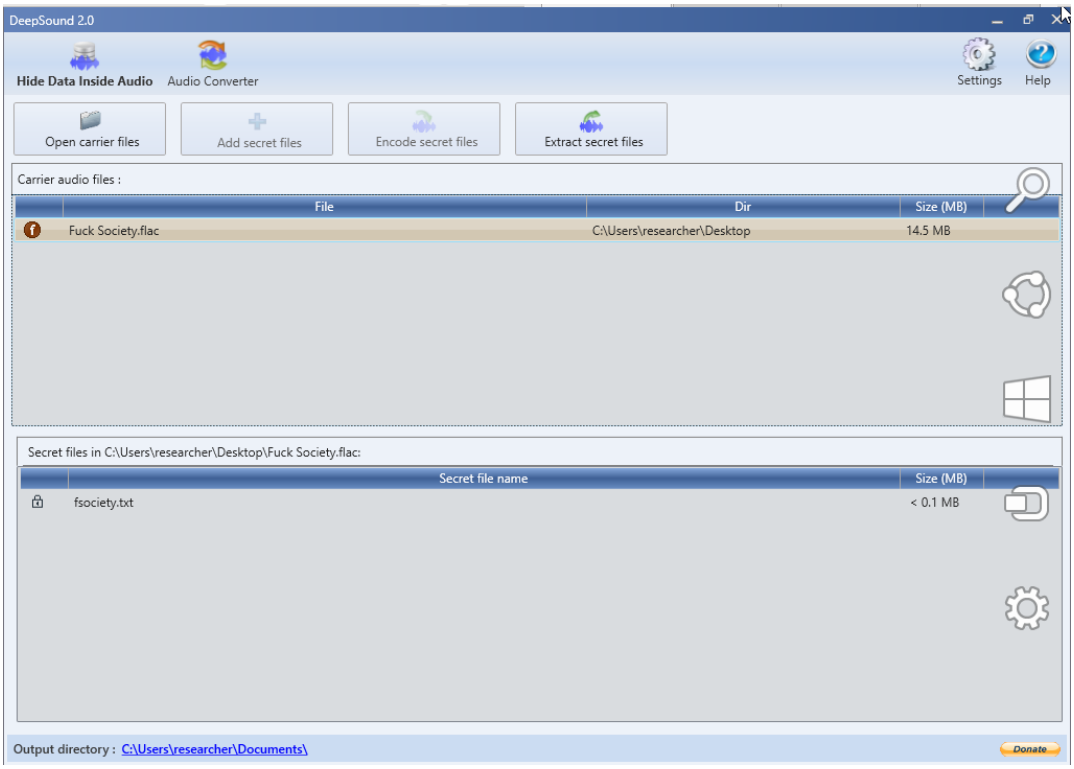
CTF FWHIBBIT WRITE-UPS BY @Roskyfrosky

ESTEGANOGRAFÍA

Ones and Zeroes
Points: 100
Country: Nigeria
Attachment: https://mega.nz/#!ApEzQC5C!43uNu_7_IVijvU-9ash0dkpTq9SS5gCsd1UTVTkkTKA
Description: The Dark Army told me that Stage 2 is ready. When you see it, you'll be pleased. It worked, Elliot...do not forget to hide all the information in those stupid CDs!!!
It's up to us now. Let me show you.

En este caso nos descargamos un fichero del enlace llamado “Fuck Society.flac”. Lo abrimos con DeepSound para ver si contiene algún fichero en su interior.
Nos pide una clave que es “ElliotAlderson” y nos muestra que contiene un fichero oculto en su interior con nombre fsociety.txt.



