ISSN 2278-3091
Volume 10, No.2, March - April 2021
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse151022021.pdf
https://doi.org/10.30534/ijatcse/2021/151022021

A Survey Paper on Malware Detection Techniques

Nahush Shetty[1], **Raja Praveen**[2]
[1]Jain University, India, nahushshett01@gmail.com
[2]Jain University, India, raja.jainuniversity@gmail.com

ABSTRACT

The invasion of machine learning on various field in engineering in recent days is quite astonishing. The recent growth in new malwares have put a burden on our traditional anti malwares that use signature based or heuristic based techniques to detect malwares as these either cannot detect zero-day malwares or it would be insufficient to detect a certain type of malware. So, we need to find some new technique to deal with this situation. In this survey paper we shall look into how machine learning can potentially be used as an anti-malware.

Key words: Heuristic based approach, Machine Learning, Malware, Malware Detection technique

1. INTRODUCTION

In this day and age, we live in an era where everything around us has been digitized. With the emergence of the vast digitization, we also face the downside of it i.e., the issues of malwares on our system. With the widespread and high dependence of us on computer systems, it is very important to make sure there are not any damage caused by malwares.

Malwares are rapidly increasing in numbers, there are new malwares being launched at a very fast phase. This is something of a great concern to the computer devices as we all are at a great risk of being infected by these malwares. Malwares can be categorized as virus, worms, spywares, adware's, trojans, Ransomware, etc. Table 1 Shows few malwares that are currently being circulated in the Cyber Space. It is important to note that these malwares can cause a great deal of damages to the systems that we use including corrupting, even deleting the files, stealing data, etc. Basically, malwares can be defined as a piece of software that is created and circulated with a bad intention to steal data, corrupt files, delete files, etc. In table 2 we can see how the malwares have changed through the generation, while comparing New Generation and old traditional malwares.

To protect genuine users and corporate space from malware, malware needs to be identified. Malware detection is the process of knowing whether a given system has malicious intentions or not. In the old days, the signature-based technique of signing was mainly used to identify malware. However, this technique has certain kinds of limitations as it cannot identify the Zero-day malwares [1].

This Zero-day malwares can be identified by machine learning techniques. There is another technique to identify malwares, i.e., using heuristic based technique although this technique can detect zero-day malwares, it still has a drawback of giving a lot of false positive cases [2].

There are various machine learning algorithms that can be used to identify malwares i.e., namely isolation forest, random forest, support vector machine, gradient booster etc. However, detecting all modern malware is a tedious task. Malwares can be entered into any kinds of systems including windows, Linux, android and cloud systems [3].

Table 1: Few Malwares that are in cyber space

Malware	**What they do**
GrooveMonitor.exe	Files do not respond or they become missing.
indexervolumeguid	Once system gets infected with this it tries to infect every file.
rundll32.exe	Overfloods CPU resources
WannaCry	Encrypts the system and asks for ransom to send key that is used to decrypt it
Smishing.c	Attracts users into a phishing site using spam

Table 2: New Generation Vs Traditional malwares

Criteria for comparison	**New Generation**	**Traditional**
Level of coding	Hard coded	Simple coded
Similarity of malwares	Copies are different	Copies are similar
Mode of infection	Uses all types of extension	Only uses .exe
Stay till	Persistent	Temporary
Attack technique	Targeted	Generalized
Defense needed	Sophisticated	Simple
Target Users	All types of users including IOT devices	Normal computer users

2. PROBLEM DEFINITION

Here we recognize the problems in detecting malwares. We can safely say that it is quite an impossible task to detect all the existing and zero-day malwares with any technique that are currently in existence. This is because the problem of identifying the malware has shown NP-complete in different researches [4]. This is crucial because before beginning to develop a sophisticated detection system, it is a great practice and experience for scholars to determine the scope, limitation, and possibility of malware identification. The scope of identifying malware is staying hectic the reason is in theory it is a tedious issue, and in practice malware developer using sophisticated methods such as obfuscation to make identifying process very difficult.

