Unknown Malcode Detection – A Chronological Evaluation

Robert Moskovitch, Clint Feher, Yuval Elovici

***Abstract*— Signature-based anti-viruses are very accurate, but are limited in detecting new malicious code. Dozens of new malicious codes are created every day, and the rate is expected to increase in coming years. To extend the generalization to detect unknown malicious code, heuristic methods are used; however, these are not successful enough. Recently, classification algorithms were used successfully for the detection of unknown malicious code. We earlier investigated the optimized conditions in which highest-level accuracy is achieved, in terms of the percentage of malicious files. In this paper we describe the methodology of detection of malicious code based on static analysis and a chronological evaluation, in which a classifier is trained on files till year k and tested on the following years. The evaluation was performed in two setups, in which the percentage of the malicious files in the training set was 50% or 16%. Using 16% malicious files in the training set showed a clear trend, in which the performance improves as the training set is more updated.**

***Index Terms*— Malicious Code Detection, Classification Algorithms**

I. Introduction

THE term *malicious code* (malcode) commonly refers to pieces of code, not necessarily executable files, which are intended to harm, generally or in particular, the specific owner of the host. The recent growth in high-speed internet connections and internet network services has led to an increase in the creation of new malicious codes for various purposes, based on economic, political, criminal, or terrorist motives (among others).

Current anti-virus technology is primarily based on two approaches. *Signature-based* methods rely on the identification of unique strings in the binary code; while being very precise, this approach is useless against unknown malicious code. The second approach involves *heuristic-based* methods, which are based on rules determined by experts, which define a malicious behavior, or a benign behavior, in order to enable the detection of unknown malcodes [1]. Generalization of detection methods to be able to detect *unknown* malcodes is therefore crucial.

Recently, classification algorithms were employed to automate and extend the idea of heuristic-based methods. Recent studies have shown that this is a very successful strategy [2,3,4]. However, these studies present evaluations based on relatively small test collections, which include roughly 3,000 files. Moreover, in their evaluation the proportion of malicious versus benign files is the same, which does not reflect real life conditions and also presents non accurate results. A recent survey[1] by McAfee indicates that about 4% of search results from the major search engines on the web contain malicious code. Additionally, it was found that above 15% of the files in the KaZaA network contained malicious code. Thus, we assume that the percentage of malicious files in real life is about or less than 10%.

Malicious code creators improve their applications along time; moreover, they are written in varying frameworks which results in different repeating patterns in the binary. This requires updating the training set in order to maintain a high level of accuracy. In this paper we evaluate the static malicious code detection using the Decision Trees classification algorithm from a chronological point of view. The dataset includes files from 2000 to 2007, and is trained each time on files till year *k* and tested on the following years.

II. Methods

A. Dataset Creation

We created a dataset of malicious and benign executables for the Windows operating system, which is the commonly used and most commonly attacked. We acquired the malicious files from the VX Heaven website. Thus, we had 7688 malicious files and 22,735 benign files which we took from computers at our campus. The files were identified and verified as malicious or benign using Kaspersky anti-virus.

B. Data Preparation and Feature Selection

We parsed the files using several n-gram lengths moving windows: 3, 4 and 5-grams, which resulted in millions of distinct terms [6]. Later each n-gram term was represented using its Term Frequency (TF), which is the number of its appearances in the file, divided by the term with the maximal

Robert Moskovitch is a PhD student at the Deutsche Telekom Laboratories at Ben Gurion University, Be'er Sheva, 84105 Israel. : robertmo@bgu.ac.il)
Clint Feher is an MSc student at the Deutsche Telekom Laboratories at Ben Gurion University (clint@bgu.ac.il).
Dr Yuval Elovici is the head of the Deutsche Telekom Laboratories at Ben Gurion University (elovici@bgu.ac.il).

[1] McAfee Study Finds 4 Percent of Search Results Malicious, by Frederick Lane, June 4, 2007
[http://www.newsfactor.com/story.xhtml?story_id=010000CEUEQO]

ISI 2008, June 17-20, 2008, Taipei, Taiwan
1-4244-2415-3/08/$20.00 ©2008 IEEE

appearances, having a value in the range [0,1]. We used feature selection to reduce the amount of features to hundreds. Based on an extensive evaluation [6], we found that the top 300 features of 5-grams from the Fisher-Score selection method outperformed. In our experiments we used several classification algorithms; however, here we report the results of Decision Trees (DT).

III. Experiments and Results

To evaluate the importance of and need for updating the training set, we divided the entire test collection into the years from 2000 to 2007, in which the files were created. Thus, we had 6 training sets, in which we had samples from year 2000 till year 200k. Each training set was evaluated separately on each following year from 200k+1 till 2007. Obviously the files in the test were not present in the training set. We present two experiments which vary in the Malicious Files Percentage (MFP) in the training set, one having 50% which is commonly used and the other 16%, which is expected to maximize the performance, and which was the same in the test set (16%) to reflect real life conditions.

Figure 1 presents the results of the chronological evaluation, in which the MFP in the training set was 50%. Out of the years 2000 and 2001 most of the results are below 0.9 accuracy. The training set of till 2002 outperforms the other training sets out of the 2006 training set. However, generally a significant decrease in performance was seen when testing on 2007.

