



PAPER

## Enhancing blockchain security: a novel approach to integrated malware defence mechanisms

To cite this article: Aastha Sharma *et al* 2024 *Eng. Res. Express* **6** 025215

View the [article online](#) for updates and enhancements.

### You may also like

- [Research on Practical Byzantine Fault Tolerant Consensus Algorithm Based on Blockchain](#)  
Xiandong Zheng and Wenlong Feng
- [Bioclimatic analysis of streets and buildings of the Byzantine period](#)  
Flora Bougiatioti and Aineias Oikonomou
- [A Byzantine consensus based on proof-of-work of nodes' behaviors](#)  
Wang Dingyu, Wang Dingmin, Qin Lupin et al.

# Engineering Research Express



## PAPER



# Enhancing blockchain security: a novel approach to integrated malware defence mechanisms

RECEIVED  
11 March 2024

REVISED  
2 May 2024

ACCEPTED FOR PUBLICATION  
14 May 2024

PUBLISHED  
22 May 2024

Aastha Sharma, Divya Upadhyay\*  and Shanu Sharma\* 

Department of Computer Science and Engineering, ABES Engineering College, Ghaziabad, Uttar Pradesh, 282007, India

\* Authors to whom any correspondence should be addressed.

E-mail: [aastha.22m0101002@abes.ac.in](mailto:aastha.22m0101002@abes.ac.in), [upadhyay.divya@gmail.com](mailto:upadhyay.divya@gmail.com) and [shanu.sharma16@gmail.com](mailto:shanu.sharma16@gmail.com)

**Keywords:** SHA-256, blockchain, security, digital signatures, reentrancy attack, Byzantine fault tolerance, DDoS

## Abstract

This paper introduces a novel integrated hybrid malware attack detection algorithm, focusing on enhancing cybersecurity within blockchain systems by addressing the prevalent challenges of Byzantine fault tolerance, Reentrancy, and DDOS attacks. The significance of this research lies in its contribution to safeguarding blockchain technology, a cornerstone for secure, decentralized digital transactions, against sophisticated malware threats. Current cybersecurity solutions frequently fall short of offering a complete defense mechanism, making it difficult to effectively combat a variety of dynamic malware attacks at the same time. Thus, the main objective of this research is to provide a hybrid framework that combines DDOS attack prevention, reentrancy attack detection, and Byzantine fault tolerance detection into a single, cohesive architecture. The proposed hybrid framework encompasses a detailed algorithmic approach integrating SHA-256 and DSA to analyze the aforementioned three malware attacks. A hybrid model combining these algorithms, implemented in one block, has been developed to mitigate malicious activity. These measures aim to improve computational complexity and expedite execution within the network of nodes. To test the efficacy of the proposed framework, the approach is tested on the NSL-KDD dataset to analyze the malicious activities. The performance analysis of the proposed frameworks presents a recall and F1 score of 73 and .68 respectively. Furthermore, for efficient mitigation, the time and space complexity analysis is performed on proposed algorithms for attack analysis, which resulted in a combination of constant and linear time complexity operations. The findings reveal that the proposed algorithm successfully identifies and mitigates the targeted malware attacks and maintains optimal performance in terms of time and space complexity. Specifically, the algorithm showcases linear and constant time complexities across different attack vectors, ensuring swift and scalable defense capabilities. This research's contribution to the cybersecurity field is significant, offering a robust, scalable solution that enhances the resilience of blockchain networks against a broad spectrum of malware attacks.

## 1. Introduction

Blockchain technology has emerged as a revolutionary and collaborative platform, enabling secure and transparent digital transactions [1]. Blockchain provides a decentralized technology that enables the creation of a secure and transparent environment for data sharing in a Peer-to-Peer (p2p) network. It operates as a decentralized and distributed ledger, where data is saved in blocks and then kept in a hyperledger or immutable ledger. These blocks are then shared over a p2p network [2]. It is also known as Distributed Ledger Technology (DLT), which creates and maintains all the user records within the block [3]. During the updation of information in the block, the copy of information gets updated at multiple decentralized locations simultaneously [4].

The blocks in the blockchain are cryptographically linked together to secure more data within the data. It also managed to transfer crypto-assets more securely and fast to maintain the optimization while executing the process [5]. These assets can be tangible (cash, car, accounts, finance management, etc) or intangible (intellectual

property (IP), copyright, etc). The blockchain's significance lies in its ability to safeguard data through cryptographic techniques, connecting it in the form of connected nodes (linked-list). It also offers transparency, flexibility, accessibility, and the ability to trace payment orders and account data [6, 7].

Blockchain technology, which provides decentralized and transparent solutions, has completely changed the way we store and move data. It operates on a decentralized ledger, allowing anyone to trade their information or data [5]. Connected members are responsible for maintaining the usability policies of the Blockchain platform. Due to its decentralized and unchangeable characteristics, it has attracted significant attention across various sectors. However, like any innovative technology, blockchain is not immune to security vulnerabilities [7]. Reentrancy, Byzantine Fault Tolerance Detection, and Distributed Denial of Service (DDoS) attacks pose significant threats to the reliability and trustworthiness of blockchain systems [5–7]. The presence of these security vulnerabilities emphasizes the necessity for ongoing innovation and advancement in blockchain technology to outpace malevolent entities [8, 9].

Considering the dynamic nature of blockchain technology and its associated security consequences, the objective of this study is to examine the principles underlying security assaults on blockchains and provide solutions to improve resilience and detection capabilities. In this paper, the following key steps are considered for the development of secured infrastructure for blockchain.

- Investigating the fundamental mechanics of blockchain technology, including analyzing its features, understanding the reasons for its adoption, and examining the procedures of generating requests and mining.
- Examining the origins, effects, and possible weaknesses caused by malware attacks on blockchain platforms, with a focus on the necessity of implementing strong security measures.
- Comprehensive analysis of malware assaults, specifically examining their development, the decision-making mechanisms in peer-to-peer networks, and the potential consequences for blockchain security to identify the current security challenges.
- Analyzing the recent work done for the development of security measures in blockchain and evaluating the effectiveness of popular security measures such as SHA-256 and digital signatures.
- Proposing a hybrid solution that combines DDOS attack prevention, reentrancy attack detection, and Byzantine fault tolerance detection into a single, cohesive architecture.
- Evaluating the effectiveness of the proposed framework by implementing it for malicious activity detection.
- Evaluating the feasibility of the proposed algorithms through time and space complexity analysis.

Following the aforementioned approach, in this paper, an integrated hybrid malware attack detection framework is proposed for enhancing existing security measures in blockchain. The framework encompasses a detailed algorithmic approach integrating SHA-256 and DSA to analyze DDOS attack prevention, reentrancy attack detection, and Byzantine fault tolerance detection into a single cohesive architecture. A hybrid model combining these algorithms, implemented in one block, has been developed to mitigate malicious activity. These measures aim to improve computational complexity and expedite execution within the network of nodes.

The proposed work presented in this paper is organized as: section 2 provides an overview of blockchain technology and examines different types of attacks, as well as their corresponding solutions. This section also provides a summary of many related studies on the subject of developing optimum security measures. In section 3, the proposed framework is described followed by its implementation-related details in section 4. In section 5, various results obtained during the implementation are presented. Major findings are then concluded in section 5.

