Securing Web Application by Embedded Firewall at Gytech Indosantara Mandiri Ltd.

Muhammad Yusup[1] , Maisyaroh[2] , Laila Septiana[3]

[1] Informatics Engineering Department; STMIK Nusa Mandiri Jakarta, Jl. Kramat Raya No. 18 Senen, Jakarta Pusat, 10420, telp (021) 31908575; e-mail: yusupmuhammad04@gmail.com
[2] Computer Engineering Department; Universitas Bina Sarana Informatika, Jl. Kamal Raya No.18, Ringroad Barat, Cengkareng, Jakarta Barat; e-mail: maysaroh.msy@bsi.ac.id
[3] Information Systems Department; STMIK Nusa Mandiri Jakarta, Jl. Kramat Raya No. 18 Senen, Jakarta Pusat, 10420, telp (021) 31908575; e-mail: laila.lsp@nusamandiri.ac.id

* Correspondence: e-mail: maysaroh.msy@bsi.ac.id

Received: 17 January 2020; Revised: 31 January 2020; Accepted: 14 February 2020; Available Online: 20 March 2020

Abstract

Gytech Indosantara Mandiri Ltd. in the last few years experienced many Cybercrime attacks on the Web Server which caused many moral and material losses. Therefore, it is necessary to consider ways to fight and prevent attacks on the webserver. One way to fight and prevent attacks is to use the Attack Signatures method by using ModSecurity and fail2ban as a Web Application Firewall (WAF). ModSecurity is used to detect and prevent the occurrence of Cyber Crime in the Http and https services. Whereas Fail2ban is used to prevent Bruteforce attacks on ssh, FTP and telnet services. Modesecurity, which acts as a Web Application Firewall (WAF) will send logs to Fail2ban when exploits occur on the Web Server. Meanwhile, Fail2ban will block the Attacker's IP address so that both can be used as a Web Application Firewall or can be used as layer 7 network security.

Keywords: Fail2ban, ModSecurity, Web Aplication Firewall (WAF)

1. Introduction

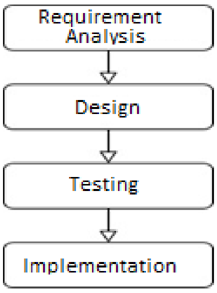
Gytech Indosantara Mandiri Ltd. is a Mecanical Engineering company. Currently the web server used is based on cloude service on AWS, Alibaba and Digital Ocean, but the web server is known to have not implemented security for the web service, so when an attack or exploitation occurs, it will cause fatal problems. Data loss and user trust are some of the negative impatcs. Therefore, the security of web services used by websites or mobile apps is something that must be considered on an ongoing basis. ModSecurity and Fail2ban are technologies that will be used in dealing with existing problems in the company, where a Web Application Firewall will be built that is able to prevent the possibility of exploitation on the web server or the existence of new information security holes in the web service.

Hacking is an intrusion activity into a computer system or network with the aim to abuse or damage the existing system (Amarudin dan Ulum, 2018). Therefore, the current study proposed and implementing network security supported by Web Application Firewall (WAF) with the Attack Signatures Method to detect and prevent exploitation of the web server PT. Gytech Indosantara Mandiri by creating a Core Rules Set or known as SecRules in ModSecurity to filter every request.

2. Research Methods

The research collected data using four stages: 1). Requirements Analysis for conducting an analysis of web attacks that have occurred on the website of Gytech Indosantara Mandiri Ltd. that has been running, then determine using the WAF with the appropriate Attack Signatures method and then collect data on how to create a network using ModSecurity and Fail2ban as WAF. 2). Design phase in which the authors propose to build ModSecurity and

Fail2ban on public cloude loadbalancer of Gytech Indosarana Mandiri Ltd. and make private cloud on each production server so that the production server can only be accessed by the devops team or using VPN and the user always requests for the loadbalancer first so that this implementation is more optimal and does not waste resources. 3). Testing stage. This proposed concept should have no obstacles because the application is the same as building a loadbalancer that has been previously applied to several Gytech Indosantara Mandiri Ltd. servers, if using WAF is only installed on Loadbalancer. After the topology design stage, the testing phase is carried out to determine the implementation system is running in accordance with the objectives that have been analyzed. The results of the topology implementation will be simulated with Virtual box 6.0. 4). Implementation, to assist in identifying Network Security problems by using the Modsecurity and fail2ban Attack Signatures method at Gytech Indosantara Mandiri Ltd., as well as to manage network security. Figure 1 shows the framework of the research method.



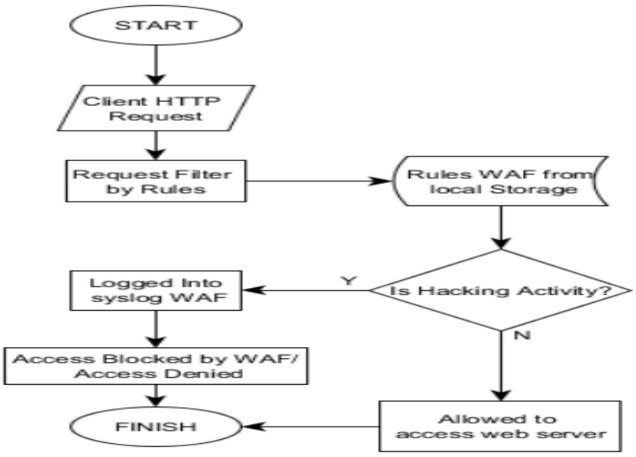
Source: Research Results (2019)

Figure 1. Research Method Framework

2.1 Literature Study

Web server is software that provides data services with a main function to receive Hypertext Transfer Protocol (HTTP) requests from users (known as web browsers) and send the results back in the form of web pages that generally display text, images, animations and videos (Rheno Widianto dan Abdullah Azzam, 2018).

