

Integrating the IBM MQ Appliance into your IBM MQ Infrastructure

Neil Casey

Andy Emmett

Rufus Russell



WebSphere



International Technical Support Organization

**Integrating the IBM MQ Appliance into your IBM MQ
Infrastructure**

November 2015

Note: Before using this information and the product it supports, read the information in “Notices” on page ix.

First Edition (November 2015)

This edition applies to the IBM MQ Appliance M2000.

© Copyright International Business Machines Corporation 2015. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	ix
Trademarks	x
IBM Redbooks promotions	xi
Preface	xiii
Authors	xiii
Now you can become a published author, too!	xv
Comments welcome	xv
Stay connected to IBM Redbooks	xv
Part 1. Overview	1
Chapter 1. Introduction	1
1.1 IBM MQ Appliance M2000	2
1.1.1 Advantages of an appliance	2
1.1.2 Business use cases	3
1.2 Features and benefits	4
1.2.1 High availability	4
1.2.2 Easy administration	5
1.2.3 Appliance options and upgrades	6
1.2.4 Support for hardware and software	7
1.2.5 Security	7
1.3 About this book	8
1.3.1 Intended audience	8
1.3.2 Covered topics	8
1.3.3 Topics not covered	8
Chapter 2. Business scenario overview	11
2.1 Current configuration	12
2.1.1 Queue manager cluster configuration	12
2.1.2 Application flow	13
2.2 Target configuration	14
2.2.1 New full repository queue managers	14
2.2.2 HA gateway	16
2.2.3 Central authentication and authorization	16
2.2.4 Adding non-HA queue managers to the cluster	17
2.2.5 Cluster configuration	17
2.2.6 Application flow	18
Part 2. Using the IBM MQ Appliance	19
Chapter 3. Planning	21
3.1 Number of appliances	22
3.2 Standards	22
3.2.1 Naming standards	22
3.2.2 Network interface standards	23
3.2.3 Interface name standards	23
3.2.4 Authentication and authorization standards	24
3.2.5 Logging standards	25

3.2.6 Coded Character Set ID standards	25
3.3 Location	26
3.4 Other equipment needed	26
3.4.1 Racks	26
3.4.2 Support servers	27
3.4.3 Cables	27
3.5 Installation prerequisites	28
3.6 Information needed before initializing the IBM MQ Appliance	30
3.7 Remote access to serial port	30
3.8 More information for complete initialization	31
Chapter 4. Installing the IBM MQ Appliance	33
4.1 Rack choice	34
4.2 Power	35
4.3 Environment	35
Chapter 5. Initial appliance configuration	37
5.1 Preparing a notebook to connect to the serial console port	38
5.2 Connecting the USB serial adapter cable	38
5.3 Running a terminal emulation application	38
5.3.1 Using a Windows system to connect	39
5.3.2 Using Mac OS X to connect	39
5.4 Logging in for the first time	39
5.5 Installation wizard	40
5.5.1 Running the installation wizard	40
5.5.2 Configuring the mgt0 network interface	41
5.5.3 Configuring Network Services	41
5.5.4 Configuring a unique system identifier	41
5.5.5 Configuring remote management access	42
5.5.6 Configuring a spare admin account	43
5.5.7 Saving the configuration	43
5.6 Accepting the appliance license terms and conditions	43
5.6.1 Connecting to the appliance web UI with a browser	44
5.6.2 Logging in and accepting the license agreement	47
5.7 Implementing IPMI	50
5.7.1 Creating IPMI objects	50
5.7.2 IPMI usage examples	51
5.8 Next steps	54
Chapter 6. Appliance administration	55
6.1 IBM MQ Appliance user model	56
6.1.1 Appliance administrators	56
6.1.2 IBM MQ users	58
6.2 CLI administration	59
6.3 IBM MQ Appliance Web UI administration	64
6.3.1 Securing the appliance web UI	64
6.3.2 Managing the IBM MQ Console	71
6.4 Network configuration	75
6.4.1 Configure network interfaces	75
6.4.2 Configuring names for interfaces	94
6.4.3 External services used by the IBM MQ Appliance	96
6.4.4 Appliance hosted Network Services	101
6.5 Appliance file system	103
6.6 Firmware upgrades	105

Chapter 7. Creating queue managers	107
7.1 Preparing to create queue managers	108
7.1.1 User and group configuration	108
7.1.2 Log file size	109
7.1.3 Queue file size	109
7.1.4 File system size	109
7.2 Using the command line to create queue managers	110
7.2.1 Default values	110
7.3 Using the IBM MQ Console to create queue managers	112
7.4 Queue manager security	112
7.4.1 LDAP considerations	113
7.4.2 Connecting the queue manager to the LDAP	113
7.5 Customizing the queue manager	118
7.5.1 Limit max channels	118
7.5.2 Configuring the number of logs	119
7.5.3 Configuring channel behavior	119
7.5.4 Configuring TLS certificate checking	120
7.5.5 Binding the listener to the data interface	120
Chapter 8. IBM MQ object security	123
8.1 Queue manager security	124
8.2 Object security	124
8.2.1 Enable remote administration	124
8.2.2 Connecting IBM MQ Explorer	129
8.2.3 Connecting runmqsc client	131
8.2.4 Enabling limited administration rights	133
8.2.5 Granting security rights to users	133
8.3 Summary	135
Chapter 9. IBM MQ channel security	137
9.1 Security policy	138
9.2 SSL and TLS	138
9.3 Replacing exits	138
9.3.1 Security exits	139
9.3.2 Channel auto-definition exits	139
9.3.3 Other exits	140
9.4 User Identity changes	140
9.5 Purpose of MCAUSERS	141
9.6 Authenticating channel partners	141
9.6.1 Message channels	141
9.6.2 MQI channels	142
9.7 Assigning MCAUSERS on the IBM MQ Appliance	142
9.7.1 MCAUSER field on the channel definition	143
9.7.2 MCAUSER attribute of a channel authentication record	143
9.7.3 MCAUSER assigned by using ADOPTCTX	145
9.7.4 Granting rights to channel MCA users	146
9.7.5 Creating and managing MCA users	147
9.8 Summary	148
Chapter 10. High availability	149
10.1 Disaster recovery and high availability	150
10.2 Historical IBM MQ HA	150
10.3 HA groups	151
10.4 Preparing for HA	153

10.5	Creating an HA group	153
10.6	Creating an HA queue manager	154
10.7	Setting the preferred appliance for a queue manager.	156
10.8	Suspending and resuming an appliance	156
10.9	Applying a fix pack to an HA group	160
10.10	HA scenario.	160
10.10.1	Defining a cluster receiver channel.	161
10.10.2	Creating a host alias for a listener.	161
10.10.3	Defining a listener	162
Chapter 11.	Application changes	163
11.1	Just another queue manager	164
11.2	Client scenarios.	164
11.2.1	Synchronous requester application.	165
11.2.2	Server application (query only)	166
11.2.3	Server application (reliable update)	167
11.3	Summary.	169
Chapter 12.	Support for the IBM MQ Appliance	171
12.1	Scope of appliance support	172
12.1.1	Removing user exits	172
12.2	Fix process	173
12.2.1	Software problem	173
12.2.2	Hardware problem	173
12.3	Support for the IBM MQ installation on the appliance	173
12.3.1	New tools for basic troubleshooting	173
12.3.2	Differences in familiar commands and familiar file locations	174
12.3.3	Location and names of IBM MQ error logs on the IBM MQ Appliance	175
12.4	Appliance support	176
12.4.1	Appliance error logs	176
12.4.2	Copy command.	179
12.5	Example support scenario	180
Part 3.	Appendixes	183
Appendix A.	IBM MQ cryptographic changes	185
	Comparison of commands	186
	Examples of using the new keystore commands	187
	Copying certificate files to an appliance	187
	Adding trusted certificates.	188
	Creating a certificate signing request	188
	Copying CSR to an external server.	189
	Listing certificate signing requests	190
	Showing details of a certificate signing request	190
	Receiving signed certificate	191
	Deleting a certificate signing request	191
	Creating a self-signed certificate.	192
	Listing the certificates for a queue manager.	192
	Showing the details of a certificate	192
	Deleting a certificate	194
Appendix B.	Transcript of IBM MQ Appliance firmware upgrade.	195
Appendix C.	Transcript of appliance initialization	199

Appendix D. Commands to enable an LDAP authenticated administrator	209
Related publications	213

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®

DataPower®

FFST™


HACMP™

IBM®

Redbooks®

Redguide™

Redpaper™

Redbooks (logo) ®

Tivoli®

WebSphere®

The following terms are trademarks of other companies:

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Find and read thousands of IBM Redbooks publications

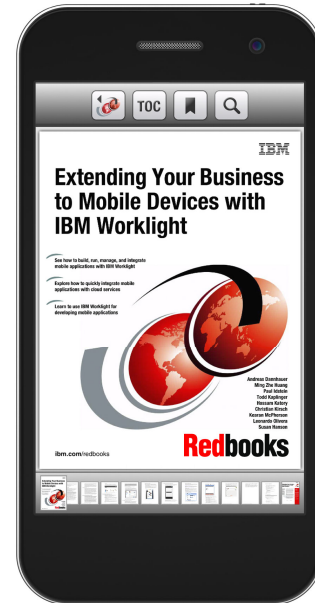
- ▶ Search, bookmark, save and organize favorites
- ▶ Get up-to-the-minute Redbooks news and announcements
- ▶ Link to the latest Redbooks blogs and videos

Get the latest version of the Redbooks Mobile App



Download
Now

iOS



Promote your business in an IBM Redbooks publication

Place a Sponsorship Promotion in an IBM® Redbooks® publication, featuring your business or solution with a link to your web site.

Qualified IBM Business Partners may place a full page promotion in the most popular Redbooks publications. Imagine the power of being seen by users who download millions of Redbooks publications each year!



ibm.com/Redbooks

About Redbooks → Business Partner Programs

THIS PAGE INTENTIONALLY LEFT BLANK

Preface

This IBM® Redbooks® publication describes the IBM MQ Appliance M2000, an application connectivity option that combines secure, reliable IBM MQ messaging with the simplicity and low overall costs of a hardware appliance.

This book presents underlying concepts and practical advice for integrating the IBM MQ Appliance M2000 into an IBM MQ infrastructure. Therefore, it is aimed at enterprises that are considering a possible first use of IBM MQ and the IBM MQ Appliance M2000 and those that already identified the appliance as a logical addition to their messaging environment.

Details about new functionality and changes in approaches to application messaging are also described. The authors' goal is to help readers make informed design and implementation decisions so that the users can successfully integrate the IBM MQ Appliance M2000 into their environments.

A broad understanding of enterprise messaging is required to fully comprehend the details that are provided in this book. Readers are assumed to have at least some familiarity and experience with complimentary IBM messaging products.

Authors

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.



Neil Casey is a Senior Consultant with Syntegrity Solutions, based in Melbourne, Australia. He has a Bachelor of Applied Science in Computer Science from the Royal Melbourne Institute of Technology. He primarily works with customers to help them use IBM MQ and IBM WebSphere® DataPower® systems efficiently and effectively. In 2012, co-wrote the IBM RedBooks publication *Secure Messaging Scenarios with WebSphere MQ*, which was based on WebSphere MQ v7.5.



Andy Emmett is a Software Engineer at the IBM Hursley Laboratory in the UK. He has been with IBM for over 15 years, working with MQ products as far back as WebSphere MQ 5.1. Primarily working in the L3 support organization, Andy is the recognized subject matter expert for the Queue Manager Clusters feature of the product. After leaving education, Andy became a senior programmer, writing programs for machining and manufacture of complex 3, 4, and 5 dimensional geometrical components. Before joining IBM, Andy worked as a consultant who developed in various fields, including Computer Aided Design and Computer Aided Manufacture (CAD/CAM). Currently, Andy is working with the MQ development team for the MQ Appliance.



Rufus Russell is a Software Engineer working for IBM in Hursley, UK. He holds a masters degree in Physics from Durham University. Since starting with IBM in January 2014, he has worked as an MQ developer, primarily in the IBM MQ Appliance team.

Syntegrity Solutions: Syntegrity Solutions is a specialist IT consultancy that provides digital innovation enablement. Founded in Melbourne in 2010, Syntegrity Solutions focuses on providing high-quality integration, API management, identity management, and access management expertise. Our background is in designing and implementing enterprise solutions that use the IBM WebSphere and Tivoli® suites. With our current focus on digital innovation enablement, Syntegrity Solutions enables our clients to quickly and securely use and modify their digital channels to provide enhanced customer engagement. Syntegrity Solutions clients include major banks, regional banks, insurance companies, federal governments, state governments, and education providers. Syntegrity Solutions is the premier DataPower consultancy in Australia and New Zealand. For more information, see this website:

<http://www.syntegrity.com.au>

This project was led by the following people:

- ▶ Martin Keen
- ▶ Shawn Tooley

Thanks to the following people for their contributions to this project:

- ▶ Steve Aiken
- ▶ Anthony Beardsmore
- ▶ Dave Lane
- ▶ Trevor Lobban
- ▶ Liam O'Neil
- ▶ Anja Jessica Paessler
- ▶ Graham Richards
- ▶ Miguel Rodriguez
- ▶ Jon Rumsey
- ▶ Andrew Schofield
- ▶ Geoff Winn

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience by using leading-edge technologies. Your efforts help to increase product acceptance and customer satisfaction as you expand your network of technical contacts and relationships. Residencies run 2 - 6 weeks in length and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

ibm.com/redbooks

- Send your comments in an email to:

redbooks@us.ibm.com

- Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on Facebook:

<http://www.facebook.com/IBMRedbooks>

- Follow us on Twitter:

<http://twitter.com/ibmredbooks>

- Look for us on LinkedIn:

<http://www.linkedin.com/groups?home=&gid=2130806>

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>

- Stay current on recent Redbooks publications with RSS Feeds:

<http://www.redbooks.ibm.com/rss.html>



Part 1

Overview

This part includes the following chapters:

- ▶ Chapter 1, “Introduction” on page 1
- ▶ Chapter 2, “Business scenario overview” on page 11



Introduction

This chapter provides an overview of the IBM MQ Appliance M2000, a new application connectivity option that combines secure, reliable IBM MQ messaging with the simplicity and low overall costs of a hardware appliance.

The IBM MQ Appliance M2000 is available in the following options:

- ▶ IBM MQ Appliance M2000A for large, demanding enterprise workloads
- ▶ IBM MQ Appliance M2000B for smaller workloads that require less processing capability

Later chapters describe the intricacies of the appliance and how to use it. Here, we address the product, why IBM introduced it, and the many benefits it offers enterprises that are faced with growing demand for secure, reliable application messaging.

The chapter includes the following sections:

- ▶ 1.1, “IBM MQ Appliance M2000” on page 2
- ▶ 1.2, “Features and benefits” on page 4
- ▶ 1.3, “About this book” on page 8

1.1 IBM MQ Appliance M2000

The concept behind the IBM MQ Appliance M2000 is simple: Combine the customer-proven scalability and security of IBM MQ messaging software with the simplicity, ease-of-use, and low total costs of a hardware appliance. Enterprises have long used IBM MQ messaging to integrate applications, systems, and services reliably and securely. Now, with the IBM MQ Appliance M2000, IBM adds a state-of-the-art hardware option that is fast to deploy and uses fewer administrative and infrastructure resources than running multiple messaging servers.

Messaging servers are only part of the cost of messaging integration. There also is the expense of configuring and maintaining the servers and software, and for many enterprises, the challenge of extending the infrastructure to multiple, far-flung geographic locations. Also, by its nature, messaging infrastructure must be highly available and responsive to enormous fluctuations in demand.

Therefore, the industry needs a new approach to application connectivity, one that is fast and easy to deploy, simple to maintain, reliably secure, and cost-effective. With the IBM MQ Appliance M2000 (see Figure 1-1), IBM offers the messaging performance of IBM MQ with the convenience and costs savings of a robust physical component.



Figure 1-1 IBM MQ Appliance M2000

1.1.1 Advantages of an appliance

The foundation of the IBM MQ Appliance M2000 is in its predecessor products, including IBM MQ version 8.0, which is the most recent update to the messaging middleware offering. Yet, whereas IBM MQ version 8.0 and IBM WebSphere MQ 7.5 are software solutions, the IBM MQ Appliance M2000 is a hardware platform that is purpose-built for messaging.

Compared to the software, the appliance performs the same universal messaging functions that enable applications, systems, and services to connect and exchange information securely, reliably, and rapidly.

However, as hardware, the appliance enables the following features and can help reduce messaging overhead:

- ▶ Queue managers that behave the same whether they are deployed on the appliance or are running as software on other servers. They can participate in clusters and exchange messages with other queue managers or IBM MQ clients.
- ▶ A new, high availability configuration that consists of a *pair* of appliances that mirror messages, therefore, if the primary appliance fails, the other can take over seamlessly.
- ▶ Lock-down features that aid in appliance security and maintenance. No extra software can be installed, including user applications and user exits.

The appliance is used and managed as a messaging hub (see Figure 1-2), with applications relying on client connections to the appliance (or other IBM MQ queue managers).

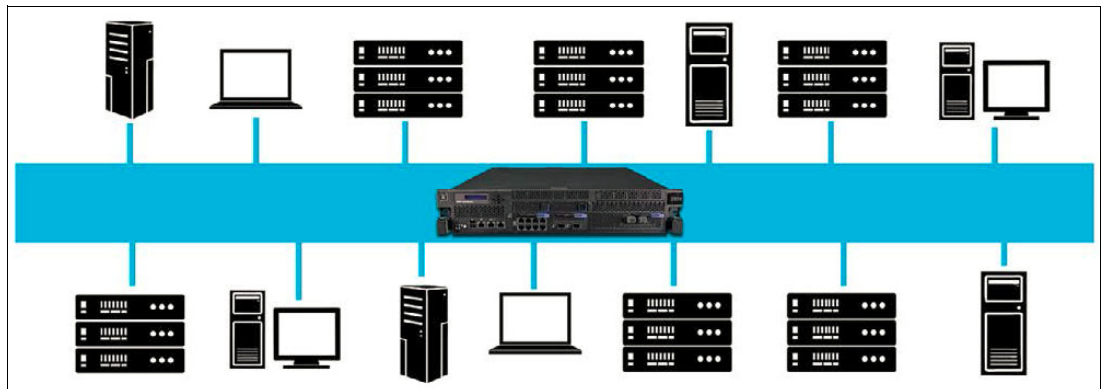


Figure 1-2 IBM MQ Appliance M2000 as a messaging hub

1.1.2 Business use cases

The IBM MQ Appliance M2000 has multiple advantages over a solution that is built on messaging software alone. It saves enterprises from having to build their own messaging servers and allows them to implement an IBM MQ-based solution with less in-house IBM MQ expertise.

Table 1-1 lists the top use cases for using the IBM MQ Appliance in four distinct business situations.

Table 1-1 Business use cases for the IBM MQ Appliance M2000

Business need	Advantages of the IBM MQ Appliance M2000
High availability	<ul style="list-style-type: none"> ▶ IBM MQ is known for high availability, but an appliance design enhances it ▶ Availability is more component-based and less configuration-dependent ▶ Failover is more assured with appliance <i>pairs</i> and mirrored messages
Consolidate an IBM MQ infrastructure	<ul style="list-style-type: none"> ▶ With fewer resource-intensive servers running IBM MQ, costs are reduced ▶ It is easier to deploy queue managers from an appliance-based hub ▶ Downtime can be reduced by using multiple appliances

Business need	Advantages of the IBM MQ Appliance M2000
Deploy messaging to remote locations	<ul style="list-style-type: none"> ▶ Connectivity at remote locations can improve with an appliance there ▶ An appliance allows a simpler infrastructure set-up compared to servers ▶ Failover support is more assured because of the appliance's HA features
Deploy messaging to business partners	<ul style="list-style-type: none"> ▶ Partners can be brought onboard quickly, regardless of their IBM MQ expertise ▶ Partners are now more likely to meet industry standards for messaging ▶ Customized administrative controls allow tight control of partner access

1.2 Features and benefits

The core features for the IBM MQ messaging product are described in the following resources:

- ▶ *IBM MQ V8 Features and Enhancements*, SG24-8218:
<http://www.redbooks.ibm.com/abstracts/sg248218.html?open>
- ▶ IBM MQ Knowledge Center:
https://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.pro.doc/q001010_.htm

Beyond the proven capabilities of IBM MQ, there are several aspects of the IBM MQ Appliance M2000 that make it an attractive alternative to traditional IBM MQ. These advantages are described in the next sections.

1.2.1 High availability

High availability (HA) is easy with the IBM MQ Appliance M2000, or rather, with two appliances that are deployed together as an HA Group.

The IBM MQ Appliance M2000 takes a new approach to high availability. The original IBM MQ product has built-in high availability features that are widely embraced by IBM customers. However, the IBM MQ Appliance M2000 improves on this high availability by trading configuration-based HA for component-based HA in which appliances are paired, and persistent messages mirrored to ensure seamless delivery if there is a failover event. If a queue manager fails on one appliance, a queue manager on the paired appliance takes over and delivers the mirrored messages that it has in storage.

This approach has the following advantages:

- ▶ Enables automatic failover policies for assured message delivery
- ▶ Supports manual failover triggers for rolling upgrades
- ▶ Easier to set up than other HA solutions (no shared system or disks)
- ▶ Synchronous replication ensures 100% message fidelity

1.2.2 Easy administration

The IBM MQ Appliance M2000 strives to make messaging administration as easy as possible by combining a powerful new web user interface with traditional command-line interface (CLI) interactions, where appropriate.

The IBM MQ Console is intuitive, which making it easy to complete numerous routine administrative chores and allows new users to get up to speed quickly. The CLI is ideal for advanced users and is the only administrative option for some advanced functions.

Tip for readers: Throughout this book, the authors show how to use the IBM MQ Console for tasks that can be completed by using the console. When a particular task *must* be performed from the CLI, the CLI method is described.

IBM MQ Console

The new IBM MQ Console is a browser-based messaging administration tool that can be used with (or in place of) earlier administrative options, such as IBM MQ Explorer and, for users of the CLI, the popular IBM MQ Script Commands (also known as runmqsc).

The IBM MQ Console (see Figure 1-3) provides administrative tools that are familiar to IBM MQ administrators and easy to use for individuals who are using IBM MQ for the first time. The console is used to define and administer various IBM MQ capabilities and enable remote administration of appliances that are installed outside a company's main IT facilities.

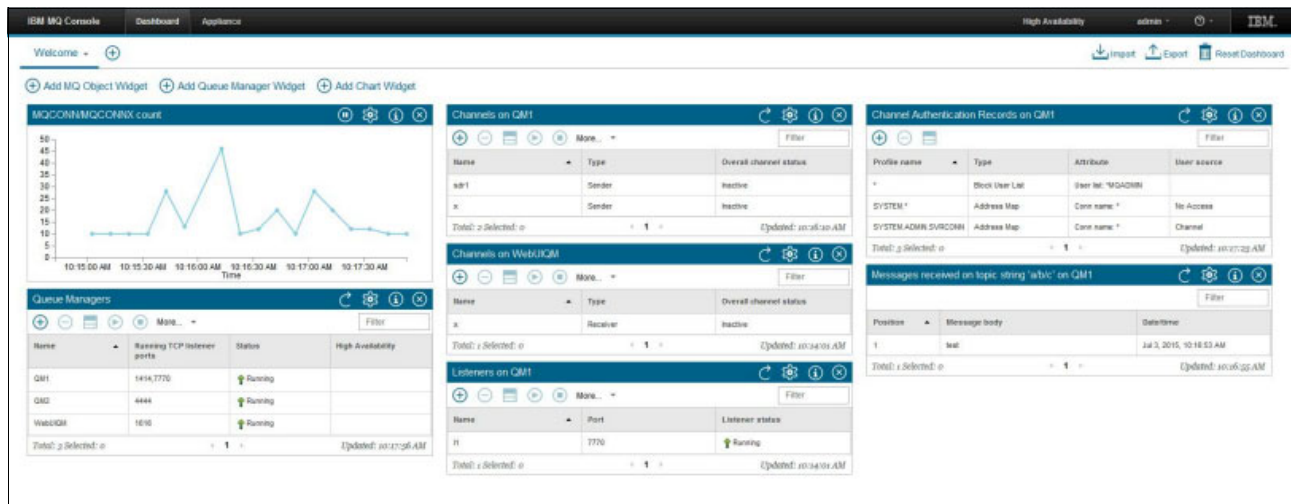


Figure 1-3 IBM MQ Console

CLI

The CLI is the first method that the administrator uses to interact with the appliance. Some tasks are only executable by using the CLI. At the time of writing, the following CLI-dependent tasks are available:

- ▶ Setting up the certificate for the IBM MQ Appliance's web user interface
- ▶ Setting up the certificate and key for each queue manager
- ▶ Configuring the queue manager HA preferred server
- ▶ Setting up logging targets for appliance error logs
- ▶ Copying files (such as certificates or diagnostic information) to or from the appliance

Although the appliance CLI (see Figure 1-4) does not provide the same scriptable environment as a UNIX shell, it is the primary means through which the IBM MQ administrator performs IBM MQ and appliance-related functions. The CLI gives access to IBM MQ commands, such as creating or deleting queue managers, runmqsc commands, and appliance-level commands.

```
M2000(config)# crypto
Crypto configuration mode
M2000(config-crypto)# password-map

Please enter alias-name and plaintext passwords pairs
- Enter a blank alias name to finish

Alias-name: WebUIPass
Plaintext password: *****
Re-enter plaintext password: *****

Alias-name:
Password-map updated (1 entry)

M2000(config-crypto)# keygen C US L Raleigh ST NC O CompanyB OU SA-W525 CN mqa00
2p.companyb.local rsa 2048 gen-object object-name WebUIKey file-name WebUI expor
t-key password-alias WebUIPass
keygen : Generating a 2048 bit RSA private key
keygen : This could take some time, please be patient
keygen : Saved private key in 'cert:///WebUI-privkey.pem'
keygen : Saved private key in 'temporary:///WebUI-privkey.pem' [INSECURE DIRECTO
RY]
keygen : Saved Certificate Signing Request in 'temporary:///WebUI.csr'
M2000(config-crypto)#
```

Figure 1-4 IBM MQ Appliance CLI

1.2.3 Appliance options and upgrades

The IBM MQ Appliance M2000 is a flexible solution not only for the deployment and configuration options it provides, but also in terms of processing capacity.

Depending on your needs, the following appliances are available:

► IBM MQ Appliance M2000A

A high-end solution for enterprise messaging consolidation. The M2000A offers the following features:

- Access to all of the CPU cores in the appliance
- Handles large IBM MQ workloads for persistent and non-persistent messaging
- Can host multiple queue managers to act as a messaging hub
- Potentially replaces multiple, separate IBM MQ servers
- Supports IBM MQ Advanced Message Security (AMS) and Managed File Transfer

► IBM MQ Appliance M2000B

A lower-cost solution for off-premise use, such as in a branch office or factory location. The M2000B offers the following features:

- Access to a subset of the CPU cores in the appliance (but with the same software and hardware as M2000A)
- Ideal for environments with less stringent messaging throughput requirements
- Supports all major features of M2000A, including HA and IBM MQ AMS
- Trade-up part adds easy, cost-effective option to upgrade to M2000A capacity, if needs change

The trade-up part option makes it easy to add capacity. The full set of CPU cores is in the appliance; therefore, the extra capacity is ready for you to use when you contact your IBM representative and make the deal.

1.2.4 Support for hardware and software

With the IBM MQ Appliance M2000, support for the hardware platform and firmware (including the IBM MQ software on the appliance) is provided through a single support infrastructure. Therefore, all support is provided by IBM.

Compare this single point-of-support model with a classic virtual infrastructure model, where separate support arrangements might be needed for the following distinct hardware components and software stacks:

- ▶ SAN controllers
- ▶ SAN disks
- ▶ SAN software or firmware
- ▶ Host Bus Adapter (HBA)
- ▶ HBA drivers
- ▶ Fiber switch fabric
- ▶ Hypervisor
- ▶ Operating system (OS)
- ▶ Hypervisor support drivers in the OS
- ▶ SAN drivers in the OS (for direct-access mode support to disk LUNs)
- ▶ IBM MQ

Even within an organization, support might be needed from teams that are responsible for Storage, Virtualization, OS, Security, and IBM MQ to configure a queue manager on a server. With the appliance deployment model that is described in this book, most queue managers can be built by the IBM MQ team alone without interaction with other support teams. The only other team that is likely to be involved is Security, which might need to provision LDAP accounts (if LDAP is used for authentication and authorization within the appliance queue managers).

1.2.5 Security

IBM Advanced Message Security is built into the IBM MQ Appliance M2000. This feature brings multiple benefits to the enterprise, including end-to-end protection, administrative logging, and, more generally, easier compliance with today's more stringent messaging security standards.

The IBM MQ Appliance M2000 includes the following security features:

- ▶ Authorize appliance administrators to perform IBM MQ administration: You can separate the roles of appliance administration from messaging administration.
- ▶ Support for secure connectivity over SSL/TLS: Certificates can be imported into the appliance.
- ▶ Scalable security administration: Define a few local users or use an offboard repository for larger communities.
- ▶ Lock-down capabilities: The appliance's pre-optimized security features cannot be altered.

1.3 About this book

In this section, we describe why this IBM Redbooks publication was produced and who will gain the most benefit from reading it.

1.3.1 Intended audience

The book presents underlying concepts and practical advice for integrating the IBM MQ Appliance M2000 into an IBM MQ infrastructure. Therefore, it is aimed at enterprises that are considering a possible first use of IBM MQ and the IBM MQ Appliance M2000 and those that identified the appliance as a logical addition to their messaging environment.

Details about new functionality and changes in approaches to application messaging are described, where appropriate. The authors' goal is to help readers make informed design and implementation decisions so that they can successfully integrate the IBM MQ Appliance M2000 into their environments.

A broad understanding of enterprise messaging is required to fully comprehend the details that are provided in this book. Readers are assumed to have at least some familiarity and experience with complimentary IBM messaging products.

1.3.2 Covered topics

The three parts of this book provide a sample environment in which to gain an understanding of new IBM MQ Appliance. This book consists of the following parts:

- ▶ Part 1, “Overview” on page 1: Provides a basic understanding of the IBM MQ Appliance M2000 and the business scenario the authors use to carry readers through the instructional material that is presented here.
- ▶ Part 2, “Using the IBM MQ Appliance” on page 19: Includes detailed instruction and advice for installing, configuring, and using the IBM MQ Appliance M2000, including creating queue managers, managing HA, and options for obtaining support.
- ▶ Part 3, “Appendixes” on page 183: Provides more follow-up information about topics, such as cryptographic changes and an IBM MQ Appliance Upgrade Transcript.

1.3.3 Topics not covered

Some functions of the IBM MQ Appliance M2000 are *not* addressed in this book. For these topics, see the following resources:

- ▶ Advanced Message Security

The IBM MQ Advanced Message Security function of the appliance allows end-to-end encryption of message contents that are at rest on disk. Organizations that must encrypt their data in this way can use IBM MQ appliances in their messaging infrastructure and be assured that message content stored on the appliance is protected.

For more information, see these resources:

- IBM MQ Advanced Message Security product page:
<http://www.ibm.com/software/products/en/ibm-mq-advanced-message-security>
- IBM MQ Advanced Message Security (part of IBM MQ Knowledge Center):
https://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.sec.doc/q014580_.htm

- *IBM WebSphere MQ V7.1 and V7.5 Features and Enhancements*, SG24-8087:
<http://www.redbooks.ibm.com/abstracts/sg248087.html>
- *Secure Messages with IBM WebSphere MQ Advanced Message Security* (IMPACT 2014 presentation):
<http://www.slideshare.net/MoragHughson/websphere-mq-ams>
- *Secure Messaging Scenarios with WebSphere MQ*, SG24-8069:
<http://www.redbooks.ibm.com/redpieces/abstracts/sg248069.html>
- **Managed File Transfer**

The IBM MQ Managed File Transfer function enables users to move data that is held in files. Queue managers that are hosted on an IBM MQ Appliance M2000 can be a coordinator or a mover of these files by using IBM MQ messages.

For more information, see the following resources:

 - WebSphere MQ File Transfer Edition product page:
<http://www.ibm.com/software/products/en/wmq-fte>
 - WebSphere MQ Managed File Transfer (part of WebSphere MQ 8.0.0 Knowledge Center):
https://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.wmqfte.doc/wmqfte_intro.htm
 - *Getting Started with WebSphere MQ File Transfer Edition V7*, SG24-7760:
<http://www.redbooks.ibm.com/abstracts/sg247760.html?open>
 - *IBM WebSphere MQ File Transfer Edition Solution Overview*, REDP-4532:
<http://www.redbooks.ibm.com/abstracts/redp4532.html>
 - *Managed File Transfer for SOA using IBM WebSphere MQ File Transfer Edition*, REDP-4533:
<http://www.redbooks.ibm.com/abstracts/redp4533.html>
- **IBM MQ Telemetry**

IBM WebSphere MQ Telemetry provides real-time access to mobile devices, remote sensors, and other telemetry equipment and is delivered as part of IBM MQ V8 on distributed platforms. However, IBM MQ Telemetry relies on the IBM MQ Telemetry Transport (MQTT) protocol, and the IBM MQ Appliance M2000 cannot host connections from MQTT clients (the service that performs the function is not supported).

For more information, see the following resources:

 - WebSphere MQ Telemetry product page:
<http://www.ibm.com/software/products/en/wmq-telemetry>
 - IBM MQ Telemetry (part of WebSphere MQ 8.0.0 Knowledge Center):
https://www-01.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.prod.doc/q002760_.htm
 - *Building Smarter Planet Solutions with MQTT and IBM WebSphere MQ Telemetry*, SG24-8054:
<http://www.redbooks.ibm.com/abstracts/sg248054.html>
 - *Building Real-time Mobile Solutions with MQTT and IBM MessageSight*, SG24-8228:
<http://www.redbooks.ibm.com/Abstracts/sg248228.html>

► IBM MQ Light

IBM MQ Light provides a messaging run time for use by developers. It enables developers to install, configure, and create and write scalable and responsive messaging applications in the shortest possible time (a matter of a few minutes). IBM MQ Light uses an API that is simple to use, which provides the basic messaging capabilities that your applications require.

For more information, see the following resources:

- Getting started with IBM MQ Light

<https://developer.ibm.com/messaging/mq-light/docs/>

- IBM MQ Light V1.0.0

http://www.ibm.com/support/knowledgecenter/SSBJCR_1.0.0/com.ibm.mqlight.help/home.v10.doc/welcome_page.htm

- IBM MQ Light for Bluemix

<https://www.ng.bluemix.net/docs/#services/MQLight/index.html>



Business scenario overview

Company B is a fictional retailer that is used in this chapter as a business scenario. Today, Company B takes orders by telephone from their business users and uses a small messaging network to submit the orders and trigger the various business processes to complete the order. A message is generated by an application, which the telephone operator uses to input the order data. The application is directly connected to a queue manager in the network when the message is put on one of the queues that it hosts.

Company B is doing well, needs to increase the throughput of orders, and eliminate the manual ordering process that is occasionally prone to errors and is becoming a bottleneck. They also want to modernize. They realize that a web-based, front-end ordering system is what they need. They reviewed product literature and discussed at some length their requirements and proposal of the use of the IBM MQ Appliance as a highly available pair to achieve proposed order processing targets.

Company B see the following advantages of integrating the IBM MQ Appliance:

- ▶ High availability (HA)
- ▶ Central Gateway for Queue Manager Cluster
- ▶ Ease of installation
- ▶ Consolidation and replacement for out of support products

This chapter describes how Company B adopt the IBM MQ Appliance and includes the following topics:

- ▶ 2.1, “Current configuration ” on page 12
- ▶ 2.2, “Target configuration” on page 14

2.1 Current configuration

Company B has a few queue managers in their configuration; however, although the number is small, they run their business on back-level and unsupported products. They realize that this configuration is no longer acceptable to their future needs and believe that it is time to improve their agility.

2.1.1 Queue manager cluster configuration

The current cluster configuration for Company B comprises three full repository queue managers on separate servers that are running the WebSphere MQ v7.5 product, as shown in Figure 2-1. It was decided when the system was originally designed that having another full repository can provide some resiliency in the system.

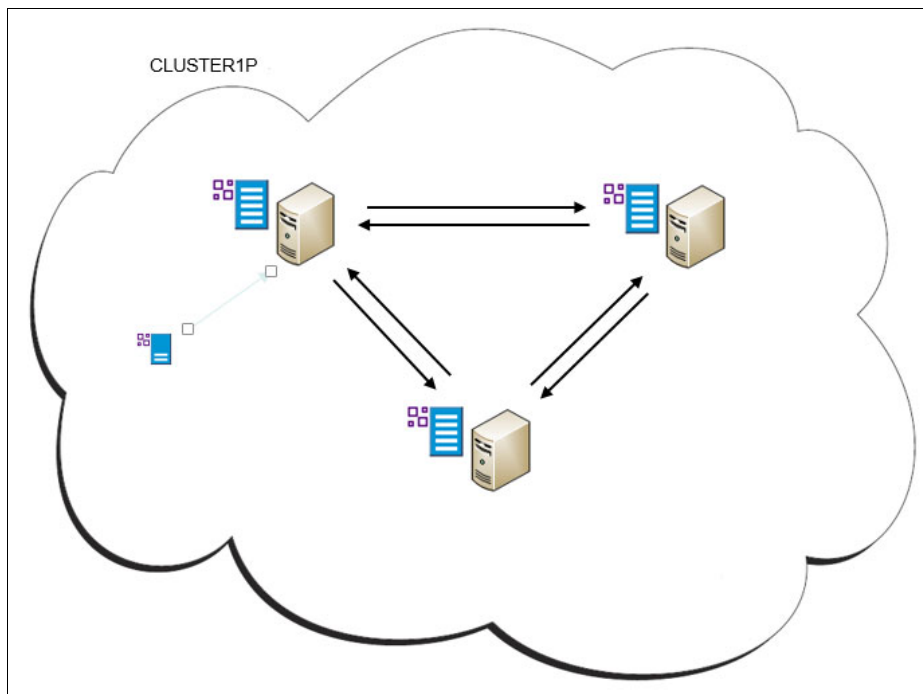


Figure 2-1 Three full repositories with six manually defined sender channels

Queues

There are several queues in the overall configuration for the order processing system. In this section, we describe only the main queues that are required to understand the scenario.

ORDER.REQUEST

This queue is a remote queue that is defined on a queue manager that is named `ORDERS` and is running WebSphere MQ v6.0. A message that represents the order is put to this queue for payment processing. The remote queue specifies the queue `PAYMENT.REQUEST` and queue manager `ACCOUNTS`.

PAYMENT.REQUEST

This queue is the local queue that is defined on the ACCOUNTS queue manager that is running WebSphere MQ V7.5. Messages arrive on this queue after they are routed by the ORDERS queue manager. An application receives the order message that contains payment information and performs the payment processing. Assuming payment is complete, the application publishes information about the order to topics in the cluster.

Topics and topic strings

After payment is successfully made, a message is published to subscribers in the clusters so that the appropriate business process can perform their various tasks. There is only one topic that is defined in the configuration, which is the same for and pre- and post-IBM MQ Appliance environments.

ORDER.PROCESSING: [/STUFF/ORDERS]

This topic is an administrative topic that defines the orders queue manager and a member of the CLUSTER1P cluster. As a result, all queue managers in this small configuration learn about the topic and the relevant topic string to use when performing the order publication.

Exits

Company B is using an exit within their production environment. They had developed an exit to change the LOCLADDR attribute of the auto-defined cluster sender channels in their cluster configuration. This change was made so that there was some control of the network adapter to which these channels bound themselves.

Note: Moving forward to the new configuration, these exits are not permitted because the IBM MQ Appliance does not support the installation of user code.

2.1.2 Application flow

The production order processing components and application flow of Company B are shown in Figure 2-2 on page 14.

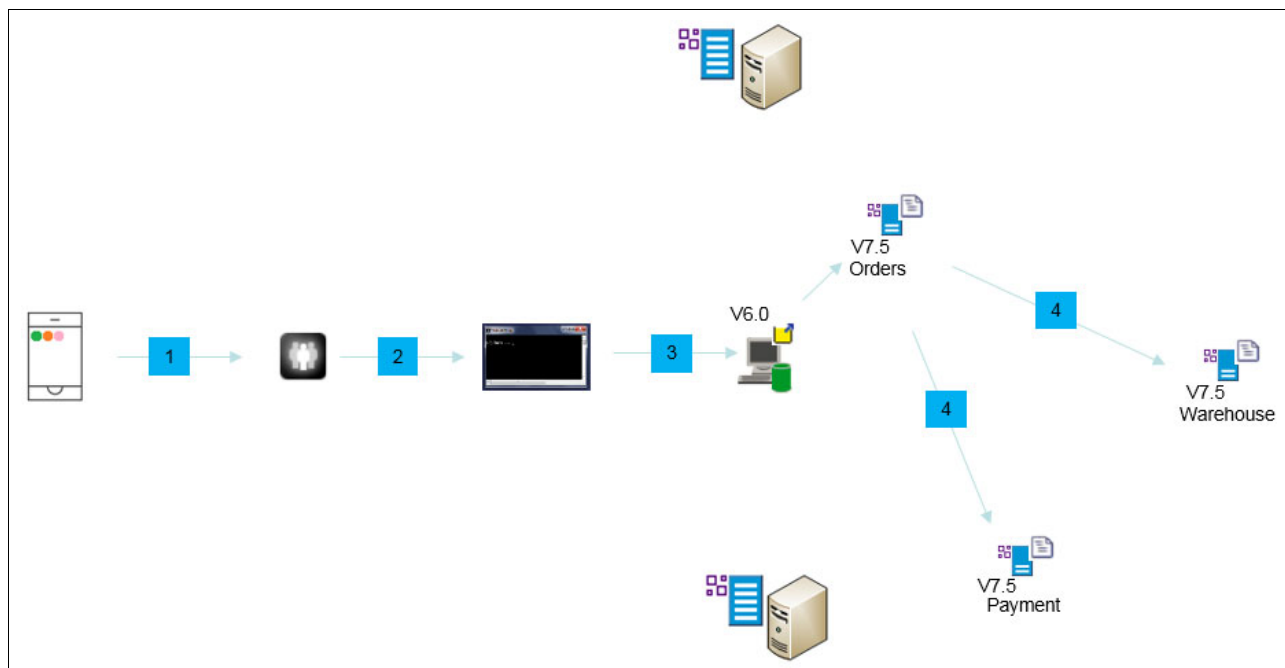


Figure 2-2 Production order processing components and application flow

The flow includes the following steps:

1. A customer contacts Company B and asks if they have a specific item.
2. An operator checks and determines that the item is available and uses a console application to place an order for the customers' item. The application that is used by the operator takes the information about the item that the customer requested and composes an IBM MQ message. The message is sent by using the MQPUT application programming interface (API) to an old version 6.0 queue manager.
3. A reply message is generated with a unique order identifier.
4. An application that uses the MQGET API retrieves the message from the ORDER.REQUEST queue and sends it to a distribution queue manager that is called ORDERS where publish subscribe is used to facilitate the rest of the order process. A new message is published to a topic with topic string ORDERS.INCOMING.

2.2 Target configuration

There are several reasons why Company B wants to introduce the IBM MQ Appliance into their environment. This section describes what changes were made to their configuration and outlines the reasons why those changes were made.

2.2.1 New full repository queue managers

Although the original configuration has minimum and maximum product levels of WebSphere MQ V6 and WebSphere MQ V7.5, the use of new configuration (including the appliances) means that the highest level of IBM MQ product becomes version 8.0. The architects recalled a discussion about this subject with the support organization and must meet the recommendation that was made by IBM to reduce the number of full repositories to two.

Tip: Only two servers are required to host full repository queue managers. Having only two is the optimum number to use. This number keeps the manual configuration changes to a minimum and the number of configuration messages that are needed for synchronization of every full repository (if there were more than two).

Company B considered the following approaches to achieve the recommended configuration:

- ▶ Add the full repository queue managers to the new appliances. Thought was given to the possibility of making them highly available while using the spare capacity that the appliance has available. Also, they were going to be using only one other queue manager on the appliance.
- ▶ Use the multi-installation features of the IBM MQ products. The thinking was that the two servers that are hosting full repositories can have IBM MQ V8.0 installed with WebSphere MQ V7.5. To achieve the recommendation that two is enough for containing the cluster topology, the full repositories FR01 and FR02 were demoted to partial repository queue managers.

This approach enables Company B to upgrade service levels for the full repository queue managers only without affecting any other queue managers and narrowing the scope of future change requests.

The option that Company B selected was to install IBM MQ 8.0 onto the servers where full repositories are hosted. The reason for choosing this option was primarily based on the ability to continually update the full repositories, as shown in Figure 2-3.

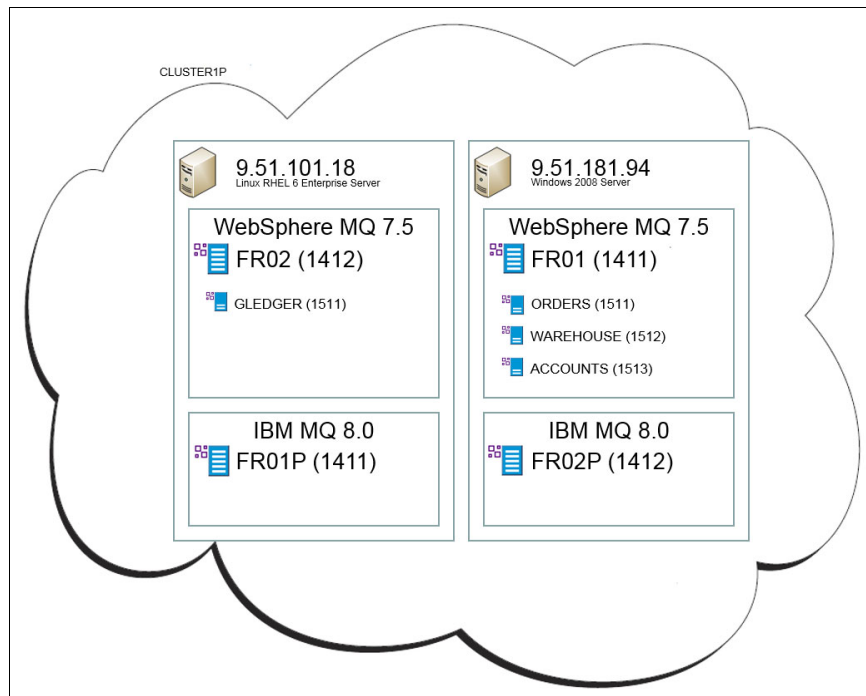


Figure 2-3 FR01P and FR02P are new full repositories in the cluster

2.2.2 HA gateway

Company B is adding a pair of new appliances to their domain; namely, `companyb.local`. A new queue manager that is joined to the cluster is created to be highly available, and others are created to replace the WebSphere MQ V6.0 queue managers in the original configuration.

The new appliances were given two new host names, MQA001P and MQA002P and are so named to begin a new standard of naming specifically to identify their environment. In this case, the appended letter P to the name identifies these appliances as belonging to the production environment.

The following two servers' fully qualified names are used:

- ▶ `mqa001p.companyb.local`
- ▶ `mqa002p.companyb.local`

For more information about naming the appliance and standards, see Chapter 3, “Planning” on page 21.

For more information about creating the HA groups, see Chapter 6, “Appliance administration” on page 55.

For more information about creating HA queue managers, see Chapter 10, “High availability” on page 149.

2.2.3 Central authentication and authorization

A directory server is added to the new environment for the purpose authorizing users that are required for messaging purposes. Company B chose IBM Security Directory Server, which uses the Lightweight Directory Access Protocol (LDAP).

LDAP is used for those queue managers in the final configuration that are running IBM MQ V8.0; that is, the full repositories and the HA queue managers that are running on the appliances. At some later point when the Company B can upgrade their other servers in the environment, Security Directory Server is used for authenticating users that are defined as deployed queue managers, as shown in Figure 2-4.

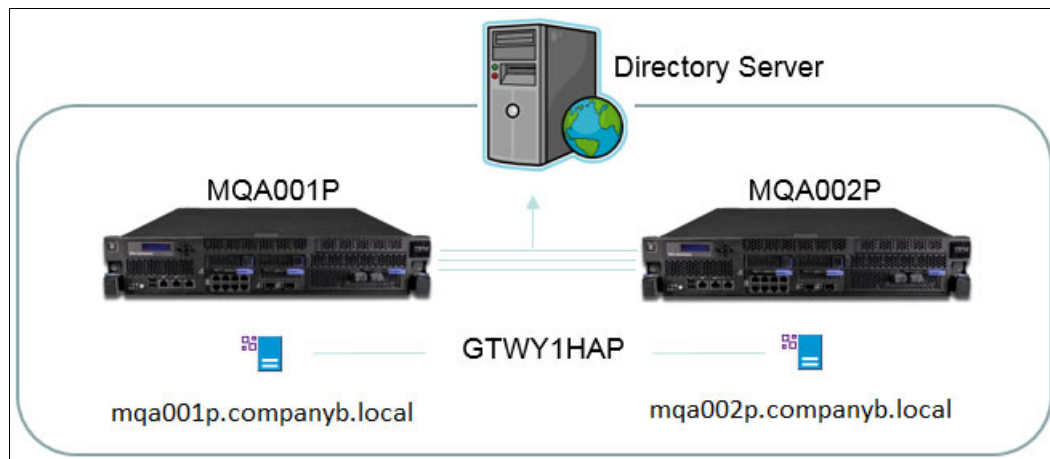


Figure 2-4 IBM MQ Appliance pair with HA queue manager GTWY1HAP

For more information about configuring users, see Chapter 6, “Appliance administration” on page 55.

For information about creating queue managers for use in an HA group, see Chapter 10, “High availability” on page 149.

2.2.4 Adding non-HA queue managers to the cluster

Company B has WebSphere MQ V6.0 running on one of its servers within the production environment. Company B must remove this queue manager and make its functions available on the IBM MQ Appliance. These functions are not considered critical and as such, are not made highly available.

For more information about creating queue managers, see Chapter 7, “Creating queue managers” on page 107.

2.2.5 Cluster configuration

In this section, we describe the cluster configuration that is used.

Queues

The following queues are used in the new configuration:

- **WEB.ORDER.REQUEST**

As with the original configuration, this queue is the queue that the order application specifies to perform the submission. In this new environment that contains the appliances, the application that is performing the submission is deployed inside WebSphere Application Server Liberty Profile.

The queue is defined on the GTWY1HAP queue manager and is an alias queue that resolves to a clustered topic, which also defined ORDER.REQUEST.

- **WEB.ORDER.REPLY**

This queue is where a confirmation message is sent when order processing completes. Within this message, there is confirmation number that can be passed back to the requester.

Note: An alias queue must resolve to a locally defined queue on the same queue manager (GTWY1HAP in this case) if that queue is not clustered. Alternatively, the name that is given in the alias queue as the target can be a cluster queue that is hosted in the cluster. For more information, see the Queue Name Resolution table in the Application Programming Guide in the IBM Knowledge Center, which is available at this website:

https://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.dev.doc/q025990_.htm

Topics

The /STUFF/ORDERS/ topic string is used with administered topics in the configuration topics that are used in the new configuration to assist the order processing system.

2.2.6 Application flow

Figure 2-5 shows the new application flow when the appliances are in place in the configuration.

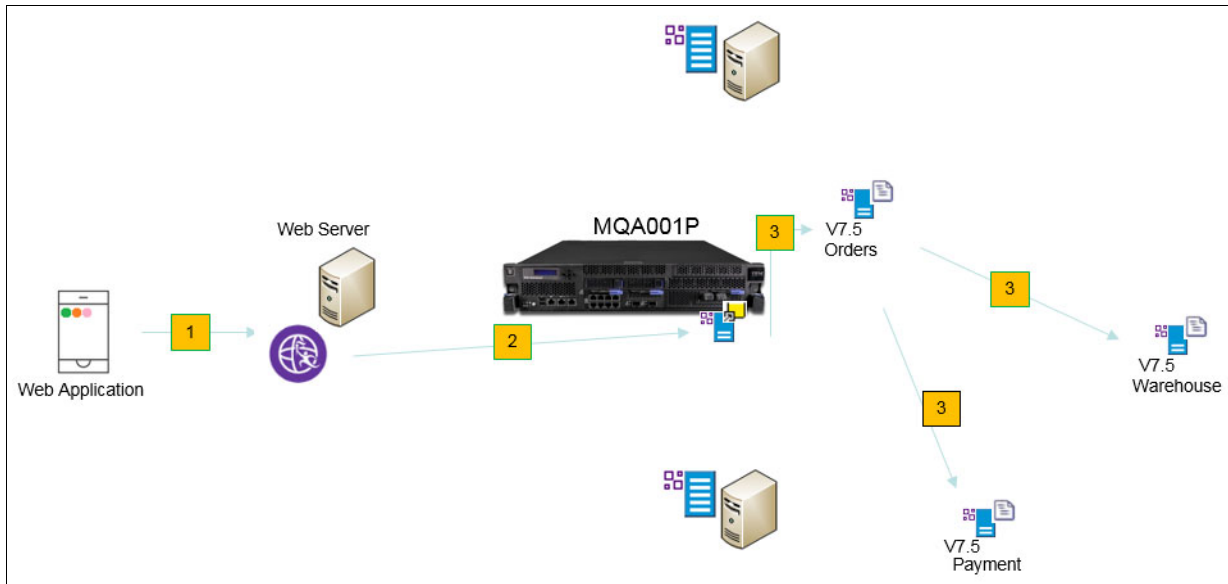


Figure 2-5 New application flow

The flow features the following steps:

1. A customer (or many customers simultaneously) uses the same smartphone to begin an order. However, the customer cannot make a phone call; instead, they use the web browser or mobile app to see all that Company B offers. The required items are found and an order is placed by using a provided web application of the website.
2. The web application that is running in this scenario on a WebSphere Application Server (Liberty Profile) passes the request to the IBM MQ Appliance.
3. The IBM MQ Appliance connects to the orders, payment, and warehouse services.



Part 2

Using the IBM MQ Appliance

This part includes the following chapters:

- ▶ Chapter 3, “Planning” on page 21
- ▶ Chapter 4, “Installing the IBM MQ Appliance” on page 33
- ▶ Chapter 5, “Initial appliance configuration” on page 37
- ▶ Chapter 6, “Appliance administration” on page 55
- ▶ Chapter 7, “Creating queue managers” on page 107
- ▶ Chapter 8, “IBM MQ object security” on page 123
- ▶ Chapter 9, “IBM MQ channel security” on page 137
- ▶ Chapter 10, “High availability” on page 149
- ▶ Chapter 11, “Application changes” on page 163
- ▶ Chapter 12, “Support for the IBM MQ Appliance” on page 171



Planning

Before implementing any new system into your organization, it is important to understand what is needed for the system to run and for you to use and support the system effectively.

For the IBM MQ Appliance, there are many things that must be considered and decided upon before setting up an appliance.

This chapter describes the environment that you need to install, configure, and use the IBM MQ Appliance quickly after it arrives at your data center.

This chapter includes the following topics:

- ▶ 3.1, “Number of appliances” on page 22
- ▶ 3.2, “Standards” on page 22
- ▶ 3.3, “Location” on page 26
- ▶ 3.4, “Other equipment needed” on page 26
- ▶ 3.5, “Installation prerequisites” on page 28
- ▶ 3.6, “Information needed before initializing the IBM MQ Appliance” on page 30
- ▶ 3.7, “Remote access to serial port” on page 30
- ▶ 3.8, “More information for complete initialization” on page 31

3.1 Number of appliances

How many IBM MQ Appliances do you need? This question is a difficult question and is analogous in many ways to the question, “How long is a piece of string?” The answer is always, “It depends”.

However, we can describe what sort of environment the IBM MQ Appliance is used in, and what is needed to meet the organizational need in that environment. Consider the following points:

- ▶ Do you need high availability (HA)? If the answer is yes, you need two appliances at your production site.

Note: Two appliances are required to implement an HA solution. More appliances might be needed for throughput. If HA is a requirement, appliances must be implemented in groups of two.

- ▶ What non-production systems are needed?

Development roles are not likely to require an appliance. IBM MQ Advanced for Developers is available in an unsupported free version or a fully supported single user licensed version. It might be cost effective to implement development-level queue managers on an appliance that is shared with testing users.

Testing environments need IBM MQ Queue Managers; however, in some environments, you might want to run these managers as a standard queue manager on another platform or on a smaller (M2000B) appliance.

Finally, most sites that use HA in production must run a pair of appliances in HA mode in a pre-production test environment. This configuration enables these sites to test their applications against failure scenarios before proceeding to production, and with new versions of firmware as they are released.

3.2 Standards

Standards are critical to the implementation of reliable systems that are easy to configure and maintain. Good standards assist systems implementation and maintenance by making things consistent. If things, such as the format of names or the network interfaces that are used for a specific function, are different between the installed appliances, working with the appliances becomes more difficult. Consistency makes working with the appliances simpler and leads to fewer problems.

3.2.1 Naming standards

All appliances should be given a unique system identifier (host name) when they are initialized. Appliances that are used in HA pairs must be uniquely identifiable, and so must have a unique identifier assigned.

Document a format standard that makes it easy to create a unique system identifier for each appliance. The format might embed values that indicate environment (for example, Prod, Test, and Dev), installed site (Room1 or Room2), or other items of interest. The site might have a server naming standard that can be used as is or enhanced to support identifying appliances and other types of servers.

Example 3-1 shows a proposal for a simple IBM MQ Appliance standard.

Example 3-1 Simple IBM MQ Appliance naming standard

Appliance Naming Standard:

Field 1: 1 character – Environment (P=Prod, T=Test, D=Dev)

Field 2: 1 character – Site (1=site 1, 2 = site 2, 3 = site 3 ...)