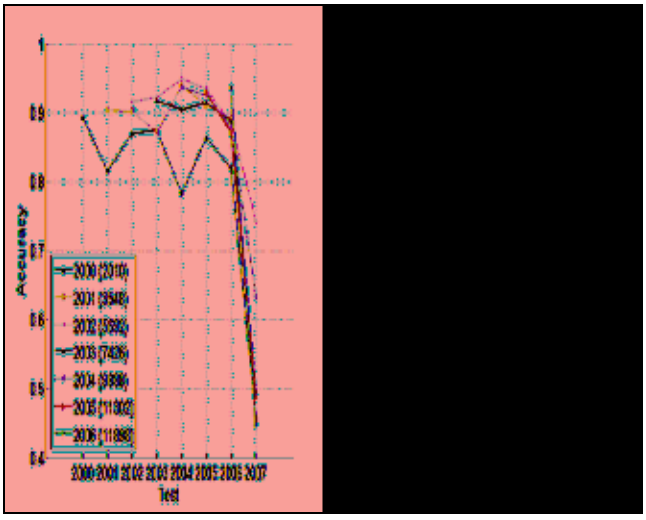

Figure 1. The results of the training set MFP, having 50%. The more updated the higher the accuracy level. A significant decrease is seen in 2007, while training on 2006 outperforms.

Figure 2 presents the results of the chronological evaluation, in which the MFP in the training set was 16%. In an extensive evaluation of several MFPs in the training set and test set, we found that having a low MFP in the training set for real life conditions (where there is low MFP) maximizes the results. In Figure 2 we again see a generally better performance. 2004 introduced a significant challenge for the training sets of till 2000 and 2001. In this set of results there is a clear trend which shows that the more the training set is updated the higher the level of accuracy in the following years, and even when evaluated on 2007 the accuracy level was above 0.9 when trained on training sets till 2004, 2005 and 2006.

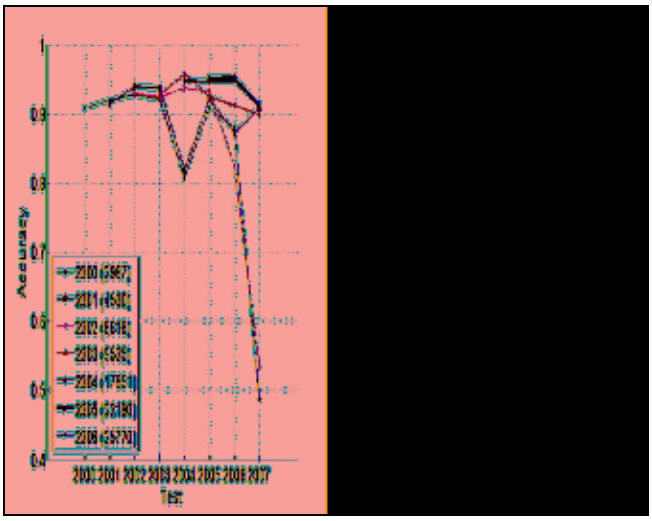

Figure 2. The results of the training set MFP, having 16%. The more updated the higher the accuracy level. Testing on year 2004 presented a challenge for the training set from 2000 and 2001. Generally the performance is higher than the 50% MFP.

IV. Discussion and Conclusions

We presented the problem of unknown malicious code detection using classification algorithms. We described our previous results from an extensive evaluation using several n-grams and feature selection methods to reduce the amount of features. We presented the creation of our test collection, which is 10 times larger than any previously presented. In a previous study, which we published elsewhere, we investigated the aspects of the percentage of malicious files in the training set to maximize the accuracy in real life conditions.

In this study we referred to the question of the importance of updating the training set with the new malicious codes in a yearly time granularity. Our results indicate that when we have 16% MFP in the training set, which corresponds to the test set, we achieve a high level of accuracy, and also a clear trend that the more updated the training set, the higher is the level of accuracy.

V. References

[1] Gryaznov, D. Scanners of the Year 2000: Heuritics, Proceedings of the 5th International Virus Bulletin, 1999.

[2] Schultz, M., Eskin, E., Zadok, E., and Stolfo, S. (2001) Data mining methods for detection of new malicious executables, in Proceedings of the IEEE Symposium on Security and Privacy, 2001, pp. 178-184.

[3] Abou-Assaleh, T., Cercone, N., Keselj, V., and Sweidan, R. (2004). N-gram Based Detection of New Malicious Code, in Proceedings of the 28th Annual International Computer Software and Applications Conference (COMPSAC'04).

[4] Kolter, J., and Maloof, M. (2006) Learning to Detect and Classify Malicious Executables in the Wild, Journal of Machine Learning Research 7, 2721-2744.

[5] Chawla, N. V., Japkowicz, N., and Kotcz, A. (2004). Editorial: special issue on learning from imbalanced data sets. SIGKDD Explorations Newsletter 6(1):1-6.

[6] Moskovitch, R., Stopel, D., Feher, C, Missim, N., Elovici, Y., Unknown Malicious Code Detection via Text Categorization and the Imbalance Problem, IEEE Intelligence and Security Informatics, Taiwan, 2008.