Detection System for Detecting Worms using Hybrid Algorithm of Naïve Bayesian classifier and K-Means

Osama Mohammed Qasim
Al-Karkh University of Science, Baghdad, Iraq
Baghdad, Iraq
usamamoh2006@yahoo.com

Karim Hashim Al-Saedi
Al-Mustansirya University Collage
Baghdad, Iraq
karimnav6@gmail.com

Abstract—According to a huge rampage of malware especially worms that network attackers use to invade the network. Detection systems need to be designed to stop these worm attacks. In this paper, we proposed a detection system that detects worms depending on a dataset of Kaspersky and McAfee companies' websites: (https://www.kaspersky.com/resource-center/threats/resource-for-virus-threats-definitions, https://home.mcafee.com/virusinfo), the dataset has been programmed by gathering information from these two websites. The proposed system has been used in host-based environment and it consist of two hybrid algorithms, which are the Naïve Bayesian classification and K-Means clustering. We applied this proposed algorithm on the dataset that involves the worms and as a result, we gained 88% accuracy, 81% of Detection Rate and 21% of False Alerts.

Keywords: *Worms, Detection Systems, Network Security, Naïve Bayesian classification, K-Means clustering.*

I. *INTRODUCTION*

Network Security is one of the Computer Science sections that refer to the protection of the data stored in computers connected through network. Nowadays networks have been developed and became popular in our world, the attackers of these networks are increasing every day, and the threats of those attackers have been evolved with it [1]. Network Security is one of the most important situations for foundations such as universities, these foundations provides a very important procedures on processes and world security [2].

Network services turned out progressively popular for the users in the last decades. The users are capable to communicate in business, information casting and participate knowledge. To decrease expanses, these services are cooperative by Information Technology (IT) associations and Internet Service Providers (ISPs) [3].

This may put the network at risk and cause malware. Malware is a software could be installed in the computers, smartphones and in any device connected to a network, and it damages these devices by making access to personal data and important information of these electronic devices [4].

In electronic devices such as computers and smartphones there are big amount of malicious software (malware) types that spreads through the network and it is multiplying rapidly, malware is increasing about 400K/day to 1M/day [5]. Kaspersky Labs found unprecedented types of mobile malware in 2015, and it was 884,774 types. This means three times more than they discovered in 2014, which it were 295,539 types [6].

Malware considered as the most dangerous attacks for the network, which threatens its confidentiality and the safety of the data. Great forms of malware increased through the network, lurking in packages. Everyday new forms of malware are discovered, so there must be a method to prevent these threats [7]. The malware makers have many selections to make decision about safeguarding their code from the anti-malware programs [8].

One of the most dangerous malware threats that the world is facing is Worm, it can spread inside the network by replicating itself [9]. Worm can defect computers especially individuals PCs and facilities. Despite the ability of the detection techniques, still its comprehensive effect is difficultly determined. Also its unknown how many devices connected to a network will end up with this kind of threat [10].

This gives rise to detection techniques. Detection techniques are the most important safeguard for the network and have major benefits for network security. Presently the malware deterrence is the antivirus programs, they are the first defense line that rise against malware. Mostly the functions used to detect a threat is Behavioral-based tool, Signature-based and Heuristic methods [11].

The internet has been spread through the world continuously, which referred to the need for more of data searching, entertaining, financing, information exchange, learning, commercial activities …etc. This was considered as a serious condition for network users, which make them venerable to many attacks. Han-Wei Hsiao, et al. proposed a detection system technique for malicious website of these kind of attacks. He used a spatial-temporal method of aggregation variables to make and made a detection module from NetFlow data. The results presented good actions for detecting these websites attacks [12].

In 2013 G. Ganesh Sundarkumar, et al. built a method of static analysis for detecting malware depending on API call

series by applying data mining and text in tandem. They applied text mining to get features from dataset involving a sequence of API calls. The joint information would be called to select a feature. Then they used the over sampling for balancing the detest. They used different data mining techniques such as Support Vector Machine, One Class Support Vector Machine, Decision Tree, Probabilistic Neural Network, and Group Method for Data Handling and Multi-Layer Perception. After balancing dataset and using Support Vector Machine in addition the One Class Support Vector Machine fulfilled sensitivity of 100 percent [13].

