MQ Set UserID Installation and Operation Manual

Capitalware Inc.
Unit 11, 1673 Richmond Street, PMB524
London, Ontario N6G2N3
Canada
sales@capitalware.com
https://www.capitalware.com

Last Updated: January 2021.

Table of Contents

1 Introduction

1.1 Overview

MQ Set UserID (MQSUI) provides the ability to explicitly set a UserID via a MQ Channel's Send, Receive or Message Exit.

MQSUI operates with IBM MQ v7.0, v7.1, v7.5, v8.0, v9.0, v9.1 and v9.2 in Windows, Unix, IBM i (OS/400) and Linux environments. It operates with Sender, Receiver, Server, Requester, Cluster-Sender, Cluster-Receiver, Server Connection and Client Connection channels of the MQ queue managers.

How to specify the UserID that MQSUI will set in the message's MQMD UserIdentifier field:

- If a value is set in the SNDDATA, RCVDATA or MSGDATA field (i.e. u=roger) then that value will be used
- If the SNDDATA, RCVDATA or MSGDATA field is blank then MQSUI will use the UserID specified in the channel's MCAUSER field
- If the SNDDATA, RCVDATA or MSGDATA field is blank and the channel's MCAUSER field is blank then MQSUI will use the value specified in the IniFile (i.e. UserID=roger)
- If MQSUI cannot find a value in any of those fields then MQSUI will stop the channel and output an error message in the MQSUI logfile.

MQSUI is a simple drop-in solution. MQSUI can be configured as a channel's Send, receive or message exit.

The MQSUI solution is licensed as '***Licensed As Free***' meaning a user can freely download the software and use it (i.e. Same as how IBM licenses MQ Client software.). If the user wants support for MQSUI then they can purchase a support subscription from Capitalware.

On AIX, HP-UX, Linux, Solaris and Windows, MQSUI can be configured and used with a non-default installation of MQ in a multi-install MQ environment.

Note: Raspberry Pi is a Linux ARM 32-bit OS (Operating System). Hence, simply follow the Linux 32-bit instructions for installing and using the solution on a Raspberry Pi.

1.2 Executive Summary

The MQSUI solution is an MQ Receive and Message exit. It is available for a wide range of platforms: AIX, HP-UX, IBM i, Linux, Solaris and Windows.

Major Features of MQSUI:

- Easy to set up and configure
- No application changes required
- Can be configured as either queue manager to queue manager or client application to queue manager solution
- Standard MQ feature, GET-with-Convert, is supported
- Provides high-level logging capability

1.3 Prerequisites

This section details the minimum supported software levels. These prerequisites apply to both client-side and server-side installations of MQ Set UserID.

1.3.1 Operating System

MQ Set UserID can be installed on any of the following supported servers:

1.3.1.1 IBM AIX

- IBM AIX 6L version 6.1 or higher

1.3.1.2 HP-UX IA64

- HP-UX v11.23 or higher

1.3.1.3 IBM i (OS/400)

- IBM i V6R1 or higher

1.3.1.4 Linux x86

- Red Hat Enterprise Linux v5, v6, v7, v8
- SUSE Linux Enterprise Server v11, v12, v15

1.3.1.5 Linux x86_64 (64-bit)

- Red Hat Enterprise Linux v5, v6, v7, v8
- SUSE Linux Enterprise Server v11, v12, v15

1.3.1.6 Linux on POWER

- Red Hat Enterprise Linux v5, v6, v7, v8
- SUSE Linux Enterprise Server v11, v12, v15

1.3.1.7 Linux on zSeries (64-bit)

- Red Hat Enterprise Linux v5, v6, v7, v8
- SUSE Linux Enterprise Server v11, v12, v15

1.3.1.8 Raspberry Pi (Linux ARM 32-bit)

- Raspberry Pi OS v9 or higher

1.3.1.9 Sun Solaris

- Solaris SPARC v10 or higher
- Solaris x86_64 v10 or higher

1.3.1.10 Windows

- Windows 2008, 2012 or 2016 Server (32-bit & 64-bit)
- Windows 7, 8, 8.1 & 10 (32-bit & 64-bit)

1.3.2 IBM MQ

- IBM MQ v7.0, v7.1, v7.5, v8.0, v9.0, v9.1 and v9.2 (32-bit and 64-bit)

Operating System	MQ v7.0, v7.1, v7.5, v8.0, v9.0, v9.1 and v9.2
AIX v6.1 or higher	64-bit
HP-UX IA64 v11.23 or higher	64-bit
IBM i (OS/400)	64-bit
Linux x86	32-bit
Linux x86_64	64-bit
Linux on POWER	64-bit
Linux on zSeries	64-bit
Raspberry Pi ARM	32-bit
Solaris SPARC v10 & v11	64-bit
Solaris x86_64 v10 & v11	64-bit
Windows 2008, 2012, 2016, 7, 8, 8.1 & 10	32-bit & 64-bit

1.3.3 Windows 32-bit

The following is the software prerequisite for Windows 32-bit:

- Microsoft Visual C++ 2010 Redistributable Package (x86)
  https://www.microsoft.com/en-ca/download/details.aspx?id=5555

1.3.4 Windows 64-bit

The following is the software prerequisite for Windows 64-bit:

- Microsoft Visual C++ 2010 Redistributable Package (x64)
  https://www.microsoft.com/en-ca/download/details.aspx?id=14632

2 Installing MQ Set UserID

This section describes how to install Capitalware's MQ Set UserID. For Windows, it is available as a Windows installer package called: **mqsui-setup.exe**. When the user runs this package, it will install the server-side exit.

