International Innovative Research Journal of Engineering and Technology
ISSN: 2456-1983 Vol: 5 Issue: 3 March 2020

Multi Feature Detection and Signature Sharing of Android Malware using Blockchain

P. Boobalan[1], R.Keerthana[2], K.Nandhini[3], P.Vignesh[4]

[1,2,3,4]Information Technology, Pondicherry Engineering College, Pondicherry, India
[1]boobalanp@pec.edu, [2]rkeerthana1999@gmail.com, [3]nandhinik251998@gmail.com, [4]vig2699@gmail.com

Abstract: Android devices like smartphones and tablets has been gaining much popularity and accelerated usage for its low cost and increase in functionality and services. Due to its openness and free availability, Android Based system has become not only a major stakeholder in the market but has also become an attractive target for cybercriminals. The main objective of this project is to reduce false positive rate in malware detection by analyzing Different malware families are developing a corresponding Multi-Feature Model (MFM) on Android-based systems by following a fuzzy method of comparison. A file hash value is generated if a malware is suspected and send to the blockchain network As alleged identity of the malware file. If there is the same file hash value on the blockchain, the file is found to be malicious by the malware detection program, and the result is sent to the blockchain network as a vote. This system can ensure data security and consistency with the use of blockchain. The results show that the proposed system can achieve efficient detection accuracy and reduced false positive and negative rates.

Keyword: Ethereum Blockchain, Malware detection, Multi feature, Smart contract, Static- dynamic analysis.

1. INTRODUCTION

Android Operating systems, is one of the most popular mobile operating systems. With the increased number of applications from different market, android provides several functionalities to its users. Unfortunately, smartphones running on android are a main target of cyber criminals and increasingly affected by malicious software. The increased malware threats on this efficient and open source android platform has made the malware detection as a challenging issue.

Anti-virus solutions identify and analyze the malware and create a special handcrafted signature that is released as an update to their customers. This process of manual analysis usually takes a long time, during which the malware stays undetected and continues to get infected. Furthermore, the developers of malware programs typically make some minor changes to their code even when detected, so the latest version is undetected by the anti-virus software. Hence an antivirus vendor alone might not be able to respond effectively to the rapidly increasing malware. The discussion above indicates that antivirus vendors should not only gather information about malware but that consumers should also exchange this information with each other. We embraced blockchain technology to realize such sharing of malware information between users.

The blockchain technology has gained considerable popularity as a new form of distributed computing paradigm because of its high performance, high data security, high reputation and low cost. Blockchain In 2008, Nakamoto suggested technology[7] for the implementation of the Bitcoin network. This technology allows for fast transactions for both low cost users, and without central authority mediation. Additionally, decentralized applications (Dapps) based on blockchain, such as uPort[8], also appeared. Dapps save and use some blockchain details, and are expected to increase its use. To address the platform of detecting malicious codes in malware and extracting the corresponding evidences in mobile devices, an Ethereum blockchain framework is to be constructed and a hash value is calculated for the suspected malware file which is shared to the blockchain using smart contract.

Specifically, in view of detecting different malware families (DroidDream, zHash, DroidKungFu, aucun, gingermaster, etc.) In the Android-based framework, a modeling feature that includes software package feature, permission and device feature, and function call feature is to be performed using statistical analysis method to extract malware family features. In addition, a multi-feature detection system for Android-based malware detection and classification is to be developed to reduce false-positive levels and boost the detection potential of

malware variants. In addition to this, The proposed framework could share the signatures (hash value) of suspected files between users, enabling them to respond quickly to could malware threats. The comparison with the existing methodologies indicates that the latest method proposed With lower false-positive and false-negative levels, can achieve higher detection accuracy within a limited time.

2. LITERATURE SURVEY

Jingjing et al.[1] introduced a platform to identify and recognize malware for mobile apps, named the Blockchain consortium for malware detection and the retrieval of proof (CB-MDEE). The CB-MDEE consists of two blockchains, a public blockchain (PB) and a blockchain consortium (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. Representatives of CB-affiliated malware detection organizations Using the information to create a knowledge base and update the malware functions database. Through test tests the CB-MDEE has achieved greater detection accuracy for android malware.

