every transaction on a blockchain network must be digitally signed by the sender using their private key. The digital signature acts as proof that the transaction was created by the owner of the private key, and ensures that the transaction cannot be altered or tampered with once it has been added to the blockchain. The encryption of the private key to digital signature is asymmetric - which means it can not be decrypted back to the private key.

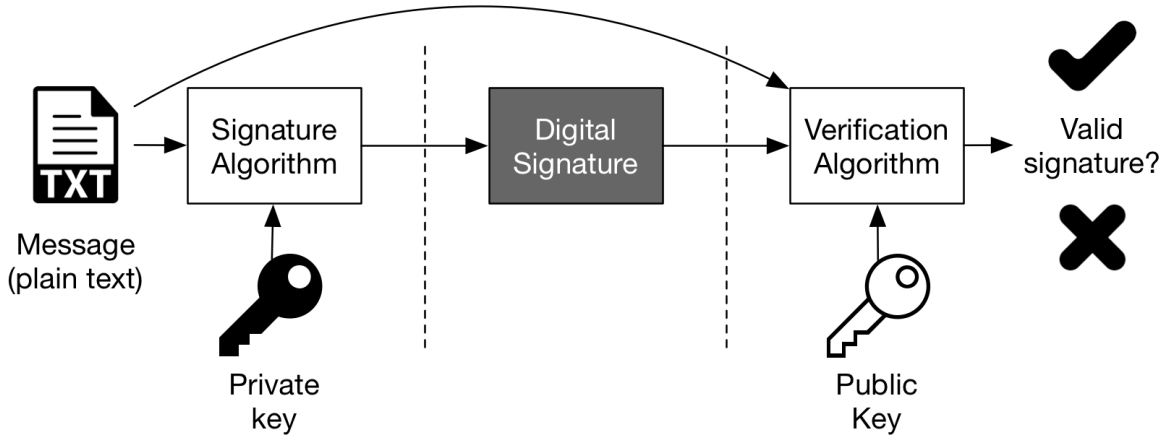



*figure 4: digital signature*

### *2.2.3 Multi-Party Computation (MPC)*

MPC is a type of cryptographic technique that allows multiple parties to jointly compute a function over their inputs, without revealing the inputs to each other. MPC allows multiple parties to securely compute a function over their private data.

### *2.2.4 Symmetric Encryption*

Symmetric encryption is a type of encryption that uses the same key for both encryption and decryption. This means that the sender and the recipient of the encrypted message must both have access to the same key in order to encrypt and decrypt the message.

## 2.3 Previous Researches and Related Work

Previous researches on MPC for private key storage has focused on developing and evaluating new MPC protocols for secure key management. These protocols typically involve the division of the private key into multiple shares, which are distributed among multiple parties. Some research has focused on the use of MPC for secure key storage in blockchain systems, where the private key is used to sign transactions and access digital assets. Other studies have explored the use of MPC in other applications, such as secure cloud computing and secure messaging.

In general, previous research on MPC for private key storage has highlighted the potential of MPC as a tool for secure and efficient key management in distributed settings and has discussed various challenges and future directions for MPC research in this context.

# 3 Expected Achievements

In this part, we will present the expected outcomes of the project.

We will review the main goals:

### 3.1 Explore the Problem State

Private key security is a critical issue in the crypto market, as it directly affects the safety and security of individuals' and organizations' digital assets. It is important to research and carefully consider the best security practices for protecting private keys.
Statistics on private key security in the crypto market can provide valuable insights into the prevalence and impact of key-related issues. For example, data on the number and value of lost or stolen keys can help highlight the importance of secure key management.

### 3.2 Investigate the Existing Approaches to the Problem Solution

Investigating the existing approaches to private key security methods can help organizations better understand the options available for improving the security and resilience of their key management systems, and make informed decisions about which methods are most suitable for their needs.

### 3.3 Web Application

Creating a web app that shows a solution to the private key security problem can be an important tool for demonstrating the effectiveness and value of the solution.
The web app will illustrate the algorithm of securely storing private keys in a distributed manner, without the need for a central authority or trusted third party. The web app will allow users to interact with the system and explore its functionality, providing a visual and user-friendly way to understand and evaluate the solution.

## 4 Engineering process

In Part A of the project, we gained an understanding of the cryptographic and cryptocurrencies field and its relevance to our project. We also addressed the problem of private key security and reviewed various solutions. In Part B of the project, we will design and build a web application to demonstrate the algorithm.

## 4.1 Existing Solutions

### *4.1.1 Cold Wallet*

This involves storing private keys on a device that is not connected to the internet. Cold storage options include storing keys on a paper wallet or on a USB drive that is kept in a secure location.

### *4.1.2 Hot Wallet*

This refers to the practice of keeping the private keys for a wallet online, on servers that are connected to the internet. While this is more convenient for users, as they can easily access their cryptocurrency from anywhere with an internet connection. However, it also makes the keys more vulnerable to hacking and theft. To mitigate this risk, exchanges that use hot storage often implement additional security measures such as multi-factor authentication and regular security audits.