2.1 Server-side Security Exit

The following files are the platform specific server-side exits and the required initialization file (IniFile).

- **mqsui.dll** is the exit DLL for Windows that will be invoked by the MQ component.
- **mqsui** is the exit shared library for Unix or Linux that will be invoked by the MQ Server component.
- **MQSUI** is the exit for IBM i that will be invoked by the MQ Server component.
- **MQSUIJ.jar** is the Java client-side exit that will be invoked by the MQ Client component. It requires Java v1.3 or higher.
- **mqsui.ini** is a sample initialization file for the server-side exit.
- **mqsui.sample.MQA1.mqsc** is a sample 'MQA1' MQSC script to define MQ channels with the message exit.
- **mqsui.sample.MQB1.mqsc** is a sample 'MQB1' MQSC script to define MQ channels with the message exit.
- **AddRegistryEntries.bat** is a simple Windows batch file to add Event Log registry entries.
- **mqsui.reg** is the registry entries for the Windows Event Log.
- **rotatelog.sh** is a simple Unix shell script roll the log file to a backup file.
- **rotatelog.bat** is a simple Windows batch file to roll the log file to a backup file.
- **setmqsui.sh** is a simple Unix shell script to set the appropriate file permissions.

2.1.1 Windows Installation

To install the exit on Windows, first unzip the **mqsui-setup.zip** and then run the mqsui_setup.exe file. Follow the on-screen instructions and the exit will be installed in the **C:\Capitalware\MQSUI\** directory (default installation).

The user may copy or ftp the mqsui.dll, mqsui.ini, AddRegistryEntries.bat and mqsui.reg files from one Windows server to another Windows server.

2.1.2 Linux 32-bit Installation

To install the 32-bit version of MQSUI on Linux, first unzip the **mqsui-setup.zip** and then select the appropriate TAR file for the target platform. You will find 3 TAR files in the original ZIP file:

- **Java/MQSUIJ.jar**
- **Linux_x86/mqsui_linux.tar**
- **RaspberryPi_ARM/mqsui_raspberrypi_arm.tar**

Steps to install the server-side exit:

1. ftp or copy the selected TAR file to the target platform to the ***/var/mqm/exits/*** directory.
2. Un-tar the mqsui_xxx.tar file into the ***/var/mqm/exits/*** sub-directory (xxx is either aix, hpux, solaris or linux)

   ```
   cd /var/mqm/exits/
   tar -xvf  mqsui_xxx.tar
   ```

3. Change directory to ***/var/mqm/exits/***
4. Next, do the following commands against ***mqsui***:

   ```
   chmod +x setmqsui.sh
   ./setmqsui.sh
   ```

2.1.3 Unix and Linux 64-bit Installation

To install the 64-bit version of MQSUI on Unix or Linux, first unzip the **mqsui-setup.zip** and then select the appropriate TAR file for the target platform. You will find 7 TAR files in the original ZIP file:

- **AIX/64-bit/mqsui_aix53_64.tar**
- **AIX/64-bit/mqsui_aix61_64.tar**
- **HPUX_IA64/mqsui_hpux64_IA64.tar**
- **Linux_x86_64/mqsui_linux_x86_64.tar**
- **Linux_POWER/mqsui_linux_power64.tar**
- **Linux_zSeries/64-bit/mqsui_linux_zseries64.tar**
- **Solaris_SPARC/64-bit/mqsui_solaris64.tar**
- **Solaris_x86_64/mqsui_solaris_x86_64.tar**

Steps to install the server-side exit:

1. ftp or copy the selected TAR file to the target platform to the ***/var/mqm/exits64/*** directory.
2. Un-tar the mqsui_xxx.tar file into the ***/var/mqm/exits64/*** sub-directory (xxx is either aix, hpux, solaris or linux)

   ```
   cd /var/mqm/exits64/
   tar -xvf  mqsui_xxx64.tar
   ```

3. Change directory to ***/var/mqm/exits64/***
4. Next, do the following commands against ***mqsui***:

   ```
   chmod +x setmqsui.sh
   ./setmqsui.sh
   ```

2.1.4 IBM i Installation

To install the MQSUI on IBM i , first unzip the **mqsui-setup.zip** and then select the files in the IBM i directory.

- **mqsui.savf** is the IBM i 'Save File' that contains the library with the security exit.

- **mqsui_iseries.tar** is the IBM i IFS TAR file that contains a sample initialization file for the server-side security exit and sample MQSC script to define MQ channels with the security exits.

