Block Chain Based Open Source Malware Detection System

A project report

submitted in partial fulfillment of the requirements

for the degree of

Bachelor of Technology

in

Computer Science & Engineering

by

Adrit Kunar Bose

(Roll No: 1902840100013)

Under the Guidance of

Dr. Amit Kumar Tiwari

(Assistant Professor)

Department of Computer Science & Engineering

UNITED INSTITUTE OF TECHNOLOGY PRAYAGRAJ

Uttar Pradesh 211010, INDIA.

(Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow)

2022-2023

Declaration of Academic Ethics

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I declare that I have properly and accurately acknowledged all sources used in the production of this project report.

I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Date: April, 2023 (Adrit Kumar Bose)

Certificate from the project guide

This is to certify that the work incorporated in the project report entitled ***"Block Chain Based Open Source Malware Detection System "*** is a record of work carried out by ***Adrit Kumar Bose(19028409100013)*** Under my guidance and supervision for the award of Degree of Bachelor of Technology in Computer Science & Engineering.
To the best of my/our knowledge and belief the project report

1. Embodies the work of the candidates themselves,
2. Has duly been completed,
3. Fulfils the requirement of the Ordinance relating to the Bachelor of Technology degree of the University and
4. Is up to the desired standard both in respect of contents and language for being referred to the examiners.

Date: April, 2023 (Dr. Amit Kumar Tiwari)

The project work as mentioned above is here by being recommended and Forwarded for examination and evaluation.

Date: April, 2023 Head of Department

Acknowledgments

I would like to thank **Dr. Amit Kumar Tiwari sir** and **Head of Departement** for giving us the honour of being a guide to my project **Blockchain based open source malware detection system** . I would like to express my deep sense of gratitude to my friends for their important help and technical suggestions. I am grateful to Prashant Kumar Dey for introducing me with this concept. I would like to thank all members of CSE lab who have directly or indirectly contributed in my project work and maintained a friendly atmosphere in the lab. Finally I would like to thank my family and my friends for their constant moral support.

Date:April, 2023
Place: UIT Prayagraj

Adrit Kumar Bose
(Name of Student/s)

Abstract

The security and integrity of computer systems and networks are seriously threatened by the ubiquity and sophistication of malware, which is on the rise. The inability of conventional malware detection approaches to keep up with new malware developments frequently results in vulnerabilities and breaches. Blockchain technology has become a potential option for many applications, including cybersecurity, in recent years. In order to increase cybersecurity overall and improve detection capabilities, this study suggests a unique blockchain-based malware detection system that makes use of the distributed nature of blockchain.

The suggested method uses the blockchain's openness, immutability, and decentralisation properties to build a safe and impenetrable environment for virus detection. The technology creates a network of nodes that jointly take part in the discovery process by merging blockchain and malware detection algorithms. With each node maintaining a copy of the blockchain, redundancy is ensured and single points of failure are avoided.

The detection process begins with the submission of suspicious files or network traffic to the blockchain network. The nodes collaborate to validate and analyze the submitted data using advanced malware detection techniques, such as signature-based scanning, behavior analysis, machine learning, and artificial intelligence algorithms. The detection results, along with relevant metadata, are recorded in blocks and added to the blockchain, ensuring transparency and auditability.

Through consensus mechanisms, such as proof-of-work or proof-of-stake, the system guarantees the integrity and reliability of the detection results. Moreover, the blockchain-based architecture enables secure sharing of threat intelligence across multiple organizations, fostering collaboration and collective defense against malware attacks.

The proposed system offers several advantages over traditional malware detection approaches. It provides a decentralized infrastructure that is resistant to single points of failure and tampering attempts. The transparent and immutable nature of the blockchain enhances the trustworthiness and verifiability of the detection process. Furthermore, the shared threat intelligence allows for the rapid identification and response to emerging malware threats.

This research concludes by presenting a blockchain-based malware detection system that takes advantage of the capabilities of distributed ledger technology to overcome the drawbacks of conventional malware detection. The suggested system provides improved security, transparency, and cooperation features, making it a potential alternative for fending off malware threats that are constantly changing in the digital sphere. Future studies can concentrate on enhancing the system's functionality, scalability, and compatibility with current cybersecurity infrastructure.

Contents

Chapter 1

Introduction

The project described is focused on the development of a blockchain-based system that can effectively detect and catalog viruses, malware, and other related threats. The system will provide a comprehensive and detailed view of the encountered threats, utilizing the immutable and distributive properties of the blockchain technology to ensure data safety and security. The open-source nature of the project aims to make this information available and accessible to as many people as possible.

The project's main objective is to create a public blockchain (permission-based) that can effectively store and manage data related to various types of malware and viruses. This will involve the creation of a detailed database containing information on the nature, origin, and effects of different types of threats.

As a public blockchain, any user can read the information stored in the database. However, only authorized users, such as antivirus and related companies, will have the ability to validate the entered data's authenticity. This means that the system will require real-time updates to keep the project functioning effectively.

The key advantage of using blockchain technology in this project is its immutability and distributive nature. This means that once the data has been entered into the blockchain, it cannot be altered, ensuring the authenticity and reliability of the information. The distributive nature of the blockchain also means that the system is resistant to attacks and failures, making it a robust and secure solution for storing and managing malware-related data.

Overall, the project's aim is to create an effective and comprehensive solution to combat malware and related threats. By utilizing the power of blockchain technology, the project aims to provide an innovative and secure solution to an increasingly prevalent problem in the digital world

In addition to the benefits mentioned, using blockchain technology in this project also provides transparency and accountability. Each entry in the blockchain will be timestamped and linked to previous entries, creating an unbreakable chain of information. This means that every change made to the database is recorded and cannot be altered or deleted. This feature can be helpful in identifying the source of malware and its origin, which can be useful in preventing future attacks.

Moreover, blockchain technology can enable data sharing and collaboration between different entities, such as antivirus companies and researchers, allowing them to work together to create a more comprehensive and accurate database of malware and related threats.

The open-source nature of the project makes it accessible to anyone interested in contributing to the database's growth and development. This can help build a community of developers, researchers, and cybersecurity experts working together to create a better understanding of malware and its effects.

To ensure the project's real-time updation and validation, the system will require a network of authorized nodes that can verify the authenticity of the data entered. This can be achieved through a consensus mechanism that requires multiple nodes to validate each entry before it is added to the blockchain. This ensures the accuracy and reliability of the data and helps prevent malicious entries from being added to the database.

Finally, the system can also utilize smart contracts to automate certain functions and ensure the timely execution of certain tasks. For example, a smart contract could automatically trigger an alert to all authorized nodes when a new type of malware is detected, ensuring that everyone is aware of the latest threat and can take appropriate measures to prevent its spread.

In summary, the blockchain-based malware detection system project aims to create a comprehensive and reliable database of malware and related threats. By utilizing the unique properties of blockchain technology, the system can provide transparency, accountability, and security, while also fostering collaboration and community involvement.

1.0.1 History

Blockchain has the potential to grow to be a bedrock of the worldwide record-keeping systems, but was launched just 10 years ago. It was created by the unknown persons behind the online cash currency bitcoin, under the pseudonym of Satoshi Nakamoto.

