ETHICAL HACKING
LAB SERIES

Lab 4: Web Pentesting with Nikto & OWSP Zap

Material in this Lab Aligns to the Following Certification Domains/Objectives		
Certified Ethical Hacking (CEH) Domains	Offensive Security (PWK) Objectives	SANS GPEN Objectives
12: Hacking Webservers 13: Hacking Web Applications	14: Web Application Attacks	6: General Web Application Probing

Document Version: 2016-03-09

Contents

Introduction

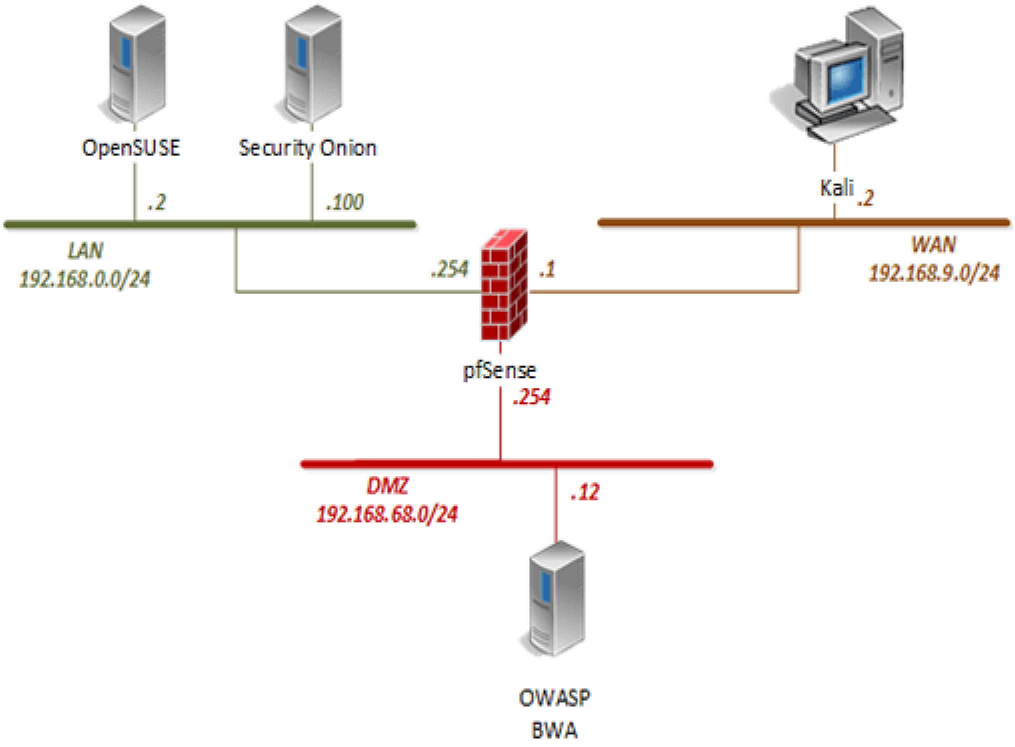
Enterprise applications are increasingly using web interfaces for their user interface. This lab uses two well-known web application assessment tools for conducting security assessments.

Objective

In this lab, you will be conducting ethical hacking practices using various tools. You will be performing the following tasks:

1. Scanning With Nikto
2. Scanning With OWASP Zap

Pod Topology



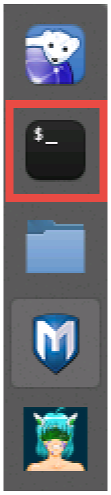
Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

Virtual Machine	IP Address	Account (if needed)	Password (if needed)
Kali Linux	192.168.9.2	root	toor
pfSense	192.168.0.254 192.168.68.254 192.168.9.1	admin	pfsense
OWASP Broken Web App	192.168.68.12	root	owaspbwa
OpenSUSE	192.168.0.2	osboxes	osboxes.org
Security Onion	192.168.0.100	ndg	password123

1 Scanning With Nikto

1. Click on the **Kali** graphic on the *topology page*.
2. Click anywhere within the *Kali* console window and press **Enter** to display the login prompt.
3. Enter **`root`** as the *username*. Click **Next**.
4. Enter **`toor`** as the *password*. Click **Sign In**.
5. Open the *Terminal* by clicking on the **Terminal** icon located on the left panel.

