International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2017/48968 Volume 4, Issue 2, March – April (2017)

RESEARCH ARTICLE

A Novel Hybrid Approach for Detection of Web-Based Attacks in Intrusion Detection Systems

Muhammet Baykara
Department of Software Engineering, Firat University, Elazig, 23119, Turkey.
mbaykara@firat.edu.tr

Resul Das
Department of Software Engineering, Firat University, Elazig, 23119, Turkey.
rdas@firat.edu.tr

Published online: 26 March 2017

Abstract – Importance of information security systems is increasing in parallel with the rapid developments in information technology. The development of new technologies brings new security weaknesses in corporate and personal meaning can lead to unavoidable losses. For this reason, many researches have been performed in order to ensure the security of information systems. In today's world, the concept of information has been moved to the digital size from conventional size. Protection of the data stored in the digital archive and is easily accessibility at any time have become a quite important phenomenon. In this concept, intrusion detection and prevention systems as security tools are widely used today. In this paper, a hybrid real time intrusion and prevention system approach has been proposed for web applications security. The proposed system uses rule-based misuse detection and anomaly detection as intrusion detection method and uses network packets as data source. The system is real-timed with accordance to data process time, centralized with accordance to architecture, and server-based with accordance to system it protects. The developed system has been tested on the current web attacks determined by OWASP (The Open Web Application Security Project) and provides a very high success rate.

Index Terms – Web Attacks, Intrusion Detection and Prevention Systems, Information Security, Network Analysis.

1. INTRODUCTION

Parallel to the extraordinary developments in the world of information technologies, the importance of information systems security in terms of individuals, institutions and organizations is also increasing. There are many different studies to ensure security of information systems. It is aimed to provide security through these studies which can be achieved in both hardware and software [1-7]. The security software used for this purpose is very important for the protection of systems belong to individuals and institutions.

Along with the developing technology, the level of progress in IT world has become a new power element in the inter-country relations. Nowadays, some new terms called "cyber space", "cyber force", and "cyber army" have been introduced to the concepts as power elements according to the countries like military, economic etc. Unlike physical domain, traditional warfare and attacks are no longer taking place rather cyber-attacks are being carried out. Therefore, the countries now establish their cyber armies besides the military armies, and the attacks are carried out in the structure called cyber space.

According to a research by Symantec published in 2013 entitled “Internet Security Threat Report”, a total of 69 million cyber-attacks have been detected in 157 countries worldwide. In 2015, the number of zero-day vulnerabilities discovered more than doubled to 54, a 125 percent increase from the year before (Symantec’s 2016 Internet Security Threat Report). According to the same report, there were over one million web attacks against people each day in 2015 and early 75 percent of all legitimate websites have unpatched vulnerabilities. When looking at the historical process of information systems, it is seen that as systems evolve, a number of new vulnerabilities caused by the weaknesses of developing technology and an increasing rate of cyber-attack numbers due to exploitation of these vulnerabilities.

Tools such as firewalls, antivirus software, Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) have been developed to protect information stored in digital media against all types of attacks and threats. IDS/IPS, one of the developed tools based on the understanding of information and computer security today, is a kind of alarming mechanism in digital systems against attacks. By using IDS/IPS, unauthorized user access to the system and threats of abuse can be detected and the damage that they may cause is prevented. With the use of IDS/IPS, it is possible to obtain detailed information about the system's vulnerabilities, the analysis and classification of attacks on the system, and the attackers [8].

IDS/IPSs first emerged in 1980 as a result of Anderson's work. Then these systems have continued to develop rapidly with many studies. The need for audit data in IDSs has increased the investigation of fields related to acquisition and use of audit data. Developed in 1988, Intrusion Detection Expert System

RESEARCH ARTICLE

(IDES) is one of the most important IDS studies that combined many of the researches done up to that time. In addition to statistical approaches, rule-based threshold values, state transition diagrams, and data mining methods have also been used in those years. However, despite the rapid development of technology and the attackers having less knowledge and experience, by taking advantage of this development, more effective and faster attacks have been made. So the security issues had to be dynamic and continuance. Thus, intelligent learning techniques such as artificial neural networks, artificial immune system, and fuzzy logic have begun to be used in the process of historical development of IDSs [8, 9]. It has been observed that the success rates of IDSs have been increased with the use of intelligent systems. In particular, the use of intelligent techniques has been one of the major factors in increasing the success rate in the anomaly detection approach used to detect new unknown attacks.

When the literature related to the subject is analyzed, it has been observed that in data collection, labeling, storage, data reduction (filtering, feature extraction and classification), identification and classification of behavioral models, determination of rules for rule-based systems, reporting and generating results phases have some difficulties. One of the biggest problems with IDSs is the lack of databases that can be used in the design and implementation phases of IDSs. Another problem encountered in IDSs is that false alarm (false positive) rates cannot be reduced. False positive is not often encountered in IDSs that uses misuse detection technique, but in systems that detect anomalies, this is usually the case. This is due to the difficulties encountered in modeling behavior or user profiles, or the nature of detecting anomalies. The way to overcome this problem is through well designed databases.

Based on the reviewed literature, it has been determined that new enhanced IDSs need to be developed in order to solve problems related to IDSs. In particular, the development of intelligent IDS/IPS systems are needed to reduce false positive alarm rate. It is also clear that new data sets that include current attack patterns for IDSs, should be developed. In this study, IDS/IPSs have been examined and the number of software have been developed for the security of web applications.

2. ANALYSIS OF INTRUSION DETECTION AND PREVENTION SYSTEMS

With the development of computer technology in today's world, attacks have gained a different dimension. Nowadays classic physical offensive and defensive activities are also observed as cyber war activities in the world of information systems. A variety of attacks can be made on systems over the cyber space, unexpectedly and unknowingly, and regardless of distance.

With the developed technology, every event that happens in computer systems are logged. With the use of this log files, it can be determined how the attacks were done and some precautions can be taken against them. Information security is a critical process that must be continuously done. For this reason, monitoring of the attacks, attacks analysis, response and prevention processes must always be done. Figure 1 presents a functional structure of how these processes are done.

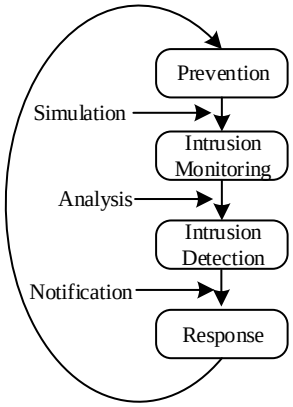

Figure 1 Functional Structure of IDS [8]

In generally for IDSs, a structure that includes a sample configuration data, audit data archive, analysis modules, result archive and user interface is proposed. Figure 2 shows the basic structure and components of IDSs. It is assumed that the audit trail data is used in this scenario. According to the type of IDSs, log records or network packets can also be used. Basically, the analysis process is performed by comparing the audit data from the archive with the event data. The results of this analysis are sent to the results archive and can also be presented in a user interface as a status report.

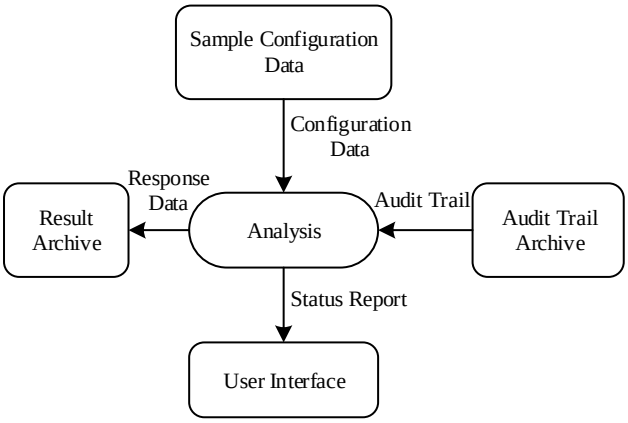

Figure 2 Basic Components of IDS

IDSs can be classified according to different characteristics. In general, data processing time, architectural aspects, methods used in intrusion detection, information source, and protected system are some features used for classification. In IDSs, the most common classification criterion is the attack detection method. According to this criterion, IDSs are divided into two

RESEARCH ARTICLE

classes as anomaly detection and misuse detection. A general classification according to the most used classification criteria for IDSs is given in Figure 4.

An intrusion detection system that may have a different feature in each category is given in Figure 4 [9]. This situation depends on the nature of the classification. For example, IDES developed in 1988 is a real-time system according to the data processing time. IDES is also a centralized system based on architecture, server based system according to the information source, and an anomaly detection system according to the protected system. If the determinant classification attribute is only the approach used for detection of attacks, IDES is in the anomaly detection class. In this case, other criteria are not mentioned. However, all these properties determine the characteristic specification of an IDS/IPS [9].

In general, there are basically two types of approaches in IDS systems, namely "misuse detection" and "anomaly detection". There are various studies deploy these two approaches in the literature; however, some studies suggest that deploying these two approaches together should continue [8, 9].

