Assessment of Web Scanner Tools

Rawaa Mohammed
Al-Mustansiriyah University
Computer engineering

ABSTRACT
Nowadays the security of web applications becomes a serious problem because of the impact of its vulnerability, so a previous consideration should be taken to diminish its harmful effect. One of the most important tools used to test the security of the web is web security scanner which is a tool that can be used by the penetration tester to give clear indication of the weakness by detecting the vulnerabilities of web pages like SQL injection, XSS attack. While the importance of web scanners are so obvious, but there effectiveness and differences need to be evaluated to find the flaws, limitations and distinguish between them. In this paper an analytical comparison is present on six open source web scanners by using manual and automatic testing of the chosen test beds then analyzing these results to assess the scanners.

General Terms
Vulnerabilities detection, Web scanners, assessment of open source tools.

Keywords
False Positive, False Negative, evaluation, analysis.

1. INTRODUCTION
The need for an automated scanner to check the flaws, vulnerabilities of websites is very important today. Web tester examines web application to identify the potential weak points. The scanning tools perform testing without knowing the source code, so it is called black box testing. The evaluation of these scanners should be done to check the effectiveness of each scanner. [1]

The code located inside the software is not one of the demands to perform the black box testing, the function of this testing is similar to the attacker action that relay on fazing method.

In this method a specified data is prepared for each type of attack then monitoring the related result to detect the vulnerabilities, this will be done automatically by different web scanner tools. [2]

The attacker examine the web to search for the vulnerabilities that can be used by him to put any type of attack like controlling the user account, reach sensitive data, inject malicious script in the user input field, so this problem should be monitored to eliminate its harmful effect. In the recent year, researchs show that almost all web applications tested by web scanner had at least one serious vulnerability. [3]

Web scanner can be classified into commercial tools or open source tools. [4]

There are differences and limitations to the usefulness of dynamic web scanners, in other word the effectiveness of these scanners need to be assessed to find how it is exact in discovering a specific attack upon another. [5]

2. RELATED WORK
Fakhreldeen A. and Eltyeb E. assess different web scanner depending on OWASP Top 10-2013 application security risks, this assessment is used by the developer to choose the best one for each application. [1]

David A. Shelly analyzes the effectiveness of commercial and open source web scanners, by using two versions of testbed one contain vulnerability and one without vulnerability, then he propose a new method to minimize the number of false positive and false negative attack. [6]

Yuliana M. presents an evaluation reports from the results of running QualysGuard WAS and Acunetix WVS against a chosen test bed. The identification of the most challenging vulnerabilities is present for WAVS to detect, and compare their effectiveness as penetration testing tools. [7]

Xiaowei Li and YuanXue consider behavior model for the applications by comparing each requests responses of the web. This model consist of two phases the first phase is used to derive the model by identifying the attacks and in the second phase this model is used to evaluate each request and response to check for the differences. [8]

Sneha Parmar uses specific scanning strategy with certain open source tools, so the accuracy of the detection will increase more than using one scanning tool. The using of different type of scanner can cover nearly every area of support by web vulnerability scanners. [2]

Mikko Vimpari uses a qualitative investigation method named Choosing by Advantage (CBA).It was established on standard resulting from the end users' requirements. Finally a list of advantage and drawback is inferred from an analytical study to evaluate each one. [9]

3. PROPOSED SYSTEM
As shown bellow in figure (1) the tester begins the testing by performing two types of testing: Automatic and manual testing.

At the first automatic testing is done by using open source web scanners tools, where the result of this testing is a number of vulnerabilities detected by a specified scanner.

Then in manual testing the developer perform testing by himself by sending the http request and receiving the response to/from the server, the result of this testing is a real number of the detected vulnerabilities.

Finally these results obtained from automatic and manual testing will be compared to calculate the number of false positive and false negative attack which will be used in the evaluation of web scanners.