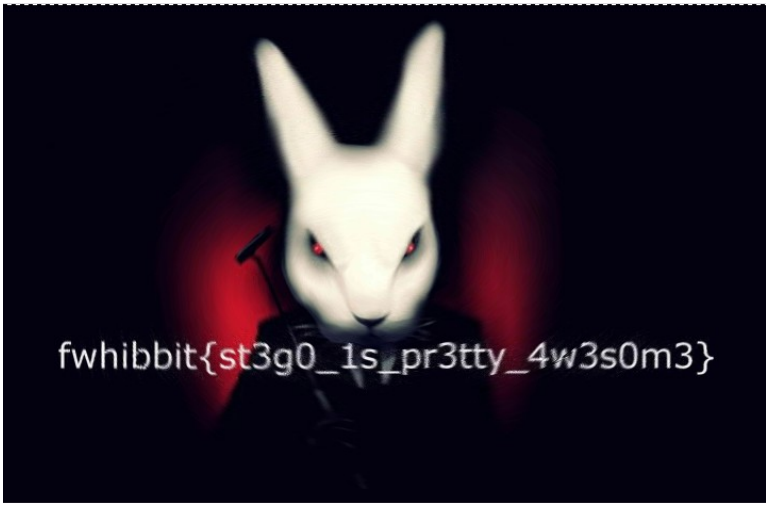
Lo abrimos y vemos la flag : fwhibbit{0ur_D3m0cr4cy_h4s_b33n_H4ck3d}

Flag Inside
Points: 125
Country: France
Attachment: https://mega.nz/#!th8EAYzZ!S8DM-gg2SAa8rrOnQ1_vHyZlY3x0r9tFzlLTevGUA8w
Description: I'm sure this file hides something ... but I do not know exactly what. Can you help me?

Nos descargamos un fichero con nombre “flag_inside.pdf”. El fichero contiene un base64 que es una imagen, por lo que utilizamos una tool online (http://codebeautify.org/base64-to-image-converter) que nos dá la siguiente imagen:

Si utilizamos gimp, vamos a filtros → Distorsión → Remolinos y aspiración.

Modificando el parámetro ángulo de remolino obtenemos la imagen con la flag:



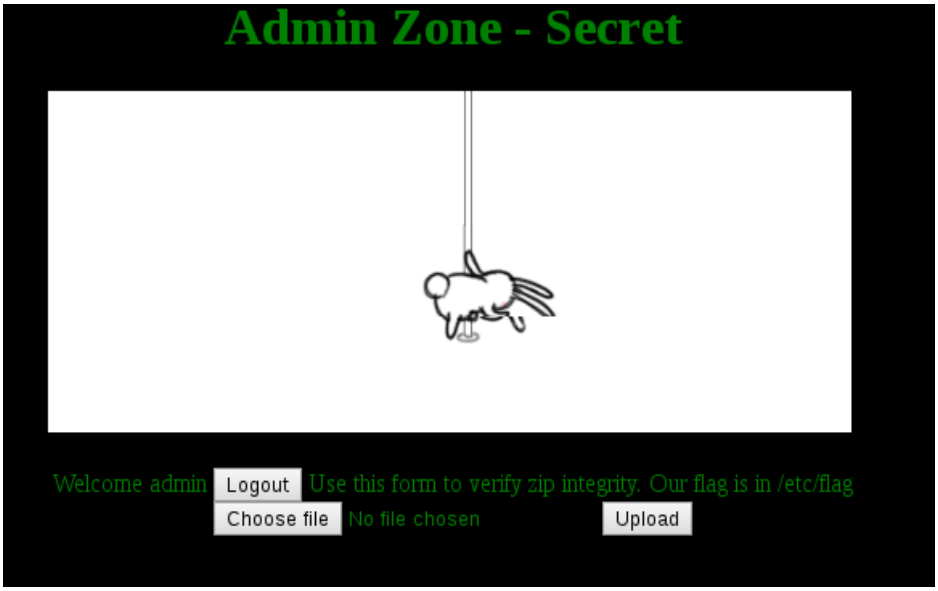
WEB

Hate
Points: 200
Country: Madagascar
Link: http://web5.ctf.followthewhiterabbit.es/
Description: There is a phisher who pretends to be us, can you help us to pwn him?

Entramos en la página y lo primero que hacemos es mirar el código fuente de la página donde encontramos que hay una zona privada.

