

# **ADVANCED WEB APPLICATION PENETRATION TESTING WITH BURP SUITE & OWASP DVWA**

**Identifying and Exploiting Web  
Vulnerabilities to Demonstrate  
Real-World Attack Surfaces**

**BY JACINTA IZUNDU**

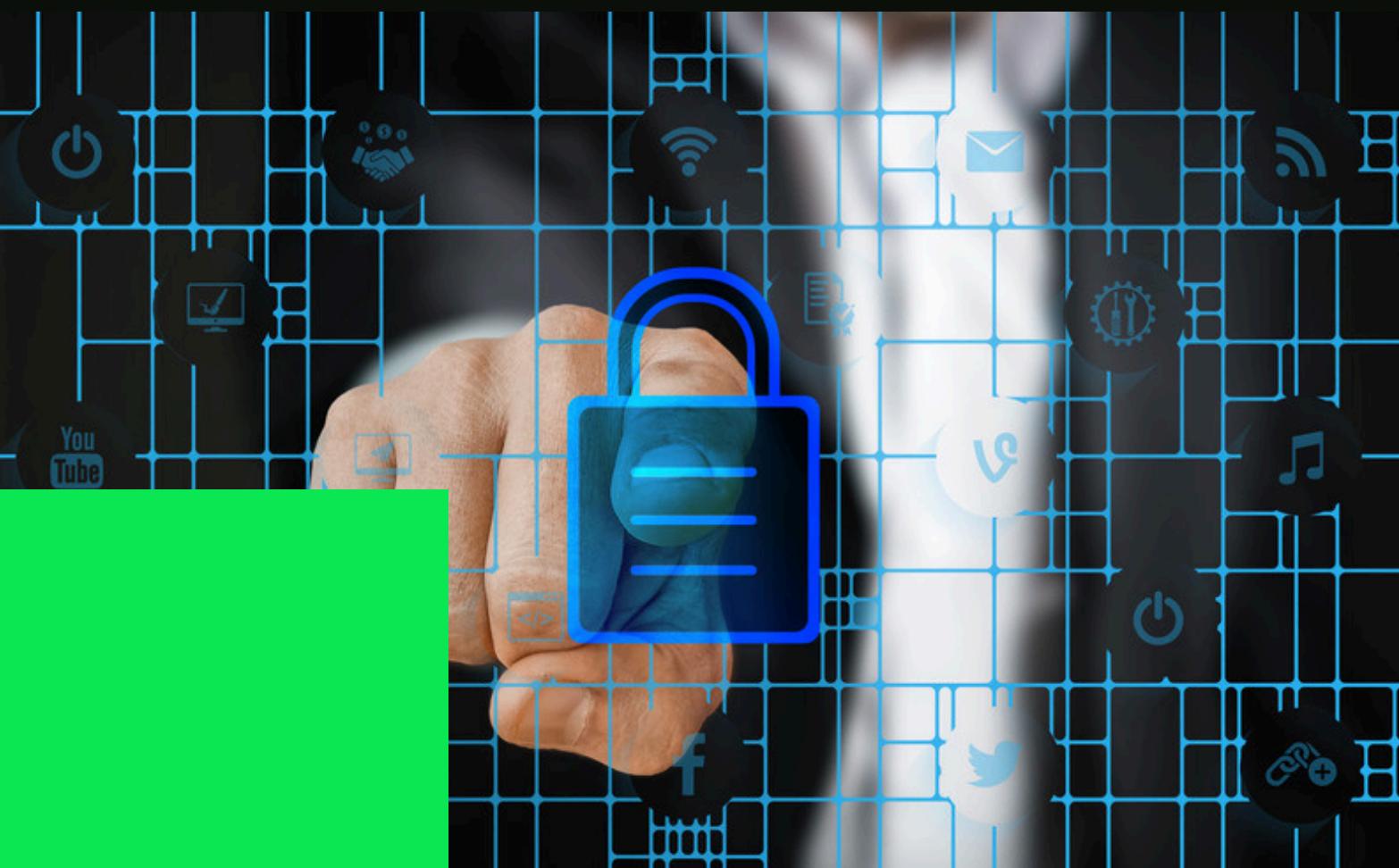
**EDUCATION**

# INTRODUCTION



- This hands-on lab teaches practical web hacking skills.
- You used tools like Nikto and Burp Suite.
- You tested a vulnerable website (OWASP BWA / DVWA).
- You practiced scanning, intercepting traffic, mapping a site, and brute forcing a login.