Field 3: 1 character – Platform (constant value “a” for “appliance”. Note: extension to existing server naming standards. These might define w=workstation, s=server, d=desktop)

Field 4: 1 character – Operating System (constant value “m” for “IBM MQ”. Note: extension to existing server naming standards. This might be w=windows, l=linux, z=zOS and so on

Field 5: 1 character - High Availability indicator (“h” if appliance is in a HA pair, “n” otherwise)

Field 6: 3 characters – unique number, assigned sequentially from zero.

First IBM MQ Appliance installed for Test usage without HA at site 1:

t1amn000

3.2.2 Network interface standards

Defining the use of network interfaces consistently helps to reduce errors that are caused by working with unfamiliar systems.

Example 3-2 shows a proposed standard that might be used for redundant, high-performance network connectivity from an appliance with HA enabled.

Example 3-2 Network interface standards

mgt0: Web UI and IPMI

mgt1: Command Line access (ssh)

eth10: Link aggregated 1 Gbps data interface

eth11: Link aggregated 1 Gbps data interface

eth12: Link aggregated 1 Gbps interface for log and SNMP traffic

eth13: HA group primary interface (determined by the firmware)

eth14: Link Aggregated 1 Gbps data interface

eth15: Link Aggregated 1 Gbps data interface

eth16: Link aggregated 1 Gbps interface for log and SNMP traffic

eth17: HA group alternative interface (determined by the firmware)

eth20: unused

eth21: HA replication interface (determined by the firmware)

3.2.3 Interface name standards

The IBM MQ Appliance is built on top of some IBM WebSphere DataPower capability. One of these capabilities is to provide a “Host Alias” for specific IP addresses that are assigned to network interfaces.

It is always better to provide consistent names to the IP addresses that are assigned to an interface to reference it rather than using the IP address. When consistent names are used, a configuration can be copied to another machine or migrated between environments without requiring extensive changes to cope with different IP addresses on the new appliance.

Example 3-3 shows a configuration of IP addresses that are assigned to network interfaces and the Host Aliases that are used to reference the network interfaces. A logging target then is created that sends data to a logging server. The log target can reference log-int instead of 10.161.121.5. The log target definition can be copied unchanged to another IBM MQ Appliance and works as is. If the IP address was referenced instead of the Host Alias, the log target must be changed on the second appliance before it can work.

Logging target: A logging target is a mechanism that can collect messages that are produced by the IBM MQ Appliance firmware and sends them to a log collector, which might be an SNMP trap collector, an NFS file system, or (most commonly) a syslog receiver. The IBM MQ error log files (AMQERR0?.LOG) are not included in the messages that are collected by a log target.

Example 3-3 Interface name standards

```
Interfaces:
eth10: 10.61.121.5/24 "Used for IBM MQ data"
eth11: 10.161.121.5/25 "Used for logging and SNMP"
eth13: 192.168.121.5/29 "Used for HA Primary"
eth17: 192.168.122.5/29 "Used for HA Secondary"
eth21: 192.168.123.5/29 "Used for replication"
Host Aliases:
data-int: 10.61.121.5
log-int: 10.161.121.5
hap-int: 192.168.121.5
has-int: 192.168.122.5
har-int: 192.168.123.5
```

3.2.4 Authentication and authorization standards

Authentication can be described as "Who is the connected person or system?". Authorization is "What is the connected entity allowed to do?". These capabilities are critical capabilities for the IBM MQ Appliance.

User IDs with access to administer the appliance can be separate from users with access to an IBM MQ queue manager that is hosted on it, although there is overlap because of the design of the appliance firmware.

Many sites have security standards and policies in place that provide direction about how authentication and authorization occur. Appliance administration users must be local to each appliance, and so these users might not meet these standards. If this issue is the case, the exception must be documented and a specific standard is written for user IDs that are managed on each appliance.

The IBM MQ Queue Managers on an appliance use LDAP for authentication and authorization. This configuration shows a single source of identity to both appliances in an HA pair. The LDAP identity provider meets corporate standards, which means that the appliance administrator does not have to try to implement them. Even in non-HA implementations, consistency and policy compliance are sufficient reasons to use LDAP rather than appliance-hosted users for the IBM MQ Queue Managers.

The IBM MQ Appliance does allow for messaging users and groups to be created locally, and this ability might be necessary for some implementations. However, this book describes the use of an external identity source as local identities provide many challenges to an IBM MQ administrator.

3.2.5 Logging standards

The IBM MQ Appliance supports the following logging options:

- ▶ cache
- ▶ console
- ▶ file
- ▶ nfs
- ▶ smtp
- ▶ snmp
- ▶ soap
- ▶ syslog
- ▶ syslog-tcp

Some of these options (cache, console, and file) collect the information and save or display it locally to the IBM MQ Appliance. The other options provide a mechanism to export the log information to another system. This ability allows for log data from all IBM MQ Appliances to be collected centrally, which can assist with incident investigation.

For more information about logging targets on the IBM MQ Appliance, see this website:

http://www.ibm.com/support/knowledgecenter/#!/SS5K6E_1.0.0/com.ibm.mqa.doc/reference/logeventcmds/logtargetcommands.htm

Decide on a logging strategy when planning to install the IBM MQ Appliance. Request logging servers that can receive the log stream if the decision is made to log appliance events to an external system.

Note: The IBM MQ error log files (AMQERR0?.LOG) cannot be handled by the appliance logging system. These files are always stored in the appliance file system.

3.2.6 Coded Character Set ID standards

The default Coded Character Set ID (CCSID) in the IBM MQ Appliance is 1208. This is a CCSID for UTF-8, which is a widely used and useful value. CCSID 1208 is also used as the default CCSID on queue managers that are running on the Linux operating system.

When migrating workloads from other queue managers to queue managers that are hosted on the IBM MQ Appliance, it might be necessary to review the application to ensure that it can operate correctly in a CCSID 1208 environment. For example, this necessity might require the application to explicitly set the CCSID when putting a message or to request conversion when getting a message.

It might be necessary to remediate code to include explicit handling the CCSID, or alter the queue manager CCSID when changing an application from using another queue manager to using a queue manager that is hosted on an appliance.

It also might be necessary to plan for several queue managers on an appliance (each running with different CCSIDs) to simplify migration of applications that are running on non-Linux platforms.

The CCSID of a queue manager on the IBM MQ Appliance can be set after the queue manager is created, as is the case on all other platforms. It is controlled as part of the queue manager configuration by using the **ALTER QMGR CCSID()** command.

3.3 Location

The IBM MQ Appliance is placed in a data center machine room, not an office or workspace. It is a high-performance machine with significant cooling requirements. The fans run at high speed and generate high noise levels, especially when air at office temperature is taken in.

Important: Install the IBM MQ Appliance in a data center machine room.

3.4 Other equipment needed

To complete the installation of the IBM MQ Appliance and connect it to your network, other equipment is needed. Your project runs more smoothly if this equipment can be ordered and installed in parallel with the delivery of the IBM MQ Appliance instead of waiting for it to arrive and then requesting items piecemeal.

3.4.1 Racks

The IBM MQ Appliance is physically designed like a network device (similar to a switch or managed router). It is not physically configured like a standard server. The standard server form factor has power and network connections at the back. The front of a standard server generally has switches or display panels only.

The IBM MQ Appliance has power connections at the back and all network connections on the front. The rack for an IBM MQ Appliance must have cable management at the front instead of at the back.

Because of this cabling requirement, the IBM MQ Appliance must be installed into a network-style rack. The rack must be configured with dual power. For more information, see this website:

https://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/8436_connectingapplianceoacpowersource.htm

Important: Install the IBM MQ Appliance in a network rack with dual power.

Important: Install IBM MQ Appliances that are in a HA group into two separate racks. This configuration is used so that rack failure does not affect both IBM MQ Appliances within the group. That is, the rack does not become a single point of failure for the system.

For more information about installing the IBM MQ Appliance into a rack, see this website:

https://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/in00000_.htm

Attention: The IBM MQ Appliance is heavy. Carefully read all of the Safety section of the manual and follow the installation instructions.

Attention: Installation of the IBM MQ Appliance might expose the installer to electrical and other hazards. Safety instructions in the manual must be read and followed.

3.4.2 Support servers

Managing the IBM MQ Appliance requires the administrator to copy files to and from the system. The IBM MQ Appliance was designed with high security in mind, and forces the administrator to start such transfers from the appliance. The IBM MQ Appliance supports the following protocols for copying files:

- ▶ HTTP
- ▶ HTTPS
- ▶ Secure Copy (SCP)
- ▶ Secured File Transfer Protocol (SFTP)

A system is required that can act as a server for at least one of these protocols. The simplest option is to use a UNIX type server that supports sshd natively. Alternatively, a Windows server can be used, which provides appropriate software that is installed to support the SCP server capability.

In this book, the examples all use the SCP capability that is provided by sshd.

Hint: Plan for a server that is used to stage files when copying on and off the appliance. A virtual server is suitable.

3.4.3 Cables

The two management interfaces of the IBM MQ Appliance use copper Ethernet cabling. These ports must be patched to the network interface in the rack that uses patch cables. Depending on your site standards, these patch cables might be supplied by your network services team or contractor.

The two copper Ethernet interfaces that are used for HA signaling (eth13 and eth17) must be cabled to the other appliance. This cabling can be implemented by connecting the ports on each appliance to a switch or by connecting the ports directly by using a single cable per port. For best performance, the interfaces must be cabled directly to the matching interface on the paired IBM MQ Appliance. Two 10-meter CAT 5e cables are included in the box. If these cables are long enough to provide direct interconnection between the appliances in your two racks, they can be used. If longer cables are needed, you must obtain them. If you choose to connect these interfaces by using switches, you need in rack patch leads instead.

If the HA primary, secondary, and replication interfaces on the appliances are connected by using switches, each pair of interfaces must be connected by using a different switch. The matching interfaces for the appliances must be connected to the same switch to reduce latency.

If you choose to install appliances for an HA pair in separate data centers, the connectivity for the different HA interfaces (eth13, eth17, and eth21) must have separate paths to reduce the risk of split brain that is caused by a single point network failure.

When implementing HA across significant distance, the latency must be less than 10 ms.

Note: Direct connection of cables between the HA signaling interfaces of the HA paired appliances is recommended for best performance. Any switch introduces some latency.

One 10 Gbps Ethernet connector (eth21) is used for replication of IBM MQ data between appliances in an HA pair. A 5-meter copper Enhanced Small Form Factor Pluggable (SFP+) Direct Attach cable is included with the IBM MQ Appliance. This cable can be used if it is sufficient to reach between the HA pair of appliances that are installed in separate racks.

Where an optic fiber connection is preferred or required, a short-range SFP+ transceiver is shipped with the appliance. An OM3 or OM4 standard optic fiber cable must be obtained in the correct length and installed between the appliances. If necessary, a connection to a switch is possible, but not recommended because of increased latency that is incurred by the switch connection.

Tip: Direct connection of the replication interfaces of HA paired appliances provides better performance than connection by using a switch. Fiber cable or the supplied 10 Gbps Ethernet cable can be used. Fiber cabling provides better performance for distances greater than 5 meters.

For high-speed, non-redundant, network connectivity for IBM MQ messages, the eth20 10 Gbps interface is available. This interface is an SFP+ interface and ships with a short-range SFP+ (10 GBASE-SR) transceiver installed. A fiber patch lead is required if you choose to use this connection. You must install fiber connectivity to the rack or a rack-mounted switch, which includes fiber interfaces.

If redundant network connections are required, up to six Ethernet interfaces can be combined to provide highly available network connectivity and increased bandwidth. These interfaces can be aggregated by using Link Aggregation Control Protocol (LACP) or other algorithms. LACP is recommended if it is supported in your switches. Patch cables are needed for each copper Ethernet connection that you choose to use.

3.5 Installation prerequisites

Before installing the IBM MQ Appliance, you must prepare your environment. The appliance must be installed into a rack (two racks are recommended for HA). These racks must be provisioned in your data center.

Power also must be available for the IBM MQ Appliances. You might need to notify your data center power manager of the power requirements of each IBM MQ Appliance so that they can ensure that enough power is available.

For more information about power requirements, see the installation manual that is available at this website:

http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/8436_specifications.htm

A minimum of two network connections are needed for each appliance. These connections must be provisioned before the appliance is installed or delays can occur while you wait for network cabling and switch configuration.

The minimal network configuration consists of the following connections:

- ▶ An mgt0 copper Ethernet. Allocate one Internet Protocol (IP) address for management operation. You need subnet mask and gateway address information in addition to the assigned address.
- ▶ A second IP address. This address is assigned to the Intelligent Platform Management Interface (IPMI) capability, which also uses the mgt0 interface. This configuration enables remote power control and remote access to the serial port (optional).
- ▶ An eth10 copper Ethernet. Allocate one IP address for IBM MQ data transfer, which is shared by all queue managers.

This minimal network configuration does not provide high performance networking or tolerance of network device failures.

It is sensible to plan your complete network requirements for the IBM MQ Appliance before installation. All of the network cabling can then be installed before the IBM MQ Appliance arrives.

Determine whether your site uses the default ports for management of the appliance. The default port Web User Interface (IBM MQ Appliance Web UI) port is 9090. The default Secure Shell interface (ssh) port is 22. These ports can be changed during initial configuration of the appliance by using the serial console port. If other ports are used, determine these ports in advance so that they can be configured on the appliances and included in firewall rules, if needed.

In some network environments, firewalls are in place to control access to the management and data network interfaces of systems. To complete the IBM MQ Appliance installation, a machine with a functional web browser must connect to the assigned management port of the management interfaces. Request firewall rules to enable connection from your administrative workstations to the IBM MQ Appliance Web UI and ssh ports on the IBM MQ Appliance.

To fully connect the IBM MQ Appliance to another appliance for HA and to provide redundant network access for management and data, the following items are required:

- ▶ Cabling and IP addressing for mgt0 and mgt1
- ▶ Cabling for eth10, eth11, eth12, eth14, eth15, and eth16 (is aggregated with LACP)
- ▶ LACP configuration of switches that support the data interfaces
- ▶ Switch configuration to enable native VLAN tagging of the IBM MQ Appliance link aggregation interface (optional)
- ▶ IP addressing for the link aggregated interfaces (one or more addresses)
- ▶ VLAN ID information if native VLAN tagging is enabled (optional)
- ▶ Cabling to direct connect eth13, eth17, and eth21 on each IBM MQ Appliance
- ▶ Assignment of private IP addresses that are used by each of the three HA network interfaces

Note: If remote site replication is implemented, the HA network interfaces likely are connected by using switches, and might use routable addresses. Ensure that the interfaces are still on separate subnets, and enforce correct routes by using static route entries on the interfaces.

3.6 Information needed before initializing the IBM MQ Appliance

After physically installing the IBM MQ Appliance in a rack, the next step is to initialize the appliance so that it can be accessed by using network connections. At a minimum, you must collect the following information before the installation begins:

- ▶ IP address that is assigned to the mgt0 interface
- ▶ Subnet mask for the mgt0 interface
- ▶ First Hop Router address (often called default gateway) for the mgt0 interface

This information is needed to complete the minimal configuration by using the setup wizard, which is accessed by using the serial console port. After the network-based management interfaces are available, the rest of the setup is performed by using network interfaces. This process reduces the amount of time the administrator must spend in the machine room.

This process also allows much of the configuration to be performed by using the IBM MQ Appliance Web UI. This interface provides an improved experience for configuring the IBM MQ Appliance, compared to performing all work by using the command-line interface (CLI).

3.7 Remote access to serial port

Depending on your site standards, you can have several choices of accessing the serial console port, including the following choices:

- ▶ Only inside the machine room, by connecting a notebook to the supplied serial cable
- ▶ From elsewhere, by connecting the appliance serial console port to a terminal server or console server
- ▶ From elsewhere, by enabling IPMI

Enabling IPMI also has the advantage of enabling remote power control. Only implement this choice if this choice does not violate security standards for your site.

Determine whether you need remote access to power control or the serial console to effectively support the IBM MQ Appliance. If this access is not normally allowed in your company policies, you might need to request policy exemptions before IPMI is implemented on the IBM MQ Appliance.

3.8 More information for complete initialization

After completing the initial configuration wizard, the administrator must connect to the Web UI of the appliance and accept the license agreement. The system then restarts.

After the restart, the rest of the appliance initialization can be performed by the administrator. More information about the environment is needed for these steps. The administrator needs the following information to complete the initialization and prepare the appliance for creating and running queue managers:

- ▶ Domain Name Server (DNS) addresses
- ▶ Network Time Protocol (NTP) server addresses
- ▶ Time zone information
- ▶ Network address information for all interfaces except mgt0, which was configured in the wizard
- ▶ Log server protocols
- ▶ Log server addresses



Installing the IBM MQ Appliance

The IBM MQ Appliance is installed in a standard 19-inch (48.26 cm) rack. The installation process is documented in the IBM Knowledge Center.

The appliance has all network interfaces on the front panel. Power is connected to receptacles on the back panel. This configuration matches the form factor of most rack-mounted switches and routers, but not of normal servers.

For more information about the work that must be done before installing the appliance, including power and cabling work, see Chapter 3, “Planning” on page 21.

Also, the IBM Knowledge Center features the following critical chapters regarding appliance installation:

- ▶ The *Safety* section must be read in its entirety by personnel who are installing the appliance and is available at this website:
http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/8436_dangernotices.htm
- ▶ The *Prepare for installation* section is available at this website:
http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/8436_prepareinstallationchapter.htm
- ▶ The *Installing the appliance in a rack* section and includes instructions for installing the rails, the appliance on the rails, and connecting the appliance to power and the network, and is available at this website:
http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/8436_installingdevice.htm

Important: Pay attention to the danger and safety warnings in the IBM Knowledge Center. The appliance is heavy and represents an injury danger if not handled correctly. Any activity that involves electricity presents a danger of death by electrocution. Follow the safety instructions that are described in the IBM Knowledge Center.

This chapter includes the following topics:

- ▶ 4.1, “Rack choice”
- ▶ 4.2, “Power” on page 35
- ▶ 4.3, “Environment” on page 35

4.1 Rack choice

The IBM MQ Appliance is constructed as a network device, with power connected to the back and network cables that are connected to the front.

Figure 4-1 shows two appliances in a standard server rack. The appliances are separated by 1U so that the cable management housings can direct the cables to the back of the rack between the appliances.

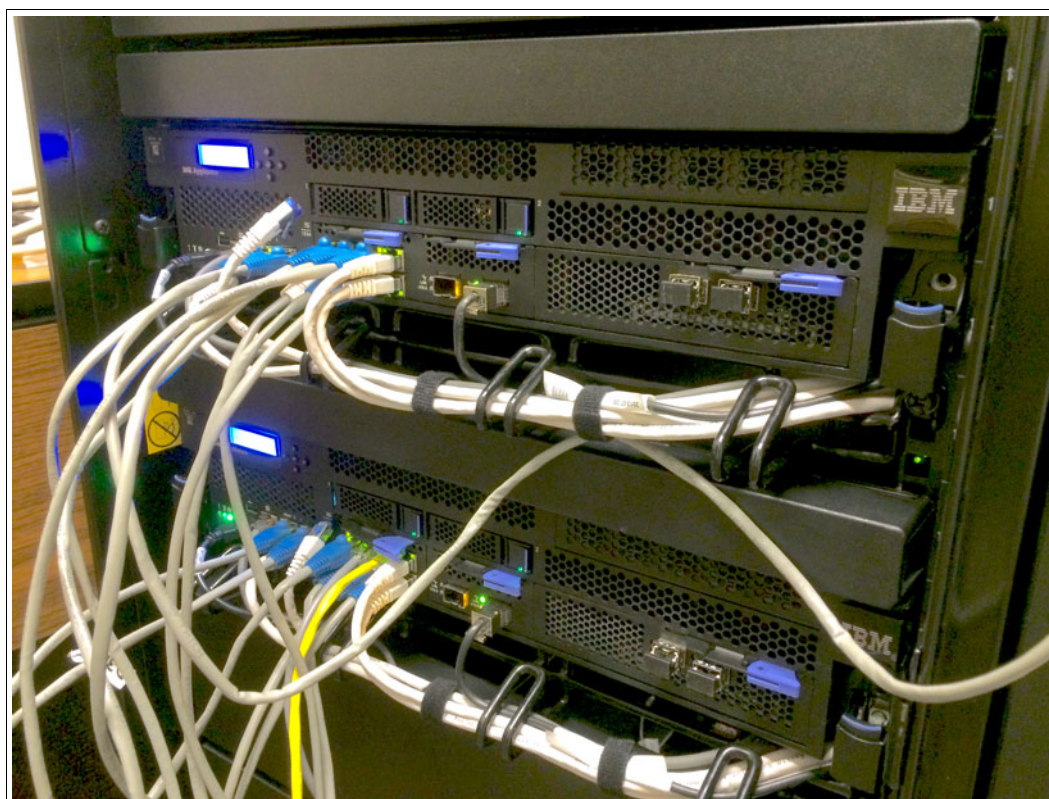


Figure 4-1 IBM MQ Appliances in a standard rack

If possible, install IBM MQ Appliances into a rack that is designed for switches and other network devices. This type of rack features cable management at the front of the rack, whereas standard server racks provide space for this management only at the back. In some cases, a standard server rack might not have enough space between the appliance and the front door of the rack. This configuration can cause the network cables to be kinked tightly or to sit against the door, which can lead to premature failure of the cables.

If you are installing the appliance into a standard server rack, leave 1U or 2U free below each appliance. This space allows network cables to be run from the back of the rack to the network interfaces on the front of the appliance. Also, ensure that there is sufficient space between the front of the appliance and the door to allow the cables to be inserted without a severe kink.

If IBM MQ Appliances are installed into a rack with front cable management, they do not need to be separated. The door is further away from the appliance and there is no requirement to run network cables from the cable management points at the back of the rack between appliances to the front.

4.2 Power

Ensure that both power supplies are correctly connected to separate power sources. If only one power supply is connected, the appliance considers the unconnected power supply to be faulty, and reports a power supply fault.

If both power supplies are connected to the same power source, a failure of the power source removes power from the appliance, which leads to an outage for non-high availability (HA) queue managers. An outage can occur, even for HA queue managers if both appliances had both power supplies connected to the same single source.

Important: Note the danger warnings in the manual before working with power supplies or connecting the IBM MQ Appliance to power sources.

4.3 Environment

The IBM MQ Appliance has specific temperature and environmental requirements, which are documented in the Rack Requirements section of the manual that is available at this website:
http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/8436_rackrequirements.htm

The temperature requirement is that the ambient temperature within the rack cannot exceed 35 °C (95 °F).



Initial appliance configuration

When the IBM MQ Appliance is delivered, only one interface is enabled. This interface is the serial interface, and is normally accessed by directly connecting a notebook system to the appliance by using a serial cable or USB to serial cable. Suitable cables are supplied with the appliance.

All network interfaces are disabled when the appliance is delivered. This configuration is a security feature to ensure that nobody that is watching devices on the network can detect a new appliance and connect to it before the default administrator password is changed.

The initial appliance configuration is performed by using the serial console port. It configures just enough on the appliance to allow the full configuration to be completed remotely by using the appliance web console and the appliance command-line interface (CLI) that is accessed by using Secure Shell (ssh).

This chapter summarizes and extends the information in the manual that is available at this website:

http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/8436_definingbaseconfiguration.htm

This chapter includes the following topics:

- ▶ 5.1, “Preparing a notebook to connect to the serial console port” on page 38
- ▶ 5.2, “Connecting the USB serial adapter cable” on page 38
- ▶ 5.3, “Running a terminal emulation application” on page 38
- ▶ 5.4, “Logging in for the first time” on page 39
- ▶ 5.5, “Installation wizard” on page 40
- ▶ 5.6, “Accepting the appliance license terms and conditions” on page 43
- ▶ 5.7, “Implementing IPMI” on page 50
- ▶ 5.8, “Next steps” on page 54

5.1 Preparing a notebook to connect to the serial console port

Most modern notebooks do not feature a serial port; instead, they provide USB ports. A USB-to-serial cable is shipped with each appliance so that a USB connection can be used to connect to the serial console port on the appliance.

A driver is needed before the cable can be used. If the driver is not installed on the notebook system, it must be installed. A suitable driver for Mac or Windows systems is included in the Resource Kit CD-ROM that is included with the appliance.

Copy the .zip image for your operating system from the driver/mac or driver/win directory.

The following files are available:

- ▶ Mac: md_PL2303_MacOSX10.6_dmg_v1.4.0.zip
- ▶ Windows: PL2303_Prolific_DriverInstaller_v1417.zip

Extract the .zip file for your operating system (OS) into a directory. Follow the instructions in the included document to install the driver. A restart might be required.

5.2 Connecting the USB serial adapter cable

Plug the USB-A end of the USB-to-serial adapter cable into the notebook.

If the USB driver is installed correctly, a serial device is dynamically created when the USB serial adapter cable is connected to the notebook.

On a Mac, this device is called /dev/tty.usbserial, or /dev/cu.usbserial, or both devices are visible and enable connection to the appliance.

On a Windows system, the serial device is COMx (that is, COM1, or COM2, and so forth). To discover which COM port is assigned, look in the Ports section of the Windows Device Manager. The COM port number might not be consistent every time the cable is attached to a Windows system.

Connect the RJ45 end of the USB-to-serial cable to the serial console port of the appliance.

Care: Do not plug the USB-to-serial cable into an Ethernet port. Do not plug an Ethernet cable into the serial console port.

5.3 Running a terminal emulation application

A terminal emulation application is needed to allow characters you enter to be sent to the appliance and the output from the appliance to be sent to a window on your display.

The appliance serial console port is configured to run at 115200 baud, with 8-bit characters, no parity, and 1 stop bit (8N1). Ensure that flow control is disabled.

5.3.1 Using a Windows system to connect

Many Windows applications can provide serial terminal emulation. You can use any of these applications to connect to the serial console port of the IBM MQ Appliance.

As of this writing, we used PuTTY to connect to the serial console when a Windows based notebook was used.

Example 5-1 describes configuring PuTTY to connect to the appliance where Windows created COM4.

Example 5-1 Use PuTTY to control the appliance by using the serial port

```
Start PuTTY
Select the Serial radio button
Enter the value COM4 for Serial line
Enter the value 115200 for Speed
Select the Category: Serial
Select the Flow control option None
Click Open
```

5.3.2 Using Mac OS X to connect

For a Mac notebook, the “screen” command line program can be used to drive a serial port. Start a terminal window, and run the following program:

```
screen -h 9999 -fn -L - /dev/tty.usbserial 115200,cs8,ixoff
```

To end the connection to the appliance, press Ctrl-A and Ctrl-\..

Example 5-2 shows an applescript that creates a terminal window, runs the screen program, and connects to the IBM MQ Appliance serial console port (assuming that the correct driver is installed and the cable is connected). Run Script Editor to create the script, and then export it as an application. The application can be started by double-clicking it in Finder.

Example 5-2 Applescript to run screen program in a terminal window

```
tell application "Terminal"
    do script with command "screen -h 9999 -fn -L - /dev/tty.usbserial
115200,cs8,ixoff"
    set custom title of window 1 to "MQAppliance: ^A ^\ to terminate"
    set number of columns of window 1 - 160
    set number of rows of window 1 - 50
    set normal text color of window 1 to "black"
    set background color of window 1 to "white"
end tell
```

5.4 Logging in for the first time

When you have a terminal emulator running and attached to the serial console port, press Enter until a login prompt appears.

Troubleshooting: If the login prompt does not appear, check that the speed (baud rate) and other settings are correct. Also, check that the USB-to-serial cable is correctly attached and that the appliance is powered on.

The initial administration user ID is admin. The initial password is listed in the manual that is available at this website:

http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/installing/8436_initializingtheappliance.htm

After the login is complete, the appliance prompts you to verify the type of license that was purchased.

If the system is licensed for Enterprise Capacity use (M2000A) then answer Yes.

The system then prompts you for a new password for the administrator account.

Care: Enter the replacement password carefully, and do not lose it. At this point, a secondary account that can reset the administrator account password is *not* created. If the administrator password is lost and cannot be recovered or reset, the appliance must be returned to IBM and reinitialized.

5.5 Installation wizard

The installation wizard is used to perform the minimum configuration that is needed to allow the full configuration to be performed remotely. While more configuration tasks can be performed by using the wizard, a machine room where the appliance is installed is not a good location to spend time while fully configuring the appliance.

The installation wizard can configure the following components and perform the following tasks:

- ▶ Network interfaces
- ▶ Network services (such as DNS)
- ▶ Unique system identifier
- ▶ Remote management access
- ▶ User account to reset the administrator password
- ▶ Save the resultant configuration

In this book, we use the wizard to configure the mgt0 Ethernet interface, web-based remote access, and command-line remote access (ssh).

All other configuration tasks to prepare the appliance are performed by using the following remote access tools:

- ▶ IBM MQ Console by using a web browser
- ▶ IBM MQ Appliance CLI by using ssh

5.5.1 Running the installation wizard

After the password is updated, the appliance prompts you to run the Installation Wizard, as shown in Example 5-3 on page 41. Respond y to start the wizard.

Example 5-3 Prompt to start the installation wizard

Do you want to run the Install Wizard? Yes/No [y/n]: **y**

The next sections show abbreviated prompts for the wizard. For more information about a full transcript of a configuration that is performed with the installation wizard, see Appendix C, “Transcript of appliance initialization” on page 199.

5.5.2 Configuring the mgt0 network interface

The first step of the wizard is configuring the network interfaces. The appliance prompts the user to confirm that network configuration is required, then prompts for whether each separate interface must be configured. If an interface is chosen, the appliance then prompts the user for information to configure that interface. A shortened version of a network configuration is shown in Example 5-4.

Example 5-4 Configuring network interface

```
Step 1 - Do you want to configure network interfaces? [y]:y
...
Do you have this information? [y]:y
Do you want to configure the eth10 interface? [y]:n
...
Do you want to configure the mgt0 interface? [y]:y
Modify Ethernet Interface configuration
Do you want to enable DHCP? [y]:n
Enter the IPv4 address for the interface in CIDR notation: 10.101.1.10/24
Enter the IPv4 address for the default gateway []:10.101.1.1
Do you want to configure the mgt1 interface? [y]:n
```

5.5.3 Configuring Network Services

Network Services from the perspective of the IBM MQ Appliance are services that are used by the appliance but are provided by other systems that are accessed over the network. These services include the Domain Name Service (DNS) and Network Time Protocol (NTP) service.

Do not configure these services yet. These services are not required for initial configuration and can be more easily configured by using the standard administration functions, as described in Chapter 6, “Appliance administration” on page 55.

Example 5-5 shows a “no” response to configuring network services by using the installation wizard.

Example 5-5 Do not configure network services by using the installation wizard

Step 2 - Do you want to configure network services? [y]:**n**

5.5.4 Configuring a unique system identifier

Always configure a unique system identifier. Although a system identifier is mandatory only if the IBM MQ Appliance joins a High Availability (HA) group, configuring the system identifier helps to ensure that it is always clear which appliance is being configured.

Example 5-6 shows the assignment of system identifier mqa001p.

Example 5-6 Setting up the system identifier by using the installation wizard

```
Step 3 - Do you want to define a unique system identifier for the appliance?
[y]:y
Enter a unique system identifier: mqa001p
```

A system identifier is recommended, as it is required to configure an HA Group.

5.5.5 Configuring remote management access

This step is an essential step in completing the wizard. Until remote management access is enabled, the only management access to the appliance is through the serial console.

The IBM MQ Appliance web UI and the ssh CLI must be enabled. Configuring the IBM MQ Appliance by using the IBM MQ Appliance web UI often is simpler than the use of the CLI, but not all aspects of the appliance can be set by using the web UI. Some attributes must be configured by using the CLI.

Example 5-7 shows responding to the remote management access prompt and then enabling ssh and IBM Appliance web UI access.

Example 5-7 Configuring remote management access by using the installation wizard

```
Step 4 - Do you want to configure remote management access? [y]:y
These configurations require the IP address of the local interface that manages
the appliance.
Do you have this information? [y]:y
Do you want to enable SSH? [y]:y
Enter the local IP address [0 for all]: 0
Enter the port number [22]: 22
...
Do you want to enable WebGUI access [y]:y
Enter the local IP address [0 for all]: 0
Enter the port number [9090]: 9090
```

After completing the wizard, remote management access is available through all configured network interfaces, which is poor security practice. However, only one interface was configured, so there is no actual risk created. The remote management access is bound to specific network interfaces by using the administration functions of the IBM MQ Appliance web UI and CLI. Implement this security enhancement after initial configuration has been completed. For more information, see Chapter 6, “Appliance administration” on page 55.

Care: Remote management of a system is a sensitive function that must be tightly controlled. On systems with more than one network interface, it is good practice to limit management functions to operate through only one or two of the interfaces. These interfaces should be connected to controlled management networks rather than to the same network that carries data.

5.5.6 Configuring a spare admin account

Before running the installation wizard, the appliance forced the administrator to change the password for the administrator account. If this password is lost or locked out and there is no account available that can reset the password, it might be necessary to return the appliance to IBM for it to be reset.

Therefore, it is always sensible to create an account that can be used to reset the administrator account password.

The account that was created by the wizard for this purpose has the same rights as the original administrator account. As shown in Example 5-8, we name the account `adminbackup`, rather than the default name `password-reset-user` so that this difference is clear.

Example 5-8 Creating a backup administrator account by using the wizard

```
Step 5 - Do you want to configure a user account that can reset passwords? [y]: y
Enter the name of the user account that can reset passwords [password-reset-user]:
adminbackup
New User configuration
Enter new password: *****
Re-enter new password: *****
Cleared RBM cache
```

5.5.7 Saving the configuration

After completing the previous sections in this chapter, the active configuration of the appliance is set. However, the appliance configuration is not yet saved to permanent storage. If the appliance is restarted before the configuration is saved, the configuration that is built by the wizard is lost.

Example 5-9 shows how to save the configuration by using the wizard, which ensures that the new configuration is restored when the appliance restarts.

Example 5-9 Using the installation wizard to save the configuration

```
Do you want to save the current configuration? [y]:y
Overwrite previously saved configuration? Yes/No [y/n]: y
Configuration saved successfully.
You have completed the Installation Wizard.
```

5.6 Accepting the appliance license terms and conditions

After the installation wizard configures the `mgt0` interface and remote management access, the administrator must accept the license terms and conditions. These terms and conditions are displayed the first time an IBM MQ Appliance administrator user logs on to the appliance by using the IBM MQ Appliance web UI.

At this point in the appliance setup process, the IBM MQ Appliance web UI is not configured with a certificate that correctly identifies the URL that it uses. This issue causes any web browser to notify the user that something might be wrong. The examples in this section show the Firefox browser, and describe how to verify that the certificate that is presented is acceptable, even if Firefox does not trust it.

As an administrator, you should always validate the identity of an untrusted certificate to ensure that you are not connecting to an incorrect appliance, or through a proxy that might capture credential or other critical information.

5.6.1 Connecting to the appliance web UI with a browser

Open the browser and connect to the URL with the address that was configured on the mgt0 Ethernet interface as described in 5.5.2, “Configuring the mgt0 network interface” on page 41. Figure 5-1 shows the Firefox untrusted connection warning (other browsers show similar warnings).

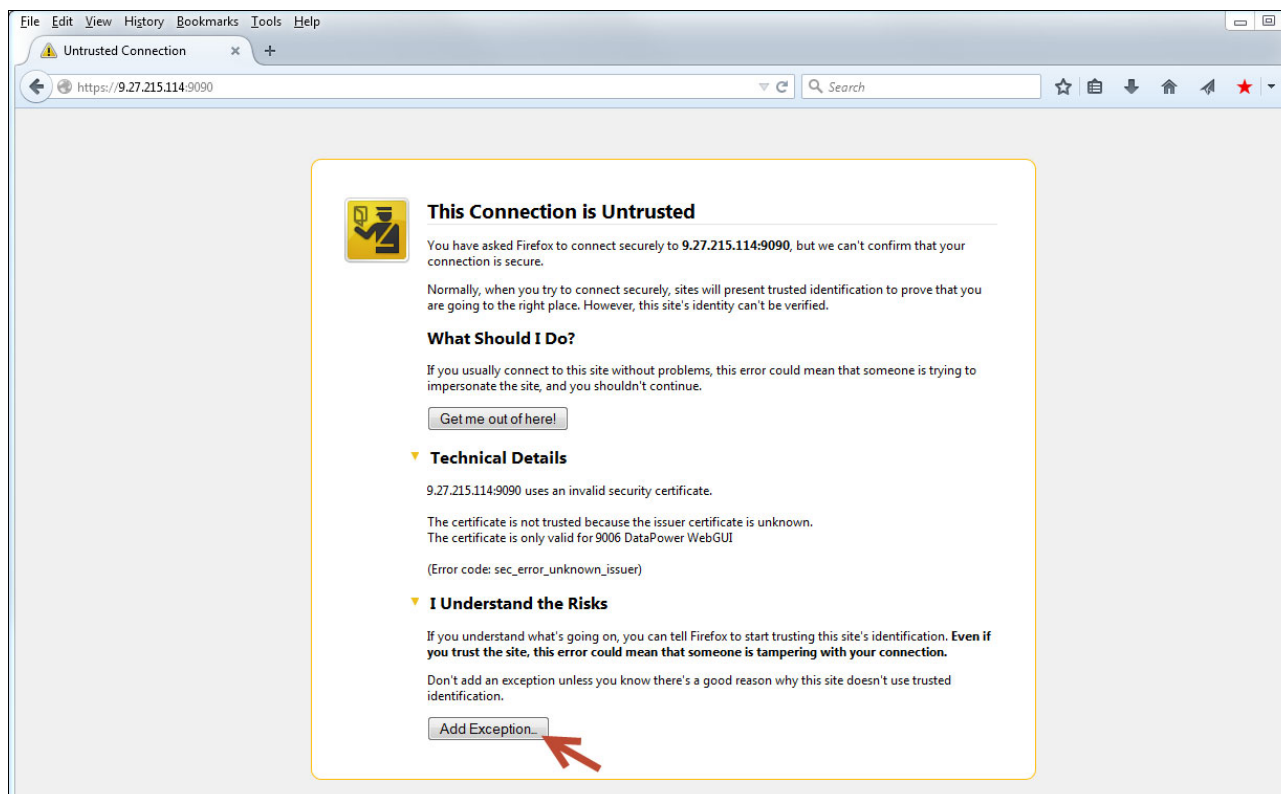


Figure 5-1 Firefox browser untrusted certificate warning

It is poor security practice to accept this warning without ensuring that it is acceptable. As shown in Figure 5-1, the following problems with the certificate are described:

- ▶ The certificate issuer is unknown
This problem means that the issuer of the certificate is not in the browser trusted CA store. This issue is expected for an appliance that uses the default web UI certificate.
- ▶ The certificate is only valid for 9006 DataPower WebGUI
The identity that is asserted by the certificate does not match the URL. This issue also is expected because it is a generic certificate that is used by all IBM MQ Appliances when they are first configured.

Click **Add Exception** to continue verifying the presented certificate before deciding whether to trust it. Figure 5-2 on page 45 shows the Add Security Exception panel that is displayed.

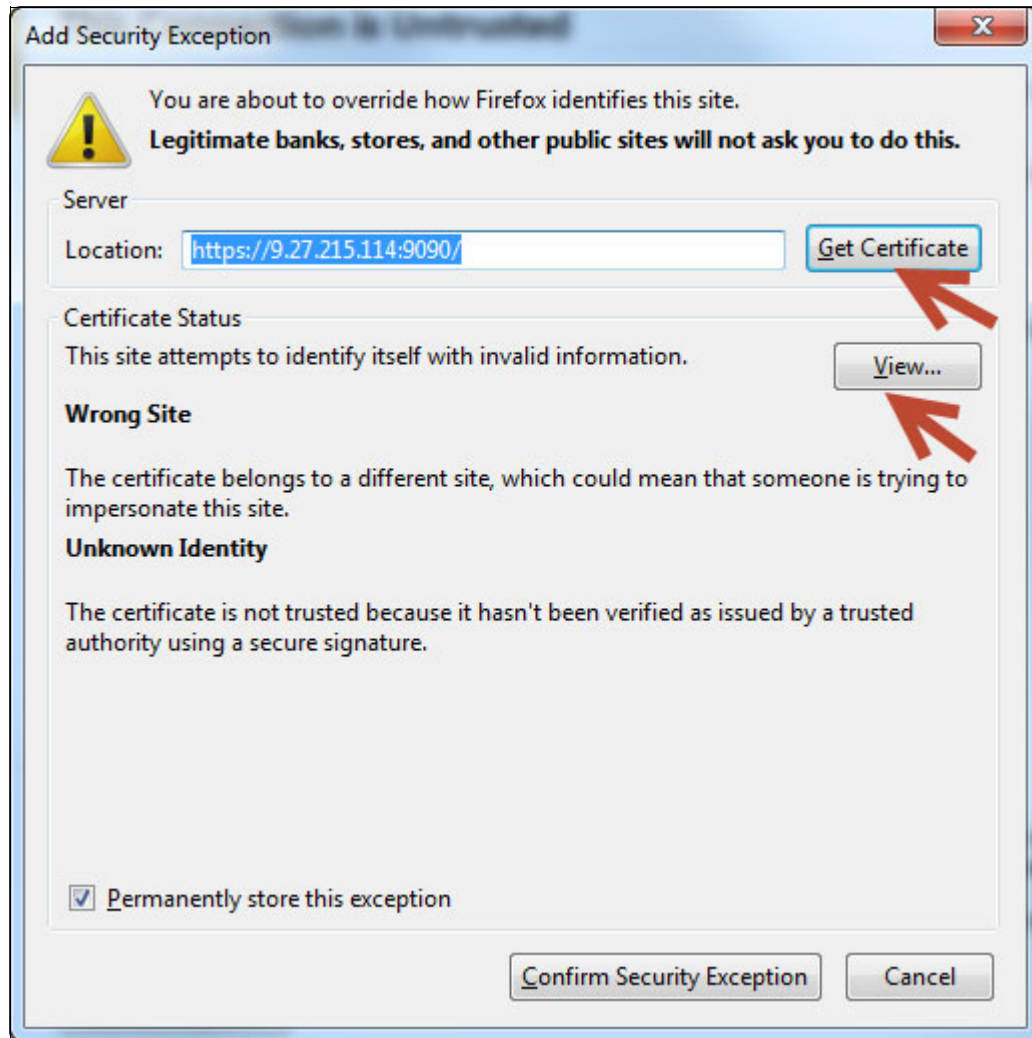


Figure 5-2 Add Security Exception window

Click **Get Certificate** and then **View**. The Certificate Viewer window opens, as shown in Figure 5-3 on page 46. Verify that the certificate that is presented includes Common Name 9006 DataPower WebGUI, and it is Issued By a certificate with Common Name SSL Server CA. The Organization in both certificates is IBM. Check that all other fields in the certificate also match.

Note: If the certificate does not match these figures, the following explanations are possible:

- ▶ IBM might change the certificate that is used by the initial WebGUI. See the IBM MQ Appliance manual for information about the certificate.
- ▶ There might be someone fraudulently imitating the IBM certificate. Investigate the certificate and ensure that you are connecting to the correct URL.

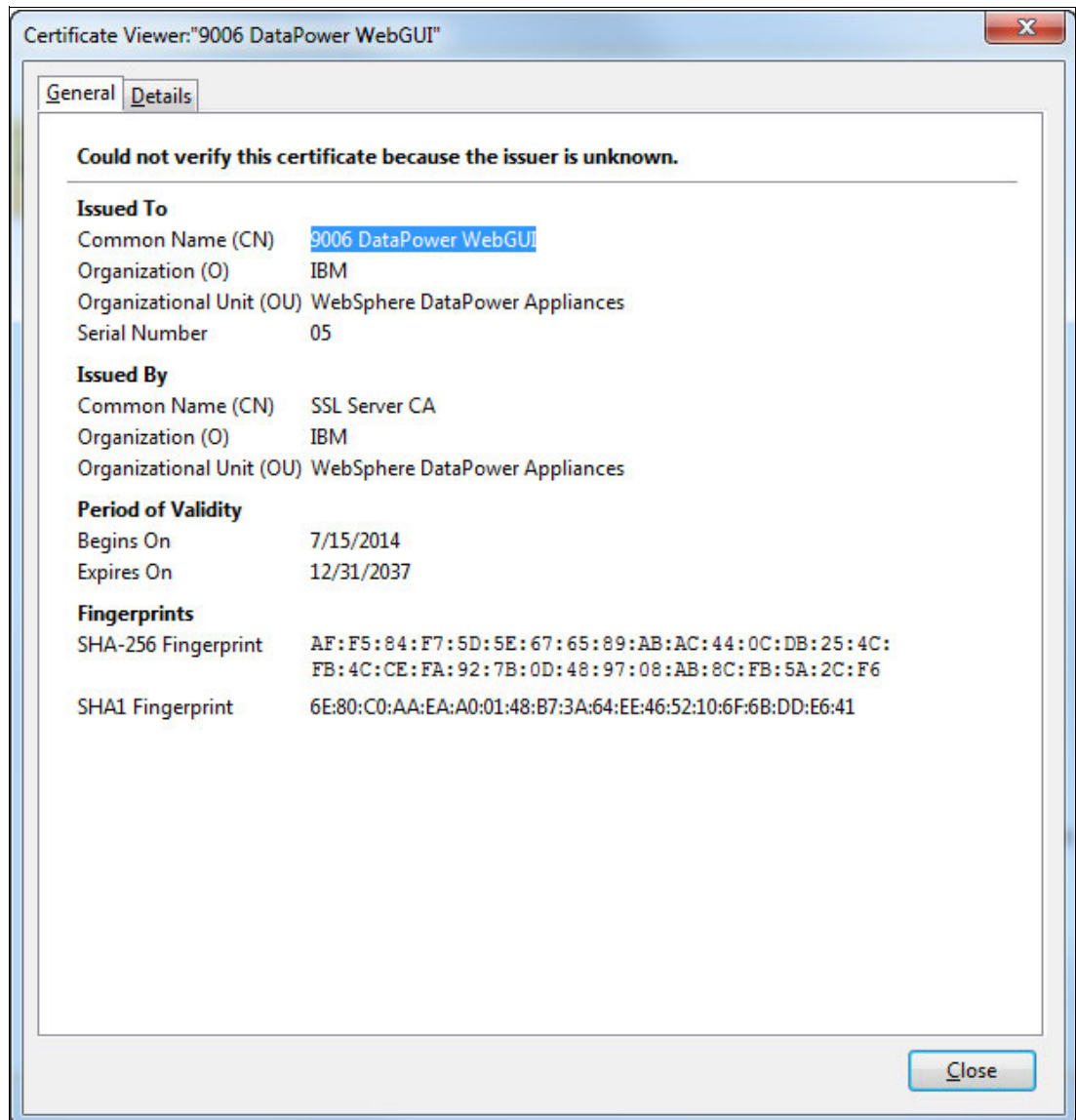


Figure 5-3 Certificate Viewer window

Click **Close**.

The Add Exception dialog is active again. If the certificate you inspected matched the expected values, click **Confirm Security Exception**, as shown in Figure 5-4 on page 47. The web UI login page is displayed.

If the certificate does not match expected values, do not confirm the security exception and click **Cancel**. Do not proceed to the web UI login page until the security issue is resolved.

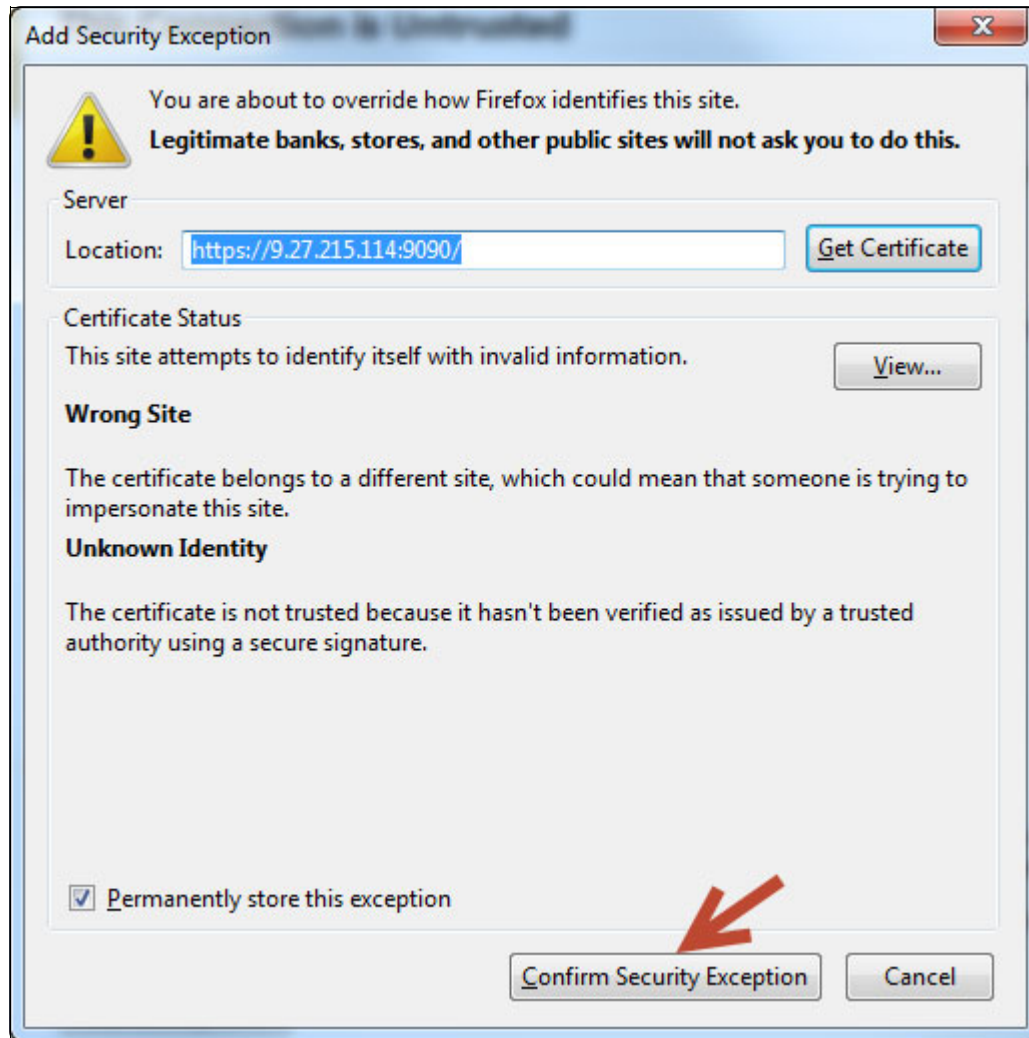


Figure 5-4 Firefox Confirm Security Exception for default certificate

5.6.2 Logging in and accepting the license agreement

Login by using the admin account and the password set during initialization, as shown in Figure 5-5 on page 48.

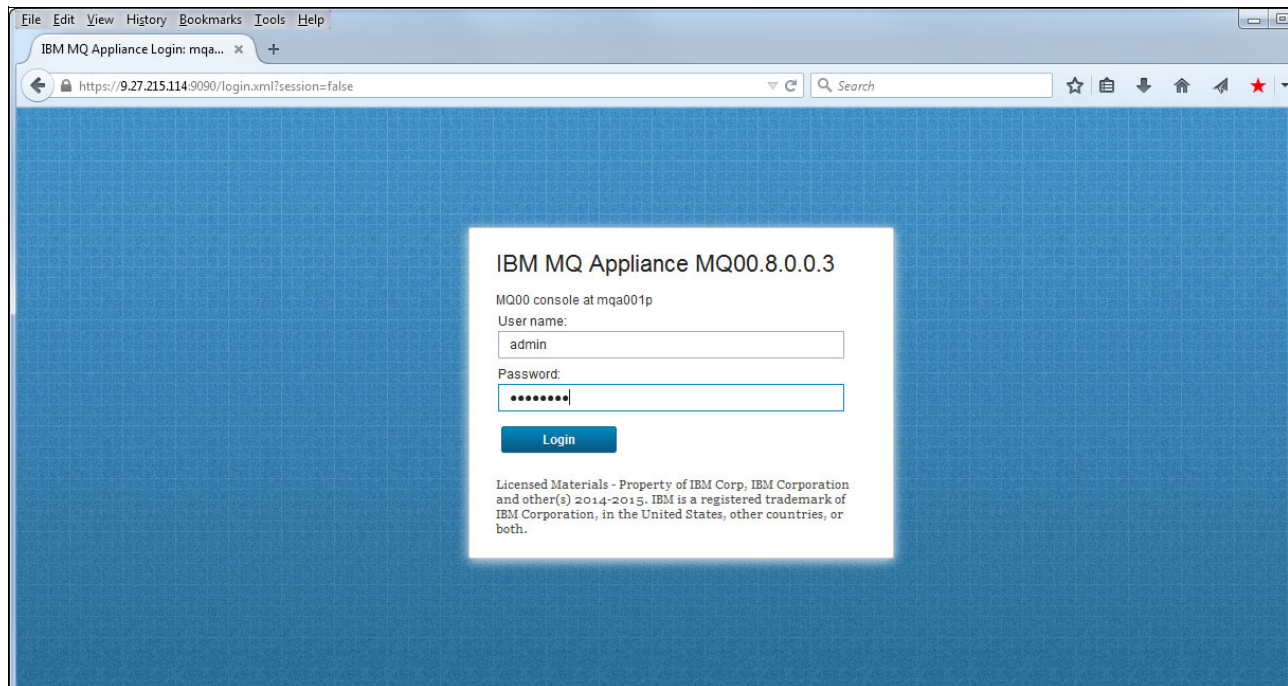


Figure 5-5 Complete the login page

The appliance displays the license terms and conditions, which must be accepted before any further configuration or use of the appliance can continue.

Figure 5-6 on page 49 shows the top of the agreement. After reading the agreement, click **I agree** only if you accept the terms and conditions. If you do not agree, the IBM MQ Appliance does not function.

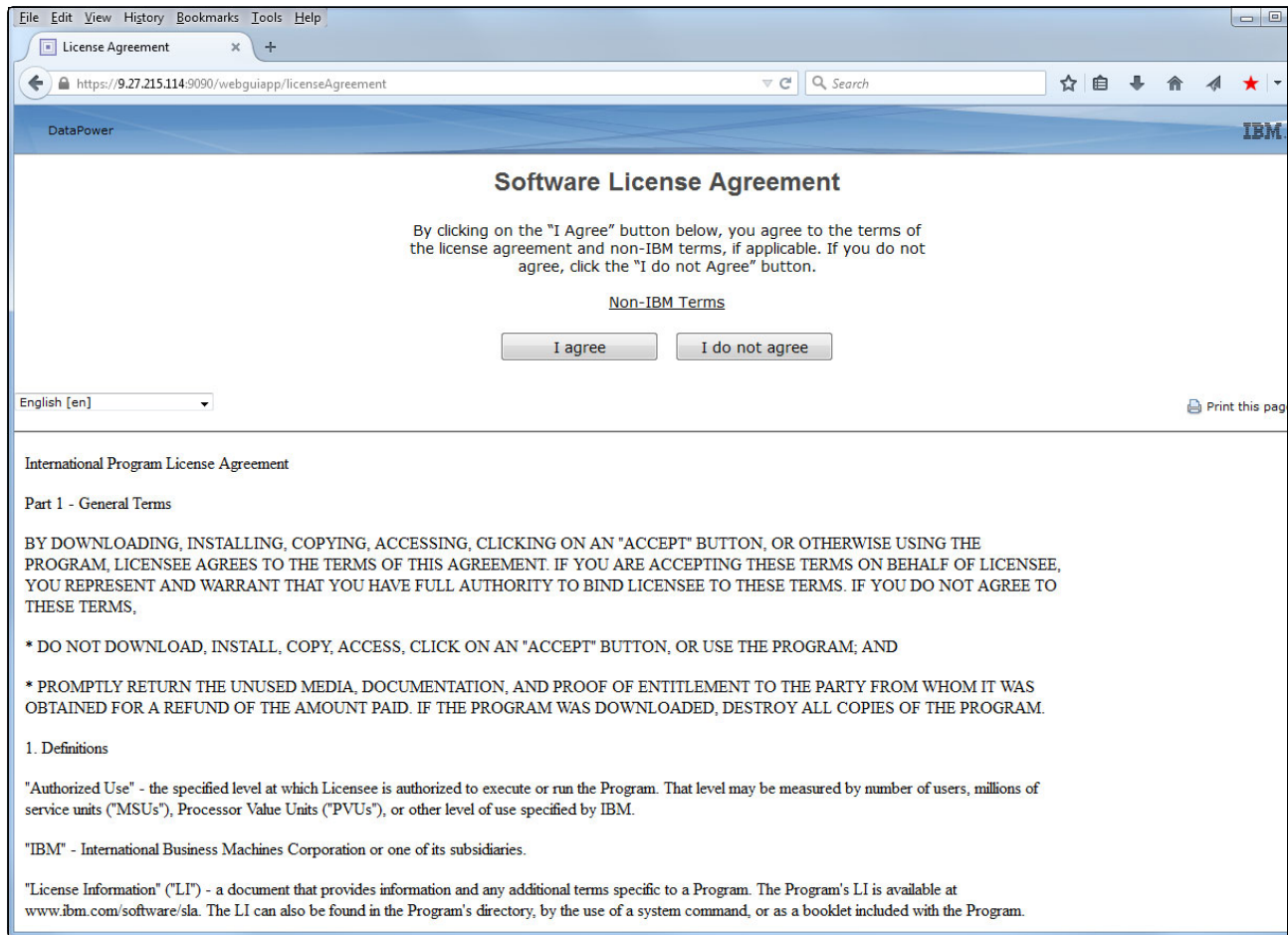


Figure 5-6 License agreement page

Acceptance of the license agreement causes the IBM MQ Appliance to start, which finalizes the initial configuration. Figure 5-7 shows the dialog that indicates that the restart began.

