A NEW APPROACH FOR ANOMALY DETECTION IN WEB APPLICATIONS

Alavi Seyed Enayatallah
Department of Computer Engineering, Faculty of Enginnerig, Shahid Chamran University of Ahwaz, Iran
Ahwaz, Iran
E-mail: se.alavi@scu.ac.ir

Abstract: In this paper, we present an approach based on XML files, which can be implemented independently from the application layer, to perform related operations to detect abnormalities in the web applications without the need for initial training. Normal behavior of web applications is automatically and accurately defined in XML files. This approach has been implemented in Medical Sciences Research Automation system (Syat) of Mazandaran, Iran. Results obtained from this evaluation indicate an increase in the accuracy of anomaly detection and reduction of the rate of incorrect detections.
Keywords: Security - anomaly detection - web application

Introduction

ДDetection of anomalous behavior of users in the Internet-based business web applications and portals is very important in maintaining security. Such users' behavior can have a significant impact on the web application. Various approaches have been proposed in this respect. Some of these approaches focus on the protection of databases and detection of abnormal commands, and others focus on the protection of all parts of web applications. To detect anomaly using these approaches, a precise picture should be presented of what the normal behavior is in a specific web application. To this end, many introduced approaches need to be learned. This Training Phase should be carefully carried out through thousands of valid requests[1-3].

I. The Proposed approach

The system presented in this paper is a simple Web Application Firewall witch analyzes HTTP requests sent by a client browser trying to get access to a web server. In our architecture, the system operates as a proxy located between the client and the web server. Likewise, the system might be embedded as a module within the server. However, the first approach enjoys the advantage of being independent of the web platform. The system follows the anomaly-based approach, detecting known and unknown web attacks. So it can be used to protect any type of web application[4-5].

Prior to the detection process, the system needs a precise picture of what the normal behavior is in a specific web application. For this purpose, our system relies on an XML file which contains a thorough description of the web application's normal behavior. Once a request is received, the system compares it with the normal behavior model. If the difference exceeds the given thresholds, then the request is flagged as an attack and an alert is launched. In this approach, the XML files are created automatically by the system (Figure 1).

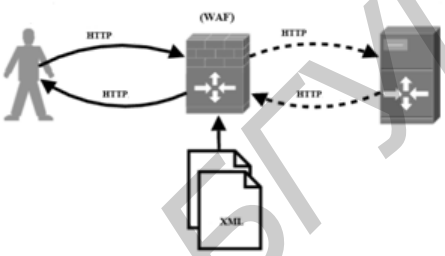

Fig.1 An overview of the proposed framework

To implement this approach, we must first clearly define acceptable entries for each field in web pages and other related parts within a web application (Verbs, Headers, etc.). To define the acceptable value of each input field, "Regular Expression" or "Regex" is used. Since most programming languages (php, java, java script, etc.) use Regex standard equally, we decided to use this feature in our system as well. Accordingly, the fields in the web forms are defined in a table named Fiel Regex. In this table, only those fields that should receive special input are defined. If an acceptable input for a particular field is not defined, the system uses the information reletaed to that field which has been defined while designing the database to define the relevant regex to be incorporated in the XML file[6,8].

II. Detection Process

TIn the detection process, our system follows an approach of the form "deny everything unless explicitly allowed", also known as positive security model. The detection process consists of several steps, each constituting a different line of defense, in which the different parts of the request are checked with the aid of the XML file. If an incoming request fails to pass one of these lines of defense, an attack is assumed: a customizable error page is returned to the user and the request is logged for further inspection. The detection process is composed of the following steps:

1. Verbs check. The verb must be present in the XML file, otherwise the request is rejected. For example, in the applications in which only GET, POST and HEAD are required to work

Библиотека БГУИР

correctly, the XML file could be configured accordingly, thus.