# LAB OBJECTIVES



1. Scanning with Nikto
  2. Setting up Burp Suite
  3. Building a Site Map in Burp Suite
  4. Performing Brute Force Attack on DVWA

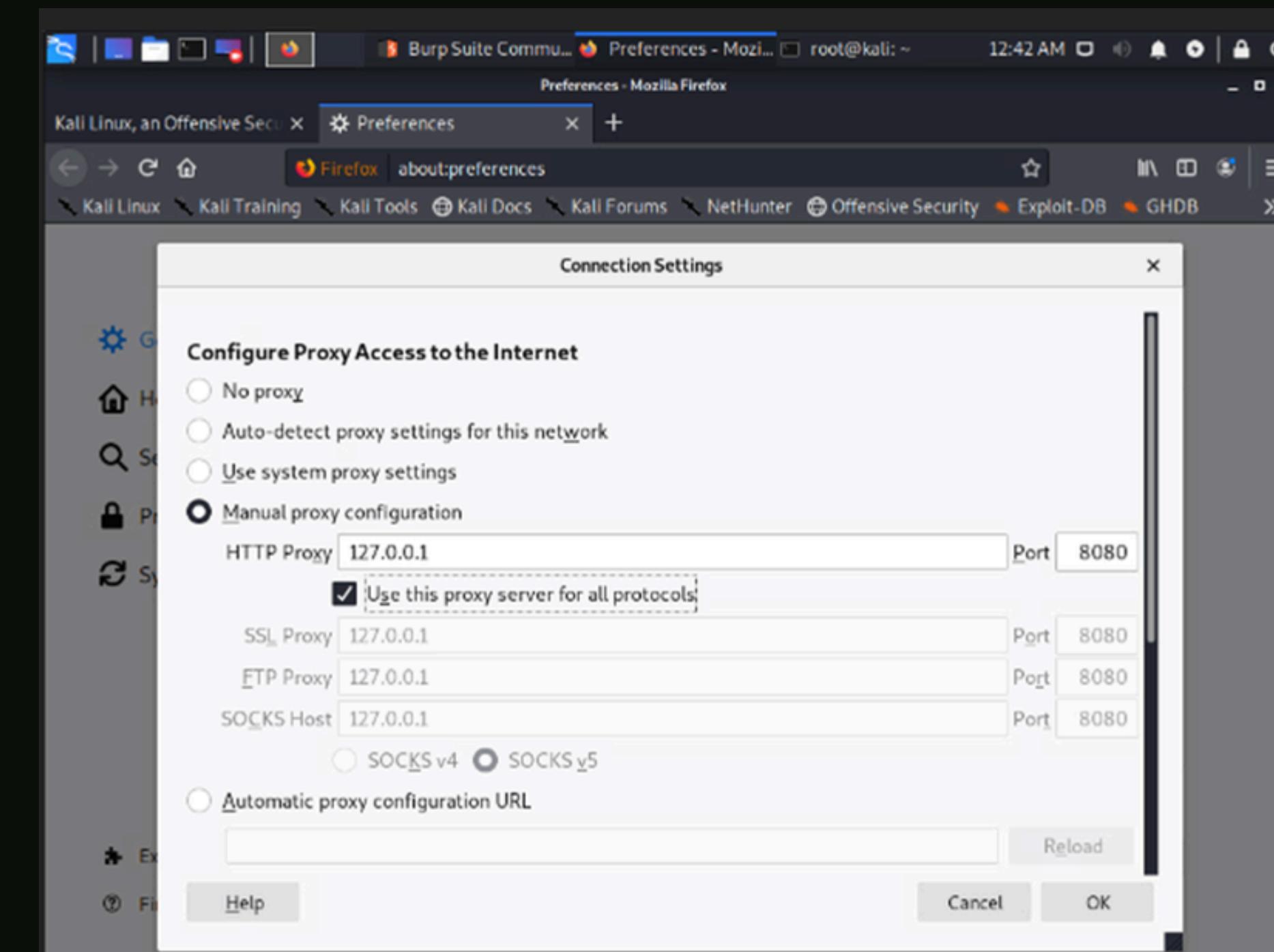
# TOOLS USED

- **Nikto:** Web server vulnerability scanner
- **Burp Suite:** Web penetration testing framework
- **Firefox:** Browser used for intercepting HTTP requests
- **OWASP BWA / DVWA:** Deliberately vulnerable target website



# NIKTO SCAN (OVERVIEW)

- I logged into Kali (root/toor)
- I ran Nikto to scan a web server
- Nikto checks for vulnerabilities, outdated software, HTTP options, plugins.



# NIKTO: PLUGIN SCAN RESULTS

- I USED NIKTO PLUGINS TO SCAN FOR:
  - Outdated software
  - HTTP options
  - Sever messages
  - Vulnerable configurations

The screenshot shows the Burp Suite interface. The top menu bar includes: Burp, Project, Intruder, Repeater, Window, Help. The sub-menu for 'Target' is open, showing options: Dashboard, Target, Proxy, Intruder, Repeater, Sequencer, Decoder, Comparer, Extender, Project options, User options. Below the menu is a toolbar with buttons: Site map, Scope, Issue definitions. A status message 'Logging of out-of-scope Proxy traffic is disabled' with a 'Re-enable' button is displayed. The main content area shows a 'Site map' view for 'http://192.168.68.12/'. A context menu is open over the entry 'http://192.168.68.12/'. The menu options are: Add to scope (highlighted), Scan (Pro version only), Engagement tools (Pro version only), Compare site maps, Expand branch, Expand requested items, Delete host, Copy URLs in this host, Copy links in this host, Save selected items, Show new site map window, and Site map documentation. To the right of the menu is a table of requests:

Method	URL	Params	Stat...	Length	MIME type	Title
GET	/		200	28533	HTML	owaspbbe OW...
GET	/animatedcollapse.js		200	12301	script	
GET	/jquery.min.js		200	57733	script	
GET	/AppSensorDemo/					
GET	/ESAPI-Java-SwingSe...					
GET	/MCIR					
GET	/OWASP-CSRFGuard...					
GET	/WackoPikko					
GET	/WebGoatAttack					
GET	/awstats/awstats.pl					
GET	/awstats/awstats.pl?...					

