Configuring the IBM Integration Bus web user interface with HTTPS

Shenfu (Mike) Fan
Doug Drake

January 07, 2015

The IBM Integration Bus web user interface (hereafter called the web UI) enables web users to access broker resources through an HTTP or HTTPS client, and gives broker administrators an alternative to IBM Integration Bus Explorer and MQSC commands for administering broker resources. This tutorial shows you how to configure the web UI with HTTPS on IBM Integration Bus V9.

Web user interface

The developerWorks tutorial Configuring role-based security with the IBM® Integration Bus V9 web UI describes basic configuration of role-based security for the web UI using HTTP. This tutorial shows you how to do advanced configuration of the web UI with HTTPS on IBM Integration Bus V9. The tutorial uses an integration node (broker) named IIB9BRK1, and two integration servers (execution groups) named IS1 and IS2. The integration node was created on Microsoft® Windows® 7.

Setting up public key infrastructure

Because SSL is required for HTTPS, you need to set up a public key infrastructure (PKI) for the integration node. Several tools supplied with IBM Integration Bus can help you set up a PKI, including a command-line tool, a keytool, and a graphical tool (ikeyman). The command line tool (runmqckm), from the Global Secure Toolkit (GSKit) supplied with WebSphere MQ, is used in this article to create and populate keystores and truststores.

Create a keystore

Run the following command to create the keystore IIB9BRK1keystore.jks under the folder C:\BrokerSSL\IIB9BRK1:

```
runmqckm -keydb -create -db C:\BrokerSSL\IIB9BRK1\IIB9BRK1keystore.jks -pw p@ssw0rd
        -type jks
```

Create a truststore

Run the following command to create the truststore IIB9BRK1truststore.jks under the folder C:\BrokerSSL\IIB9BRK1:

```
runmqckm -keydb -create -db C:\BrokerSSL\IIB9BRK1\IIB9BRK1truststore.jks -pw p@ssw0rd
        -type jks
```

Create a self-signed certificate

Run the following command to create the self-signed certificate IIB9BRK1Cert.arm with the label IIB9BRK1Cert for test purposes. In production, you should request a certificate from a certificate authority.

```
runmqckm -cert -create -db C:\BrokerSSL\IIB9BRK1\IIB9BRK1keystore.jks -pw p@ssw0rd
        -label IIB9BRK1Cert -dn "CN=IIB9BRK1.server,O=IBM,OU=ESB,L=Charlotte,C=US"
```

Extract a certificate

Run the following command to extract a self-signed certificate as a signer certificate to import into a truststore:

```
runmqckm -cert -extract -db C:\BrokerSSL\IIB9BRK1\IIB9BRK1keystore.jks -pw p@ssw0rd
        -label IIB9BRK1Cert -target C:\BrokerSSL\IIB9BRK1/IIB9BRK1Cert.arm -format ascii
```

Add the extracted certificate to the truststore

Run the following command to add the extracted certificate as a signer certificate to the truststore:

```
runmqckm -cert -add -db C:\BrokerSSL\IIB9BRK1\IIB9BRK1truststore.jks -pw p@ssw0rd
        -label IIB9BRK1Cert -file C:\BrokerSSL\IIB9BRK1/IIB9BRK1Cert.arm -format ascii
```

Use the following command to view the details of the certificate:

```
runmqckm -cert -details -db C:\BrokerSSL\IIB9BRK1\IIB9BRK1truststore.jks -pw p@ssw0rd
        -label IIB9BRK1Cert
```

Configuring the PKI at the integration node level

To configure the PKI at the integration node level, define the integration node registry properties to identify the location, name, and password of the keystore and truststore files. Make sure the integration node is running -- if it is not, use the following command to start it:

```
mqsistart IIB9BRK1
```

To set the keystore and the truststore properties, run the following commands:

```
mqsichangeproperties IIB9BRK1 -o BrokerRegistry -n brokerKeystoreFile
                        -v C:\BrokerSSL\IIB9BRK1\IIB9BRK1keystore.jks
mqsichangeproperties IIB9BRK1 -o BrokerRegistry -n brokerTruststoreFile
                        -v C:\BrokerSSL\IIB9BRK1\IIB9BRK1truststore.jks
```