Accedemos a http://web5.ctf.followthewhiterabbit.es/admin.php y volvemos a mirar el source de la página y vemos que nos indica cuales son las credenciales:

Metemos las credenciales que nos indican (admin: admin) y nos aparece un upload para subir ficheros en el cuál indican que debe de ser un zip:



Trás varias pruebas vemos que al subirel fichero zip, lo desempaqueta, comprueba si existe el fichero y en caso correcto muestra el resultado. Como en el enunciado nos dicen que la flag está en /etc/flag creamos un link simbólico y lo empaquetamos.

```
➜  ~ ln -s /etc/flag flag
➜  ~ zip --symlinks flag.zip flag
  adding: flag (stored 0%)
➜  ~
```

Subimos el Zip y nos muestra la flag:

FORENSE

Amnesia
Points: 200
Country: Chile
Attatchment: https://mega.nz/#!hh0RFKaI!nGKDQXItaLQiZnw_WygT3-ga7alXUKkisl2wC8uej6s
Description: Our favorite rabbit has lost its pendrive, inside, you can find sensitive information that should not be discovered by the queen. He doesn't remember the password of the file or where the information was hidden! Can you help him?

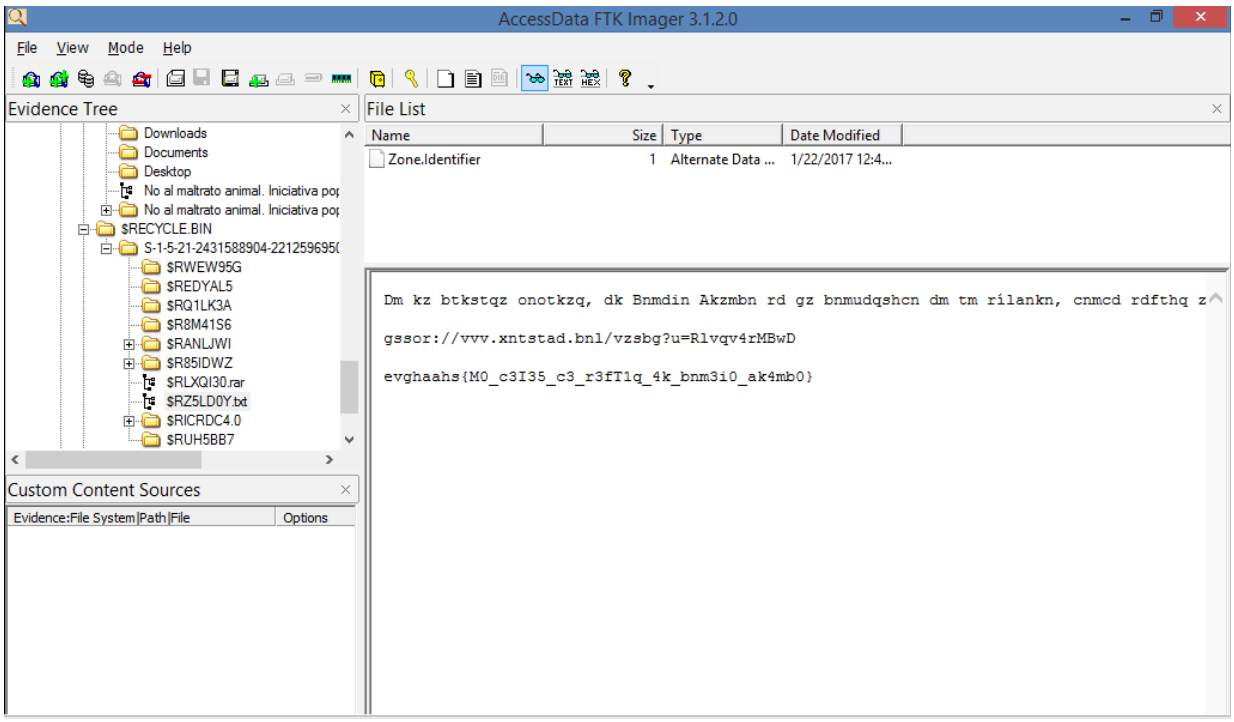
Desde el enlace nos bajamos un fichero con nombre “ch4ll3nger.rar” que está protegido con contraseña.Pasamos a intentar creakear la contraseña con la versión de John de ripper mejorada(john jumbo) de manera que utilizamos el siguiente comando para generar el hash de la password del fichero que más tarde pasaremos a crackear.:

./rar2john ch4ll3nger.rar > challenger.john

Una vez que tenemos el hash de la password del fichero rar, pasamos a crackearlo.

```
→  ~ ./john-1.8.0-jumbo-1/run/john --incremental challenger.john
Loaded 1 password hash (rar, RAR3 [SHA1 AES 32/64])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
password         (ch4ll3ng3r.rar)
1g 0:00:45:31 DONE (2017-02-19 14:54) 0.000366g/s 149.8p/s 149.8c/s 149.8C/s password..passw0la
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

La contraseña resulta ser “password”, por lo que descomprimimos y obtenemos un fichero llamado “ch4ll3ng3r.ad1”. La extensión del fichero indica que es una imagen que se ha tomado con FTK Imager, por lo que la abrimos con FTK Imager y en la papelera nos encontramos con un fichero de texto en rot13.



Lo ponemos en claro y vemos el siguiente mensaje donde está la flag:

```
En la cultura popular, el Conejo Blanco se ha convertido en un símbolo, donde seguir al
Conejo Blanco describe el acto de seguir a algo o a alguien ciegamente,
y cuya persecución desemboca en aventuras y descubrimientos. Esta metáfora ha sido utilizada numerosas
veces en la literatura, el teatro, el cine y la televisión. https://www.youtube.com/watch?v=Smwrw4sNCxE
fwhibbit{N0_d3J35_d3_s3gU1r_4l_con3j0_bl4nc0}
```

CRIPTOGRAFÍA

The burrow needs you
Points: 175
Country: Australia
Description: The new cryptographic algorithm? Can you help us?
xoCnrQoFNCiIvQImryKTIpLRTsYLSFTEfmWrvdYbJEsPxNWlxygTmHtufmysnGDtTCemYVgGlocDLg
ObvxeIRQRbvUwPuJoGPJYgjFCCfetUGEqYVcYpBJkpJHKYDUphbWWodHgerNcWxLWsFyleEoqyLFo
qlQKJtjMGhsPFXejIiqUGrGOyFjKLNOIUtyrwHeKXOMlRhdOclVQMjIsKfdUMBYgqiWVg==

Viendo el texto a simple vista parece un base64, pero cuando lo intentamos decodear no arroja ningun texto en claro, por lo que decio buscar una variante de este.
Me encuentro con un script que oculta una cadena de texto o archivo y simula ser un base 64 (https://github.com/hecky/stegb64).

```
→  ~ python stegb64.py -r xoCnrQoFNCiIvQImryKTIpLRTsYLSFTEfmWrvdYbJEsPxNWlxygTmHtufmysnGDtTCemYVgGlocDLgObvxeIRQRbvUwPuJoGPJYgjFCCfetUGEqYVcYpBJkpJHKYDUphbWWodHgerNcWxLWsFyleEoqyLFoqlQKJtjMGhsPFXejIiqUGrGOyFjKLNOIUtyrwHeKXOMlRhdOclVQMjIsKfdUMBYgqiWVg==

fwhibbit{f4k3_b4s3_64_rul3s}
```

Lo pruebo y obtenemos la flag:

fwhibbit{f4k3_b4s3_64_rul3s}

REVERSING

Reversing ‘like’ a boss

Points: 100
Country: Romania
Attatchment: https://mega.nz/#!vpgFgJYB!wYwVOMhSEbVoXpeRBm4qnpLGzmQBD5VPV7fU7gPvXJE
Description: We have this file, but we aren't able to extract the secret info inside, can you help us? The world's future is in your hands!

Se trata de un binario .Net , por lo que lo decompilamos y se ve la flag en claro.

