Digital Investigation 11 (2014) 323–335

Contents lists available at ScienceDirect

Digital Investigation

journal homepage: www.elsevier.com/locate/diin

Malware categorization using dynamic mnemonic frequency analysis with redundancy filtering

BooJoong Kang [a], Kyoung Soo Han [a], Byeongho Kang [a], Eul Gyu Im [b, *]

[a] Department of Computer Software, Hanyang University, Seoul, Republic of Korea
[b] Division of Computer Science & Engineering, Hanyang University, Seoul, Republic of Korea

ARTICLE INFO

Article history:
Received 5 July 2013
Received in revised form 12 June 2014
Accepted 13 June 2014
Available online 1 July 2014

Keywords:
Malware analysis
Dynamic analysis
Malware categorization
Mnemonic frequency
Redundancy filtering

ABSTRACT

The battle between malware developers and security analysts continues, and the number of malware and malware variants keeps increasing every year. Automated malware generation tools and various detection evasion techniques are also developed every year. To catch up with the advance of malware development technologies, malware analysis techniques need to be advanced to help security analysts. In this paper, we propose a malware analysis method to categorize malware using dynamic mnemonic frequencies. We also proposed a redundancy filtering technique to alleviate drawbacks of dynamic analysis. Experimental results show that our proposed method can categorize malware and can reduce storage overheads of dynamic analysis.

Introduction

Malware is short for malicious software which is designed to disrupt or deny operation, to gather information that leads to loss of privacy or exploitation, or to gain unauthorized access to system resources and other abusive behavior (Nash, 2005). Malware analysis and detection are important to reduce damages caused by malware. To defend against malware, many methods have been proposed and these methods mostly focused on malware detection and classification. For malware detection and classification, various features of malware can be used including binary signatures, instructions, control flow graphs, call graphs, behavioral information and so on.

When the number of newly found malware per year was small enough, malware detection and classification were effective with only simple features such as binary signatures of malware. However, malware technologies keep advancing to avoid detection and classification. To keep up with the malware technology improvement, malware detection and classification methods also have advanced in various ways, such as using features that are harder to be avoided or hidden by malware developers. Since we try to defend against moving targets of malware technologies, new analysis techniques should be researched from different aspects of malware to help security analysts.

In this paper, we proposed a malware dynamic analysis method using redundancy filtered mnemonic frequencies to categorize malware. Assembly instructions are low-level instructions which compose a program and assembly instructions consist of a) an opcode which represents an operation to be performed and b) operands which represent data to be processed. Some opcodes perform the same operation with different operand types, so such opcodes are named as the same operator name, called *mnemonic*, such as *MOV*, *PUSH*, *JMP* and so on. We focused on mnemonics instead of opcodes because we want to categorize malware based on statistics of what operations are performed.

Mnemonic frequencies are appearance counts of each mnemonic in an executable file and can be extracted from a sequence of disassembled assembly instructions. However,

* Corresponding author. Tel.: +82 2 2220 4321; fax: +82 2 2281 2381.
E-mail addresses: kang@cs.hanyang.ac.kr (B. Kang), 1hanasun@hanyang.ac.kr (K.S. Han), null@hanyang.ac.kr (B. Kang), imeg@hanyang.ac.kr (E.G. Im).

http://dx.doi.org/10.1016/j.diin.2014.06.003
1742-2876/

it is hard to disassemble some malware if obfuscation techniques are applied, such as packing techniques. To overcome the problems caused by the obfuscation, our method is based on dynamic analysis to get mnemonic frequencies from malware. One of differences between static analysis and dynamic analysis is that some instruction sequences can be executed repeatedly in dynamic analysis because a program includes loops and repeated calls to the same function. This difference can increase the size of execution traces in dynamic analysis methods. To handle this problem, we also proposed a redundancy filtering technique to filter out repeated blocks of instructions.

To evaluate effectiveness of our proposed method, various classification experiments were performed using *WEKA* (Hall et al., 2009). In our experiments, categorization accuracies are about 90% using only 1% of the original execution trace data. Our proposed method can be used to reduce workloads of security analysts as a part of preprocessing in everyday malware analysis by providing classification information of malware.