1. 1991

   A cryptographically secured chain of blocks is described for the first time by Stuart Haber and W Scott Stornetta

2. 1998

   Computer scientist Nick Szabo works on ‘bit gold’, a decentralised digital currency

3. 2000

   Stefan Konst publishes his theory of cryptographic secured chains, plus ideas for implementation

4. 2008

   Developer(s) working under the pseudonym Satoshi Nakamoto release a white paper establishing the model for a blockchain

5. 2009

   Nakamoto implements the first blockchain as the public ledger for transactions made using bitcoin

6. 2014

   Blockchain technology is separated from the currency and its potential for other financial, interorganisational transactions is explored. Blockchain 2.0 is born, referring to applications beyond currency

The Ethereum blockchain system introduces computer programs into the blocks, representing financial instruments such as bonds. These become known as smart contracts.

The second generation

Other blockchains include those that run the several hundred “altcoins” – other similar currency projects with different rules – as well as truly different applications, such as:

Ethereum: the second largest blockchain implementation after bitcoin. Ethereum distributes a currency called ether, but also allows for the storage and operation of computer code, allowing for smart contracts

1.0.2 Problem statement

The increasing reliance on digital technology and the internet has made cybersecurity a critical concern for individuals, businesses, and governments worldwide. Cybersecurity refers to the practice of protecting computer systems, networks, and sensitive information from unauthorized access, theft, damage, or disruption.

One of the biggest challenges in cybersecurity is the availability of unidentified threats and malware information. Cyber threats can take various forms, including viruses, worms, Trojan horses, spyware, ransomware, and other malicious software. Cybercriminals often use sophisticated techniques to evade detection and remain undetected for long periods.

When a system is infected with malware, it can result in unexpected server or system failures, leading to downtime, data loss, and financial losses. Malware can cause significant harm to the infected systems by stealing sensitive information, corrupting files, or rendering the system unusable.

The unavailability of unidentified threat and malware information makes it difficult for cybersecurity experts to develop effective defense mechanisms to prevent cyber attacks. Traditional cybersecurity systems often rely on signature-based detection techniques that require prior knowledge of the malware's signature. However, this approach is becoming less effective as cybercriminals continuously develop new techniques to evade detection.

Therefore, the need for a more robust and reliable cybersecurity system that can detect and prevent unidentified threats is becoming increasingly critical. Blockchain-based malware detection systems offer a promising solution by providing a decentralized, secure, and immutable system that can store information about various malware signatures.

By using blockchain technology, the system can ensure that the data remains secure and accessible to authorized parties while preventing unauthorized access, tampering, or deletion. This approach allows cybersecurity experts to access real-time information about malware signatures, enhancing their ability to develop effective defense mechanisms and prevent cyber attacks.

1.0.3 Solution

The proposed solution to the problem of unidentified threats and malware information in cybersecurity is the development of a decentralized blockchain network that can store and update malware signatures in real-time. This system aims to provide a reliable and secure method of storing and sharing malware information, allowing cybersecurity experts to access the latest information about new and emerging threats.

The system will work by creating a blockchain network that will contain the signatures of each malware encountered. These signatures will include all the required technologies to resolve that threat, including information about the malware's behavior, propagation techniques, and recommended solutions.

As new threats are discovered, the blockchain network will be updated with new blocks containing the latest information about the malware signatures. The decentralized nature of the blockchain network ensures that the system remains resilient against attacks and tampering, making it difficult for cybercriminals to modify the information or delete it.

The use of blockchain technology ensures that the information stored in the network remains secure and transparent, as all transactions on the blockchain are verified and recorded on a public ledger. Authorized users can access the network and validate the authenticity of the entered data.

As the blockchain network continues to evolve and new blocks are added, the system becomes more robust and resistant to attacks. This system's decentralized nature ensures that there is no single point of failure, making it more secure and resilient than traditional centralized systems.

In summary, the proposed solution of developing a decentralized blockchain network that stores malware signatures and updates in real-time offers a promising solution to the problem of unidentified threats and malware information. The system's decentralized nature ensures that the information is secure, transparent, and resilient, making it an effective tool in the fight against cyber threats.

1.0.4 Why Blockchain

The Block chain not only store the signature and details of each and every Malware but along with that it will keep increasing its blocks as the list of these threats increaing day by day. And along with all that as per as the alogorithum of the blockchain i.e, the tampering or hacking of the data isnt possible with blockchain, it will be a great asset fr the storage of all the problems alongwith there solution

1.0.5 Current Scenario –

The current approach of antivirus and security systems working individually poses a significant challenge in the identification and mitigation of cyber threats. When an organization identifies a threat, they tend to keep the information to themselves without sharing it with others. This siloed approach limits the effectiveness of cybersecurity efforts, as other organizations remain vulnerable to the same threat.

This project proposes a solution to this problem by promoting a mutual relationship between different organizations in the cybersecurity industry. By creating a decentralized blockchain network that allows for the sharing of information about identified threats, a comprehensive list of all threats and malware can be developed.

The blockchain network will enable different organizations to share information about identified threats, allowing for faster and more effective response times. The list of threats will be constantly updated in real-time as new threats are identified, making it a comprehensive resource for the cybersecurity industry.

The use of blockchain technology ensures that the list is tamper-proof and resistant to attacks, as each user of the blockchain is a server in themselves, making it nearly impossible to tamper with the data. This feature of the blockchain network ensures that the database remains secure, transparent, and resilient.

In summary, promoting a mutual relationship between different organizations in the cybersecurity industry and developing a decentralized blockchain network to share information about identified threats offers a promising solution to the current challenge of siloed cybersecurity efforts. The blockchain network's decentralized nature ensures that the database remains secure, transparent, and resilient, making it an effective tool in the fight against cyber threats.

Chapter 2

SDLC Model

SDLC stands for Software Development Life Cycle, which is a process used by software development teams to design, develop, and test high-quality software. There are several SDLC models, each with its own set of strengths and weaknesses. Here are some of the most popular SDLC models:

Waterfall Model: This model is the most traditional SDLC model, and it works best when the requirements are clear and fixed. It proceeds in a linear, sequential manner, with each phase dependent on the previous one. This model is rigid and inflexible, and it's difficult to incorporate changes once a phase has been completed.

Agile Model: This model is designed to be more flexible than the Waterfall model. It focuses on delivering working software quickly and making changes as needed. This model is particularly useful when requirements are uncertain or subject to change.

Spiral Model: This model combines elements of both the Waterfall and Agile models. It allows for more flexibility and is particularly useful for large and complex projects. The Spiral model involves risk analysis and prototyping in each phase, making it more adaptable to changing requirements.

Iterative Model: This model involves breaking a project into smaller pieces, and each piece is developed in an iterative manner. This model is particularly useful for projects with uncertain requirements, as it allows for testing and feedback throughout the development process.

V-Model: This model is an extension of the Waterfall model and emphasizes testing at every stage of the process. Each phase in the development process has a corresponding testing phase, ensuring that the software is thoroughly tested before release.

DevOps Model: This model involves the integration of development and operations teams to ensure faster and more efficient delivery of software. It emphasizes collaboration and automation, with a focus on continuous delivery and continuous deployment.

Each SDLC model has its own set of advantages and disadvantages, and the choice of which model to use will depend on the project's specific requirements and constraints

2.0.1 Waterfall Model

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In ”The Waterfall” approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially. The following illustration is a representation of the different phases of the Waterfall Model. The sequential phases in Waterfall model are • Requirement Gathering and analysis All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document. • System Design The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture. • Implementation With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its func- tionality, which is referred to as Unit Testing. • Integration and Testing All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures. • Deployment of system Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market. • Maintenance There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment. • All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name ”Waterfall Model”. In this model, phases do not overlap.