## 2. Literature review

The recent development of blockchain technology has fundamentally transformed how data is stored and transmitted, providing decentralized and transparent solutions [1]. Blockchain is a secure, immutable ledger database used for digital transactions. It operates on peer-to-peer networks and uses end-to-end encryption to prevent malicious or unauthorized activity [2, 3]. The secure hashing algorithm i.e., SHA-256 is used to add blocks to the blockchain. It is a cryptographic function that converts text into a unique alphanumeric string, known as a hash value. Miners solve a mathematical puzzle called Proof of Work (POW) and Proof of Stake (POS) to create blocks [10].

The decentralized and unchangeable characteristics of blockchain technology have attracted significant attention across various industries. Nevertheless, similar to many groundbreaking technologies, blockchain is susceptible to security flaws [6, 7].

Security attacks on blockchain can target various aspects of the technology, including the underlying consensus mechanisms, smart contracts, and the network itself [7]. In the past few years, various attacks have been identified and analyzed by researchers to further generate a secure mechanism to overcome these attacks. The most common security attacks and their impacts on the blockchain are summarized in table 1 [6, 7, 11, 12].

Understanding the concerns presented in table 1 and executing resilient solutions is crucial for maintaining the integrity and reliability of blockchain-based systems. Over the past few years, researchers have tried to explore the applicability of blockchain technology in a variety of industries [10–16]. They have explored various implementation and security-related issues also that may affect the implementation of blockchain. A lot of secure mechanisms have also been explored by researchers to mitigate these issues [17–27]. To provide a secured blockchain infrastructure, in this section, a thorough analysis has been performed on the existing literature to identify the various security issues and their possible solutions.

An overview of the fundamental ideas behind blockchain technology, including its architecture, consensus processes, and projected advancements, is provided by authors in [1–5]. Additionally, researchers in [6–8, 11] provide a thorough analysis of the many forms of cyberattacks and countermeasures. Authors have also examined the suitability of blockchain technology for many purposes across several sectors, including cryptocurrency [10, 11], decentralized banking [13], secured healthcare [14, 15], wireless communication [16], Government Sector [17], etc. During the implementation of blockchain for different applications, authors have reported different types of security attacks at different phases of the implementation. Furthermore, different solutions have been provided by researchers to overcome these issues. To provide a systematic analysis of security attacks, the work reported by different authors is summarized in table 2. It presents the analysis of the work done in the field of implementing blockchain in different industries, various security issues, and their possible solutions. Here several security challenges arise with blockchain technology and viable strategies to deal with these vulnerabilities are addressed.

Over the past few years, experts investigated thoroughly into numerous security procedures to provide secured blockchain infrastructure as presented in table 2. A secure blockchain infrastructure ensures data integrity and authenticity. Cryptographic techniques are used in blockchain to check the integrity and authenticity of data during transactions. Digital signatures are generally used to encrypt and authenticate the electronic information. A public-key cryptography algorithm called DSA is used to create and authenticate digital signatures [28]. DSA is used in blockchain to create digital signatures that attest to the integrity and authenticity of transactions. A user creates a digital signature with their private key and appends it to the transaction data when they wish to conduct a transaction on the blockchain. The sender's public key and the digital signature that is attached can then be used by other network users to confirm the transaction's integrity and legitimacy [28, 29]. DSA provides non-repudiation and guarantees that transactions cannot be changed or falsified by assuring that only the owner of the private key may provide a legitimate signature. Furthermore, a series of cryptographic hash algorithms called Secure Hash Algorithms (SHA) is intended to generate a fixed-size output, or hash value, from variable-sized input data. Blockchain hashes a wide range of data types, including transactions, blocks, and public keys, using SHA functions [30]. The term 'blockchain' refers to the chain of blocks that are created in a blockchain since each block usually includes a hash of the header from the preceding block. A SHA function, like SHA-256 in Bitcoin, is used to generate this hash. It is computationally impossible to tamper with blockchain data without being detected because of SHA functions, which guarantee that each change to the data being hashed produces an entirely different hash value [31, 32].

A lot of work has been done in literature, where authors have explored SHA and DSA for creating a secured blockchain infrastructure [28–32]. Table 3 summarizes the work focusing on implementing SHA and DSA. It presents an overview of methodologies and applications in blockchain security, focusing primarily on employing the DSA algorithm to establish a framework for public and private key signatures in the blockchain and the implementation of the SHA-256 algorithm to analyze behavior patterns in the blockchain. Data is hashed using SHA to guarantee its integrity and immutability within the blockchain, and digital signatures are created and verified using DSA to guarantee transaction integrity and authenticity. All of these cryptographic building blocks are essential for keeping the blockchain network safe and participants' confidence.

In this section, a thorough analysis has been performed on malware analysis and detection in blockchain technology. A detailed analysis of the literature presents, that blockchain infrastructure is vulnerable to dynamic malware attacks. However, due to its decentralized and unchangeable characteristics, it has a future across various sectors. By addressing these attacks through innovative solutions, robust protocols, and continuous vigilance, blockchain ecosystems can maintain their integrity, security, and trustworthiness in an increasingly interconnected digital landscape. To provide a secured infrastructure, here a hybrid framework is proposed combining SHA-256 and DSA cryptographic algorithms to deal with the Reentrancy, Byzantine fault tolerance

**Table 1.** Security attacks and blockchain and their analysis.

Attack	Description	Impact on blockchain networks	Possible solutions	Application
Byzantine Fault Tolerance Problem [12]	Ability to form some malicious fail node in between the nodes.	It depicts malicious activity in mining blocks, potentially compromising the consensus mechanism.	Implementing consensus algorithms such as ByzCoin, practical Byzantine Fault Tolerance (PBFT) etc and employing cryptographic techniques such as digital signatures and hash algorithms etc.	Bitcoin, Monero, Smart Contract Deployment
Sybil Attack [7]	This attack involves one malevolent actor creating several fictitious identities, or ‘Sybil identities,’ to take over a system or network. In the case of blockchain, it can disrupt the network’s normal operation, manipulate consensus mechanisms, or gain unfair advantages, such as controlling a majority of nodes.	Impairs the validation and decision-making process in the blockchain network, potentially allowing malicious nodes to exert disproportionate influence.	Implementing identity verification mechanisms or reputation systems can help prevent Sybil attacks by requiring participants to authenticate their identities. PoW and PoW are popular approaches to avoid Sybil attack.	General Blockchain Transactions
DDoS Attack [7, 11]	A denial-of-service attack (DDoS) is an intentional attempt to spoof a server, service, or network by flooding it with unsolicited internet traffic. It leverages multiple compromised devices (often a botnet) distributed across the internet to coordinate the attack.	Disrupts blockchain network operations by overwhelming traffic, potentially leading to service unavailability.	Implementing DDoS mitigation techniques such as rate limiting, IP blocking, traffic filtering, and using decentralized hosting solutions can help mitigate the impact of DDoS attacks on blockchain networks.	Bitcoin Transactions, Blockchain Network Operations
Reentrancy Attack [6, 11]	It refers to a vulnerability where a function within a smart contract can be called recursively before the previous calls are completed. This can lead to unexpected behavior and potential security exploits, allowing an attacker to manipulate the contract’s state or drain its funds.	Exploits vulnerabilities in smart contracts, potentially causing financial losses and disrupting contract execution.	Mutual Exclusion Locks are generally used to prevent multiple calls to the same function from occurring at the same time to avoid a reentrancy attack.	Smart Contract Deployment, Decentralized Applications (DApps)
51% Attacks [7]	In Proof of Work blockchains, a 51% attack occurs when a single entity or group controls more than half of the network’s mining power.	It manipulate transactions, double-spend coins, or halt confirmations.	Employing mechanisms such as multi-algorithm consensus, where multiple hashing algorithms are used simultaneously, or transitioning to Proof of Stake consensus, which requires attackers to control a majority of the cryptocurrency to compromise the network, can mitigate the risk of 51% attacks.	Bitcoin Mining, Blockchain Network Consensus Mechanism

