MQ Message Encryption Overview

Capitalware Inc.

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 Message Encryption (MQME) provides encryption for MQ message data while it resides in a queue or topic and in the MQ logs (i.e. all data at rest). In cryptography, encryption is the process of transforming information into an unreadable form (encrypted data). Decryption is the reverse process. It makes the encrypted information readable again. Only those with the key (PassPhrase) can successfully decrypt the encrypted data. MQME uses Advanced Encryption Standard (AES) to encrypt the data. AES is a data encryption scheme, adopted by the US government, that uses three different key sizes (128-bit, 192-bit, and 256-bit).

One of the features that MQME offers is the ability to control who accesses protected queues/topics. This control is obtained through the use of UserId grouping. MQME can query the local OS group or a group file. Group files are implemented in a similar manner to the way they are implemented in Unix and Linux (i.e. **/etc/group** file). Normally, the 'mqm', 'QMQM' or 'MUSR_MQADMIN' MQ UserIds or any UserId in the 'mqm' group get full access to all messages in all queues/topics. For queues/topics protected by MQME, those privileged UserIds do **not** get access to the messages in the protected queues/topics unless they are explicitly added to the authorized list of users or groups.

Another feature of MQME is its ability to generate and validate the message via a digital signature. MQME uses the SHA-2 to create a cryptographic hash function (digital signature) for the message data. The digital signature provides verification that the message data has not been altered.

MQME is an MQ API Exit that operates with IBM MQ v7.1, v7.5, v8.0, v9.0, v9.1 and v9.2 in Windows, Unix, IBM i (OS/400) and Linux platforms.

On AIX, HP-UX, Linux, Solaris and Windows, MQME 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

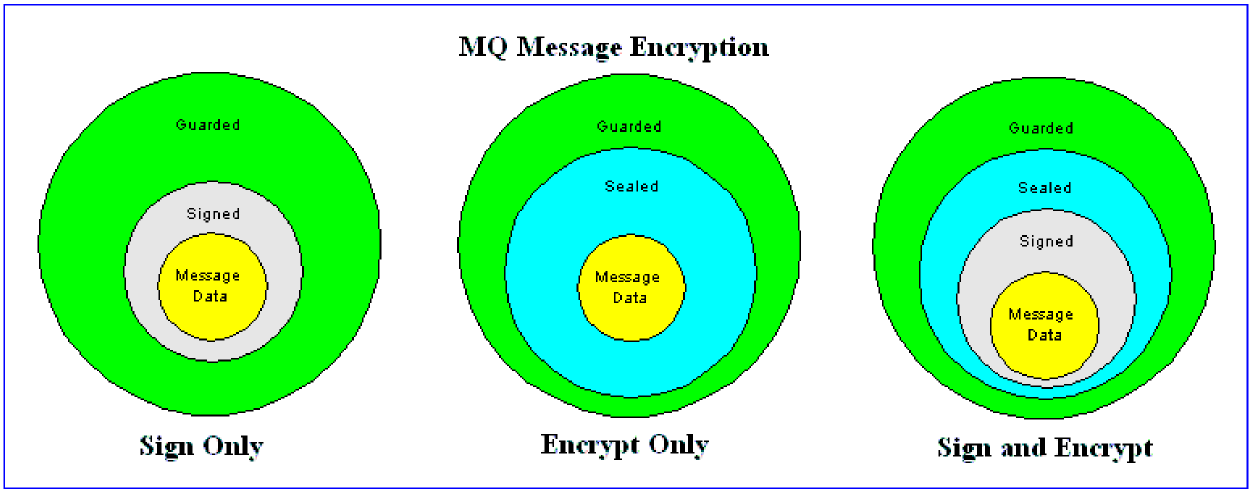
MQME is an MQ API Exit. The MQ API Exit is available in 3 forms:

- Windows DLL
- Shared library for AIX, HP-UX, Linux, and Solaris.
- IBM i exit module

