

MWR InfoSecurity Security
Advisory

Websphere MQ MCAUSER
Setting Bypass Vulnerability

15th May 2007

MWR  INFOSECURITY

CONTENTS

| | | |
|----------|--|----------|
| 1 | Detailed Vulnerability Description..... | 5 |
| 1.1 | Introduction | 5 |
| 1.2 | Technical Background | 5 |
| 1.3 | Exploit Information | 7 |
| 1.4 | Dependencies | 8 |
| 2 | Recommendations..... | 9 |

Websphere MQ Security Exit Authentication Bypass Vulnerability

| | |
|---------------------------|---|
| Package Name: | Websphere MQ |
| Date: | 30 th April 2007 |
| Affected Versions: | Websphere MQ 5.1 – 5.3 on Solaris are confirmed to be vulnerable Websphere MQ 6.0.0.0 on Windows is confirmed to be vulnerable |

CVE Reference: CVE-2008-1130

Severity: High Risk

Local/Remote: Remote

Vulnerability Class: Authorisation Bypass

Vendor Response: The vendor has been contacted, and a fix has been released

Exploit Details Included: Yes (although no exploit code is provided)

Affected OS: Solaris, Windows (although others could also be affected)

Overview:

The Websphere MQ service can be used to transfer messages between systems and applications. It is possible to lock down access to channels by setting an invalid MCAUSER. A method of bypassing this authorisation control has been discovered which would enable unauthorised access to be gained.

Impact:

The vulnerability could enable an attacker to access sensitive channels that have been restricted with the MCAUSER parameter. This would enable an attacker to gain full read and write access to all queues defined within the channel. Additionally, an attacker could perform remote fingerprinting of the software, alter the Queue Manager configuration and potentially execute Operating System commands through the creation of an appropriate trigger process. The latter would be dependent on the presence of an existing trigger monitor process running on the system. Additionally, on version 6 of the software an attacker could create a service to execute commands on the system.

Cause:

The vulnerability arises from an error in the state model responsible for permitting connections to a channel. A connection can be established to the Queue Manager if the "2035 Not Authorised" response is ignored and the connection attempt continues.

Interim Workaround:

The workarounds at the present time involve using additional security mechanisms such as Security Exits to protect all channels on a Queue Manager. Additionally, it is possible to use network filtering to restrict access to Websphere MQ services to trusted IP addresses only. It is also possible that correctly deploying and mandating the use of SSL client certificates will prevent an attacker from accessing the channels on the Queue Manager. IBM have released a fix pack, the link of which is provided below.

Solution:

IBM has released the following solution(s) and fix packs:

<http://www-1.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 manners and the technology requires a feature rich API and extensions in order to integrate with an Enterprise environment.

1.2 Technical Background

Technical Introduction to MQ

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. The Queue Manager can communicate using a number of protocols including IP and SNA (LU 6.2) amongst others.

MQ services on a given system can be detected by the presence of open TCP ports. By default, WebSphere MQ listens on TCP port 1414; however, it is often the case that other ports are used to run Queue Managers and an attacker would therefore be required to search for these services.

Queue detection requires a handshake to be attempted on each open TCP port. A tool was written in-house and used to search for legitimate message queues by sending an MQ “INITIAL_DATA” packet (the segment type was set to 01h in the Transmission Segment Header).

To connect to an instance of Websphere MQ it is necessary to establish communications with a specific channel. A channel is essentially a conduit through which communication with the message queues can occur. Therefore, it is necessary to know a valid channel on the remote system before connection can occur. A number of server connection channels are present by default and these will often be the first target of an attacker. In most cases the default channel “SYSTEM.DEF.SVRCONN” can be used to attempt attacks against the service.

If a legitimate MQ service is running on a port a valid MQ packet will be returned containing the name of the Queue Manager. Other possible responses could be error messages; for instance, stating the channel name was incorrect or that the Queue Manager was not available. This technique enables an attacker to identify the presence of a Queue Manager even when located on a port away from the default value.

MCAUSER Parameter

Access control within Websphere MQ is handled based on the user ID of the process making calls (MQI) on the system running the Queue Manager. When connecting using a client it is the process associated with the Server Connection channel that issues the MQI calls. The user ID making the MQI call is ultimately dependent on the MCAUSER Identifier that is defined for the channel.

The MQI call is then made using a user ID that is determined based on a series of rules that relate to the value specified in the packets from the user and the MCAUSER value configured on the channel. However, it is widely accepted that by specifying an invalid username in the MCAUSER parameter for a channel it is not possible to access the channel remotely.

MQ Connection Process

When a connection to the MQ service is attempted a number of steps are involved. The first step involves an exchange of initial data which continues until the systems have agreed on parameters such as protocol version; heartbeat interval and sequence wrap value. At this stage, the remote Queue Manager also notifies the client whether authentication is required by setting the "Server Connection Security" bit. The initial data exchange can involve a number of packets being exchanged until suitable values have been agreed by the client and server.