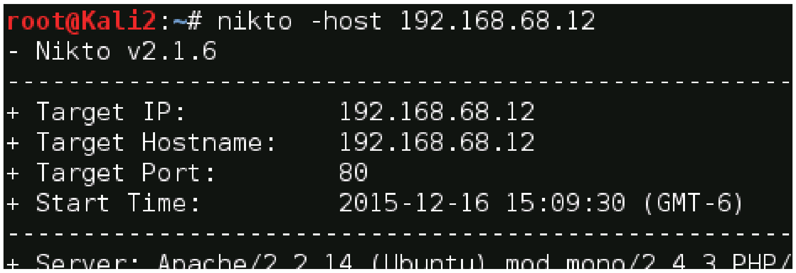


6. In the new *Terminal* window, observe the options available for *nikto*. Type the command below followed by pressing the **Enter** key.

```
nikto -help
```

7. Type the *nikto* command below to initiate a host scan with no options followed by pressing the **Enter** key.

```
nikto -host 192.168.68.12
```



```
root@Kali2:~# nikto -host 192.168.68.12
- Nikto v2.1.6
---------------------------------------------------
+ Target IP:          192.168.68.12
+ Target Hostname:    192.168.68.12
+ Target Port:        80
+ Start Time:         2015-12-16 15:09:30 (GMT-6)
---------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/
```

8. Once the scan completes, notice the large amount of information given. To narrow down the scan, first check which *nikto* plugins are available. Enter the command below.

```
nikto -list-plugins
```

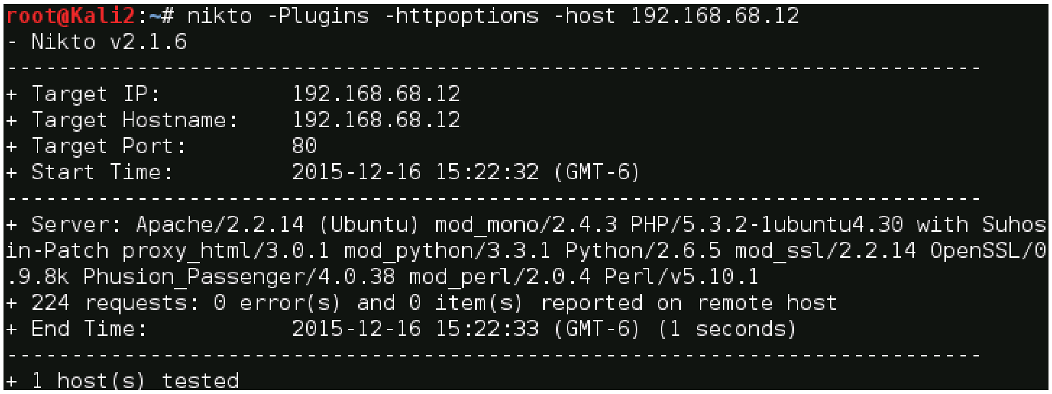
9. After examining the plugins, test the versions of software on the server. Enter the command below.

```
nikto -Plugins outdated -host 192.168.68.12
```

> Make sure to include a capital "P" in the word Plugins, otherwise, the command will not be accepted properly.