Cryptocurrency exchanges often use a split storage system to keep the private keys for their user's wallets secure. This system involves the use of both hot and cold wallets. The hot wallet is stored on an online server and is used to facilitate quick transactions when users withdraw funds from the exchange. It typically contains a relatively small amount of cryptocurrency, as determined by the exchange operator.

It is likely that the small amount referred to is significantly larger than anticipated, as demonstrated by the trade volumes depicted in the chart for the period of January 1st to 8th, 2023 on the Binance exchange.

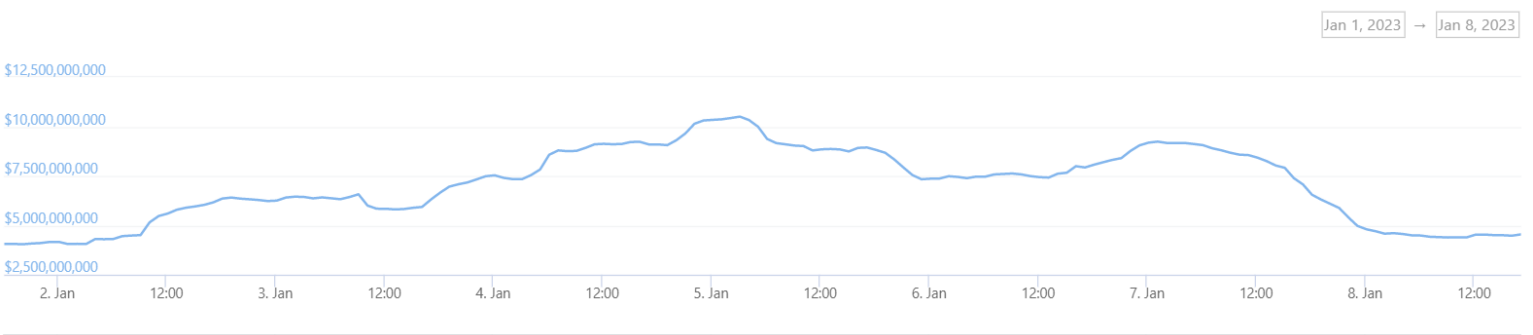



*figure 5: Binance exchange volume graph*

In contrast, the cold wallet is a physical device that is not connected to the internet and stores the private keys for the majority of an exchange's cryptocurrency assets. Cold wallets are considered to be more secure than hot wallets because they are not connected to the internet and are therefore less vulnerable to hacking and theft. Exchanges may transfer funds between the hot and cold wallets as needed, such as when the hot wallet has a surplus or deficiency of cryptocurrency.

It is important to note that when an individual stores their cryptocurrency on an exchange, they do not possess private keys for their assets. Instead, they are reliant on the exchange's promise to provide access to their coins under certain conditions.

## 4.2 The Process

In the initial phase of our research process, we defined the research question that we sought to address through our work. This involved identifying the main problem that we wanted to address and formulating a specific research question.

Subsequent to this initial phase, we conducted a literature review in order to gain a deeper understanding of the relevant fields of study. This involved reading and reviewing academic articles and other materials written by experts in these fields, in order to gain a comprehensive understanding of the current state.

Following the initial phase of our research process, we defined the specific requirements. This involved identifying the specific goals and performance criteria that the algorithm needed to meet, such as the required level of security and scalability.

Based on these requirements, we designed the overall structure and functionality of the algorithm, including the specific protocols that would be used and UML diagrams.

As part of the algorithm development process, we conducted extensive research on the management and storage of private keys, as well as various cryptography methods.

Once the design of the algorithm was complete, we developed a testing plan in order to evaluate its performance and functionality. This may have involved identifying specific test cases and test data to be used, and outlining the specific methods and techniques that would be used to evaluate the algorithm.

The algorithm can provide improved security compared to traditional cryptographic algorithms. This is because it relies on the cooperation of multiple parties, and does not rely on the assumption that a single party is trustworthy.

One of the main advantages of the algorithm is that it increases the difficulty for hackers to access the keys. This is because, in the system, the private key is divided into several smaller pieces, which are stored on separate servers. To access the private key, the parties must be combined and decrypted using a specific process. As a result, a hacker would need to attack multiple parties across different systems and locations in order to access the private key, rather than just targeting a single server or individual.

Even if an attacker were able to gain access to one of the validators, they would only be able to access a portion of the private key and would not have the full key needed to access the assets. Additionally, by storing each character of the private key twice in two different validators, the system is able to protect against data loss or corruption. If one of the servers fails or is compromised, the private key can still be reconstructed from the remaining validators. It also has the potential to improve scalability. The algorithm can be designed to scale up to a large number of servers, making it suitable for use in large-scale systems. The more parties involved, the more secure the system is.