2.0.2 Adantage

- This paradigm is straightforward and easy to understand. Because of this, teamwork is incredibly simple and everyone is on the same page.
- With this model, deadlines are met with ease. This is because the team has previously been given precise instructions and no time is lost trying to figure out how to proceed.
- Since this type of architecture involves a lot of paperwork, it is better suited for larger projects with greater teams.

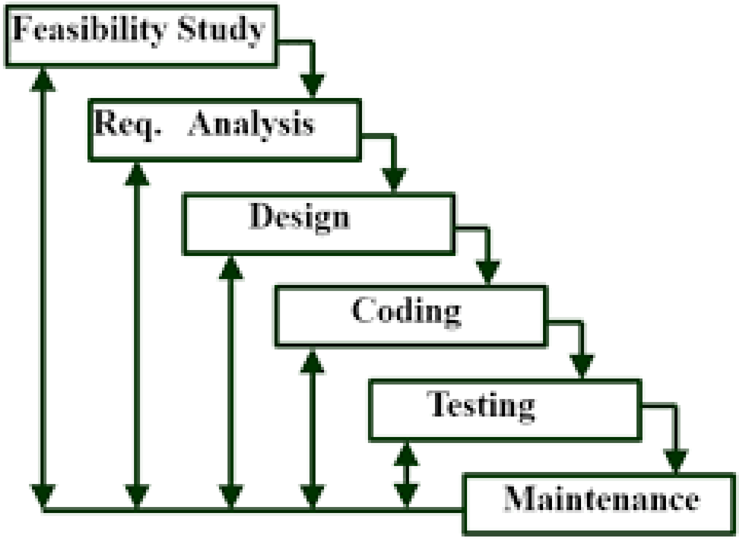
2.0.3 Disadvantage

- Making adjustments to the product at a later stage in the project is practically difficult, which is one of this model's main drawbacks. This is because the steps are tied to one another in a sequential process, thus changing anything would be exceedingly difficult.
- The inability of the stakeholders and customers to utilise or view the product right away is another drawback of this approach. The customer will have to wait a few months before they can view the goods, which occasionally causes them to feel uncomfortable or disappointed.
- As the product cannot be altered at a later stage of the development process, there is also no space for error. Therefore, extensive investigation is required.

2.0.4 Iterative waterfall Model

Iterative Model - Design

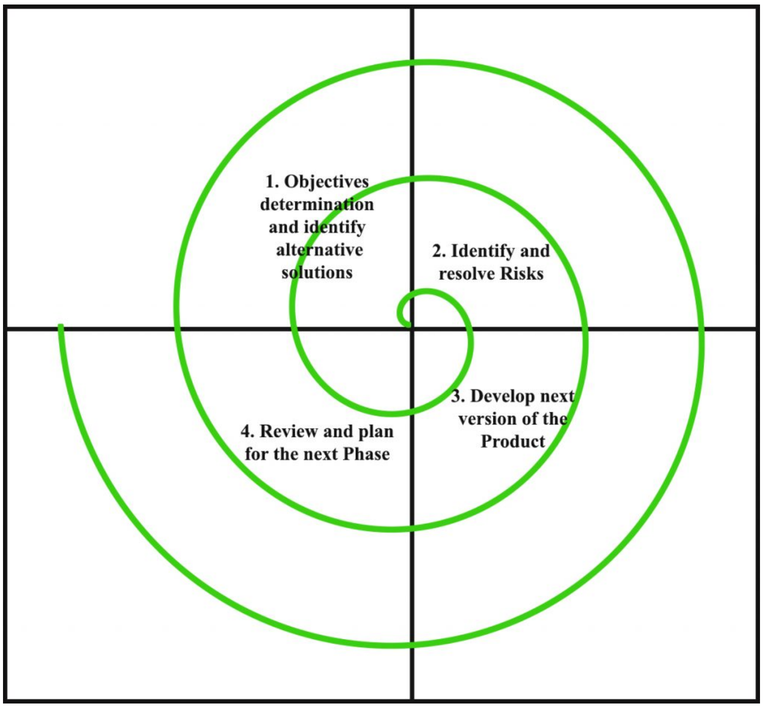
Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental). The following illustration is a representation of the Iterative and Incremental model Iterative and Incremental development is a combination of both iterative design or iterative method and incremental build model for development. ”During software development, more than one iteration of the software development cycle may be in progress at the same time." This process may be described as an "evo- lutionary acquisition" or "incremental build" approach." In this incremental model, the whole requirement is divided into various builds. During each iteration, the development module goes through the requirements, design, implementation and testing phases. Each subsequent release of the module adds function to the previous release. The process continues till the complete system is ready as per the requirement. The key to a successful use of an iterative software development lifecycle is rigorous validation of requirements, and verification testing of each version of the software against those requirements within each cycle of the model. As the software evolves through successive cycles, tests must be repeated and extended to verify each version of the software.



2.0.5 Spiral Model

Spiral Model - Design

The spiral model has four phases. A software project repeatedly passes through these phases in iterations called Spirals. • Identification This phase starts with gathering the business requirements in the baseline spiral. In the subsequent spirals as the product matures, identification of system requirements, subsystem requirements and unit requirements are all done in this phase. This phase also includes understanding the system requirements by continuous communication between the customer and the system analyst. At the end of the spiral, the product is deployed in the identified market. • Design The Design phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and the final design in the subsequent spirals. • Construct or Build The Construct phase refers to production of the actual software product at every spiral. In the baseline spiral, when the product is just thought of and the design is being developed a POC (Proof of Concept) is developed in this phase to get customer feedback. Then in the subsequent spirals with higher clarity on requirements and design details a working model of the software called build is produced with a version number. These builds are sent to the customer for feedback. • Evaluation and Risk Analysis Risk Analysis includes identifying, estimating and monitoring the technical feasibility and management risks, such as schedule slippage and cost overrun. After testing the build, at the end of first iteration, the customer evaluates the software and provides feedback. The following illustration is a representation of the Spiral Model, listing the activities in each phase. Based on the customer evaluation, the software development process enters the next iteration and sub- sequently follows the linear approach to implement the feedback suggested by the customer. The process of iterations along the spiral continues throughout the life of the software


1. Objectives determination and identify alternative solutions
2. Identify and resolve Risks
3. Develop next version of the Product
4. Review and plan for the next Phase

2.0.6 Advantage

- It enables the developers to easily add new functionality to the software that was before impossible to do

- We are able to receive feedback following each spiral, which is a highly advantageous feature that allows us to continuously enhance our offering
- Tasks can be ranked in order of importance based on market demands. We may focus on the iteration that the market demands, for instance, and finish the other iterations later.

2.0.7 Disadvantage

- In order to manage these spirals, highly qualified managerial personnel are needed. If management is poor, there is a chance that the spiral will continue forever.
- Due to the complexity of this process, there is an increased risk of error.

When selecting a particular model for a project, it is important to consider various factors such as the project's goals, the available resources, and the technology stack. In the case of a blockchain-based malware detection system, it is essential to select a model that is suitable for the unique features and requirements of the technology.

To choose the appropriate model for the project, the researcher may have reviewed several research papers and articles on the topic. During this process, the researcher may have come across a research paper that discussed the deficiencies of using traditional Software Development Life Cycle (SDLC) models for blockchain-enabled smart contract applications.

