INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 3)
Available online at: www.ijariit.com

Log based internal intrusion detection for web applications

S. Chanthini
chanthini1995@gmail.com
Anna University BIT-Campus, Tiruchirappalli, Tamil Nadu

Dr. K. Latha
erklatha@gmail.com
Anna University BIT-Campus, Tiruchirappalli, Tamil Nadu

ABSTRACT

The web environment security has become a high priority for e-businesses communities. As the usability and popularity of web applications have increased, various types of attacks in web applications also increase. The Intrusion detection system has been used for the purpose of detecting the attacks by looking at the network traffic or by looking at operating system events. Log-based intrusion detection system makes use of user logs from the web servers, where the access information about each and every request is saved. The drawback of Log-based web intrusion detection is that it can be executed only after transactions take place; hence, the prevention of attack is unattainable. In this paper we are focusing on user's behavior, to find out the difference between the common user and malicious user and try to prevent the attack before it takes place

Keywords— *Intrusion detection system, Web applications, Web log, Web security*

1. INTRODUCTION

Due to the rapid increase in network technology, the security becomes a major issue of the network. Most of the organizations depend upon the internet for communication. Owing to the rapid development in the technology and extensive use of the Internet, a lot of problems have to be tackled to secure the system's critical information within or across the networks as there are millions of people attempting to attack the system and extract critical information. Intrusion Detection and Systems (IDS) plays a vital role in those attacks by protecting the system's critical information. As firewalls and anti-viruses are not sufficient to offer full protection to the system, organizations have to implement the Intrusion Detection System to protect their critical information against various types of attacks.

The attacks may be an outsider attack or insider attack. In case of insider attack, the authorized users try to compromise the integrity, confidentiality or availability of resources. To safeguard the computers from these attacks, efficient intrusion detection systems (IDS) need to be considered. An intrusion detection system is a security tool which strengthens the security of communication and information systems. This intrusion detection system is considered to be a combination of software and hardware for analyzing the network traffic and detecting the different malicious patterns and warn the abnormal activity to the administrator.

The IDS involves the following tasks such as, First, the data is collected from the web server. Then, from the feature vector containing the different data, the required data is generated. It checks whether the collected data is malicious or not, using various techniques. Then the presence of attack is intimated to the network administrator. A security system, named the Internal Intrusion Detection System, is used to detect insider attack by using the weblogs collected from the web server. The system identifies a user's forensic features by analyzing the weblogs to improve the accuracy of attack detection and reduces the response time.

Over the last few decades, the rise in the use of web-based applications such as E-banking, e-commerce, online blogs, and social networking sites has been drastically increasing. Web applications have become a common platform for transmitting Information and delivering online services. People choose these web applications to get things done effortlessly and the developers are trying to achieve more reliability by compromising in standard coding.

As a result, web applications are subjected to attacks. Attackers exploit these vulnerabilities for adding malicious code to the application which can destroy the application. Attackers mostly exploit, the vulnerability of client-server interaction phase in a web application architecture. In client-server interaction, the client sends HTTP request to the server, by using this request attacker will be able to make changes to the web application. It is important to detect and prevent the attacks in a web application by using suitable security methods.

Over the last few decades, the rise in the use of web-based applications such as E-banking, e-commerce, online blogs, and social networking sites has been drastically increasing. Web applications have become a common platform for transmitting Information and delivering online services. People choose these web applications to get things done effortlessly and the developers are trying to achieve more reliability by compromising in standard coding.

As a result, web applications are subjected to attacks. Attackers exploit these vulnerabilities for adding malicious code to the application which can destroy the application. Attackers mostly exploit, the vulnerability of client-server interaction phase in a web application architecture. In client-server interaction, the client sends HTTP request to the server, by using this request attacker will be able to make changes to the web application. It is important to detect and prevent the attacks in a web application by using suitable security methods.

In this paper we focus on user behaviour, to find out the difference between the common user and the malicious user by using a pattern matching algorithm to prevent the attack before it takes place. The rest of this paper is organized as follows. In section 2, we described the related works of web application intrusion detection systems. The system architecture and components are described in section 3, section 4 describes the experimental results and section 5 describes the conclusion and future work.

2. RELATED WORK

Dilip Motwani et.al (2017) implemented two IDS model which includes an anomaly detection technique measured by cross entropy and a signature based attack detection using a genetic algorithm. Both the methods, that is signature based as well as anomaly-based IDS are used to detect even more attacks than either approach could detect alone. This paper focuses on the detection of three most reported application layer attacks, SQL Injection (SQLI), Cross-Site Scripting(XSS), Remote File Inclusion(RFI). This approach can address the limitations of existing signature-based IDS. The more the population size of a chromosome becomes the more it will be able to detect the new attack.