Once this exchange of initial data has occurred the client will typically send an authentication packet to the server (those using the MQ APIs will do so by default). This will contain the authentication information required to access the channel. If a security exit is not defined the Queue Manager will ignore this authentication packet and the client system can attempt to connect.

The MCAUSER parameter set on the Queue Manager will be the primary factor in determining the success or failure of this connection attempt. If a valid MCAUSER is set for the channel the Queue Manager will return an MQCONN REPLY packet to inform the client their connection was successful. A client can then continue to communicate with the channel and request access to the Queues that are defined. If the MCAUSER is not set to an authorised user the Queue Manager will return a packet with the reason code set to 2035 which is a "Not Authorised" error message. Upon receiving this response the client should then terminate its connection attempt to the Queue Manager.

Vulnerability Details

A vulnerability was discovered such that an attacker could bypass the "2035 Not Authorised" reason code and gain access to the channel. On affected systems this could result in unauthorised access being gained to the channels that were supposedly restricted through an invalid MCAUSER parameter.

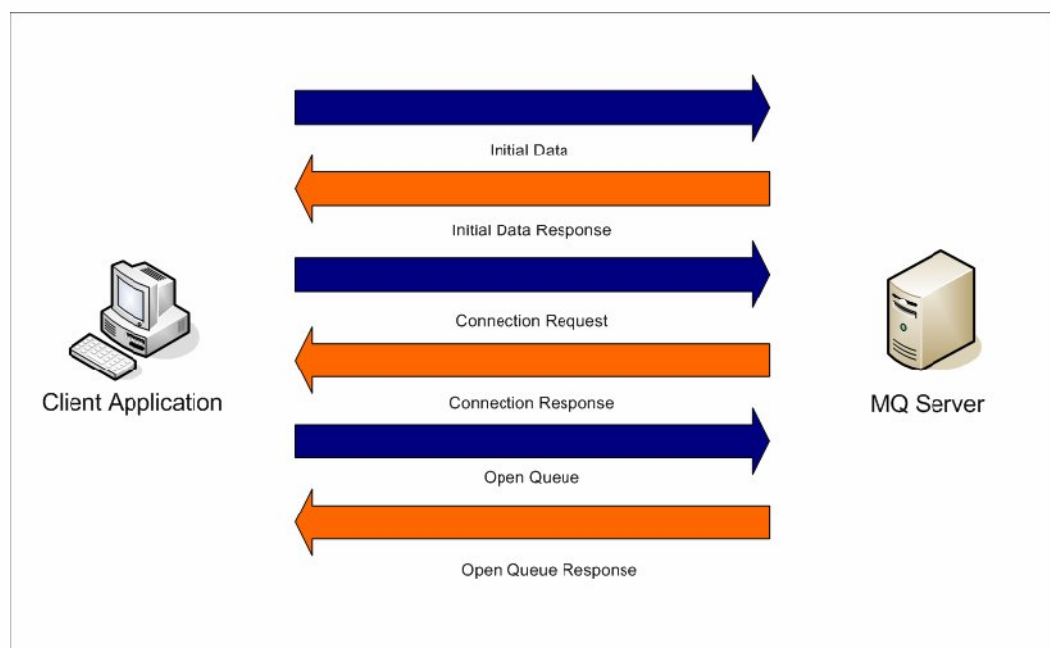
The sequence of packets required to gain access to an affected system is described in the Exploit Information section of this document. This vulnerability was confirmed on Websphere MQ versions 5.1, 5.2 and 5.3 on Solaris and version 6.0 on Windows.

Vulnerability Impact

An attacker capable of bypassing the authorisation controls can gain full access to a restricted channel. Depending on the configuration of the software this could result in a serious security breach. An attacker would be able to communicate with the queues associated with the channel and this could result in the compromise of the confidentiality or integrity of the Queue data. Additionally, if an attacker were able to access the Command Server through the channel it would be possible to alter the configuration of the Queue Manager and so could potentially result in Operating System commands being executed.

