

Mission 5

<http://unaaldia.hispasec.com/2018/03/una-al-mes-mision005.html>

Misión#005	
Información personal: Nombre: Thomas A. Anderson Fecha de nacimiento: 11 de Marzo del 1962 Trabajo: Programador Empresa: Metacortex	
Misión: Nivel: Medio Introducción: ¡Neo, tenemos un problema! Han secuestrado a Morfeo y no sabemos donde lo pueden tener. Necesitamos que investigues y descubras su localización para rescatarlo. La única pista que tenemos es una URL que conseguimos. ¿Serás capaz de encontrarle?	
Información adicional: URL conseguida: <code>http://34.253.233.243/search/localizacion.php</code> Tip: La flag es el nombre del sitio donde se encuentra con el formato UAM{Localización}. Tip2: El nombre del sitio en la flag es con "_" en lugar de espacios. Tip3: El archivo ".zip" se descomprime con "123mango". Tip4: Hay una flag trampa la cuál no tiene localización.	

Parece que han atrapado a Morfeo y debemos averiguar dónde.

Cuando entramos en <http://34.253.233.243/search/localizacion.php>

vemos que hace una redirección a <http://34.253.233.243/search/index.php>

No todo es lo que parece...



Si miramos el código de la página antes de que nos redirija, encontramos el texto al final del html:

Para continuar deberéis sacar X información del primer archivo (la cuál está encriptada) y pasársela al segundo archivo:
archivo 1: <https://goo.gl/K1dcbG>
archivo 2: <https://drive.google.com/open?id=1CAz5xxsf9YxGISWDgOVURsvFmT6A1Swn>

El archivo 1

<https://goo.gl/K1dcbG>



Nos descargamos la imagen y con un simple *head* vemos que entre los metadatos hay cosas interesantes:

```
$ head morfeo.jpg
****
: http://ns.adobe.com/xap/1.0/?xpacket begin='' id='WSM0MpCehiHzreSzNTczkc9d'?>
<x:xmpmeta xmlns:x='adobe:ns:meta/' x:xmptk='Image::ExifTool 10.75'>
<rdf:RDF xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'>

  <rdf:Description rdf:about=''
    xmlns:dc='http://purl.org/dc/elements/1.1/'>
    <dc:creator>
      <rdf:Seq>
        <rdf:li>Pass:UAM</rdf:li>
      </rdf:Seq>
```

Con Steghide podemos averiguar lo que esconde la imagen, y como es lógico, probamos con la pass UAM

```
$ steghide extract -sf morfeo.jpg -p UAM
anotó los datos extraídos e/"morf.txt".
```

El jpg escondía este texto:

```
$ cat morf.txt
AABBBBAABBBAAABBBBBAABA AABBAABBBAAABBB AABBAABABB AABABABABABAAAABAAABAAABA
```

Está codificado en Baconian y uso cualquier utilidad online:

Baconian Cipher

Rumkin.com >> Web-Based Tools >> Ciphers and Codes Search:

Francis Bacon created this method of hiding one message within another. It is not a true cipher, but just a way to conceal your secret text within plain sight. The way it originally worked is that the writer would use two different typefaces. One would be the "A" typeface and the other would be "B". Your message would be written with the two fonts intermingled, thus hiding your message within a perfectly normal text.

There are two versions. The first uses the same code for I and J, plus the same code for U and V. The second uses distinct codes for every letter.

For example, let's take the message "Test It" and encode it with the distinct codes for each letter. You get a result like "BAABBAABAABAABABAABB ABAAAABAABB". The original message is 6 characters long so the encoded version is $6 * 5 = 30$ characters. If I were to find a 30-character message and put in "B" letters as bold and italics, we will get "***This is a test message*** ***with bold for "B"***".

When decoding, it will use "0", "A", and "a" as an "A"; "1", "B", and "b" are all equivalent as well. Other letters are ignored.