Prof. H. K. Khanuja(2016) proposed a tool called Distributed Vulnerability and Attack Detection Tool (DADT), which can be used to analyze the current security mechanism of a web application by injecting possible vulnerabilities in the web application and that vulnerability can be exploited using this tool. This tool consists of a vulnerability injection mechanism, vulnerability exploits mechanism and attack analysis and classification mechanism. This tool analyses the web application and generates a set of possible vulnerabilities in the distributed environment.

Lei Wang et.al (2017) proposed a new web anomaly method to identify user's anomalies by using FCER (Frequent Closed Episode Rules Mining) Mining algorithm to analyze the web access logs. The novel FCER Mining algorithm parallel mines the frequently closed episode rules on Spark, which handles massive data rapidly. Meanwhile, it reduces a part of rules which are redundant for anomaly detection to improve the matching efficiency. Then they proposed a grouping scheme to improve the parallel efficiency of the FCER Mining algorithm.

Hui Sun (2016) presented an approach to mine frequent attack sequence based on Prefix Span from the weblogs. Their method was effective in identifying both the behaviour of scanners and attack sequences in weblogs. But it was impossible to obtain new unknown attack sequences.

A. Juvonen et al. (2015) proposed an anomaly detection framework and several dimension reduction techniques, which included random projection, principal component analysis and diffusion map, were utilized to detect intrusion from the high-dimensional datasets in real time. However, larger volumes of data might lead to the low efficiency of the system.

J. Yang et al (2016) proposed an anomaly detection method, which used sequential pattern mining to model the software behaviour of the sample program. In the detection phase, their method compared the program library in the normal sequence mode with the current pattern to be detected and calculated sequence similarity identified the target behaviour.

3. PROPOSED SYSTEM

Several defence mechanisms have been adopted to ensure adequate security for web applications. Web Application Firewall (WAF) is the most prominent defences mechanism used to protect the web application after the deployment. WAF first analyses the web requests before they are sent to the web application and block if malicious, but is insufficient to understand the context of custom web applications. IPS (Intrusion Prevention System) is the extension of IDS which is capable of responding to attack incidents and block if essential. IPS works similar to WAF, but it can inspect in-depth traffic details to detect an attack. These tools offer a more dynamic solution than WAF as they can utilize both signature-based technique and anomaly-based technique for detecting known threats and identifying abnormal behaviour respectively.

In this article we focus on the user's behaviour, to find out the difference between the common user and malicious user and try to prevent the attack before it takes place. The Remote authentication mechanism is used to avoid the blocking of a legitimate user, the rollback mechanism is used to recover modified data and Tracking of the intruder by analyzing the corresponding system IP address.

4. SYSTEM FRAMEWORK

The Web Intrusion Detection System (Web IDS) analyses the access log files generated by a Web server. It analyses these files to detect Web server attacks. A Web server keeps track of requests in an access log file. The access log files produced by Web servers contain the requests that are posted to the Web server as well as status information that is generated by the Web server. In our proposed system we use Boyer Moore algorithm for pattern matching and the modules of the proposed framework are described as follows.

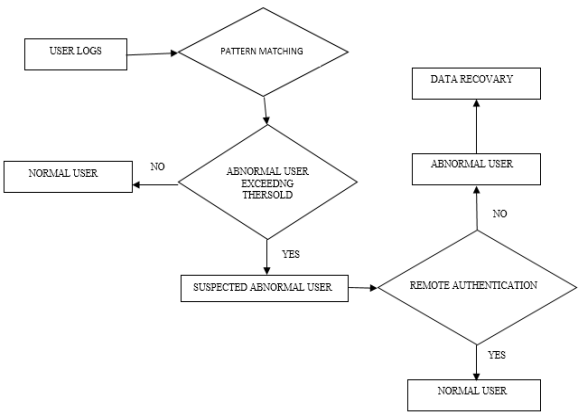

Fig. 1: Overview of IIDS for web application

4.1 Data collection

This module collects the HTTP request from the web server which contains parameters. More precisely, web applications Use POST or GET command to send the user's input value. It is very important to analyze them, which exposes factors needed for a specific application and the role of them. The analysis on GET/POST requests is a process of analyzing weak points and preparing for attacks as well as a key to check up Input Validation Attacks. We collect data into the same form with the query string and saved as a file.

