Blockchain-based Deep Learning Algorithm for Detecting Malware

Dmytro Denysiuk[a], Olena Geidarova[a], Mariia Kapustian[a], Sergii Lysenko[a], Anatoliy Sachenko[b,c]

[a] *Khmelnytskyi National University, Instytutska Str., 11, Khmelnytskyi, Ukraine*
[b] *Western Ukrainian National University, Ternopil, Ukraine*
[c] *Kazimierz Pulaski University of Technology and Humanities in Radom, Poland*

Abstract

Nowadays malware detection is a very important task in information security. Criminals are constantly looking for new ways to attack computer networks and systems, so it is important to have a reliable mechanism to detect and prevent these threats.

Existing anti-virus programs are not always effective in detecting new types of viruses or malware, which can compromise your system and steal important information. Therefore, it is critically important to investigate and create new methods for detecting malicious software, especially using modern technologies like Blockchain.

One of the ways to detect malicious software is to use deep learning algorithms. For this, a Deep Learning Algorithm using Blockchain technology was developed to detect malicious programs. The basic idea is to use the blockchain to ensure security and accuracy of malware detection.

The algorithm proposed in this work is based on the subsystems and Proof-of-Action. The first mechanism provides parallel analysis of potentially dangerous software by different participants. The second mechanism is used to validate the results of analysis and increase the accuracy of detection.

The application of the proposed approach allows to detect malicious software with an accuracy of from 98.81% to 99.33%, which is quite a high result.

Keywords

malware, malware detection, cybersecurity, Blockchain, Proof-of-Action

1. Introduction

Cyber threats have continued to grow over the past three years, especially with the widespread adoption of video conferencing platforms, telecommuting and other remote work solutions. According to a study by Risk Based Security[1], more than 29 billion data records were stolen in cyberattacks in 2021, that is the highest number on record.

At the beginning of 2022, cybercrime has increased significantly, posing a threat to business and other sectors of the economy. A study by McAfee[2] showed that the number of cyber attacks on enterprises increased by 146% compared to the previous year. Most of these attacks target large companies, but small and medium-sized businesses also fall victim to cybercriminals.

IntelITSIS'2023: 4th International Workshop on Intelligent Information Technologies and Systems of Information Security, March 22–24, 2023, Khmelnytskyi, Ukraine
EMAIL: denysiuk@khmnu.edu.ua (D.Denysiuk); geydarova@ukr.net (O.Geidarova); kapustian.mariia@gmail.com (M.Kapustian); sirogyk@ukr.net (S.Lysenko); as@tneu.edu.ua (Anatoliy Sachenko)
ORCID: 0000-0002-7345-8341 (D. Denysiuk); 0000-0002-7253-893X (O. Geidarova); 0000-0001-9200-1622(M.Kapustian); 0000-0001-7243-8747 (S. Lysenko); 0000-0002-0907-3682 (Anatoliy Sachenko)

Cyberattacks have used a variety of methods, including phishing, spreading malware via email and social media, and using cryptocurrencies to demand ransom.

In 2020, SonicWall[3] registered 2.9 billion cyberattacks, which is 18% more than in 2019. Most of these attacks targeted small and medium-sized businesses, as well as home computers.

In 2019, a study by Symantec[4] noted that more than 60% of cyber attacks were aimed at small and medium-sized businesses. In addition, research has shown that criminals are increasingly using social engineering techniques, such as phishing attacks, to gain access to sensitive information.

According to Symantec's "Internet Security Threat Report" [5], the number of new vulnerabilities used by hackers increased by 60% in 2021 compared to 2019.

Also, according to Cybersecurity Ventures [6], it is predicted that by 2025 spending on cyber security will reach $10.5 trillion.

Thus, the problem of cyber security is very urgent and requires constant attention, research, solutions and investment. And malware detection is a very important task in information security. Criminals are constantly looking for new ways to attack computer networks and systems, so it is important to have a reliable mechanism to detect and prevent these threats.

2. Related works

Different approaches to detecting malicious programs are widely described in scientific resources. For example, the classification of malware using user feedback is described in [7], but this approach leads to an increase in the number of false positives in the case of confidential resources.

Another approach proposed in [8] combines permissions and intents, which are supplemented by some stages of classifiers like decision trees, multi-level perceptrons, and decision tables. These stages combine the help of three schemes: determination of the average value of probabilities, product of probabilities and majority voting.

In [9], a technique for detecting malware through analysis of system call logs is introduced. This approach achieves a high level of detection accuracy, but it neglects the possibility of certain applications being able to detect sandbox-like environments.

On the other hand, the system for malware detection proposed in [10] employs a deep convolutional neural network (CNN) to classify malware. The classification is based on a static analysis of the raw code sequence obtained from the disassembled program.