TaeGuen et al[2] suggested a Multimodal Deep Learning System for the Detection of Android Malware using Various Features. It uses various kinds of features to reflect the properties of Android applications from various aspects, and the features are refined using our existence-based or similarity-based feature extraction method for effective feature representation on malware detection, with this approach, it was possible to maximize the benefits of encompassing multiple feature types.

Li J et al [3] proposed a system based on permission usage analysis. Instead of extracting and analyzing all Android permissions, they developed 3-levels of pruning by mining the permission data to identify the most significant permissions that can be effective in distinguishing between benign and malicious apps. SIGPID (Significant permission identification) then utilizes machine-learning based classification methods to classify different families of malware and benign apps. Their evaluation finds that only 22 permissions are significant.

Iqbal S and Zulkernine M [4] proposed A system for using several, real-time android malware detectors called SpyDroid. It consists of two modules of operating system (monitoring and detection), and supports subdetectors of application layers. Sub-detectors are standard Android applications that use the monitoring module to track and analyze different runtime details, and report their findings to the detection module. The monitoring module determines when to mark an application as malware. Results showed that decisions from multiple subdetectors on a real system would significantly increase the malware detection rate.

Howard M et al [5] proposed a Method for that machine learning-based malware detection systems by predicting signatures of potential variants of malware and injecting them into the defense network as a vaccine. Their approach utilizes deep learning to learn from family history trends in malware evolution. These patterns of evolution are then used to predict future changes within the family. The tests showed that a detection system combined with such future malware signatures is capable of detecting future variants of malware that the detection system alone could not detect.

3. PROPOSED WORK

The description of the program proposed is shown at fig1.1. The blockchain network is comprised of users who want to exchange and receive knowledge about malware. Here, the device of each user is presumed to host the proposed malware detection system and a signature-based system. The blockchain is used to store signatures of suspected malware files (hash values). and other information from them. The proposed system focuses on improving the accuracy rate by using multi features detection along with signature-based method which Can detect and remove malware using our own malware tests detection system and votes (no of users who have already identified whether a software is malicious or benign) of other users stored on the blockchain.

A. Contribution

- When a user downloads an apk file, the proposed malware detection is executed first by performing feature modeling by utilizing statistical analysis method to extract malware family features, including software package feature, permission and application feature, and function call feature.
- Analyzing Specific malware families use Android-based systems and a corresponding Multi-Feature Model (MFM) by following a fuzzy method of comparison. To lower false-positive rates and boost malware detection capability variants, multiple marking functions are used [1].
- · When the downloaded apk file is found to be Malware, the user sends the hash value as presumed file identity to the blockchain network for malware.
- If another User downloads the same apk file, first checking whether the executable hash value of the file is

International Innovative Research Journal of Engineering and Technology
ISSN: 2456-1983 Vol: 5 Issue: 3 March 2020

already documented as a suspected file identity for malware on the blockchain.

Extracted samples	Compared sample	If probability >50%	Marking function
S= {SBS_1, SBS_2.SBS_N}	SBSs	SBS^C	α
F= {F_1, F_2……F_M}	F= {F_1^c, F_2^c, F_3^c, F_4^c}	F^c =1	β (f_k^c)
P= (p_1, p_2,….. p_l)	P= {p_1, p_2…p_l}	p^c	γ (P^C)

If another user installs the same apk file, the user first checks if the executable file hash value is already known as a suspected file identity on the blockchain.

➢ If a blockchain has the same file hash value, then the malware detection system can decide if the file is malicious and submit the result as a vote (malicious or benign) to the blockchain network.

➢ The detection system Then determine whether to uninstall the suspicious file based on the results of the blockchain vote and the results of its own malware evaluation.