**Table 2.** Blockchain implementation and related security issues in different applications.

References	Description	Methodology
[18] (2024)	Presents a novel approach to improve cloud computing security through the integration of lightweight blockchain technology and encryption techniques.	The lightweight design of the blockchain implementation addresses issues with performance overhead and scalability. This entails optimizations including utilizing effective consensus procedures and storing transaction data in a distributed hash table (DHT).
[19] (2024)	Smart contract reentrancy vulnerabilities have the potential to cause significant security lapses as well as monetary losses. This paper focuses on discovering possible attacker contracts to detect vulnerable reentrancy issues in smart contracts.	The paper describes methods for identifying contracts that are attackers based on how they behave and interact with vulnerable contracts. To find unusual activity suggestive of reentrancy attacks, may entail examining transaction data, contract interactions, and transaction histories.
[20] (2023)	This study offers a comprehensive analysis of how malware has taken advantage of blockchain technology. It looks at how secret means of communication are used in secure settings.	For Ethereum smart contracts, a mechanism known as Smart-Zephyrus is being developed to allow for covert channel communication. A large number of real-world smart contracts are used, and a public proof of concept is made accessible as an open-source project, to preserve the confidentiality of the information.
[21] (2023)	It focuses on ransomware attacks in the healthcare industry, in which malevolent parties encrypt private patient information and demand payments in exchange for the key. Serious hazards to patient privacy, data integrity, and healthcare operations are presented by these attacks.	A blockchain-based system designed especially for digital healthcare contexts is presented in this study. This architecture probably consists of elements like ransomware detection and mitigation, immutable audit trails, secure data storage, and access control mechanisms.
[22] (2023)	It focuses on the secure and efficient sharing of electronic medical records (EMRs) using blockchain technology while mitigating the risk of malicious propagation.	The article addresses particular methods or algorithms intended to identify and reduce the spread of harmful software or unapproved changes inside the blockchain network. To guarantee the integrity of shared EMRs, this involves consensus-based validation techniques, signature-based detection, or anomaly detection.
[23] (2022)	Avatar-based metaverse environments are becoming more popular as problem-solving tools. Although Roblox, Minecraft, and Fortnite are platforms that provide services, they are susceptible to security breaches.	This work suggests a secure communication system paradigm for metaverse environments that makes use of reciprocal authentication via 'Elliptic Curve Cryptography (ECC)' and biometric data. The 'Burrows-Abadi-Needham (BAN)' logic, the 'Real-or-Random (ROR)' model, the 'Automated Validation of Internet Security Protocols and Applications (AVISPA)' system, and informal security analysis all serve to illustrate the security of the method.
[24] (2022)	Cryptocurrency mining is carried out using miner malware, commonly referred to as cryptocurrency mining hijacking assaults, which take advantage of the victim's computer resources to mine cryptocurrency.	It presents MBGNet, method that uses graph neural networks (GNNs) and behavior patterns to identify bitcoin mining malware. It examines the characteristics of the function call and control flow graphs, extracts connection features from crucial nodes, and converts those features into feature vectors.
[25] (2022)	Traditional ways of authenticating and authorizing (A&A) people in smart grids may have flaws like single points of failure, not being clear, and being easy to hack.	Using a consortium blockchain approach, the A&A protocol is developed and deployed on the FISCO and Hyperledger platforms. Moreover, smart contracts are maintained and recorded via patching techniques.
[26] (2021)	In this paper, the predominance of distributed software systems—composed of linked components operating on several network nodes—is discussed. These systems are vulnerable to malware infections, insider assaults, and illegal access, among other security risks.	Accurately differentiating between malicious and normal behavior is one of the obstacles in detecting assaults in distributed software systems, along with the complexity of system interactions and the diversity of attack paths. This research suggests addressing these issues by incorporating blockchain technology into stand-alone behavior-based attack detection methods.
[27] (2018)	It examines the safety features of blockchain technology with a focus on finding and reducing the risks connected with 51% attacks.	This study uses anomaly detection methods, transaction histories, and network hash rate distribution monitoring to find unusual behavior suggestive of a possible 51% attack. Enhancements to the consensus method are also included, encouraging decentralization and boosting network security with features like checkpointing and delayed finality.

**Table 3.** DSA-SHA-based cryptographic solutions for Data Security in Blockchain Infrastructure.

References	Methodology	Analysis
[28] (2023)	The focus of this paper is on new developments in cryptocurrency wallet technology, especially the addition of digital signatures to improve security and usability.	The study explores the function of digital signatures in Bitcoin transactions. Digital signatures enable wallets to securely authorize and protect transactions against tampering. In addition, the article explores sophisticated cryptographic methods like multi-signature (multisig) wallets and hierarchical deterministic (HD) wallets, which provide enhanced security and flexibility.
[29] (2023)	The paper investigates how cryptographic algorithms can be used to enhance malware detection techniques.	The paper evaluates the use of cryptography algorithms in malware detection, with a focus on multi-signature hashes and key-delivery algorithms.
[30] (2023)	This paper presents an analysis of the digital signature and hash algorithms used in the Bitcoin and Ethereum blockchain networks.	The study explores the utilization of digital signature algorithms, specifically Elliptic Curve Digital Signature Algorithm (ECDSA), in Bitcoin and Ethereum to authenticate and verify transactions. It also implemented cryptographic hash functions, specifically SHA-256 (Secure Hash Algorithm 256-bit) in Bitcoin and Keccak-256 (a derivative of SHA-3) in Ethereum.
[31] (2022)	Hash functions are extremely useful and can be found in almost all applications that deal with information security; consequently, hashing techniques are the most secure of the available options.	Here SHA-512, a novel hash-based technique is suggested for blockchain infrastructure, which operates on 1024-bit blocks and employs 64-bit words. Authors demonstrated that SHA 512 is more resistant to collisions than its predecessor.
[32] (2019)	Conventional methods for finding malware often rely on signature-based detection, which checks files that might be malicious against a database of known forms of malware.	This paper proposes a blockchain-based malware detection method that utilizes shared signatures of suspected malware files, enabling users to respond quickly to increasing threats.

detection, and Distributed Denial of Service (DDoS) attacks in a combined way. A detailed discussion of the proposed approach and its implementation is presented in further sections.

### 3. Proposed integrated hybrid malware attack detection algorithm

This section presents the architecture and algorithms that are suggested to assess blockchain attacks by combining SHA-256 hashing algorithms with digital signatures to improve the security of blockchain transactions. Figure 1 shows the suggested three-layered architectural paradigm for attack analysis inside a blockchain framework.