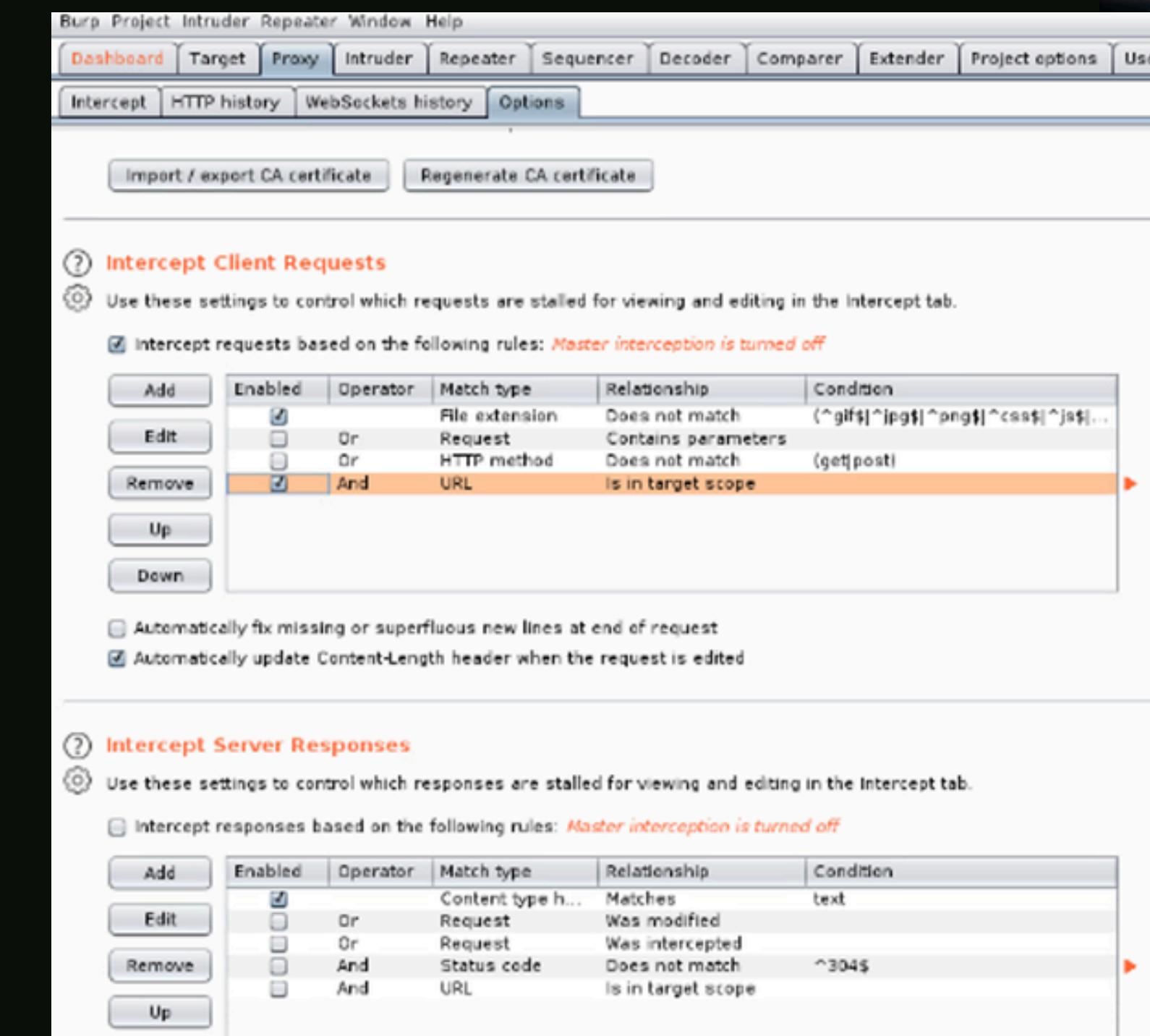
Below the table, the 'Raw' tab of a request details view is selected, showing the following HTTP request:

```
GET / HTTP/1.1
Host: 192.168.68.12
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
```

At the bottom of the interface, there is a search bar: 'Type a search term' and a status message: '0 matches'.

# NIKTO: STANDARD TESTS & OSVDB FINDINGS

- Nikto: Web server vulnerability scanner
- Burp Suite: Web penetration testing framework
- Firefox: Browser used for intercepting HTTP requests
- OWASP BWA / DVWA: Deliberately vulnerable target website



The screenshot shows the 'Intercept Client Requests' settings in Burp Suite. It includes a table for defining request interception rules and sections for intercepting server responses.

**Intercept Client Requests**

Use these settings to control which requests are stalled for viewing and editing in the Intercept tab.