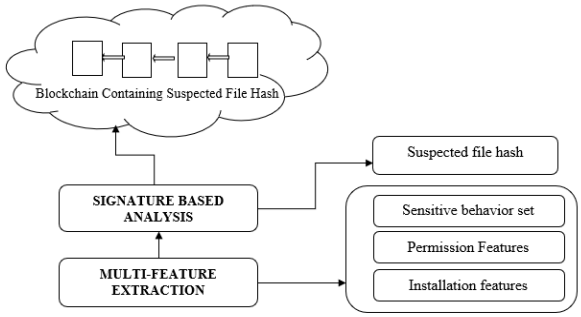

Figure 1. Overall Architecture Diagram

B. Feature Extraction

- Android-based device features can be derived from a range of features including package configuration features, program and authorization features, system call sequence features, and system call context.
- SBS (Sensitive Behavior Set) is a set of secure sensitive methods, attributes in malware installation Packages, and software applications permission documents.
- Essential Feature Representation (CFR) calculates F as a 0-1 software feature vector for testing if a lib / so file exists, and P as a software application permission list.
- Multi-Feature Model (MFM) is developed by extracting malware family features and creating multiple marking functions, and taking the probability of certain sample behaviors as weights in the malware family.

C. Multi Feature Detection

- VC Improving the identification capability of malware variants by naming a blurred comparison process
- functions to detect the multi-feature is to be used.
- The correlation between the program sample features and a certain family of malware is thus determined [SBS(Sensitive Behavior Graph),F(Feature Set),P(Permission List).]

a) Fuzzy comparison

- Calculating the similarity for the obtained SBS^C, F^C, P^C and add a correction factor ω (number of all sets (ω_{abs}, ω_f, ω_p)). Here the proportion of the similar elements in the two sets is greater than 80%

$$S_{SBS} = \omega_{SBS} \sum_{i=1}^{n} \alpha(S_i^c)\zeta(S_i^c), S_i^c \in SBS^C \quad (1)$$

$$\zeta(S_i^c) = \begin{cases} 0, \ if \ semiContain(SBS, S_i^c) = false \\ 1, \ if \ semiContain(SBS, S_i^c) = true \end{cases}$$

$$S_p = \omega_p \sum_{i=1}^{n} \gamma(p_i^c)\psi(p_i^c), p_i^c \in P^C \quad (2)$$

$$\varphi(p_i^c) = \begin{cases} 0, \ if \ p_i^c \notin P \\ 1, \ if \ p_i^c \in P \end{cases}$$

$$S_f = \omega_f \sum_{i=1}^{m} f_i f_i^c)\beta(f_i) \quad (3)$$

- Thus, calculate the similarity between features of a software sample and a certain malware family as

$S_{score} = S_{sbs} + S_f + S_p$ (4)

- Choose the average value of the outcomes of similarity measurements and decide whether the threshold exceeds If the threshold is reached, it is assumed to be malware and the corresponding malware family ID is released. Otherwise, the check software is assumed to be the original software.

b) Signature sharing

- If the downloaded executable file is considered malware, the user sends the hash value as a presumed identity to the blockchain network of the file.
- When a blockchain has the same file hash value, the user's heuristic or behavior-based malware detection system will decide whether the file is malicious, and the result. will be sent to the blockchain network as a vote (malicious or benign).
- If the file hash value on the blockchain does not exist, and the malware detection system assumes that the downloaded file is malware the machine sends the file

hash value to the blockchain network to transmit it.It also measures the degree of maliciousness and removes the file downloaded according to the result.

c) Measures against Mass Voting by Misleading Users

- The database information stored in includes
- Hash Value, Address of users who have voted, and Votes
- Access the web to participate in the Ethereum blockchain network and register an address to be used for voting Fig. 1.2.
- The Web server accesses the Ethereum blockchain Register Smart Contract and records the address.
- After that, the user enters the network and gets signatures, registers, and votes.
- The Vote Smart Contract, which is responsible for the procurement, authentication and voting of signatures, accesses the Register Smart Contract and checks if the Register Smart Contract includes an address.
- Where the address exists, the creation, registration and vote of the signature shall be accepted; otherwise they shall be refused

