Detection of Malware and Malicious Executables Using E-Birch Algorithm

Dr. Ashit Kumar Dutta
Associate Professor
Department of Computer Science
Alquwayiya College of Science and Humanities
Shaqra University

***Abstract*—Malware detection is one of the challenges to the modern computing world. Web mining is the subset of data mining used to provide solutions for complex problems. Web intelligence is the new hope for the field of computer science to bring solution for the malware detection. Web mining is the method of web intelligence to make web as an intelligent tool to combat malware and phishing websites. Generally, malware is injected through websites into the user system and modifies the executable file and paralyze the whole activity of the system. Antivirus application utilizes the data mining technique to find the malware in the web. There is a need of heuristic approach to solve the malware problem. Dynamic analysis methods yield better result than the static methods. Data mining is the best option for the dynamic analysis of malware or malicious program. The purpose of the research is to apply the enhanced Birch algorithm to find the malware and modified executables of Windows and Android operating system.**

Keywords—Birch; Malware; Executables; Android and Windows

I. INTRODUCTION

Web Intelligence is the new field of computer science. There is a need of protective environment for users in the web. Users are isolated in web and web criminals are grab their attention towards them. Hacking websites, software and Internet accounts, phishing, malwares are the issues remains unsolved in internet. Arrival of web Mining (WM) gave a new definition for internet security; even famous software firms started using WM to find malware and malicious websites in Internet. Web Intelligence is the shield for internet users to protect them from the criminals exists in the web. The following applications are very useful for the internet communities [1][2]. Malware study the system and send the details to the Malware provider and that details will be useful for them to hack the system. Figure 1 shows the application of web mining and finding malicious website is one prime application of it.

A. Find Malicious website

WM methods are more intelligent to find the websites offering malicious software and deposits harmful files in the visitor system. The nature of ordinary websites differs from the malicious websites and WM easily identifies the difference and inform users about the site.

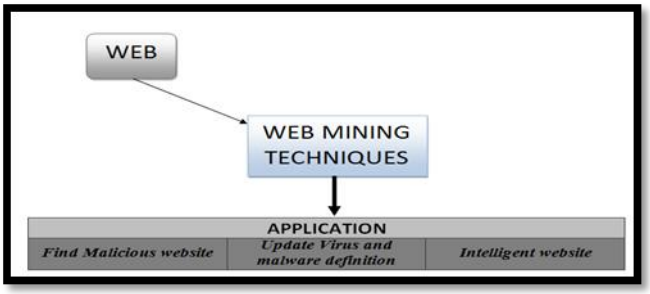

Fig. 1. Application of Web Mining

B. Update Virus and malware definition

It is easy to develop a database to store the virus and malware information. WM identifies newly arrived viruses in internet and stores automatically into the database. User can utilize the updated information to avoid storing harmful information into the system.

C. Intelligent website

Machine learning methods are used to build robots and can be used to design a website to become more intelligent to find their clones exist in internet. It has the ability to eradicate the malicious websites targeting their visitors in web.

Signature and Behavior based approaches are widely used in malware detection. The signature based approach is not effective because of the nature of detecting the malware but the behavior based approach is more effective in the process of detecting the malware and malicious executables. The purpose of research is to build an effective algorithm to detect the malware and malicious executables of operating system. The following sections will provide the data collection, design of algorithm and results and discussion.

Section II discuss the relevant research and its result and section III gives detail about the data collection for the study and Section IV discuss the experimentation and the result of the proposed research and finally section V gives the conclusion and future scope of the research.