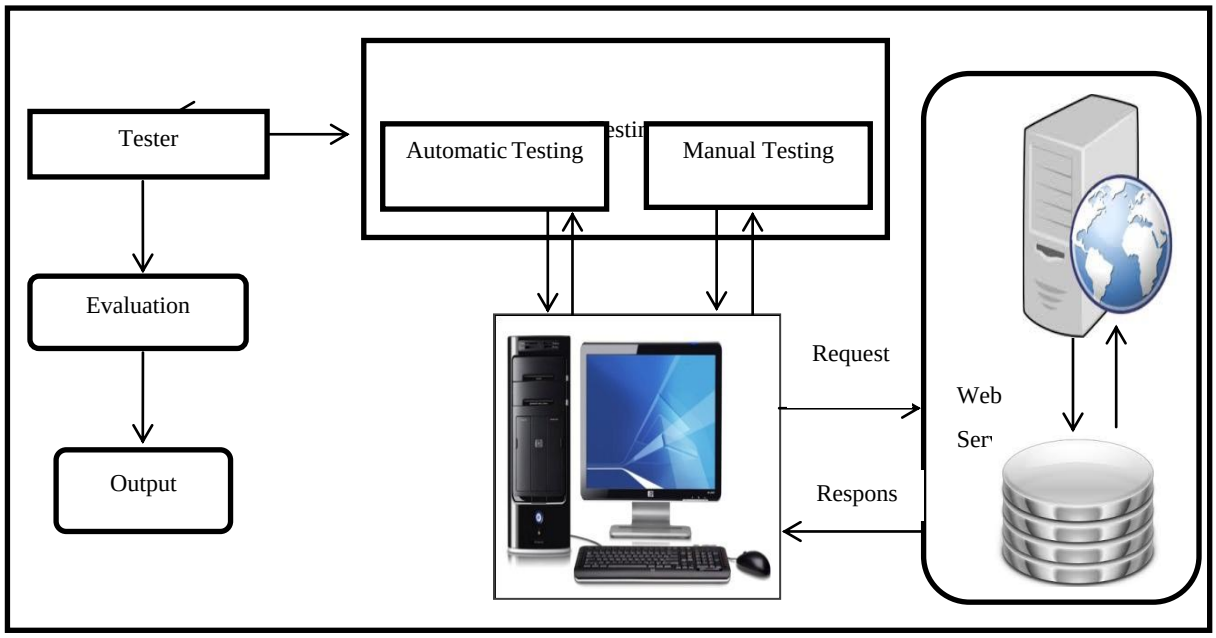

Fig. 1: The cycle of assessment steps

This experimental study contains three steps as will be described here:

The first step was to setup Apache software server and a number of previously prepared web sites as a test bed in this work six vulnerable web application called (bwapp, OWASP_Bricks, dvwa, WackoPicko, Webgoat, gruyere) will be hosted on this server.

In web application there are many types of vulnerabilities that is varied according to its harmful, in this study XSS and SQL will be examined because they are classified as the most crucial for detection. In the first XSS attack will be checked against the test bed, which is result from exploiting the input field by injecting malicious script on the transaction tag. The second type of vulnerability that will be considered is SQL injection that is caused by saving improper SQL statement in a data base instead of legitimate transaction this attack can cause future attack to any user that interact with this database.

The second step was to setup six web application scanners for use against the vulnerable web application, all of them are free/open.

The final step was sorting the results into false-positives or false-negatives these two metrics is the base for analyzing the results and identifying the limitations of these scanners.

4. RESULTS EVALUATION

In this work the evaluation of six scanners is done by analyzing the results that is obtained from the execution of web scanners against the vulnerable webs then comparing the number of detected vulnerabilities to the real number of attacks obtained from manual test; this comparison will give clear indication of the number of false negative and false positive attack.

False positive means the scanner detect vulnerabilities in the site but in real it doesn't exist.

False negative refer to vulnerabilities that is not reported by a vulnerability scanner but in real, it does exist, Finally the Precision, recall, and F-measure will be determined in order to assess these scanners.

4.1 Precision of Scanners

It refers to the percentage of correct relevant vulnerabilities to the total number of information. It is also known as positive predictive value and can be calculated as the following equation:

$$Precision = \frac{TP}{TP + FP}$$

From the results that are obtained after applying six scanners on six web sites, the precision of each scanner for SQLI attack will be as shown in table (1)

Table(1) Precision SQLI

Web Scanner	TP	TP+FP	Precision %
Paros proxy	14	16	87.5
Wapiti	11	18	61.1
skipfish	10	13	76.9
Nikto	9	11	81.8
Wfuzz	9	14	64.2
Netsparker	8	9	88.8
HP WebInspect	8	15	53.3

As can be shown from the results above, Netsparker scanner has the highest precision of about 88 % followed by Paros Proxy with 87 %, whereas HP WebInspect has the lowest precision (53 %).