1.3 Exploit Information

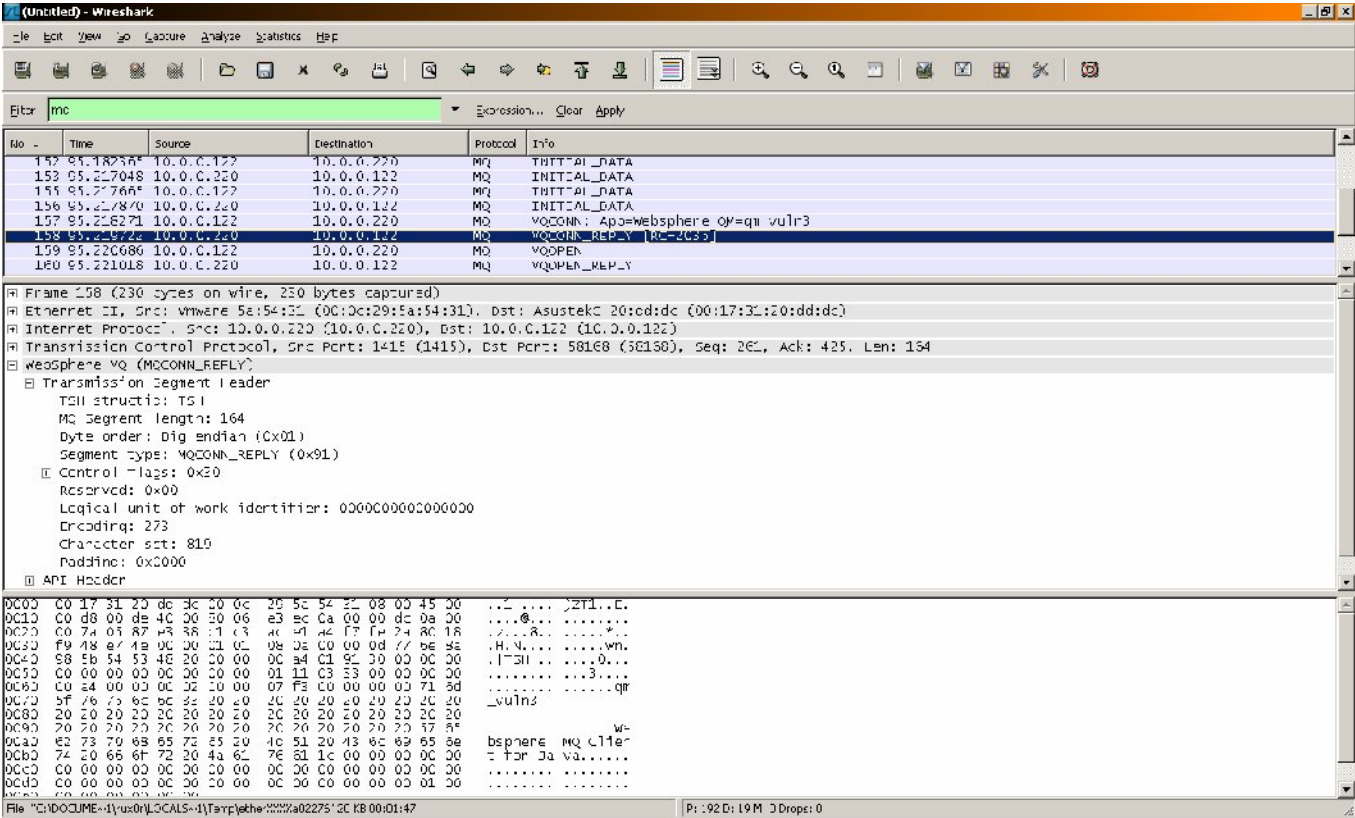
The invalid MCAUSER defined for a channel can be bypassed by using a modified connection process. Therefore successful exploitation requires a custom MQ client to be written to perform this bypass attack. Code to perform the attack will not be provided at the present time; however, the sequence of packets that can be used to gain access to a protected channel is included here: -



As can be observed, the authentication sequence is not attempted during the handshake. If the MCAUSER parameter is set to a non-valid user such as "nobody" the Connection response packet will contain the "2035 Not Authorised" message. However, this does not prevent subsequent requests in order to manipulate queues on the channel.

The result of this is that the connection request is accepted and access is granted to the channel. This exploit does not enable GET and PUT restrictions applied to queues to be bypassed but if the channel has Command Server access it becomes possible to reconfigure the Queue Manager using PCF commands.

A screenshot of a "Wireshark" packet capture obtained whilst performing this process can be observed below with the highlighted packet indicating the authorisation error that is returned.



Exploitation of a test system was demonstrated by placing a message on a queue on a restricted channel. This was also possible on MQ installations which have been tested by MWR InfoSecurity.

1.4 Dependencies

To exploit this vulnerability a channel must not be configured with a security exit as this provides additional authentication checks. However, the vendor recommended method for securing DEFAULT and SYSTEM channels is to set the MCAUSER parameter to an invalid user, with no specific recommendation as to security exits.

This vulnerability has been tested on Websphere MQ versions 5.1, 5.2 and 5.3 on a Solaris platform and version 6.0.0.0 (evaluation download) on the Windows platform. It is possible that further versions of the software are vulnerable to this issue.

2 Recommendations

To resolve the issue it is important that the State Machine implemented by the software is reviewed. This should ensure that communication cannot continue if a "2035 Not Authorised" error is returned from the connection request. It is therefore recommended that the vendor perform investigations to establish the cause of the vulnerability and provide an appropriate resolution.

It is recommended that all users install any security patches released by the vendor in response to this issue.

Interim Workaround

The use of a security exit is one of the most effective mechanisms for protecting an installation of Websphere MQ. It is therefore recommended that appropriate exits be applied to all channels, even those set with an invalid MCAUSER.

An additional method of mitigation is the use of client SSL certificates to protect all access to channels. It is important that the SSL filtering is performed in a rigorous manner and that all trusted Certificate Authorities are removed from the MQ system except for those that are under direct control of the organisation.

Additionally, 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.

If possible all unused channel definitions should be removed from the Queue Manager to lower the attack surface area.

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