The work [11] proposed a system of static analysis, which consists of four stages. First, a call graph is built for each program, from which sequences of unique API calls are obtained. Each call is then assigned to a specific class, package, or family. The subsequent step involves constructing Markov chains from sequences of API calls to model the behavior of each application. The system then utilizes the probabilities of transitions between these calls as a feature vector to classify applications as either benign or malicious.

In [12], a framework for assessing the potential risk of applications was developed using triage. This approach employs a probabilistic model to predict the presence and significance of information flows in both benign and malicious applications. The results of the experiments demonstrated that this approach is effective in predicting the availability of information flows and can significantly reduce resource usage.

Meanwhile, in [13], the approach to detecting malicious programs involves both static and dynamic analysis. In order to improve the efficacy of static analysis-based malware detection, traditional features such as permissions and API calls are utilized. The proposed method incorporates feature selection and clustering techniques to normalize the features obtained from call graphs of different sizes.

In the study [14], the possibility of using meaningful and connective features for the detection of malware was investigated. For this, various types of entities and their semantic relationships were simulated, including relationships between files, archives, machines, APIs, and DLLs. A method was developed to map relationships between files using a structured heterogeneous information network (HIN) and metagraph-based approach. In order to identify the HIN, it was necessary to apply effective methods of studying hidden ideas. For this, a new metagraph2vec model was proposed, which is based on the creation of metagraph schemes.

In [15], the potential use of evolutionary computations is being explored to generate new iterations of malware capable of evading protection systems that rely on static analysis. Furthermore, these methods could be employed to automatically devise more effective measures for preventing such malware.

In work [16], a novel approach to malware detection is suggested, involving the analysis of information flows to detect behavior patterns and related flows that share common computation paths. These intricate streams are able to accurately capture complex behavior, which can indicate either malicious or benign programs. To identify unique and shared behavior patterns, the method utilizes an N-gram analysis of API calls present in these complex flows.

The paper [17] proposes the use of a deep discriminative adversarial network (DAN) to classify applications into malicious and benign using three types of features: raw code operations, permissions, and API calls. By using this approach, it is possible to detect malicious programs that employ obfuscation techniques to avoid detection.

An examination of literature reveals the significance of detecting malicious programs, with current methods demonstrating considerable effectiveness but also a high rate of false positives. A notable limitation of this approach is its substantial computational demands and inability to dynamically counteract both known and unknown malware attacks.

In addition, some of these methods share common weaknesses, such as ignoring packaged malware and failing to protect devices from zero-day attacks and malware that can modify their code.

Existing anti-virus programs are not always effective in detecting new types of viruses or malware, which can compromise your system and steal important information. Therefore, it is critically important to investigate and create new methods for detecting malicious software, especially using modern technologies like Blockchain.

2.1 Blockchain in cybersecurity

Blockchain is a technology that secures data using cryptographic techniques such as hash functions and digital signatures. These methods make it possible to guarantee the integrity of data stored in blockchain blocks (Figure 1).

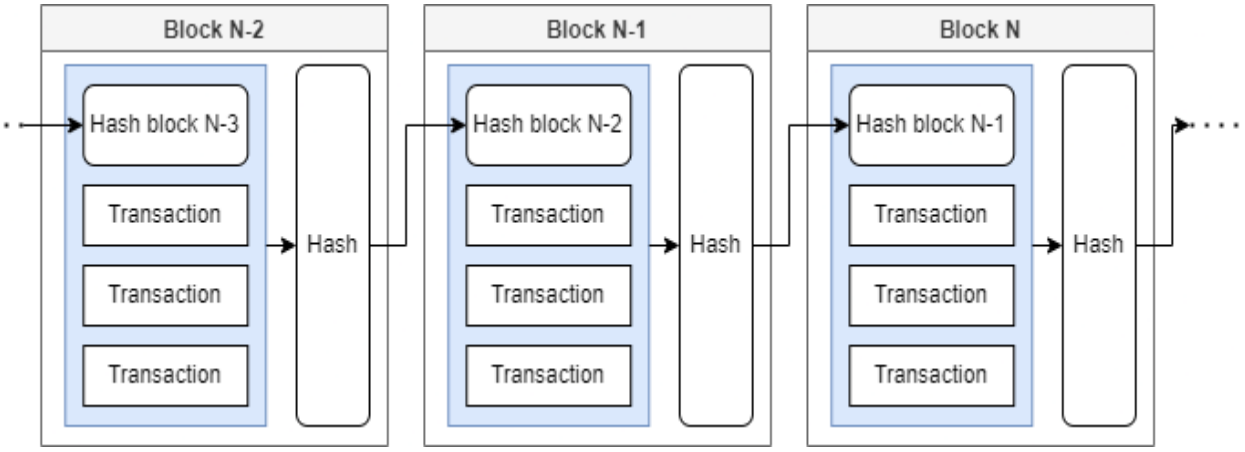

Figure 1. An example of the general structure and organization of blocks Blockchain

Blockchain technology has practical applications in the realm of cybersecurity, providing a secure and dependable infrastructure for data processing, storage, and transmission. For instance, blockchain can be used to store sensitive information like financial information, personal data, and medical data. With encryption methods, data can be stored in an encrypted form and access to it can be controlled using digital signatures and other methods.

