

# Lab - Perform a Vulnerability Scan Using OWASP Zed Attack Proxy

## Overview

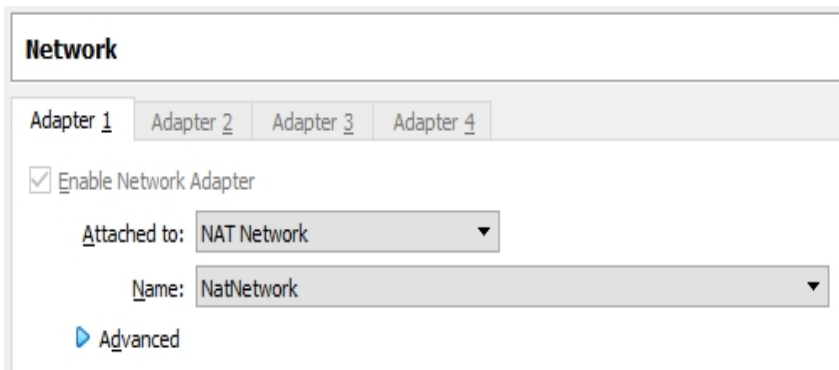
Penetration testing helps in finding vulnerabilities before an attacker does.

OSWAP ZAP is an open-source, free tool used to perform penetration tests. The main goal of Zap is to allow easy penetration testing to find the vulnerabilities in web applications.

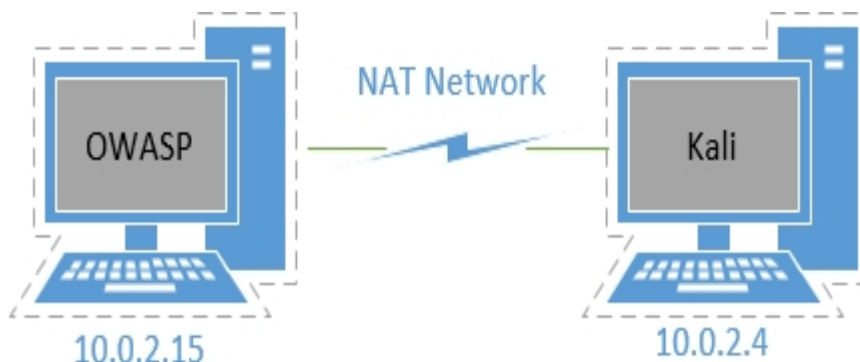
ZAP creates a proxy server and makes the website traffic pass through the server. The use of auto scanners in ZAP helps in finding the vulnerabilities on the website.

## Lab Configuration:

- One virtual install of Kali Linux
- One virtual install of OWASP Broken Web Application VM
- Ensure that both VirtualBox network adapters are set to NAT network



## Lab Diagram



These are my IP addresses. Yours may differ!

The OWASP VM will show you its current IP address once you log on to the terminal. Username and password are provided at the terminal window.

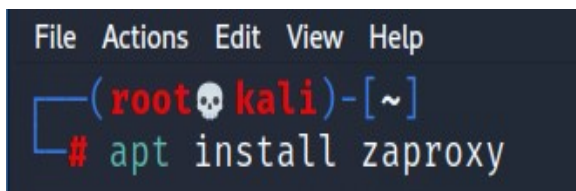
For your Kali, open a terminal and use the `ifconfig` command to find the IP address assigned to your `eth0` adapter.

## Begin the Lab!

### Install OWASP ZAP

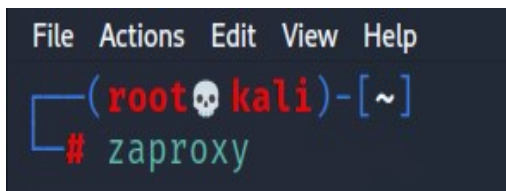
By default, Kali does not come with **OWASP ZAP** installed. To install **OWASP ZAP** from your Kali desktop, launch a terminal and at the prompt, type the following command:

```
apt install zaproxy
```



Once the application has been installed, to launch OWASP ZAP, type the following at your terminal prompt:

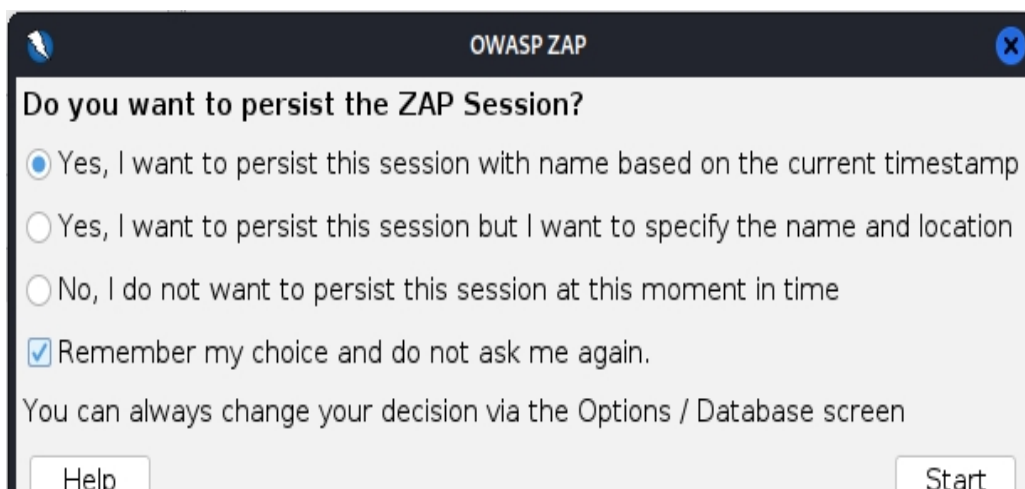
```
zaproxy
```




It takes a few minutes to load up.

After initial processing in the terminal window, the OWASP ZAP window is displayed.

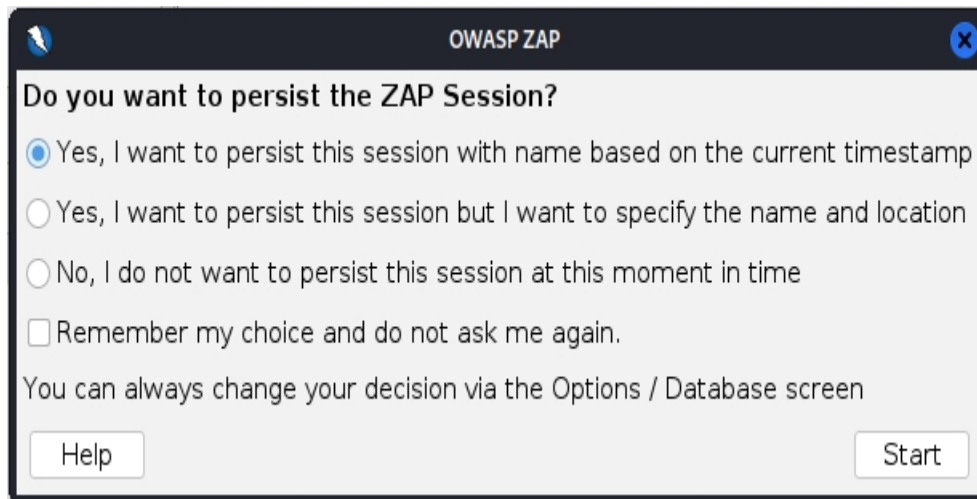
The OWASP ZAP dialog box is as follows:



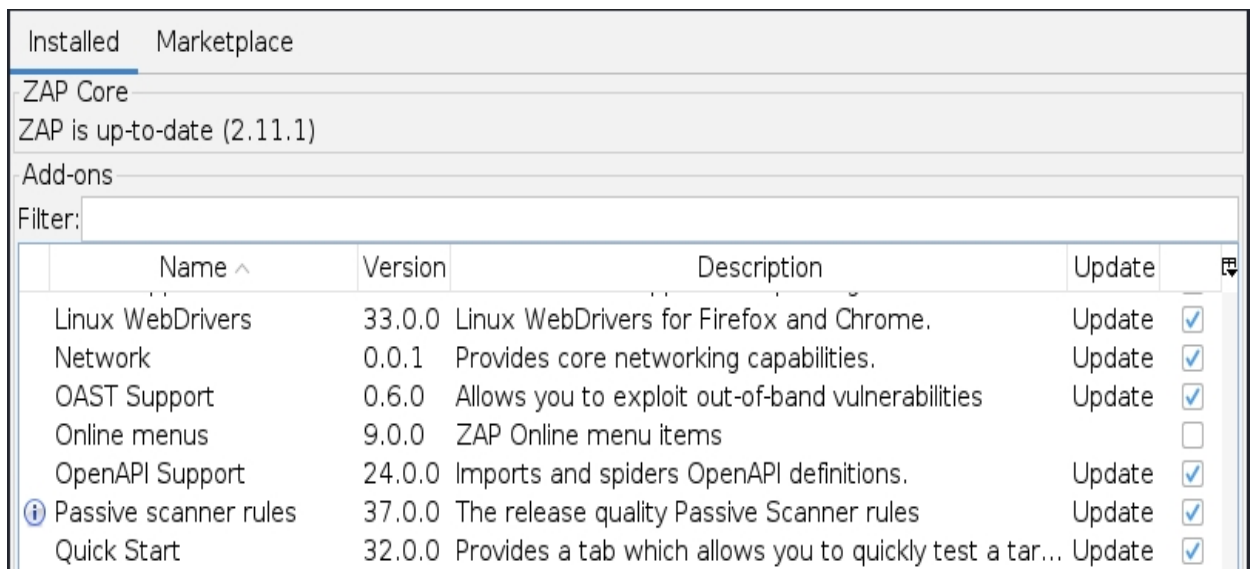


Select Yes on this dialog box; I want to persist this session with name based on the current timestamp. Lastly, select Remember my choice and do not ask

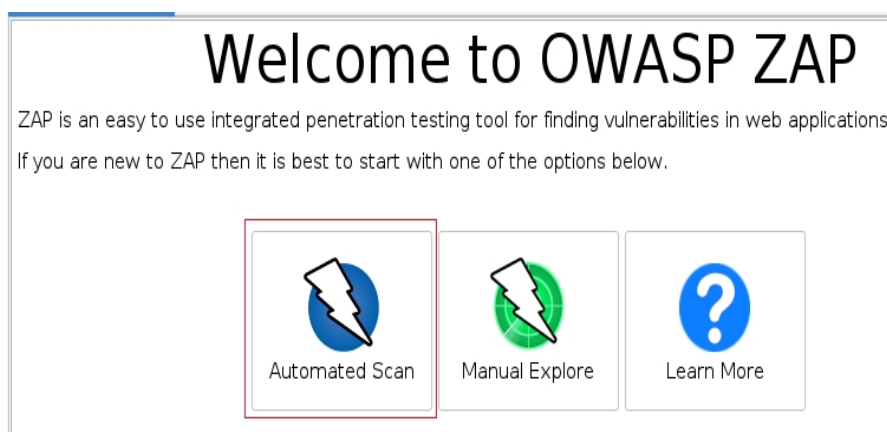
me again and click Start.



The Manage Add-ons window will open asking for updates. The updates are not needed for this lab, but they can be installed. Be patient; the updates will take some time.



In the OWASP ZAP window, in the middle right pane, click Automated Scan:




In the middle right pane, in the Automated Scan section, type the following URL in the **URL to attack** text box:

<http://10.0.2.15/bWAPP>


Remember, Linux is case-sensitive. Be sure to correctly type in the web application's name, bWAPP.



Click **Attack**.

Please be aware that you should only attack applications that you have been specifically given permission to test.

URL to attack:   Select...

Use traditional spider: ☒

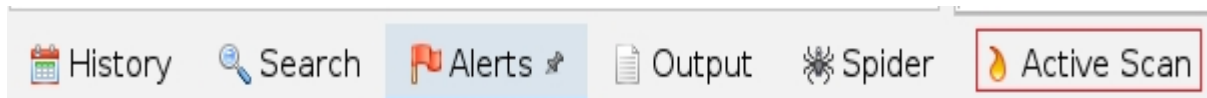
Use ajax spider: ☐ with Firefox Headless 

