Web Anomaly Detection Using Artificial Immune System and Web Usage Mining Approach

Masoumeh Raji *
Yazd University
Electrical and Computer Engineering Department
raji.n@yazduni.ac.ir

Vali Derhami
Yazd University
Electrical and Computer Engineering Department
vderhami@yazuni.ac.ir

Reza Azmi
Alzahra University
Computer Engineering Department
azmi@alzahra.ac.ir

Abstract: The analogy between immune systems and intrusion detection systems encourage the use of artificial immune systems for anomaly detection in computer networks, Web servers and web-based applications which are popular attack targets. This paper presents a web anomaly detection based on immune system and web usage mining approach for clustering web sessions to normal and abnormal. In this paper the immune learning algorithm and the attack detection mechanism are described. Theoretical analysis and experimental evaluation demonstrate that the proposed approach is more suitable for detecting unknown attacks, and are able to provide a real time defense mechanism for detecting web anomalies.

Keywords: Intrusion Detection Systems; Artificial Immune Systems; Anomaly; Normal behavior; Session.

1 Introduction

The World Wide Web (WWW) considered as the largest distributed collection of information and plays an important role in human life. Web applications are becoming increasingly popular in all aspects of human activities; ranging from science and business to entertainments. Consequently, web servers and web application are becoming the major targets of many attacks. Due to the growing number of computer crimes, needs for techniques that can secure and protect web servers and web applications against malicious attacks have been highlighted. Unfortunately, current security solutions, operating at network and transport layers, have insufficient capabilities in providing acceptable level of protection against web-based attacks [1]. Attaining desired information has become a difficult task for users even in a particular website. Web usage mining techniques tries to extract patterns from the data that are collected from the interaction of users with the web. The aim of any web usage mining process is to learn models of users behavior and use these models for any application that tries to ease use of the web [2].

The Artificial Immune System (AIS) is a powerful paradigm for learning which is originally inspired from the natural immune system. There are a number of motivations for using the immune system as inspiration for clustering web users which include recognition, diversity, memory, self regulation and learning [3]. The vertebrate immune system is composed of special type of white blood cells (called Bcells), which are responsible for detecting antigens and defending against them. When an antigen is detected by the B-cells, an immune response is promoted resulting in antigen elimination. One type of response is the secretion of antibodies by B-cells (cloning). Antibodies are Y-shaped molecules
on the surface of B-cells that can bind to antigens and recognize them. Each antibody can recognize a set of antigens which can match the antibody. The strength of the antigen-antibody interaction is measured by the affinity of their match [2].

Many artificial immune models have been discussed in literature such as negative selection, danger theory and Artificial Immune Networks (AINs). We use the AIN model which was initially proposed by Jern [4]. Access log files of web servers are an important source of information for Web Intrusion Detection Systems (WIDSs).

In this paper, is worked on access log files of Apache server and an anomaly detection system for detecting web-based attacks. In the training phase, the anomaly detection system tries to learn how to distinguish normal behaviors from attack by considering several parameters. These parameters include: the number of values assigned to variables of each request within a session [1]; the length of URL of each request [7], the depth of path of each request, attribute character distribution [5], attribute length [5].

The remainder of this paper is organized as follows. In Section 2, a review on some available IDSs is presented. Section 3 discusses the goals of this study and introduces algorithm regarding the data representation. In Section 4, the experimental evaluation of the proposed system is presented. Moreover, the detection ability of the system is tested to other area dataset. Finally, Section 5 concludes our study.

## 2 Related Work

There are two possible approaches for intrusion detection. Intrusion detector can be provided by a set of rules or specifications of what is regarded as normal behavior based on the human expertise. This approach could be considered as an extension of misuse detection systems. In the second approach, the anomaly detector automatically learns the behavior of the system under normal operations and then generates an alarm when a deviation is detected from the normal model [1].

Vigna et al.[5] proposed an IDS that operates on multiple event streams and use similar features to our work. The system analyzes the HTTP GET requests that use parameters to pass values to server-side programs. However, these systems are misuse-based and therefore not able to detect attacks that have not been previously modeled. Guangmin [6] presents an immune based active defense model for web attacks (IADMW) which is on the basis of the clone selection and hyper-mutation. Http queries is considered as the antigens. An http query is represented by a vector of attributes extracted from the http query, with associated weights represented the importance of the attribute in the http query. Danforth [7] presents the Web Classifying Immune System (WCIS) which is a prototype system to detect attacks against web servers by examining web server requests. Focused on distinguishing self from non-self and laid the foundations for the negative selection algorithm. WCIS considers some features: length of the URI, number of variables and distribution of characters. Guangmin and Danforth are not considered web sessions, and a http query is labeled as an attack. Rassam [8] proposed an immune network clustering method that is robust in detecting novel attacks in the absence of labels. The purpose of this study is to enhance the detection rate by reducing the network traffic features and to investigate the feasibility of bio-inspired immune network approach for clustering different kinds of attacks and some novel attacks. Rough Set method was applied to reduce the dimension of features in DARPA KDD Cup 1999 intrusion detection dataset. Immune network clustering was then applied using ainet algorithm to cluster the data.

