Journal of Theoretical and Applied Information Technology
15th March 2022. Vol.100. No 5
2022 Little Lion Scientific

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CLASSIFICATION OF ANDROID MALWARE TYPES USING SUPPORT VECTOR MACHINE

[1]HENDRA SAPUTRA, [2]AMALIA ZAHRA
[12]Computer Science Department, BINUS Graduate Program, Master of Computer Science,
Bina Nusantara University, Jakarta, Indonesia 11480
E-mail: [1]hendra.saputra@binus.ac.id, [2]amalia.zahra@binus.edu

ABSTRACT

In the past years, the amount of Android malware is on the increase. This statement is supported by the data from VirusShare showing an increase in the amount of malware each passing year. Hence it is necessary to classify the malware for identifying types of malware attacking smartphone which consequently will help resolve the issue easier. For addressing the issue, this research classifies Android malware based on their types. The attributes employed are activities, permission, and receiver located inside the Androidmanifest.xml file. This research obtains the malware data from VirusShare database in 2018. Classification algorithms used was Support Vector Machine (SVM) with RBF kernel and k-fold = 10. In addition, this research also employed gain ratio feature selection to minimize unnecessary attributes on the data. The accuracy of classification using feature selection was 72.5%. This number was 0.3 lower compared to the classification result without feature selection with the accuracy of 72.8%. However, the data classified using feature selection can reduce the process of classification model creation by 206 seconds.

Keywords: *Malware Android, SVM, Euphony, Gain Ratio, Virusshare, Virustotal*

1. INTRODUCTION

Android is the name of an operating system in a gadget such as smartphone or tablet currently that is being widely used compared to others. Android provides an open platform for developers to create applications to be used by users across cellular devices. According to appbrain.com, there are an array of Android applications available in 2021. Three million applications to be exact [1].

As the number of Android based smartphone users increases among government officials, citizens, and companies, there is bigger chance for intruders to target and commit crime using malicious software or commonly referred to as malware. Malware is a software explicitly designed to perform dangerous activities or destroy other devices, for example Trojan, virus, spyware, and exploit [2]. Malware activity can be malicious to users because it steals private information, damages the system, and taps as well as monitoring smartphones activity. This is a proof of cybercrime committed by intruders with the help of malware. Google has developed cloud-based security scanner named bouncer aimed at detecting malicious applications in Play Store. Despite the measure taken, applications carrying malware still manage to infiltrate Play Store. Because of the lack of caution from users in using third-party applications, their smartphones are still in danger of being infected by malware. As a result, a large number of researches has been conducted in the topic of malware classification in Android applications by employing machine learning statically or dynamically.

In order to obtain satisfactory result, research on the classification of applications carrying malware or benign employing Bayesian classifier which focuses on improved mutual information (IMIFS) as feature selection will be used [3]. Research which employs permission and API Calls as features for classifying applications carrying malware or benign with the help of SVM algorithm, J48, and Bagging shows accuracy rate of more than 90% [4]. The research which classifies Android applications containing malware or benign by using permission and base-code property as the feature and Bayesian as classification algorithm also provides good result. Before the data are tested, they are undergoing feature selection by using Mutual Information (MI) [5]. The research on the topic of malware or benign classification of Android applications using deep learning algorithm also provides good result [6]. Research conducted using hybrid feature on logistic regression algorithm, naive bayes, and random forest provides a moderately accurate classification of Android malware [7]. While for research using clustering method of Android malware which employs Community Detection algorithm with network similarity approach also provides a quite satisfactory result [8].

Despite the variety of researches, none of them analyzes the detection of malware types. The detection of malware types is highly essential for researches aiming at delving into malware. There are plenty researches focusing on one type of malware, however researchers are in difficulty to obtain several malwares by merely relying on one type. Several malware repository providers such as antivirus programs also set out stringent requirements for obtaining the said malware. Even though there is provider such as VirusShare, the malwares in their repository are unlabeled, which makes researchers experience difficulty in identifying the types of malware. Because of those reasons, it is expected that this research will be helpful for other researchers in collecting the desired malware types. Malware types detection is also necessary as the continuation for previous researches which only detect malware or benign. once a file has been detected as carrying malware, further action is needed to determine the type so that smartphone users know how to deal with problems caused by the malware based on the attack pattern and the attributes required by the malware.