In information systems, IDSs are positioned behind router devices in general, in order to be able to analyze attacks that can occur both from the external network and from the internal network. On the other hand, IDSs are used to detect attacks that can overcome the firewall with dynamic monitoring capabilities after firewall devices with static monitoring capability. Figure 3 shows a generic localization model for the IDSs on the network.

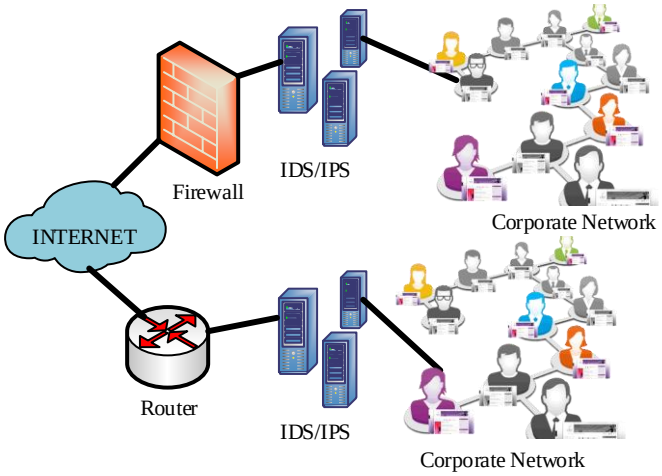

Figure 3 General Localization of IDS/IPS in Computer Networks

The approach called anomaly detection models the user behavior by analyzing the activities in the system. While the approach called detection of misuse (signature) models the behaviors of attackers. Some metrics are used during the evaluation of IDSs. The metrics and definitions used in this study are given below.

True Positives*:* Used for correctly classified intrusions.

True Negatives: Used for correctly classified non-malicious activities.

False Positives: Used for non-malicious activities that are classified incorrectly as an intrusion.

False Negatives: Used for activities that are classified as non-malicious activities incorrectly, in fact attacks.

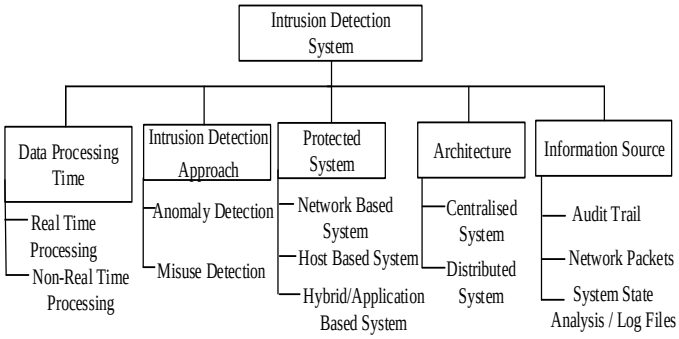

Figure 4 Classification of Intrusion Detection Systems

2.1. Data processing time based IDSs

The data processing time for IDSs is one of the criteria used in classification and refers to the time elapsed between the monitoring of the events and the evaluation of the events.

In real-time IDS/IPS system, the data is analyzed in the real-time zone provided by the communication, and if an attack is detected, the necessary operations are performed in response to the attack. It is difficult and costly to implement real-time IDS/IPSs in large networks/systems where information flow is congested. However, these systems are compulsory structures that must be used in organizations that have critical importance where it should actively producing real-time responses. There are many commercial IDS/IPS in this type.

In non-real time IDS/IPS system; the received data is stored and sent to the relevant IDS for analysis. In this structure, instantaneous real-time attack detection cannot be achieved. Because of the difference in time which the attack analysis is made and the time which the possible performed attack. The threats and possible attacks identified in these systems generate alarms to alert the system user. In order to define the profile of the attack and to avoid any further exposure to such attack, the results are obtained by analysis of past stored data. An IDS/IPS in this structure is used to overcome specified security vulnerabilities in the system [9].

2.2. Architecture Based IDSs

The architectural structure of IDSs is related to how functional components that are to be actively involved in the system are

RESEARCH ARTICLE

positioned relative to each other. Basic functional components in an information security system or computer network system: the monitored system, the server and its environment, network devices and target systems which is monitoring.

In terms of architectural structure, IDSs are divided into two types, namely central system and distributed system. In central system structure, all monitoring, detection and reporting operations are controlled at a central location. Physical proximity is not an important factor for central control. Most of the known IDSs fall into this category.

In distributed systems, all attack analysis and network monitoring are done by agent based approach at the analysis point. The distributed IDS structure is typically used in very large networks. For example, IDSs that content all computer systems of a corporation in different location fall into this category.

2.3. Information Source Based IDSs

IDSs use information sources for analysis and comparison to identify attacks. Basically, the information sources that can be used for intrusion detection can be obtained from the computer or network packets, as well as from the behavioral patterns of the users. These information sources are audit trail, network packages and application log files.

Audit trails are the system activities arranged in time order for the security of information and communication. This structure is used when reconfiguring the system and performing test operation. This happens when the sequence of events occur in the system is corrupt or changes occur in these events. IDSs use the audit trail to extract behavioral patterns of users or groups that are defined in the system. User profiles are created according to daily activities and the authorities over these activities. The accuracy of these profiles is verified by various analyses. IDSs perceive any kind of movement that is contrary to those determined profiles as an attack [8].

Network packages, other structures of information sources for IDSs, are obtained by listening to the network traffic by sniffers. These logs are mainly used to detect Denial of Service (DoS) attacks. Through the information obtained by listening to network packets, unlike server-based IDSs, it is possible to detect attacks occur at the network layer.

Application log files can be used to detect attacks that may occur at the application layer. This data source can be easily obtained more than the other two sources, but the attack detection rate is lower. In addition, if a real-time intrusion detection is required, this information source is not used because it is not possible in practice to build and analyze application log files simultaneously. Analyses of log records are made in non-real time systems where the analysis phase time is not the same with attacks time. An IDS may use application enrollment files such as WebWatcher's application log files, a software developed to detect web-based attacks [8], as an information source.

2.4. Detection Method Based IDSs

In IDS/IPS, two different approaches are used as attack detection methods: anomaly detection and misuse detection. These two approaches are explained in the following sub-sections.

2.4.1. Anomaly Detection

Basically, anomaly detection approach works by the logic of distinguishing abnormal events from normal events in the system. For an IDS in an information security system, the anomaly corresponds to any deviation from normal activities. The normal behavior of the system can be obtained as a result of a long analysis. Detection of behavioral profiles of user or user groups in the system is the most basic task for detecting anomalies. After the normal behavior profile is determined, the behaviors that differ are identified as the attack. Correct detection of attacks in detecting anomalies is proportional to how well the normal behavior profile is determined [8, 9].

Systems that detect anomalies are also called intelligent systems. This is due to fact that the normal activities in the system are constantly updated. All activities in the system are constantly monitored and compared with the normal detected activities. If an anomaly is detected, it is considered an attack. It is assumed that all exploiting activities will occur absolutely unusual while detecting anomalies. For example, when looking at the statistics in the system, normal usage patterns (CPU usage, working hours, etc.) are determined, and those outside normal usage are considered attacks [9]. The difficulty of this method is determining normal system usage patterns. In this method, due to incorrect or insufficient modeling, normal operations can be mistakenly considered as an attack. The advantage of this method is that new attacks can be detected. The following profiles are generally used to determine abnormal behaviors [9].

- **Individual Profiles*:*** Includes the general activities (user's working hours and working style) expected to be done by the user.
- **Group Profiles*:*** Behavior of the users in a group; the way the users work, the resources they use and the historical activities.
- **Source Profile:** How resources such as applications, user accounts and the used communication ports are observed.
- **Other Profiles:** They observe how executable programs use system resources.

Three basic techniques can be mentioned for the anomaly detection approach. These are statistical anomaly detection, data mining methods and artificial intelligence methods [9].

RESEARCH ARTICLE

- **Statistical Anomaly Detection:** In this method, two profiles are kept for each user. One is the profile stored in the system and obtained as a result of monitoring each user, and the other is the instant profile of the same user. By looking at the log records on the computer, the incoming profile is updated at regular intervals and compared to the profile stored in the system. If the difference is greater than the predetermined threshold, the system will mark it as an attack and give an alarm. The advantage of this method is that there is no need for prior knowledge of attacks and it can recognize new attacks. The disadvantage of this method is that it can be trained by skilled attackers to make the system recognize an abnormal activity as a normal activity.
- **Data mining:** In this technique, data is taken as input and attempts are made to reveal associations that cannot be easily seen by the naked eye. For example, defining the boundaries of the correct network traffic via the help of data mining; this helps the analyst to distinguish the attack from normal network traffic.
- **Artificial Intelligence:** Learning with artificial intelligence techniques is defined as the ability of a system to learn a specific event and update itself over time [8, 9]. In other words, the system can adapt to react to new events.

The advantage of the anomaly detection approach compared to the misuse detection approach is the ability to detect previously unknown attacks. However, the disadvantage is that false alarms, activities that are not actually attacks, are also detected as attacks. A variety of different techniques such as statistical methods, artificial intelligence techniques, artificial neural networks, data mining, and artificial immune systems can be used to detect anomalies [8, 9]. The operational structure of the anomaly detection method is shown in Figure 5.