4.2 Pattern Matching

The pattern Matching is a technique used to find the first occurrence of a given pattern in a stream of given text but the match should be exact which is known as Exact Pattern Matching. Text T is a sequence of characters. An alphabet set Σ is the set of alphabets that are used in a string. Pattern P is a string of length m and the prefix of P is a substring of type P [0…k] and suffix of P is a substring of type P [k……m-1]. So the problem consists of finding a substring of T equal to P. Then find the occurrence of pattern in the text 'Buffer overflow attack is performed' is known as Exact Pattern Matching. Various algorithms used different techniques to find a pattern in the text string. In our proposed system we use Boyer Moore Algorithm since it is an efficient string matching algorithm in the usual application. The reason is that it works faster when the alphabet is moderately sized and the pattern is relatively long. The algorithm scans the character of the pattern from right to left beginning with the rightmost character.

4.3 Algorithm

The pseudo code for Boyer Moore Pattern Matching Algorithm is stated as follows.

```
BOYER_MOORE_MATCHER (T, P)
Input: Text with n characters and Pattern with m characters
Output: Index of the first substring of T matching P
Compute function last
i ← m-1
j ← m-1
Repeat
   If P[j] = T[i] then
      if j=0 then
         return i
      else
         i ← i -1
         j ← j -1
   else
      i ← i + m - Min(j, 1 + last[T[i]])
      j ← m -1
until i > n -1
Return "no match"
```

4.4 User Server Authentication

The user server authentication in our proposed system is done by sending OTP to the registered email address or phone number in the user profile. OTP Verification process verifies Email Address or Mobile Number of users by sending verification code (OTP). It removes the chance of a user registering with a fake Email Address/Mobile Number. This module checks the presence of the Email Address/Mobile Number and checks whether the user can access that Email Address/Mobile Number. On the Authenticating process, an Email/SMS with OTP is sent to the email address/mobile number provided by the user. Once the OTP is entered, it is verified and the user is considered as a genuine user.

4.5 Data recovery (rollback mechanism)

All modifications within the data of web application are stored within the transaction log, with additional space also kept in the log for the undo records, in the event that it has to rollback. Each transaction log has adequate information to reverse the change it has made, so that it can undo the change if required. If the transaction has to be rolled back, the undo space reserved is going to be used, and the row would be re-inserted. The reason for the undo reservation of space is to confirm that the transaction log cannot be occupied by mid-transaction, leaving it no space to complete or rollback.

5. EXPERIMENTAL EVALUATION

The most widely adopted mechanism in web security to access the effectiveness of our system is accuracy.

Accuracy matrices target the accuracy of the detection algorithm. Hence higher accuracy value means the better overall performance of the detection system.

Accuracy is the ratio of a number of correctly detected samples to that of all samples.

$$\text{Accuracy} = (TP+TN)/N$$

$$\text{Detection Rate} = TP/(TP+FN)$$

$$\text{False alarm Rate} = FP/(FP+TN)$$

$$\text{Precision} = TP/(TP+FP)$$

$$\text{Recall} = TP/(TP+FN)$$

$$\text{F-Measure} = (2*\text{precision}*\text{recall})/(\text{Precision} + \text{recall})$$

Where,
TP-true positive
TN-true negative
FP-false positive
FN-False Negative
N-number of users

Table 1: Accuracy Detection Table

N	TP	TN	FP	FN	Accuracy
10	1	1	3	5	0.2
20	3	5	5	7	0.4
30	7	9	4	10	0.5
40	11	12	5	9	0.7
50	24	23	4	16	0.9

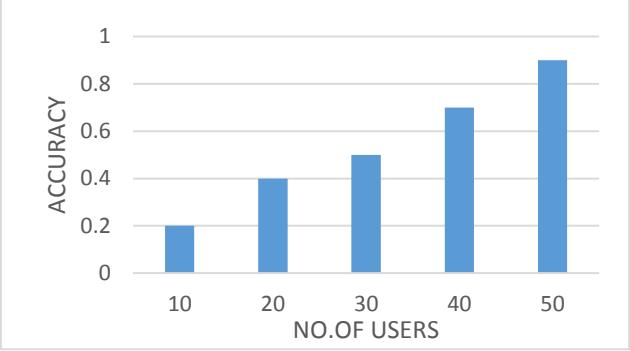

Fig. 2: Accuracy Detection

Table 3: Detection Rate Calculation

N	TP	TN	FP	FN	False alarm rate
1	1	3	5	0.1	0.75
3	5	5	7	0.2	0.55
7	9	4	10	0.4	0.3
11	12	5	9	0.5	0.25
24	23	4	16	0.6	0.1

