Jurnal Ilmiah KURSOR
Menuju Solusi Teknologi Informasi

Vol. 9, No.4, Desember 2018

ISSN 0216 – 0544
e-ISSN 2301– 6914

MITIGATION HANDLING OF SQL INJECTION ATTACKS ON WEBSITES USING OWASP FRAMEWORK

[a]Imam Riadi, [b]Rusydi Umar, [c]Wasito Sukarno

[a,b,c]Department of Information System, Universitas Ahmad Dahlan
Jl. Prof. Dr. Soepomo, S.H., Janturan Warungboto, Umbulharjo, Yogyakarta 55164 Indonesia
E-mail: imam.riadi@is.uad.ac.id, rusydi_umar@rocketmail.com, wasito1508048006@webmail.uad.ac.id

Abstract

The development of the security system on the application of a website is now more advanced. But software that has vulnerabilities will threaten all fields such as information system of health, defense, finance, and education. Information technology security issues will become a threat that made managers of the website (web admin) alerted. This paper is focused on how to handle various application web attacks, especially attacks that use SQL Injection, using The Open Web Application Security Project (OWASP), the aim is to raise awareness about application security web and how to handle an occurred attack. OWASP is a non-profit organization that focuses on web application security. OWASP provides security resources so that everyone can improve website security. the existence of security holes on the website is very vulnerable to being broken with dangerous character. to prevent it from being able to periodically replace the user name and password. Testing can be done to mitigate the security gap in the SMS broadcast application service by updating the filter character in such a way that the attacker can be minimized. Mitigation is done by limiting characters to enter making it difficult for attackers.

Keywords: SQL Injection, OWASP, Security, Website

.

.

INTRODUCTION

The development of rapid information technology can't be separated from the Internet, to handle the information that is up to date every second, the website is necessary. Websites can contain text, pictures, or videos. The more varied and interesting a website will become a management for users all over the world to access it. The number of developed websites that did not follow by a good security system could have vulnerabilities that are not known by the admin or manager of the website [1]. The same impact could happen on broadcast SMS website the Bureau of Student Affairs and Alumni.

The website is used to provide information to students. Because of the website, SMS broadcast Bureau of Student Affairs and Alumni of Ahmad Dahlan University is often used and very important, this website should be safe from attacks, especially SQL Injection. In today's world, SQL injection is a serious security threat on the Internet for various dynamic webs that are on the internet. Because the use of the internet for various online services is increasing, so are the security threats that exist on the web increasing. [2]. SQL Injection technique is well known in the world of hacking as one of the web hacking techniques that are destructive to the database of a site. The technique used in SQL injection is to enter standard commands in SQL (DDL, DML, DCL) such as create, insert, update, drop, alter, union and select along with other commands that are not foreign. [3]

To find out whether the website is completely safe, an attack simulation is conducted. SQL Injection is used to determine whether there is a security loopholes website or not. A website that has vulnerabilities will be vulnerable to attacks. If an attacker successfully attacked a website then it is a possibility that an attacker can manipulate the data. This study aims to find security holes in applications web SMS broadcast Bureau of Student Affairs and Alumni of Ahmad Dahlan University. This research uses a gray system theory. Gray system theory is used because this method does not require a lot of data. this study uses little data, 12 data samples. to prevent it from being able to periodically replace the user name and password.

RESEARCH METHOD

A website loophole can be detected by using SQL Injection, by entering certain characters in the login form might cause an attacker can get to the admin page and know the contents of the database, even extract, transform or remove it [4]. SQL Injection occurs when an attacker gives an SQL query command input to manipulate the query language so that the attacker gets database information [5]. SQL Injection attacks are very dangerous, attackers who master the database or have entered the database without permission can manipulate data on the database system, this might cause the injected website became unusable. Hacked data can be misused by irresponsible parties [6].

SQL Injection can be done in many ways one of them is by giving the character aims to inject a character such as string or quotation marks, exclamation points, or equal to, and the other characters to produce the condition is always true [7]. In SQL Injection attack the attacker can use malicious characters to be injected in the login form of the website application so that it can control the database. If the security system is good then the database can be recovered [8]. The character is shaped like quotation marks ('=,!) Injected into the login form. Other characters that can be used as OR1 = 1--, or'1 = '1'. Character is what used to attack websites login form [9]. SQL Injection attacks are used to take control of the system through the database. By leveraging the success of SQL injection attacker can enter into the website system without going through the login process and password [10].

The weakness of the website requires the security of related information on unrestricted databases thus allowing attackers to retrieve information data. Tests using SQL injection security hole could tell the difference before and after the applied patch to renew or update a malicious code to block SQL injection attacks, the patch that is used to improve the security of applications that require validation against a database [11].

Attacks using SQL Injection can be detected

but it is difficult to know the identity of the attacker and might not be tracked. Therefore, it is important to build a security system on a website [12]. Website's vulnerability to attack from SQL Injection can be prevented by renewing Password with the latest patches and updates security system otherwise website will be easily susceptible to attack [13].

Analysis handling attacks SQL Injection by Rudi Samuel Pardosi [14] can be done in the following way:

1. Perform an SQL Injection attack by entering a malicious character in the login form that is at the time of initializing in the programming code that retrieves data from the database.
2. Giving the code constraint input limitation, the input that makes the attacker can't inject long input in the login form.
3. Eliminate or hide the program code to resolve error messages that come out of the database. This research uses the OWASP Framework. The advantages of the OWASP Framework compared to other frameworks are simple approaches to calculating and assessing the risks associated with applications. wherewith this method can be decided what should be done to these risks. By knowing the risks that will occur, many benefits will be obtained including saving time and reducing the occurrence of more serious risks [15]. In The Open Web Application Security Project (OWASP) attack SQL Injection was ranked first, this can be evidenced by the release of the OWASP Top notes 10-2017[16] and can be seen in Figure 1.