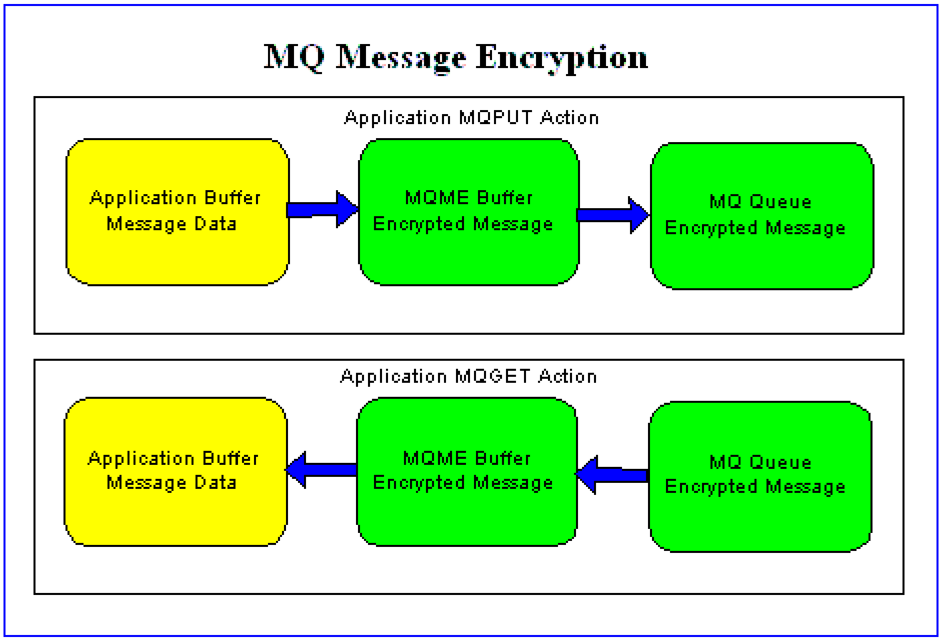
The major features of MQME are as follows:

- No application changes required
- All message data written to a selected queue and/or topic will be encrypted (nothing missed or forgotten)
- Secure encryption/decryption methodology using AES with 128, 192 or 256-bit keys
- Uses the SHA-2 to create a cryptographic hash function (digital signature)
- No application changes required
- Group authority checking against the local OS groups or a group file
- Standard MQ feature, GET-with-Convert, is supported
- Provides high-level logging capability for encryption/decryption processing

1.3 Message Diagram (Logical View)



1.4 Message Flow (Logical View)


MQ Message Encryption
Application MQPUT Action
Application Buffer Message Data
MQME Buffer Encrypted Message
MQ Queue Encrypted Message
Application MQGET Action
Application Buffer Message Data
MQME Buffer Encrypted Message
MQ Queue Encrypted Message

1.5 Prerequisites

This section provides the minimum supported software levels. These prerequisites apply to server-side installations of MQ Message Encryption.

1.5.1 Operating System

MQ Message Encryption can be installed on any of the following supported servers:

1.5.1.1 IBM AIX

- IBM AIX 6L version 6.1 or higher

1.5.1.2 HP-UX IA64

1. HP-UX v11.23 or higher

1.5.1.3 IBM i (OS/400)

- IBM i V6R1 or higher

1.5.1.4 Linux x86

1. Red Hat Enterprise Linux v5, v6, v7, v8
2. SUSE Linux Enterprise Server v11, v12, v15

1.5.1.5 Linux x86_64 (64-bit)

3. Red Hat Enterprise Linux v5, v6, v7, v8
4. SUSE Linux Enterprise Server v11, v12, v15

1.5.1.6 Linux on POWER

5. Red Hat Enterprise Linux v5, v6, v7, v8
6. SUSE Linux Enterprise Server v11, v12, v15

1.5.1.7 Linux on zSeries (64-bit)

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

1.5.1.8 Raspberry Pi (Linux ARM 32-bit)

7. Raspberry Pi OS v9 or higher

1.5.1.9 Sun Solaris

- Solaris SPARC v10 & v11
- Solaris x86_64 v10 & v11

1.5.1.10 Windows

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

1.5.2 IBM MQ

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

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

1.5.3 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

1.5.4 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