Steps to install the server-side security exit:

1. Log onto the target IBM i server and do the following command:

   ```
   CRTSAVF FILE(QGPL/MQSUI)
   ```

2. ftp the IBM i files to the IBM i server as follows:

   ```
   ftp -s:mqsui_iseries.ftp  iseries_hostname
   ```

   ```
   your-IBM i-userid
   your-IBM i-password

   binary
   cd QGPL
   put mqsui.savf

   quote SITE NAMEFMT 1

   cd /QIBM/UserData/mqm/
   put mqsui_iseries.tar
   quit
   ```

3. Log onto the target IBM i server and do the following commands:

   ```
   RSTLIB SAVLIB(MQSUI) DEV(*SAVF) SAVF(QGPL/MQSUI)
   CLRSAVF FILE(QGPL/MQSUI)
   CHGOBJOWN  OBJ(MQSUI) OBJTYPE(*LIB)  NEWOWN(QMQM)
   qsh
   cd /QIBM/UserData/mqm/
   tar -xvf mqsui_iseries.tar
   chown -R QMQM mqsui
   rm mqsui_iseries.tar
   ```

3 Configuring QMgr to QMgr Channels

This section describes how to configure the exit. By default, MQSUI will use the UserID specified in the channel's MCAUSER field.

Optional, configuration parameters can be specified in the MSGDATA attribute field.

- MSGDATA

  `U=roger`
  Where 'roger' is the UserID to be written to each messages MQMD UserID field.

  `D=Y`
  Where 'Y' enables LogMode of Debug.

Note: Use a semicolon to separate the MQSUI parameters in the MSGDATA attribute field.

Note: Message Exit Data must NOT exceed 32 characters.

3.1 Receiver Channel

This section describes the necessary entries to enable the exit. The MQ Administrator will need to update 2 fields of the RECEIVER Channel that the exit will be applied to.

3.1.1 Windows

On Windows, MSGEXIT and MSGDATA will contain the following values assuming a default install.

- MSGEXIT
  `C:\Capitalware\MQSUI\mqsui(RE)`

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE( RCVR ) +
       TRPTYPE( TCP ) +
       MSGEXIT('C:\Capitalware\MQSUI\mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.1.2 Linux 32-bit

On Linux, MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT
  `/var/mqm/exits/mqsui(RE)`

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE( RCVR ) +
       TRPTYPE( TCP ) +
       MSGEXIT('/var/mqm/exits/mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.1.3 Unix and Linux 64-bit

On Unix and Linux (excluding Linux x86), MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT
  /var/mqm/exits64/mqsui(RE)

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE( RCVR ) +
       TRPTYPE( TCP ) +
       MSGEXIT('/var/mqm/exits64/mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.1.4 IBM i

On IBM i, MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT is made up of 10 characters for program name (padded with blanks) followed by 10 characters for the LIBRARY name (padded with blanks).

  MQSUI   MQSUI

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE( RCVR ) +
       TRPTYPE( TCP ) +
       MSGEXIT('MQSUI     MQSUI     ') +
       MSGDATA('u=roger') +
       REPLACE
```

3.2 Requester Channel

This section describes the necessary entries to enable the exit. The MQ Administrator will need to update 2 fields of the REQUESTOR Channel that the exit will be applied to.

3.2.1 Windows

On Windows, MSGEXIT and MSGDATA will contain the following values assuming a default install.

- MSGEXIT
  C:\Capitalware\MQSUI\mqsui(RE)

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE( RQSTR ) +
       TRPTYPE( TCP ) +
       MSGEXIT('C:\Capitalware\MQSUI\mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.2.2 Linux 32-bit

On Linux, MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT
  /var/mqm/exits/mqsui(RE)

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE( RQSTR ) +
       TRPTYPE( TCP ) +
       MSGEXIT('/var/mqm/exits/mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.2.3 Unix and Linux 64-bit

On Unix and Linux (excluding Linux x86), MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT
  /var/mqm/exits64/mqsui(RE)

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE( RQSTR ) +
       TRPTYPE( TCP ) +
       MSGEXIT('/var/mqm/exits64/mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.2.4 IBM i

On IBM i, MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT

  ```
  MQSUI     MQSUI
  ```

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE( RQSTR ) +
       TRPTYPE( TCP ) +
       MSGEXIT('MQSUI     MQSUI      ') +
       MSGDATA('u=roger') +
       REPLACE
```

3.3 Cluster Receiver Channel

This section describes the necessary entries to enable the exit. The MQ Administrator will need to update 2 fields of the CLUSRCVR Channel that the exit will be applied to.

3.3.1 Windows

On Windows, MSGEXIT and MSGDATA will contain the following values assuming a default install.

