ETHICAL HACKING V2
LAB SERIES

Lab 10: Web Pentesting

Document Version: 2020-08-24

Material in this Lab Aligns to the Following	
Books/Certifications	**Chapters/Modules/Objectives**
All-In-One CEH Chapters ISBN-13: 978-1260454550	6: Web-Based Hacking: Servers and Applications
EC-Council CEH v10 Domain Modules	13: Hacking Webservers 14: Hacking Web Applications 15: SQL Injection
CompTIA Pentest+ Objectives	2.2: Given a scenario, perform a vulnerability scan 2.3: Given a scenario, analyze vulnerability scan results 2.4: Explain the process of leveraging information to prepare for exploitation 3.2: Given a scenario, exploit network-based vulnerabilities 3.4: Given a scenario, exploit application-based vulnerabilities 4.2: Compare and contrast various use cases of tools 4.3: Given a scenario, analyze tool output or data related to a penetration test
CompTIA All-In-One PenTest+ Chapters ISBN-13: 978-1260135947	4: Vulnerability Scanning and Analysis 9: Web and Database Attacks

Contents

Introduction

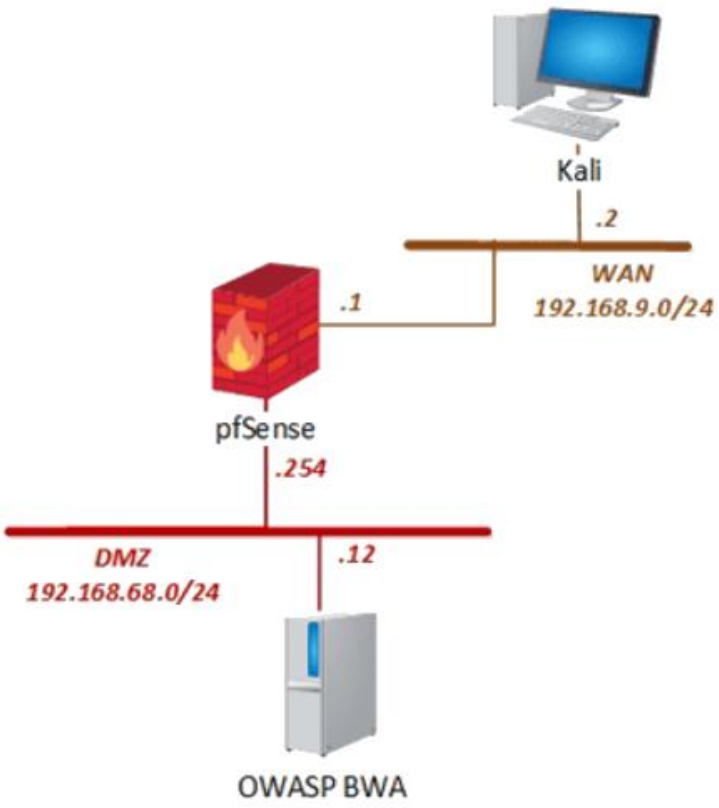
Enterprise applications are increasingly using web interfaces for their user interface. This lab uses two well-known web application assessment tools for conducting security assessments.

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. Setting up Burp Suite
3. Building a site map with Burp Suite
4. Brute Forcing a Web Application

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	192.168.9.2 192.168.0.2	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

1 Scanning With Nikto

1. Click on the **Kali** tab.
2. Click within the console window and press **Enter** to display the login prompt.
3. Enter **root** as the **username**. Press **Tab**.
4. Enter **toor** as the *password*. Click **Log In**.
5. Open a new terminal by clicking on the **Terminal** icon located at the top of the page if the terminal is not already opened.
6. In the new *Terminal* window, observe the options available for *nikto*. Type the command below, followed by pressing the **Enter** key.

```
nikto -help
```

```
root@kali:~# nikto -help
Unknown option: help

       -config+            Use this config file
       -Display+           Turn on/off display outputs
       -dbcheck            check database and other key files for syntax errors
       -Format+            save file (-o) format
       -Help               Extended help information
       -host+              target host/URL
       -id+                Host authentication to use, format is id:pass or id:pass:realm
       -list-plugins       List all available plugins

Output omitted...
```