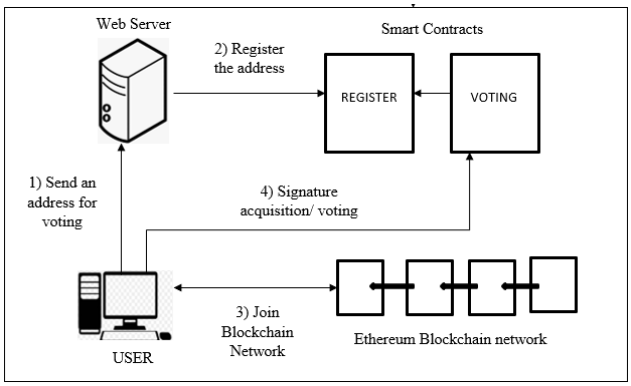

Figure 2.1. Signature sharing process

4. RESULT ANALYSIS

Data sets and Experimental environment Setting

We perform our experiments on the Intel(R) Core (TM) i5-6200U CPU with 8 GB of main memory. The operating system consists of Ubuntu. Also, to build a blockchain we installed truffle suite to deploy register and vote smart contracts. Android studio is used to perform the feature extraction and detection of malware. The malware dataset comes from **CICAndMal2017.**We installed 5,000 of the collected samples (426 malware and 5,065 benign). Our malware samples in the CICAndMal2017 dataset are classified into four categories,

- Adware
- Ransomware
- Scareware
- SMS Malware

Experiments of malware Detection

In order to evaluate the detecting capability of our malware detection system a set of benign software were mixed with malware samples and given as input data, the feature extraction were carried out for each software and based on the MFM model, the software were classified as malware and benign for the given set of samples. Only few samples of dataset were used in this experiment to test the accuracy, all the input samples were detected accurately in less time as shown in Table 1(Only few files and details are shown).

Evaluation Parameter

The parameters used to evaluate the accuracy of the proposed system are as follows

• FPR (False Positive Rate) -a software that does not belongs to a malware family but the system judges it as malware software

• FNR (False Negative Rate) - a software that belongs to a malware family but the system judges it as benign software

Table I

Table 1.1: Feature extraction by pre-static, static and dynamic analysis

Md5 value	***Sha256***	***VT positives***	***Analysis time(sec)***
0d33685eff9b6df4e87d799689e99a32	5d3deaa5c24e304f17cc511d2371c393f43b78b29b07b252db7936a35a70511d	32	***3.5***
0f52bf7d7b6cf39cbe438b141372e923	e6939b3784f2e10032abd4e5ada25adc81f1afa7dd6887f1b577db88e0c27af2	29	4.78
0b2f0c2ddf4a6a1101036286d75709f8	e4d65c43aeff2ced0258c39d57d80a24ee6fa9d2ccf845ca1a8c887e58ce5b96	33	4.2

$$FNR = \frac{FN}{TP + FN}$$

$$FPR = \frac{FP}{TN + FP}$$

Where,

FP ⟶ False Positive
FN ⟶ False Negative
TP ⟶ True Positive
TN ⟶ True Negative

5. CONCLUSION

In this project, a framework Malware is designed to be identified and listed on Android-based mobile devices using Blockchain technology. Different Malware Community Features are analyzed, and a malware feature model--*MFM* is constructed. The generated hash value sends to the Ethereum Blockchain network to determine the final result. The experimental The results of the CICandMal data set and the benign software data set show that the proposed program can effectively detect and identify known malware and identify malware on unknown software with higher accuracy and lower time-consuming.

REFERENCES

[1] Gu, Jingjing, Binglin Sun, Xiaojiang Du, Jun Wang, Yi Zhuang, and Ziwang Wang. "Consortium blockchain-based malware detection in mobile devices." IEEE Access, vol 6, pp.12118-12128, 2018.'