- MSGEXIT
  `C:\Capitalware\MQSUI\mqsui(RE)`

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE(CLUSRCVR) +
       TRPTYPE( TCP ) +
       MSGEXIT('C:\Capitalware\MQSUI\mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.3.2 Linux 32-bit

On Linux, MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT
  `/var/mqm/exits/mqsui(RE)`

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE(CLUSRCVR) +
       TRPTYPE( TCP ) +
       MSGEXIT('/var/mqm/exits/mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.3.3 Unix and Linux 64-bit

On Unix and Linux (excluding Linux x86), MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT
  `/var/mqm/exits64/mqsui(RE)`

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE(CLUSRCVR) +
       TRPTYPE( TCP ) +
       MSGEXIT('/var/mqm/exits64/mqsui(RE)') +
       MSGDATA('u=roger') +
       REPLACE
```

3.3.4 IBM i

On IBM i, MSGEXIT and MSGDATA will contain the following values assuming a default install:

- MSGEXIT is made up of 10 characters for program name (padded with blanks) followed by 10 characters for the LIBRARY name (padded with blanks).

  MQSUI   MQSUI

```
DEFINE CHANNEL( 'MQB1.TO.MQA1')  CHLTYPE(CLUSRCVR) +
       TRPTYPE( TCP ) +
       MSGEXIT('MQSUI      MQSUI      ') +
       MSGDATA('u=roger') +
       REPLACE
```

4 Configuring SVRCONN and CLNTCONN Channels

This section describes how to configure the exit.

For normal operation of the MQSUI solution, configuration parameters can be specified in the SENDDATA and RCVDATA attribute fields.

- SENDDATA
- RCVDATA

  U=roger
  Where 'roger' is the UserID to be written to each messages MQMD UserID field.

  D=Y
  Where 'Y' enables LogMode of Debug.

Note: Use a semicolon to separate the MQSUI parameters in the SENDDATA and RCVDATA attribute fields.

Note: Send and Receive Exit Data must NOT exceed 32 characters.

4.1 Server Connection Channel

This section describes the necessary entries to enable the server-side exit. The MQ Administrator will need to update 2 fields of the SVRCONN Channel that the server-side exit will be applied to.

4.1.1 Windows

On Windows, SENDEXIT, SENDDATA, RCVEXIT and RCVDATA will contain the following values assuming a default install.

- SENDEXIT
  `C:\Capitalware\MQSUI\mqsui(RE)`
- RCVEXIT
  `C:\Capitalware\MQSUI\mqsui(RE)`

```
DEFINE CHANNEL ('MQA1.APP.CH01') CHLTYPE(SVRCONN) +
       TRPTYPE(TCP) +
       RCVEXIT('C:\Capitalware\MQSUI\mqsui(RE)') +
       RCVDATA('u=roger') +
       REPLACE
```

4.1.2 Linux 32-bit

On Linux, SENDEXIT and SENDDATA will contain the following values assuming a default install:

- SENDEXIT
  `/var/mqm/exits/mqsui(RE)`
- RCVEXIT
  `/var/mqm/exits/mqsui(RE)`

```
DEFINE CHANNEL ('MQA1.APP.CH01') CHLTYPE(SVRCONN) +
       TRPTYPE(TCP) +
       RCVEXIT('/var/mqm/exits/mqsui(RE)') +
       RCVDATA('u=roger') +
       REPLACE
```

4.1.3 Unix and Linux 64-bit

On Unix and Linux (excluding Linux x86), SENDEXIT and SENDDATA will contain the following values assuming a default install:

- SENDEXIT
  /var/mqm/exits64/mqsui(RE)
- RCVEXIT
  /var/mqm/exits64/mqsui(RE)

```
DEFINE CHANNEL ('MQA1.APP.CH01') CHLTYPE(SVRCONN) +
       TRPTYPE(TCP) +
       RCVEXIT('/var/mqm/exits64/mqsui(RE)') +
       RCVDATA('u=roger') +
       REPLACE
```

4.1.4 IBM i

On IBM i, SENDEXIT and SENDDATA will contain the following values assuming a default install:

- SENDEXIT is made up of 10 characters for program name (padded with blanks) followed by 10 characters for the LIBRARY name (padded with blanks).

  MQSUI   MQSUI

- RCVEXIT is made up of 10 characters for program name (padded with blanks) followed by 10 characters for the LIBRARY name (padded with blanks).
- **MQSUI   MQSUI**

```
DEFINE CHANNEL ('MQA1.APP.CH01') CHLTYPE(SVRCONN) +
       TRPTYPE(TCP) +
       RCVEXIT('MQSUI      MQSUI      ') +
       RCVDATA('u=roger') +
       REPLACE
```

4.2 Client Connection Channel

This section describes the necessary entries to enable the server-side exit. The MQ Administrator will need to update 2 fields of the CLNTCONN Channel that the server-side exit will be applied to.

Note: Receive Exit Data must NOT exceed 32 characters.

4.2.1 Windows

On Windows, RCVEXIT and RCVDATA will contain the following values assuming a default install.

- RCVEXIT
  `C:\Capitalware\MQSUI\mqsui(RE)`

```
DEFINE CHANNEL ('MQA1.APP.CH01') CHLTYPE(CLNTCONN) +
       TRPTYPE(TCP) +
       CONNAME('127.0.0.1(1414)') +
       QMNAME('MQA1') +
       RCVEXIT('C:\Capitalware\MQSUI\mqsui(RE)') +
       RCVDATA('u=roger') +
       REPLACE
```

4.2.2 Linux 32-bit

On Linux, RCVEXIT and RCVDATA will contain the following values assuming a default install:

- RCVEXIT
  `/var/mqm/exits/mqsui(RE)`

