Investigation on Sharing Signatures of Suspected Malware Files Using Blockchain Technology

Ryusei Fuji, Shotaro Usuzaki, Kentaro Aburada, Hisaaki Yamaba, Tetsuro Katayama, Mirang Park, Norio Shiratori and Naonobu Okazaki

***Abstract*—In recent years, the amount of new malware has been rapidly increasing. Because malware has an adverse effect on the Internet, upon which modern society is increasingly dependent, its detection is very important. In addition, blockchain technology has attracted the attention of many people in recent years due to its four main characteristics of decentralization, persistency, anonymity, and auditability. In this paper, we propose a system for sharing the signatures of suspected malware files using blockchain technology. The proposed system can share the signatures of suspected files between users, allowing them to rapidly respond to increasing malware threats. Further, it improves the accuracy of detection and removal of malware by utilizing signatures recorded by the blockchain. In the evaluation experiment, we created a prototype of the proposed system and investigated its effect on the accuracy of detection and removal of malware. Compared with heuristic methods or behavior-based methods only, the proposed system which uses these methods plus signature-based method using shared signatures on the blockchain improved the false negative rate by about 4% and the false positive rate by about 2.5%.**

***Index Terms*—malware detection, blockchain technology, Ethereum, smart contract.**

I. Introduction

MALWARE is a portmanteau word combining "malicious" and "software", and it operates illegally with the purpose of theft or destruction of a computer's internal information. Malware achieves its objectives in an infected computer by performing illegal operations without detection of its presence. An example of damage caused by malware infection is ransomware, which interrupts operation of the computer or encrypts the data inside it. The attacker will request a ransom from the user in exchange for releasing the restriction on access to the computer or its data. The amount of pecuniary damage from ransomware in 2017 is said to be USD 5 billion [1], which is a serious problem. In addition, the presence of malware in Internet of Things (IoT) equipment has been confirmed, with the "Mirai" malware at the head of the list. In 2016, denial of service attacks of up to 1.5 Tbps have been executed by exploiting IoT devices infected with "Mirai" [2]. According to Security Report 2017/2018 [3] published by AV-TEST, in recent years the number of new malware programs observed is more than 100 million per year, which means that about 4 new malware programs are discovered per second. Clearly, malware has a serious adverse effect on modern society by impacting the Internet that it has been founded upon.

Malware detection techniques are roughly divided into three types: signature-based methods, behavior-based methods, and heuristic methods [4]. Signature-based methods are commonly used to detect malware. The signature is a sequence of bytes with features extracted from malware, and if the contents of an inspected file match one of these signatures, the file is determined to be malware. Signature-based methods have the advantage of reliably detecting known malware, but they have the disadvantage of not being able to detect unknown malware. Behavior-based methods perform their malware detection by actually executing the file under inspection and observing its behavior. These methods can detect unknown malware that cannot be detected by signature-based methods. However, these methods have the disadvantage of a high false positive rate (FPR), which is the rate of benign files being labeled as malicious files. Heuristic methods are techniques that detect malware using data mining and machine learning techniques. The features utilized in heuristics methods include API call sequences issued to the operating system and machine language instruction. The main advantages and disadvantages of heuristics methods are similar to those of the behavior-based detection methods.

To prevent malware infection, usually anti-virus software provided by a vendor is installed on the computer. Generally the malware signatures used by the anti-virus software are distributed from the anti-virus vendors, who provide and update these signatures by collecting and analyzing malware-related information from sources such as users and online malware inspection and analysis services. Hashimoto et al. [5] provided information on malware that could not be detected by anti-virus software to the vendor to calculate the subsequent malware detection rates, and evaluated the anti-virus software. According to the study, the malware detection rates 30 days after providing the malware information were 50% at most. In other words, it is conceivable that an anti-virus vendor alone cannot respond adequately to malware that is rapidly increasing.

The above discussion suggests that not only should anti-virus vendors collect malware information, but also users should share this information with each other. To realize such sharing of malware information among users, we adopted blockchain technology.