OWASP Top 10-2013 (Previous)		OWASP Top 10-2017 (New)	
A1	- Injection	A1	- Injection
A2	- Broken Authentication and Session Management	A2	- Broken Authentication and Session Management
A3	- Cross Site Scripting (XSS)	A3	- Cross Site Scripting (XSS)
A4	- Insecure Direct Object Reference-Merged with A7	A4	- Broken Access Control (Original Category in 2003/2004)
A5	- Security Misconfiguration	A5	- Security Misconfiguration
A6	- Sensitive Data Exposure	A6	- Sensitive Data Exposure
A7	- Missing Function Level Access Control- Merged with A4	A7	- Insufficient Attack Protection (NEW)
A8	- Cross Site Request Forgery (CSRF)	A8	- Cross Site Request Forgery (CSRF)
A9	- Using Components with Known Vulnerabilities	A9	- Using Components with Known Vulnerabilities
A10	- Unvalidated Redirects and Forwards-Dropped	A10	- Underprotected APIs (NEW)

Figure 1. The level of security attacks OWASP

Figure 1 [16] shows the security attack on injection occupying the first level because often used to attack web applications. In Figure 1 the release notes OWASP Top 10-2017 has been updated regularly, namely:

1. Merger 2013-A4: Direct Object Reference Unsafe and 2013-A7 level control function is lost into 2017-A4: Damage control access.
2. New additional 2017-A7: Protection of attack that is not full.
3. New additional 2017-A10: No protected with an Application Programming Interface (API).
4. Eliminate 2013-A10: Redirects and Forwards are not validated.

Figure 2 [16] indicates a potential attacker to use a variety of techniques for entering applications. website dangerous database. Sometimes, these techniques can easily be found and exploited, otherwise sometimes can be difficult, as well as damage from a simple factor improved to an irretrievable.

Figure 3 [16] shows the update of the OWASP Top 10 focused on the identification of the most serious risk, for each risk there is general information about the likelihood and impact by using a simple grading scheme is based on the OWASP Risk Rating Methodology. For each application the possibility of no threat of attack and impact that makes a change. In the previous version focused on identifying common vulnerabilities that are designed based on risk. Risks in the Top 10 come from this type of attack, weakness, and its effects.

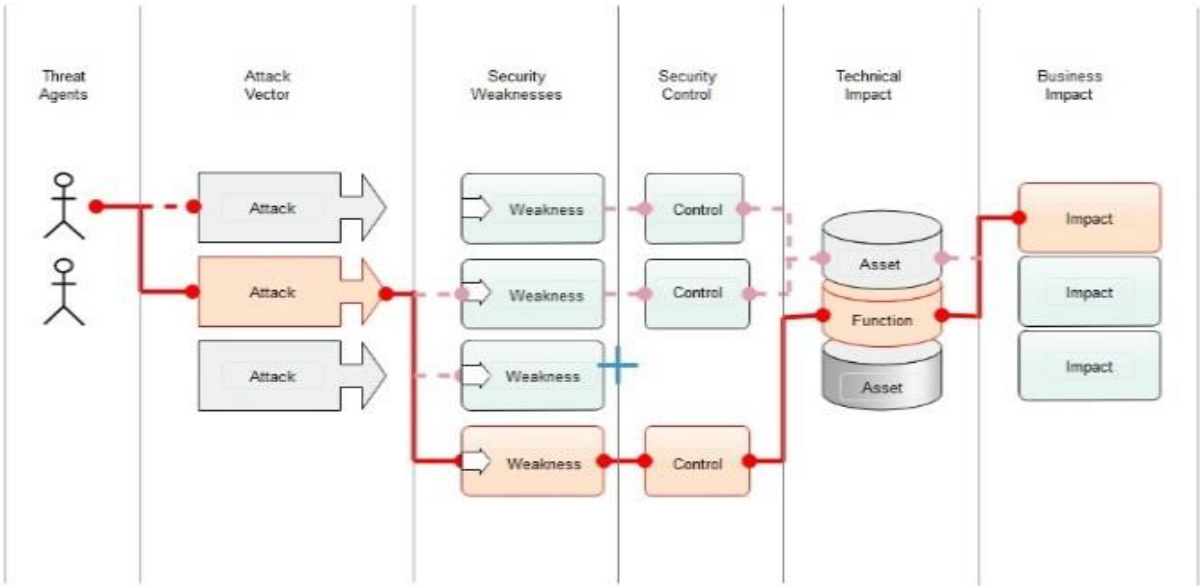

Figure 2. Application Security Risks In

Threat Agent	Attack Vector	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact
Application Specific	Easy	Widespread	Easy	Severe	Application/ Business Specific
	Average	Common	Average	Moderate	
	Difficult	Uncommon	Difficult	Minor	

Figure 3. Risk Assessment Methodology in the OWASP

History in surfing the browser where the user opens user id email or password will be stored on the hard drive or computer, otherwise, it is also stored in random access memory. This activity also includes accessing internet banking login, PayPal, Bitcoin, and Facebook. Login access is user id and password. Linux Extractor (LiME) memory can capture memory so that information obtained from random access memory can be completed and can be used as evidence in digital crime management and involves evidence of the Linux-based laptop operating system. Forensic Tool Kit (FTK) Imager can analyze digital evidence well because the data evidence of encrypted and unencrypted information can also be opened by these tools.[17]

Attacks on structured networks come from multiple sources and assemble to form a large packet flow is a type of Denial of Service (DoS) attack. This attack can disrupt the service on the target network by flooding the bandwidth or system capacity on processing to be able to make the target server network becomes overloaded. The tool used to detect DoS network attacks router and perform network traffic analysis is Wireshark. it can be concluded that attacks on Router analysis, starting from the attack process can be obtained information that the DoS attack can Ping or send data/messages can repeatedly and make the network Router down using DNS Flooding application. This application has characteristics in the research results and it can be proven that forensic investigators can succeed in using the Wireshark app to analyze DoS attacks on the Router[18].