```
};
Console.WriteLine("User info: ");
Console.WriteLine("   {0}\n", BitConverter.ToString(array));
string text = Convert.ToBase64String(array);
Console.WriteLine("Loaded payload:\n   {0}\n", text);
byte[] value = Convert.FromBase64String(text);
Console.WriteLine("Saving your info: ");
Console.WriteLine("   {0}\n", BitConverter.ToString(value));
string text2 = "Easy Reversing Challenge";
string str = "fwhibbit{";
Console.Title = text2;
Console.WriteLine(text2);
Console.WriteLine("Enter your name:");
string str2 = Console.ReadLine();
Console.WriteLine("Hi " + str2 + " how are you?");
Console.WriteLine("Enter your password:");
string a = Console.ReadLine();
string value2 = "password";
string b = "user";
Console.WriteLine(value2);
bool flag = a == b;
if (flag)
{
    Console.WriteLine("Congrats! Your flag is:" + str + "OBFUSCATE_NET!}}");
}
else
{
```

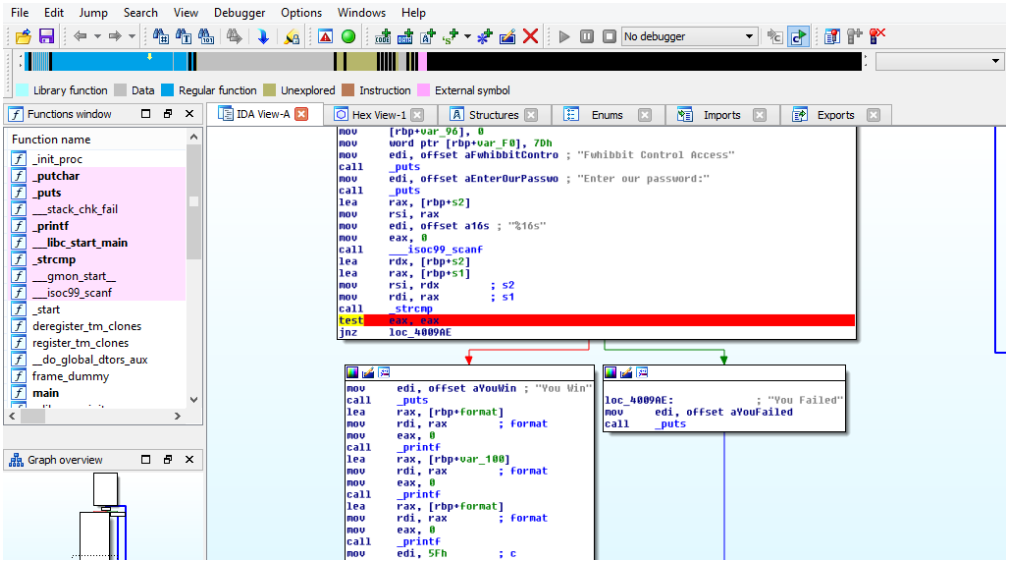
Flag: flag fwhibbit{OBFUSCATE_NET!}

Mayday Mayday

Points: 150
Country: South Africa
Attatchment: https://mega.nz/#!HpYxUIIZ!TjDhMDCvazuay1Cats4zObHuRmixGhVa7Sy0-5hnLTg
Description: Hi aspirant, we lost all our carrots, for this reason we need your skills so please... try to steal the private bank of carrots for us.
The time begins...NOW!

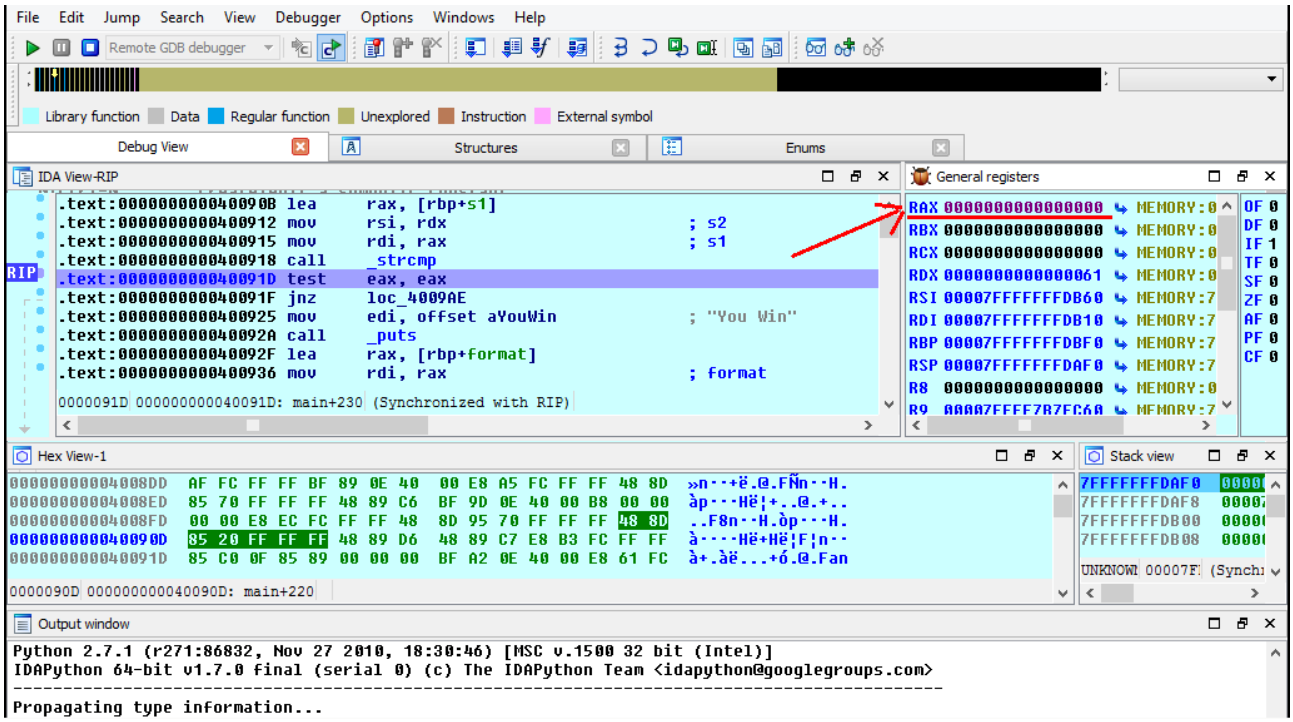
Nos dan un ELF de 32 bits, lo ejecutamos para ver que ocurre y nos aparece un mensaje de un dollar y nos pregunta una password.Como no la sabemos vamos a abrirlo con el IDA y vemos como funciona.

Comprobamos donde se hace el You win and You lose y ponemos un breakpoint en la comprobación.



Empezamos a debuggear con gdbserver, para que se alcance el breakpoint hay que meter una password cualquiera, yo por ejemplo he metido el carácter “a”.

Una vez alcanzado el breakpoint lo único que hay que hacer es que se cumpla la condición para que salte al código que queremos y es que eax(en este caso rax porque es de 64 bits) sea 0.
Al haber metido el caracter “a” de contraseña, en rax nos aparece 0x00....D, por lo que le seteamos a 0:



y volvemos a pulsar el boton de run para que nos escupa la flag:

MISCELLANEOUS

Information Leakage
Points: 75
Country: Tanzania
Link: flag.followthewhiterabbit.es
Description: Our experts claim that we have suffered an important information leak thanks to our domain: flag.followthewhiterabbit.es
Can you check if this is true?

Nos dan una url, por lo intentamos acceder a ella por el navegador y nos devuelve que no es alcanzable. Por lo que lanzamos el comando host -a para ver toda la información del subdominio.