10. Check for the *HTTP* options the server accepts.

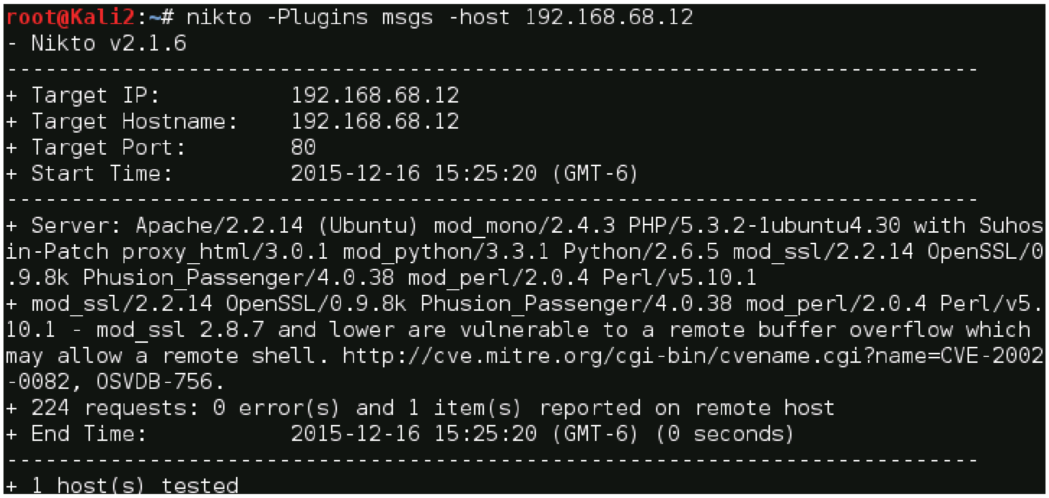
```
nikto -Plugins -httpoptions -host 192.168.68.12
```



```
root@Kali2:~# nikto -Plugins -httpoptions -host 192.168.68.12
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.68.12
+ Target Hostname:    192.168.68.12
+ Target Port:        80
+ Start Time:         2015-12-16 15:22:32 (GMT-6)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhos
in-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0
.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
+ 224 requests: 0 error(s) and 0 item(s) reported on remote host
+ End Time:           2015-12-16 15:22:33 (GMT-6) (1 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

11. Notice the server accepts all HTTP options and is susceptible to cross-site tracing. Check which client policies the server accepts. Enter the command below.

```
nikto -Plugins msgs -host 192.168.68.12
```

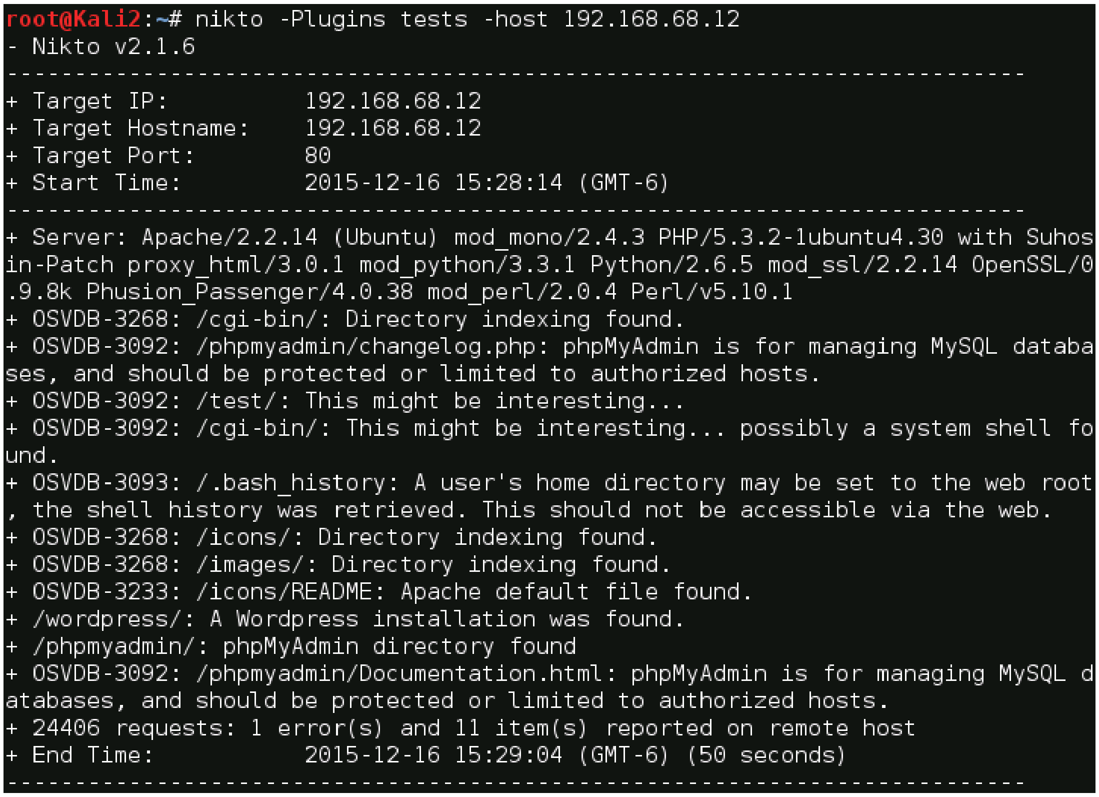


