iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 Issue: VI Month of publication: June 2021

DOI: https://doi.org/10.22214/ijraset.2021.36191

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI June 2021- Available at www.ijraset.com

Behavioral Host-Based Intrusion Detection and Prevention System for Android

Yashavant Darange[1], Sanket Patekar[2], Jitesh Narode[3], Ayur Shete[4]
[1, 2]B.E. Student, Dept. of Information Technology Sanjivani College of EngineeringKopargaon, India

Abstract: Intrusion Detection System (IDS) is vital to protect smartphones from about to happen security breach and make sure user privacy. Android is the most popular mobile Operating System (OS), holding many markets share. Android malware detection has received important concentration, existing solutionstypically rely on performing resource intensive analysis on a server, assuming an uninterrupted link between the device and the server. In this paper, we propose a behavior Host-based IDS (HIDS) by using permissions incorporating arithmetical and ML algorithms. The benefit of our proposed IDS is two folds. First, it is completely independent and runs on the smartphone device, without need any link to a server. Second, it requires only training dataset consisting of some of examples from both benignand malicious datasets for tuning. though, in put into practice, collecting malicious examples is exciting since its important infecting the device and collecting many of samples in orderto characterize the malware's behavior and the labelling has to be done. The evaluation outcome show that the proposed IDS gives a very hopeful accuracy.
Index Terms: Android, Security and privacy, Intrusion detec- tion and prevention system(IDPS),Malware detection, Behavior analysis, Machine learning.

I. INTRODUCTION

The model System proposed in this paper focuses on variousdata machine learning techniques that are used in intrusion detection prediction system. Now a days smartphone is a most important part of a life. It regulates work throughout our day. Any permission difference in smartphone can cause intrusion in other applications of smartphone. Smartphones play essential role in modern life. They provide a broad range of attractive features enabling mobile users to accessan excess of high-quality personalized services , which makes them attractive for cybercriminals. Android is the most popular mobile operating system, capturing approximately the majorityof global market share, which renders it a major target for attackers. In particular, its open operating system characteristicallows the user to install applications from not only trusted,but also untrusted sources (i.e. third-party markets). therefore, malwares looking like an innocent software (e.g., games, utilities, etc.) might be downloaded and installed, which can pose serious security threats. Smartphone malwares also allowattackers to use the stored personal data on the device or to launch attacks. This paper presents techniques to analysis of Random Forest for predicting intrusion at an early stage [3]. Earlier research efforts on designing an IDPS for Androidmostly rely on rooting the device or collecting data from re- mote devices and processing them in a command and organize hub inside the cloud However, these approaches have some severe limitations: i) they require a continuous link betweena mobile device and a central IDPS server, that might notbe always possible due to the network's problems or partial coverage; and ii) they increase the risk of personal informationleakage, which may lead to the violation of user's privacy.

II. LITERATURE SURVEY

Malware detection methods are divided into three major categories: 1) static, 2) dynamic, and 3) hybrid [2]. The static techniques (also known as misuse- or signature-based) maintain an updated database of malicious code patterns (i.e. attack signatures) and scan the code, with no running it, for those signatures. Behavior-based IDPSs basically build a data- driven model for the benign behavior. The data can usually contain access permissions requested by an application, e.g.to read/send SMS, accessing the camera, microphone, contact list, device's location, etc. [3], [4]. Based on the location wherethe detection algorithm is deployed, IDPSs are more divided into three main categories [5]:

1) *Host-based:* The complete system, including the detectionengine, is deployed on the smartphone device itself, an IDPS [6], [7], [8].
2) *Centralized:* An authority and manage center in the cloud monitors the smartphone devices. In-depth analyses are performed on powerful servers, taking benefit of their plentifulcalculation power and memory capability [9], [10], [11], [12], [13], [14].
3) *Distributed:* The system is partially deployed on the smartphone device and partly within the cloud. The data collection agent and some lightweight analyses are performed on the device, whereas computationally expensive analyses arecarried out on a remote server computer [15], [16].