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@kali:~# 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:         2020-07-02 22:32:07 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch prox
Output omitted...
```

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
```

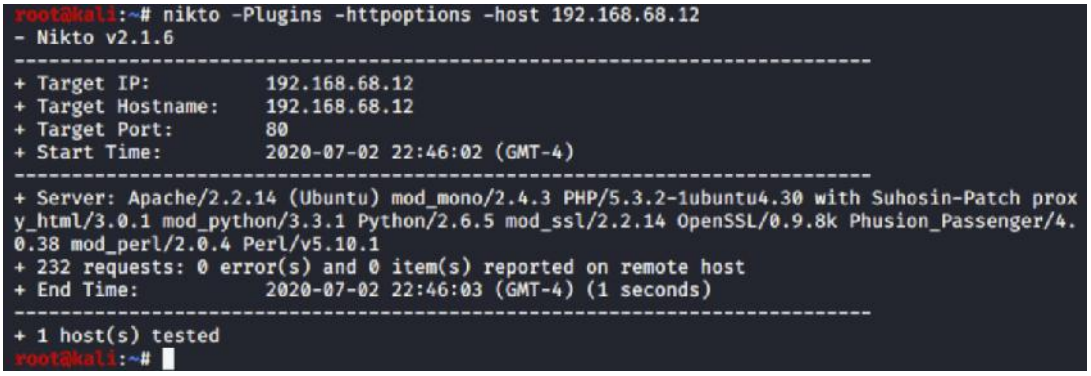

9. After examining the plugins, test the versions of software on the server. Enter the command below.

```
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.

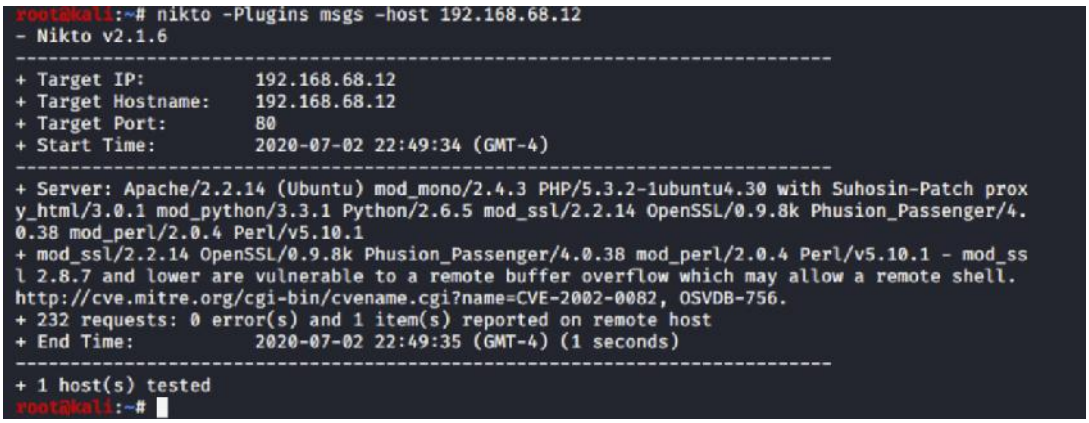
10. Check for the *HTTP* options the server accepts.

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



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
```



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)*.

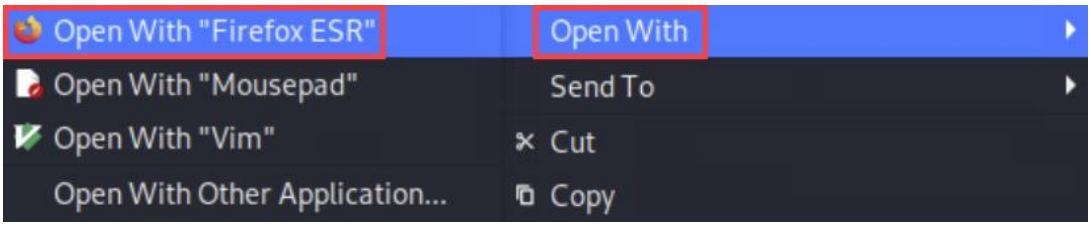
```
root@kali:~# nikto -Plugins tests -host 192.168.68.12
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.68.12
+ Target Hostname:    192.168.68.12
+ Target Port:        80
+ Start Time:         2020-07-02 22:52:17 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch prox
y_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 databases, and shou
ld be protected or limited to authorized hosts.
+ OSVDB-3092: /test/: This might be interesting...
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3268: /images/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /phpmyadmin/: phpMyAdmin directory found
+ OSVDB-3092: /phpmyadmin/Documentation.html: phpMyAdmin is for managing MySQL databases, and
 should be protected or limited to authorized hosts.
+ 24629 requests: 1 error(s) and 8 item(s) reported on remote host
+ End Time:           2020-07-02 22:53:01 (GMT-4) (44 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
root@kali:~#
```