```
root@Kali2:~# nikto -Plugins msgs -host 192.168.68.12
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.68.12
+ Target Hostname:    192.168.68.12
+ Target Port:        80
+ Start Time:         2015-12-16 15:25:20 (GMT-6)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhos
in-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0
.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
+ mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.
10.1 - mod_ssl 2.8.7 and lower are vulnerable to a remote buffer overflow which
may allow a remote shell. http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2002
-0082, OSVDB-756.
+ 224 requests: 0 error(s) and 1 item(s) reported on remote host
+ End Time:           2015-12-16 15:25:20 (GMT-6) (0 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

12. Notice the server is susceptible to buffer overflow. Try a set of standard nikto tests against the server.

```
nikto -Plugins tests -host 192.168.68.12
```

After the scan completes, notice a number of vulnerabilities from the *Open Source Vulnerability Database (OSVDB)*.

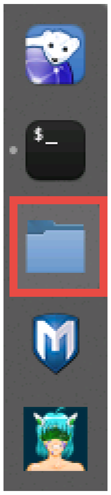


```
root@Kali2:~# nikto -Plugins tests -host 192.168.68.12
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.68.12
+ Target Hostname:    192.168.68.12
+ Target Port:        80
+ Start Time:         2015-12-16 15:28:14 (GMT-6)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhos
in-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0
.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
+ OSVDB-3268: /cgi-bin/: Directory indexing found.
+ OSVDB-3092: /phpmyadmin/changelog.php: phpMyAdmin is for managing MySQL databa
ses, and should be protected or limited to authorized hosts.
+ OSVDB-3092: /test/: This might be interesting...
+ OSVDB-3092: /cgi-bin/: This might be interesting... possibly a system shell fo
und.
+ OSVDB-3093: /.bash_history: A user's home directory may be set to the web root
, the shell history was retrieved. This should not be accessible via the web.
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3268: /images/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /wordpress/: A Wordpress installation was found.
+ /phpmyadmin/: phpMyAdmin directory found
+ OSVDB-3092: /phpmyadmin/Documentation.html: phpMyAdmin is for managing MySQL d
atabases, and should be protected or limited to authorized hosts.
+ 24406 requests: 1 error(s) and 11 item(s) reported on remote host
+ End Time:           2015-12-16 15:29:04 (GMT-6) (50 seconds)
---------------------------------------------------------------------------
```

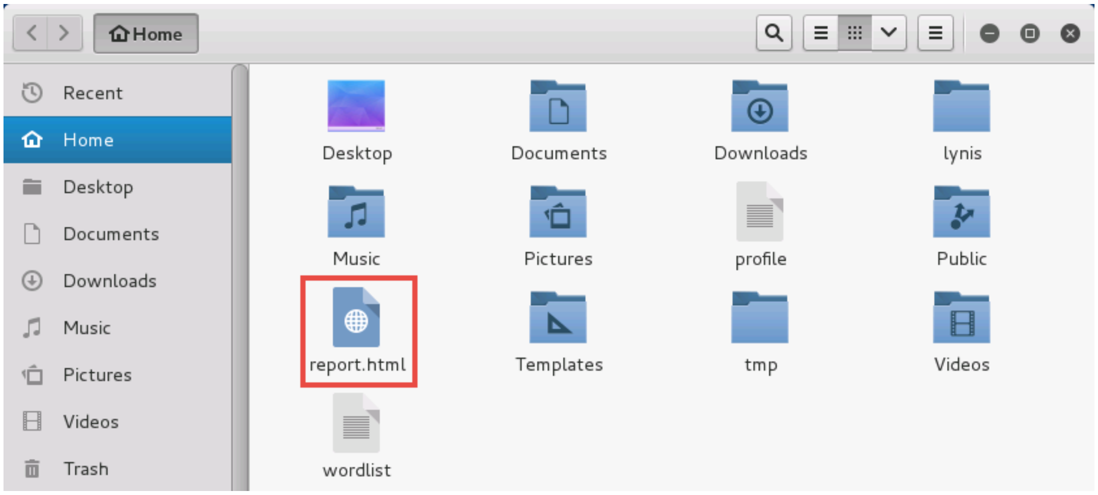
13. Now that a number of tests have been established, generate a comprehensive report in *HTML*. Type the command below followed by pressing the **Enter** key.

```
nikto -host 192.168.68.12 -output report.html
```