Referring to the existing issue, this research focuses on the detection of Android malware types by employing static method and using dataset from VirusShare with unlabeled malware. The result from uploading malware files to Virustotal exhibit vast differences in the detection of malware types among antivirus programs. This makes the process of malware labelling incoherent. In order to address the dataset labelling issue, the researcher uses Euphony can parse malware labels from Virustotal and produce a single family per file [16]. The features which will be used on the dataset are activities, permission, and receiver. Machine learning method used is SVM. This method is chosen due to the fact that in the previous researches it provides a quite satisfactory result in detecting malware types or benign. Before applying machine learning algorithm, the data dimension will be reduced by using Gain Ration feature selection and Ranker method. Attribute which shows no value will be removed to minimize the data dimension prior to classification. The attribute removal is expected to prevent bias and speed up classification process.

2. RELATED WORK

Researches on Android malware by employing machine learning has been conducted quite frequently. The researches were conducted either by clustering or classifying malware and non-malware. Several machine learning algorithms such as SVM, Bayesian Classifier, J48, bagging and k-Means provide moderately optimal result as seen in Table I.

Table 1: Related Work.

No	Papers	Title	Algorithms	Result
1	[5]	Analysis of Bayesian Classification Based Approaches for Android Malware Detection	Bayesian classifier	AUC > 0.92
2	[3]	An Innovative Technique to Detect Malicious Applications in Android	Bayesian classifier	There is an increase in accuracy by using the feature selections
3	[4]	Machine Learning for Android Malware Detection Using Permission and API Calls	Support Vector Machine (SVM), J48 dan Bagging	Accuracy > 92%
4	[6]	Android Malware Detection Using Deep Learning ON API Method Sequences	Deep Learning	Accuracy > 95%
5	[9]	Android Malware Classification Using K-Means Clustering Algorithm	Random Forest dan K-Means	Provides a comparison of the results from the Virustotal and malgenome datasets
6	[10]	Android Malware Clustering	Fuzzy Hashing	Algorithm is less suitable in

		through Malicious Payload Mining		Android malware similarity analysis.
7	[11]	Android Malware Prediction by Permission Analysis and Data Mining	Logistic Regression Model, Tree Model with Ensemble techniques, Neural Network	From the model made gives good accuracy results
8	[7]	Ensemble Machine Learning Approach for Android Malware Classification Using Hybrid Features	Logistic Regression, Naïve Bayes, Random Forest	Able to provide classification results for malware types with fairly good accuracy.
9	[12]	Structural analysis and detection of Android botnets using machine learning techniques	Support Vector Machine	Able to classified files containing botnets and not
10	[8]	Android Malware Clustering using Community Detection on Android Packages Similarity Network	Community Detection	87% were successful at detecting Android malware

The research classified Android malware and benign [3]. Android applications are extracted by apk tools to obtain several attributes such as permission and code property from file class.dex. The next step is feature selection using Improved Mutual Information Feature Selection Algorithm (IMIFS). The data from feature selection will be sorted to eliminate redundant attributes for optimal classification result. Algorithm which is used in classification process is Bayesian. The result shows that the data selected using IMFS are better than those processed without feature selection. Research classifies malware and benign [4]. Attributes used in classification process are permission, API Calls, and com+. The process compares several algorithms, for example SVM, J48 and Bagging. It turns out Bagging algorithm with com+ dataset provides the most satisfactory result. Research employing permission and code property attributes in detecting malware and benign also provides satisfactory result [5]. Feature selection with Mutual Information algorithm and Bayesian Classifier are also accurate since they increase Area Under the Curve (AUC) to above 0,92. Research on malware and benign classification with API Calls as the attribute using deep learning algorithm and MalDozer framework provides an accuracy of more than 95% [6].

Referring to the literature review, all researchers have attempted to determine whether an application is a malware or non-malware by employing machine learning. It can be concluded, from the results of previous researches, that machine learning can aid researchers in differentiating malware from non-malware. However, further action is required to determine the type of malware in an application. By knowing the type of malware in each application, users can more easily solve problems caused by malware attacks. Based on those reasons, this research will focus on classifying malware types.

3. RESEARCH METHOD

The first step is collecting malware data from VirusShare. VirusShare is chosen because it does not set out as many requirements as known antivirus programs for obtaining malware dataset. The malware used from VirusShare is from their 2018 dataset. The workflow can be seen in Figure 1.