The paper may have argued that the immutability feature of blockchain-enabled smart contracts, which is inherited from the underlying architecture of distributed ledger technologies, makes traditional SDLC models a poor fit for such applications. This is because traditional SDLC models assume that changes to software requirements or design can be made easily, and that the software can be modified or updated as needed. However, the immutability of smart contracts means that once they are deployed on the blockchain, they cannot be modified.

Therefore, the researcher may have concluded that traditional SDLC models are not suitable for the blockchain-based malware detection system project. Instead, the researcher may have chosen a model that is more suitable for the unique characteristics of blockchain technology and smart contracts.

For example, the researcher may have chosen an Agile development model, which emphasizes flexibility and adaptability, and can accommodate changes to requirements and design throughout the development process. Alternatively, the researcher may have chosen a DevOps model, which emphasizes collaboration and communication between development and operations teams, and can help ensure the efficient and effective deployment of smart contract-based applications.

In summary, selecting the appropriate model for a blockchain-based malware detection system project requires careful consideration of the unique features and requirements of blockchain technology and smart contracts. By selecting a suitable model, the project team can ensure that the development process is efficient, effective, and tailored to the needs of the project.
Research paper source :- "https://arxiv.org/ftp/arxiv/papers/2001/2001.10589.pdf"

Chapter 3

Requirement Analysis and Specification

3.1 System Requirement Specifications (SRS)

The System Requirement Specifications for Blockchain based malware detection system include the following key aspects:

Input Sources: The system should support various input sources, such as images, video streams, or live camera feeds.

Code compiler: A basic code compiler should be installed in it.

Integration: The system should be capable of integrating with other systems or applications, such as handling large database.

Scalability: The system should be scalable to handle varying workloads and accommodate increasing data volume.

Deployment Flexibility: The system should be deployable on edge devices for local processing or on any servers for wider network access.

- The steps of Requirement Analysis covers :
- Creating the Context Diagram
- Developing Prototypes
- Modeling the Requirements
- Finalizing the Requirements

3.1.1 Design Phase

In SDLC, the design phase is where the developers define the details of the project. Depending on the product, these details may include the analysis phase, data, drawings, schematics, and designs, starting with these designs, where each design determines the project of a major product. The design phase is an important step in creating a digital product. This is when the team successfully and thoroughly interprets thoughts and ideas about the product. In short, the real work but as there's a property of blockchain that is it is immutable, can't be changed at any step hence any present SDLC model won't be appplicable.

3.1.2 Software Requirements:

Developer Perspective

- **Operating system :** Windows 10 and above
- **Coding Language :** • Python (3.7.6) Open Libraries/Modules

User Perspective

- **Operating system :** Windows 7 and above, Linux.

3.1.3 Hardware Requirements

Developer perspective

- **Processor:** i5 Processor and above.
- **Hard Disk :** would be able to acces cloud
- **RAM :** 4GB(min), 8GB 0r above (recommended)

User Perspective

- **Processor:** i3 Processor and above.
- **Hard Disk :**
- **RAM:** 4GB(min)

Chapter 4

Literature Review

4.0.1 Blockchain

Blockchain Blockchain is a technology to secure integrity and reliability of transaction records without trusted 3rd service provider, by having all the participants in the network create, record, store and verify transaction information jointly, and has the structure to realize various application services based on distributed network infrastructure using security technologies including Hash, Digital Signature and Cryptography (Bahga and Madisetti, 2016). This Blockchain technology was designed to save and use a cryptocurrency called Bitcoin safely. The Blockchain 1.0, which had main functions of issuing, distributing and transacting digital currencies, as the core technology of Bitcoin, is now overcoming the limitations of the existing Bitcoin and being developed into Blockchain 2.0, aiming for expansion into various areas (Financial Services Commission, 2016). The representative technology of Bitcoin 2.0 is Ethereum. Along with the cryptocurrency function, Smart Contracts, in which various types of programs for the transaction scripts of Bitcoins are made possible, are realized (UK, 2016). It is expected that Blockchain will be expanded to a platform in which various decentralized applications are developed and operated, including contracts for real estate and online voting (Tapscott and Tapscott, 2016).

4.0.2 Characterstics of Blockchain

The characteristics of Blockchain are that, since it is a distributed structure, the cost incurred in P2P transactions can be reduced without the need for a trusted 3rd service provider, and there is no need for centralized organizations or trusted 3rd parties to guarantee trust (Dorri et al., 2012). There is no need for centralized organizations such as Korea Financial Telecommunications Clearings Institute, or a public certificate authority, and since new innovative processes can be introduced, the expenses necessary for the operation, maintenance, security and financial transactions of various centralized systems can be reduced (Financial Services Commission, 2016). Furthermore, since all users (nodes) have transaction ledger, even if some parts of the network encounter problems, they do not affect the whole Blockchain, and since it is a distributed structure, it is expected that it will not show vulnerability in security to the attacks such as DDoS (Lee, 2017). For these reasons,it has the advantages of being more transparent, and easily tracked than those of the existing financial transactions

4.0.3 Types Of blockchain

Public Blockchain is open type, in which anyone can participate. All participants may freely access data and make transactions, but since numerous unverified users are participating, advanced encryption and verification are needed, and thus, network expansion is difficult and it is very slow. Furthermore, public Blockchain forms a perfect distributed structure, and participants of Blockchain network are pseudo-anonymous, and thus Public Blockchain is not appropriate for financial services that need to be controlled by the centralized information management system. Therefore, financial institutions are paying attention to Blockchain for consortium (Consortium Blockchain) and Private Blockchain that will make the most of the advantages of Blockchain such as cost reduction, while not losing the system control authority and initiative, which are needed in the financial service. Unlike Public Blockchain, which provided pseudo-anonymity, it is possible to identify the subject in Private Blockchain. The transactions are handled fast, network expansion is easy and could be modified whichever way the user wishes, and thus, is suitable for the financial service. Therefore, it is receiving attention from the companies and financial institutions recently. Private Blockchain is the Blockchain in which the owner generates and manages the Blockchain. This is appropriate if the Blockchain owner wishes to manage the Blockchain as the centralized system. For example, you could consider the following: for the transaction system in which real-time transaction is important, it should be operated as the centralized system, while using Private Blockchain for the purpose of storing and verifying transaction details at a safe and low cost, after the transaction has been made (Financial Security Institute, 2015).

4.0.4 Consensus Mechanism in Blockchain:A Beginner's Guide

Why Blockchains Need Consensus Mechanisms

Consensus mechanisms form the backbone of all cryptocurrency blockchains, and are what make them secure. Before we delve into the different consensus mechanisms, we need to first define what it means for blockchains to achieve consensus.

1. Proof of work (POW)
2. Proof of Stake (PoS)
3. Proof of Activity (PoA)
4. Proof of Authority (PoA)
5. Proof of Burn (PoB)
6. Proof of Elapsed Time (PoET)

4.0.5 Blockchain Architecture Basics: Components, Structure, Benefits Creation- Anastasiia Lastovetska

, blockchain technology has the core characteristics of decentralization, accountability, and security. This technique can improve operational efficiency and save costs significantly. The demand and usage of applications built on blockchain architecture will only evolve.

1. What is Blockchain Architecture?
2. Database vs. Blockchain Architecture
3. Core Components of Blockchain Architecture: How Does It Work
4. How to Make a Private Blockchain Architecture
5. Blockchain Network Creation
6. Blockchain Code Creation
7. Key Characteristics of Blockchain Architecture
8. Create Your Own Blockchain Architecture

4.0.6 The Ethereum Virtual Machine—How doesit work? - Luit Hollander

Creating a smart contract

Smart contracts are often written in a programming language called “Solidity”, a language similar to JavaScript and C++. Other languages for writing smart contracts include Vyper and Bamboo. Before Solidity was released, other languages like Serpent (deprecated) and Mutan (deprecated) were used.