14. Once the operation completes, click the **Files** icon located in the left panel.



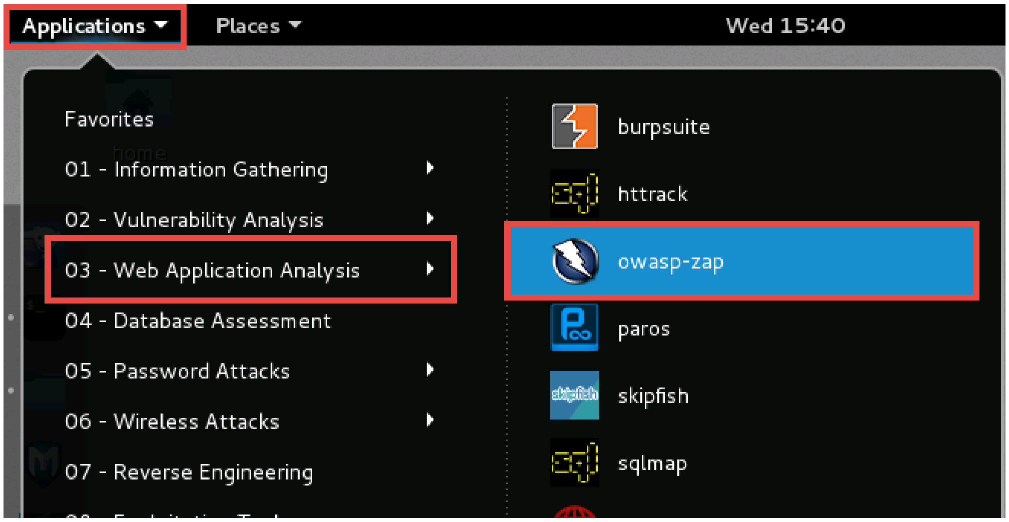
15. While viewing the *Home* directory (default), double-click on the **report.html** file.



16. Analyze the contents of the *report.html* file. When finished, minimize the *Iceweasel* browser.

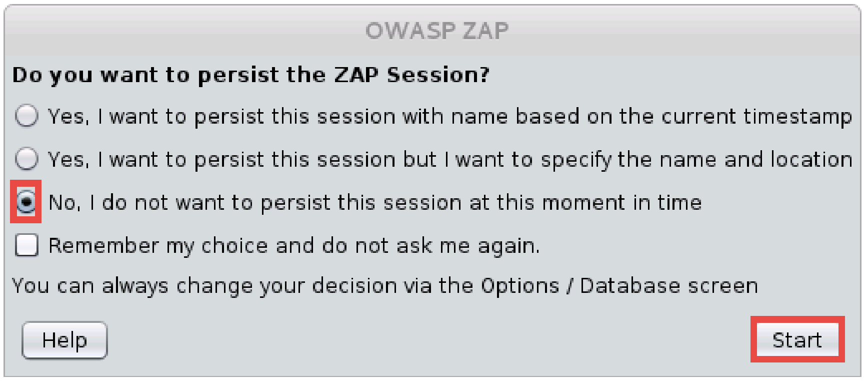
2 Scanning With OWASP Zap

1. OWASP Zap is a proxy server that can be used to delve deeper into a web server's vulnerabilities. Launch *OWASP Zap* by navigating to **Applications > Web Application Analysis > owasp-zap** using the *Application Launcher* found in the top-left corner.