In addition, the blockchain can be used to confirm the authenticity and integrity of data that is transmitted over the network. Blockchain can help prevent hacking and other forms of cybercrime by creating a safe and secure infrastructure.

However, it is important to note that blockchain alone is not a universal defense against cybercrime. Criminals can use a variety of techniques to gain access to data, even if it is stored on the blockchain. However, the application of blockchain can be a useful addition to other cybersecurity methods already in use.

In addition to data storage, blockchain can also be used to verify the identity of users and create secure payment mechanisms. Many blockchain platforms, such as Bitcoin and Ethereum, are used for secure and anonymous transactions, allowing users to transact without the use of intermediaries such as banks.

One of the fundamental advantages of the blockchain is that it works on the principle of decentralization, that is, data is stored on several computers at the same time (Figure 2), which makes it impossible to hack the system by invading one central server. Furthermore, for any modifications to the blockchain, the consensus of the majority of network participants is necessary, ensuring the system's reliability and its ability to withstand hacking attempts and manipulation.

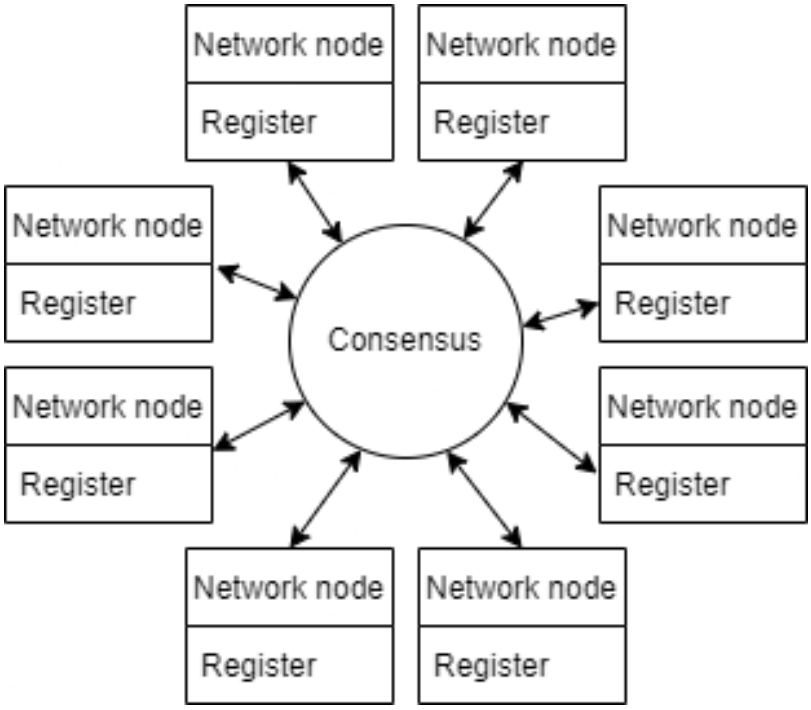

Figure 2: An example of the general structure of a distributed ledger Blockchain

However, like any technology, blockchain also has its limitations and drawbacks. For example, processing transactions on the blockchain can take quite a long time and have high fees. Additionally, some types of blockchain attacks can be successful if criminals can gain access to sufficient computing resources.

One of the unique features of blockchain technology is that in order to compromise sequences of blocks, one needs to have more than 51% participation in the computing power used to create new blocks. The use of blockchain technology offers substantial advantages to numerous users who need instantaneous, trustworthy access to shared transactions. As the blockchain does not have a singular data storage location, it lacks a central point of weakness. This enhances the security and accessibility of data for each participant in the network.

2.2. Incorporating Blockchain Mechanism for the Implementation of Malware Detection Technologies

The article outlines a flexible approach to identifying malicious software in networks, which relies on a load distribution mechanism among network participants. Malware poses a serious threat to the

security of information systems, and therefore the development of effective methods of searching for it is extremely important.

The demonstrated method employs various machine learning techniques to identify code fragments that may pose a threat. Based on the received data, the software tool determines whether this piece of code is malware or benign software. This enables a successful search for malware, which is particularly advantageous during the swift dissemination of new malicious programs. Furthermore, the method utilizes a range of machine learning techniques such as data classification and deep learning. These methods make it possible to improve the accuracy of detection of malware and reduce the number of false signals.

