Project Synopsis

on

Signature and machine learning based web application firewall

Submitted as a part of course curriculum for

Bachelor of Technology

in

Computer Science

Submitted by

Prachi Sharma 2000290120108

Manya Varshney 2000290120093

Under the Supervision of

Mr. Abhishek Goyal

KIET Group of Institutions, Ghaziabad

Department of Computer Science

Dr. A.P.J. Abdul Kalam Technical University

2022-2023

DECLARATION

We hereby declare that this submission is our work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

Signature of Students

Name: Manya Varshney
Roll No.: 2000290120093

Name: Prachi Sharma
Roll No.: 2000290120108

Date: 11/11/2022

CERTIFICATE

This is to certify that Project Report entitled "**Signature and machine learning based web application Firewall**" which is submitted by **Prachi Sharma and Manya Varshney** in partial fulfilment of the requirement for the award of degree B. Tech. in Department of Computer Science of Dr A.P.J. Abdul Kalam Technical University, Lucknow is a record of the candidates own work carried out by them under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

Date: 11/11/2022

Supervisor Signature
Mr. Abhishek Goyal

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the synopsis of the B.Tech Mini Project undertaken during B.Tech. Third Year. We owe a special debt of gratitude to Mr. Abhishek Goyal, Department of Computer Science, KIET Group of Institutions, Delhi- NCR, Ghaziabad, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavours have seen the light of the day.

We also take the opportunity to acknowledge the contribution of Dr. Ajay Kumar Shrivastava, Head of the Department of Computer Science, KIET Group of Institutions, Delhi- NCR, Ghaziabad, for his full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all the faculty members of the department for their kind assistance and cooperation during the development of our project.

Last but not the least, we acknowledge our friends for their contribution to the completion of the project.

Signature:
Date : 11/11/2022

Name : Manya Varshney
Roll No: 2000290120093

Name: Prachi Sharma
Roll No: 2000290120108

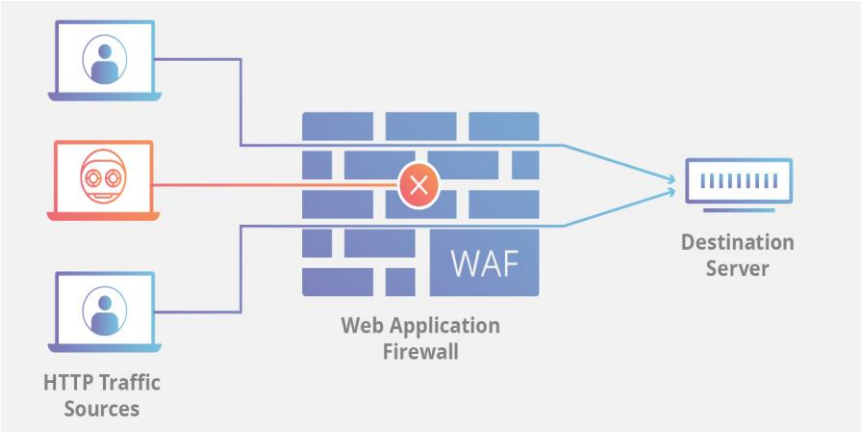
ABSTRACT

Attacks on web applications and web-based services were conducted using Hyper-Text Transfer Protocol (HTTP), which is also used as the communication protocol of web-based applications. Due to the dynamic structure of web applications and the fact that they have many variables, detection and prevention of web-based attacks are made more difficult. In this study, a hybrid learning-based web application firewall (WAF) model is proposed to prevent web-based attacks, by using signature-based detection (SBD) and anomaly-based detection (ABD). SBD is a rule-based model, defined as misuse. It is revealed by examining the characteristics, behaviors, and content of the attack. Signature-based methods generally work fast and are effective against attack types listed in the signature database. In general, intrusion detection systems work on the basis of a signature database . When a new attack technique is developed, the signature database needs to be updated in order to set the system to be efficient. Detection of known web-based attacks is done by using SBD, while detection of anomaly HTTP requests is done by using ABD. Learning-based ABD is implemented by using Artificial Neural Networks (ANN). Thus, an adaptation of the model against zero-day attacks is ensured by learning-based ABD by using ANN. a. In this project, we used methods based on deep-neural-network and parallel-feature-fusion that features engineering as an integral part of them and plays the most important role in their performance. The proposed methods use stacked autoencoder and deep belief network as feature learning methods. . Results show that deep model and feature fusion model demonstrated as hierarchical feature learning which had better performance in terms of accuracy and generalisation in a reasonable time.