According to (Laitupa, Ismail, dan Rizal, 2015), Web Application Firewall (WAF) is a method for securing web applications, which attempts to prevent threats from attackers. The concept of WAF is very similar to the traditional firewall works. WAF works based on a set of configurable rules called the Core Rules Set (CRS). This CRS selectively allows or rejects HTTP Requests. CRS is also capable of detecting common attacks such as SQL injection, XSS RFI, LFI can even recognize these requests as hacking methods. WAF workflow diagram shown in Figure 1.



Source: Research Results (2019)

Figure 2. WAF workflow diagram

WAF system start from dari *client HTTP request.* After client request, WAF filters the request. Rules from local storage will analyze automatically. If the request cannot be identified by WAF as a hack attack, the request will be processed by webserver, but if the request is identified as a hack attack the webserver will block and deny the access.

Fail2ban function is to monitor the number of SSH login failures on the server, which then the IP will be blocked. Fail2ban can also be used to prevent bruteforce attacks on http, ftp and telnet servers (Kurniawan, Mulyanto, and Nandiasa 2016; Anugrah and Rahmanto 2018).

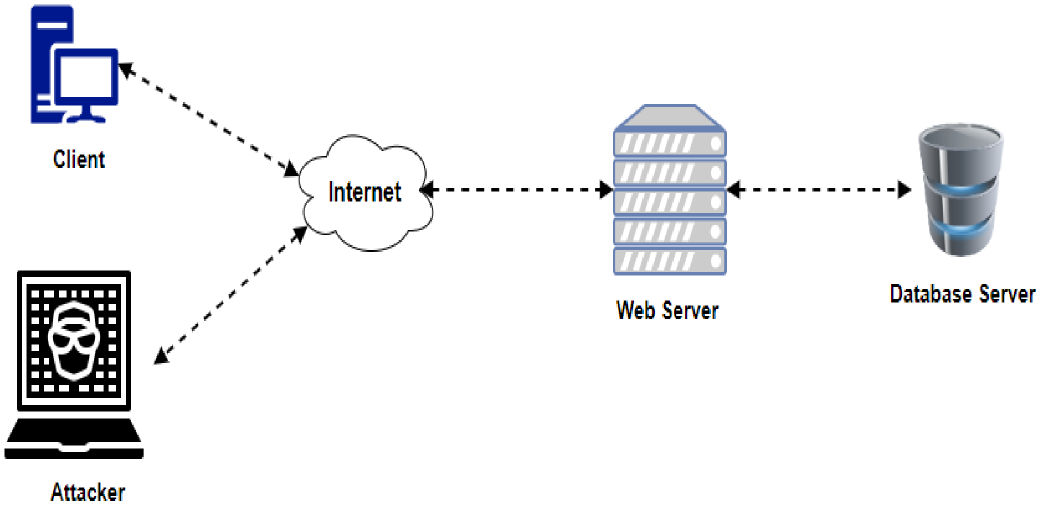
Web Application Firewall (WAF) widely uses Modsecurity as a solution for securing a web server. Modsecurity which is an open source application which is an additional module to apache which currently can also be used on nginx (Suartana, Endah Wahanani, dan Noor Sandy, 2015)

Attack signatures are rules or patterns that identify as an attack or class of attacks on a web application and its components. Security policies compare attack signature patterns with the contents of requests and responses that look for potential attacks. Some Signatures are designed to protect operating systems, web servers, databases, frameworks or certain applications.

3. Result and Discussion

Gytech Indosantara Mandiri Ltd. needs network security. a company engaged in Mecanical Engineering, is in dire need of computer network security to secure existing data or systems such as websites, things that can cause exploitation must be minimized to a minimum and the length of server downtime due to attackers must also be minimized. In Web Application Firewall, one of which can be used to minimize server or application exploitation is Modsecurity combined with Fail2ban. Thus, in the proposed network security at the branch office of Gytech Indosantara Mandiri Ltd. Based on the above understanding the authors propose to add Modsecurity and fail2ban in order to detect and prevent exploitation on the server and use the Attack Signatures method related to the journal themes that the authors take including: 1). Add or install Web Application Firewall on loadbalancer, and 2). Combining Modsecurity and Fail2ban in order to block every suspected request is a pattern of attack or hacking methods

I. Initial Network Schema

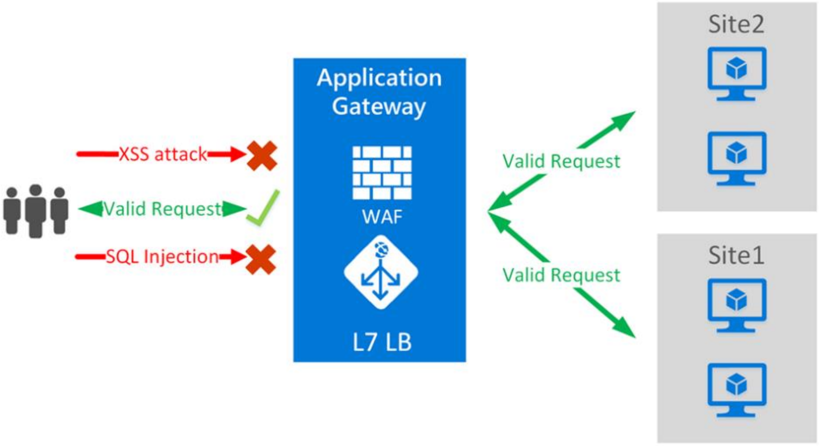


Source: Research Results (2019)

Figure 3. Initial Network Scheme at Gytech Indosantara Mandiri Ltd.

