Metamorphic Malware Detection using Heuristic Signature

Milan Rajpara[1] Girish Khilari[2]
[1] P.G. Student [2] Senior Consultant, CDAC
[1] Computer Engineering Department
[1]Gujarat Technological University PG School, Ahmedabad (GUJ), India.
[2] CDAC,Pune, Maharashtra, India.

Abstract— Present day malicious programs are comes with dynamic packing capability which helps them to evade detection from traditional anti-malware scanner who works on OpCode pattern matching technique. The metamorphic engine resides in malware, changes the structure of malware, which changes OpCode hence the previous signature will not work for the new variant. Here we are using a bioinformatics technique of signature alignment, to generate a heuristic signature, based on the previous database of malware. We made a signature set of malicious (single MSA signature) and benign file (group signature) and computed a threshold to detect malwares. Proposed method calculates to predict the signatures, of the metamorphic malware variants. This method showed a good result with high true positive and low false positive ratio.

Keywords: Malware Signature; Heuristic Detection; Bioinformatics; Multiple Signature Alignment.

I. INTRODUCTION

A malicious program is a piece of code, specifically developed to harm the existing computer system [1] by use of any vulnerability in it. In past, it was simple in design and, easy to detect via single signature. But nowadays malware becomes far more advance with use of different concealment techniques. Now malware writers are using metamorphic technique to hide the malicious code.

The metamorphic malware uses metamorphic engine, which changes the structure and code of malware without affecting its malicious function, on each replication and makes a different variants of it. The metamorphic engine try to use very small space in file to evade detection. This engine changes the instruction, inserts the junk code, changes variable name, and try many other tactics to change the structure of program. And tries to look similar to benign file. Even though the changes in the file, there are some handwritten assembly code which cannot be changed in much amount or else it will lose its functionality. So this gives a chance to catch this malware. And also the metamorphic engine uses an algorithm so it makes a minor pattern in making of new mutants of malware. Here we try to make a signature using a sequence alignment to detect a malware mutant.

In human DNA / protein receives the functions and anatomical structure from parents. We can compare this transformation with malware mutants, by comparing DNA sequence transformation with OpCode in a computer code. In bioinformatics there is a technique of sequence alignment, used in detection of similar DNA patterns. Here we extended the work of the author proposed in [2], we are using sequence alignment technique to make single and group signature of malicious and benign file respectively instead of using only malware signature. By calculating a threshold value we can detect unknown malwares. The group signature derives from the benign programs are going reduce the false positive ratio by marking the benign file as benign.

II. RELATED WORK

Many researchers are working in this field and explored many techniques. There are static and dynamic, signature, behavior and heuristic based detection techniques. G. Jacob et al [3] proposed their technique with a behavioral based detection and Ahmed et al [4] showed a behavioral signature to catch malware. As in a heuristic based author [5] and [6] proposed their solutions with use of API (Application Programming Interface) and CFG (Control Flow Graph) respectively; they are good at detection with good true positive ratio but also gives a higher false positive too.

I. Santos et al [7], [8] and N. Ranwal [9] proposed a solution by machine learning techniques, HMM (Hidden Markov Models) and vector machine respectively. They gives a good result but it's difficult to make and also takes much time to make a data set for it. And metamorphic malware are able to bypass this HMM scheme. To overcome this, the proposed work of author in [10] shows multilayer HMM, but it take much larger amount of time to carry out this work.

Authors in [11] proposed a "phylogeny" model and [2] proposed a MSA, group and probabilistic signature of malware variants, this gives good result with almost zero false positive by group signature but also drops the detection rate. Single MSA is able to achieve more than 09% of detection rate but is also give a higher number of false positive too. This results are from heuristic and signature based combination. Here we try to reduce the false positive by using a group signature of benign file along with a single MSA signature.

III. SEQUENCE ALIGNMENT METHODS

In biological field to get the similarity between two or more DNA / protein sequences different alignment methods are used. This alignment methods tries to discover the highly similar parts in this sequences. This calculation helps to find out inheritance and extract other useful data in bioinformatics domain.

Computer programs are also a sequence of an OpCode. Conventional antivirus scanner checks the OpCode sequence as a signature. In metamorphic malware the metamorphic engine makes a malware mutant with changes in it, that changes the OpCode string, but they left few similarities in new replica. So different variants from the same parent show some similarities. If we compare the

OpCode string with DNA sequences of bioinformatics, we can use this sequence alignment to find out malware variants. By use of the sequence alignment techniques we can conclude the passing sequence from the same malware variants. Authors in [2] used a global, local and multiple sequence alignment, we followed their proposed calculation to generate MSA here.

