International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 4 Issue 6 ǁ June. 2016 ǁ PP. 30-36

Detection Malware Web Services Using Data Mining Based on WSDL File

Mohammed Zoroub
Information Technology Department, Islamic University of Gaza – Palestine

Abstract: In these days reliance on web services has become widely and very important for daily uses to many companies , such as sending , receiving and saving a sensitive data, unfortunately, this popularity uses of web services, has become a target for developers of malware to spy and steal sensitive information. There are Several researches have proposed a new approaches based on Penetration Testing emulate attacks on web services, but these limited to low coverage of existing vulnerabilities and the high percentage of false positives.
In our paper we propose a new framework for detection malware web services using datamining and our results very promised with high accuracy up to 88% to detection the malware web services **.**

Keywords Machin learning , detection malware, WSDL file

I. INTRODUCTION

In recent years, the widely of web services remarkably increased particularly in business transactions and are meant to be used business to business (B2B) model, Web-service is mainly based on Extensible Markup Language (XML) and Hyper Text Transfer Protocol (HTTP). Communication protocol for web services is Simple Object Access Protocol (SOAP). SOAP messages are transported using HTTP, SMTP etc. Web services are described by Web Service Description Language (WSDL). We use Universal Description, Discovery and Integration (UDDI) to locate and access a service via service metadata. , the popularity used to store and process highly sensitive information,

Unfortunately this platform has become an important goal for developers of malware.

In general Malware, is one of the major threats on the privacy of sensitive data and the availability of critical services. A common feature of malware is that they are specifically designed to damage, disrupt, steal, or in general inflict some other bad or illegitimate actions. Malware can infect any computing machine running user programs (i.e. applications) [1]

Thera are many types of malware web services that hacking web services resources like information clients , servers and database by attackers through vulnerabilities such as :-

(1) Denial of Service Attacks also called Dos that causes runs out of resources on server hosting the service and cannot respond to legitimate consumers this will lead to unavailable services .

Injection Attacks

By using an SQL injection attack a web-service enabled database application can be targeted to reveal unauthorized information. In XML injection attacker can insert XML tags in such a way that cannot be detected by parser and may lead to undesired effect, e.g. parameter overriding., This may lead the attacker to insert a script which can harm system operation.

(3) WSDL Spoofing Maliciously changing the content of the WSDL file. This attack is also known as WSDL Parameter Tampering. That focused in our study

Some efforts has been well studied and proposed approaches to detection malware application in Windows from executable file [2, 3] using datamining ,However these efforts not extended to web services even with the increasing popularity of web services application development.

On the other hand, Many researches have contributed to the detection of malware in web services based on penetration testing and Fault injection[4], However these efforts limited to low coverage of existing vulnerabilities and the high percentage of false positives.

.

In our approach we proposed a new framework to detection malware web services using datamining that include three phases as follow :-

1- Collection malware web services that contain the manipulation WSDL file then extraction features .

2- Implementation behavioral-based technique to determine what is main specific behaves for fake WSDL file .

3- Learning a machine the most frequently of malware web services then apply classification method to determine benign or malware W.S.

This paper is organized as follows: in next section we explain some related works. Section three explains the Our methodology, and. Dictation the results approach is covered in section four, while the last section contains the conclusion of the paper.

II. RELATED WORKS

Data mining is application from machine learning that helps many researchers to detection malware file in multi platforms such as windows[2] to detection the malware executable file , and several publisher apply data mining method on android O.S platform[5-8] to determine the malware android application, However these approaches using datamining techniques and proposed an efficient way to detection the malware, but not extended to web service application, So in our approach we proposed a new framework to detection malware web services .

On the other hand , several efforts proposed many approaches to detection malware web services based on penetration testing and fault injection[4, 9, 10] that called emulator attacker such as XXS attack , spoofing and Denial of service to determine how much vulnerabilities in web services and using external tools to show the results, However these techniques based on known previously malware and ignores newly malware attacks , So in our approach we implemented the behavioral- based technique to detection unseen previously malware web services .