Intercept requests based on the following rules: *Master interception is turned off*

Add	Enabled	Operator	Match type	Relationship	Condition
Add	<input checked="" type="checkbox"/>	File extension	Does not match	(^gif\$ ^jpg\$ ^png\$ ^css\$ ^js\$ ...	
Edit	<input type="checkbox"/>	Or	Request	Contains parameters	
Remove	<input type="checkbox"/>	Or	HTTP method	Does not match	(get post)
	<input checked="" type="checkbox"/>	And	URL	Is in target scope	

Automatically fix missing or superfluous new lines at end of request  
 Automatically update Content-Length header when the request is edited

**Intercept Server Responses**

Use these settings to control which responses are stalled for viewing and editing in the Intercept tab.

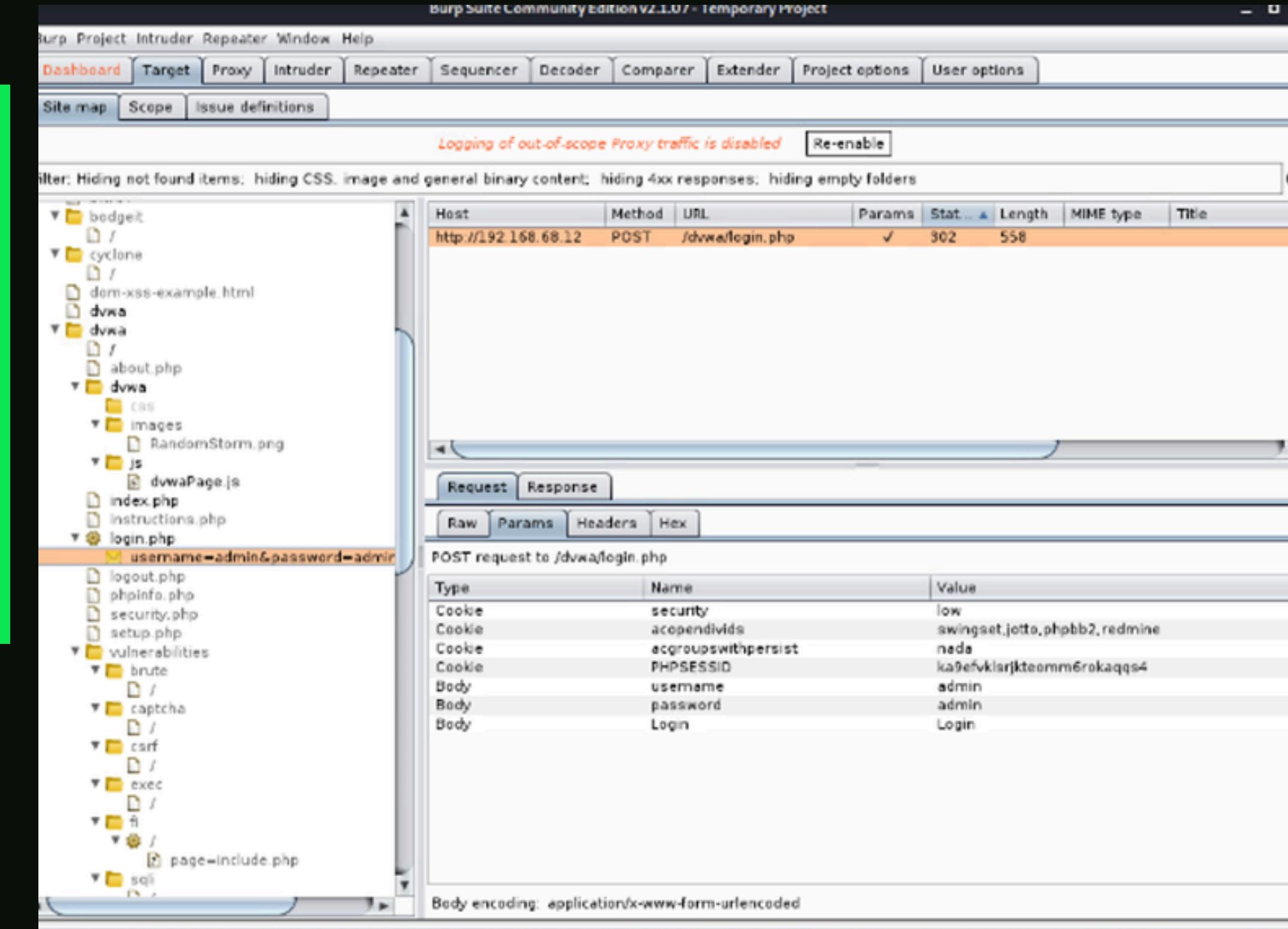
Intercept responses based on the following rules: *Master interception is turned off*

Add	Enabled	Operator	Match type	Relationship	Condition
Add	<input checked="" type="checkbox"/>	Content type h...	Matches	text	
Edit	<input type="checkbox"/>	Or	Request	Was modified	
Remove	<input type="checkbox"/>	Or	Request	Was intercepted	
	<input type="checkbox"/>	And	Status code	Does not match	^304\$
	<input type="checkbox"/>	And	URL	Is in target scope	



# NIKTO: HTML REPORT

- I generated a complete HTML vulnerability report
- Opened it in Firefox
- Reviewed findings



