# Writeup Zion: 1.0/1.1- Vulnhub

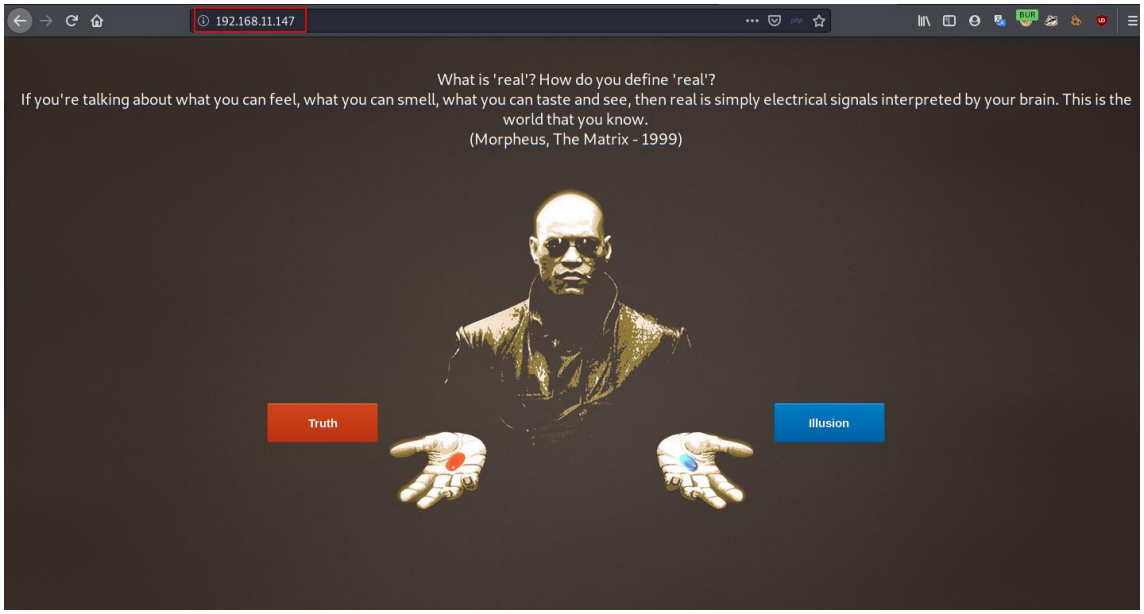
VM Creada por: @mrhenrike

Empiezo como siempre, lanzando nmap a la dirección IP para obtener los servicios disponibles.

