An Enhanced Method for Identifying Android Malware Detection

P. Jayanthi, K. Nirmaladevi, N. Krishnamoorthy

Abstract—In worldwide, all people are living with mobile applications for most of their life time. A statistical survey shows that mobile user exceeds 5 billions by 2019. There is a necessity to download different kinds of applications in different occasions. The library in android OS used for displaying media content has multiple vulnerabilities which enable the attackers to exploit media files and run the malicious code. The new ranges of attacks have been opened up today. The malware application does fraudulent activities automatically in the mobile without the knowledge of users. It is very difficult to identify the malware among such applications. Thus a challenge rises for protecting the mobile phones from these attacks. The existing method, "Significant Permission Identification for Machine-Learning-Based Android Malware Detection (SIGPID)", which uses Multi-Level Data Pruning process to identify significant permissions. In SIGPID, three level pruning process namely Permission Ranking with Negative Rate (PRNR), Support based Permission Ranking (SPR) and Permission Mining with Association Rule (PMAR) are applied to the dataset followed by SVM classification. The large dimension of the dataset negatively affects the malware detection efficiency. To reduce features of malicious apps further, an enhanced method called "Enhanced Model of Significant Permission Identification (ESID)" is proposed to identify android malware applications using data mining techniques. It adds the process to remove non-significant permissions and to classify the benign apps and malicious apps using SVM before installing an android application in the mobile. The experimental result shows that the better accuracy of 93.75% in identifying the malicious apps..

Keywords: Android Malware Detection, Datamining Techniques, Rank Based Approach, Feature Reduction

I. INTRODUCTION

Nowadays, the mobile device becomes an essential part of life of the people. For most of the activities in our day-to-day life is being completed with the help of the mobile device[19]. It is necessary to protect the data stored in the mobile from malware applications.

Because of presence of the mutli-vulnerabilities in the android mobile devices, the attackers may easily hack the data stored in the mobile, which leads to the possibility of fraudalent activities. The permissions asked during installation may provide the ability to access the mobile data without of much difficulty. Hence Android Malware Detection system is necessary for mobile devices. If it detects an android application is a malware one during installation of that application, it sends a notification to the user; hence it prevents the mobile from malware app installation.

Revised Version Manuscript Received on 16 October, 2019.

Dr. Jayanthi P. Department of Computer Science and Engineering, Kongu Engineering College, Perundurai, Erode, Tamilnadu, India.(Email:pjayanthikec@gmail.com)

Dr.Nirmaladevi K., Department of Computer Science and Engineering, Kongu Engineering College, Perundurai, Erode, Tamilnadu, India.(Email:k_nirmal@kongu.ac.in)

Dr. Krishnamoorthy N., Department of Computer Science and Engineering, Kongu Engineering College, Perundurai, Erode, Tamilnadu, India. (Email: krishnamoorthyer@gmail.com)

II LITERATURE SURVEY

Different approaches [1] [7] [10] [11] [16] [18] [22] are being used in the detection of malware apps. Sample approaches have been described in the following sections.

Wanga, et. al[23], illustrated a method to find the malware applications among the other applications. It uses different classification approaches like K-Nearest Neighbor (KNN), Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes (NB) and Classification and Regression Tree (CART) as ensemble classifiers. To characterize the behaviors of the applications, eleven different varieties of features which are static in nature are extracted.

Milosevic, et. al, [10], demonstrated the usage of machine learning approach for detecting malware applications. It includes two different methods to analyze the android malware applications statically. The permissions based method is used first and the second method uses a bag-of-words representation model to analyze the source code. The results shows that an F-score of 95.1% and F-measure of 89% for these methods respectively.

Lou, et.al,[20] shown that a novel method TFDroid used to detect malware apps by topics and sensitive data flows using machine learning techniques. The results show that this method can correctly identify 93.7% of all malware.

Meenu Ganesh, et. al., [15] developed an android malware detection method which investigates permission patterns based on a convolutional neural network. The result shows the accuracy of 93% in identifying the malware apps among a set of 2000 malicious and 500 benign apps.

Hui-Juan Zhu, et.al, [9] developed an effective and robust detection model namely, "DroidDet"which uses static analysis to extract the features like permissions, sensitive APIs, monitoring system events and permission-rate and employ the rotation forest model. The results show that it achieves accuracy of 88.26% with 88.40% sensitivity at the precision of 88.16%.

Chit La PyaeMyoHein andKhin Mar Myo, [2] have selected the features based on permissions to detect the malware applications. It is based on manifest file analysis to reduce the features of applications. It is followed by Score-based Approach which uses correlation and information gain.