Our contributions can be summarized as follows:

- ✓ We proposed a malware categorization method using dynamic mnemonic frequencies.
- ✓ We proposed a redundancy filtering technique to remove repeated blocks of instructions from execution traces.
- ✓ We provide experimental results of our proposed method with various classifiers for malware categorization.

The remainder of this paper is organized as follows: Section 2 discusses the related work. Section 3 describes our proposed method, and Section 4 explains the experimental results of our proposed method. Finally, Section 5 concludes the paper and outlines avenues for future work.

Related work

Bilar (2007) showed that different malware has different frequencies of mnemonics. They showed that rare mnemonics are better indicators for malware classification than the others. However, there is no classification experiment shown in the paper.

Rad and Masrom (2010) proposed a malware family classification method based on mnemonic frequencies. They extracted mnemonic frequencies from each function statically and compared the functions using *Minkowski-form distance*. They defined the distance of two malware as the average of the minimum distances between functions. They showed that distances among malware variants in the same malware family are small but distances among variants in different families are big. However, they did not consider all function pairs and they used the minimum distance of some possible function pairs. It may produce false positives because a function may show low distances with many other functions and this function will affect the average of distances.

Santamarta (2006) proposed a polymorphic malware classification method based on mnemonic frequencies. A mnemonic frequency is extracted from the first 150 executed instructions appeared. In their experiments, the known polymorphic engines were identified using the neural pattern recognition algorithm. Using only 150 instructions seems to be acceptable because most of unpacking engines are executed before executing actual or original instructions of malware. However, it is easy to avoid this method by injecting fake instructions at the beginning of executions.

Ye et al. (2010) proposed a malware family classification method using mnemonic frequencies and function-based mnemonic sequences. As a classifier, they used the cluster ensemble of two clustering algorithms, i.e., the *hierarchical clustering algorithm* and the *k-medoid algorithm*. Since they analyzed mnemonic frequencies with static analysis, they used *K32Dasm* to disassemble and unpack malware. To increase the classification accuracy, they applied the term frequency and inverse document frequency (*TF–IDF*) technique (Baeza-Yates and Ribeiro-Neto, 1999) to mnemonic frequencies. In their experiments, they showed that the cluster ensemble and *TF–IDF* can improve the classification accuracy. There are a couple of problems with their approach: 1) malware unpacking may not be possible and 2) the mnemonic sequences can be changed by obfuscation techniques such as instruction substitution, permutation, injection and so on.

Santos et al. (2011) proposed a malware classification method using *n*-gram mnemonic frequencies, where an *n*-gram mnemonic is a sequence of *n*-mnemonics. They used *NewBasic* as a disassembler and only non-packed malware files are used in their experiments because their method is based on static analysis. They showed that the *n*-gram mnemonic frequencies are good attributes for malware classification using several different classifiers such as *k-nearest-neighbor algorithm*, *decision trees*, *support vector machines* and *Bayesian networks*. However, the *n*-gram approach is an expensive algorithm because its complexity increases exponentially with the increase of *n*.

The above mnemonic frequency analysis methods have a common limitation. They extracted mnemonic frequencies by static analysis. To hinder static analysis, most of malware take advantage of using packing techniques. Unpackers can be used to overcome this problem but unpacking may not be possible in some cases.

O'Kane et al. (2013) presented a malware detection analysis using mnemonic frequencies and support vector machine (*SVM*) (Hsu et al., 2010) based on dynamic analysis. They showed that mnemonic frequencies can be used to distinguish between malware and benign software.

Dynamic mnemonic frequency analysis with redundancy filtering

Dynamic mnemonic frequency

Opcodes specify operations of instructions to be performed and those operations can be expressed in character strings, called *mnemonics*, such as *MOV*, *PUSH*, *JMP* and so on (Intel, 2014a). Opcode frequencies and mnemonic frequencies are similar but not identical, as shown in Fig. 1. The aggregate of all opcode frequencies equals to the