```
→  ~ host -a flag.followthewhiterabbit.es
Trying "flag.followthewhiterabbit.es"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19151
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 2, ADDITIONAL: 0

;; QUESTION SECTION:
;flag.followthewhiterabbit.es.  IN      ANY

;; ANSWER SECTION:
flag.followthewhiterabbit.es. 3600 IN   RRSIG   TXT 7 3 3600 20170317170954 2017
0215170954 43673 followthewhiterabbit.es. U05Ns0BE0xhhOVeRYyKWyjT/kXJRWpRlSH6NFZ
yTrF2w0iWZM9jGbz/p pW6KPDggEhtYpO0YncAnWHpqBywDO+/bi+g1A3Kwc92u/+x5dIwE9H/s 0iGB
agMO/l6Z6mngfL/yyG3CmoJZKKjiauKnrREeL/LY8yW0ParcWrUJ C5U=
flag.followthewhiterabbit.es. 3600 IN   TXT     "fwhibbit{DnS_L3ak}"
flag.followthewhiterabbit.es. 3600 IN   RRSIG   A 7 3 3600 20170317170954 201702
15170954 43673 followthewhiterabbit.es. WtRy2JYB0BLU8C5VlsHRZs5UGWEfV0827ymstdjd
iK+/lkvuLKYQSCkl 0lbopys+DT5tzTm2zHBJXLQbWCikjk37dCwnThCsRmP3X6e/ZfedLO6+ 7q/uU9
O61UAJFjamCQOEOMJSLbHPig9swYP7X/pZpjbhyPnvAEuGOA+P yUk=
flag.followthewhiterabbit.es. 3600 IN   A       127.0.0.1

;; AUTHORITY SECTION:
followthewhiterabbit.es. 60032  IN      NS      ns20.ovh.net.
followthewhiterabbit.es. 60032  IN      NS      dns20.ovh.net.