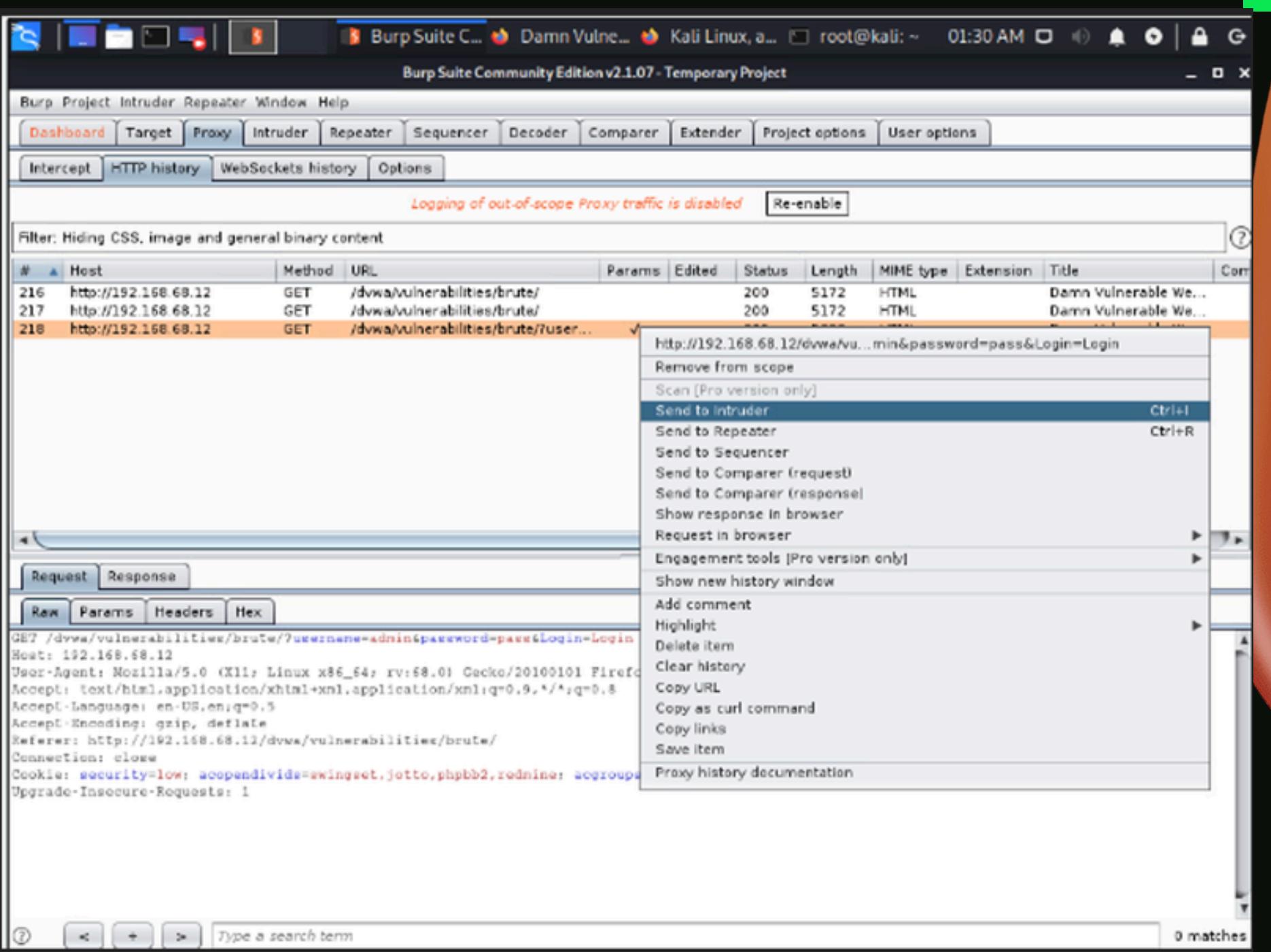
The screenshot shows the Burp Suite Community Edition interface. The title bar reads "Burp Suite Community Edition v2.1.0.7 - Temporary Project". The menu bar includes "Burp", "Project", "Intruder", "Repeater", "Window", and "Help". The tabs at the top are "Dashboard", "Target", "Proxy", "Intruder", "Repeater", "Sequencer", "Decoder", "Comparer", "Extender", "Project options", and "User options". The "Proxy" tab is selected. Below the tabs are buttons for "Site map", "Scope", and "Issue definitions". A message in the center says "Logging of out-of-scope Proxy traffic is disabled" with a "Re-enable" button. The main pane shows a list of files in a tree view on the left, including "bodgeit", "cyclone", "dvwa" (which is expanded to show "about.php", "dvwa" (expanded to show "CSS", "images" (with "RandomStorm.png"), "js" (with "dvwaPage.js"), "index.php", "instructions.php", and "login.php"), "logout.php", "phpinfo.php", "security.php", "setup.php", "vulnerabilities" (expanded to show "brute", "captcha", "csrf", "exec", "fi" (expanded to show "page=include.php"), and "sql"). On the right, a table shows a POST request to "http://192.168.68.12 /dvwa/login.php". The table columns are Host, Method, URL, Params, Status, Length, MIME type, and Title. The status is 302 and the length is 558. Below the table, the "Params" tab is selected in a sub-pane showing the POST request parameters. The parameters are:

Type	Name	Value
Cookie	security	low
Cookie	acopendivids	swingset,jotto,phpbb2,redmine
Cookie	acgroupswithpersist	nada
Cookie	PHPSESSID	ka9efvklarjkteomm6rokaqq84
Body	username	admin
Body	password	admin
Body	Login	Login

At the bottom, it says "Body encoding: application/x-www-form-urlencoded".