Blockchain technology is the fundamental technology of various virtual currencies, including Bitcoin, and it has attracted much attention in recent years. Blockchain technology was proposed by Nakamoto [6] in 2008 to realize the Bitcoin network. This technology enables rapid transactions

Manuscript received December 20, 2018; revised February 6, 2019. This work was supported by the Japan Society for the Promotion of Science, Kakenhi Grants JP17H01736, JP17K00139, and JP18K11268.

R. Fuji, S. Usuzaki, K. Aburada, H. Yamaba, T. Katayama, and N. Okazaki are with the Faculty of Engineering, Univ. Miyazaki, Gakuen-kibanadai-nishi-1-1, Miyazaki 889-2192, Japan e-mail: aburada@cs.miyazaki-u.ac.jp

M. Park is with Kanagawa Institute of Technology, 1030 Shimo-Ogino, Atsugi, Kanagawa 243-0292, Japan.

N. Shiratori is with Research and Development Initiative, Chuo University, 1-13-27 Kasuga, Bunkyo-ku, Tokyo 112-8551, Japan

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

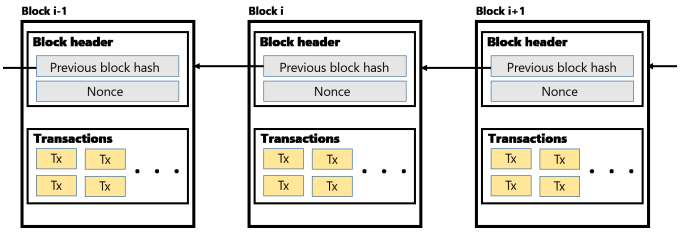

Fig. 1. Blockchain example

between users with a low cost and without mediation of a central authority. Further, blockchain-based decentralized applications (Dapps), such as uPort [7], have been emerging. Dapps save and use certain information on a blockchain, and their use is expected to increase.

In this paper, we propose a system for sharing the signatures of suspected malware files using blockchain technology. The proposed system allows the signatures of suspected files to be shared between users, and we can more rapidly respond to increasing malware. Furthermore, the shared signatures make it is possible to improve the malware detection accuracy.

II. Related work

Jingjing et al. [8] proposed a framework, called Consortium Blockchain for Malware Detection and Evidence Extraction (CB-MDEE), that detects and classifies malware for mobile devices. The CB-MDEE is composed of two blockchains, a public blockchain (PB) and a consortium blockchain (CB). Users belonging to the PB use a multi-feature model created from, for example, sensitive behavior graphs and installation packages, to detect and classify malware, and store the information on the PB for subsequent malware detection and classification. Members of malware detection organizations belonging to the CB use the information to create a fact base for updating the malware feature database. In evaluation experiments, the CB-MDEE achieved a classification accuracy of 94% for android malware.

Roman et al. [9] proposed a system to support cyber analysts by classifying and managing cyber incident reports using blockchain technology and a deep autoencoder neural network. When a cyber expert enters a cyber incident report into the system, the system classifies the report and returns past similar incident reports. Because the classification and management are executed automatically, the cyber expert can adopt suitable countermeasures quickly. In the evaluation, they used 5,850 training documents and 584 test documents to validate the effectiveness of their proposed system. For the "fulldisclosure" category, they achieved a true positive rate 0.991 and an FPR of 0.059.

This study is based on the assumption that users belonging to the blockchain have a different malware detection system using behavior-based methods or heuristic methods. Then, these detection results are saved as votes on the blockchain for later use. As a result, we can detect and eliminate malware by utilizing the results of our own malware detection system and votes of other users stored on the blockchain, which is different from the above studies.

III. Blockchain technology

A. Overview of blockchain technology

Blockchain technology was proposed as a fundamental technology for realizing Bitcoin in the paper published by Nakamoto [6] in 2008. Bitcoin was realized by combining several inventions to decentralize functions, such as currency issuance and mediation of transactions, which banks typically do. Due to the decentralized function, Bitcoin makes it possible to issue currency and create transactions among users without third-party institutions, such as banks. An example of the blockchain is shown in Fig. 1.