Figure 3 explains the network scheme used by Gytech Indosantara Mandiri Ltd. before using Web Application Security (WAF). A valid user/Client and Attacker has the same permissions because on a web server, Public IP that can be accessed via the internet is used, this makes it easy for Attacker to exploit the Gytech Indosantara Mandiri Ltd. system or web server.

II. Proposed Network Schema



Source: Research Results (2019)

Figure 4. Proposed Network Schema at Gytech Indosantara Mandiri Ltd.

Several applications on Gytech Indosantara Mandiri Ltd. were: Loadbalancer and Web Application Firewall to check each http request to allow or reject if the request is a hacking method and if the request is valid it will be forwarded to the web server.

III. Application Design

The proposed application design was a network security system using a Web Application Firewall. Modsecurity combined with fail2ban can automatically block any attempts that are considered dangerous. Modsecurity uses the Attack signatures method so that it can compare any valid requests or an attack pattern, modsecurity is an open source Web Application Firewall that is built using Regulation Expression (Regex) so that the rules of modsec are easy to use and implement. Web Application Firewall is installed on the loadbalancer so that automatically every request that passes through the loadbalancer will pass through the WAF so that the request can be filtered before the request is received by the web server.

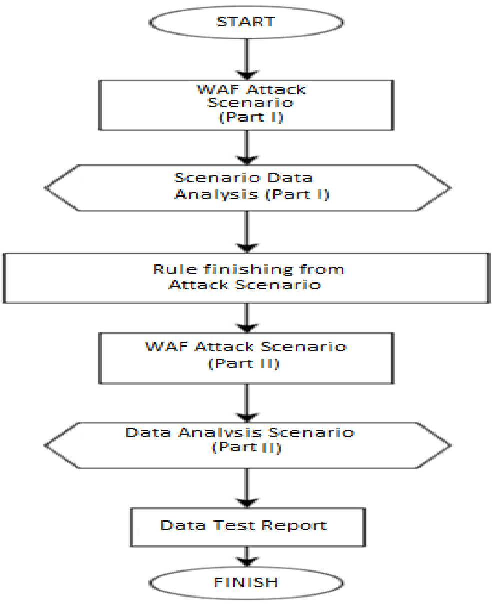
IV. System Testing Stages

Testing must be performed to test the security of WAF on a web server. Test scenarios were shown in Table 1 and Figure 4.

Table 1. List of Testing Scenarios

No.	Scenario	Scenario Description
1	Hacking simulation using SQL Injection technique	The SQL Injection attack test is carried out with SQLMap software and browser, the response time will also be taken into account in this analysis
2	Hacking simulation using Cross-site Scripting (XSS) technique	Trial of XSS attacks is done with browser software, response time will also be taken into account in this analysis
3	Hacking simulation using the Local File Inclusion (LFI) technique	Trial Local File Inclusion (LFI) attacks using browser software, response time will also be calculated in this analysis
4	Hacking simulation using Remote File Inclusion (RFI) technique	Trial Remote File Inclusion (RFI) attacks using browser software, response time will also be calculated in this analysis
5	Hacking attack simulation using the Remote Command Execution (RCE) technique	Trial Remote Command Execution (RCE) attacks using browser software, response time will also be calculated in this analysis
6	Improved rules from analyst results	Improvement of WAF rules is done by adding rules manually to the configuration of WAF rules, response time will also be taken into account in this analysis

Source: Research Results (2019)