David Dagon et al. cautioned the local area in 2004 foresee-ing the attainability of malware in cell phones. Indeed on the off chance that wi-fi and bluetooth were considered as the mostlikely disease ways, the development of cell phone deals with nonstop Internet availability made the expectation come valid. Solidly, in June of that very year, the first malware specificallycomposed for Symbian OS stage was found. After the disease achievement did by Cabir malware what's more, its variations,analysts proposed approaches what's more, created diffierent instruments to distinguish malware in cell phones.

Work	Advantage	Disadvantage
Andromaly	Behavior-based IDS that analyzesresource utilization of the mobile device	Requires labeled data and only has been tested against malicious data collected froma few artificial malware's
Aurasium	Enforces arbitary runtime securitypolicies	Uses repacking-modifies the original application, can be treated as a malware by other IDSs
Crowdroid	Uses crowdsourcing to acquire data from different sources	Needs root access and analyzes one application at a time
Drozer	IPC to monitor in-stalled apps easy to implement new models	Uses a command line interface thatis not user friendly
Kirin	Verifies the apps permissions against a set of predefined rulesand provides amethodology forupgrading security requirements	Analyzes the ap-plication in install time and is not designed for mon- itoring the appli-cation behavior in runtime

III. HOST-BASED IDS

The proposed HIDS utilizes ML or measurable inconsistency recognition calculations to recognize dubious conduct on Android cell phones, dissecting the framework log records and figuring the likelihood of interruption. We recognize highlights that adequately describe the effect ofversatile malwares furthermore, boost the adequacy of the fundamental calculation for identifying dubious exercises. These highlights are checked continuously by the created HIDS to gather the information and feed it to the recognition calculation, for examination.

The engineering of the proposed HIDS is made out of the following segments, represented by Fig. 1: A. Gathering data B. Data pre-processing C. Researching the model that will be best for the type of data D. Training and testing the model E. Evaluation

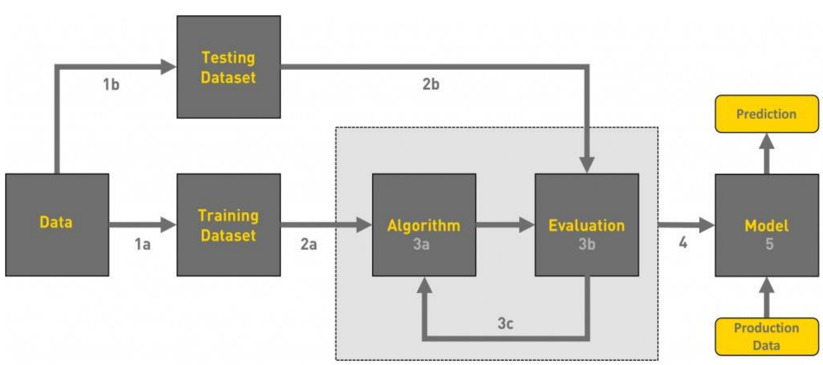


Fig. 1. System Architecture.

A. Data Gathering

The Real-Time Data Gathering component is responsible for collecting the data in real-time.The Real-Time Data Gathering component is responsible for collecting the data in real- time.The process of gathering data depends on the type of project we desire to make, if we want to make an ML project that uses real-time data. The data set can be collected from various sources such as a file, database and many other such sources but the collected data cannot be used directly forperforming the analysis process as there might be a lot of missing data, extremely large values, unorganized text data or noisy data. Therefore, to unravel this problem Data Preparationis completed .

B. Dataset Pre-processing

Information pre-handling is quite possibly the main strides in AI. It is the main advance that aides in building AI models all the more precisely. In AI, there is a 80/20 standard. Each information researcher ought to invest 80 percent energy for information pre-preparing and 20 percent chance to really playout the examination.