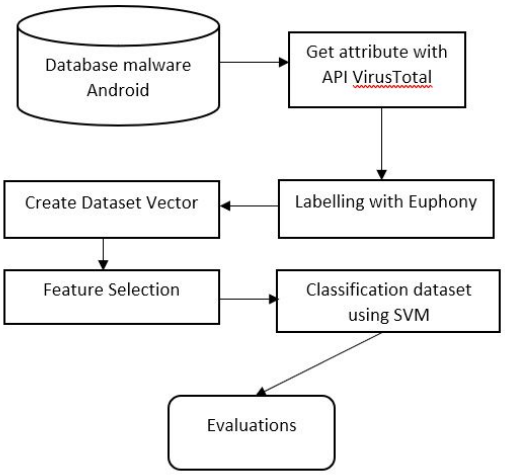


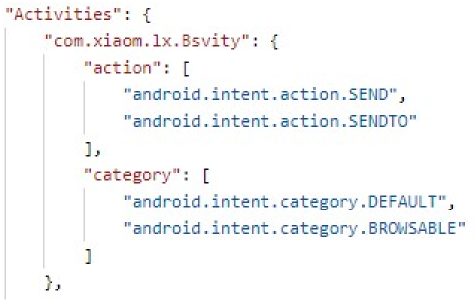
Figure 1: Receiver from API VirusTotal

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.1 Collecting Attribute Malware Android

The researcher will use the API from Virustotal in order to collect attributes namely activities, permission, and receiver as well as the detection of malware types to be used as reference for labelling with the help of Euphony application.

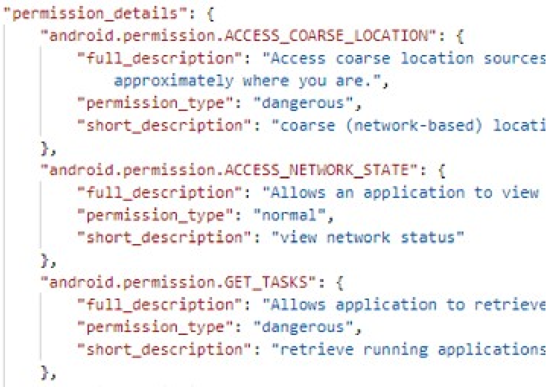
Activities is a component in Android application which displays and manages the screen as a way for users to interact with the application itself, for example making calls, taking pictures, sending messages, or sending emails.



```
"Activities": {
    "com.xiaom.lx.Bsvity": {
        "action": [
            "android.intent.action.SEND",
            "android.intent.action.SENDTO"
        ],
        "category": [
            "android.intent.category.DEFAULT",
            "android.intent.category.BROWSABLE"
        ]
    },
```

Figure 2: Activities from API VirusTotal

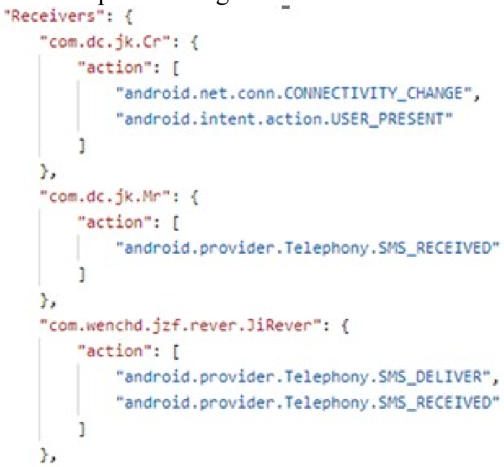
Permission is the most essential and common property used to detect malware in previous researches. A large number of researches have made successful attempt at developing classification model for detecting Android malware by using permission property. It is due to the fact that the permission property requested by an application can be potentially harmful to Android smartphones. One of the most commonly used permission by Android malware is short message service (SMS) permission for sending, reading, and receiving messages. This permission allows an application to receive and read the incoming messages as well as sending one. For example, a criminal can exploit this permission for subscribing to unwanted paid services unbeknownst to victim [13].



```
"permission_details": {
    "android.permission.ACCESS_COARSE_LOCATION": {
        "full_description": "Access coarse location sources
            approximately where you are.",
        "permission_type": "dangerous",
        "short_description": "coarse (network-based) locati
    },
    "android.permission.ACCESS_NETWORK_STATE": {
        "full_description": "Allows an application to view
        "permission_type": "normal",
        "short_description": "view network status"
    },
    "android.permission.GET_TASKS": {
        "full_description": "Allows application to retrieve
        "permission_type": "dangerous",
        "short_description": "retrieve running applications
    },
```

Figure 3: Permission from API VirusTotal

Receiver is a component in Android application waiting for broadcast message (or occurring event) from various sources including its own application, system, or other applications. Therefore, the researcher also uses receiver property as a feature for classifying because an application can monitor or send notification to other applications causing the smartphone being monitored.

