

MWR InfoSecurity Security
Advisory

IBM WebSphere MQ -
rriLookupGet Remote Denial
of Service Vulnerability

4th March 2010



Contents

1	Detailed Vulnerability Description	4
1.1	Introduction	4
1.2	Technical Background.....	5
1.3	Exploit Information	6
1.4	Dependencies	7
2	Recommendations.....	8

"rriLookupGet" Remote Denial of Service

Package Name:	IBM WebSphere MQ
Date:	2010-01-15
Affected Versions:	WebSphere 7.0.0.2 and 7.0.0.1 on Windows are confirmed to be vulnerable. Other versions and platforms may also be affected by this issue.

CVE Reference	CVE-2009-3161
Author	A. Plaskett
Severity	High Risk
Local/Remote	Remote
Vulnerability Class	Denial of service
Vendor URL	http://www-306.ibm.com/software/integration/wmq/
Vendor Response	A patch is available from the following URL: http://www-01.ibm.com/support/docview.wss?uid=swg24024153
Exploit Details Included	Yes (although no exploit code is provided). Proof of concept code is available on request.

Overview:

The WebSphere MQ service can be used to transfer messages between systems and applications. A vulnerability was identified with the packet handling routines which would allow a malicious attacker to cause a denial of service condition.

Impact:

This vulnerability could be exploited to prevent or disrupt legitimate users from accessing a Queue Manager. It is also possible that the MQ service could be forced to produce large memory dumps potentially disclosing sensitive information or resource exhaustion.

Cause:

The vulnerability is caused by a logic error in the code which allows a pointer to be dereferenced when the MQ state machine processes an unexpected packet with a segment type of 0xE.

Interim Workaround:

The use of SSL for authentication would mitigate this risk and restrict exploitation to those users in possession of a valid SSL client certificate. It is not expected that a security exit could prevent exploitation of this vulnerability, although only a limited number of exits have been tested at this point.

Solution:

The vendor supplied patches should be installed to resolve this issue. Links to the updated software can be found at the following location:
<http://www-01.ibm.com/support/docview.wss?uid=>

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.

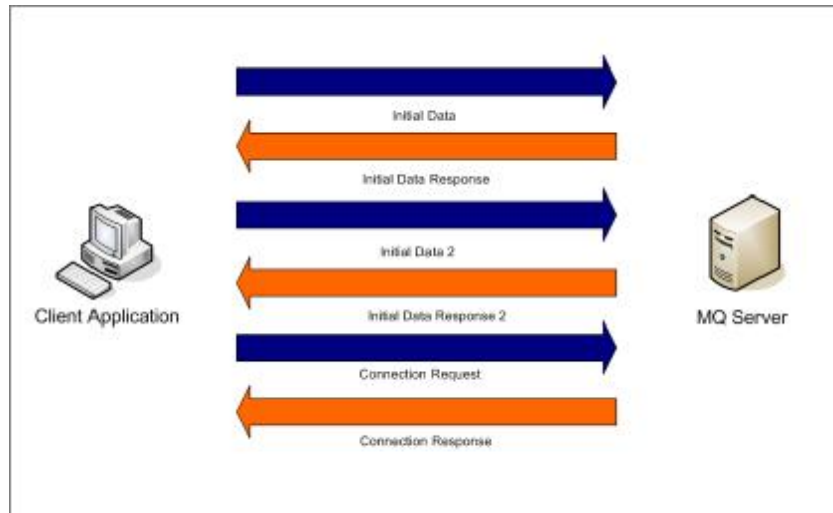
The main component of a WebSphere MQ instance is the Queue Manager. This is a process that manages the message queues and provides interfaces (known as channels) to the applications wishing to communicate with them. By default, a Queue Manager will listen on a network interface for incoming connections 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 vulnerability in the code used to parse the data after it has been passed from the network socket can potentially be exploited by an attacker. 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. 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.