- **Input Layer:** The topmost layer outlines the components necessary to form a block within the blockchain. It includes elements such as the block number, nonce, timestamp, the hash value of the previous and current block, and references to Proof of Work (PoW) and Proof of Stake (PoS) mechanisms.
- **Security Layer:** The middle layer specifies the core security features of the blockchain. It includes the SHA-256 hashing algorithm, which encrypts the block data with a 64-bit output. This layer also encompasses digital signatures, which verify the authenticity and integrity of transactions. Additionally, it highlights the use of public and private keys, which are essential components of blockchain cryptography.
- **Data Layer:** The bottom layer represents the data types or entities managed within the blockchain. It consists of transactions, which are the fundamental operations conducted on the blockchain, Bitcoin (or other cryptocurrencies), which may be the subject of the transactions; personal documents, which can be encrypted and stored or transferred securely; and private information, which underscores the need for security and privacy in the blockchain network.

The combination of SHA-256 and DSA is then deployed to detect and mitigate various malware attacks, including Byzantine fault tolerance, reentrancy attacks, and DDOS attacks. The process of detection of the aforementioned attacks is further summarized in detail.

#### 3.1. Byzantine fault tolerance detection

The Byzantine Fault Tolerance Detection algorithm involves acquiring messages for attacked blocks, making decisions based on problem checks, and implementing a blocking mechanism. The algorithm detects and prints

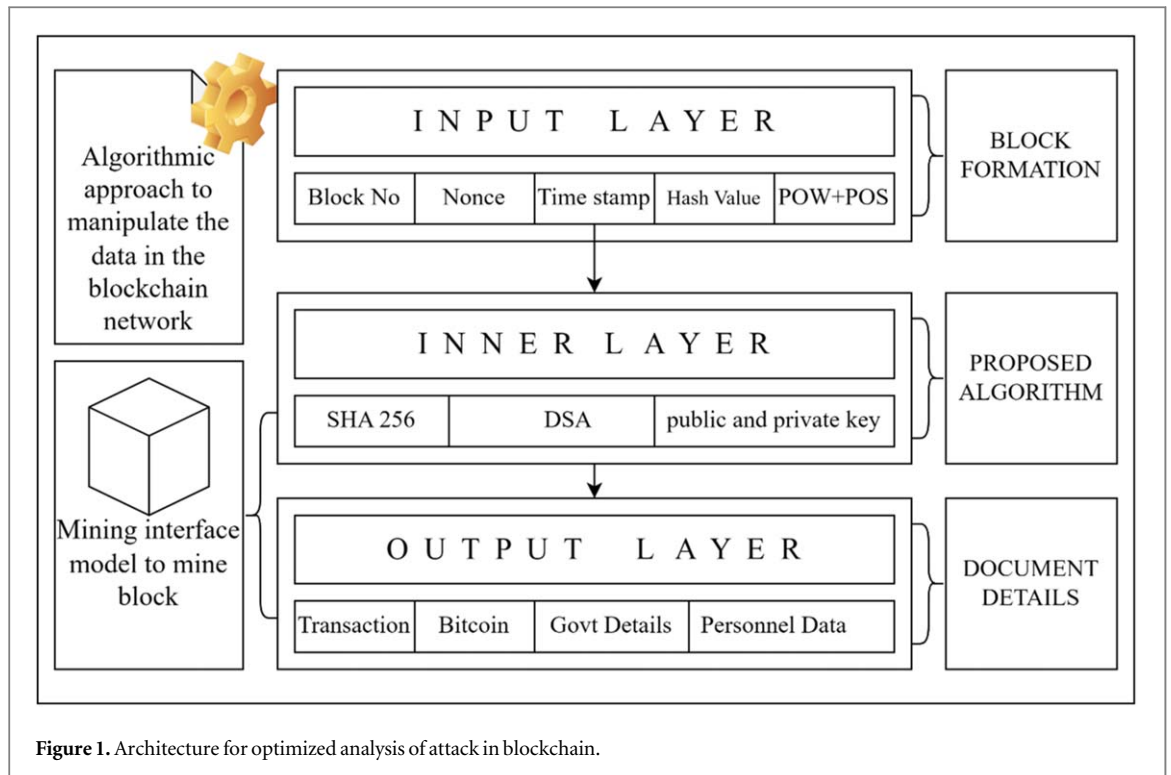


Figure 1. Architecture for optimized analysis of attack in blockchain.

whether the block is attacked for each block, utilizing a comprehensive process to address Byzantine fault tolerance within a blockchain system. Figure 2 presents the proposed algorithm for byzantine fault tolerance detection in blockchain networks.

### 3.2. Reentrancy attack detection

The Reentrancy Attack Detection algorithm involves the creation of fake files by the attacker to manipulate smart contracts. The attacker contracts with a bank account, establishes a code for looping to receive Bitcoin, and creates a fallback function for bank statement commands. The execution includes calling the fallback function, initiating the attack command, and retrieving the balance through a public function. This comprehensive approach aims to detect and prevent reentrancy attacks in blockchain systems. Figure 3 represents the proposed approach for reentrancy attack detection in a blockchain Network.

### 3.3. DDOS attack detection

The DDOS Attack Detection algorithm employs a multi-step mechanism. It detects denial of service attacks by scrutinising payload drops and takes actions such as adding source IP addresses to blacklists or graylists for effective defence. The algorithm efficiently handles various scenarios, offering robust protection against DDOS attacks in blockchain systems. Figure 4 presents the adopted approach for DDOS attack detection in blockchain networks.

The proposed approaches presented here, present an integrated approach to deal with byzantine fault tolerance, reentrancy attack, and DDOS attacks through a secured mechanism combining SHA-256 and DSA. It has the capability of providing a robust defense against various malware threats in blockchain systems.

## 4. Implementation

In this section, a detailed discussion on the implementation of the above-proposed framework and various algorithms for attack analysis is thoroughly discussed. The basic steps involved during the implementation involve data preprocessing, feature extraction, and integration of multiple detection methods. A step by step procedure of implementation is presented in Algorithm 1 and is analyzed further.

The implementation of the mentioned hybrid malware detection approach which combines various techniques like anomaly detection, signature-based detection, and behavior analysis is executed in Python. To check the efficacy of the proposed approach, the algorithm is tested on the NSL-KDD dataset [33, 34]. It consists of several features related to network connections, which are indicators of normal or malicious activity (e.g.,