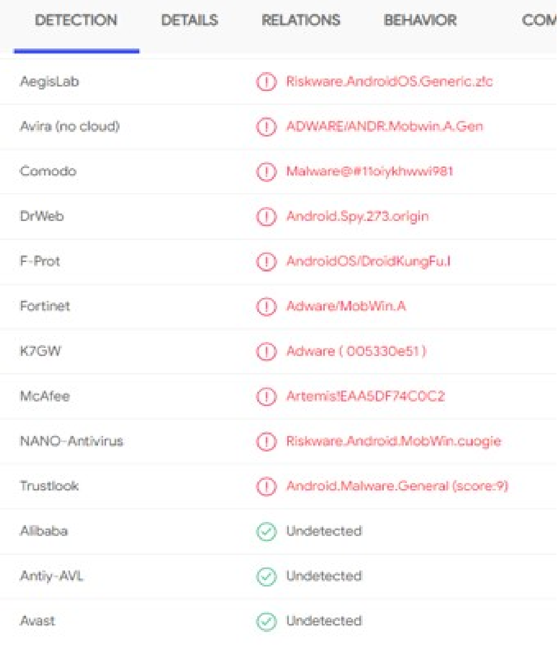


```
"Receivers": {
    "com.dc.jk.Cr": {
        "action": [
            "android.net.conn.CONNECTIVITY_CHANGE",
            "android.intent.action.USER_PRESENT"
        ]
    },
    "com.dc.jk.Mr": {
        "action": [
            "android.provider.Telephony.SMS_RECEIVED"
        ]
    },
    "com.wenchd.jzf.rever.JiRever": {
        "action": [
            "android.provider.Telephony.SMS_DELIVER",
            "android.provider.Telephony.SMS_RECEIVED"
        ]
    },
```

Figure 4: Receiver from API VirusTotal

3.2 Labelling Malware Types

The Dataset obtained from VirusShare contains unlabeled malware which causes differences when uploaded to Virustotal. The differences are in terms of malware types in the antivirus programs datasets provided by Virustotal as shown on Figure 4. The labelling will ease the researcher in evaluating the classification result.



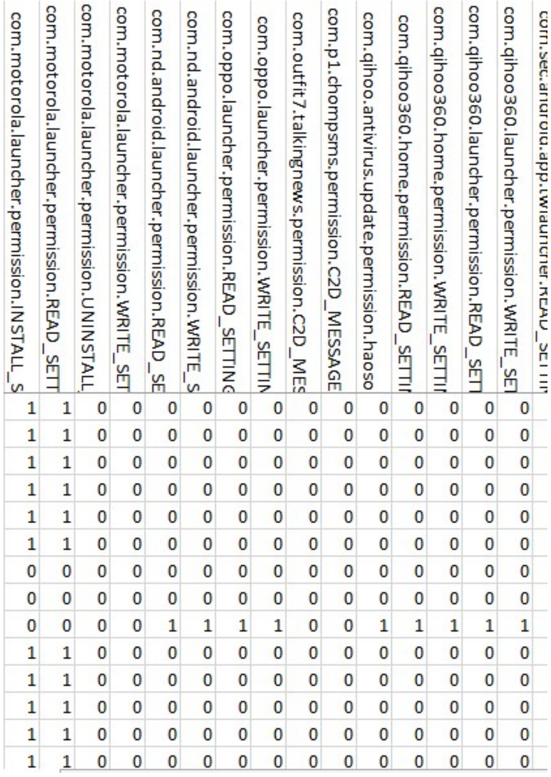
AegisLab	ⓘ Riskware.AndroidOS.Generic.zfc
Avira (no cloud)	ⓘ ADWARE/ANDR.Mobwin.A.Gen
Comodo	ⓘ Malware@#11oiykhwwi981
DrWeb	ⓘ Android.Spy.273.origin
F-Prot	ⓘ AndroidOS/DroidKungFu.I
Fortinet	ⓘ Adware/MobWin.A
K7GW	ⓘ Adware (005330e51)
McAfee	ⓘ Artemis!EAA5DF74C0C2
NANO-Antivirus	ⓘ Riskware.Android.MobWin.cuogie
Trustlook	ⓘ Android.Malware.General (score:9)
Alibaba	✓ Undetected
Antiy-AVL	✓ Undetected
Avast	✓ Undetected

Figure 5: Uploaded to Virustotal

The labelling process will be conducted using java-based Euphony application. Prior to processing the data, it is necessary to obtain all Virus total scan results for each data. Then the scan results will be processed in accordance with Euphony format. Once all data are labelled, the 10 most frequently appeared malware are then used for the classification. The amount of data from those 10 malware will not be the same, thus an adjustment will be made.