2.1 Difficulty of problem in theory

The very initial malware that came into existence in the cyber space was a virus, many of the research had been performed in theory were based on the identification of virus. With regards to old research, the identification of virus is not possible [5]–[7] and NP-complete [8]–[11]. According to F. Cohen, the identification of computer virus is an unprecedented task because identification method in itself has a contradiction [3], [5], [6]. If the identification issue is seen as a choice problem, D(decision-maker) will get to judge whether P is a malware or not. According to Cohen, it is not possible to be able to decide whether it is a malware because if P is a malware, it will raise flag as D as a malware and will not be able to make changes to other programs, as it will not act as a malware. If D choice maker did not detect P as a malware, P will interact with other programs to infect and become infected.

2.2 Difficulty of problem in practice

The recent gen malware uses the very usual obfuscation methods such as encryption, oligomorphic, polymorphic, metamorphic, stealth, and packing techniques to make identification method more tedious. This type of malware can easily breach defense software that is being run on kernel mode such as firewalls, antimalware software, etc. and some malware instances can also present the characteristics of multiple classes simultaneously. This makes it practically not possible to identify all malware with single identification technique.
The definition of most used obfuscation methods is:
•Encryption: In this technique malwares change the context of the code making it look like something else to the host [10].
•Polymorphic: Here the malware uses a different key for encryption and decryption [12] similarly the key used in oligomorphic method. The encryption payload portion has many copies of the decoder and can be encrypted in layered [13]. Therefore, it is highly difficult to identify polymorphic malware when comparing to oligomorphic malware.
•Metamorphic: This technique does not use any kind of encryption. It makes use of dynamic code hiding which the opcode changes in each turn when the malicious process is executed [14]. It is highly difficult to identify such malware as each new copy has a whole different signature.
•Stealth: Also known as code protection, puts forward various counter methods to eliminate it from being analyzed properly [11]. Consider, it is capable of making changes on the device and keep it hidden from identifying systems.

3. MALWARE DETECTION APPROACHES

We shall be looking at main three approaches that could be used namely signature-based approach, Heuristic based approach and Machine learning approach

3.1 Signature based:

It basically detects every malware in a unique way. That is, it has to know the malware in prior to detect it. We need to keep updating it in order to keep it functional to new virus. It cannot detect zero-day malwares. Even the malware of same family can go undetected if it's a zero-day malware. Although, it is highly used in commercial purpose as it is fast and efficient in many ways.

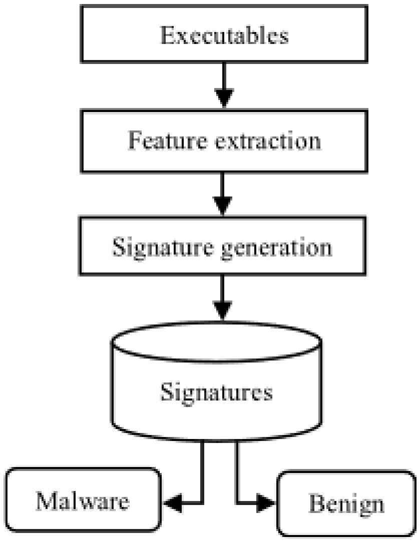

Figure 1: Signature based method

Figure 1 shows the signature-based method. Initially the executables are taken i.e., it could be any executable file like .exe flies, it is then sent to extract features of that executable file, all the features of the file are extracted and then a signature is generated, later the signature is checked to determine if it's a malware or Benign.