Opcodes

Under the hood, the EVM uses a set of instructions (called opcodes) to execute specific tasks. At the time of writing, there are 140 unique opcodes. Together, these opcodes allow the EVM to be Turing-complete. This means the EVM is able to compute (almost) anything, given enough resources. Because opcodes are 1 byte, there can only be a maximum of 256 (16^2) opcodes. For simplicity’s sake, we can split all opcodes into the following categories:

1. Stack-manipulating opcodes (POP, PUSH, DUP, SWAP)
2. Arithmetic/comparison/bitwise opcodes (ADD, SUB, GT, LT, AND, OR)
3. Environmental opcodes (CALLER, CALLVALUE, NUMBER)
4. Memory-manipulating opcodes (MLOAD, MSTORE, MSTORE8, MSIZE)
5. Storage-manipulating opcodes (SLOAD, SSTORE)
6. Program counter related opcodes (JUMP, JUMPI, PC, JUMPDEST)

4.0.7 A case study on business model innovations using Blockchain: focusing on financial institutions

Purpose – Blockchain is a distributed ledger, in which the blocks containing transaction details are connected chronologically to form a series of chains, thus raising the possibility of improving the process and innovating business model for the financial institutions. The purpose of this paper is to study the actual cases of Blockchain applied in Korea in 2017, so that a vision of business model innovation of financial institutions can be drawn.

Design/methodology/approach – The financial institutions in Korea are in the technology verification stage to introduce Blockchain technology. Since there is an insufficient amount of actual measurement data, case study method was adopted. The authors interviewed ICT officers of major banks in Korea. The purpose of the interview was to understand the relationship between Blockchain and business models of financial institutions, and the effects and challenges that Blockchain has on the business model of financial institutions.

Findings – From the perspective of financial institutions, the emergence of Blockchain does not just have technical significance – emergence of highly efficient database system – but has the possibility that if the business model of existing financial intermediaries disappears or get reduced, the financial services relying on them can disappear altogether, or some of them can be replaced, and financial transaction patterns of consumers can be changed. As a case studies researched for this paper, it was discovered that the distributed characteristic of Blockchain cannot be applied when actually developing financial services.

4.0.8 Block Chain - Blueprint for a new technology by. Melanie Swan

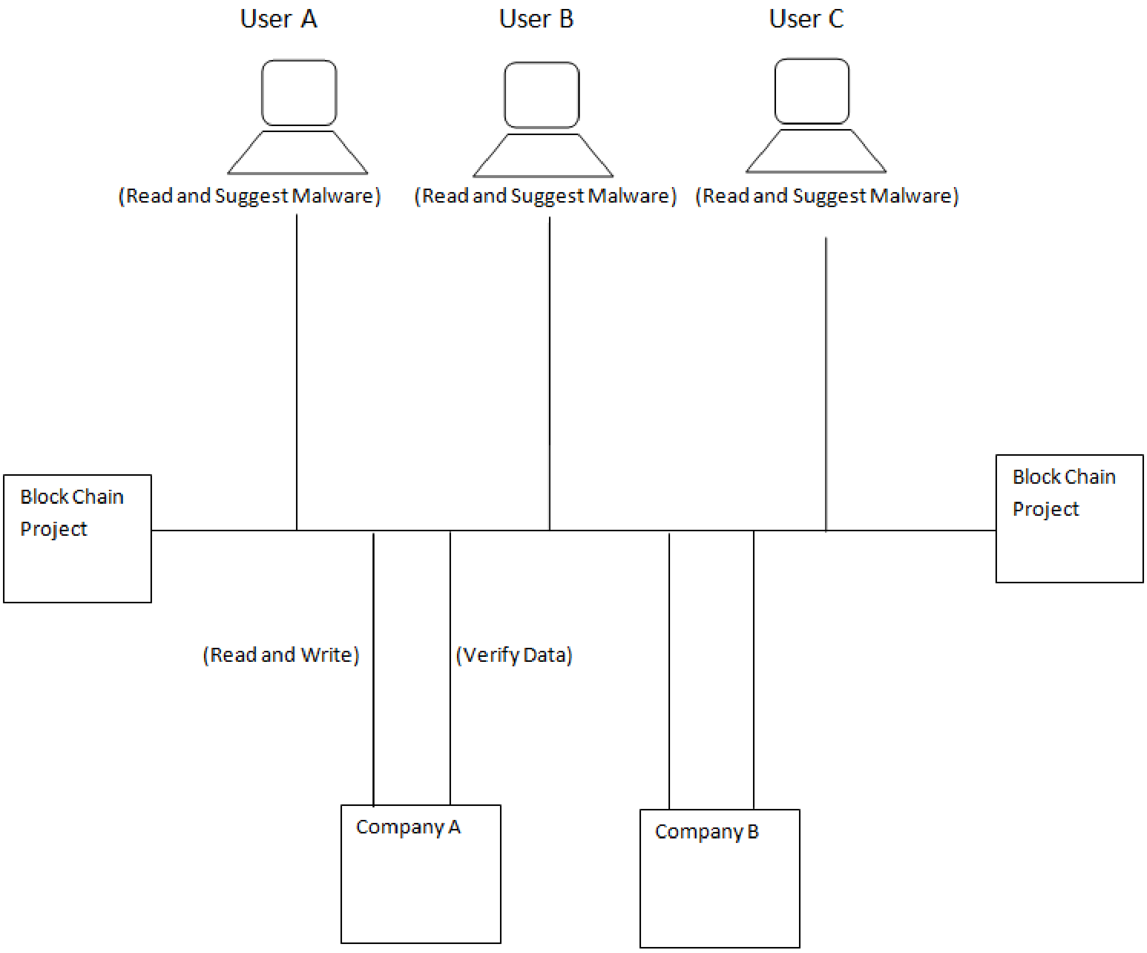
Topice Include :-

1. Concepts, features, and functionality of Bitcoin and the blockchain
2. Using the blockchain for automated tracking of all digital endeavors
3. Enabling censorship-resistant organizational models
4. Creating a decentralized digital repository to verify identity
5. Possibility of cheaper, more efficient services traditionally provided by nations
6. Blockchain for science: making better use of the data-mining network
7. Personal health record storage, including access to one's own genomic data
8. Open access academic publishing on the blockchain

Chapter 5

Design

5.1 Block Diagram



Chapter 6

Implementation

6.1 Methodlogy

Blockchain technology works by creating an environment that is secure and transparent for the financial transactions of virtual values such as Bitcoin. Hash codes of each block keep records safe in the blockchain. This is mainly because irrespective of the size of the information or document, the mathematical hash function provides a hash code of the same length for each block. So, attempting to change a block of information would generate a completely new hash value . A network that is open to everyone and concurrently maintains user's anonymity undoubtedly raises trust issues regarding the participants. So, to build the trust the participants need to go through several consensus algorithms such as Proof of Work and Proof of Stake. The digital cryptocurrency Bitcoin uses the first-ever blockchain technology .

It is a digital store of value that enables peer to peer transactions over the internet without the intervention of a third party. The blockchain network is a decentralized structure that consists of scattered nodes (computers) that inspect and validate the authenticity of any new transactions that attempt to take place. This combine agreement is done through several consensus models by the process of mining. The process of mining demonstrates that each node trying to add a new transaction has gone through and solved the complex computational puzzle through extensive work and deserves to get a reward in return for their service. For the validation of a transaction, the network must confirm the following conditions: The sender account holds sufficient Bitcoin balance that it intends to transfer.