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
```

```
root@kali:~# nikto -host 192.168.68.12 -output report.html
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.68.12
+ Target Hostname:    192.168.68.12
+ Target Port:        80
+ Start Time:         2020-07-02 22:56:16 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch prox
y_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
+ Server may leak inodes via ETags, header found with file /, inode: 286483, size: 28067, mti
me: Thu Jul 30 22:55:52 2015
Output omitted...
```

14. Once the operation completes, click on the **Folder > Open Folder** located at the top of the Desktop.
15. While viewing the *root* directory (default), right-click on **report.html** file and select **Open With > Open with "Firefox ESR"**.



The file does not seem to launch with the Chromium Web Browser, which is the default browser.

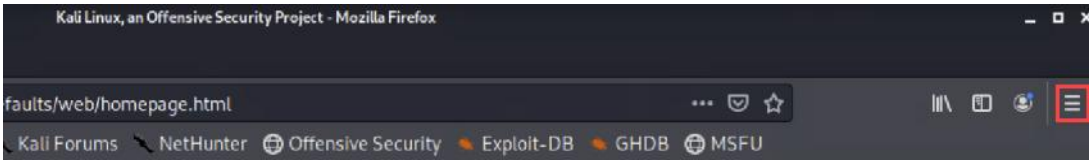
16. Analyze the contents of the *report.html* file. When finished, close the *Mozilla Firefox* browser and the *File Manager*.

2 Setting up Burp Suite

1. Open a terminal and type the following command:

```
burpsuite
```

2. In the Java popup, click **OK** to continue.
3. Review the Terms and Conditions, then click **I Accept** to continue.
4. Make sure **Temporary project** is selected and then click **Next.**
5. Make sure **Use Burp defaults** is selected and then click **Start Burp.**
6. In the *Burp Suite is out of date* window, click **OK** to continue. This will not affect the lab.
7. *Burp Suite* primarily functions with the *Proxy* and the *Target* tabs. *Burp Suite* functions as a proxy to capture traffic. Before we can start capturing traffic, we need to enable the proxy in Mozilla Firefox. Click on the **Applications > Web Browser** link from the Desktop.
8. Click on the **Menu** icon in the upper right, then select **Preferences.**



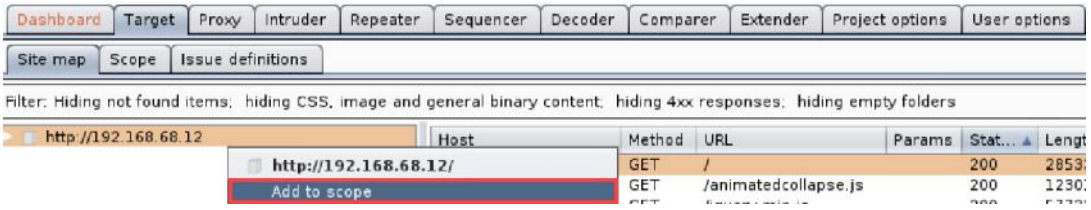
9. Scroll to the bottom and click on **Settings…** under Network Settings.
10. Select **Manual proxy configuration** and add the information below:

HTTP Proxy	Port
127.0.0.1	8080

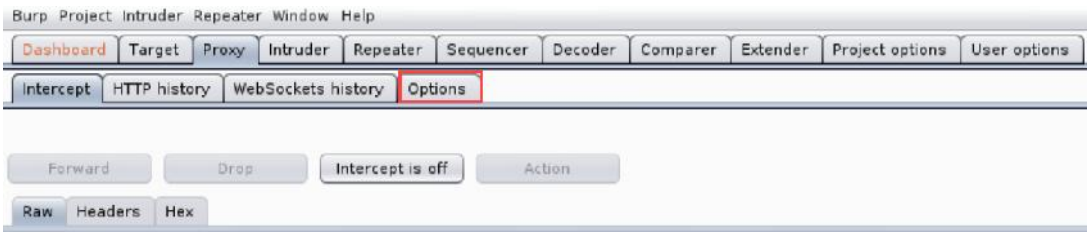
11. Click the checkbox for **Use this proxy server for all protocols.**
12. Click on **OK** to save settings.
13. Click on the **X** to close the preferences tab.
14. Click back on the **Burp Suite** tab.
15. Click on the **Proxy** tab. You may notice a request to http://detectportal.firefox.com. Right now, *Burp Suite* is intercepting all traffic outbound through the browser. You could drop individual requests or forward them on.

16. For the purposes of this lab, we want to turn intercept off, and just allow Burp Suite to log information. Click on **Intercept is on** to turn it off.

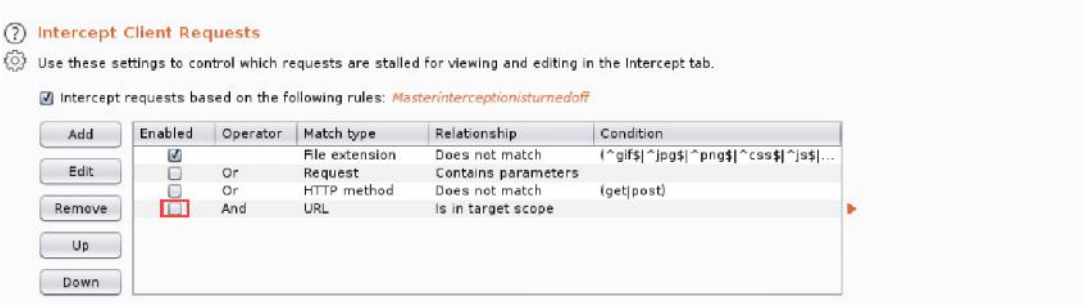
17. Click back on the **Mozilla Firefox** tab at the top.
18. In the address bar, type **`192.168.68.12`** and press **Enter.**
19. Once the page has loaded, click on the **Burp Suite** tab at the top to switch back.
20. Click on the **Target** tab.
21. If you were to browse various sites, they would start showing up on the left pane. Because this lab is a sandbox environment without internet access, there will be minimal traffic here. However, you want to focus your attention on the 192.168.68.12 web server. You will need to add this host to the scope for filtering. Right-click on the **http://192.168.68.12** host and select **Add to scope.**



22. In the popup window, it asks if you want to stop logging items outside of the scope. Click on **Yes** so we can focus on the OWASP web server.
23. Click on the **Scope** tab under the *Target* tab and see that the host has been added to the **Include in scope.**
24. This will filter the site map, but it will not filter the intercept mode of the proxy unless you specify it to. Click on the **Proxy** tab, then click on the **Options** tab.



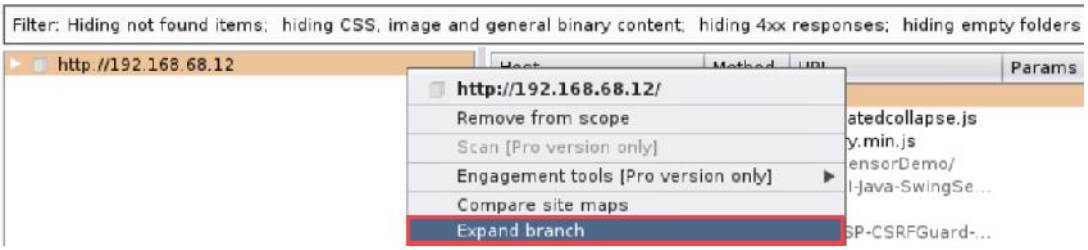