Ali Feizollah, et al. in 2014 applied the k-means and the mini batch k-means clustering algorithms they used in detecting malware for Android. The Android applications made the network traffic, for detection malware, they analyzed threat and benign data. They chose 800 samples out of 1260 samples of Android malware. The dataset have been made from MalGenome data sample. They gathered huge normal applications from official market of Android. According to the results mini batch k-means showed better performance than the other algorithm in detecting Android malware [14].

A huge set of consumers made by Luiza Sayfullina, et al. they worked mobile malware detection because the report of modern F-Secure is a 97% for Android platform of mobile malware threat that, has been used. To protect users from downloaded applications that contain the malware threat they applied the Naïve Bayes classification algorithm to classify and detect the malware especially the new kinds of malware by extracting the package of Android application (APK). They used a huge APKs dataset that is gained from F-Secure, they achieved by many tests 0.1% rate of false positive with 91% overall accuracy [15].

James B. Fraley, et al. they detected the polymorphic malware threats by using the algorithms of data mining and feature extraction techniques in 2016. The results of their study was 0.0030 low false positive ratios, and high true positive was 0.9978 for unknown files with size about (4k) [16].

II. *PROPOSED SYSTEM MODEL*

In this paper, the authors designed a system that detects malware worm type. The proposed system includes three models two of them are the main models and one sub-model. The main models include the hybrid of Naïve Bayesian classification and K-Means clustering and the second one include the Naïve Bayesian classification as comparison algorithm with proposed one.

First, the system will read the data by using the two models and then the results of both of them will be calculated in the third model, which it will present the final statistical results for both models. All of mentioned will be explained as the figure1 below which include the three models that have been used in this paper depending on the dataset that has been taken from Kaspersky and McAfee websites.

Finally, after applying the two models the output results will be taken to the statistical sub-model and calculate the Accuracy, Detection Rate and False Alerts and then presenting the statistical results of the two models performance. This sub-model also calculate the elapsed time for the models and show each model how long time they took to perform the detection and presenting the results.

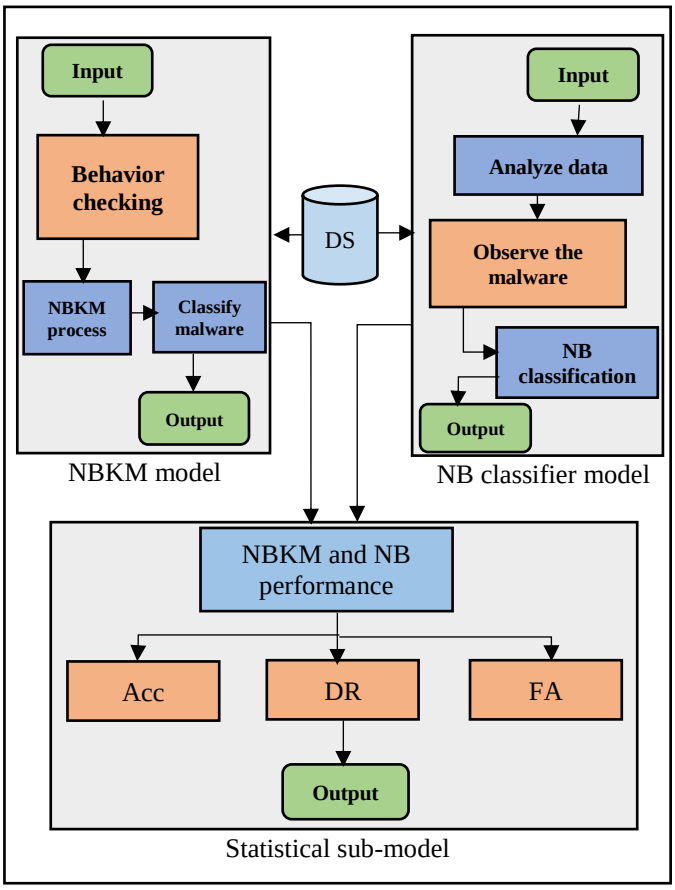

Fig 1 Proposed System Model

4. *NBKM MODEL*

In this field, we will speak about the proposed NBKM model. It is a hybrid between Naïve Bayesian classification and K-Means clustering, they both Data mining and Machine learning techniques. The hybrid model will use the benefits of the two algorithms the Naïve Bayesian classification and the K-Means clustering, the data detection could be achieved by two phases classification and clustering.