When remitting coins between users, the user on the remittance side issues a transaction describing the transfer of value, including information such as remittance amount, address of destination, and digital signature of the user on the remittance side. The issued transactions are transmitted and received between mutually connected nodes. The node that received the transaction verifies the transaction, and if the transaction is valid, the node sends it to the next node. By this transaction transmission and reception activity, the issued transaction is propagated to the entire blockchain network. Ultimately, the transaction is included in the block by a miner, and the remittance process is completed by becoming a part of the blockchain. The blockchain maintains data integrity by using consensus algorithms, such as Proof of Work (PoW), Proof of Stake, and Delegated Proof of Stake. Bitcoin adopted PoW, which determines a cryptographic nonce so that the block hash value satisfies a specific hash value and generates the next block. Blocks generated by PoW are propagated to the entire Bitcoin network by transmission and reception of blocks between the nodes, and independently verified. As a result of the verification, if the block is valid, it is accepted into the Bitcoin network and becomes part of the blockchain.

The blockchain has four main characteristics: decentralization, persistency, anonymity, and auditability [10]. Based on the above characteristics, this study adopted blockchain technology, which enables rapid sharing of suspected malware signatures between users without the intervention of a central organization, such as an anti-virus vendor. It is possible to use these signatures to eliminate malware.

B. Blockchain platforms

Blockchain is the basis for various platforms, such as Ethereum [11] and Hyperledger [12]. Ethereum is a platform for building Dapps in an open-source development environment. We can develop various applications by executing a programmed contract called a smart contract on the Ethereum blockchain. We used Ethereum because it is already used as a blockchain platform in many Dapps, such as uPort [7].

IV. Proposed system

In this section, we explain the proposed system.

A. Overview of proposed system

An overview of the proposed system is shown in Fig. 2. The blockchain network is composed of users who want to share and obtain malware information. Here, it is assumed that each user's computer hosts a heuristic or behavior-based

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE I
DEFINITIONS FOR EACH SYMBOL

Symbols	Definitions
M_d	Malicious degree
M_t	Threshold for malicious degree
V_t	Threshold for total votes
V_b	The number of votes of 'benign'
V_m	The number of votes of 'malicious'
R_v	Voting confidence rate
R_s	Self confidence rate
D_r	Detection result of own malware detection system
F_v	Voting fee
R_v	The number of votes for compensation distribution

malware detection system and a signature-based system. Also, we suppose these malware detection systems use different features or methods. For example, the computer of user 1 might have a heuristic malware detection system based on API call sequences issued to the operating system and the computer of user 2 might have a heuristic malware detection system based on machine language instructions. Both users also have a signature-based detection system. The blockchain is used to store signatures (file hash values) and other information from suspected malware files.

When a user downloads an executable file, heuristic or behavior-based malware detection is executed first. If the downloaded executable file is judged as malware, the user sends the file hash value to the blockchain network as a suspected malware file identity. When another user downloads the same executable file, the user first checks whether the file hash value of the executable file is already registered as a suspected malware file identity on the blockchain. If the same file hash value exists on the blockchain, the user's heuristic or behavior-based malware detection system judges whether the file is malicious, and the result is sent as a vote (malicious or benign) to the blockchain network. Thereafter, based on the voting results on the blockchain and the results of its own malware assessment, the user's detection system decides whether to remove the suspect file.

A flowchart showing the process for each user is provided in Fig. 3, and symbols used in this paper are defined in Table I.

B. Malware detection systems on user computers

In this study, we assume that each user belonging to the blockchain network installs the following two malware detection systems on the computer:

- Malware detection system using heuristic or behavior-based methods
  This program is executed when the user downloads an executable file. In this study, it is assumed that each user detects malware using different features or methods.