In XSS attack the results of table (2) show that Skipfish reach to 78 % with highest precision but HP WebInspect has the lowest precision (39 %). The precision of other scanners is in between theses percentage values.

Table(2) Precision XSS

Web Scanner	TP	TP+FP	Precision
Paros proxy	13	21	61.90

Wapiti	4	10	40
skipfish	11	14	78.57
Nikto	11	17	64.70
Wfuzz	11	25	44
Netsparker	12	22	54.54
HP WebInspect	11	28	39.28

4.2 Recall of Scanners

It is the proportion of positive cases that were correctly identified, this metric refer to the true detected vulnerabilities to all number of real vulnerabilities (TP+FN), it can be found as the following equation:

$$Recall = \frac{TP}{TP + FN}$$

From table (3) that refers to the percentage of recall for each scanner in SQLI, we can see that the recall of Paros Proxy occupy higher percentage (73%) than the others, next to it Wapiti with (57%), the recall of the other scanner was very close about 40%.

Table (3) Recall of SQLI

Web Scanner	TP	TP+FN	Recall %
Paros proxy	14	19	73.6
Wapiti	11	19	57.8
skipfish	10	18	55.5
Nikto	9	19	47.3
Wfuzz	9	20	45
Netsparker	8	19	42.1
HP webInspect	8	19	42.1

On the other hand, it can be shown from table (4) that the recall of all scanners in XSS attack was higher than recall in SQLI except for Wapiti which has about 30 %.

Table (4) Recall of XSS

Web Scanner	TP	TP+FN	Recall
Paros proxy	13	18	72.22
Wapiti	4	13	30.76
skipfish	11	16	68.75
Nikto	11	17	64.70
Wfuzz	11	16	68.75
Netsparker	12	18	66.66
HP WebInspect	11	17	64.70

4.3 F-Measure

This value gives a clear indication of the effectiveness of each scanner because it combines precision and recall into a single measure which is a Harmonic mean of them as the following equation:

$$F = 2pr/(p + r)$$

Table (5) show the F-measure of each scanner in SQLI, as can be shown the percentage of Paros proxy and skipfish have higher efficiency than the others with 79, 64 respectively.

Wapiti and Nikto have the same F-measure (59 %), next to them was Netsparker (57 %).

The lower F-measure was for wfuzz (26 %) Preceded by Hp-Webinspect with 47 %.

Table (5) F-measure of SQLI

Web Scanner	Precision %	Recall %	2*p*r	P+r	F
Paros proxy	87.5	73.6	12880	161.1	79.95
Wapiti	61.1	57.8	7063.16	118.9	59.4
skipfish	76.9	55.5	8535.9	132.4	64.47
Nikto	81.8	47.3	7738.28	129.1	59.94
Wfuzz	64.2	45	2889	109.2	26.45
Netsparker	88.8	42.1	7476.96	130.9	57.11
HP WebInspect	53.3	42.1	4487.86	95.4	47.04

Table (6) calculate the F-measure for XSS attack, here the higher F-measure was occupied by Skipfish and Paros Proxy with 73% , 66% respectively, next to them was Nikto with 64 %.

Netsparker and Wfuzz have near efficiency about 59, 53 percent respectively.

Finally Hpwebinspect has 48 %, followed by Wapiti which has the lower F-measure among all scanners with about 35 %.

Table(6) F-measure of XSS

Web Scanner	Precision %	Recall %	2*p*r	P+r	F
Paros proxy	61.90	72.22	8940.83	134.1	66.6
Wapiti	40	30.76	2460.8	70.76	34.7
skipfish	78.57	68.75	10803.37	147.3	73.3
Nikto	64.70	64.70	8372.18	129.4	64.7
Wfuzz	44	68.75	6050	112.7	53.6
Netsparker	54.54	66.66	7271.27	121.2	59.9
HP WebInspect	39.28	64.70	5082.83	103.9	48.8

The results of table (1), table (3) and table (5) can be concluded as shown in figure 2 that put Paros proxy in the first with highest F-Measure, whereas Wfuzz has the lowest F-Measure.

The values of F-Measure are very close in Wapiti, Skipfish, Nikto, Netsparker, and Hp webinspect.

