

MWR InfoSecurity Security
Advisory

WebSphere MQ xcsGetMem
Heap Overflow Vulnerability

12th January 2009



Contents

| | | |
|----------|---|----------|
| 1 | Detailed Vulnerability Description | 5 |
| 1.1 | Introduction | 5 |
| 1.2 | Technical Background..... | 5 |
| 1.3 | Vulnerability Details..... | 6 |
| 1.4 | Exploit Information | 6 |
| 1.5 | Dependencies | 7 |
| 2 | Recommendations..... | 8 |

WebSphere MQ xcsGetMem Heap Overflow

| | |
|-------------------------------|---|
| Package Name: | WebSphere MQ |
| Vendor Notified: | 20 th September 2007 |
| Advisory Release Date: | 12 th January 2009 |
| Affected Versions: | WebSphere MQ 6.0.0.0 on Windows is confirmed to be vulnerable. Other versions and platforms may also be affected by this issue. |

| | |
|---------------------------------|---|
| CVE Reference | CVE-2008-4289 |
| Author | M. Ruks |
| Severity | High Risk |
| Local/Remote | Remote |
| Vulnerability Class | Integer Overflow leading to Heap Buffer Overflow |
| Vendor Response | Updated packages have been created and are included in MQ release 6.0.2.4. |
| Exploit Details Included | Yes (although no exploit code is provided) |
| Affected OS | Microsoft Windows is confirmed to be vulnerable although other platforms are also expected to be affected |

Overview:

The WebSphere MQ service can be used to transfer messages between systems and applications. An integer overflow and subsequent heap overflow vulnerability has been identified in the packet parsing routines. This vulnerability is associated with the memory allocation code and can result in the overwriting of data on the heap. This vulnerability could be exploited to execute arbitrary code.

Impact:

The vulnerability could enable an attacker to remotely execute arbitrary code on the affected system with the privileges of the MQ process. An attacker could also use this vulnerability to create a Denial of Service condition for legitimate users.

Cause:

The vulnerability arises from an error in the wrapper function written around the 'malloc' call within the "amqxcs2.dll" library. An integer overflow and heap buffer overflow vulnerability is present in the 'xcsGetMem' function. It is possible to trigger a memory overwrite as length fields contained within specific packet types are not checked correctly before being passed to the vulnerable function.

Interim Workaround:

One method for mitigating the risk associated with this issue would be to use network filtering to restrict access to WebSphere MQ services to trusted IP addresses only.

Solution:

The vendor supplied patches should be installed to resolve this issue. Links to the updated software can be discovered at the following location: -

<http://www-01.ibm.com/support/docview.wss?rs=171&uid=swg27006037>

1 Detailed Vulnerability Description

1.1 Introduction

WebSphere MQ is an Enterprise level application that can be used to provide a unified platform for messaging within an organisation. IBM describes their technology as follows: -

“WebSphere MQ provides an award-winning messaging backbone for deploying your enterprise service bus (ESB) today as the connectivity layer of a service-orientated architecture (SOA).”

Source: <http://www-306.ibm.com/software/integration/wmq/>

Communication with MQ services can be achieved in a number of ways and the technology requires a feature rich API and extensions in order to integrate with an Enterprise environment.

1.2 Technical Background

The main component of a WebSphere MQ instance is the Queue Manager. This is a process that manages the message queues and provides interfaces (known as channels) to the applications wishing to communicate with them. By default, a Queue Manager will listen on a network interface for incoming connections and process the data accordingly. A Queue Manager will accept any type of MQ data and begin processing it before determining whether the packet is authorised or has been received at the correct point within the application's “state machine”.

The result of this fact is that a large amount of MQ code is exposed to unauthenticated users. Consequently, any vulnerabilities in the code used to parse the data after it has been passed from the network socket can potentially be exploited by an attacker. For example, if an MQSET packet is sent to a Server Connection channel before the MQCONN connection call has been completed, MQ will parse the data structure by making calls to functions such as ‘xcsGetMem’. Once the parsing has been completed MQ will check whether the state machine is setup such that the packet belongs to a session that has been correctly established with the Queue Manager at the application level.

If the session has not been established, an error message will be returned; for example, stating that the request was “Not Authorised” or that the communication was “Terminated by a Remote Exit”. However, the parsing of the packet would still be completed before the state machine disallowed the execution of the command itself. The ‘xcsGetMem’ function is called when parsing all MQ packets; however, the data fields which are interpreted vary between packet types.

After receiving data from the network socket the MQ application will process and parse the data in various ways. The exact nature of this parsing is not within the scope of this document; however, it is important to note that a large amount of this activity occurs before a connection to the Queue Manager is fully established.