TABLE OF CONTENTS

INTRODUCTION

A WAF protects your web apps by filtering, monitoring, and blocking any malicious HTTP/S traffic traveling to the web application, and prevents any unauthorized data from leaving the app. It does this by adhering to a set of policies that help determine what traffic is malicious and what traffic is safe. Just as a proxy server acts as an intermediary to protect the identity of a client, a WAF operates in similar fashion but in the reverse—called a reverse proxy—acting as an intermediary that protects the web app server from a potentially malicious client.



Generally, there are two approaches to detect attacks. The first is the signature-based method, which is to look for specific attack patterns in requests; The second is anomaly based which is to establish normal request profiles so that anomalous requests can be discriminated from normal ones.

The signature-based method is adopted more wildly than the anomaly based method because usually the signature-based one has lower false alarm rate and achieves higher accuracy. For example, ModSecurity, one popular open source Web Application Firewall (WAF), builds up OWASP ModSecurity Core Rule Set (CRS) containing a massive number of rules which can detect SQL Injection, Cross Site Scripting, HTTP Protocol Violations and etc. As effective as it is, the rule-based method is still problematic.
Firstly, it is only as good as the extent of the rule set, which means it is incapable of identifying attacks that are not in its signature dataset.

A. Problem Statement

An unprotected website is a security risk to customers, other businesses, and public/government sites. It allows for the spread and escalation of malware, attacks on other websites, and even attacks against national targets and infrastructure. In many of these attacks, hackers will try to harness the combined power of thousands of computers and sites to launch this attacks, and the attacks rarely lead directly back to the hackers.

B. Objective

The purpose of this project is to develop a web application firewall that uses Signature-Based Detection (SBD), Anomaly-Based Detection (ABD) and Artificial Neural Networks (ANN) as one of the artificial intelligence techniques. By using SBD, the detections were made against known web-based attack types such as SQL (Structured Query Language) Injection, Cross-Site Scripting (XSS), Command Injection, and Directory Traversal Attacks. HTTP requests that do not conform to the structure of the web application architecture were detected using ABD.

C. Scope of the Project

Increasing importance of web applications and rising instances of web attacks, such as cyber theft, espionage, vandalism, and fraud are driving the growth of the web application firewall market.
Furthermore, introduction of machine learning/AI-powered web application firewall and high penetration in SMEs are anticipated to create market opportunities for the web application firewall market during the forecast period.
The advantages of a web application firewall include protection against automated temporary patches, zero-data exploits, and prevents leakage of data.