Information pre-preparing is a cycle of cleaning the crude information for example the information is gathered in reality and is changed over to a perfect informational collection. As such, at whatever point the information is assembled from various sources it is gathered in a crude configuration and this information isn't possible for the investigation.

Subsequently, certain means are executed to change overthe information into a little perfect informational index, this piece of the interaction is called as information pre-handling.

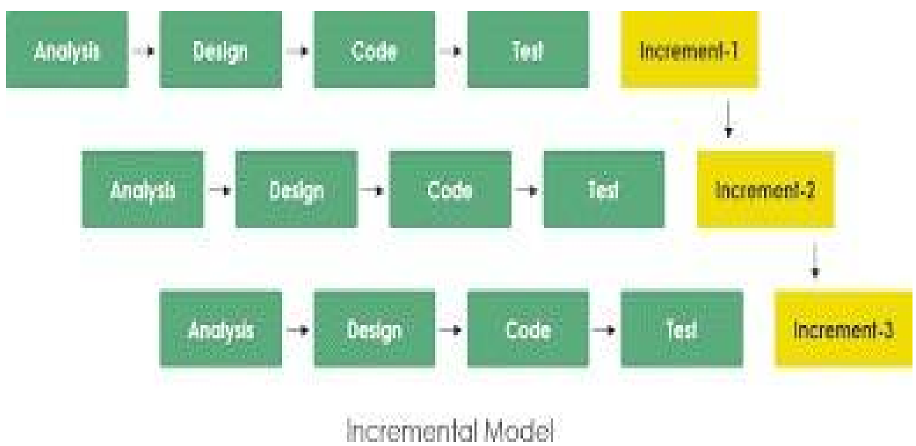


Fig. 2. Incremental Model.

C. Researching The Model That Will Be Best For The Type Of Data

The researching the model that will be best for the type of data utilizes either a ML or a measurable calculation to order every section of the standardized dataset. For every section in the dataset, the calculation yields either zero (benign) or one (malicious). Consequently, the yield of Recognition Algorithmis a double vector, the length of which, is equivalent to the quantity of sections in the standardized dataset. This vectoris then taken care of to the Intrusion Probability Assessment module.

D. Training

The Training module is liable for building profiles for benevolent and pernicious practices. It very well may be performed either disconnected, to save versatile assets, or on the web, by the cell phone, every now and then, to update thoseprofiles since the thought for ordinary conduct can change aftersome time on the grounds that the way the gadget is being utilized may not really consistently steady.

E. Evaluation

Model Evaluation is a fundamental piece of the model improvement measure. It assists with tracking down the best model that addresses our information and how well the pickedmodel will function later on.

To further develop the model we may tune the hyper- boundaries of the model and attempt to work on the exactness and further more looking at the disarray lattice to attempt to build the quantity of genuine positives and genuine negatives.

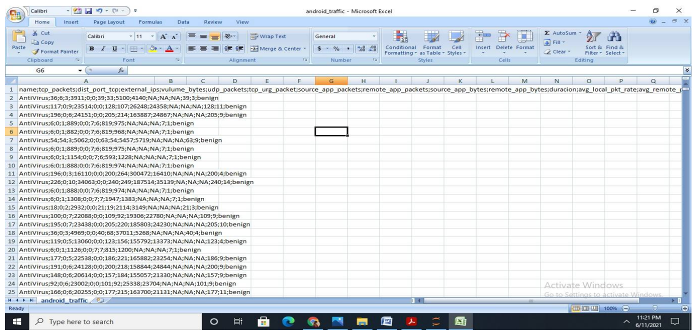
International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI June 2021- Available at www.ijraset.com

IV. ALGORITHM

The proposed IDS makes use of following Machine Learn- ing and statistical algorithms so as to classify a run-time behaviour as benign or malicious.

A. Dataset

We first of all collect benign dataset that contain 7846 samples The samples collected during one data acquisition interval were saved in a CSV-file, every row representing one example and each column representing one feature.