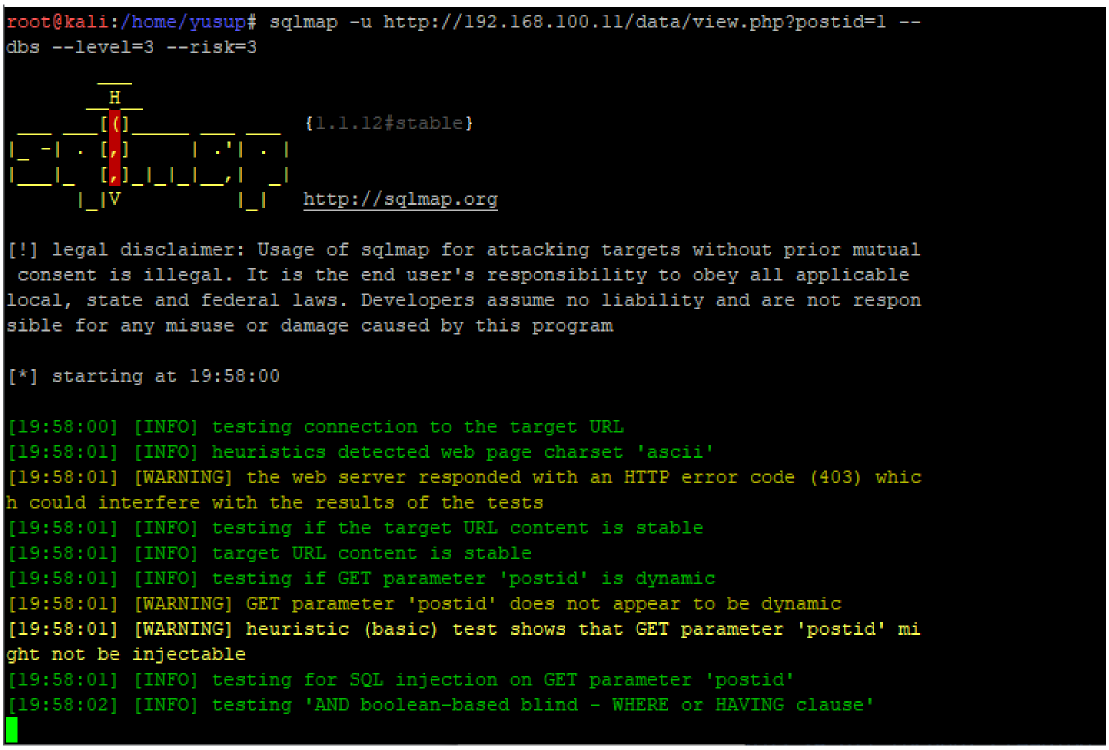

START
WAF Attack Scenario (Part I)
Scenario Data Analysis (Part I)
Rule finishing from Attack Scenario
WAF Attack Scenario (Part II)
Data Analysis Scenario (Part II)
Data Test Report
FINISH

Source: Research Results (2019)

Figure 5. Flow of Test Scenarios

Figure 5 shows the flow of the test scenario on the WAF that starts from the attack scenario of Part I. After the attack, the data of attack scenario for Part I will be analyzed and the rules of the results of the attack scenario part I will be refined. Attack part II will be carried out after attacking part I. Finally, the scenario II attack data will be analyzed again. If the attack can be overcome, WAF will generate a test data report.

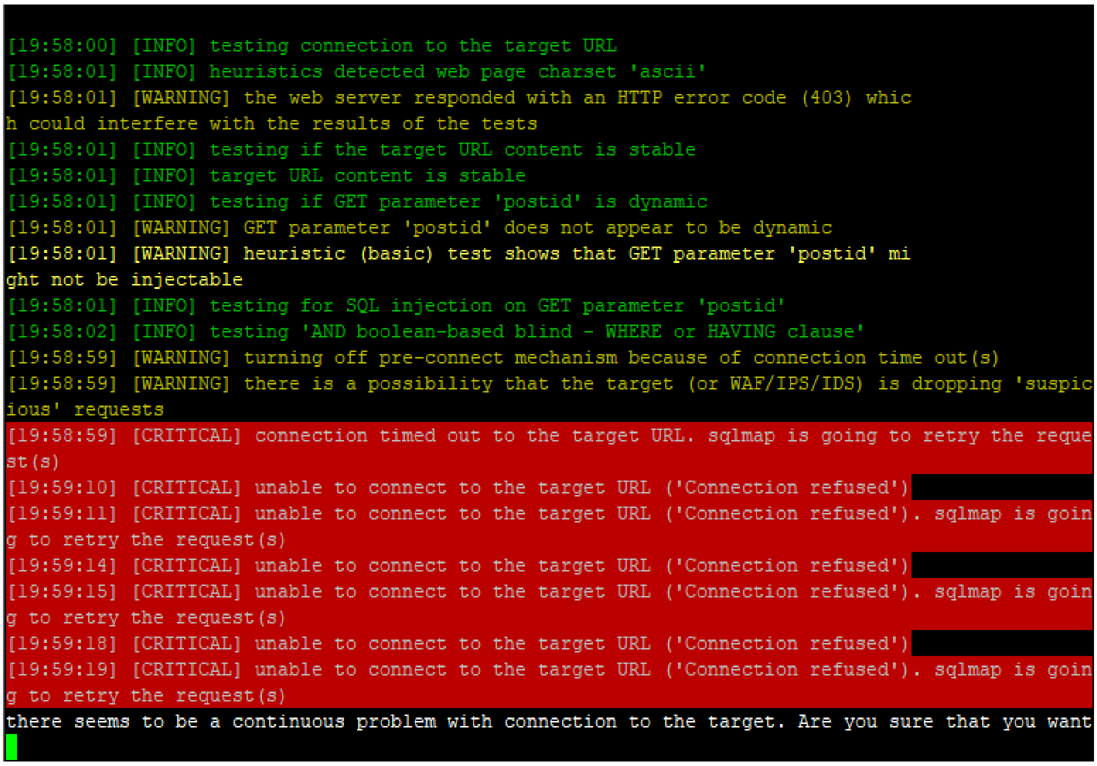
1. SQL Injection Attack Test



```
root@kali:/home/yusup# sqlmap -u http://192.168.100.11/data/view.php?postid=1 --
dbs --level=3 --risk=3

        ___
       __H__
 ___ ___[(]_____ ___ ___  {1.1.12#stable}
|_ -| . [,]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V          |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual
 consent is illegal. It is the end user's responsibility to obey all applicable
local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting at 19:58:00

[19:58:00] [INFO] testing connection to the target URL
[19:58:01] [INFO] heuristics detected web page charset 'ascii'
[19:58:01] [WARNING] the web server responded with an HTTP error code (403) whic
h could interfere with the results of the tests
[19:58:01] [INFO] testing if the target URL content is stable
[19:58:01] [INFO] target URL content is stable
[19:58:01] [INFO] testing if GET parameter 'postid' is dynamic
[19:58:01] [WARNING] GET parameter 'postid' does not appear to be dynamic
[19:58:01] [WARNING] heuristic (basic) test shows that GET parameter 'postid' mi
ght not be injectable
[19:58:01] [INFO] testing for SQL injection on GET parameter 'postid'
[19:58:02] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
```