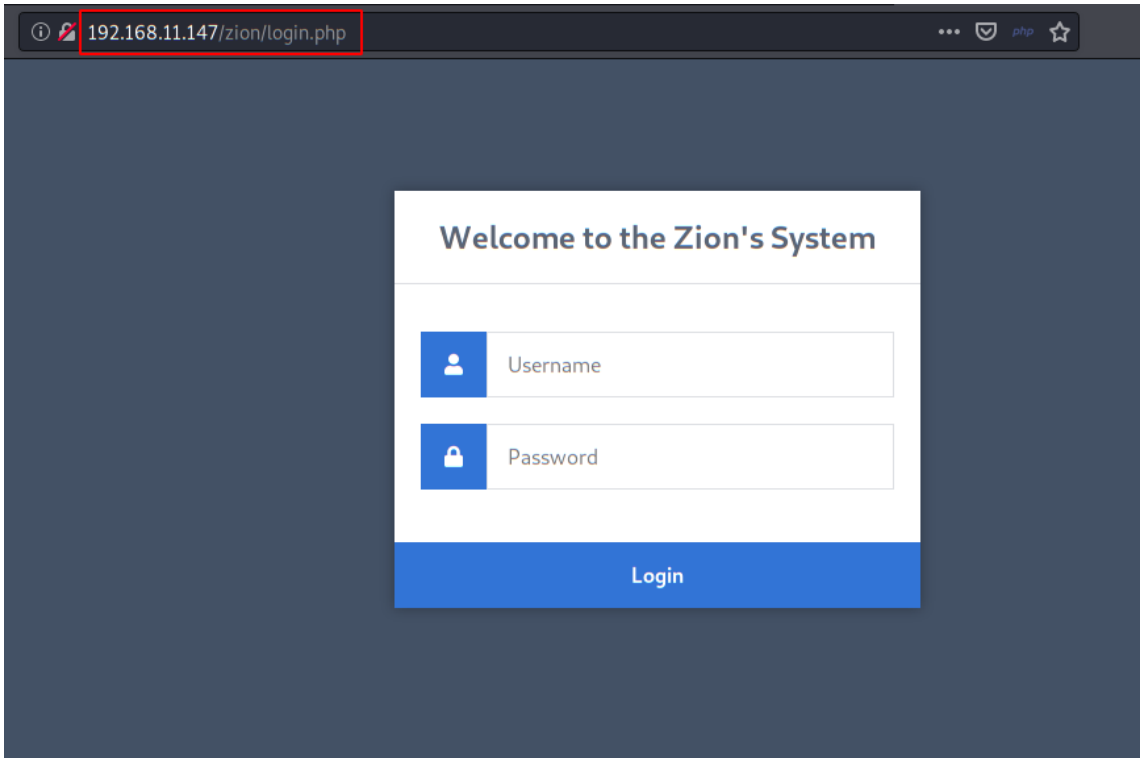
```
root@m3n0sd0n4ld:~/Documentos/OSCP/machines/Zyon# nmap -sV -sC -p- 192.168.11.147 -o 192.168.11.147
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-16 01:27 EDT
Nmap scan report for 192.168.11.147
Host is up (0.0012s latency).
Not shown: 65533 filtered ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.0 (protocol 2.0)
| ssh-hostkey:
|   2048 92:4b:37:54:79:d2:a8:e2:b1:90:f6:f0:95:73:75:14 (RSA)
80/tcp open  http    Apache httpd (PHP 7.4.5)
|_http-server-header: Apache
|_http-title: 403 Forbidden
MAC Address: 00:0C:29:6D:94:7F (VMware)
```

Como se aprecia en la imagen, solo existe el servicio SSH y un Apache en el puerto 80.

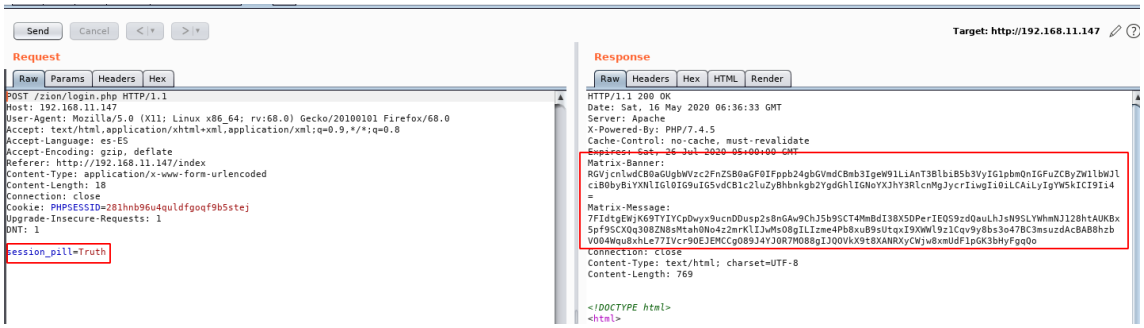
Accediendo al servicio web, tenemos la famosa elección de *Neo* en la película *Matrix*.



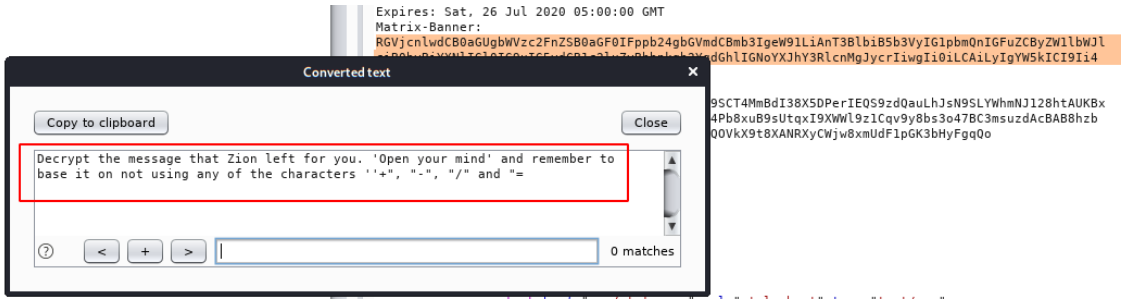

Elijo verdad (Truth), me redirige a un panel de autenticación.



