An Improved Approach for Detecting Host Based Malware using Genetic Algorithms and Support Feature Vectors

Dalimlata Nag
Department of Computer Science & Engg. Dept.Shankaracharya
Group of Institutions, Bhilai (C.G.), India

Reetika Singh
Asst.Professor, Computer Science & Engg. Dept.
Shankaracharya Group of Institutions, Bhilai (C.G.), India

Abstract – A Malware is a set of instructions that run on your computer and make your system do something according to attacker's intentions. A computer worm is a self-replicating computer program. It propagates through network to send copies of itself to other nodes without any user assistance. Email worm, as per the name spreads through infected messages sent by email. The worm may be in the form of attachment, or may contain links to an infected website; the host is immediately infected when the user opens the attachment, or clicks the link. The proposed work is aimed at suspecting the e-mails which consist of unwanted elements and block them. Genetic Algorithms (GAs) are adaptive heuristic search algorithm which is based on ideas of natural selection and genetic for evolution. The basic concept of algorithm is designed to simulate natural evolution, specifically the principles suggested0020by Charles Darwin of survival of the fittest. Genetic algorithms as combined with other techniques can very useful for detecting malware. Future work covers detecting malwares in other host based applications like exes by analyzing and applying the same concept for assembly code of exes.

Index Terms – Malware, Genetic Algorithm, Email Worm, Virus, Spyware.

1. INTRODUCTION

Malware is term signifying "malicious software." This is programming that is particularly designed to access or harm a computer without the information of the user [13]. There are different sorts of malware including spyware, worms, or any kind of noxious code that invades a PC [14]. By and large, programming is considered malware in light of the purpose of the inventor as opposed to its real peculiarities. Malware creation is on the ascent because of the sheer volume of new sorts made every day and the income that can be made through composed web wrongdoing. Malware was initially made as experiment, however in the end prompted vandalism focused on machines. Today, a lot of malware is made for benefit through constrained promoting (adware), taking touchy data (spyware), and spreading email spam or tyke erotic entertainment (zombie PCs).

1.1. Categories of Malware

There are various types of malware. Some of them are:

Virus: Software that can replicate it and spread to other computers or are programmed to damage a computer by deleting files, or using up memory.

Adware: Software that financially supports another program by displaying ads when connected to the Internet.

Spyware: Software that gathers information in background and transmits it to interested parties. Gathered information includes the list of visited websites, browser and system information, and your computer IP address.

Browser hijacking software - Software that modifies your browser settings creates desktop shortcuts, and displays advertising pop-ups in between. After a browser is infected, it may redirect links to other sites that advertise, or sites that collect Web usage information.

1.2. How Malware propagates

Malware can abuse security gaps in your program as a method for attacking your machine. Now and then sites express that product is needed to view the site, trying to trap clients into clicking "Yes" options installing program onto their machines. An alternate trap is whether you click "No," numerous error windows are displayed. Different locales will let you know that utilizing a declaration makes their site "safe" which is not the situation. Authentication check implies just that the organization that composed the product is the same as the organization whose name shows up on the download brief. Some malware gives no uninstall alternative, and introduces code in surprising and concealed spots (e.g., the Windows registry) or changes the working framework, accordingly making it harder to uproot.

2. RELATED WORK

Author Blaine Alan Nelson in his work "Designing, Implementing, and Analysing a System for Virus Detection", March 19, 2006 stated that In spite of advances in viral detection, the rapid proliferation of novel mass mailing worms continues to pose a daunting threat to network administration. The crux of this problem is the slow dissemination of the up-to-date virus signatures required by traditional systems to effectively halt viral spread.

Authors Mohammad M. Masud, Latifur Khan , and Ehab Al-Shaer in their work "Email Worm Detection Using Naïve Bayes and Support Vector Machine ", Springer-Verlag Berlin Heidelberg 2006 stated that There has been a significant amount of research going on to combat worms. The traditional way of dealing with a known worm is to apply signature based detection.But the problem with this approach is that it involves significant amount of human intervention and it may take long time (from days to weeks) to discover the signature. Since worms can propagate very fast, there should be a much faster way to detect them before any damage is done.