Source: Research Results (2019)

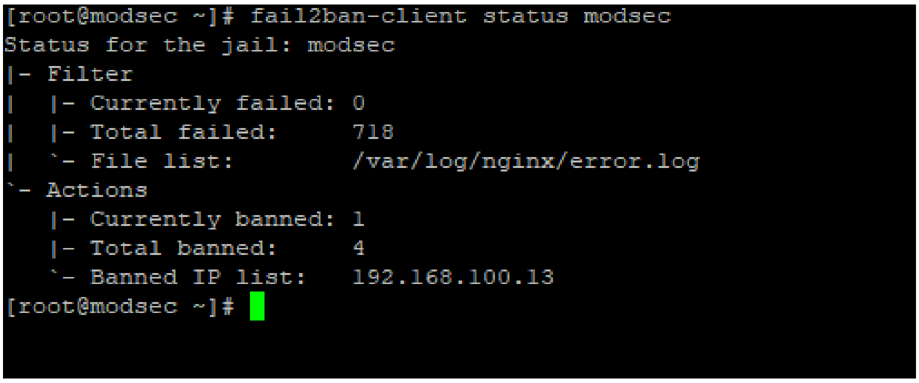
Figure 6. Testing SQL Injection Using SQLmap



```
[19:58:00] [INFO] testing connection to the target URL
[19:58:01] [INFO] heuristics detected web page charset 'ascii'
[19:58:01] [WARNING] the web server responded with an HTTP error code (403) whic
h could interfere with the results of the tests
[19:58:01] [INFO] testing if the target URL content is stable
[19:58:01] [INFO] target URL content is stable
[19:58:01] [INFO] testing if GET parameter 'postid' is dynamic
[19:58:01] [WARNING] GET parameter 'postid' does not appear to be dynamic
[19:58:01] [WARNING] heuristic (basic) test shows that GET parameter 'postid' mi
ght not be injectable
[19:58:01] [INFO] testing for SQL injection on GET parameter 'postid'
[19:58:02] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[19:58:59] [WARNING] turning off pre-connect mechanism because of connection time out(s)
[19:58:59] [WARNING] there is a possibility that the target (or WAF/IPS/IDS) is dropping 'suspic
ious' requests
[19:58:59] [CRITICAL] connection timed out to the target URL. sqlmap is going to retry the reque
st(s)
[19:59:10] [CRITICAL] unable to connect to the target URL ('Connection refused')
[19:59:11] [CRITICAL] unable to connect to the target URL ('Connection refused'). sqlmap is goin
g to retry the request(s)
[19:59:14] [CRITICAL] unable to connect to the target URL ('Connection refused')
[19:59:15] [CRITICAL] unable to connect to the target URL ('Connection refused'). sqlmap is goin
g to retry the request(s)
[19:59:18] [CRITICAL] unable to connect to the target URL ('Connection refused')
[19:59:19] [CRITICAL] unable to connect to the target URL ('Connection refused'). sqlmap is goin
g to retry the request(s)
there seems to be a continuous problem with connection to the target. Are you sure that you want
```

Source: Research Results (2019)

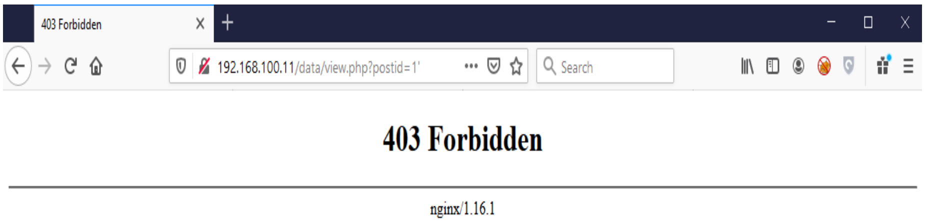
Figure 7. Testing SQL Injection Using SQLMap



```
[root@modsec ~]# fail2ban-client status modsec
Status for the jail: modsec
|- Filter
|  |- Currently failed: 0
|  |- Total failed:     718
|  `- File list:        /var/log/nginx/error.log
`- Actions
   |- Currently banned: 1
   |- Total banned:     4
   `- Banned IP list:   192.168.100.13
[root@modsec ~]#
```

Source: Research Results (2019)

