Cross-Site Scripting Attack Detection using Rule-Based Signature

Deris Stiawan [a,1,*], Gonewaje [b,2], Ahmad Heryanto [a,3], Rahmat Budiarto [b,4]

[a] Department of Computer Engineering, Faculty of Computer Science, Indonesia
[b] College of Computer Science and IT, Albaha University, Al Aqiq, Saudi Arabia
[1] deris@unsri.ac.id*; [3] hery@unsri.ac.id; [4] rahmat@bu.edu.sa
* corresponding author

ARTICLE INFO

Article history
Received
Revised
Accepted

Keywords
Signature-Based Detection,
Cross-Site Scripting,
Snort
IDS

ABSTRACT

Rule-Based Signature or also known as Misuse Detection is IDS which rely on matching data captured on retrieval of attack pattern which in system that allow attacks. If the attack activity detected according to existing signature, then it will be read by system and called as attack. The advantage of this Signature-Based IDS is the accuracy of detecting matched attack which in the system with low false-positive result and high true-positive. Cross-Site Scripting is type of attack which is perform by injecting code (usually) JavaScript to a site. XSS is very often utilized by attacker to steal web browser resource such as cookie, credentials, etc. Dataset which used in this research is dataset which created by injecting script into a website. Once obtained the dataset, then feature extraction is performed to separate the attribute which used. XSS attack pattern can be easily recognized from URI, and then detected using engine which has been created. Detection result of algorithm which used is evaluated using confusion matrix to determine detection accuracy value which performed. Obtained accuracy detection of research result reached 99.4% with TPR 98.8% and FPR 0%.

1. Introduction

Attack which on website or HTTP protocol are growing where there is HTML tag and JavaScript function. Vulnerability toward Cross-Site Scripting (XSS) is very often utilized by attacker to steal web browser resources such as cookie, credentials, and etc. They inject malicious JavaScript code into web which is vulnerable to this attack [1].

SQL Injection and Cross-Site Scripting (XSS) attack detection by creating various types of "Regular expression signatures ". However, some of these rules often warning despite slight attack and may result as false positive. These rule can be used further and then modified to get even better result [2]. Comparison of two IDS technique which are, Signature Based and Anomaly Based, is based on resource RAM Signature-Based which consume greater resource, but based on accuracy Signature-Based is better [3].

Web application based attack can be detected using Rule-Based Signature But there is still lot of false positive value which need improvement. Some of these attack patterns can be used as reference for creating new rules in order to suppress false positive values.

This paper is structured in the following format: Literature review in section 2, in section 3 discuss about methodology, section 4 discusses the temporary result, and conclusion also further activities. will be discussed in section 5.

Sriwijaya Journal of Informatic and Applications
Vol. 2, No. 1, January 2021, pp. 40-44

2. Previous Research

Research [4], discusses how to use Intrusion Detection Systems to prevent Cross-Site Scripting (XSS) attack. It is also said that currently best prevention method of XSS attack is follow the twelve rules of OWASP XSS Although those method is quite time-consuming and yet able to recognize all the existing attack.

On research [5], discusses about IDS system used to detect XSS attack using three identical web pages containing mix of simple HTML tag and JavaScript whcih are then uploaded to the Apache Web server. After that web for XSS testing was uploaded, experiment conducted using IDS XSS program then further tested using various vector of XSS attack. From the test result it was obtained that IDS XSS system was able to detect Cross-Site Scripting (XSS) attack but there is weakness found that there was a undetectable XSS attack vector containing nul (/0) character.

Furthermore [6], HTTP detection with rule option or rule-based which utilized snort as detection tool and adding new rules still get FPR (False Positive Rate) and TNR (True Negative Rate) value above 40%-60% for each test.

3. Research Methodology

In this research, it use several software and dataset which have been created as research reference, as well as a PC used for data processing with spesification as follows:

Table 1. Software Requirement

System	*Tools*	Information
IDS	Snort	Version 2.9.11
Compiler	Python	Version 2.7.6
IDE	Geany	Version 1.23.1

Table 2. Hardware Requirement

Hardware / SO	Information
CPU	Intel Core i3
RAM	4 GB
HDD	500 GB
Operating System	Linux

Creation of dataset is perform in three scenarios, the first scenario is normal data retrieval, the second scenario of attack data retrieval and the last scenario is retrieval of combined data (normal and attack). These following topology used in this study:

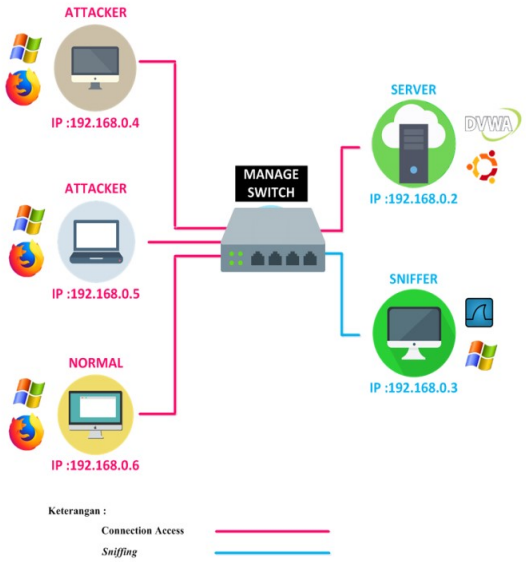


Fig. 1. Topology of Dataset

Scenario of dataset retrieval is perform in approximately 15 minute. Based on Figure 1, if normal data scenario is performed then only computer with IP 192.168.0.6 or computer with "normal" label doing activities and other idle. Then, if the scenario of attack data then which performed attack activity only the computer with label "attack". Similarly, if combined scenario then the three computers are doing activities according to their respective labels. Whole activity to the server and on the switch is configured port mirroring to capture data packet traffic.

After dataset is obtained, the next step is performing feature extraction to make it easier to find attack pattern. Feature extraction aim to separate the attributes which will be used and then save them in comma-separated values format. Here is flowchart of feature extraction:

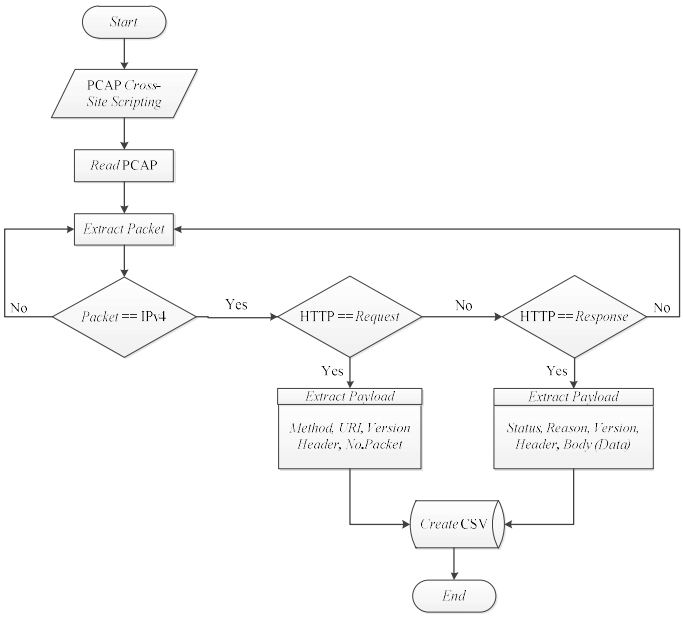


Fig. 2. Feature Extraction Flowchart

Sriwijaya Journal of Informatic and Applications
Vol. 2, No. 1, January 2021, pp. 40-44

4. Result

Results obtained from the implementation of rule-based signature method are stored in .csv extension log file. After getting detection result then perform matching between data packet detection result and raw data .pcap, these following matching result.

From validation result there are 6 point match which are :

A. Point 1, 2, and 3, point 1 indicate matching of packet number,
B. point 2 indicate the time of packet starting from year, month, date to hour-second and
C. point 3 indicate IP source. B.
D. Point 4, 5, and 6, on point 4 is source port, point 5 is IP destination and point 6 is destination port.