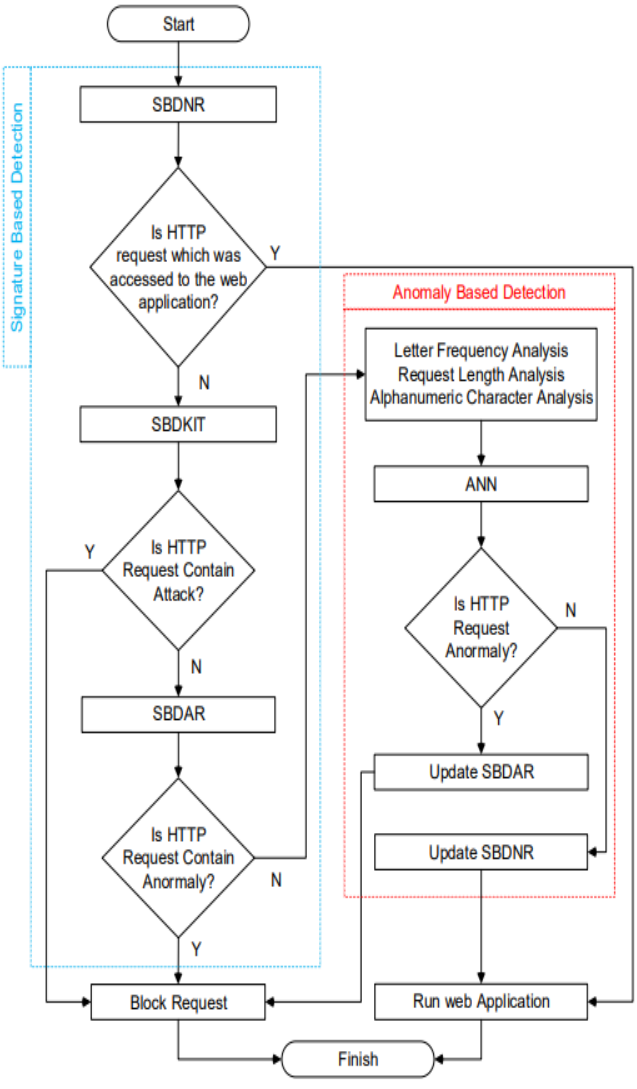
LITERATURE REVIEW

In this section, in order to support the purpose of this study and to present why our proposed model is superior to existing models, some findings of previous studies discussed. Stephan et al. (2015) developed a prototypic WAF to detect zero-day webbased attacks which did not require signature updates. They used a neural network back-propagation approach for identifying attacks that are not detected at the stage of signature analysis. The proposed study was tested on a set of parameters and some additional information about user behaviours on web application. The system was founded to obtain a good performance in comparing and matching the test patterns with the existing patterns therefore the test data has been correctly identified by a 95 % detection rate success. In another study, proposed by Nguyen et al. (2013) a web attack detection model was proposed using Generic Feature Selection (GeFS). For GeFS, CSIC 2010 and ECVL/PKDD 2007, datasets were used, and the detection was accomplished by generating 30 different features that were determined by expert opinions in datasets with 4 different sorters consisting of C4.5, CART, Random Tree, and Random Forest. Sorting was made in this study for Cross-Site Scripting (XSS), SQL injection, LDAP Injection and XPATH Injection attack types. Pa lka and Zachara (2011) discussed a selection of issues related to the implementation and deployment of the WAF protecting the target application by verifying the incoming requests and their parameters by way of matching them to recorded usage patterns. These patterns, in turn, were learned from the traffic generated by users of the application. A learning-based WAF offered a flexible, application-tailored yet easy to deploy solution. There were certain concerns, however, regarding the classification of the data that was used for the learning process which could, in certain cases, impair the firewall ability to classify traffic correctly. Basile and Lioy (2015) have developed a model for the analysis of packet filters to the policy anomaly analysis in application firewalls. Both rule-pair and multiple anomalies were detected, in order to see reducing the likelihood of conflicting and suboptimal configurations. The expressiveness of this model was successfully tested against the features of Squid, a popular Web caching proxy offering various access control capabilities. The tool implementing this model was tested on the basis of various scenarios, and it exhibited good performance. In another study, Torrano Gimenez et al. (2009) developed a system that followed the anomaly approach. It could detect both known and anomaly web-based attacks. The system decides whether the incoming requests are attacks or not by an XML signature list. Any request which deviates from the normal behavior is considered an anomaly. The system has been applied to protect a real web application. An increasing number of training requests have been used to train the system. Experiments showed that when the XML file has enough data to closely characterize the normal behavior of the target web application, a very high detection rate is reached while the false alarm rate remains very low. Kruegel and Vigna (2003) proposed a character distribution method for web attack detection. They explained

that normal web access data would possess a normal character distribution, and that anomaly data, on the contrary, would have a different character distribution. In a study conducted by Singh et al. (2011) , an iSVM-Improved Support Vector Machine algorithm developed for sorting cyber-attack data sets was proposed. Results exhibited a 100 % detection and false alarm accuracy against iSVM, Normal and Denial of Service (DoS) types as well as comparability of training and test durations. Traditional SVM performance was enhanced to enlarge spatial resolution around a conformal mapping margin with Gauss Kernel. Thus, the divisibleness between the attack types will increase. All these studies have been conducted in order to prevent web-based attacks. Unlike previous findings, in our study, a hybrid learning-based WAF model which uses SBD and ABD methods together in order to prevent web-based attacks is proposed. While the SBD method is effective against known web-based attacks and runs faster than ABD, the ABD method is effective against zero-day attacks and is able to self renew itself for new types of attack patterns. The disadvantage of ABD is that it is slow in running to identify attack patterns. Thus, methods like ABD are not preferred at real-time intrusion detection systems. In this study, the proposed algorithm uses the SBD method which is effective against known web-based attack and runs faster as well together with the ABD method, renewing itself for new types of attack patterns, for its effectiveness against zero-day attacks. With the SBD, detections have been made against known web-based attack types such as SQL Injection, Cross-Site Script (XSS) coding, Command Injection, and Directory Traversal Attacks. Through the ABD, anomaly HTTP requests have been detected with three selected features which are using ANN on a learning basis. While SBD method works quite fast, it does not work efficiently against zero-day attacks. The ABD method, however, is effective against zero-day attacks. Therefore, disadvantages of SBD and ABD methods have been eliminated by using both methods simultaneously. The Signature-Based Detection of Anomaly Requests (SBDAR) checklist has been updated with HTTP requests that were previously identified as an anomaly. When HTTP requests that were previously identified as an anomaly reach to a web application a second time, it can be blocked by SBDAR without using ABD. The other distinctive characteristic of the proposed WAF model is its ability to determine the normal requests that reach to the web applications. We also updated Signature-Based Detection of Normal Requests (SBDNR) with HTTP requests that were previously identified as normal HTTP request. With this new proposed model, clients who requested normal HTTP requests will easily access a web application without the interference of ABD. This feature will increase the speed performance of the proposed model.