To set the passwords for the keystore and truststore, run the following commands:

```
mqsisetdbparms IIB9BRK1 -n brokerKeystore::password -u ignore -p p@ssw0rd
mqsisetdbparms IIB9BRK1 -n brokerTruststore::password -u ignore -p p@ssw0rd
```

Restart the broker to implement the password changes. To display and verify the broker registry properties, run the following command:

```
mqsireportproperties IIB9BRK1 -o BrokerRegistry -r
BrokerRegistry
   uuid='BrokerRegistry'
   brokerKeystoreType='JKS'
   brokerKeystoreFile='C:\BrokerSSL\IIB9BRK1\IIB9BRK1keystore.jks'
   brokerKeystorePass='brokerKeystore::password'
   brokerTruststoreType='JKS'
   brokerTruststoreFile='C:\BrokerSSL\IIB9BRK1\IIB9BRK1truststore.jks'
   brokerTruststorePass='brokerTruststore::password'
   brokerCRLFileList=''
   httpConnectorPortRange=''
   httpsConnectorPortRange=''
   brokerKerberosConfigFile=''
   brokerKerberosKeytabFile=''
   modeExtensions=''
   operationMode='advanced'
   shortDesc=''
   longDesc=''
BIP8071I: Successful command completion.
```

These broker-level settings are used as default settings for the broker-wide HTTP listener and for all embedded HTTP listeners in integration servers on the broker. You can configure the PKI for the broker-wide HTTP listener. These settings override any PKI configuration set at the broker level. You can also configure the PKI for an embedded HTTP listener in an integration server. These settings override any PKI configuration set at the broker level.

Enabling broker administration security

If the broker administration security is not enabled, anyone can access the web UI, so to secure access to it, enable broker administration security by running the following commands:

```
mqsistop IIB9BRK1
mqsichangebroker IIB9BRK1 -s active
mqsistart IIB9BRK1
```

To check whether broker administration security is enabled, run the following command and check the value for administration security:

```
mqsireportbroker IIB9BRK1
BIP8927I: Broker Name 'IIB9BRK1'
   Last mqsistart path  = 'C:\IBM\IIBV9'
   mqsiprofile install path  = 'C:\IBM\IIBV9'
   Work path = 'C:\ProgramData\Application Data\IBM\MQSI'
   Broker UUID = '9e33f967-868c-4d91-8d8b-1c064707688c'
   Process id = '33696'
   Queue Manager = 'IIB9BRK1QM'
   User lil path = ''
   User exit path = ''
   Active user exits = ''
   LDAP principal = ''
   LDAP credentials = ''
   ICU converter path = ''
   Trusted (fastpath) Queue Manager application = 'false'
   Configuration change timeout = '300' seconds
   Internal configuration timeout = '60' seconds
   Statistics major interval = '60' minutes
```

```
   Operation mode = 'advanced'
   Fixpack capability level = '' (effective level 'unrestricted')
   Broker registry format = 'v9.0'
   Administration security = 'active'
   Multi-instance Broker = 'false'
   Shared Work Path = 'none'
   Start as WebSphere MQ Service = 'undefined'
   HTTP listener port = '7080'
   Cache manager policy = 'disabled'
   Cache manager port range = '2860-2879'
   Integration registry hostname = ''
BIP8071I: Successful command completion.
```

Enabling the web UI server with HTTPS

When a new broker is created, the web UI with HTTP protocol using port 4414 is enabled automatically. To change the default settings and use HTTPS protocol instead, run the following command to set up the HTTPSConnector object with PKI properties:

```
mqsichangeproperties IIB9BRK1 -b webadmin -o HTTPSConnector
-n port,keystoreFile,keystorePass,truststoreFile,truststorePass
-v "4434","C:\BrokerSSL\IIB9BRK1\IIB9BRK1keystore.jks","p@ssw0rd",
   "C:\BrokerSSL\IIB9BRK1\IIB9BRK1truststore.jks","p@ssw0rd"
```