II. REVIEW OF LITERATURE

Komashinkskiy.D and Kotenko[1] have proposed a research on malware detection based on positionally

dependent features. The research considers the specificities of object's file format of potential malware containers. It also describes the realization and investigation of the common methodology for design of data mining based malware detectors using positionally dependent static information. Muazzam siddiqui et.al.[2], proposed a work presents a novel idea of extracting variable length instruction sequences that can identify worms from clean programs using data mining techniques. The research deployed tree based classifiers including decision tree, bagging and random forest. Mohammad M. Masud et. al.[3], proposed a research on cloud based malware detection for evolving data streams. The research proposed a multi partition ensemble classifier in which a collection of classifiers trained with fold partitioning of the data, yielding an ensemble of classifiers. The work also proposed a feature extraction and selection technique for data streams that do not have any fixed feature set. J.k.kolter et.al[4]., have done a research in classifying malicious executables in the wild using machine learning methods. The work has gathered malicious executables and encoded each as a training example using n-grams of byte code as features. They have evaluated the methods classified executables based on the function of their payload. The work could be used as the basis for an operational system for detecting previously undiscovered malicious executables.J.Dai et.al.[5], have presented a novel approach to detect unknown virus using dynamic instruction sequences mining techniques. They have built a program monitor which is able to capture runtime instruction sequences of an arbitrary program. The monitor utilizes the derived classification model to make an intelligent guess based on the information extracted from instruction sequences to decide whether the tested program is benign or malicious.

III. Data Collection

Identification of malware is a difficult task, many commercial websites failed to detect it. Some websites are offering malware executables for the purpose of research[6][7][8]. 250 malwares were downloaded from www.vxnetlux.org, www.29a.net and www.vxheaven.org. 25 infected executables of windows 7 and 10 infected executables of Android 4.4.2 were collected and included in the research to investigate the way of injecting malwares to infect the executables of the operating system. Numerical values were assigned to collected malwares and executables to reduce the complex nature and improve the performance of Birch algorithm[9][10][11].

IV. Birch Algorithm

Balanced iterative reducing and clustering using hierarchical (BIRCH) is an efficient clustering algorithm for large data set. It is a type of hierarchical clustering uses a clustering feature (CF) tree to reduce the memory size of the algorithm. The CF – tree uses 3 parameters (N, L S,SS): N – Number of objects in the cluster ; LS – Linear sum of data points; SS – Square sum of data points; The following formula use to evaluate LS and SS

$$LS = \sum_{pi \leftarrow N} \overrightarrow{pi}$$

$$SS = \sum_{pi \leftarrow N} |\overrightarrow{pi}|^2$$

CF tree is a multi-level compression of data that has the inherent clustering structure of the data. The limitation of main memory minimizes the time complexity of the data[12][13[14]. Birch algorithm is sensitive to the order of data, lacks in performance to cluster the non – numeric data. Generally it works well in spherical shape clusters but does not perform well for other shapes [15][16][17].

V. Results and Discussion

Birch algorithm performance is limited for certain circumstances and it is not possible to produce better results. The CF tree parameters have been modified to two parameters and the value of LS is neglected for the study of malware and malicious executables[19][20][21][22]. The CF tree structured as an image to reduce the size and the total time of the algorithm[23][24][25].

The figure 2 shows the comparison of Birch and Enhanced Birch performance on 250 malware detection[26][27][28]. Initially, the performance of both algorithms was same but the Enhanced Birch shown better result in the later part of the malware[29][30][31].

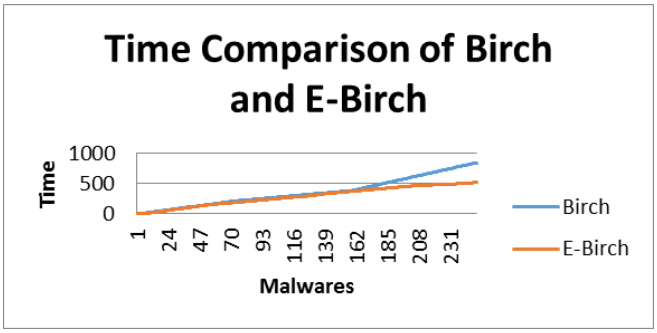

Fig. 2. Detection of Malware

The figure 3 shows the performance of the 25 executables of windows operating system. Both algorithms generate the optimum result with the mild difference in the execution time.