2. Headers check. If the header appears in the XML file, its value must be included too. Different values will not be accepted, thus preventing attacks embedded in these elements.
3. Resource test. The system checks whether the requested resource is valid.
4. 4. Arguments test:
   a) It is checked that all arguments are allowed for the resource.
   b) It is confirmed that all mandatory arguments are present in the request.
   c) Argument values are cheched. An incoming request will be allowed if all parameter values are identified as normal.

III. Analysis

The evaluation of this approach has been performed in Syat system. This system is used in Mazandaran University of Medical sciences, Iran to mechanize research plans.

To evaluate the effectiveness of the proposed approach, the assessment criteria of true positives (number of legitimate requests that have been detected correctly), true negative (number of abnormal requests that have been detected correctly), false positives (number of legitimate requests that have been detected as abnormal) and false negative (number of abnormal requests that have been detected as legitimate) will be used. In this assessment, the model proposed by Sun, J. [7] and also the model proposed by Torrano-Giménez[2] were implemented and compared with the current model. To this end, each of these approaches were tested in up to 3 thousand requests. The results obtained from this evaluation is described in Figure 2.

IV. Conclusions

The advantages of this approach could be summarized as being independent of a certain application, automatic training (without human intervention and quick application release), simple and fast implementation in complex web application, and that there is no need to retraining after changes are made in the application. Whereas, in approaches based on manual training, the time-consuming training phase should be performed again after any changes are made.

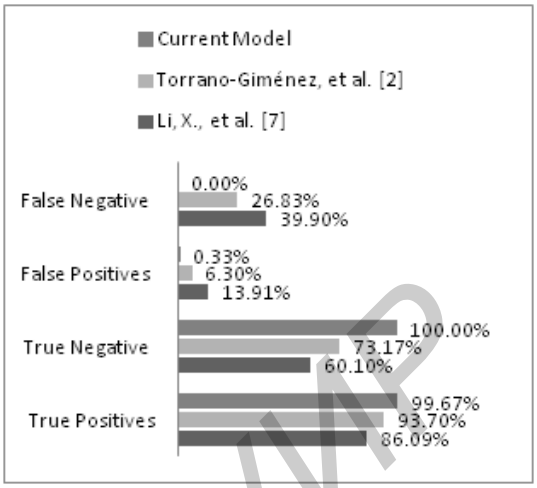

Fig.2 The results obtained from this evaluation

1. Torrano-Giménez, C., et al. An Anomaly-Based Approach for Intrusion Detection in Web Traffic / Torrano-Giménez, C., et al. // 2010.
2. Qin, Z., et al. Suspicious user tracking based on web data analysis. Systems, Man, and Cybernetics (SMC) / Torrano-Giménez, C., et al. // 2011 IEEE International Conference on, IEEE.
3. Kirchner, M. A framework for detecting anomalies in HTTP traffic using instance-based learning and k-nearest neighbor classification / Kirchner, M. // Security and Communication Networks (IWSCN), 2010 2nd International Workshop on, IEEE.
4. Zolotukhin, M., et al. Analysis of HTTP Requests for Anomaly Detection of Web Attacks. Dependable / Zolotukhin, M., et al. // Autonomic and Secure Computing (DASC), 2014 IEEE 12th International Conference on, IEEE.
5. Viswanathan, R. P., et al. Application attack detection system (AADS): An anomaly based behavior analysis approach / Viswanathan, R. P., et al. // Computer Systems and Applications (AICCSA), 2011 9th IEEE/ACS International Conference on, IEEE.
6. Threepak, T. and Watcharapupong A. Web attack detection using entropy-based analysis / Threepak, T. and A. // Information Networking (ICOIN), 2014 International Conference on, IEEE.
7. Li, X., et al. Detecting Anomalous User Behaviors in Workflow-Driven Web Applications / Li, X., et al. // 31st Symposiun on Reliable Distributed Systems, 2012 International Conference on, IEEE.
8. Ruzhi, X. and Liwu D. A learning-based anomaly detection model of SQL attacks /Ruzhi, X. and Liwu D. // Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference on, IEEE.

Библиотека БГУИР