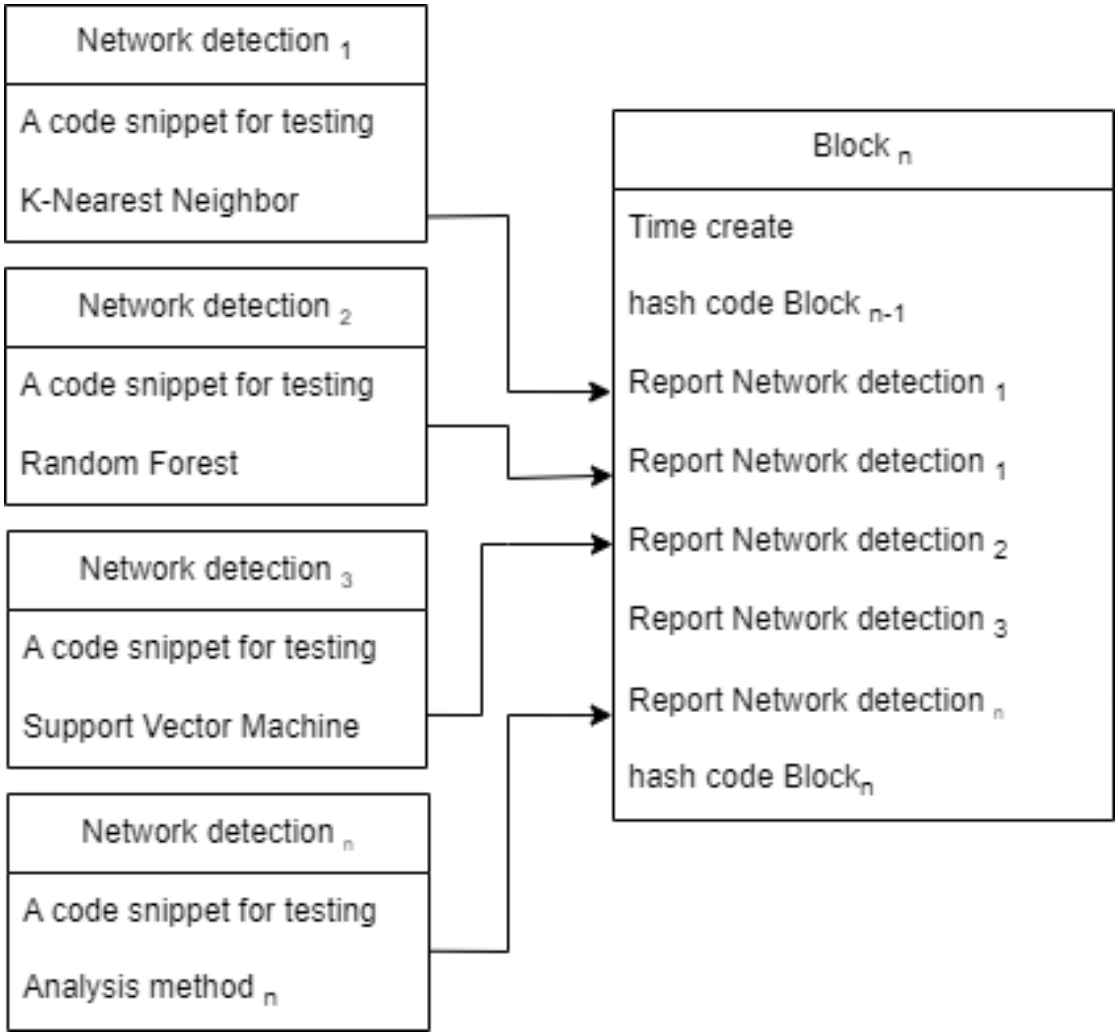

Figure 3: An example of the organization of the network structure

The primary objective of the suggested approach is to enhance malware detection efficacy through the segregation of malware identification and detection mechanisms.

The developed method is comprised of two subsystems, which prevent information compromise and improve malware identification.

2.3. Multi-Network Malware Detection Subsystem

A collection of malware detection networks refers to a group of networks, wherein each network utilizes a relevant algorithm to identify potentially harmful code, and is composed of a cohort of network participants. This mechanism makes it possible to quickly scale the network, to increase the

number of subnets with the appropriate search algorithms for the malware detection, and to scale the number of participants in the subnets. Figure 3 shows the general structure of the network, as a result of which we receive a Blockchain block with the result of the work, for further analysis by the neural network. Thus, the set of subnets can be denoted as $B = \{N_{Pi}\}_{i=1}^{K}$, where K is the total number of subnets engaged in examining potentially hazardous code segments, N_i is the i-th subnet participating in the validation; P_i is the a set of users associated with N_i subnet.