	A
1	name;tcp_packets;dist_port_tcp;external_ips;vulume_bytes;udp_packets;tcp_urg_packet;source_app_packets;remote_app_packets;source_app_bytes;remote_app_bytes;duracion;avg_local_pkt_rate;avg_remote_p
2	AntiVirus;36;6;3;3911;0;0;39;33;5100;4140;NA;NA;NA;39;3;benign
3	AntiVirus;117;0;9;23514;0;0;128;107;26248;24358;NA;NA;NA;128;11;benign
4	AntiVirus;196;0;6;24151;0;0;205;214;163887;24867;NA;NA;NA;205;9;benign
5	AntiVirus;6;0;1;889;0;0;7;6;819;975;NA;NA;NA;7;1;benign
6	AntiVirus;6;0;1;882;0;0;7;6;819;968;NA;NA;NA;7;1;benign
7	AntiVirus;54;54;3;5062;0;0;63;54;5457;5719;NA;NA;NA;63;9;benign
8	AntiVirus;6;0;1;889;0;0;7;6;819;975;NA;NA;NA;7;1;benign
9	AntiVirus;6;0;1;1154;0;0;7;6;593;1228;NA;NA;NA;7;1;benign
10	AntiVirus;6;0;1;888;0;0;7;6;819;974;NA;NA;NA;7;1;benign
11	AntiVirus;196;0;3;16110;0;0;200;264;300472;16410;NA;NA;NA;200;4;benign
12	AntiVirus;226;0;10;34063;0;0;240;249;187514;35139;NA;NA;NA;240;14;benign
13	AntiVirus;6;0;1;888;0;0;7;6;819;974;NA;NA;NA;7;1;benign
14	AntiVirus;6;0;1;1308;0;0;7;7;1947;1383;NA;NA;NA;7;1;benign
15	AntiVirus;18;0;2;2932;0;0;21;19;2114;3149;NA;NA;NA;21;3;benign
16	AntiVirus;100;0;7;22088;0;0;109;92;19306;22780;NA;NA;NA;109;9;benign
17	AntiVirus;195;0;7;23438;0;0;205;220;185803;24230;NA;NA;NA;205;10;benign
18	AntiVirus;36;0;3;4969;0;0;40;68;37011;5268;NA;NA;NA;40;4;benign
19	AntiVirus;119;0;5;13060;0;0;123;156;155792;13373;NA;NA;NA;123;4;benign
20	AntiVirus;6;0;1;1126;0;0;7;7;815;1200;NA;NA;NA;7;1;benign
21	AntiVirus;177;0;5;22538;0;0;186;221;165882;23254;NA;NA;NA;186;9;benign
22	AntiVirus;191;0;6;24128;0;0;200;218;158844;24844;NA;NA;NA;200;9;benign
23	AntiVirus;148;0;6;20614;0;0;157;184;155057;21330;NA;NA;NA;157;9;benign
24	AntiVirus;92;0;6;23002;0;0;101;92;25338;23704;NA;NA;NA;101;9;benign
25	AntiVirus;166;0;6;20255;0;0;177;215;163700;21131;NA;NA;NA;177;11;benign

Fig. 3. Dataset.

B. Naive Bayes

It's anything but a grouping method dependent on Bayes' hypothesis with a presumption of freedom between indicators. In basic terms, a Naive Bayes classifier accepts that the presence of a specific element in a class is disconnected tothe presence of some other element. For instance, a natural product might be viewed as an apple on the off chance thatit is red, round, and around 3 crawls in distance across. Regardless of whether these highlights rely upon one another or upon the presence of different highlights, an innocent Bayesclassifier would think about these properties to freely add to the likelihood that this organic product is an apple.

Naive Bayesian model is not difficult to assemble and espe-cially valuable for extremely enormous informational indexes.Alongside straightforwardness, Naive Bayes is known to beat even exceptionally refined grouping strategies.