METHODOLOGY

A. FLOWCHART



B. Algorithm Proposed

Signature-Based Detection of Normal Requests (SBDNR).

SBDNR is a signature-based detection stage where HTTP requests directed at web applications are normally detected through ABD. Normally, the detected HTTP requests are added to the normal signature lists so when they appear in the web application again, they are directed to the web application without running the anomaly detection process. As such, an increased speed of its performance can be ensured.

Signature-Based Detection of Known Intrusion Types (SBDKIT)

Known attacks are the web-based attack types that target web applications revealing their weaknesses by using security vulnerabilities of web applications and containing intrusion qualities. In this study, the most common web-based attack types are used as an example. These are: SQL Infection, Cross-Site Scripting (XXS), Directory Traversal Attack, Command Infection and Signature-Based Detection of Anomaly Requests (SBDAR). SQL Injection: SQL is a structural language specialized in querying within database management systems. Relational database applications in various sizes are reached through SQL queries. Many database management systems that support SQL apply special add-ons in the standard language. Web applications may be used to generate different SQL sentences for user originated inputs and requests. The SQL injection method is a penetration test. It allows for misuse of SQL sentences using vulnerabilities that arise from user inputs not being verified or being verified inadequately. If the web application does not efficiently detect user requests, the structure of the SQL sentences inside the web application could be changed; a database management system penetrated and privileges of the web application's administrator could be captured through the SQL Injection method. When performing a SQL injection, characters containing special meanings unique to SQL need to be used.

Cross-Site Scripting (**XSS**) : It is generally defined as running desired client originated scripts in users browser by means of adding client-originated scripts inside HTML codes. XSS are generally written in HTML/JavaScript languages; however, scripting can also be done in VBScript, ActiveX, Java, Flash or other languages that are supported by web applications.

Command Injection: A command injection attack is an intrusion type aimed at running scripts on the operating system containing the application through an unprotected application. These attacks are only possible when an application runs on system shell user-originated data that are unsafe (forms, cookies, HTTP Headers etc)

Signature-Based Detection of Anomaly Requests (SBDAR)

SBDAR is a signature-based detection process accomplished to block HTTP requests detected as an anomaly through the ABD process of HTTP requests. When HTTP

requests are detected as an anomaly and then re-enter the web application, they will be blocked via SBDAR without ABD processes.

Anomaly-Based Detection (ABD)

Anomaly HTTP requests behave differently than normal HTTP data. In order to carry out detection of HTTP requests which do not conform to the normal HTTP request structure, three features have been selected. According to the results of the experiment conducted with WAF 2015, CSIC 2010 and ECML-PKDD 2007 data sets, Letter Frequency Analysis, Request Length Analysis and Alphanumerical Character Analysis features are selected.

Alphanumerical Character Analysis:
The term alphanumeric is used to define a character sequence of letters and numerals (A-Z, a-z, 0-9) in the Latin alphabet. Similarly, each member of this sequence is defined as Alphanumeric. It is a cluster of definitions generated to allow for optimum memory use during data storage of computers. Generally, in one byte, the ASCII correspondence of the alphanumerical value is kept. HTTP Requests coming to the web application have a specific character sequence. This specific character sequence can be analyzed by using the Eq. 1.

According to the HTTP request structure, we can compute the specific character sequence which is found in defined global cluster (e) and generated by using equation (1). The equation (1) which is used for alphanumeric character analysis is presented. In equation (1), "|" means conditional probability, which suggests that the total alphanumerical character value is determined, based on the specific character sequence which is also a condition in the global cluster (e).

$$\mathbf{a = Xn\ i\ ri \in e\ |(a + 1),} \qquad (1)$$

where a = Total of Alphanumerical Character Values, r = HTTP Request, n = Total Number of Characters Forming Request, e = Global Cluster. Request Length Analysis: Requests coming to web applications have a specific request structure depending on the developing structure of the web application. One of the request structure feature is the request length. Request length values of memory overflow and cross-site scripting attacks are different to normal requests. Following [9], average length and variance values of the requests are used.