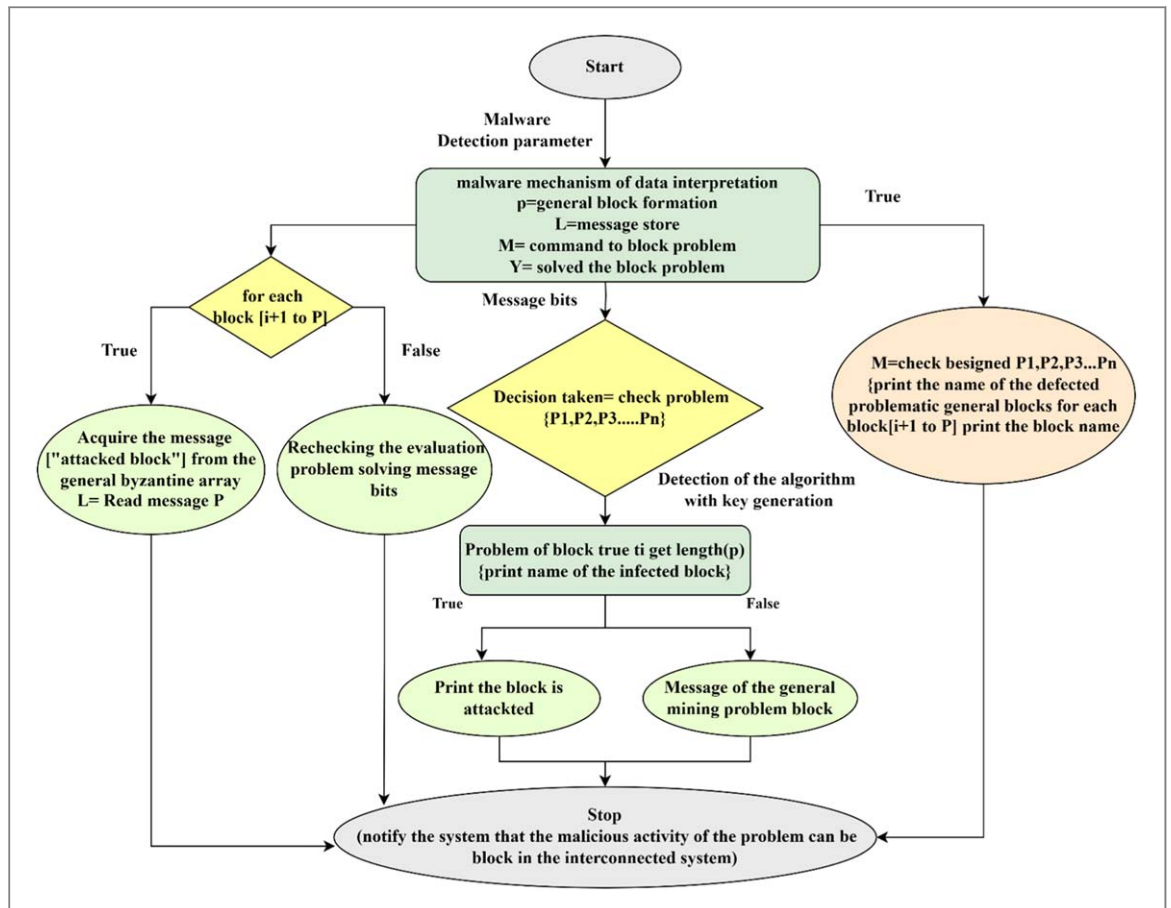


Figure 2. Approach for Byzantine fault tolerance detection in blockchain network.

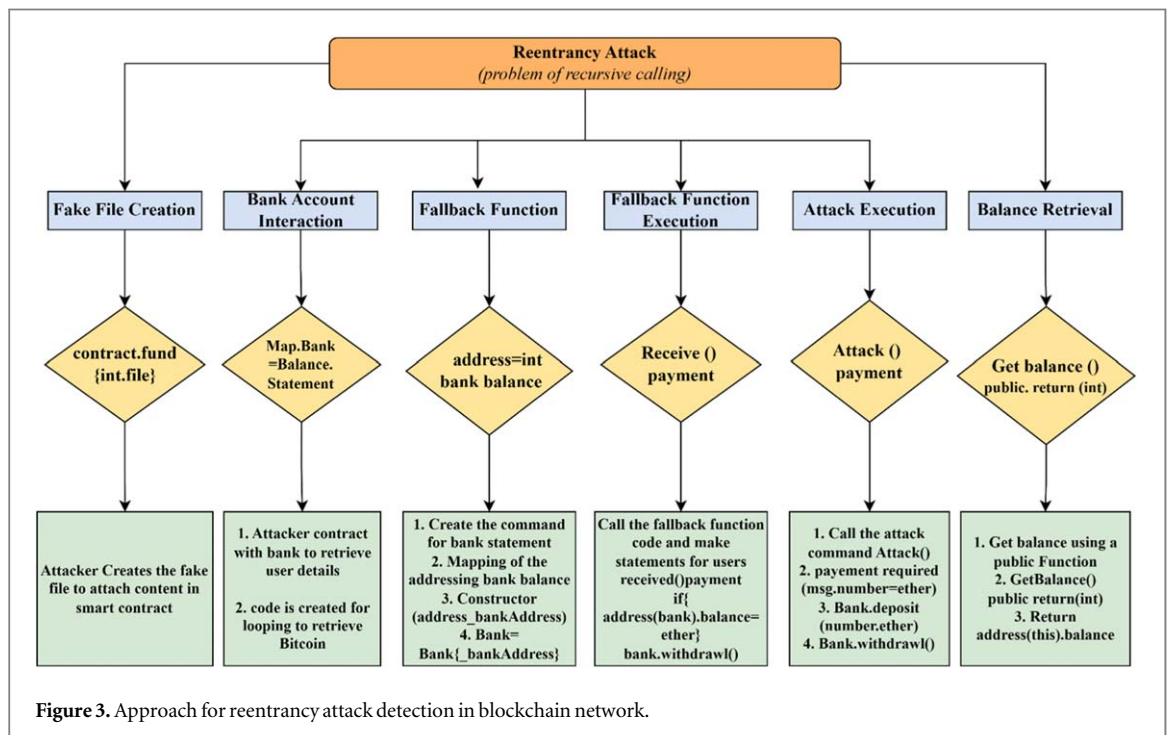
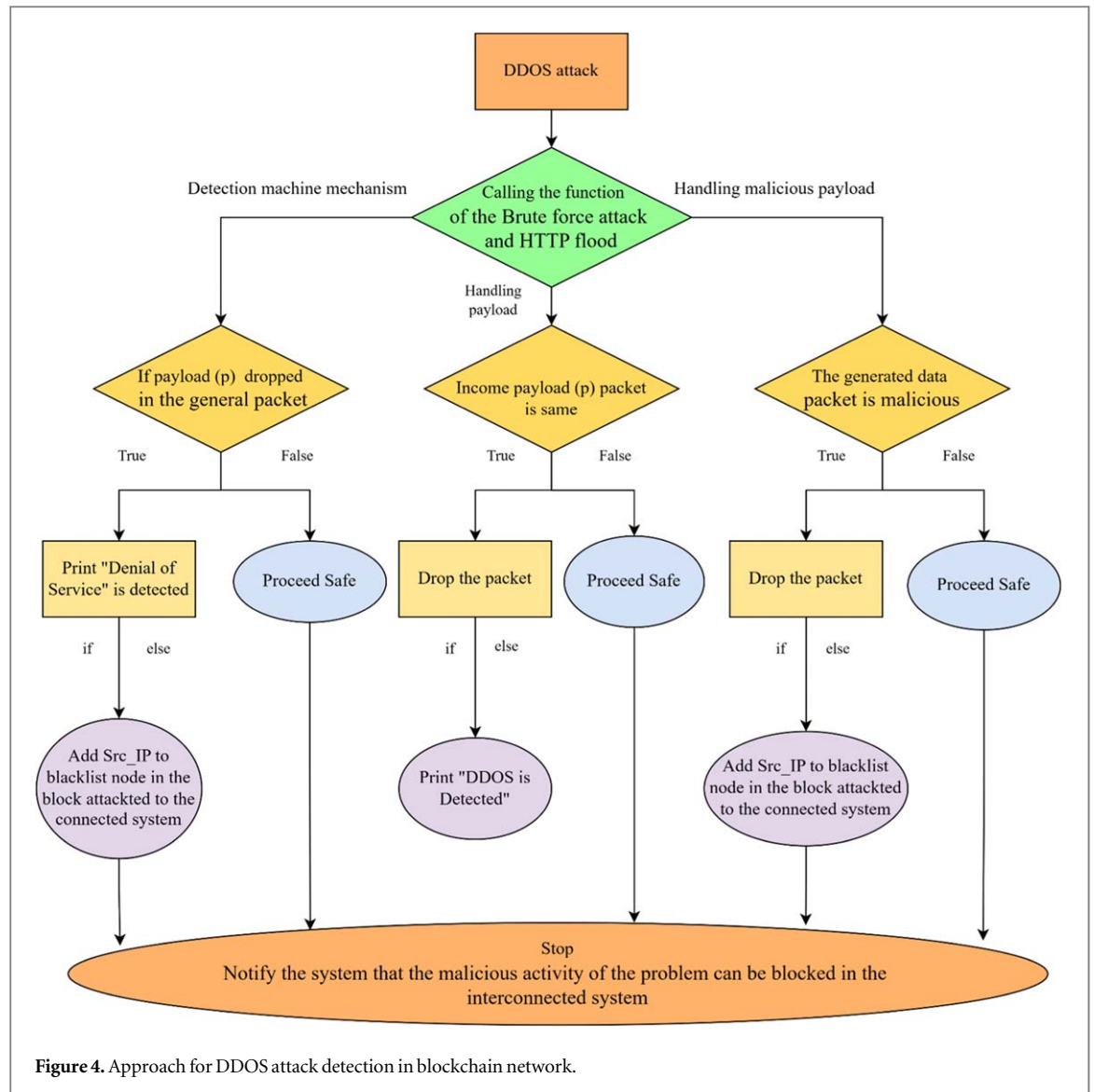