Bayes hypothesis gives a method of figuring back likelihood P(c—x) from P(c), P(x) and P(x—c). Take a gander at the condition beneath:

Likelihood　　Class Prior Probability

$$P(c \mid x) = \frac{P(x \mid c)\,P(c)}{P(x)}$$

Posterior Probability　　Predictor Prior Probability

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Here, P(c—x) is the posterior probability of class (target)given predictor (attribute).

P(c) is the prior probability of class.

P(x—c) is the likelihood which is the probability of predictorgiven class.

P(x) is the prior probability of predictor.

C. kNN (k-Nearest Neighbours)

It very well may be utilized for both arrangement and relapse issues. In any case, it is all the more broadly utilizedin order issues in the business. K closest neighbors is a straightforward calculation that stores every accessible case and groups new cases by a larger part vote of its k neighbors. The case being appointed to the class is generally basic among its K closest neighbors estimated by a distance work.These distance capacities can be Euclidean, Manhattan, and Hamming distance. Initial two capacities are utilized for consistent capacity and third one (Hamming) for all out factors. In the event that K = 1, the case is basically doled out to the class of its closest neighbor. On occasion, pickingK ends up being a test while performing kNN demonstrating method. KNN can without much of a stretch be planned toour genuine lives model/. On the off chance that you needto find out about an individual, of whom you have no clue,you may jump at the chance to get some answers concerning his dear companions and the circles he moves in and access his/her information.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI June 2021- Available at www.ijraset.com

Things to consider before selecting kNN
1) KNN is computationally expensive
2) Variables should be normalized else higher range vari-ables can bias it
3) Works on pre-processing stage more before going forkNN like an outlier, noise removal

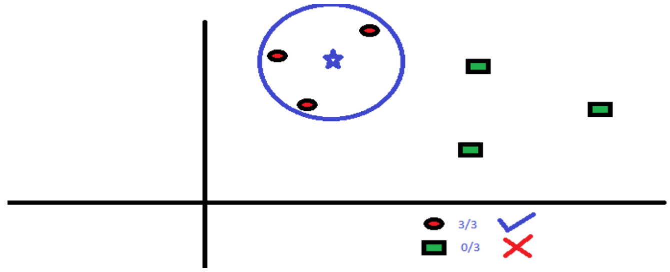


Fig. 4. kNN.

D. Random Forest

Random Forest is a brand name term for an outfit of choice trees. In Random Forest, we've assortment of choice trees (so it is known as "Woods"). To group another article dependent on ascribes, each tree gives an arrangement and we say the tree "votes" for that specific class. The woodspicks the arrangement having the greatest votes (over every one of the trees in the timberland).

Each tree is planted and developed as follows:
1) If the quantity of cases in the preparation set is N, then, at that point test of N cases is taken aimlessly yet with substitution. This example will be the preparation set for developing the tree.
2) If there are M info factors, a number m¡¡M is determinedwith the end goal that at every hub, m factors are chosen aimlessly out of the M and the best parted on these m is utilized to part the hub. The worth of m is held consistent during the backwoods developing.
3) Each tree is developing to the biggest degree conceivable.There is no chance of pruning.

A benign dataset, the all out number of right and wrong choices are dismissed by True Negatives (TNs) and FPs, individually. Conversely, for a malignant dataset, the all out number of right and wrong choices are dismissed by TPsand False Negatives (FNs), individually. We use Accuracy, review, F1 score,True Positive Rate (TPR), and False Positive Rate (FPR) as execution measurements. . Exactness is theproportion of right choices out of the complete number of choices that the Intrusion Detection framework takes.