To confirm that the HTTPS-related properties are set correctly, run the following command:

```
mqsireportproperties IIB9BRK1 -b webadmin -o HTTPSConnector -a
HTTPSConnector
   uuid='HTTPSConnector'
   algorithm='Platform Default'
   clientAuth='Platform Default'
   keystoreFile='C:\BrokerSSL\IIB9BRK1\IIB9BRK1keystore.jks'
   keystorePass='********'
   keystoreType='Platform Default'
   truststoreFile='C:\BrokerSSL\IIB9BRK1\IIB9BRK1truststore.jks'
   truststorePass='********'
   truststoreType='Platform Default'
   sslProtocol='Platform Default'
   ciphers='Platform Default'
   address=''
   port='4434'
   maxPostSize=''
   acceptCount=''
   compressableMimeTypes=''
   compression=''
   connectionLinger=''
   connectionTimeout=''
   maxHttpHeaderSize=''
   maxKeepAliveRequests=''
   maxThreads=''
   minSpareThreads=''
   noCompressionUserAgents=''
   restrictedUserAgents=''
   socketBuffer=''
   tcpNoDelay=''
   enableLookups='false'
BIP8071I: Successful command completion.
```

Run the following command to enable SSL for the web UI component:

```
mqsichangeproperties IIB9BRK1 -b webadmin -o server -n enableSSL -v true
```

Run the following command to verify that the web UI component is enabled with HTTPS:

```
mqsireportproperties IIB9BRK1 -b webadmin -o server -a
server=''
   uuid='server'
   enabled='true'
   enableSSL='true'
BIP8071I: Successful command completion.
```

The attribute enabledSSL='true' is for HTTPS. After the HTTPS protocol is enabled, the HTTP protocol is automatically disabled. Restart the broker to ensure that the changes take effect. You can also enable the web UI server with HTTPS using IBM Integration Explorer.

Setting up role-based security for the web UI

Web access to broker resources is controlled by associating web user IDs with roles. A role has security permissions assigned to it. The broker administrator can set up multiple system user IDs for the system on which the broker is running, with different permissions set on the authorization queues. These permissions then apply to web users through their assigned role.

If administration security is not enabled, web users can interact with the web UI as the default user and can access all data and broker resources.

If administration security is enabled, web users can access the web UI only when they have logged on using their web user account. Their access to data and broker resources is controlled by the permissions associated with their role.

Use the procedures described in the developerWorks article Configuring role-based security with the IBM Integration Bus V9 web UI to grant access to web users based on their assigned roles.

Accessing the web UI

After completing the role-based security configuration, you can access the broker and broker resources by logging on to the web UI in a web browser using the URL https://localhost:4434/, where localhost is the web UI server address, and 4434 is the port number that was specified for the HTTPS connector object in the preceding steps.

Enter the Web user ID and password, as shown in Figure 1, and then click Log in:

Figure 1. Web UI login screen

Administrators and web users can have different access to broker resources based on their roles. The web UI is tailored to your role, so that you see only the options that are available to you based on your role.

A window opens as shown in the Figure 2, in which you can view and administer your broker resources. The Navigator view is displayed on the left side of the window, and the content on the right side of the window varies according to the resource selected in the Navigator view:

Figure 2. IBM Integration Bus web UI
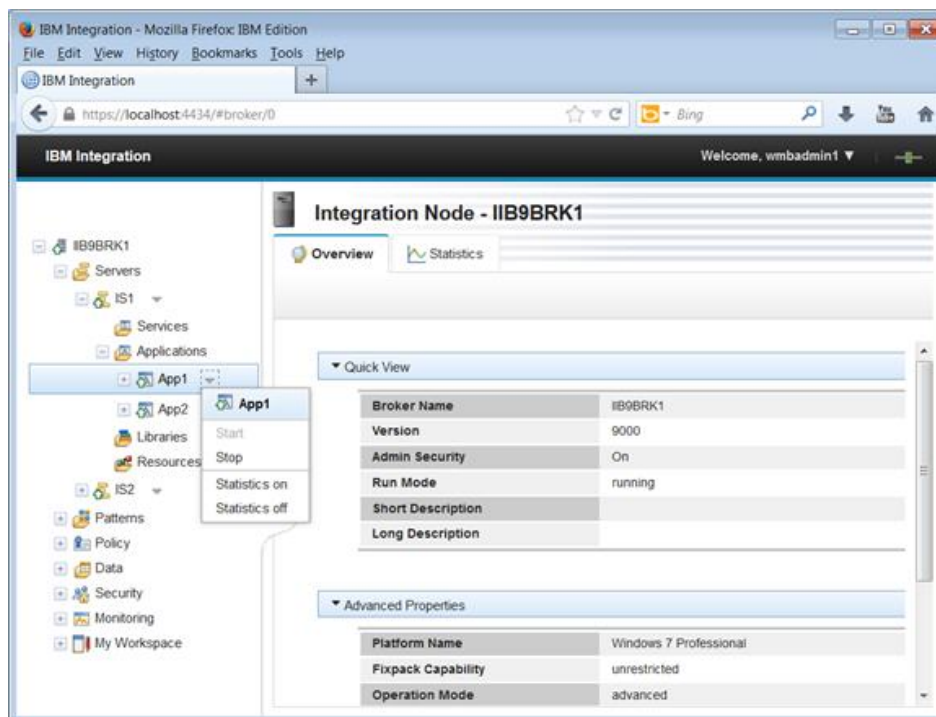
Managing broker resources using web UI