Figure 8. IP Address Attacker is blocked by Fail2ban



403 Forbidden

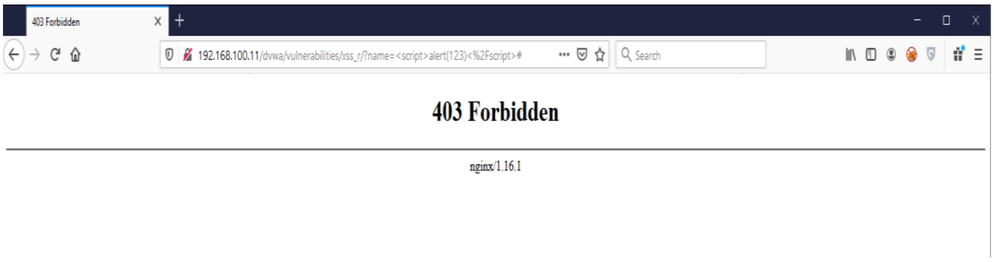
nginx/1.16.1

Source: Research Results (2019)

Figure 9. Testing SQL Injection Using a Browser

Figure 6 and 7 show SQL Injection attack testing using SQLMap tools after the implementation of Web Application Firewall (WAF). The attack was blocked by showing an error response to the SQLmap tool and shown with banned IP and 403 Forbidden status shown in Figure 8 and 9

2. Cross-site Scripting (XSS) Test



192.168.100.11/dvwa/vulnerabilities/xss_r/?name=<script>alert(123)<%2Fscript>#

403 Forbidden

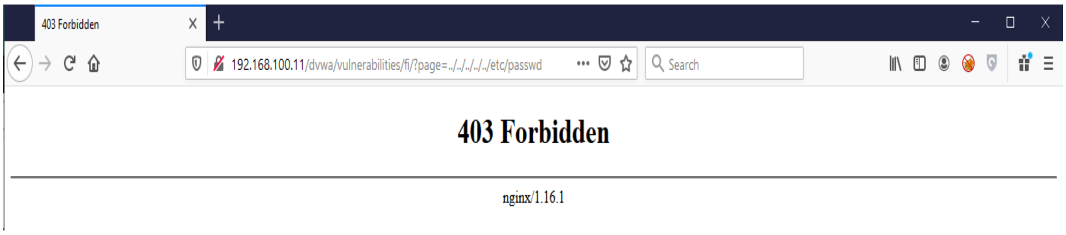
nginx/1.16.1

Source: Research Results (2019)

Figure 10. Testing XSS Using a Browser

Figure 10 shows the http error code with 403 forbidden, XSS payload requests such as "<script> alert (123) <script>" which triggers alert 123. It means that the payload for stealing cookies such as "(document cookie)" cannot be done or has been done recognized by WAF because it is a hacking method.

3. Local File Inclusion (LFI) Trial Attacks



192.168.100.11/dvwa/vulnerabilities/fi/?page=../../../../../etc/passwd

403 Forbidden

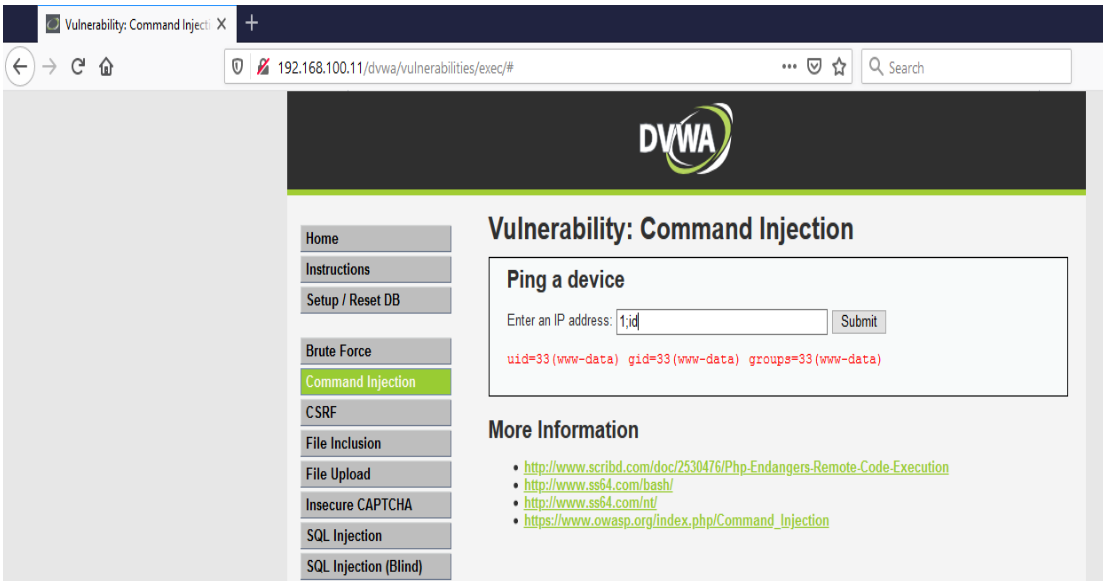
nginx/1.16.1

Source: Research Results (2019)

Figure 11. Testing LFI Using a Browser

Figure 11 shows the http error code with 403 forbidden, LFI payload requests such as "../../../../../../etc/passwd" which calls the passwd file on the server. The page parameters are recognized by WAF because it is a hacking method.