### *4.2.1 The Challenges*

In the research process, three primary challenges were encountered, specifically in relation to secure channel communication, single component failure, and synchronization. To address these challenges, solutions were integrated into the algorithm. These solutions include the implementation of encryption, the distribution of servers across various locations and cloud providers, and the adoption of the Master-slave replication method.

The establishment of secure channels of communication is to mitigate the risk of man-in-the-middle attacks. The attacker could intercept the communication between the servers and modify the private key or the way it is divided and distributed. This would allow the attacker to gain access to the private key and compromise the assets associated with it, despite the algorithm's security measures. To mitigate the risk of these attacks, we implement additional security measures - symmetric encryption and SSH secure channel. To ensure the integrity and confidentiality of the communication channel between the servers.

Another potential vulnerability occurs when a system has a single component that if it fails, it will cause the entire system to fail. It would occur if all the servers that store the private key are located in the same location or hosted by the same cloud provider. If the location or cloud provider is compromised, an attacker could access all the servers and steal the private key, compromising the assets associated with it. To mitigate the risk it is important to distribute the servers across multiple locations and cloud providers.
Synchronization is another challenge that needs to be addressed. The servers need to be kept in sync with each other in order to ensure that the private key can be reconstructed when needed. We plan to use the Master-slave replication method in which the primary server is tasked with managing and distributing the private key, while the secondary servers are responsible for maintaining the key partitions.

### *4.2.2 Stakeholders*

The algorithm is useful for a variety of different stakeholders in a blockchain system, including cryptocurrency exchanges, blockchain developers, and end users of blockchain systems.
Cryptocurrency exchanges often hold large amounts of assets on behalf of their users and need to secure the private keys associated with these assets. The algorithm can provide a secure and scalable way to manage these keys, without the need for a central authority or trusted third party. This can help protect the assets of the users and ensure the integrity of the exchange's operations.
Blockchain developers can also benefit from using it in their applications. Developers can provide strong security for private keys and other sensitive data, while still enabling the decentralized nature of the blockchain. This can help developers build more secure and scalable blockchain applications, and make it easier for users to trust and adopt these applications.
End users of blockchain systems can also benefit from it by having a secure and scalable way to manage their private keys, users can protect their assets and ensure the integrity of their transactions. This can be especially important for users that handle large amounts of assets or transactions.

Overall, The algorithm is a valuable tool for stakeholders in blockchain systems and can provide improved security and scalability for key management in these systems.

### *4.2.3 Market Analysis*

In our research on how individuals and organizations secure their private keys, we identified a significant issue with the prevalence of hacking attacks involving private keys. As a result, we developed an algorithm with the goal of mitigating the financial impact of such attacks. The design of this algorithm was informed by our findings on existing practices for private key security.

# 5 Product

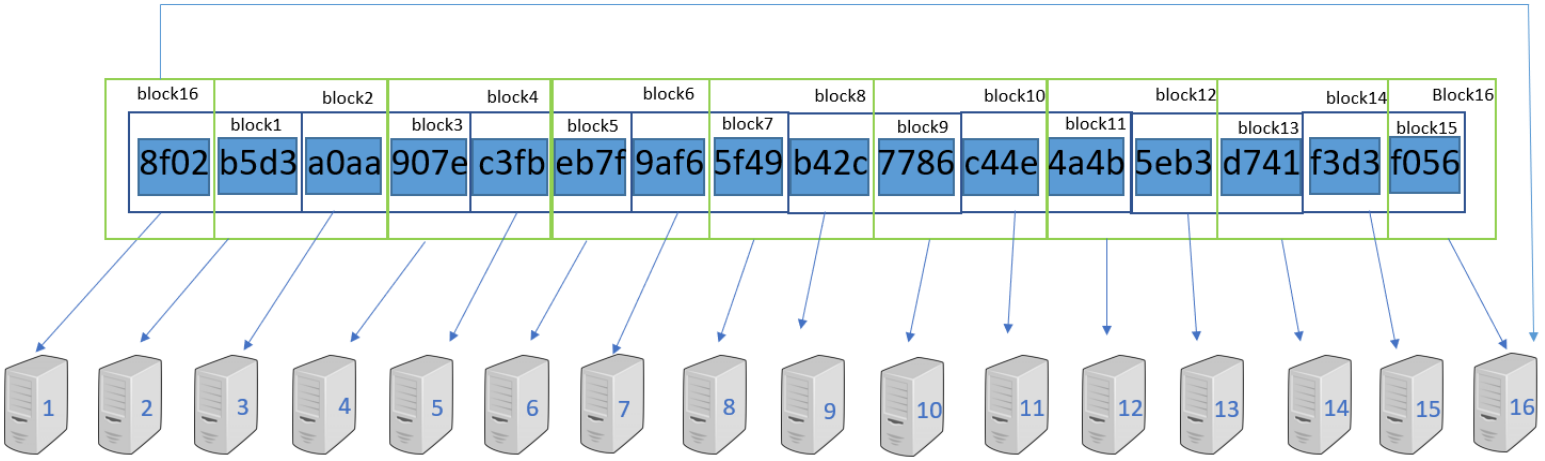
## 5.1 Proposed Solution

We have discussed the main problems of private key security. and the consequences of revealing or losing the private key. In this part, we will present a solution that may help cryptocurrency exchanges and individuals as well.
Our proposed solution is a way to divide and distribute a private key across multiple servers for improved security. It contains several components and includes the following steps to implement:

1. The following algorithm is suitable for 16 servers, although it could be easily managed for more or fewer servers, according to security needs.
2. The 64-character (256-bit) private key is divided into 16 smaller parts of 4-character. These smaller parts are stored in 8-character blocks, with each block containing 2 of the smaller parts. The first block, containing the first 8-character of the private key, is stored in the first server. The second block, containing the last 4-character from the first block and next 4-character of the private key, is stored in the second server. This process continues until all 16 smaller parts are stored in 8-character blocks and distributed across 16 servers. This ensures that each bit of the private key is stored twice in two different servers, adding an extra layer of security to the key. In case one of our servers fails we can still access the private key. By storing the private key this way, a potential hacker will have to hack all 16 servers in order to access the private

key and compromise the assets associated with it, which becomes much more difficult.

Example of the divided and stored private key:




3. Each part will be encrypted using symmetric encryption and sent to a different server.
4. To improve security, the private key can be divided into several parts and stored on different servers located in different locations and companies, such as AWS and Google's servers. For example, servers 1 to 8 could store parts of the private key on AWS servers, while servers 9 to 16 could store other parts of the key on Google's servers. This ensures that if one company's servers are compromised, the attacker would still not have access to the entire private key.
5. Upon completion of a transaction, a new random key will be employed to encrypt each partition of the divided private key.

Upon initiating a withdrawal, the main computer initiates a request to each of the servers to retrieve their respective portions of the private key. Once receiving responses from all servers, the individual portions of the private key are combined to form a composite key. This composite key is then decrypted using symmetric decryption to reveal the true private key, which is then utilized to sign the transaction. The signed transaction is then transmitted.

Subsequent to the transmission of the transaction, the private key is re-encrypted using a different random key and symmetric encryption.

### *5.1.1 Cryptography*

The symmetric cryptography chosen for the encryption of the private key partitions is the BlowFish algorithm.

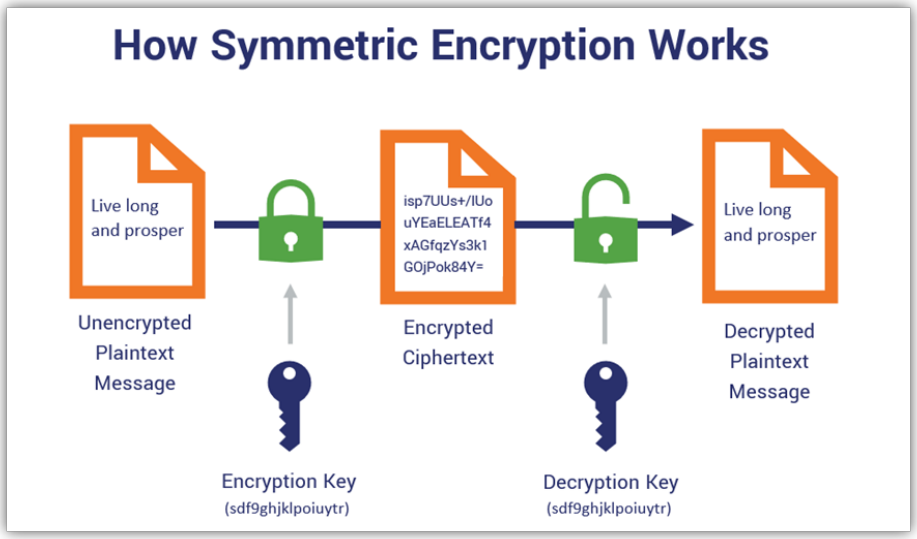



*figure 6: symmetric encryption*

In our design, the private key serves as the plaintext and is encrypted using a randomly initialized key, with this process being repeated for each transaction. This approach allows for the dynamic re-encryption of the private key using a unique key for each transaction, enhancing security by making it more difficult for potential attackers to gain access to the key. Following the re-encryption process, the key will be divided into separate parts and transmitted to the same servers for storage. This approach allows for the secure distribution of the key among multiple servers, enhancing the overall security of the key.