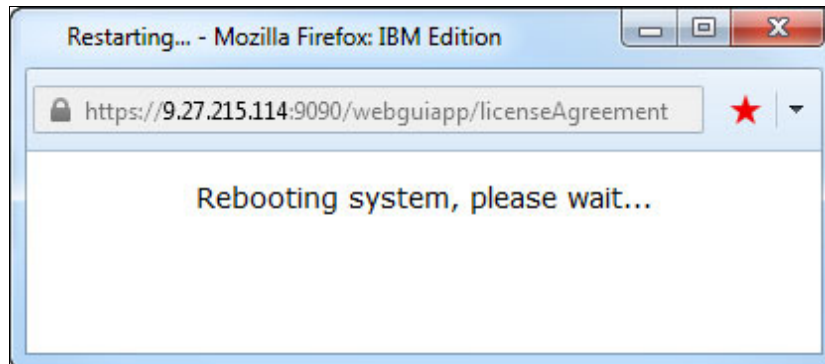


Figure 5-7 Restart dialog presented after the license is accepted

The log in panel is displayed again after the restart process is complete.

5.7 Implementing IPMI

The Intelligent Platform Management Interface (IPMI) is a standard that allows access to a Baseboard Management Controller (BMC) from a remote system. The BMC is powered whenever power is available to the appliance, even if the appliance is powered down. This feature is useful as it enables control over appliance power status remotely. That is, the appliance can be powered up or powered down remotely.

Note: This step is optional, but provides improved ability to remotely control the appliance, which can be useful when problem determination is necessary.

On the IBM MQ Appliance, the BMC shares a physical network connection with the mgt0 interface, which means that mgt0 must be physically cabled to use IPMI.

The normal configuration for IPMI requires the allocation of a second IP address in the same VLAN and subnet as the mgt0 interface. Often it is necessary for the IPMI network address to be configured in the same VLAN and subnet as mgt0 appliance interface because the two interfaces share a network cable and switch port.

It is also possible to use mgt0 only for IPMI, and not configure an address on the appliance mgt0 at all.

Also, it might not be technically necessary for the IPMI address to be in the same subnet as an mgt0 address. The two addresses must be in subnets that are appropriate for the same broadcast domain. The simplest way to ensure that the two addresses are appropriate for the same broadcast domain is to allocate them in the same subnet.

Important: Because of the nature of IPMI and the level of control that can be obtained over the IBM MQ Appliance that uses this interface, connect mgt0 to a secure management network or do not connect it at all.

5.7.1 Creating IPMI objects

IPMI is configured by using the CLI. The following items must be created before IPMI can function:

- ▶ IPMI user
- ▶ IPMI LAN channel

Example 5-10 shows a transcript of the configuration of IPMI.

Example 5-10 Configuration of IPMI

```
M2000# config
M2000(config)# ipmi-user ipmiadmin
New IPMI User configuration
M2000(config ipmi-user ipmiadmin)# user-id 3
M2000(config ipmi-user ipmiadmin)# password ipmiISuseful
M2000(config ipmi-user ipmiadmin)# exit
M2000(config)# ipmi-lan-channel mgt0
New IPMI LAN Channel configuration
M2000(config ipmi-lan-channel mgt0)# ip address 10.101.1.12/24
M2000(config ipmi-lan-channel mgt0)# ip default-gateway 10.101.1.1
M2000(config ipmi-lan-channel mgt0)# allowed-user ipmiadmin operator on 0
```



```
M2000(config ipmi-lan-channel mgt0)# maximum-channel-privilege-level operator
M2000(config ipmi-lan-channel mgt0)# sol-enabled on
M2000(config ipmi-lan-channel mgt0)# sol-required-user-privilege-level user
M2000(config ipmi-lan-channel mgt0)# exit
M2000(config)#
```

Note: Always specify mgt0 as the name of the ipmi-lan-channel. Although the name that is provided on the ipmi-lan-channel command is a variable and the interface accepts any value, the creation of the lan channel fails unless the name is mgt0.

Note: Setting maximum-channel-privilege-level to operator enables IPMI to manage chassis power, which allows remote power down and power up. The IPMI interface address must be protected by a firewall to limit access.

5.7.2 IPMI usage examples

There are several tools available that implement the client side of the IPMI protocol, which allows various IPMI commands to be issued to the BMC. These examples use the open source ipmitool, which is available in most Linux distributions. On Windows, ipmiutil has similar capabilities, but with different syntax.

Both tools display what options are available if an incomplete command is given.

Other tools might be available that have graphical user interfaces, but these tools were not used during the production of this book.

General help in ipmitool is obtained by using the **ipmitool -h** command.

Example 5-11 shows the command to print the System Event Log (SEL).

Example 5-11 Print System Event Log by using ipmitool

```
[root@fedora ~]# ipmitool -I lanplus -U ipmiadmin -a -L OPERATOR -H
mqa001p-ipmi.companyb.local sel
Password:
SEL Information
Version      : 1.5 (v1.5, v2 compliant)
Entries      : 0
Free Space   : 65502 bytes
Percent Used : 0%
Last Add Time : 07/20/2015 18:08:30
Last Del Time : 07/20/2015 18:13:14
Overflow     : false
Supported Cmds : 'Delete' 'Partial Add' 'Reserve' 'Get Alloc Info'
# of Alloc Units : 3639
Alloc Unit Size : 18
# Free Units    : 3639
Largest Free Blk : 3639
Max Record Size : 14
[root@fedora ~]#
```

Example 5-12 shows use of the chassis status command. This command returns information about several critical parameters for the IBM MQ Appliance, such as whether components are powered up or they have faults.

Example 5-12 ipmitool chassis status command

```
[root@fedora ~]# ipmitool -I lanplus -U ipmiadmin -a -L OPERATOR -H
mqa001p-ipmi.companyb.local chassis status
Password:
System Power          : on
Power Overload        : false
Power Interlock       : inactive
Main Power Fault      : false
Power Control Fault   : false
Power Restore Policy  : previous
Last Power Event      : command
Chassis Intrusion     : active
Front-Panel Lockout   : inactive
Drive Fault           : false
Cooling/Fan Fault     : false
Front Panel Control   : none
[root@fedora ~]#
```

Displaying the chassis power status is shown in Example 5-13. The chassis power command can also be used to power the appliance on or off from a remote location.

Example 5-13 Show power status by using ipmitool

```
[root@fedora-casey-fam ~]# ipmitool -I lanplus -U ipmiadmin -a -L OPERATOR -H
mqa001p-ipmi.companyb.local chassis power status
Password:
Chassis Power is on
[root@fedora ~]#
```

The locate LED can be controlled directly by using the ipmitool chassis identify command, as show in Example 5-14.

Example 5-14 Activate locate LED for 15 seconds by using ipmitool

```
[root@fedora ~]# ipmitool -I lanplus -U ipmiadmin -a -L OPERATOR -H
mqa001p-ipmi.companyb.local chassis identify
Password:
Chassis identify interval: default (15 seconds)
[root@fedora ~]#
```

Example 5-15 shows how ipmitool can collect information from all of the sensors in the IBM MQ Appliance. This command produces a large volume of output that is wide and does not fit within a document. Only the command is shown in the example, not the response.

Example 5-15 ipmi sensor command

```
[root@fedora ~]# ipmitool -I lanplus -U ipmiadmin -a -L OPERATOR -H
mqa001p-ipmi.companyb.local sensor
```

The last ipmitool example (as shown in Example 5-16) shows how to access the serial console by using a remote LAN connection. This configuration is called Serial Over LAN (SOL). IPMI SOL can be a useful tool, especially when the configuration of management interfaces is changed. If an error is made that disables network access via the normal administrative interfaces, the serial console can be accessed without the need to travel to the data center and enter the machine room to connect a system by using the physical serial interface.

Example 5-16 Attaching to the serial console by using IPMI SOL command

```
[root@fedora ~]# ipmitool -I lanplus -U ipmiadmin -a -L OPERATOR -H
mqa001p-ipmi.companyb.local sol activate
Password:
[SOL Session operational. Use ~? for help]
M2000(config)# show int
```

interface	IP Address	RX (kb/pkts/errs)	TX (kb/pkts/errs)
-----	-----	-----	-----
lo	127.0.0.1	49/1535/0	49/1535/0
sit0		0/0/0	0/0/0
ip6tnl0		0/0/0	0/0/0
mgt0	10.101.1.10	203/741/0	22/184/0
mgt1	10.101.1.11	51/471/0	7/12/0
eth20		0/0/0	0/0/0
eth21		0/0/0	0/3/0
eth10		0/0/0	0/2/0
eth11		0/0/0	0/0/0
eth12		0/0/0	0/0/0
eth13		0/0/0	0/3/0
eth14		0/0/0	0/0/0
eth15		0/0/0	0/0/0
eth16		0/0/0	0/0/0
eth17		0/0/0	0/3/0

```
M2000(config)# ~. [terminated ipmitool]
[root@fedora ~]#
```

Care: The IPMI SOL capability takes over the physical serial connection. This example does not show the user logging off before disconnecting SOL. If you do not log off before disconnecting SOL, the serial console is still logged on and anyone that connects to the serial console or IPMI SOL has access to the command line without logging in again.

5.8 Next steps

The initial configuration is now complete and the IBM MQ Appliance can respond to administration requests by using the ssh CLI or the IBM MQ Appliance web UI.

However, more configuration tasks must be performed before the appliance is ready to run queue managers or participate in an HA group.

The following items must be configured, as described in Chapter 6, “Appliance administration” on page 55:

- ▶ Ethernet interfaces
- ▶ Link Aggregation interfaces
- ▶ VLAN interfaces
- ▶ Host Aliases
- ▶ DNS Servers
- ▶ Static Hosts
- ▶ NTP servers
- ▶ IBM MQ HA group (if HA is used)
- ▶ Appliance users
- ▶ IBM MQ users (there might not be any users if LDAP is used, or there might still be some who are also Appliance users)
- ▶ Certificate for IBM MQ Appliance web UI
- ▶ Apply improved security controls to ssh and IBM MQ Appliance web UI



Appliance administration

When the IBM MQ Appliance is first installed and powered on, you can administer it by using only a serial port connection to the command-line interface (CLI), as described in Chapter 5, “Initial appliance configuration” on page 37. After you complete the initial setup, you can administer the appliance and IBM MQ on the appliance by using the web user interface (the IBM MQ Appliance Web UI) or the CLI.

The Appliance Web UI is more intuitive, but the CLI is more powerful. If it is possible to complete a task by using the web UI, we decided to document that process. It is not possible to perform some tasks by using the web UI. In these cases, we use the CLI to complete the task.

The IBM MQ Appliance is based on recent versions of the IBM DataPower SOA Gateway Appliances. Parts of the underlying hardware and operating system (OS) are shared between the two appliances; therefore, users who are skilled in DataPower SOA Gateway Appliance administration find elements of IBM MQ Appliance administration familiar. For more information about command references, see the following resources:

- ▶ IBM MQ Appliance Knowledge Center:
<http://www.ibm.com/support/knowledgecenter/SS5K6E/welcome>
- ▶ DataPower Gateway Knowledge Center:
http://www.ibm.com/support/knowledgecenter/SS9H2Y_7.1.0/com.ibm.dp.doc/welcome.html

This chapter guide focuses on new and changed features that are specific to the IBM MQ Appliance.

For users who are unfamiliar with IBM DataPower Gateway, it is important to note that the CLI does not grant access to any normal shell environment; therefore, some tasks that you might expect to complete by using familiar shell-based methods have distinct differences for the IBM MQ Appliance. Many familiar IBM MQ commands are available in mqcli mode from the CLI, although some new commands are implemented there to support the needs of running IBM MQ on an appliance.

In this chapter, the term *configurable* is used. This term refers to options that an administrator can manage. For example, the eth0 network interface is a configurable.

This chapter includes the following topics:

- ▶ 6.1, “IBM MQ Appliance user model”
- ▶ 6.2, “CLI administration” on page 59
- ▶ 6.3, “IBM MQ Appliance Web UI administration” on page 64
- ▶ 6.4, “Network configuration” on page 75
- ▶ 6.5, “Appliance file system” on page 103
- ▶ 6.6, “Firmware upgrades” on page 105

6.1 IBM MQ Appliance user model

There are two distinct types of administrator on the IBM MQ Appliance. These types can be separated by their administrative roles.

It might be helpful to imagine appliance users and groups as “operating system” users and groups. IBM MQ can then be considered a program that is running on the appliance OS with its own messaging users and groups.

To manage high availability (HA) groups, create queue managers, manage certificates, manage messaging users, and more, you must first log in to the operating system or appliance as an appliance administrator. Then, you must enter IBM MQ administration mode.

However, you can connect and administer a queue manager as a messaging user that belongs to a messaging group without needing to log in as an appliance administrator. For more information about connecting and administering queue managers, see Chapter 8, “IBM MQ object security” on page 123 and Chapter 9, “IBM MQ channel security” on page 137.

6.1.1 Appliance administrators

Appliance administrators are authorized to perform system administration tasks, such as firmware upgrades, network interface management, user management, and setting the language, and accessing IBM MQ administration mode.

Creating appliance administrators

On first start, there is only a single account on the appliance, which is named “admin”. To comply with your site security regulations, you might need to create multiple appliance administrators. Figure 6-1 on page 57 shows how to create appliance administrators.

Tip: Use of the default “admin” user is not recommended for the following reasons:

- ▶ If you lose the password, your appliance must be returned to IBM for reset.
- ▶ If multiple people use the same account, you might be violating site security policy.

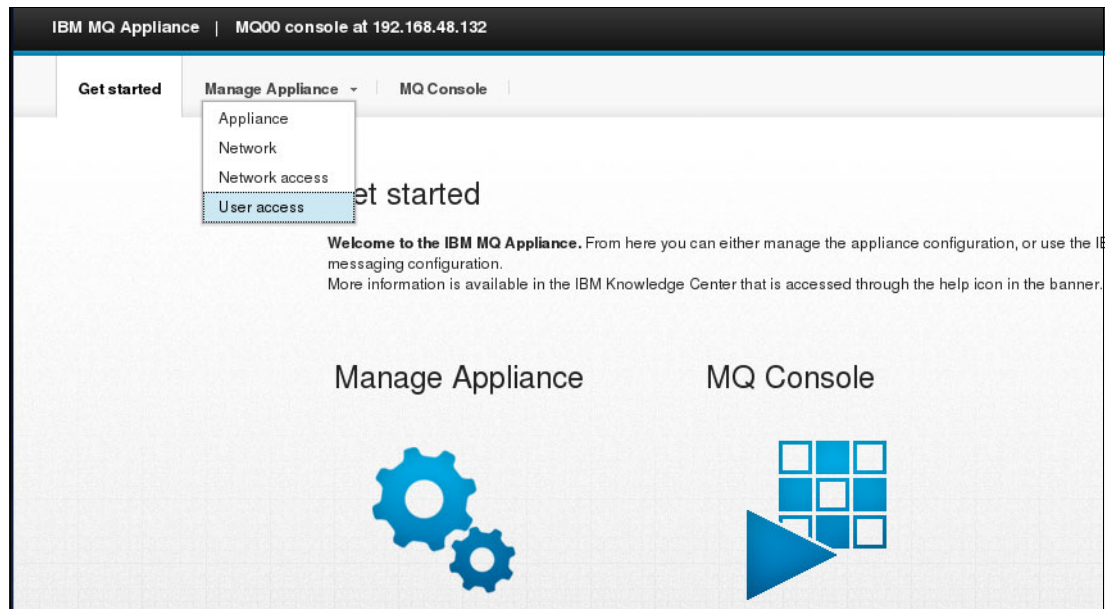


Figure 6-1 Entering the User access section in the web UI

After you open the user access page, you can create a user by clicking **New**. You then see a form, such as the form that is shown in Figure 6-2.

User Account

User Account
CompanyBApplianceAdmin *

Name:

Main

Enable administrative state: ☒

Comments:

Password:

Access level:

Group defined
Privileged
User (deprecated)

Domain restriction (deprecated):

SNMPv3 credentials:

Figure 6-2 Setting up a new account; privileged account access highlighted in the drop-down menu

In Figure 6-2, the Enable administrative state option must be selected if you want to enable the account for use after you save the configuration.

Tip: Throughout the appliance administration process, enable or disable administrative state is used to indicate enabled or disabled configurables.

When you create the user account, assign the account a generic password. The first time the user logs in with that account (through the web UI or CLI), they are prompted to change the password.

If you want to create an account that can be used for appliance and IBM MQ administration tasks, you must change the drop-down menu item Access level: to Privileged. The other option in this drop-down is Group defined, which is not covered in detail here, but allows account privileges to be set according to the group to which they belong.

In the column to the left of the User access page, there is a link to User Group, from where groups can be managed and administered. By creating an appliance user group, the administrative privileges of members can be fine-tuned.

When you complete all of the sections, click **Apply**. The new user should appear in the table that is displayed in the User access page. The final step is to click **Save configuration** in the upper right area of the page.

You are notified about the status of your saved configuration via a small dialog that appears in the lower right area of the web UI page. Figure 6-3 shows a successful configuration change.

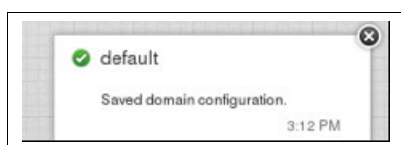


Figure 6-3 Dialog that indicates save configuration success

For any configuration change that is made through the web UI, clicking **Save configuration** persists the change. The change is active before you save it, but disappears if the appliance is restarted. The equivalent in the CLI is to enter `write memory`.

6.1.2 IBM MQ users

You must use IBM MQ configuration mode (`mqcli`) in the CLI for IBM MQ user administration. To access `mqcli` mode, you must first log in to the appliance as an appliance administrator.

IBM MQ users can connect to queue managers remotely to send and receive messages. They can also be authorized to remotely manage some aspects of queue managers through programs, such as IBM MQ Explorer, and `runmqsc` in client mode (new for IBM MQ version 8).

There are two ways to manage your IBM MQ users: You can use an external Lightweight Directory Access Protocol (LDAP) repository, or you can use user and group commands that are built into `mqcli` mode within the CLI.

If you have a configuration of users that you want to replicate on another appliance but do not use LDAP, there are also user backup and restore commands available. However, these commands are outside the scope of this book. The recommended strategy is to use an external LDAP repository for the following advantages:

- ▶ Security administrators need only to apply company security policy in a single place
- ▶ Built-in synchronization between appliances:
 - Machines in the same HA group are guaranteed to have the same user model
 - Production emulation in staging is simplified
- ▶ Far reduced setup times if your company already uses LDAP

For more information about setting up queue managers with an external LDAP repository, see Chapter 7, “Creating queue managers” on page 107 and Chapter 8, “IBM MQ object security” on page 123.

6.2 CLI administration

As the CLI is the first way that an administrator interacts with the appliance, we describe CLI administration before web UI administration in this chapter.

Some tasks are only executable via the CLI, which can be accessed in the following ways:

- ▶ By using a serial port connection directly into the appliance
- ▶ By using ssh if enabled on the target appliance
- ▶ By using Intelligent Platform Management Interface (IPMI), as described in Chapter 5, “Initial appliance configuration” on page 37

Within the CLI, there are several modes and submodes. There are different administration activities available under each mode. Generally under any mode you can perform the following tasks:

- ▶ Enter `help` or `?` to view mode specific help documentation.
- ▶ Enter `help <option>` or `? <option>` to view help specific to that section or command.
- ▶ Enter `top` to return to the top level mode.
- ▶ Enter `exit` to apply configuration and go back up a level.
- ▶ Enter `show` to view the configuration of the current object, or in a higher level mode to view a complete list of commands available. This option is not an option in IBM MQ configuration mode.
- ▶ Enter `show <object>` or `show <object type> <object name>` to view the configuration of an object that was configured in a lower level mode. This option is not an option in IBM MQ configuration mode.
- ▶ Enter `cancel` to revert configuration changes in current mode to previous values and go back up a mode. This option is not an option in IBM MQ configuration mode.
- ▶ Enter `no <object>` or `no <object name>` to delete an object. This option is not an option in IBM MQ configuration mode.
- ▶ Enter `write memory` to persist configurations over a restart. This option is not an option in IBM MQ configuration mode. In IBM MQ configuration mode, all changes are persisted after they are made unless documented otherwise.
- ▶ Entering enough of the mode name for it to be unique is enough to enter that mode. For example, entering `co` enters configuration mode from the top-level mode. However, for readability, we use `config`. This option is not an option in IBM MQ configuration mode.

In the appliance CLI, all non-management traffic is disabled if you make network configuration changes incorrectly. This safeguard avoids security pitfalls in configurations that otherwise can be missed.

A DataPower CLI command reference document is available that describes a superset of the commands that are available on the IBM MQ Appliance. For CLI-based tasks that are not covered by examples in this book, the DataPower document is a good point of reference. However, the DataPower document does not cover commands that are available in IBM MQ administration mode.

The DataPower reference document is available at this website:

ftp://ftp.software.ibm.com/software/integration/datapower/library/prod_docs/4Q2008/XI-3.7.2-CommandReference.pdf

The IBM DataPower Gateway Knowledge Center also includes good reference material and is available at this website:

http://www.ibm.com/support/knowledgecenter/SS9H2Y_7.2.0/com.ibm.dp.doc/welcome.html

CLI configurables

CLI configurables are objects or collections of objects that you can create, manipulate, and link through the CLI. To manipulate a CLI object, you must enter the correct configuration mode. For example, to change properties that are related to the eth0 interface, you must enter the configuration mode and then enter `interface eth0`. From there, you can enter `show` to view the current configuration, and each of the parameters shown can be altered by entering their type directly into the CLI. Examples of CLI object manipulation are shown in the examples throughout this book that are related to securing the Appliance Web UI.

There is a basic rule for creating configurables and a partner rule for deleting configurables through the CLI.

Basic rule for creating configurables through the CLI

In the appliance CLI, there is a general method that you must follow to create any configurable. The basic rule is: you can create only objects that do not have dependencies on other uninitialized objects. Creating objects must start with the most specific object first.

For example, after you have a signed certificate to certify the IBM MQ Appliance Web UI with, you must first put the signed certificates in the right location, then create the certificate object that references the linked certificates as files. Then, you build up the configurations with it in place.

Basic rule for deleting configurables through the CLI

If you want to delete an object, you must follow the opposite rule. The basic rule for deleting objects is: you cannot delete any objects on which other objects depend.

For example, if you want to delete objects that are related to your signed IBM MQ Console certificate, you cannot delete the certificate object first. You must start by deleting the SSL proxy object. You might want to run a `show` command on the SSL proxy object before deleting it, as showing the object shows you the names of all of the referenced objects that you might want to delete next.

An example of deleting configuration is shown in Example 6-1 on page 61 in the context of the Appliance Web UI X.509 certificate. As there are many steps to implementing a certificate for the Web UI, this example was chosen for practical advice concerning deleting configuration in the CLI by using the top-down approach as described in this section.

Note: For more information about a step-by-step guide to secure the Appliance Web UI, see 6.3.2, “Managing the IBM MQ Console” on page 71.

Example 6-1 Deleting CLI objects (here certificate configuration) by using the top-down approach

```
M2000# config
Global configuration mode
M2000(config)# web-mgmt
Modify Web Management Service configuration
M2000(config web-mgmt)# show
  admin-state enabled
  ip-address 0.0.0.0
  port 9191
  ssl WebUISSL [up]
  user-agent default [up]
  save-config-overwrite on
  idle-timeout 600 Seconds
  acl web-mgmt [up]
M2000(config web-mgmt)# no ssl
M2000(config web-mgmt)# show
  admin-state enabled
  ip-address 0.0.0.0
  port 9191
  user-agent default [up]
  save-config-overwrite on
  idle-timeout 600 Seconds
  acl web-mgmt [up]
M2000(config web-mgmt)# exit
M2000(config)# show ssl WebUISSL
sslproxy: WebUISSL [up]
-----
  admin-state enabled
  SSL Direction reverse
  Reverse (Server) Crypto Profile WebUICP [up]
  Server-side Session Caching on
  sess-timeout 300 seconds
  cache-size 20 entries (x 1024)
  client-cache on
  client-sess-timeout 300 seconds
  client-cache-size 100 entries
  client-auth-optional off
  client-auth-always-request off
  permit-insecure-servers off
M2000(config)# no sslproxy WebUISSL
sslproxy WebUISSLProxy - Configuration deleted.
M2000(config)# crypto
Crypto configuration mode
M2000(config-crypto)# profile WebUICP
Modify Crypto Profile configuration
M2000(config profile WebUICP)# show
  admin-state enabled
  idcred WebUIIDCred [up]
  ciphers HIGH:MEDIUM:!aNULL:!eNULL:@STRENGTH
  option-string
  OpenSSL-default+Disable-SSLv2+Disable-SSLv3+Enable-Legacy-Renegotiation
  clientcalist off
M2000(config profile WebUICP)# exit
M2000(config-crypto)# no profile WebUICP
profile WebUICP - Configuration deleted.
```

```

M2000(config-crypto)# idcred WebUIIDCred
Modify Crypto Identification Credentials configuration
M2000(config idcred WebUIIDCred)# show
  admin-state enabled
  key WebUIKey [up]
  certificate WebUICert [up]
  ca CompanyBCA [up]
M2000(config idcred WebUIIDCred)# exit
M2000(config-crypto)# no idcred WebUIIDCred
idcred WebUIIDCred - Configuration deleted.
M2000(config-crypto)# key WebUIKey
Modify Crypto Key configuration
M2000(config key WebUIKey)# show
  admin-state enabled
  file-name cert:///WebUI-privkey.pem
  password-alias on
M2000(config key WebUIKey)# exit
M2000(config-crypto)# no key WebUIKey
key WebUIKey - Configuration deleted.
M2000(config-crypto)# show certificate WebUICert
certificate: WebUICert [up]
-----
  admin-state enabled
  file-name cert:///mqa002pWebUI.cer.pem
  password-alias off
  ignore-expiration on
M2000(config-crypto)# no certificate WebUICert
certificate WebUI - Configuration deleted.
M2000(config-crypto)# show certificate CompanyBCA
certificate: CompanyBCA [up]
-----
  admin-state enabled
  file-name cert:///mqa002pWebUI.cer.pem
  password-alias off
  ignore-expiration off
M2000(config-crypto)# no certificate CompanyBCA
certificate CompanyBCA - Configuration deleted.
M2000(config-crypto)# exit
Exiting Crypto Configuration mode
M2000(config)# dir cert://

```

File Name	Last Modified	Size
mqa002pWebUI.cer.pem	Jul 28, 2015 3:42:02 PM	1253
WebUI-privkey.pem	Jul 28, 2015 3:33:05 PM	1834
CompanyBCA.cert.pem	Jul 28, 2015 1:59:04 PM	1521

```

  12912.1 MB available to cert://
M2000(config)# delete cert:///WebUI-privkey.pem
File deletion successful
M2000(config)# delete cert:///CompanyBCA.cert.pem
File deletion successful
M2000(config)# delete cert:///mqa002pWebUI.cer.pem
File deletion successful

```

Configuration mode

You can enter configuration mode by entering `config` from the top-level CLI. Within configuration mode, you can perform appliance mode administration tasks. The scope of this book cannot cover every aspect of the configuration mode. However, throughout the book use of configuration mode is documented in examples. See these examples for more information about configuration mode, such as adding a custom certificate to the appliance web UI and upgrading the firmware level.

IBM MQ administration mode

You can enter IBM MQ administration mode by entering `mqcli` from the top-level CLI. In IBM MQ administration mode, you can access CLI tools that are familiar to IBM MQ administrators. Example 6-2 shows all of the commands that are available in `mqcli` mode. The help documentation for commands is grouped into several topics to help aid readability. However, all commands can be accessed under the basic IBM MQ administration mode (there are no submodes).

Example 6-2 Command topics available in IBM MQ administration mode

```
M2000# mqcli
M2000(mqcli)# help
The following help topics are available. Type help <topic name> for more
information.

mq          General IBM MQ administration commands
cert       Channel security certificate administration commands
diag       IBM MQ problem diagnosis commands
ha         High availability administration commands
user       Messaging user and group administration
```

Basic IBM MQ commands are shown in Example 6-3.

Example 6-3 Basic IBM MQ commands available in IBM MQ administration mode

```
M2000# mqcli
M2000(mqcli)# help mq
The following IBM MQ commands are available. Type help <command> for more
information.

crtmqm      Creates a queue manager
dlmqm      Deletes a queue manager
dspmq       Displays information about queue managers
dspmqini    Displays queue manager initialization parameters
dspmqtrn    Displays in-doubt and heuristically completed transactions
dspmqvar    Displays global and queue manager environment variables
endmqm      Stops a queue manager
rsvmqtrn    Resolves in-doubt and heuristically completed transactions
runmqsc     Runs MQSC commands
runswchl    Switches transmission queue for cluster channel
setmqini    Configures queue manager initialization parameters
setmqvar    Configures global and queue manager environment variables
status      Displays status information
strmqm      Starts a queue manager
```

Diagnostic commands are shown in Example 6-4.

Example 6-4 Diagnostic commands available in IBM MQ administration mode

```
M2000# mqcli
M2000(mqcli)# help diag
The following diagnostic commands are available. Type help <command> for more
information.
```

dltmgras	Deletes IBM MQ Appliance diagnostic information
dmpmqcfg	Dumps the configuration of a IBM MQ Appliance queue manager
dspmqrerr	Displays error logs and FDC files
dspmqrte	Determines the route that a message has taken through a queue manager network
dspmqrver	Displays IBM MQ Appliance version and build information
endmqtrc	Ends trace for some or all of the entities that are being traced
mqrcc	Displays information about return codes
runmgras	Gathers IBM MQ Appliance diagnostic information together into a single archive, for example to submit to IBM Support
strmqtrc	Enables trace at a specified level of detail, or reports the level of tracing in effect

Most of the basic IBM MQ control commands have equivalent functions that are available from the IBM MQ Console area in the Appliance Web UI. The Web UI is described next.

6.3 IBM MQ Appliance Web UI administration

By default, you can access the IBM MQ Appliance Web UI through a secure https connection to the configured IP address at port 9090 (<https://<IP Address>:9090>). You must accept the license the first time that you start the web UI, as described in Chapter 5, “Initial appliance configuration” on page 37.

However, you might not want to use port 9090. After you secure the appliance web UI, the port number that it runs through it can be changed.

6.3.1 Securing the appliance web UI

When the appliance web UI is first accessed, a self-signed certificate is used. Securing the appliance web UI is done by using certificates over several steps that use the CLI.

If you find you made an error while attempting to secure the appliance web UI, see 6.2, “CLI administration” on page 59 for the example of how to delete CLI configurables.

For more information about securing the appliance, see the following resources (the steps to secure the appliance web UI are not included in these resources):

- ▶ https://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/security/se00000_.htm
- ▶ https://www.ibm.com/support/knowledgecenter/SS9H2Y_7.1.0/com.ibm.dp.doc/securingcommunication.html

You must use a certificate from your certificate authority (CA) to certify the Appliance Web UI for your organization. For your CA to generate a unique certificate for the Appliance Web UI, you must generate a public/private key pair and a certificate signing request for each appliance.

To generate a certificate signing request, you can use your plain text password or generate a password alias. In the next section, we describe how to generate a password alias. If a plain text password is used, the appliance configuration file might contain the password for the private key in a visible form.

Creating a password alias

A password alias links your password to a plain text alias, which can be encoded into the certificate request. Creating a password alias is required if you do not want to use your password in plain text in the certificate request. We use the CLI to generate a certificate request, as shown in Example 6-5.

Example 6-5 Generating a password alias

```
M2000# config
Global configuration mode
M2000(config)# crypto
Crypto configuration mode
M2000(config-crypto)# password-map

Please enter alias-name and plaintext passwords pairs
Enter a blank alias name to finish

Alias-name: WebUIPass
Plaintext password: *****
Re-enter plaintext password: *****

Alias-name:
Password-map updated (1 entry)
M2000(config-crypto)#
```

Generating a key

After you generated a password alias, you can use the keygen function to generate a private key, the matching public key, and a certificate signing request. The **keygen** command takes key-value pairs that are separated by spaces, as shown in Example 6-6. This example assumes that a password alias named WebUIPass is set up, as shown in Example 6-5.

Example 6-6 Generating a public/private key pair and a certificate signing request

```
M2000# config
Global configuration mode
M2000(config)# crypto
Crypto configuration mode
M2000(config-crypto)# keygen C US L Raleigh ST NC O CompanyB OU SA-W525 CN
mqa001p.companyb.local rsa 2048 gen-object object-name WebUIKey file-name WebUI
export-key password-alias WebUIPass
keygen : Generating a 2048 bit RSA private key
keygen : This could take some time, please be patient
keygen : Saved private key in 'cert:///WebUI.pem-privkey.pem'
keygen : Saved private key in 'temporary:///WebUI.pem-privkey.pem' [INSECURE
DIRECTORY]
```

```

M2000(config)# copy
scp://rrussell@mqsupportsvr.companyb.local//var/tmp/mqa002pWebUI.cer
cert://mqa002pWebUI.cer.pem
Password: *****
File copy success
M2000(config)# dir cert://

```

File Name	Last Modified	Size
mqa002pWebUI.cer.pem	Jul 28, 2015 3:42:02 PM	1253
WebUI-privkey.pem	Jul 28, 2015 3:33:05 PM	1834
CompanyBCA.cert.pem	Jul 28, 2015 1:59:04 PM	1521

```

10599.3 MB available to cert://

M2000(config)# crypto
Crypto configuration mode
M2000(config-crypto)# certificate CompanyBCA
New Crypto Certificate configuration

M2000(config certificate CompanyBCA)# show
admin-state enabled
password-alias off
ignore-expiration off
M2000(config certificate CompanyBCA)# file-name cert:///CompanyBCA.cer.pem
M2000(config certificate CompanyBCA)# exit
M2000(config-crypto)# certificate WebUICert
New Crypto Certificate configuration

M2000(config certificate WebUICert)# show
admin-state enabled
password-alias off
ignore-expiration off
M2000(config certificate WebUICert)# file-name cert:///mqa002pWebUI.cer.pem
M2000(config certificate WebUICert)# exit
M2000(config-crypto)# show certificate CompanyBCA

certificate: CompanyBCA [up] (new)
-----
admin-state enabled
file-name cert://CompanyBCA.cer.pem
password-alias off
ignore-expiration off
M2000(config-crypto)# key WebUIKey
Modify Crypto Key configuration

M2000(config key WebUIKey)# show
admin-state enabled
file-name cert:///WebUI-privkey.pem
password-alias on
M2000(config key WebUIKey)# exit
M2000(config-crypto)#

```

As shown in Example 6-7 on page 66, the certificates are copied onto the appliance. The certificates are then linked to certificate objects in crypto mode in the CLI. A similar object is also created for the private key. The use of these objects simplifies certificate and key management by abstracting the physical filename away from other configuration objects, which makes the configuration more portable and less prone to errors.

The admin-state commands that are shown in Example 6-7 on page 66 requested that the state of the key and certificate objects is [up]. This state is what you should see if your certificate was accepted. If the state of either certificate is not up, examine your certificate file for errors or corruption and ensure that the entire certificate is in the file, including the BEGIN CERTIFICATE and END CERTIFICATE lines. If the key object is not up, the most likely cause is an incorrect password.

After a key object for the key and a certificate object for each certificate is created, identification credentials can be associated with these certificate objects.

Creating an identification credential

An identification credential object groups the linked certificate and key objects that are set up in Example 6-7 on page 66. In the CLI, identification credential is shortened to idcred.

Identification credentials associate the public and private keys that the Web UI uses for SSL authentication, as shown in Example 6-8.

Example 6-8 Creating the identification credential object

```
M2000(config)# crypto
Crypto configuration mode
M2000(config-crypto)# idcred WebUIIDCred
New Crypto Identification Credentials configuration

M2000(config idcred WebUIIDCred)# show
  admin-state enabled
M2000(config idcred WebUIIDCred)# certificate WebUICert
M2000(config idcred WebUIIDCred)# key WebUIKey
M2000(config idcred WebUIIDCred)# ca CompanyBCA
M2000(config idcred WebUIIDCred)# show
  admin-state enabled
  key WebUIKey  [up]
  certificate WebUICert  [up]
  ca CompanyBCA  [up]
M2000(config idcred WebUIIDCred)# exit
M2000(config-crypto)#
```

After identification credentials are established and you ensured that all of your new certificate and key objects are reporting that they are in [up] state (as shown in Example 6-8), you can create a cryptographic profile object. You must associate the ID credentials that you created with this crypto profile object.

Creating a cryptographic profile

Cryptographic profiles identify a collection of SSL resources that support secure connections with remote peer appliances, or with web browsers that are used to administer the appliance. The crypto profile object allows users to customize options, such as cipher strength, which we change to high only for increased security.

You must assign your identification credentials to your cryptographic object. You can also assign validation credentials, which validate the authenticity of received certificates and digital signatures. You can create a validation credential object in the CLI by using `valcred <valcredName>` from cryptographic mode. A validation credential object is created at the same level as the identification credential object, as shown in Example 6-9.

Example 6-9 Creating a cryptographic profile through the CLI

```
M2000(config-crypto)# profile WebUICP
New Crypto Profile configuration

M2000(config profile WebUICP)# show
  admin-state enabled
  ciphers HIGH:MEDIUM:!aNULL:!eNULL:@STRENGTH
  option-string OpenSSL-default+Disable-SSLv2+Disable-SSLv3
  clientcalist off
M2000(config profile WebUICP)# idcred WebUIIDCred
M2000(config profile WebUICP)# ciphers HIGH:!MEDIUM:!aNULL:!eNULL:@STRENGTH
M2000(config profile WebUICP)# show
  admin-state enabled
  ciphers HIGH:!MEDIUM:!aNULL:!eNULL:@STRENGTH
  option-string OpenSSL-default+Disable-SSLv2+Disable-SSLv3
  clientcalist off
M2000(config profile WebUICP)# exit
M2000(config-crypto)# exit
Exiting Crypto Configuration mode
```

Creating an SSL proxy profile

The SSL proxy profile defines SSL behavior for the appliance. For more information, see this website:

https://www.ibm.com/support/knowledgecenter/SS9H2Y_7.1.0/com.ibm.dp.doc/sslproxyprofile.html

Initializing the SSL proxy profile requires a name and that you specify a cryptographic profile object, which for this set of examples is WebUICP, as shown in Example 6-10.

Example 6-10 Creating the SSL proxy profile and associating it with the crypto profile object

```
M2000(config)# sslproxy WebUISSL server WebUICP
Creating SSL Proxy Profile 'WebUISSL'

M2000(config)# show sslproxy WebUISSL

sslproxy: WebUISSL [up]
-----
  admin-state enabled
  SSL Direction reverse
  Reverse (Server) Crypto Profile WebUICP  [up]
  Server-side Session Caching on
  sess-timeout 300 seconds
  cache-size 20 entries (x 1024)
  client-cache on
  client-sess-timeout 300 seconds
  client-cache-size 100 entries
  client-auth-optional off
```

```
client-auth-always-request off
permit-insecure-servers off
```

```
M2000(config)#
```

Designating IBM MQ Appliance Web UI to use the new SSL proxy object

After the SSL proxy object is created and reporting [up] status, the final step is to associate it with the IBM MQ Appliance Web UI. This association is done by entering web-mgmt mode, which is found under config mode, and carrying out the following commands, as shown in Example 6-11.

Example 6-11 Switching the SSL proxy object to the new object

```
M2000(config)# web-mgmt
Modify Web Management Service configuration

M2000(config web-mgmt)# ssl WebUISSL
M2000(config web-mgmt)# user-agent default
M2000(config web-mgmt)# local-address WebUI-interface
M2000(config web-mgmt)# show
  admin-state enabled
  ip-address WebUI-interface
  port 9090
  ssl WebUISSL [up]
  user-agent default [up]
  save-config-overwrite on
  idle-timeout 600 Seconds
  acl web-mgmt [up]
M2000(config web-mgmt)# exit
M2000(config)# show web-mgmt

web-mgmt [up] (modified)
-----
  admin-state enabled
  ip-address WebUI-interface
  port 9090
  ssl WebUISSL [up]
  user-agent default [up]
  save-config-overwrite on
  idle-timeout 600 Seconds
  acl web-mgmt [up]
```

After the web-mgmt status is [up], the IBM MQ Appliance Web UI presents a valid certificate, which correctly represents the IP address or host name that is used to access it and signed appropriately for your organization.

Checking for success and persisting changes

If the command **show web-mgmt** prints a status of [up] (modified), as shown in Example 6-11 on page 70, you might want to persist your changes so that they stay over a restart. If you send the command **write memory** to the CLI in configuration mode, you can save your changes, as shown in Example 6-12 on page 71.

```
M2000(config)# show web-mgmt
```

```
web-mgmt [up] (modified)
```

```
-----
```

```
admin-state enabled
ip-address 9.20.87.135
port 9090
ssl WebUISSL [up]
user-agent default [up]
save-config-overwrite on
idle-timeout 600 Seconds
acl web-mgmt [up]
```

```
M2000(config)# write memory
```

```
Overwrite previously saved configuration? Yes/No [y/n]: y
```

```
Configuration saved successfully.
```

```
M2000(config)#
```

After you correctly configure the IBM MQ Appliance Web UI with your certificate, you can perform appliance administration tasks, such as changing the Web UI port from within the Web UI.

Many of these tasks are described in their specific sections of this book; for example, setting up network interfaces is described in 6.4, “Network configuration” on page 75. However, a brief introduction to the IBM MQ Console is provided in this section. The IBM MQ Console is accessed from the IBM Appliance Web UI.

6.3.2 Managing the IBM MQ Console

The IBM MQ Console is a new, web-based way of administering IBM MQ. You can perform the following tasks by using the IBM MQ Console:

- ▶ Manage the following components:
 - Queue managers (including viewing and downloading error logs)
 - Client-connection channels
 - Queues
 - Channels
 - Topics
 - Listeners
 - Channel authentication records
- ▶ View live IBM MQ charts and analytics through chart widgets
- ▶ Save and load customizable dashboards by using JSON files

The IBM MQ Console is linked from the home page of the IBM MQ Appliance Web UI, as shown in Figure 6-4.

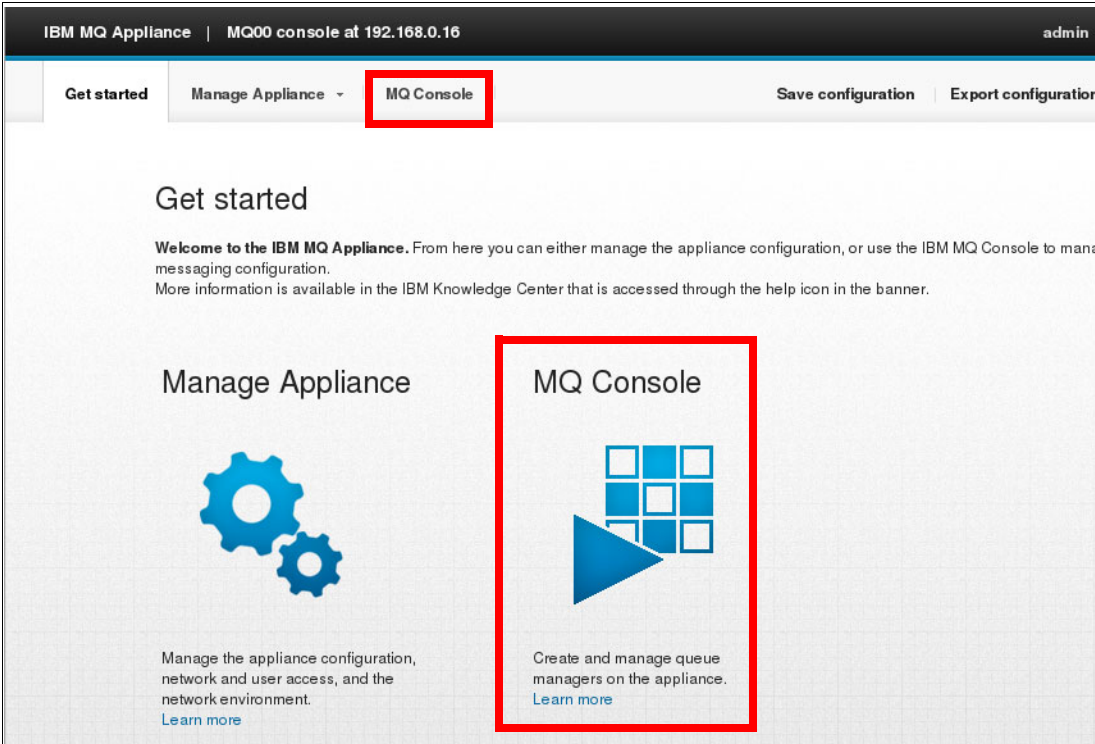


Figure 6-4 Accessing the IBM MQ Console

Dashboard

The IBM MQ Console introduces user dashboards, which include support for chart widgets that are based on live data and basic IBM MQ administration, as shown in Figure 6-5.

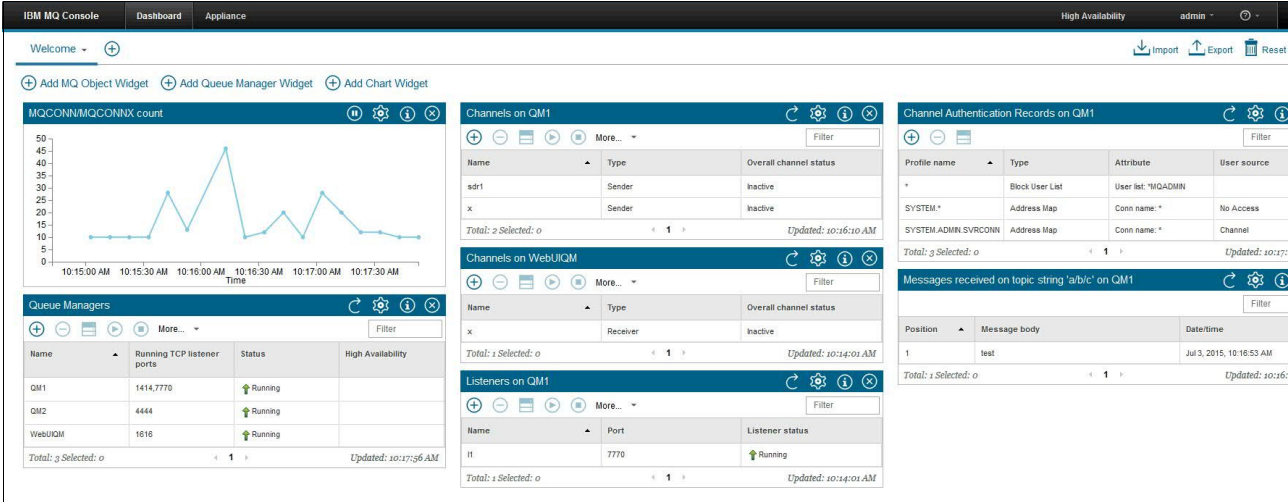


Figure 6-5 Example IBM MQ Console dashboard

Dashboard configurations can be imported to and exported from tabs on the dashboard. In our scenario, this ability can be useful to ensure that both appliances in an HA pair have the same dashboard.

In the following example, a chart widget is created, the dashboard configuration JSON(JavaScript Object Notation) file is exported and examined, and the file is then imported onto the second appliance dashboard. Complete the following steps:

1. Create a tab on the IBM MQ dashboard by clicking + next to the default Welcome tab, which we named Charts by using the tabs drop-down menu option, as shown in Figure 6-6.

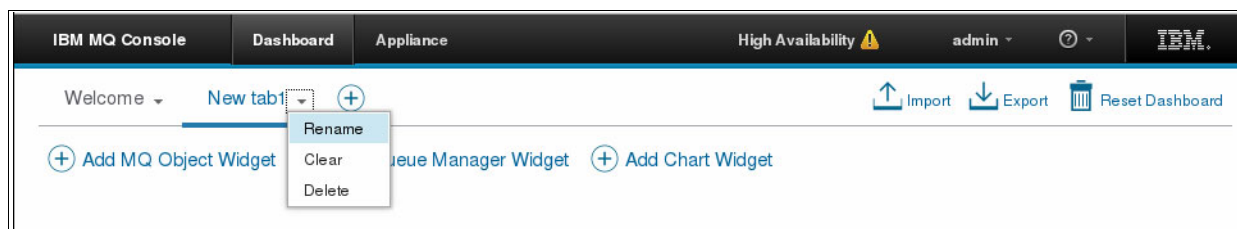


Figure 6-6 Creating a tab

2. Click + next to Add Chart Widget and chose the chart that is wanted, as shown in Figure 6-7.

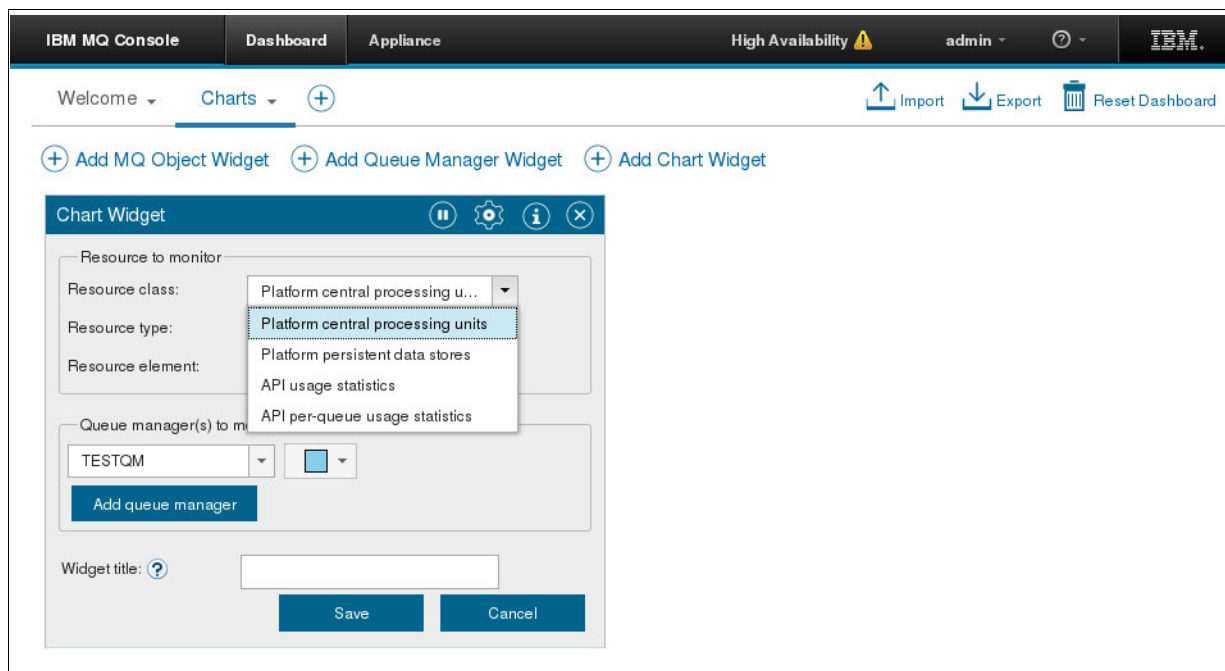


Figure 6-7 Chart widget

The following options are available:

- ▶ The User and System CPU and RAM usage system wide or per queue manager
- ▶ Data persisted on disk, including:
 - Statistics for various appliance volumes, including errors volume and first failure system trace/failure data capture (FFST/FDC) count. If errors are ramping up, this condition can be an early warning or act as a health check.
 - Usage statistics for selected queue manager.

- ▶ IBM MQ API usage including statistics for the following components:
 - MQCONN and MQDISC
 - MQOPEN and MQCLOSE
 - MQINQ and MQSET
 - MQPUT
 - MQGET
 - Commit and rollback
 - Subscribe count
 - Publish
- ▶ IBM MQ API usage per queue manager:
 - MQOPEN and MQCLOSE
 - MQINQ and MQSET
 - MQPUT and MQPUT1 (MQPUT1 puts a single message)
 - MQGET

Many statistics are available, which can help you to easily monitor queue manager health. The term *platform* was used instead of *appliance* in the widget selection menus to ensure the IBM MQ Web UI is platform-independent.

You must choose a target queue manager for the widget, and can optionally name the chart. If you do not name the chart, by default it is assigned a descriptive name according to its type. If you are experiencing problems choosing a queue manager for the chart widget, ensure that you have at least one running (and HA active) queue manager on the appliance you are administering.

After you successfully create the widget, it is gradually populated with data that was collected since it was created. If you create more than one widget, you can click and drag widgets around the dashboard as needed.

To export your configuration to the paired appliance or to other appliances that need similar configuration, you can use the Export function that is available in the upper right of the IBM MQ dashboard. This function downloads a JSON object that contains all configuration for all of the tabs. The file is named `MQConsole-dashboard-<username>.json`.

This file can then be uploaded through the IBM MQ Web UI to load the same dashboard on another appliance, which is useful if you want to load the same configuration across many appliances or many Web UI instances.

If you want to have similar dashboard configurations but different queue manager names, it might be worth manually editing the JSON file and replace instances of the queue manager name with the queue manager that is wanted on the target appliance. For large-scale rollouts of the dashboard, it might be best to create a JSON template of your dashboard with appropriate tags for queue manager names, which can then be used to create many JSON configuration files with different queue manager names.

6.4 Network configuration

It is essential to position the appliance correctly within your company firewall so that it can successfully connect to everywhere it needs to (for more information, see Chapter 3, “Planning” on page 21).

In our scenario, the HA pair of appliances is acting as a gateway into the cluster. Therefore, firewall rules must be set up such that any applications that client connect to the queue managers on the appliances can reach the IBM MQ listener port. Other cluster members also must reach the appliance on the queue manager listener port, and queue managers on the appliance must reach the listeners for other queue managers in the IBM MQ cluster.

Configuring the networking components of the IBM MQ Appliance include the following major elements:

- ▶ Network interfaces
- ▶ Appliance visible interface names
- ▶ External services that the appliance uses over the network
- ▶ Appliance provided services.

6.4.1 Configure network interfaces

There are several ways that network interfaces can be configured. The different options can provide higher bandwidth, network interface redundancy, connectivity flexibility, or a combination of these attributes.

Higher bandwidth can be achieved by using the eth20 Ethernet interface in 10 Gb mode. An SFP+ SR transceiver ships with the appliance, which can be used with OM3 or OM4 cabling to connect this interface. However, the second 10 Gb interface (eth21) is reserved for use as the replication interface for the HA group, which means that network failover and network HA cannot be implemented by using the 10 Gb interfaces.

Network interface redundancy is achieved by connecting several (two or more) of the six available 1 Gb Ethernet interfaces. The interfaces should be connected to at least two separate switches. For maximum redundancy and the least impact if there is a switch failure, each interface should ideally be connected to a separate switch.

If supported in your organization, the switch fabric should operate with a Virtual Link Aggregation (VLAG). This VLAG makes interface failover faster, as the switch fabric does not need to perform spanning tree re-convergence.

Rather than individually configuring each of the Ethernet interfaces, a Link Aggregation Group (LAG) is created. There are several options for protocols that can be used to implement a LAG; however, Link Aggregation Control Protocol (LACP) provides the best behavior if it is supported by your switch fabric. The Link Aggregation interface is then configured with the IP address and routing information.

Finally, flexibility to configure network interfaces into multiple Virtual local area networks (VLANs) can be implemented by creating VLAN interfaces, which use Ethernet interfaces or Link Aggregation interfaces.

In this chapter, we describe the process to configure highly available networking that uses LACP and native VLAN tagging. This process requires correct configuration of the switch ports that the appliance is connected to, but the switch port configuration is not shown.

After the interfaces are configured, it is important to consider the routing that is needed. The simplest configuration is to set up a default route on the aggregated data interface, and remove the default route from the management interfaces. A high metric (that is, low preference) static route is added to the management interfaces so that traffic that is arriving at them can be routed back correctly.

Configure Ethernet interfaces

There are six available 1 Gb Ethernet network interfaces on the IBM MQ Appliance; eth13 and eth17 cannot be used for normal networking because they are reserved for HA group usage.

Each of these interfaces is configured in the same way by using the IBM MQ Appliance Web UI. In this section, only the configuration of eth10 is shown. Each of the other interfaces, eth11, eth12, eth14, eth15, and eth16 must be configured in the same way. Figure 6-8 shows the selection of the Network menu to access the Web UI Network Configuration page.