4. Remote Command Execution (RCE) Test



192.168.100.11/dvwa/vulnerabilities/exec/#

Vulnerability: Command Injection

Ping a device

Enter an IP address: 1;id

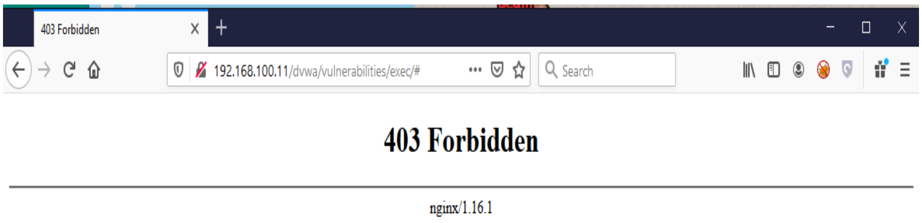
```
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

More Information

- http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution
- http://www.ss64.com/bash/
- http://www.ss64.com/nt/
- https://www.owasp.org/index.php/Command_Injection

Source: Research Results (2019)

Figure 12. Remote Command Execution (RCE) Test



403 Forbidden

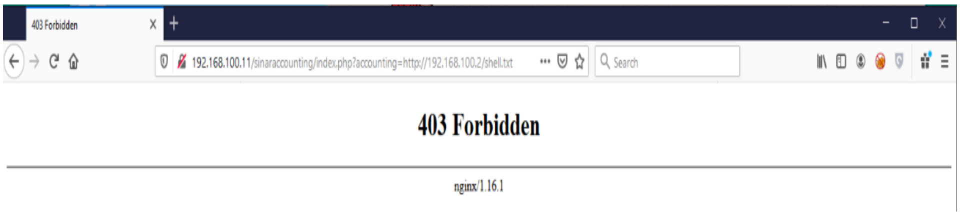
nginx/1.16.1

Source: Research Results (2019)

Figure 13. Testing RCE Using a Browser

Figure 12 shows the RCE attack or command injection on the ip address parameter was successfully carried out. The first test phase WAF did not recognized the various types of Injection command or RCE payload so that it can bypass the WAF. In Figure 13, after the Core rules Set (CRS) was performed, update and re-testing the results of the two results giving a 403 forbidden error response. This proves that the WAF has succeeded in recognizing hacking methods with various methods if a SystemAdmin or Security Operational recognizes various types of hacking method requests was implemented. Therefore, administrators would be able easily to add rules.

5. Remote File Inclusion (RFI) Attack Test


403 Forbidden
192.168.100.11/sinaraccounting/index.php?accounting=http://192.168.100.2/shell.txt
Search

403 Forbidden

nginx/1.16.1

Source: Research Results (2019)

Figure 14. Testing RFI Using a Browser

Figure 14 shows the http error code with 403 forbidden. The request payload by using the redirect function on the accounting parameter has been recognized by the WAF since it is a kind of hacking method.

6. Test Result

Based on the following tables (table 2, table 3 and table 4), the first and second scenarios in numbers 1 to 4 were successfully overcome by giving an error code 403 with Severity High, but in number 5, it cannot recognize the Remote Code Execution attack in the first scenario after updating the rules later the second scenario was successfully tested by recognizing error code 403. Respone time in table 3 and table 4 is shown by the average time in the first and second scenarios is stable and does not cause server side downtime.

Table 2. Testing Results of First and Second Scenarios.

No.	Scenario	Severity Scenario 1	Severity Scenario 2	Response Scenario 1	Response Scenario 2
1	SQL Injection	High	High	403 Forbidden	403 Forbidden
2	Cross-site Scripting	High	High	403 Forbidden	403 Forbidden
3	Local File Inclusion	High	High	403 Forbidden	403 Forbidden
4	Remote File Inclusion	High	High	403 Forbidden	403 Forbidden