 Attack  Stop

Progress: Not started

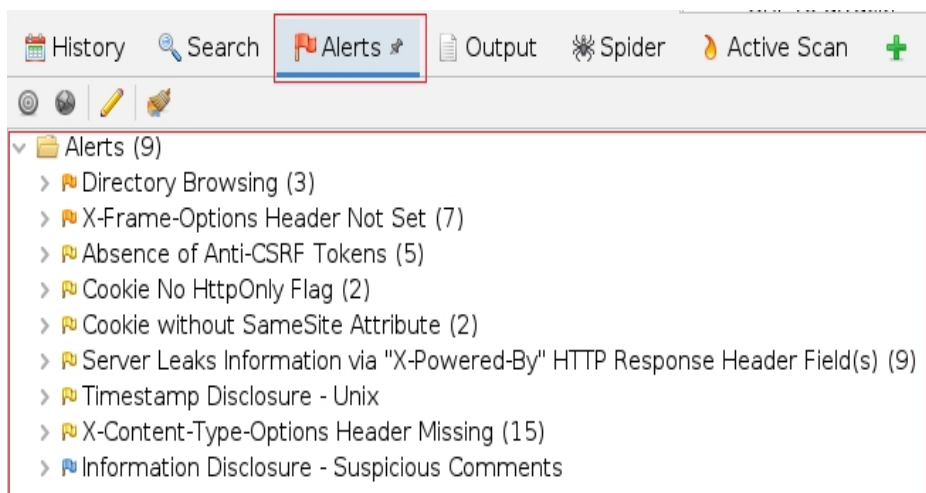
The automated attack starts. Notice that the Spider tab is now active in the bottom pane and shows current activities. The Attack button in the middle pane is now disabled, and the Stop button is active. You can manually stop the scan.

In the bottom pane, a new tab named Active Scan is displayed. It shows the current scanning activities.



When the scan is completed, the Alerts tab becomes active. This tab lists the vulnerabilities found in the bWAPP application.

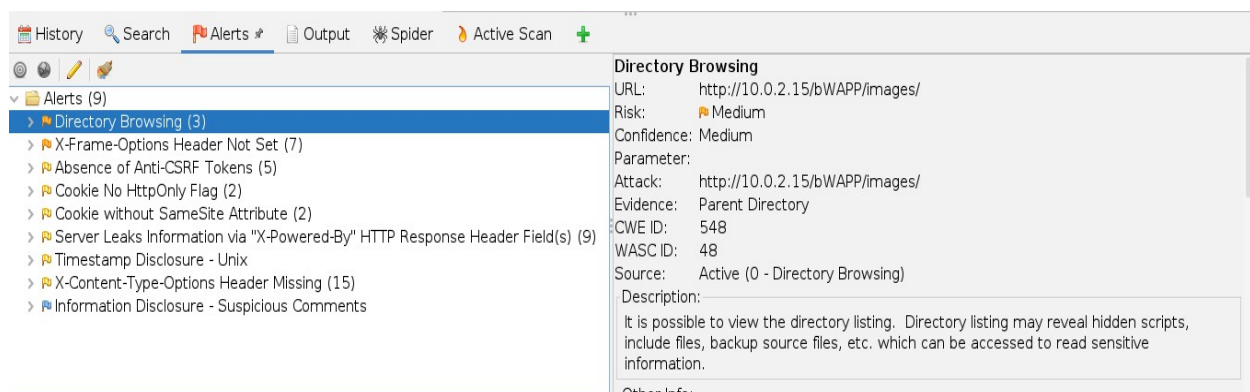
Note: You can adjust the bottom pane size for better content visibility.



From the Alerts windowpane, select Directory Browsing. When selected, the right

pane displays the details of the vulnerability:

