I.J. Wireless and Microwave Technologies, **2020, 5, 35-40**
Published Online October 2020 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2020.05.04

Application of Artificial Neural Networks for Detecting Malicious Embedded Codes in Word Processing Documents

Sisay Tumsa
Arba Minch University, Arba Minch Institute of Technology
Faculty of Computing and Software Engineering, Arba Minch, Ethiopia
Email: sisay.tumsa@amu.edu.et

Received: 06 May 2020; Accepted: 20 July 2020; Published: 08 October 2020

Abstract: Artificial Neural Networks have been widely used in security and privacy domains for alleviating the issues of malicious attacks. Several embedded codes like Visual Basic for Application Macros are reasonably powerful scripts that can help to automate iterative processes in word processing documents. It has been observed that, unethical hackers exploit these embedded scripts for their malicious intents. Since most of the Microsoft Word users are unaware of such malicious attacks because they are layman end users and mistakenly considers less suspicious contents. And therefore, these hackers prefer to use Microsoft Office documents as most vulnerable items for or Attack vectors. As a general approach, non-executable files are assumed to be less vulnerable than executable files. This implies that these document files could provide an easy and convenient exploitable pathway that can allow hackers to execute their intended malicious actions on the victim's machine. This research paper presents an automatic detection of malicious embedded codes in general and Microsoft Office documents as a specific case for experimental analysis. This research paper considered only malicious behavior of the embedded codes i.e. checks the status of inclusion or exclusion of the executable code. The malicious datasets are developed to create a knowledgebase where documents are pre-processed. Thereafter the data sets are disassembled using reverse engineering and then malicious features are extracted from the documents. In this research paper, nineteen different malicious keys were extracted. Later, feature reduction technique was applied. Based upon actions; these malicious keys were reduced to eight behaviors. Finally, a machine is trained using artificial neural network with eight input features; extracted from individual disassembled scripts. Afterwards, output nodes that represent malicious or benign behavior classify the existence of attack i.e. exists or does not exists. Based on the training model, a total of seven hundred ninety-two samples of documents were tested. Finally, the research has achieved an average accuracy of 92.2% in the identification of maliciousness of embedded codes in Microsoft Office documents as a case. This result shows that the proposed system has high accuracy in detecting malicious Embedded in word processing documents.

Index Terms: Non-executable; Malicious; behavior; suspicious; knowledgebase.

1. Introduction

An office document of a Microsoft product family is being used in documentation work, and macro functionality is provided to all of the Microsoft product families for user's convenience. The most computer users are most familiar with Microsoft office package tools such as MS word, MS excel, MS power point and MS Access etc. As a general functionality the Microsoft office documents use macro scripts to automate repetitive tasks. Today the recent nature of attack vector has also changed from system level to application level. It has been observed in several studies that the application level attack uses mostly used application software's such as PDF files and Microsoft office Documents. Several recent studies revealed that attackers embed the malicious code in this application software's.

As a matter of facts, the Microsoft office products have their own default security mechanism that prevents the executions of macros by themselves up to a certain level. But on several cases, attackers can forcefully bypass those security mechanisms. Nowadays attackers have changed their attack vector from operating system level to the application level. Particularly, they concentrate their efforts on finding vulnerabilities in application software's such as Microsoft office documents and Portable Document Format (PDF) files. This is because of its complex data structure, which allows the embedding of code, such as JavaScript and Visual Basic for Application (VBA) macro and provides different kinds of Application Programming Interfaces (APIs) to control the way documents are displayed. These complex data structures and rich functionality make such applications prone to vulnerabilities [1]. According to the report [2] most popular file types for targeted attacks were Portable Document Format (PDF) and Microsoft office files. In addition, the report of Kaspersky labs stated that, in 2013, 91% of organizations were hit by the cyber-attacks and 9% were the victim of targeted attacks. Since most of the official documents in government and private organizations having confidential information are stored in Microsoft Office Documents. And also, the most user are unaware of such kinds of attacks. Today Microsoft Office Documents have become a popular avenue for exploitation of such kinds of attacks. In order to overcome such challenges, the installation of antivirus software is not sufficiently effective in capturing embedded executable. This encouraged this research paper to develop an additional detection mechanism that can enhance the protection of the aforementioned confidential and precious documents [14].

Macro viruses make up the majority of mobile code attacks in the world. Macro viruses account for over half the infections reported each month. The U.S Department of Energy, which maintains the virus response team for the government, claims macro viruses represent 85% of their tract infections [3]. In addition a study [4]exposes the weakness identified both in Microsoft Office and OpenOffice documents and how one can bypass those security setting and execute embedded scripts. This paper covers three main aspects; including macro security level, trusted locations and digital signature of document. The other paper focuses on only detecting embedded malicious executable code an apparatus includes: an office document extension name searching module for, when the office document is opened, checking whether or not the corresponding office document has an office document extension name; a macro detecting module for detecting whether or not the office document having the extension name has a macro function; and an execution code checking/parsing module for checking whether or not the office document having the macro function has an execution code, and checking whether or not the execution code is executable [5,6].