3.3 Creating Dataset from Malware Attribute

The process of creating dataset will be completed in two steps. The initial step is creating a feature where all attributes namely activities, permission and receiver are gathered. Any duplicates will be eliminated to avoid similarity in the feature. The second step, each attribute in the malware file will be compared to the features created during the initial step. If the attribute is present in the feature, then the value will be 1. If the attribute is not present in the feature, the value will be 0. Example of the dataset can be seen in Figure 5.



com.motorola.launcher.permission.INSTALL_S	com.motorola.launcher.permission.READ_SETT	com.motorola.launcher.permission.UNINSTALL	com.motorola.launcher.permission.WRITE_SET	com.nd.android.launcher.permission.READ_SE	com.nd.android.launcher.permission.WRITE_S	com.oppo.launcher.permission.READ_SETTING	com.oppo.launcher.permission.WRITE_SETTIN	com.outfit7.talkingnews.permission.C2D_MES	com.p1.chompsms.permission.C2D_MESSAGE	com.qihoo.antivirus.update.permission.haoso	com.qihoo360.home.permission.READ_SETTII	com.qihoo360.home.permission.WRITE_SETTII	com.qihoo360.launcher.permission.READ_SETT	com.qihoo360.launcher.permission.WRITE_SET	com.sec.android.app.twlauncher.READ_SETTII
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	1	1	1	1	0	0	1	1	1	1	1	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	

Figure 6: Example of dataset in vector

3.4 Feature Selection

Several features from data extraction are redundant or irrelevant. This condition can influence the performance result of machine learning algorithm. Hence feature selection is employed to minimize unnecessary or irrelevant features in the data. Gain Ratio is the chosen feature selection which will be used with scoring method for nominal attribute weighting or discrete and continuous data by using maximum entropy [14]. Features with no value will be eliminated to minimize the data for the sake of speeding up machine learning.

3.5 Classifying

SVM algorithm will be used in the classification process. The process is conducted with two data, they are data without feature selection and data with feature selection. Referring to previous researches, SVM exhibits higher endurance and generalization compared to other algorithms [15]. The value displayed by this algorithm is >90% on malware or non-malware detection by using permission and API Calls attributes [4].

3.6 Evaluation the Classification

In the classification step, both data, with and without feature selection, consisting of 10 types of malware will be classified by using SVM algorithm with RBF kernel and k-fold 10. The accuracy of both classification processes will be used to evaluate the performance of previously developed model.

4. RESULT AND DISCUSSION

The process of malware data labelling which refers to the 2018 VirusShare dataset using Euphony application displays uneven amount of malware as seen in Table II.

Table 2: Top 10 Types Android Malware from Virusshare 2018.

No	Name of Malware	Total
1	Shedun	10861
2	Revo	2989
3	Dnotua	1385
4	Jiagu	1355
5	Artemis	954
6	Triada	785
7	Smsspy	676
8	Debugkey	637
9	Wapron	604
10	Gexin	438

Referring to the data labelling process in Table II, there is a significant imbalance in the amount of data for each type of malware. Due to this reason, the maximum amount of malware used for the classification is 785 samples as seen in Table II. This measure is taken to ensure that the comparison of accuracy result is even.

Table 3: Total data that will be used for classification.

No	Name of Malware	Total
1	Shedun	785
2	Revo	785
3	Dnotua	785
4	Jiagu	785
5	Artemis	785
6	Triada	785
7	Smsspy	676
8	Debugkey	637
9	Wapron	604
10	Gexin	438

The total of 7,065 dataset samples used for the research as shown in table III generates 15,935 features. This enormous amount will be highly influential in the duration of classification process. Because of that, the data will be selected using gain ratio to minimize unnecessary features. After undergoing gain ratio feature selection, the amount of features which will be used is reduced to 858.

