A Novel Approach for Predicting the Malware Attacks

Ekta Rokkathapa
DIT University
Dehradun, India

Soumen Kanrar
Asst. Professor
DIT University
Dehardun, India

ABSTRACT
Malware means malicious software. Detecting malware over a system is malware analysis. It consists of two parts static analysis and dynamic analysis. Static analysis includes analyzing a suspicious file and dynamic analysis means observing a file during its process time. In this paper, we have proposed a framework for malware analysis based on semi automated malware detection usually machine learning which is based on dynamic malware detection . The framework shows the quality of experience (QoE) to maintain the efficiency tradeoffs and uses the method of classification. The samples of malware also shows that the framework create a strong detection method.

Keywords
Malware, attacks , disassembler, evasion attacks, machine learning

1. INTRODUCTION
Malware analysis is a process like thief and corps. In the past decade cyber attacks is at top. The reason is that more number of people perform their daily activities and transaction digitally. According to a survey report minimum effort is required to launch a cyber attack because of the attacker tool kits. Malicious software is a major cause of cyber attack incidences.[2] In 2016 , 20% of 40 million files in network were verified as malware. Analysis of malware contains two classification. Static analysis consists of reverse engineering which is implemented by disassembler like IDA pro[1] .But dynamic malware analysis exactly shows malware operation[2]. Some of the tools are regshot,process explorer[1]. To conquer cyber attack by malware the blockage method should be applied from the network traffic. Section 1.1 specifies the methods of malware detection.

1.1 Methods of malware detection
Malware detection usually uses signature methods of viruses to defend against malicious software. Most of the antivirus tools depends on regular expression and pattern to categorised malware. Antivirus lessly update their databases for malware detection and prevention as file features has to update a newly created malware. To generate signature from the updated files required maximum human efforts practically. Malware sample from spreading from signature based malware detection fails to identify new malwares attacks from these challenges. Signature based methods for detection gets fails to identify new malware codes. As the drawback in signature based detection many researchers are on malicious file detection using machine learning. The proposed method if huge amount of malicious file has been extracted with the help of cross validation method in machine learning one can classify malware samples and also those samples can also predicted about the maliciousness present in the sample. Many researchers has studied on static file detection of malicious content using machine by automatic malware detection [3,4,7,8,9]. However , D.Maiorca, N.Srndic, W.Xu et al[10,11,12] has identified that threat of evasion attack is more on static file base malware identification.

W.Xu et al[12] applied methods of generic programming to avoid the evasion attacks by generating and achives 100% success with the samples of evasive variants. Various efforts has been applied in runtime behaviour in suspicious file . Rieck et al[13] proposed machine clustering tool which works on machine learning and it collects behaviour reports. Bayer et al[14,15] generated a report by clustering various malware files and grouping them by data analysis methods. In comparison to static file content malware runtime behaviour cannot be modified easily to create mimicry attacks. Machine learning based methods on dynamic files is superior as it is very hard to conquer by malicious code. But dynamic feature requires more complex implementation methods and higher resource consumption. In developing a dynamic behaviour detection system some researchers has proposed the following works.

Firstly , model using partial behavioural features consisting of a dynamic monitoring usually within few minutes from start of execution , but this method has no surety whether the selected execution time length is near to optimal time for effective report without performing degradation . Secondly , the researcher has focus on achieving high efficiency of system neglecting the cost of the system. But neglecting the cost make those system less interactive malware solution.So as a conclusion QoE provide bridge between accuracy and resource usage of malware detection system under different cases.

In this paper , we have proposed a system which fill the gap user interactive malware identifier system and dynamic behaviour feature. We have taken the consideration that malware specification collected from different samples can uplift resource cost and time with different accuracy . In this work we have proposed efficient online machine learning algorithms that gain its experience over time from samples files for the best matching classifier with QoE metric.

2. RELATED WORK

2.1 Traditional method against malware threats
Various search engine like Google, Bing protect the systems while downloading same file by the user when file seems suspicious [16]. It is done by Matching the URLs against updated malicious URLs by search engine .This methods gets fails if the URL frequently mutated by changing its binaries

2.2 Machine learning based malware detection
Attackers some times gain access to the victim system and can easily modified the targets by neglecting signature detection Signature based detection method stops malware from spreading and also it fails the mutation of new malware. Xu et

al[17] implemented a generic programming model to stop mutation of the malicious samples. The experimental results has shown the classifier achieving 100% evasion rate.Runtime behaviour of malware is much more difficult to modify Bayer et al[18][19][20] has proposed dynamic malware clustering system .They proposed a clusters method to reduce the complexity of distance calculation from n2 to n2/2. Trinus, Reick et al[19][21] has proposed a method to analyze the runtime behaviour of malware using machine learning.

Sommer et al[27] have researched in network intrusion detection system and shown the areas where machine learning can wok successfully. But this paper has shown only limited static features.

2.3 Executable Behaviour of malware

Lo et al[32] has proposed a method to optimize the resource allocation in computer. Although their work has improved the throughput in clusters, but it was not clear that their features are usefull in a security setting.