Forensic networks, requiring traffic logs to analyze the activity of each computer connected to the network to be able to know what hackers

do. This requires the router information. Accessing router-related information such as RouterOS on Mikrotik devices can be used to maintain some data that uses the API in remotely accessing the router. Forensics on router-based OS devices Routers can be done with live forensics via the Media API. In the extraction of router data through the API can access information related to various activities on the network. The applications developed are the success of data from the router, Log Activity, IP Address List, ARP, Rent DHCP, RouterBoard Info, Users, and DNS Cache. The data used in observing network-based attacks is a scenario, the DNS Cache role has no correlation on the FTP Service for the case of attack scenarios. Analysis of linked links in each field of data acquisition variables greatly helps digital forensic investigators to determine the attack activity of the Network. To obtain forensic acquisition information should do so quickly before the Router is turned off or rebooted[19].

The threat of malicious networks for security on Web servers resulting in loss of bandwidth and overload for users and web servers of service providers is flooding. Flooding attacks on the network is by implementing an Intrusion Detection System (IDS) detection system such as Snort. An open-source system that can be used to detect flooding attacks using Snort's special rules. various activities will be recorded on Snort then stored in the log file to record all network traffic activity. Log files are used for investigation into forensic process modeling methods to find evidence. Results from the analysis of this study found that 15 IP addresses recorded perform illegal acts on the webserver. The IDS system that applies to this research has worked well as expected, the system can record all network activities in the form of log files with p. cap extension, the file can be analyzed with Wireshark. The analysis carried out, found 15 IP address of the webserver has done illegal acts, thus causing overload on the network traffic. With the forensic process, the IDS system on the webserver can help meet forensic needs, besides, administrators can monitor and prevent attacks[20].

Structured network attacks that originate from multiple sources and converge to form a large packet flow are the Distributed Denial of Service Attacks (DDoS). This attack to interfere with the service on the target network by flooding the target bandwidth or excessive load capacity on the target network server. The method of the network defense system on the Internet to avoid a DDoS attack is the classification of network packets. This classification is done by an Artificial Neural Network (ANN) method[21].

Cloud service applications that Cloud service providers offer, but most companies build private cloud computing. Cloud system violations may be an internal user or due to a configuration error or there may be flaws in the system. This research introduces the ADAM (Advanced Data Acquisition Model) method, which refers to the result of the ADAM investigation process, can also verify some parameters of successful investigation; the investigation by using ADAM in the future can work well and correctly. To identify weaknesses in the service system used its ownCloud user list from a group that can change the password of other users[22].

The ADAM (Advanced Data Acquisition Model) method is used for the private cloud computing service investigation process that has been successfully performed. The process in data acquisition of service can work either directly or by writing block acquisition per device so that the problems that occur are the mainstay of evidence as reliable digital data in court. In the misuse of XYZ hospital data against the dissemination of confidential data occurs due to system flaws, or misconfiguration, this can occur due to the misuse of policies on private cloud computing services.

Forensic Analysis and Prevention Cross-Site Scripting use an open web application framework (OWASP) of three important stages: Attack, Analysis, and Closing. The stages are as follows:

1. Stage Attack is by single victim method using OWASP Xenotix XSS Attack Exploit Framework v6.2 for Information Collection, keylogger, Download spoofer, and screenshot Webcam directly to the victim by using Mozilla Firefox browser.
2. Stage Analysis is performed using Live Forensic by Wireshark, HTTP Header and Tcpdump. The use of live forensic methods can certainly capture all types of attacks that occur such as payload, and scripts. results on analysis and script files. some hash value test files with apps to use in comparing file values

that are already downloaded by files on the victim's premises that have been stored on the server.

The Closing Phase is a process by patching the user's security gap by first installing the add-on on the extension in the Mozilla Firefox browser using the XSSFilterAde extension name. XSSFilter is available for early warning, turn off plugins, restrict, authorize payload/script to the victim when opening website address[23]

METHODOLOGY

The method by [24] can be described as follows:

1) Identify websites, Internet networks, Web servers.
2) Testing with attack SQL Injection to finding loopholes that can be penetrated by malicious code
3) attacks Analyze the results to find weaknesses in the website
4) report the results of the following research documentation and evidence of research.

This journal emphasizes the attacks carried out by the perpetrators of the crime through the security holes of the website, and successfully entered the website database so that the perpetrators of the crime can change or delete the database on the website that will harm the website owner.

The analysis in this research is to create an attack SQL Injection on the website SMS broadcast Bureau of Student Affairs and Alumni it is to determine whether there are security loopholes broadcast SMS Bureau of Student Affairs and Alumni so by knowing their security holes.

The test is conducted to prove the existence of vulnerabilities perforated so it can be known of the slits to shut it down immediately so that an attacker can't log back in using a unique character. Attacks trials SQL Injection that

Table 1. Characteristics of SQL Injection Used attacks directly to the target website SMS broadcast.