Decrypt ▾

Distinct codes ▾

Your message: ([Swap A and B](#))

AABBBBAABBBAAABBBBBAABA AABBAABBBAAABBB AABBAABABB AABABABABABAAAABAAABAAABA%

This is your encoded or decoded text:

HTTPS GOO GL FKQRC

El archivo 2

<https://drive.google.com/open?id=1CAz5xxsf9YxGISWDgOVURsvFmT6A1Swn>

```
#!/usr/bin/python3

string = input("Introduce la información que hayas sacado de la imagen: ")

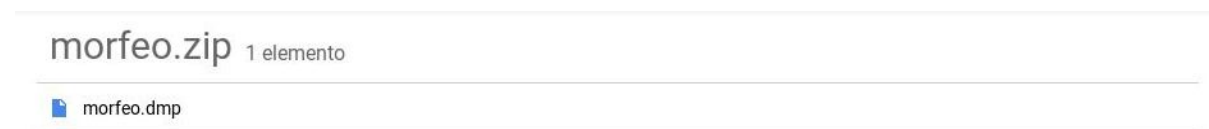
a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r = string

a = a.lower()
b = b.lower()
c = c.lower()
d = d.lower()
e = e.lower()
f = '://'
g = g.lower()
h = h.lower()
i = i.lower()
j = '.'
k = k.lower()
l = l.lower()
m = '/'
n = n.upper()
o = o.lower()
p = p.upper()
q = q.upper()
r = r.lower()
s = '2'

print (a + b + c + d + e + f + g + h + i + j + k + l + m + n + o + p + q + r + s)
```

Un script en python que nos ayuda a construir la url de verdad, transformando a mayúsculas y minúsculas como corresponde. Además añade un 2 de postre :)

Esta es la URL finalmente <https://goo.gl/FkQRc2>



El archivo zip tiene password pero no lo conocemos, y tras un rato probando diccionarios sin éxito, con john the ripper trato de hacer brute force... llevaba 30 minutos funcionando cuando nos informaron los admins del CTF por el canal de Telegram que esperaban que funcionara con algunos diccionarios típicos, pero al probar que no sacaba el password optaron por darnos el password a todos: 123mango

Lo descomprimos usando ese password y extraemos el archivo morfeo.dmp

Parece un dump de memoria así que vamos a usar volatility para explorarlo. Después de probar algunas cosas miro sus archivos, buscando los que contengan 'uam'...

volatility -f morfeo.dmp --profile=Win7SP1x64 filescan

```
0x000000007df0b120 1 1 RW-rwd \Device\HarddiskVolume2\Users\anubis\AppData\Local\Microsoft\Win
0x000000007df07070 7 0 R--r-d \Device\HarddiskVolume2\Windows\SysWOW64\dbghelp.dll
0x000000007df0a3e0 3 0 R--r-d \Device\HarddiskVolume2\Windows\System32\stobject.dll
0x000000007df0b200 4 0 R--r-d \Device\HarddiskVolume2\Windows\SysWOW64\taskschd.dll
0x000000007df0b350 16 0 R--r-- \Device\HarddiskVolume2\Users\anubis\Desktop\uam.jpg
0x000000007df0cf20 6 0 R--r-d \Device\HarddiskVolume2\Program Files (x86)\Google\Update\1.3.3
0x000000007df0d070 10 0 R--r-d \Device\HarddiskVolume2\Windows\System32\msshooks.dll
0x000000007df1af20 10 0 R--r-d \Device\HarddiskVolume2\Windows\System32\msidle.dll
0x000000007df26500 17 0 RW-rwd \Device\HarddiskVolume2$\Directory
0x000000007df27330 4 1 RW-rw- \Device\HarddiskVolume2\ProgramData\Microsoft\Search\Data\Appli
```

Extraigo el archivo.

volatility -f morfeo.dmp --profile=Win7SP1x64 dumpfiles -Q 0x000000007df0b350 --name -D files/

```
$ cat file.None.0xfffffa8003c58af0_uam.jpg.dat
<html>
  <head>
    <title>UAM FLAG</title>
  </head>
  <body>
    <h1>UAM{N30_i5_4_G0D}</h1>
  </body>
</html>
```