```
DEFINE CHANNEL ('MQA1.APP.CH01') CHLTYPE(CLNTCONN) +
       TRPTYPE(TCP) +
       CONNAME('127.0.0.1(1414)') +
       QMNAME('MQA1') +
       RCVEXIT('/var/mqm/exits/mqsui(RE)') +
       RCVDATA('u=roger') +
       REPLACE
```

4.2.3 Unix and Linux 64-bit

On Unix and Linux (excluding Linux x86), RCVEXIT and RCVDATA will contain the following values assuming a default install:

- RCVEXIT
  /var/mqm/exits64/mqsui(RE)

```
DEFINE CHANNEL ('MQA1.APP.CH01') CHLTYPE(CLNTCONN) +
       TRPTYPE(TCP) +
       CONNAME('127.0.0.1(1414)') +
       QMNAME('MQA1') +
       RCVEXIT('/var/mqm/exits64/mqsui(RE)') +
       RCVDATA('u=roger') +
       REPLACE
```

4.2.4 IBM i

On IBM i, RCVEXIT and RCVDATA will contain the following values assuming a default install:

- RCVEXIT is made up of 10 characters for program name (padded with blanks) followed by 10 characters for the LIBRARY name (padded with blanks).

  MQSUI   MQSUI

```
DEFINE CHANNEL ('MQA1.APP.CH01') CHLTYPE(CLNTCONN) +
       TRPTYPE(TCP) +
       CONNAME('127.0.0.1(1414)') +
       QMNAME('MQA1') +
       RCVEXIT('MQSUI      MQSUI      ') +
       RCVDATA('u=roger') +
       REPLACE
```

4.3 Java based Applications

For Windows, Unix or Linux, set MQSUI_FILE JVM arguments.

4.3.1 Java Code Samples

This section describes how to code a Java application to invoke MQSUI.

4.3.1.1 Java Send Exit Code Sample

```
MQEnvironment.sendExit = new MQSUIJ();
```

4.3.1.2 Java Receive Exit Code Sample

```
MQEnvironment.receiveExit = new MQSUIJ();
```

4.3.2 Java Run-Time Settings

To use JVM arguments to specify the IniFile that will contain the MQSUI parameters, do the following:

- Add the following JVM argument to your java command-line parameters to specify an IniFile:

  On Windows:
  java -DMQSUI_FILE=C:\Capitalware\MQSUI\mqsuij.ini com.acme.run.Thing

  On Unix / Linux:
  java -DMQSUI_FILE=/home/user/mqsuij.ini com.acme.run.Thing

4.4 Configuring MQSUIJ for use in J2EE Application Server

This section describes the necessary steps to enable Security Exits in a J2EE Application Server like IBM's WebSphere Application Server or BEA's WebLogic Server.

4.4.1 Batch or Quiet mode for J2EE based applications

To run in batch or quiet mode, the user can explicitly set the UserId value in the channel's SendExitInit / ReceiveExitInit field or specify a file in the SendExitInit / ReceiveExitInit field.

4.4.1.1 Updating Application Server's JVM Classpath

Windows:
The JAR file is located at (assuming a default install of **C:\Capitalware\MQSUI**):

```
SET CLASSPATH=C:\Capitalware\MQSUI\MQSUIJ.jar;%CLASSPATH%
```

Unix and Linux (32-bit):
The JAR file is located at (assuming a default install of **/var/mqm/exits/**):

```
export CLASSPATH=/var/mqm/exits/MQSUIJ.jar;%CLASSPATH%
```

Unix and Linux (64-bit):
The JAR file is located at (assuming a default install of **/var/mqm/exits64/**):

```
export CLASSPATH=/var/mqm/exits64/MQSUIJ.jar:$CLASSPATH
```

4.4.1.2 Updating Application's JMS binding file

Use IBM MQ's JMSAdmin command to define or alter a QCF (QueueConnectionFactory) or TCF (TopicConnectionFactory). The client-side security exit also works with the XA versions of QCF and TCF (i.e. XAQCF and XATCF). In the SendExitInit / ReceiveExitInit field, include the UserId value as follows:

```
define tcf(tcfClient) qmgr(MY.QMGR)
channel(SYSTEM.DEF.SVRCONN) hostname(MYHOSTNAME) port(1414)
transport(CLIENT) RCVEXIT(biz.capitalware.mqsui.MQSUIJ)
RCVEXITINIT(u=roger)