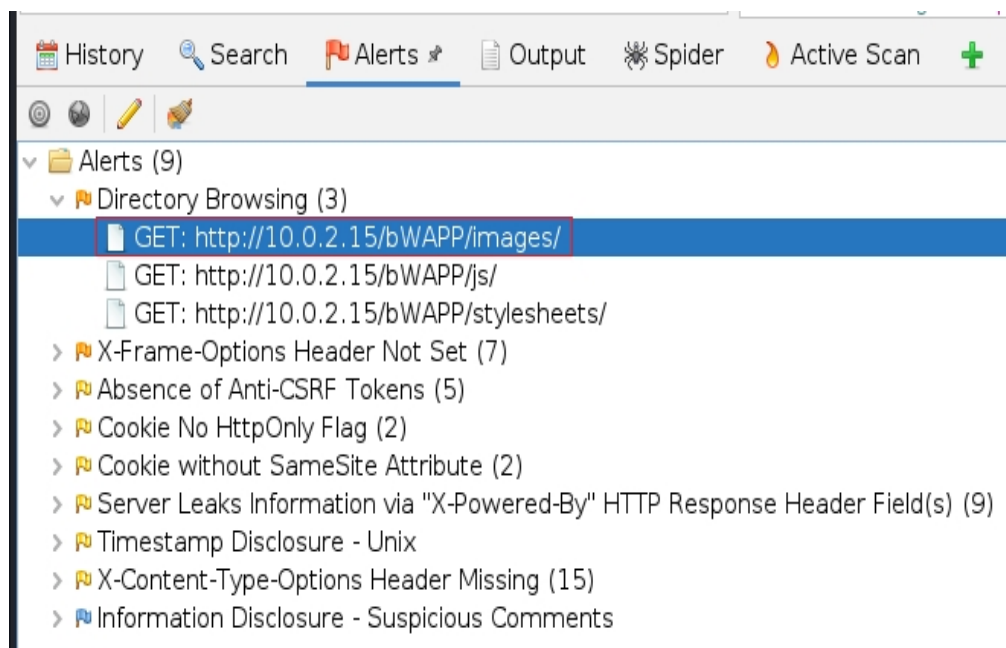




The right windowpane lists the following:

- CWE ID, which is the Common Weakness Enumeration ID
- WASC ID, which is the Web Application Security Consortium ID
- Source
- Description
- Solution
- Reference

Expand Directory Browsing. It shows three directory paths that are visible and can be navigated to. Double-click the first one with /images in the path:

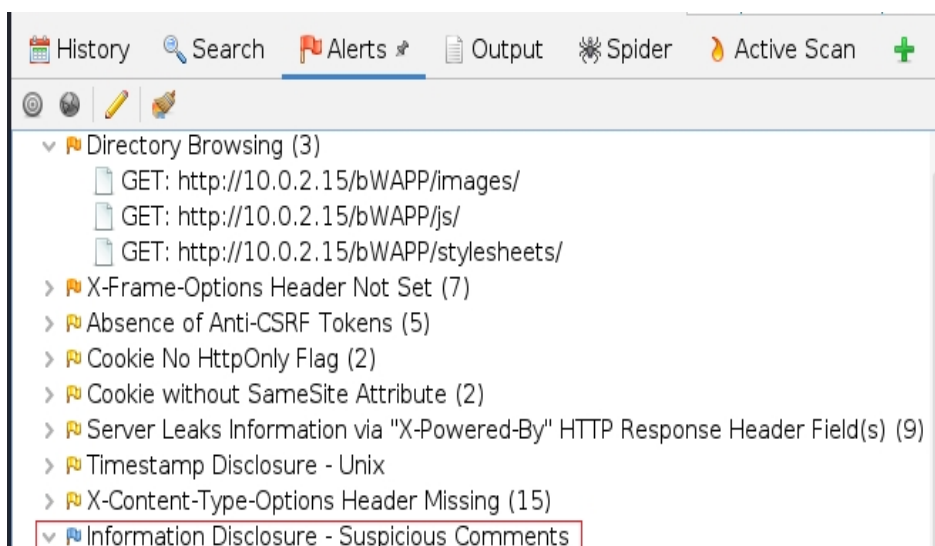


The Edit Alert dialog box is displayed. There is additional information regarding Risk, Confidence, and Attack. A solution is displayed and a link to the