The IBM Integration Bus web UI provides basic functions for an administrato to manage broker resources, such as starting and stopping integration servers and deployed applications. You can view the administration log, including administration requests, runtime responses, and change notifications, as shown in Figure 3:

Figure 3. View of admin log
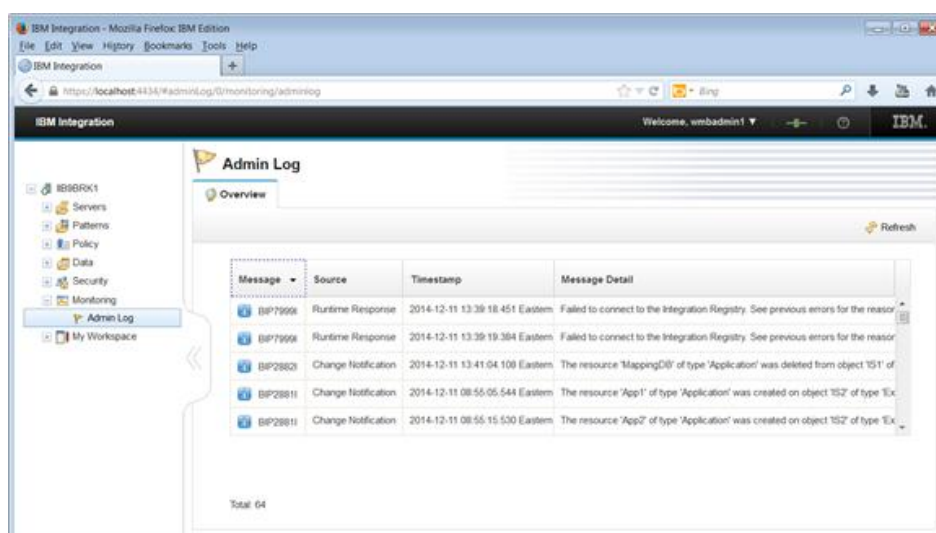
You can get and view message flow statistics reports (including message rate, elapsed time, and CPU time), as shown in Figures 4 and 5. To generate the statistics report, select the Statistics tab

in the web UI after you turn the message flow statistics on. If you start the collection of message flow statistics data using the web UI, the statistics are emitted in JSON format in addition to any other formats that are already being emitted.