2.4 Malware information sharing platform

Threat analysis and information sharing gain is more popular to avoid cyber attack but due to limited area ,actions are restricted. Webroot's Bright Cloud Platform [2] is a threat analyze intelligence system that anaylize ,classify , samples and groups of cyber threats. They have shown in their paper that 85,000 malicious URL generate daily and among 40 million new files 15% contains malware content. However , Webroot's cloud didn't proved its accuracy during analysis and classification.

NATO designed Malware info sharing Platform [22] to show cyber defence into system. It is an source project contains a regular updation of knowledge on Malware and containing specification like STIX,TAXII, and CYbox[23].

Some more sharing platform are Alien vault Open Threat Exchange [24], Virus total [18] , Cyber Threat Alliance [25].

3. PROPOSED METHOD

System Model

Malware classification using Behavioural specification are modelled under supervised learning of post experience . Under the model Malware classifier (f) trained the frame work with labelled history feature which is applied by Vectorized features to calculate malicious effectiveness. Malware classifier depends on monitoring time (T) . More T leads to classifier accuracy . Best and worse case of classifier are defined under T.

We have proposed a learning process for multiple classifier f1fk trained for different monitoring length T1Tk. Malware detection system learns in real time for the best classifer. System can be modelled as multi –arm bandit having malware content information .For selecting the best classifier and online classification selection problem has been formulated and the accuracy of the classifier fk can be shown by

A(fk) =E[rt|ft=fk]

The QoE of classifier is done at time t+Tt by selecting the kth classifier

Q(fk)=q(fk)- BC(Tk)

where B€[0,1] shows trade off parameter.

Algorithm Description

We have proposed a new Algorithm from upper confidence bound (UCB) [76,77] , unlike UCB our

algorithm uses sample context to identify best classifier and maximizing QoE for malware detection source. Algorithm maintains multiple counters and accuracy qe (fk) and QoE for every classifier f={f1……fk) under different VL. Nk records maintains classifier fk for round f for the future classification estimated QoE is Q(fk)

Upper selection bound for malware detector selection

Input

A €R+, S={(Θ1,x1),(Θ2,x2)..……,(Θt,xt)},

£={f1,f2,……..,fk), Ķ={1,……,K},

B€[0,1],

M={v1,v2,…….vL} , Ļ={1,……..,L}

Output:

$\{y^1,.....y^t\}€\{0,1)$

1. Initialization:
2. for l€L do
3. for k€K do
4. Randomly select $(Θ^m,X^m)$
5. Set $q_l(f_k)←f_k(X^m)$
6. Set $Q_l(f_k)← q_l(f_k)$-$Bc(T_k)$
7. Set $N_l^k← 1$
8. End for
9. Set $N^l← K$,
10. End for
11. Set N←LK
12. For each malware detection request $(Θ^t,x^t)$ do
13. $L^*=arg_l€L\ \min||Θ^t-V_l||^2$
14. $K^*=arg_{\ k\ €K}max(Θ_t^*(f_k)+ alnN^{l*}/N_k^{l*}$
15. Set $r_t=f_k^*(x^t)$
16. Set $r_t=f_k^*(x^t)$
17. Set $q_{l*}(f_{k*})← q_{l*}(f_{k*})+1/N_{k*}^{l*}[r_t-q_{l*}(fk_*)]$
18. Set $Q_{l*}(f_{k*})← q_{l*}(f_{k*})$-$Bc(T_{k*})$
19. Set $N^{l*}← N^{l*}+1$
20. Set $N^{l*}_{k*}+1$
21. Set N← N+1
22. end for

Algorithm maintains multiple counters and accuracy q_e (f_k) and QoE for every classifier f=(f_1f_k) under differen t V_l. N_k records maintains classifier f_k for round for future classification estimatedQoE is $Q(f_k)$, the algorithm Willrun under the clustering runtime , and then the classifier f_k should select context V_L. to estimate QoE for f_k will maintained unchanged.

4. EXPERIMENTAL RESULTS

We have implemented our algorithm in python evaluated its aspects of its performance. The set of real world malware samples has collected from internet to select the best classifier based on malware context. Three major component of our model includes user agent, runtime malware analysis , and system calculation component. Chrome Extension is used user

agent. K means cluster feature is applied to dynamic analysis of malware and once the classifier is selected , the length of malware analysis and it can be defined accordingly. Thus traditional methods are too costly to be maintained . So to maintained effectiveness , we have applied machine learning on malware data set.Experiment include 3000 dataset among which 1400 were malicious programs labeling of classification are defined by Virus Total Online scanner sample categorised into 1000 samples each . Fig 3.1 shows scatter plot of the context feature of 3000 samples.