25. Under the **Intercept Client Requests** section, click on the checkbox for **And URL Is in target scope.** This will filter down to just the scope you specified before.



26. You will use this later to gather information from the website.

3 Building a site map with Burp Suite

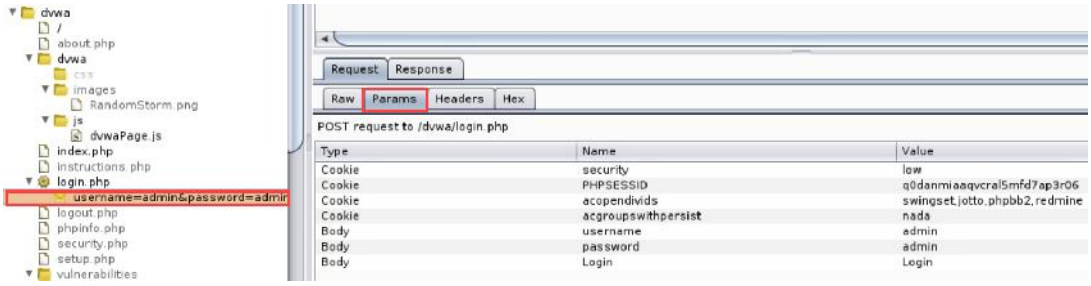
1. Click on the **Target** tab and then click on the **Site map** tab.
2. Note that *Burp Suite* is creating a site map of the site. To see this, right-click on **http://192.168.68.12** host and select **Expand branch.**



3. Even though you just went to the main page, *Burp Suite* has collected quite a bit of information. Items are grey when you have not requested them, while those in black you have requested. Notice that the *dvwa* file is grey. To expand this further, click on the **Mozzilla Firefox** tab at the top.
4. Scroll down and click on **Damn Vulnerable Web Application**.

5. Log in to the page with the username **`admin`** and the password **`admin`**.
6. Click back on the **Burp Suite** tab at the top.
7. Notice that *dvwa* is now black and shows a folder. Right-click on the **dvwa** folder and select **Expand branch.**
8. *Burp Suite* has started mapping out the web server even further, including logging information such as usernames and passwords, as indicated under the *login.php* file. Click on the **username=admin&password=admin** link below *login.php.*

9. Under the *Request* tab, click on the **Params** tab to clearly view the request parameters. Here you can access the cookie information, as well as the username and password information.

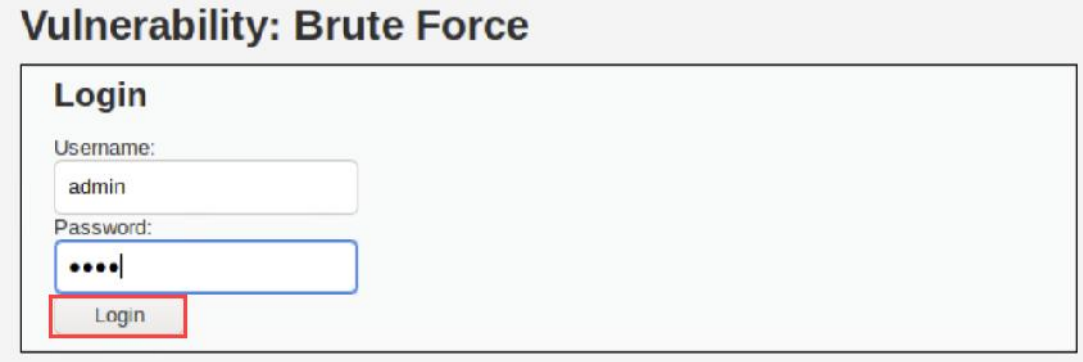


You will see this information used in a later lab about SQL injections.

4 Brute Forcing a Web Application

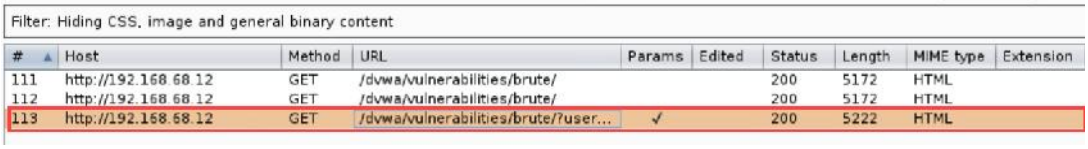
In this section, you will use the Intruder Attack feature in BurpSuite to brute force a login page.

1. Click on the **Proxy** tab, and then click on the **HTTP history** tab.
2. Right-click anywhere in the list in the middle pane and select **Clear history.**
3. Click on the **Yes** button to confirm.
4. Click on the **Mozilla Firefox** tab at the top of the screen to switch back to the DVWA application.
5. In the left pane, click on **Brute Force.**
6. In the *Username* field, type **`admin`** and in the *Password* field, type **`pass`**. Click on the **Login** button to continue.



Notice the application gives the feedback, "Username and/or password incorrect."

7. Click on the **Burp Suite** tab at the top of the screen.
8. Click on the line with a **GET** method that has the **Params** column with a check.

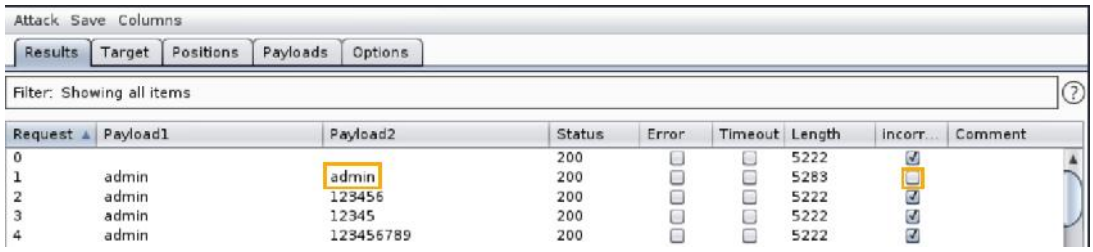


9. In the *Raw* tab below, we can see it highlights different variables. Right-click on the line you selected in the middle and select **Send to Intruder.**