The structure of Microsoft office 2007 and higher versions have the security features and techniques to bypass this security mechanism discussed [7,8,9]. Discusses structure of Microsoft office 2007, techniques that reveal how data are concealed and how to bypass the security mechanism set to Microsoft Office files. This clearly shows that there are different advanced techniques that allow attackers to use Microsoft office files as a weapon to attack organizations and individuals.

Thus, the prime goal set for this research is to identify malicious behaviors of Macro scripts and propose a system that automatically detect malicious behavior of macros in Microsoft office documents.

2. Relate Work

Jonathan Dechaux, Eric Filiol and Jean-Paul Fizaine in their paper “Office Documents: New Weapons of Cyber warfare” explores different weakness identified both in Microsoft Office and OpenOffice to bypass those security setting and execute automatically malicious macros. The paper covers three main aspects: macro security level, trusted locations and digital signature of documents [5].

In macro security level the paper discloses the vulnerability of macro security and how to bypass this security level simply by changing the values set under the registry. The paper presents the different security issues with respect to macros. By default, in office security policy the macros are not allowed to be run unless they are in a “trusted location”. However, it is possible to modify this default security setting according to the need and wish of every user. Macro security level is handled in the windows registry base. Indeed, the registry key involved in the security is located in the HKEY_CURRENT_USER section of the registry base.

When installing the Microsoft Office suite, various trusted locations are set up, as defined in the Office security policy, any macro in those trusted locations will be executed by default, and whatever may be the level of security chosen by the user. Some of these trusted locations offer the possibility that the subfolders are also considered and managed as trusted ones. Trusted locations are managed at the registry base level, in the registry section.[17] HKEY_CURRENT_USER

Finally, in digital signature of document the paper presents the vulnerability of documents and how one can bypass the security under digital signature. Creating and adding digital signatures in Microsoft Office 2007are extremely easy. The user has just to click on the “office” button and then choose the prepare option and then the “Add a signature” one. Whenever no certificate is present, Microsoft Office asks the user to create one and to select a certificate type Microsoft Office we mainly use the RegOpenKeyEx, RegSetValueEx, RegCreateKeyEx and RegCloseKey function to manipulate the registry key in a suitable way for our attacks [16, 13].

This paper did not provide absolute solution for the mentioned vulnerability instead it discloses the vulnerability to the attackers.

Another analysis made by “Philippe Lagadec” entitled with” OpenDocument and Open XML security (OpenOffice.org and MS Office 2007)” focused on security issues linked to native features of the file formats and applications. In this analysis most interesting features for security concern, such as VBA Macros, are not covered in the open XML specifications. Microsoft and ECMA consider these features as proprietary technologies that are outside the scope of the open XML standards [6].

3. Proposed System Architecture

Today, identification of an object and classifying it into its appropriate classes become a wide area of research. In the architecture proposed by this research paper (i.e. figure 1) the identification of an object passes through a series of steps/procedures that is proposed to be applied to differentiated items, in which each new item must be categorized in to one of a predefined class based on the basis of observed attribute or features. This section discusses the design and architecture of the proposed system for the detection of malicious macros in Microsoft office documents.

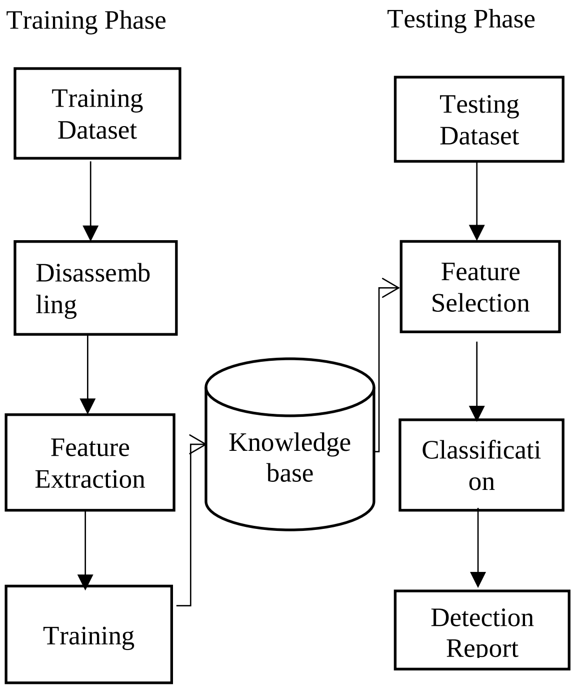

Fig 1. Architecture of the proposed system

Most frameworks for the detection of malware bearing documents are almost the same. Fig 1 shows the proposed system architecture for detection of malicious macros in Microsoft office documents and each component will be discussed hereafter.