## 3 Proposed Method

The proposed Web Host Immune Based Intrusion Detection System (WHIBIDS) introduces immune principles into IDSs to improve the capability of learning and recognizing web attacks, especially unknown web attacks. In the proposed algorithm sessions and requests are constructed from web logs in which the clickstream data are stored. Clickstream data are generated as a result of user interaction with a website. Antigen and antibodies are represented same form and their length is equal.

**Antigen Presenting:** Define each users request as the antigens set Ag. Each request is represented by a vector of attributes extracted from the access log file. The form of the vector of the antigen set Ag is listed as following: Ag= ag| =< SessionID, URLlength, numberofvariables, distributionofcharacters, attributelength, depthofpath >

There are some shortcomings to common access log files generated by web servers such as Apache. One of these problems is to define the web sessions. Since the boundaries of sessions are not clearly defined, extraction of web sessions from these log files is not a
straightforward process. In this paper sessions are generated like [1] that demonstrate real sessions. In [1] log file is generated with software which is written by PHP and is called PHP log generator. A log file which is generated with PHP log generator includes sessions and other parameters. Session-ID shows each request which is owned by sessions. Calculate length of the URL and number of variables of each request [7].

Distributions of characters have a regular pattern [5]. For example in some of the attacks like buffer overflow, it is possible to see a completely different distribution of parameters and also this subject appears to hold true for attacks that use manifold iteration of a special character like multiple use of dot character in directory traversal flaws [9]. For each character, existence percentage of a character in the proportion to length of a parameter is calculated and then for each character average percent of these values in whole parameters of a request are computed. Calculating attribute length and depth of path that are a part of each request, for example depth of the following path is 3.

index/wp-admin/export.php

Finally, the vector that is corresponded to that request is normalized. The range of output is between 0 and 1. The normalized value for each field in a vector of a request is calculated by dividing the value of that field by the sum of values over all the fields in that vector.

Affinity function: similarity measure between tow antigen is Euclidean distance determines the distance between two web application requests. Precisely, the similarity between two requests $a_g_i$ and $a_g_j$ is defined as:

$$\text{dis}(a_g_i, a_g_j) = \sqrt{\sum_{n=1}^{k} (a_{gn_i} - a_{gn_j})^2}$$

(1)

Where $k$ is the number of features is extracted for each request. The pseudo code of the proposed algorithm is presented as following:

```
initialization;
Fix the Maximal population size $N_B$;
Initialize B-cell population and $\sigma_i^2=\sigma_{\text{init}}$ using a number of random antigen;
while all antigens are presented;
do
  Present antigen to each B-cell;
  if activated the B-cell $w_{ij} > w_{\text{min}}$;
    then
      Refresh age($t = 0$);
      Add the current B-cell ad its KNN to working sub-network;
    else
      Increment the age of B-cell by one;
  end
  if for all B-cells $w_{ij} < w_{\text{min}}$;
    then
      Create a new B-cell = antigen;
    else
      Repeat for each B-cell in working sub-network;
      Compute B-cell stimulation;
      Update B-cell $\sigma_i^2$;
  end
  antigens of a session is presented;
  then
    Clone B-cell B-cell based on their stimulation level;
    if populationsize > $N_B$;
      then
        Remove extra least stimulated B-cells;
      else
        end
    else
      end
  end
```

The modified algorithm of [2]

Algorithm 1:

As it is shown in proposed algorithm, when an antigen is unable to activate any B-cell, this antigen may represent a noise or a new emerging pattern. In this condition, a new B-cell is created which is a copy of the presented antigen. If this antigen is a noisy data and does not present a new emerging pattern, it would not get enough chance to get stimulated by incoming antigens and is probably eliminated. After each antigens of a session is presented to the network, the B-cells go under cloning operation based on their stimulation level. When the population of the network exceeds a defined threshold, the least stimulated B-cells are removed from the network. The distance measure presented in this study is used in all the steps for calculating the internal and external (B-cell to antigen) interactions of B-cells. The detailed information about cal-
Calculating stimulation level and updated it are described by [2]. In the training phase, two profiles of normal and abnormal behaviors using the proposed algorithm are built. Then, they are applied to new request and new session in order to detect abnormal behaviors in the testing phase.

4 Experimental Evaluation

There are no available data on web attacks and pure non-attack that can be used as a benchmark test; therefore, we used dataset has been gathered by [1] that it has vast variety vulnerability tests such as SQL injection, XSS vulnerability and directory traversal flaws. The empirical evaluation reported in this paper is performed on web requests of sessions. The original data used in our experiment, contains 43602 requests and 6677 sessions from log files of the web server for seven random days. Duplicate records in dataset are removed.