Figure 4. Statistics report for all message flows on the integration node IIB9BRK1
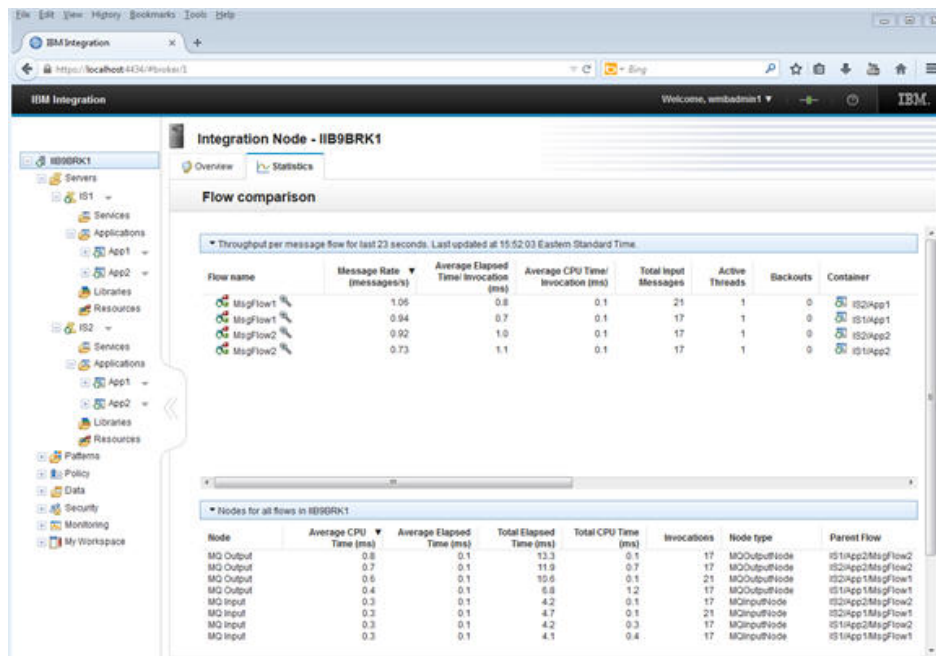
Figure 5. Statistics report for message flow MsgFlow1
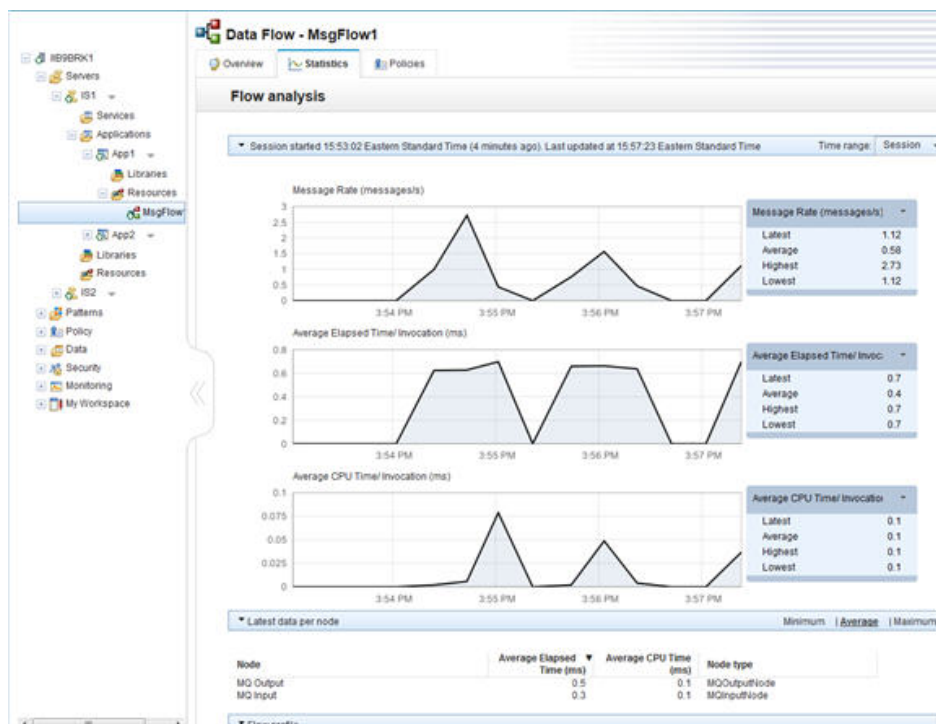
You can also view a list of recorded messages, or view details of a specific message, as shown in Figures 6 and 7, after a data capture store is created and configured. Of course, you can also replay the recorded messages, as shown in Figure 8:

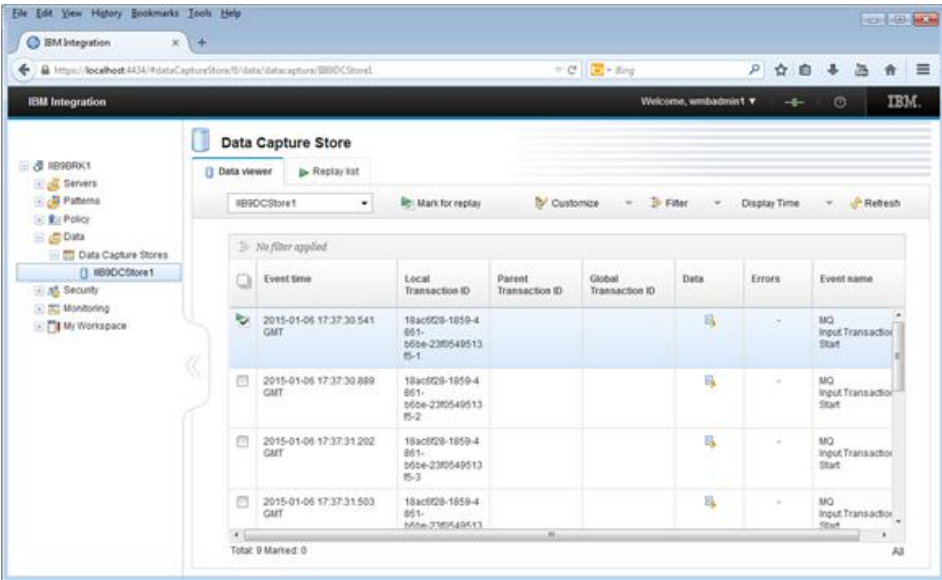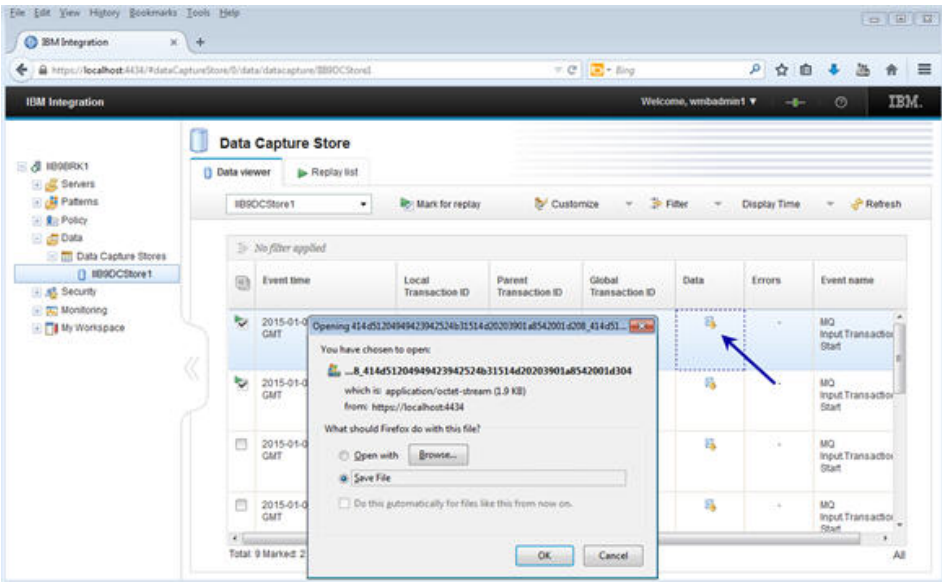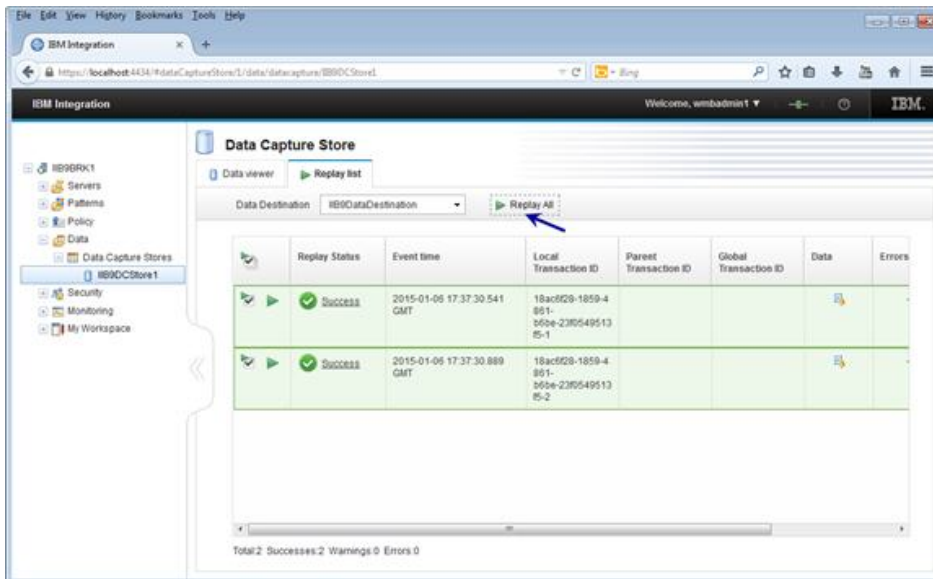## Figure 6. List of recorded messages in a data capture store

Figure 7. Opening a recorded message in a data capture store

Figure 8. Replaying recorded messages in a data capture store

There are some actions you cannot perform in the web UI, such as creating a new integration server, and removing deployed applications.