1- WSDL Structure : A WSDL document has various elements, but they are contained within these three main elements, which can be developed as separate documents and then they can be combined or reused to form complete WSDL files Show Fig1

```
<definitions>
  <types>
    definition of types.....
  </types>

  <message>
    definition of a message....
  </message>

  <portType>
    <operation>
      definition of a operation.....
    </operation>
  </portType>

  <binding>
    definition of a binding....
  </binding>

  <service>
    definition of a service....
  </service>
</definitions>
```

Fig-1 show wsdl structure

Definition : It is the root element of all WSDL documents. It defines the name of the web service, declares multiple namespaces used throughout the remainder of the document, and contains all the service elements described here.

Data types : The data types to be used in the messages are in the form of XML schemas.

Message : It is an abstract definition of the data, in the form of a message presented either as an entire document or as arguments to be mapped to a method invocation.

Operation : It is the abstract definition of the operation for a message, such as naming a method, message queue, or business process, that will accept and process the message.

Port type : It is an abstract set of operations mapped to one or more end-points, defining the collection of operations for a binding; the collection of operations, as it is abstract, can be mapped to multiple transports through various bindings.

Binding : It is the concrete protocol and data formats for the operations and messages defined for a particular port type.

Port : It is a combination of a binding and a network address, providing the target address of the service communication.

Service : It is a collection of related end-points encompassing the service definitions in the file; the services map the binding to the port and include any extensibility definitions

III. Methodology

In our approach we implementation a new framework to detection malware web services that include collection dataset via internet and extracting features as phase one then we analysis the behavioral of WSDL file preparing to a learning machine in phase two then we apply classification method as a decision tree to classifier a WSDL benign or malware show Fig-2

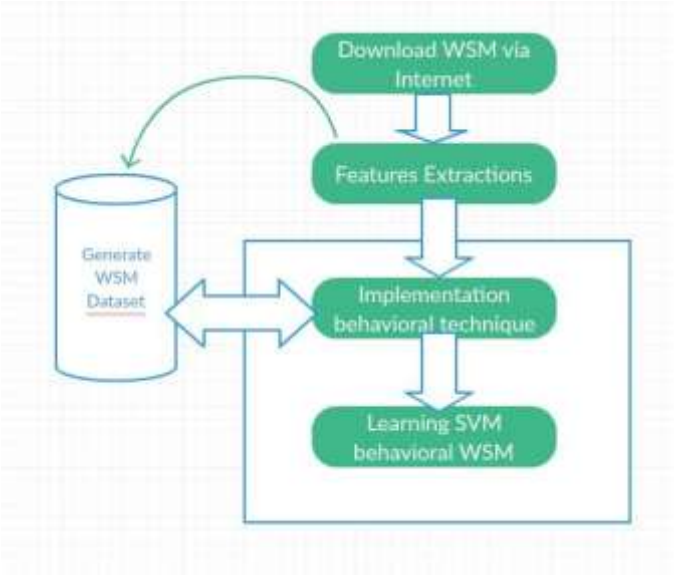

Fig- 2 our diagram framework

1- **Dataset collection and Features Extraction :-**

a)-In this phase we collected the malware web services that contain WSDL spoofing or changing metadata from many sources via internet as dataset#1 and generate dataset#2 that contains web services normal WSDL file .

b)- After step one we decompress the WSDL file to extract the content. During the first two steps we retrieve the information from this source. We process the files.wsdl to extract these data

Attribute	**Description**	**Possible Value**
Datatypes	Primitive or Doc	String ,float ,int
Messages	abstract	RPC or Doc
	definition	
Operations	abstract	messages
	definition	
Port-type	abstract set	endpoints
Binding	Concrete	Messages and
	protocol	operations

IV. ANALYSIS BEHAVIORAL OF MALWARE WSDL

After previous step we implement analysis behavioral technique to determine which are most frequently process in malware wsdl files we found some malicious operations such as :

- Change endpoint URL and target namespace Man-in-the-middle scenario
- Change message schema operations

Attach spoofed WS-Security Policy Add/remove/change/fake operations
- Spoofed WS-Security Policy: Modify security assertions
- Change cryptographic algorithms to use Encryption becomes breakable

Remove security assertions Eavesdropping and data modification
- The size of WSDL spoofing Large than normal WSDL
- Include executable file as a parameters .
- Include malicious code like JavaScript through Tags.

