IJITCE

ISSN 2347- 3657

International Journal of Information Technology & Computer Engineering

Email : ijitce.editor@gmail.com or editor@ijitce.com

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 12, Issue 1, 2024

MALWARE DETECTION USING MACHINE LEARNING

P.Prashanth, Professor, Department Of CS SICET, Hyderabad
Vanama Tharun Kumar, Kalabandalapati Mounika, Kasoju Akshay, Sura Sai Kiran
UG Student, Department Of CS, SICET, Hyderabad

ABSTRACT:
While aiming to reduce the number of negatives, we propose various functions where different machine le arning can be used to get the difference between malware files and clean files. In this paper, we first use ca scaded singlesided perceptrons to illustrate the idea behind our framework, and then use cascaded nucleate d singlesided perceptrons. After successful testing of the average malware and clean archive, the idea behi nd the framework was sent to the expansion process, which allowed us to solve large malware and collect clean data.

Introduction

Malware is defined as software that enters or damages a computer without the prior permission of the own er. Malware is a general term for all types of computer threats. Basic classifications of malware include pr ogram files and standalone malware. Another way to classify malware is by their specific behavior: worms , backdoors, Trojans, rootkits, spyware, adware, etc. This becomes even more difficult as all malware appli cations now have multiple polymorphic layers to avoid detection or use a utility to update themselves to ne w versions on short notice to avoid being caught by antivirus software. For an example of dynamic data an alysis for malware detection (tested in a virtual environment), interested readers can refer to [2]. The classi cal method of detecting metamorphic organisms is described in [3]. Here we provide some information ab out such procedures. work becomes more efficient. Association rules are also used, but there are known sy mbols on the honey symbols as they are [7]. No) replacement of previous program files. To achieve simila r goals, [9] adopted the Profile Hidden Markov model, which has been previously successful in sequence a nalysis in bioinformatics. This capability is in [10]. In [11], selfmapUsed to describe the behavior of the vi rus in Windows executable files. The search software aims to obtain as many parameters as possible by si mple and easy multilayer combination (cascade) of different models of the perceptron algorithm [12]. Oth er automatic classification algorithms [13] can also be used in this framework, but we do not explore this a lternative. The main steps taken by this framework are summarized below:

TABLE I
NUMBER OF FILES AND UNIQUE COMBINATIONS OF FEATURE VALUES IN THE TRAINING, TEST, AND SCALE-UP DATASETS.

	Files		Unique combinations	
Database	malware	clean	malware	clean
Training	27475	273133	7822	415
Test	11605	6522	506	130
Scale-up	approx. 3M	approx. 180M	12817	16437

TABLE II
MALWARE DISTRIBUTION IN THE TRAINING AND TEST DATASETS.

	Training Dataset		Test Dataset
Malware type	Files	Unique combinations of feature values	Files
Backdoor	35.52 %	40.19%	9.16%
Hacktool	1.53 %	1.73%	0.00%
Rootkit	0.09 %	0.15%	0.04%
Trojan	48.06 %	43.15%	37.17%
Worm	12.61 %	12.11%	33.36%
Other malware	2.19 %	2.66%	20.26%

Because most features are designed to indicate some aspect of the malware profile
Map algorithms on large (large) datasets. Not all malware is actually malware, and not all clean samples ar e clean. This is because the larger the data, the higher the probability that the samples will not be classified in training. Since our algorithm aims to reduce the number of false positives to 0, the detection value (sen sitivity) obtained in large data sets will be less (due to negative problems). In Figure 4, we can see how the detection rate decreases as the data grows. Table X shows that accuracy, specificity, and the number of art ifacts generally decrease as data size increases.

Conclusion and future work

Our main goal is to propose a machine learning system that can detect as many types of malware as possib le, including the hard limit that zero is not a good value. Although our actual accuracy is still not zero, we are close to our goal. In order for this framework to become part of a highly competitive product, many sp ecial exemption procedures need to be added. We believe that malware detection through machine learnin g will complement, not replace, the detection methods used by antivirus vendors. All antivirus operations a re subject to some speed and memory limitations; Therefore, the most reliable algorithms

TABLE X
DETECTION RATE (SE) COMPARISON ON THE SCALE-UP (LARGE) DATASET WHEN TRAINING THE COS-P ALGORITHM.