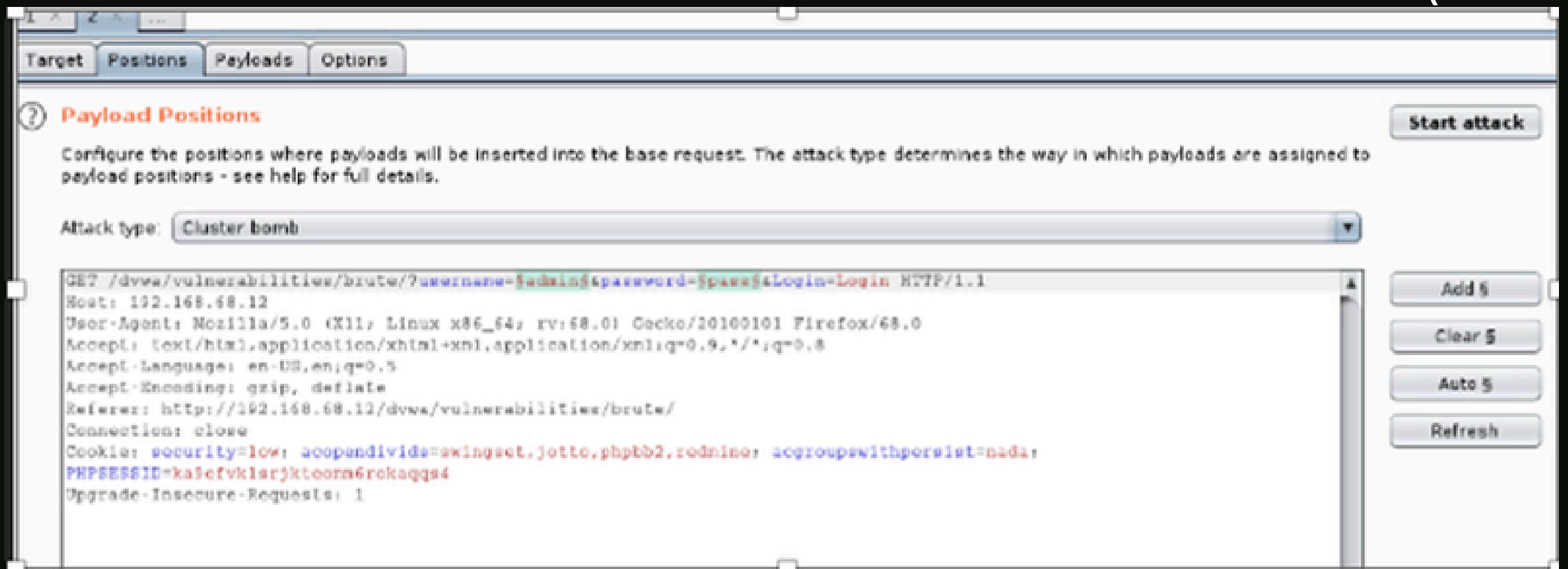
# SETTING UP BURP SUITE

- I launched Burp Suite
- Set up a temporary project
- Enabled it as Firefox's proxy
- Configured Firefox to route traffic through Burp (127.0.0.1:8080)