Figure 3. Approach for reentrancy attack detection in blockchain network.

duration, protocol, service, flag, src\_bytes, dst\_bytes, various rates like dst\_host\_same\_srv\_rate, and a target variable indicating the class of activity, such as normal, Neptune, saint, scan).

The NSL-KDD dataset is an improved version of the KDD'99 dataset designed to solve some inherent problems, such as redundant records in train and test sets. The NSL-KDD dataset includes a set number of



selected records, ensuring that the results from different research works are consistent and comparable. The dataset can be accessed from the NSL-KDD repository, which provides files in both ARFF and TXT formats for training and testing purposes.

The dataset comes with predefined training (KDDTrain+.txt) and testing sets (KDDTest+.txt), which were utilized in this study to ensure comparability with other research in the field. The training set (KDDTrain+.txt) and testing set (KDDTest+.txt) are used as provided, with no further random subdivision in our experiments. This choice is based on the need to maintain the integrity of comparative evaluations across various studies using the same dataset. Although the NSL-KDD dataset is pre-split, further cross-validation within the training set for internal experiments was controlled using a fixed random\_state parameter, set to a specific value (random\_state = 42). This approach ensures that subset generation within the dataset is reproducible by other researchers. A Python code snippet for data loading and splitting is given below.

```

from sklearn.model_selection import train_test_split
# Load your dataset
X, y = load_data('KDDTrain+.txt')
# Splitting the dataset into training and validation sets
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size = 0.3, random_state = 42)

```

The algorithm, employing techniques for Byzantine fault tolerance, reentrancy attack detection, and DDOS attack mitigation, demonstrates a robust framework for safeguarding blockchain systems. Using multiple detection strategies enhances the comprehensive defense mechanism against diverse malware attacks. The architectures (figure 1) and flowcharts (figures 2–4) outline the structured approach to tackling specific vulnerabilities in blockchain, which helps pinpoint and respond to security breaches effectively.

**Algorithm 1.** Proposed Hybrid Malware Detection System.**Input:** Dataset with features and labels**Output:** Prediction results**Steps:****Dataset and Model Training:**

- **Dataset Composition:** This consists of network connection features categorised as normal or malicious activities.
- **Feature Details:** Includes key indicators such as duration, protocol type, service type, and byte counts.
- **Dataset Split:**
  - *Training set:* 70% of the data is used for model training.
  - *Testing set:* 30% of the data is reserved for evaluating the model's performance.
  - *Purpose of Split:* Ensures sufficient data for effective model learning and provides an ample subset for unbiased performance testing.

**Feature Extraction:**

- Implement feature extraction method (in this case, the method simply returns the data without modification).

**Define Hybrid Detection Model:**

- Initialize the model with a dictionary containing instances of **RandomForestClassifier**.
- Define a training method that iterates through the models, training each on the dataset.
- Define a prediction method that aggregates predictions from each model. For each instance in the test set, predictions from all models are averaged to form a final prediction.

**Execution:**

- Load NSL KDD dataset [33].
- Preprocess the data to obtain scaled features and labels.
- Split the dataset into training and testing sets using `train_test_split`.
- Initialize the hybrid model.
- Train the model using the training data.
- Generate predictions on the testing set.
- Evaluate the model by comparing the predictions to the true labels, printing out a classification report.

**End of Algorithm**

## 5. Analysis

To analyze the feasibility of the proposed framework, the time-space complexity of the framework is performed for different types of attacks and a combined accuracy is also extracted. Furthermore, security analysis is also performed to check the correctness and soundness of the framework.

### 5.1. Performance analysis

The integrated hybrid malware attack detection algorithm exhibits commendable outcomes across three key malware attacks.

For Byzantine Fault Tolerance Detection, the proposed algorithm efficiently acquires messages from Byzantine blocks ( $O(P)$ ), makes decisions based on Byzantine problems ( $O(n)$ ), and blocks besieged blocks ( $O(P)$ ). It successfully detects Byzantine attacks, offering detailed information about attacked blocks. The overall time complexity is  $O(P + n)$ , and the space complexity is  $O(P)$ . For Reentrancy Attack Detection, the proposed algorithm with constant time complexities ( $O(1)$ ) for creating fake files, contracting with the bank, and executing fallback functions, the algorithm successfully identifies reentrancy attacks, providing insights into balance retrieval and attack execution. Both time and space complexities are  $O(1)$ . Finally, DDOS Attack Detection, it swiftly detects denial of service attacks ( $O(1)$ ), efficiently handles the same payload ( $O(1)$ ), and manages malicious payloads ( $O(1)$ ). It identifies and handles DDOS attacks promptly, managing multiple types of nodes simultaneously. The overall time and space complexities are  $O(1)$ .

In an integrated hybrid malware detection algorithm,  $O(P)$  and  $O(n)$  are used to describe the time complexity of the proposed algorithms in terms of the number of operations required relative to the size of the input data.  $O(P)$  of operations is linear concerning the number of blocks ( $P$ ) being processed or analyzed. It refers to the part of the Byzantine Fault Tolerance Detection where each block within the blockchain is individually assessed to determine if it has been compromised.  $O(n)$  of the operations scales linearly with the number of decision checks or messages ( $n$ ) being handled. It refers to processes that involve multiple decision-making steps & transactions within the system based on certain criteria that are evaluated one at a time.

The efficiency and performance of the proposed algorithm is calculated as

Overall Efficiency and Performance:

*Efficiency:*

*Time Complexity:*  $O(P + n + 3)$  - Sum of time complexities from each component.