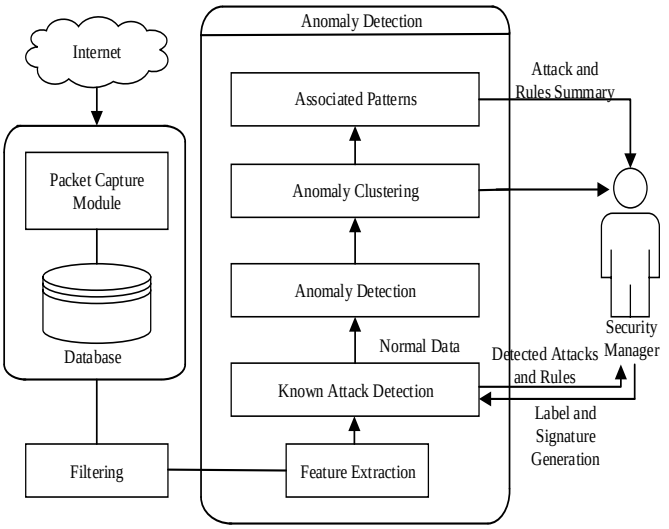

Figure 5 Anomaly Detection

2.4.2. Misuse Detection

In the misuse detection approach, extraction of behavioral models pattern of attackers is essential. These models, which are also known as the signature of the attack, are obtained by analyzing previously encountered attacks and extracting their characteristic specifications. After creating signature databases, the activities that match with those signatures are identified as attacks.

The advantage of the misuse detection approach compared to the anomaly detection is that every known attack can be detected and does not produce false positives. However, the disadvantage is that unknown attacks cannot be detected, and thus the rate of false negatives may be high. In order to reduce the false negative rate, the signature database must be updated with new attack signatures [8]. Figure 6 shows operational structure of misuse detection approach.

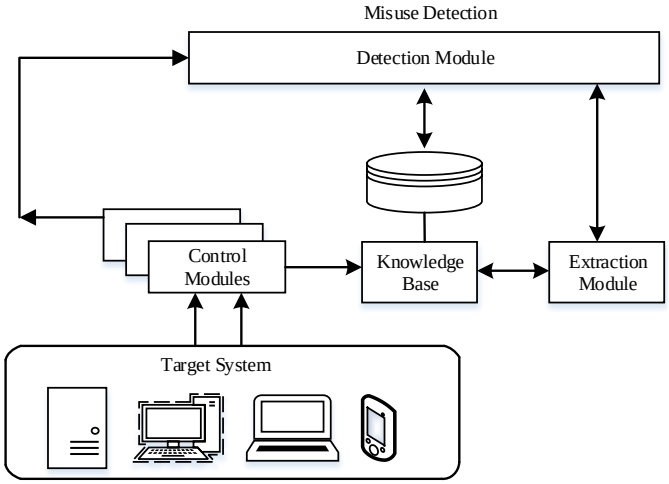

Figure 6 Misuse Detection [8]

In misuse detection method, known attack patterns are defined, and these actions are compared with existing activities to decide whether or not there is an attack. For example, for the "guess password" attack, "If there are four unsuccessful login attempts within two minutes, this is an attack" rule can be defined [9].

The greatest advantage of this method is that known attack patterns can be introduced to the system. Thus, known attacks can be detected completely and accurately. Nevertheless, it cannot recognize new attacks that have not been introduced to the system. The basic logic in the misuse approach is that attacks can be defined as a specific pattern. In the method of anomaly detection, while attempting to detect unknown malicious behaviors, the mechanism tries to identify the malicious behaviors known in the misuse method. Methods of misuse detection system can be divided into four groups [9];

Expert Systems: Experts with coding knowledge try to detect attacks by defining rules if-then.

RESEARCH ARTICLE

Model Based Reasoning Systems: When misuse occur, misuse models are used together to try to get correct results.

State Transition Analysis: This method models the attacks as a series of state transitions in the system where attacks are monitored.

Keyboard usage analysis: By checking the user's speed of keyboard use, it is possible to detect the attack when another user is using the computer. This method only tests keyboard usage, so it cannot detect attacks with malicious software.

IDSs can collect and process data at various levels. Generally collected data are as follows [12]:

- Network data (collected from router or firewall devices),
- Operating system command prompt,
- Operating system calls,
- Internal information of applications,
- All transmitted characters,
- Keyboard strokes.

2.5. Protected System Based IDSs

Intrusion detection systems can also be classified according to the structure of the system they protect. According to this classification method, IDSs can be divided into three groups. These are network, server, and application-based IDS structures. IDSs can monitor a small or large area in terms of protection scope. IDSs that monitor a small area are generally known as server-based systems, while IDSs that monitor a large area are known as network-based system.

Network-based IDSs are intended to detect attacks that may be done against the system by monitoring network traffic. These systems detect attacks by analyzing network packets. Commercial IDSs are usually network based. Network-based IDS implementations should not be confused with firewalls. Firewalls only monitor various service/system accesses according to specific rule definitions. It does not check the contents of packages. However, IDSs check the content of network packets. Within this structure, a network-based IDS is used especially detection of attacks that occur because of the weakness in the network such as SYN Flood and TCP port scanning.

Server-based IDSs operate to detect malicious attacks made to a server computer. Malicious activities can be detected by collecting and analyzing data such as system and event logs on the relevant computer and operating system account trails. With regard to the operation carried out in a new record added, a decision is made as "attack" or "normal activity". In the process of making this decision, the IDS signature database is used. Server-based IDSs run on a server computer, they cannot recognize and analyze the entire network structure they are installed on. However, they are faster than network-based systems and can detect some attacks which cannot be detected by network-based IDSs. The Remote-to-Local (R2L) and User-to-Remote (U2R) attacks can be given as a sample for such attacks.

Application-based IDSs are considered as a subset and a smaller version of server-based IDSs in some studies [8]. They are included as application software on the server computer. Application-based IDSs that analyze events with this application software use log records generated by the application as information source.

3. PROPOSED HYBRID APPROACH TO DETECT WEB ATTACKS

In this study, various applications have been developed to detect current web attacks using real-time network data. The applications have been implemented and tested to be used on both Linux-based servers and all web applications running on Windows-based servers. The software developed for web application servers, running on Windows operating system, was written by C# and ASP.NET programming languages.

For Linux operating system servers, PHP-IDS based web applications, especially for WordPress based web applications, have been realized. Applications can monitor incoming and outgoing network traffic via an optional user-selectable network adapter and all web requests made over the HTTP protocol.

The Windows-based application, which basically used the TCP/IP protocol, was developed using SharpPcap and PacketDotNet ActiveX objects in the C# console environment. The Linux server based application was developed with PHP language using PHP-IDS. With the developed applications, IP packets in the network traffic are captured. The user requests contained in the TCP packet and the requests to be received to the web server in real-time are interpreted, and web attacks are detected with the help of a rule based expert system. For example, SQL injection attacks can be detected by filtering the keywords that are analyzed in Table 1, which can be used in SQL injection attacks.

In this case, the application is an example of a rule-based and anomaly-based hybrid intrusion detection system that can detect real-time web attacks. Intrusions that can be detected are attacks such as XSS, CSRF, Ldap Injection, Xpath Injection, Header Injection, Directory Traversal, RFI/LFI, DoS/DDoS, and SQL Injection. These are considered the most common web vulnerabilities offered by OWASP.

In attacks such as DoS/DDoS, real-time anomaly analysis are needed. In the applications developed, anomaly detection and misuse detection methods were used together. Figure 7 shows the flowchart of misuse detection method, and Figure 8 shows flowchart anomaly detection method.

RESEARCH ARTICLE

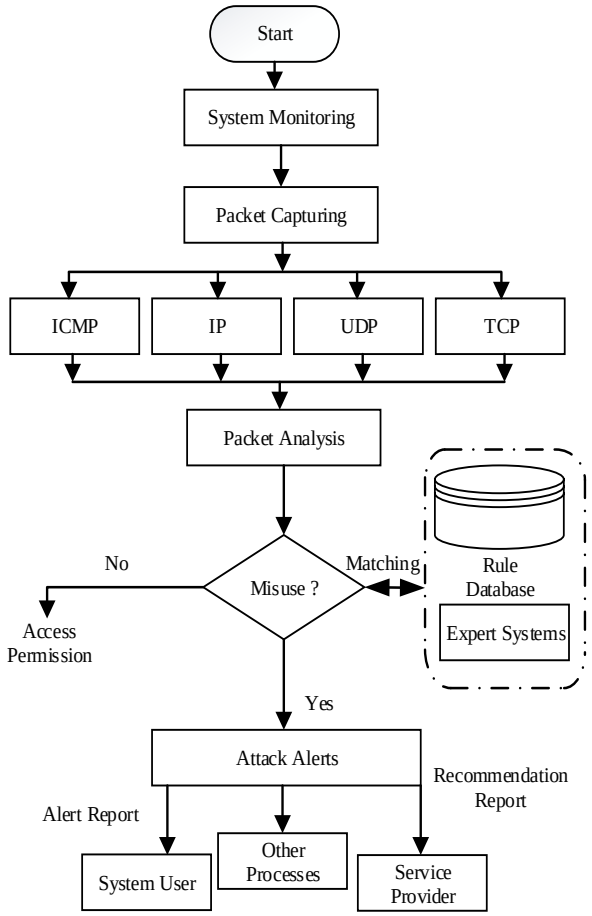

Figure 7 Flowchart of Misuse Detection Method

The words that can be used in an SQL injection attack are listed in Table 1. Each of these words is separated by a comma. These words are compared to requests in the contents of the network packets from the selected network interface in the application. Matching records in this comparison result are perceived as an attack. These attack records are listed on the console screen in red texted color. In Figure 9, packet 7 is sent from the client address "192.168.2.3". When the request address is considered, it is understood that the "Having" SQL word and SQL injection attack attempt have been made.