Testing Activities	**Injection Character**	**Expectation Of Testing Results**	**Testing Results From**
Open the site/web SMS broadcast BIMAWA	'OR1 = 1	can not enter the login form code injection because blockaded Managed Sign in with character unblocked injection	Failed
	'OR1 = 1--		Failed
	'or'1 = 1		Failed
	OR1 = 1--'OR1 = 1 #		Failed
	'or'1' = '1		Failed'
	or'1 '=' 1		Successfully
	'or a = a - 'or'a' = 'a	did not sign the form because of the character of the injection login blockaded	Failed failure'
	'OR1 = 1--		Failed'
	OR1 = 2 #		Failed
	'or'1' = '1 #		Failed

Bureau of Student Affairs and Alumni 172.10.23.161/SMS then insert the malicious code as shown in Table 1 below.

Implementation Test totals 12 times the only one who managed to enter the following test SQL Injection results. 12 (Twelve) times the tests, one of them managed to get in. The following shows an example of testing that did not work and that work.

Figure. 4. Display the front page of SMS broadcast

Figure 4 is a page views website SMS broadcast Bureau of Student Affairs and Alumni, these pages display the login form and password. In the user id field input SQL Injection characters while Password is emptied after input then press enter. Picture 3 has not entered a character so the display has not changed.

In Figure 5 login form SMS broadcast view Bureau of Student Affairs and Alumni gave input 'or1 = 1 for password deliberately emptied, the results obtained for the input characters above can't enter the menu page because it is blocked with the characters mentioned above. Existing display after given the input character 'or1 = 1 as in Figure 6 that is a warning that the character is not recognized by the system.

Figure 5. Page form login by input characters' OR1 = 1

Figure. 6. Results of input characters' OR1 = 1

The experiment was conducted 12 times, among the twelve experiments one could successfully enter the login form. this experiment can know the website SMS broadcast that has been injected dangerous characters there are security holes that can be exploited by the attacker to manipulate the existing data in the system website.

Figure 7 shows the login form page in the user id field entered character 'or'1' = '1. And in the password field is not given any input, only on the column id user only inputted after the login button pressed then the results obtained from the input characters above are as shown in Figure 7. In this experiment successfully entered by using the character 'or'1' = '1.