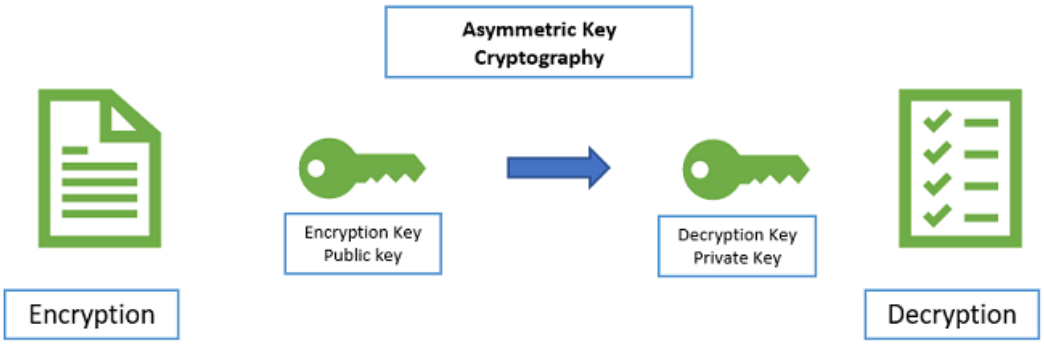



*figure 7: asymmetric encryption*

Asymmetric encryption is used in blockchain to make transactions. The public key is used to encrypt the data, while the private key is used to decrypt it. This allows the sender to encrypt the data using the recipient's public key, while only the recipient has access to the corresponding private key needed to decrypt the data.
Encrypting transactions of blockchain helps to protect the confidentiality of the data being transmitted and ensures that it can only be accessed by authorized parties. It also helps to prevent tampering with the data, as any attempt to alter the encrypted data would result in the decryption process failing. This helps to maintain the integrity of the data and ensures that the blockchain remains a reliable and trusted source of information.

In the algorithm, we propose to use the SSH protocol to establish secure communication channels between the main server and the validators. To accomplish this, we will leverage both symmetric and asymmetric encryption techniques to encrypt the private key parties during transmission. Additionally, we will employ digital signature and authentication mechanisms, such as SSH key-based authentication, to verify the identity of the servers and ensure that the private key parties are only received from authorized sources. This combination of encryption and authentication techniques will provide a high level of security for the private key storage system, and it will help to protect against potential threats such as man-in-the-middle attacks.

## 5.2 Requirements

### *5.2.1 Functional Requirements*

1. The system should be capable of dividing the private key.
2. The system should be able to encrypt the individual parties.
3. The system should be able to send the encrypted private key parties to specific servers.
4. The system should be able to enumerate the different servers.
5. The system should identify the different validators.
6. The system should identify the part number.
7. The system should be able to assign unique identifiers to each validator.
8. The system should be able to identify the specific parties of the private key.
9. The system should be able to assign unique identifiers to each party.
10. The system should use encrypted communication channels.
11. The system should provide the option to restore the private key.
12. The system should be able to utilize the private key.
13. The system should be able to sign blockchain transactions.
14. The system should be able to transmit signed transactions to blockchain nodes.
15. The system should be able to scale up to support a large number of servers.
16. The system must be able to ensure the integrity of the data.

### *5.2.2 Non-Functional Requirements*

1. The private key should be divided into multiple parties.
2. each party contains a partial representation of the private key.
3. The encryption using Blowfish cryptographic algorithm.
4. restore the private key without requiring access to all parties.
5. Sign blockchain transactions using the private key.
6. The system must be easy for users to interact with.

## 5.3 Used Technology

### *5.3.1 Web3 and Ethers Libraries*

web3 and ethers are collections of libraries, which allow interaction with a local or remote Ethereum node, using HTTP. It is a JavaScript library that allows developers to interact with the Ethereum blockchain through their web applications.

Ethers.js is more lightweight and developer-friendly. Ethers.js is compatible with both Node.js and web browsers.

Ethers.js is also a more powerful and flexible library, it has a lot of functionalities that are missing from web3.js. it also has a lot of built-in functionalities like signing, contract deployment, contract interactions, and more.

We will use Ethers.js in order to create and sign the transaction and send it to the blockchain nodes.

### *5.3.2 Infura API*

Infura is a web3 infrastructure provider that offers API access to decentralized protocols, including the Ethereum blockchain. By using Infura, developers are able to access blockchain networks and build applications without the need to operate their own full nodes.