Each subnet can have an unlimited number of users in order to increase the speed of malware detection. In order to effectively use the power of the network, a part of the participants performs the role of a feature extraction mechanism, which can indicate belonging to the appropriate method of malware detection. Each member of the network checks a specific piece of code for potentially dangerous elements and submits a report to create an overall subnet report. Each network technique can have a distinct approach to representing potentially hazardous code in a format amenable to machine learning analysis, achieved through the selection of suitable features, such as n-grams, control flow graphs, feature vectors, opcode sequences, etc.

An instance of the arrangement of network nodes for scrutinizing potentially hazardous code is depicted in Figure 4. Each of the nodes consists of a set of participants of the network P, which perform verification according to the given method of the network N_i.

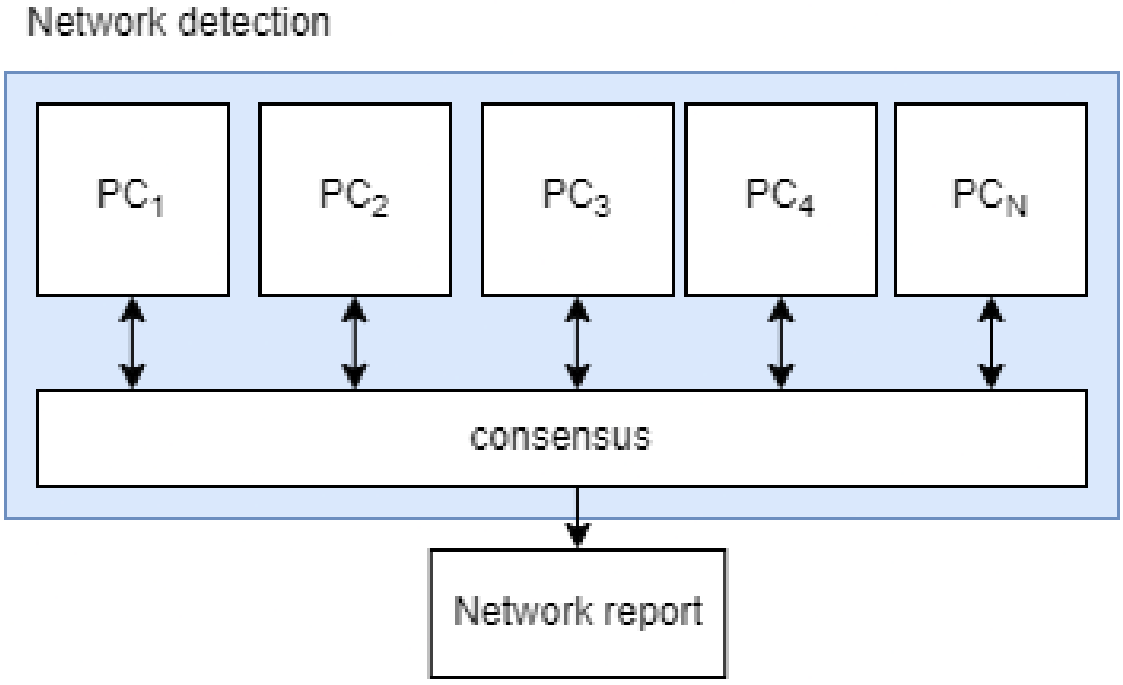

Figure 4: Example of subnet organization

Since network participants do not know about each other to ensure network security from data compromise, a consensus mechanism based on the Proof-of-Action (PoA) algorithm was used. Thanks to the Proof-of-Action algorithm, the maximum efficiency of the use of computing power is achieved. Because in order to achieve consensus, Proof-of-Action uses a mechanism for validating results from a group of nodes that have completed the verification. Thus, a randomly generated group of validators is used for validation. To increase the accuracy and reliability of the validation results, the number of validation steps can be increased. If a conflict of validation results occurs during validation, the validation iteration is repeated with a change in the number of validators. After conducting the validation stage and drawing up the corresponding report, the participants of the validation groups receive efficiency coefficients, which are later used to obtain the validator's rating. This coefficient is taken into account when creating a general report on the results of the network, and the soft voting method is used.

2.4. Subsystem of the analysis of malware detection results

The Blockchain method is used for efficient and transparent storage of information in the system. This approach will provide an opportunity to store the results of the check in a decentralized manner. With the formalized reports of the detection sub-networks and the Blockchain structure, the validation results are used to train a deep neural network. Thanks to this, the system has the ability to check potentially dangerous code fragments without involving subnets. If the accuracy of the check is low, the system will start a mechanism for checking potentially dangerous codes over networks. Figure 5 shows the interaction mechanism of the sequence of blocks and the neural network.