Parece que encontramos un flag, no? La maldita trampa... Envié la flag por email pensando que lo había conseguido pero justo leo en telegram el recordatorio... nos piden un lugar, no esto :(

Pero hay más archivos:

```
0x000000007dd73d00 1 1 RW--- \Device\HarddiskVolume2\Windows\SYSTEM32\LOCALS-1\AppData\Roaming\PEERNE-1\A8A453-1.HOM\86C374-1\grouping\db.mdb
0x000000007dd73e50 19 1 RW--- \Device\HarddiskVolume2\Windows\SYSTEM32\LOCALS-1\AppData\Roaming\PEERNE-1\A8A453-1.HOM\86C374-1\grouping\tmp.edb
0x000000007dd73760 11 0 R--r-d \Device\HarddiskVolume2\Windows\System32\drtrtransport.dll
0x000000007dd76760 1 1 ----- \Device\Afd\Endpoint
0x000000007dd78070 16 0 R--r-d \Device\HarddiskVolume2\Windows\System32\drt.dll
0x000000007dd78f20 101 1 ----- \Device\Afd\Endpoint
0x000000007ddae270 1 1 R--r-d \Device\HarddiskVolume2\Windows\System32\es-ES\KernelBase.dll.mui
0x000000007ddafe50 1 1 R--r-d \Device\HarddiskVolume2\Windows\System32\es-ES\msctf.dll.mui
0x000000007ddb0540 16 0 -W-r-- \Device\HarddiskVolume2\Users\anubis\Desktop\uam.jpg.jpgVirtualBox Dropped Files\2018-03-12T20_33_51_765201500Z\uan (2).jpg
0x000000007ddb08f0 1 1 R--rw- \Device\HarddiskVolume2\Windows\winsxs\amd64_microsoft.windows.common-controls_6595b64144ccf1df_6.0.7601.17514_none_fa3960871755a
0x000000007ddb1280 2 0 R--rwd \Device\HarddiskVolume2\Users\anubis\Searches\desktop.ini
0x000000007ddb1490 18 1 RW-r-- \Device\HarddiskVolume2\Users\anubis\System\EXT-MS-WIN-KERNEL32-PACKAGE-CURRENT-L1-1-0.DLL
0x000000007ddb1e20 2 1 R--rwd \Device\HarddiskVolume2\Users\Public\Desktop
0x000000007ddb2530 4 0 RW-rwd \Device\HarddiskVolume2$\Directory
0x000000007ddb2680 2 0 R--rwd \Device\HarddiskVolume2\Users\anubis\Downloads\desktop.ini
0x000000007ddb27d0 2 0 R--rwd \Device\HarddiskVolume2\Users\anubis\Contacts\desktop.ini
0x000000007ddb2dd0 4 0 RW-rwd \Device\HarddiskVolume2$\Directory
0x000000007ddb2f20 2 0 R--rwd \Device\HarddiskVolume2\Users\anubis\Favorites\desktop.ini
0x000000007ddb3970 2 0 R--rwd \Device\HarddiskVolume2\Users\anubis\Links\desktop.ini
0x000000007ddb4000 2 0 R--rwd \Device\HarddiskVolume2\Users\anubis\Saved Games\desktop.ini
```

\$ volatility -f morfeo.dmp --profile=Win7SP1x64 dumpfiles -Q 0x000000007ddb0540 --name -D files

Y este sí tiene algo interesante

```
$ cat file.None.0xfffffa8003bb1f10.uam\ \ (2\).jpg.dat
<html>
  <head>
    <title>Coordenadas de Morfeo</title>
  </head>
  <body>
    <h1>40.7484405, -73.9856644</h1>
  </body>
</html>
```

Vamos a maps y vemos que....



Morfeo está en el Empire State Building!

Y como nos advertían en el tip...

Tip2: El nombre del sitio en la flag es con "_" en lugar de espacios.

Tienen a Morfeo en el **UAM{Empire_State_Building}**!!!

Saludos!

Herramientas usadas:

Firefox
7-zip
steghide
volatility

José Ángel Sánchez (j0n3)
@_j0n3