Authors Mohammad Narubordee Sarnsuwan, Naruemon Wattanapongsakorn and Chalermpol Charnsripinyo "Internet Worm Detection and Classification with Data Mining Approaches", November 2008, presently, trend of many malwares focuses on network end-point with diverse behaviors. In this paper, techniques tare presented to detect and classify many types of internet worm at network end-point byusing data mining approaches which are Bayesian network, C4.5 Decision tree and Random forest.

Author Christie Williams in her work titled "Applications of Genetic Algorithms to Malware Detection and Creation", December 16, 2009 stated that Malware, or "malicious" software such as viruses, worms, Trojans, denial-of-service tools, etc., is becoming an increasingly major issue. Malwares are becoming increasingly sophisticated, making it both more damaging and more difficult to detect.

Authors Sadia Noreen, Shafaq Murtaza†, M. Zubair Shafiq‡, Muddassar Farooq in their work titled Evolvable Malware", July 8–12, 2009 proposed an evolvable malware framework. We have proposed an evolvable malware. They validated the notion of evolution in viruses on a well-known virus family, called Bagle.

3. PROBLEM IDENTIFICATION

Today, malware is used by both black hat hackers and governments, to steal personal, financial, or business information. Since the rice of widespread broadband Internet access, malicious software has more frequently been designed for profit. Since 2003, the majority of widespread viruses and worms have been designed to take control of users' computers for illicit purposes. The major problem areas that have been identified are:

- Malware, or "malicious" software such as viruses, worms, Trojans, denial-of-service tools, etc, is becoming an increasingly major issue.
- Current methods of protecting against malware, which are often based on static "signatures" updated after the malware is already "live", cannot adequately protect users. The process of creating and releasing signatures for "known" malware does not have a fast enough response time, which leads to frequent "zero-day" vulnerabilities which users are not protected from.

Rather than rely on signatures produced from known malware, the system instead determines whether a process or file is malware based on characteristics that can be observed in real time, such as behavior or common "malicious" patterns in the executable code.

4. PROPOSED METHODOLOGY

4.1. Genetic Algorithm

Genetic algorithms were formally introduced in the United States in the 1970s by John Holland at University of Michigan. The continuing price/performance improvements of computational systems have made them attractive for some types of optimization. To use a genetic algorithm, you must represent a solution to your problem as a genome (or chromosome). The genetic algorithm then creates a population of solutions and applies genetic operators such as mutation and crossover to evolve the solutions in order to find the best one(s).The three most important aspects of using genetic algorithms are: (1) definition of the objective function, (2) definition and implementation of the genetic representation, and (3) definition and implementation of the genetic operators. Once these three have been defined, the generic genetic algorithm should work fairly well. Basic steps are:

[**Start]**
Generate random population of *n* chromosomes (suitable solutions for the problem)
[Fitness]
Evaluate the fitness *f(x)* of each chromosome *x* in the population
[New population]
Create a new population by repeating following steps until the new population is complete
[Selection]
Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected)

- **[Crossover]**
  With a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.

- **[Mutation]**
  With a mutation probability mutate new offspring at each locus (position in chromosome).
- **[Accepting]**
  Place new offspring in a new population
- **[Replace]**
  Use new generated population for a further run of algorithm
- **[Test]**
  If the end condition is satisfied, **stop**, and return the best solution in current population
- **[Loop]**
  Go to step **2**

4.2. Basic algorithm of proposed work

[1] Generate sample email worm dataset.
[2] Generate initial population as support vector i.e. a feature vector for email dataset.
[3] Select fittest parents according to function criteria.
[4] Apply crossover on selected parents.
[5] Apply mutation and generate the offspring.
[6] Generate candidate support vector for candidate email.
[7] Compare the feature values with the offspring.
[8] If the no of features in candidate vector match with the feature vector of the offspring then classify candidate as worm or else healthy,