In this section, information about the current web vulnerabilities detected by the implemented applications, is given briefly and the results of the applications are presented in section 4.

3.1. SQL Injection

Today, almost all of the professional web applications use databases. These online applications communicate with the database via SQL, a structured query language. SQL Injection can be defined as the manipulation of SQL queries created by user input from web applications [11]. In data-driven web applications where users interact, the information in the database tables is filtered according to certain conditions and transferred to the application interface using queries or parameters. These result values are presented to the user or the manager in certain formats according to the design of the application. The SQL injection method is performed exactly while performing these operations. Attackers can perform SQL injection attacks by adding malicious code to the web browser address bar or to the login controls found in the application. Information that is not publicly available, but obtained in this way, can be important and confidential. The attacker can gain access to other information in the database by providing different dimensions for the SQL injection scenario, using the gained information about the system and the database. By using the information the attacker gained, later the attacker can achieve his goal [11]. An example of an SQL injection attack that can be used in web applications is given below.

Figure 9 shows a sample administrator form used in a web application. The SQL query that results when the administrator logs in correctly through this form is as below.

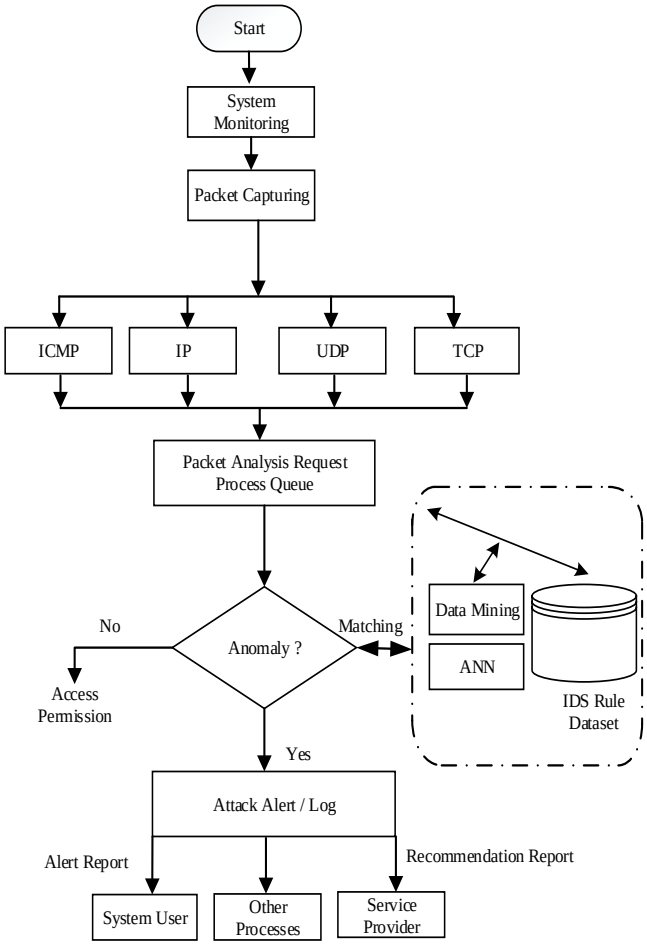

Figure 8 Flowchart of Anomaly Detection Method

RESEARCH ARTICLE

SELECT * FROM Users WHERE Username = 'admin' AND Password = '1234'

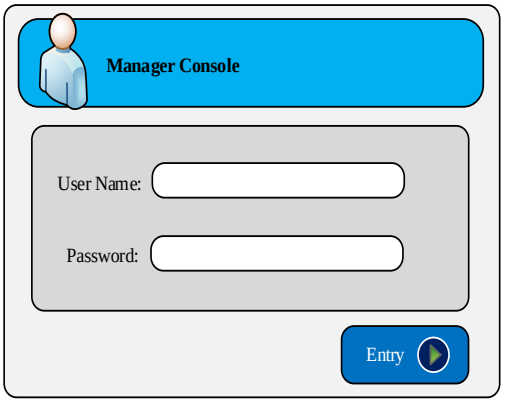

Figure 9 A Sample Entry Form

The attacker can attempt to attack the user name and password fields in such input forms by using 'OR 1=1--' statement given in Table 1, which can be used for SQL injection. The SQL query that results from this attack attempt is as below.

SELECT * FROM Users WHERE Username = 'admin' AND Password = ''OR 1=1--

So after this code injection, the attacker will have gained access to the system successfully.

SQL Injection
*', \, --, ' --, --';, ' ;, = ', = ;, = --, \x23, \x27, \x3D \x3B', \x3D \x27, \x27\x4F\x52 SELECT *, \x27\x6F\x72 SELECT *, 'or select *, admin'--, ';shutdown--, <>\'%;)(&+, ' or ''=', ' or 'x'='x, \ or \x\=\x, ') or ('x'='x, 0 or 1=1, ' or 0=0 --, \ or 0=0 --, or 0=0 --, ' or 0=0 #, \ or 0=0 #, or 0=0 #, ' or 1=1--, \ or 1=1--, ' or '1'='1'--, \' or 1 --'\, or 1=1--, or%201=1, or%201=1 --, ' or 1=1 or ''=', \ or 1=1 or \\=\, ' or a=a--, \ or \a\=\a, ') or ('a'='a, \) or (\a\=\a, hi\ or \a\=\a, hi\ or 1=1 --, hi' or 1=1 --, hi' or 'a'='a, hi') or ('a'='a, hi\) or (\a\=\a, 'hi' or 'x'='x';, @variable, ,@variable, PRINT, PRINT @@variable, select, insert, procedure, limit, order by, asc, desc, delete, update, distinct, having, truncate, replace, like, handler, bfilename, ' or username like '%, ' or uname like '%, exec xp, exec sp, '; exec master..xp_cmdshell, '; exec xp_regread, t'exec master..xp_cmdshell 'nslookup www.google.com'--, --sp_password, \x27UNION SELECT, ' UNION SELECT, ' UNION ALL SELECT, ' or (EXISTS), ' (select top 1, '||UTL_HTTP.REQUEST, 1;SELECT%20*, to_timestamp_tz, tz_offset, <>"%;)(&+, '%20or%201=1, %27%20or%201=1, %20$(sleep%2050), %20'sleep%2050', char%4039%41%2b%40SELECT, '%20OR, 'sqlattempt1, (sqlattempt2), \|, %7C, *\|, %2A%7C, *(\|(mail=*)), %2A%28%7C%28mail%3D%2A%29%29, *(\|(objectclass=*)), %2A%28%7C%28objectclass%3D%2A%29%29, %28, %29, %26, !, %21, ' or 1=1 or ''=', ' or ''=', x' or 1=1 or 'x'='y, //, //*, */*, @*, SELECT @@VERSION, SELECT * from v$version;, union, delete, alter, having*

Table 1 SQL Injection Keywords

Table 2 shows a sample of captured network packet contents. As a result of the analysis of the network packets obtained in real-time, as seen in Table 2, malicious attacks can be detected by the methods given in the flow charts in Figure 7 and Figure 8.

A captured packet
GET /deneme/en/DuzenlemeKurulu.aspx HTTP/1.1 **Host:** 192.168.2.2 **Connection:** keep-alive **Accept:** text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8 **User-Agent:** Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/31.0.1650.63 Safari/537.36 **Referer:** http://192.168.2.2/deneme/en/Kurullar.aspx **Accept-Encoding:** gzip,deflate,sdch **Accept-Language:** tr-TR,tr;q=0.8,en-US;q=0.6,en;q=0.4 HTTP/1.1 200 OK **Content-Type:** text/html **Content-Encoding:** gzip **Last-Modified:** Mon, 06 Jan 2014 19:43:50 GMT **Accept-Ranges:** bytes **ETag:** "1f1ea59f17bcf1:0" **Vary:** Accept-Encoding **Server:** Microsoft-IIS/8.0 **X-Powered-By:** ASP.NET **Date:** Tue, 14 Jan 2014 23:03:57 GMT **Content-Length:** 3523

Table 2 A Sample Packet Content

3.2 XSS

XSS (cross site scripting) is a type of attacks in the form of a code run between websites. This attack is carried out by running scripting codes in areas that do not contain the control structure on the site. This script can be composed of HTML or JavaScript codes. It is generally written by JavaScript code. However, these attacks can also be done with scripts written by VBScript, Ajax, ActiveX-like languages supported by web browsers. Since the script code is executed between HTML tags, this HTML injection is used instead of XSS [13].