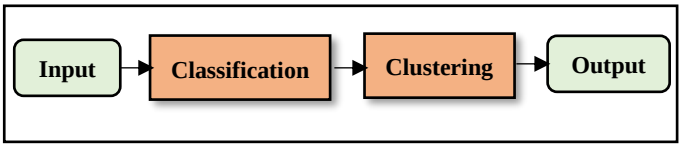

Fig 2 NBKM Phases

The algorithm will classify the data into 1's and 0's, the 1 is for the threat data and 0 is for benign data, below is the NBKM model algorithm pseudo steps and how it works:

Algorithm 1 NBKM model

Input: dataset include malware behavior.
Output: detecting types of malware.

1. TCP connection (IP = 127.0.0.1 , port)
2. Initialization i = 0, j = 0, k = 1, d = 1;
3. for m = 0 to 21 do
4. if (behavior == true)
5. then behavior = 1, i++ , k++;
6. else behavior = 0, j++ , d++;
7. end if;
8. Apply probability equation:
9. PY = i / k;
10. PN = j / d;
11. Choose centroid between PY and PN values.
12. Apply Euclidean equation for distances:
13. DISPY = sqrt ($\sum$ (PY – m)^2);
14. DISPN = sqrt ($\sum$ (PN – m)^2);
15. Round the distances by two decimal values:
16. Round (DISPY, 2);
17. Round (DISPN, 2);
18. if (DISPY >= DISPN)
19. then print "threat";
20. else print "benign";
21. end if;
22. end for;
23. Choose the nearest values to centroids and make it as the new values.
24. Repeat steps 3 to 21 until centroid values not exceed the number of iterations.

The algorithm will check the ports that the worm would make as a gateway to access the network with the IP address (127.0.0.1). The algorithm will check the behavior of the worm by analyzing the dataset from the number of data in the file that is 21 which it (m) in the algorithm above, if it the behavior was a threat then it is true and it will be dealt with the threat and if it was a benign then it will pass because it is not threat which make false, so the true value is for the threat behavior is for the false is for the benign. Then it will have the values that is gained from analyzing the dataset and compute the probability for each 1's and 0's, the values will be incremented each time the condition is fulfilled like the X malware uses port Y as a gateway to gain access to the computer and do its malicious work, by using if-then condition the values will increment by one. Two equations used for this purpose (py) and (pn), py for 1's and pn for 0's, the py will compute the probability of 1's and find how many 1's are found the same for pn but it works with 0's. For the number of 1's and 0's and the items used four counters used to calculate them, the counters are (i), (k), (j) and (d), i will count the number of 1's and j will be the number of 0's found; as for k and d they both will be the number of items found according for the behavior of malware.

By using the mathematical equation of probability for the Naïve Bayesian classifier the probability of the threats and benign as mentioned before 1 for threat and 0 for benign files could be calculated, this is the general form of the used equation:

$$P(1) = \text{number of 1's} / \text{number of items} \ldots E1$$
$$P(0) = \text{number of 0's} / \text{number of items} \ldots E2$$

Where P stand for Probability of (1) and (0), as mentioned before this equation calculate the probability ,which is number of 1's and 0's divided on the total number of items, the items are number of the behaviors that is discovered through the process.

After calculating the probability, the results of this process will be calculated by the Euclidean equation for distance, Euclidean is a method to calculate the distances, in this thesis it has been used to calculate the distance between the value of probability and the total number of dataset used for this work. By applying these mathematical methods, a detection will occur and the accuracy could be calculated. This is the general form for Euclidean equation:

$$Dis1 = SQRT\ (\sum (P(1) - \text{number of items})^2) \ldots E3$$
$$Dis2 = SQRT\ (\sum (P(0) - \text{number of items})^2) \ldots E4$$

Where the (Dis1, Dis2) are the distances between the probabilities and the used items number of dataset in this work. The result of this equation always will be positive values because of square and besides no distance have a negative value.

IV. *NB Classification Model*

In this field, we will talk about the NB classifier that we used to compare the results with the proposed model NBKM to measure the NBKM performance. In addition, as the previous model, it will classify the data as 1's and 0's, 1 for the threat data and 0 for the benign as well, and it will check the ports that the worm would gain access through the network with IP address (127.0.0.1). The pseudo code below for this model will be explained briefly:

Algorithm 2 NB classifier

Input: dataset including behavior of malware.
Output: detecting and classifying the malware.