For Request Length Analysis,

Eq. 2 is used. The Eq. 2 is given for computing the probability of an HTTP request whether it is an anomaly or not. However, we could not decide whether a HTTP request is an anomaly or not by only using the Eq. 2. Eq. 2 produces one of the parameters that is used to digitize a HTTP request for ANN input data.

$$P = \sigma 2 (l - \mu) 2 \quad (2)$$

where P = Probability of Length, μ = Average (Average Value of Requests), σ 2 = Variance (Average Variance Value of Requests), l = Length

Artificial Neural Networks (ANN)

To implement learning-based ABD, the best ANN learning model is produced by using HTTP datasets. Firstly, ANN input and output data were normalized by using min-max method linearly between 0-1. The purpose of ANN training is to reach minimum error rate, by using the lowest number of hidden layers and neurons. In the training of ANN weights, a feedforward backpropagation algorithm is selected.

The most important process at this stage is the process of determining the number of hidden layers, the number of neurons in the hidden layers, and the activation function. Hyperbolic tangent is used as the activation function. The hidden layer and neuron numbers are determined experimentally to obtain the best ANN result.

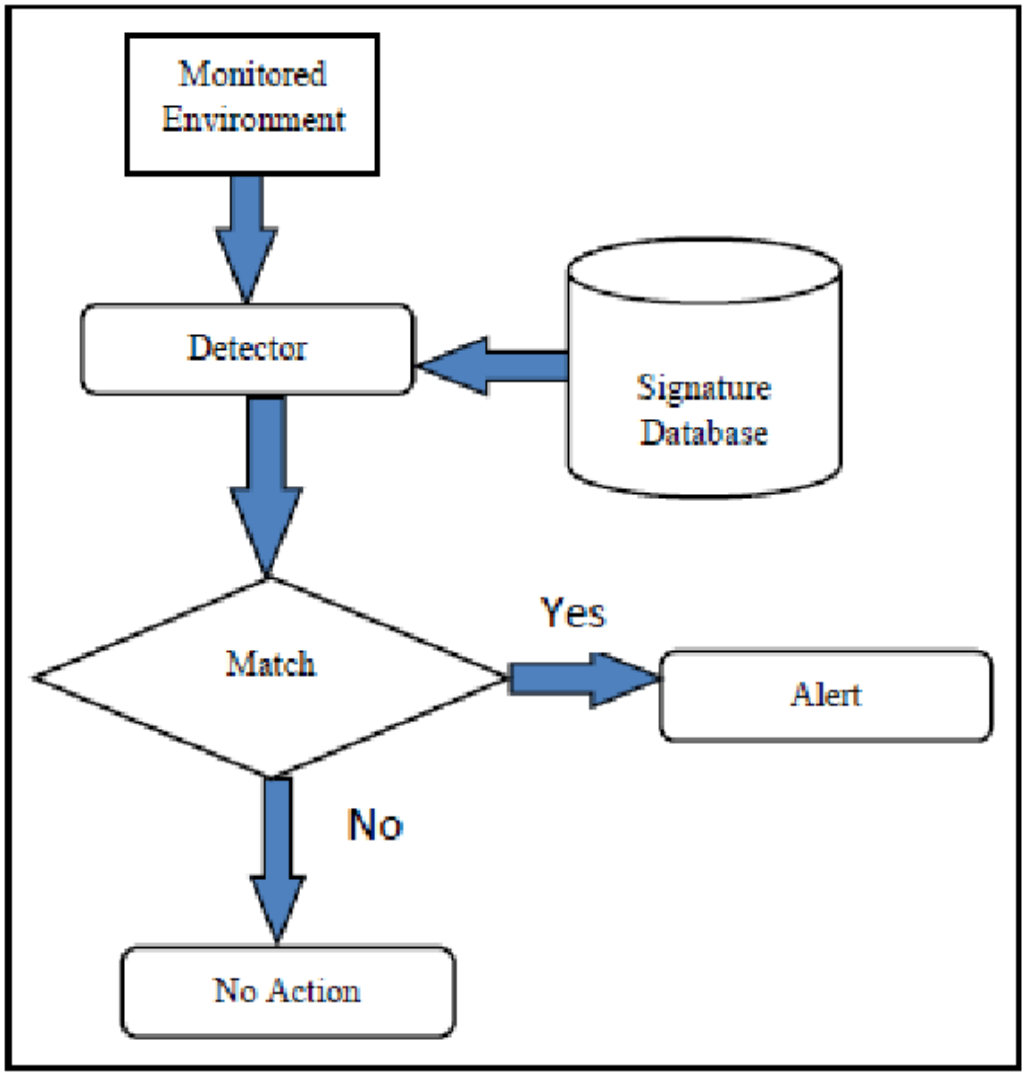
The number of layers can be increased according to the difficulty of the problem, but if the number of layers is much more than required, it causes an increase of process time and memorization of the network. To generate the best ANN model, Mean Squared Error (MSE) value is selected as the lowest possible (close to 0) and the value of regression is selected as the highest (close to 1). In order to train the network, 80 % of all datasets were determined as training data, and 20 % of the dataset as test data. The network has one hidden layer and ten neurons in that hidden layer.