Related work on signature-based detection:
F. Zolkipli and Jantan has given a new technique using genetic algorithm and S-based algorithm, this is the framework that has been used to identify malwares [12]. Although the claim of author says it can identify malwares but there simply isn't enough data to support that claim such as test results, total malware analyzed, and comparison of

proposed method with other existing studies. Tanget al. has given an innovative method of bioinformatic method which gives a pin point exploit-based signatures for polymorphic worms [13]. This method contains three steps: multiple sequence alignment to reward consecutive sub-string extractions, noise elimination to remove noise effects, and signature transformation to make the simplified regular expression signature compatible with current IDSs. The authors have claimed that given schema is sound-resistant, is much more accurate and precise than those generated by some other exploit-based signature generation schemas. The reason behind it is it removes more polymorphic worm features like single-byte invariants and distance restrictions between invariant bytes. However, proposed schema is limited to polymorphic worm and cannot be generalized to other malware types. Borojerdi and Abadi proposed a Malware Hunter identification system which is a new technique based on sequence clustering and alignment [14]. It produces signatures computer generated based on malware features for polymorphic malware. The method functions as shown here: Initially, from various malware samples, behavior sequences are generated. After that, considering on identical behavioral sequences, different clusters are saved in the database. To identify malware sample, behavior sequences are taken and compared with sequences which was already extracted earlier and saved in the database [15]. Based on the comparison, the sample is marked as malware or benign.

3.2 Heuristic based approach

In the ongoing years, heuristic based recognition approach has been utilized often [16] and it is an unpredictable discovery strategy which uses encounters and various strategies. For example, rules and ML methods [10]. Despite the fact that it has a high exactness rate to distinguish zero-day malware in a specific way, it can't identify convoluted malware and Heuristic-based identification blueprint can be found in Figure 2.

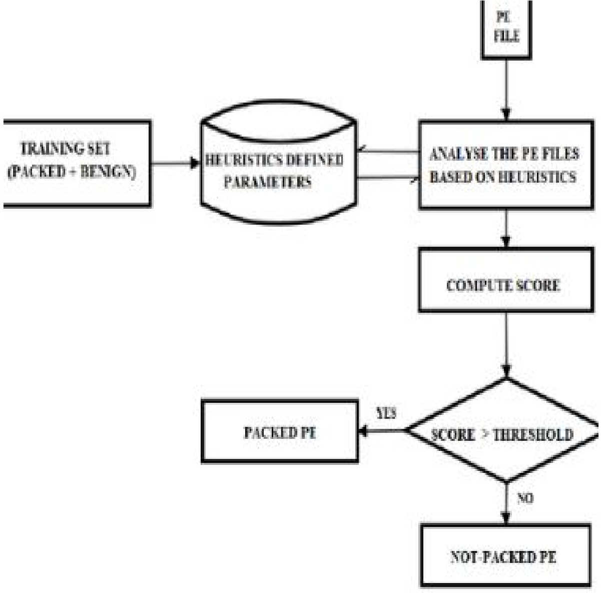
PE FILE
TRAINING SET (PACKED + BENIGN)
HEURISTICS DEFINED PARAMETERS
ANALYSE THE PE FILES BASED ON HEURISTICS
COMPUTE SCORE
YES
SCORE > THRESHOLD
PACKED PE
NO
NOT-PACKED PE

Figure 2. Heuristic based Approach

Related work on heuristic based approach:

Arnold and Tesauro proposed a therefore generated Win32 heuristic disease area in and they automatically construct distinctive neural association classifiers which can detect unknown Win32 contaminations. Overall, heuristic chart has high FP rate, anyway the makers ensure that by joining the individual classifier yields using a majority rule framework. The risk of FP is diminished to an emotionally low level. The assessment is limited to Win32 contamination. It can be contacted other mal-item and more malware ought to be broke down for this technique. Moreover, Expert-arranged heuristic features can improve the performance.

Yanfanget al. proposed post-dealing with techniques of helpful portrayal for malware recognizable proof [17] and the proposed structure unfathomably decreased the number of generated rules by using rule pruning, rule situating, and rule selection. This way, the system didn't need to deal with a large database of rules, which similarly stimulates the disclosure time and exactness rate and according to the paper, the proposed system beat notable antivirus programming contraptions such as McAfee, VirusScan and Norton Antivirus. It outperformed data-mining-based acknowledgment systems including guileless Bayes, support vector machine (SVM), and decision tree techniques and to accumulate all the more Programming interface calls which can give more information about malware and perceive complex associations among the Programming interface calls may improve the performance. Since standard imprint-based adversary of disease structures misfire to detect polymorphic, extraordinary, and heretofore unknown malicious executables heuristic-based malware area is explained in [18], [19].

