

9

Level09

Al abrir el proyecto tenemos de nuevo un script y un archivo llamado token. Este script no se le puede hacer debug y cuando lo ejecutamos con el token nos da una cadena con caracteres no imprimibles.

Para ello voy a optar por usar [ghidra](#) un software open source de reversing de binarios. Con Ghidra vemos lo siguiente. Nuestro binario procesa el argumento que recibe tras hacer varias comprobaciones anti debug.

Nuestro script lo que hace es ofuscar una cadena sumándole un valor al hexadecimal de cada carácter. Eso lo hace aquí:

```

08048987 0f be c0      MOVSX    EAX,AL
0804898a 03 44 24 20   ADD      EAX,dword ptr [ESP + local_120]
0804898e 89 04 24      MOV      dword ptr [ESP]=>local_140,EAX
08048991 e8 2a fb      CALL    <EXTERNAL>::putchar

```

Lo único que tenemos que hacer es modificar esa operación para que en lugar de sumar reste.

```

08048987 0f be c0      MOVSX    EAX,AL
0804898a 2b 44 24 20   SUB     EAX,dword ptr [ESP + 0x20]
0804898e 89 04 24      MOV      dword ptr [ESP]=>local_140,EAX
08048991 e8 2a fb      CALL    <EXTERNAL>::putchar

```

Ahora ejecutamos el script pasándole el token como argumento y sacamos la cadena fácilmente.

```

level09@SnowCrash:/var/tmp$ ls
level09_FILE level09_resta myToken.txt token
level09@SnowCrash:/var/tmp$ ./level09_resta $(cat myToken.txt)
f3iji1ju5yuevaus41q1afiuq
level09@SnowCrash:/var/tmp$ 

```

Pass para flag09: [f3iji1ju5yuevaus41q1afiuq](#)

Flag final [s5cAJpM8ev6XHw998pRWG728z](#)