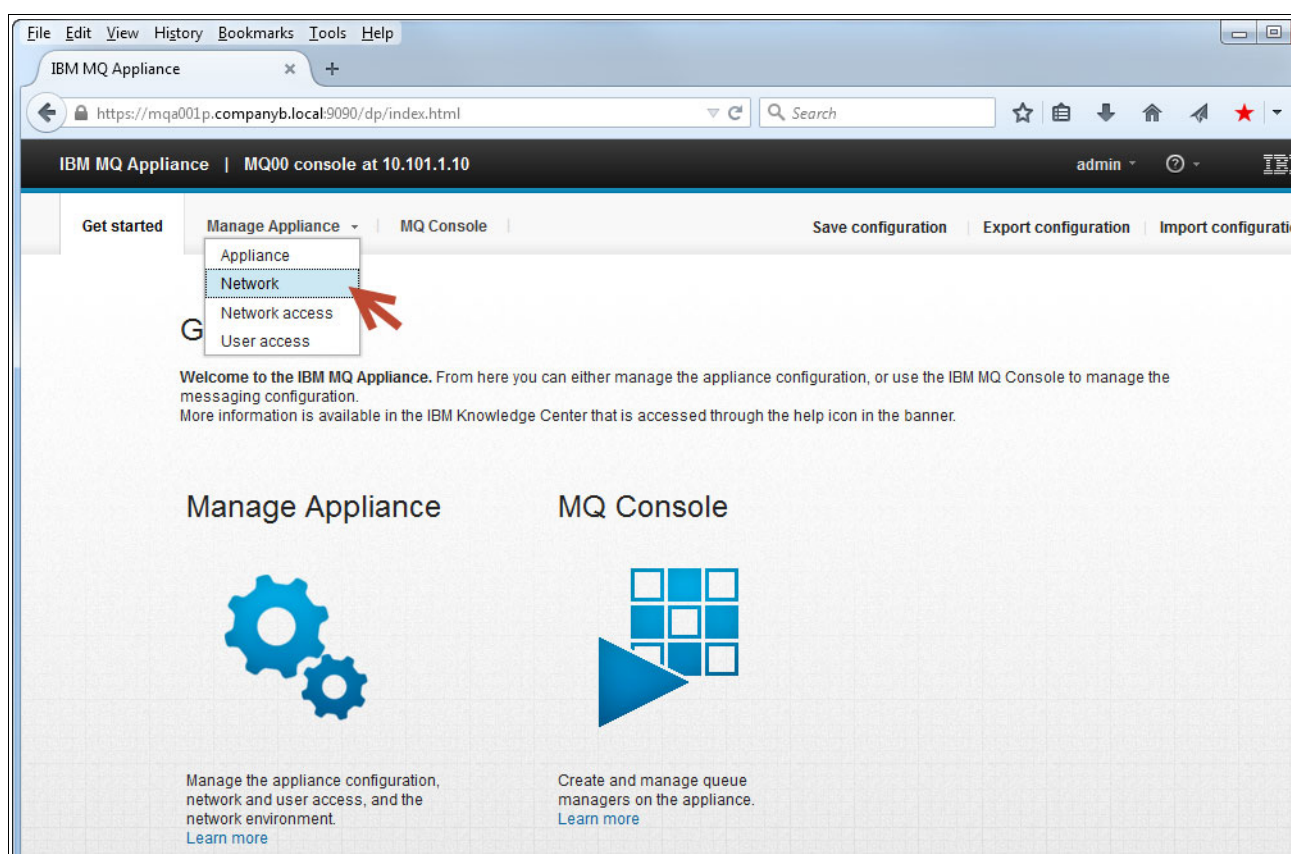


Figure 6-8 Network menu item

Complete the following steps:

1. Select **Ethernet Interface** from the Network list, then click **eth10**, as shown in Figure 6-9.

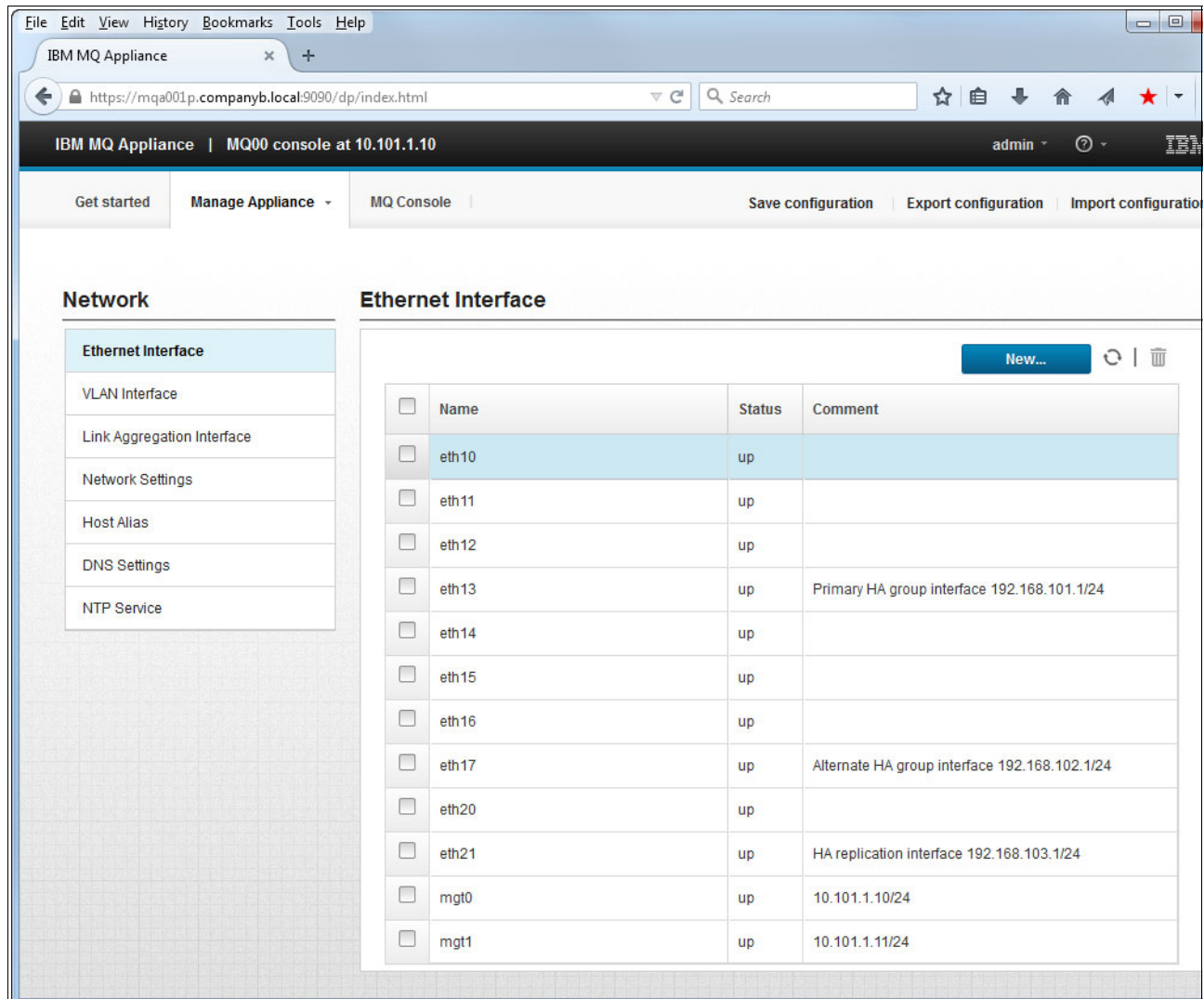


Figure 6-9 Ethernet interface selection

2. Select **Enable for link aggregation**, as shown Figure 6-10. Then, click **Apply**.

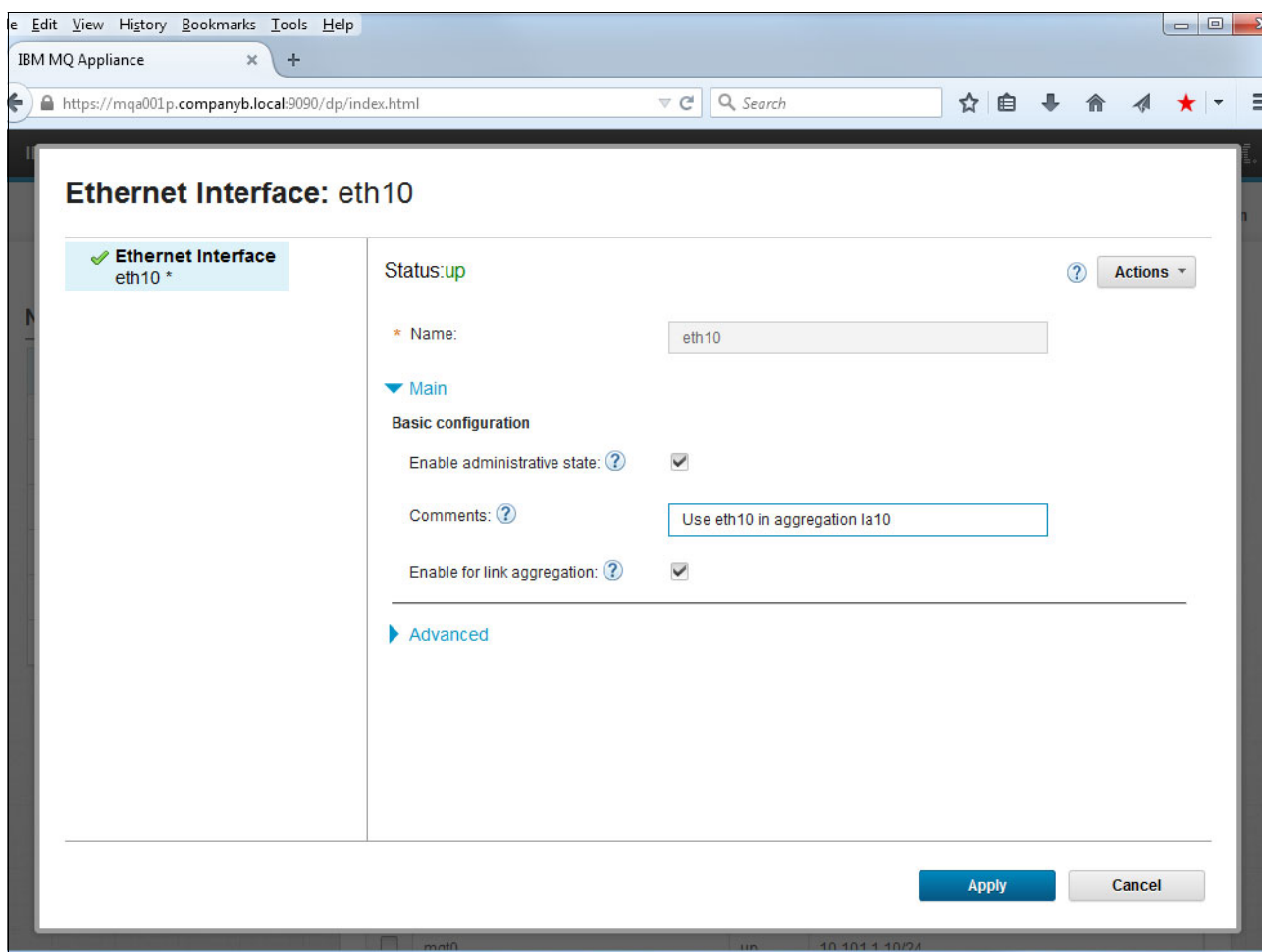


Figure 6-10 Enable Ethernet interface for link aggregation

3. Repeat the configuration of Enable for link aggregation for each of the other Ethernet interfaces that are included in the link aggregation.

Note: It is important that the network switches are configured to enable link aggregation on the ports to which the appliance interfaces are connected. Work with your network team to ensure that this configuration is done correctly on the switches.

Creating a link aggregation interface

The IBM MQ Appliance can access a group of network interfaces as though it provides a single point of access to the network. An object called a link aggregation interface is created to perform this function. Other configurations then use the link aggregation interface as though it were a regular Ethernet interface. Complete the following steps to create a link aggregation interface:

1. From the Network configuration page in the Web_UI, select **Link Aggregation Interface**, and then click **New**, as shown in Figure 6-11 on page 79.

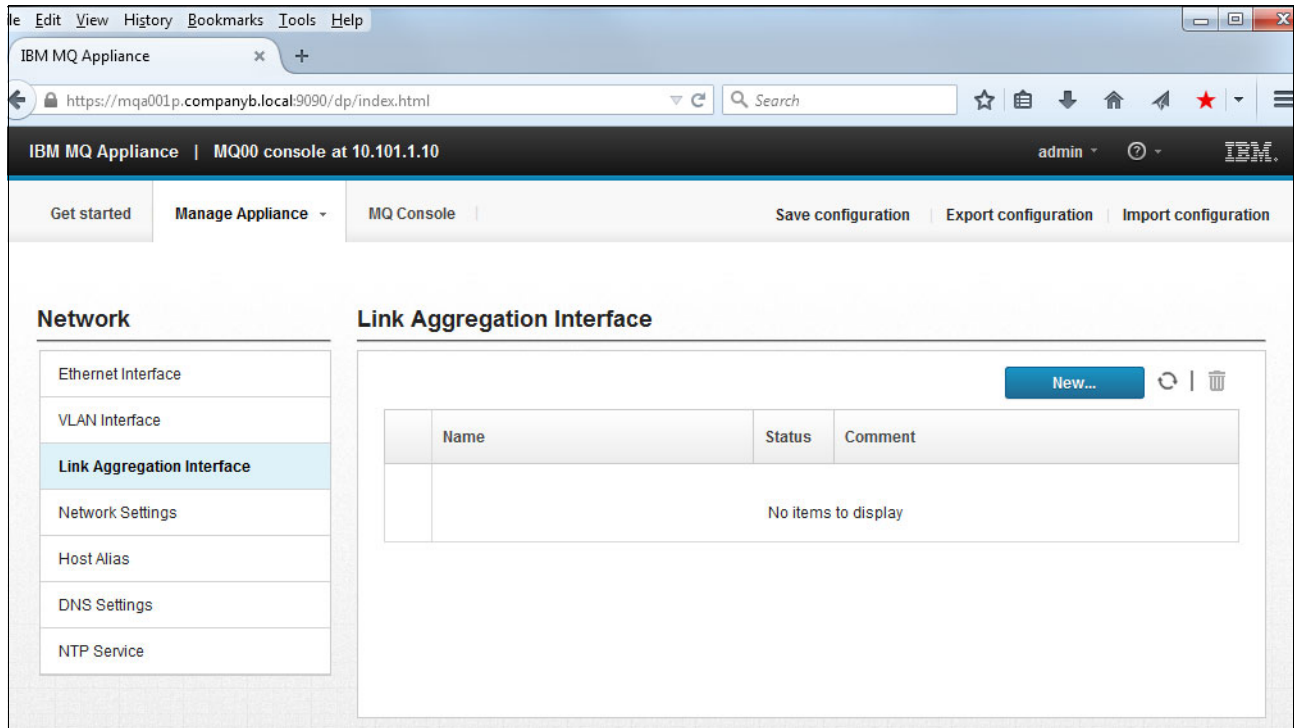


Figure 6-11 Creating a link aggregation interface

2. Enter a name for the link aggregation interface. The example that is used here uses the name **la10** and includes a comment, which is good practice but not required. Set the Aggregation Mode to **LACP**. This should be set to **Active-backup** or **Transmit-based** if the switches do not support LACP.
3. Click **Add** to create a connection to each of the six Ethernet interfaces. Figure 6-12 on page 80 shows the result after all six Ethernet interfaces are added into the link aggregation.
4. Select **Stable** for the LACP selection policy. Set MAC and IP addresses for the LACP distribution algorithm.
5. The IP addressing and IP routing sections of the form are not shown in Figure 6-12 on page 80. They do not need to be completed because a VLAN interface is created by using this link aggregation interface. If native VLAN tagging is not enabled for these switch ports, only a single VLAN is accessible and the IP addressing and IP routing sections of the Link Aggregation Interface form must be completed.
6. Click **Apply** to activate the new configuration.

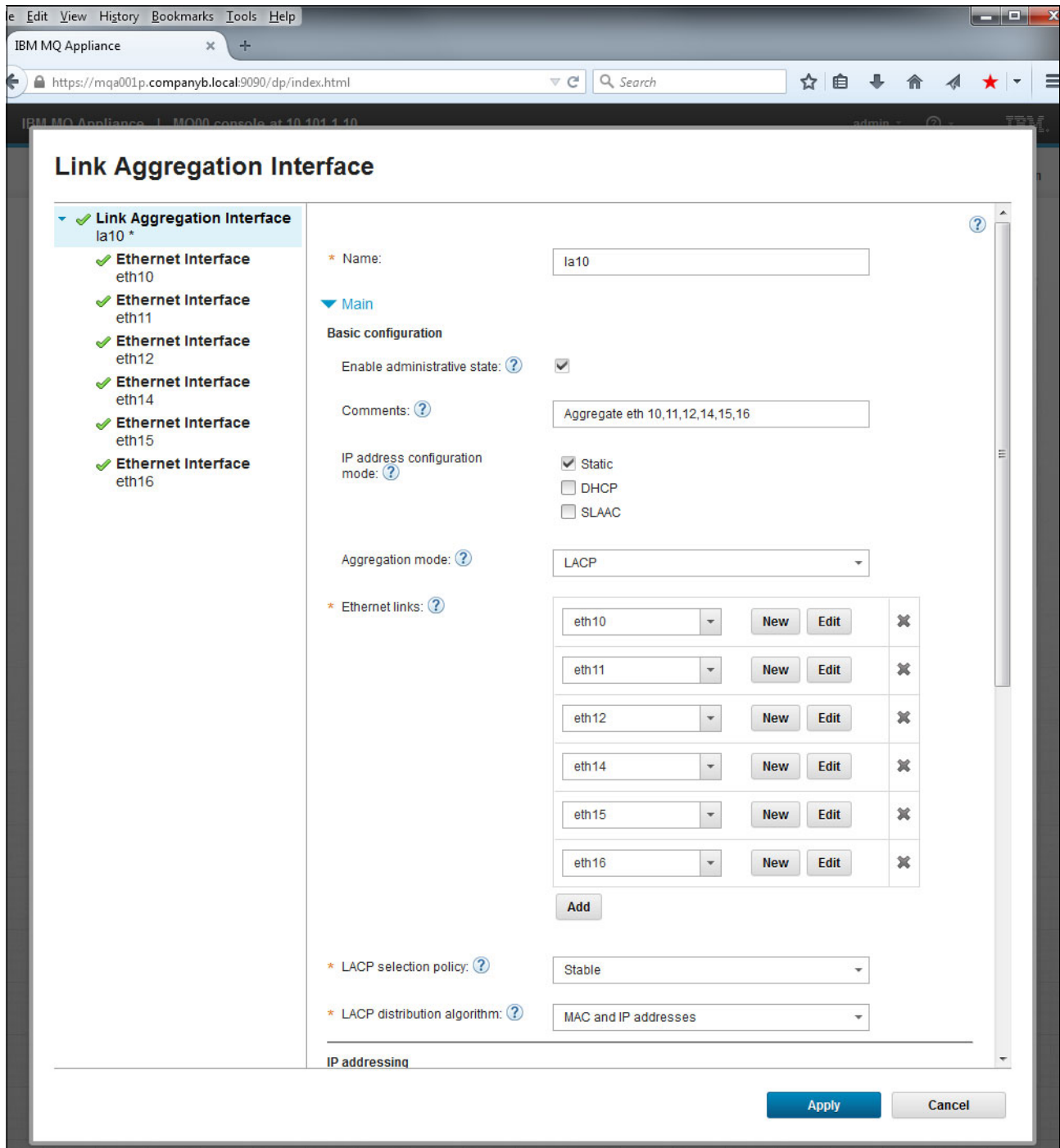


Figure 6-12 Apply configuration of link aggregation interface

- Figure 6-13 on page 81 shows the Network, Link Aggregation Interface page after the la10 interface is created.

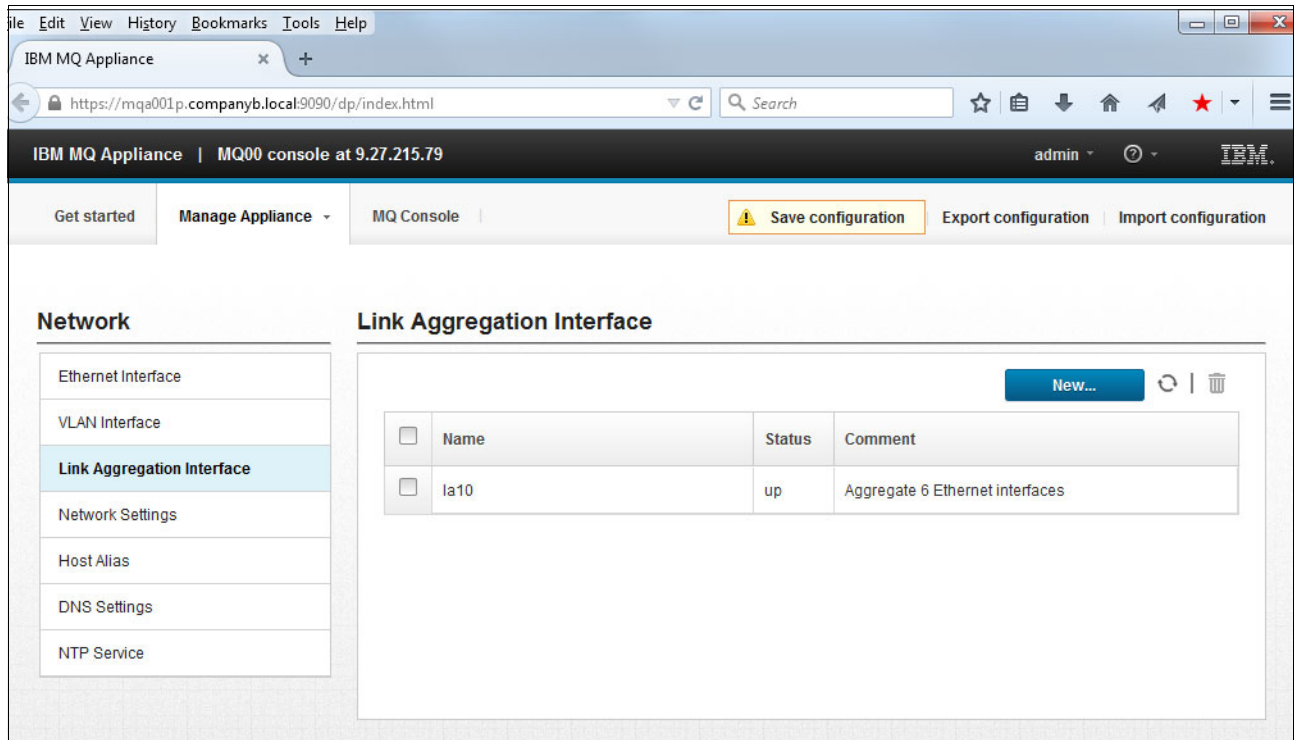


Figure 6-13 Link aggregation is created; configuration not saved

Creating a VLAN interface

If native VLAN tagging is used, it allows the IBM MQ Appliance to connect to many VLANs by using a single network interface. In this case, the single interface is a link aggregation across many Ethernet interfaces. This configuration provides higher bandwidth than a single interface unless 10 Gb Ethernet is used and provides highly available access to the network.

Complete the following steps to create a VLAN interface:

1. Select **VLAN Interface** from the Network page, as show in Figure 6-14 on page 82. Then, click **New**.

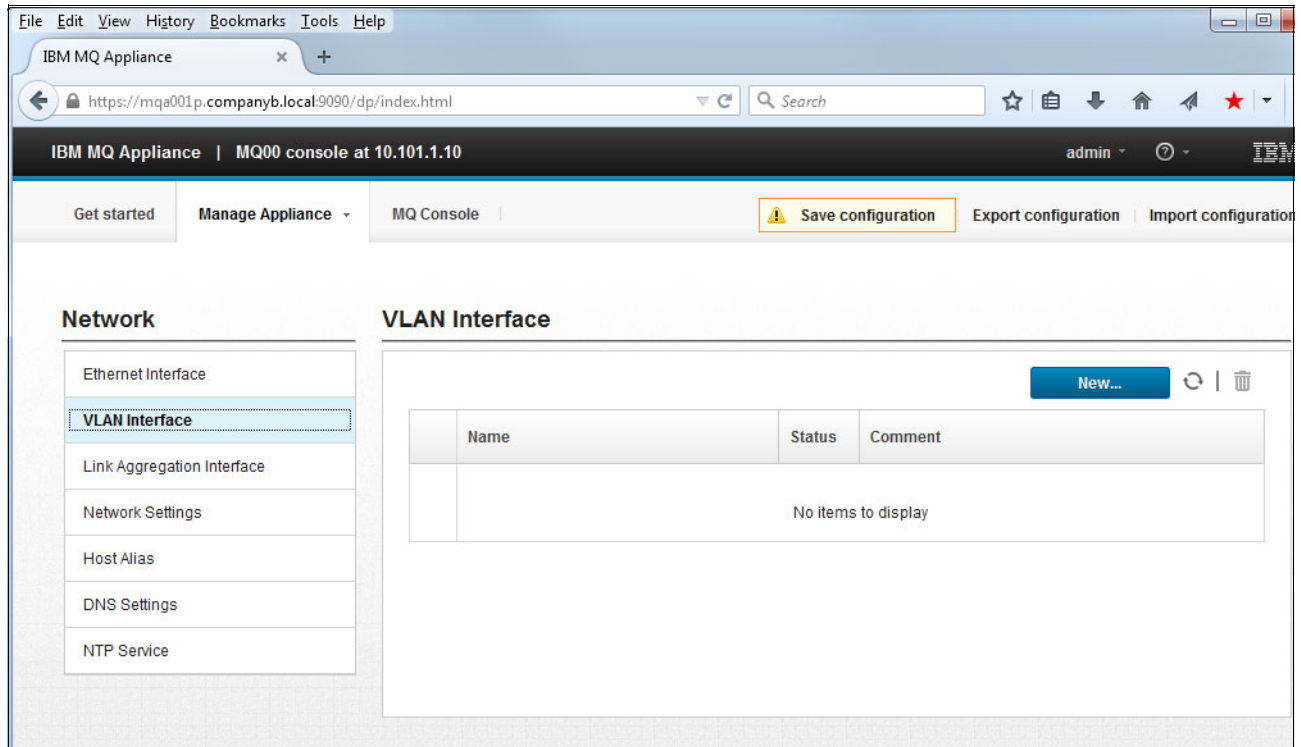


Figure 6-14 Access the form to create a VLAN interface

2. By using the first section of the page that is shown in Figure 6-15 on page 83, you can name the new VLAN interface and describe it by adding a comment. In this example, the name that is used is `vl01`. Set VLAN hosted on to **Link aggregation** so that the `la10` aggregation can be selected in Link aggregation interface. Set the VLAN identifier to the agreed identifier, which is 2012 in this example. Set the Outbound priority as required (normally zero). For more information about this field, see the Help section in the form.

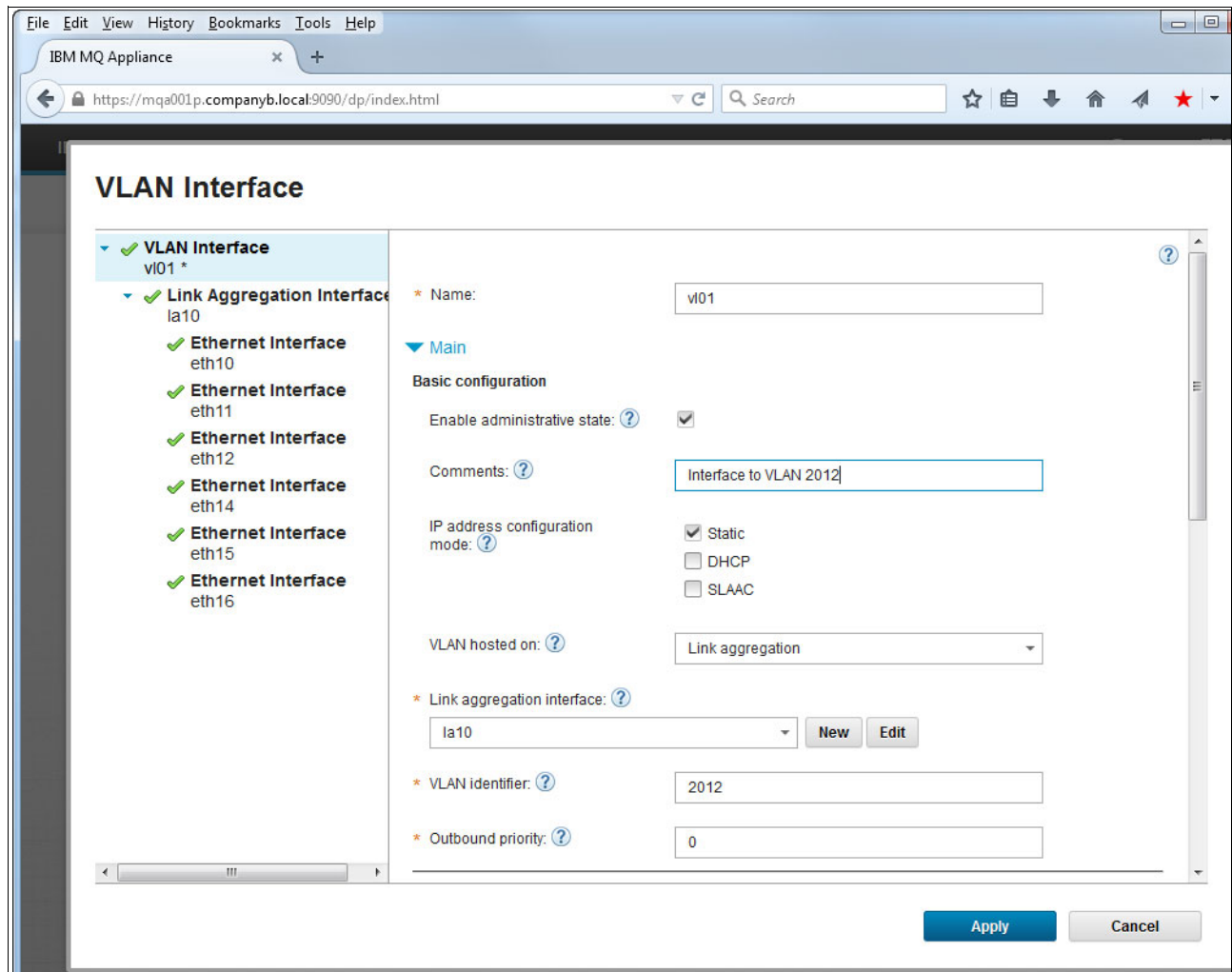


Figure 6-15 Set up and name VLAN base interface

3. Scroll down to show the next sections of the window. These sections are the IP addressing and IP routing. Configure the assigned network address and subnet mask in Classless Inter-Domain Routing (CIDR) format. This format places the number of bits in the network portion of the IP address after the dotted decimal address, and separates from it by a slash character. Configure secondary addresses in the same subnet if required by clicking **Add** and entering the extra addresses.
4. Configure a default gateway or static routes in the IP routing section. In this example, only a Default IPv4 Gateway is assigned.
5. If a single Ethernet interface is used, the IP addressing and IP routing information must be configured there instead. If a link aggregation interface is created but native VLAN tagging is not used, the information is configured on the Link Aggregation Interface because the VLAN Interface does not exist, as shown in Figure 6-16 on page 84.

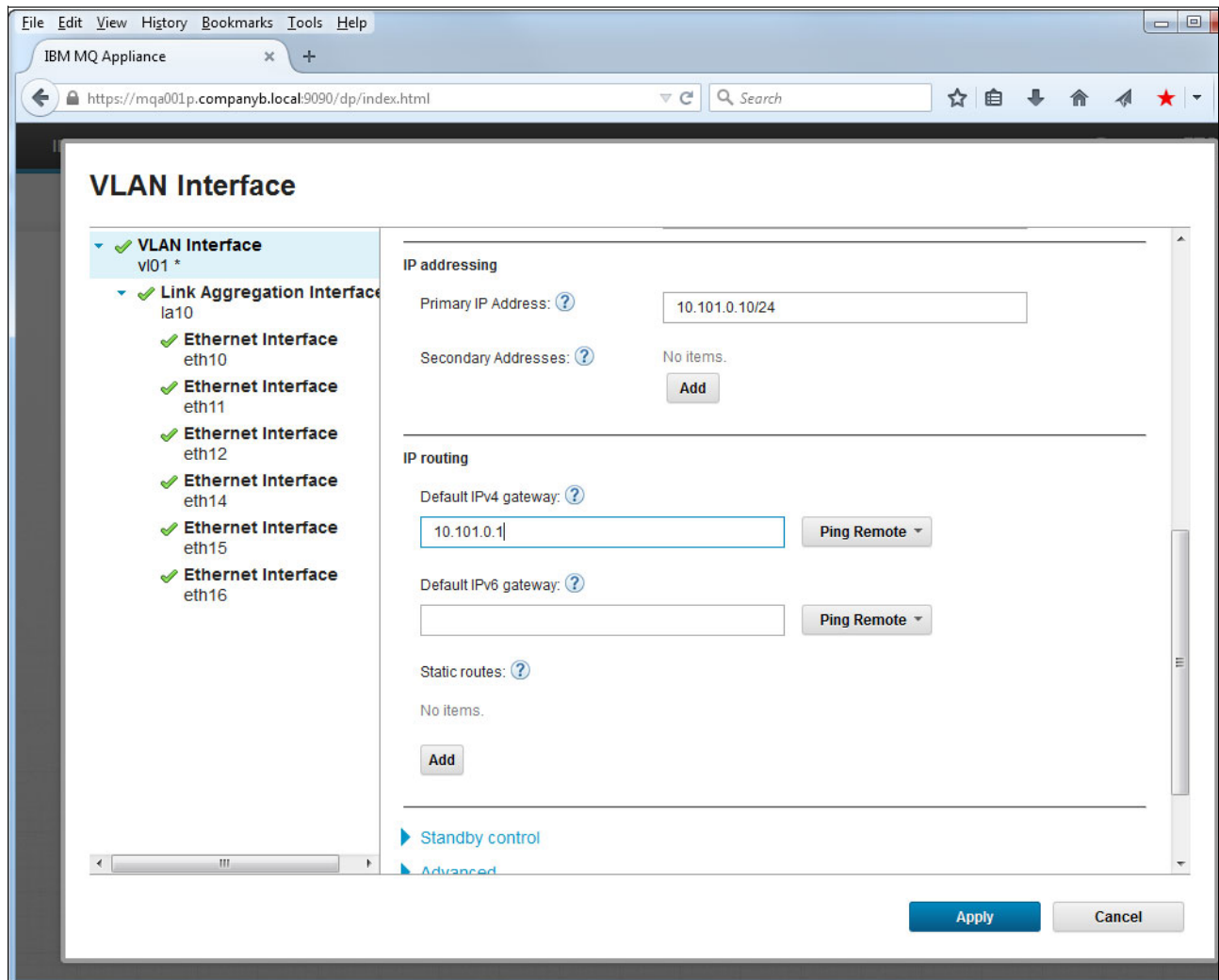


Figure 6-16 Apply updated values from the bottom part of the VLAN Interface panel

Figure 6-17 on page 85 shows the VLAN Interface page with the newly created VLAN interface shown.

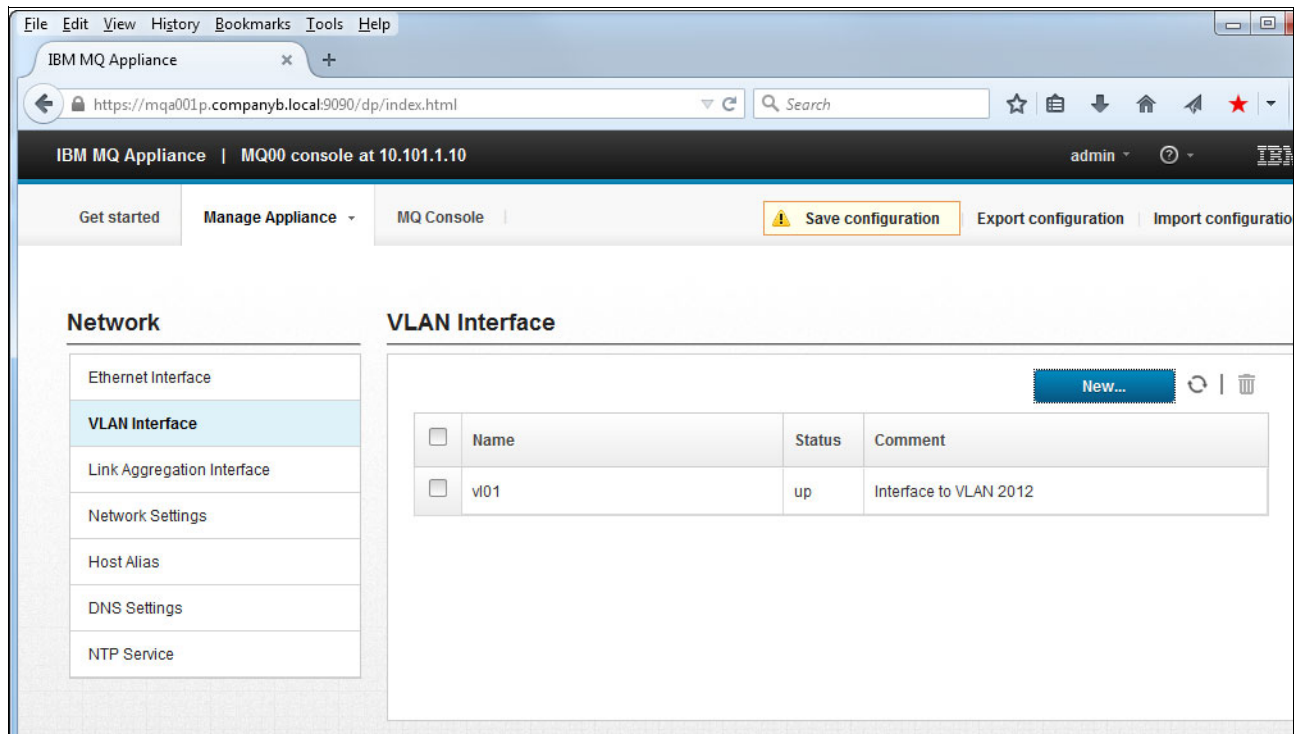


Figure 6-17 New VLAN interface shown on VLAN Interface page

Note: Multiple VLAN interfaces can be configured, but each VLAN interface must be for a separate VLAN ID or be configured on a separate Link Aggregation or Ethernet interface. Only one VLAN interface for each VLAN ID functions correctly on each interface.

Note: Conventional server or sender channels can bind outgoing requests to a specific interface by using the LOCLADDR attribute.

Cluster channels on a queue manager can be directed to use a specific interface address by setting the MQ_LCLADDR environment variable. In the absence of the MQ_LCLADDR environment variable, the outgoing interface address for a cluster sender channel are chosen by the routing table.

Configuring management interfaces

During the initial setup wizard, only the mgt0 interface was configured and a default route was established. In the previous section, the major data interfaces were configured and another default route was established, which has the same metric.

In this section, the mgt0 interface is altered to remove the default route and establish a static route instead so that there is no ambiguity in the routing table. It also sets up the mgt1 interface so that management functions can still be accessed, even if a network interface or switch fails.

Reconfiguring mgt0 interface

Complete the following steps:

1. From the Manage Appliance menu, select **Network**, then select **Ethernet Interface** and click **mgt0**, as shown in Figure 6-18 on page 86.

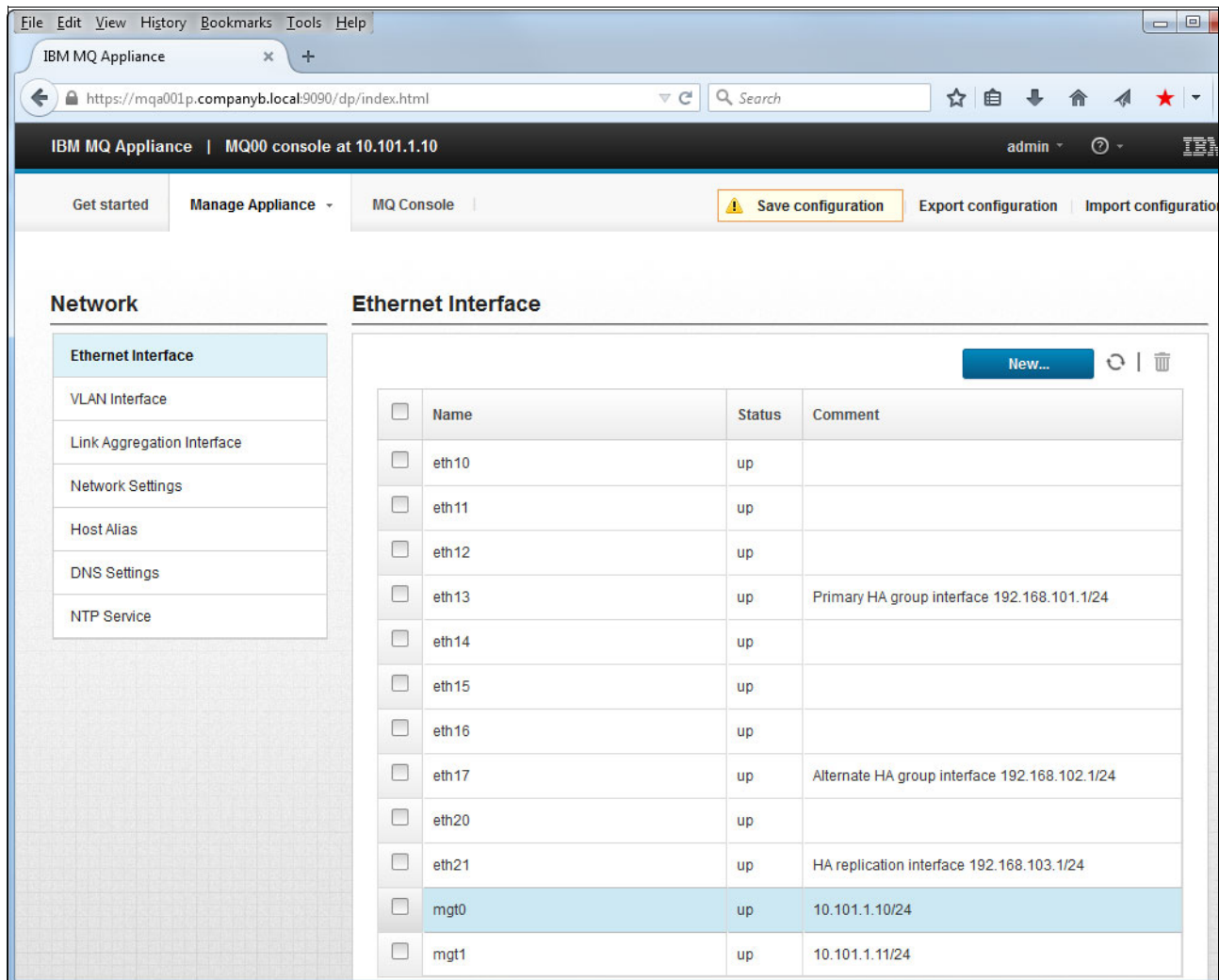


Figure 6-18 Select mgt0 interface

- The Basic configuration and IP addressing sections do not need to be changed. Scroll down to the IP routing section. Remove the address from the Default IPv4 gateway field. Click **Add** (below Static routes).
- Enter the Destination address 0.0.0.0/0, Next-hop router address <router address for the subnet> and Metric 255, as shown in Figure 6-19 on page 87. The interface address is 10.101.1.10/24. The gateway address must be in the same subnet, and in this case is the lowest address in the range (10.101.1.1).

Metric 255 is used so that any other route, including the default route, is used in preference to this route, except for sockets that originated against this interface. The destination address 0.0.0.0/0 matches all addresses and is used so that management access can be originated from anywhere. In many networks, the management interfaces are protected by firewalls and this route is acceptable. In some cases, you might want to create a separate /32 static route for each system that is allowed to connect to the management interfaces. The extra route is made available by clicking **Add** again. A route to a file access server also can be specified.

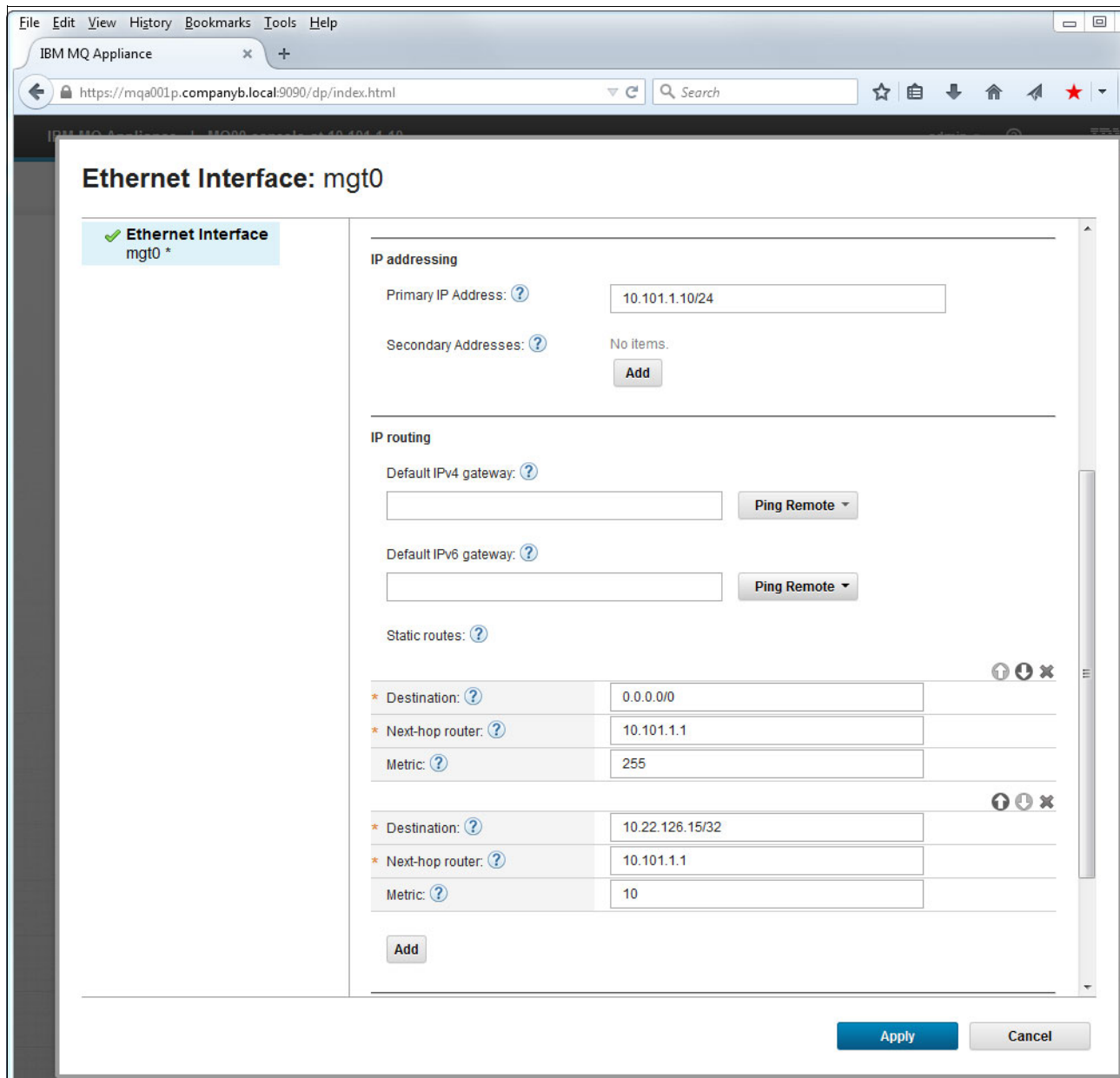


Figure 6-19 Configuration of mgt0 route

If you have a dedicated group of servers for staging files on and off the appliance, defining specific routes for each of those servers via the mgt0 or mgt1 interfaces might be appropriate.

Configuring mgt1 interface

The mgt1 interface provides an alternative to mgt0 in case that interface or the switch it is attached to fails. In this section, we describe the process that is used to configure the mgt1 interface with an address in the same subnet as mgt0.

Note: It is not necessary that mgt1 and mgt0 are in the same subnet, although it is common practice.

Complete the following steps to configure the mgt1 interface:

1. Select the mgt1 interface as the mgt0 interface was selected previously. Figure 6-20 shows the Basic configuration and IP addressing sections, which are populated with the values that are used at the time of this writing.

The screenshot shows a web browser window with the title "IBM MQ Appliance". The address bar shows the URL "https://mqa001p.companyb.local:9090/dp/index.html". The main content area is titled "Ethernet Interface: mgt1". On the left, there is a sidebar with a green checkmark and the text "Ethernet Interface mgt1". The main configuration area is divided into two sections: "Basic configuration" and "IP addressing".

Basic configuration

- Status: up (with a question mark icon and an "Actions" dropdown menu)
- Name: mgt1
- Enable administrative state: ☒ (with a question mark icon)
- Comments: 10.101.1.11/24 (with a question mark icon)
- IP address configuration mode: ☒ Static, ☐ DHCP, ☐ SLAAC (with a question mark icon)
- Enable for link aggregation: ☐ (with a question mark icon)

IP addressing

- Primary IP Address: 10.101.1.11/24 (with a question mark icon)
- Secondary Addresses: No items. (with a question mark icon and an "Add" button)

At the bottom right, there are "Apply" and "Cancel" buttons.

Figure 6-20 Basic mgt1 configuration

2. Scroll down to the IP routing section. Click **Add** to create as many static route entry fields as are needed, and enter the static route information. Click **Apply**. The completed IP routing section is shown in Figure 6-21 on page 89.

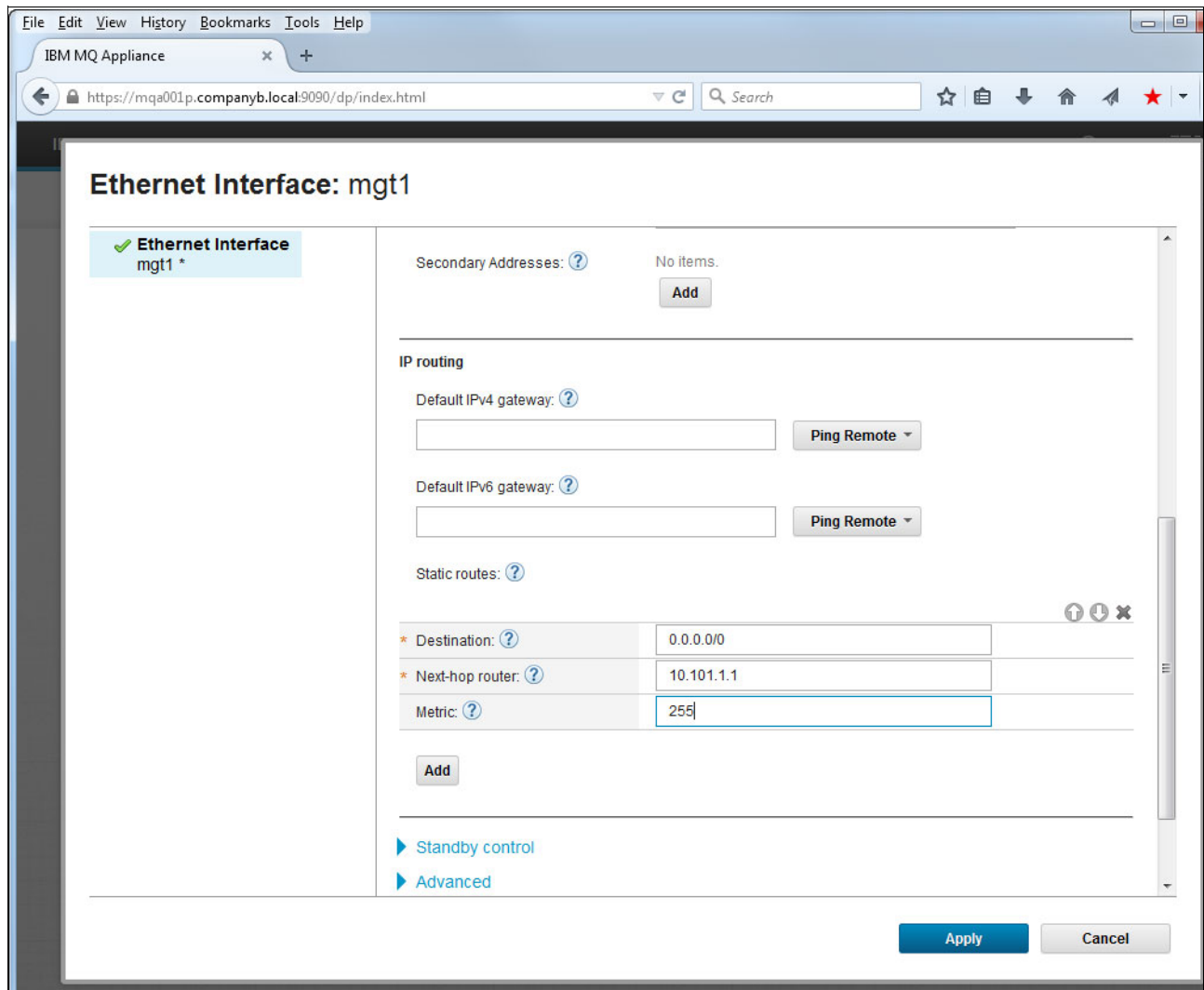


Figure 6-21 mgt1 IP routing

Configuring HA group interfaces

The IBM MQ Appliance uses three network interfaces for HA-related functions. Two 1 Gb Ethernet interfaces, eth13, and eth17 are the HA group primary interface and HA group alternative interface. The eth21 10 Gb interface is the HA group replication interface.

For more information about other options when these interfaces are configured, see this website:

http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/configuring/co00240_.htm

The configuration option that provides the highest performance and best reliability is to directly connect the eth13, eth17, and eth21 interfaces of the two appliances in the HA group by using a single cable for each connection. Suitable cables are included with the appliances when they ship. The supplied cables can be used if the cable path between the appliances is less than 5 meters (approximately 16 feet).

When the distance is up to 10 meters (32 feet), the provided 1 Gb cables can be used, but the 10 Gb direct connect copper interface cable should be replaced with a longer OM3 or OM4 fiber cable. The fiber cable operates with the installed original SFP+ transceivers instead of the direct connect cable. For connections of over 10 meters (32 feet), your site must provide 1 Gb Ethernet cabling in addition to fiber cabling.

This chapter describes the configuration of one half of the HA group. The second appliance is configured identically, except that the address that is assigned to each of the three HA group interfaces is one address higher. For example, where the first appliance has an eth13 address of 192.168.101.1/24, the second appliance eth13 address is 192.168.101.2/24.

It is required that each of the HA group interfaces is configured in a separate subnet. (To make this distinction clear, we use /24 subnet in this book.) This choice means that the third octet of the address changes between the interfaces. Because there can be only two appliances in an HA group, a /30 subnet in theory provides enough addresses; however, it might be more difficult to use any future enhancements in this area.

Because the HA interfaces are directly connected, there is no need for any IP routing definitions. Only the Basic Configuration and IP addressing sections are needed.

As the book was written, we configured jumbo frames (that is, a large Message Transfer Unit, or MTU size) on the replication interface. If this interface is not directly connected to the other appliance in the HA group, this configuration normally requires extra setup in other network devices. We established that the HA configuration operated correctly by using the large MTU, but did not have the resources to test whether performance was improved compared with the use of normal 1500 byte MTUs.

The two appliances in our scenario are mqa001p and mqa002p. The HA network configuration values for these appliances are listed in Table 6-1.

Table 6-1 HA group network values

Appliance	Interface	IP address	Subnet prefix length	MTU size
mqa001p	eth13	192.168.101.1	24	1500
mqa001p	eth17	192.168.102.1	24	1500
mqa001p	eth21	192.168.103.1	24	9000
mqa002p	eth13	192.168.101.2	24	1500
mqa002p	eth17	192.168.102.2	24	1500
mqa002p	eth21	192.168.103.2	24	9000

Configuring eth13

Figure 6-22 on page 91 shows the Web UI configuration of eth13, which is the primary HA group interface. This pane can be reached in much the same way as any of the other interfaces that were configured in this chapter.

Complete the following steps to configure eth13:

1. Select **Enable administrative state**.
2. Enter a descriptive comment. We recommend usage and address information in the comment for any interface.
3. Set the IP address configuration mode to **Static**.
4. Unset **Enable for link aggregation**.

5. Enter the Primary IP Address in CIDR format, as shown in Figure 6-22. For the purposes of this book, the values are taken from Table 6-1 on page 90. When appliances are directly connected for the HA group, any non-routable address that is not used in the network is suitable.

Note: In your environment, work with your network team to determine appropriate subnets and host numbers. The subnet must be private and does not normally need to be routable. There might be special circumstances that require the appliances to be in separate subnets, especially if they were placed a considerable distance apart and connected via switches and routers rather than directly. Connecting the HA group interfaces by using switches and routers increases latency and can affect throughput. This issue is especially true of the replication interface (eth21).

The screenshot displays the IBM MQ Appliance web interface for configuring the Ethernet Interface eth13. The interface is titled "Ethernet Interface: eth13". On the left, there is a sidebar with a green checkmark and the text "Ethernet Interface eth13". The main content area shows the configuration details for this interface. The status is "up". Under the "Main" section, the "Basic configuration" includes: "Enable administrative state" (checked), "Comments" (Primary HA group interface 192.168.101.1/24), "IP address configuration mode" (Static selected, DHCP and SLAAC unselected), and "Enable for link aggregation" (unchecked). The "IP addressing" section shows the "Primary IP Address" as 192.168.101.1/24 and "Secondary Addresses" as "No items". There is an "Add" button for secondary addresses. At the bottom right, there are "Apply" and "Cancel" buttons.

Figure 6-22 Configure eth13 Basic and IP addressing sections

6. Scroll down to the Advanced section. Then, click **Advanced** to expand the section.

- Set the Physical mode to 1000baseTx-FD (that is, 1 Gb Full Duplex). When connecting to a switch, this mode normally is set to the default value, which is Auto. However, when directly connecting appliances, it is sensible to directly configure the wanted mode. This configuration is shown in Figure 6-23. All other Advanced section fields can be left at their default values.