Received 505 bytes from 10.28.5.1#53 in 61 ms
```

Vemos que nos muestra la flag.

fwhibbit{DnS_L3ak}

TORxicity
Points: 300
Country: Colombia
Link: http://rabbit3yfa6dcgka.onion
Description: We recently found that a group of people are selling female rabbit extract. You need to find their real server, and deanonimize them!

Este reto era igual que uno que salío en el CTF de la ekoparty por lo que ya sabíamos como se resolvía. Intentamos conectarnos por ssh al dominio facilitado para obtener el fingerprinting de la clave del host.

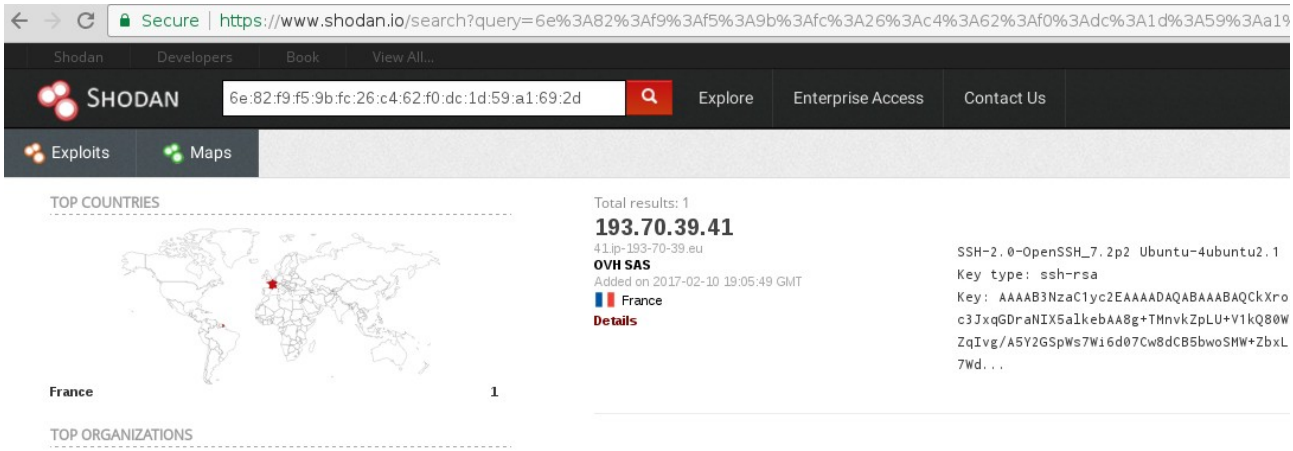
```
→  ~ torify ssh rabbit3yfa6dcgka.onion
The authenticity of host 'rabbit3yfa6dcgka.onion (127.42.42.0)' can't be establi
shed.
ECDSA key fingerprint is 88:5b:86:ce:e6:40:96:3a:c7:42:11:bf:0e:86:05:78.
Are you sure you want to continue connecting (yes/no)? no
Host key verification failed.
```

Y probamos a buscarlo en shodan, por si acaso hubiera suerte.

No lo hemos encontrado, por lo que utilizamos el comando ssh-keygen para generar la clave en diferentes algoritmos.

```
→  /tmp ssh-keygen -l -f keys
2048 6e:82:f9:f5:9b:fc:26:c4:62:f0:dc:1d:59:a1:69:2d rabbit3yfa6dcgka.onion (RSA)
256 88:5b:86:ce:e6:40:96:3a:c7:42:11:bf:0e:86:05:78 rabbit3yfa6dcgka.onion (ECDSA)
256 b2:61:6b:64:1d:ad:99:27:52:fd:59:fb:1f:42:96:27 rabbit3yfa6dcgka.onion (ED25519)
→  /tmp
```

Y pasamos a comprobar si alguno de esos algoritmos se encuentra en shodan como hemos hecho anteriormente.



Esta vez ha habido suerte y obtenemos una IP. Entramos a la IP con el navegador y obtenemos la flag.