TECHNOLOGY USED

- **Python**
- **Anaconda Distribution**
- **Tensorflow, Keras**
- **Oracle**
- **Database Management System**

DIAGRAMS

This architecture uses the detector to find and compare activity signatures found in the monitored environment to the known signatures in the signature database. If a match is found, an alert is issued and there is no match the detector does nothing.



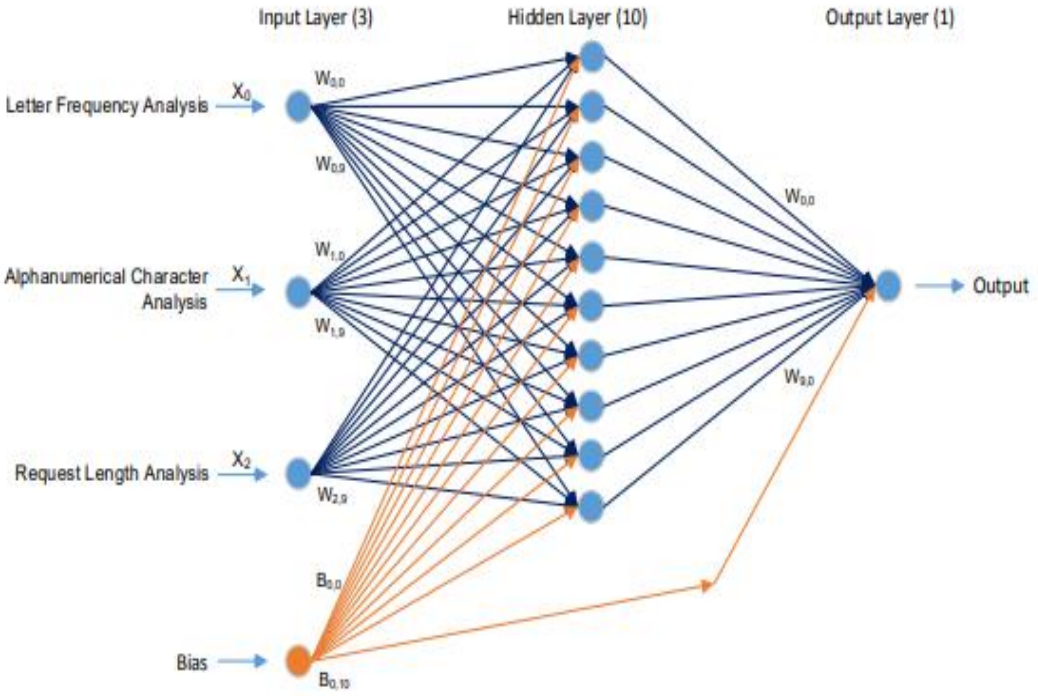


Fig. 3 *Proposed artificial neural network model (3-10-1).*

The ANN model is developed to determine the learning method for the purpose of determining anomaly or normality of the HTTP requests within the web application.

RESULT

Detection times of stages in the SBD are identified as 2.7 ms, 2.4 ms and 3.2 ms for SBDNR, SBDAR and SBDKIT, respectively. Detection time of the ABD is identified as 12.3 ms. Because the ABD detection time is higher than the SBD detection time, considering the system's speed performance, SBD is implemented before ABD.