Other approaches used network flow-based features [13], defense techniques [14], and sensitive data flow [20] to identify the malware apps.

Jin Li, et. al[12], implemented "Significant Permission Identification for Machine-Learning-Based Android Malware Detection (SIGPID)", which uses Multi-Level Data Pruning process to identify permissions which have impact on the system. The large dimension of the dataset negatively affects the malware detection efficiency. Hence three level pruning process namely Permission Ranking with Negative Rate (PRNR), Support based Permission Ranking (SPR) and Permission Mining with Association Rule (PMAR) are applied to the dataset. Then SVM classification method is applied to classify the malware apps and benign apps.

To improve the existing system (SIGPID), identification and elimination of non-sensitive columns has been introduced in the proposed system. In this paper section 3, illustrates the phases in the proposed system, section 4 shows the implementation of the proposed system, section 5 summarizes the results found and section 6 concludes the paper with future expansion of the work.

III PROPOSED SYSTEM

The main objective of the proposed system is to detect android malware applications and to protect the mobile devices. An enhanced model of Significant Permission Identification (ESID) is proposed to identify dangerous malicious applications by considering the list of permissions associated with them. Different phases of the proposed model are explained in the following section.

A. Phases of Proposed System:

The overall architecture of the proposed model ESID includes different phases as shown in the figure 1.

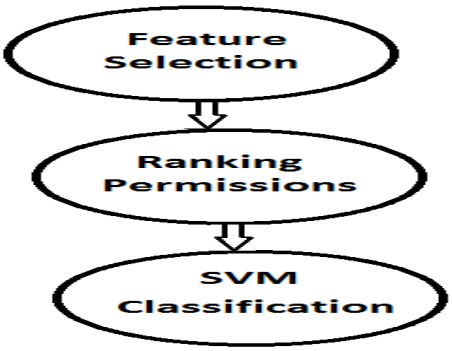

Fig. 1 Phases in the proposed system

The android malware dataset with their permission list is given as input for the system. The proposed model starts with the phase feature selection which eliminates the non-sensitive permissions from the dataset. Then permission rankings have been done based on various strategies. Finally the classification is done using the SVM approach. The steps in the above three phases are explained in the following sections.

B. Phase I - Feature Selection: Elimination of Non Sensitive Columns

While downloading and installing various android applications from various resources, the applications require permissions from the user. The permission represents a specific operation such that an application is allowed to perform. Permission INTERNET refers that whether an application can access to the Internet.

The features i.e. permissions listed in the dataset do not have significant effect in the detection of malware applications. The proposed system focuses on reducing the number of non-sensitive permissions. Non-sensitive permissions mean that most of the values in the column is almost same, the reason behind the elimination of the columns are they are insensitive to any calculations. In this phase, the model focuses on finding such non-sensitive columns and eliminating those columns from the dataset. Thereby reducing the number of permission list from the original dataset helps in achieving higher accuracy than the existing system.

Dataset is in the form of matrix, where rows represents the android applications and column represents the permission used by the android applications. Let the matrices M and B represents malware apps and benign apps respectively. The permissions required are represented using the following criteria as shown below:

M_{ij}= 1, if i^{th} malware application needs j^{th} permission
= 0, otherwise.
B_{ij} = 1, if i^{th} benign application needs j^{th} permission
= 0, otherwise.

The non-significant permissions are to be eliminated based on the following criteria as shown below:

(i) For each j^{th} permission in $M_{i,}$
M_j =eliminated, if j^{th} permission = c, for M_i, 0< i< n
= not eliminated, otherwise.
(ii) For each j^{th} permission in $B_{i,}$
B_j =eliminated, if j^{th} permission = c, for B_i, 0< i< n
= not eliminated, otherwise.

The above said criteria are applied to the dataset. The reduced feature data set is used for the next phases.

C. Ranking Permissions

The permissions requested for both benign apps and malicious apps needs to be analyzed to make an effective malware detection system. The permissions with high-risk attacks, permissions frequently asked, rarely asked permissions, commonly asked permissions are to be analyzed. It becomes an essential activity to differentiate the type of applications. While excluding the commonly asked permissions, a care to be taken for identifying the way of using such permissions by both benign apps and malicious apps.