- Malware detection system using signature-based methods.
  This system is responsible for investigating whether signatures already exist on the blockchain. Also, it calculates the degree of maliciousness and eliminates the downloaded file according to the result.

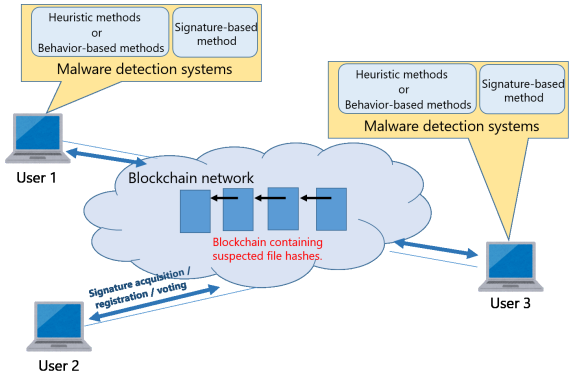

Fig. 2. Overview of proposed system

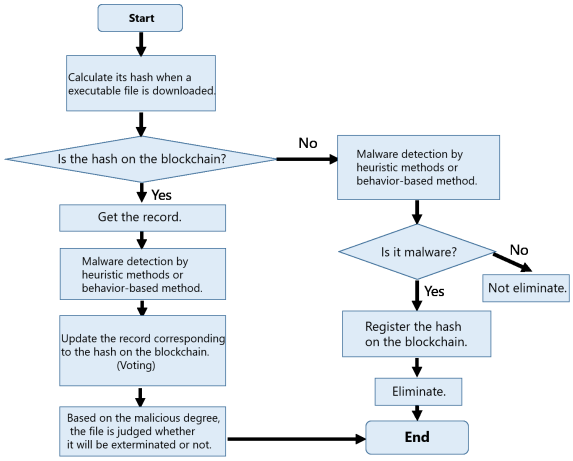

Fig. 3. Flowchart of malware detection performed by each user

C. Detection of suspected files and transmission of file hash values

When a user downloads an executable file, the heuristic or behavior-based malware detection is executed first. Next, the user's computer checks whether the file hash value is already registered as a suspected malware file hash value on the blockchain. If the file hash value does not exist on the blockchain, and if the malware detection system determines that the downloaded file is malware, the computer sends the file hash value to the blockchain network to share it and then eliminates the file. When the file hash value exists on the blockchain, the user's detection system sends out the result of its own malware analysis as a vote ("malicious" or "benign") to the blockchain network and then decides whether to remove the file with the elimination decision formula (see section IV-E).

D. Record components

Here we describe data such as file hash values and the number of votes to be stored on the blockchain. Data stored on the blockchain can be represented as a record, and its details are shown in Fig. 4. As can be seen from Fig. 4, the record is represented by the following five elements:

- Suspected file hash value
- Number of votes for "malicious"
- Number of votes for "benign"
- Addresses of users who voted "malicious"
- Addresses of users who voted "benign"

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

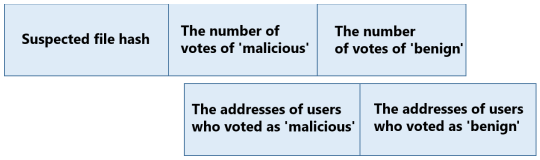

Fig. 4. Details of a record

The numbers of votes for "malicious" and "benign" are used to calculate the degree of maliciousness in the elimination decision formula (section IV-E) to determine whether to remove the file. The recording of user addresses prevents the same user from illegally voting more than once. Here, the user address is not an IP address but the address used on the blockchain, such as "0xca35b7d915458ef540ade6068dfe2f44e8fa733c".