5	Remote Command Execution	Unknown	High	200 Ok	403 Forbidden

Source: Research Results (2019)

Table 3. First and Second Scenario Response Time

Assault technique	Scenario I	Scenario II	Average
SQL Injection	146ms	166ms	156ms
Cross-site Scripting	77ms	31ms	54ms
Local File Inclusion	26ms	27ms	26.5ms
Remote File Inclusion	33ms	35ms	34ms
Remote Code Execution	26ms	27ms	26.5ms

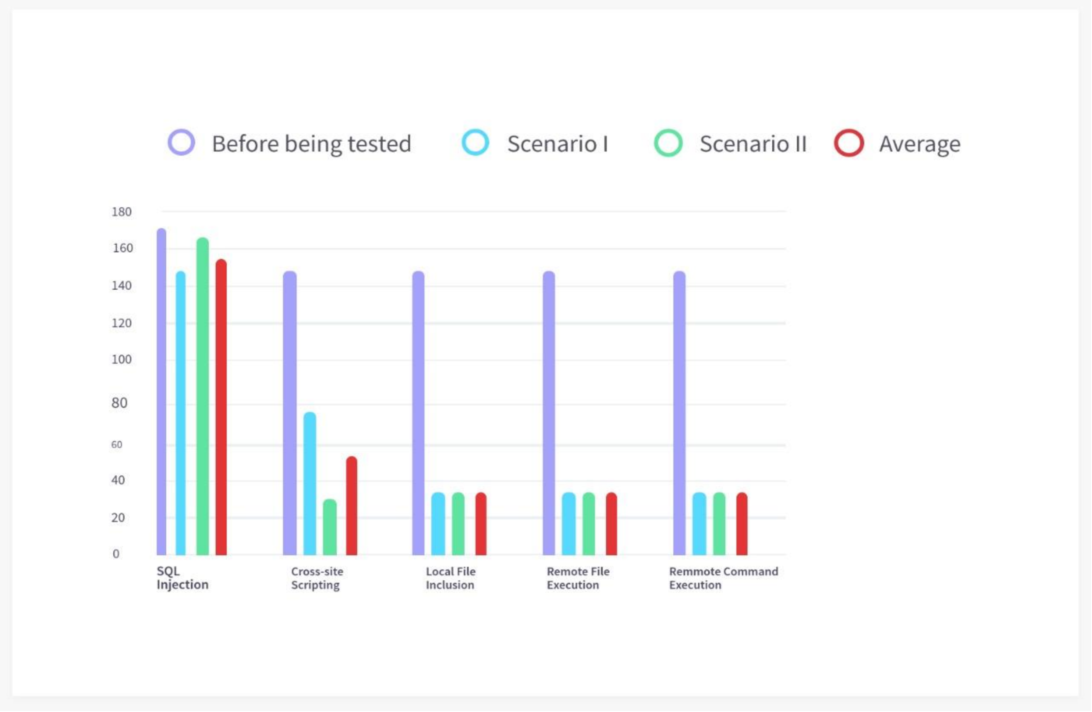
Source: Research Results (2019)

Table 4. Report Data and Test Response Times in the First and Second Test

No.	Scenario	Severity Scenario 1	Severity Scenario 2	Response Scenario 2	Time Response Miliseconds
1	SQL Injection	High	High	403 Forbidden	156ms
2	Cross-site Scripting	High	High	403 Forbidden	54ms
3	Local File Inclusion	High	High	403 Forbidden	26.5ms
4	Remote File Inclusion	High	High	403 Forbidden	34ms
5	Remote Command Execution	Unknown	High	403 Forbidden	26.5ms

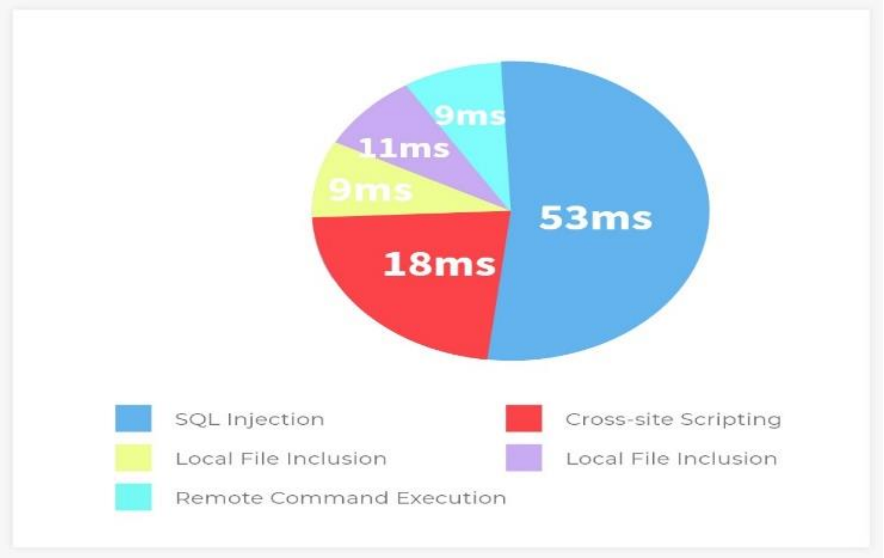
Source: Research Results (2019)

7. Data Analysis



Source: Research Results (2019)

Figure 15. Graph of Response Time (Scenarios I and II)



Source: Research Results (2019)

Figure 16. Response Time Graph When WAF Conducts Hacking Attack

4. Conclusion

Based on testing results and measurement analysis in previous section, it can be concluded as follows: 1). Web server with WAF can prevent and detect exploitation, 2). The use of Modsecurity and Fail2ban with open source services, enables customization and can add rules so that they can be easily adapted to the needs of the company, 3). The use of Modsecurity and Fail2ban with open source services is not paid or free, so that it can ease the financial burden on the company.

Reference

Amarudin, and F. Ulum. 2018. Desain Keamanan Jaringan Pada Mikrotik Router OS Menggunakan Metode Port Knocking. *Jurnal Teknoinfo* 12, no. 2: 72–75.

Anugrah, I., and R.H. Rahmanto. 2018. Sistem Keamanan Jaringan Local Area Network Menggunakan Teknik De-Militarized Zone. *PIKSEL : Penelitian Ilmu Komputer Sistem Embedded and Logic* 5, no. 2: 91–106.

Kurniawan, I., F. Mulyanto, and F. Nandiasa. 2016. Sistem Pencegahan Serangan Bruteforce Pada Ubuntu Server Dengan Menggunakan Fail2Ban. *Infomatek* 18, no. 2: 89–96.

Laitupa, D.R.H., S.J.I. Ismail, and M.F. Rizal. 2015. IMPLEMENTASI MODSECURITY SEBAGAI SISTEM MONITORING KEAMANAN APLIKASI WEB SECARA REAL TIME. *EProceedings of Applied Science* 1, no. 3: 2132–2134.

Rheno Widianto, S., and I. Abdullah Azzam. 2018. Analisis Upaya Peretasan Web Application Firewall Dan Notifikasi Serangan Menggunakan Bot Telegram Pada Layanan Web Server. *Elektra* 3, no. 2: 19–28.

Suartana, I.M., H. Endah Wahanani, and A. Noor Sandy. 2015. Sistem Pengaman Web Server Dengan Application Firewall (WAF). *Scan* X, no. 1: 39–44.