The application may take 1-2 minutes to initialize and appear on the screen.

2. Upon startup, choose the **No, I do not want to persist this session at this moment in time** radio button and click **Start**.



OWASP ZAP

Do you want to persist the ZAP Session?

○ Yes, I want to persist this session with name based on the current timestamp

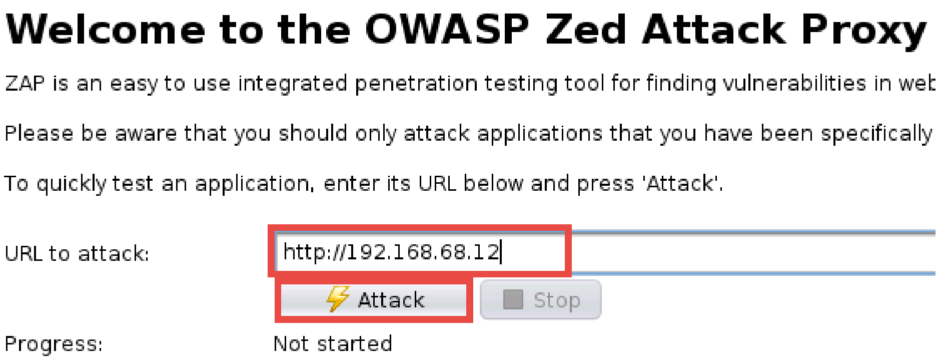
○ Yes, I want to persist this session but I want to specify the name and location

● No, I do not want to persist this session at this moment in time

□ Remember my choice and do not ask me again.

You can always change your decision via the Options / Database screen

3. Enter the IP address [`192.168.68.12`] in the **URL to attack** text box so that it reads **http://192.168.68.12**. Click the **Attack** button.



Welcome to the OWASP Zed Attack Proxy

ZAP is an easy to use integrated penetration testing tool for finding vulnerabilities in web

Please be aware that you should only attack applications that you have been specifically

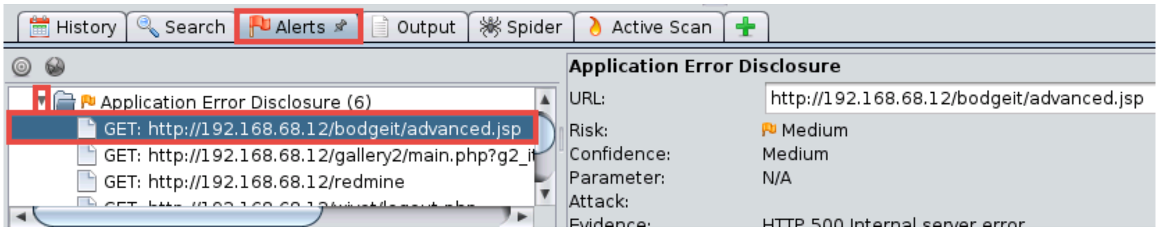
To quickly test an application, enter its URL below and press 'Attack'.

URL to attack: http://192.168.68.12

Progress: Not started

Wait about 5-8 minutes for the scan to complete before moving on to the next step.

4. Once finished, click on the **Alerts** tab located on the bottom panel.
5. In the *Alerts* panel, expand the **Application Error Disclosure** from the inventory tree and select the first **GET** request.



Application Error Disclosure (6)
GET: http://192.168.68.12/bodgeit/advanced.jsp
GET: http://192.168.68.12/gallery2/main.php?g2_i
GET: http://192.168.68.12/redmine
GET: http://192.168.68.12/wivet/logout.php

Application Error Disclosure

URL: http://192.168.68.12/bodgeit/advanced.jsp
Risk: Medium
Confidence: Medium
Parameter: N/A
Attack:
Evidence: HTTP 500 Internal server error

6. Notice the *OWASP Zap* tool dives deeper into the vulnerabilities found. Compare a few of the vulnerabilities with the *nikto report.html* file.
7. Close the **Kali** PC viewer.