*Space Complexity:*  $O(P + 2)$  - Sum of space complexities from each component.

*Performance Parameters:*

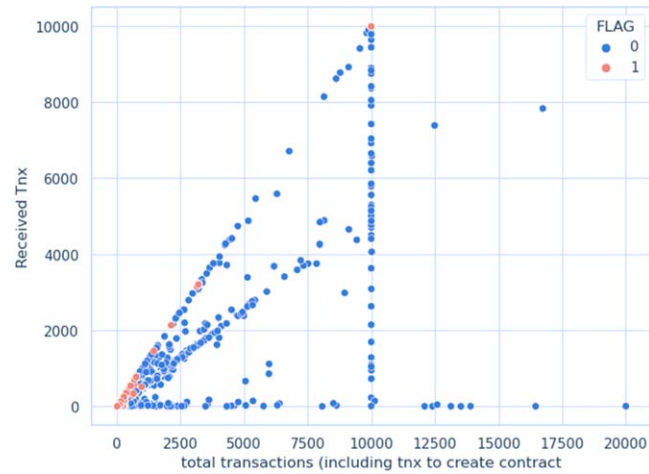


Figure 5. Scatter plot representing fully requested successful transactions and malicious transactions.

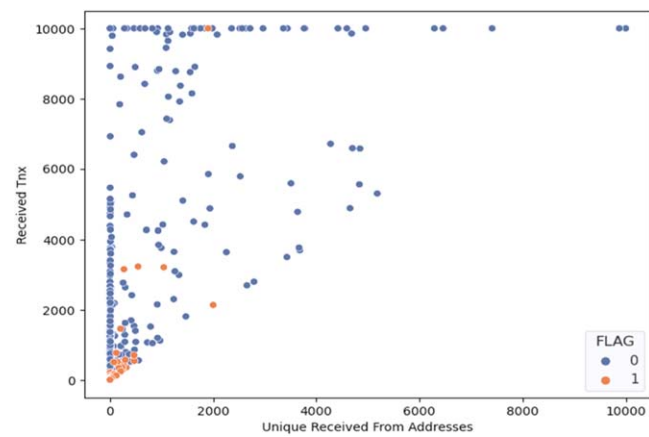


Figure 6. Scatter plots representing behavior of transaction.

*Time Complexity:*  $O(P + n)$

*Space Complexity:*  $O(P)$

Overall, the integrated algorithm provides a holistic defense against Byzantine, reentrancy, and DDOS attacks, efficiently managing and analyzing different types of malware. The combined time complexity is  $O(P + n + 3)$ , and the space complexity is  $O(P + 2)$ .

Furthermore, the analysis of the proposed approach for detecting the malicious activities is done through random forest classifier as discussed in Algorithm 1. The analysis of the transactional database through the Random Forest Classifier to detect malicious activity can be visualized through the scatter plots presented in figures 5 and 6. These figures demonstrate the analysis of fraudulent files designed to compromise the entire system. In figure 5, 0 represents transactions that are not fully requested and detected as a malicious file in the block, and 1 represents the successful transactions in the block network. Figure 6 represents the behavior analysis of each transaction i.e., a request to generate a block to add that block to the blockchain network. Here, 0 represents a situation of successfully authenticated blocks and the addition of blocks in the blockchain network, whereas 1 represents the malicious nodes who could not pass the authentication process of blocks.

For a comprehensive evaluation of performance various metrics such as precision, recall, and F1-score are considered. These metrics are essential for evaluating the algorithm's effectiveness in identifying and mitigating malware attacks within blockchain systems. Various formulas used for their calculation are given below:

**Accuracy:** This metric evaluates the overall correctness of the model across all predictions.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

In this study, it is applied as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Where TP is true positive, TN is true negative, FP is false positive, and FN is false negative.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

The generated results show that the proposed algorithm achieves an accuracy of 73.33% with the following other metrics.

*Precision: 64.29%*

*Recall: 73.33%*

*F1-Score: 68.03%*

*Execution Time: Approximately 0.99 s*

## 5.2. Security analysis

To perform a security analysis on the proposed Integrated Hybrid Malware Attack Detection Algorithm based on its correctness and soundness, including mathematical derivations, we focus on the key components of the algorithm: Byzantine Fault Tolerance Detection, Reentrancy Attack Detection, and DDOS Attack Detection. Each component's effectiveness is measured through its operational complexity, and the algorithm's overall efficiency and performance are quantified in terms of time and space complexity.

### 5.2.1. Correctness

*Byzantine Fault Tolerance Detection* operates with a complexity of  $O(P)$  for acquiring messages and  $O(n)$  for decision-making, ensuring correct identification of Byzantine attacks.

*Reentrancy Attack Detection* exhibits constant complexity  $O(1)$  across its operations, indicating it correctly identifies attacks without scaling issues.

*DDOS Attack Detection* also demonstrates constant complexity  $O(1)$ , correctly identifying and handling attacks efficiently.

### 5.2.2. Soundness

The integrated approach combines these mechanisms to cover a broad spectrum of attacks with the overall time complexity of  $O(P+n+3)$  and space complexity of  $O(P+2)$ , showcasing the algorithm's comprehensive coverage and soundness in defending against diverse malware threats.

## 6. Conclusion

This paper proposes and deploys a comprehensive hybrid solution to detect and mitigate various malware attacks, including Byzantine fault tolerance, reentrancy attacks, and DDOS attacks. It presents a comprehensive defense mechanism against these diverse cyber threats. The algorithm demonstrates efficiency through linear time complexities ( $O(P + n)$ ) for Byzantine fault tolerance, constant time complexities ( $O(1)$ ) for reentrancy attack detection, and DDOS attack detection components. The Byzantine fault tolerance module efficiently acquires and analyses messages from Byzantine blocks, providing detailed insights into attacked blocks with a linear time complexity proportional to the block size ( $\backslash(P\backslash)$ ). Simultaneously, the reentrancy and DDOS attack detection components operate with constant time complexities, swiftly identifying and handling respective attacks. The amalgamation of these components yields a holistic defence system with promising outcomes for real-time malware detection and prevention. The overall performance parameters reflect the algorithm's effectiveness in managing time and space complexities, offering a balanced trade-off between computational efficiency and resource utilization. The algorithm showcases a robust defense strategy suitable for complex and dynamic cyber landscapes by addressing Byzantine, reentrancy, and DDOS attacks within a unified framework. Its scalability and adaptability make it a valuable contribution to the field of cybersecurity, providing a versatile solution for safeguarding distributed systems and blockchain networks against multifaceted malware threats.

## Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

## Declarations

## Funding

Not Applicable.

## Conflicts of interest/Competing interests

There is no conflict of interest.

## Ethics approval

Not Applicable.

## Consent to participate

I consent to participate.

## Consent for publication

I consent to publish.

## Availability of data and material

Publicly available NSL KDD Dataset.

## Code availability

Custom code.

## Credit author statement

**Aastha Sharma:** Conceptualization, Methodology, Software, Data curation, Validation, Writing- Original draft preparation.