Table 4: Classification Results with SVM

Total Data	7,065 Samples
Before Feature Selection	15,935 Features
After Feature Selection	865 Features
SVM accuracy without feature selection	72.8013%
SVM accuracy with feature selection	72.5978%

The classification result of Android malware using SVM without feature selection shows an accuracy of 72.8%, while the ones classified using feature selection has an accuracy of 72.5%, 0.3% lower. Referring to the accuracy result, it can be concluded that the attributes of activities, permissions, and receivers can provide moderately satisfactory result. However, the result is less optimal compared to the previous research which simply classifying Android malware or non-malware with the accuracy rate of more than 90%. This is due to the fact that the malware is unlabeled which becomes the cause of labeling errors. The result of feature selection is also quite satisfactory even though the accuracy decreases by 0.3%. Nevertheless, the initial 15,935 features are successfully minimized to become 858 which consequently speeds up classification process.

5. CONCLUSION

Based on the classification test on 10 types of malware with the accuracy of approximately 72%, activities, permission, and receiver attributes in Android are proven quite effective in detecting malware types using SVM. In addition, the feature selection using Gain Ratio can minimize data dimension or unnecessary feature which helps to speed up machine learning process without significantly reducing the accuracy rate. Despite all those results, there are several errors in the detection process which may be caused by labeling errors due to unlabeled malware obtained from VirusShare. The detection result from Virustotal also shows differences among the antivirus programs which potentially cause obscurity in the labeling process.

As a result, it can be said that there are many android apks that have more than one malware label in the VirusShare 2018 database causing labeling errors. To overcome the error in the labeling of malware types, further research is needed to analyze the proximity of the attributes used in malware that has more than one type of malware label from virustotal.

REFRENCES:

[1] www.appbrain.com, "Number of Android apps on Google Play" 2020. [Online]. Available: https://www.appbrain.com/stats/number-ofAndroid-apps/. [Accessed: 21-01-2021].

[2] Wahanggara, V., & Prayudi, Y. Sistem Deteksi Malicious Software Berbasis System Call untuk Klasifikasi Barang Bukti Digital Menggunakan Metode Support Vector Machine. 2015. Yogyakarta: UII.

[3] Chorghe, S. P., & Shekokar, N. An Innovative Technique to Detect Malicious Applications in Android. 2013. IJSR, ISSN (Online): 2319-7064.

[4] Peiravian, N., & Zhu, X. Machine Learning for Android Malware Detection Using Permission and API Calls. 2013. USA: Florida Atlantic University.

[5] Yerima, S. Y., Seze, S., & McWilliams, G. Analysis of Bayesian Classification Based Approaches for Android Malware Detection. IET Information Security, olume 8, Issue 1. 2014.

[6] Karbab, E. B., Derhab, A., Debbabi, M., & Mouheb, D. Android Malware Detection Using Deep Learning ON API Method Sequences. ResearchGate. 2017.

[7] Pektas, A., & Acarman, T. Ensemble Machine Learning Approach for Android Malware Classification Using Hybrid Features.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Computer Engineering Department, Galatasaray University. 2017.

[8] Karbab, E. B., Debbabi, M., Derhab, A., & Mouheb, D. Android Malware Clustering using Community Detection on Android Packages Similarity Network. arXiv. 2020.

[9] A Hamid, I. R., Khalid, N. S., Abdullah, N. A., Ab Rahman, N. H., & Wen, C. C. Android Malware Classification Using K-Means Clustering. International Research and Innovation Summit (IRIS 2017).

[10] Li, Y., Jang, J., Hu, X., & Ou, X. Android Malware Clustering through Malicious Payload Mining. 2017.

[11] Dong, Y. Android Malware Prediction by Permission Analysis and DataMining. Athesis. 2017.

[12] Venkatesh, G. K., & Anitha, R. Structural analysis and detection of Android botnets using machine learning techniques. Int. J. Inf. Secure. 2018.

[13] Unucheck, R. All about Android app permissions. https://www.kaspersky.com/blog/Android-permissions-guide/14014/, p. 2017. [Accessed: 05-05-2020].

[14] Djatna, T. Pembandingan Stabilitas Algoritma Seleksi Fitur menggunakan. 2008.

[15] Hermawati, F. A. Data Mining. 2013.

[16] Hurier, M., Suarez-Tangil, G., Dash, S. K., Bissyande, T. F., Le Traon, Y., Klein, J., &amp; Cavallaro, L. Euphony: Harmonious Unification of Cacophonous Anti-Virus Vendor Labels for Android Malware. 2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR). 2017. https://doi.org/10.1109/msr.2017.57.