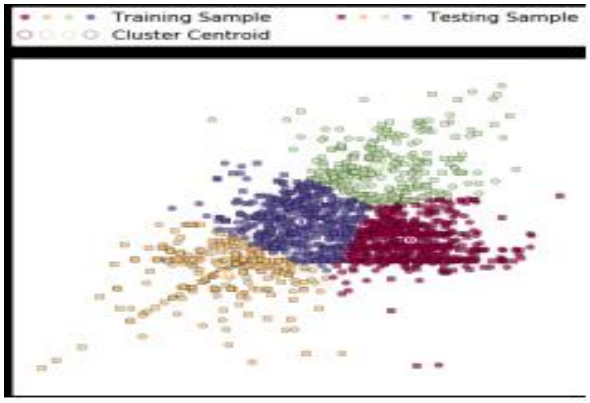

Fig 3.1 Context Clustering

5. CONCLUSION

As the tremendous increase in the past decade , malware threat become a threat in information security. Traditional malware detection method depends on human interfere. So not enough security methods are available for detection of signature. Thus traditional methods are too costly to be maintained . So to maintained effectiveness , we have applied machine learning on malware data set.

6. REFERENCES

[1] Sikorski, Michael, and Andrew Honig. *Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software.* No Starch Press, 2015.

[2] Egele, Manuel, et al. "A survey on automated dynamic malware-analysis techniques and tools." *ACM Computing Surveys (CSUR)* 44.2 (2016): 6.

[3] R. Perdisci, A. Lanzi, and W. Lee, "McBoost: Boosting Scalability in Malware Collection and Analysis using Statistical Classification of Executables," 2011, pp. 301–310.

[4] S. M. Tabish, M. Z. Shafiq, and M. Farooq, "Malware Detection using Statistical Analysis of Byte-Level File Content," CSI-KDD '09 Proceedings of the ACM SIGKDD Workshop on CyberSecurity and Intelligence Informatics, pp. 23–31, 2009.

[5] D. Wagner and P. Soto, "Mimicry Attacks on Host-Based Intrusion Detection Systems," Proceedings of the 9th ACM

[6] Conference on Computer and CommunicationsSecurity, pp. 255–264, 2002.

[7] A. Walenstein and M. Venable, "Exploiting Similarity Between Variants to Defeat Malware," Proceedings of BlackHat Briefings DC 2007, pp. 1–12, 2007.

[8] A. Karnik, S. Goswami, and R. Guha, "Detecting Obfuscated Viruses Using Cosine Similarity Analysis," First Asia International Conference on Modelling & Simulation (AMS'07), pp. 165–170, 2007.

[9] M. Gheorghescu, "An Automated Virus Classification System," Virus Bulletin Conference, pp. 294–300, 2005.

[10] C. LeDoux and A. Lakhotia, "Malware and machine learning," in Intelligent Methods for Cyber Warfare, 2015.

[11] X. Hu, T. Chiueh, and K. G. Shin, "Large-scale Malware Indexing Using Function-Call Graphs," Proceedings of the 16th ACM Conference on Computer and Communications Security, 2009.

[12] D. Maiorca and G. Giacinto, "Looking at the Bag is not Enough to Find the Bomb : An Evasion of Structural Methods for Malicious PDF Files Detection,"

[13] Proceedings of the ASIA CCS'13, pp. 119–129, 2013.N. Srndic and P. Laskov, "Practical Evasion of A Learning-based Classifier: A case study," Proceedings - IEEE Symposium on Security and Privacy, pp. 197–211, 2014.

[14] W. Xu, Y. Qi, and D. Evans, "Automatically evading classifiers: A case study on pdf malware classifiers," NDSS, 2016.

[15] K. Rieck, P. Trinius, C. Willems, and T. Holz, "Automatic Analysis of Malware Behavior using Machine Learning," pp. 1–30, 2011.

[16] U. Bayer, "Large-Scale Dynamic Malware Analysis," PhD Thesis, pp. 1–109, 2009.

[17] U. Bayer, P. M. Comparetti, C. Hlauschek, C. Kruegel, and E. Kirda, "Scalable , Behavior-Based Malware Clustering," NDSS, pp. 51–88, 2009.

[18] Google Safe Browsing, "Google Safe Browsing."

[19] [Online]. Available: https: //safebrowsing.google.com/

[20] W. Xu, Y. Qi, and D. Evans, "Automatically evading classifiers: A case study on pdf malware classifiers," NDSS, 2016.

[21] U. Bayer, "Large-Scale Dynamic Malware

[22] Analysis," PhD Thesis, pp. 1–109, 2009.

[23] 22. U. Bayer, P. M. Comparetti, C.Hlauschek, C.Kruegel, and E. Kirda, "Scalable , Behavior- Based Malware Clustering," NDSS, pp. 51–88, 2009.

[24] P. Trinius, C. Willems, T. Holz, and K.Rieck, "A Malware Instruction Set for Behavior-Based Analysis," Sicherheit Schutz undZuverl¨assigkeit SICHERHEIT, pp. 1–11, 2011.

[25] "Malware Information Sharing Platform,"

[26] http://www.misp-project.org/, 2016, [Online; accessed March, 2016].

[27] "Information Sharing Specifications for Cybersecurity," https://www.us-cert. gov/Information-Sharing-specifications Cybersecurity, 2016, [Online; accessed March, 2016].