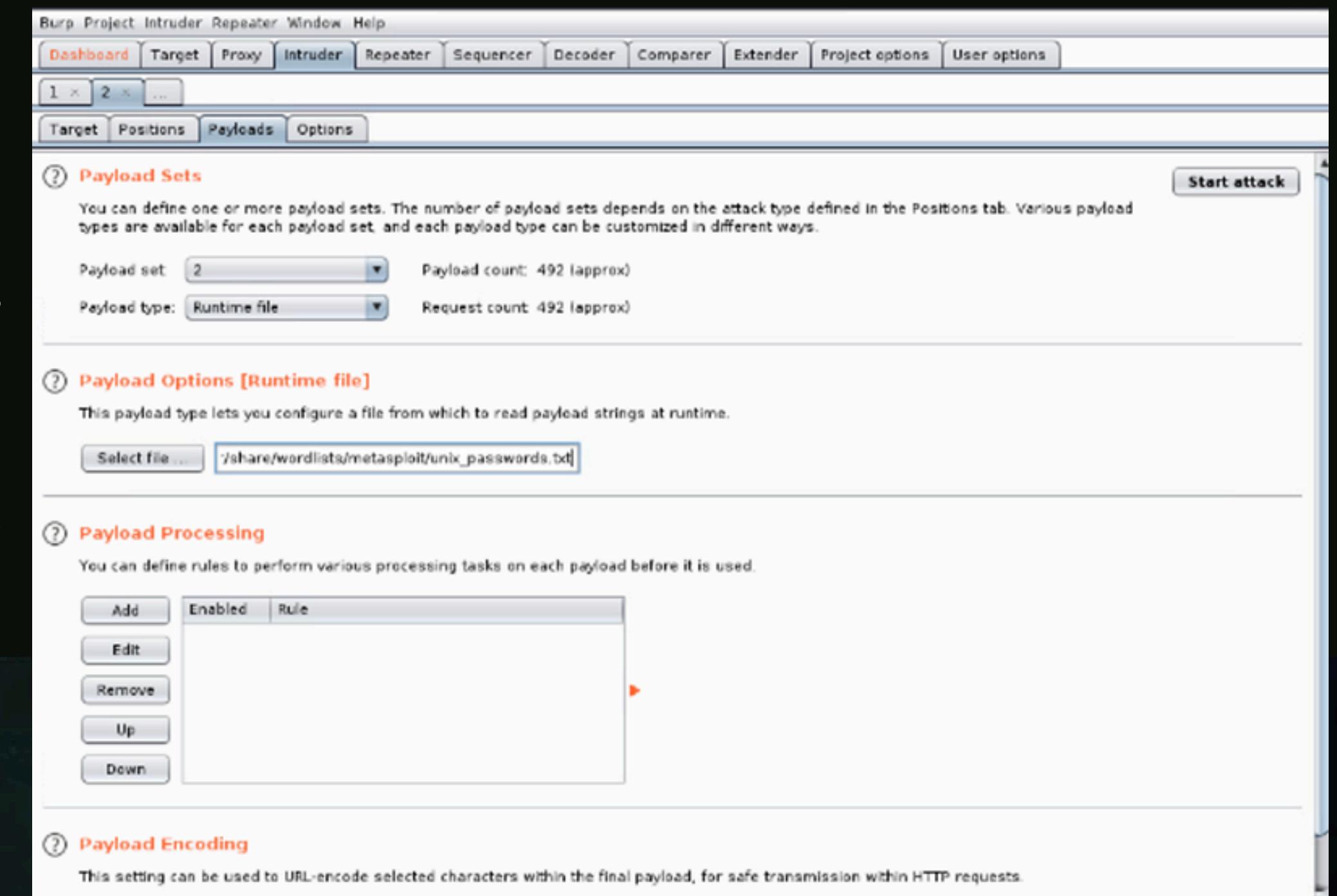
# BURP SUITE INTERCEPT MODE

- I turned intercept OFF to allow traffic to flow
- I navigated to the vulnerable web app (192.168.68.12)