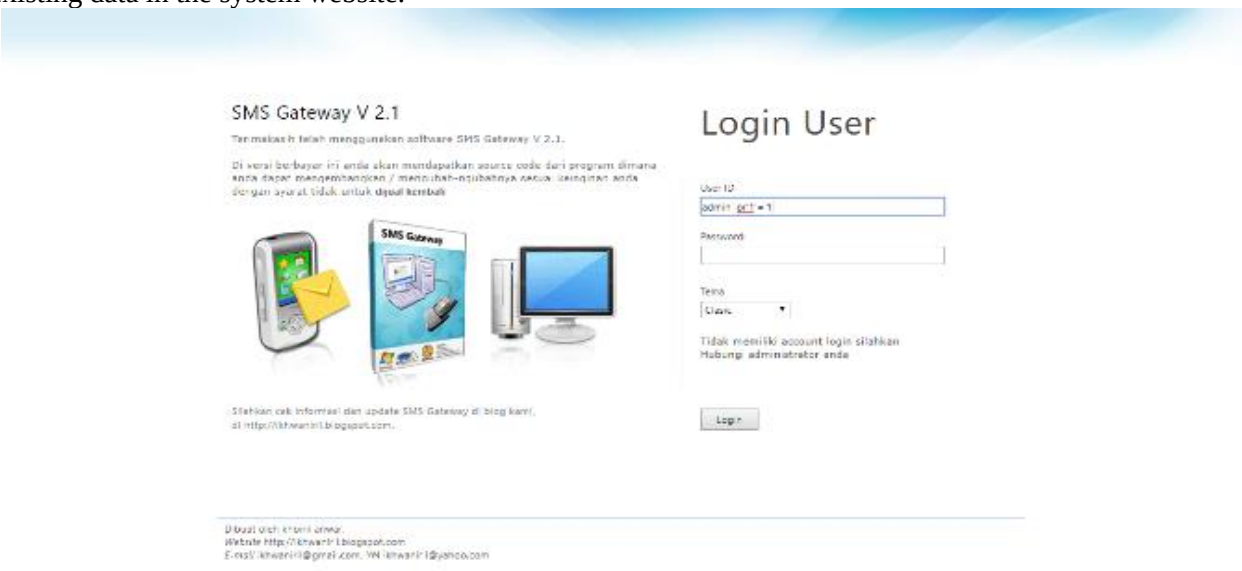

Figure 7. Page Feedback Form Login
Given Characters' or'1 '=' 1

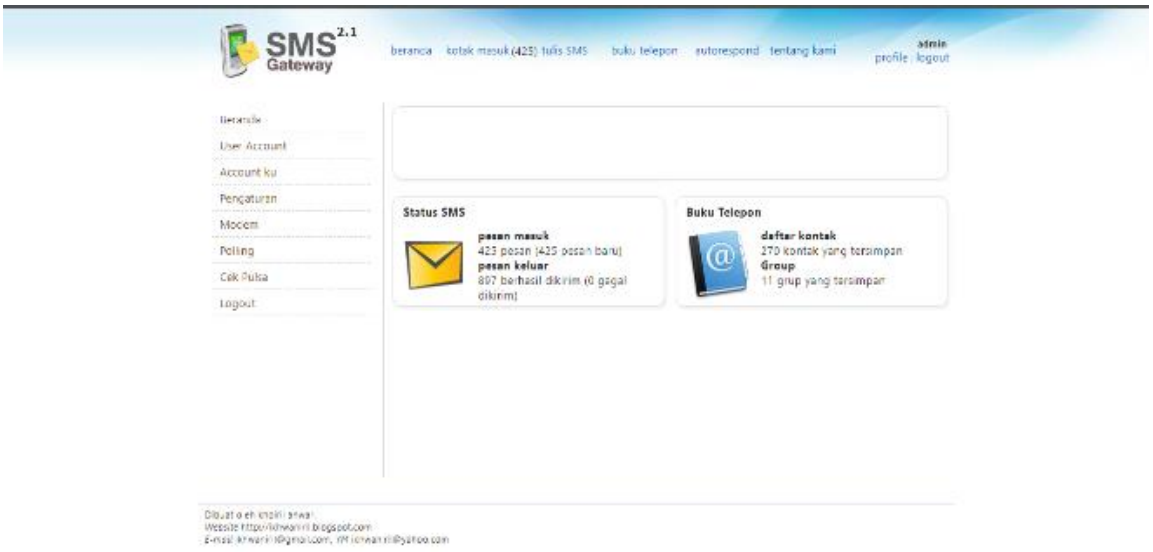

Figure 8. Results of input characters' or'1 '=' 1 testing 12

The result of input 'or'1' = '1 has shown successful entry as shown in figure 8. the above test successfully entered into the web admin page so that in this research it can be said that the security hole has been open for character above. The successful entry into the administrator page occurs due to validation error or not filtered malicious characters entered into the login form. The dangerous thing is the successful attacker entering without a password by entering characters or inputs as in the twelfth test and the attacker SMS broadcast BIMAWA or false information or false news that causes the receiver's loss of information.

Tests twelfth managed to get to a page web admin so in this study it can be said that the security hole has been open to the characters mentioned above. The success goes to page administrator was due to a validation error or not a filtered dangerous character is entered into the login form. It is causing a dangerous attacker is successful without a password by entering characters or inputs such as in testing the twelfth and the attacker does SMS broadcast Bureau of Student Affairs and Alumni or false information or false information that causes harm the recipient of the information.

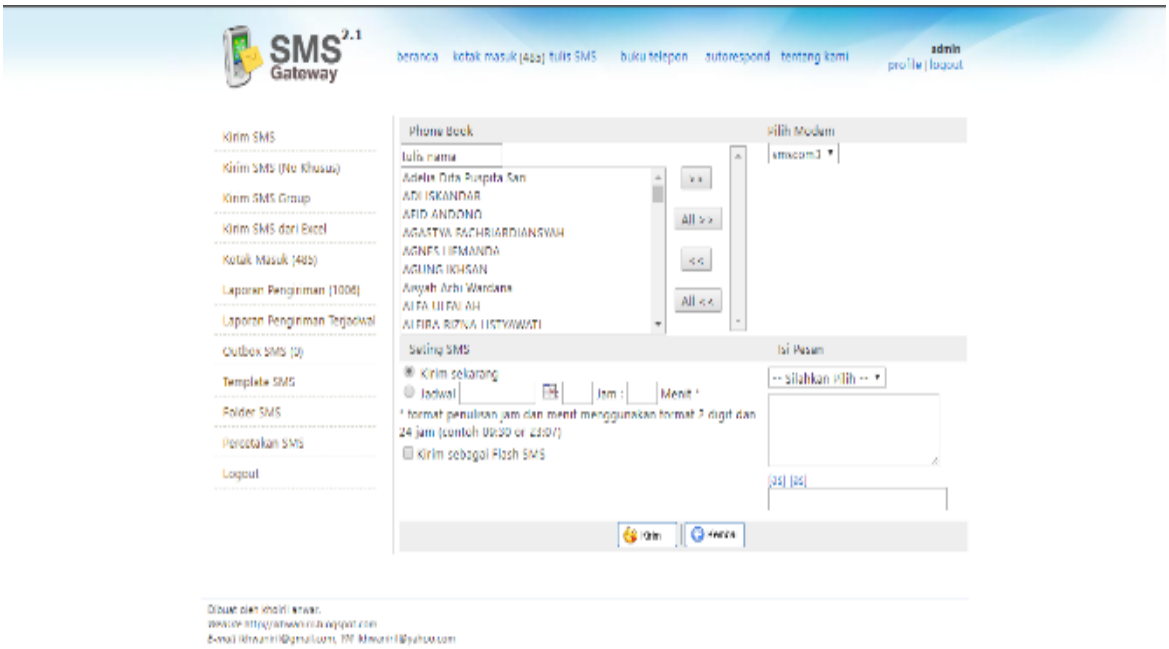

Figure 9. Page SMS Messages broadcast Bureau Of Students Affairs and Alumni To Conduct

Figure 9 is a page to the message carried by admin. Until this page, the attacker can perform a message to the destination number. An attacker can create fake messages or sending chain messages that cause users no phone intended to follow what is required or authorized by the attacker, who in this case the mobile phone number will think that that sends a message is admin though not the admin but the attacker's website SMS broadcast Bureau of Student Affairs and Alumni.

For typing SMS and no mobile phones are shown in Figure 10. The data on this page the attacker can write a message and no mobile phones are desirable. It is harmful to users no phone because he thought that sending messages is the admin of the SMS broadcast Bureau of Student Affairs and Alumni. At 12 attempts are being made to go to the website SMS broadcast Bureau of Student Affairs and Alumni one trial has made it into the login form. After learning that website there are vulnerabilities that are vulnerable to attackers who will be able to manipulate the data in the database SMS broadcast Bureau of Student Affairs and Alumni such as giving false message to the user or a student, then made an effort to close the gap so that the attacker can't enter it again. Trials to close the gap made in the official Bureau of Student Affairs and Alumni because Server SMS broadcast is in the room Bureau of Student Affairs and Alumni. Steps to be taken are as follows: Opening folder HTDOC on the server and then open the file PHP Login. with notepad

where the program code to enter the login form here. Inline 24, there is a user id where the source code is used to enter the login form.