E. Elimination decision formula

When the hash value of the downloaded file exists on the blockchain, the user's detection system determines whether to remove the file based on the maliciousness degree given by the elimination decision formula. That is, when equation (1) is satisfied, the file is not deleted, and when equation (2) is satisfied, the file is deleted.

$$M_d \leq M_t, \tag{1}$$
$$M_d > M_t, \tag{2}$$
$$0 \leq M_d \leq 1.$$

1) When $V_m + V_b \geq V_t$: The user's system uses only the voting result on the blockchain and calculates the maliciousness degree with equation (3).

$$M_d = \frac{V_m}{V_m + V_b}. \tag{3}$$

2) When $V_m + V_b < V_t$: The user's system calculates maliciousness degree with expression (4), the results of voting on the blockchain, and its own malware detection results by heuristic or behavior-based methods. Here, it is assumed that the malware detection system outputs 1 when the file is malware and 0 when it is benign. That is, $D_r \in \{0, 1\}$.

$$M_d = \frac{V_m}{V_m + V_b} \times R_v + D_r \times R_s. \tag{4}$$

where R_v and R_s are defined by the following expressions:

$$R_v = \frac{V_m + V_b}{V_t}, \tag{5}$$
$$R_s = 1 - R_v. \tag{6}$$

3) Example: Suppose that an executable file is downloaded and voting for the file hash value on the blockchain is 10 "malicious" votes ($V_m = 10$) and 5 "benign" votes ($V_b = 5$). Also, the malware detection system judges the file to be malware ($D_r = 1$), the threshold for total votes is set to 20 ($V_t = 20$), and the threshold for maliciousness degree is set to 0.5 ($M_t = 0.5$). The malicious degree M_d in this example is calculated as follows:

$$M_d = \frac{10}{10+5} \times \frac{10+5}{20} + 1 \times (1 - \frac{10+5}{20}) = \frac{3}{4}.$$

Because $\frac{3}{4} > 0.5$, the file will be deleted.

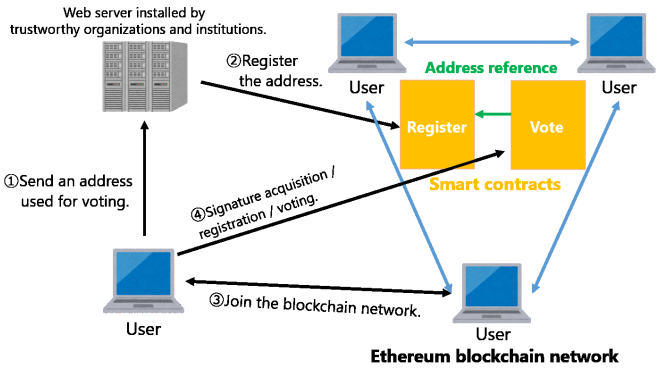

Fig. 5. Blockchain address registration

F. Countermeasure against mass voting by malicious users

The records stored on the blockchain include the addresses of users who voted in order to prevent duplicate voting by the same address. However, since any user can generate an unlimited number of addresses, it is insufficient to use only the measures described above. Therefore, we tried to solve this problem by establishing a web server by trustworthy organizations and institutions to register voting addresses and limit the addresses eligible to vote. Fig. 5 shows the countermeasures against mass voting by malicious users.

First, users who wish to participate in the Ethereum blockchain network access the web server installed by trustworthy organizations and institutions and register an address to be used for voting. The web server accesses the Register smart contract on the Ethereum blockchain and registers the address. After that, the user joins the network and acquires, registers, and votes for signatures. The Vote smart contract, which is responsible for signature acquisition, registration, and voting, accesses the Register smart contract and checks whether the address exists in the Register smart contract. If the address exists, acquisition, registration, and voting of the signature are accepted; otherwise, these are rejected.

Here, the web server prohibits the registration of consecutive addresses from the same IP address and confirms the human by CAPTCHA. From the above, it is possible to prevent the registration of addresses by malicious users and bots.