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

Here's how to calculate Precision

$$Precision = \frac{TP}{TP + FP}$$

And here's how we can calculate Recall:

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

In practice, when we try to increase the precision of our model, the recall goes down, and vice-versa. The F1-score captures both the trends in a single value:

$$F1 - score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$$

In some cases in AI we are confronted with a multi-class order issues. Cohen's kappa measurement is a best measure that can deal with very well both multi-class and imbalanced class issues. Cohen's kappa is characterized as: where po isthe noticed arrangement, and pe is the normal understanding.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI June 2021- Available at www.ijraset.com

V. RESULT

In this section, we present our numerical results for bothML and statistical algorithms.

```
naive_bayes
0.8375
              precision    recall  f1-score   support

           0       0.91      0.76      0.83        41
           1       0.78      0.92      0.85        39

    accuracy                           0.84        80
   macro avg       0.85      0.84      0.84        80
weighted avg       0.85      0.84      0.84        80
```

Fig. 5. Naive Bayes Accuracy.

```
kneighbors 3
0.8875
              precision    recall  f1-score   support

           0       0.94      0.82      0.88        39
           1       0.85      0.95      0.90        41

    accuracy                           0.89        80
   macro avg       0.89      0.89      0.89        80
weighted avg       0.89      0.89      0.89        80
```

Fig. 6. kNN Accuracy.

```
kneighbors 6
0.85
              precision    recall  f1-score   support

           0       0.94      0.76      0.84        42
           1       0.78      0.95      0.86        38

    accuracy                           0.85        80
   macro avg       0.86      0.85      0.85        80
weighted avg       0.87      0.85      0.85        80


kneighbors 9
0.8625
              precision    recall  f1-score   support

           0       0.94      0.78      0.85        41
           1       0.80      0.95      0.87        39

    accuracy                           0.86        80
   macro avg       0.87      0.86      0.86        80
weighted avg       0.87      0.86      0.86        80


kneighbors 12
0.85
              precision    recall  f1-score   support

           0       0.94      0.76      0.84        42
           1       0.78      0.95      0.86        38

    accuracy                           0.85        80
   macro avg       0.86      0.85      0.85        80
weighted avg       0.87      0.85      0.85        80
```

Fig. 7. kNN Accuracy.

```
RandomForestClassifier(max_depth=50, n_estimators=250, random_state=45)
0.9172625127681308
              precision    recall  f1-score   support

      benign       0.93      0.94      0.93      1190
   malicious       0.90      0.88      0.89       768

    accuracy                           0.92      1958
   macro avg       0.91      0.91      0.91      1958
weighted avg       0.92      0.92      0.92      1958

cohen kappa score
0.8258206083396299
[[1117   73]
 [  89  679]]
```

Fig. 8. Random Forest Accuracy.

VI. CONCLUSION

We have developed a system for classifying Android ap- plications as malicious or benign applications using machine- learning techniques and algorithms. To generate the models, we have used android traffic datasets. This application gives high accuracy rate for different machine learning algorithms.

REFERENCES

[1] JOSÉ RIBEIRO, FIROOZ B. SAGHEZCHI ,GEORGIOS MANTAS, JONATHAN RODRIGUEZ ,AND RAED A. ABD-ALHAMEED , "HIDROID: Prototyping a Behavioral Host-Based Intrusion Detection and Prevention System for Android," IEEE Access, Vol. 8, pp. 23154- 23168, doi:10.1109/ACCESS.2020.2969626.

[2] A. Damodaran, F. D. Troia, C. A. Visaggio, T. H. Austin, and M.Stamp,"A comparison of static, dynamic, and hybrid analysis for mal- ware detection,". Comput. Virol. Hacking Techn., vol. 13, no. 1, pp. 112,Feb. 2017,doi: 10.1007/s11416-015-0261-z.

[3] P. Faruki, A. Bharmal, V. Laxmi, V. Ganmoor, M. S. Gaur, M. Conti, and M. Rajarajan, "Android security: A survey of issues, malware penetration, and defenses," IEEE Commun. Surveys Tuts., vol. 17, no.2, pp. 9981022,2nd Quart., 2015, doi: 10.1109/comst.2014.2386139.

[4] N. Peiravian and X. Zhu, "Machine learning for Android malware detection using permission and API calls," in Proc. IEEE 25th Int.Conf.Tools with Artif. Intell., Washington, DC, USA, Nov. 2013, doi: 10.1109/ictai.2013.53.