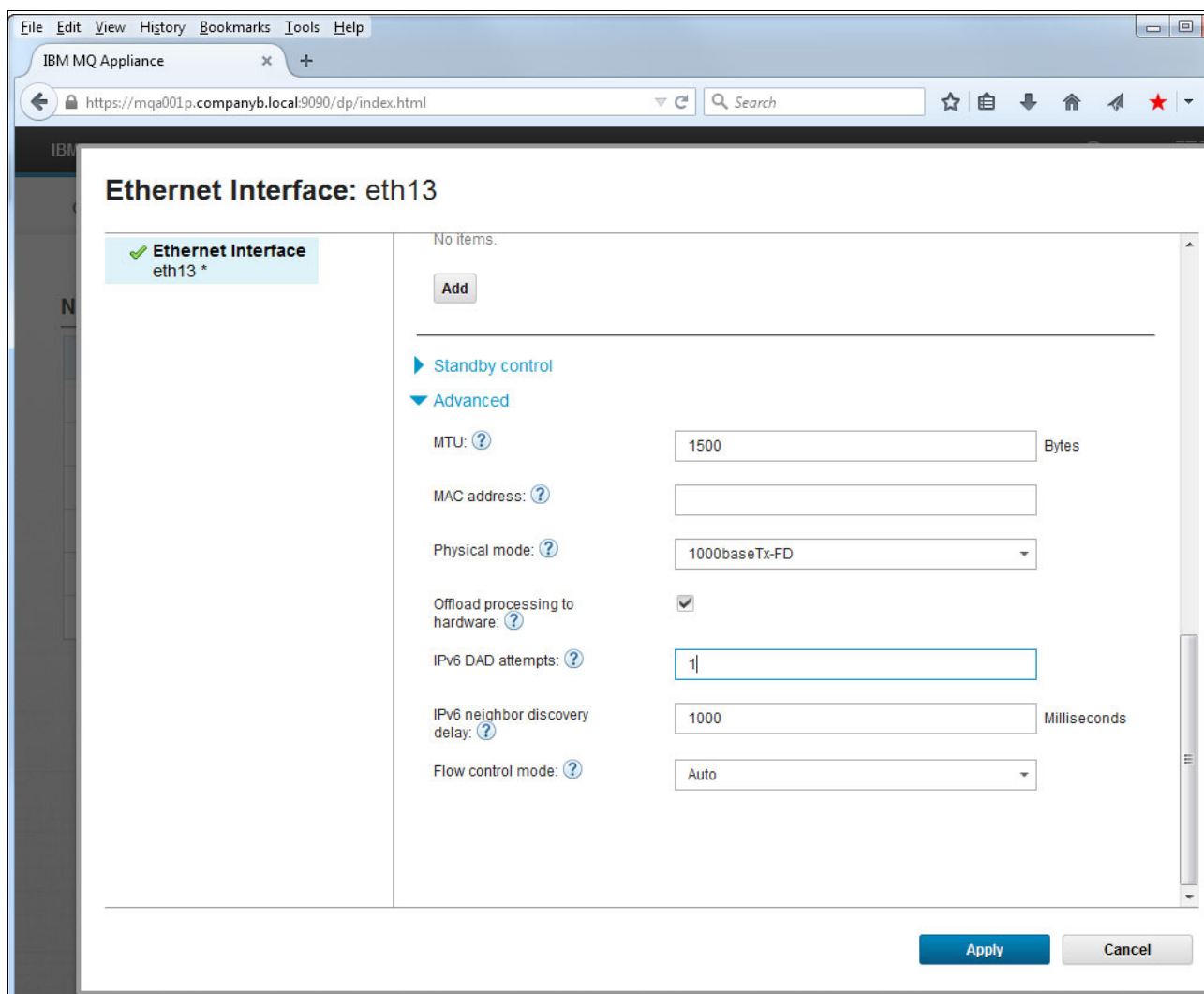


Figure 6-23 Configure eth13 Advanced section

- Click **Apply**.

Configuring eth17

The alternative HA group interface is eth17. It is configured in the same way as eth13, but by using different IP address information, which is taken from Table 6-1 on page 90.

Configuring eth21

In addition to status information that is detected and exchanged by using the primary and secondary HA group interfaces, the HA group uses the replication interface to forward data between appliances. The eth21 10 Gb interface is used for this function.

The configuration of eth21 is shown in Figure 6-24 and Figure 6-25 on page 94. It is configured as is the HA group interfaces eth13 and eth17, except that a large MTU size is used to improve efficiency of the data transfers by reducing the packet overheads relative to the payload.

Figure 6-24 shows the Basic configuration and IP addresses sections of the form.

The screenshot displays the 'Ethernet Interface: eth13' configuration page in a web browser. The browser's address bar shows 'https://mqa001p.companyb.local:9090/dp/index.html'. The page title is 'Ethernet Interface: eth13'. On the left, a sidebar lists 'Ethernet Interface' and 'eth13'. The main content area shows the interface status as 'up' and an 'Actions' button. The 'Main' section is expanded, showing the 'Basic configuration' and 'IP addressing' sections. In the 'Basic configuration' section, 'Enable administrative state' is checked, 'Comments' is 'Primary HA group interface 192.168.101.1/24', 'IP address configuration mode' is set to 'Static' (with 'DHCP' and 'SLAAC' as options), and 'Enable for link aggregation' is unchecked. The 'IP addressing' section shows the 'Primary IP Address' as '192.168.101.1/24' and 'Secondary Addresses' as 'No items', with an 'Add' button. At the bottom right, there are 'Apply' and 'Cancel' buttons.

Section	Field	Value
Basic configuration	Name	eth13
	Enable administrative state	<input checked="" type="checkbox"/>
	Comments	Primary HA group interface 192.168.101.1/24
	IP address configuration mode	<input checked="" type="checkbox"/> Static <input type="checkbox"/> DHCP <input type="checkbox"/> SLAAC
	Enable for link aggregation	<input type="checkbox"/>
IP addressing	Primary IP Address	192.168.101.1/24
	Secondary Addresses	No items. <button>Add</button>

Figure 6-24 Basic configuration of eth21 replication interface

Figure 6-25 shows the Advanced section, and configuration of Physical mode and MTU size.

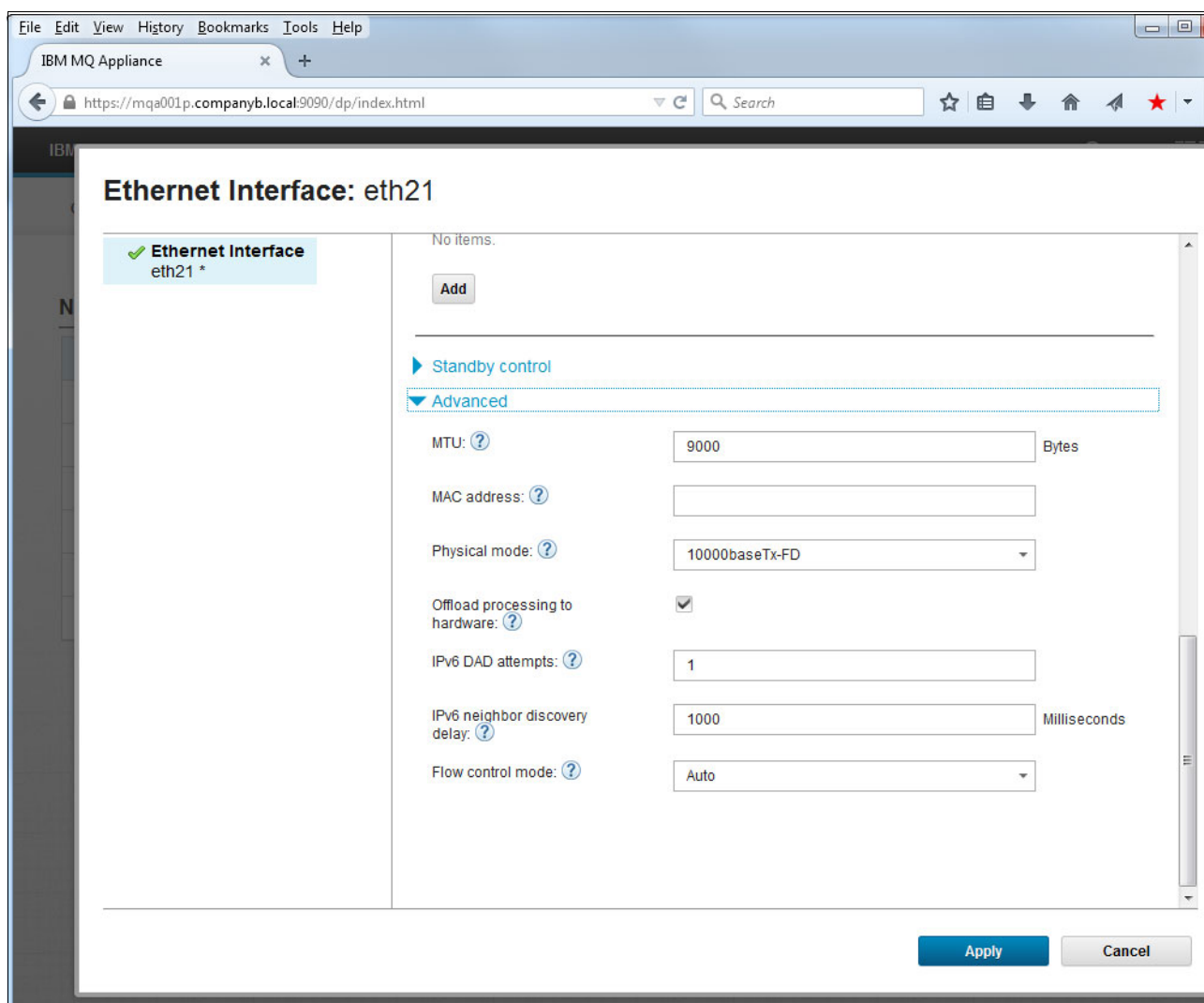


Figure 6-25 Advanced configuration of eth21 replication interface

6.4.2 Configuring names for interfaces

Various listeners that are used by IBM MQ queue managers on the appliance or by the IBM MQ Appliance should bind only to specific interfaces. Listeners should not bind to all interfaces as is the default. This configuration is used so that administrative traffic is not possible by using the data interface and data traffic cannot reach the queue managers by using the management interfaces.

To make configuring the listeners simpler and more consistent, the appliance supports configuring host aliases. These aliases have a name that can be used in place of an IP address when a listener is configured for the Web UI, SSH service, or queue manager listeners. These aliases simplify configuration because the same alias name can be used for the same purpose on all appliances.

The actual names of the host aliases do not matter. However, they appear in the configuration of other objects, so it is useful if the name conveys the correct usage. For the purposes of this book, we use the following names for the host aliases:

- ▶ WebGUI-interface
- ▶ SSH-interface
- ▶ Data-interface

Interface names are not needed for the HA group interfaces because these interfaces are used by internal components only; they are never referenced directly.

Creating a host alias for each interface address

Complete the following steps to create a host alias:

1. Click **Manage Appliance** → **Network** → **Host Alias**. Click **New**. Figure 6-26 shows the creation of a host alias for the mgt0 address, which is used for the WebGUI listener.

The screenshot shows a web browser window titled "IBM MQ Appliance" with the URL "https://mqa001p.companyb.local:9090/dp/index.html". The main content area is titled "Host Alias: WebGUI-interface". On the left, there is a sidebar with a "Host Alias" section containing "WebGUI-interface *". The main configuration area shows the following fields:

- Status:** up
- Name:** WebGUI-interface
- Main** (expanded section):
 - Enable administrative state:** ☒
 - Comments:** Bind web gui to this address (mgt0) 10.101.1.10
 - IP address:** 10.101.1.10

At the bottom right, there are "Apply" and "Cancel" buttons.

Figure 6-26 Creating host alias for an interface address

2. Complete in the fields and click **Apply**.

The same process is used to create other host aliases as needed for the planned configuration. Normally, at least three aliases are needed: one each for the Web UI, SSH, and data.

The host aliases that are created are used later in the network access configuration process so that the Web UI and SSH services can be accessed only via the designated interfaces. The Data interface is used when queue manager listeners are configured so that they bind only to the data interface.

Note: Multiple data interfaces can be configured as individual Ethernet interfaces, smaller aggregations, or several VLANs. Define a host alias for each defined address so that these aliases can be used when queue managers are configured. Different queue managers can then bind to different addresses to listen for incoming connections.

Outgoing connections for cluster channels can be controlled by using the routing table or MQ_LCLADDR environment variable. The routing table is created by using static routes that are defined against each interface. For information about setting the MQ_LCLADDR environment variable, see Example 6-2 on page 63.

The following host aliases were created for use in the examples and figures on the next section:

- ▶ WebGUI-interface
- ▶ SSH-interface
- ▶ Data-interface

6.4.3 External services used by the IBM MQ Appliance

The IBM MQ Appliance uses external services to resolve host name-to-IP address mappings, and to keep the internal clock in sync with other computers within the organization.

These items are described in this section.

Configuring DNS Settings

It is inconvenient to configure every host address as an IPv4 or IPv6 address. The solution to this issue is provided by the Domain Name Service (DNS). The DNS Settings page, which is accessed by clicking **Manage Appliance** → **Network** → **DNS Settings**, is used to configure how the IBM MQ Appliance uses DNS.

The following items within the DNS Settings page must be configured:

- ▶ Comments
Describes how DNS is used.
- ▶ IP preference
IPv4 is still widely used. If your site switched to the use of IPv6, select that option.
- ▶ Search domains
These domains are a shortcut. The DNS expects to see a query for a fully qualified host name, including the domain that has the host name. If a bare host name with no domain information is queried, the system tries the query with each of the Search domains that are listed.

This process is convenient for a person on a workstation because the user does not have to enter as much information. However, it is a potential security flaw in a server system and should be avoided. Use fully qualified host names in all cases, and leave Search domains empty.

- DNS servers

This item is a list of servers that can be queried for domain name resolution. Each server must return information about the same set of domains. The appliance does not query all of the servers in the list looking for one that knows about a specific domain. To eliminate the single server risk, specify more than one DNS server or specify an address that is load balanced to multiple DNS servers. These addresses must be specified as static host names, IPv4 dotted decimal, or IPv6 hexadecimal addresses because there is no way to resolve a name without knowing the DNS server address.

- Static hosts

If specific hosts are not defined in DNS, name-to-address mappings can be defined on the appliance. This mapping also can occur if the DNS lookup for the addresses is not available in the DNS service that is configured.

The Static hosts capability is commonly used where Network Address Translation is needed to traverse a firewall. The host name maps to the actual address, but IBM MQ must use the translated address. The translated address can be defined in the Static hosts list.

- Load balancing algorithm

The options are round-robin, which allows load balancing dynamically between all available DNS servers, and first alive. Round-robin is often a good choice because it is simple and spreads the load between many servers. First alive might be a better choice when there is one DNS server address at each of several sites. Configure the local site DNS server address first, then the DNS server addresses for other sites. It is good practice in this case for the DNS server address to be externally load balanced across several actual servers to eliminate single points of failure within a site.

Figure 6-27 on page 98 and Figure 6-28 on page 99 show a DNS Settings configuration with two DNS servers, and no Search domains or Static hosts.

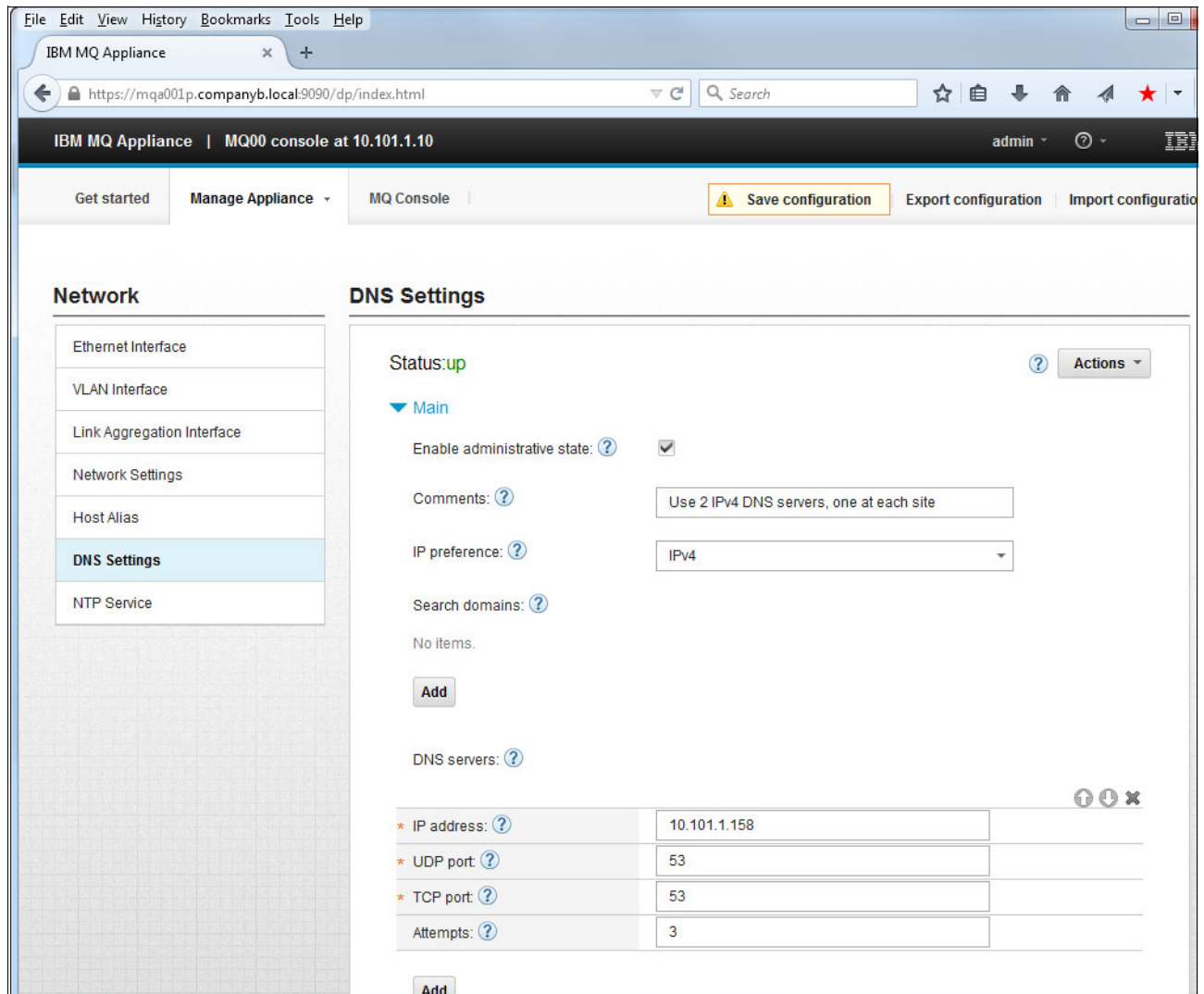


Figure 6-27 Top part of DNS Settings configuration

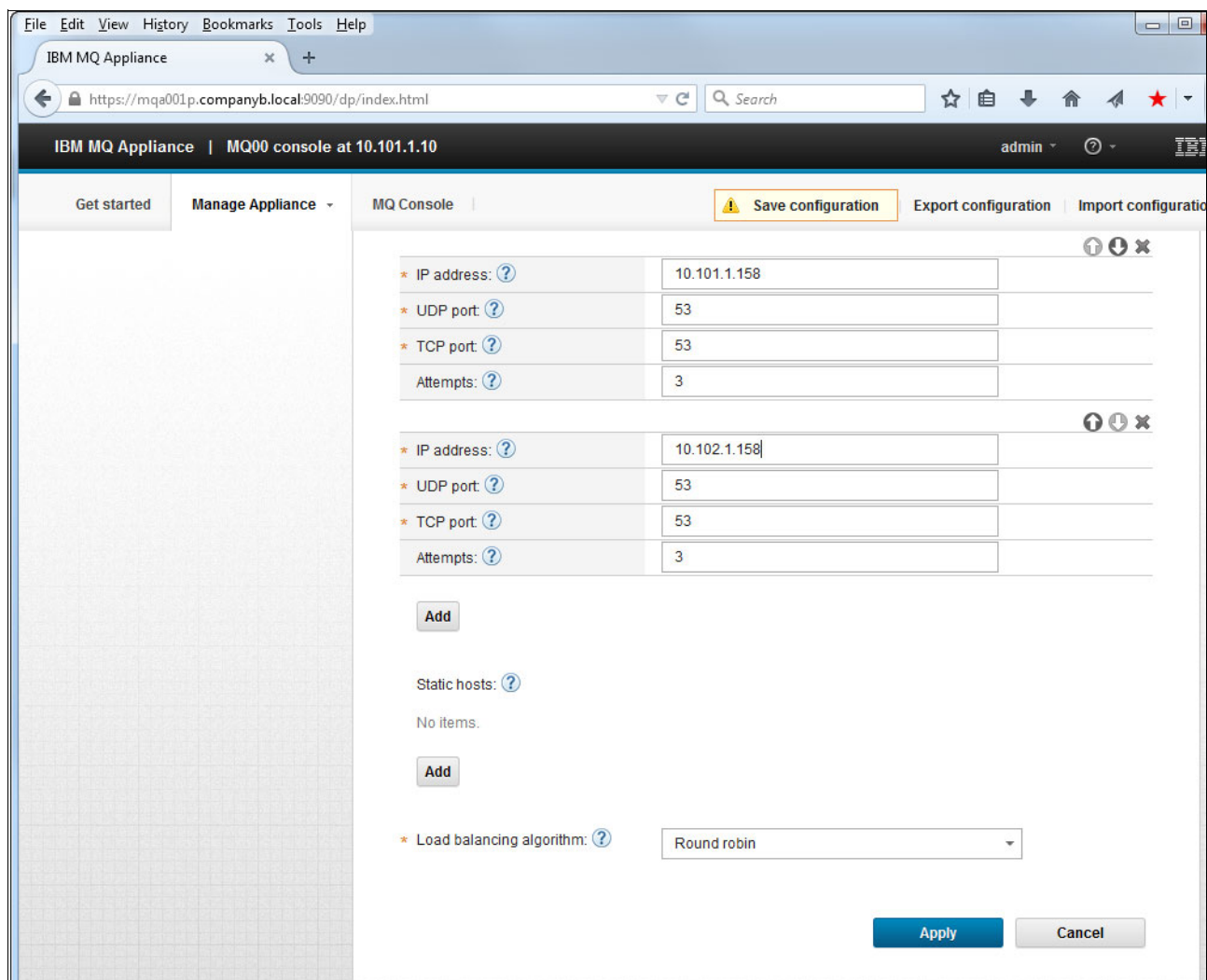


Figure 6-28 Bottom part of the DNS Settings page

Click **Apply** when the field values are populated to activate the new configuration.

Configuring the NTP Service

Time values on a system can be important, especially for audit and problem determination. There also are other considerations that make accurate time values important. The values must be consistent between machines in a network.

Consistent time values between IBM MQ Appliances in an HA group are especially important. If the clocks drift too far apart, the HA group cannot keep data synchronized between the systems.

Computer internal clocks are notorious for not keeping time accurately. The clocks can drift by seconds or even minutes each week.

The Network Time Protocol (NTP) was established to address this issue. A time source is made available by using NTP, which provides an accurate source of time information to all servers. The servers then periodically check their internal clocks against the accurate time source. The servers can estimate the drift in their own clocks to make them more accurate. They can also adjust their own clocks to bring them into line with the external source.

Complete the following steps to configure the NTP service on the IBM MQ Appliance:

1. Select the **NTP Service** option on the Network page, which is accessed by clicking **Manage Appliance** → **Network**.
2. Select the **Enable administrative state** option.
3. A comment should describe the time sources that are configured. These sources can be internal to your company or an external source. If you have more than a few servers in your company, you should have your own time source.
4. Add one or more NTP server addresses. Each address can be defined as a dotted decimal IPv4 address, an IPv6 address, or as a host name if DNS can resolve them or if a Static host is defined. More than one address should be defined so as to remove risks that are associated with a single point of failure. An externally load balanced group of NTP servers can be addressed by using a single value in the IBM MQ Appliance configuration.
5. The default Refresh interval of 900 seconds is an acceptable value at most sites. Check with your site NTP administrator to see if a different value must be used.
6. Click **Apply** to complete the configuration and active it.

The result of these steps is shown in Figure 6-29. The Status changes to “up” after you click Apply to activate the new configuration.

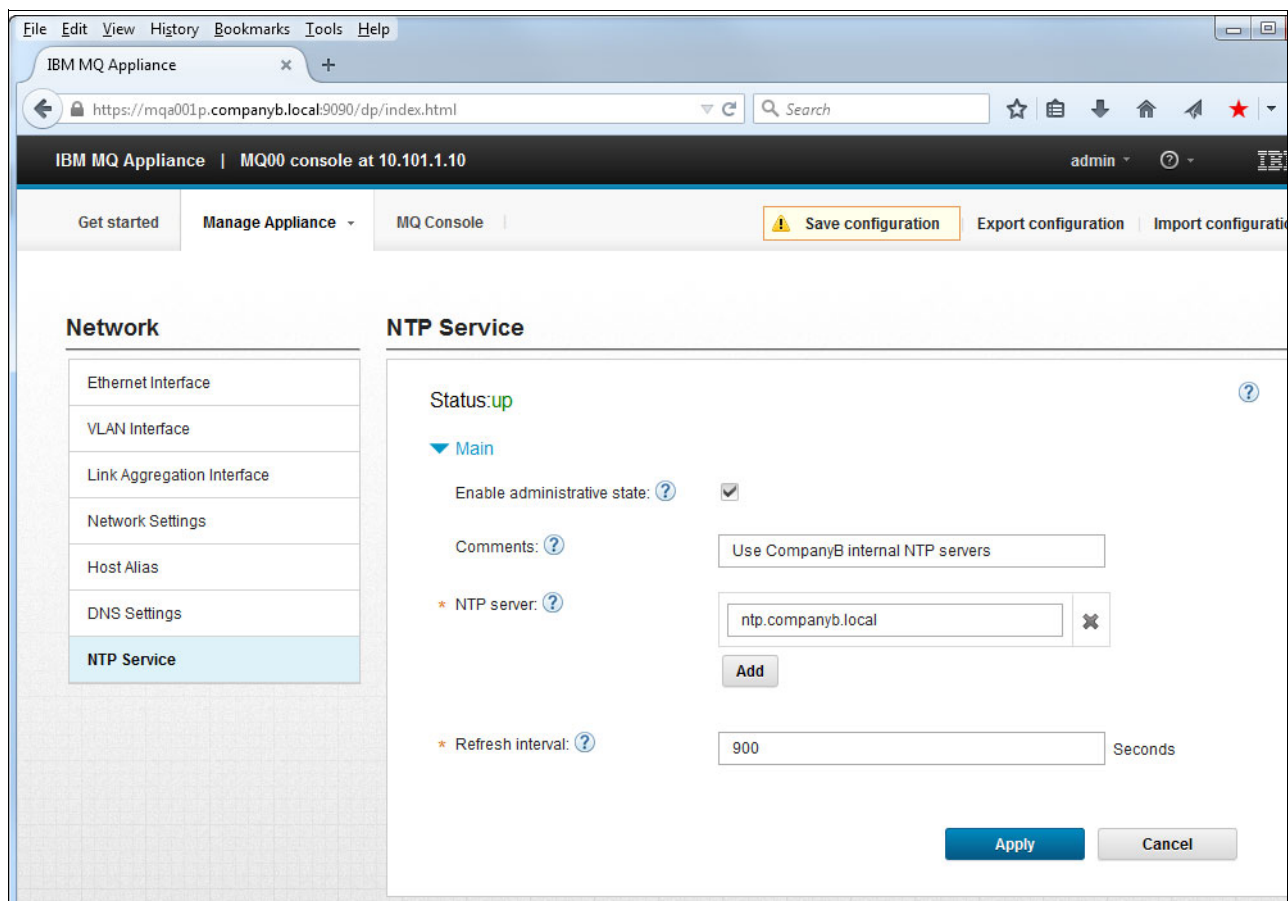


Figure 6-29 NTP Service configuration

6.4.4 Appliance hosted Network Services

The IBM MQ Appliance hosts the following services, which are used to manage the appliance:

- ▶ SSH Service
- ▶ Web Management Service

These services were enabled during the initial configuration of the appliance by using the initial wizard. However, both services were defined so that they allow connections from any interface on the appliance.

Good security practice requires that management access should only be permitted on designated management interfaces, and should be blocked on the data interfaces.

This section describes how to configure the SSH service to the mgt1 interface and the Web Management Service to the mgt0 interface. The services are configured to use separate interfaces so that if access to an interface fails, such as when a switch fails, administration is possible by using the other management service.

Configuring the SSH service

Complete the following steps to configure the SSH service:

1. Click **Manage Appliance** → **Network access**.
2. Select **SSH Service**.
3. In the Local address field, enter the Host alias for the IP address that is assigned to the mgt1 interface. This alias was created in “Creating a host alias for each interface address” on page 95. If you do not use a Host alias, the actual IP address of the interface can be used.

Figure 6-30 on page 102 shows the SSH Service panel after the Local address is configured.

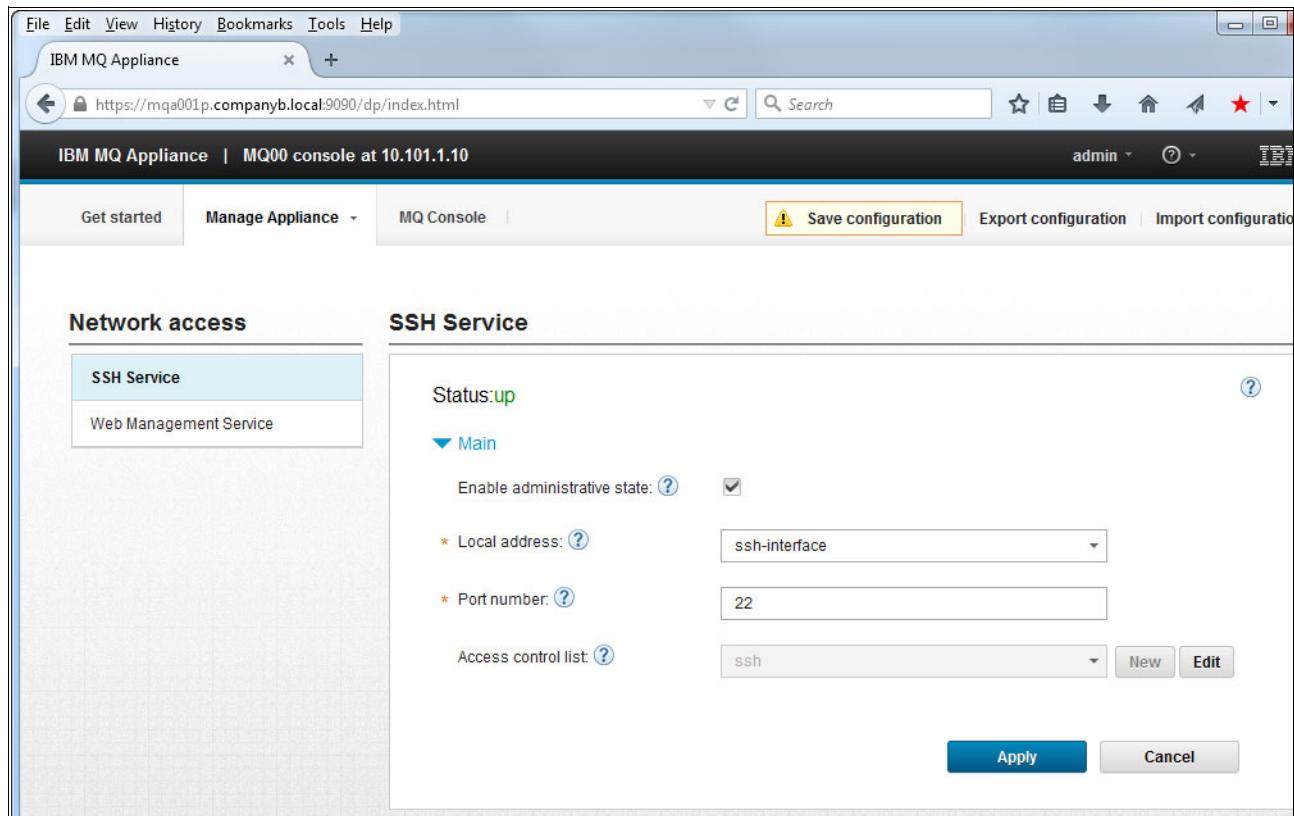


Figure 6-30 SSH service configuration

Configuring the Web Management service

Complete the following steps to configure the Web Management service:

1. Click **Manage Appliance** → **Network access**.
2. Select **Web Management Service**.
3. In the Local address field, enter the Host alias for the IP address that is assigned to the mgt0 interface. This alias was created in “Creating a host alias for each interface address” on page 95. If you do not use a Host alias, the actual IP address of the interface can be used.

Figure 6-31 on page 103 shows the Web Management Service panel after the Local address is configured. Other configuration of the Web Management Service, including use of a certificate that is signed by your company’s Certificate Authority, is described in 6.3.1, “Securing the appliance web UI” on page 64.

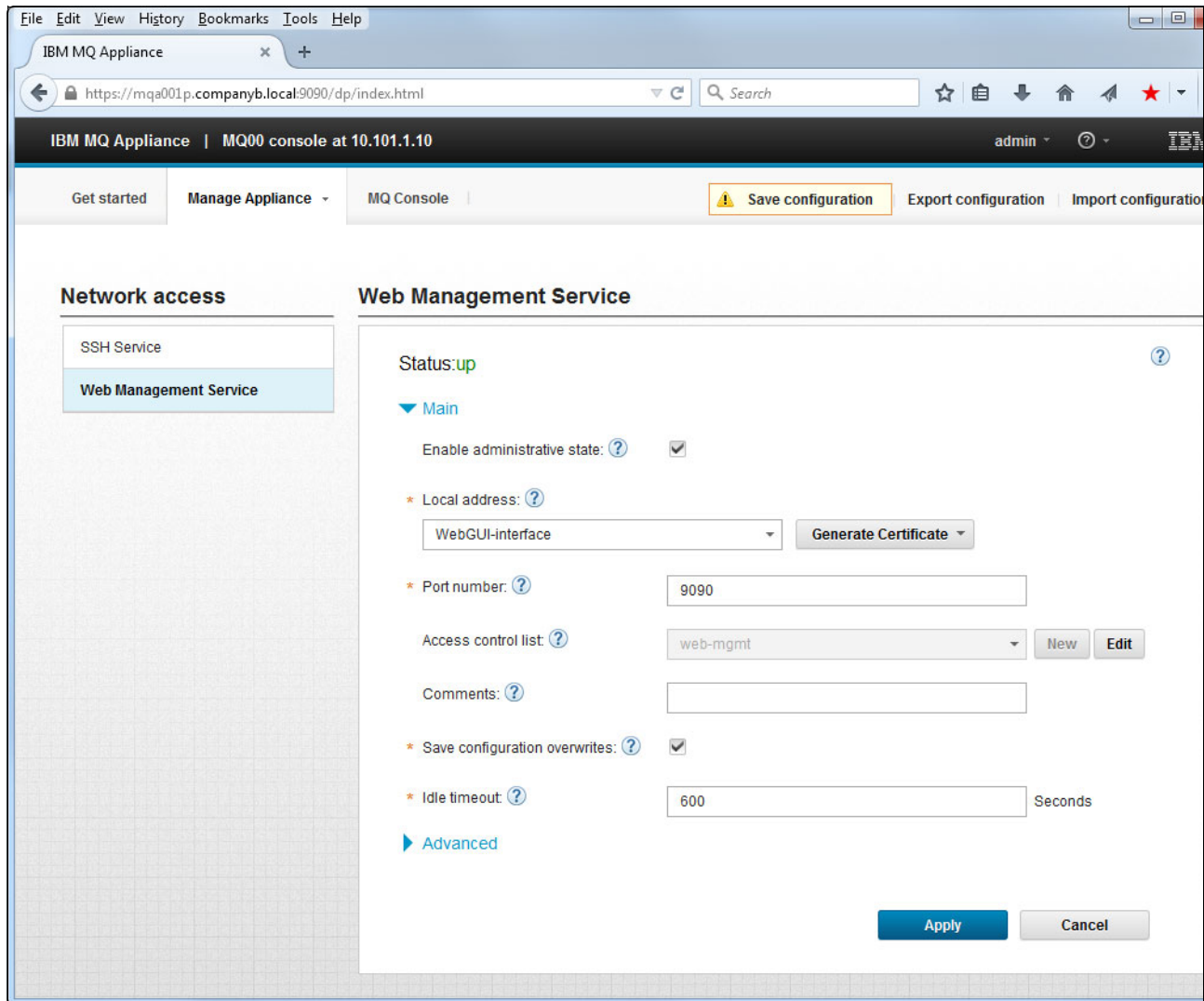


Figure 6-31 Web Management Service configuration

6.5 Appliance file system

On the appliance, there is no traditional file system available. Instead, file transfer on and off the appliance can be achieved by using universal resource indicators (URIs) as listed in Table 6-2. From the configuration mode in the CLI, you can enter `dir` to view a complete list of URIs.

Table 6-2 Useful URIs for the IBM MQ Appliance

URI	Purpose
mqbackup://	User and certificate backup and restore Queue manager configuration file (generated through <code>dmpmqcfg</code>) Client channel definition tables (CCDTs)
mqdiag://	Output of IBM MQ diagnostics tools (to be used on advice from the IBM MQ support team)

mqerr://	System error logs Queue manager error logs under 'qmgrs' subdirectory First-failure support technology (FFST™) files
mqpubcert://	Certificate requests Certificates
mqtemporary://	FFSTs before the file system is fully initialized
mqtrace://	IBM MQ trace Web UI trace, error logs and FFSTs (under /webui subdirectory)
mqwebui://	Saving and loading user dashboard json files. This can also be achieved via the IBM MQ Console UI directly and does not require use of the CLI copy command.
config://	Appliance configuration files
image://	New firmware images must go to this directory
logtemp://	Appliance logs
store://	Appliance configuration files

From the configuration mode in the CLI, the following commands are available with which you can perform basic file manipulation:

- ▶ **copy**: Place a file on the appliance or copy it to another system.
- ▶ **dir**: View files under a certain URI.
- ▶ **show file**: Print file contents to window (in plain text format).
- ▶ **delete**: Remove files.

These commands are shown in Example 6-13 in the context of generating and viewing a channel security certificate signing request.

Example 6-13 File manipulation in the context of IBM MQ channel security certificates

```
M2000(mqcli)# createcertrequest -m GTWY1HAP -dn "CN=HAGTWY1HAP, OU=/CLUSTER1P/,
OU=SA-W525, O=CompanyB, L=Raleigh, ST=NC, C=US"
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqcli)# exit
M2000# config
Global configuration mode
M2000(config)# dir mqpubcert://
      File Name                               Last Modified                               Size
      -----                               -
      GTWY1HAP_ibmwebspheremqgtwy1hap Jul 15, 2015 1:37:05 PM          1041
      973.4 MB available to mqpubcert://
M2000(config)# show file mqpubcert://GTWY1HAP_ibmwebspheremqgtwy1hap

-----BEGIN NEW CERTIFICATE REQUEST-----
MIICwTCCAakCAQAwfDELMakGA1UEBhMCVVMxCzAJBgNVBAgTAk5DMRAwDgYDVQQH
EwdSYWxlaWdoMREwDwYDVQQKEwhDb21wYW55QjEQMA4GA1UECzMHU0EtVzUyNTEU
MBlGA1UECxMLLONMVVNURVIXUC8xEzARBgNVBAMTCkhBR1RXWTFIQVAwggEiMAOG
CSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQRDSaWQyuLhiQ/0zVxPx4Wzakpw0ZdL
Fjd0HhQ/xG18JOaVez56TbnRKUi4ESC3Ssp6SkL8Ya10P1MjspYriuYmrq1irWbm
2q/BazYfHuvBNXDBMEf1NPVgY0yQYAVWnVFUVUk3yRV5x4aePAI8ezXzw7Kpq7ds
zyE9Ux5mgdwZh+DtrZNR/xyTU5tMREwgqXkl2+Jz/F6IRLET1kzTLMLKJzdJHyla
```

```

3EkgyF+j5zoyjFk8pjZyg039khhYeLlnBa/k4Lf19JRvzYHQofk4qKxVx+bN4sgd
3x48NgYEPcOKx+aox8tmLaQhS5an3foV/3P0cSR0MrDKrK/QfWcI4ufNagMBAAGg
ADANBgkqhkiG9w0BAQsFAA0CAQEUAU80KwQFBtbhPCMiVXoF7BP130RGE79KaGJub
rOxPz6dyY8DeDGNRo2MLbtEYXVyHayN/0B9qsIPiNozrQZ0Aw+9VQm45L7yEDwCI
NCfI7yn1bs4Evvwro8PypV0vyrMt0Esb0Kxjn9oFNkApf5ugYaV4mIAvR9nZwCdV
ONJEnKi+xXbLD7UfqyDtFjQMzNs51c1jFvT7T5U3NQshJYo1DzIE7TOSWKFda3Dx
rtMw8c8/KTFszB/YcdN4XLHJJvTm1SEr7cHg+B85WzAc0KHF/Mh0LEKZpZf0r1xe
Hkd1GLo1hT5z0znj2ue8v1nxCayM9jRPhbAT2aRqbuDnr7UajA==
-----END NEW CERTIFICATE REQUEST-----

```

```

M2000(config)# copy mqpubcert://GTWY1HAP_ibmwebspheremqgtwy1hap scp://user@<ip
address>//full/path/to/directory/
M2000(config)# exit
M2000# mqcli
M2000(mqcli)# deletecertrequest -m GTWY1HAP -label ibmwebspheremqgtwy1hap
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqcli)#

```

Example 6-14 shows the appliance OS way of deleting a file in the context of acquiring service support. Before the start of the example, the IBM MQ must-gather tool, **runmqras**, was run. Under these circumstances, you want to copy the tool output to an external server (as shown) to be passed on to IBM and then remove the file from the appliance.

Example 6-14 File manipulation in the context of appliance support

```

M2000# config
Global configuration mode
M2000(config)# dir mqdiag://

```

File Name	Last Modified	Size
runmqras_150715_140126.zip	Jul 15, 2015 2:01:04 PM	132195

```

1948.6 MB available to mqdiag://
M2000(config)# copy mqdiag://runmqras_150715_140126.zip scp://user@<ip
address>//full/path/to/directory/
File copy success
M2000(config)# delete mqdiag://runmqras_150715_140126.zip
File deletion successful
M2000(mqcli)#

```

Tip: You cannot set permissions on files on the IBM MQ Appliance. If files need specific permissions, you must set them on the origin machine and then copy them onto the appliance.

6.6 Firmware upgrades

Firmware upgrades on the IBM MQ Appliance follow the same rules that are used to upgrade DataPower appliances. To upgrade your appliance, you must download the latest scrypt3 file from the following IBM Fix Central website:

<http://www.ibm.com/support/fixcentral/>

An script3 file contains an entirely new image, which means that the level of IBM MQ and the level of the underlying OS can be upgraded. The use of an script3 file is the only way to upgrade your appliance, which often means that upgrading your appliance is significantly easier and quicker than the equivalent patching and software upgrade tasks on a conventional distributed system.

After you download the script3, we suggest that you place it on a secure copy (scp) server in a network position that the appliance can access so that it is accessible via scp. There is no scp server on the appliance, which means that you can only “pull” files onto it.

To copy the fix onto your appliance, you must use the **copy** command, which can be found in the configuration (config) mode of the CLI. An example of a successful copy is shown in Example 6-15. You must copy the script3 file into the image:// URI.

Tip: You can only access configuration mode as a privileged appliance user.

Example 6-15 Use of CLI copy command to copy a firmware upgrade onto an IBM MQ Appliance

```
M2000# config
M2000(config)# copy scp://user@server//full/path/to/script3/upgrade.script3
image://
Password: *****
File copy success
M2000(config)# flash
Flash configuration mode
M2000(config-flash)# boot image accept-license upgrade.script3
```

After the steps in the example are complete, the appliance checks that you want to proceed, followed by beginning the firmware upgrade. For complete details of an upgrade process, see Appendix B, “Transcript of IBM MQ Appliance firmware upgrade” on page 195.

Two appliances can be upgraded in an HA group without incurring downtime by suspending the HA group, upgrading the appliance, and rejoining the HA group on each appliance in turn. For more information about HA firmware upgrades, see the chapter about HA in the IBM MQ Appliance manual, which is available at this website:

http://www.ibm.com/support/knowledgecenter/#!/SS5K6E_1.0.0/com.ibm.mqa.doc/administering/ad00161_.htm



Creating queue managers

This chapter describes the practical advice for creating queue managers for the IBM MQ Appliance. The advice applies to standard queue managers and those managers that are intended to be used for high availability (HA).

The product documentation describes all of the options that are associated with the `crtmqm` command, which is used to create a queue manager. In this chapter, we discuss the most important options relating to creating queue managers on an IBM MQ Appliance.

This chapter includes the following topics:

- ▶ 7.1, “Preparing to create queue managers” on page 108
- ▶ 7.2, “Using the command line to create queue managers” on page 110
- ▶ 7.3, “Using the IBM MQ Console to create queue managers” on page 112
- ▶ 7.4, “Queue manager security” on page 112
- ▶ 7.5, “Customizing the queue manager” on page 118

7.1 Preparing to create queue managers

Before a queue manager is created, several factors should be considered and addressed. These considerations are briefly described in the following sections of this chapter.

A more detailed view of establishing the configuration, and calculating file system and log file sizes, is included following the section which describes creating queue managers.

7.1.1 User and group configuration

Consider the user configuration that the queue manager needs. These configurations are the user IDs that are assigned as Message Channel Agent Users (MCAUSERS). These user IDs can be members of one or more groups. Permission to administer or use IBM MQ objects can be granted to these users or to the groups.

Our recommendation is that these users and groups are defined in your corporate Lightweight Directory Access Protocol (LDAP), not as messaging users within the appliance. For more information, see Chapter 3, “Planning” on page 21.

Decide which users are needed for different messaging channels that are used by the new queue manager. If permissions are granted to groups rather than principals, the group names also must be defined.

Request the security or identity management team of your organization to create the accounts for these user IDs.

Apart from MCAUSERS for messaging channels, it is likely that you also need the following types of users for MQI or client channels:

- ▶ Regular users who log on by using user ID and password to administer the queue manager in some way.
- ▶ Administration and monitoring user IDs that are used by monitoring applications.
- ▶ Application user IDs in which the application authenticates with user ID and password.
- ▶ Application user IDs in which the application authenticates with a Transport Layer Security (TLS) certificate.

These classes of users can have different attributes from an LDAP perspective. The users can be created in different organizational units (OUs). Some of the users might not have passwords. Some users might be in a disabled state to ensure that they cannot be used to log on to an actual system.

Your organization might also have rules regarding separation of resource groups and role groups, and different types of groups might again be stored in different OUs. If separate resource groups and role groups are used within your organization, you must use the NESTGRP(YES) option in your AUTHINFO TYPE(IDPWLDAP) record.

From an IBM MQ perspective, the following critical factors must be considered:

- ▶ BASEDNU must be high enough in the LDAP tree that all users that are required by the queue manager can be found by a subtree search for the short user name.
- ▶ BASEDNG must be high enough in the LDAP tree that all groups that are required by the queue manager can be found by a subtree search for the group name.
- ▶ Short user IDs must be unique within the subtree that IBM MQ can see.
- ▶ Group names must be unique within the subtree that IBM MQ can see.

- ▶ Short names that are assigned to user records must be no longer than 12 characters.

7.1.2 Log file size

All queue managers that are running on the IBM MQ Appliance use circular logs. The following default size of these logs is small and likely to be too small for many potential messaging environments:

- ▶ Three primary logs
- ▶ Two secondary logs
- ▶ 4096 blocks per log file
- ▶ Each block is 4 KB
- ▶ Total default log space is approximately 81 MB

Note: The number of log files can be increased after the queue manager is created. The size of each log file cannot be changed without re-creating the queue manager; therefore, consider starting with a larger number of blocks in each log file than the default. The maximum number of blocks in each file is 65535.

A more reasonable starting point for circular logs might be the following configuration:

- ▶ 10 primary logs
- ▶ Five secondary logs
- ▶ 32768 blocks per log file
- ▶ Total log space that is allocated at creation is 1920 MB

7.1.3 Queue file size

The size of a queue file is entirely dependent on the workload of the queue manager. This issue is not driven by the queue manager; instead, it is driven by the behavior of applications that connect to the queue manager and put and get messages.

Although a queue manager is designed to provide asynchronous messaging by using queues as destinations (a means to store messages when a receiving application is not available to immediately handle an incoming message), it is optimized to perform best when the queues are not deep. For this reason, queues are not considered to be the correct place to store messages for extended periods, especially when other applications require faster access to messages on the same queue as a deep queue.

However, there can be times when the size of queues can expand rapidly, and the queue manager file system must cope with the demand for space. It is better to overestimate these requirements because the file system that is used by each queue manager cannot be expanded after it is created.

7.1.4 File system size

When a queue manager is created on an appliance, a file system is created for it. The default size for this file system is 64 GB. This file system is allocated from the available disk space from the appliance storage. For the IBM MQ Appliance A, B, and B+ models, the factory shipped disks are 1 TB.

In some environments, this disk space might be insufficient to store all messages that a queue manager must hold at one time, although this issue is likely to be a rare situation unless messages are held for long periods.

As with any queue manager, the IBM MQ Appliance is optimized to move and process messages and not to store them as a database might do.

If your application scenarios for a queue manager indicate that only 4 GB of total messages must be stored at once, allowing for message overheads, some growth, log files, error report files, trace files and such, a file system size of 10 - 16 GB is likely sufficient.

7.2 Using the command line to create queue managers

It is necessary to log in to the appliance and switch into the command-line mode (**mqcli**) before the **crtmqm** command can be run.

A list of options can be displayed for the command as shown in the following example:

```
crtmqm ?
```

Output from the **crtmqm help** command is shown in Example 7-1.

Example 7-1 Output of crtmqm help

```
M2000# crtmqm ?
Usage: crtmqm [-c Text] [-d DefXmitQ] [-h MaxHandles]
              [-p Port] [-t TrigInt] [-u DeadQ] [-x MaxUMsgs]
              [-lp LogPri] [-ls LogSec] [-lf LogFileSize]
              [-fs FileSystemSize] [-sx] QMgrName

-c    Descriptive text.
-d    Default transmission queue name.
-fs   File system size in gigabytes (GB).
-h    Maximum number of handles per connection handle.
-lf   Log file size, specified in units of 4 KB pages.
-lp   Primary log files allocated when the queue manager is created.
-ls   Secondary log files allocated when the primary files are exhausted.
-p    Port number for the managed TCP listener.
-sx   Make this queue manager a high availability (HA) queue manager.
-t    Trigger interval in milliseconds.
-u    Dead-letter queue name.
-x    Maximum number of uncommitted messages under any one syncpoint.
M2000(mqcli)#
```

7.2.1 Default values

The simplest form of the **crtmqm** command allows a user to specify only the name of the queue manager that they want to be created. When this command is used, default values are applied for all possible creation options.

Example 7-2 shows a queue manager that is called TESTQM being created.

Example 7-2 Creating a queue manager with default values

```
M2000(mqcli)# crtmqm GTWY1HAP
Please wait while 64 GB file system is initialized for queue manager 'GTWY1HAP'.
IBM MQ Appliance queue manager created.
The queue manager is associated with installation 'MQAppliance'.
Creating or replacing default objects for queue manager 'dspmq'.
Default objects statistics : 83 created. 0 replaced. 0 failed.
```

```
Completing setup.  
Setup completed.  
5724-H72 (C) Copyright IBM Corp. 1994, 2014.  
M2000(mqccli)#
```

As is shown in the first line of information during the creation process, there is a 64 GB portion of the file system that is allocated with which this queue manager can function. This portion is a large area regarding other attributes that might be applied.

Log file pages

The log data is held in a series of files called log files. The log file size is specified in units of 4 KB pages.

The default number of log file pages is 4096, which gives a log file size of 16 MB. The minimum number of log file pages is 64 and the maximum is 65535. Therefore, the maximum size that a log file can be is 256 MB.

Note: The size of the log files for a queue manager is specified during queue manager creation, and cannot be changed. Therefore, all primary and secondary log files for a queue managers are always the same size.

Primary log files

Primary log files (-lp) specify the log files that are allocated when the queue manager is created.

Primary log files are created and sized according to the log file pages value when the queue manager is created. The default number of log files that can be created is three; however, this amount can be changed by using the -lp option of the **crtmqm** command.

There can be 2 - 510 primary log files; the default is three. The total number of primary and secondary log files must not exceed 511 and must not be less than 3.

You can change this value after the queue manager is created. However, the change is not effective until the queue manager is restarted.

Secondary log files

Secondary log files (-lf) specify the log files that are allocated when the primary files are exhausted.

There can be 2 - 509 secondary log files; the default is two. The total number of primary and secondary log files must not exceed 511 and must not be less than 3.

You can change this value after the queue manager is created. However, the change is not effective until the queue manager is restarted.

File system size

File system size (-ls) specifies that the queue manager is created with the file system size `FileSystemSize`.

`FileSystemSize` is a numeric value, which is specified in GB. You can specify a value in MB by entering the value followed by the character M. For example, to specify a `FileSystemSize` of 3 GB, enter 3. To specify a `FileSystemSize` of 1024 MB, enter 1024M. The default value is 64.

Note: After a queue manager is created, you cannot resize the file system. Ensure that the value that is specified here is sufficient for the current and any future workload.

7.3 Using the IBM MQ Console to create queue managers

The IBM MQ Console can be used to create a queue manager; however, many options cannot be specified and defaults are taken. The use of the Web UI to create queue managers can be appropriate for a development environment on which each queue manager can have a short lifetime and be used by only one or two developers. For full control of queue manager creation, use the command line.

7.4 Queue manager security

Queue managers on the IBM MQ Appliance can use special messaging users and groups. Users that provide credentials can then be authenticated against the local user account and authorized based on object authority manager (OAM) authorities granted to the user, or to the groups to which the user belongs.

However, this configuration puts a burden of identity management onto the IBM MQ Appliance administrator, especially for other administrative accounts, such as limited administrators. The IBM MQ Administrator must provide the following services that are normally part of enterprise identity management:

- ▶ Account management
- ▶ Password reset
- ▶ Password expiry management
- ▶ Password complexity control
- ▶ Group membership
- ▶ Role mapping

These functions must be performed independently on each appliance because the appliances do not have a scriptable mechanism to replicate user and group information between appliances.

Of even more concern, correct behavior of queue manager security in an HA group requires that all users and groups that are referenced by OAM are identical in both appliances. However, there is no simple mechanism to implement or validate that the user and group configuration matches this requirement.

Therefore, the most practical approach to user and group management for queue managers that are running on the IBM MQ Appliance is to set up each queue manager to use LDAP for user authentication and for authorization mapping.

Authorization mapping is the process of discovering to which groups a specified user is a member. The actual authorization decision is then based on the OAM rules that are defined within the queue manager. These rules are always consistent with whichever HA group appliance is hosting the queue manager because the OAM rules are stored as part of the queue manager.

7.4.1 LDAP considerations

Most LDAP schemas for users and groups can support the requirements for IBM MQ queue managers that are running on the IBM MQ Appliance.

The following requirements must be met for authorization mapping to work:

- ▶ For an authorization method that is based on groups that are listing the users who are members, use AUTHORMD(SEARCHGRP). The full distinguished name (DN) of each user must appear in the attribute that lists each user that is a member of the group. The queue manager forms a query that includes the full user DN, not just the short name. This attribute is commonly called `member`, although this name is not required. The attribute can have another name.
- ▶ For an authorization method that is based on users that are listing the groups that they are in, AUTHORMD(SEARCHUSR) is used. The full DN of each group of which the user is a member must appear in the attribute that lists these groups. The queue manager forms a query that includes the full DN of the group, not just the short or common name. This attribute is commonly called `memberOf`, although this name is not required. The attribute can have another name.
- ▶ Users must be a member of a group and connected by using one of these two methods. The primary group ID method of connecting a user to a group is not sufficient for the queue manager to discover membership in an LDAP context.

Many directory implementations include `member` lists in each group and `memberOf` lists in each user. If so, it is normally best to configure the queue manager to use AUTHORMD(SEARCHUSR) as it is more efficient in most directory implementations.

The LDAP server that was used when this book was written did not support defining group memberships in user records. Therefore, the examples in this chapter use the AUTHORMD(SEARCHGRP) method.

Because the LDAP server is used for authentication, the password (or a representation of it) flows between the queue manager and the LDAP server. Therefore, it is important that TLS is used to secure the connection. Ensure that the LDAP server that is used for authenticating users to the queue manager supports LDAPS and has secure certificate and key management practices.

Secure communication to the LDAP server is controlled by the SECCOMM parameter of the AUTHINFO object. An example of an AUTHINFO object is described in the next section.

7.4.2 Connecting the queue manager to the LDAP

To direct each queue manager to use LDAP for authentication and authorization information, the following items must be configured:

- ▶ Set up TLS trust of the LDAP server. This setup is done by copying the certificate files to the appliance, and then by using the **addcert** command to add each certificate to the keystore of the queue manager.
- ▶ Refresh TLS security to ensure that the new certificates are available to the queue manager.
- ▶ Create an authentication information (AUTHINFO) record that describes how the queue manager connects to the LDAP server.
- ▶ Set the queue manager to use the new AUTHINFO record.

Setting up trust of the LDAP server in the queue manager

For the LDAPS protocol (secure LDAP), trust is based on the signer of the certificate that is presented by the LDAP server. The signer chain must be trusted by the queue manager, which means that it must be in the queue manager truststore.

The certificate chain of the LDAP server must be obtained. This process is outside the scope of this section, but certificates might be obtained from the LDAP provider (perhaps your company security team), or you can use a tool to extract certificates from a TLS exchange. For example, `openssl s_client -showcerts`, might be used to capture the certificates, which can then be placed in Privacy Enhanced Mail (PEM) format files.

There are other parameters to the `openssl` command that are needed. These parameters can be found via a simple web search or by using the `openssl` command with a “?” at the end of the command. After you have the certificate files available, they must be copied to the appliance file store so that they can be added to the queue manager keystore.

As with all files that must be copied to or from the IBM MQ Appliance, the appliance is in charge. This configuration provides a reduced security exposure because a potential attacker cannot push files onto an IBM MQ Appliance or pull files off the appliance.

As shown in Example 7-3, the certificate files for the CA trust chain are stored in the `/var/tmp/certs` directory on the `mqsupportsvr.companyb.local` server. The certificate files are called `CompanyBrootCA.pem` and `CompanyBSigner.pem`. The `mqsupportsvr` server is a server that is running a service that supports the secure copy protocol (scp). The server can be a UNIX server or a Windows server that is running software to provide the service.

Certificates that are added to a queue manager keystore must be placed in the `mqpubcert:` directory of the appliance, as shown in Example 7-3. The copy command can be used only when in configuration mode. This mode is reached by using the `configure terminal` command. As with all appliance commands, this command can be abbreviated to the shortest unique value, which in this case is `co`.