In certain types of packet the ‘xcsGetMem’ function will use a value supplied within a packet as an argument to a memory allocation operation. Within this function the allocation always occurs after a static value has been added to the length field, which is defined as an integer.

1.3 Vulnerability Details

A vulnerability was identified such that an attacker could affect the size of the memory allocation for a 'malloc' call within the 'xcsGetMem' function. The size of the memory allocation used within the 'malloc' call is passed to the function through the EAX register (this data is the size of the data which will be processed in a later operation). A static value of 20h is added to the data in EAX before the call as can be observed in the disassembly included here: -

```
.text:4E615849          mov     ebx, [esp+14h+arg_8]
.text:4E61584D          lea   eax, [ebx+20h]
.text:4E615850          push  eax                ; size_t
.text:4E615851          call  ds:malloc
```

If a value between 0xfffffe0 and 0xffffffff is sent in the packet the integer will overflow and the 'malloc' call will be made for an area of memory which is smaller than the size of the data in the packet.

After the call is made execution will continue and the heap buffer which has been assigned will be used when conversion routines are called later in the packet parsing. WebSphere MQ will accept a number of encoding types and therefore the application is required to convert these into a standard format before performing further processing on the data.

The conversion function operates by copying the converted bytes in the packet into the heap buffer allocated previously. However, as the size checking function has been manipulated through the integer overflow the application performs an incorrectly bounded copy into the heap buffer.

This operation results in the overwriting of data on the heap with arbitrary data supplied in the data packet. Therefore this condition is an exploitable heap overflow as a result of the integer overflow within the 'xcsGetMem' function.

Vulnerability Impact

An attacker capable of exploiting the vulnerability would be able to gain execution control within the process and thereby execute arbitrary code. The "amqrmppa" process, which handles data from network connections, runs with the privileges of the MQ user (this is dependent on the platform, for example, mqm on UNIX or MUSR_MQADMIN on Microsoft Windows). Therefore, an attacker would be able to execute arbitrary code with the privileges of that user.

1.4 Exploit Information

The vulnerability exists within the 'xcsGetMem' memory allocation function after which an overflow can occur within the conversion routines. Therefore, the vulnerability potentially affects all packets parsed by the function where the length field is read from the packet data.

When an MQSET packet is received the Character Length field is passed to the vulnerable function. This is data specified within the packet and is therefore controllable by an attacker. By setting this length field to a value which will cause the integer overflow it is possible to

trigger the integer overflow and allow execution to pass to the conversion function. The method of exploitation chosen utilised the standard ASCII character set which was specified in the Transmission Segment Header of the MQ packet.

It should be noted that it may be possible to exploit this vulnerability using other types of MQ packet. In the example provided here an Initial Data handshake must have occurred for the MQSET packet to be processed in a manner which will trigger the vulnerability.

The packet payload can be constructed to overwrite data structures on the heap after the allocated buffer. On the Windows 2000 platform it is possible to overwrite the subsequent heap header with the location of a known function (for example, PEB Lock) and the address of the shellcode to be executed. Once the overwritten header is used in internal heap management it will cause the pointer to the PEB Lock function to be overwritten with the address of the shellcode.

When a lock operation now occurs the shellcode will be called and control over execution will be gained. It should be noted that a number of additional operations are required to repair the stack and the overwritten function pointer in order to prevent exceptions from occurring which could prevent execution of the shellcode.

MWR InfoSecurity has been able to construct a working exploit for Microsoft Windows platforms although the code will not be released into the public domain at the present. The decision to release such code in the future will be taken based on MWR InfoSecurity's obligations to protect its customers and Critical National Infrastructure (CNI) whilst also enabling the security community to accurately assess the vulnerability of systems running the software.

1.5 Dependencies

This vulnerability has been tested on WebSphere MQ version 6.0 on the Microsoft Windows platform. However, as the vulnerability is present within the memory allocation wrapper of the MQ product it is expected to be present in every version of the software. It should be noted that successful exploitation will be dependent on utilising a technique appropriate to the nature of the heap on the affected platform.

2 Recommendations

It is recommended that all users install the appropriate security patches released by the vendor in response to this issue. Links to the updated software can be discovered at the following location: -

<http://www-01.ibm.com/support/docview.wss?rs=171&uid=swg27006037>

Interim Workaround

Both network and host based traffic filtering controls could be implemented to protect access to the Queue Manager service on the affected hosts. This should be considered at both the packet filtering level and by the use of IPSec tunnels between hosts.

MWR InfoSecurity
St. Clement House
1-3 Alencon Link
Basingstoke, RG21 7SB
Tel: +44 (0)1256 300920
Fax: +44 (0)1256 844083
mwrinfosecurity.com