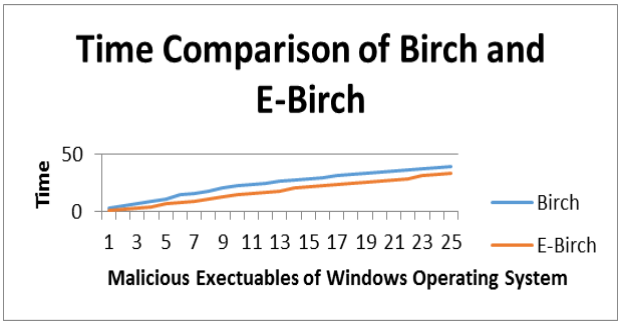

Fig. 3. Detection of Windows Malicious Executables

The figure 4 shows the performance of algorithm on 10 executables of Android operating system. The results are optimum but enhanced birch algorithm execution time is better than the Birch algorithm

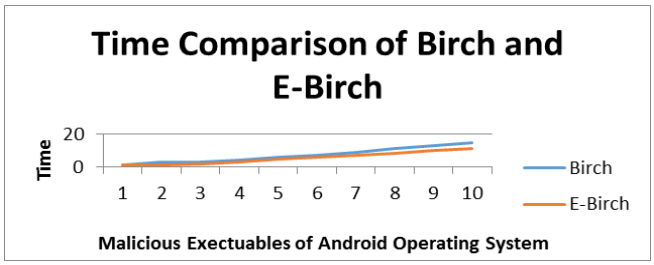

Fig. 4. Detection of Android Malicious Executables

TABLE I. EXECUTION TIME OF ALGORITHMS ON MALWARE AND MALICIOUS EXECUTABLES

Algorithm	Time to detect Malware (seconds)	Time to detect Malicious Executables of Windows operating system (seconds)	Time to detect Malicious Executables of Android operating system (seconds)
Birch	841	39	15
E-Birch	519	33	11

VI. CONCLUSION

The study proved that enhancement of BIRCH generates the accurate result with less memory and computation time. Dynamic analysis of malware detection will also generate the optimum results as the algorithm has learnt the behavior of malware and malicious executables. Malwares of Windows and Android were deployed in BIRCH AND E-BIRCH and found that E-BIRCH overall performance is better than the BIRCH. The future scope of the study is to design a hybrid application of signature and behavior based malware detection by implementing E-BIRCH.

REFERENCES

[1] Komashinskiy, D.; Kotenko, I., "Malware Detection by Data Mining Techniques Based on Positionally Dependent Features," in Parallel, Distributed and Network-Based Processing (PDP), 2010 18th Euromicro International Conference on , vol., no., pp.617-623, 17-19 Feb. 2010 doi: 10.1109/PDP.2010.

[2] Muazzam Siddiqui, Morgan C.Wang and Joohan Lee, " Detecting Internet worms using data mining techniques", Journal of Systemics, Cybernetics and Informatics, 6, 48-53. (2009).

[3] Mohammad M.Masud, Tahseen M.Al-Khateeb, K.W.Hamlen, Jing Gao, Latifur Khan, Jiawei Han and Bhavani Thuraisingham," Cloud based malware detection for evolving data streams", ACM – Transaction, September 22, 2011,12.6.

[4] J.Zico Kolter and Marcus A.Maloof, " Learning to detect and classify malicious executables in the wild", Journal of Machine learning research 7(2006),2721 – 2744.

[5] J.Dai, R.Guha, and J.Lee, "Efficient Virus Detection Using Dynamic Instruction Sequences," Journal Of Computers, Vol. 4, No 5, May 2009.

[6] G.H.John, and P.Langley, "Estimating Continuous Distributions in Bayesian Classifiers," Proceedings of the 11-th Conference on Uncertainty in Artificial Intelligence, 1995.

[7] R.Kohavi, "The Power of Decision Tables," Proceedings of European Conference on Machine Learning, 1995.

[8] J.Kolter, and M.Maloof, "Learning to Detect Malicious Executables in the Wild,". Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2004.

[9] T.Mitchell, Machine Learning. The Mc-Graw-Hill Companies, Inc., 1997.

[10] M.Pietrek, "An In-Depth Look into the Win32 Portable Executable File Format," MSDN Magazine, 2002.

[11] R.Quinlan, "C4.5," Programs for Machine Learning, Morgan Kaufmann Publishers, 1993.