The feature reduced dataset i.e. M and B matrices from the phase 1 to be given as input to the phase 2 to reduce the permissions based on the ranking method. The balanced matrices are appropriate for ranking method. The support value of a permission of a larger one scales down that of the same permission of a smaller one. For the alternate case, then the support of each permission SB(P) is calculated as shown in equation 1 :

$$\mathbf{SB\ (P_j\ ) = \sum B_{ij}\ size(B_j) * size(M_j\ )} \quad (1)$$

where $SB(P_j)$ denotes the support of jth permission in B_i app. Then Permission Ranking with Negative Ranking (PRNR) can be implemented as shown in equation 2 :

$$\mathbf{R(P_j) = \sum M_{ij} - SB(P_j) / \sum M_{ij} + SB(P_j)} \quad (2)$$

The values of $R(P_j)$ falls in a range of (-1, 1). The permission P_j needed only in the malware applications is considered as high risk with a value 1. Similarly the permission Pj needed only in the benign applications is considered as low risk with a value -1. The permission having the value $R(P_j) = 0$ is noted for its insignificant effect in malware detection system.

D. Support Based Permission Ranking

If a dataset contains more number of features or attributes, the results may be overfitting. To get the more accurate classification, the features may be reduced properly. The low support value of permission does not affect the performance metrics used in the malware detection approaches. Consider the permission BIND_TEXT_SERVICE which is used only in benign applications. So, any application which uses this permission is considered as benign app.

E. Permission Mining with Association Rule

Association rule mining [17] is a process through which the permissions always appear together can be found. The permissions for writing SMS and reading SMS are always used together. Consider only one of this permission is enough to characterize some of the behaviours in the applications. The other one associated with this permission can be found and to be eliminated further.

F. Classification of Applications

To classify the benign app and malicious app, the Support Vector Machine (SVM) is used. The feature reduced dataset obtained from the previous ranking phase is given as input to this classification phase. It classifies the malicious app and benign app into two groups.

IV EXPERIMENTAL ANALYSIS

A. Experimental Setup

To experiment the proposed method with the real time android maleware dataset, the system with Intel octa core processor, 8 GB RAM and disk capacity of 256 GB SSD is used. The program was written using R Studio Version 3.4.4 [8] and executed in a Windows 10 operating system environment.

B. Dataset Description

The proposed system is experimented with two datasets. The different datasets [4] are available on the web. The first dataset [5] used from android malgenome project includes 1260 malware apps and 2539 benign apps. The dataset includes 215 features extracted from 3799 applications. The second dataset [6] used in Drebin project [3] includes 5,560 malware apps and 9,476 benign apps. It consists of 215 attributes extracted from 15,036 applications.

The sample features are explained below: (i)GET_SYSTEM_DIRECTORYA – used to retrieve the path of the system directory which contains system files such as dynamic link libraries and drivers.

(ii) WRITE_SMS - used to allow an application to send SMS messages which has been approved as top ten dangerous permissions by the Google.

(iii) CAMERA – used to enforce automatically the uses-feature manifest element for all camera features.

(iv) WRITE_CONTACTS - used to allow an application to write the user's contacts data. The protection level of this permission is dangerous.

(v)ACCESS_BACKGROUND_LOCATION – used to allow an application to access location in the background which requests ACCESS_FINE_LOCATION. Requesting by itself is not sufficient to give the location access. The protection level for this permission is dangerous. Attributes have only two values 0 and 1. If the attribute value is 0, then the permission is not used by the particular app. If the attribute value is 1, then the permission is used by the particular app.

V RESULTS AND DISCUSSIONS

After the removal of non-sensitive permissions in the phase 1, the features of dataset1 has been reduced by 38 such permissions and that of the dataset2 has been reduced by 47 permissions. This is an added process to the existing system SIGPID.

To verify the performance of the proposed system, the two data set described in section 4.2 are experimented. A comparison is made between existing system (SIGPID) and the proposed system (ESID). The metrics used are precision, recall and accuracy. In existing system 22 permissions have been identified as the dangerous permission after applying three pruning levels but in proposed system 8 permissions in dataset1 and 12 permissions in dataset2 have been identified as the dangerous permission after applying feature selection along with the three pruning levels.

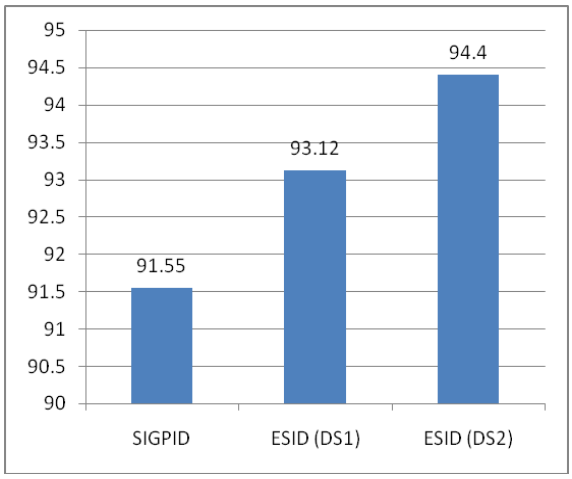

Fig. 2: Precision values

The above figure 2 shows the results of comparison made between the precision values obtained in SIGPID and that of the proposed system. It is obvious that the precision value obtained is 91.55 but in the proposed system the precision value is 93.12 and 94.4 for dataset1 and dataset2 respectively.