Only one broker and its resources are shown in the web UI. If you have more than one broker on the same server, and the port for HTTP or HTTPS is the same, the broker shown in the web UI is the one started first. To manage multiple brokers on the same server, assign a different HTTP or HTTPS port number to each broker.

Conclusion

This tutorial showed you how to configure the web UI with HTTPS on IBM Integration Bus V9, and described the common functions to manage broker resources using the web UI.

Acknowledgments

The authors would like to thank IBM Senior Managing Consultant and Integration Architect Sudhakar Bodapati for his help in reviewing and helping to improve this article.

Related topics

- **IBM Integration Bus resources**
  - IBM Integration Bus V9 Knowledge Center
    A single portal to all IBM Integration Bus documentation, with conceptual, task, and reference information on installing, configuring, migrating to, and using IBM Integration Bus.
  - What's new in IBM Integration Bus V9
    This IBM developerWorks article describes the highlights of IBM Integration Bus V9 for both existing WebSphere Message Broker and WebSphere ESB customers, and for those new to integration.
  - Configuring role-based security with the IBM Integration Bus V9 web UI
    This IBM developerWorks article walks you through the IBM Integration Bus V9 web UI and shows you how to use it to configure role-based security to control access to a broker and its resources.
  - IBM Integration Bus developer resources page
    Downloads, tutorials, education, product info, and other resources to help you use IBM Integration Bus to enable connectivity and transformation in heterogeneous IT environments for businesses of any size using a wide range of platforms, including cloud and z/OS.
  - IBM Integration Bus product family page
    Product features, use cases, and resources.
  - Video: What's new in IBM Integration Bus
    A short YouTube video showing key IBM Integration Bus features.
  - Download IBM Integration Bus Developer Edition
    A lightweight edition that you can use for evaluation, development, unit test, and other scenarios.
  - Follow IBM Integration Bus on Twitter
    Latest IBM Integration Bus news and announcements,
  - IBM Integration Bus forum
    Forum on mqseries.net for user questions, answers, and tips.
  - Track IBM Integration Bus user requirements
    Create, view, and track IBM Integration Bus user requirements.
- **WebSphere resources**
  - developerWorks WebSphere
    Technical resources for developers who use WebSphere and related middleware products. Includes downloads, how-to information, support resources, and a free technical library of tutorials, best practices, IBM Redbooks, and product manuals.
  - Most popular WebSphere trial downloads
    No-charge trial downloads for key WebSphere products.
  - WebSphere forums
    Product-specific forums where you can get answers to your technical questions and share your expertise with other WebSphere users.
  - WebSphere demos

Download and watch these self-running demos, and learn how WebSphere products can provide business advantage for your company.

- WebSphere-related tutorials on developerWorks
  Nearly 3000 tutorials on WebSphere and related technologies by top practitioners and consultants inside and outside IBM. Search for what you need.
- WebSphere-related books from IBM Press
  Convenient online ordering through Barnes & Noble.
- WebSphere-related events
  Conferences, trade shows, Webcasts, and other events around the world.
- **developerWorks resources**
  - Trial downloads for IBM software products
    No-charge trial downloads for selected IBM® DB2®, Lotus®, Rational®, Tivoli®, and WebSphere® products.
  - developerWorks blogs
    Join a conversation with developerWorks users and authors, and IBM developers.
  - Technical events for developers
    Conferences, seminars, and technical events to deepen your technical expertise and help you succeed in your software projects.
  - developerWorks podcasts
    Listen to interesting and offbeat interviews and discussions with software innovators.
  - developerWorks on Twitter
    Check out recent Twitter messages and URLs.
  - IBM Education Assistant
    A collection of multimedia educational modules that will help you better understand IBM software products and use them more effectively to meet your business requirements.