1. TCP connection (IP = 127.0.0.1, port)
2. Initialization: i = 0, j = 0, k = 1, d = 1;
3. for n = 0 to 21 do
4. if (behavior == true)
5. then behavior = 1, i++, k++;
6. else behavior = 0, j++, d++;
7. end if;
8. Apply the probability equations:
9. P1 = i / k;
10. P2 = j / d;
11. Round the results of P1 and P2 equations by two decimal values:
12. Round (P1, 2);
13. Round (P2, 2);

14. if (P1 >= P2)
15. then print "threat";
16. else print "benign";
17. end if;
18. end for;
19. Repeat steps 3 to 17

The performance of this algorithm applied on the same dataset that was used to the first algorithm (NBKM) to make a comparison between results of both algorithms. It uses four counters i, j, k and d, the counters i and k are for the threat calculating and counters j and d for benign ones, it will be spoken briefly in the next paragraphs.

After analyzing dataset, the algorithm will work as the concept of (NBKM) algorithm; every time the behavior is checked whether it was a threat or benign it will give values of 1's and 0's, and as the same before 1 for threat and 0 for benign files.

When the behavior is equal to 1 then it is a threat the counters i and k will increment its values by one, when the opposite behavior equal to 0 counters j and d increment its values by one. After this process of the behavior of the malware, the probability will be calculated.

When all of above is set, the probability of both threat and benign values will be calculated by the probability equation to classify the data as a threat or benign, here is the equations of probability for both threat and benign data:

$$P1 = \sum (i / k) \ldots E4$$
$$P2 = \sum (j / d) \ldots E5$$

Where P1 is the probability of the threat, i is a counter of 1's and k is a counter for the number of items for threat data, P2 is the probability of the benign data; j is the number of 0's found, d is a counter, which represents the number of items for benign data.

After calculating the probabilities, the algorithm will classify the data as a threat or benign file and then it will have a full detection for malware according to the used dataset.

V. *STATISTICAL SUB-MODEL*

After applying, the two models and having the detection results the statistical sub-model will take these results and calculate the Accuracy, Detection Rate and False Alerts for both algorithms to make a compression between the two models performance. It has three mathematical equations that used for this purpose:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \ldots E6$$

$$\text{Detection Rate} = \frac{TP}{TP+FP} \ldots E7$$

$$\text{False Alert} = \frac{FP}{FP+TN} \ldots E8$$

All of these symbols of these mathematical equations are explained in the table below:

Table 1 statistical model

Actual	Predicted benign	Predicted threat
benign	TN	FP
threat	FN	TP

4. *RESULTS DISCUSSION*

After applying the system models, the results will be presented as the tables shown below:

Table 2 NBKM results

id	DISPY	DISPN	Threat	Type
1	20.12	20.52	No	Mydoom
2	20.12	20.52	No	Mimal
3	20.12	20.52	No	Doomjuice
4	20.12	20.86	No	Sobig
5	20.12	11	Yes	ILOVEYOU
6	20.12	11	Yes	ExplorerZIP
7	20.12	11	Yes	Badtrans
8	20.12	11	Yes	Brontok
9	20.12	20.52	No	Welchia
10	20.12	20.66	No	Sasser
11	11	20.12	No	Bagle
12	11	20.12	No	Zotob
13	11	20.52	No	Blaster
14	11	20.12	No	SCA
15	11	20.12	No	Zero-Access botnet
16	20.12	20.12	Yes	Zlob
17	20.66	20.12	Yes	ILOVEYOU

In the table above are the results of worm detection by using the proposed model of the hybrid Naïve Bayesian and the K-Mean (NBKM) algorithms. The values in the table above are the distances between probabilities of threat and benign data. After discovering, seventeen types of worms, the results of the distances of the both benign and threat data were very high. After combining the two algorithms together, the detection became more accurate but it took more time to complete the detection through this process implementation. The table involve the index of each row, the values of the distances for benign and threat data, the worm detection probability and the type of the worm malware that have been discovered. The chart below will show the result of file detection, the file that has been searched have fifty percent worm and the other fifty percent is benign:

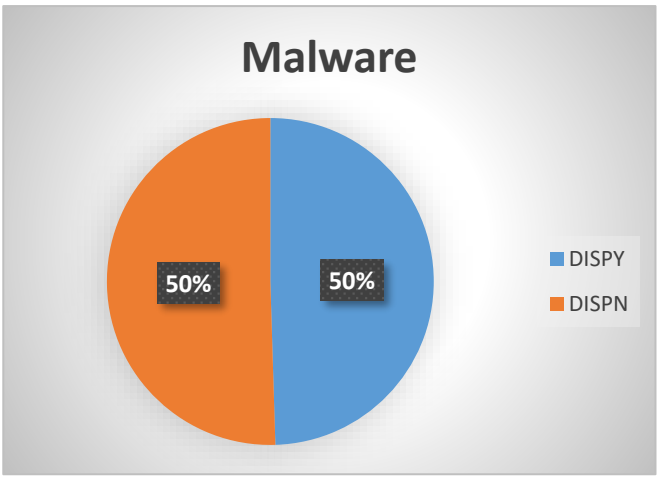

Chart 1 NBKM file detection rate

As for the NB classifier results table:

Table 3 NB classifier results

id	P1	P2	Threat	Type
1	0.96	0.98	No	Mydoom
2	0.96	0.98	No	Mimal
3	0.96	0.98	No	Doomjuice
4	0.96	1	No	Sobig
5	0.96	0	Yes	ILOVEYOU
6	0.96	0	Yes	ExplorerZIP
7	0.96	0	Yes	Badtrans
8	0.96	0	Yes	Brontok
9	0.96	0.98	No	Welchia
10	0.96	0.99	No	Sasser
11	0	0.96	No	Bagle
12	0	0.96	No	Zotob
13	0	0.98	No	Blaster
14	0	0.96	No	SCA
15	0	0.96	No	Zero-Access botnet
16	0.96	0.96	Yes	Zlob
17	0.99	0.96	Yes	ILOVEYOU

The NB classifier model detected the same worms as the NBKM model but the results are less accurate with less time to implement than the NBKM model. The table has the probability equations values that shows the probability results of the worm detection.

This is the chart of the probabilities of NB classifier model for detecting the worm threat for the same file as NBKM algorithm which have 51% of threat detection and 49% for benign detection:

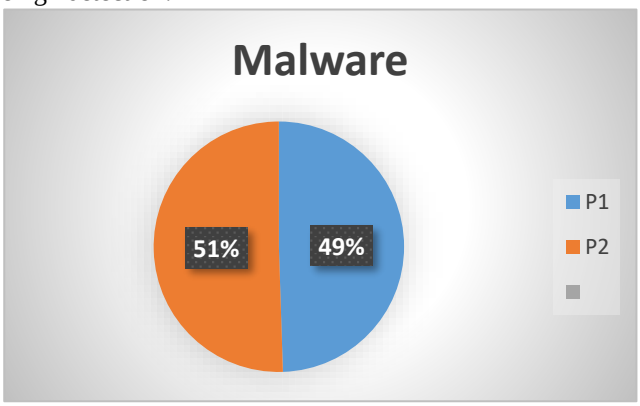

Chart 2 NB classifier file detection

After applying, the models and gaining the results of each model statistical results will be presented to show the comparison between the proposed hybrid model NBKM and the other model the NB classifier to compare the performance between them, as it shown in the table below:

Table 4 statistical model

Algorithm	Accuracy	DR	FA	Time

NBKM	0.88	0.82	0.21	00:01:11
NB	0.81	0.82	0.21	00:01:10

As in the table above, the NBKM achieved 88% of accuracy while the NB classifier achieved 81% of accuracy. Both of the models achieved an equal ratio in detection rate which it 82%, the same goes for the false alerts, which it were 21% of the data in both models performance. As for time the NBKM model took 00:01:11 to detect, while the NB classifier model took less time which it 00:01:10. The chart below shows the statistical results of both NBKM and NB algorithms:

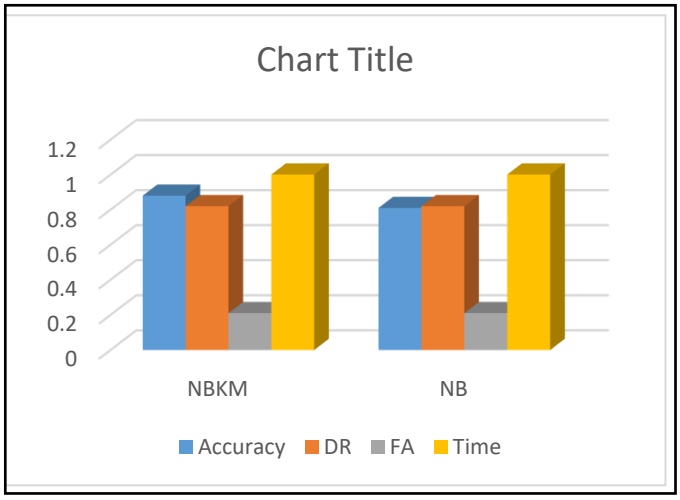

Chart 3 statistical model

VII. *CONCLUSIONS*

In this paper, a proposed system model has been designed, hybrid model has been designed between Naïve Bayesian classifier and K-Means clustering. We made a hybrid between Naïve Bayesian and K-Means because the Naïve Bayesian is easy to build and efficient in detection the malware and as for the K-Means it was used to enhance and raise the accuracy of detection. The hybrid model showed a great performance in detecting worms, while the Naïve Bayesian classifier model showed less performance in worm detection. The statistical of both models NBKM and NB classifier has been resulted as follows: the accuracy has been increased by 7% in NBKM model more than the accuracy in the NB classifier model. The detection rate for both models was 81% and the false alerts were 21%. The time in NBKM model was higher comparing with the NB classifier model because of the hybrid of two algorithms NB classifier and K-Means clustering by combining the benefits of both algorithms especially their equations the probability and the distances equations, which needed more coding for these algorithms that makes the debugging takes more time to debug.

VIII. *REFERENCES*

[1] A. . Fallis, "Neural Network Model," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[2] M. S. Kacar and K. Oztoprak, "Network Security Scoring," *2017 IEEE 11th Int. Conf. Semant. Comput.*, pp. 477–481, 2017.

[3] H. Li, P. W. C. Prasad, A. Alsadoon, L. Pham, and A. Elchouemi, "An improvement of Backbone Network security using DMVPN over an EZVPN structure," pp. 14–16, 2016.

[4] A. Juan *et al.*, "Native Malware Detection in Smartphones with Android OS Using Static Analysis , Feature Selection and Ensemble Classifiers," pp. 67–74, 2016.

[5] J. R. Upchurch, "Malware Provenance : Detecting Code Reuse in Malicious Software," pp. 101–109, 2016.

[6] M. Ping, B. Alsulami, and S. Mancoridis, "On the Effectiveness of Application Characteristics in the Automatic Classification of Malware on Smartphones," pp. 75–82, 2016.

[7] E. Bocchi *et al.*, "MAGMA network behavior classifier for malware traffic," *Comput. Networks*, vol. 0, pp. 1–15, 2016.

[8] L. Jones, A. Sellers, and M. Carlisle, "CARDINAL : Similarity Analysis to Defeat Malware Compiler Variations," 2015.

[9] M. A. Ahmad, S. Woodhead, and D. Gan, "Early Containment of Fast Network Worm Malware," pp. 195–201, 2016.

[10] L. Xue and Z. Hu, "Research of Worm Intrusion Detection Algorithm Based on Statistical Classification Technology," *2015 8th Int. Symp. Comput. Intell. Des.*, pp. 413–416, 2015.

[11] M. Martens, H. Asghari, M. van Eeten, and P. Van Mieghem, "A time-dependent SIS-model for long-term computer worm evolution," *2016 IEEE Conf. Commun. Netw. Secur.*, pp. 207–215, 2016.

[12] D. Chen, "Detecting Hiding Malicious Website Using Network Traffic Mining Approach," 2010.

[13] G. G. Sundarkumar and V. Ravi, "Malware Detection by Text and Data Mining," vol. 500057, pp. 0–5, 2013.

[14] A. Feizollah, N. B. Anuar, R. Salleh, and F. Amalina, "Comparative Study of K-means and Mini Batch K-means Clustering Algorithms in Android Malware Detection Using Network Traffic Analysis," no. February, pp. 1–5, 2014.

[15] L. Sayfullina, E. Eirola, D. Komashinsky, P. Palumbo, and Y. Miche, "Efficient detection of zero-day Android Malware using Normalized Bernoulli Naive Bayes," pp. 1–8, 2015.

[16] J. B. Fraley and M. Figueroa, "Polymorphic malware detection using topological feature extraction with data mining," *SoutheastCon 2016*, pp. 1–7, 2016.