4. Experiment

In this research, malicious features of Macro Scripts were identified and a machine is trained by their behavior to detect later those behaviors later. A knowledgebase was created as a result of the training process. The knowledgebase is later used during the identification of unknown samples of the given dataset. Fig 1 shows the proposed system architecture.

The research expected to have

A. Feature Exraction

In this research study, a total of nineteen suspicious keys were extracted from these eight features selected by a method called feature reduction. The selected eight features are discussed as follows-

Auto-execute: suspicious keys that were included in this behavior executes when a new excel work book or an office document is opened.

Bind malicious action with keys: this class of suspicious behavior binds an action with frequently used keys.

Connect to remote server: this class of suspicious behaviors connects to remote server.

Create object: May create an Object Link Embedded (OLE) objects.

Manipulate registry value: this class of suspicious behavior manipulates the registry value so that the default settings may be changed forcefully.

Obfuscation: this class of suspicious behavior attempt to obfuscate strings.

Open port: this class of suspicious behavior open windows port.

Run executable file: this class of suspicious behavior may run an executable file or a system command.

B. Maintaining the Integrity of the Specifications

First features were extracted from the collected datasets and the malicious features were defined. The system was trained with the extracted features and knowledgebase was created by the system through which the system would check the features for the next level i.e. for the testing phase. Since labelled data were used for the system, a supervised machine learning method were used to identify the class of the document. In this paper, an Artificial Neural Network with feed forward multilayer perceptron architecture was used for training because of its favorable properties that make it an excellent choice for object classification.

In feed forward multi-layer perceptron architecture, the neural networks have distinct input, output and hidden layers where the output from one layer of neurons feed forward into the next layer of neurons. There are never any backward connections, and connections never skip a layer. Typically, the layers were fully connected, meaning that all units at one layer are connected with all units at the next layer. A feed forward multilayer neural network consists of a layer of input units, one or more of hidden units, and one output layer of units.

For this experiment, the eight measures of central tendency and dispersions representing malicious features of macros were used as input values (nodes) and two were used as output values (nodes), which leads to have two nodes (benign and malicious) in the output layer. The number of hidden layers is determined to be four. Using this the system classifies the given input datasets as it shows in table 1.

Table 1. Summary of the identification

Target Class Output Class	1	2
1	51	8
2	0	43
Total	51	51
Percent corrected	100%	92.2%

The summary result of the identification showed that from the total test set of 102 samples, 94 (92.2%) were identified correctly and 8 (7.8%) were not identified in its correct classification. Numbers 1 and 2 represents benign and malicious classification

The training process finally generates a knowledgebase which contain the complex relationship between various feature values malicious data. The knowledgebase become the primary input for the decision-making process in this research study.

5. Dataset Identification

Identification of dataset is made by making use of the knowledgebase created during the training phase. The procedure used in this phase was similar to that of the training phase except that the dataset was not labelled. Unknown dataset samples pass through pre-processing processes. Then, the features discussed above were computed where the system matches against the knowledgebase to predict the maliciousness of the dataset.

6. Experiment

A. Dataset Collection

A total of 792 data are collected from the VXheaven machine learning repository. For this experiment eight features were selected that identified the maliciousness behavior of Visual Basic for Application (VBA) macro scripts.

B. Training

Identification of an object using a machine learning approach has two basic phases: training and testing. In the training phase, data is repeatedly presented to the pattern recognizer, while weights are updated to obtain a desired response. Thus, we design the identifier by partitioning the dataset into training, validation and testing dataset. From the total dataset 70% was used for training, 15% for validation and the remaining 15% was used for testing. Since the expected output is a sequence of binary digits, a sigmoid transfer function was used to output 1 in the correct class of the output vector and to fill the rest of the output vector with 0.

During the training, the connection weight of the neural network was initialized with some random values. The training samples were input to the neural network in random order and the connection weight were modified according to the error. This process was repeated until the mean squared error (MSE) fell below a predefined tolerance level or the maximum number of iterations is achieved. The validation set was used for improving generalization of the results. During the training process the error in the validation set is monitored. The validation error normally decreased during the initial phase of training, as does the training set error. However, when the network begins to over-fit the data, the error on the validation set typically begins to rise. When the validation error increases for a specified number of iterations, the training is stopped, and the weights and biases at the minimum of the validation performance as mentioned in the figure-II was measured at 16 iteration and the validation error was 0.042025.