G. Incentive design

To encourage user voting, we designed incentives for dominant votes. The user pays a small voting fee when voting for a suspected file hash value. The user issues a message that an execution fee (Gas in Ethereum) is required to vote, but in this study, we set the fee to 0 (that is, we set gasPrice to 0). The Vote smart contract distributes the collected voting fee for each R_v for a suspected file hash value. Here, we define the votes belonging to the detection result with more votes as the dominant vote and the result with fewer votes as the inferior vote. The voting fees are distributed only to users who voted for the dominant vote for every R_v. Voting fees are not distributed to users who cast an inferior vote. Suppose that the number of dominant votes is DV and number inferior votes is IV. Then, the voting fee VF_{dist} to

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE II
PARAMETER DEFINITIONS FOR EQUATIONS (10) AND (11)

Symbol	Definition
True Positive (TP)	The number of malware detected as malware correctly
True Negative (TN)	The number of benign files judged as benign files correctly
False Positive (FP)	The number of benign files detected as malware mistakenly
False Negative (FN)	The number of malware judged as benign files mistakenly

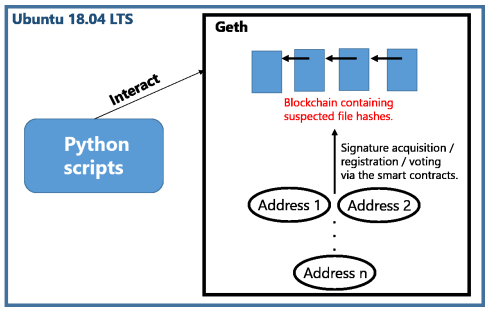

Fig. 6. Overview of experiment environment

be distributed is determined by the following equations:

$$VF_{dist} = \frac{F_v \times R_v}{DV}(for\ dominant\ votes), \quad (7)$$

$$VF_{dist} = 0(for\ inferior\ votes), \quad (8)$$

$$DV > IV, DV + IV = R_v. \quad (9)$$

By the above incentive design, the frequency and correctness of user voting will be promoted.

V. EVALUATION

In this section, we explain an evaluation experiment to verify the effectiveness of the proposed system. The purpose of the experiment is to investigate whether the accuracy of detecting and removing malware is improved by using the proposed system. To conduct the experiment, we created a prototype of the proposed system.

A. Overview of evaluation experiment

An overview of the experimental environment is shown in Fig. 6. To build a virtual environment on the computer, we installed Ubuntu 18.04 LTS using Oracle VM VirtualBox. Also, to build a blockchain within a private network, we installed Geth, an Ethereum client, and interacted with Geth through a Python script using Web3.py.

B. Detection accuracy indicators

We used the false negative rate (FNR) and FPR as evaluation indexes for malware detection and removal accuracy. FNR is the rate at which malware is mistakenly judged as a benign file, and FPR is the rate at which a benign file is erroneously detected as malware.

$$FNR = \frac{FN}{TP + FN}, \quad (10)$$

$$FPR = \frac{FP}{TN + FP}. \quad (11)$$

The definitions of the symbols in equations (10) and (11) are given in Table II.

C. Simulation procedure

In the evaluation experiment, it is assumed that no malicious user exists and each user possesses the malware detection systems described in section IV-B. In this experiment, rather than implementing user-specific malware detection systems, we created pseudo malware detection systems that have FPR and FNR as parameters. In addition, it is assumed that each user performs the malware detection of heuristic methods or behavior-based methods for all the predefined files regarded as malicious or benign, and registers, votes, and obtains information from the blockchain as necessary.

First, we created an address representing each user using a Python script and interacting with Geth. Next, we deployed the Register and Vote smart contracts on the blockchain and registered the addresses that we originally created with the Register smart contract.

The simulation continued until each address representing the user performed malware detection and removal for all predefined files regarded as malicious or benign.

D. Parameters

In this experiment using the system prototype, the number of user addresses was 30, the number of file hash values assumed as malicious or benign was 30 each, $V_t = 15$, and $M_t = 0.5$.

In addition, the FPR and FNR of the pseudo malware detection systems were set with reference to the literature [13], where Windows malware was detected using machine instruction sequences. Specifically, malicious instruction extraction and malicious sequential pattern extraction (MSPE) were used to efficiently and effectively obtain malicious sequences by heuristic methods. In the evaluation experiment, the detection result with MSPE combined with all-nearest-neighbor was the best result, achieving a detection rate of 96.17% (FNR of 3.17%) and FPR of 6.13%

They also experimented with combinations of other classifiers, and we set the FNR to 5% and FPR to 6% based on their experiment results.