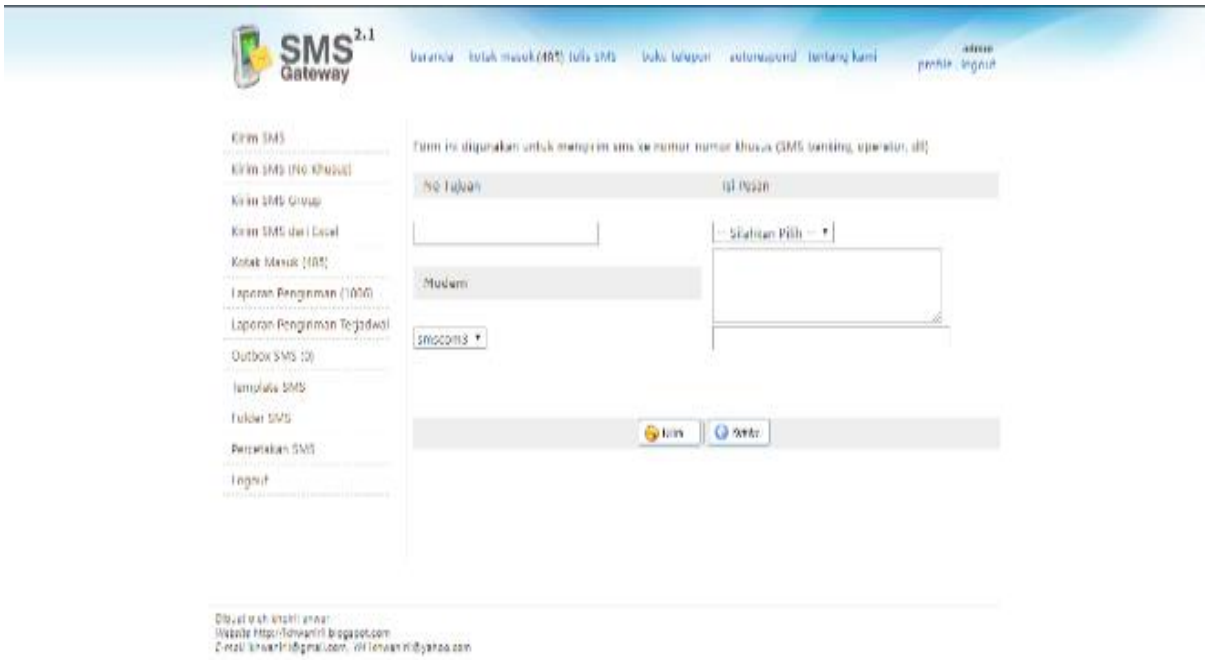

Figure 10. Broadcast SMS page Bureau of Student Affairs and Alumni to Send SMS The Characteristically Shipping Per No Mobile

```php
<?php
$handle = @fopen("conf.txt", "r");
if ($handle) {
    while (!feof($handle)) {
        $buffer = fgets($handle, 4096);
        $pieces = explode(" ", $buffer);

    }
    fclose($handle);
}

$user=$pieces[0];
$password=$pieces[1];
$db=$pieces[2];

$hostname_config = "localhost";
$database_config = "$db";
$username_config = "$user";
$password_config = "$password";
$config = mysql_pconnect($hostname_config, $username_config, $password_config) or trigger_error(mysql_error(),E_USER_ERROR);
mysql_select_db($database_config, $config);

$tema = $_POST['tema'];
$user_id = $_POST['no_anggota'];
$password = md5($_POST['password']);
$sql = mysql_query("select user_id,password,status FROM user WHERE user_id='$user_id' && password='$password'");
$jumlah=mysql_num_rows($sql);
$row=mysql_fetch_array($sql);

if($jumlah>0)
{
session_start();

$_SESSION['namauser']=$row[user_id];
$_SESSION['passuser']=$row[password];
$_SESSION['status']=$row[status];

$l=mysql_query("select Id from theme where aktif='1'");
$tl=mysql_fetch_array($l);

$up=mysql_query("update theme set aktif='0' where Id='$tl[Id]'");
$up1=mysql_query("update theme set aktif='1' where jenis='$tema'");
```

Figure 11. Code Programs on file PHP login broadcast SMS server BIMAWA

Figure 11 shows the time has not changed the program code on line 24, which is located in the log file HTDOC broadcast SMS in the Bureau of Student Affairs and Alumni server. Program code above causes the attacker can log in with the character of admin 'or'1' = '1.

```php
<?php
$handle = @fopen("conf.txt", "r");
if ($handle) {
    while (!feof($handle)) {
        $buffer = fgets($handle, 4096);
        $pieces = explode(" ", $buffer);

    }
    fclose($handle);
}

$user=$pieces[0];
$password=$pieces[1];
$db=$pieces[2];

$hostname_config = "localhost";
$database_config = "$db";
$username_config = "$user";
$password_config = "$password";
$config = mysql_pconnect($hostname_config, $username_config, $password_config) or trigger_error(mysql_error(),E_USER_ERROR);
mysql_select_db($database_config, $config);

$tema = $_POST['tema'];
$user_id = preg_replace('/[^a-zA-Z0-9]/','', $_POST['no_anggota']);
$password = md5($_POST['password']);
$sql = mysql_query("select user_id,password,status FROM user WHERE user_id='$user_id' && password='$password'");
$jumlah=mysql_num_rows($sql);
$row=mysql_fetch_array($sql);

if($jumlah>0)
{
session_start();

$_SESSION['namauser']=$row[user_id];
$_SESSION['passuser']=$row[password];
$_SESSION['status']=$row[status];

$l=mysql_query("select Id from theme where aktif='1'");
$tl=mysql_fetch_array($l);

$up=mysql_query("update theme set aktif='0' where Id='$tl[Id]'");
$up1=mysql_query("update theme set aktif='1' where jenis='$tema'");
```

Figure 12. The program code on the file PHP login on the Bureau of Student Affairs and Alumni broadcast SMS server has been added with the program code to filter the characters.