or

define qcf(qcfClient) qmgr(MY.QMGR)
channel(SYSTEM.DEF.SVRCONN) hostname(MYHOSTNAME) port(1414)
transport(CLIENT) RCVEXIT(biz.capitalware.mqsui.MQSUIJ)
RCVEXITINIT(u=roger)
```

5 IniFile Keywords

5.1 Logging

This section describes the necessary entries to enable MQSUI to write log information. To enable and control logging, you need 3 keywords in the IniFile:

- **LogMode** specifies what type of logging the user wishes to have. LogMode supports 4 values [Q / N / V / D] where Q is Quiet, N is Normal, V is Verbose and D is Debug. The default value is Q.

- **LogFile** LogFile specifies the location of the log file. The default is as follows:

  For Windows:
  ```
  LogFile=C:\Capitalware\MQSUI\mqsui.log
  ```
  For IBM MQ 32-bit on Linux:
  ```
  LogFile=/var/mqm/exits/mqsui.log
  ```
  For IBM MQ 64-bit on Unix and Linux:
  ```
  LogFile=/var/mqm/exits64/mqsui.log
  ```
  For IBM MQ on IBM i:
  ```
  LogFile=/QIBM/UserData/mqm/mqsui/mqsui.log
  ```

  Token Replacement for LogFile keyword:
  - **%QM%** - Substitutes the name of the queue manager
  - **%CHL%** - Substitutes the name of the channel
  - **%UID%** - Substitutes the UserID
  - **%PID%** - Substitutes the Process ID
  - **%TID%** - Substitutes the Thread ID

- **RotateLogDaily** specifies whether or not the log files will be rotated on a daily basis. A Y value for ' RotateLogDaily' will activate this feature; otherwise, the log files will left as is. The default value is Y.

  In other words, it is possible to keep up to 9 backup log files. The first connection request after midnight (and not at midnight) will cause it to roll/rotate the log files. If there are already 9 backup log files, the ninth backup log file will be deleted and 8 becomes 9, 7 becomes 8, etc...

- **BackupLogFileCount** specifies the number of backup log files that should be kept by MQSUI. The default value is 9. This keyword is only used if RotateLogDaily is set to 'Y'.

5.2 UserID

Specifying a UserID in the IniFile is optional.

- **UserID** specifies the UserID to be set in the message's MQMD UserId field.

6 Miscellaneous

This section describes the extra files that were included to help the user get MQSUI up and running in a very quick manner.

6.1 Windows

Sample IniFile

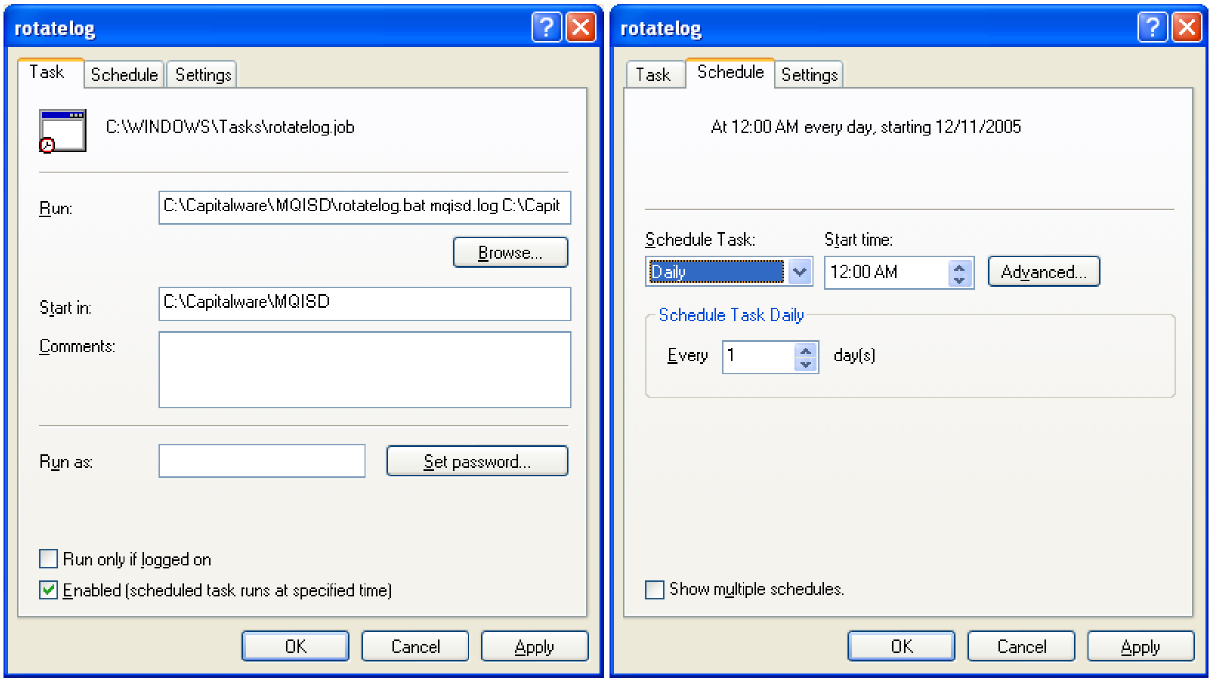
The '***mqsui.ini***' file is a basic MQSUI IniFile. It has the standard IniFile parameters that the user may need to use or update. The '***mqsui.ini.readme***' file is a plain text help file with a description of the parameters.

Sample MQSC scripts

The '***mqsui.sample.mqsc***' file is a sample MQSC script to update the 2 system defined channels with the MQSUI exit information.

Rotate log script

The '***rotatelog.bat***' file is a Windows batch script to rotate (backup) the mqsui.log file. Actually, it is generic in implementation; hence, it can be used to rotate any log file that the user wishes to be rotated. The batch script requires 2 parameters: log file name and the directory of log file.



C:\WINDOWS\Tasks\rotatelog.job

Run: C:\Capitalware\MQISD\rotatelog.bat mqisd.log C:\Capit

Start in: C:\Capitalware\MQISD

Comments:

Run as:

☐ Run only if logged on

☑ Enabled (scheduled task runs at specified time)

At 12:00 AM every day, starting 12/11/2005

Schedule Task: Daily

Start time: 12:00 AM

Schedule Task Daily

Every 1 day(s)

☐ Show multiple schedules.

6.2 Unix and Linux

Sample IniFile
The '***mqsui.ini***' file is a basic MQSUI IniFile. It has the standard IniFile parameters that the user may need to use or update. The '***mqsui.ini.readme***' file is a plain text help file with a description of the parameters.

Sample MQSC scripts
The '***mqsui.sample.MQA1.mqsc***' and '***mqsui.sample.MQB1.mqsc***' files are sample MQSC scripts to be used as reference channels with the MQSUI exit.

Rotate log script
The '***rotatelog.sh***' file is a Unix / Linux shell script to rotate (backup) the mqsui.log file. Actually, it is generic in implementation; hence, it can be used to rotate any log file that the user wishes to be rotated. The shell script requires 2 parameters: log file name and the directory of log file.

Sample daily CRON entry for IBM MQ 32-bit on Unix and Linux:

```
0 0 * * * /var/mqm/exits/rotatelog.sh mqsui.log /var/mqm/exits/  > /tmp/mqsui.log.run 2 > &1
```

Sample daily CRON entry for IBM MQ 64-bit on Unix and Linux:

```
0 0 * * * /var/mqm/exits64/rotatelog.sh mqsui.log /var/mqm/exits64/  > /tmp/mqsui.log.run 2 > &1
```

6.3 IBM i

Sample IniFile
The '***mqsui.ini***' file is a basic MQSUI IniFile. It has the standard IniFile parameters that the user may need to use or update. The '***mqsui.ini.readme***' file is a plain text help file with a description of the parameters.

Sample MQSC scripts
The '***mqsui.sample.mqsc***' file is a sample MQSC script to update the 2 system defined channels with the MQSUI security exit information.

6.4 Server-side Log File

To verify that the process flow was successful, you can view the log file for the events that are generated.

7 Appendix A – mqsui.ini file

The sample IniFile below is the mqsui.ini file supplied for Windows. The IniFile supports the following keywords and their values:

```
LogMode=N
LogFile=C:\Capitalware\MQSUI\mqsui.log
```

Note: Keywords are case sensitive.

Keyword	Description of Server-side keywords
BackupLogFileCount	**BackupLogFileCount** specifies the number of backup logfiles that the security exit will be keeping. The default value is 9. e.g. BackupLogFileCount=9
LogFile	**LogFile** specifies the location of the log file. The default is as follows: For Windows: LogFile=C:\Capitalware\MQSUI\mqsui.log For IBM MQ 32-bit on Linux: LogFile=/var/mqm/exits/mqsui.log For IBM MQ 64-bit on Unix and Linux: LogFile=/var/mqm/exits64/mqsui.log For IBM MQ on IBM i: LogFile=/QIBM/UserData/mqm/mqsui/mqsui.log
LogMode	**LogMode** specifies what type of logging the user wishes to have. LogMode supports 4 values [Q / N / V / D] where Q is Quiet, N is Normal, V is Verbose and D is Debug. The default value is Q. e.g. LogMode=Q
RotateLogDaily	**RotateLogDaily** specifies whether or not daily log file rotation should take place. RotateLogDaily supports 2 values [Y / N]. The default value is Y. e.g. RotateLogDaily=Y

Keyword	Description of Server-side keywords
UserID	**UserID** specifies the UserID to be set in the message's MQMD UserId field. e.g. UserID=fred

8 Appendix B – MQSUI Upgrade Procedures

To upgrade an existing installation of MQSUI from an older version to a newer version, do please do the following in the appropriate section below.

8.1.1 Windows Upgrade

- Stop all of the channels using the MQSUI exit or completely stop the queue manager.
- Backup all MQSUI IniFiles in the MQSUI install directory
- If MQSUI was installed using the Windows Installer then
  - Click the ***Start*** -> ***All Programs*** -> ***Control Panel*** -> ***Add or Remove Programs***, select MQSUI from the list and click the ***Remove*** button then follow the prompts to remove it
  - Run the ***mqsui-setup.exe*** file from the ***Windows*** directory to install the new version
- Otherwise copy the following files (latest version) to the MQSUI install directory:
    - mqsui.dll
    - AddRegistryEntries.bat
    - mqsui.reg
    - rotatelog.bat
- Run AddRegistryEntries.bat batch file
- Restore the MQSUI IniFiles if they were altered / deleted.
- Start all of the channels using the MQSUI server-side security exit or restart the queue manager if it was previously stopped.

8.1.2 Linux 32-bit Upgrade

- Login under the mqm account
- Stop all of the channels using the MQSUI exit or completely stop the queue manager.
- Backup all MQSUI IniFiles in the MQSUI install directory
- Copy the appropriate tar file to the ***/var/mqm/exits/*** directory
- Un-tar the contents of the tar file.
  i.e. For AIX, do the following command:
  ```
  tar -xvf  mqsui_linux.tar
  ```
- Run the script as follows:
  ```
  ./setce.sh
  ```
- Restore the MQSUI IniFiles if they were altered / deleted.
- Delete the MQSUI tar file
- Start all of the channels using the MQSUI exit or restart the queue manager if it was previously stopped.

8.1.3 Unix and Linux 64-bit Upgrade

- Stop all of the channels using the MQSUI exit or completely stop the queue manager.
- Backup all MQSUI IniFiles in the MQSUI install directory
- Copy the appropriate tar file to the ***/var/mqm/exits64/*** directory
- Un-tar the contents of the tar file.
  i.e. For AIX, do the following command:
  ```
     tar -xvf  mqsui_aix.tar
  ```
- Run the script as follows:
  ```
    ./setce.sh
  ```
- Restore the MQSUI IniFiles if they were altered / deleted.
- Delete the MQSUI tar file
- Start all of the channels using the MQSUI exit or restart the queue manager if it was previously stopped.

8.1.4 IBM i Upgrade

- Stop all of the channels using the MQSUI exit or completely stop the queue manager.
- Backup all MQSUI IniFiles in the MQSUI install directory
- ftp the IBM i files to the IBM i server as follows:

```
ftp -s:mqsui_iseries.ftp  iseries_hostname
```

```
your-IBM i-userid
your-IBM i-password

binary
cd QGPL
put mqsui.savf

quote SITE NAMEFMT 1

cd /QIBM/UserData/mqm/
put mqsui_iseries.tar
quit
```

- Log onto the target IBM i server and do the following commands:

```
RSTLIB SAVLIB(MQSUI) DEV(*SAVF) SAVF(QGPL/MQSUI)
CLRSAVF FILE(QGPL/MQSUI)
CHGOBJOWN  OBJ(MQSUI) OBJTYPE(*LIB)  NEWOWN(QMQM)
qsh
cd /QIBM/UserData/mqm/
tar -xvf mqsui_iseries.tar
chown -R QMQM mqsui
rm mqsui_iseries.tar
```

- Restore the MQSUI IniFiles if they were altered / deleted.
- Start all of the channels using the MQSUI exit or restart the queue manager if it was previously stopped.

9 Appendix C – Capitalware Product Display Version

MQSUI includes a program to display the product version number. The command to display the product version number is:

`cwdspver`

9.1 Examples

9.1.1 Windows

To use the cwdspver program on Windows, open a Command prompt and change the directory to `C:\Capitalware\MQSUI\` and type the following:

```
cwdspver.exe
```

9.1.2 Linux 32-bit

To use the cwdspver program on Unix/Linux for MQ 32-bit, open a shell prompt and change directory to `/var/mqm/exits/` and type the following:

```
./cwdspver
```

9.1.3 Unix and Linux 64-bit

To use the cwdspver program on Unix/Linux for MQ 64-bit, open a shell prompt and change directory to `/var/mqm/exits64/` and type the following:

```
./cwdspver
```

9.1.4 IBM i

To use the cwdspver program on IBM i for MQ, issue the following command on the prompt:

```
CALL MQSUI/CWDSPVER
```

10 Appendix D – Support

The support for MQ Set UserID can be found at the following location:

Online Help Desk Ticketing System at
www.capitalware.com/phpst/

By email at:
support@capitalware.com

By regular mail at:

Capitalware Inc.
Attn: MQSUI Support
Unit 11, 1673 Richmond Street, PMB524
London, Ontario N6G2N3
Canada

11 Appendix E – Summary of Changes

- MQ Set UserID v1.0.5
  - Enhanced the code for dumping the pointers passed into exit.
  - Fixed an issue in the subroutine that removes trailing blanks
  - Fixed an issue with default exit path

- MQ Set UserID v1.0.4
  - Added code to append trailing slash for ExitPath if it is missing.
  - Tuned the code that is called on entry
  - Tuned the logging code

- MQ Set UserID v1.0.3
  - Fixed an issue in the logging framework where a constant was being modified.

- MQ Set UserID v1.0.2
  - Enhanced logging - the LogFile keyword now supports the following tokens: %QM%, %CHL%, %UID%, %PID% & %TID%

- MQ Set UserID v1.0.1
  - Fixed an issue on Windows with freeing environment variable memory (error with FreeEnvironmentStrings Windows API call)
  - Fixed an issue with using "size_t" variable type when it should have been "int"

- MQ Set UserID v1.0.0
  - Initial release.

12 Appendix F – License Agreement

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

13 Appendix G – Notices

Trademarks:

AIX, IBM, MQSeries, OS/2 Warp, OS/400, iSeries, MVS, OS/390, WebSphere, IBM MQ and z/OS are trademarks of International Business Machines Corporation.

HP-UX is a trademark of Hewlett-Packard Company.

Intel is a registered trademark of Intel Corporation.

Java, J2SE, J2EE, Sun and Solaris are trademarks of Sun Microsystems Inc.

Linux is a trademark of Linus Torvalds.

Mac OS X is a trademark of Apple Computer Inc.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation.

UNIX is a registered trademark of the Open Group.

WebLogic is a trademark of BEA Systems Inc.