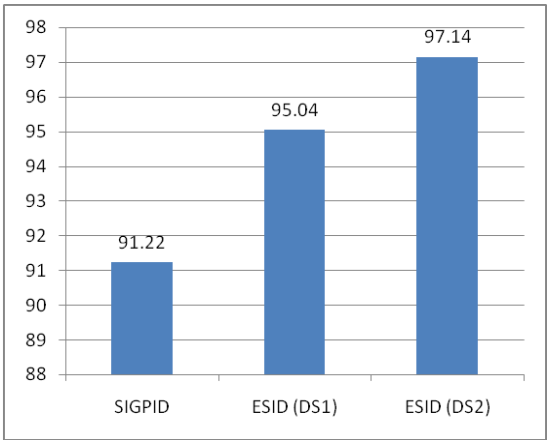

Fig. 3: Recall values

From the figure 3, it is observed that the recall value obtained for SIGPID is 91.22, whereas in the proposed system the recall value obtained is 95.04 for dataset1 and 97.14 for dataset2. From figure 4, it clearly shows that the proposed system provides better accuracy value than the existing system. i.e. accuracy value obtained in the existing system (SIGPID) is 91.34 and that of the proposed system is 93.68 and 93.75 for the dataset1 and dataset2 respectively.

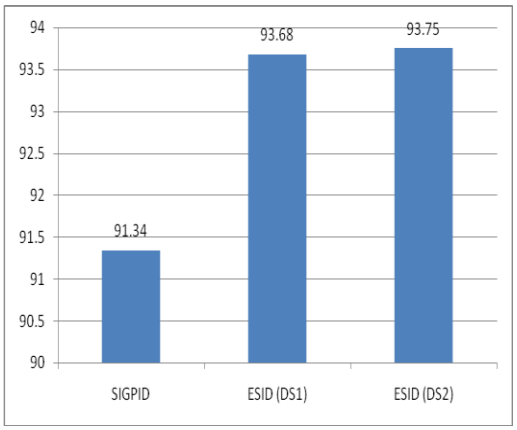

Fig. 4: Accuracy values

VI CONCLUSION

This proposed framework demonstrated the possibility of reducing the features by considering the significance of them. Then for further reduction, a systematic three-level pruning approach is followed. The proposed system uses only 12 permissions in the pruning phase. When compare to the existing system, significant number of permissions were reduced due to non–sensitive permission feature reduction. Because of the reduced features, the proposed system shows the better accuracy 93.68 and 93.75 respectively for the dataset1 and dataset2 respectively than the existing system.

Furthermore, the enhancements can be made by identifying such more permission sets by using advanced techniques like deep learning, etc. and apply a more suitable classification approach to reach most accuracy in the identification of malware apps and benign apps.