Example 7-3 Copying CA certificates to the appliance mqpubcert: directory

```
M2000# configure terminal
M2000(config)# copy
scp://neil@mqsupportsvr.companyb.local//var/tmp/certs/CompanyBrootCA.pem
mqpubcert://CompanyBrootCA.pem
Password: *****
File copy success
M2000(config)# copy
scp://neil@mqsupportsvr.companyb.local//var/tmp/certs/CompanyBIssuer.pem
mqpubcert://CompanyBIssuer.pem
Password: *****
File copy success
M2000(config)#
```

After the certificates are available on the appliance file system, the certificates can be added into the queue manager keystore by using the `addcert` command. Example 7-4 shows moving from the appliance configuration command line to the IBM MQ command line and setting up the trusted certificates in the keystore for the GTWY1HAP queue manager.

Example 7-4 Adding certificates to the queue manager keystore

```
M2000(config)# exit
M2000# mqcli
```

```
M2000(mqcli)# addcert -m GTWY1HAP -label "CompanyB root CA" -file
CompanyBrootCA.pem -format ascii
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqcli)# addcert -m GTWY1HAP -label "CompanyB issuer CA" -file
CompanyBIssuer.pem -format ascii
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqcli)#
```

Making the certificates available to the active queue manager

Queue managers cache the contents of the keystore for high performance. They do not reload the cache unless the queue manager restarts or a directive is given to the queue manager that tells it to reload the keystore. The **REFRESH SECURITY(*) TYPE(SSL)** command that is used to tell the queue manager to reload the keystore is shown in Example 7-5.

Caution: The **REFRESH SECURITY(*) TYPE(SSL)** command causes all running channels that are using TLS to be stopped. They then restart based on the normal retry parameters or the application behavior for server connection channels. This issue can cause application interruption, delivery delays, or loss of non-persistent messages. If a refresh is needed, it must be planned with care.

Example 7-5 Reloading the keystore

```
M2000(mqcli)# runmqsc GTWY1HAP
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Starting MQSC for queue manager GTWY1HAP.

REFRESH SECURITY(*) TYPE(SSL)
  1 : REFRESH SECURITY(*) TYPE(SSL)
AMQ8560: IBM MQ Appliance security cache refreshed.
END
  2 : END
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
M2000(mqcli)#
```

Creating an authentication information record

The details of how the queue manager connects to the LDAP server and how the LDAP hierarchy is structured (including attribute names, which the queue manager must use) are included in an authentication information (AUTHINFO) record. This record is created by using the **runmqsc** command on the appliance.

For more information about the AUTHINFO record, see the following IBM Knowledge Center website:

http://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.ref.adm.doc/q085490_.htm?lang=en

In Example 7-6 on page 116, the following configuration is shown:

- ▶ LDAP servers are `ldap1.companyb.local` and `ldap2.companyb.local`.
- ▶ Authenticated users are adopted as the MCAUSER of the channel (for more information, see Chapter 10, “High availability” on page 149).
- ▶ The LDAP objectClass for a group of users is `posixGroup`.

- ▶ The LDAP objectClass for a user is posixAccount.
- ▶ All users records are suffixed with OU=users,DC=companyb,dc=local.
- ▶ All group records are suffixed with OU=groups,DC=companyb,DC=local.
- ▶ Group records contain a list of users that is stored as memberUid attributes.
- ▶ The queue manager logs on (binds) to the LDAP with user UID=mqbind,OU=users,DC=companyb,DC=local so that it can search for records. The password for this account is mqbind.
- ▶ The short name of each user is stored in the uid attribute.
- ▶ The name of the group is the cn attribute.
- ▶ Nested groups cannot be used (the site is not using nesting to support separation of resource groups and role groups).
- ▶ The LDAP server is accessed by using TLS, but the queue manager does not use its certificate to authenticate. The bind account is used instead.

Example 7-6 Creating AUTHINFO record for CompanyB LDAP server

```
M2000(mqcli)# runmqsc GTWY1HAP
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Starting MQSC for queue manager GTWY1HAP.
DEFINE AUTHINFO(COMPANYB.LDAP) -
  AUTHTYPE(IDPWLDAP) -
  ADOPTCTX(YES) -
  DESCR('Use CompanyB LDAP for authentication and authorization') -
  CONNAME('ldap1.companyb.local(636),ldap2.companyb.local(636)') -
  CHCKCLNT(REQDADM) -
  CHCKLOCL(OPTIONAL) -
  CLASSGRP('posixGroup') -
  CLASSUSR('posixAccount') -
  FAILDLAY(1) -
  FINDGRP('memberUid') -
  BASEDNG('OU=groups,DC=companyb,DC=local') -
  BASEDNU('OU=users,DC=companyb,dc=local') -
  LDAPUSER('UID=mqbind,OU=users,DC=companyb,DC=local') -
  LDAPPWD('mqbind') -
  SHORTUSR('uid') -
  GRPFIELD('cn') -
  USRFIELD('uid') -
  AUTHORMD(SEARCHGRP) -
  NESTGRP(NO) -
  SECCOMM(ANON) -
  REPLACE
  1 : DEFINE AUTHINFO(COMPANYB.LDAP) -
    : AUTHTYPE(IDPWLDAP) -
    : ADOPTCTX(YES) -
    : DESCR('Use CompanyB LDAP for authentication and authorization') -
    : CONNAME('ldap1.companyb.local(636),ldap2.companyb.local(636)') -
    : CHCKCLNT(REQDADM) -
    : CHCKLOCL(OPTIONAL) -
    : CLASSGRP('posixGroup') -
    : CLASSUSR('posixAccount') -
    : FAILDLAY(1) -
    : FINDGRP('memberUid') -
    : BASEDNG('OU=groups,DC=companyb,DC=local') -
```



```

      :   BASEDNU('OU=users,DC=companyb,dc=local') -
      :   LDAPUSER('UID=mqbind,OU=users,DC=companyb,DC=local') -
      :   LDAPPWD('mqbind') -
      :   SHORTUSR('uid') -
      :   GRPFIELD('cn') -
      :   USRFIELD('uid') -
      :   AUTHORMD(SEARCHGRP) -
      :   NESTGRP(NO) -
      :   SECCOMM(YES) -
      :   REPLACE
AMQ8563: IBM MQ Appliance authentication information object created.
      :
END
      2 : END
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
M2000(mqccli)#

```

Note: The BASEDNU and BASEDNG values are the start point for a subtree search within the directory. Users and groups do not need to have the exact base distinguished name listed.

For example, with the configuration that is shown in Example 7-6 on page 116, an account with the DN `uid=_mqrepo,OU=serviceaccounts,ou=users,dc=company,dc=local` is found by the search.

However, the DN `uid=_mqrepo,OU=serviceaccounts,dc=company,dc=local` is not within the `ou=users` subtree, and is not found by the search.

Ensure that all users and groups that are required by the queue manager can be found within the subtrees that are specified in the BASEDNU and BASEDNG values.

Setting the queue manager to use LDAP authentication and authorization

This step is simple, but must be done with care. If the LDAP configuration does not work, client access to the queue manager is lost, and receiver channels do not restart.

Care: Test LDAP configurations carefully before implementing them in important environments. Always test any change to LDAP configurations immediately and as much as possible.

Tip: Make note of the connection authentication settings before activating the LDAP configuration. These settings can be found by using `DISPLAY QMGR CONNAUTH` in the `runmqsc` command.

Example 7-7 on page 118 shows the command to display the current CONNAUTH setting and alter the queue manager configuration to use the new AUTHINFO record that was created in Example 7-6 on page 116. It then refreshes connection authentication (CONNAUTH) security so that the queue manager starts to use LDAP-based authentication.

Example 7-7 Activating LDAP authentication and authorization

```
M2000(mqcli)# runmqsc GTWY1HAP
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Starting MQSC for queue manager GTWY1HAP.

DIS QMGR CONNAUTH
  1 : DIS QMGR CONNAUTH
AMQ8408: Display Queue Manager details.
      QMNAME(GTWY1HAP)
      CONNAUTH(SYSTEM.DEFAULT.AUTHINFO.IDPWOS)
ALTER QMGR CONNAUTH(COMPANYB.LDAP)
  2 : ALTER QMGR CONNAUTH(COMPANYB.LDAP)
AMQ8005: IBM MQ Appliance queue manager changed.
REFRESH SECURITY TYPE(CONNAUTH)
  3 : REFRESH SECURITY TYPE(CONNAUTH)
AMQ8560: IBM MQ Appliance security cache refreshed.
END
  4 : END
3 MQSC commands read.
No commands have a syntax error.
All valid MQSC commands were processed.
M2000(mqcli)#
```

Note: If testing of LDAP-based authentication and authorization fails, the previous configuration can be restored with another ALTER QMGR command, followed by REFRESH SECURITY(*) TYPE(CONNAUTH).

7.5 Customizing the queue manager

When a queue manager is created on the IBM MQ Appliance, it has several attributes that must be updated before it is ready to be used by application.

This section describes the options that are available and demonstrates how to configure these options for queue managers that are hosted on the IBM MQ Appliance. Many of the mechanisms that are shown are the same as are used for normal queue managers. Other mechanisms are specific to the appliance.

7.5.1 Limit max channels

On a normal queue manager, the maximum number of channels and the maximum number of active channels are set by using the `mq.ini` file. This ability requires a queue manager restart to activate the change.

On the IBM MQ Appliance, the `qm.ini` file sets unlimited values for these parameters. The `MAXINST` and `MAXINSTC` attributes of channels are used instead. These parameters are controlled by using `runmqsc`, **ALTER CHANNEL**.

Do not change the provided values for the maximum channel instances in the `qm.ini` file. Do not try to limit these at a queue manager level.

7.5.2 Configuring the number of logs

The correct number of primary and secondary logs must be created with the queue manager by using parameters of the `crtmqm` command.

If more logs are needed or too many were allocated, these logs can be changed by using the `setmqini` command to set values in the `qm.ini` file. On the appliance, it is not possible to directly edit the `qm.ini` file.

Example 7-8 shows two commands that change the number of primary and secondary log files for the GTWY1HAP queue manager.

Example 7-8 Configuring the number of log files

```
M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# setmqini -m GTWY1HAP -s Log -k LogPrimaryFiles -v 10
Key LogPrimaryFiles was successfully updated in stanza Log for queue manager
GTWY1HAP.
M2000(mqcli)# setmqini -m GTWY1HAP -s Log -k LogSecondaryFiles -v 5
Key LogSecondaryFiles was successfully updated in stanza Log for queue manager
GTWY1HAP.
M2000(mqcli)#
```

7.5.3 Configuring channel behavior

IBM MQ provides options that can help make channel reconnection faster after failover occurs. These options are controlled through `mq.ini` file parameters for `AdoptNewMCA` and some related parameters. Although many sites do not need to change these settings, it might be critical for some.

Attention: `AdoptNewMCA` can produce unpredictable results when `FASTPATH` channels are used. `FASTPATH` channels are the default on the IBM MQ Appliance, which is different than normal queue managers where the default is `STANDARD`. If you must use `AdoptNewMCA`, consider changing the `MQIBindType=STANDARD` instead of the default of `FASTPATH`.

Example 7-9 shows how `AdoptNewMCA` can be activated for an appliance-hosted queue manager.

Example 7-9 Setting up AdoptNewMCA for a queue manager

```
M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# setmqini -m GTWY1HAP -s Channels -k MQIBindType -v STANDARD
Key MQIBindType was successfully updated in stanza Channels for queue manager
GTWY1HAP.
M2000(mqcli)# setmqini -m GTWY1HAP -s Channels -k AdoptNewMCA -v ALL
Key AdoptNewMCA was successfully updated in stanza Channels for queue manager
GTWY1HAP.
M2000(mqcli)# setmqini -m GTWY1HAP -s Channels -k AdoptNewMCACheck -v QM,NAME
Key AdoptNewMCACheck was successfully updated in stanza Channels for queue
manager GTWY1HAP.
M2000(mqcli)#
```

7.5.4 Configuring TLS certificate checking

Several options can be set related to the way that a queue manager checks the validity of certificates that are presented to it.

The options can affect how the queue manager checks Certification Revocation Lists (CRLs) and whether failure to locate a CRL or Online Certificate Status Protocol (OCSP) responder affects the decision to accept or reject the presented certificate.

If your certificates include CRL or OCSP information but the addresses cannot be reached reliably, it might be critical to change the default values.

Example 7-10 demonstrates deactivating the requirement for OCSP responders that are named in certificates to be available. This configuration might not be appropriate for your environment. For effective trust of certificates, revocation information must be available to queue managers and checking of that information must be enforced wherever possible.

Example 7-10 Disable OCSP and CDP checking

```
M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# setmqini -m GTWY1HAP -s SSL -k CDPCheckExtensions -v NO
Key CDPCheckExtensions was successfully updated in stanza SSL for queue manager
GTWY1HAP.
M2000(mqcli)# setmqini -m GTWY1HAP -s SSL -k OCSPAuthentication -v OPTIONAL
Key OCSPAuthentication was successfully updated in stanza SSL for queue manager
GTWY1HAP.
M2000(mqcli)# setmqini -m GTWY1HAP -s SSL -k OCSPCheckExtensions -v NO
Key OCSPCheckExtensions was successfully updated in stanza SSL for queue
manager GTWY1HAP.
M2000(mqcli)#
```

Attention: Configuring your queue manager as shown in this section reduces the reliability of certificate-based authentication. It can allow a revoked certificate to be used to connect to your queue manager.

7.5.5 Binding the listener to the data interface

When the queue manager is created on an appliance, a listener can automatically be created. This process always is done when a queue manager is created by using the IBM MQ Console. It is controlled by the **-p** parameter of the **crtmqm** command.

This automatically created TCP listener is called `SYSTEM.LISTENER.TCP.1`. It is created with the `IPADDR` attribute set to blank, which means that the queue manager listener binds to all available network interfaces.

In secure environments, it is often not permitted for systems to bind listeners to all interfaces. Specifically, it is often not allowed to bind to management and data interfaces.

This example shows how to configure the automatically defined listener to bind to the data interface, which is described in Chapter 6, “Appliance administration” on page 55.

In Chapter 6, “Appliance administration” on page 55, several Ethernet interfaces were connected and aggregated to form Link Aggregation la10. A VLAN interface was then defined, called vl01 with an IP address and subnet information in addition to a VLAN ID.

A host alias named Data-interface was created, which the queue manager can use to bind to the correct interface address. The same name for the interface can be used on any appliance because the same Host Alias can contain different IP addresses on different appliances.

Example 7-11 shows the command to alter the listener so that it binds to the correct interface. It then restarts the listener to force the new configuration to come active and displays the status of the listener to show that it is now bound to a specific IP address.

Example 7-11 Binding the listener to a specific interface address

```

M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# runmqsc GTWY1HAP
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Starting MQSC for queue manager GTWY1HAP.

ALTER LISTENER(SYSTEM.LISTENER.TCP.1) TRPTYPE(TCP) IPADDR('Data-interface')
  1 : ALTER LISTENER(SYSTEM.LISTENER.TCP.1) TRPTYPE(TCP)
IPADDR('Data-interface')
AMQ8623: IBM MQ Appliance listener changed.
STOP LISTENER(SYSTEM.LISTENER.TCP.1)
  2 : STOP LISTENER(SYSTEM.LISTENER.TCP.1)
AMQ8706: Request to stop IBM MQ Appliance Listener accepted.
START LISTENER(SYSTEM.LISTENER.TCP.1)
  3 : START LISTENER(SYSTEM.LISTENER.TCP.1)
AMQ8021: Request to start IBM MQ Appliance listener accepted.
DIS LSSTATUS(SYSTEM.LISTENER.TCP.1) ALL
  4 : DIS LSSTATUS(SYSTEM.LISTENER.TCP.1) ALL
AMQ8631: Display listener status details.
LISTENER(SYSTEM.LISTENER.TCP.1)          STATUS(RUNNING)
PID(692846)                               STARTDA(2015-07-29)
STARTTI(18.19.58)                         DESCR( )
TRPTYPE(TCP)                             CONTROL(QMGR)
IPADDR(::ffff:10.101.0.11)                PORT(4001)
BACKLOG(100)

END
  5 : END
4 MQSC commands read.
No commands have a syntax error.
All valid MQSC commands were processed.
M2000(mqcli)#

```

To provide improved separation between queue managers, secondary IP addresses or separate VLAN interfaces can be used. In this type of environment, a host-alias is defined for each of the defined IP addresses. The host-alias might be the name of the queue manager that will use that specific address. The listener definition can then bind to the specific IP address, which is used by that queue manager and no other. This configuration allows all queue managers on an appliance to listen on the same TCP port number, but different interface addresses.

If IP address separation is required, outgoing channels must also bind to the same queue manager-specific address. This requirement can be enforced by using LOCLADDR for conventional channels, or the (undocumented) MQ_LCLADDR environment variable. For information about the MQ_LCLADDR environment variable, see Example 9-3, “Define a channel with MCAUSER” on page 143.



IBM MQ object security

The most important feature of object security on the IBM MQ Appliance M2000 is that it is almost entirely the same object security as on other distributed queue managers.

This chapter describes aspects of the Object Authority Manager (OAM) as it relates to the IBM MQ Appliance. In particular, the differences between other queue managers and those managers that are hosted on the IBM MQ Appliance are reviewed.

In addition to the specifics of OAM security for queue managers on the appliance, this chapter begins by describing setting up a queue manager to use Lightweight Directory Access Protocol (LDAP) for authentication and authorization. The reason for the use of LDAP rather than local appliance accounts and groups is described in Chapter 3, “Planning” on page 21.

The queue manager that is used for examples in this chapter is called GTWY1HAP.

The chapter also includes information about connecting IBM MQ Explorer or `runmqsc -c` to a queue manager on an IBM MQ Appliance that was set up in the examples.

This chapter includes the following topics:

- ▶ 8.1, “Queue manager security” on page 124
- ▶ 8.2, “Object security” on page 124
- ▶ 8.3, “Summary” on page 135

8.1 Queue manager security

Queue managers that are running on the IBM MQ Appliance M2000 can use local accounts and groups for authentication and authorization. However, this configuration imposes security management burdens and compliance issues on the appliance administrator, who is most likely the IBM MQ administrator.

Chapter 7, “Creating queue managers” on page 107 describes the use of LDAP for group membership information and to authenticate users who are connecting to a queue manager. The user ID and group names can then be used to grant authorities by using the OAM. The queue manager enforces these authorities when applications or administrators access objects on the queue manager.

8.2 Object security

When an external LDAP repository handles connection authentication and authorization, OAM rules store authorized users and groups by using the full distinguished name of the LDAP object. Therefore, any rules that are in place before LDAP-based security is in place does not have any effect.

The OAM rules are managed by using the normal IBM MQ commands or Programmable Command Facility (PCF) messages and are stored within the queue manager. The LDAP is used to provide authentication and group membership only.

8.2.1 Enable remote administration

When LDAP-based security is first activated, all security rules in the queue manager cease to function because the identity format that is used by the OAM changes. Only administrative tasks that are done by using appliance administration accounts (such as the admin account) continue to function.

Until new OAM authorizations are granted, the queue manager can be configured only by using the IBM MQ Appliance Web UI or IBM MQ Configuration mode from the command-line interface (CLI). After OAM is configured, it can be administered by using the `runmqsc -c` command or IBM MQ Explorer.

You must complete the following prerequisite tasks before enabling queue manager administration that uses LDAP accounts:

1. Define an LDAP group that grants IBM MQ administration rights to members.
2. Add IBM MQ administrator user IDs to the LDAP group.
3. Grant administration authority to the group.

The first two of these steps are outside the scope of this book; however, we can describe the results and then use them in an example to grant full administration capability on the queue manager.

The example that we use in this section is based on the following information:

- ▶ The LDAP group for full administrators of production queue managers is `cn=mqadmin`.
- ▶ Users `neil` and `rufus` are members of this group.
- ▶ The queue manager which will be administered is `GTWY1HAP`.

Figure 8-1, Figure 8-2, and Figure 8-3 on page 126 show the tree view of the directory (Figure 8-1), and the details of a user (Figure 8-2) and a group (Figure 8-3 on page 126) within the directory to provide context for the commands that are shown in our example.

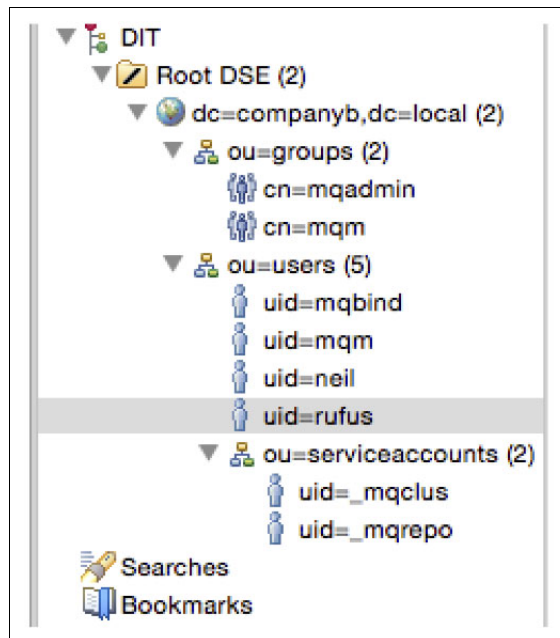


Figure 8-1 Tree view of the LDAP directory

DN: uid=neil,ou=users,dc=companyb,dc=local	
Attribute Description	Value
objectClass	account (structural)
objectClass	posixAccount (auxiliary)
objectClass	shadowAccount (auxiliary)
objectClass	top (abstract)
cn	neil
gidNumber	100
homeDirectory	/home/neil
uid	neil
uidNumber	16859
gecos	neil
loginShell	/bin/bash
shadowLastChange	0
shadowMax	0
shadowWarning	0
userPassword	CRYPT hashed password

Figure 8-2 User detail from the LDAP directory

DN: cn=mqadmin,ou=groups,dc=companyb,dc=local	
Attribute Description	Value
objectClass	posixGroup (structural)
objectClass	top (abstract)
cn	mqadmin
gidNumber	100
memberUid	uid=neil,ou=users,dc=companyb,dc=local
memberUid	uid=rufus,ou=users,dc=companyb,dc=local

Figure 8-3 Group detail from the LDAP directory

In Example 8-1, the `-e` option is used on the `runmqsc` command to suppress echoing of the commands to reduce the amount of output in the example. For more information about the commands that are included here (without the `runmqsc` response text), see Appendix D, “Commands to enable an LDAP authenticated administrator” on page 209.

Example 8-1 Grant full administration rights to IBM MQ administrators group

```

M2000(mqcli)# runmqsc -e GTWY1HAP
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Starting MQSC for queue manager GTWY1HAP.
SET AUTHREC -
  PROFILE('**') -
  OBJTYPE(AUTHINFO) -
  GROUP('mqadmin') -
  AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
  PROFILE('**') -
  OBJTYPE(CHANNEL) -
  GROUP('mqadmin') -
  AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
  PROFILE('**') -
  OBJTYPE(CLNTCONN) -
  GROUP('mqadmin') -
  AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
  PROFILE('**') -
  OBJTYPE(COMMINFO) -
  GROUP('mqadmin') -
  AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
  PROFILE('**') -
  OBJTYPE(LISTENER) -
  GROUP('mqadmin') -
  AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -

```

```

    PROFILE('**') -
    OBJTYPE(NAMELIST) -
    GROUP('mqadmin') -
    AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(PROCESS) -
    GROUP('mqadmin') -
    AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(Queue) -
    GROUP('mqadmin') -
    AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(QMGR) -
    GROUP('mqadmin') -
    AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(RQMNAME) -
    GROUP('mqadmin') -
    AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(SERVICE) -
    GROUP('mqadmin') -
    AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(TOPIC) -
    GROUP('mqadmin') -
    AUTHADD(ALL)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(AUTHINFO) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(CHANNEL) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -

```

```

    OBJTYPE(CLNNTCONN) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(COMMINFO) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(LISTENER) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(NAMELIST) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(PROCESS) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(QUEUE) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(SERVICE) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(TOPIC) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
AMQ8862: IBM MQ Appliance authority record set.

END
21 MQSC commands read.
No commands have a syntax error.
All valid MQSC commands were processed.
M2000(mqc1i)#

```

Note: Example 8-1 on page 126 shows all of the object types that are supported by a version 8 queue manager. Some of these objects (such as the SERVICE object) might not be useful on an appliance.

With rights to the queue manager and objects set in OAM, users who are authorized can connect to the queue manager by using IBM MQ Explorer `runmqsc -c` or another IBM MQ client-based administration tool.

8.2.2 Connecting IBM MQ Explorer

The previous section showed how administration rights can be granted to a group of users. This section describes how one of those users can configure IBM MQ Explorer to connect to the queue manager on an IBM MQ Appliance.

A channel is needed to allow the client to connect to the queue manager. The channel uses Transport Layer Security (TLS) so that credentials passed are secured. The channel is set up with an invalid user that is defined in MCAUSER to ensure that only authenticated users can connect to the queue manager. The user ID is adopted from the authenticated credentials. This option is controlled at the queue manager level. For more information about the ADOPTCTX option, see 7.4.2, “Connecting the queue manager to the LDAP” on page 113 and Example 7-6, “Creating AUTHINFO record for CompanyB LDAP server” on page 116.

Example 8-2 shows creating the channel and a channel authentication rule so that the incoming user ID is used as the MCAUSER after the password is validated.

Example 8-2 Defining a channel for administration access

```
M2000(mqcli)# runmqsc -e GTWY1HAP
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Starting MQSC for queue manager GTWY1HAP.
DEFINE CHANNEL(MQEXPLORER.SVRCONN) -
  CHLTYPE(SVRCONN) -
  MCAUSER('*NOBODY') -
  SSLCAUTH(OPTIONAL) -
  SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) -
  REPLACE
16 : DEFINE CHANNEL(MQEXPLORER.SVRCONN) -
    : CHLTYPE(SVRCONN) -
    : MCAUSER('*NOBODY') -
    : SSLCAUTH(OPTIONAL) -
    : SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) -
    : REPLACE
AMQ8014: IBM MQ Appliance channel created.

SET CHLAUTH(MQEXPLORER.SVRCONN) -
  TYPE(ADDRESSMAP) -
  ADDRESS('*') -
  USERSRC(CHANNEL) -
  CHCKCLNT(REQUIRED) -
  ACTION(REPLACE)

17 : SET CHLAUTH(MQEXPLORER.SVRCONN) -
    : TYPE(ADDRESSMAP) -
    : ADDRESS('*') -
```

```

:    USERSRC(CHANNEL) -
:    CHCKCLNT(REQUIRED) -
:    ACTION(REPLACE)

```

AMQ8877: IBM MQ Appliance channel authentication record set.

The CLNTCONN end of this channel is defined in IBM MQ Explorer. It requires a JKS keystore to be created, and then the channel defined with a user ID and password.

Figure 8-4, Figure 8-5, Figure 8-6 on page 131, and Figure 8-7 on page 131 show the definition windows for the channel, but not creating the JKS keystore. For more information about creating the required JKS keystore, see this website:

http://www.ibm.com/support/knowledgecenter/#!/SSFKSJ_8.0.0/com.ibm.mq.adm.doc/q020430_.htm

Figure 8-4 General connection window for IBM MQ Explorer

Figure 8-5 User connection window for IBM MQ Explorer

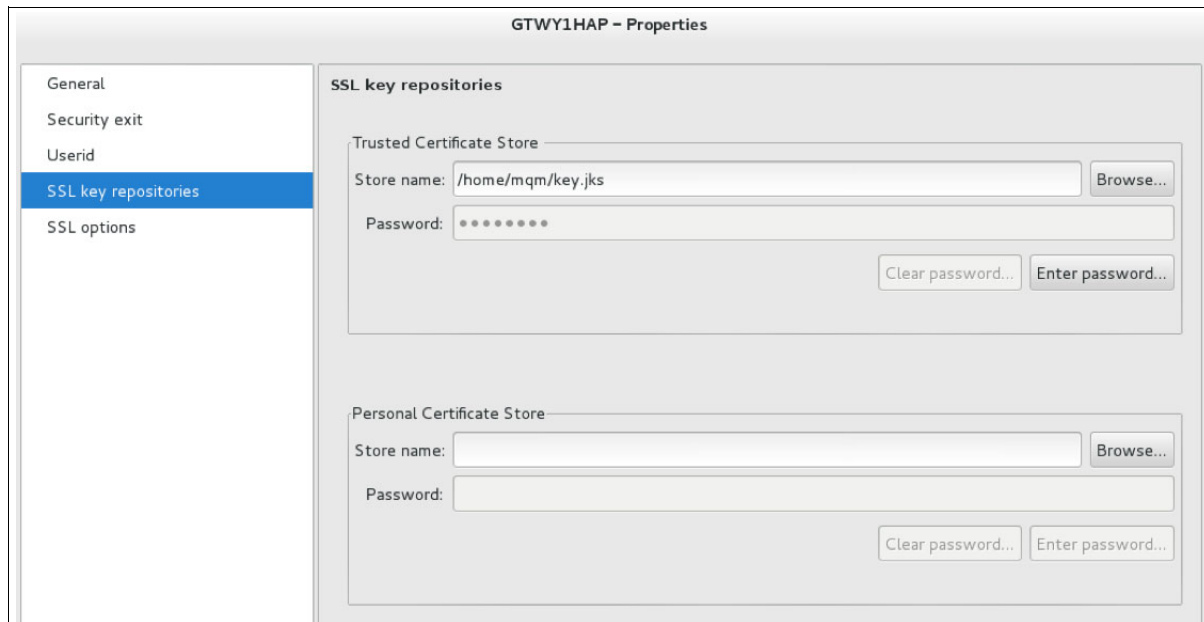


Figure 8-6 SSL key repositories window for IBM MQ Explorer

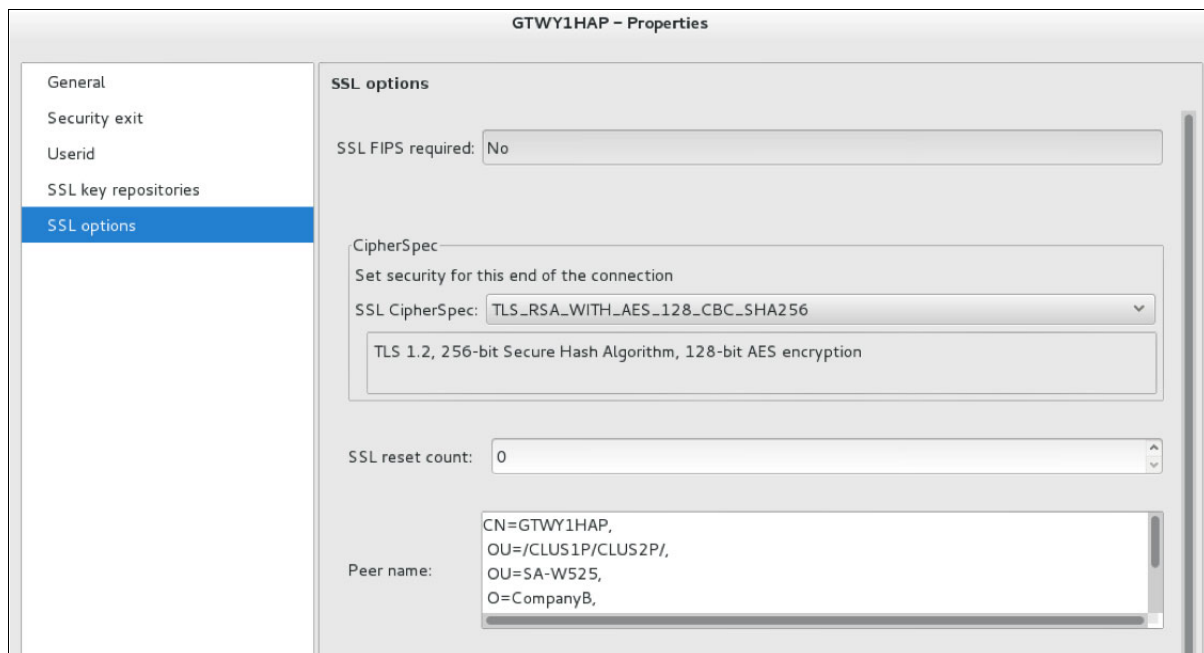


Figure 8-7 SSL options window for IBM MQ Explorer

8.2.3 Connecting runmqsc client

The **runmqsc** command that provides command-line administration of IBM MQ queue managers now can connect to queue managers by using an IBM MQ Client connection. This connection is useful in the context of an IBM MQ Appliance because **runmqsc** can be called from within a script on a support server and connect to the appliance queue manager to perform some work.

The runmqsc client can use the same channel as IBM MQ Explorer, although you might prefer to define a separate channel. You can also define a channel that uses TLS certificates for authentication instead of user ID and password.

Example 8-3 shows the creation of a CLNTCONN channel on a Linux server by using **runmqsc -n**. The connection to the queue manager is then made by using the **runmqsc -c** program and the client channel table, which is shown in Example 8-4.

Example 8-3 Defining a CLNTCONN channel by using runmqsc -n by using Linux

```
-bash-4.2$ runmqsc -n
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Starting local MQSC for 'AMQCLCHL.TAB'.

DEFINE CHL(MQEXPLORER.SVRCONN) -
  CHLTYPE(CLNTCONN) -
  CONNAME('10.10.0.30(1415)') -
  SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) -
  SSLPEER('CN=GTWY1HAP, OU=/CLUS1P/CLUS2P/, OU=SA-W525, O=CompanyB, L=Raleigh,
ST=NC, C=US') -
  QMNAME(GTWY1HAP) -
  REPLACE

1 : DEFINE CHL(MQEXPLORER.SVRCONN) -
  : CHLTYPE(CLNTCONN) -
  : CONNAME('10.10.0.30(1415)') -
  : SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) -
  : SSLPEER('CN=GTWY1HAP, OU=/CLUS1P/CLUS2P/, OU=SA-W525, O=CompanyB,
L=Raleigh, ST=NC, C=US') -
  : QMNAME(GTWY1HAP) -
  : REPLACE
AMQ8014: WebSphere MQ channel created.
```

Example 8-4 Connect to the queue manager by using runmqsc -c

```
-bash-4.2$ export MQCHLLIB=/var/mqm
-bash-4.2$ export MQCHLTAB=AMQCLCHL.TAB
-bash-4.2$ export MQSSLKEYR=/home/mqm/key
-bash-4.2$ runmqsc -c GTWY1HAP -u neil
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Enter password:
****
Starting MQSC for queue manager GTWY1HAP.

dis chl(MQEXPLORER.SVRCONN)
1 : dis chl(MQEXPLORER.SVRCONN)
AMQ8414: Display Channel details.
CHANNEL(MQEXPLORER.SVRCONN)          CHLTYPE(SVRCONN)
ALTDATE(2015-09-06)                  ALTTIME(18.05.41)
CERTLABL( )                          COMPHDR(NONE)
COMPMSG(NONE)                        DESCR( )
DISCINT(0)                           HBINT(300)
KAINT(AUTO)                          MAXINST(999999999)
MAXINSTC(999999999)                 MAXMSGL(4194304)
MCAUSER(*NOBODY)                    MONCHL(QMGR)
RCVDATA( )                          RCVEXIT( )
```


SCYDATA()	SCYEXIT()
SENDDATA()	SENDEXIT()
SHARECNV(1)	SSLCAUTH(OPTIONAL)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)	
SSLPEER()	TRPTYPE(TCP)

The example shows a channel table that was created in the default location. The MQCHLLIB environment variable can be set to force the table to be created in a different directory.

8.2.4 Enabling limited administration rights

In your environment, it might be appropriate to group users according to their wanted administration authority for one or more queue managers. These groupings can provide read-only access to configuration. They also might allow message browsing on all or some queues, or they might allow creation or changes to all or some objects.

From a queue manager administration perspective, permissions can be granted by using SET AUTHREC statements in **runmqsc** directly on the appliance in IBM MQ administration mode on the command line or by using a remotely attached **runmqsc** program. IBM MQ Explorer or other programs that create programmable command format (PCF) messages can also be used to set authorities.

Note: The **setmqaut** command, which can be used on other platforms to grant or remove authorities, is not available on the IBM MQ Appliance. Use SET AUTHREC in MQSC or a PCF-based program (such as IBM MQ Explorer) instead.

For more information about the configuration that is needed to enable limited administration by using IBM MQ Explorer, see Chapter 9 of the IBM Redbooks publication *Secure Messaging Scenarios with WebSphere MQ*, SG24-8069, which is available at this website:

<http://www.redbooks.ibm.com/abstracts/sg248069.html>

The requirements on an IBM MQ Appliance are no different, although there are some differences in implementation specifics. The main difference is that the user repository is local on the appliance or is an LDAP directory that is accessed by the queue manager. The appliance cannot make available an external user repository to an appliance queue manager as can be done on distributed systems.

There are significant enhancements in version 8 of IBM MQ that enable authentication by using a user ID and password instead of TLS certificates. These enhancements are included in the version of IBM MQ that runs on the IBM MQ Appliance. Authentication of administrative users with a user ID and password can eliminate the need to manage certificates for each user.

The examples in 8.2.3, “Connecting runmqsc client” on page 131 show all of the types of objects against which OAM security is active as of IBM MQ v8.0.0.3.

8.2.5 Granting security rights to users

As described in this book, all users of IBM MQ queue managers on the appliance are LDAP users. There are many ways to express an identity within an LDAP directory. When authenticating or creating OAM rules, the queue manager can use any of these options. In other places (especially when setting the MCAUSER) only the SHORTUSR can be used.

The absolute value of the user ID within the LDAP directory is the distinguished name, or DN. This value is the value that the queue manager stores in the OAM when rights are granted.

Note: The OAM stores the full DN of a user ID or group, not the short name. If a user ID or group is moved within the directory, its DN changes, even if the short name does not change. Although the permissions are still visible in the OAM, they do not work because the search for the short name returns the new DN. If a user ID or group is moved, the rights that are granted to it on the OAM must be granted again.

The following attributes in various directory implementations can contain short user names:

- ▶ uid
- ▶ sAMAccountName
- ▶ Common Name (CN)
- ▶ shortName

The specific name fields in your company's directory must be configured in the AUTHINFO records for your queue managers. At the time of this writing, our directory used the field "uid" to store the short user name (SHORTUSR). The configuration of the queue manager to use this short user name field is described in Chapter 7, "Creating queue managers" on page 107. We did not specify a different USRFIELD. For more information about SHORTUSR and USERFIELD, see this website:

http://www.ibm.com/support/knowledgecenter/#!/SSFKSJ_8.0.0/com.ibm.mq.sec.doc/q123460_.htm

Example 8-5 shows granting GET, BROWSE, and INQUIRE rights to queue GL.UPDATE.REQUEST to user neil. The short user name is used in the SET AUTHREC statement. This directory field is specified in the SHORTUSR attribute of the AUTHINFO record. However, the display of the resultant record shows the full DN for Neil's user ID, which is uid=neil,ou=users,dc=companyb,dc=local.

Example 8-5 Grant and display rights on a queue

```
M2000(mqcli)# runmqsc NEILHA
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Starting MQSC for queue manager NEILHA.

SET AUTHREC -
  PROFILE(GL.UPDATE.REQUEST) -
  OBJTYPE(Queue) -
  PRINCIPAL('neil') -
  AUTHRMV(ALL) -
  AUTHADD(BROWSE,GET,INQ)
  1 : SET AUTHREC -
    : PROFILE(GL.UPDATE.REQUEST) -
    : OBJTYPE(Queue) -
    : PRINCIPAL('neil') -
    : AUTHRMV(ALL) -
    : AUTHADD(BROWSE,GET,INQ)
AMQ8862: IBM MQ Appliance authority record set.
DISPLAY AUTHREC -
  PROFILE(GL.UPDATE.REQUEST) -
  MATCH(EXACT)
  2 : DISPLAY AUTHREC -
    : PROFILE(GL.UPDATE.REQUEST) -
    : MATCH(EXACT)
```

```
AMQ8864: Display authority record details.  
  PROFILE(GL.UPDATE.REQUEST)  
  ENTITY(uid=neil,ou=users,dc=companyb,dc=local)  
  ENTTYPE(PRINCIPAL)                      OBJTYPE(QUEUE)  
  AUTHLIST(BROWSE,GET,INQ)  
END  
  3 : END  
2 MQSC commands read.  
No commands have a syntax error.  
All valid MQSC commands were processed.  
M2000(mqc1i)#
```

Note: In Example 8-5 on page 134, AUTHRMV(ALL) is specified before AUTHADD, which ensures that the exact rights that are granted are the result of the statement. Without the AUTHRMV option, the three new rights are added to whatever rights user ID neil already had over the queue. If this result is the intended result, include the rights in the statement or do not include AUTHRMV(ALL). Although it appears to be simpler to ignore the AUTHRMV and add the new rights, doing so can make longer term management and auditing of the OAM rules more difficult.

The name of the principal that is shown in Example 8-5 on page 134 was a short user name (SHORTUSR). In the SET AUTHREC statement, a full distinguished name or a value that matches USRFIELD can be used instead.

In other contexts, such as Message Channel Agent user IDs (MCAUSERS), only the short name can be used.

Tip: It is good practice to use short names in SET AUTHREC and then verify that the resolved DN is correct. This practice enables you to verify that the correct account is resolved when the short name is used in an MCAUSER field.

For more information about assigning user IDs to channels, see Chapter 9, “IBM MQ channel security” on page 137.

8.3 Summary

This chapter described where an administrator can expect to see differences in managing security for queue managers on the IBM MQ Appliance rather than on any other distributed platform.

Most aspects of security of the queue manager and how they are managed by OAM were not covered because the focus of this book is not IBM MQ security. Apart from the differences that are highlighted in this chapter, security for queue managers on the appliance does not differ from security for other queue managers. If you choose to implement connection authentication by using LDAP, and authorization method that also points to LDAP is almost the same as a distributed queue manager that uses the same options.

The important difference is that there is no ability for the appliance firmware to provide an enterprise view of user identity. On a distributed system, whether it is Windows, UNIX, or something else, the operating system can use an enterprise directory for user identity and an IBM MQ queue manager can use that user identity.

To achieve something similar on the IBM MQ Appliance, the queue manager must interact with the LDAP service and only LDAP can be used for this function.

For more information about creating and administering local messaging users, see this website:

http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/administering/ad00080_.htm

Although it is feasible to implement queue managers on the IBM MQ Appliance by using the user repository that is provided by the appliance, we advise against this practice for the following reasons:

- ▶ It makes implementing and administering HA queue managers more difficult because all users and groups must be created and managed in multiple places.
- ▶ Strong security management requires identity management to be separated from other administration roles.

By using a corporate directory as the queue manager identity source, issues concerning password reset, provisioning, removal, and validation can be handled by the normal security team without any special effort.



IBM MQ channel security

Most aspects of channel security on the IBM MQ Appliance are the same as security that is implemented on any queue manager that is running at a similar software level.

The IBM MQ Appliance that was used during the writing of this book reports software version 8.0.0.3 and command level 802. Queue managers on the appliance always set and check authorizations for principals and groups, like non-appliance hosted queue managers on Windows, or queue managers on other distributed platforms with SecurityPolicy=user set in the qm.ini file.

This chapter describes the differences between channel security on an IBM MQ Appliance and channel security on a distributed queue manager. Suggestions are provided for how to effectively manage the differences to implement the IBM MQ Appliance into your organization.

This chapter includes the following topics:

- ▶ 9.1, “Security policy” on page 138
- ▶ 9.2, “SSL and TLS” on page 138
- ▶ 9.3, “Replacing exits” on page 138
- ▶ 9.4, “User Identity changes” on page 140
- ▶ 9.5, “Purpose of MCAUSERS” on page 141
- ▶ 9.6, “Authenticating channel partners” on page 141
- ▶ 9.7, “Assigning MCAUSERS on the IBM MQ Appliance” on page 142
- ▶ 9.8, “Summary” on page 148

9.1 Security policy

Implementing security in a computer system is enforcement of a security policy. Security policies are set by the organizations that run computer systems and can be set to meet regulatory or other external requirements and the needs of the organization.

Most such policies are phrased in fairly generic ways to enable them to be enforced by using whatever capability is available on the target system.

Example 9-1 shows a possible policy statement that requires traffic to be encrypted but does not prescribe the specific algorithms that must be used.

Example 9-1 Security policy statement

All data exchanged between applications and message queue providers must be encrypted using strong cipher suites. Cipher suite choice must be reviewed at 1 year intervals to ensure that the chosen cipher suite is still suitable.

Enforcement of some policies might be implemented on non-appliance based queue managers by using an exit, such as a security exit. A channel auto-definition exit might be used to ensure that the security exit was correctly attached to a cluster channel. This approach to policy enforcement by using channel exits is not possible when an IBM MQ Appliance is used.

When an appliance is integrated into an IBM MQ environment, the limitation regarding exits can extend to other queue managers, even if they are not running on an appliance.

9.2 SSL and TLS

Secure Sockets Layer (SSL) and Transport Layer Security (TLS) are terms that are often used interchangeably. SSL is an older standard that was superseded by TLS. SSL is now deprecated as all forms of it include known security weaknesses. TLS is used in preference to SSL in all cases.

To avoid problems with compatibility with an earlier version, many IBM MQ options, such as SSLCIPH, or SSLPEER, still refer to SSL, even when they are used to configure TLS connections.

In this book, we refer to TLS rather than SSL in all cases, except those cases where the IBM MQ commands or options explicitly use the term SSL.

SSL and TLS rely on X.509 certificates that are signed by trusted authorities. These certificates assert the identity of the holder based on the distinguished name that is defined in the certificate. When we refer to certificates in this book, we are referring to X.509 certificates.

9.3 Replacing exits

The most obvious difference between a regular queue manager and a queue manager that is hosted on the IBM MQ Appliance is that the appliance does not allow any exits to run. This restriction is absolute and includes the following types of channel-related exits:

- Security

- ▶ Message
- ▶ Send
- ▶ Receive
- ▶ Channel auto-definition

9.3.1 Security exits

Most functions of security exits can now be implemented by using channel authentication (CHLAUTH) rules. CHLAUTH rules are defined by using the IBM MQ Console, **runmqsc** and **SET CHLAUTH** commands, or the IBM MQ Explorer. CHLAUTH rules were introduced in version 7.1, and are now an established part of IBM MQ security control.

Where an exit was implemented to enforce a site security policy and that policy cannot be met by using CHLAUTH rules, a policy change or exemption might be required to integrate an IBM MQ Appliance into your infrastructure.

For example, suppose that a security exit was developed to exchange a shared secret between the sender and receiver channel instances during start. This exit might be an acceptable authentication mechanism to meet a site policy. There is no equivalent function available by using CHLAUTH, although stronger authentication of the channel partner can be achieved by using TLS, with or without CHLAUTH rules. To implement an interoperating queue manager on an IBM MQ Appliance, the exit-based authentication method is removed and replaced with a method that operates correctly without exits.

9.3.2 Channel auto-definition exits

On other versions of IBM MQ, channel auto-definition exits can be started in the following places:

- ▶ Dynamic definition of cluster sender channels
- ▶ Dynamic creation of server connection (svrconn) channels

These options are not available on an IBM MQ Appliance queue manager. For the situation where svrconn channels were dynamically used, static definitions of the svrconn channels are required.

The more complex issue is the cluster channel auto-definition exit, which can be used to customize exit definitions that appear on dynamically created cluster sender channels. Exits that are defined on cluster receiver channels propagate to dynamic cluster sender channels. The cluster channel auto-definition exit can remove the propagated exit from the cluster sender if it is not needed or customize the exit name and path so that it can function correctly on the platform that is running the exit.

The lack of channel auto-definition exits means that all exits (message, send, receive, or security) must also be removed from cluster receiver channels that interoperate with an IBM MQ Appliance.

Another function of a cluster channel auto-definition exit might be to customize the value for LOCLADDR on the dynamic cluster sender channel. This field defines the source address that is used for the TCP socket supporting the channel.

There is an environment variable (currently undocumented) that can be used to address this need, at least in part. Set the MQ_LCLADDR (note that the name contains LCL, not LOCL) environment variable. An example of the command is shown in Example 9-2.

Example 9-2 Set MQ_LCLADDR environment variable

```
setmqvar -m GTWY1HAP -k MQ_LCLADDR -v 10.101.0.30
```

Any sender channel without the LOCLADDR attribute uses the supplied IP address as the source instead of binding to the address with the best route as defined by the routing table. This configuration allows all cluster sender channels in a queue manager to use the same, non-default address but does not allow for different addresses to be used by different cluster sender channels.

9.3.3 Other exits

Some systems might use message or send and receive exits for functions, such as capturing message text for audit purposes.

Because exits cannot be run on the IBM MQ Appliance, they cannot be used on appliance-hosted queue managers. They also cannot be defined on cluster receiver channels that interoperate with an appliance-hosted queue manager because the dynamic definition of the cluster sender channel includes the exit, which prevents the cluster sender channel from running.

It might be possible to use application activity trace instead of an exit if the objective is to capture this type of data from the channel. For more information about application activity trace, see the following resources:

- ▶ IBM MQ Appliance Knowledge Center:

https://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/monitoring/mo00020_.htm

- ▶ IBM MQ Knowledge Center:

http://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.mon.doc/q037520_.htm

IBM MQ Support Pack mh06 can assist in collecting and interpreting the collected trace information. For more information, see this website:

<http://www.ibm.com/support/docview.wss?uid=swg24036430>

9.4 User Identity changes

As described in Chapter 7, “Creating queue managers” on page 107, queue managers that are running on an appliance should use an external Lightweight Directory Access Protocol (LDAP) based identity source. LDAP is used so that organizational security policy enforcement does not fall to IBM MQ Administrators.

For example, if appliance-based queue managers use internal users, the IBM MQ Administrator must be involved in the process to remove accounts when a person leaves an organization. The IBM MQ Administrator also is required to perform password reset and group membership type functions, which is problematic for many organizations with strong security policies in place. Security policy enforcement is enhanced by limiting the use of appliance accounts as much as possible.

Because all queue manager user identity is in LDAP rather than the appliance, channel Message Channel Agent user IDs (MCAUSERS) must also be LDAP-hosted accounts. This requirement is true whether the user ID is assigned to MCAUSER in the channel definition, mapped by using CHLAUTH, or adopted from a user ID and password authenticated account on a server connection channel.

9.5 Purpose of MCAUSERS

Queue managers can control the level of access that is given to different users that connect. This control is managed by a component that is called the Object Authority Manager (OAM).

The basis for granting or denying access is the identity of the user. In this context, the user might not be a person. It might be a program that is running on a server that connected to the queue manager.

Because it is difficult to have a single view of identities across all queue managers, IBM MQ does not generally flow the user context from one queue manager to another. Instead, each queue manager establishes a local security context, including the user ID.

The value of the user ID in the local security context is set by using the MCAUSER channel attribute.

9.6 Authenticating channel partners

Running encrypted channels and authenticating channel partners by using certificates is essentially the same for standard queue managers and queue managers that are hosted on an IBM MQ Appliance. The configuration of CMS keystores is different when performed on an appliance rather than for a distributed queue manager. This difference is described in Appendix A, “IBM MQ cryptographic changes” on page 185.

9.6.1 Message channels

Message channels are used to communicate between queue managers. They are called message channels because messages flow over them directly from the source queue manager to the target queue manager. They are also called queue manager-to-queue manager channels.

TLS is the only strong method available to authenticate queue manager-to-queue manager channels when an IBM MQ Appliance is involved. If a TLS client certificate is presented, the CHLAUTH rule can strongly authenticate based on possession of the certificate and key.

IBM MQ queue managers perform two functions when authenticating a TLS-based connection. The TLS component verifies that the presented certificate is valid; that is, the certificate was signed by a trusted authority, reached its validity date, and has not expire. The channel or a CHLAUTH rule checks the Distinguished Name (DN) of the certificate against expected values, which are found in SSLPEER attributes.

The TLS client end of the TLS connection can be required to present a certificate by defining channels with the SSLCAUTH(REQUIRED) attribute.

Note: When connecting to a queue manager that always runs on the same IBM MQ Appliance, the address localhost (127.0.0.1) can be used. This scheme reduces the need for TLS and strong authentication. However, this scheme does not scale well into a high availability (HA) environment because there is no way to ensure that the two queue managers always run on the same appliance.

A TLS client is a program that starts a TLS connection. In the context of an IBM MQ queue manager, this side is the side that starts the channel. This side is generally the Sender side of a Sender/Receiver pair. It also can be a requester in a Requester/Server pair. There are also other possibilities, but the rule always remains: the TLS client starts the connection.

If a queue manager-to-queue manager channel is set up without requiring the TLS client to present a certificate, CHLAUTH rules can filter on the IP address or the queue manager name only. Queue manager or IP address-based filters cannot be considered effective authentication mechanisms. An IP address can be spoofed, which allows an attacker access that must be denied. A queue manager name can be adopted even more easily by any new queue manager that is defined on any machine with network access. This configuration allows a connection to occur, which must be rejected.

If the site security policy requires that connections between queue managers are authenticated to ensure that only valid connections are allowed, a TLS key pair and certificate must be created for each queue manager and SSLCAUTH(REQUIRED) must be implemented on all message channels.

9.6.2 MQI channels

Message Queue Interface (MQI) channels are commonly referred to as client connection channels (defined as server connection channels at the queue manager). MQI channels are referred to in this way because MQI calls flow across them, rather than messages.

MQI channels in the IBM MQ Appliance environment provide more authentication options as compared to message channels. As with message channels, TLS can be used with the client application presenting its certificate to the queue manager for validation and verification.

MQI channels also support the application passing a user ID and password in the API flow. The queue manager can validate these against a local user registry or against an LDAP directory. This choice is made at the queue manager level by creating an Authentication Information (AUTHINFO) record of type IDPWOS or IDPWLDAP. The record is then named in the Connection Authentication (CONNAUTH) field of the queue manager by using IBM MQ Explorer or the `runmqsc` and `ALTER QMGR` commands. For more information, see Chapter 7, “Creating queue managers” on page 107.

We recommend the use of LDAP to host all user and group information that is used by queue managers. Therefore, IBM MQ Administrators are not responsible for managing user accounts according to company policies. They also do not implement policy enforcement controls in parallel with those controls that are already in place in a corporate directory.

9.7 Assigning MCAUSERs on the IBM MQ Appliance

After authentication of the channel partner is complete, a user ID must be associated with the connection. On an appliance, this process works in the same way as on normal queue managers, with the restriction that an exit cannot be used to assign the user ID.

There are several options that can be used to assign an MCAUSER to a channel, as described in the following sections.

9.7.1 MCAUSER field on the channel definition

The simplest way to assign the user ID that is used for subsequent authorization activities is to assign it in the MCAUSER attribute of the channel, as shown in Example 9-3. It shows a receiver channel on queue manager GTWY2HAP, which receives messages from queue manager GTWY1HAP.

Example 9-3 Define a channel with MCAUSER

```
runmqsc GTWY1HAP
  DEFINE CHANNEL(GTWY1HAP.GTWY2HAP.01) -
    CHLTYPE(RECEIVER) -
    SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) -
    SSLCAUTH(REQUIRED) -
    SSLPEER('CN=GTWY1HAP,OU=SA-W525,O=CompanyB,ST=NC,C=US') -
    MCAUSER('mq_GTWY1HAP')
```

9.7.2 MCAUSER attribute of a channel authentication record

There are several formats of channel authentication records (CHLAUTH) that can assign an MCAUSER. The CHLAUTH record matches some attribute of the incoming request and the user ID is assigned based on the match. The matching attribute is selected based on the TYPE attribute of the CHLAUTH record.

Using CHLAUTH TYPE(SSLPEERMAP)

Example 9-4 shows a validation that is based on the incoming X.509 certificate, which is the strongest assertion available.

Example 9-4 Assign MCAUSER based on certificate DN

```
runmqsc GTWY1HAP
  SET CHLAUTH(GTWY1HAP.GTWY2HAP.01) -
    TYPE(SSLPEERMAP) -
    SSLPEER('CN=GTWY1HAP,OU=*,OU=SA-W525,O=CompanyB,ST=NC,C=US') -
    USERSRC(MAP) -
    MCAUSER('mq_GTWY1HAP')
```

Assigning the MCAUSER with a CHLAUTH record is commonly combined with defining an invalid MCAUSER on the channel. In the examples that are shown in this section, the invalid user is “NOACCESS”. This user never matches a valid user ID and causes all requests to fail unless a CHLAUTH rule matches the incoming request and replaces the invalid MCAUSER with a valid user.

When CHLAUTH rules are combined with a cluster receiver channel, each queue manager in the cluster that connects can be assigned its own dedicated user ID. Cluster members that are not permitted to communication with this queue manager fail to connect because the invalid user ID that is assigned by the channel definition blocks the connection, as shown in Example 9-5 on page 144, where GTWY2HAP defines a cluster receiver channel and CHLAUTH rules to allow GTWY1HAP and QM3 to connect. QM4 is blocked from connecting because it does not have a certificate that matches any rule. The example also shows a generic SSLPEER matching the cluster name in the OU of the incoming certificate.

Example 9-5 Assign invalid MCAUSER in cluster receiver channel. Fix MCAUSER in CHLAUTH

```
runmqsc GWTY1HAP
  DEFINE CHL(TO.GTWY2HAP.CLUSTER1P) -
    CHLTYPE(CLUSRCVR) -
    MCAUSER('*NOACCESS') -
    CONNAME('mqa001p-mq.companyb.local(1414)') -
    SSLPEER('*') -
    CLUSTER(CLUSTER1P)

  SET CHLAUTH(TO.GTWY2HAP.CLUSTER1P) -
    TYPE(SSLPEERMAP) -
    SSLPEER('CN=GTWY1HAP,OU=*/CLUSTER1P/*,OU=SA-W525,O=CompanyB,ST=NC,C=US') -
    USERSRC(MAP) -
    MCAUSER('mq_GTWY1HAP')

  SET CHLAUTH(TO.GTWY2HAP.CLUSTER1P) -
    TYPE(SSLPEERMAP) -
    SSLPEER('CN=QM3,OU=*/CLUSTER1P/*,OU=SA-W525,O=CompanyB,ST=NC,C=US') -
    USERSRC(MAP) -
    MCAUSER('mq_qm3')
```

Use of CHLAUTH TYPE(QMGRMAP)

By using the TYPE(QMGRMAP) attribute in a CHLAUTH record, the queue manager name that is asserted by the partner queue manager can be checked and mapped onto an assigned MCAUSER value. This check is simpler to set up than an SSLPEERMAP, but provides a weaker assertion of the identity of the partner's identity.