**Divya Upadhyay:** Conceptualization, Methodology, Supervision, Reviewing and Editing.

**Shanu Sharma:** Conceptualization, Methodology, Supervision, Reviewing and Editing.

## ORCID iDs

Divya Upadhyay  <https://orcid.org/0000-0003-3664-7415>

Shanu Sharma  <https://orcid.org/0000-0003-0384-7832>

## References

- [1] Zheng Z, Xie S, Dai H, Chen X and Wang H 2017 An overview of blockchain technology: architecture, consensus, and future trends. *2017 IEEE Int. Congress on Big Data (BigData Congress)* (IEEE) **557–64**
- [2] Lakhani I K 2017 The truth about blockchain *Harv. Bus. Rev.* **95** 118–27
- [3] Purwono P, Ma'arif A, Rahmani W, Haq Q M ul, Herjuno D and Naseer M 2022 Blockchain technology *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika* **8** 199
- [4] Tripathi G, Ahad M A and Casalino G 2023 A comprehensive review of blockchain technology: underlying principles and historical background with future challenges *Decision Analytics Journal* **9** 100344

- [5] Sarwar M I, Maghrabi L A, Khan I, Naith Q H and Nisar K 2023 Blockchain: a crypto-intensive technology—a comprehensive review *IEEE Access* **11** 141926–55
- [6] Dave D, Parikh S, Patel R and Doshi N 2019 A survey on blockchain technology and its proposed solutions *Procedia Computer Science* **160** 740–5
- [7] Anita N and Vijayalakshmi M 2019 Blockchain security attack: a brief survey *10th Int. Conf. on Computing, Communication and Networking Technologies (ICCCNT)* (IEEE) **1–6**
- [8] Sherman A T, Javani F, Zhang H and Golaszewski E 2019 On the origins and variations of blockchain technologies *IEEE Security & Privacy* **17** 72–7
- [9] Barde S 2022 Blockchain and cryptocurrencies *Emerging Computing Paradigms* ed U Singh et al (<https://doi.org/10.1002/9781119813439.ch13>)
- [10] Li S-N, Yang Z and Tessone C J 2020 Mining blocks in a row: a statistical study of fairness in Bitcoin mining *2020 IEEE Int. Conf. on Blockchain and Cryptocurrency (ICBC)* (<https://doi.org/10.1109/icbc48266.2020.9169436>)
- [11] Sheela S, Shalini S, Harsha D, Chandrashekar V T and Goyal A 2023 Decentralized malware attacks detection using blockchain *ITM Web of Conferences* **53** 03002
- [12] Suliayanti W N and Sari R F 2023 Blockchain-based double-layer byzantine fault tolerance for scalability enhancement for building information modeling information exchange *Big Data and Cognitive Computing* **7** 90
- [13] Ozili P K 2022 Decentralized finance research and developments around the world *Journal of Banking and Financial Technology* **6** 117–33
- [14] Chaturvedi S 2023 IoT-based secure healthcare framework using blockchain technology with a novel simplified swarm-optimized bayesian normalized neural networks *International Journal of Data Informatics and Intelligent Computing* **2** 63–71
- [15] Tyagi P and Manju Bargavi S K 2023 Using federated artificial intelligence system of intrusion detection for IoT healthcare system based on blockchain *International Journal of Data Informatics and Intelligent Computing* **2** 1–10
- [16] Chandan R R, Balobaid A, Cherukupalli N L S, Gururaj H L, Flammmini F and Natarajan R 2023 Secure modern wireless communication network based on blockchain technology *Electronics* **12** 1095
- [17] Meirobie I, Irawan A P, Sukmana H T, Lazirkha D P and Santoso N P L 2022 Framework authentication e-document using blockchain technology on the government system *International Journal of Artificial Intelligence Research* **6**
- [18] Shrivastava P, Alam B and Alam M 2024 A hybrid lightweight blockchain based encryption scheme for security enhancement in cloud computing *Multimed Tools Appl* **83** 2683–702
- [19] Yang S, Chen J, Huang M, Zheng Z and Huang Y 2024 Uncover the premeditated attacks: detecting exploitable reentrancy vulnerabilities by identifying attacker contracts in *2024 IEEE/ACM 46th Int. Conf. on Software Engineering (ICSE)* 912
- [20] Gimenez-Aguilar M, de Fuentes J M and Gonzalez-Manzano L 2023 Malicious uses of blockchains by malware: from the analysis to smart-zephyrus *Int. J. Inf. Secur.* **22** 1445–80
- [21] Lakhan A, Thinnukool O, Groenli T M and Khuwuthyakorn P 2023 RBEF: ransomware efficient public blockchain framework for digital healthcare application *Sensors* **23** 5256
- [22] Lin C, Huang X and He D 2023 Efficient blockchain-based electronic medical record sharing with anti-malicious propagation *IEEE Trans. Serv. Comput.* **16** 3294–304
- [23] Ryu J, Son S, Lee J, Park Y and Park Y 2022 Design of secure mutual authentication scheme for metaverse environments using blockchain *IEEE Access* **10** 98944–58
- [24] Zheng R, Wang Q, He J, Fu J, Suri G and Jiang Z 2022 Cryptocurrency mining malware detection based on behavior pattern and graph neural network *Security and Communication Networks* **2022** 1–8
- [25] Zhong Y, Zhou M, Li J, Chen J, Liu Y, Zhao Y and Hu M 2021 Distributed blockchain-based authentication and authorization protocol for smart grid *Wireless Communications and Mobile Computing* **2021** 1–15
- [26] Aljihani H, Eassa F, Almarhabi K, Algarni A and Attaallah A 2021 Standalone behaviour-based attack detection techniques for distributed software systems via blockchain *Applied Sciences* **11** 5685
- [27] Ye C, Li G, Cai H, Gu Y and Fukuda A 2018 Analysis of security in blockchain: case study in 51%-attack detecting *2018 5th Int. Conf. on Dependable Systems and Their Applications (DSA)* 2018 15–24
- [28] Ji P 2023 The advance of cryptocurrency wallet with digital signature *Highlights in Science, Engineering and Technology* **39** 1098–103
- [29] Krishna M and Manjunath C R 2023 Investigating the role of applied cryptography algorithms for malware detection In *2023 2nd Int. Conf. on Futuristic Technologies (INCOFT)* 1–5 (IEEE)
- [30] Liu J 2023 Digital signature and hash algorithms used in bitcoin and ethereum In *Third Int. Conf. on Machine Learning and Computer Application (ICMLCA 2022)* Vol. 12636, 1302–21 (SPIE)
- [31] Davda Y 2022 Design of hash algorithm for blockchain security *Blockchain Applications in Cryptocurrency for Technological Evolution* ed A Taghipour (IGI Global) **118–35**
- [32] Fuji R, Usuzaki S, Aburada K, Yamaba H, Katayama T, Park M, Shiratori N and Okazaki N Blockchain-based malware detection method using shared signatures of suspected malware files *Advances in Networked-based Information Systems. NBIS - 2019 2019. Advances in Intelligent Systems and Computing* vol 1036 ed L Barolli et al (Springer) ([https://doi.org/10.1007/978-3-030-29029-0\\_28](https://doi.org/10.1007/978-3-030-29029-0_28))
- [33] Tavallae M, Bagheri E, Lu W and Ghorbani A A 2009 A detailed analysis of the KDD CUP 99 data set *IEEE Symp. on Computational Intelligence for Security and Defense Applications* (IEEE) **1–6**
- [34] 2024 KDD Cup 1999 Data. (1999, October 28). Retrieved March 15, 2024, from <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>