E. Results and discussion

The experimental results are shown in Table III. Case A and Case B in the table are as follows:

- Case A
  Heuristic methods or behavior-based methods only.
- Case B
  Heuristic methods or behavior-based methods plus signature-based method using signatures on the blockchain.

FNR and FPR in Case B are the average for all users.

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE III
EXPERIMENTAL RESULTS

	FNR	FPR
Case A	0.05	0.06
Case B	0.013	0.035

FNR and FPR in Case A are the values from section V-D. From Table III, FNR and FPR in Case B improved by about 4% and 3%, respectively. Therefore, the proposed system that utilizes the signatures from the blockchain can improve the accuracy of detection and removal of malware. However, the standard deviations of FNR and FPR were 0.044 and 0.022, respectively. For this reason, it was revealed that some users who use the proposed system have degraded malware detection and removal accuracy.

VI. SUMMARY

In this paper, we proposed a system to share and utilize signatures of suspected malware files using blockchain technology. This system aims to quickly share signatures of suspected files among users and improve the accuracy of detection and removal of malware without a centralized organization, such as an anti-virus vendor. In the evaluation experiment, we created a prototype of the proposed system and investigated its accuracy for detecting and removing malware. The evaluation experiment showed that the proposed system improved the FNR by about 4% and the FPR by about 2.5%. For future work, it is necessary to evaluate the proposed system using malware detection systems based on actual heuristic or behavior-based methods. In addition, it is necessary to set each parameter and precondition so as to more accurately reflect the real world.

REFERENCES

[1] Sultan, Hirra, et al. "A SURVEY ON RANSOMWARE: EVOLUTION, GROWTH, AND IMPACT." *International Journal of Advanced Research in Computer Science* vol 9, No. 2 (2018).
[2] Barrera, David, Ian Molloy, and Heqing Huang. "IDIoT: Securing the Internet of Things like it's 1994." *arXiv preprint arXiv:1712.03623* (2017).
[3] AV-TEST "SECURITY REPORT 2017/18" available at:https://www.av-test.org/fileadmin/pdf/security_report/AV-TEST_Security_Report_2017-2018.pdf (accessed 2018/12/02).
[4] Bazrafshan, Zahra, et al. "A survey on heuristic malware detection techniques." *Information and Knowledge Technology (IKT), 2013 5th Conference* (2013):113-120.
[5] Ryota Hashimoto, Katsunari Yoshioka , and Tsutomu Matsumoto . "Evaluation of Anti-Virus Software based on the Correspondence to Non-Detected Malware" (in Japanese) *Distributed Processing System (DPS)* (2012): 1-8.
[6] Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." (2008).
[7] uPort.me at:https://www.uport.me/ (accessed 2018/12/02).
[8] Gu, Jingjing, et al. "Consortium Blockchain-Based Malware Detection in Mobile Devices." *IEEE Access 6* (2018): 12118-12128.
[9] Graf, Roman, and Ross King. "Neural network and blockchain based technique for cyber threat intelligence and situational awareness." 2018 *10th International Conference on Cyber Conflict (CyCon)*. IEEE, 2018.
[10] Zheng, Zibin, et al. "An overview of blockchain technology: Architecture, consensus, and future trends." *IEEE 6th International Congress on Big Data* (2017): 557-564.
[11] Ethereum Project available at:https://www.ethereum.org/ (accessed 2018/12/02).
[12] Hyperledger - Open Source Blockchain Technologies available at:https://www.hyperledger.org/ (accessed 2018/12/02).
[13] Fan, Yujie, Yanfang Ye, and Lifei Chen. "Malicious sequential pattern mining for automatic malware detection." *Expert Systems with Applications 52 (2016): 16-25.*

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)