[5] G. Suarez-Tangil, J. E. Tapiador, P. Peris-Lopez, and A. Ribagorda, "Evolution, detection and analysis of malware for smart devices," IEEE Commun. Surveys Tuts., vol. 16, no. 2, pp. 961987, 2nd Quart., 2014, doi: 10.1109/surv.2013.101613.00077.

[6] J. Ribeiro, F. B. Saghezchi, G. Mantas, J. Rodriguez, S. J. Shep- herd, and R. A. Abd-Alhameed, "An autonomous host-based intrusion detection system for Android mobile devices," in Mobile Networksand Applications MONET. New York, NY, USA: Springer, 2019, doi: 10.1007/s11036-019-01220-y

[7] A. Shabtai, U. Kanonov,Y. Elovici, C. Glezer, andY.Weiss, "'Andro- maly': A behavioral malware detection framework for android devices," J. Intell. Inf. Syst., vol. 38, no. 1, pp. 161190, Feb. 2012, doi: 10.1007/s10844-010-0148-x.

[8] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Surveyof intrusion detection systems: Techniques, datasets and challenges," Cybersecurity, vol. 2, no. 1, p. 20, Dec. 2019, doi: 10.1186/s42400-019-0038-7.

[9] I. Burguera, U. Zurutuza, and S. Nadjm-Tehrani, "Crowdroid: Behaviorbased malware detection system for Android," in Proc. ACM Conf.Comput. Commun. Secur., 2011, pp. 1526, doi:10.1145/2046614.2046619.

[10] Y. Mehmood, M. A. Shibli, U. Habiba, and R. Masood, "Intrusiondetection system in cloud computing: Challenges and opportunities,"in Proc. 2nd Nat. Conf. Inf. Assurance (NCIA), Dec. 2013,pp. 5966.

[11] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya, and R. Ranjan, "A hybrid deep learning-based model for anomaly detection in cloud datacenter networks," IEEE Trans. Netw. Service Man- age., vol. 16,no. 3, pp. 924935, Sep. 2019, doi: 10.1109/tnsm.2019. 2927886.

[12] S. Garg, K. Kaur, N.Kumar, S. Batra, and M. S. Obaidat, "HyClass: Hybrid classication model for anomaly detection in cloud environment,"in Proc. IEEE Int. Conf. Commun. (ICC), Kansas City, MO, USA, May 2018,pp. 17, doi: 10.1109/icc.2018.8422481.

[13] S. Garg, K. Kaur, S. Batra, G. Kaddoum, N. Kumar, and A. Bouk- erche,"A multi-stage anomaly detection scheme for augmenting the security in IoT-enabled applications," Future Gener. Comput. Syst., vol.104, pp. 105118, Mar. 2020, doi: 10.1016/j.future.2019.09.038.

[14] S. Garg, K. Kaur, S. Batra, G. S. Aujla, G. Morgan, N. Ku-mar, A. Y. Zomaya, and R. Ranjan, "En-ABC: An ensemble arti- cial bee colony based anomaly detection scheme for cloud environ- ment," J. Parallel Distrib. Comput., vol. 135, pp. 219233, Jan. 2020, doi:10.1016/j.jpdc.2019.09.013.

[15] A.-D. Schmidt, R. Bye, H.-G. Schmidt, J. Clausen, O. Kiraz, K. A. Yuksel,S. A. Camtepe, and S. Albayrak, "Static analysis of executables for collaborative malware detection on Android," in Proc. IEEE Int. Conf. Commun.,Dresden, Germany, Jun. 2009, p. 631.

[16] G. Dini, F. Martinelli, A. Saracino, and D. Sgandurra, MADAM: A Multi- Level Anomaly Detector for Android Malware (Lecture Notes in Com- puter Science: Lecture Notes in Artificial Intelligence: Lecture Notes in Bioinformatics),vol. 7531. Berlin, Germany: Springer, 2012, pp. 240253, doi:10.1007/978-3-642-33704-8$_2$1.