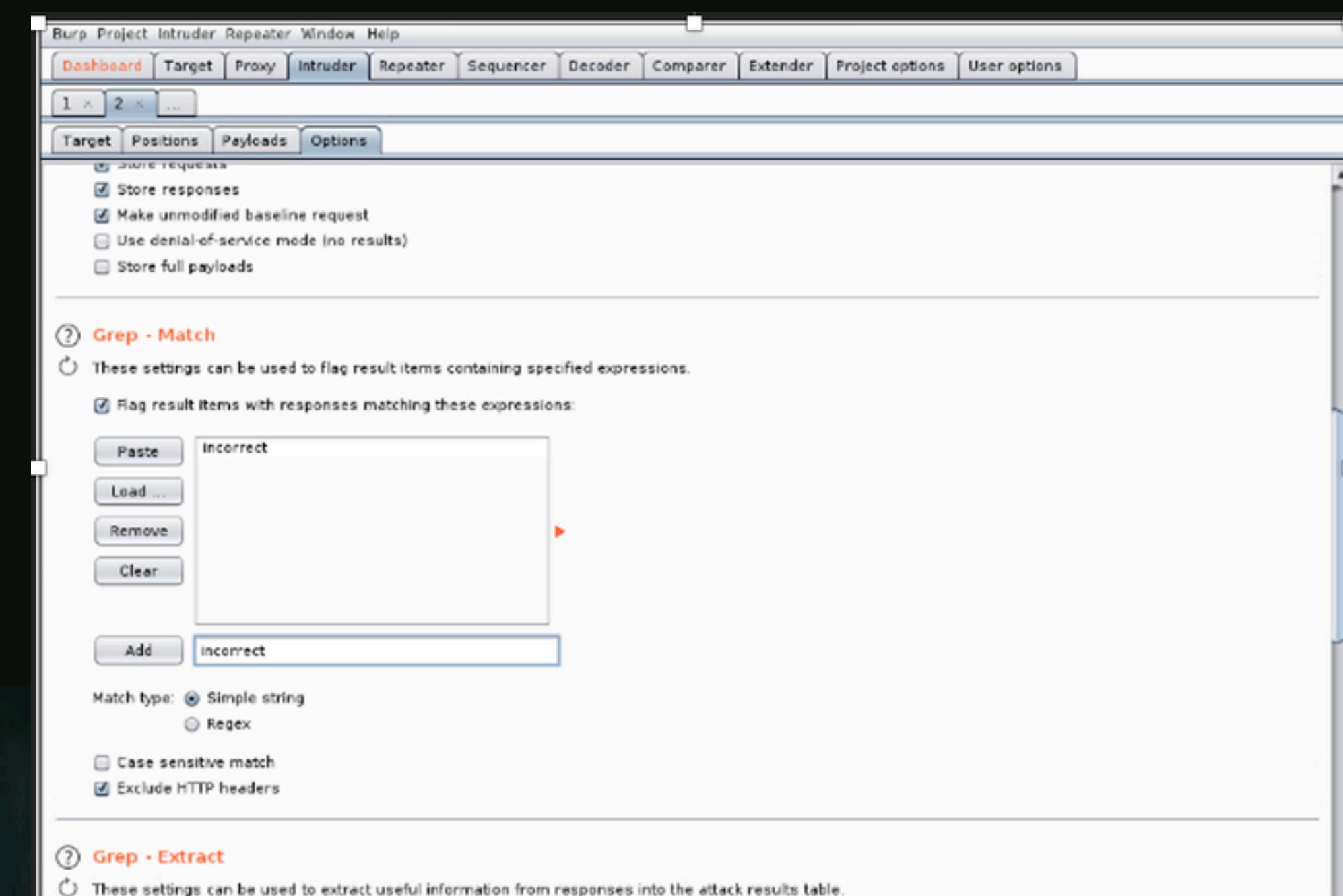
# BUILDING A SITE MAP

- Burp Suite begins mapping the site
- As I clicked DVWA, directories turned from grey → black
- Burp logs usernames, passwords, and HTTP parameters



# CAPTURED CREDENTIALS

- Burp Suite captured DVWA login parameters
- You viewed them under the “Params” tab
- Shows `username=admin` & `password=admin`



# BRUTE FORCE ATTACK (OVERVIEW)

- Used Burp Intruder to brute-force the DVWA login pagSteps:
  - Clear history
  - Submit login attempt
  - Send request to Intruder
  - Set payloads (username + password list)
  - Run attack
  -

# SENDING REQUEST TO INTRUDER

- I grabbed the login request
- Sent it to the intruder module
- Intruder turned orange

The screenshot shows the OWASp ZAP Intruder module interface. The title bar says "Intruder attack1". The main window has tabs for "Results", "Target", "Positions", "Payloads", and "Options". The "Results" tab is selected, showing a table of attack results. The table has columns: Request, Payload1, Payload2, Status, Error, Timeout, Length, Incomplete, and Comment. The "Payload2" column for the first row is highlighted in orange. The table contains 13 rows of data. Below the table, there are tabs for "Request" and "Response", with "Request" selected. The "Raw" tab is active, showing an HTTP GET request to "/dvwa/vulnerabilities/brute/". The request includes parameters: "username=admin&password=admin&Login=Login". The "Headers" tab shows standard HTTP headers. The "Params" tab shows the "username" and "password" parameters. The "Headers" tab shows the "Content-Type" header. The "Hex" tab is also present. At the bottom, there are buttons for "Type a search term" and "0 matches".

Request	Payload1	Payload2	Status	Error	Timeout	Length	Incomplete	Comment
0			200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
1	admin	admin	200	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	5283	<input type="checkbox"/>	
2	admin	123456	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
3	admin	12345	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
4	admin	123456789	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
5	admin	password	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
6	admin	iloveyou	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
7	admin	princess	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
8	admin	1234567	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
9	admin	12345678	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
10	admin	abc123	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
11	admin	nicole	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
12	admin	daniel	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	
13	admin	halimah	200	<input type="checkbox"/>	<input type="checkbox"/>	5222	<input checked="" type="checkbox"/>	

Request

Raw Params Headers Hex

```
GET /dvwa/vulnerabilities/brute/?username=admin&password=admin&Login=Login HTTP/1.1
Host: 192.168.68.12
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.68.12/dvwa/vulnerabilities/brute/
Connection: close
Cookie: security=low; acceptdivide=swingest,jetico,phpbb2,redmine; nogroupswithparent=nada;
```

Type a search term

0 matches

# CONFIGURING PAYLOAD POSITIONS

- I cleared all variables
- Selected admin and pass parameters
- Added \$ markers for both

# LOADING PASSWORD WORDLIST

- For usernames, I used **admin**
- For passwords, I loaded a wordlist
  - **/usr/share/wordlists/metasploit/unix\_passwords.txt**

# SETTING GREP-MATCH RULE

- To identify failed attempts, I added rule: **Incorrect**
- Correct password will not contain this word

# RUNNING THE ATTACK

- I started the brute force attack
- Burp Suite Community Edition slows results
- I looked for the password without an “incorrect” flag



# **SUCCESSFUL PASSWORD FOUND**

- I identified the correct DVWA password
- It appears under Payload2 without the “incorrect” mark

# KEY FINDINGS

- Nikto revealed outdated software & weak HTTP options
- Burp Suite intercepted all web traffic
- Site map exposed sensitive parameters
- Intruder successfully brute-forced DVWA login
- Demonstrates why strong passwords & secure configs matter

# CONCLUSION

- The exercise taught real-world ethical hacking skills
- Showed how vulnerable misconfigured systems can be
- Reinforced importance of monitoring, patching and hardening web apps

# THANK YOU!

Thank you for being part of the secure digital journey.  
Together, we create a safer internet for everyone.



+1 217 670 4436



[clemsjacy@gmail.com](mailto:clemsjacy@gmail.com)



<http://linkedin.com/in/jacinta-izundu-b36a20233>