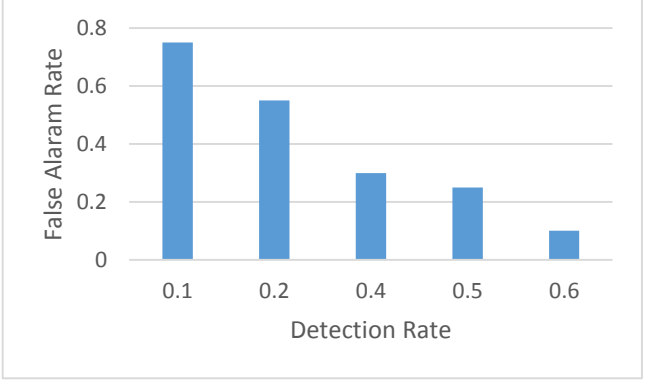

Fig. 3: Detection Rate

Table 3: Performance table

TP	TN	FP	FN	Precision	Recall	F-factor
1	1	3	5	0.25	0.16	0.15
3	5	5	7	0.37	0.3	0.32
7	9	4	10	0.65	0.4	0.48
11	12	5	9	0.6	0.5	0.54
24	23	4	16	0.8	0.6	0.68

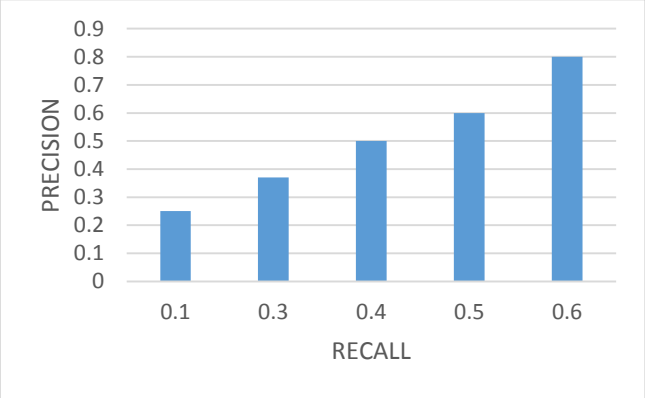

Fig. 4: System performance

6. CONCLUSION

In this paper, we developed WAIDS; a new intrusion detection method for detecting attacks against web applications. Existing detection system; NIDS will not detect the unknown attacks against web applications. The Proposed system; WAIDS can detect unknown abnormal web request and reduce false positives rate, and it can detect abnormal requests at runtime. Because of this, it can prevent web application attacks during the whole service time. Thus our proposed system can be used widely in web application security. The future work is to track the malicious user by tracking system IP and recover the data modified by the intruder by using rollback mechanism.

7. REFERENCES

[1] Khan, S., & Motwani, D. (2017, June). Implementation of IDS for web application attack using the evolutionary algorithm. In 2017 International Conference on Intelligent Computing and Control (I2C2) (pp. 1-5). IEEE.

[2] Kozik, R., Choraś, M., Renk, R., & Hołubowicz, W. (2015, September). A Proposal of an algorithm for web applications cyber-attack detection. In IFIP International Conference on Computer Information Systems and Industrial Management (pp. 680-687). Springer, Berlin, Heidelberg.

[3] Fasnamol, A. A., Fasnamol, A. A., & Ajith, S. (2018). INTRUSION DETECTION TECHNIQUES FOR WEB APPLICATION ATTACKS. International Journal of Creative Research Thoughts (IJCRT), 6(2), 914-919.

[4] Gao, Y., Ma, Y., & Li, D. (2017, October). Anomaly detection of malicious users' behaviours for web applications based on weblogs. In Communication Technology (ICCT), 2017 IEEE 17th International Conference on (pp. 1352-1355). IEEE.

[5] Pałka, D., & Zachara, M. (2011, August). Learning web application firewall-benefits and caveats. In International Conference on Availability, Reliability, and Security (pp. 295-308). Springer, Berlin, Heidelberg.

[6] Bherde, G. P., & Pund, M. A. (2016, September). Recent attack prevention techniques in web service applications. In Automatic Control and Dynamic Optimization Techniques (ICACDOT), International Conference on (pp. 1174-1180). IEEE.

[7] Al-Khurafi, O. B., & Al-Ahmad, M. A. (2015, December). Survey of Web Application Vulnerability Attacks. In Advanced Computer Science Applications and Technologies (ACSAT), 2015 4th International Conference on (pp. 154-158). IEEE.