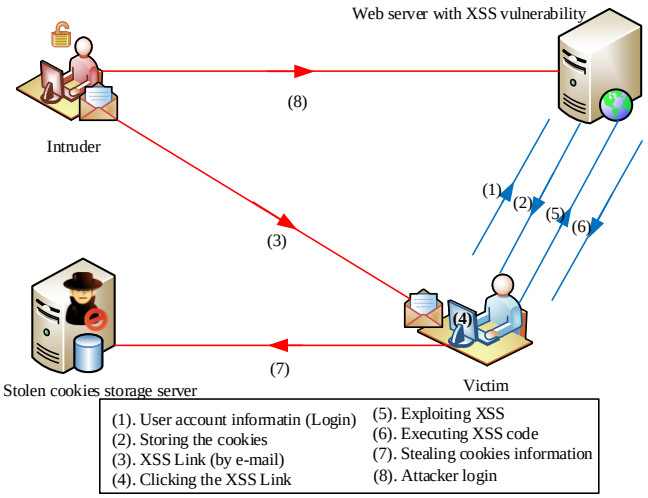

Figure 10 A Sample XSS Attack Scenario

With XSS attack, websites session cookies can be seized, web browser can be directed to the desired address, and DoS attacks can be done. The XSS attack is shown in Figure 10 with an example scenario. Here, when the victim logs in to the website that has the XSS vulnerability, the session information will

RESEARCH ARTICLE

normally be uploaded to the machine. In step 3, the attacker sends the link of a malicious XSS script via e-mail. When the victim runs the XSS script, the server will be interacted with by running the code and the victim's session information will be stolen.

3.3 Cross Site Request Forgery

CSRF means cross site request forgery and also called sleeping giant, is similar to XSS attacks. CSRF is a type of attack that is carried out by ensuring that any Internet user is able to perform actions on his/her behalf in the application used. Transactions such as sending money from Internet bank accounts and sending e-mails are among the activities that can be used by CSRF. The scenario in Figure 11 is illustrating the CSRF attack through a sample scenario. For example, when a user logs in to a site named "website.com" (step 1), the user receives login information from the web site (step 2), and after logging in, when the user clicks "saldirgan.com/index.html", the registration information on website.com can be changed even if the user is not aware when the user clicks on the attack link (step 3).

With this type of attack, different processes can be performed depending on the type of web application. For example, if it is a shopping site, new products can be added to the shopping cart. In addition, the user is walking through the site specified in step 3 as if it were a normal site, which makes the user unaware that it contains an attack link. These domain names are also given as an example.

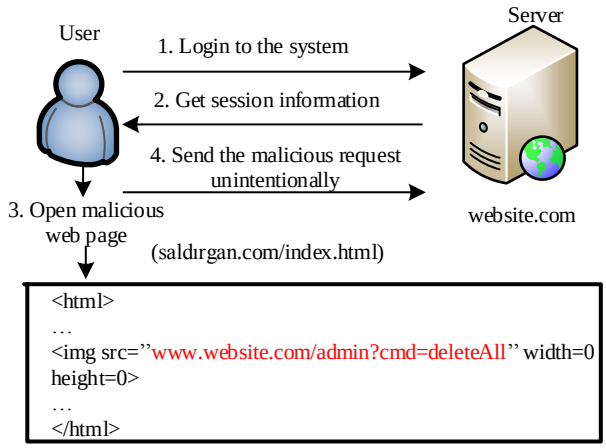

Figure 11 CSRF Attack Scenario

3.4 LDAP Injection

LDAP (Lightweight Directory Access Protocol) is a protocol used for directory management. LDAP is a kind of directory service standard. LDAP injection refers to attacks against web applications where the data is stored as LDAP statements. A sample LDAP attack is given in Figure 12. As it can be seen in Figure 12, in the query-response traffic between the client and the LDAP server, data that on the target system can be obtained using the parameters added to the queries.

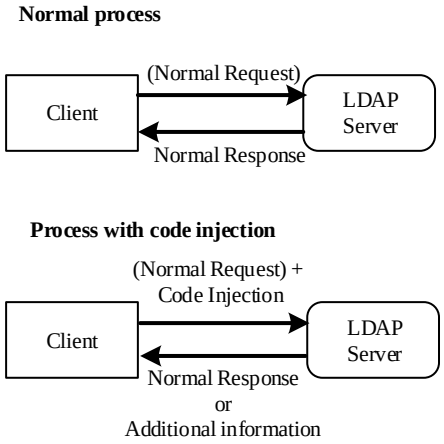

Figure 12 LDAP Injection

3.5 Header Injection

This type of attack exploits vulnerabilities by manipulating the HTTP header. In particular, the PHP language contains vulnerabilities that allow making a random header injection via URLs. By adding random headers, attackers can perform CSRF, XSS, HTML-injection, and other attacks.

3.6 Remote File Inclusion / Local File Inclusion

RFI/LFI are the type attacks that exploit vulnerability which is called file inclusion. These attacks are often encountered in web applications written in the PHP language. In this type of attacks, the attacker can run both local and remote code. LFI refers to code execution by adding files locally, and RFI to executing code by adding files remotely.

RFI/LFI attacks are caused by the fact that web applications coded in PHP language, there is no value assigned to the defined variables or the values that are assigned but not filtered. The code part below can be given as an example of this situation.

```
<?php
$page = $_GET[page];
include($page);
?>
```

Basically, with this program part, the aim is *“*index.php?page=index.php*”*. But here, it developed a structure using the GET method with variable named *page* without performing any control. If any attacker is aware of the inadvertent coding there, then the attacker can run a shell like *“*index.php?page=malicious.php*”* to navigate through the

RESEARCH ARTICLE

server and edit the files. In Figure 13, a sample can be seen as a scenario of RFI/LFI attack.

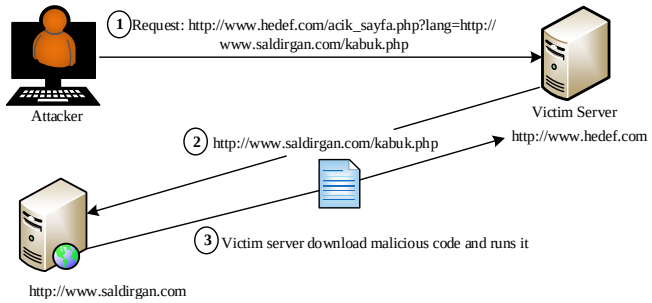

Figure 13 RFI/LFI Attack

3.7 Denial of Service

Denial of service (DoS) is a type of cyber-attack that aimed to make a server or a system unavailable with a large number of requests. As it can be understood from this definition, DoS attack is a different type of cyber-attack compared to other intrusions such as unauthorized access, password cracking, etc. The only purpose of this type of attack is to make the target system unreachable.

DoS attacks are divided into groups. “The ping of death” attack is an intrusion made by exploiting protocol errors. In this type of attack, a single large ICMP echo message is sent so that the system cannot respond. Another type of DoS attack is when a packet is sent to the TCP SYN packet with the same source and destination address.

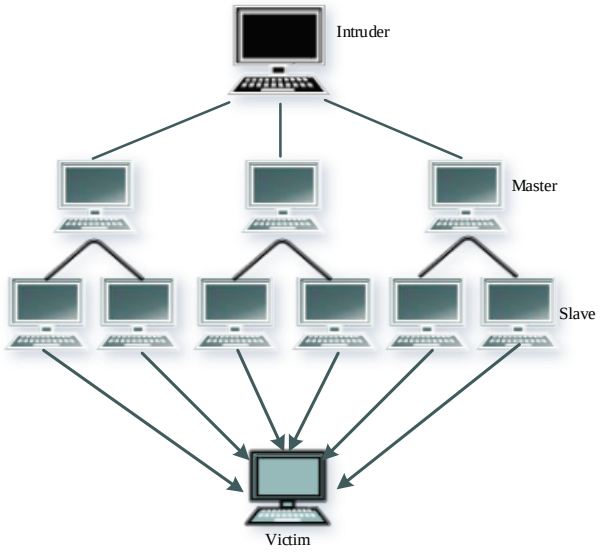

Figure 14 DDoS Attack

These are attacks that use the exploits of protocols, which can be carried out with a single packet or with less packets. The attacks in the other group are based on continuous request. Both the server machine and the network are busy. For example, this could be making a persistent connection request to a server. The attack can be from a single machine, or it can be done with many machines compromised over the network called zombies. Reflector, which can be any server [10], is also used if a zombie is not being used. Figure 14 shows the logical structure of a distributed denial of service attacks.

3.8 Directory Traversal

In this type of attack, the attacker can access some directories without any permission. In this attack, the attacker can send a URL containing a set of invalid Unicode UTF-8 code that has been consciously created by the IIS server. Therefore, the server can be moved out of an index and the desired scripts can be run. The working structure of directory traversal attack is given in Figure 15.