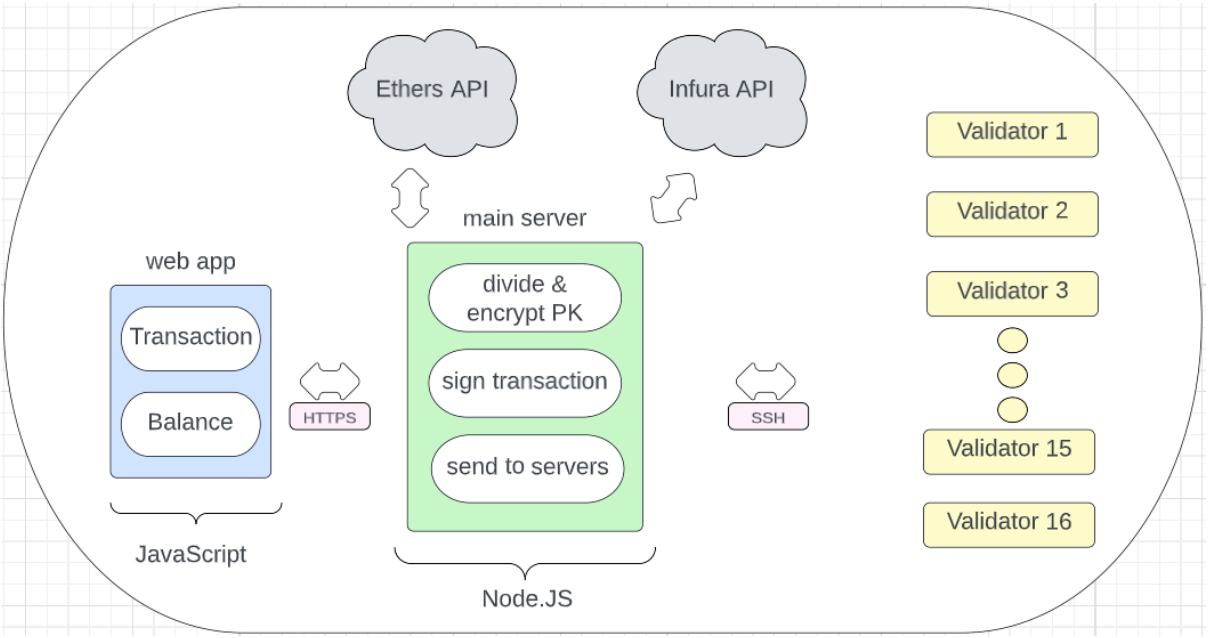
In this project, we plan to utilize Infura as a platform for communicating with blockchain nodes. Its API will allow us to communicate with the blockchain nodes without having to host the nodes on our own server, which can be resource-intensive.

### *5.3.3 SSH*

SSH (Secure Shell) is a protocol that provides secure encrypted communications between two computers over an insecure network. It provides strong authentication and allows us to execute commands and move data from one computer to another in a secure way.

We will use the SSH protocol in order to make secure communication between the main server and the validators. This protocol can help us to significantly reduce the chances of a cyber attack, that in these attacks there can be an interference in communication or that the data can be stolen.