Yanfanget al. proposed intelligent malware disclosure structure (IMDS) [20] and the IMD Sussed objective-orchestrated association (OOA) mining that works based on windows Programming interface calls. The procedure contains 3parts: PE (helpful executables) parser, OOA rule genera-pinnacle, and rule-based classifier. PE parser removed Windows API execution calls from PE. OOA Fast FP-Advancement algorithm used Programming interface calls and made alliance rules and finally based on the connection rules, OOA mining estimations pre-molded and executables stepped toxic or kind. The paper claims that the proposed structure performed better than various methods including against contamination programming such as Norton Anti-Virus, McAfee VirusScan and KAV, likewise as the systems using data mining techniques, for instance, naïve Bayes, SVM and decision tree, and to vanquish the disadvantages of imprint, and social based malware detection approaches

B. Zahra,et al. proposed heuristic sort of method which can recognize malware that can't be distinguished by previous two approaches [19]. Makers applied learning count to generate a model which resembled mark and based on the signature, new questionable tasks were stepped mal-item or friendly. The paper referred to Programming interface structure calls, operational code (Opcode), n-grams, control stream graph (CFG). Cross variety incorporates that are used broadly in heuristic approach [19]. A genuine assessment of opcode repeats flows to identify and separate current (polymorphic and meta-morphic) malware is explained in and a whole of 67 malware executables were reviewed statically destroyed and their statistical opcode repeat flows were compared with the absolute experiences of 20 non-malicious samples. Test results demonstrated that there is a really gigantic difference in opcode dispersal among malware and benign and to get more reliable results, more models ought to be analyzed and suggested methodology results' ought to be compared with other striking heuristic procedures. An area system that joins static and dynamic features has been suggested in [21] and according to the paper, combining static and dynamic features improve the procedure execution. By combining these features, the part vector was assembled and classified using ML classifiers and the paper attests that the detection rate of the proposed structure is pleasant and extended when compared to their first assessment. In any case, the probability of recognizing dark malware is still low and FPR is high and using more specific features and train the model with more malware may improve the strategy execution for unknown malware.

3.3 Machine learning approach

In this approach we use various machine learning algorithms, ML is basically a collection of algorithms which can give output without specifically programming it. It takes the input from a dataset and gives an appropriate analyzed output. They are various task that can be done on machine learning such as regression, clustering and classification, this approach for detecting malware has been used from quite a few years [22]. Some of the famous machine learning algorithms are linear regression, logical regression, k-nearest neighbour(knn), Naïve Bayes, Random forest, Gradient Boosting, isolation forest, multi-layered perceptron (MLP) etc.

Dataset

As in other exploration territories, there are very few datasets distributed beforehand which are acknowledged and generally utilized for malware identification and moreover, the greater part of the current datasets is not open for research, and as a rule the datasets got are not in the suitable organizations for information mining cycles and ML calculations. The datasets utilized in malware identification can be recorded as follows: NSL-KDD, Darebin, Microsoft malware grouping challenge, Brace (order of Malware with PE headers), AAGM, and Coal dataset

.NSL-KDD dataset (2009), It is a refreshed adaptation of the KDD'99 dataset which comprises of approximately125,000 records and 41 highlights [23] and It shows the organization related assaults which are utilized for interruption discovery system.

Darebin dataset (2014), this dataset is made for cell phones to look at the viability of the current enemy of infection programming [24] and It comprises of 5560 malware across20 families and 123,453 kind hearted samples.