The TYPE(QMGRMAP) CHLAUTH record can be combined with a generic SSLPEER in the channel definition to ensure that only a queue manager with access to an acceptable certificate can impersonate another queue manager, as shown in Example 9-6. The invalid MCAUSER '*NOACCESS' is again used to force all connection attempts to fail unless a CHLAUTH rule matches the request and asserts a valid MCAUSER.

Example 9-6 Use of SSLPEER on a channel and QMGRMAP on the CHLAUTH record

```
runmqsc GWTY1HAP
  DEFINE CHL(TO.GTWY2HAP.CLUSTER1P) -
    CHLTYPE(CLUSRCVR) -
    MCAUSER('*NOACCESS') -
    CONNAME('mqa001p-mq.companyb.local(1414)') -
    SSLPEER('CN=*,OU=*/CLUSTER1P/*,OU=SA-W525,O=CompanyB,ST=NC,C=US') -
    CLUSTER(CLUSTER1P)

  SET CHLAUTH(TO.GTWY2HAP.CLUSTER1P) -
    TYPE(QMGRMAP) -
    QMNAME(GTWY1HAP) -
    USERSRC(MAP) -
    MCAUSER('mq_GTWY1HAP')
```

9.7.3 MCAUSER assigned by using ADOPTCTX

Client applications, including administration applications (such as IBM MQ Explorer) connect to a queue manager by using a Server Connection (SVRCONN) channel. The application can authenticate to the queue manager in two ways, as described in 9.6.2, “MQI channels” on page 142.

A queue manager setting (CONNAUTH) specifies an Authentication Information (AUTHINFO) record that controls some aspects of queue manager security. One option in the AUTHINFO record is to adopt the context of the user ID, which is authenticated by using a password. The user ID and password flow across the channel during setup before other API calls are allowed.

An example that shows the creation of the AUTHINFO record and assigning it to the queue manager is shown in Example 9-7.

Example 9-7 Creating AUTHINFO to activate adopt context

```
runmqsc GWTY1HAP
  DEFINE AUTHINFO(COMPANYB.LDAP) -
    AUTHTYPE(IDPWLDAP) -
    ADOPTCTX(YES) -
    DESCR('Authenticate to LDAP, and adopt context of the user ID') -
    CONNAME('ldap1.companyb.local(636),ldap2.companyb.local(636)') -
    CHCKCLNT(REQUIRED) -
    CHCKLOCL(OPTIONAL) -
    CLASSGRP('posixGroup') -
    CLASSUSR('posixAccount') -
    FAILDLAY(1) -
    FINDGRP('memberUid') -
    BASEDNG('OU=groups,DC=companyb,DC=local') -
    BASEDNU('OU=users,DC=companyb,dc=local') -
    LDAPUSER('UID=mqbind,OU=users,DC=companyb,DC=local') -
    LDAPPWD('mqbind') -
    SHORTUSR('uid') -
    GRPFIELD('cn') -
    USRFIELD('uid') -
    AUTHORMD(SEARCHGRP) -
    NESTGRP(NO) -
    SECCOMM(ANON) -
    REPLACE

  ALTER QMGR CONNAUTH(COMPANYB.LDAP)

  REFRESH SECURITY TYPE(CONNAUTH)
```

For more information about the definition of AUTHINFO objects, see this website:

http://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.ref.adm.doc/q085490_.htm

Authentication of incoming connections can be forced by using the CHCKCLNT(REQUIRED) attribute, as shown in Example 9-7. This authentication process forces all client connections to the queue manager to authenticate with user ID and password, even if they also present an X.509 certificate.

An alternative is to set CHCKCLNT(OPTIONAL) in the AUTHINFO object, which is used by the queue manager, and enforce user ID and password checking by using CHLAUTH rules that match the channel names. MQI Channels where the user ID and password are used for authentication have a CHLAUTH rule with CHCKCLNT(REQUIRED). Other channels have SSLCAUTH(REQUIRED) to ensure that clients present a certificate, or might supply a low capability MCAUSER on the channel definition.

To enforce user ID and password-based authentication on a client connection, create a SVRCONN channel that specifies that the client does not have to present a certificate. This specification means that SSLCAUTH(OPTIONAL) is set. Then, create a Channel Authentication (CHLAUTH) rule that forces the application to authenticate with user ID and password. To achieve this configuration, the CHLAUTH rule includes CHCKCLNT(REQUIRED). An example of a SVRCONN channel and CHLAUTH rule to implement this scheme is shown in Example 9-8.

Example 9-8 Define a channel to force user ID and password-based authentication

```
runmqsc GTWY1HAP
  DEFINE CHL(AUTH.CLIENT.CHL) -
    CHLTYPE(SVRCONN) -
    TRPTYPE(TCP) -
    DESCR('Allow clients to connect without certificates') -
    MCAUSER('*NOACCESS') -
    SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) -
    SSLCAUTH(OPTIONAL) -
    REPLACE

  SET CHLAUTH(AUTH.CLIENT.CHL) -
    TYPE(ADDRESSMAP) -
    ADDRESS('*') -
    DESCR('Force clients to connect with user ID and password') -
    CHCKCLNT(REQUIRED) -
    USERSRC(CHANNEL) -
    ACTION(ADD)
```

9.7.4 Granting rights to channel MCA users

The rights that are granted to an MCAUSER on an MQI channel are the same as those rights for an application that is bound directly to the queue manager. The mechanism for obtaining the user ID that is checked is different, but the actual rights that are needed are the same.

The rights that are granted to an MCAUSER on a receiver channel are somewhat different from those rights that are granted to applications that use MQI channels, but they must still be controlled tightly. In particular, message channels should not be allowed to put messages on system queues.

This rule has the following likely exceptions:

- ▶ Cluster channel MCAUSERS at a full repository must be permitted to PUT to the SYSTEM.CLUSTER.COMMAND.QUEUE.
- ▶ Cluster channel MCAUSERS at a partial repository that receive messages from a full repository must be allowed to PUT to the SYSTEM.CLUSTER COMMAND.QUEUE. This exception means that Full Repository queue managers in a cluster are assigned a different MCAUSER than normal queue managers.

- If `SYSTEM.DEAD.LETTER.QUEUE` is used as the DLQ for the queue manager, all channels must have access to it.

In addition, message channel MCAUSERS need authorization to set all context for each queue to which they have access. This authorization is granted with the `SETALL` authority.

None of these exceptions are any different than the requirements for IBM MQ that is running in a distributed environment.

9.7.5 Creating and managing MCA users

In this book, we described the use of LDAP as the identity source for queue managers that are running on an IBM MQ Appliance and why this configuration makes sense in the context of the appliance.

Therefore, all user IDs that are seen by the queue manager must be defined in LDAP.

Current IBM MQ administrators might be familiar with getting channel MCAUSERS defined in the local Operating System (OS), while administration user IDs might be enterprise users that are held in a directory that the OS can also use. This configuration is not possible on the IBM MQ Appliance. All users must either be defined as messaging users on the appliance, or defined in the LDAP that is identified to the queue manager. This distinction is specified in the queue manager connection authentication setting by using `runmqsc` command **ALTER QMGR CONNAUTH** to point to an `AUTHINFO` object with `TYPE(IDPWOS)` for local account management or `TYPE(IDPWLDAP)` for LDAP-based accounts. When LDAP is used, the MCAUSER values (which are limited to 12 characters) are obtained from the field named in the `SHORTUSR` attribute of the `AUTHINFO` object.

For the appliance, work with your enterprise security team to define a mechanism for MCAUSERS to be created and assigned to groups. These user IDs might need to be managed differently to normal user IDs because they do not represent an actual person. In this respect, they are more like system or service accounts than personal accounts.

It is likely that message channel MCAUSER IDs can be placed in a different part of the directory to that used for normal user accounts.

In the Company B directory that is used to demonstrate configurations for this book, regular users have a DN suffix of `ou=users,dc=companyb,dc=local`.

The queue manager can expand the user ID (neil) to the full DN because it can find `uid=neil` by doing a subtree search down from the `BASEDNU` point of the directory tree, which is `dc=companyb,dc=local`. The search returns the correct record, and the queue manager discovers the DN `'uid=neil, ou=users,dc=company,dc=local'`.

However, Company B stores service accounts in a different OU, `ou=serviceaccounts,dc=companyb,dc=local`. The `BASEDNU` is high enough in the LDAP tree so that regular users and service accounts are within the searched subtree.

9.8 Summary

This chapter provided an overview of the difference that an administrator encounters when implementing channel security on an IBM MQ Appliance rather than on a regular distributed queue manager.

The examples are intended to show how the standard IBM MQ features can be used to avoid the need for the use of exits. In the past, exits were used to extract and validate identity and assign MCAUSERS to channels.

The examples do not show all of the possible options of queue manager configuration, authentication information objects, channel definitions, and channel authentication records. Instead, they show how these items can be used to enforce strong authentication by using TLS or user ID and password.

The examples then show how the authenticated user is mapped to an MCAUSER, which is the basis for granting access to resources that are managed by the queue manager.



High availability

There often is a misunderstanding about high availability (HA). This subject often is confused with Disaster Recovery (DR). Before describing HA concerning the IBM MQ Appliance, we describe the differences between HA and DR.

This chapter includes the following topics:

- ▶ 10.1, “Disaster recovery and high availability” on page 150
- ▶ 10.2, “Historical IBM MQ HA” on page 150
- ▶ 10.3, “HA groups” on page 151
- ▶ 10.4, “Preparing for HA” on page 153
- ▶ 10.5, “Creating an HA group” on page 153
- ▶ 10.6, “Creating an HA queue manager” on page 154
- ▶ 10.7, “Setting the preferred appliance for a queue manager” on page 156
- ▶ 10.8, “Suspending and resuming an appliance” on page 156
- ▶ 10.9, “Applying a fix pack to an HA group” on page 160
- ▶ 10.10, “HA scenario” on page 160

10.1 Disaster recovery and high availability

DR is the process and procedures that are used to restore business continuity in the time soon after an event that caused significant disruption to normal service. The event can be the result of natural causes or as a direct result of some human intervention. The aim of DR is to restore services that are needed for continued business activity. It can be assumed that within a disaster situation, there can be some limited loss of data and a time for which services can be unavailable. Perhaps unexpectedly, this time can be several hours and not only minutes.

Typically, within the DR pattern, there is a plan for the business IT infrastructure. This plan might consist of an ordered list of infrastructure resources to be brought online for the most important services to provide the minimal system. This plan is then built upon, bringing up secondary, tertiary services, and so on, until a complete IT environment is available that functions as a whole to provide all services that a business needs.

Tip: DR is often performed at sites that are remote from the point the original disaster that caused and plans and policies to be put into action.

The focus on HA often is single components within the whole system. However the aim is opposite to DR. For example, HA designs usually want to provide a rapid takeover of a failed system (including the data) by another equivalent system. The data that is on a disk might be owned temporarily by one server (which exhibits a failure) and later taken over by another server that also is attached to the same disk where the business data is stored.

Comparatively DR designs tend to restore functions of an entire data center and might not have to provide immediate recovery, or recovery of all data to the point of the failure.

10.2 Historical IBM MQ HA

Traditional IBM MQ has over the years introduced features related to HA concepts. The IBM MQ Appliance brings with it a new feature in the form of the HA Group. The following types of features were available in previous versions of the IBM MQ product:

- ▶ Queue Manager Clusters

These clusters make it possible to distribute queues of the same name across multiple queue managers. Applications can connect to a specific queue manager and send messages to any instances of a queue. A built-in workload management routine distributes messages to these instances by default, taking into account the volume of messages that flowed across the channels to those queue managers. This feature enables the detection of unreachable destinations and the routing of messages away from a failed queue manager. By using clustering capabilities, end points for the messages do not need to be a single point of failure.

- ▶ Multi-Instance Queue Managers

These queue managers are one of several IBM MQ technologies that provides the automatic switching of queue managers between servers. A queue manager is started normally or in a standby mode where it continually attempts to acquire the queue manager resource and becomes the active queue manager if possible. Therefore, this instance is the first instance of the queue manager to capture the resources or the owning queue manager or server had a failure and released the resources for capture by an alternative instance.

- ▶ HA Clusters

HA clusters are another automatic switching capability. Clustering technology, such as Microsoft Cluster Server (MSCS), or High Availability Cluster Multi-Processing (HACMP™) for AIX®, manage resources and trigger fail over of the resources onto secondary hardware. For example, with MSCS, the IBM MQ implementation provides the callback routines for the MSCS application programming interface. These callback routines perform queue manager *health* validation and status verification that all is working as it should be with the queue managers under the control of MSCS. If the response from the callback reports a problem, it is the MSCS software that performs the failover of the dependent resources, such as disks and IP addresses.

- ▶ Client Reconnection

IBM MQ client applications can automatically reconnect to the same or another (ideally equivalent) queue manager to continue processing if there is a failure. The possible destinations are encoded into the client connection channel and can work with multi-instance queue managers to reconnect to any now active queue manager.

All HA configuration on the appliance occurs in `mqcli` command-line interface (CLI) mode. In this chapter, every command that is referenced assumes that you are in `mqcli` CLI mode.

10.3 HA groups

An HA group defines an association between appliances that act together so that they are highly available. That is, the appliances in the group monitor each other for failures and if necessary take over the responsibility for running the queue managers that are defined to be highly available. They also make their resources available to client applications.

The two appliances are connected on the following available interfaces:

- ▶ Replication (eth21)
- ▶ Primary interface: eth13
- ▶ Secondary interface: eth17

An HA group directly connects two appliances for HA. An HA group is a configuration of IBM MQ Appliances that monitor each other and any HA queue managers. The aim of the HA group is to ensure that each HA queue manager can run on one appliance but also can fail over to the other, if necessary.

An appliance can be in only one HA group at a time. HA queue managers are owned by the HA group, not by a specific appliance in the group, as shown in Figure 10-1 on page 152.

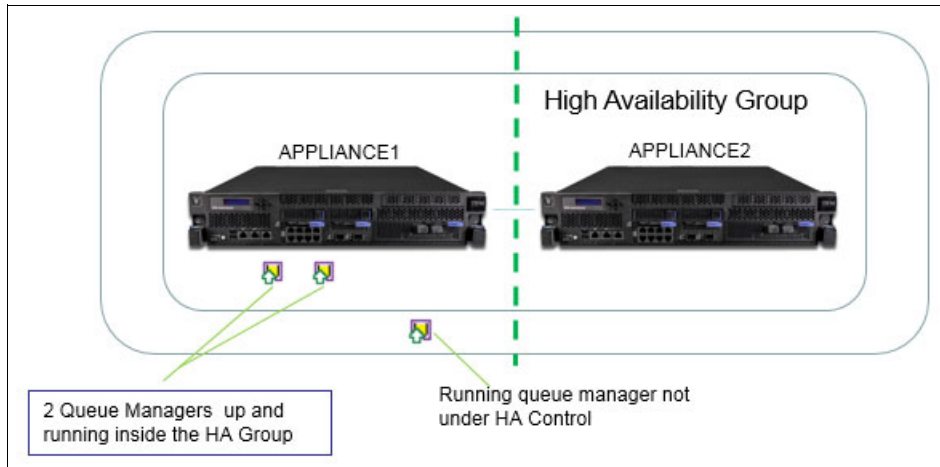


Figure 10-1 Two appliances in an HA group configuration

Note: Appliances in HA groups can also support queue managers that are not in the HA group.

It is up to the IBM MQ administrator to select which appliance an HA queue manager “prefers” to run on. Under normal circumstances, the HA queue manager runs on the preferred appliance. The state of each appliance is monitored. If an appliance or queue manager fails, queue managers that are running on that appliance begin running on the other appliance.

The seamless switch-over is achieved by using synchronous replication of queue manager data and logs. Synchronous replication was chosen for reasons that are related to DR. Supposing a disaster event occurs that affects a single appliance in the HA pair. If the compromised appliance was running a certain HA queue manager, the latest state should be replicated on the healthy appliance. When monitoring the healthy appliance detects the failure, it starts its copy of the queue manager, which minimizes loss of service. This approach is the only supported approach to HA on the appliance.

Regarding failures that can occur and trigger the failover of queue managers from one appliance to another, the following types of monitoring can occur:

- ▶ Within the HA group in which there is an appliance that is running an HA queue manager, the health of the queue manager is monitored. If a problem is detected, the queue manager can be failed over to the secondary appliance by the primary appliance for that queue manager).
- ▶ Each appliance also monitors the other appliance in the HA group. If APPLIANCE1 that is shown in Figure 10-2 on page 153 were to experience a total failure of the power supply unit for example, APPLIANCE2 rapidly notices that APPLIANCE1 is no longer communicating. It can then take action to bring ownership of the queue managers that are in under HA control.

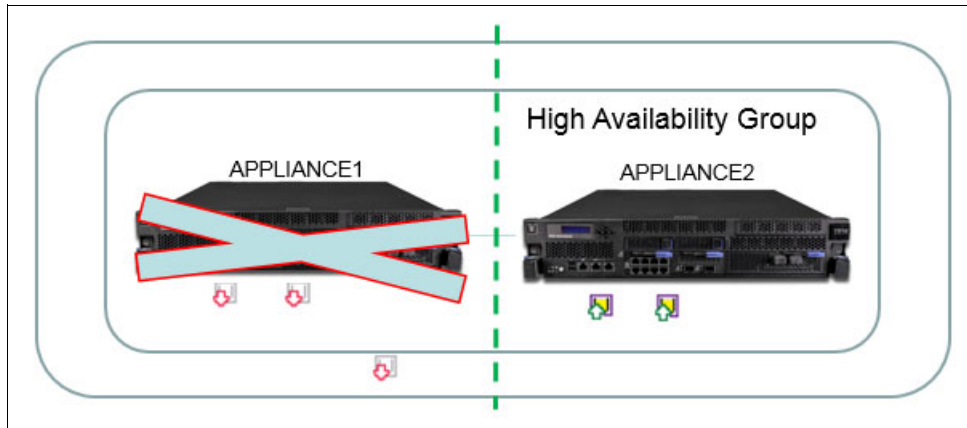


Figure 10-2 Queue managers that are running on the available IBM MQ Appliance after failure

10.4 Preparing for HA

The following preparatory tasks must be completed before an HA group can be created:

- ▶ An IBM MQ administrator account is used on both appliances.
- ▶ CLI access is available to both appliances.
- ▶ Addresses must be decided on and the HA interfaces configured for each appliance. These addresses must be used for HA only.
- ▶ The IP address of the HA primary interface on each appliance (eth13) is established.
- ▶ The HA primary, secondary, and replication interfaces are physically connected.
- ▶ The round-trip latency time for a message is less than 10 ms.

10.5 Creating an HA group

Example 10-1 shows the available HA commands.

Example 10-1 Available HA commands in IBM MQ administration mode

```

M2000# mqcli
M2000(mqcli)# help ha
The following HA commands are available. Type help <command> for more
information.
crthgrp      Creates an HA group with a prepared appliance
dlthgrp      Deletes an HA group
dsphgrp      Displays the status of the appliances in an HA group
makehaprimary Resolves a partitioned situation in an HA group
prepareha    Prepares an appliance to be part of an HA group
sethgrp      Pauses and resumes an appliance in an HA group
clearhapreferred Clears the preferred appliance for an HA queue manager
sethapreferred Sets the preferred appliance for an HA queue manager

```

Table 10-1 lists the tasks that must be completed in the order that are presented to create an HA group.

Table 10-1 Creating an HA group

Appliance A	Appliance B
<pre>M2000# mqcli M2000(mqcli)# prepareha -s ABCD1234 -a 192.168.101.2 Waiting up to 600 seconds for remote system to respond. M2000(mqcli)# M2000(mqcli)# dsphagrp This Appliance: Online Appliance mqa002p: Online M2000(mqcli)#</pre>	<pre>M2000# mqcli M2000(mqcli)# crthagrp -s ABCD1234 -a 192.168.101.1 Checking network configuration Configuring HA Group on this appliance Configuring HA Group on the other appliance This Appliance: Online Appliance mqa001p: Online M2000(mqcli)# M2000(mqcli)# dsphagrp This Appliance: Online Appliance mqa001p: Online M2000(mqcli)#</pre>

The command prompt becomes available on the primary appliance before the secondary appliance finishes configuring the HA group. However, both appliances in the HA group are online only when the commands complete on both appliances.

Note: An appliance cannot be a member of more than one group.

10.6 Creating an HA queue manager

The following methods can be used to add a queue manager to an HA group:

- ▶ Add the manager when it is created
- ▶ Use the **sethagrp -i** command, which adds the queue manager that is specified
- ▶ Create the HA queue manager by using the IBM MQ Console

The following options are available:

- ▶ Option 1

Use the following command:

```
crtmqm -sx QMgrName
```

This command creates and starts the queue manager on the appliance to which you sent the command and uses the other appliance as secondary. This method is the simplest and quickest way of creating an HA queue manager.

► Option 2

Use the following command:

```
crtmqm QMgrName
strmqm QMgrName
--- Non-HA configuration steps ---
sethagrp -i QMgrName
```

This command breaks up the queue manager setup and enables non-HA configuration and testing before a queue manager is added to an HA group. If you want to consolidate a queue manager onto an appliance and use HA, this method is likely to be preferred because you can ensure that the queue manager settings are correct before enabling replication and it allows the tasks to be separated.

If you use the second method to create the queue manager, the queue manager must be started to add it to the HA group. After it is added, the queue manager must synchronize across both appliances. This process can take some time (approximately 30 minutes for a queue manager with a 64 GB file system). During this time, the queue manager can remain running or it can be stopped. You can perform other configuration steps while the synchronization process is progressing.

Until synchronization is complete, you can run only the queue manager on the appliance on which it was created. The synchronization status can be queried by using the **status <QMGrName>** command, as shown in Example 10-2.

Example 10-2 Creating an HA queue manager and querying queue manager synchronization status

```
M2000(mqcli)# crtmqm -sx GTWY1HAP
Please wait while 64 GB file system is initialized for queue manager 'GTWY1HAP'.
Performing initial high availability configuration
Configuring appliance mqa001p
Configuring this appliance
Initial high availability configuration succeeded
IBM MQ Appliance queue manager created.
The queue manager is associated with installation 'MQAppliance'.
Creating or replacing default objects for queue manager 'GTWY1HAP'.
Default objects statistics : 83 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
Performing final high availability configuration
Final high availability configuration succeeded
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqcli)# status GTWY1HAP
```

QM(GT)	Status(Running)
CPU:	0.03%
Memory:	198MB
Queue manager file system:	229MB used, 63.0GB allocated [0%]
HA role:	Primary
HA status:	Synchronization in progress
HA control:	Enabled
HA preferred location:	This appliance
Synchronization progress:	2.4%
Estimated synchronization time:	2015-07-27 17:47:57.877

10.7 Setting the preferred appliance for a queue manager

One of the appliances in the group can be selected as the preferred appliance to run a specific highly available queue manager. This method is a useful way to enable the two appliances in the group to both be active.

Although this method helps to achieve a good workload balance across appliances, care must be taken not to overload appliances. If one appliance fails in an HA group, the other appliance must perform satisfactorily while carrying the full load that normally is split between the systems.

After a queue manager is added to an HA group, you can use the **sethappreferred** command to specify a preferred appliance for it. Although this specification can be set while the queue manager is synchronizing, the appliance that is running the queue manager changes only after the synchronization is complete.

10.8 Suspending and resuming an appliance

You might want to suspend an appliance from your HA group; for example, for a firmware upgrade. You can suspend an appliance by using the **sethagrp** command, as shown in Table 10-2 on page 157 and Table 10-3 on page 158.

Table 10-2 Suspending an appliance in an HA group

Appliance A	Appliance B
<pre> M2000# mqccli M2000(mqccli)# dspmq -o all QMNAME(GTWY1HAP) STATUS(Running) DEFAULT(no) HA(Replicated) QMNAME(GTWY2HAP) STATUS(Running elsewhere) DEFAULT(no) HA(Replicated) M2000(mqccli)# sethagrp -s AMQ6586: The sethagrp command succeeded. Initially... M2000(mqccli)# dspmq -o all QMNAME(GTWY1HAP) STATUS(Ended normally) DEFAULT(no) HA(Replicated) QMNAME(GTWY2HAP) STATUS(Running elsewhere) DEFAULT(no) HA(Replicated) M2000(mqccli)# status GTWY1HAP QM(GTWY1HAP) Status(Running) CPU: 0.00% Memory: 199MB Queue manager file system: 229MB used, 63.0GB allocated [0%] HA role: Primary HA status: Normal HA control: Enabled HA preferred location: This appliance After a short time... M2000(mqccli)# dspmq -o all QMNAME(GTWY1HAP) STATUS(Running elsewhere) DEFAULT(no) HA(Replicated) QMNAME(GTWY2HAP) STATUS(Running elsewhere) DEFAULT(no) HA(Replicated) M2000(mqccli)# status GTWY1HAP QM(GTWY1HAP) Status(Running elsewhere) HA role: UNKNOWN HA status: UNKNOWN HA control: Enabled HA preferred location: This appliance </pre>	<pre> M2000# mqccli M2000(mqccli)# dspmq -o all QMNAME(GTWY1HAP) STATUS(Running elsewhere) DEFAULT(no) HA(Replicated) QMNAME(GTWY2HAP) STATUS(Running) DEFAULT(no) HA(Replicated) Initially... M2000(mqccli)# dspmq -o all QMNAME(GTWY1HAP) STATUS(Running elsewhere) DEFAULT(no) HA(Replicated) QMNAME(GTWY2HAP) STATUS(Running) DEFAULT(no) HA(Replicated) M2000(mqccli)# status GTWY1HAP QM(GTWY1HAP) Status(Running elsewhere) HA role: Secondary HA status: Normal HA control: Enabled HA preferred location: Other appliance After a short time... M2000(mqccli)# dspmq -o all QMNAME(GTWY1HAP) STATUS(Running) DEFAULT(no) HA(Replicated) QMNAME(GTWY2HAP) STATUS(Running) DEFAULT(no) HA(Replicated) M2000(mqccli)# status GTWY1HAP QM(GTWY1HAP) Status(Running) CPU: 0.01% Memory: 199MB Queue manager file system: 229MB used, 63.0GB allocated [0%] HA role: Primary HA status: Secondary appliance unavailable HA control: Enabled HA preferred location: Other appliance </pre>

Table 10-3 Resuming an appliance in an HA group

Appliance A	Appliance B
M2000(mqcli)# M2000(mqcli)# sethagrp -r AMQ6586: The sethagrp command succeeded. <i>Initially...</i> M2000(mqcli)# status GTWY1HAP QM(GTWY1HAP) Status(Running elsewhere) HA role: Secondary HA status: Normal HA control: Enabled HA preferred location: This appliance <i>After a short time...</i> M2000(mqcli)# status GTWY1HAP QM(GTWY1HAP) Status(Running) CPU: 0.03% Memory: 199MB Queue manager file system: 229MB used, 63.0GB allocated [0%] HA role: Primary HA status: Normal HA control: Enabled HA preferred location: This appliance	M2000(mqcli)# <i>Initially...</i> M2000(mqcli)# status GTWY1HAP QM(GTWY1HAP) Status(Running) CPU: 0.00% Memory: 199MB Queue manager file system: 229MB used, 63.0GB allocated [0%] HA role: Primary HA status: Normal HA control: Enabled HA preferred location: Other appliance <i>After a short time...</i> M2000(mqcli)# status GTWY1HAP QM(GTWY1HAP) Status(Running elsewhere) HA role: Secondary HA status: Normal HA control: Enabled HA preferred location: Other appliance

A suspend request safely ends all HA queue managers on the appliance on which it is requested and starts them on the other appliance in the group. A resume restarts the HA group. After a short time, it ensures that each queue manager is running on its preferred appliance. If you suspend both appliances in an HA pair, the queue managers are ended.

The IBM MQ Console can be used to suspend an IBM MQ Appliance from the HA group. Complete the following steps:

1. Log in to the console, as shown in Figure 10-3.

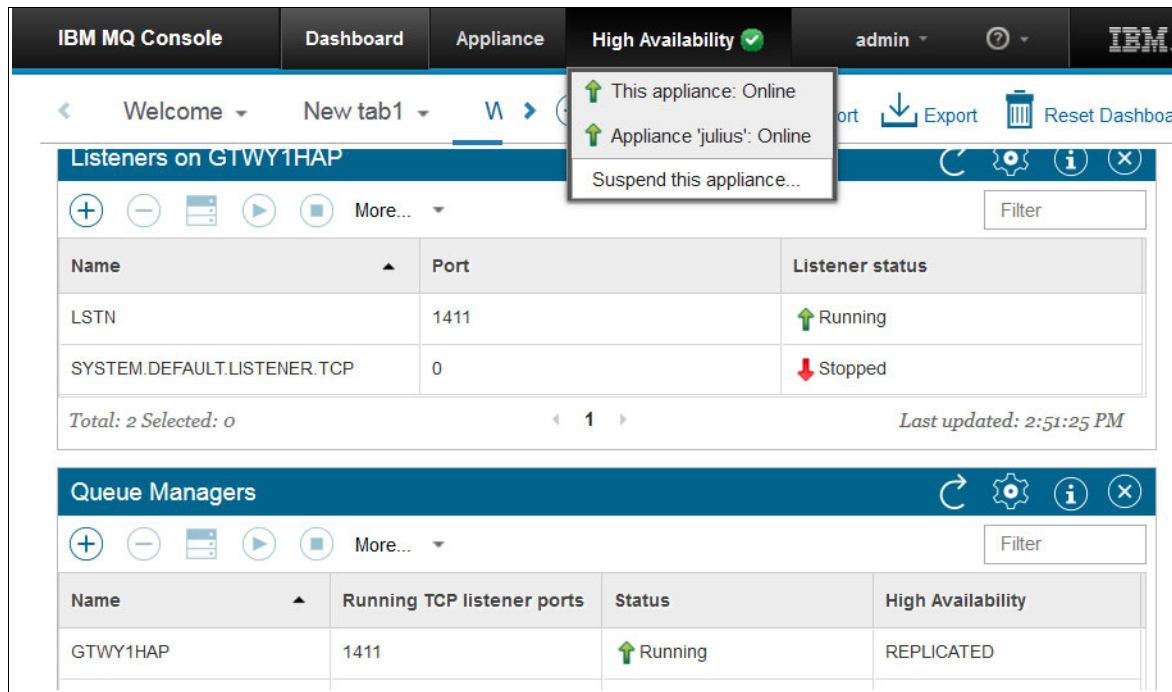


Figure 10-3 IBM MQ Console High Availability Menu

2. Select the **Suspend** option. A warning message displays, as shown in Figure 10-4. After a short time, the appliance is suspended from the HA group.

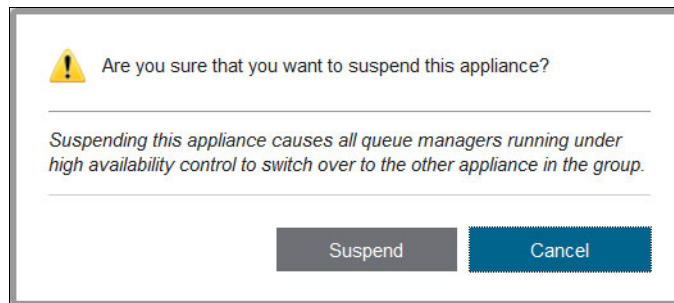


Figure 10-4 Suspend Warning message

3. Any HA Queue managers move to the available member of the HA group and continue to function. As shown in Figure 10-5 on page 160, the High Availability menu option now displays a warning symbol. This message informs users that the appliance to which the user is logged in to is not functioning as a member for the HA group to which it belongs.

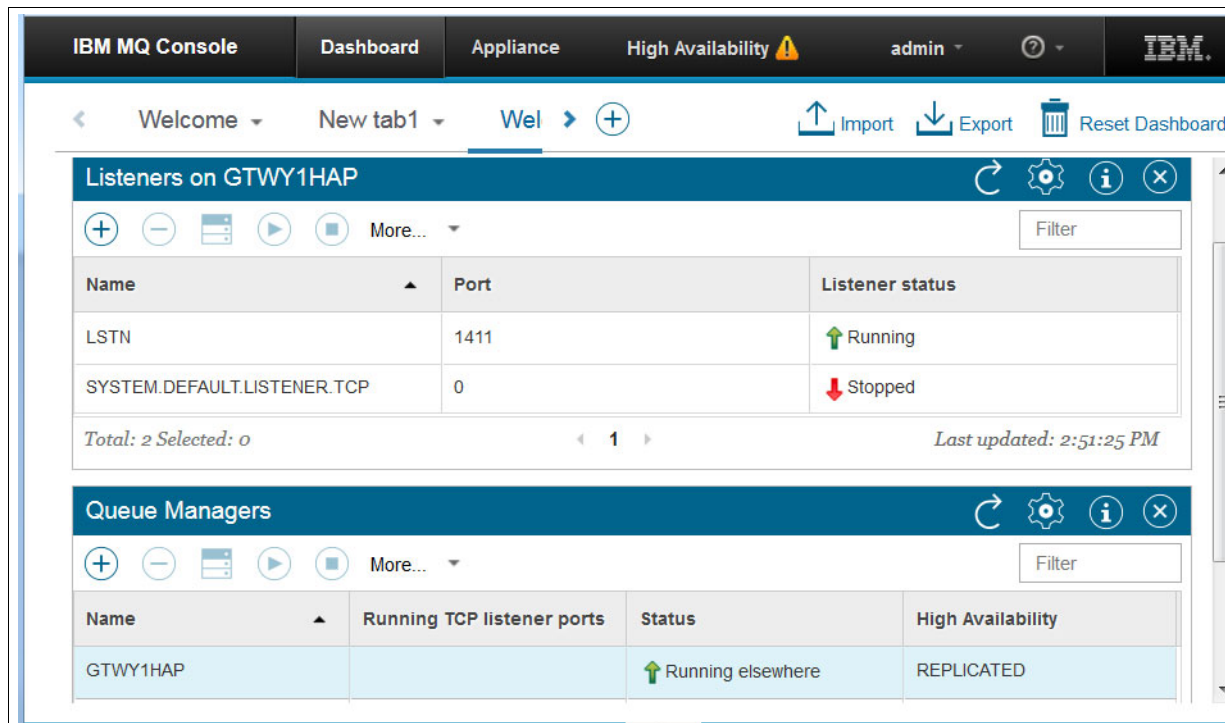


Figure 10-5 Appliance that is suspended from its HA Group

10.9 Applying a fix pack to an HA group

If you have two appliances (A and B) in an HA group and are using only HA-enabled queue managers, appliances can be upgraded without incurring downtime. Complete the following steps:

1. Suspend the HA group on appliance A. Ensure that the suspend process ran correctly and the queue manager is running successfully on appliance B. Avoid outages on appliance B.
2. Run the normal firmware upgrade on appliance A, as described in Chapter 6, “Appliance administration” on page 55.
3. When appliance A is available, rejoin the HA group and ensure that all queue managers are fully synchronized and replicated.
4. Repeat step 1 for appliance B.
5. Run the normal firmware upgrade on appliance B as described in Chapter 6, “Appliance administration” on page 55. When appliance B is available, rejoin the HA group and ensure that all queue managers are synchronized and replicated.

10.10 HA scenario

This section describes HA configuration requirements in terms of the business scenario that is described in Chapter 2, “Business scenario overview” on page 11.

10.10.1 Defining a cluster receiver channel

GTWY1HAP requires that a cluster receiver channel (CLUSRCVR) channel is defined as with any other queue manger that wants to join a cluster. However, because this queue manager can be running in two different locations (that is, the two appliances), the channel must be defined with each of the appliances IP addresses that are specified as the connection name. The basic definition for our CLUSRCVR channel is as shown in Example 10-3.

Example 10-3 CLUSRCVR channel definition

```
DEFINE CHANNEL(TO.GTWY1HAP) +  
          CONNAME('mqa001p.companyb.local(1411),  
                  mqa002p.companyb.local(1411)') +  
CLUSTER(CLUSTER1P)
```

10.10.2 Creating a host alias for a listener

The listener that is defined for our queue manager must run at both locations. Unfortunately, we cannot specify two different IP addresses in the listener definition. If a failover of the GTWY1HAP occurs and assuming it was not a problem with the listener that triggered a failover to occur, the listener is still listening on the original IP address to which it was bound.

This situation can be resolved by defining a host alias definition for each appliance that is configured with the wanted IP address of the data port that is used.

It is possible to specify only a single address for a listener. However, we want the listener to listen on the two appliance addresses. To enable this configuration, a host alias is created on both appliances, each provided its own host name for the alias.

A host alias can be defined by using the Manager Appliance section of the web-based administration interface, as shown in Figure 10-6 on page 162. Complete the following steps:

1. Click **Network** → **Interface** → **Host Alias**.
2. Click **New**.
3. In the Name field, enter the alias name. This name cannot begin with the reserved letters “eth” or “mgt”.
4. Set the Enable administrative state to identify the administrative state of the configuration.
5. (Optional) In the Comments field, enter a descriptive summary.
6. In the IP Address field, enter a local IP address.
7. Click **Apply** to save the changes to the running configuration.
8. (Optional) Click **Save Config** to save the changes to the start configuration.

Host Alias

✓ Host Alias

lstn-alias *

?

★ Name:

lstn-alias

▼ Main

Enable administrative state: ?

☒

Comments: ?

host alias for queue manager listener

★ IP address: ?

9.20.87.135

Figure 10-6 Use of web-based interface to configure a host alias

Now that the alias is defined on the two appliances of the HA configuration, we can define the listener to use a listener with an IP address that uses the host alias name.

10.10.3 Defining a listener

The new HA queue manager GTWY1HAP can now have a listener that is defined for inbound connections to the available queue manager. The listener is defined as shown in Example 10-4.

Example 10-4 Listener definition

```

DEFINE LISTENER(LSTN) +
    TRPTYPE(TCP) +
    CONTROL(QMGR) +
    IPADDR('lstn-alias') +
    PORT(1411)
  
```

162 Integrating the IBM MQ Appliance into your IBM MQ Infrastructure



Application changes

A queue manager that is running on the IBM MQ Appliance M2000 is a distributed IBM MQ version 8 queue manager with the following special characteristics:

- ▶ No server bindings: All applications use client bindings.
- ▶ No exits on the queue manager. Client-side exits are allowed.
- ▶ New high availability (HA) support, but the client interface is the same as for a multi-instance queue manager.

This chapter describes what these changes mean for applications that use MQ. It also describes what must change in an application to achieve the same sort of capabilities that were provided by queue managers on distributed platforms.

It also describes what does not change. There is a compatibility expectation between versions and platforms. The IBM MQ Appliance delivers on that expectation.

This chapter includes the following topics:

- ▶ 11.1, “Just another queue manager” on page 164
- ▶ 11.2, “Client scenarios” on page 164
- ▶ 11.3, “Summary” on page 169

11.1 Just another queue manager

One of the most important attributes of queue managers that are running on the IBM MQ Appliance is that to an application, they are just another queue manager. Any well-written application that uses IBM MQ Client libraries to connect to a distributed queue manager can instead connect to queue manager on an appliance.

Depending on the platform your current queue managers are on, the default Coded Character Set ID (CCSID) might change. However, this change can occur when connecting to a different distributed queue manager that is running on a different platform. If the application is sensitive to the CCSID of the queue manager, you might have to change the queue manager CCSID or change the application to request conversion of messages to the correct CCSID.

Support for HA is provided by a pair of IBM MQ Appliances that are configured in an HA group. How the queue managers are configured and the infrastructure that is needed is different from creating queue managers that run as multi-instance queue managers. For more information about creating an HA group and an HA queue manager, see Chapter 6, “Appliance administration” on page 55 and Chapter 7, “Creating queue managers” on page 107.

However, the client interface is identical to multi-instance queue managers. There is one queue manager with two possible IP addresses. Both IP addresses are included in the Connection Name (CONNAME) of the client channel definition. This definition can appear in a CLNTCONN channel, an MQSERVER environment variable, a JNDI connection factory, or anywhere else the specific client obtains its connection information.

The behavior of HA queue managers after failure and recovery is slightly different. A multi-instance queue manager continues to run wherever it is until it fails or is forced to fail over. An HA queue manager always returns to its preferred appliance as soon as it can. However, the behavior that is visible to the client remains unchanged. A client that is running in automatic reconnection mode sees the recovery back to the preferred appliance as just another failover.

11.2 Client scenarios

Different client applications feature different requirements for recoverability and different usage of queue manager services. Possible scenarios of these application styles are described in the following sections. Examples of queue manager and channel table configurations to support the application requirement also are provided.

The examples in the scenarios show the creation of client connection channels, which in turn create client channel tables that an application uses. The application can also achieve the same results by using other techniques, such as MQCONN calls with appropriate structures.

In each case, queue manager resources, such as channels and queues, are also needed, but the process of creating these components is not shown.

The examples are based on an assumption that the current application is using server binding and must change to client binding to use an appliance-based queue manager. If the application is client-connected to the queue manager, it can use the appliance queue manager without changes other than the queue manager name and connection string.

11.2.1 Synchronous requester application

This application style is common in interactive front-end programs. It is commonly called a *requester application*, or a *service requester*. A user makes a request for information, which is handled by the front end application. The application formats a request message and sends the message to queue that is hosted by a queue manager on an IBM MQ Appliance. The application then waits for a limited time (for example, 30 seconds or less) for a response message to arrive on a ReplyToQueue. While the application waits, IBM MQ sends the message on to a server application. The server application processes the request, formats a response message, and puts the response message to the ReplyToQueue at the ReplyToQueueManager.

Figure 11-1 shows two IBM MQ Appliances, mqa001p and mqa002p. They each have one queue manager, REQ1QMP and REQ2QMP. They each have one

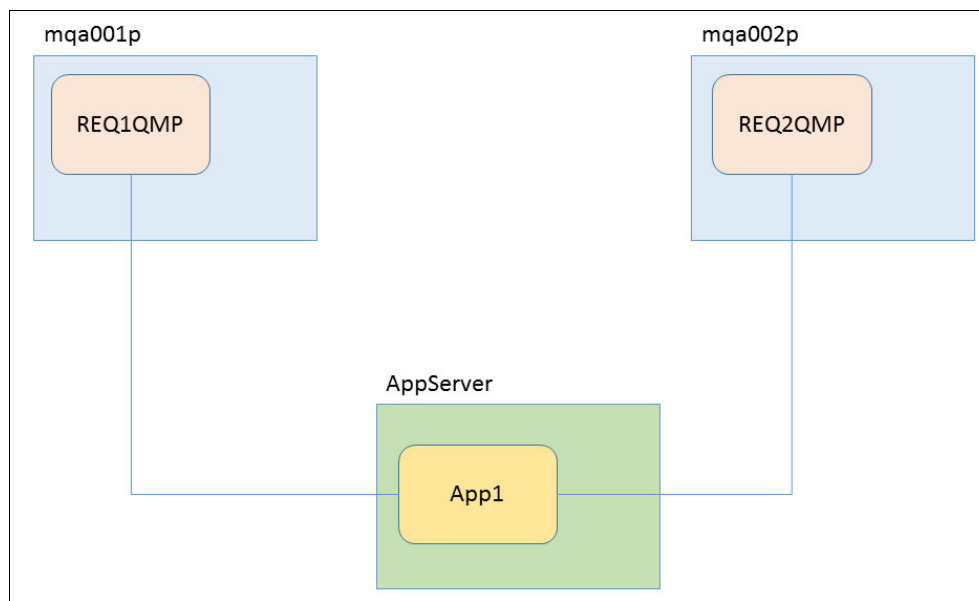


Figure 11-1 Synchronous Request-Response application

It does not matter which queue manager (REQ1QMP or REQ2QMP) to which the application connects. In either case, the message can reach the intended destination and the response can return to the ReplyToQueue.

A suitable channel table for this application might contain client connection channels, as shown in Example 11-1. The application requests connection to the queue manager *REQ.

*Example 11-1 Client channel table for logical queue manager *REQ used by App1*

```
DEFINE CHANNEL(REQ1QMP.APP1) -  
  CHLTYPE(CLNTCONN) -  
  TRPTYPE(TCP) -  
  CONNAME('mqa001p.companyb.local(5001)') -  
  QMNAME(REQ) -  
  DEFRECON(NO) -  
  AFFINITY(NONE) -  
  CLNTWGHT(50) -  
  SSLCIPH(TLS_RSA_WITH_AES_256_CBC_SHA256) -  
  SSLPEER('CN=*,OU=*,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US') -  
  REPLACE
```

```

DEFINE CHANNEL(REQ2QMP.APP1) -
  CHLTYPE(CLNTCONN) -
  TRPTYPE(TCP) -
  CONNAME('mqa002p.companyb.local(5002)') -
  QMNAME(REQ) -
  DEFRECON(NO) -
  AFFINITY(NONE) -
  CLNTWGHT(50) -
  SSLCIPH(TLS_RSA_WITH_AES_256_CBC_SHA256) -
  SSLPEER('CN=*,OU=*,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US') -
  REPLACE

```

If the connected queue manager fails, the application receives an error (MQRC=2009). The application can then choose to end or clear down the connection and start again. The connection attempt succeeds because one of the queue managers is still available.

11.2.2 Server application (query only)

Applications that receive messages, process them, and send back a response can be called “server” applications or service provider applications. That is, they provide a service to others.

In an IBM MQ environment with multiple instances of a queue that is used for requests to the service, every instance of the queue must have an application that is connected to it. Otherwise, if all application instances connected to one of the two queue instances, messages that went to the other instance are not processed.

The configuration of these service provider applications must be different from the service requester that does not care to which queue manager they connect.

Figure 11-2 shows two application instances, each of which is configured to connect to a specific queue manager that hosts an instance of the queue. They can share a single channel table, as shown in Example 11-2 on page 167, but each application instance is configured to use a specific queue manager from the channel table.

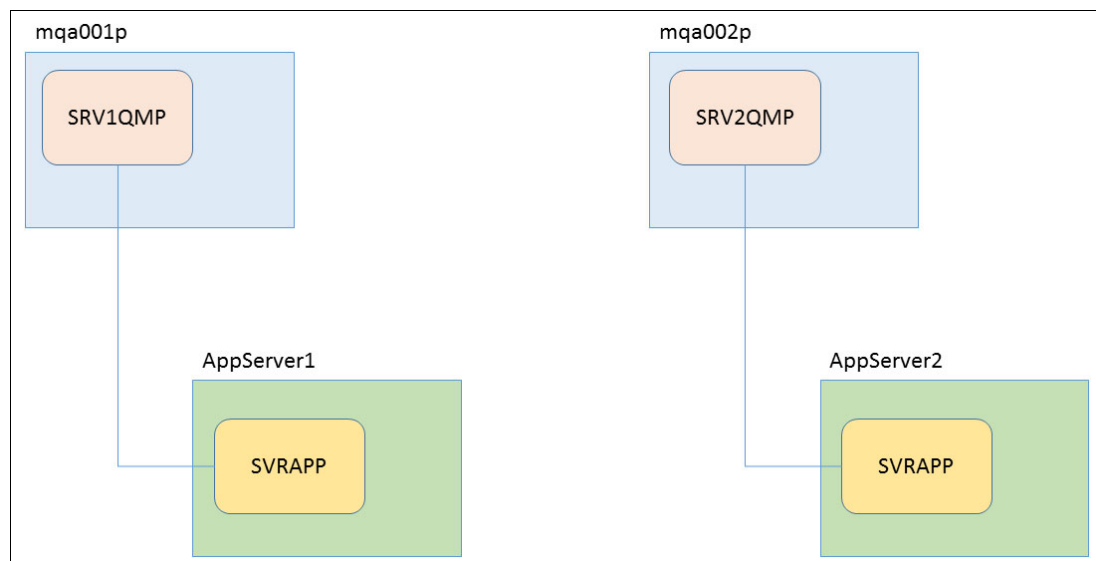


Figure 11-2 Two instances of a service application, connecting to two queue managers

Example 11-2 Client channel table for two instances of the service provider

```
DEFINE CHANNEL(SRV1QMP.SVRAPP) -
  CHLTYPE(CLNTCONN) -
  TRPTYPE(TCP) -
  CONNAME('mqa001p.companyb.local(5011)') -
  QMNAME(SRV1QMP) -
  DEFRECON(NO) -
  AFFINITY(NONE) -
  CLNTWGHT(0) -
  SSLCIPH(TLS_RSA_WITH_AES_256_CBC_SHA256) -
  SSLPEER('CN=SRV1QMP,OU=*,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US') -
  REPLACE

DEFINE CHANNEL(SRV2QMP.SVRAPP) -
  CHLTYPE(CLNTCONN) -
  TRPTYPE(TCP) -
  CONNAME('mqa002p.companyb.local(5012)') -
  QMNAME(SRV2QMP) -
  DEFRECON(NO) -
  AFFINITY(NONE) -
  CLNTWGHT(0) -
  SSLCIPH(TLS_RSA_WITH_AES_256_CBC_SHA256) -
  SSLPEER('CN=SRV2QMP,OU=*,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US') -
  REPLACE
```

Note: In the requester example, the application can connect to either queue manager because it used the queue manager name *REQ to search the channel table. In this example, two instances of the application are started. One configured for queue manager SRV1QMP, and the other for SRV2QMP. Each application instance always connects to only one queue manager.

11.2.3 Server application (reliable update)

The final simple client application scenario is for a server that is performing some sort of persistent update (see Figure 11-3 on page 168). The requesting applications want high assurance that the update can occur and that the message is not lost.

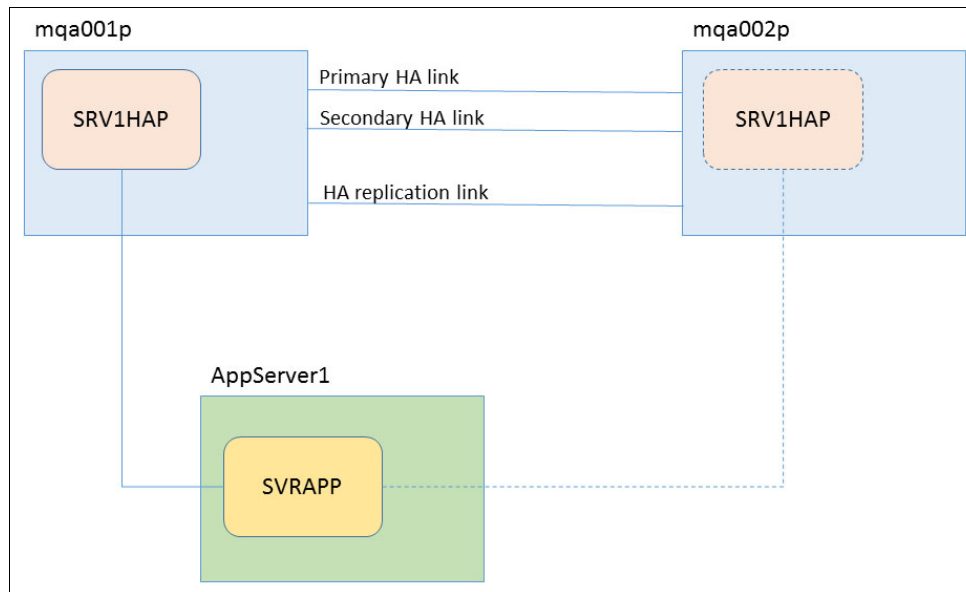


Figure 11-3 One server application connecting to a highly available queue manager

Persistent messages are sent to the queue manager that is hosting the service queue. The queue manager is hosted on an HA group on two appliances. At any time, one appliance hosts the queue manager while the other appliance keeps its files in sync with the first and watches for failures.

The client channel table that is shown in Example 11-3 has only one channel definition because there is only one queue manager. In this case, the connection name includes two addresses and the reconnection parameter (DEFRECON) is set to YES. The application connects to queue manager SRV1HAP.

Example 11-3 Client channel table to enable reconnection to a highly available queue manager

```

DEFINE CHANNEL(SRV1HAP.APP1) -
  CHLTYPE(CLNTCONN) -
  TRPTYPE(TCP) -
  CONNAME('mqa001p.companyb.local(5003),mqa002p.companyb.local(5003)') -
  QMNAME(SRV1HAP)
  DEFRECON(YES) -
  AFFINITY(NONE) -
  CLNTWGHT(0) -
  SSLCIPH(TLS_RSA_WITH_AES_256_CBC_SHA256) -
  SSLPEER('CN=SRV1HAP,OU=*,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US') -
  REPLACE
  
```

The application is written so that all IBM MQ operations occur inside a unit of work (UOW). Other persistent operations, such as database inserts, updates, or deletes also are inside the UOW. If a failover occurs during the unit of work, this failover is recognized when the application attempts to commit. At that point, the application is informed of a successful commit or a rollback. If a rollback occurs, the message is restored to the input queue and any database changes are undone. An MQGET operation can receive the message and reprocess the transaction.

11.3 Summary

The following major differences between connecting applications to a queue manager on an appliance and another distributed queue manager are important:

- ▶ They must be connected by using a client channel
- ▶ Server-side exits are not possible

Any client application must connect to an appliance-hosted queue manager instead with minimal issues. Make note of the CCSID requirements of the application based on the current application and queue manager platforms. If there is a change in the CCSID, more testing is needed and an application code change might be required so that IBM MQ performs conversions on behalf of the application.

Changes also might be required to an application that uses server bindings to connect to a queue manager to make it run well by using an appliance-hosted queue manager. These changes are the same as are required migrating the application to client bindings on any platform. The changes are not related to the queue manager that is running on an appliance.

Different application styles need different client connection styles and use client channel connection tables in different ways. All of the different styles can coexist in a single channel table. Multiple tables can be created for different applications.



Support for the IBM MQ Appliance

This chapter describes some basic steps that you can take to troubleshoot your IBM MQ Appliance and where to find the information that you need to fix the problem. Because of the appliance form factor, support for the IBM MQ Appliance differs from support for IBM MQ software support; therefore, the scope of IBM Support also is described.

At time of this writing, development of IBM MQ uses a continuous delivery model. This development means that there are more frequent updates to IBM MQ, which incrementally delivers small fixes or extra function. If customers upgrade their infrastructure in step with the frequent updates, they can avoid the large risks that are associated with upgrading from one major version to another.

To reduce these risks, the IBM Support organization suggests that companies support the continuous delivery model by upgrading their infrastructure in step with releases. The first step in any request for IBM MQ Appliance support can be to ensure that you are upgraded to the latest firmware.

This chapter includes the following topics:

- ▶ 12.1, “Scope of appliance support” on page 172
- ▶ 12.2, “Fix process” on page 173
- ▶ 12.3, “Support for the IBM MQ installation on the appliance” on page 173
- ▶ 12.4, “Appliance support” on page 176
- ▶ 12.5, “Example support scenario” on page 180

12.1 Scope of appliance support

The aim of support for the IBM MQ Appliance is to make problem ownership clear. Because IBM provides the hardware and software, IBM also provides the support to fix the appliance if you encounter a problem.

To clarify this distinction further, custom code cannot be run on the IBM MQ Appliance. This restriction has implications for any user exits that you might be using on other queue managers that you might want to consolidate onto an IBM MQ Appliance.

12.1.1 Removing user exits

User exits allow users to view and log detailed information that is related to the behavior of applications that are connected to queue managers. Typically, user exits were used for queue manager monitoring, message statistic gathering, and security.

Although user exits are useful, they include the following undesirable qualities:

- ▶ User exits are often loaded and run in the queue manager address space, which can affect performance or even queue manager stability.
- ▶ Exit development and testing are entirely the customers' responsibility.
- ▶ Debugging problems for customer exits might not be possible because of environment constraints and might not be permitted.
- ▶ If the IBM MQ infrastructure is upgraded, the exit is not guaranteed to work without more modification.

User exits typically allowed users to capture and, if required, log detailed information that is related to the behavior of applications that are connected to queue managers. Exits that are used with queue managers also were used for queue security duties and modifying channel behavior.

Over the years, versions of IBM MQ made available alternative methods of performing the roles that exits played in most customer situations. For example, Channel Authentication (CHLAUTH) features that were added to WebSphere MQ v7.1 provided a means to add security measures that users exits performed. Also introduced in WebSphere MQ V7.1, the Application Activity Trace features enabled users to gather information about an application's use of MQPUT and MQGET, for example.

Therefore, user exit installation is not a capability of the IBM MQ Appliance. For more information about configuring the Activity Tracing on the IBM MQ Appliance, refer to the following product documentation:

- ▶ Setting initialization (setmqini):
http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/reference/mqconfigcmd/setmqini.htm
- ▶ Displaying initialization file (dspmqini):
http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/reference/mqconfigcmd/dspmqini.htm

On the IBM MQ Appliance, activity trace is used for generating monitoring widgets within the IBM MQ Console and for the new **status** command that is found in IBM MQ administration mode within the command-line interface (CLI).

12.2 Fix process

The fix process differs for software or hardware problems. However, unless some basic troubleshooting as described in this chapter can solve the problem in either case, your primary task is to contact IBM support.

12.2.1 Software problem

If a software problem is found, the fix is provided in the form of a firmware upgrade file (an `script3`). There is no other way of applying a fix to the IBM MQ Appliance.

By using the firmware upgrade process that is described in the firmware upgrade section of Chapter 6, “Appliance administration” on page 55, you must upgrade your appliance. If your appliance is in a high availability (HA) group, ensure that you suspended the HA group on the appliance you want to upgrade and took care to avoid outages on the other appliance before upgrading.