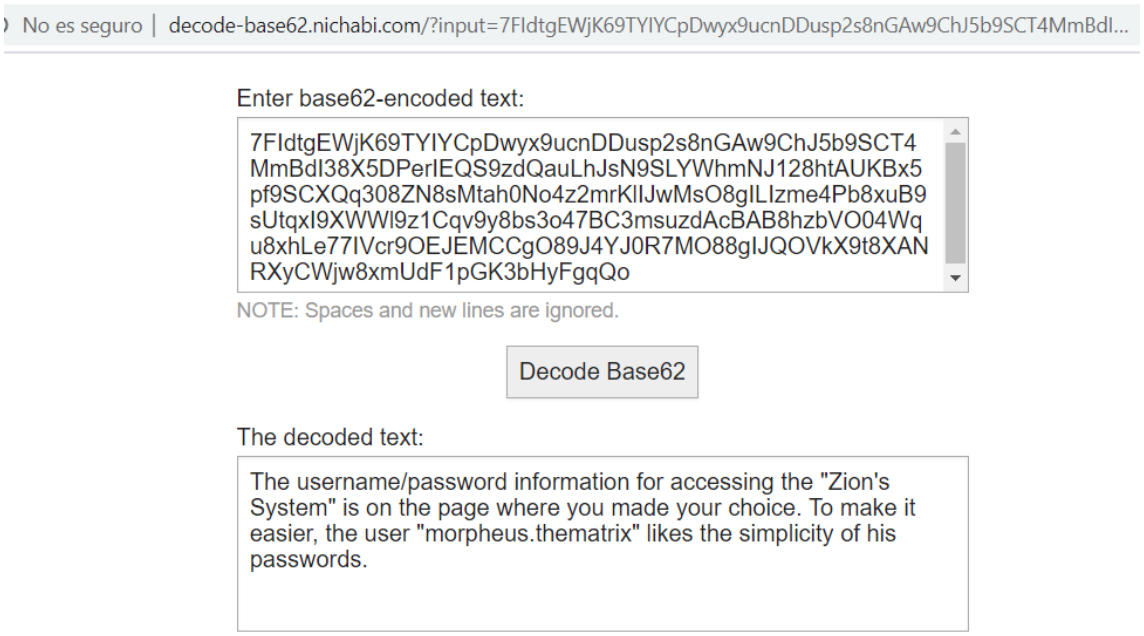

Dependiendo de la opción que se seleccione, un banner en la respuesta del servidor nos aparecerá con información para continuar el reto. Elijo *Truth* y compruebo las cabeceras de respuesta:



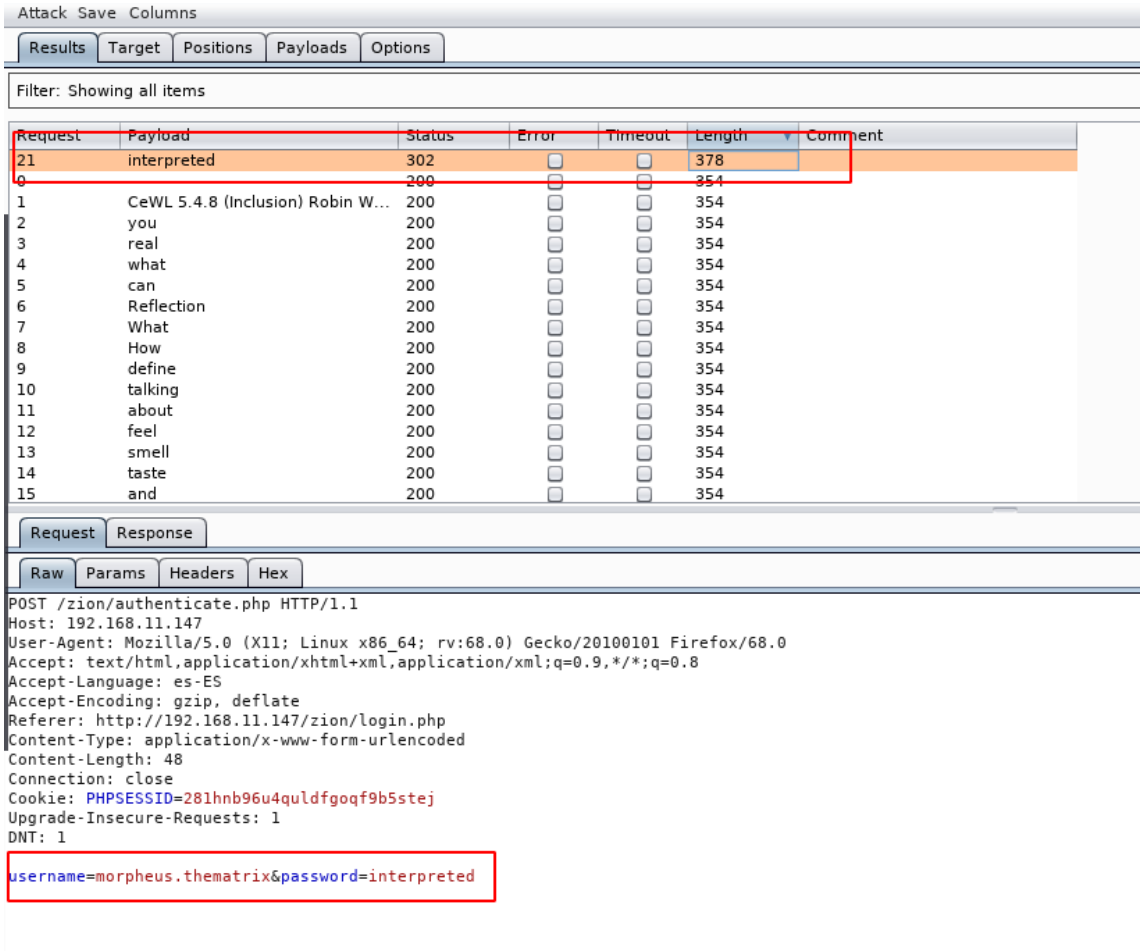

Decodifico la información de *Matrix-Banner* que está en base64, esta información es una pista para conseguir descifrar el *Matrix-Message*.