Show Fig-3 to see sample of malware wsdl that contain payload to executable file like calculation in client machine that request service from server .

```
<?xml version="1.0"?>
<definitions name="StockQuote"
          targetNamespace="http://example.com/stockquote.wsdl"
          xmlns:tns="http://example.com/stockquote.wsdl"
          xmlns:xsd1="http://example.com/stockquote.xsd"
          xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
          xmlns="http://schemas.xmlsoap.org/wsdl/">

 <types>
  <schema targetNamespace="http://example.com/stockquote.xsd"
        xmlns="http://www.w3.org/2000/10/XMLSchema">
   <element name="Payload" default="${=Runtime.getRuntime().exec('calc.exe')};" type="string">
    <complexType>
     <all>
      <element name="tickerSymbol" type="string"/>
     </all>
```

Fig-3 show mlawre wsdl

3. Apply Data mining techniques

a) **Feature Selection**

In Machine Learning applications, a large number of extracted features, some of which redundant or irrelevant, present several problems such as— misleading the learning algorithm, over-fitting, reducing generality, and increasing model complexity and run-time. These adverse effects are even more crucial when applying Machine Learning methods on WSDL files, Applying fine feature selection in a preparatory stage enabled to use our malware detector more efficiently, with a faster detection cycle. Nevertheless, reducing the amount of features should be performed while preserving a high level of accuracy. In this section we select the k best features from the extracted features of WSDL

files by using feature selection method: Information Gain. This method depends on entropy of the attributes and it selects the largest value of gain as the best feature. Gain of an attribute A on a collection of examples S is given by Splitting criteria used for splitting of nodes of the tree is Information gain. To determine the best attribute for a particular node in the tree we use the measure called Information Gain. The information gain, Gain (S, A) of an attribute A, relative to a collection of examples S, is defined as:

$$Entropy = \sum_{j} -P_j \log_2 P_j$$

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

b- Machine Learning and Malware Detection

The selected features are collected into the signature database and divided into training data and test data and used by standard machine learning techniques to detect the web services malware. In this case we apply clustering method using K-Means with k=3 ,it split the data to 3 cluster. we use sample 300 record. K-Means

require changing the data type from nominal to numerical to work. Data is divided into 3 clusters as shown below Fig-4 :

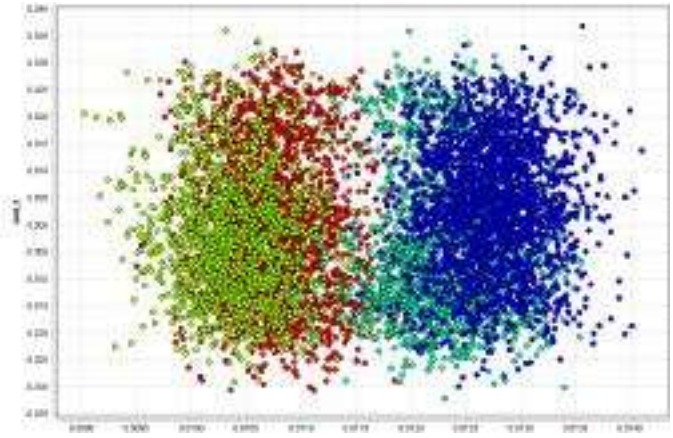

Fig-4 show clusters

J48 Decision Tree

To work out the information gain for A relative to S, we first need to calculate the entropy of S. Here S is a set of 300 examples.

To determine the best attribute for a particular node in the tree we use the measure called Information Gain.

The information gain, Gain (S, A) of an attribute A, relative to a collection of sample S.

"target namespace", has the highest gain, therefore it is used as the root node.