12.2.2 Hardware problem

If a hardware problem occurs, see your IBM Hardware Warranty Information documentation. By default, support is provided 9 hours a day, 6 days a week. If you have the Business Critical Support package, 24/7 support is available.

If a certain part fails in an IBM MQ Appliance, see Chapter 7 of your Installation and User's Guide, which is found on your IBM MQ M2000 Resource CD that is included with each IBM MQ Appliance.

If you are shipped a new IBM MQ Appliance, set your entitlement ID as the serial number of the previous IBM MQ Appliance, as described at this website:

http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/reference/syscmds/entitlement_systemsettings.htm

12.3 Support for the IBM MQ installation on the appliance

There are various differences to the overall picture of IBM MQ on the IBM MQ Appliance.

Some new tools were written to make certain tasks possible, some existing tools were updated, and certain IBM MQ files were put in slightly different places because of the different file structure.

12.3.1 New tools for basic troubleshooting

To view logs, you can use the `dspmqrerr` command, which is found in `mqcli` mode. You can view specific error files or list all available files, as shown in Example 12-1.

Example 12-1 Viewing IBM MQ logs on the IBM MQ Appliance

```
M2000# mqcli
M2000(mqcli)# help dspmqrerr
Usage: dspmqrerr [-f | -m QmgrName | -s | -w] [-l | FileName]

-f    File type: FDCs.
```

```
-l List files of specified type.
-m File type: Queue-manager error log.
-s File type: System error log.
-w File type: Web UI log.
M2000(mqcli)# dspmqerr -l
MQSystem.log 14843 bytes Mon Jul 13 13:07:42 2015
M2000(mqcli)# dspmqerr MQSystem.log [this opens in a viewer]
```

Deleting old logs

You can delete files under most URIs by using the delete function that is found in the configuration mode of the CLI, as described in Chapter 6, “Appliance administration” on page 55. However, this function deletes only one file at a time. You can delete old IBM MQ log files in bulk in `mqcli` mode by using the **dltmqras** command, as shown in Example 12-2.

Example 12-2 Deleting files in bulk on the IBM MQ Appliance

```
M2000# mqcli
M2000(mqcli)# help dltmqras
Usage: dltmqras [ -y ] -a | -c | -d | -f | -h | -m QMName | -p | -t

-a Purge files of all listed types except QM diagnostics.
-c File type: MQ Console.
-d File type: General diagnostics.
-f File type: FDCs.
-h File type: HA.
-m File type: Queue Manager diagnostics.
-p File type: Temporary.
-t File type: Trace.
-y Run without prompt.
M2000(mqcli)# dltmqras -a
File found: messages.log
File found: messages_15.07.13_08.16.29.0.log
File found: messages_15.07.10_06.39.30.0.log
3 removable diagnostic file(s) found.
Do you want to delete these files? [Y/N]
y
3 diagnostic file(s) removed.
M2000(mqcli)#
```

As shown in Example 12-2, the usage message is requested first, which shows each file type that is covered by **dltmqras**. Then, **dltmqras -a** is requested, which searches for old log files of every listed type. Any files that are found are displayed. If you want to delete these files, you must enter a lower or uppercase `y`. The listed files are then deleted.

12.3.2 Differences in familiar commands and familiar file locations

The IBM MQ MustGather command, **runmqras**, underwent several changes for use on the IBM MQ Appliance to remove some switches that are irrelevant for the appliance. To collect normal IBM MQ trace (used with IBM Support) or new HA trace, you must specify **runmqras -section trace**. The IBM MQ MustGather command also starts the WebSphere Application Server Liberty MustGather tool if **-section webui** is specified.

Note: IBM MQ trace is meant to be used with IBM Support. Therefore, although you can start and end trace as normal through the CLI in IBM MQ configuration mode, there is no trace formatter available on the IBM MQ Appliance. To collect trace output, you can use `runmqgras -section trace`.

Because there is no way to capture displayed output, `dmpmqcfg` (which is used to save queue manager configurations) was changed to write its output to file under the `mqbackup://` URI.

Client Channel Definition Tables (CCDTs), which allow users to back up channel definitions for queue managers CCDTs on the appliance, have an altered filename and appear under the `mqbackup://` URI, named `<QmgrName>_AMQCLCHL.TAB`.

12.3.3 Location and names of IBM MQ error logs on the IBM MQ Appliance

Because the appliance file system is based on URIs, the process to access error logs is different. You might want to export logs from the appliance for further troubleshooting or monitoring. Before you can export error logs, you must know where they are found on the appliance.

A new IBM MQ error log was added to IBM MQ for the IBM MQ Appliance (see Table 12-1). The file is called `MQSystem.log` and it logs appliance-specific events, such the status of various daemon processes, warning messages, and errors.

Table 12-1 Log types, where to find them and how to view them

Log type	Log name	Log location	Command to view log (from IBM MQ administration mode in the CLI)
General IBM MQ log	<code>MQSystem.log</code>	<code>mqerr://</code>	<code>dspmqerr</code> <code>dspmqerr -s</code>
Queue manager error logs	<code>AMQERR01.LOG</code> <code>AMQERR02.LOG</code> <code>AMQERR03.LOG</code>	<code>mqerr://qmgrs/<QmgrName>/</code> Queue manager error logs also can be downloaded from the IBM MQ Console area of the WebGUI.	<code>dspmqerr -m <QmgrName> <logname></code> Note: If logname is omitted, <code>AMQERR01.LOG</code> is opened.
First failure system technology (FFST), where N is any number 0-9	<code>AMQNNNN.N.FDC</code>	<code>mqerr://</code>	<code>dspmqerr -f AMQNNNN.N.FDC</code>
IBM MQ Console log	<code>messages.log</code> <code>messages_<date>.log</code>	<code>mqtrace://webui/</code>	<code>dspmqerr -w <logname></code> Note: if logname is omitted, IBM MQ Console <code>messages.log</code> is opened.
IBM MQ Console JMS FFSTs	<code>mqjms.log.*</code>	<code>mqtrace://webui/FFDC/</code>	<code>dspmqerr -w <logname></code> Note: if logname is omitted, IBM MQ Console <code>messages.log</code> is opened.

IBM MQ Console Liberty FFSTs	trace*.log	mqtrace://webui/ffdc/	dspmqr -w <logname> Note: if logname is omitted, IBM MQ Console messages.log is opened.
HA Trace	messages_<date> <Many different names>.fmt	mqtrace://	None in mqcli From configuration mode: show file mqtrace://<logname> To list: dir mqtrace://

You can also list all logs of a certain type by using the **dspmqr** command with the **-l** switch with any option that uses the **dspmqr** command.

12.4 Appliance support

The IBM MQ Appliance inherited many of the functions from the DataPower SOA Gateway Appliances. For more information about troubleshooting, see the DataPower reference material. However, the IBM MQ Appliance does not support the old DataPower WebGUI; therefore, many logging tasks are possible through the CLI only.

12.4.1 Appliance error logs

There is an overview of appliance logs in the IBM Redbooks publication *IBM WebSphere DataPower SOA Appliances Part I: Overview and Getting Started*, REDP-4327, which is available at this website:

<http://www.redbooks.ibm.com/redpapers/pdfs/redp4327.pdf>

In this book, key common concepts are introduced. However, on the IBM MQ Appliance, all capabilities must be managed via the CLI unless otherwise stated.

If you want to view the default appliance log, use the **show log** command in the top-level CLI or in configuration mode. This command has the same effect as entering `show file logtemp://default-log` into the configuration mode of the CLI.

Logging to a syslog server can be set up by configuring log targets through the CLI. For more information about the steps for this process, see this website:

http://www.ibm.com/support/knowledgecenter/SS9H2Y_7.1.0/com.ibm.dp.doc/logtargetcommmands.html?lang=en

Creating a log target through the CLI

A log target allows you to view appliance logs from an external server, with which you can externally monitor your appliances. Before configuring a log target on the appliance, you must set up a remote syslog server, which is beyond the scope of this book.

Note: To monitor IBM MQ on the IBM MQ Appliance, you can use activity trace. For more information, see 12.1, “Scope of appliance support” on page 172.

The following information is needed before a log target can be configured:

- ▶ Management interface IP address (as the syslog server sees it).
- ▶ IP address or host name and port for remote syslog server.
- ▶ The type of logs and level of detail you want to send to log target.
- ▶ Unique name for the logs of this type and level from this appliance for the syslog server.
- ▶ Upload-method protocol.

For more information, see the DataPower Gateway Appliances manual, which is available at this website

http://www.ibm.com/support/knowledgecenter/SS9H2Y_7.1.0/com.ibm.dp.doc/logtargetco mmmands.html?lang=en

You can enter CLI configuration mode as a privileged user. From that mode, Example 12-3 shows the CLI commands and the responses from the CLI. The unique identifier that is chosen here is allmq001p.

Example 12-3 Configuring a log target through the CLI

```
M2000(config)# logging target syslog-server
New Log Target configuration
M2000(config logging target syslog-server)# summary "Remote logging to servername"
M2000(config logging target syslog-server)# show
admin-state enabled
summary "Remote logging to servername"
priority normal
soap-version soap11
format xml
timestamp syslog
fixed-format off
size 500 kilobytes
archive-mode rotate
rotate 3
ansi-color off
facility user
rate-limit 100 events/second
connect-timeout 60 seconds
idle-timeout 15 seconds
active-timeout 0 seconds
feedback-detection off
event-detection off
suppression-period 10 seconds
M2000(config logging target syslog-server)# type syslog-tcp
M2000(config logging target syslog-server)# timestamp syslog
M2000(config logging target syslog-server)# local-ident allmq001p
M2000(config logging target syslog-server)# upload-method ftp
M2000(config logging target syslog-server)# remote-address "Remote IP Address"
"Remote Port"
M2000(config logging target syslog-server)# local-address "Local IP Address"
M2000(config logging target syslog-server)# event "all" "debug"
M2000(config logging target syslog-server)# show
admin-state enabled
summary "Remote logging to servername"
type syslog-tcp
priority normal
soap-version soap11
```

```

format text
timestamp syslog
fixed-format off
local-ident allmq001p
size 500 kilobytes
archive-mode rotate
upload-method ftp
rotate 3
ansi-color off
remote-address "Remote IP Address"
remote-port "Remote Port"
local-address mq001p-ssh
facility user
rate-limit 100 events/second
connect-timeout 60 seconds
idle-timeout 15 seconds
active-timeout 0 seconds
feedback-detection off
event-detection off
suppression-period 10 seconds
event all debug
M2000(config logging target syslog-server)# exit

```

% Pending

```

M2000(config)# write memory
Overwrite previously saved configuration? Yes/No [y/n]: y
Configuration saved successfully.
M2000(config)#

```

Example 12-3 on page 177 shows setting up a log target at the finest level of detail, **debug**. In practice, it might be better to decide on wanted logging levels and set up individual log targets for each log type available.

Example 12-4 shows some output that was produced on the syslog server.

Example 12-4 Sample syslog output after the previous log target was successfully set up

```

Jul 28 16:24:32 allmq001p [0x8040006b][system][notice] logging target(syslog-server):
trans(111): Logging started.
Jul 28 16:24:32 allmq001p [0x00360001][mgmt][info] logging target(syslog-server): trans(111):
Pending
Jul 28 16:24:32 allmq001p [0x00330019][mgmt][error] logging target(syslog-server): trans(111):
Operation state transition to up failed
Jul 28 16:24:32 allmq001p [0x810001f0][cli][debug] (admin:default:secure-shell:mq001p-ssh):
exit
Jul 28 16:24:32 allmq001p [0x8040006b][system][notice] logging target(syslog-server):
trans(111): Logging started.
Jul 28 16:24:32 allmq001p [0x00350014][mgmt][notice] logging target(syslog-server): trans(111):
Operational state up
Jul 28 16:24:37 allmq001p [0x80e0067a][ipmi][info] trans(1455): BMC SEL polling cycle starting
Jul 28 16:24:37 allmq001p [0x80e00681][ipmi][debug] trans(1455): BMC SEL: No records to read,
done
Jul 28 16:24:39 allmq001p [0x8100015b][file][notice] trans(5807): Creating file
"config:/temp_00001"

```

```

Jul 28 16:24:39 allmqa001p [0x8100000c][mgmt][notice] trans(5807): Saved current configuration
to 'config:///autoconfig.cfg'
Jul 28 16:24:39 allmqa001p [0x81000040][mgmt][notice] domain(default): trans(5807): Domain
configuration has been saved.
Jul 28 16:24:39 allmqa001p [0x810000d1][cli][notice] trans(385): Configuration saved
successfully.
Jul 28 16:24:39 allmqa001p [0x810001f0][cli][debug] (admin:default:secure-shell:mqa001p-ssh):
write memory

```

After a log target is set up, more work can be done to analyze and present the output.

12.4.2 Copy command

The appliance **copy** command that is found in configuration mode is DataPower's own implementation of several different file transfer protocols. This book focuses on use of the **copy** command for secure copy (SCP). If you experience problems with the **copy** command, the following steps can be used to help diagnose the problem:

1. Ensure that the path and filename are correct.

For files that are copied from the IBM MQ Appliance to an external server, you must use the DataPower URI format. For example, for the first error log of a queue manager that is named GTWY1HAP, `mqerr://qmgrs/GTWY1HAP/AMQERR01.LOG`.

For files that are copied from an external server to the IBM MQ Appliance, you must follow the convention that is shown in Figure 12-1 regarding the copy command, including the correct number of forward slashes. You can enter `help copy` from the configuration mode to see some examples. Figure 12-1 guides the user through use of the **copy** command.

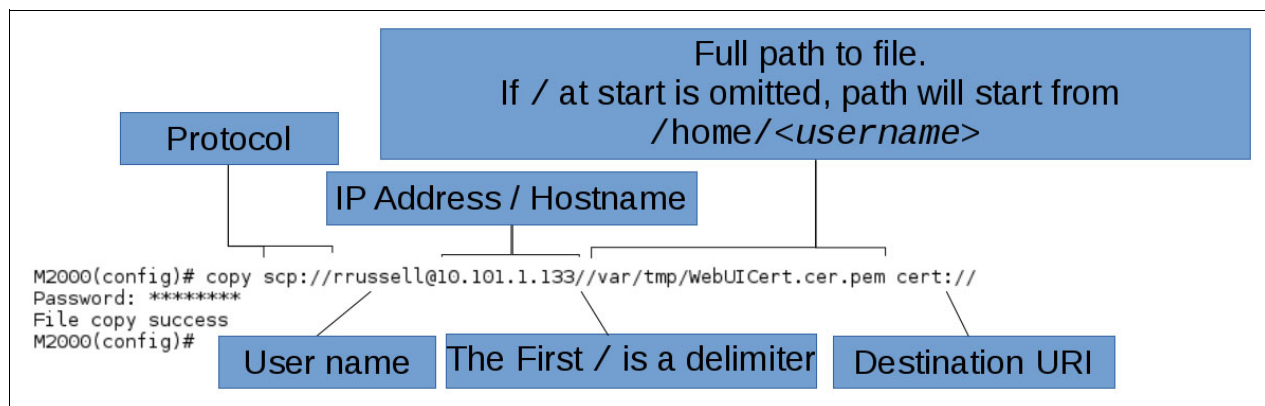


Figure 12-1 Guided use of the copy command for copying a signed certificate to certify the IBM MQ Console

2. Ensure that your IBM MQ Appliance is part of the network and that you can ping the destination server. You can access ping from the DataPower CLI. A successful ping is shown in Example 12-5.

Example 12-5 Successfully pinging a destination SCP server from an IBM MQ Appliance

```

M2000# ping
Usage: ping [-6] [-4] <host>
M2000# ping 10.101.1.133
PING (10.101.1.133): with 56 data bytes
64 bytes from 10.101.1.133: seq=0, ttl=64, rtt=61.0 ms
64 bytes from 10.101.1.133: seq=1, ttl=64, rtt=4.0 ms

```

```
64 bytes from 10.101.1.133: seq=2, ttl=64, rtt=6.0 ms
64 bytes from 10.101.1.133: seq=3, ttl=64, rtt=15.0 ms
64 bytes from 10.101.1.133: seq=4, ttl=64, rtt=12.0 ms
64 bytes from 10.101.1.133: seq=5, ttl=64, rtt=10.0 ms
6 packets transmitted, 6 received, 0% loss, time 6006 ms
M2000#
```

3. For SCP, ensure that the ssh daemon is running on the destination SCP server. In /etc/ssh/sshd_config, ensure that the line PasswordAuthentication yes exists and is not prefixed with a # (indicates that it is commented out).

12.5 Example support scenario

The appliances were running for some time and HA queue manager performance degraded. Company B's IBM MQ administrator does not know why, and contacts IBM Support.

IBM Support ask the administrator to turn on trace and provide them with its output, collect IBM MQ Console diagnostics, and collect the appliance default log. The administrator carries out the steps on both appliances, as shown in Example 12-6.

Example 12-6 Collecting some basic troubleshooting information for IBM Support

```
[rrussell@PK1Y0W0 log]$ ssh mqa001p-ssh
```

```
mqa001p
```

```
Unauthorized access prohibited.
```

```
login: CompanyBAdmin
```

```
Password: *****
```

```
Welcome to IBM MQ Appliance M2000A console configuration.
```

```
Copyright IBM Corporation 1999-2015
```

```
Version: MQ00.8.0.0.3 build 262660mq on Jun 2, 2015 3:28:04 PM
```

```
Serial number: 7800318
```

```
M2000# mqcli
```

```
M2000(mqcli)# strmqtrc -t all -e
```

```
IBM MQ Appliance trace for installation 'MQAppliance' has started.
```

```
M2000(mqcli)# endmqtrc
```

```
IBM MQ Appliance trace for installation 'MQAppliance' has stopped.
```

```
M2000(mqcli)# runmqras -section trace,webui
```

```
Executing command: ps
```

```
Processing MQ files...
```

```
Executing command: server
```

```
Processing MQ files...
```

```
Processing File: wlp_dump.zip
```

```
The zip file can be found at mqdiag://runmqras_150728_140122-mqa001p.zip
```

```
M2000(mqcli)# exit
```

```
M2000# config
```

```
Global configuration mode
```

```
M2000(config)# copy mqdiag://runmqras_150728_140122-mqa001p.zip
```

```
scp://CompanyBAdmin@CompanyBSCPServer//full/path/to/destination/folder/
```

```
Password: *****
```

```
File copy success
```



```
M2000(config)# copy logtemp://default-log
scp://CompanyBAdmin@CompanyBSCPServer//full/path/to/destination/folder/default-log
-mqa001p
Password: *****
File copy success
M2000(config)#
```

Note: The example copy command that is shown in Example 12-6 on page 180 renames the log file at the destination to preserve the appliance name.

After the logs and outputs of runmqras are copied from mqa001p and mqa002p, the administrator can then transfer them to IBM Support, who provides more help. IBM Support might ask for more commands to be run on the appliances, whose output must need be copied away before sending back to IBM Support in a similar way.

If the problem required a fix, IBM Support supplies the fix in the form of a firmware image (with suffix scrypt3). The steps to apply the fix are the same as those to apply a firmware upgrade, as described in Chapter 6, “Appliance administration” on page 55.

Appendixes

This part includes the following appendixes:

- ▶ Appendix A, “IBM MQ cryptographic changes” on page 185
- ▶ Appendix B, “Transcript of IBM MQ Appliance firmware upgrade” on page 195
- ▶ Appendix C, “Transcript of appliance initialization” on page 199
- ▶ Appendix D, “Commands to enable an LDAP authenticated administrator” on page 209



A

IBM MQ cryptographic changes

Queue managers that are running on the IBM MQ Appliance M2000 communicate with applications and other queue managers over an Internet Protocol network. In many cases, encryption of network traffic is required, so the queue manager must be prepared for TLS operations.

This process primarily consists of populating the keystore of the queue manager with trusted CA certificates, trusted self-signed certificates, and one or more personal certificates for the queue manager.

The appliance uses CMS keystores (as do all other distributed queue managers) but uses a different command-line interface to operate on the keystore for each queue manager.

Comparison of commands

The appliance uses different commands that are equivalent to **runmqakm -fips**.

Table A-1 lists the equivalent commands for IBM MQ v7.1 or later (**runmqakm** commands) and the appliance commands.

Table A-1 Certificate command equivalents

Task	IBM MQ command	IBM MQ Appliance command
Create the CMS keystore	runmqakm -fips -keydb -create -db <key.kdb> ...	No equivalent. CMS keystore is built with the queue manager automatically
Add a trusted certificate (usually a CA certificate) to the keystore	runmqakm -fips -cert -add -db <key.kdb> -stashed ...	addcert -m <QmgrName> ...
Create a self-signed certificate and private key in the keystore	runmqakm -fips -cert -create -db <key.kdb> -stashed ...	createcert -m <QmgrName> ...
Create a certificate signing request and private key in the keystore	runmqakm -fips -certreq -create -db <key.kdb> -stashed ...	createcertrequest -m <QmgrName> ...
Delete a certificate from the keystore. Also, delete the matching private key if one exists	runmqakm -fips -cert -delete -db <key.kdb> -stashed ...	deletecert -m <QmgrName> ...
Delete a certificate signing request and the matching private key from the keystore	runmqakm -fips -certreq -delete -db <key.kdb> -stashed ...	deletecertrequest -m <QmgrName> ...
Print the details of a certificate from the keystore	runmqakm -fips -cert -details -db <key.kdb> -stashed ...	detailcert -m <QmgrName> ...
Print the details of a certificate signing request from the keystore	runmqakm -fips -certreq -details -db <key.kdb> -stashed ...	detailcertrequest -m <QmgrName> ...
Extract a certificate from the keystore	runmqakm -fips -cert -extract -db <key.kdb> -stashed ...	No equivalent. When a self-signed certificate is created, the certificate is automatically placed in the mqpubcert: directory.
Back up the keystore	No equivalent. Performed using OS commands	keybackup -m <QmgrName> ...
Restore the keystore	No equivalent. Performed using OS commands	keyrestore -m <QmgrName> ...
List the certificates in a keystore	runmqakm -fips -cert -list -db <key.kdb> -stashed ...	listcert -m <QmgrName> ...
List the unfulfilled certificate signing requests from a keystore	runmqakm -fips -certreq -list -db <key.kdb> -stashed ...	listcertrequest -m <QmgrName> ...
Receive a signed certificate to pair with a private key	runmqakm -fips -cert -receive -db <key.kdb> -stashed ...	receivecert -m <QmgrName> ...
Recreate the certificate request file for a specific key	runmqakm -fips -certreq -recreate -db <key.kdb> -stashed ...	recreatecertrequest -m <QmgrName> ...

For more information about IBM MQ commands, see this website:

http://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.ref.adm.doc/q083800_.htm

For more information about IBM MQ Appliance certificate management commands, see this website:

http://www.ibm.com/support/knowledgecenter/SS5K6E_1.0.0/com.ibm.mqa.doc/reference/certificatecmd/certintro.htm

Examples of using the new keystore commands

IBM MQ uses X.509 certificates for authentication via SSL/TLS. The private keys, personal certificates, and trusted certificates that IBM MQ uses for identity and validation are all stored in Certificate Management Services (CMS) keystores. CMS keystores on an IBM MQ Appliance are managed by using a new set of commands.

The examples in this appendix are based on the examples from *Secure Messaging Scenarios with WebSphere MQ*, SG24-8069, which is available at this website:

<http://www.redbooks.ibm.com/abstracts/sg248069.html>

Each example was rewritten by using the keystore control commands that are available on the IBM MQ Appliance. Where necessary, more commands were shown to demonstrate how to manage files on the appliance. The command to create a keystore was removed because it is not relevant on the appliance. The keystore for each queue manager is automatically created by the appliance version of the **crtmqm** command.

Copying certificate files to an appliance

Before a certificate can be received into or added to a keystore, the certificate file must be copied to the appliance and placed in the `mqpubcert:` directory. Example A-1 shows the root certificate file for Company B (`CompanyBrootCA.pem`) being copied from the support server (`mqsupportsvr.companyb.local`) to the appliance. The file is at `/var/tmp/CompanyBrootCA.pem` on the support server.

Example A-1 Copying a certificate file to an appliance

```
M2000(config)# top
M2000# config
Global configuration mode
M2000(config)# copy
scp://neil@mqsupportsvr.companyb.local//var/tmp/CompanyBrootCA.pem mqpubcert:
Password: *****
File copy success
M2000(config)#
```

Note: The copy command is available in configure terminal mode only.

Adding trusted certificates

Example A-2 shows the command that is used to add CA certificates and self-signed certificates that IBM MQ should trust to the keystore for a queue manager. The CompanyBrootCA.pem file must be placed into the mqpubcert: directory on the appliance.

Example A-2 Adding a trusted certificate to the keystore

```
M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# addcert -m GTWY1HAP -file CompanyBrootCA.pem -label "CompanyB Root
CA"
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqcli)#
```

Creating a certificate signing request

Example A-3 shows creating a private key in the queue manager keystore, and a certificate signing request (CSR) file in the mqpubcert: directory. The CSR must then be sent to a CA, which signs it and returns a certificate. The **createcertrequest** command is preceded by a command in config mode that shows the content of the *mqpubcert:* directory before the certificate request is created. After the certificate request is created, the directory content is listed again to show the file that was created by the command.

Example A-3 Creating certificate signing request

```
M2000(config)# dir mqpubcert:
  File Name                               Last Modified                               Size
  -----
  CompanyBrootCA.pem                      Jul 29, 2015 9:13:04 AM                      1521
  CompanyBIssuer.pem                      Jul 24, 2015 4:26:00 PM                      1948
  15952.8 MB available to mqpubcert:

M2000(config)# top
M2000# mqcli
M2000(mqcli)# createcertrequest -m GTWY1HAP -label ibmwebspheremqgtwy1hap -dn
"CN=GTWY1HAP,OU=/CLUS1P/CLUS2P/,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US"
-sig_alg sha256 -size 2048
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqcli)# top
M2000# co
Global configuration mode
M2000(config)# dir mqpubcert:
  File Name                               Last Modified                               Size
  -----
  CompanyBrootCA.pem                      Jul 29, 2015 9:13:04 AM                      1521
  CompanyBIssuer.pem                      Jul 24, 2015 4:26:00 PM                      1948
  GTWY1HAP_ibmwebspheremqgtwy1hap       Jul 29, 2015 10:45:01 AM                      1041
  15952.8 MB available to mqpubcert:
```

The CSR file is stored in the mqpubcert: directory, and is called <QMGRNAME>_<label>. Given Example A-3 with queue manager name GTWY1HA, and label ibmwebspheremqgtwy1hap, the resulting certificate request file is mqcert:///GTWY1HAP_ibmwebspheremqgtwy1hap, as shown in the directory listing.

Copying CSR to an external server

Example A-4 shows how to copy the certificate signing request file to an external server. The file can then be sent to a CA to be signed.

Example A-4 Copying certificate request file to an external server

```
M2000(mqcli)# top
M2000# co
Global configuration mode
M2000(config)# copy mpubcert:///GTWY1HAP_ibmwebspheremqgtwy1hap
scp://neil@mqsupportsvr.companyb.local//var/tmp/gtwy1hap.certreq
Password: *****
File copy success
M2000(config)#
```

It is also possible to display the content of the certificate request file and paste the data into a file on another system or paste the text directly into the CA signing form. Displaying the content of a file is shown in Example A-5.

Example A-5 Displaying the content of a file, allowing copy and paste operations

```
M2000(mqcli)# top
M2000# co
Global configuration mode
M2000(config)# show file mpubcert:///GTWY1HAP_ibmwebspheremqgtwy1hap

-----BEGIN NEW CERTIFICATE REQUEST-----
MIICwzCCAAsCAQAwfjELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAk5DMRAwDgYDVQQH
EwdSYWx1aWdoMREwDwYDVQQKEwhDb21wYW55QjEQMA4GA1UECxMHU0EtVzUyNTEY
MBYGA1UECxMPLONMVMxUC9DTFVTM1AvMREwDwYDVQQDEwhHVFDZMUhBUDCCASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALXJtnCXfoDH95cHc+yCbwp+IJCf
etHpgdqUBYc+bvUE9Hz4CYQeqGF/rM4rBMqTMMIdD9hPXySNOUEhtWsZAeaoZLf/
1PwsqTh8Mox6XbZXKnLhWR4vfCjQYNSTTNM3rkcWSPoals/XTA48qls4x116g5Sy
1Ck5s3cYPYIe6LwJmISs910dUAm8hVksxMwAZcLxJPYgnxF0x0F7BiAXfx197WyW
01j0YRcbJzydzHsYffx122M0t1prJuf6Hf2k1/51NNIa5/S60bWnG7cY1zTGDEbn
ft0kwBZwjYudxSM/M9EfozCTEq6wgCYWp7Ky9ouw101bLcGmZrfWr3UQoC0CAwEA
AaAAMA0GCSqGSIB3DQEBChUAA4IBAQB54jfWM1BX0cT8r9LD/20H1pMXa140A2IN
sxHJxvrzPMcnu12B/2Fb1sQixRf8ELWUosAjn0ZJn9ah/z2rShmk08RLPHU3gLuW
x1Rff18+L+vXA6xA2eFE1Wepu2U1007Jd1WN6nEiPMIwH9/ZKqBqIGFK5s5SoXnc
2VN2kKchmUL9Scnuw5NfUCnwI6og/9k57eV3igDhSjZmBe1Ki/LmnNc71u8Fo/7E
ORIjzYtghzBJQ9Wr1j/7zaG70A0YJzgvbsQhq5sm9xGpiqY8YeB5ovz19f9P9eei
/Os1bZ3TmGD95xnbkFU15UG3FUOP1GUWwftPfXdDwfZyd9YzEqQi
-----END NEW CERTIFICATE REQUEST-----

M2000(config)#
```

Listing certificate signing requests

Example A-6 shows how to list the certificate signing requests that were created in a keystore, but where the certificate was not yet received. After the certificate is received, the CSR information is no longer available.

Example A-6 Listing certificate signing requests

```
M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# listcertrequest -m GTWY1HAP
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Certificates requests found
      ibmwebspheremqgtwy1hap
M2000(mqcli)#
```

Showing details of a certificate signing request

Example A-7 shows how to display the details of a certificate signing request. Showing these details is possible only before the certificate is received. After the certificate is received, only the certificate details can be shown, not the certificate request details.

Example A-7 Showing certificate request details

```
M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# detailcertrequest -m GTWY1HAP -label ibmwebspheremqgtwy1hap
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Label : ibmwebspheremqgtwy1hap
Key Size : 2048
Subject :
CN=GTWY1HAP,OU=/CLUS1P/CLUS2P/,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US
Public Key
  30 82 01 22 30 0D 06 09 2A 86 48 86 F7 0D 01 01
  01 05 00 03 82 01 0F 00 30 82 01 0A 02 82 01 01
  00 B5 C9 B6 70 97 7E 80 C7 F7 97 07 73 EC 82 6F
  0A 7E 20 90 9F 7A D1 E9 81 DA 94 05 87 3E 6E F5
  04 F4 7C F8 09 84 1E A8 61 7F AC CE 2B 04 CA 93
  30 C2 1D 0F D8 4F 5F 24 8D 39 41 21 B5 6B 19 01
  E6 A8 64 B7 FF 94 FC 2C A9 38 7C 32 8C 7A 5D B6
  57 2A 72 E1 59 1E 2F 7C 28 D0 60 D4 93 4C D3 37
  AE 47 16 48 FA 1A 96 CF D7 4C 0E 3C AA 5B 38 C6
  5D 7A 83 94 B2 D4 29 39 B3 77 18 3D 82 1E E8 BC
  09 98 84 AC F7 5D 1D 50 09 BC 85 59 2C C4 CC 00
  65 C2 F1 24 F6 20 9F 11 74 C4 E1 7B 06 20 17 7F
  19 7D ED 6C 96 3A 58 CE 61 17 1B 27 3C 9D CC 7B
  18 7D FC 65 DB 63 34 B7 5A 6B 26 E7 FA 1D FD A4
  D7 FE 75 34 D2 1A E7 F4 BA 39 B5 A7 1B B7 18 D7
  34 C6 0C 46 E7 7E D3 A4 C0 16 70 8D 8B 9D C5 23
  3F 33 D1 1F A3 30 93 12 AE B0 80 26 16 A7 B2 B2
  F6 8B B0 D7 4D 5B 2D C1 A6 66 B7 D6 AF 75 10 A0
  2D 02 03 01 00 01
Public Key Type : RSA (1.2.840.113549.1.1.1)
Fingerprint :
0589727dfa5f50701569e6c35f56b2d6
```

```

1c6e03a0
Attributes
Signature Algorithm : SHA256WithRSASignature (1.2.840.113549.1.1.11)
Value
  6C E2 37 D6 32 50 57 39 C4 FC AF D2 C3 FF 63 87
  D6 93 17 6B 5E 0E 03 62 0D B3 11 C9 C6 FA F3 3C
  C0 A7 BB 5D 81 FF 61 5B 96 C4 22 C5 17 FC 10 B5
  94 A2 C0 23 9C E6 49 9F D6 A1 FF 3D AB 4A 19 A4
  3B C4 4B 3C 75 37 80 BB B0 C6 54 5F 7E 5F 3E 2F
  EB D7 03 AC 40 D9 E1 44 D5 67 A9 BB 65 35 3B 4E
  C9 76 55 8D EA 71 22 3C C2 30 1F DF D9 2A A0 6A
  20 61 4A E6 CE 52 A1 79 DC D9 53 76 90 A7 21 99
  42 FD 49 C9 EE C3 93 5F 50 29 F0 23 AA 20 FF D9
  39 ED E5 77 8A 00 E1 4A 36 66 05 ED 4A 8B F2 E6
  9C D7 3B D6 EF 05 A3 FE C4 39 12 23 CD 8B 60 87
  30 49 43 D5 AB D6 3F FB CD A1 BB D0 0D 18 27 38
  2F 6E C4 21 AB 9B 26 F7 11 A9 8A A6 3C 61 E0 79
  A2 FC F5 F5 FF 4F F5 E7 A2 FC EB 25 6D 9D D3 98
  60 FD E7 19 DB 90 55 25 E5 41 B7 15 4D 0F 94 65
  30 59 FB 4F 7D 77 43 59 F6 72 77 D6 33 12 A4 22
M2000(mqccli)#

```

Receiving signed certificate

Example A-8 shows the command to receive a certificate that is signed by a CA into the keystore and match it with the private key that is in place. The private key is created when the CSR is originally created. The certificate file must be copied to the appliance `mqpubcert:` directory. Follow the command format that is shown in Example A-1 on page 187 to copy the certificate to the appliance.

Example A-8 Receiving the signed certificate

```

M2000(mqccli)# top
M2000# mqccli
M2000(mqccli)# receivecert -m GTWY1HAP -file gtwy1hap.cer
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqccli)#

```

Deleting a certificate signing request

By using the command that is shown in Example A-9, the certificate signing request information and the matching private key are deleted; therefore, use this command carefully. It might be necessary to delete a CSR if a problem is found with it before the certificate is received from the CA. A new CSR must be created that has the same label; therefore, the previous CSR must be deleted first.

Example A-9 Deleting a certificate request

```

M2000(config)# top
M2000# mqccli
M2000(mqccli)# deletecertrequest -m GTWY1HAP -label ibmwebspheremqgtwy1hap
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqccli)#

```

Creating a self-signed certificate

In some cases, self-signed certificates can be used instead of certificates signed by a certificate authority. A self-signed certificate is a certificate that is signed by using its own private key. The command that is shown in Example A-10 creates a self-signed certificate.

Example A-10 Creating a self-signed certificate for a queue manager

```
M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# createcert -m GTWY1HAP -label ibmwebsphermqgtwy1hap -dn
"CN=GTWY1HAP,OU=/CLUS1P/CLUS2P/,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US"
-sig_alg sha256 -size 2048
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Certificate has been extracted to
'mqpubcert://GTWY1HAP_ibmwebsphermqgtwy1hap'.
M2000(mqcli)#
```

Listing the certificates for a queue manager

Example A-11 shows the command that is used list the certificates in a keystore.

Example A-11 List the certificates for a queue manager

```
M2000(mqcli)# top
M2000# mqcli
M2000(mqcli)# listcert -m GTWY1HAP
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Certificates found
* default, - personal, ! trusted, # secret key
!      "CompanyB Root CA"
      ibmwebsphermqgtwy1hap
M2000(mqcli)#
```

Showing the details of a certificate

The command that is shown in Example A-12 displays the details of a certificate.

Example A-12 Show details of a certificate

```
M2000(config)# top
M2000# mqcli
M2000(mqcli)# detailcert -m GTWY1HAP -label ibmwebsphermqgtwy1hap
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
Label : ibmwebsphermqgtwy1hap
Key Size : 2048
Version : X509 V3
Serial : 03
Issuer : C=US,ST=NC,L=Raleigh,O=CompanyB,OU=SA-W525,CN=DevOps Untrusted CA
Subject :
CN=GTWY1HAP,OU=/CLUS1P/CLUS2P/,OU=SA-W525,O=CompanyB,L=Raleigh,ST=NC,C=US
Not Before : July 29, 2015 3:44:26 PM EDT
Not After : July 28, 2016 3:44:26 PM EDT
Public Key
30 82 01 22 30 0D 06 09 2A 86 48 86 F7 0D 01 01
```

```

01 05 00 03 82 01 0F 00 30 82 01 0A 02 82 01 01
00 C2 B7 BD E2 91 ED 48 5D 24 D0 2E A6 01 51 F6
EF 6E AC D4 4D E3 24 86 0F EA 37 72 1B 7B 50 3D
43 7E 17 4C 35 D4 DA EA 52 F5 06 88 19 FC 57 0F
BA C2 AF 53 21 D4 10 3F 67 8E DA FB 92 11 F5 26
EF DD 90 CB B5 00 D4 4D 48 C3 24 B8 7D B4 8A 38
2A 60 B3 56 BF 0C 88 F4 A6 DC FC BC E6 5B 66 26
99 DD E8 63 2D F0 2F 16 3C D6 E9 8D A7 BF 99 9C
16 98 69 12 20 06 BF BC E7 B3 90 1E 6E CB 10 28
AF 97 1C 56 E7 2B 01 32 E5 C9 85 93 0F E0 04 A8
09 13 84 9B 67 EB C9 AD 38 D6 3B 48 74 82 DD D2
EF 63 F1 DE 9F 92 A3 27 6B 0A 35 F3 E1 33 2F B6
58 F8 33 03 52 A7 70 D3 88 2C 56 78 3F DF 60 28
90 10 6F 5C 29 3B 4F 65 02 A8 C5 B7 06 57 F9 00
08 03 6C 6A D1 15 C2 42 E3 66 7A 95 B6 1C B0 B3
FA 97 22 D9 6E 51 8D 6E B1 23 83 59 D1 C0 FB 33
13 28 3A 81 8B 61 38 18 01 97 59 5D 4E 21 E5 90
A5 02 03 01 00 01
Public Key Type : RSA (1.2.840.113549.1.1.1)
Fingerprint : SHA1 :
    0E 30 1E DC 3D 9D C7 A6 D8 27 AE 9B CB 69 B1 30
    5B 08 AD F3
Fingerprint : MD5 :
    10 97 68 26 8B B4 29 A1 6D 8C 11 5A 06 F5 E0 5B
Fingerprint : SHA256 :
    DD DF F3 3B 60 B0 18 D1 ED CF 90 8E D0 AF 71 61
    DC 05 D7 B8 83 79 5B 82 09 CC 6F FE 31 A8 FA 88
Extensions
    basicConstraints
        ca = false
        pathLen = 140736185198800
        critical
Signature Algorithm : SHA256WithRSASignature (1.2.840.113549.1.1.11)
Value
    95 F0 49 2F 6C C8 28 38 59 95 CD 1E 11 41 F5 A3
    38 62 7F 35 2A 8E C4 C4 18 E3 58 7E 2D 0D 14 01
    9A 99 B1 23 1A 47 CE 78 53 F8 91 59 E5 6A 23 D0
    3A CB CA 75 72 65 B5 77 D9 19 EA BF 69 B2 FA C1
    C2 B9 FF 58 8C 45 07 4F 0F B8 0C 3E 27 93 18 48
    AB 3E D1 DA 91 14 73 1E 62 C8 28 D8 0A 6B 43 88
    8A 53 FF 45 CA E9 5E 0F 25 92 46 46 D7 41 97 73
    5E 54 30 25 B1 39 00 07 3B BD 92 0F 25 7B 9A 2B
    E1 8D 54 9A 1B 3D 6C DE 00 FA A8 23 23 A3 17 B3
    D6 BE B0 7B EA 35 64 94 38 9E 69 D0 0C 09 89 CD
    44 41 DB DA 64 A8 A9 E1 22 9F 7B 25 5D 9B 26 D9
    BE EB 63 75 11 5D 27 12 F5 D9 CB 52 4B FB 6B 48
    63 9D EA B3 34 C5 71 F8 F6 F0 C7 8B 53 68 C9 AD
    B1 0F 50 CD 9A B0 EF 47 C3 F5 4D 13 6C 40 42 E4
    16 A1 4F E3 8A AF BB 15 6A D3 3C CD 24 21 D9 36
    ED F1 84 5F D7 BD 9C 14 17 50 14 4D B0 9F 9E 92
Trust Status : Enabled
M2000(mqccli)#

```

Deleting a certificate

The command that is shown in Example A-13 deletes a trusted or personal certificate. If a private key is associated with the certificate, it is deleted. The command does not prompt for confirmation; therefore, use this command carefully.

Example A-13 Deleting a certificate

```
M2000(config)# top
M2000# mqcli
M2000(mqcli)# deletecert -m GTWY1HAP -label ibmwebspheremqgtwy1hap
5724-H72 (C) Copyright IBM Corp. 1994, 2014.
M2000(mqcli)#
```



B

Transcript of IBM MQ Appliance firmware upgrade

This appendix features a typical IBM MQ Appliance firmware upgrade transcript, as shown in Example B-1.

Example B-1 IBM MQ Appliance upgrade transcript

```
M2000(config-flash)# boot image accept-license rel-mq.script3
Invoking dynamic loader
.....
.....
Firmware upgrade successful
Device is rebooting now.
Version 2.15.1236. Copyright (C) 2012 American Megatrends, Inc.
**** DP-9006 BIOS V0.16 (10/27/2014) ****
Press <TAB> to enter setup.                                Press <L> to
boot to LAN.                                                Press <B> for BBS MENU
Copyright (c) 1997-2014 Emulex. All rights reserved.
Press <Alt E> or <Ctrl E> to enter Emulex BIOS configuration utility. Press
<s> to skip Emulex BIOS
Emulex LightPulse FC x86 BIOS, Version 10.2.261.0           Emulex BIOS is
Disabled on Adapter 01                                     Emulex BIOS is Disabled on
Adapter 02                                                 Emulex FC BIOS is not installed!!!

Version 2.15.1236. Copyright (C) 2012 American Megatrends, Inc.
**** DP-9006 BIOS V0.16 (10/27/2014) ****                 Press <TAB> to
enter setup.                                                Press <L> to boot to LAN.
Press<B>forBBSMENU
Version 2.15.1236. Copyright (C) 2012 American Megatrends, Inc.
**** DP-9006 BIOS V0.16 (10/27/2014) ****                 Press <TAB> to
enter setup.                                                Press <L> to boot to LAN.
Press<B>forBBSMENU
Progress:
```

```

DATAPOWER: Configuring loopback interface
DATAPOWER: Check for uEFI configuration update
DATAPOWER: Reading LUKS subroutines
DATAPOWER: Starting tcscd..done
insmod: can't insert '/lib/modules/2.6.32-431.29.2.dp_800.15.x86_64/aesni-intel.ko': unknown
symbol in module, or unknown parameter
DATAPOWER: Finding encrypted flash./dev/sda2
done
DATAPOWER: Unlocking encrypted flash from primary key
DATAPOWER: Unlocked encrypted flash from primary key
DATAPOWER: Stopping tcscd
DATAPOWER: Checking encrypted flash filesystem
e2fsck 1.41.12 (17-May-2010)
DPLUKS: recovering journal
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information

DPLUKS: ***** FILE SYSTEM WAS MODIFIED *****

    64225 inodes used (6.57%)
      37 non-contiguous files (0.1%)
      58 non-contiguous directories (0.1%)
        # of inodes with ind/dind/tind blocks: 5536/124/0
1126921 blocks used (28.87%)
    0 bad blocks
    1 large file

50386 regular files
 9434 directories
   10 character device files
    2 block device files
    0 fifos
11100 links
 4382 symbolic links (3499 fast symbolic links)
    2 sockets
-----
 75316 files
DATAPOWER: Mounted encrypted flash filesystem

DPOS boot - press <ESC> within 7 seconds for boot options.....
DATAPOWER: Loading system from encrypted flash filesystem
UNABLE TO GATHER EFI DATA
DATAPOWER: Unmounting encrypted flash filesystem
DATAPOWER: Starting system
DATAPOWER: Starting udevd
DATAPOWER: Triggering udevd
DATAPOWER: Settling udevd
DATAPOWER: Getting partnum
insmod: can't insert '/lib/modules/2.6.32-431.29.2.dp_800.15.x86_64/aesni-intel.ko': unknown
symbol in module, or unknown parameter
DATAPOWER: Finding flash device

```


DATAPOWER: Waiting to find encrypted flash
DATAPOWER: Found encrypted flash
DATAPOWER: Creating ramdisk 1
DATAPOWER: Enabling loopback interface
DATAPOWER: Enabling LUKS-encrypted flash device
DATAPOWER: Starting tcscd
DATAPOWER: Waiting for tcscd to startup
DATAPOWER: Unlocking LUKS from TPM key
DATAPOWER: Unlocked LUKS from TPM key
DATAPOWER: Removed unused upgrade key
DATAPOWER: Checking flash filesystems
DATAPOWER: Stopping tcscd
DATAPOWER: TPM Firmware version 3.17, ROM CRC 9ae5
DATAPOWER: Stopping udev before executing supervisor
DATAPOWER: Executing supervisor process
(unknown)
Unauthorized access prohibited.
login:



Transcript of appliance initialization

This appendix features a transcript of appliance initialization, as shown in Example C-1.

Example C-1 Transcript of appliance initialization

login: admin

Password: *****

Capacity licensing: If you are unsure you should answer 'no' to the following.
If configured incorrectly you will need to reinitialise or contact IBM support.
Is this system licensed for Enterprise Capacity use (M2000A): yes

Take note of the new admin password. If you lose or forget the admin password,
security best practice dictates that you return the appliance to IBM to
reset this password.

After the appliance is returned to you, you must perform an initial
firmware setup as described in the Installation Guide. Therefore,
none of your configuration data is on the appliance.

However, when another user account can log in and has the appropriate access
permission, that user can reset the password for the admin account.

Please enter new password: *****

Please re-enter new password to confirm: *****

Do you want to run the Install Wizard? Yes/No [y/n]:y

Welcome to IBM MQ Appliance M2000A console configuration.

Copyright IBM Corporation 1999-2015

Version: MQ00.8.0.0.3 build 262660mq on Jun 2, 2015 3:28:04 PM
Serial number: 7800318

Global configuration mode

The Installation Wizard helps to perform the following tasks:

- (1) Configure network interfaces
- (2) Configure network services (such as DNS)
- (3) Configure a unique system identifier
- (4) Configure remote management access
- (5) Configure a user account that can reset passwords
- (6) Review and save the configuration

Step 1 - Do you want to configure network interfaces? [y]:y

To perform these tasks, you will need the following information:

- (1) The interfaces that are connected
- (2) Whether to use DHCP or a static IP address and subnet mask
- (3) The IP address of the default gateway

Do you have this information? [y]:y

Do you want to configure the eth10 interface? [y]:n

Do you want to configure the eth11 interface? [y]:n

Do you want to configure the eth12 interface? [y]:n

Do you want to configure the eth13 interface? [y]:n

Do you want to configure the eth14 interface? [y]:n

Do you want to configure the eth15 interface? [y]:n

Do you want to configure the eth16 interface? [y]:n

Do you want to configure the eth17 interface? [y]:n

Do you want to configure the eth20 interface? [y]:n

Do you want to configure the eth21 interface? [y]:n

Do you want to configure the mgt0 interface? [y]:y

Modify Ethernet Interface configuration

Do you want to enable DHCP? [y]:n

Enter the IPv4 address for the interface in CIDR notation: 9.27.215.68/25

Enter the IPv4 address for the default gateway []:9.27.215.1

Do you want to configure the mgt1 interface? [y]:n

Step 2 - Do you want to configure network services? [y]:y

Do you want to configure DNS? [y]:n

A system identifier is recommended, as it is required to configure an HA Group.

Step 3 - Do you want to define a unique system identifier for the appliance? [y]:y

Enter a unique system identifier: mqa001p

Modify System Settings configuration

Step 4 - Do you want to configure remote management access? [y]:y

These configurations require the IP address of the local interface that manages the appliance.

Do you have this information? [y]:y

Do you want to enable SSH? [y]:y

Enter the local IP address [0 for all]:

Enter the port number [22]:

% Pending

SSH service listener enabled

Do you want to enable WebGUI access [y]:y

Enter the local IP address [0 for all]:

Enter the port number [9090]:

Modify Web Management Service configuration

Attention: If the password for the admin account is lost or forgotten, security best practice dictates that you return the appliance to IBM Support to reset the password. However, if another user account can log in and if that account has the appropriate access permission, that user can reset the password for the admin account.

Note: If you specify an existing user account, you will change the password for this account.

Step 5 - Do you want to configure a user account that can reset passwords? [y]: y

Enter the name of the user account that can reset passwords [password-reset-user]: adminbackup

New User configuration

```
Enter new password: *****
Enter new password: *****
Enter new password: *****
Enter new password: ****
Enter new password: ***
Enter new password: **
Enter new password: *
Enter new password: *****
```

```
Re-enter new password: *****
```

```
Cleared RBM cache
```

```
Step 6 - Do you want to review the current configuration? [y]:y
```

```
ethernet: eth10 [up]
```

```
-----
```

```
admin-state enabled
ip-config-mode static
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto
hardware-offload on
flow-control auto
```

```
ethernet: eth11 [up]
```

```
-----
```

```
admin-state enabled
ip-config-mode static
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto
hardware-offload on
flow-control auto
```

```
ethernet: eth12 [up]
```

```
-----  
admin-state enabled  
ip-config-mode static  
ipv6-dadtransmits 1  
ipv6-nd-retransmit-timer 1000 Milliseconds  
standby-enable off  
standby-group 1  
standby-preempt off  
standby-priority 100  
standby-authentication 0x5841333500000000  
standby-hello-timer 3 Seconds  
standby-hold-timer 10 Seconds  
link-aggregation-mode off  
mtu 1500 Bytes  
mode Auto  
hardware-offload on  
flow-control auto
```

ethernet: eth13 [up]

```
-----  
admin-state enabled  
ip-config-mode static  
ipv6-dadtransmits 1  
ipv6-nd-retransmit-timer 1000 Milliseconds  
standby-enable off  
standby-group 1  
standby-preempt off  
standby-priority 100  
standby-authentication 0x5841333500000000  
standby-hello-timer 3 Seconds  
standby-hold-timer 10 Seconds  
link-aggregation-mode off  
mtu 1500 Bytes  
mode Auto  
hardware-offload on  
flow-control auto
```

ethernet: eth14 [up]

```
-----  
admin-state enabled  
ip-config-mode static  
ipv6-dadtransmits 1  
ipv6-nd-retransmit-timer 1000 Milliseconds  
standby-enable off  
standby-group 1  
standby-preempt off  
standby-priority 100  
standby-authentication 0x5841333500000000  
standby-hello-timer 3 Seconds  
standby-hold-timer 10 Seconds  
link-aggregation-mode off  
mtu 1500 Bytes  
mode Auto  
hardware-offload on  
flow-control auto
```

ethernet: eth15 [up]

admin-state enabled
ip-config-mode static
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto
hardware-offload on
flow-control auto

ethernet: eth16 [up]

admin-state enabled
ip-config-mode static
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto
hardware-offload on
flow-control auto

ethernet: eth17 [up]

admin-state enabled
ip-config-mode static
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto


```
hardware-offload on
flow-control auto
```

```
ethernet: eth20 [up]
```

```
-----
admin-state enabled
ip-config-mode static
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto
hardware-offload on
flow-control auto
```

```
ethernet: eth21 [up]
```

```
-----
admin-state enabled
ip-config-mode static
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto
hardware-offload on
flow-control auto
```

```
ethernet: mgt0 [up] (modified)
```

```
-----
admin-state enabled
ip-config-mode static
ip-address 9.27.215.68/25
ipv4-default-gateway 9.27.215.1
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
```

```
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto
hardware-offload on
flow-control auto
```

```
ethernet: mgt1 [up]
```

```
-----
```

```
admin-state enabled
ip-config-mode static
ipv6-dadtransmits 1
ipv6-nd-retransmit-timer 1000 Milliseconds
standby-enable off
standby-group 1
standby-preempt off
standby-priority 100
standby-authentication 0x5841333500000000
standby-hello-timer 3 Seconds
standby-hold-timer 10 Seconds
link-aggregation-mode off
mtu 1500 Bytes
mode Auto
hardware-offload on
flow-control auto
```

```
ssh [up] (modified)
```

```
---
```

```
admin-state enabled
ip-address 0.0.0.0
port 22
acl ssh [up]
```

```
web-mgmt [up] (modified)
```

```
-----
```

```
admin-state enabled
ip-address 0.0.0.0
port 9090
save-config-overwrite on
idle-timeout 600 Seconds
acl web-mgmt [up]
```

```
Do you want to save the current configuration? [y]:y
```

```
Overwrite previously saved configuration? Yes/No [y/n]: y
```

```
Configuration saved successfully.
```

```
You have completed the Installation Wizard.
```

```
You must read and agree to the terms of the license agreement using the WebGUI.
```

If you did not configure the Web Management Interface, you must do it now with the following command:

```
configure terminal;web-mgmt;admin-state enabled;local-address 0 9090;exit
```

```
M2000# exit
```

Goodbye.

mqa001p

Unauthorized access prohibited.

login:

Commands to enable an LDAP authenticated administrator

This appendix shows commands to enable an LDAP authenticated administrator, as shown in Example D-1.

Example D-1 Commands to enable an LDAP authenticated administrator

```
SET AUTHREC -  
  PROFILE('**') -  
  OBJTYPE(AUTHINFO) -  
  GROUP('mqadmin') -  
  AUTHADD(ALL)  
SET AUTHREC -  
  PROFILE('**') -  
  OBJTYPE(CHANNEL) -  
  GROUP('mqadmin') -  
  AUTHADD(ALL)  
SET AUTHREC -  
  PROFILE('**') -  
  OBJTYPE(CLNTCONN) -  
  GROUP('mqadmin') -  
  AUTHADD(ALL)  
SET AUTHREC -  
  PROFILE('**') -  
  OBJTYPE(COMMINFO) -  
  GROUP('mqadmin') -  
  AUTHADD(ALL)  
SET AUTHREC -  
  PROFILE('**') -  
  OBJTYPE(LISTENER) -  
  GROUP('mqadmin') -  
  AUTHADD(ALL)  
SET AUTHREC -  
  PROFILE('**') -  
  OBJTYPE(NAMELIST) -
```

```

GROUP('mqadmin') -
AUTHADD(ALL)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(PROCESS) -
GROUP('mqadmin') -
AUTHADD(ALL)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(QUEUE) -
GROUP('mqadmin') -
AUTHADD(ALL)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(QMGR) -
GROUP('mqadmin') -
AUTHADD(ALL)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(RQMNAME) -
GROUP('mqadmin') -
AUTHADD(ALL)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(SERVICE) -
GROUP('mqadmin') -
AUTHADD(ALL)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(TOPIC) -
GROUP('mqadmin') -
AUTHADD(ALL)

SET AUTHREC -
PROFILE('**') -
OBJTYPE(AUTHINFO) -
GROUP('mqadmin') -
AUTHADD(CRT)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(CHANNEL) -
GROUP('mqadmin') -
AUTHADD(CRT)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(CLNTCONN) -
GROUP('mqadmin') -
AUTHADD(CRT)
SET AUTHREC -
PROFILE('**') -
OBJTYPE(COMMINFO) -
GROUP('mqadmin') -
AUTHADD(CRT)
SET AUTHREC -
PROFILE('**') -

```

```

        OBJTYPE(LISTENER) -
        GROUP('mqadmin') -
        AUTHADD(CRT)
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(NAMELIST) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(PROCESS) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(QUEUE) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(SERVICE) -
    GROUP('mqadmin') -
    AUTHADD(CRT)
SET AUTHREC -
    PROFILE('**') -
    OBJTYPE(TOPIC) -
    GROUP('mqadmin') -
    AUTHADD(CRT)

REFRESH SECURITY(*) TYPE(AUTHSERV)

END

```

Related publications

For more information about the IBM MQ Appliance, see the following resources:

- ▶ IBM DataPower Gateway Knowledge Center:

http://www.ibm.com/support/knowledgecenter/SS9H2Y_7.1.0/com.ibm.dp.doc/welcome.html

- ▶ DataPower XML Integration Appliance XI50DP Command Reference:

ftp://ftp.software.ibm.com/software/integration/datapower/library/prod_docs/4Q2008/XI-3.7.2-CommandReference.pdf

For more information about earlier versions of IBM MQ and IBM WebSphere MQ, see the following resources:

- ▶ WebSphere MQ V6 Fundamentals, SG24-7128:

<http://www.redbooks.ibm.com/abstracts/sg247128.html>

- ▶ WebSphere MQ V7.0 Features and Enhancements, SG24-7583:

<http://www.redbooks.ibm.com/abstracts/sg247583.html>

- ▶ IBM MQ V8.0 Feature and Enhancements, SG24-8218:

<http://www.redbooks.ibm.com/abstracts/sg248218.html>

- ▶ Secure Messaging Scenarios with WebSphere MQ, SG24-8069:

<http://www.redbooks.ibm.com/abstracts/sg248069.html>



SG24-8283-00

ISBN 0738441112

Printed in U.S.A.

Get connected