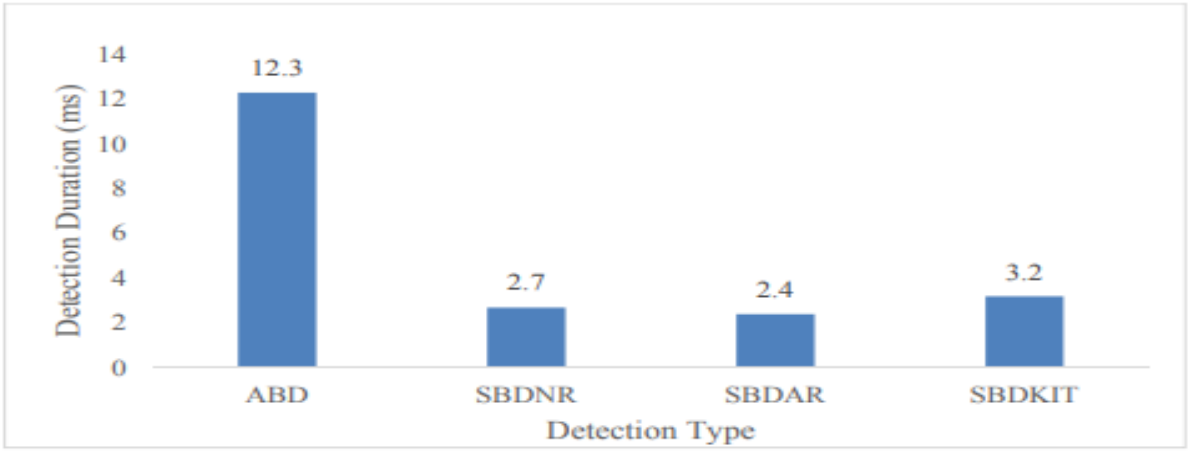


Fig. 7 *Comparison time performance of detection stages.*

The proposed model is an approach for a WAF algorithm combined with ANN. It is intended for the prevention of web-based attacks. The model test results were shown in Fig. 8 with different datasets. The test results as shown in Fig. 8 were obtained from WAF 2015, CSIC 2010 and ECML-PKDD 2007 datasets by using the WAF software. According to the test results, 98.52 % of the HTTP requests that contain attacks were detected in the WAF 2015 dataset. 96.74 % of the HTTP requests that contain attacks were detected in the CSIC 2010 dataset. 94.53 % of the HTTP requests that contain attacks were detected in the ECML-PKDD 2007 dataset. 14 Fig. 7. Comparison time performance of detection stages The proposed model is an approach for a WAF algorithm combined with ANN. It is intended for the prevention of web-based attacks. The model test results were shown in Fig. 8 with different datasets. The test results as shown in Fig. 8 were obtained from WAF 2015, CSIC 2010 and ECML-PKDD 2007 datasets by using the WAF software.