Microsoft malware order challenge dataset (2015): It has been distributed by Microsoft and comprises of 20,000 malwares [25]. Malware has been breaking down utilizing the IDA parcel disassembler and the yield ought to be prepared utilizing information mining preceding ML

ClaMP dataset (2016), it comprises of 5184 records and has55 properties [26] and the dataset utilizes Programming interface exhibits, contains instances of malignant and favorable programming with their features

.AAGM dataset (2017), It is an organization based dataset for android malware [27] and It comprises of 400 malware and 1500 kind hearted examples from 12 families [27]

EMBER dataset (2018), it comprises of 1 million records. It holds malware and amiable highlights [28] and these datasets can be utilized for explores who need to get some insight before proposing another malware recognition approach.

Related work on machine learning approach:

Gavriluț, D., Cimpoeşu, et al this paper standard objective was to create an artificial intelligence structure that traditionally perceives as much malware tests as it can, with the extraordinary restriction of having a zero false positive rate and were outstandingly close to their target, despite the way that they still have a non-zero false positive rate and all together that this framework to end up being significant for a significantly genuine business thing, different deterministic exception frameworks have to be incorporated [28]. As they might want to think, malware area by methods for machine learning won't displace the standard area strategies used by against malware merchants, anyway will come as an extension to them and any business against malware is needy upon certain speed additionally, memory limitations, in this way the most reliable computations among those presented here are the course lopsided perceptron (COS-P) and its explicitly arranged variety (COS-PMap). Since most Antivirus make sense of how to have an ID movement of over 90%, it follows that an extension of the outright acknowledgment movement of 3% − 4% as the one made according to our observations, is particularly critical and as of this second, our framework was exhibited to be a significant assessment device for the PC security experts at Bitdefender Antimalware Investigation Labs [29].

Zhao, J., Zhang, et al This paper presented another malware area strategy using Computer based intelligence subject to the mix of dynamic and static features and the static segment uses IDA Python to isolate strings and DLLs information in PE reports, while the dynamic feature uses instrumentation development reliant on Intel PIN to remove get together direction courses of action and Programming interface call repeat. For features, for instance, string, DLL, Programming interface, information gain figuring is used to pick more huge features and for get together direction course of action, n-gram estimation is used to eliminate features. The Honest Bayesian and SVM gathering models made last refined 97% and 98% precision for the disclosure rate of malware programs [30]

Evaluation of the approaches using graphical presentation:

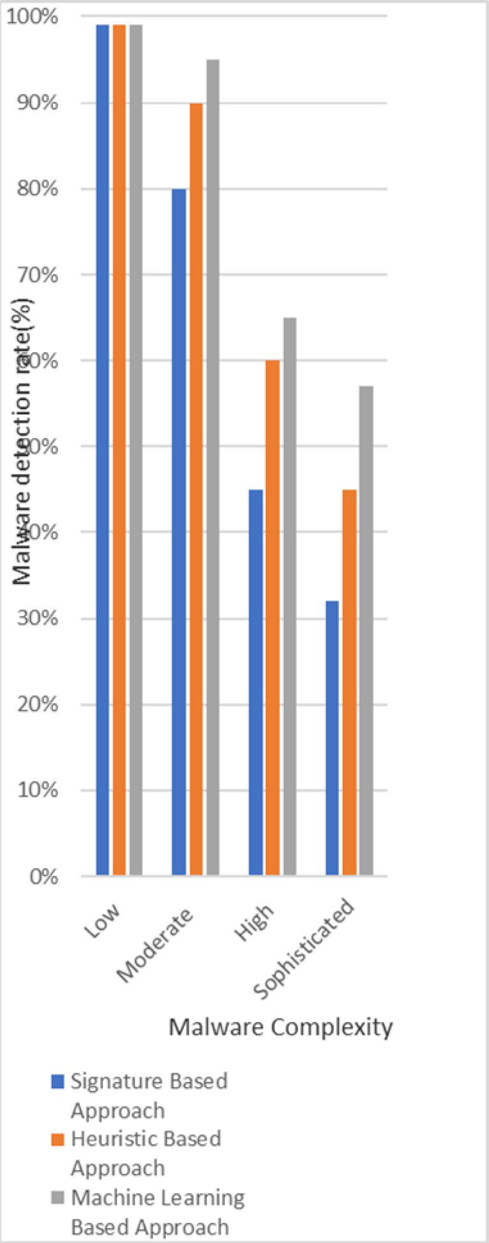

Figure 3. Malware detection rate Versus the complexity of Malware

In Figure 3 we can notice the relation between Malware detection rate when we take complexity of Malware into consideration. We can clearly make a conclusion that Machine Learning Approach has outperformed in these instances.

4. CONCLUSION

Although we have various different method in existence to detect malwares, we still lack in detecting highly sophisticated malwares. We have a huge gap to fill in detecting sophisticated malwares. We can clearly see that Signature based performs well with known malwares but it fails miserably when it comes to malwares that are unknown or Zero-day malwares. Heuristic Based has performed well compared to signature-based approach in detecting new or zero-day malwares, but still it has a lot of gap to fill in detecting malwares and also it results in false positive cases. Machine learning Based approach has done the best compared to other two approaches. It has also shown positive result in detecting unknown or Zero-day malwares and also has shown very less false positive cases but still even this approach has a lot of gap to fill in detecting highly sophisticated malwares. The new malwares that are emerging in the cyber space is very strong against the current anti-malwares techniques we have. We could add Machine learning as an added module with current signature based anti-malware to make it more efficient.

REFERENCES

1. Zhao, J., Zhang, S., Liu, B. and Cui, B., 2018, July. **Malware detection using machine learning based on the combination of dynamic and static features**. In *2018 27th International Conference on Computer Communication and Networks (ICCCN)* (pp. 1-6). IEEE.
2. Aslan, Ö.A. and Samet, R., 2020. **A comprehensive review on malware detection approaches**. *IEEE Access*, *8*, pp.6249-6271.
3. Gupta, M.K., Shaw, S. and Chakraborty, S., **Pattern Based Malware Detection Technique in Cloud Architecture.** (table1)
4. Gao, Y., Lu, Z. and Luo, Y., 2014, August. **Survey on malware anti-analysis.** In *Fifth International Conference on Intelligent Control and Information Processing* (pp. 270-275). IEEE.
5. F. Cohen, **Computer viruses**, Ph.D. dissertation, Univ. Southern California, Los Angeles, CA, USA, 1986.
6. F. Cohen, **A formal definition of computer worms and some related results,** Comput. Secur., vol. 11, no. 7, pp. 641–652, Nov. 1992.
7. D. M. Chess and S. R. White, **An undetectable computer virus**, inProc.Virus Bull. Conf., vol. 5, 2000.
8. F. Cohen, **Computer viruses: Theory and experiments**, Comput. Secur.,vol. 6, no. 1, pp. 22–35, 1987.
9. L. M. Adleman, **An abstract theory of computer viruses**, **in Advances in Cryptology—CRYPTO**. New York, NY, USA: Springer-Verlag, 1990.
10. D. Spinellis, **Reliable identification of bounded-length viruses is NP-complete**, IEEE Trans. Inf. Theory, vol. 49, no. 1, pp. 280–284,Jan. 2003.
11. Z. Zuo, Q. Zhu, and M. Zhou, **On the time complexity of computer viruses**, IEEE Trans. Inf. Theory, vol. 51, no. 8, pp. 2962–2966,Aug. 2005.
12. M. F. Zolkipli and A. Jantan, **A framework for malware detection using combination technique and signature generation**, in Proc. 2nd Int. Conf.Comput. Res. Develop., May 2010.