10. Notice the **Intruder** tab went orange in color. Click on the **Intruder** tab.
11. The *Target* tab has been filled out based on the information from the previous screen. Click on the **Positions** tab.
12. In the *Attack type* dropdown, select **Cluster bomb.**
13. In the pane below, notice all the variables are highlighted. You only want to focus on *$admin$* and *$pass$* as they were the options you tried on the page. Click on the **Clear $** button on the right to clear the selected variables.
14. Double-click on the word **admin.** Then, click the **Add $** button.

15. Double-click on the word **pass.** Then, click the **Add $** button.

16. Click on the **Payloads** tab.
17. The first payload set will be for the username. You are looking to just brute force for a single user, *admin*. In the *Add* box under *Payload Options*, enter **`admin`** then click the **Add** button. You should see *admin* added to the list above.
18. Click on the dropdown for **Payload set** under the *Payload Sets* section. Select **2.**
19. This will allow you to manipulate the password field. Change the dropdown for *Payload type* to **Runtime file**. This will allow you to use a dictionary list.
20. In the *Payload Options* section, click on the **Select file…** button.
21. In the *Look In* dropdown, select the **/** directory.
22. Navigate to the **/usr/share/wordlists/metaploit/** folder and double-click on the **unix_passwords.txt** file.
23. Click the **Options** tab. Scroll down to the **Grep - Match** section. You need to add a rule so we can determine if the password used is correct. Remember the feedback from Step 7 had the phrase, "Username and/or password incorrect", when there was an invalid entry. We need to grep on the word "incorrect" to mark the invalid passwords, leaving those not marked as valid passwords.
24. Click the **Clear** button to clear the existing flags. In the *Confirm* window, click **Yes.**
25. In the *Add* field, type **`incorrect`** and then click the **Add** button. You should see **incorrect** added to the list.

26. You are now ready to begin your attack. Click on the **Intruder** tab.
27. Then click on the **Start attack** button.
28. In this Community Edition of *Burp Suite*, the attacks are time throttled. Click on **OK** to continue.
29. As it starts to attack, look in the **incorrect** column for any passwords without a checkmark. You should see the password in the **Payload2** column of **admin.** This is a valid password for the DVWA application.



This attack will continue with the many passwords on the list. You can close this window at any time. The output may differ from the screenshot.

Burp Suite has many more features available for web application analysis. However, many of those will require the use of the Pro edition.

30. You may now end your reservation.