4.3. Flow Chart

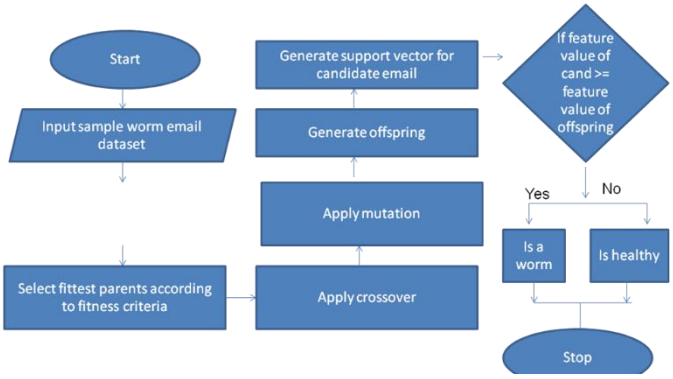

Figure 1: Detecting Email Worms

5. CONCLUSION

Overall it appears that genetic algorithms do have many promising applications to detecting malware, as well as for creating malware that is better able to evade existing detection systems. Genetic algorithms as combined with other techniques can very useful for detecting malware. Future work covers detecting malwares in other host based applications like exes by analyzing and applying the same concept for assembly code of exes.

- The efficiency of algorithm can be further increased by combining the algorithm with more efficient algorithms in near future.
- The detection approach can be further enhanced for other Malwares also.
- The same approach can be used for filtering out malwares in executable but reading and applying algorithm on assembly code of the program.

The best protection from malware continues to be the usual advice: be careful about what email attachments you open, be cautious when surfing and stay away from suspicious websites, and install and maintain an updated, quality antivirus program.

- Security can be increased by applying efficient encryption/decryption algorithms.

REFERENCES

[1] Optimization. In GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation, pages 103–110, New York, NY, USA, 2006. ACM.
[2] S. B. Mehdi, A. K. Tanwani, and M. Farooq. Imad: in-execution malware analysis and detection. In GECCO '09: Proceedings of the 11th Annual conference on Genetic and evolutionary computation, pages 1553–1560, New York, NY, USA, 2009. ACM.
[3] S. Noreen, S. Murtaza, M. Z. Shafiq, and M. Farooq. Evolvable malware. In GECCO '09: Proceedings of the 11th Annual conference on Genetic and evolutionary computation, pages 1569–1576, New York, NY,USA, 2009. ACM.
[4] M. Z. Shafiq, S. M. Tabish, and M. Farooq. On the appropriateness of evolutionary rule learning algorithmsfor malware detection. In GECCO '09: Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference, pages 2609–2616, New York, NY, USA, 2009. ACM.
[5] N. Weaver, V. Paxson, S. Staniford and R.Cunningham, "Taxonomy of computer worms," Proc of the ACM workshop on Rapid malcode, WORM03, 2003, pp. 11-18
[6] S. A. Khayam, H. Radha and D. Loguinov, "Worm Detection at Network Endpoints Using InformationTheoretic Traffic Perturbations", IEEE Inter Conf on Communications (ICC), 2008, pp. 1561-1565.
[7] Symantec Internet Security Threat Report XI – Trends for July – December 07," 2007.
[8] O. Sharma, M. Girolami and J. Sventek, "Detecting Worms Variants Using Machine Learning", Proc of the ACM CoNEXT conference, 2007
[9] C. Smith, A. Matrawy, S. Chow and B. Abdelaziz, "Computer Worms: Architecture, Evasion Strategies, and Detection Mechanisms," J. of Information Assurance and Security, 2009, pp. 69-83
[10] M. Siddiqui, M. C. Wang and J, Lee, "Detecting Internet Worms Using Data Mining Techniques", Cybernetics and Information Technologies, Systems and Applications: CITSA, 2008.
[11] X. Wang, W. Yu, A. Champion, X. Fu and D. Xuan, "Detecting Worms via Mining Dynamic Program Execution", Security and Privacy in Communications Networks and the Workshops 2007, Nice, France, June 24, 2008
[12] Wireless and Secure Networks (WiSNet) Research Lab at the NUST School of Electrical Engineering and Computer Science (SEECS), http://wisnet.seecs.edu.pk/
[13] M. Milton Joe, R.S. Shaji, K. Ashok Kumar, "Prevention of Worm at Router Level for Providing Seamless Communication in Network Environment" International Journal of Engineering and Technology, Vol 5 No 2 Apr-May 2013.
[14] M. Milton Joe, R.S. Shaji, F. Ramesh Dhanaseelan", Detection of m-worm to provide secure computing in social networks", Elixir Comp. Sci. & Engg. 50 (2012) 10363-10365.