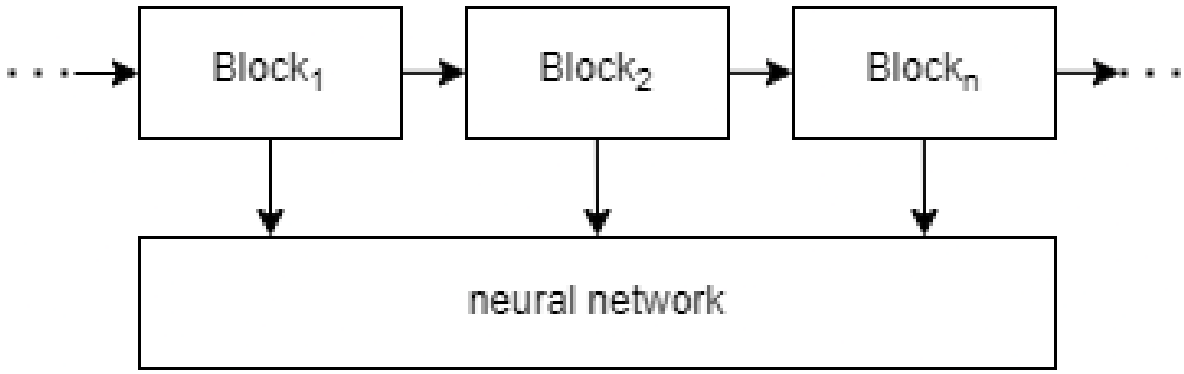

Figure 5: An example of the interaction of Blockchain and a neural network

When a new block is created, the neural network starts a learning mechanism based on the generated network results.

Figure 6 shows the general algorithm of the system for analyzing a potentially dangerous piece of code.

3. Experiments

A network consisting of 120 computer systems divided into 12 subnets was used for software analysis experiments. Each computer subnet was designed to perform certain functions. For example, the S group was engaged in extracting signs that indicated that the programs belonged to the malicious class.

The P team analyzed potentially dangerous code to determine whether it was malicious or safe software. Group J, for its part, was responsible for verifying and validating the malware analysis results that were received from group P.

This network allowed for more accurate and detailed analysis of the software, which ensured greater efficiency and reliability of the experiments.

In addition, this division of responsibilities among different groups has helped in faster and more accurate detection of malicious software, as well as in reducing the number of false reports about safe software.

The following methods based on machine learning [18-25] were employed to scrutinize segments of code that may pose a threat: K-Nearest Neighbor, Random Forest, Support Vector Machine, Rotation Forest, Decision Trees [26-28].