The maximal population size of the network is set to 50; the control parameter for the number of nearest neighbors ($K$) is set to 3. The activation threshold ($w_{\min}$) is 0.5, the similarity threshold $\theta = 0.75$. If the weighted distance is greater than $\theta$, each B-cell is activated and several of them who are belonging to a session are represented as user behavior. Evaluation is based on two criteria on two datasets, one of the criteria upon request and the other based on session, indeed the array of requests that indicated user behavior. We believe that an attack is a series of actions, so the set of requests as actual sessions are considered. To show that the proposed algorithm works on each dataset, we use Linux syscall data set that there also exists the concept of sessions. Second data set contains 13217 sessions and 66159 syscall. Both data types have been tested in the same algorithm parameters.

Different kinds of metrics are measured to evaluate the ability of the algorithm to learn the properties of the features of the data and also detecting the anomaly activities. Detection rate is the fraction of true positive rates to the number of all cases that should have been classified as positive. The false alarm rate can be defined as the proportion of actually normal cases that were incorrectly classified as anomalous.

<table>
<thead>
<tr>
<th>Table 1: evaluation on web access log dataset</th>
<th>$\theta = 0.75$</th>
<th>Accuracy</th>
<th>False alarm rate</th>
<th>Detection rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request based</td>
<td>97/3%</td>
<td>0</td>
<td>92%</td>
<td></td>
</tr>
<tr>
<td>Session based</td>
<td>98/9%</td>
<td>0</td>
<td>95%</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2: evaluation on Linux syscall dataset</th>
<th>$\theta = 0.75$</th>
<th>Accuracy</th>
<th>False alarm rate</th>
<th>Detection rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request based</td>
<td>97%</td>
<td>0.03</td>
<td>98%</td>
<td></td>
</tr>
<tr>
<td>Session based</td>
<td>98/6%</td>
<td>0.01</td>
<td>98/5%</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 3: evaluation on web access log dataset by adding 20% noise</th>
<th>$\theta = 0.75$</th>
<th>Accuracy</th>
<th>False alarm rate</th>
<th>Detection rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request based</td>
<td>80%</td>
<td>0.18</td>
<td>76%</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 4: comparison of WHIBIDS vs IADMW IDS</th>
<th>Accuracy</th>
<th>False alarm rate</th>
<th>Detection rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>WHIBIDS</td>
<td>97/3%</td>
<td>0</td>
<td>92%</td>
</tr>
<tr>
<td>IADMW</td>
<td>85%</td>
<td>0.065</td>
<td>67%</td>
</tr>
</tbody>
</table>

We run the proposed algorithm 5 times with 5-folds cross validation and the final values for evaluation measures is the average of these 5 runs. Table 1 and Table 2 represent the proposed system’s high capabilities in both criteria and both datasets. As the results show that performance of session based is the better than request based and we can claim that the proposed algorithm can detect malicious activities with high accuracy. Patterns may be repeated in multiple B-cells within the population. This is called a loss of diversity or overfitting which essentially leads to redundancy (e.g. multiple requests have the same signature). To show that there has not been overfitting in training data, 20% noise is added to the test data. Table 3 shows the noise, about 15 percent impact on the results. If overfitting had occurred would have a significant impact on results.Table 4 shows the comparison of WHIBIDS vs IADMW IDS, which comes from [6]. The detection rate of WHIBIDS is 92%, but the detection rate IADMW is 67%. Simultaneously, WHIBIDS is also capable of classifying web attacks and has a high accuracy rate 97.3%. These results show that WHIBIDS is a competitive alternative for detecting web attacks.

5 Acknowledgment

This research was supported by APA center at Yazd University. The authors would like to thank APA for its support.
6 Conclusions

In this paper we proposed an intrusion detection system, based on the principles of the immune system (WHIBIDS) that can detect known and unknown attacks. Here an attack as a series of actions is considered. The requests obtained from the preprocessed log files of web server are presented to the system as antigens. The network of the B-cells represents a summarized version of the antigens encountered to the network. Also, they are able to adapt to emerging usage patterns proposed by new antigens at any time. The results show the ability of the proposed AIS to clustering web sessions to normal and abnormal. The results indicate designing an immune base IDS that has several advantages: (1) Self learning and immune learning make the model can detect both the known and unknown web attacks. (2) Ability to detect anomaly in real time. (3) Capability to recognize abnormal behavior with regard to the actual sessions. (4) Using immune network algorithm achieved high detection rates. (5) Can be used as a general classifier. There was limitation such as determination of similarity threshold with testing. Future work will determine this threshold by reinforcement learning.

References