REFERENCES

1. Chandini S B, Rajendra A B, NitinSrivatsa G, "A Research on Different types of malware and detection techniques", International Journal of Recent Technology and Engineering, Vol. 8, Issue-2S8, August 2019.
2. Chit La PyaeMyo Hein, Khin Mar Myo, "Permission-based Feature Selection for Android Malware Detection and Analysis", International Journal of Computer Applications, (0975 – 8887), Volume 181 – No. 19, September 2018.
3. D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, K. Rieck, and C. Siemens, "DREBIN: Effective and explainable detection of android malware in pocket," presented at Annu. Symp. Netw. Distrib. Syst.Security, 2014.
4. G. DATA, "8,400 new android malware samples every day." 2017. [Online].Available: https://www.gdatasoftware.com/blog/2017/04/29712-8-400-new-android-malware-samples -every-day
5. https://figshare.com/articles/Android_malware_dataset_for_machine_ learning_1/5854590/1
6. https://figshare.com/articles/Android_malware_dataset_for_machine_ learning_2/5854653/2
7. https://www.tutorialspoint.com/data_mining/dm_overview.html
8. https://www.tutorialspoint.com/r/index.htm
9. Hui-Juan Zhu, Zhu-Hong You, Ze-Xuan Zhu, Wei-Lei Shi, Xing Chen, Li Cheng, "DroidDet: Effective and robust detection of android malware using static analysis along with rotation forest model", Neurocomputing 272 (2018) 638–646.
10. Ikola Milosevic, Ali Dehghantanha, Kim-Kwang Raymond Choo, "Machine learning aided Android malware classification", Computers and Electrical Engineering, Vol. 61 (2017) 266–274.
11. Ji Wang, Qi Jing, JianboGao, "SEdroid: A Robust Android Malware Detector using Selective Ensemble Learning", CCS '19, November 11–15, 2019, London, UK.
12. Jin Li, Lichao Sun, QibenYan, Zhiqiang Li, WitawasSrisaan, and Heng Ye, "Significant Permission Identification for Machine-Learning-Based Android Malware Detection", IEEE Transactions on Industrial Informatics, Vol. 14, No. 7, July 2018.
13. Joshua Sopuru, Arif Sari, Murat Akkaya, "Modeling A Malware detection and categorization system based on seven network flow-based features", International Journal of Innovative Technology and Exploring Engineering,Vol.8, Issue 7, May 2019, 2982-2989.
14. MA Rahim Khan, RC Tripathi, Ajit Kumar, "A Malicious Attacks and Defense Techniques on Android-Based Smartphone Platform", International Journal of Innovative Technology and Exploring Engineering,Vol.8, Issue 8S3, June 2019, 361-369.
15. Meenu Ganesh, PriyankaPednekar, PoojaPrabhuswamy, DivyashriSreedharan Nair, Younghee Park, "CNN-Based Android Malware Detection", International Conference on Software Security and Assurance (ICSSA), July 2017.
16. M. Grace, Y. Zhou, Q. Zhang, S. Zou, and X. Jiang, "RiskRanker: Scalable and accurate zero-day android malware detection," in Proc. 10th Int. Conf.Mobile Syst., Appl., Services, 2012, pp. 281–294.
17. R. Agrawal et al., "Fast algorithms for mining association rules," in Proc.20th Int. Conf. Very Large Data Bases, 1994, vol. 1215, pp. 487–499.

18. ShaikhBushraAlmin, MadhumitaChatterjee, "A Novel Approach to Detect Android Malware", International conference on advanced computing technologies and applications, ICACTA 2015, Procedia Computer Science 45 (2015) 407 – 417.
19. Statistics – "Cumulative number of apps downloaded from the GooglePlay as of may 2016," May2016,[Online], https://www.statista.com/statistics/281106/number-of-android-app-downloads-fromgoogle-play
20. Songhao Lou, Shaoyin Cheng, Jingjing Huang, Fan Jiang, "TFDroid: Android Malware Detection by Topics and Sensitive Data Flows Using Machine Learning Techniques", IEEE 2nd International Conference on Information and Computer Technologies (ICICT), 2019.
21. W. Wang, X. Wang, D. Feng, J. Liu, Z. Han, and X. Zhang, "Exploring permission-induced risk in android applications for malicious application detection," IEEE Trans. Inf. Forensics Security, vol. 9, no. 11, pp. 1869–1882, Nov. 2014.
22. Wei Wanga, Yuanyuan Li, Xing Wang, Jiqiang Liu, Xiangliang Zhang, "Detecting Android malicious apps and categorizing benign apps with ensemble of classifiers", Future Generation Computer Systems 78 (2018) 987–994

AUTHORS PROFILE.

Dr. Jayanthi P has completed BE(CSE), ME(CSE), Ph.D from Anna University, Chennai, India. She is presently working as an Associate Professor in the Department of Computer Science and Engineering, Kongu Engineering College, Erode, India. Her research area focuses mainly on Data mining, Web Services and Cloud Computing. She has been presented papers in national, regional and international conferences and in high impact factor journals.

Dr. Nirmaladevi K has completed MCA, ME(CSE), Ph.D from Anna University, Chennai, India. She is presently working as an Assistant Professor(Selection Grade) in the Department of Computer Science and Engineering, Kongu Engineering College, Erode, India. Her research area focuses mainly on Data mining, Web Services and Big Data Analytics. She has been presented papers in national, regional and international conferences and in high impact factor journals.

Dr. Krishnamoorthy N has completed BE(CSE), ME(CSE), Ph.D from Anna University, Chennai, India. He is presently working as an Assistant Professor(Senior Grade) in the Department of Computer Science and Engineering, Kongu Engineering College, Erode, India. His research area focuses mainly on Grid Computing, Operating Systems and Data Structures. He has been presented papers in national, regional and international conferences and in high impact factor journals.