6.1.1 Consensus protocol

The term consensus mechanism refers to the entire stack of protocols, incentives and ideas that allow a network of nodes to agree on the state of a blockchain.

Ethereum uses a proof-of-stake-based consensus mechanism that derives its crypto-economic security from a set of rewards and penalties applied to capital locked by stakers. This incentive structure encourages individual stakers to operate honest validators, punishes those who don't, and creates an extremely high cost to attack the network.

Then, there is a protocol that governs how honest validators are selected to propose or validate blocks, process transactions and vote for their view of the head of the chain. In the rare situations where multiple blocks are in the same position near the head of the chain, there is a fork-choice mechanism that selects blocks that make up the 'heaviest' chain, measured by the number of validators that voted for the blocks weighted by their staked ether
balance.

Some concepts are important to consensus that are not explicitly defined in code, such as the additional security offered by potential out-of-band social coordination as a last line of defense against attacks on the network.

1. Proof of Work (PoW): This consensus algorithm is used to select a miner for the next block generation. Bitcoin uses this PoW consensus algorithm. The central idea behind this algorithm is to solve a complex mathematical puzzle and easily give out a solution. This mathematical puzzle requires a lot of computational power and thus, the node who solves the puzzle as soon as possible gets to mine the next block. For more details on PoW, please read Proof of Work (PoW) Consensus

2. Practical Byzantine Fault Tolerance (PBFT):Practical Byzantine Fault Tolerance is a consensus algorithm introduced in the late 90s by Barbara Liskov and Miguel Castro. pBFT was designed to work efficiently in asynchronous(no upper bound on when the response to the request will be received) systems. It is optimized for low overhead time.

3. Proof of Stake (PoS): This is the most common alternative to PoW. Ethereum has shifted from PoW to PoS consensus. In this type of consensus algorithm, instead of investing in expensive hardware to solve a complex puzzle, validators invest in the coins of the system by locking up some of their coins as stake. After that, all the validators will start validating the blocks. Validators will validate blocks by placing a bet on it if they discover a block which they think can be added to the chain. Based on the actual blocks added in the Blockchain, all the validators get a reward proportionate to their bets and their stake increase accordingly. In the end, a validator is chosen to generate a new block based on their economic stake in the network. Thus, PoS encourages validators through an incentive mechanism to reach to an agreement.

4. Proof of Burn (PoB): With PoB, instead of investing into expensive hardware equipment, validators 'burn' coins by sending them to an address from where they are irretrievable. By committing the coins to an unreachable address, validators earn a privilege to mine on the system based on a random selection process. Thus, burning coins here means that validators have a long-term commitment in exchange for their short-term loss. Depending on how the PoB is implemented, miners may burn the native currency of the Blockchain application or the currency of an alternative chain, such as bitcoin. The more coins they burn, the better are their chances of being selected to mine the next block. While PoB is an interesting alternative to PoW, the protocol still wastes resources needlessly. And it is also questioned that mining power simply goes to those who are willing to burn more money.

5. Proof of Capacity: In the Proof of Capacity consensus, validators are supposed to invest their hard drive space instead of investing in expensive hardware or burning coins. The more hard drive space validators have, the better are their chances of getting selected for mining the next block and earning the block reward.
   In this project we will be using

   (a) Proof of work (POW)

   (b) Proof of Stake (PoS)

6.1.2 Smart Contract Layer

Smart contracts are simply programs stored on a blockchain that run when predetermined conditions are met. They typically are used to automate the execution of an agreement so that all participants can be immediately certain of the outcome, without any intermediary's involvement or time loss. They can also automate a workflow, triggering the next action when conditions are met.

How smart contracts work

Smart contracts work by following simple "if/when…then…" statements that are written into code on a blockchain. A network of computers executes the actions when predetermined conditions have been met and verified. These actions could include releasing funds to the appropriate parties, registering a vehicle, sending notifications, or issuing a ticket. The blockchain is then updated when the transaction is completed. That means the transaction cannot be changed, and only parties who have been granted permission can see the results.

Within a smart contract, there can be as many stipulations as needed to satisfy the participants that the task will be completed satisfactorily. To establish the terms, participants must determine how transactions and their data are represented on the blockchain, agree on the "if/when...then…" rules that govern those transactions, explore all possible exceptions, and define a framework for resolving disputes.

Then the smart contract can be programmed by a developer – although increasingly, organizations that use blockchain for business provide templates, web interfaces, and other online tools to simplify structuring smart contracts.

Benefits of smart contracts

1. Speed, efficiency and accuracy Once a condition is met, the contract is executed immediately. Because smart contracts are digital and automated, there's no paperwork to process and no time spent reconciling errors that often result from manually filling in documents.
2. Trust and transparency Because there's no third party involved, and because encrypted

records of transactions are shared across participants, there's no need to question whether information has been altered for personal benefit.

3. Security Blockchain transaction records are encrypted, which makes them very hard to hack. Moreover, because each record is connected to the previous and subsequent records on a distributed ledger, hackers would have to alter the entire chain to change a single record.

4. Savings Smart contracts remove the need for intermediaries to handle transactions and, by extension, their associated time delays and fees.

6.1.3 Communication layer

The idea to keep this software as an open source is that only to keep the system communicated with each and its every user
we are using Git to keep the software open and editable.

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is easy to learn and has a tiny footprint with lightning fast performance. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like cheap local branching, convenient staging areas, and multiple workflows.

6.1.4 Data Store and abstraction

Blockchain storage is a way of saving data in a decentralized network, which utilizes the unused hard disk space of users across the world to store files. The decentralized infrastructure is an alternative to centralized cloud storage and can solve many problems found in a centralized system

How blockchain storage works

Block chain relies on distributed ledger technology (DLT). The DLT acts as a decentralized database of information about transactions between various parties. Operations fill the DLT in chronological order and are stored in the ledger as a series of blocks. An interconnected chain is formed between blocks with each one referring to the block before it, thus creating a blockchain.

In blockchain storage, files are first broken apart in a process called sharding. Each shard is copied to prevent loss of data should an error occur during transmission. The files are also encrypted with a private key that makes it impossible for it to be viewed by other nodes in the network. The replicated shards are distributed among decentralized nodes all over the world. The interactions are recorded in the blockchain ledger, allowing the system to confirm and synchronize the transactions across the nodes in the blockchain. Blockchain storage is designed to save these interactions forever and the data can never be changed.

Blockchainstorage vs. cloud storage

Blockchain storage is a potentially cheaper, more secure and more reliable alternative to centralized cloud storage.

Providers of centralized cloud storage prevent data loss by making copies of the data and storing it in different data centers. The large amount of data that is duplicated in this process can create excessive amounts of surplus information. Also, cloud storage requires enterprise-grade hardware for its data centers. These factors can make centralized data storage significantly more expensive than blockchain storage.

Blockchain Current blockchain storage projects

Sia, InterPlanetary File System (IPFS), MaidSafe and Storj are some of the more prominent companies offering blockchain storage.

Sia is an open source software company that specializes in decentralized cloud storage technology. It provides a platform where users run their own private decentralized cloud. It does not accept payment from customers.