[12] M.Schultz, E.Eskin, E.Zadok, and S.Stolfo, "Data Mining Methods for Detection of New Malicious Executables," Informatics and Computer Science, Volume 172, Issue 1-2, 2001.

[13] VX Heavens Site. http://vx.netlux.org/ .

[14] J.-H.Wang, P.S.Deng, Y.-S.Fan, L.-J.Jaw, and Y.-C.Liu, "Virus Detection using Data Mining Techniques," Proceedings of IEEE 37th Annual 2003 International Carnahan Conference, 2003.

[15] N.Ye. (ed.). The Handbook of Data Mining. Lawrence Erlbaum Associates, Publishers, London, 2003.

[16] B.-Y.Zhang, J.-P.Yin, J.-B.Hao, D.-X.Zhang, and S.-L.Wang. "Using Support Vector Machine to Detect Unknown Computer Viruses," International Journal of Computational Intelligence Research, Vol.2(1), 2006.

[17] 30L.Breiman, "Random Forest," Machine Learning 45 (1), 2001.

[18] Bayer, U., Kruegel, C. and Kirda, E. (2006) TTAnalyze: A Tool for Analyzing Malware. Proceedings of the 15th European Institute for Computer Antivirus Research Annual Conference.

[19] Dinaburg, A., Royal, P., Sharif, M. and Lee, W. (2008) Ether: Malware Analysis via Hardware Virtualization Extensions. Proceedings of the 15th ACM Conference on Computer and Communications Security, CCS'08, Alexandria, 27-31 October 2008, 51-62.

[20] ThreatExpert. http://www.threatexpert.com/submit.aspx

[21] Schultz, M., Eskin, E., Zadok, F. and Stolfo, S. (2001) Data Mining Methods for Detection of New Malicious Executables. Proceedings of 2001 IEEE Symposium on Security and Privacy, Oakland, 14-16 May 2001, 38-49.

[22] Cohen, W. (1995) Fast Effective Rule Induction. Proceedings of 12th International Conference on Machine Learning, San Francisco, 115-123.

[23] Kolter, J. and Maloof, M. (2004) Learning to Detect Malicious Executables in the Wild. Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 470-478.

[24] Nataraj, L., Karthikeyan, S., Jacob, G. and Manjunath, B. (2011) Malware Images: Visualization and Automatic Classification. Proceedings of the 8th International Symposium on Visualization for Cyber Security, Article No. 4.

[25] Nataraj, L., Yegneswaran, V., Porras, P. and Zhang, J. (2011) A Comparative Assessment of Malware Classification Using Binary Texture Analysis and Dynamic Analysis. Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence, 21-30.

[26] Kong, D. and Yan, G. (2013) Discriminant Malware Distance Learning on Structural Information for Automated Malware Classification. Proceedings of the ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems, 347-348.

[27] Tian, R., Batten, L. and Versteeg, S. (2008) Function Length as a Tool for Malware Classification. Proceedings of the 3rd International Conference on Malicious and Unwanted Software, Fairfax, 7-8 October 2008, 57-64.

[28] Tian, R., Batten, L., Islam, R. and Versteeg, S. (2009) An Automated Classification System Based on the Strings of Trojan and Virus Families. Proceedings of the 4th International Conference on Malicious and Unwanted Software, Montréal, 13-14 October 2009, 23-30.

[29] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. (2009) The WEKA Data Mining Software: An Update. ACM SIGKDD Explorations Newsletter, 10-18.

[30] Santos, I., Nieves, J. and Bringas, P.G. (2011) Semi-Supervised Learning for Unknown Malware Detection. International Symposium on Distributed Computing and Artificial Intelligence Advances in Intelligent and Soft Computing, 91, 415-422.

[31] Moskovitch, R., Stopel, D., Feher, C., Nissim, N. and Elovici, Y. (2008) Unknown Malcode Detection via Text Categorization and the Imbalance Problem. Proceedings of the 6th IEEE International Conference on Intelligence and Security Informatics, Taipei, 17-20 June 2008, 156-161.