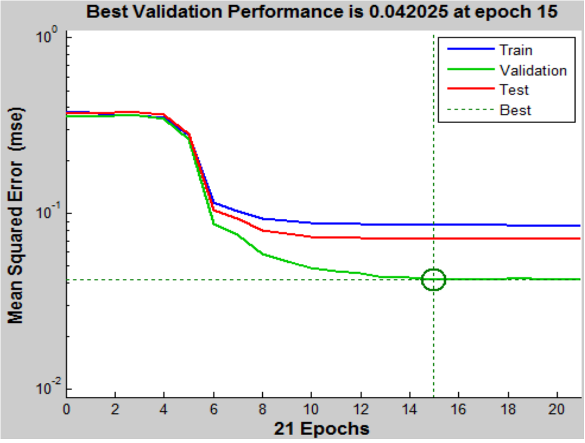

Fig. 2. Best validation performance

C. Test Result

From the collected malicious dataset, 24% of the total which was not included in the training set was used to test the performance of the system. Among this 92.2% of documents are correctly classified and 7.8% of misclassified by the machine after training. From the result it was observed that the proposed system architecture performance found very high.

From the obtained result we can conclude that the developed architecture which performs sound on classifying new documents to benign or malicious based on the trained behaviors.

7. Conclusion and Future Works

In this research paper, the experimental analysis process has extracted features that uniquely identifies malicious behavior of macros in Microsoft office documents. This research is a unique and new knowledge contribution to the researchers and malware analysts. Antivirus companies may be the exclusive beneficiary as a new dimension for detecting malicious actions in Microsoft documents in offline and real-time modes. This can be done by selected features to be incorporated to their database signatures.

In addition this research uses machine learning approach inorder to train the machine to identify malicious behavior of macros in word processing documents.

As an extension and recommendation; better results may be achieved by using hybrid approach for both static and dynamic machine learning approach and changing the learning algorithms.

References

[1] G. A. M. O. I. a. M. O. E. Mohamed Ahmed Mohamed, "A Novel Method to Protect Content of Microsoft Word," International Journal of Computer Theory and Engineering, vol. 7, no. 4, pp. 292-296, 2015.
[2] Parliament, "Inquiry into Cyber Crime," 2018.
[3] "Vernalabiity Assessment," Carnegie Mellon University, 2010. [Online]. Available: https://www.cert.org/historical/advisories/CA-1999-04.cfm.
[4] K. M. H. &. H. I. H. Jassam. T. Sarsoh, "An Effective Method for Hidding Data in Microsoft Word," Global Journal of Computer Science and Technology.
[5] E. F. J.-P. F. onathan Dechaux, "Office Documents: New Weapons of Cyberwarfare".
[6] P. Lagadec, "OpenDocument and Open XML security (OpenOffice.organd MS Office 2007)".
[7] D. R. M. A. a. R. M. Dr. Maad Kamal Al-Anni, "Text Steganography in Font color of MS Excel Sheet," 2018.
[8] M. F. S.Panchal, "Review on Methods of Selecting Number of Hidden Nodes in Artificial Neural Network,," International Journal of Computer Science and Mobile Computing, pp. 455-464, 2014.
[9] W.Bhaya, "Supporting Macro Antivirus Programs By Designing Undetected Virus," 2013.
[10] J. Rollins, "The Stuxnet Computer Worm: Harbinger of an Emerging Warfare Capability," 2010.
[11] H.Flake, "Structural comparison of executable objects," in IN Proceeding of the IEEE conference on Detection of Intrusions and Malware and Vulnerability Assessment, 2004.
[12] K. M. Krahl, "Using Microsoft Word to Hide Data," 2017.

[13] K. E. a. M. F. mmar Odeh, "Stegnography in Text by Using MS Word Symbols," in COnference of the American Society for Engineering Education, 2014.

[14] P. S. Narpat Singh Shekhawat, "Cloud Computing Security through Cryptography for Banking Sector," Proceedings of the 5th National Conference; INDIACom, 2011.

[15] H. S. S. DP Sharma, "Hybrid cloud computing in e-governance: Related security risks and solutions," Research Journal of Information Technology, vol. 4, no. 1, pp. 1-6, 10 3 2012.

[16] R. K. S. A. A. J. Durga Prasad Sharma, "Convergence of Intranetware in Project Management for Effective Enterprise Management," Journal of Global Information Technology (JGIT)-USA, vol. 4, no. 2, pp. 65-85, 2008.

[17] K. Fred B. Schneider, "Language-Based Security for Malicious Mobile Code," vol. 5, 2018.

Authors' Profiles

Sisay Tumsa Lecturer, Faculty of Computing and Software Engineering, AMIT, AMU, Ethiopia . He has Studied BSC in Information Science (Jimma University) and MSc in Computer Science (Arba Minch University). Presently working as a Lecturer in the Faculty of Computing and Software Engineering, Arba Minch University, Ethiopia.

How to cite this paper: Sisay Tumsa, " Application of Artificial Neural Networks for Detecting Malicious Embedded Codes in Word Processing Documents", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.10, No.5, pp. 35-40, 2020.DOI: 10.5815/ijwmt.2020.05.04