Figure 12 shows the code when the program has been changed in line 24 by adding on the user id
$ user_id = preg_replace ('/ [^ a-zA-Z0-9] /', '', $ _ POST ['no_anggota']);
In the above code preg_replace function is to replace the unwanted unique character as the characters are successfully used to enter login form 'or'1' = '1.

$ user_id = $ _ POST ['no_anggota'];

The program code above should be changed to close security gaps or attacker can't sign in using the characters have been used to enter the login form in the twelfth trial. Changes in the above program code are to add the program code.

For allowed into is like the source code above is only uppercase and lowercase letters from A to Z and then only the numbers from 0 to 9.

In this research has proven that website SMS broadcast in Bureau of Student Affairs and Alumni there are vulnerabilities that are vulnerable penetrated by attackers who would give information or fictitious message, and has closed the gap by adding code to filter unique character.

SQL Injection Attack appears as the main threat to the web application. The proposed solution for detecting SQLIA vulnerabilities in web applications is very large. Based on the Analyzer and dynamic tester well done to detect and block SQLIA, the response time is also very good compared to other tools. No need to change the source code of a web

application can use the minimum system resources. One advantage of the proposed solution is that it can handle advanced SQLIA techniques as a knowledge base to be updated in handling modern types of threats. The proposed solution uses an MS SQL analyzer to allow for detecting vulnerabilities and tagging pages. The detector needs to be improved so that all types of analysis can be configured for analysis. knowledge base using techniques and knowledge of various attacks.[25]

Intrusion detection system applies a learning vector quantization algorithm by applying a method of capturing data to the MySQL service port, converting data into ASCII code, extracting data into several alphanumeric features, punctuation, special combination and remaining character then processing that value into learning vector quantization algorithms so that you get an accurate SQL injection query pattern, the application enters text mode as process runs to capture and classify queries that go to the database. Evaluation at the level of accuracy is done by testing applications that use query data that varies to learning vector quantization algorithms when the application is installed on a network. By using parameters, the maximum accuracy of SQL injection detection applications reaches 80%.[26]

Based on the results of testing methods for securing internet access using VPN, SSH Tunneling, DNS over HTTPS, DNS over TLS, DNSCRYPT, and Tor can avoid the sensor system. The use of the VPN, SSH Tunneling and Tor destination address in the form of IP Address and Hostname are not detected by the sensor system while the use of the DNS over HTTPS, DNS over TLS and DNSCRYPT methods that are secured is DNS Queries even though the IP Address can still be tracked. The sensor system applied in Indonesia uses the Domain Name System filtering method that records negative addresses entered into the blacklist. There for the use of the DNS over HTTPS, DNS over TLS and DNSCRYPT methods will still escape the Indonesian government censorship system. DNS over HTTPS, DNS over TLS or DNSCRYPT were created to protect Man-in-the-Middle attacks by certain parties. So that the use of the DNS over HTTPS, DNS over TLS or DNSCRYPT method is to protect the insertion of malicious codes, protect from annoying advertisements, protect pornography, and so on. The installation of applications on secure internet access is not to avoid the sensor system.[27]

CONCLUSION

This research concludes that a security hole can be penetrated by giving input to login using a dangerous character. How to tell if on the web site there is a security hole or not ie by using SQL Injection. If a malicious character successfully escapes means the website is vulnerable to SQL Injection attacks. The existence of such vulnerabilities because the website has not been closed properly. SQL Injection can be used on other websites that have security holes. Attackers who can enter login form and successfully log in can manipulate data on the database, so it can harm the data on the website. A solution of this SQL Injection attack is to update patch, user name and password periodically.

The use of SQL Injection is used to determine the security hole and can detect threats to the SMS Broadcast web application so that the manager can immediately prevent it or immediately close the security hole in the web application.

REFERENCES

[1] A. Sagala, E. Manurung, B. Siahaan, and R. Marpaung, "Deteksi, Identifikasi Dan Penanganan Web Menggunakan SQL Injection Dan Cross-Site Scripting," *Inst. Teknol. Del*, vol. 2014, pp. 20–24, 2014.

[2] S. Lika, R. Dwi, P. Halim, and I. Verdian, "Analisa Serangan Sql Injeksi Menggunakan Sqlmap," *Jurnal Sistem dan Teknologi Informasi*, vol. 4, no. 2. pp. 88–94, 2018.

[3] F. widya Putra, "Analisis Keamanan Website Dari Serangan SQL Injection Menggunakan Web Application Firewall," *Skripsi Tek. Inform. S1 Univ. Pas. Bandung*, 2018.

[4] B. A. Harahap, H. Lubis, and T. M. Diansyah, "Penetration Testing Keamanan Web Menggunakan SQL Injection," *Biltek*, vol. 5, no. 70, pp. 1–5, 2015.