IPFS is a peer-to-peer (P2P) Hypermedia distribution system designed to provide a permanent, decentralized method for storing and sharing files. Nodes within the IPFS network form a distributed file system that can be accessed in many ways, including the Linux-based FUSE interface and HTTP (Hypertext Transfer Protocol). Local files can be added to the IPFS network and made available to the world.

MaidSafe is a company whose goal is to create a new "backbone" for the internet that can be used to access, exchange and store data. The MaidSafe network can also be used to build blockchain decentralized applications (dapps).

Storj aims to provide private, secure and efficient P2P-based cloud storage based on the Ethereum blockchain platform. The Storj platform uses sharding and end-to-end encryption (E2EE) to store and protect data.

Sia, Storj and IPFS have launched their own cryptocurrencies (Siacoin, Storjcoin X and Filecoin) in an attempt to create a market for buying and selling decentralized storage and encouraging its use. The major obstacle that companies launching blockchain storage projects will face is scalability.

Chapter 7

Algorithm

The SHA256 hashing algorithm is a critical component of the blockchain technology that is being used in this project. The algorithm ensures that all functions are performed securely by assigning a unique hash to each block in the blockchain. This is essential for ensuring the integrity of the data, as any changes made to the data will result in a completely different hash.

The way that the SHA256 algorithm works is by taking the output of one block and using it as the input for the next block. This process is repeated until the final 512-bit block is reached, at which point the output of this block is considered the ultimate hash digest. This digest is what is used to verify the integrity of the data within the block.

One of the key benefits of using the SHA256 algorithm is that it is extremely secure. Even if just one digit is changed in the input, the output will be completely different. This makes it virtually impossible for anyone to alter the data without being detected.

In addition to providing security for the data, the SHA256 algorithm also serves as a moderator for the creation and management of addresses, as well as verifying transactions. By using this algorithm, the blockchain is able to maintain a high level of security and reliability, which is essential for the success of this project.

The SHA256 hashing algorithm is widely used in blockchain technology because it is highly secure and ensures the integrity of the data stored in the blockchain. The algorithm generates a fixed-length, 256-bit hash value for each block in the blockchain, and this hash value serves as a unique identifier for that block. It is worth noting that the SHA256 algorithm is not the only algorithm used in blockchain technology, but it is one of the most widely used due to its security and efficiency.

The SHA256 algorithm is a one-way cryptographic function, which means that it is not possible to derive the original input data from the hash value. Additionally, the slightest change to the input data will result in a completely different hash value, making it virtually impossible to tamper with or modify the data stored in the blockchain without being detected.

This algorithm operates by taking an input, processing it through multiple rounds of mathematical operations, and generating a fixed-length hash output. The output of one block is used as the input for the next adjacent block in the blockchain, creating a chain of linked blocks that are all secured using the same algorithm.

The SHA256 algorithm has been extensively tested and is considered one of the most secure hash functions available today. It is used in various applications, such as digital signatures, password storage, and SSL/TLS certificates. In the context of blockchain technology, the SHA256 algorithm serves to create secure, tamper-proof data storage that can be shared across a network of users without the need for a central authority or intermediary.

The SHA256 algorithm is widely used in the field of cryptography due to its strength and security. It is a one-way hashing algorithm, which means that it is not possible to obtain the original data from the output of the algorithm. Instead, the output, or hash, can be used to verify the integrity and authenticity of the data without exposing it.

In the context of blockchain, the SHA256 algorithm is used to create a unique digital signature for each block in the chain. This signature is generated based on the content of the block and is included in the next block as a reference, thereby creating a continuous chain of secure and verifiable data.

One of the main advantages of using the SHA256 algorithm in blockchain is its resistance to tampering. Since the hash of each block is dependent on the contents of the previous block, any attempt to modify the data in a single block would result in an entirely different hash for that block and all subsequent blocks, thus making it easy to detect any tampering attempts.

Furthermore, since the SHA256 algorithm is widely used and has been extensively studied and tested, it is considered to be a secure and reliable choice for blockchain applications. Its widespread adoption in cryptocurrencies such as Bitcoin has further solidified its position as a trusted algorithm for securing digital transactions and data.

Chapter 8

Result

This project is unique in that it is entirely based on blockchain technology. The blockchain is a decentralized, distributed ledger that records transactions across a network of computers. It is secure, transparent, and tamper-proof, making it an ideal technology for storing sensitive data, such as malware and virus information.

As more users interact with the project, the chain of data will continue to grow longer, making it more robust and reliable. The programme will output the total data of all viruses/malware along with their related solutions, making it a comprehensive resource for the cybersecurity industry.

The development of this project would involve every authorised antivirus/security company, as they would continually find and present the fundamental information about the malware, ensuring that the database remains up-to-date and relevant. This open-source system would benefit each individual in the industry by providing a centralised resource for all malware and virus information.

One potential criticism of the project could be its effectiveness. While it may not be perfect, the project will continue to improve with time as the blockchain network grows and more companies contribute to it. The potential benefits of this project could be significant in terms of improving cybersecurity and reducing the impact of malware and virus threats.

Since this project is entirely based on blockchain technology, it offers several advantages that traditional centralized systems do not have. One of the significant benefits is that the blockchain network's data keeps getting longer and longer as user interaction increases, which gradually strengthens the mass of the project. This makes it a highly secure and reliable platform for storing sensitive data, such as malware signatures.

The output of the program will show the total data of all viruses/malware along with their related solutions. This would be possible due to the decentralized nature of the blockchain network, which allows for easy sharing of data among authorized antivirus/security companies. The community that would be involved in the development of this project would be every authorized antivirus/security company, who would continually find and present fundamental information about malware so that other companies in the same industry can stay in touch with one another.

By contributing to this open-source system, authorized companies can benefit each other by keeping their clients safe from potential cyber threats. It is important to note that the effectiveness of the system may not be very high initially, but with time, it will continue to climb as the database becomes more extensive and more companies contribute to its development.

In summary, the project aims to provide a decentralized platform for storing malware signatures and related information to prevent cyber threats. It encourages collaboration between authorized antivirus/security companies, which can contribute to its development and help make it more effective over time.

Chapter 9

Conclusion and Future Work

A blockchain-based malware detection system is a type of security solution that aims to detect and prevent malware attacks using distributed ledger technology. The system uses a decentralized network to store and share information related to malware signatures, patterns of attack, and other relevant data.

One of the main benefits of a blockchain-based system is that it provides a secure and tamper-proof way to store and share data. Each node in the network has a copy of the blockchain, and any changes to the data must be verified by consensus among the nodes. This ensures that the system is resistant to tampering or manipulation by malicious actors.

Another advantage of using blockchain technology for malware detection is that it provides a fast and efficient way to analyze large amounts of data. The distributed nature of the network means that processing power can be shared among many different nodes, allowing for faster and more accurate analysis of potential threats.

One way that a blockchain-based malware detection system can work is by using machine learning algorithms to analyze data from different sources, such as network traffic, system logs, and other security tools. The algorithms can then use this data to identify patterns of attack and other indicators of malicious activity.

The results of this analysis can be stored on the blockchain, allowing other nodes in the network to access the information and use it to enhance their own security defenses. By sharing information in this way, the system can become more effective over time, as it learns from new threats and adapts to evolving attack patterns.

There are several areas where future research and development can be focused to enhance the effectiveness of a blockchain-based malware detection system. One area is to improve the accuracy of the machine learning algorithms used to analyze data. This could involve developing more advanced algorithms or leveraging artificial intelligence techniques to improve the accuracy and speed of threat detection.