13. Y. Tang, B. Xiao, and X. Lu, **Using a bioinformatics approach to generate accurate exploit-based signatures for polymorphic worms**, Comput.Secur., vol. 28, no. 8, pp. 827–842, Nov. 2009.
14. H. Razeghi Borojerdi and M. Abadi, **MalHunter: Automatic generation of multiple behavioral signatures for polymorphic malware detection**, inProc. ICCKE. Mashhad, Iran: Ferdowsi Univ. Mashhad, vol. 1, Oct. 2013.
15. J. Newsome, B. Karp, and D. Song, **Polygraph: Automatically gener-ating signatures for polymorphic worms**, inProc. IEEE Symp. Secur.Privacy (Samp;P), Oakland, CA, USA, May 2005, pp. 226–241.
16. Adkins, L. Jones, M. Carlisle, and J. Upchurch, **Heuristic malwaredetection via basic block comparison,** inProc. 8th Int. Conf. MaliciousUnwanted Softw., Amer. (MALWARE), Oct. 2013.
17. Y. Ye, T. Li, Q. Jiang, and Y. Wang, **CIMDS: Adapting postprocessing techniques of associative classification for malware detection**, IEEETrans. Syst., Man, Cybern. C, Appl. Rev., vol. 40, no. 3, pp. 298–307,May 2010.
18. Y. Ye, D. Wang, T. Li, D. Ye, and Q. Jiang, **An intelligent PE-malware detection system based on association mining,** J. Comput. Virol., vol. 4,no. 4, pp. 323–334, Nov. 2008.
19. Z. Bazrafshan, H. Hashemi, S. M. H. Fard, and A. Hamzeh, **A survey on heuristic malware detection techniques,** inProc. 5th Conf. Inf. Knowl.Technol., May 2013.
20. D. Bilar, **Opcodes as predictor for malware**, Int. J. Electron. Secur.Digit. Forensics, vol. 1, no. 2, p. 156, 2007.
21. R. Islam, R. Tian, L. M. Batten, and S. Versteeg, **Classification of mal-ware based on integrated static and dynamic features**, J. Netw. Comput.Appl., vol. 36, no. 2, pp. 646–656, Mar. 2013.
22. E. Gandotra, D. Bansal, and S. Sofat, **Malware analysis and classification: A survey**, J. Inf. Secur., vol. 5, no. 2, pp. 56–64, 2014.
23. Tavallaee, **A detailed analysis of the KDD CUP 99 data set**, in Proc.IEEE Symp. Comput. Intell. Secur. Defense Appl., 2009, pp. 1–6.
24. D. Arp, M. Spreitzenbarth, M. Hübner, H. Gascon, and K. Rieck, **Drebin: Effective and explainable detection of Android malware inyour pocket**, inProc. Netw. Distrib. Syst. Secur. Symp., vol. 14, 2014,pp. 23–26.
25. R. Ronen, M. Radu, C. Feuerstein, E. Yom-Tov, and M. Ahmadi, **Microsoft malware classification challenge**, 2018,arXiv:1802.10135.[Online]. Available: https://arxiv.org/abs/1802.10135
26. **Classification of Malware PE Headers**. Accessed: Nov. 14, 2019.[Online]. Available: https://github.com/urwithajit9/ClaMP
27. A. H. Lashkari, A. F. A. Kadir, H. Gonzalez, K. F. Mbah, andA. A. Ghorbani, **Towards a network–based framework for Android mal-ware detection and characterization**, in Proc. 15th Annu. Conf. Privacy,Secur. Trust (PST), Aug. 2017.
28. S. Anderson and P. Roth, **EMBER: An open dataset for training static PEmalware machine learning models**, 2018,arXiv:1804.04637. [Online].Available: https://arxiv.org/abs/1804.04637
29. Gavriluţ, D., Cimpoeşu, M., Anton, D. and Ciortuz, L., 2009, October. **Malware detection using machine learning**. In *2009 International Multiconference on Computer Science and Information Technology* (pp. 735-741). IEEE.
30. Zhao, J., Zhang, S., Liu, B. and Cui, B., 2018, July. **Malware detection using machine learning based on the combination of dynamic and static features**. In *2018 27th International Conference on Computer Communication and Networks (ICCCN)* (pp.1-6).IEEE