[5] T. R. Yudantoro, "SQL Injection pada Sistem Keamanan Database," *J. Teknol. Inf. dan Komunikasi. STMIK ProVisi Semarang*, pp. 89–93, 2013.

[6] M. Dahlan, A. Latubessy, M. Nurkamid, and

L. H. Anggraini, “Pengujian Dan Analisa Keamanan Website Terhadap Serangan SQL Injection (Studi Kasus : Website UMK),” *Jurnal Sains dan Teknologi UMK Kudus*, vol. 7, no. 1. pp. 13–19, 2015.

[7] J. O. Atoum and A. J. Qaralleh, "a Hybrid Technique for SQL Injection Attacks Detection and Prevention,” *Int. J. Database Manag. Syst. (IJDMS)*, vol. 6, no. 1, pp. 21–28, 2014.

[8] R. Ellysa, M. Husni, and A. Pratomo, “Pendeteksi Serangan SQL Injection Menggunakan Algoritma SQL Injection Free Secure pada Aplikasi Web,” *Tek. Pomits*, vol. 2, no. 1, pp. 1–6, 2013.

[9] I. Riadi and E. I. Aristianto, “An Analysis of Vulnerability Web Against Attack Unrestricted Image File Upload,” *Comput. Eng. Appl. J.*, vol. 5, no. 1, pp. 19–28, 2018.

[10] A. Lazzez and T. Slimani, “Forensics Investigation of Web Application Security Attacks,” *Int. J. Comput. Netw. Inf. Secur.*, vol. 7, no. 3, pp. 10–17, 2015.

[11] S. G. Nugraha, S. Djanali, and A. Pratomo, “Sistem Pendeteksi dan Pencegah Serangan SQL Injection dengan Penghapusan Nilai Atribut Query SQL dan Honeypot,” vol. 2, no. 1, pp. 1–5, 2013.

[12] K. Randhe and V. Mogal, “Security Engine for prevention of SQL Injection and CSS Attacks using Data Sanitization Technique,” *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 3, no. 6, pp. 5890–5898, 2015.

[13] V. C. Amit Chaturvedi, Shailendra Bagdi, “Analysis of SQL Injections Attacks and Vulnerabilities,” *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 6, no. 3, pp. 106–110, 2016.

[14] R. Pardosi, “Kalilinux Top Hacking.pdf.” pp. 43–46, 2015.

[15] B. Ghozali, K. Kusrini, and S. Sudarmawan, “Mendeteksi Kerentanan Keamanan Aplikasi Website Menggunakan Metode Owasp (Open Web Application Security Project) Untuk Penilaian Risk Rating,” *Creat. Inf. Technol. J.*, vol. 4, no. 4, p. 264, 2019.

[16] D. W. Jeff Williams, *OWASP Top 10-2017 rcl*, 1st ed. Maryland, Amerika, 2017.

[17] D. S. Yudhistira, I. Riadi, and Y. Prayudi, “Live Forensics Analysis Method For Random Access Memory On Laptop Devices,” *Int. J. Comput. Sci. Inf. Secur.*, vol. 16, no. 4, pp. 188–192, 2018.

[18] M. A. Zulkifli and U. A. Dahlan, “Live Forensics Method for Analysis Denial of Service (DOS) Attack on Routerboard,” vol. 180, no. 35, pp. 23–30, 2018.

[19] M. I. Mazdadi, I. Riadi, and A. Luthfi, “Live Forensics on RouterOS using API Services to Investigate Network Attacks,” *Int. J. Comput. Sci. Inf. Secur.*, vol. 15, no. 2, pp. 406–410, 2017.

[20] D. Mualfah and I. Riadi, “Network Forensics For Detecting Flooding Attack On Web Server,” *IJCSIS) Int. J. Comput. Sci. Inf. Secur.*, vol. 15, no. 2, 2017.

[21] I. Riadi, A. W. Muhammad, and Sunardi, "Neural network-based DDoS detection regarding hidden layer variation," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 15, pp. 3684–3691, 2017.

[22] N. Widiyasono, I. Riadi, and A. Luthfi, “Investigation on the services of private cloud computing by using ADAM Method,” *Int. J. Electr. Comput. Eng.*, vol. 6, no. 5, pp. 2387–2395, 2016.

[23] A. Kurniawan, I. Riadi, and A. Luthfi, "Forensic analysis and prevent of cross-site scripting in single victim attack using open web application security project (OWASP) framework," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 6, pp. 1363–1371, 2017.

[24] M. Dahlan, A. Latubessy, and M. Nurkamid, “Analisa Keamanan Web Server Terhadap Serangan Possibility Sql Injection Studi Kasus: Web Server UMK,” *Pros. SNATIF*, vol. 0, no. 0, pp. 251–258, 2015.

[25] R. Muhammad, R. Muhammad, R. Bashir, and S. Habib, “Detection and Prevention of SQL Injection Attack by Dynamic Analyzer and Testing Model,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 8, pp. 209–214, 2017.

[26] A. S. Irawan, E. S. Pramukantoro, and A. Kusyanti, “Pengembangan Intrusion Detection System Terhadap SQL Injection Menggunakan Metode Learning Vector Quantization,” *J. Pengemb. Teknol. Inf. dan Ilmu Komputer. Univ. Brawijaya*, vol. 2, no. 6, pp. 2295–2301, 2018.

[27] D. Hariyadi, M. R. Jinan, N. S. Bayuaji, and A. S. Hasan, “Analisis Jaringan Pada

Aplikasi Pengamanan Akses Internet," *Cybersecurity Dan Forensik Digit.*, VOL. 2, NO. 1, PP. 16–23, 2019.