Dataset	TP	FP	SE	SP	ACC
S10	170	5	51.76%	97.75%	71.94%
S20	309	5	46.94%	98.91%	69.73%
S30	438	6	44.24%	99.22%	68.32%
S40	555	6	42.13%	99.36%	67.18%
S50	648	5	39.32%	99.61%	65.72%
S60	764	5	38.66%	99.68%	65.39%
S70	842	2	36.55%	99.89%	64.29%
S80	969	2	36.82%	99.90%	64.45%
S90	1092	3	36.89%	99.87%	64.48%
S100	1100	3	33.45%	99.88%	62.56%

Presented here are the Cascaded One-Sided Perceptron (COS-P) and its clearly defined model (COS-P-Map). It can be seen that the total detection rate produced by our algorithm increases by 3%~ 4%, which is very significant. (Please note that training was conducted on malware samples that were not detected by the detection process.)

To date, our framework has proven to be useful research for computer security experts in Bit Defender's anti-malware research department. In the future, we plan to integrate more classification algorithms such as wideedge perceptrons [18] and support vector machines [14], [19], [20].

REFERENCES

1.The authors would like to thank BitDefender management for their support on these issues.

Penya, J. Devesa and P. G. Garcia, "N-gram based data signatures for malware detection," 2009. Holz, C. Willems, P. Dussel, and P. Laskov,

2."Learning and Classification of Malware Behavior," DIMVA '08: Proceedings of the 5th International Conference on Intrusion and Malware Detection, and Vulnerability Assessment. Berlin, Heidelberg: Springer-Verlag, 2008, p. 108–125.

[3] E. Konstantinou, “Metamorphosis viruses: analysis and detection”, 2008, Technical Repor t RHUL-MA-2008-2, Search Security Award Master's Thesis, 93 Nplooj . 6, s. 2669–
2672, 2006. Maloof, “Learning to detect and identify malicious behavior in wildlife,” Journal of Machine Learning Research, vol. 7, p. 2721–
2744, December 2006, Special Issue on Machine Learning in Computer Security. Li and D. Y e, “Imds: An intuitive malware detection system,” in KDD, P. Berkhin, R. Caruana, and X. Wu, editors. ACM, 2007, nr 17. 1043-
1047 Ib. Upadhyaya, “Spycon: Simulating user activity to detect and bypass spyware,” in,IPC CC. IEEE Computer Society, 2007, p. 502-509. Walenstein thiab A. Lakhotia, "Machine-passed variants of malware filtering using Markov
chains," Malware and Unwanted Software, 2008. Conference, 2008, p. 77–
84. Stamp, S. Attaluri and S. McGhee, “Examining the latent signature model and metamorph ic virus detection,” Journal of Computational Virology, 2008. Comprehensive detection and c lassification of polymorphic malware, 2006. and Research Data Proceedings. New York, NY, AB: ACM. 82–89. . . . . McGraw-
Hill Education (ISE Edition), October 1997. Cambridge University Press, March 2000. J. Sm ola, Learning with Kernels: Support vector machines, optimization, optimization, and more. MIT Press, 2002. Chauvin, C. A. Andersen, and H. Nielsen, “Accuracy evaluation of distribu tion prediction algorithms,” Bioinformatics, vol. 5, page 5. 412-
424, Mart 2000. K. Mishra, “De novo SVM-like i-
fixation of precursor microRNAs from genomic pseudohairpins using global and internal fold ing methods.” . Freund and R. E. Schapire, “Classification of large flowers using perceptron a lgorithm,” Machine Learning, vol. 37, 1999, s. 277-296.
[20] V. N. Vapnik, Information Science and the Nature of Statistics. Springer, November 199 9.