Global alignment are useful when the strings are nearly similar in length, it means it try to compare the whole string. Needleman–Wunsch [12] is widely used method in this calculation, it works on two strings. Local alignment is useful in comparing smaller set of strings. Multiple sequence alignment is done by applying the local and global alignment repeatedly on the set of strings. Distance calculation is done to generate polygenetic tree for final MSA signature generation.

IV. IMPLEMENTATION METHODOLOGY

As shown in the figure 1, we used malware and benign portable executable to generate dataset for the signature, then; unpack, making set, alignment, generating MSA, (single, group) signature, compute threshold in inline order.

To generate dataset we collected 715 malwares from VX Heaven [13] and also constructed using malware creators (ex. G2, NGVCK, VLC32, PSMPC ...), and 92 benign files from System32 (here we used only one family for experimental purpose) of windows OS. To classify different malware families we used 4 different anti-virus scanners on created data set. And also manually labeled with some known malwares.

Metamorphic malware are generally packed (means encrypted), to hide themselves from malware scanners. To get the actual OpCode pattern we required portable executable unpacker. Generally these malwares are packed by known packers so we can used openly available unpacker like VMPacker and GNUPacker. These packers are publicly available. Else you can use Ether unpack (http://ether.gtisc.gatech.edu/) (only available for research work). The malware which is packed in multilayer are difficult to unpack.

Now the unpacked PE is opened in IDA Pro [14] (it is a disassembler), to get the OpCode sequence. The obtained code is in assembly language we require to parse that code to retrieve the OpCode. Now this OpCode sequence is aligned in pair. Now the local and global alignment method is used on the data. Here the data is in two parts which is benign and malicious, the procedure is happening on both files but separately and also family wise.

A. Signature generation

1) Single MSA signature of melisious file

Multiple sequence alignment is calculated using progressive alignment [15] technique. The most common string is taken as a base string and rest of the strings are aligned from more similar to less similar and phylogenetic tree is constructed. The single MSA signature is the OpCode which is in a maximum occurrence in a row.

Here table 1 shows the five OpCode sequences of malicious files from different families. The last column is a MSA signature, that string is made from the OpCode which occurred most in the row.

M_1	M_2	M_3	M_4	M_5	MSA Sign
mov	mov	-	mov	-	mov
push	push	lea	-	xor	
call	call	call	call	call	call
jump	jump	jz	jump	Jz	jump
-	mov	mov	mov	push	mov

Table. 1: MSA Signature

2) Group signature of Benign file

Because are using a single benign group and also benign file do not generate its variant like malicious files do, we don't require to do more calculation for it. The phylogenetic tree for group signature of benign files can be generated through iterative method. This method works as its name and improves overall alignment score.

S_1	S_2	S_3	S_4	S_5	Group Sign
mov	mov	-	mov	-	mov
push	push	lea	-	xor	push | lea | xor
call	call	call	call	call	call
jump	jump	jz	jump	Jz	jump | jz
-	mov	mov	mov	push	mov | push

Table. 2: Group Signature

This signature is helpful to detect the false positive files and again label them to benign one. It lists the most common signature of benign files which nearly all benign files contains.

B. Calculation and thresold value generation

After creating MSA of a malicious file and group signature of a benign file we are going to use this to calculate the threshold value. Higher the signature match of single MSA found in file the toxic level will be that high. The calculation will be done for both the benign and malicious file the max and min value will be calculated and the threshold T will be:

$T = [B_{max} + M_{min}] / 2$

Where, B_{max}, M_{max} and B_{min}, M_{min} are maximum and minimum score of benign and malware files.

The file is first scan with a malicious signature set (which is single MSA signature) and the more OpCode strings matched with a file, the toxic value will be increased, then the file will be scanned with a benign signature set (which is group signature), if match will found in file it will decrease the toxic level for that file.

Match in single signature → raise toxic level

Match in group signature → fall in toxic level

If the final value of toxic level is above the threshold value it will be declared as malicious otherwise benign file.

V. RESULT AND ANALYSIS

From the beginning we separated training and test data set. For training we used 542 files with 78 benign file and 464 malwares. And for test set we used 372 files with 312 malicious and 60 benign files. The threshold value is

calculated from a training set. The test set also contains some unknown files.