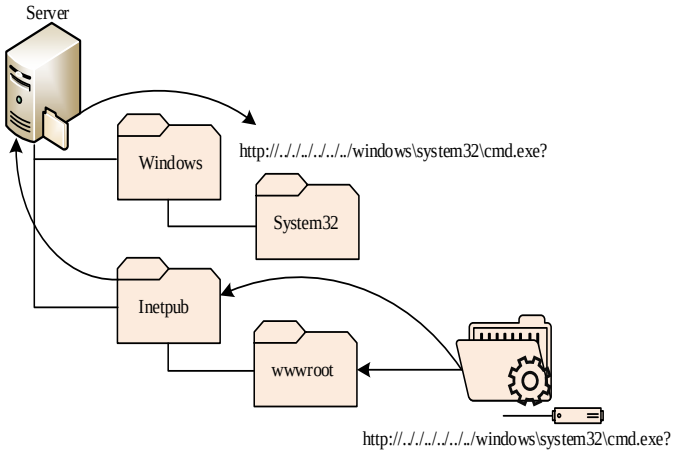

Figure 15 Directory Traversal Attack

4. APPLICATION RESULTS

In this section, the results of the implemented application methods are given. The proposed method has an alarm mechanism as seen in the flow charts given in Figures 7 and 8 in the previous sections. This alarm mechanism may differ according to the system in which IDSs are used. Figure 16 shows the applications developed in ASP.NET based web application and the capture of a sample SQL injection attack. In this application, red colored text and audible alert are used in the console environment for the detected attack.

Besides, the results related to web sites using PHP IDS structure and WordPress based web applications of this study are given in this section. Figure 17 shows the detection of the XSS attack, referred to as cross-site scripting. Figure 18 shows the real-time results of the SQL injection. Figure 19 shows RFI / LFI attacks, Figure 20 shows denial of services attacks, Figure 21 shows directory traversal attacks, and also Figure 22 shows CSRF attacks based on PHPIDS rules on WordPress sample systems.

RESEARCH ARTICLE

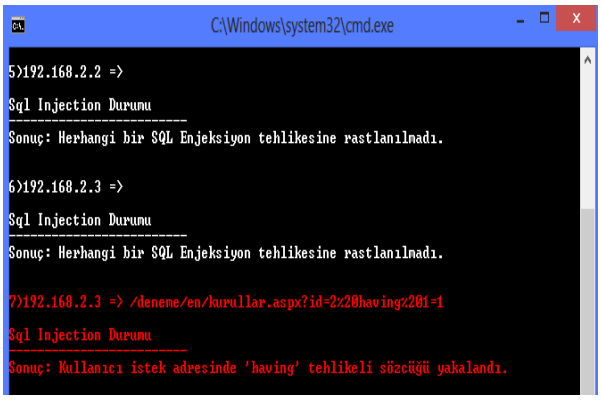

Figure 16 Screenshot of SQL Injection Detection

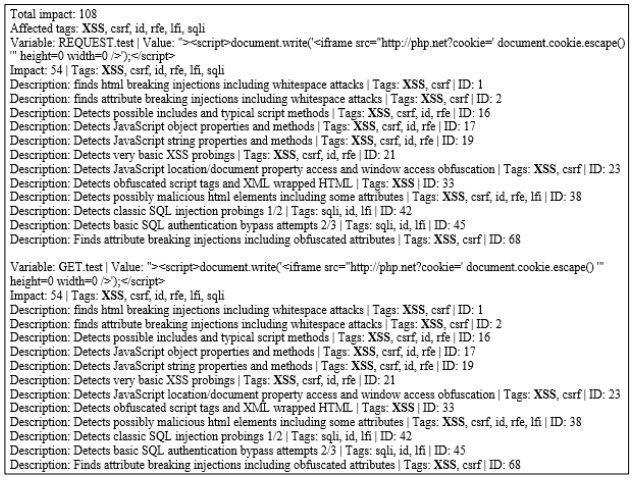

Figure 17 Screenshot of XSS Detection

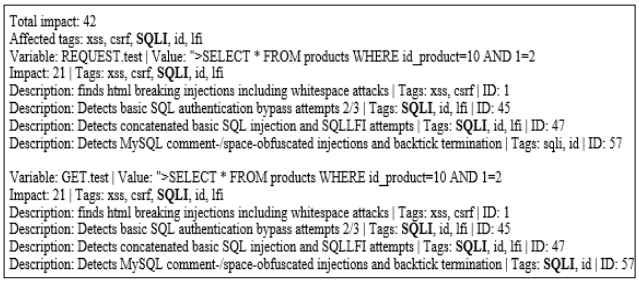

Figure 18 Screenshot of SQL Injection Detection

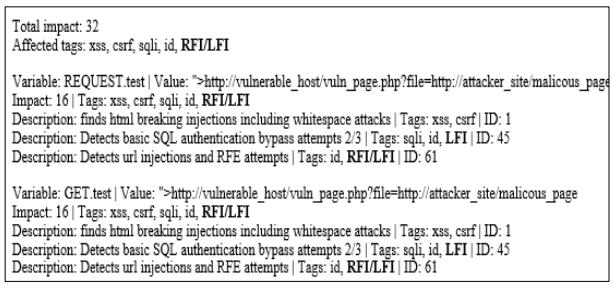

Figure 19 Screenshot of RFI/LFI Attack Detection

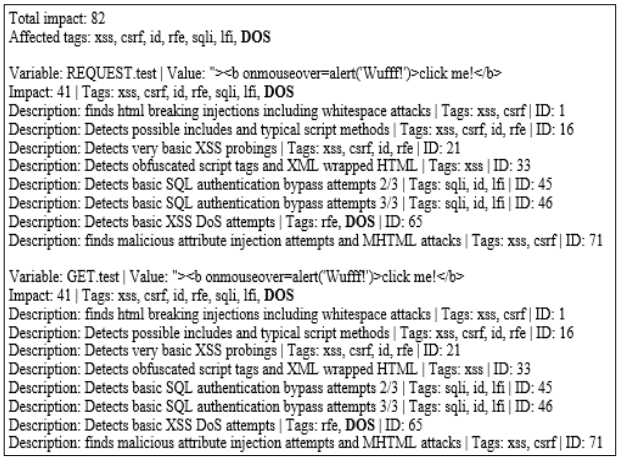

Figure 20 Screenshot of DoS Attack Detection

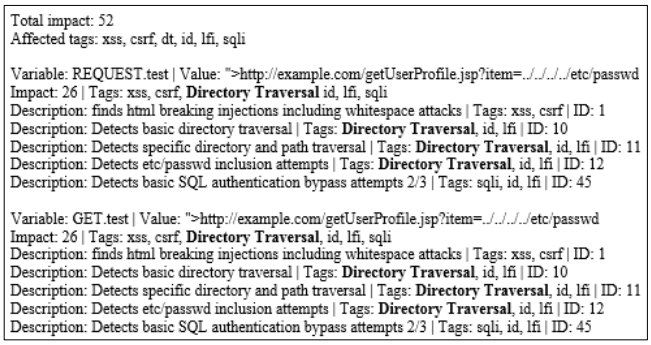

Figure 21 Screenshot of Directory Traversal Attack Detection

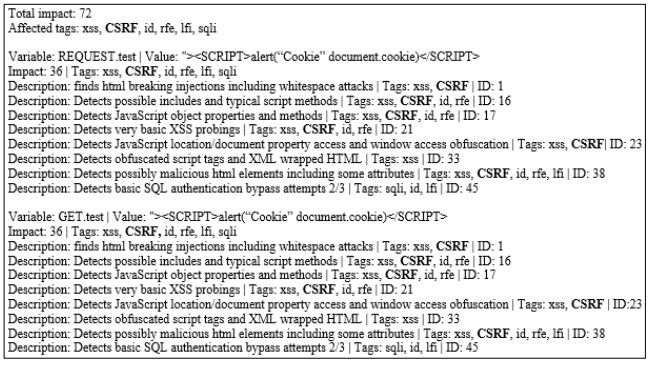

Figure 22 Screenshot of CSRF Attack Detection

Figure 23 shows the performance of WordPress-based applications with IDS and without IDS. There are five specification points to compare the applications. These specifications are the total load time, site load time, full capacity runtime, plugin load time, and theme load time. As it can be seen in Figure 24, web applications were compared according to some features, such as without installing IDS plugin, IDS-idle, IDS-active, cached and no cached. When the IDS plugins are installed and actively worked, the total load time and full capacity runtime features affected more than the

RESEARCH ARTICLE

other features. Because these plugins try to monitor and detect the attacks.

According to the analysis, when the intrusion detection plugins are installed, the total load time, site load time, full capacity runtime, plugin load time, and theme load time have obviously increased. This increase has slowed the system down a little, but it has led to performance improvement in terms of security. In this study, to get some improvements according to the run time, some IDS applications have been tried. The Expose plugin, one of the tried IDS applications is compared with PHPIDS, as shown in Figure 24. The reason behind exploring the effect of including expose plugin is to inspect the performance in terms of the runtime. As shown in Figure 24, the performance of the needed runtime has been improved considering Expose compared to PHPIDS.