Another area of focus could be on integrating blockchain technology with other security systems, such as intrusion detection and prevention systems, to create a more comprehensive security framework. This would involve developing protocols and standards for interoperability between different security tools, allowing them to work together seamlessly to detect and prevent malware attacks.

Overall, a blockchain-based malware detection system has the potential to be a valuable tool in the fight against cybercrime. By leveraging the power of distributed ledger technology and machine learning algorithms, it can provide a secure, efficient, and effective way to detect and prevent malware attacks. Ongoing research and development in this area can help to further improve the system and make it even more effective in protecting against cyber threats.

Appendix A

Citations and references

1. Smart Contracts and Chaincode
A smart contract and the ledger, as seen from the perspective of an application developer, make up the core of a Hyperledger Fabric blockchain system. A smart contract specifies the executable logic that creates new facts that are added to the ledger, as opposed to a ledger, which contains facts about the present and past states of a collection of business objects. A chaincode can be used for low level system programming of the Fabric blockchain, however administrators often utilise it to aggregate related smart contracts for deployment. This chapter will concentrate on the existence of smart contracts and chaincode as well as their use.

2. A case study on business model innovations using Blockchain: focusing on financial institutions (JaeShup Oh SookMyung Women's University, Seoul, Republic of Korea, and Ilho Shong Dongguk University, Seoul, Republic of Korea)

Blockchain is a distributed ledger created by blocks containing transaction details connected in

chronological order to form a series of chain. It is a distributed ledger in which participants of Blockchain peer-to-peer (P2P) network, and not the central administrator, generate blocks. The possibilities of use of Blockchain are acknowledged in many different fields, resulting in many developments and studies being conducted, and investments are being made actively (Cho and Park, 2017). From the perspective of financial institutions, the emergence of Blockchain does not just have technical significance – emergence of highly efficient database system – but has the possibility that if the business model of existing financial institutions or financial intermediaries disappear, the financial services relying on them may disappear altogether or be partially replaced, and financial transaction patterns of consumers can be changed. On the other hand, it is expected that the areas of use of Blockchain will be expanded to become the means to increase financial inclusion beyond being a new business model for the financial institutions (Santander, 2015).

3. Building a Transparent Supply Chain (by Vishal Gaur and Abhinav Gaiha)

Blockchain, the digital record-keeping technology behind Bitcoin and other cryptocurrency networks, is a potential game changer in the financial world. But another area where it holds great promise is supply chain management. Blockchain can greatly improve supply chains by enabling faster and more cost-efficient delivery of products, enhancing products' traceability, improving coordination between partners, and aiding access to financing

4. Blockchain in capital market

The capital markets industry is going through profound changes in business dynamics due to regulation, technology-led market disruption, and transformed economics of core business areas. The era of digitalization has resulted in sweeping changes to the industry mindset – while many firms took nearly a decade to stabilize after the 2008 crisis, they were soon confronted with expectations of a new way of doing business as a result of the digital revolution. These new expectations meant changing norms in an industry with long-persisting issues

5. Blockchain in Institutional Capital Markets

Hyperledger Composer is a set of collaboration tools for building blockchain business networks that make it simple and fast for business owners and developers to create smart contracts and blockchain applications to solve business problems. Built with JavaScript, leveraging modern tools including node.js, npm, CLI and popular editors, Composer offers business centric abstractions as well as sample apps with easy to test devops processes to create robust blockchain solutions that drive alignment across business requirements with technical development.

6. Blockchain Cyber Security.

Blockchain is gaining traction today, but critics who question the scalability, security, and sustainability of the technology remain. Deloitte member firms across the globe are continuing to collaborate to build blockchain capabilities to develop world class solutions and services for clients.

7. Why Block chain security

it produces a structure of data with inherent security qualities. It's based on principles of cryptography, decentralization and consensus, which ensure trust in transactions. In most blockchains or distributed ledger technologies (DLT), the data is structured into blocks and each block contains a transaction or bundle of transactions. Each new block connects to all the blocks before it in a cryptographic chain in such a way that it's nearly impossible to tamper with. All transactions within the blocks are validated and agreed upon by a consensus mechanism, ensuring that each transaction is true and correct.

8. Create and test smart contracts using Python

This project uses a Python wrapper around Algorand SDK, so you should have Python 3 installed on your system. Also, this project uses python3-venv package for creating virtual environments and you have to install it if it's not already installed in your system. For a Debian/Ubuntu based systems,

Bibliography

Narayanan, A., Chandrasekaran, K. (2019). Blockchain-based malware detection and analysis. In 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) (pp. 28-35). IEEE.

Tariq, M. H., Yaqoob, I. (2018). A blockchain-based malware detection and mitigation framework for IoT devices. IEEE Access, 6, 29725-29733.

Gupta, R., Kaur, P. (2021). Malware detection using blockchain technology. Journal of Ambient Intelligence and Humanized Computing, 12(7), 6875-6888.

Alazab, M., Meinel, C., Song, H. (2018). Blockchain-based malware detection: a new solution for cyber security. In 2018 IEEE/ACM 4th International Conference on Big Data Computing Applications and Technologies (BDCAT) (pp. 211-216). IEEE.

Abdellatif, A., Elshazly, H. (2019). Blockchain-based malware detection system for cloud computing environments. In 2019 5th IEEE International Conference on Computer and Communications (ICCC) (pp. 1575-1579). IEEE.

Zhang, J., Li, J., Li, B., Guan, X. (2019). Malware detection based on blockchain technology. In 2019 IEEE International Conference on Smart Internet of Things (SmartIoT) (pp. 179-184). IEEE.

Kuo, T. T., Lin, H. F., Chen, Y. L. (2020). BlockChain-based IoT Malware Detection and Prevention System. IEEE Transactions on Industrial Informatics, 16(4), 2763-2772.

Haq, S. U., Ahmed, E., Yasin, U., Qureshi, A. R. (2021). A blockchain-based malware detection system for healthcare. Journal of Ambient Intelligence and Humanized Computing, 12(12), 12707-12723

Narula, S., Gupta, M., Jangir, S. (2020). A review of blockchain technology for cyber security. International Journal of Engineering and Advanced Technology (IJEAT), 9(4), 5024-5030. This article provides an overview of how blockchain can be used for cyber security, including for malware detection and prevention.

Dangi, N., Chaudhary, R., Chaudhary, V. (2020). Block Chain based Anti-Malware System. International Journal of Engineering Research and Technology, 9(9), 1248-1252. This paper describes a blockchain-based anti-malware system that can detect and prevent malware attacks by leveraging blockchain's secure and decentralized architecture.

Ren, X., Li, H., Wang, H., Huang, Y. (2019). Malware Detection Based on Blockchain Technology. In Proceedings of the 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (pp. 754-758). IEEE. This conference paper presents a malware detection system that uses blockchain technology to securely store and share malware signatures, enhancing the accuracy of malware detection.

Chen, M., Ma, X., Wu, J. (2019). Malware detection using blockchain-based machine learning approach. In Proceedings of the 2019 IEEE International Conference on Intelligence and Security Informatics (ISI) (pp. 46-50). IEEE. This conference paper proposes a machine learning approach for malware detection that uses blockchain to securely store and share data between different systems.

Shrivastava, S., Shukla, S., Sharma, S., Shahi, A. (2020). Malware Detection and Prevention using Blockchain. International Journal of Engineering and Computer Science, 9(9), 28555-28558. This paper presents a malware detection and prevention system that uses blockchain to securely store and share malware signatures and metadata, improving the efficiency and accuracy of malware detection.