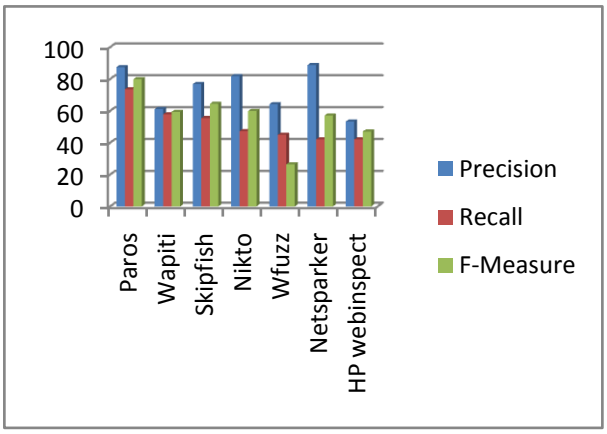

Fig 2: precision, recall, and F-Measure in SQLI attack

Figure 3 is related to XSS attack, it is concluded from the values of table (2), table (4), and table (6), here Skipfish come in the first with highest F-Measure, next to it is Paros then Nikto.

Wfuzz, Netsparker, Hp webinspect get almost near values of F-Measure.

Finally it can be shown that Wapiti has the lowest value of F-Measure.

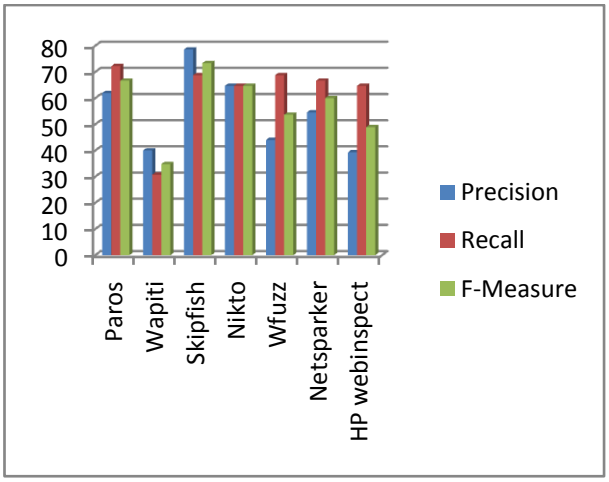

Fig 3: precision, recall, and F-Measure in XSS attack

5. CONCLUSION

This analysis and evaluation of black-box Web application vulnerability scanners was focused on SQLI and XSS attacks due to there sever effect on the proper service of any web sit. It can be concluded from the results that the efficiency of each scanner is different for each type of attack, because each scanner uses its own injection string set or has generated the injection strings using different patterns, as found from the result in SQLI Parosproxy have the higher F-measure (79%) whereas in XSS Skipfish was in the first with (73%).

In general we can see from the results that the efficiency of all scanners in SQLI is higher than the efficiency in XSS attack.

So when the developer would like to check any web application for any intended vulnerability, the appropriate scanner should be chosen for every vulnerability depending on previous assessment.

6. REFERENCES

[1] Fakhreldeen A. and Eltyeb E., "Assessment of Open Source Web Application Security Scanners", College of Computer Science and Information Technology, KAU, Khulais, Saudi Arabia, march 2014.

[2] Pakorn I., "A Comparative Study of Security Vulnerabilities in Responsive Web Design Framework", Malardalen University School of Innovation Design and Engineering, June 2015.

[3] Sneha P., "Vulnerability Checker for Infosecurity", SRM University, 2013.

[4] Fakhreldeen A., "Using WASSEC to Evaluate Commercial Web Application Security Scanners", International Journal of Soft Computing and Engineering (IJSCE), 2014.

[5] Kinnaird M., "Open Source Web Vulnerability Scanners", Marymount University, 2014.

[6] David A. Shelly, "Using a Web Server Test Bed to Analyze the Limitations of Web Application Vulnerability Scanners", Faculty of the Virginia Polytechnic Institute and State University, July 2010.

[7] Yuliana M., "Security Evaluation of Web ApplicationVulnerability Scanners' Strengths and Limitations Using Custom Web Application", California State University, October 2012.

[8] XiaoweiLi and YuanXue, "BLOCK: A Black-box Approach for Detection of State Violation Attacks towards Web Applications", Vanderbilt University, 2011.

[9] Mikko V.," An Evaluation of Free Fuzzing Tools", University of Oulu Department of Information Processing, May 2015.