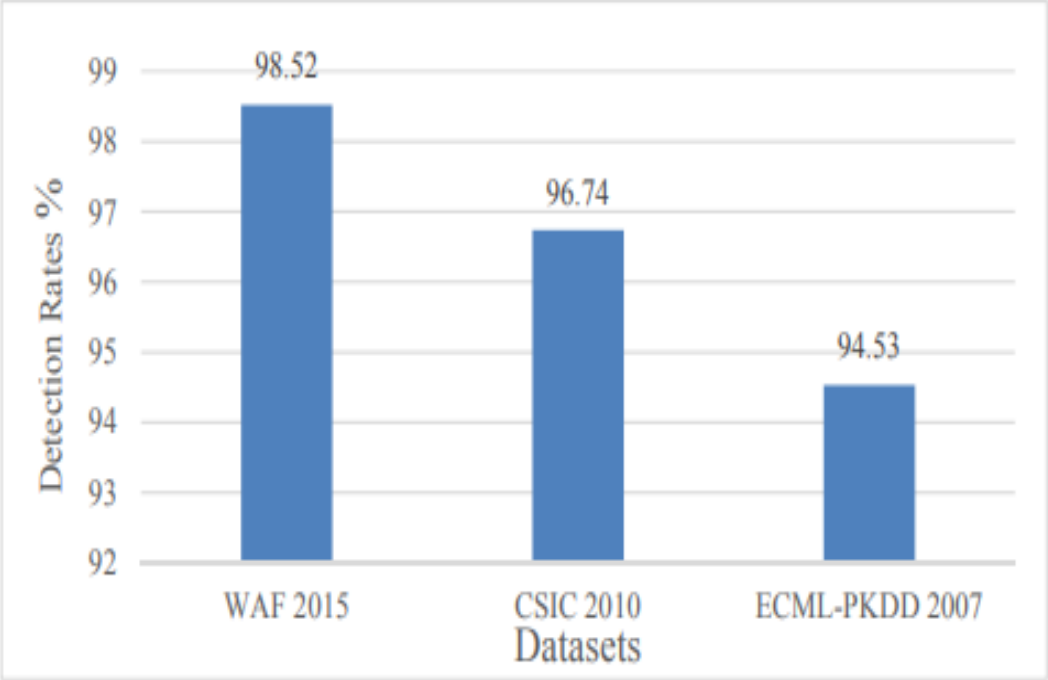


Fig. 8 *Success rates of test results for datasets.*

CONCLUSION

In this study, a learning-based hybrid WAF model is proposed and implemented to prevent web-based attacks. Detection of web-based attacks to the web applications is performed by using SBD and ABD methods in tandem. In this way, respective disadvantages of SBD and ABD methods could be eliminated. Through the SBD, three detection stages are designed. The first stage is the SBDKIT which ensures detection of some known web-based attack types such as SQL Injection, Cross-Site Script (XSS) coding, Command Injection and Directory Traversal Attacks. The second is the SBDAR checklist, which is updated with HTTP requests that were previously identified as an anomaly. When such HTTP requests reach a web application a second time, they can be blocked by SBDAR without using ABD. The last stage of SBD is to update SBDNR with HTTP requests that were previously identified as normal HTTP requests by using ABD. HTTP requests which are not able to be detected by using the SBD, are directed to the ABD.

REFERENCES

[1] BHOR R.V., KHANUJA K.H. Analysis of Web Application Security Mechanism and Attack Detection Using Vulnerability Injection Technique, 2016 International Conference on Computing Communication Control and automation (ICCUBEA), Pune, 2016, pp. 1–6, doi: 10.4236/jcc.2015.39004.

[2] VALEUR F., MUTZ D., VIGNA G. A learning-based approach to the detection of SQL attacks. In: International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, pp. 123–140, 205, Springer, Berlin, Heidelberg, doi: 10.1007/11506881_8.

[3] SANGHI D. Web Application Firewall, Kanpur, 2007.

[4] STEPHAN J.J., MOHAMMED S.D., ABBAS M.K. Neural network approach to web application protection. International Journal of Information and Education Technology, 2015, 5(2), p. 150.

[5] NGUYEN H.T., TORRANO-GIMENEZ C., ALVAREZ G., FRANKE K., PETROV˙IC S. ´ Enhancing the effectiveness of web application firewalls by generic feature selection, Logic Journal of IGPL, 2012, 21(4), pp. 560–570, doi: 10.1093/jigpal/jzs033.

[6] PA LKA D., ZACHARA M. Learning web application firewall-benefits and caveats. In: International Conference on Availability, Reliability, and Security, 2011, pp. 295–308, Springer, Berlin, Heidelberg, doi: 10.1007/978-3-642-23300-5_23.

[7] BASILE C., LIOY A. Analysis of application-layer filtering policies with application to HTTP. IEEE/ACM Transactions on Networking (TON), 2015, 23(1), pp. 28–41, doi: 10. 1109/TNET.2013.2293625.

[8] TORRANO-GIMENEZ C., PEREZ-VILLEGAS A., ALVAREZ G. A self-learning anomalybased web application firewall. In: Computational Intelligence in Security for Information Systems, 2009, pp. 85–92, Springer, Berlin, Heidelberg, doi: 10.1007/978-3-642-04091-7_ 11.

[9] KRUEGEL C., VIGNA G. Anomaly detection of web-based attacks. In: Proceedings of the 10th ACM conference on Computer and communications security, 2003, pp. 251–26, doi: 10.1145/948109.948144.

[10] SINGH S., AGRAWAL S., RIZVI M.A., THAKUR R.S. Improved Support Vector Machine for Cyber Attack Detection. In: Proceedings of the World Congress on Engineering and Computer Science WCECS. San Francisco, USA, Vol. 1, 2011.

[11] TEKEREK A., GEMC˙I C., BAY O.F. Design and Implementation of a Web-Based Intrusion Prevention system: A New Hybrid Model, Journal of the Faculty of Engineering and Architecture of Gazi University, 2016, 31(3), pp. 645–653

[12] JIA X. Design, Implementation and Evaluation of an Automated Testing Tool for Cross-Site Scripting Vulnerabilities. Darmstadt University of Technology (TUD), 2006.

[13] VURAL Y. Enterprıse Informatıon Securıty and Penetratıon Testing. Gazi University, 2007.

[14] KAPODISTRIA H., MITROPOULOS S., DOULIGERIS C. An Advanced Web Attack Detection and Prevention Tool. Information Management & Computer Security, 2011, 19(5), pp. 280–299,