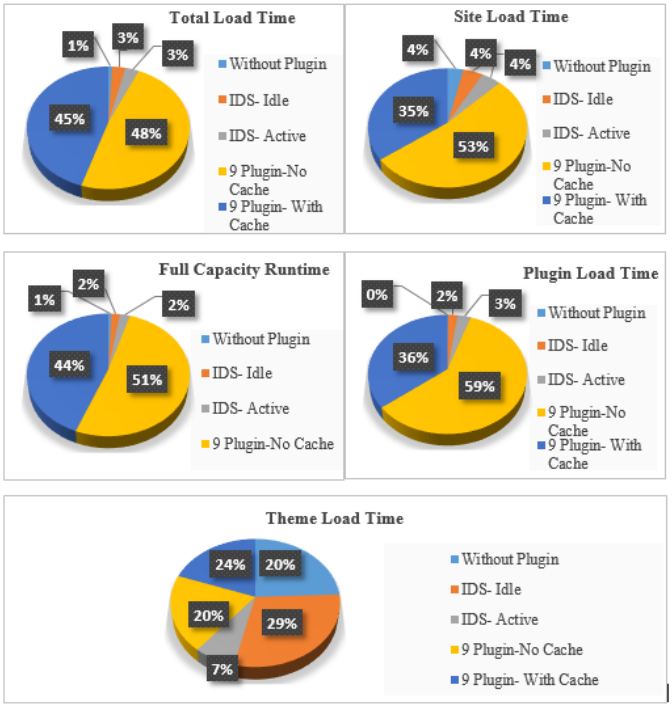

Figure 23 Performance Test Results

For the real-time analysis, firstly 'composer' which is package manager used for PHP, was installed. Then the 'Expose' files were exported to the main directory folder and the required libraries were downloaded via the composer. After this downloading, a certain set of attacks were given as an array. The test was performed by defining attack on each value of the array. In rule database, there are many attack rules. As seen in Figure 24, test results show how Exposé and PHPIDS rate each threat. The rate of attacks can be obtained by running 'phpidsTest' and 'exposeTest' program. For example, according to rule number '4', PHPIDS impact rate is '51' and the impact rate of Expose is '9'.

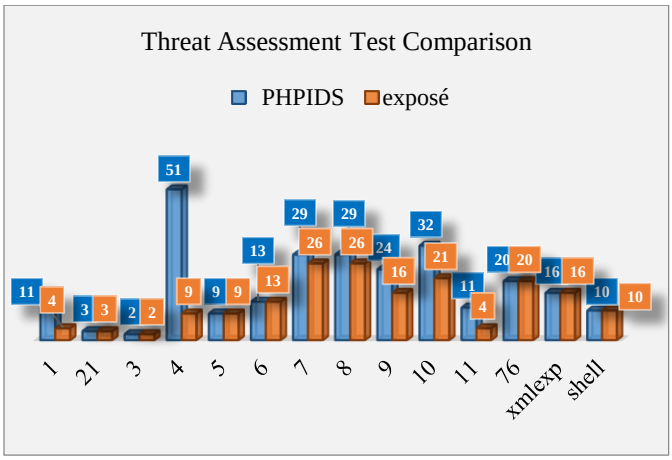

Figure 24 Comparison of a Threat Assessment PHPIDS and Expose

Attack Type	Accuracy	
	Number of Event	Number of True Positive
XSS	48	43
SQL Injection	22	21
RFI/LFI	34	32
DoS	41	40
Directory Traversal	52	45
CSRF	36	36

Table 3 Number of Events and Obtained Accuracy over a Period of Time (hourly)

Figure 25 shows the total impact rate of Expose and PHPIDS. In addition, Figure 25 presents the total spent time when the analysis of the attack array given in Figure 24. In the developed application, we focused on sensitivity, selectivity, and performance. High sensitivity rate is better for high rate to detection of attacks. Higher selectivity is better for separation of threat risk. These two features directly affect the performance. PHPIDS and Exposé have different levels of sensitivity. It means, the threshold level should be lower for the same detection level. The systems produce more noise when they have lower impact threshold. Finally, both will have a hit performance, but PHPIDS more so due to the higher sensitivity.

Exposé is a great tool for PHP based intrusion detection. It is up to the latest PHP coding standards, can be easily integrated with existing systems, and shows excellent results. If impact thresholds a bit lower, it can be get roughly the same detection with some performance gain. The IDS plugins such as PHPIDS and Expose have a great monitoring, logging and reporting features of the intrusion attempt.

Table 3 shows the number of events and obtained accuracy over a period of time. Table 3 presents the results were obtained

RESEARCH ARTICLE

from real-time analysis. 'Number of Event' field shows the count of the attacks during that time. And 'Number of True Positive' field shows the count of the correctly detected attacks. According to the analysis as given in Table 3, accuracy of the developed system can be calculated easily as seen in Figure 26. The developed applications have averagely 94% percent success rate.

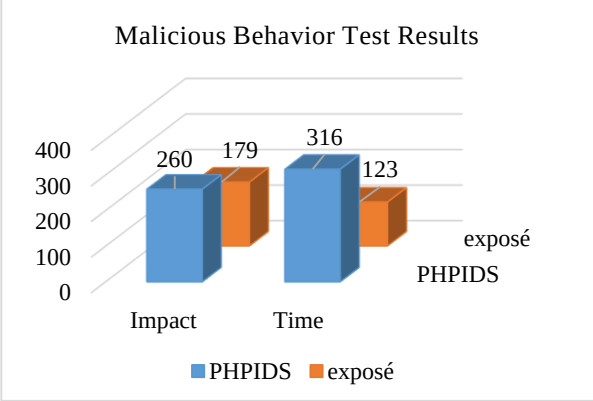

Figure 25 Test Results of Malicious Behavior

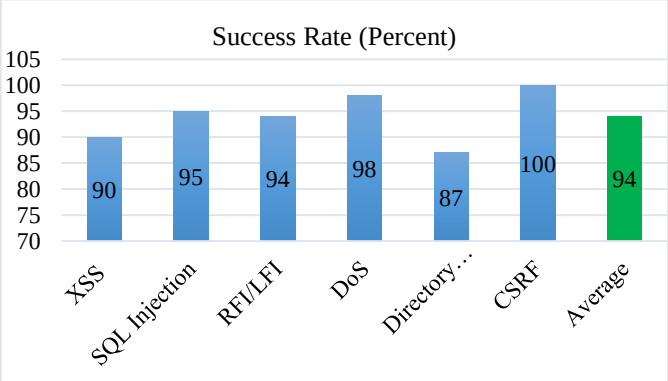

Figure 26 Performance Results of the Developed Applications

5. CONCLUSION

Nowadays, information security systems have great importance in terms of both institutional and individual aspects. The ability to store information on digital storage, be securely accessed, and be continuously available are essential features of information security systems. Different structures and tools can be used to provide security in information systems and to protect these systems against intrusion attacks that may occur internally and externally. One of the most attractive structures that can be used for this purpose is intrusion detection and prevention system.

In this study, intrusion detection systems were investigated in details, research on the logic of the study was carried out and information about the developed IDS structures in the literature up to date was given. In addition, real-time intrusion detection and prevention systems have been developed and evaluated by analyzing network traffic in real-time. This system will assist in detecting many kinds of attacks.

The developed software can detect attacks such as SQL Injection, XSS, CSRF, Header Injection, LDAP Injection, Directory Traversal, RFI/LFI, DoS, and Brute Force according to the analysis of packets' requests. This is done by capturing and analyzing the network traffic in real-time. This intrusion detection and prevention system software is an IDPS that operates in a hybrid structure and in real-time, which provides a well-structured rule-based and anomaly detection approaches together.

The developed software has 100% success rate in detecting attacks using the rule-based misuse detection approach. In the case of DoS-like attacks that work according to the anomaly detection, the performance would be less. This is due to the fact that the system has to adapt itself to many parameters according to the changes in network traffic conditions.

IDSs are systems intended for capturing attacks but they can provide protection at a certain level. Other protection systems such as a firewall cannot replace IDS's place. Defining too many rules in intrusion detection systems can lead to serious problems. Producing too many alarms in IDSs can cause critical issues to be overlooked. When IDSs are used in conjunction with firewall and other protective products, they form an indispensable security element for an organization. In practice, 100% security is impossible. The combined use of IDS and other security tools aims at identifying possible new types of attacks and ensuring the highest possible safety rate.

Existing IDSs still need some improvements. For example, the fact that the address in IP packet can be faked which creates a problem to IDSs. Likewise, encrypted packets cannot be processed by IDSs. In addition, the analytical module has limited ability to analyze resource information collected during intrusion detection. This is caused by the memory bottleneck. An IT expert who monitors the system will alert when abnormal behavior is detected, but cannot tell where the attack is coming from. In order to respond to this issue, the network access should be granted only to authorized parties. If more information is available, the IT expert may demonstrate a defensive approach to prevent future attacks.