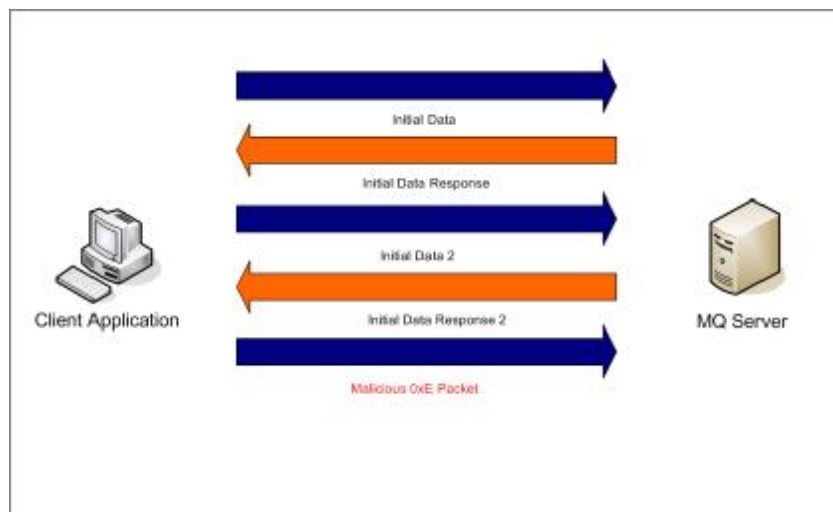
1.2 Technical Background

A vulnerability exists in the state machine which handles incoming MQ networking packets; this issue could be exploited to disrupt the MQ service for legitimate users.

Typically a legitimate protocol session would occur as follows:



However, if the packet immediately following the second initial data handshake response is sent with the segment type set to 0xE a denial of service occurs. This is because the “amqrmppa” process generates an exception and is terminated. A diagram which illustrates a malicious exchange is included here: -



The vulnerability is triggered by the following code located in the “rriLookupGet” function. If EBP is null, the dereference operation fails and an exception is thrown.

```
.text:4E8FCE4E      mov     esi, [ebp+18h] ; ebp is null at this point.
```

The assignment of the EBP register in the “rriLookupGet” function occurs here which highlights that the value is obtained from a value passed to the function on the stack:

```
.text:4E8FCE1E          mov     ebp, [esp+28h+arg_C]
```

At this location in the code the EBP register has been assigned a value that has been passed from the previous function. This is the third argument passed to the called function “rriLookupGet” from the callee “rstReceiveMessageRequest”. This can be observed in the output below which shows the ECX register being used to pass this value within the “rstReceiveMessageRequest” function. It should be noted that in this instance EBP is being used as a general purpose register.

```
.text:4E8FC181          push   ecx
.text:4E8FC182          push   ecx
.text:4E8FC183          lea   edx, [esp+34h+var_18]
.text:4E8FC187          push   edx
.text:4E8FC188          push   eax
.text:4E8FC189          push   ecx           ; null passed here
.text:4E8FC18A          mov   ecx, [esp+40h]
.text:4E8FC18E          push   ecx
.text:4E8FC18F          push   edi
.text:4E8FC190          call  _rriLookupGet
```

However, before it is pushed to the stack the third argument passed to the “rriLookupGet” function (in this instance ECX) has already been XOR'd, thus its value is 0 and hence a null pointer in the operation that generates the exception.

```
.text:4E8FC17C          xor    ecx, ecx
```

To reach these functions in the code the “rstReceiveMessageRequest” function must first be executed; the “rstReceiveMessageRequest” function is itself called from the “rriServerAsyncRcv” function when the segment type in the packet is set to 0xE.

It is expected that it is intended that the “rriLookupGet” function should only be called once a valid session has been established. Therefore, by sending a packet with a segment type of 0xE before a session is established an unexpected condition occurs due to the presence of a null pointer. The cause of the vulnerability is therefore the incorrect state machine handling of a poorly sequenced MQ packet and the failure to check for a valid session before calling the affected functions. Consequently, if an MQCONN is not made before a segment type 0xE is received by the Channel Process a denial of service condition will occur within the “amqrmppa” process.

1.3 Exploit Information

In order to trigger this vulnerability an MQ packet with the segment type of 0xE has to be sent following the initial and second handshake. Normally an MQCONN

would be made following the completion of the initial and second handshake; however, to trigger this vulnerability a 0xE segment type must be sent.

An MQCONN packet can be used with the TSH(M) segment type set to 0xE in order to create the denial of service.

It is not expected that this vulnerability is further exploitable due to the type of vulnerability. However, this may change due to future advances in exploitation and architectural differences between platforms affected. However, if an attacker were in a position to control memory layout this could become an exploitable condition. It is therefore recommended that the cause of this issue is further investigated.

1.4 Dependencies

Initial testing indicated that it would be possible to prevent this specific, vulnerable code path being executed by the use of a security exit. However, the security exit tested 'amqrspin.dll' was prone to another issue which would cause an earlier access violation to happen which would also cause the process 'amqrmppa' to terminate. Consequently, this issue is rated as high risk as an unauthorised attacker could cause the 'amqrmppa' process to terminate regardless of whether a security exit had been implemented or not. The use of SSL for authentication would mitigate this risk and restrict exploitation to those users in possession of a valid SSL client certificate.



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 found at the following location:

<http://www-01.ibm.com/support/docview.wss?uid=swg24024153>

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