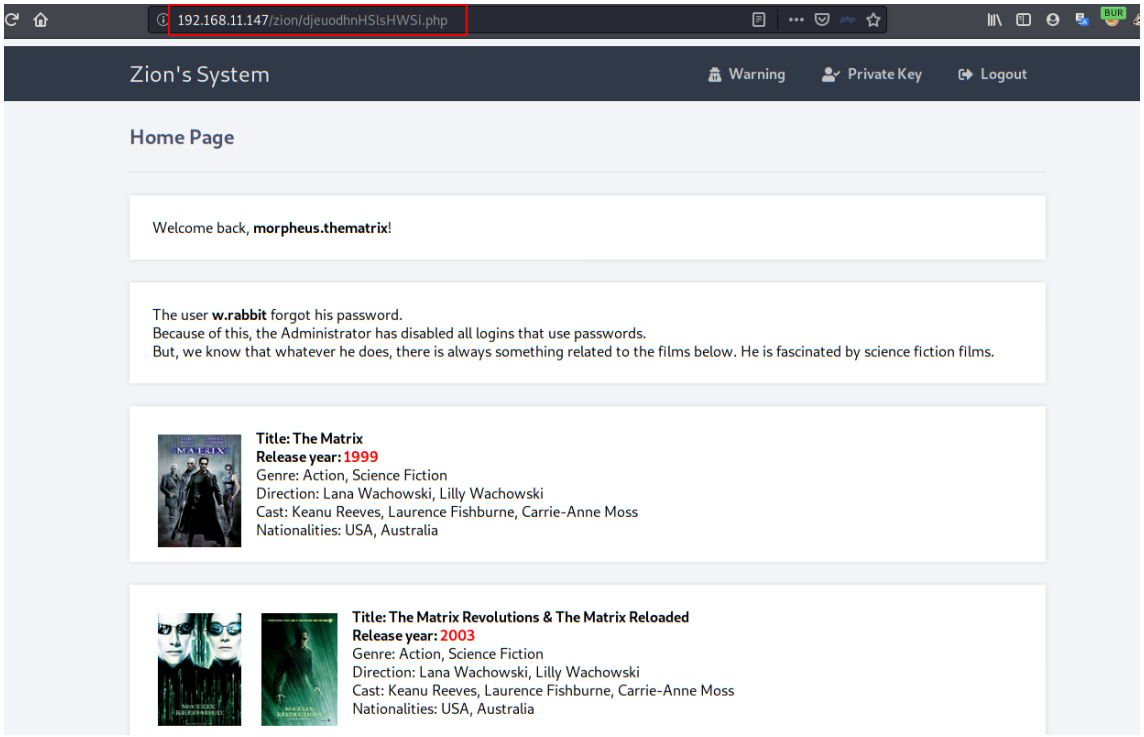

Indica que el cifrado/codificación del texto no utiliza 4 caracteres, por lo tanto, si el anterior está en base64 y le restamos 4… ¿Este debe de estar en base60? ¿Eso existe? Pues no, pero sí que existe base62 y base58, pruebo en base62:



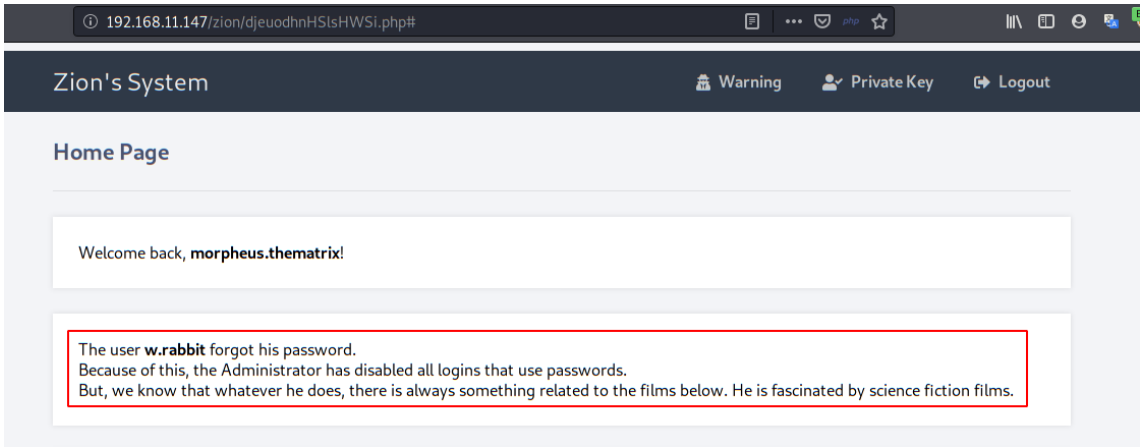

¡Genial! Ya tengo enumerado el usuario, ahora faltará encontrar la contraseña. Pruebo con contraseñas por defecto (guessing style), pero sin éxito. También pruebo con un diccionario con las 1000 contraseñas más utilizadas, pero tampoco ayuda. Finalmente, utilizo CeWL indicando el sitio web de la elección de la pastilla y ejecuto un ataque de fuerza bruta con Burp Suite.




¡Por fin! La contraseña es "*interpreted*", accedo al panel anterior:

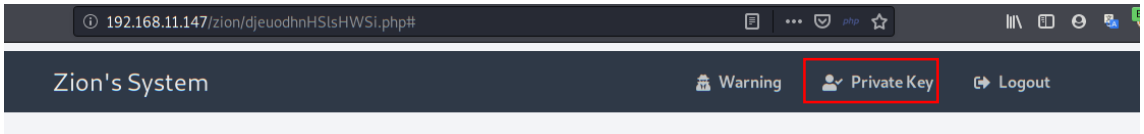



Dentro ya indica que la contraseña para el usuario *w.rabbit* ha sido deshabilitada:

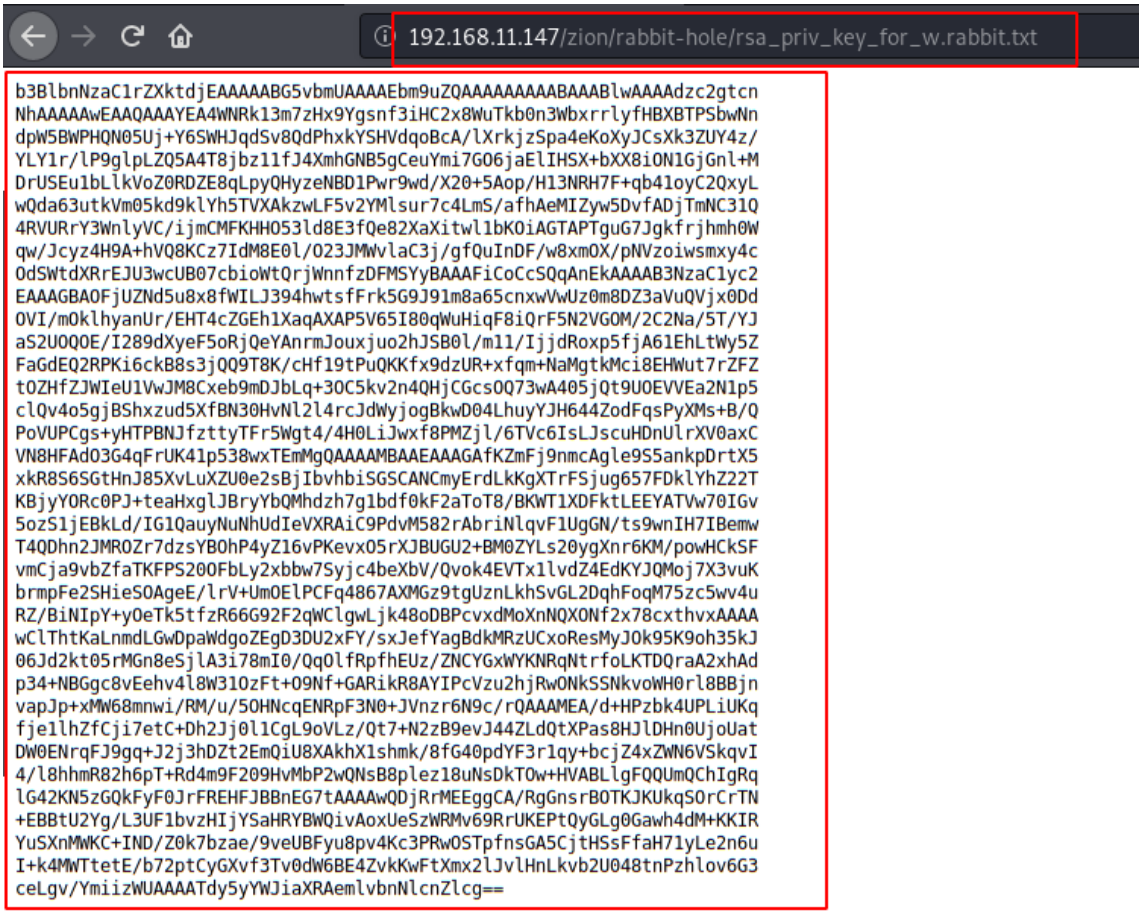



Haciendo clic en “Warning” nos da la pista de los “dos patitos”, por lo tanto 22, que es el puerto del SSH, ya sabemos que el usuario del SSH será ***w.rabbit***, como no tiene contraseña, la opción que queda es acceder por clave privada.

