

LS-20060330

## Computer Associates BrightStor ARCserve Backup Remote Buffer Overflow Vulnerability

**Release Date:**

10/05/2006

**Date Reported:**

03/30/2006

**Severity:**

Critical (Remote Code Execution)

**Vendor:**

Computer Associates

**Product:**

BrightStor® ARCserve® Backup provides a complete, flexible and integrated backup and recovery solution for Windows, NetWare, Linux and UNIX environments.

<http://www3.ca.com/solutions/ProductFamily.aspx?ID=115>

**Systems Affected:**

- BrightStor ARCserve Backup R11.5 Server
- BrightStor Enterprise Backup 10.5
- BrightStor ARCserve Backup v9.01
- CA Server Protection Suite r2
- CA Business Protection Suite r2

**Overview:**

LSsec has discovered a vulnerability in Computer Associates BrightStor ARCserve Backup, which could be exploited by an anonymous attacker in order to execute arbitrary code with SYSTEM privileges on an affected system. The flaw specifically exists within the Message Engine (msgeng.exe) due to incorrect handling of RPC requests on TCP port 6503. The interface is identified by dc246bf0-7a7a-11ce-9f88-00805fe43838. **Opnum 45** specifies the vulnerable operation within this interface.

**Vulnerability Details:**

ASCORE.dll version 11.5.3884.0 contains a buffer overflow vulnerability due to incongruous use of strcpy() in QSIGetQueuePath(). The source address of strcpy() is specified by the first DWORD of stub data and must be calculated as follows to get it point to user-supplied data which is stored on the heap.

$$heapAddr = (DWORD - 0x01) * 0x194 + 0xc200b8$$

**Copyright © 2006 LS Security**

Permission is granted for the redistribution of this alert electronically. It may not be edited in any way without the express written consent of LSsec. If you wish to reprint the whole or any part of this alert in any other medium other than electronically, please email request@lssec.com for permission.

**Disclaimer**

The information within this paper may change without notice. Use of this information constitutes acceptance for use in an AS IS condition. There are no warranties, implied or express, with regard to this information. In no event shall the author be liable for any direct or indirect damages whatsoever arising out of or in connection with the use or spread of this information. Any use of this information is at the user's own risk.

```

.text:2122EAFD ; int __cdecl QSIGetQueuePath(int,char *,int,int)
.text:2122EAFD public QSIGetQueuePath
.text:2122EAFD QSIGetQueuePath proc near ; CODE XREF: QSIGetQueueUsersFile+1D↓p
.text:2122EAFD ; QSIGetQueueJobsFile+1D↓p ...
.text:2122EAFD var_C = dword ptr -0Ch
.text:2122EAFD var_8 = dword ptr -8
.text:2122EAFD var_4 = dword ptr -4
.text:2122EAFD arg_0 = dword ptr 8
.text:2122EAFD arg_4 = dword ptr 0Ch
.text:2122EAFD arg_8 = dword ptr 10h
.text:2122EAFD arg_C = dword ptr 14h
.text:2122EAFD push ebp
.text:2122EAFE mov ebp, esp
.text:2122EB00 sub esp, 0Ch
.text:2122EB03 mov eax, [ebp+arg_0] ; src address
.text:2122EB06 sub eax, 1
.text:2122EB09 mov [ebp+var_4], eax
.text:2122EB0C mov ecx, [ebp+arg_4] ; dst address
.text:2122EB0F mov byte ptr [ecx], 0
.text:2122EB12 cmp [ebp+arg_C], 0
.text:2122EB16 jz loc_2122EBAE
.text:2122EB1C mov edx, dword_212603BC
.text:2122EB22 mov eax, dword_212603BC
.text:2122EB27 add eax, [edx+0Ch]
.text:2122EB2A mov ecx, [ebp+var_4]
.text:2122EB2D imul ecx, 194h
.text:2122EB33 add eax, ecx
.text:2122EB35 mov [ebp+var_8], eax
.text:2122EB38 cmp [ebp+arg_8], 0
.text:2122EB3C jz short loc_2122EB46
.text:2122EB3E mov edx, [ebp+arg_8]
.text:2122EB41 mov [ebp+var_C], edx
.text:2122EB44 jmp short loc_2122EB4F
.text:2122EB46 ; -----
.text:2122EB46 loc_2122EB46: ; CODE XREF: QSIGetQueuePath+3F↑j
.text:2122EB46 mov eax, [ebp+var_8]
.text:2122EB49 add eax, 50h
.text:2122EB4C mov [ebp+var_C], eax
.text:2122EB4F loc_2122EB4F: ; CODE XREF: QSIGetQueuePath+47↑j
.text:2122EB4F mov ecx, [ebp+var_C]
.text:2122EB52 push ecx ; char *
.text:2122EB53 mov edx, [ebp+arg_4]
.text:2122EB56 push edx ; char *
.text:2122EB57 call strcpy

```