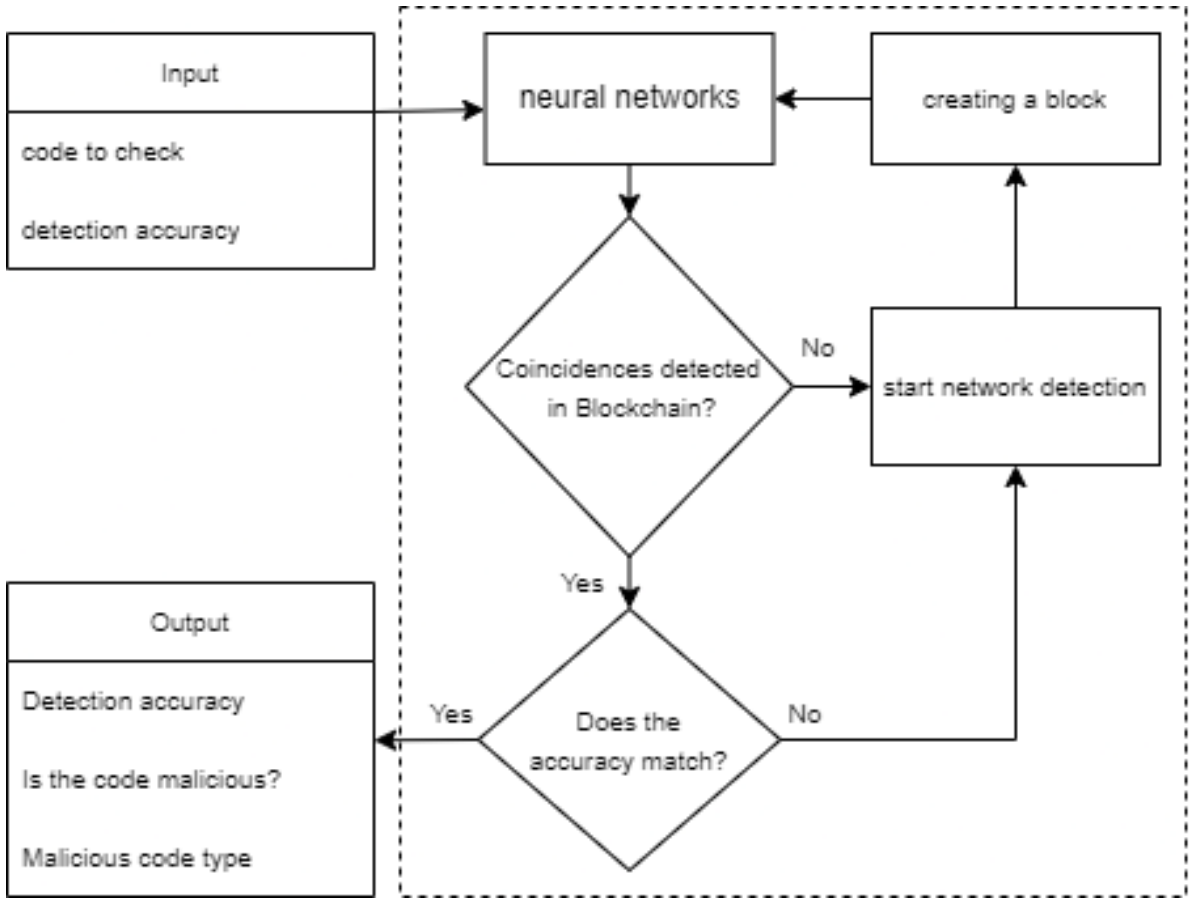

Figure 6: System operation algorithm

The publicly available data set [29] was used for the experiments. It contains 3214 samples of different malware classes. The samples of benign software were taken from the Microsoft store [30] and consist of 3597 units. Table 1 displays the outcomes of the conducted experiments.

Table 1
The outcomes of the conducted experiments, TP - True positive, TN - True negative, FN - False positive, FP - False negative.

Experiment order number	Maleware class	TP	TN	FN	FP	Overall accuracy, %
1	Rootkit	371	366	2	3	99.33
2	Backdoor	398	395	2	4	99.25
3	Worm	313	300	3	3	99.03
4	Polymorphic virus	357	355	5	2	99.03
5	Dropper	307	301	3	3	99.02
6	Downloader	346	331	2	5	98.98
7	Trojan	326	320	6	1	98.93
8	Adware	333	330	7	1	98.81

4. Conclusions

This work proposes a new approach to malware detection using deep learning and Blockchain technology. In this method, network participants use a distributed system of subnets to parallelize the analysis of potentially dangerous pieces of code.

This guarantees efficient identification of malware and mitigates the risk of utilizing analysis findings from a potentially compromised participant.

The basic idea is to use the blockchain to ensure security and accuracy of malware detection.

Network participants are divided into three groups, depending on their functional purpose. Group S eliminates indications that may suggest the software belongs to a particular malware class. Group P scrutinizes potentially hazardous code to distinguish whether it belongs to a specific malware class or benign software using machine learning techniques. Group J is responsible for validating the malware analysis results provided by the participants.

The algorithm proposed in this work is based on the subsystems and Proof-of-Action. The first mechanism provides parallel analysis of potentially dangerous software by different participants. The second mechanism is used to validate the results of analysis and increase the accuracy of detection.

The Proof-of-Action algorithm is employed to validate the analysis outcomes of participants' segments of potentially hazardous code. The soft voting method is utilized to ascertain the final analysis outcome based on the participants' results.

The outcomes of the experiments demonstrate the high effectiveness (98.81-99.33%) of the proposed malware detection approach.

The advantage of this method is scalability, that is, the possibility of adding new methods of detecting malicious software as new networks. Accordingly, the number of participants for network validation can also be scaled, giving the potential for rapid code validation and faster results.

Future research will focus on identifying optimal methods for dividing participants into groups according to their functional purposes and the most efficient network topology.

5. References

[1] Cybersecurity benchmark study reveals risk-based approach prevents security breaches URL: https://www.skyboxsecurity.com/blog/cybersecurity-benchmark-study-reveals-risk-based-approach-prevents-security-breaches/

[2] The McAfee Consumer Mobile Threat Report URL: https://www.mcafee.com/content/dam/consumer/en-us/docs/reports/rp-mobile-threat-report-feb-2022.pdf

[3] SonicWall 2022 cyber threat report URL: https://www.infopoint-security.de/media/2022-sonicwall-cyber-threat-report.pdf

[4] SymantecTM Universal Link Installation Guide URL: https://techdocs.broadcom.com/content/dam/broadcom/techdocs/symantec-security-software/endpoint-security-and-management/integrated-cyber-defense-exchange/generated-pdfs/Symantec_Universal_Link_Installation_Guide_4.0.0.pdf

[5] Internet Security Threat Report URL: https://docs.broadcom.com/doc/istr-24-2019-en

[6] Cybercrime To Cost The World $10.5 Trillion Annually By 2025 URL: https://cybersecurityventures.com/cybercrime-damage-costs-10-trillion-by-2025/

[7] B. Amro, Personal Mobile Malware Guard PMMG: a mobile malware detection technique based on user's preferences, International Journal of Computer Science and Network Security 18(1) (2018) 18–24.

[8] F. Idrees, M. Rajarajan, M. Conti, T. Chen, Y. Rahulamathavan, Pindroid: a novel android malware detection system using ensemble learning methods, Computers & Security, 68 (2017) 36–46.