It was observed that with the use of proposed method we are able to achieve 86% of detection rate with only 2% of a false positive ratio. The false positive ratio is down because of the group signature of a benign file, reduces the toxic rate for a benign file below the threshold level.

If the single MSA signature is only calculated then it can give higher detection rate but it also gives a higher rate of false positive too. To decrees this false positive we require to use group signature of benign file. It reduces the toxic value of a benign file if matching signature found.

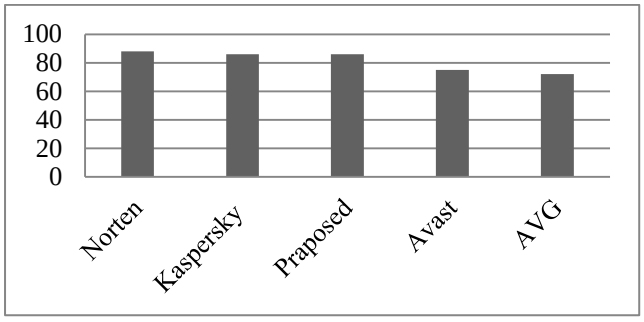

Fig. 1: Detection rate comparison of proposed method with other anti-viruses

Fig shows the result comparison of a proposed method with a four other antivirus. The result of this method shows that more improved data set will able give more fruitful results.

More similarity of a malware with a benign file decreases the detection rate.

VI. Conclusion

Our experiment with limited set of malicious and benign files gives good result with higher detection and low false positive rate. Use of group signature, extracted from benign file, along with a single MSA signature of malicious file helps to detect unknown malwares. This result is from the data set of the specific set of malware, with a bigger data set of malicious and benign files, we may get more profitable results.

References

[1] J. Aycock. "Computer Viruses and Malware". Springer, 2006
[2] Vinod P., V. Laxmi, M. S. Gaur, G. Chauhan, "MOMENTUM: MetamOrphic Malware Exploration Techniques Using MSA signatures", International Conference on Innovations in Information Technology (IIT), 2012
[3] G. Jacob, H. Debar, and E. Filiol, "Behavioral detection of malware: from a survey towards an established taxonomy," Journal in Computer Virology, pp. 251–266, 2008.
[4] A. Ahmed, E . Elhadi, M. A. Maarof and A. H. Osman, "Malware Detection Based on Hybrid Signature Behaviour Application Programming Interface Call Graph Information Assurance and Security Research Group." Journal, A., Sciences, A., & Publications, S., Faculty of Computer Science and Information Systems, 9(3), 283–288,2012.
[5] J. Lee, K. Jeong, and H. Lee, "Detecting metamorphic malwares using code graphs" In Proceedings of the ACM Symposium on Applied Computing, ser. New York, NY, USA: ACM, pp. 1970-1977, 2010.
[6] Z. Zhao, "A virus detection scheme based on features of Control Flow Graph." 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), pages 943- 947, 2011.
[7] I. Santos, F. Brezo, B. Sanz, C. Laorden, P.G. Bringas, "Using opcode sequences in single-class learning to detect unknown malware", IET Information Security, vol. 5, no. 4, p. 220, 2011.
[8] I. Santos, X. U. Pedrero, B. Sanz, C. Laorden, P. G. Bringas, "Collective Classification for Packed Executable Identification", CEAS, Perth, Australia, ACM, 2011
[9] Neha Runwal, Richard M. Low, Mark Stampz, "Opcode Graph Similarity and Metamorphic Detection", Journal in Computer Virology, vol. 8, no. 1–2, pp. 37–52, Apr. 2012
[10] Thomas H. Austin, Eric Filiol, S´ebastien Josse, and Mark Stamp, "Exploring Hidden Markov Models for Virus Analysis: A Semantic Approach", 46th Hawaii International Conference on System Sciences, 2013
[11] Md.Enamul Karim, Andrew Walenstein, and Arun Lakhotia, "Malware Phylogeny Generation using Permutations of Code", Journal in Computer Virology, 1(1–2):13–23, 2005.
[12] Sagle B. Needleman and Christian D. Wunsch. A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins. pages 443–453, 1970.
[13] VX Heavens: www.vxheaven.org (for malware samples)
[14] The IDA Pro Disassembler. http://www.datarescue.com/
[15] ClustalW2 - Multiple Sequence Alignment http://www.ebi.ac.uk/Tools/msa/clustalw2.