[2] Kim, TaeGuen, BooJoong Kang, Mina Rho, Sakir Sezer, and Eul Gyu Im. "A multimodal deep learning method for Android malware detection using various features." IEEE Transactions on Information Forensics and Security 14, no. 3, pp. 773-788, 2018.

[3] Li, Jin, Lichao Sun, Qiben Yan, Zhiqiang Li, Witawas Srisa-an, and Heng Ye. "Significant permission identification for machine-learning-based android malware detection." IEEE Transactions on Industrial Informatics 14, no. 7 , pp.3216-3225, 2018.

[4] Iqbal, Shahrear, and Mohammad Zulkernine. "SpyDroid: A Framework for Employing Multiple Real-Time Malware Detectors on Android." IEEE, 13th International Conference on Malicious and Unwanted Software (MALWARE), pp. 1-8, 2018.

[5] [5] Howard, Michael, Avi Pfeffer, Mukesh Dalai, and Michael Reposa. "Predicting signatures of future malware variants." IEEE, 12th International Conference on Malicious and Unwanted Software (MALWARE), pp. 126-132. 2017.

[6] [6] Ryota Hashimoto, Katsunari Yoshioka, and Tsutomu Matsumoto." Evaluation of Anti-Virus Software based on the Correspondence toNon-Detected Malware" (in Japanese) Distributed Processing System (DPS) (2012): 1-8

[7] .[7] Nakamoto, Satoshi." Bitcoin: A peer-to-peer electronic cash system." (2008).
[8] [8] uPort.me at: https://www.uport.me/ (accessed 2018/12/02).

[9] [9] R.Uncheck, "Mobile malware evolution 2016," Kaspersky Lab., Tech. Rep. 28. Feb. 2017.

[10] [10] N J. Percoco, and S. Schulte, "Adventures in Bouncer Land -Failures of Automated Malware Detection within Mobile Application Markets," Trustwave Holdings, Inc., Chigo, USA, Tech. Rep. 2012.

[11] [11] X. Du, et al., "Secure and Efficient Time Synchronization in Heterogeneous Sensor Networks," IEEE Trans. on Vehicular Technology, vol. 57, no 4, pp. 2387-2394, July 2008.

[12] [12] M. Grace, et al., "Riskranker: scalable and accurate zero-day android malware detection," in Proc. of the 10th Int. Conf. on Mobile systems, applications, and services. Low Wood Bay, Lake District, UK, 2012, pp. 281-293.

[13] [13] Y. Xiao, X. Du, J. Zhang, and S. Guizani, "Internet Protocol Television (IPTV): the Killer Application for the Next Generation Internet," IEEE Commun. Magazine, vol. 45, no. 11, pp. 126–134, Nov. 2007.

[14] [14] L. Wu, X. Du, and X. Fu, "Security Threats to Mobile Multimedia Applications: Camera-based Attacks on Mobile Phones," IEEE Commun. Magazine, vol. 52, no. 3, pp. 80-87, Mar. 2014.

[15] [15] M. Nofer, P. Gomber, O. Hinz, and D.Schiereck, "Blockchain," Business & Info. Systems Engineering, vol. 59, no. 3, pp. 183-187, Mar. 2017.

International Innovative Research Journal of Engineering and Technology
ISSN: 2456-1983 Vol: 5 Issue: 3 March 2020

BIOGRAPHIES

Dr.P.Boobalan was born on July 24, 1969. He has obtained Master of Technology in Computer Science and Engineering and Ph. D. in Computer Science and Engineering from Pondicherry University. He is an Associate Professor of the Department of Information Technology in Pondicherry Engineering College, Puducherry, India. He is a life member of Indian Society for Technical Education (ISTE).

Keerthana R, She is a student in Pondicherry Engineering College, pursuing B.Tech in the department of Information Technology

Vignesh P, He is a student in Pondicherry Engineering College, pursuing B.Tech in the department of Information Technology

Nandhini K, She is a student in Pondicherry Engineering College, pursuing B.Tech in the department of Information Technology