One of the most important problems in IDSs is the generation of a large number of false positive alarms. These false alarms are increased in congested networks. Intrusion detection systems provide comprehensive defense to identify the attacker, and means for information mining and network attacks. However, IDSs can lead to loss of business and revenue by accidentally generated alarms.

IDSs are used in many different current areas. Honeypot systems, SDN, cloud computing, and IoT are some of these areas. Different technologies can be used to solve problems

RESEARCH ARTICLE

such as the high false alarm level that IDS owns. Honeypots are specially created systems for analyzing methods used in attacked systems. Honeypot-based systems can be used to detect zero-day attacks and generate signatures. Thus, IDSs may be able to detect unknown new attacks more easily.

Software Defined Network (SDN) architecture provides new opportunities to implement security mechanisms in terms of unauthorized activities detection. Datasets provided by a virtual SDN environment can be used in machine learning methods to detect unauthorized activities. Therefore, SDN can be a useful methodology for an IDS/IPS, due to its capability to both mirror the network traffic and block the malicious flow as soon as the IDS notifies the controller.

Cloud computing is a one of the new areas in information technologies. Cloud computing provides a number of benefits, especially in commercial aspects. With these benefits, this model has emerged as new security issues. There are many studies on improving a variety of security mechanisms for security threats, vulnerabilities and insufficiencies in cloud computing. It has been seen that host-based intrusion detection system can be used for securing the virtualization and virtual machine in cloud environment.

The developed intrusion detection and prevention system software will be able to operate on real-time data and detect all known attack types. As a direction for further research in this field, it is thought that the implemented software in this work will be served in cloud computing environments. In addition, by examining the honeypot systems, the performance of the anomaly detection approach can be increased by reducing the false alarm level in the IDS in different architectural structures in the literature.

REFERENCES

[1] Baykara, M., Daş, R., Karadogan, İ., "Bilgi Güvenliği Sistemlerinde Kullanılan Araçların İncelenmesi", 1st International Symposium on Digital Forensics and Security, pp. 231-239, 20-22 May. 2013, Firat University, Elazığ-Turkey.

[2] Razzaq, A., Hur, A., Masood, M., Latif, K., Ahmad, H.F., Takahashi, H., "Foundation of Semantic Rule Engine to Protect Web Application Attacks," 10th International Symposium on Autonomous Decentralized Systems (ISADS), pp. 95-102, 23-27 March 2011.

[3] Lounis, O., Bouhouita Guermeche, S.E., Saoudi, L., Benaicha, S.E., "A new algorithm for detecting SQL injection attack in Web application," Science and Information Conference (SAI), pp. 589-594, 27-29 Aug. 2014.

[4] Liang Guangmin, "Modeling Unknown Web Attacks in Network Anomaly Detection," Third International Conference on Convergence and Hybrid Information Technology, ICCIT '08, vol.2, no., pp. 112-116, 11-13 Nov. 2008.

[5] Ludinard, R., Totel, E., Tronel, F., Nicomette, V., Kaaniche, M., Alata, E., Akrout, R., Bachy, Y., "Detecting attacks against data in web applications", 7th International Conference on Risk and Security of Internet and Systems (CRiSIS), pp. 1-8, 10-12 Oct. 2012.

[6] Zolotukhin, M., Hamalainen, T., Kokkonen, T., Siltanen, J., "Analysis of HTTP Requests for Anomaly Detection of Web Attacks," 12th International Conference on Dependable, Autonomic and Secure Computing (DASC), IEEE, pp. 406,411, 24-27 Aug. 2014.

[7] Takcı, H., Akyuz, T., & Sogukpınar, İ., "Web Atakları İçin Metin Tabanlı Anormallik Tespiti (Wamtat)", Journal of The Faculty of Engineering and Architecture of Gazi University, Vol: 22, No: 2, pp. 247-253, 2007.

[8] Sağıroğlu, Ş., Güven, E.N., Yavanoğlu, U., "Zeki Saldırı Tespit Sistemi Tasarımı ve Gerçekleştirilmesi", Journal of The Faculty of Engineering and Architecture of Gazi University, Vol:26, No:2, pp. 325-340, 2011.

[9] Sancak, S., "Saldırı Tespit Sistemi Tekniklerinin Karşılaştırılması", Master Thesis, Gebze Technical University, 2008.

[10] Baykara, M., "Design and Implementation of Intrusion Detection and Prevention Approaches for Information Systems", Ph.D Thesis, Fırat University, Graduate School of Natural and Applied Sciences, Department of Software Engineering 2016.

[11] Demirol D., Daş R., Baykara M., "SQL Enjeksiyon Saldırı Uygulaması ve Güvenlik Önerileri", 1st International Symposium on Digital Forensics and Security, 20-22 Mayıs 2013, Firat University, Elazığ.

[12] İnternet: Ar, İ., "Nüfuz Tespit Sistemleri", http://anibal.gyte.edu.tr/hebe/AblDrive/59669005/w/Storage/104_2010_2_673_59669005/Homeworks/lktan-ar-nufuz-tespit-sistemleri.pdf, (Access Date: 10.03.2017).

[13] Vural, Y., Sağıroğlu, Ş., "Kurumsal Bilgi Güvenliği ve Standartları üzerine bir İnceleme", Journal of The Faculty of Engineering and Architecture of Gazi University, Vol: 23, No: 2, pp. 507-522, June 2008.

[14] Vural, Y., Sağıroğlu, Ş., "Kurumsal Bilgi Güvenliğinde Güvenlik Testleri ve Öneriler", Journal of The Faculty of Engineering and Architecture of Gazi University, Vol:26, No:1, pp. 89-103, 2011.

[15] Özhan, E., Paket ve Port Analizi İle Ağ Saldırı Tespit Sistemleri, Master Thesis, Trakya University, Graduate School of Natural and Applied Sciences, 2006.

[16] Sazli, H., M., Tanrıkulu, H., "Saldırı Tespit Sistemlerinde Yapay Sinir Ağlarının Kullanılması", XII. Türkiye'de İnternet Konferansı, 8-10 Kasım, Ankara, 2007.

[17] Huang at all, "A Multi-Agent-Based Distributed Intrusion Detection System", 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), 2010.

[18] Canbek, G., Sağıroğlu, Ş., "Bilgisayar Sistemlerine Yapılan Saldırılar ve Türleri: Bir İnceleme", Erciyes University Journal of Institue of Science and Technology, 23(1-2), pp. 1-12, 2007.

[19] Patcha, A., Park, J.M., "An overview of anomaly detection techniques: Existing solutions and latest technological trends", Computer Networks, 51(12): pp. 3448-3470, 2007.

[20] Anderson, J.P., "Computer Security Threat Monitoring and Surveillance", Technical Report, James P. Anderson Co., Fort Washington, PA. 15 April 1980.

[21] Arıs, A., Oktug S. F. and Yalçın, S. B. Ö., "Nesnelerin Interneti Güvenligi: Servis Engelleme Saldırıları Internet-of-Things Security: Denial of Service Attacks", 2015.

[22] Öğretmen, F. D., Aydın, M. A. and Ahmet Sertbaş., "Saldırı Tespit Sisteminin Bulut Bilişimde Kullanımı ve Etkileri", ISC-Turkey, 30-31 October 2015.

[23] Yavuz, G., Bektaş, O., Soysal, M., and Yiğit, S., "Sanal Ipv6 Balküpü Ağı Altyapısı: Kovan", National IPv6 Conference 2011.

[24] Baykara, M., Das, R., "A survey on potential applications of honeypot technology in intrusion detection systems", International Journal of Computer Networks and Applications, 2(5), pp. 1-9, 2015.

[25] Lobato, A. G. P., da Rocha Figueiredo, U., & Duarte, O. C. M., "An Architecture for Intrusion Prevention using Software Defined Networks.", Universidade Federal do Rio de Janeiro-GTA/COPPE-Rio de Janeiro, Brazil.

[26] Raza, S., Wallgren, L., Voight, T., "SVELTE: Real-time intrusion detection in the Internet of Things. Ad hoc networks", 11.8: pp. 2661-2674, 2013.

[27] A. A. Gendreau and M. Moorman, "Survey of Intrusion Detection Systems towards an End to End Secure Internet of Things," IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud), Vienna, pp. 84-90, 2016.

RESEARCH ARTICLE

Authors

Muhammet Baykara was born in Elazig, Turkey. He received his BS and MSc. in Computer Engineering from Firat University in 2006, 2009 respectively. He received his Ph.D. in Software Engineering from Firat University in 2016. Currently, he is a research assistant in the Department of Software Engineering at Firat University. His research interests are Information Security, Honeypots, Intrusion Detection and Prevention Systems.

Resul Das received his B.Sc. and M.Sc. in Computer Science from Firat University in 1999, 2002 respectively. Dr. Das received his Ph.D. degree from Electrical and Electronics Engineering Department at the same university in 2008. He is currently an Associate Professor at the Department of Software Engineering of Firat University, Turkey. He has authored several papers in international conference proceedings and refereed journals, and has been actively serving as a reviewer for international journals and conferences. His current research interests include complex networks, computer networks, web mining, knowledge discovery, and information and network security.