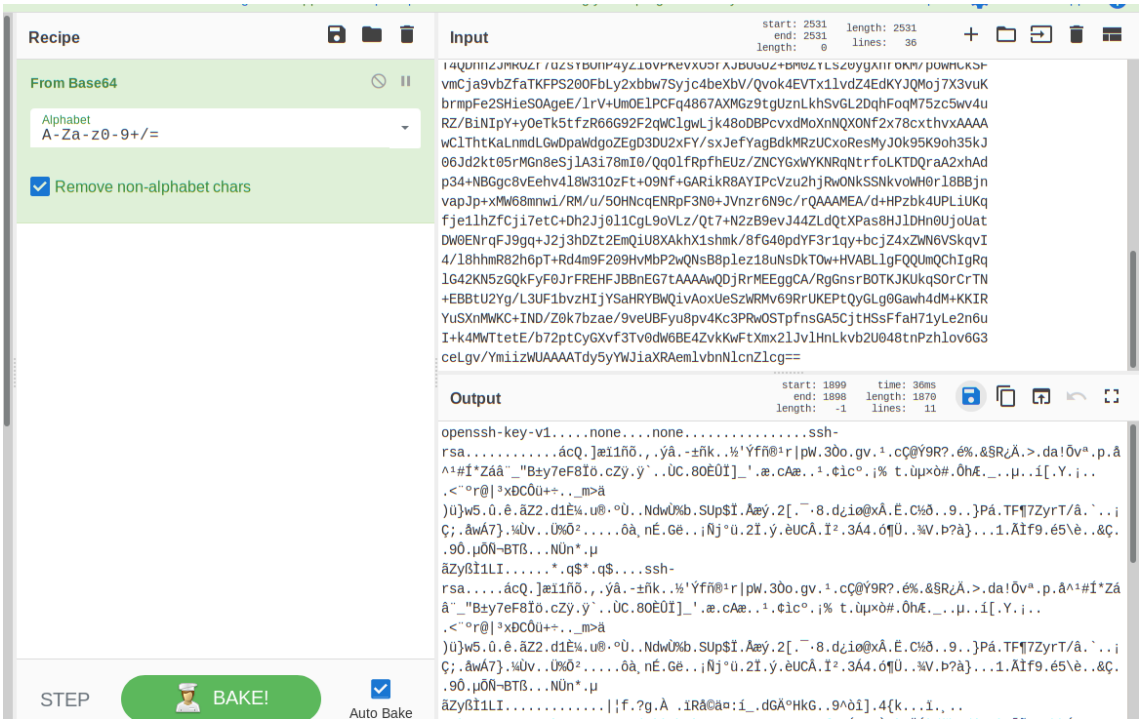
Arriba en el menú, tenemos la opción de “Private Key”:




Si hago clic en la opción del menú la tendremos en un txt (aunque no está completa)

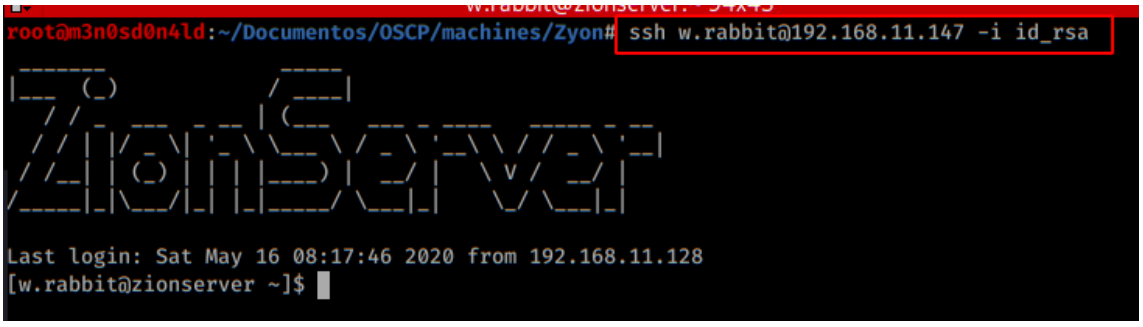



Aunque en el enlace ponga que es un Rabbit Hole, podemos decodificar parte y comprobar que realmente es una Private Key para OpenSSH.




La arreglamos….


Y la pruebo para conectarme por SSH, efectivamente funciona.




Compruebo los fichero que están disponibles:

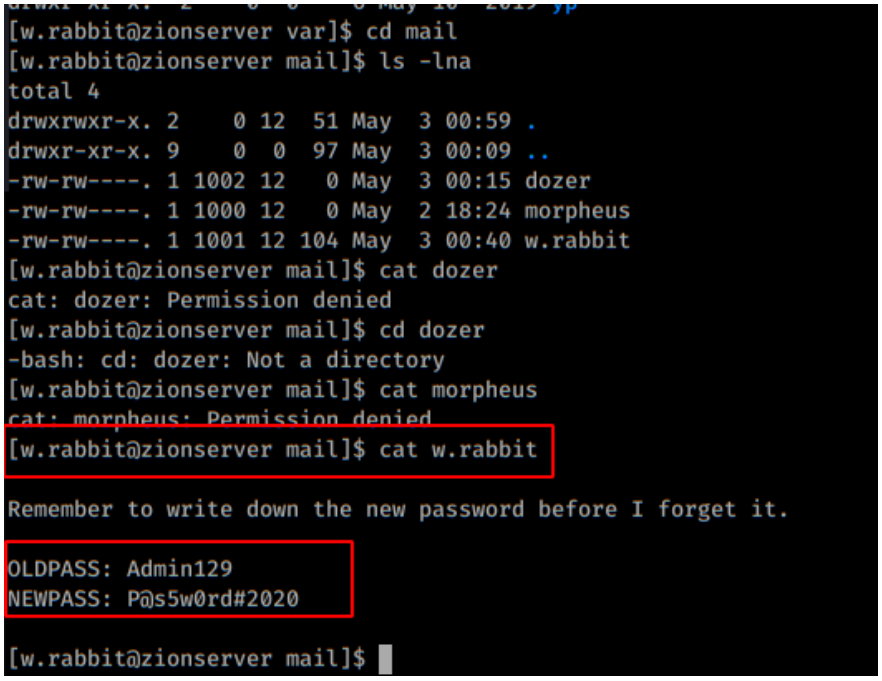
```
[w.rabbit@zionserver ~]$ ls -lna
total 28
drwx------. 8 1001 1001 237 May  3 02:42 .
drwxr-xr-x. 5    0    0  51 May  3 00:15 ..
-rw-------. 1 1001 1001   0 May  3 01:25 .bash_history
-rw-r--r--. 1 1001 1001  18 Nov  8  2019 .bash_logout
-rw-r--r--. 1    0    0 141 Nov  8  2019 .bash_profile
-rw-r--r--. 1    0    0 319 May  3 02:36 .bashrc
-rw-r--r--. 1    0    0 310 May  3 02:35 .bashrc~
-rw-r--r--. 1    0    0 310 May  3 02:35 .bashrz~
drwx------. 3 1001 1001  19 May  3 02:08 .config
-rw-------. 1 1001 1001  16 May  3 02:08 .esd_auth
drwxrwxr-x. 3 1001 1001  17 May  3 00:32 .local
drwx------. 2 1001 1001  43 May  3 02:04 .ssh
drwxrwxr-x. 2 1001 1001   6 May  3 00:32 backup
drwxrwxr-x. 2 1001 1001   6 May  3 00:32 personal
drwxrwxr-x. 2 1001 1001   6 May  3 00:32 scripts
-rw-rw-r--. 1 1001 1001 135 May  3 00:35 warning.txt
[w.rabbit@zionserver ~]$
```

Leo el fichero “warning.txt” y nos indica la ruta donde está el fichero de la flag.

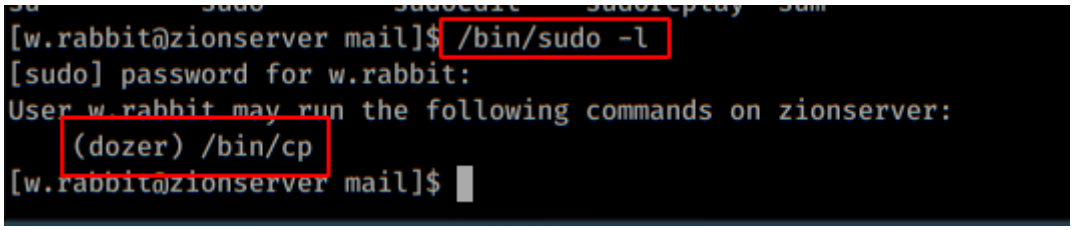
```
[w.rabbit@zionserver ~]$ cat warning.txt

Congratulations on making it this far.
The goal is to read the /home/dozer/flag.txt file.

Use the method and techniques you prefer.

[w.rabbit@zionserver ~]$
```

Al hacer un “sudo –l” pide contraseña, lógicamente no la sé, por lo que tendré que buscarla para poder autenticarme y utilizar algún binario o SUID para escalar privilegios.

Tras repasar los ficheros del sistema, encuentro estas credenciales en la carpeta mail:




Ahora sí, con la contraseña en mi poder, ejecuto “sudo –l” y veo que tengo permisos para utilizar el binario “***cp***” con el usuario *dozer*.




Pues fácil, ejecuto con sudo al binario y copio el fichero flag.txt a la carpeta temporales y leerla con el comando “***cat***”.