[9] S. Chaba, R. Kumar, R. Pant, M. Dave, Malware Detection Approach for Android systems Using System Call Logs, arXiv preprint arXiv:1709.0880, 2017.

[10] N. McLaughlin, J. Martinez del Rincon, B. Kang Deep android malware detection, Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy, 2017. – pp. 301–308.

[11] E. Mariconti, L. Onwuzurike, P. Andriotis, E. De Cristofaro. MaMaDroid: Detecting Android Malware by Building Markov Chains of Behavioral Model, ACM Trans. Priv. Sec., 1 1 (2019) 1–33.
[12] O. Mirzaei, G. Suarez-Tangil, J. Tapiador, J. M.de Fuentes Triflow: Triaging android applications using speculative information flows, Proceedings of the 2017 ACM on Asia Con-ference on Computer and Communications Security, 2017. – pp. 640-651.
[13] Y. Liu, K. Guo, X. Huang, Z. Zhou, and Y. Zhang. Detecting Android Malwares with High-Efficient Hybrid Analyzing Methods. Mobile Information Systems (2018) 1–12, doi: 10.1155/2018/1649703.
[14] Y. Fan, S. Hou, Y. Zhang, Y. Ye, and M. Abdulhayoglu. Gotcha - Sly Malware!: Scorpion A Metagraph2vec Based Malware Detection System. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (2018) 253-262.
[15] S. Sen, E. Aydogan, and A. I. Aysan. Coevolution of Mobile Malware and Anti-Malware. IEEE Trans.Inform.Forensic Secur., 13 10 (2018) 2563–2574, doi: 10.1109/TIFS.2018.2824250.
[16] F. Shen, J. D. Vecchio, A. Mohaisen, S. Y. Ko, and L. Ziarek, Android Malware Detection Using Complex-Flows. IEEE Trans. on Mobile Comput., 8 6 (2019) 1231–1245, doi: 10.1109/TMC.2018.2861405.
[17] S. Millar, N. McLaughlin, J. Martinez del Rincon, P. Miller, Z. Zhao. DANdroid: A multi-view discriminative adversarial network for obfuscated Android malware detection. Proceedings of the tenth ACM conference on data and application security and privacy, 2020, pp. 353-364.
[18] L. Gaoqi, et al. "Distributed blockchain-based data protection framework for modern power systems against cyber attacks." IEEE Transactions on Smart Grid 10.3 (2018): 3162-3173.
[19] N. Moustafa, B. Turnbull and K. -K. R. Choo An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things, IEEE Internet of Things Journal 6.3 (2018) 4815-4830.
[20] A. Ishtiaque, M. Darda and S. Nath. Blockchain: A New Safeguard to Cybersecurity, Blockchain Technology: Applications and Challenges (2021) 271-284.
[21] B. Savenko, S. Lysenko, K. Bobrovnikova, O. Savenko, G. Markowsky. Detection DNS Tunneling Botnets // Proceedings of the 2021 IEEE 11th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), IDAACS'2021, Cracow, Poland, September 22-25, 2021.
[22] S. Lysenko, K. Bobrovnikova, R. Shchuka, O. Savenko. A Cyberattacks Detection Technique Based on Evolutionary Algorithms. 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), 2020. Vol.1. pp. 127-132.
[23] S. Lysenko, O. Savenko, K. Bobrovnikova. DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering, CEUR-WS 2104 (2018) 688-695.
[24] K. Bobrovnikova, S. Lysenko, B. Savenko, P. Gaj, O. Savenko. Technique for IoT malware detection based on control flow graph analysis. Radioelectronic and Computer Systems, 1 (2022) 141–153.
[25] S. Lysenko, K. Bobrovnikova, B. Savenko, P. Gaj, O. Savenko, Botnet Detection Approach Based on DNS CEUR WS 3156 (2022) 400–410
[26] Y.Gao, H. Hasegawa, Y. Yamaguchi, H. Shimada, Malware Detection Using Gradient Boosting Decision Trees with Customized Log Loss Function. In 2021 International Conference on Information Networking (ICOIN), pp. 273-278.
[27] Savenko O. Nicheporuk, A., Hurman, I., Lysenko, S. Dynamic signature-based malware detection technique based on API call tracing. CEUR-WS. 2019. Vol. 2393. P.633-643, ISSN: 1613-0073.
[28] Q. Wang and H. Meng, Blockchain-based Federated Learning with Limited Resources, 2022 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL & ICCEA), Changchun, China, 2022, pp. 449-452, doi: 10.1109/CVIDLICCEA56201.2022.9825317.
[29] MalwareBazaar | Malware sample exchange. URL: https://bazaar.abuse.ch/
[30] Microsoft Store. URL: https://apps.microsoft.com/store/apps