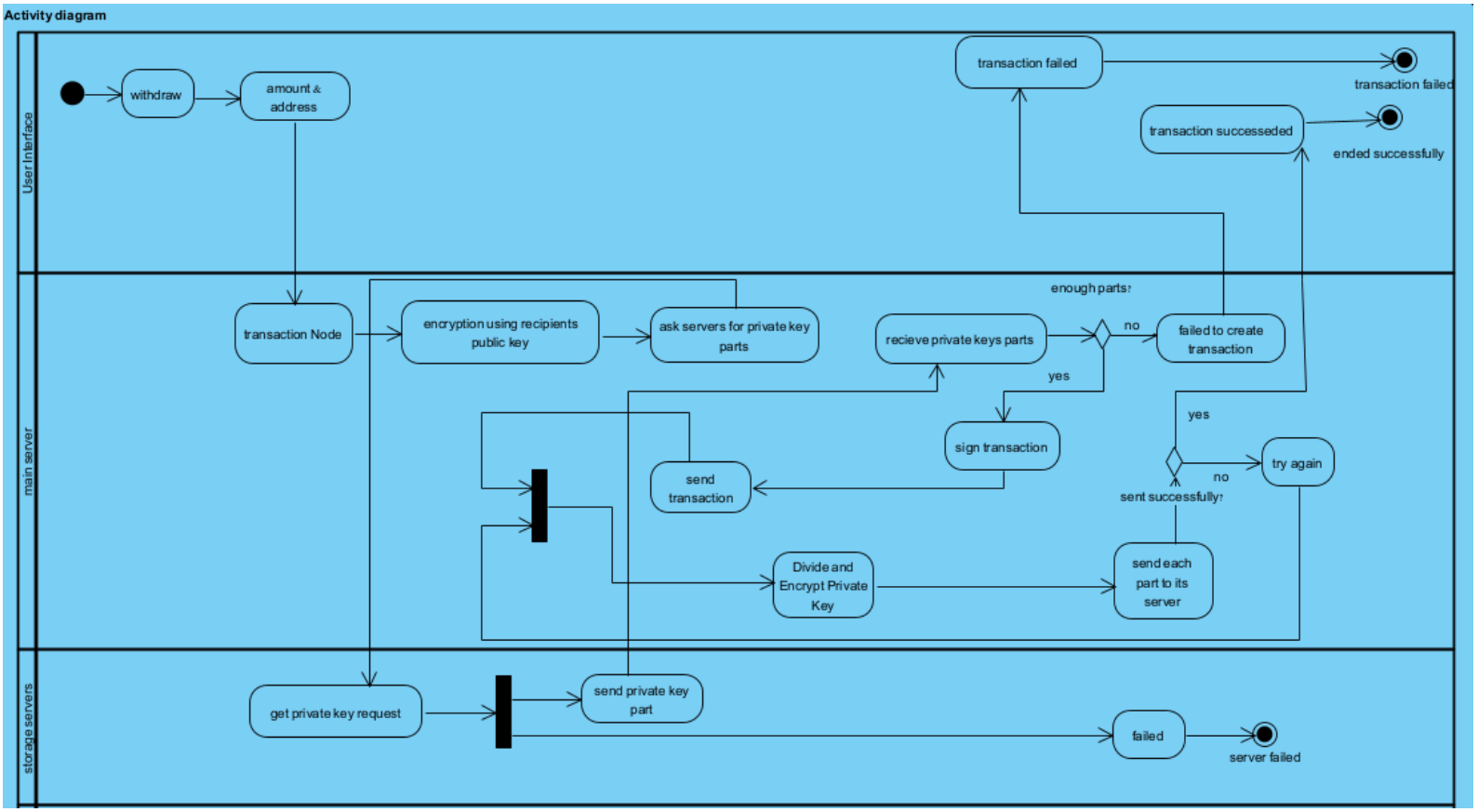
## 5.4 Architecture Diagram




In the architecture diagram, we can observe the web app which serves as the user interface, allowing users to interact with the system. The main server, which is responsible for executing the logic of the algorithm, is also depicted. In order to establish a secure communication channel and protect the confidentiality of private keys, we used symmetric encryption algorithms. The remaining servers are utilized solely for storage purposes.

The web application to be implemented in the deployment will be that of a cryptocurrency exchange. It should be noted that this web application is being developed solely for demonstration purposes.

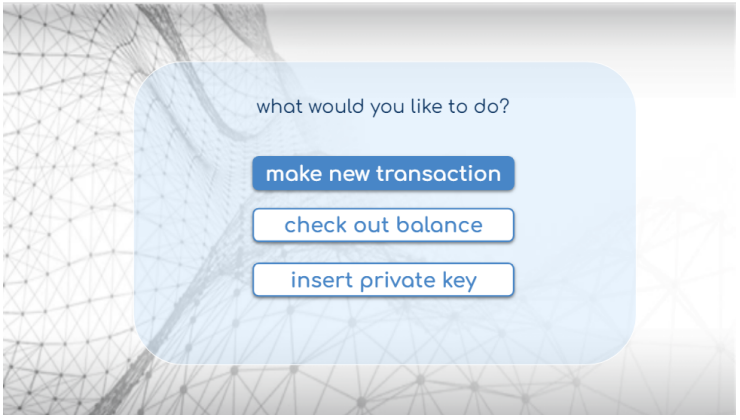
## 5.5 Activity Diagram

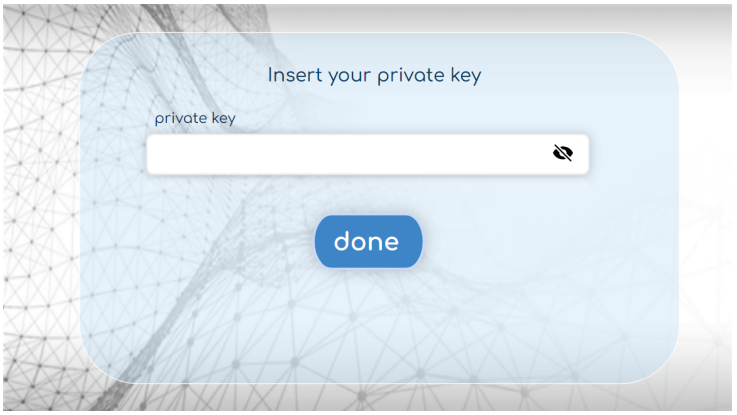


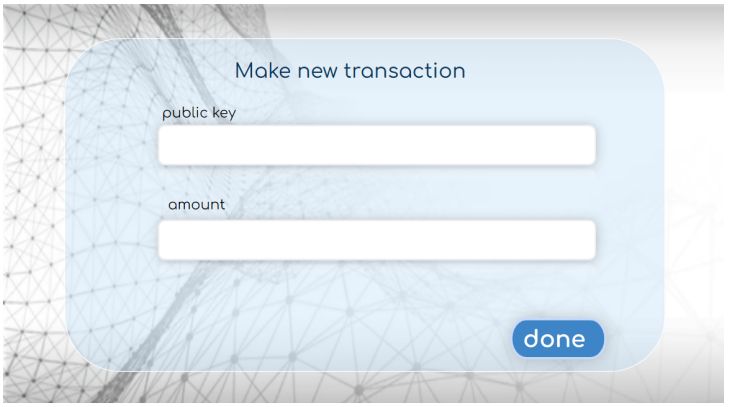

Sequence:

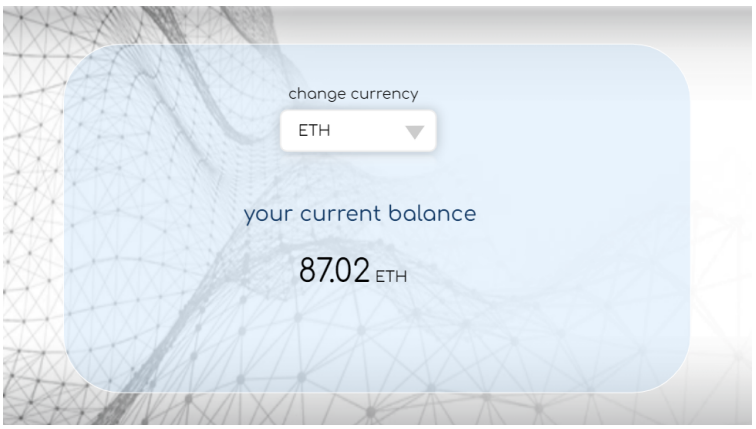
- User initiates the withdrawal by interacting with the application
- The app sends a request to the main server to retrieve the private key
- Main computer sends requests to all validators to retrieve their respective portions of the private key
- validators respond with their portions of the private key
- Main server combines and decrypts the individual portions to reveal the true private key.
- Private key is used to sign the transaction.
- Transaction is transmitted to the network

## 5.6 Web Screens



## 6 Evaluation/Verification Plan

The web application will serve as a test platform to evaluate the functionality and effectiveness of the proposed algorithm. It will allow us to simulate real-world scenarios and test the algorithm's ability to handle different types of transactions, security breaches, and other potential issues. This allows us to identify any bugs, weaknesses, or areas for improvement, and make any necessary adjustments before implementing the algorithm in a live environment. Additionally, it will enable us to demonstrate how the algorithm works and explain how it improves the security of the system. In our evaluation of the system, we plan to assess the system's ability to create valid transactions. This will include generating test transactions using a variety of different inputs and configurations, and verifying that they are correctly signed and accepted by the blockchain system. To do this, we will use a test Ethereum blockchain environment and create test transactions using the system. Once the transactions have been submitted, we will verify that they are correctly signed and accepted by the blockchain. This will involve checking that the transactions are properly formatted and that they include the correct signatures, as well as verifying that the transactions are recorded on the blockchain as expected. By testing the system's ability to create valid transactions, we can ensure that the private key is functioning correctly and that the system is able to sign transactions as expected. This is an important step in the evaluation process.

some of the test cases we would implement:

| Test number | Test case | Expected result |
|---|---|---|
| 1 | generating new private key | valid private key |
| 2 | Dividing the private key | division should be according to the algorithm |
| 3 | Encrypting and decrypting the private key parties | each part should be encrypted |
| 4 | Sending the private key parties to the correct servers | each server should have its part |
| 5 | making a new transaction | the transaction is accepted in blockchain |
| 6 | making a new transaction in case one of the servers if down | the transaction is accepted in blockchain |

# 7 References


1. cryptopedia website (2022): https://www.gemini.com/cryptopedia/public-private-keys-cryptography
2. Daniel Escudero (2022) “An Introduction to Secret-Sharing-Based Secure Multiparty Computation”
3. Giancarlo Giudici, Alistair Milne, Dmitri Vinogradov (2019) “Cryptocurrencies: market analysis and perspectives”
4. Peter M. Krafft, Nicolás Della Penna, Alex “Sandy” Pentland (2018) “An Experimental Study of Cryptocurrency Market Dynamics”
5. Satoshi Nakamoto (2008) “Bitcoin: A Peer-to-Peer Electronic Cash System “
6. Sinergija University, Faculty of computing and informatics, Bijeljina, BiH (2019) “COMPARATIVE ANALYSIS OF CRYPTOCURRENCY WALLETS VS TRADITIONAL WALLETS “
7. Usman W. Chohan, MBA, PhD (2022) “The Problems of Cryptocurrency Thefts and Exchange Shutdowns”
8. Magazine by coin telegraph website: https://cointelegraph.com/magazine/crypto-exchange-hacks/


figures:

figure 1:https://www.hedgewithcrypto.com/cryptocurrency-exchange-hacks/

figure 2: https://www.zdnet.com/article/2019-saw-more-cryptocurrency-hacks-than-any-other-year/

figure 3: https://en.wikipedia.org/wiki/Peer-to-peer

figure 4: https://stakey.club/en/verifying-digital-signatures/

figure 5: https://www.coingecko.com/en/exchanges/binance#statistics

figure 6: https://securityboulevard.com/2020/11/symmetric-encryption-algorithms-live-long-encrypt/

figure 7: https://cheapsslsecurity.com/blog/what-is-asymmetric-encryption-understand-with-simple-examples/