A Novel Anti-Obfuscation Model for Detecting Malicious Code

Yuehan Wang, Beijing University of Technology, Beijing, China
Tong Li, Beijing University of Technology, Beijing, China
Yongquan Cai, Beijing University of Technology, Beijing, China
Zhenhu Ning, Beijing University of Technology, Beijing, China
Fei Xue, Beijing Wuzi University, Beijing, China
Di Jiao, National Engineering Laboratory for e-Government Integration and Application, Beijing, China

ABSTRACT

In this article, the authors present a new malicious code detection model. The detection model improves typical n-gram feature extraction algorithms that are easy to be obfuscated. Specifically, the proposed model can dynamically determine obfuscation features and then adjust the selection of meaningful features to improve corresponding machine learning analysis. The experimental results show that the feature database, which is built based on the proposed feature selection and cleaning method, contains a stable number of features and can automatically get rid of obfuscation features. Overall, the proposed detection model has features of long timeliness, high applicability and high accuracy of identification.

KEYWORDS

Anti-Obfuscation, Feature Extraction, Feature Selection, Machine Learning, Malicious Code Detection, Malicious Code Family, N-Gram, Random Forest

1. INTRODUCTION

The malicious code is a kind of software that is intended to damage or disable computers and computer systems, including computer Trojans, blackmail software, spyware, and so on. According to Symantec (2015), more than 44.5 million new pieces of malware created in May 2015. One of the main reasons for this high volume of malware samples is the extensive use of obfuscation and metamorphic techniques by malware developers. So the most of new malicious code can be divided into several families by the original code.

The malicious code detection technologies are usually based on features, which represent the original software code. Thus, same malware families should have the same features (e.g., Wolkowicz & Kešelj (2013) and Preda & Giacobazzi (2005)). By extracting the family features in each malware family, the defense systems can constructs a feature database for detecting variants. However, the obfuscation techniques can help variants to escape the detection by interfering the feature extraction. For example, in the malicious defense system (Lu, Wang, Zhao, Wang, & Su, 2013) which extracting the key string as a feature. Variants escape the detection by equivalently replacing the key string or adding the invalid string. Many scholars (Shafiq, Tabish, Mirza, & Farooq, 2009; Sung, Xu, Chavez, & Mukkamala, 2004; Gaudesi, Marcelli, Sanchez, Squillero, & Tonda, 2016; Tabish, Shafiq, & Farooq, 2013)
2009) have proposed various feature extraction methods to defend against this kind of obfuscation technology. But such extraction methods can also be broken by emerging obfuscation technology. On the other hand, more effectively extraction methods will also lead to excessive computing resources, systems real-time poor and so on.

Machine learning model (Tahan, Rokach, & Shahar, 2012; Narouei, Ahmadi, Giacinto, Takabi, & Sami, 2015; O.W.D.C., 1992) are used to deal with detection malicious code, which have achieved good results. Through the feature database and labels, the model will train a set of classifiers to identify the variants. However, the accuracy of machine learning model depend on the quality of feature database, so that the feature extraction method will determine the accuracy of model. When the extraction method is broken, the obfuscation technologies (Nataraj, Karthikeyan, Jacob, & Manjunath, 2011; Fredrikson, Jha, Christodorescu, Sailer, & Yan, 2010; Svetnik et al., 2003) will make feature database contains a lot of obfuscation features and the accuracy will be seriously influenced. For the machine learning model used in detection malicious code, ensuring the effectiveness of feature database is an essential research task. In particular, due to the rapid growth of malicious code, the timeliness of feature extraction method becomes more and more short. In addition, it becomes increasingly difficult to maintain the security of the system by using the replacement feature extraction method.

In this article, we propose a method to ensure the effectiveness of feature database which cleans the feature database rather than changing the extraction method. The method was guided by the obfuscation features cleaning and feature selection. The final database will be used in the random forests algorithm. The main contributions of this paper are summarized as follows:

1. An algorithm based on multi-sample analysis is proposed to identity obfuscation features dynamically. This method get through analyzing some numbers of sample data in detail and builds a linear regression algorithm. This linear algorithm is used to compute the thresholds of the obfuscation features dynamically for each sample.
2. A feature selection algorithm is proposed to select family feature. The method first normalizes the eigenvector and identify the family feature according to the number of input data set.
3. Achieving the malicious code detection model. The model use random forest algorithm to reduce the effect of obfuscation technologies furtherly and improves the data utilization. The detection of result by the classifier voted.

2. METHODOLOGY

In this paper, the main research content is to clean the feature database based on the machine learning malicious code detection system. The n-gram algorithm is one of the earliest feature extraction algorithms for malware code and the feature has a stronger readable and interpretable that extract by the extraction method. It is impossible to guarantee the feature extraction algorithm never be broken. So this paper choose n-gram feature extraction method to build the feature database. The modern obfuscation technologies make n-gram algorithm is invalid and the feature database is full of obfuscation and noisy feature. So the final feature database was guide by the obfuscation features cleaning method and feature selection method. The two clean method make the feature database has a good anti-interference, and replace the bad features automatically with the training samples increased.

As shown in Figure 1. Firstly, this paper construct the linear regression algorithm to identity the obfuscation features and clean dynamically. Secondly, the normalizing method make the eigenvalues range be uniform so that the feature selection method will not be affected by the range of eigenvalues. The feature selection method will guide the train database to select the family features. Finally, this paper choose random forest algorithm to construct the classifier cluster. For the test set samples, the final result is voted on by the cluster. The overall flow chart of the model is shown in Figure 1:
Measuring the Efficiency of Free and Open Source Software Projects Using Data Envelopment Analysis
www.igi-global.com/chapter/measuring-efficiency-free-open-source/10081?camid=4v1a

Greasemonkey and a Challenge to Notions of Authorship
www.igi-global.com/chapter/greasemonkey-challenge-notions-authorship/21175?camid=4v1a