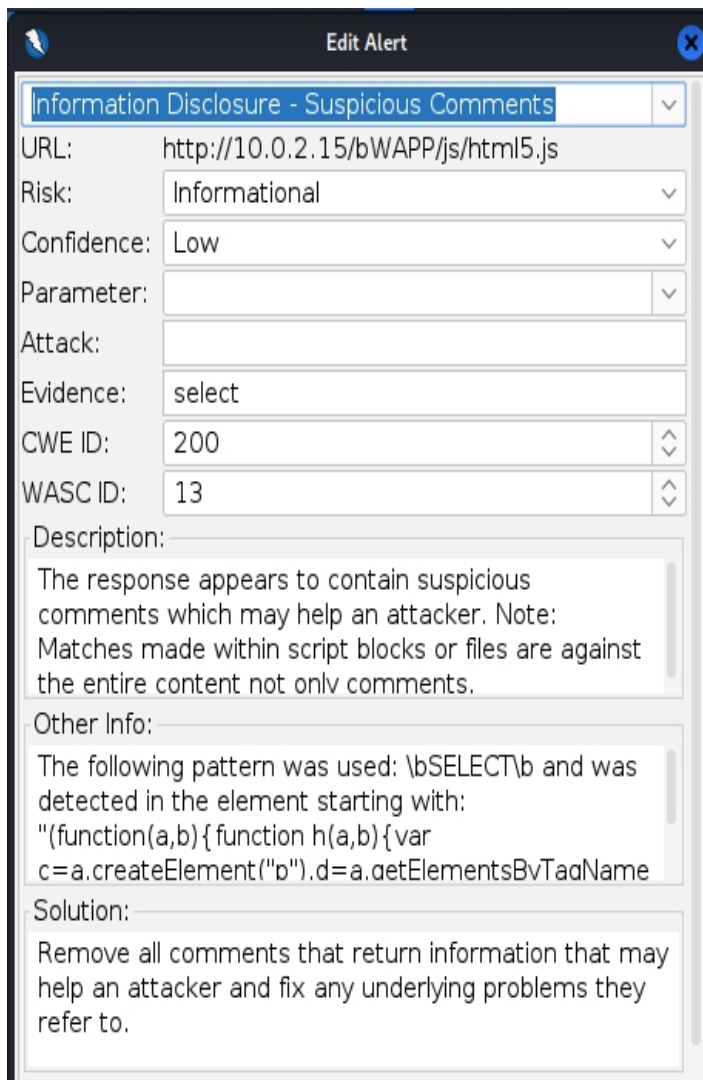
Risk, Confidence, and Attack. A solution is also mentioned. Close the dialog box.

Directory Browsing	
URL:	http://10.0.2.15/bWAPP/images/
Risk:	Medium
Confidence:	Medium
Parameter:	
Attack:	http://10.0.2.15/bWAPP/images/
Evidence:	Parent Directory
CWE ID:	548
WASC ID:	48
<b>Description:</b> It is possible to view the directory listing. Directory listing may reveal hidden scripts, include files, backup source files, etc. which can be accessed to read sensitive information.	
<b>Other Info:</b> 	
<b>Solution:</b> Disable directory browsing. If this is required, make sure the listed files does not induce risks.	
<b>Reference:</b> <a href="http://httpd.apache.org/docs/mod/core.html#options">http://httpd.apache.org/docs/mod/core.html#options</a> <a href="http://alamo.satlug.org/pipermail/satlug/2002-February/000053.html">http://alamo.satlug.org/pipermail/satlug/2002-February/000053.html</a>	

Expand Information Disclosure - Suspicious Comments and double-click the available alert:





The image shows a screenshot of the 'Edit Alert' dialog box in a security tool. The dialog has a title bar with a close button. It contains several fields for configuring an alert. The 'Title' field is set to 'Information Disclosure - Suspicious Comments'. The 'URL' field contains 'http://10.0.2.15/bWAPP/js/html5.js'. The 'Risk' field is set to 'Informational', 'Confidence' is 'Low', and 'Parameter' is empty. The 'Attack' field is empty, and the 'Evidence' field contains 'select'. The 'CWE ID' is '200' and the 'WASC ID' is '13'. There are three sections at the bottom: 'Description' with a text area containing a note about suspicious comments, 'Other Info' with a text area containing a code snippet, and 'Solution' with a text area containing advice to remove comments.

Information Disclosure - Suspicious Comments

URL: http://10.0.2.15/bWAPP/js/html5.js

Risk: Informational

Confidence: Low

Parameter:

Attack:

Evidence: select

CWE ID: 200

WASC ID: 13

Description:

The response appears to contain suspicious comments which may help an attacker. Note: Matches made within script blocks or files are against the entire content not only comments.

Other Info:

The following pattern was used: \bSELECT\b and was detected in the element starting with:  
"(function(a,b){ function h(a,b){ var  
c=a.createElement("p").d=a.getElementsBvTadName

Solution:

Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.

The Edit Alert dialog box is displayed. It is important to pay attention to the Other Info section. It mentions the code that exists as a comment, which can be helpful for an attacker.

Close the dialog box.

End of the lab!