This process goes on until all data classified perfectly or run out of attributes. The knowledge represented by decision tree can be extracted and represented in the form of IF-THEN rules.

c)-Performance Evaluation Metrics

To evaluate our approach, we will use the following estimates to evaluate the performance of the proposed approach:

- True Positive (TP):Number of correctly detected malware wsdl .
- False Positive (FP): Number of wrongly detected benign wsdl.
- True Negative (TN): Number of correctly detected benign wsdl.
- False Negative (FN): Number of wrongly detected malware wsdl.
- Detection Rate (DR): Percentage of correctly detected malware wsdl.

Detection Rate = TP/(TP+FN).

- False Alarm: False Positive Rate is defined as FP/(TN+FP).
- Overall Accuracy Rate (OA) is defined as (TP+TN)/(TP+TN+FP+FN).

d)- Experimental Results

we extracted the necessary features to analyze from sample WSDL (benign and malware). Then, we built dataset in from the extracted features. We used these two datasets to distinguish malware and benign web services by machine learning approaches. Table 1 shows the details of two datasets used in web services malware detection framework and the experimental results of j48 machine learning approach from two datasets is shown in Table 2.

V. CONCLUSION

In this paper, we proposed a new approach to detection malware web services using datamining techniques based on WSDL file that have contributed for detection unseen previously malware we have extracted several elements features from several downloaded web services.

Some of the malware web services are used from malware sample database and both malware and normal web services are classified by using machine learning techniques. In order to validate our methods, we have collected 300 samples of web services and we have extracted the features for each WSDL file and we have trained the models which have been evaluated. Regarding future work, we will extend our framework to train models with larger dataset as soon as we obtain enough samples of malicious web services and we will extract more features from samples. We will even classify the types of malware web services (Trojan, XSS, etc).

REFERENCES

[1]. Idika, N. and A.P. Mathur, A survey of malware detection techniques. Purdue University, 2007. **48**.

[2]. Shahzad, R.K., S.I. Haider, and N. Lavesson. Detection of spyware by mining executable files. in Availability, Reliability, and Security, 2010. ARES'10 International Conference on. 2010. IEEE.

[3]. Wang, T.-Y., et al. A surveillance spyware detection system based on data mining methods. in Evolutionary Computation, 2006. CEC 2006. IEEE Congress on. 2006. IEEE.

[4]. Salas, M. and E. Martins, Security testing methodology for vulnerabilities detection of xss in web services and ws-security. Electronic Notes in Theoretical Computer Science, 2014. **302**: p. 133-154.

[5]. Aung, Z. and W. Zaw, Permission-based android malware detection. International Journal of Scientific and Technology Research, 2013. **2**(3): p. 228-234.

[6]. Burguera, I., U. Zurutuza, and S. Nadjm-Tehrani. Crowdroid: behavior-based malware detection system for android. in Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices. 2011. ACM.

[7]. Dua, S. and X. Du, Data mining and machine learning in cybersecurity. 2011: CRC press.

[8]. Quan, D., et al. Detection of Android Malicious Apps Based on the Sensitive Behaviors. in Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on. 2014. IEEE.

[9]. de Melo, A.C. and P. Silveira, Improving data perturbation testing techniques for Web services. Information Sciences, 2011. **181**(3): p. 600-619.

[10]. Valenti, A.W., Testes de robustez em web services por meio de injeção de falhas. 2011.

Table 1. Datasets for Malware Detection Framework

Dataset Name	Number of Samples	Number of Features
Dataset #1	300	10
Dataset #2	150	10

Table 2. Experimental Results of Two Datasets

Dataset Name	**Method Name**	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**ROC Area**	**Correctly Classified Instances(%)**	**Incorrectly Classified Instances(%)**
Dataset#1	**J48**	**0.880**	**0.080**	**0.880**	**0.880**	**0.882**	**88.3%**	**10%**
Dataset #2	**J48**	**0.880**	**0.131**	**0.880**	**0.880**	**0.880**	**88%**	**13%**