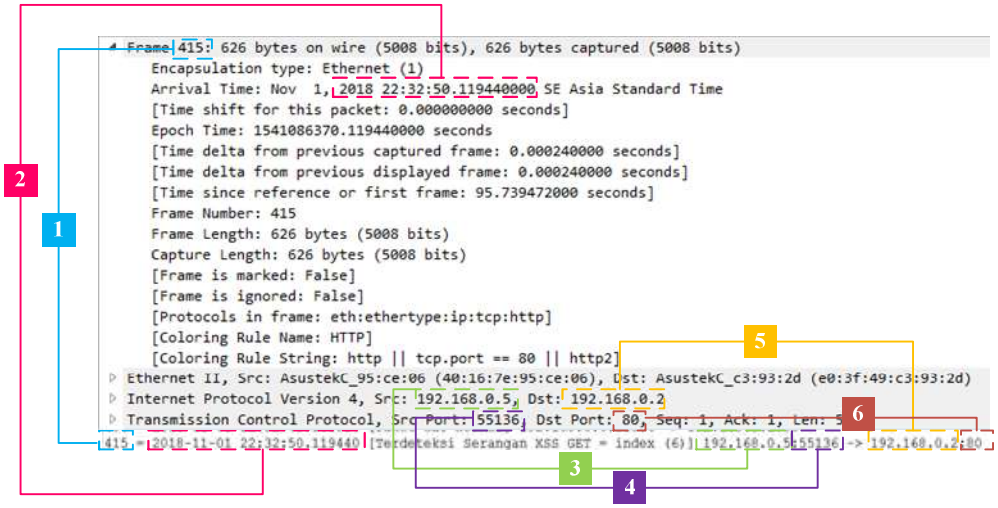
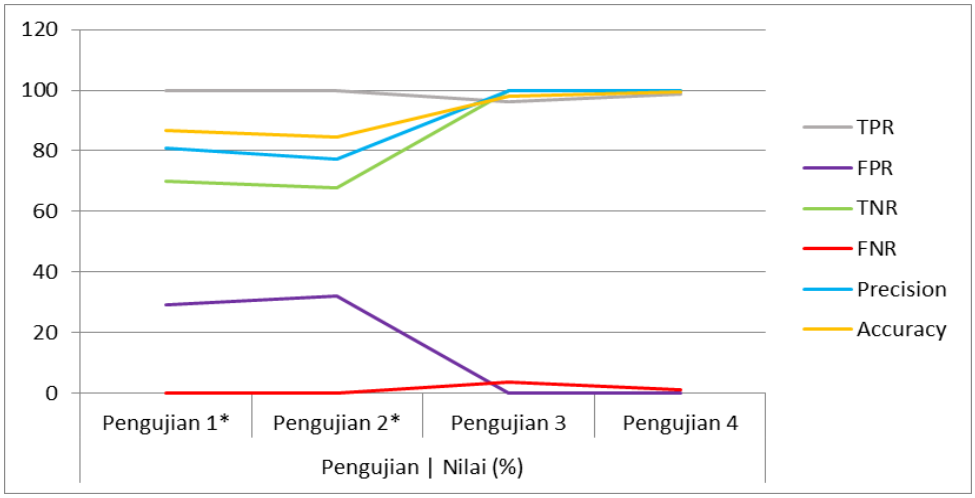


Fig. 3. Validation of attack packet detection using raw data *(.pcap)*

Lastly, calculation of confusion matrix to obtain accuracy, precision, TPR, FPR and other value to measured research. These following result from confusion matrix of this research:



Note: (*) The test phase uses the previous research [6].

Fig. 4. Graph of *detection rate confusion matrix*

5. Summary and Further Research

From detection result which perform, obtained summary as follow :

A. XSS Reflected attack reflect script which being injected by attacker on URI with GET method, while in XSS Stored attack the injected script is stored in file system or database (mtxMessage) with POST method.

B. Both Reflected and Stored type of XSS attack occur as result of Web server encoding error or there is no encoding process by web administrator so that the server process malicious script which should not be allowed to be processed or executed.

C. Attribute which used in HTTP Request payload to recognize XSS attack packet which are Method, URI, Version, Header Request, Packet Number and HTML form URL Encoded.

D. Cross-Site Scripting attack pattern focus on character percent encoding because attack string has been encoded by the system automatically.

Further Research will perform to answer some issues, including: (i) conducting real-time detection. (ii) Attack detection on other Web application such as Cross-Site Request Forgery, Path Traversal, Code Injection and Remote File Inclusion. (iii) Focus on preventing attack to block attack.

References

[1] S. Gupta and L. Sharma, "Exploitation of Cross-Site Scripting (XSS) Vulnerability on Real World Web Applications and its Defense," *nternational J. Comput. Appl. (0975 – 8887)*, vol. 60, no. 14, pp. 28–33, 2012.

[2] K. K. Mookhey and N. Burghate, "Detection of SQL Injection and Cross-site Scripting Attacks," *Symantec*, pp. 3–7, 2010.

[3] N. Sagita, N. D. Cahyani, and F. A. Yulianto, "Perbandingan Performansi Antara Signature Based dan Anomaly Based Dalam Pendeteksian Intrusi," 2011.

[4] C. Obimbo, K. Ali, and K. Mohamed, "Using IDS to prevent XSS Attacks," *Int'l Conf. Secur. Manag.*, pp. 233–239, 2017.

[5] C. M. Frenz and J. P. Yoon, "XSSmon: A Perl based IDS for the detection of potential XSS attacks," *2012 IEEE Long Isl. Syst. Appl. Technol. Conf. LISAT 2012*, pp. 0–3, 2012.

[6] M. R. Zalbina, T. W. Septian, D. Stiawan, M. Y. Idris, A. Heryanto, and R. Budiarto, "Payload recognition and detection of Cross Site Scripting attack," *2017 2nd Int. Conf. Anti-Cyber Crimes, ICACC 2017*, pp. 172–176, 2017.