Deployment Guide

Public JMX Authorization

12/14/2025

Contents

Public JMX Authorization

By default, JMX is not enabled. If you want to enable JMX, you must protect yourself from the Java deserialization vulnerability and other vulnerabilities. You should secure JMX by using the configuration below or by deploying into a secure zone like a DMZ.

Enabling Remote JMX

To enable remote JMX complete the following:

1. Enabling Remote JMX Configuration
2. Setting Remote JMX Authentication

Enabling Remote JMX Configuration

You can enable remote JMX configuration for Co-browse Server or external Cassandra.

> **Important**
> Starting in 9.0.005.15, Cassandra support is deprecated in Genesys Co-browse and Redis is the default database for new customers. Support for Cassandra will be discontinued in a later release.

Enabling Remote JMX for Co-browse Server

To enable remote JMX for Co-browse Server, open your **setenv.bat/sh** file and uncomment the JMX settings below this line:

```
Uncomment to enable JMX Remote
```

Enabling Remote JMX for External Cassandra

> **Important**
> Starting in 9.0.005.15, Cassandra support is deprecated in Genesys Co-browse and Redis is the default database for new customers. Support for Cassandra will be discontinued in a later release.

If you use external Cassandra and want to monitor Co-browse column family attributes, open your

cassandra.bat (Windows) or **cassandra-env.sh** (UNIX) and enable these JMX settings:

In **cassandra.bat**, enable the settings below the line:

```
... JMX REMOTE ACCESS SETTINGS ...
```

In **cassandra-env.sh** add:

```
JMX_PORT="7199"
JVM_OPTS="$JVM_OPTS -Dcom.sun.management.jmxremote.port=$JMX_PORT"
JVM_OPTS="$JVM_OPTS -Dcom.sun.management.jmxremote.rmi.port=$JMX_PORT"
JVM_OPTS="$JVM_OPTS -Dcom.sun.management.jmxremote.ssl=false"
JVM_OPTS="$JVM_OPTS -Dcom.sun.management.jmxremote.authenticate=false"
JVM_OPTS="$JVM_OPTS -Dcom.sun.management.jmxremote.password.file=/etc/cassandra/
jmxremote.password"
JVM_OPTS="$JVM_OPTS -Dcom.sun.management.jmxremote.access.file=/etc/cassandra/
jmxremote.access"
```

Now that you enabled remote JMX, you can set remote authentication.

Setting Remote JMX Authentication

1. In the JMX remote settings you enabled above, set the following:

   ```
   -Dcom.sun.management.jmxremote.authenticate=true
   ```

2. Specify your credentials:

   a. Find the **jmxremote.access** and **jmexremmote.password.template** files in the **<JAVA_HOME>/[jre]/lib/management** directory.

   b. Rename **jmxremote.password.template** to **jmxremote.password**.

   c. By default, the JMX remote settings you enabled in **setenv.bat/sh** use the **jmxremote.access** and **jmxremote.password** files for authentication.

      To enable default roles, uncomment the role/password settings at the bottom of the **jmxremote.password** file. For example:

      ```
      monitorRole  QED
      controlRole  R&D
      ```

      > **Tip**
      > You can edit the role names and passwords but you should first make sure the defaults work.

   d. If you use the same credentials for all host applications, you can use the default password and access files. Otherwise, do the following:

      i. Copy a pair of **jmxremote.access** and **jmxremote.password** files to the path related to each application.

      ii. Add paths to the access and password files in your JMX remote configuration. For example, in Windows:

      ```
      set JMX_PORT=7199
      set JAVA_OPTS=%JAVA_OPTS% -Dcom.sun.management.jmxremote ^
      ```

```
-Dcom.sun.management.jmxremote.port=%JMX_PORT% ^
-Dcom.sun.management.jmxremote.ssl=false ^
-Dcom.sun.management.jmxremote.authenticate=true ^
-Dcom.sun.management.jmxremote.password.file=<Path>\jmxremote.password ^
-Dcom.sun.management.jmxremote.access.file=<Path>\jmxremote.access
```

3. Set the owner of the **jmxremote.password** file to the owner of the application process:
   - In Windows, open the **jmxremote.password** file properties and set the owner in **Security Tab > Advanced > Owner**.
   - In UNIX, run this command:

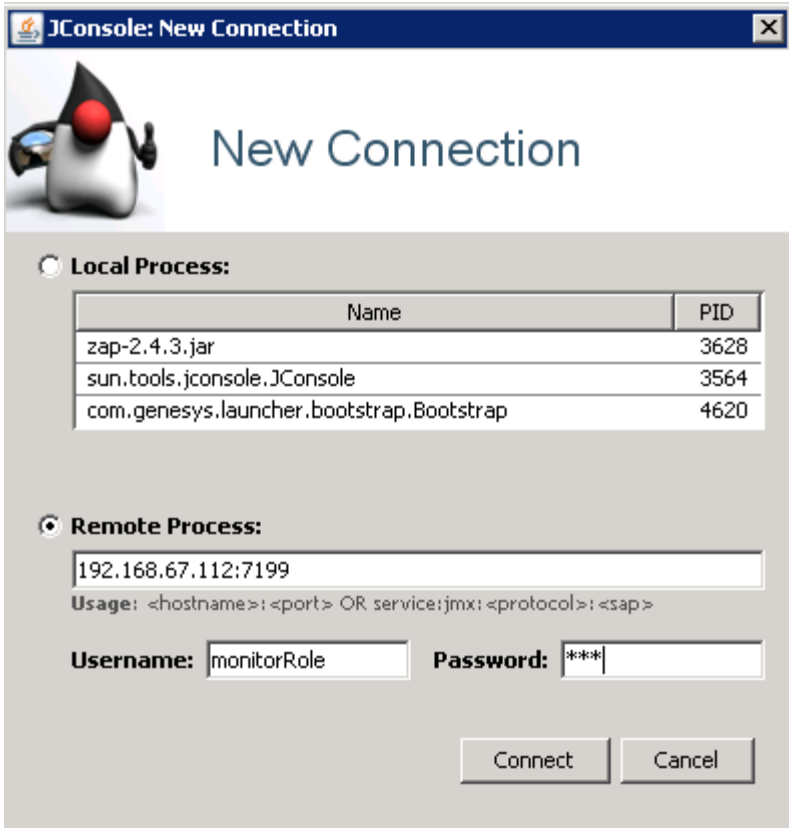
     ```
     chown <username> <path to jmxremote.password>
     ```
4. Update the permissions of the **jmxremote.password** file:
   - In Windows, open the **jmxremote.password** file's **Permissions**:
     1. Add read permissions, if absent.
     2. Remove any write permissions, for example, remove **create files/write data** and **create folders/append data**.
   - In Unix, run this command:

     ```
     chmod 444 <path to jmxremote.password>
     ```

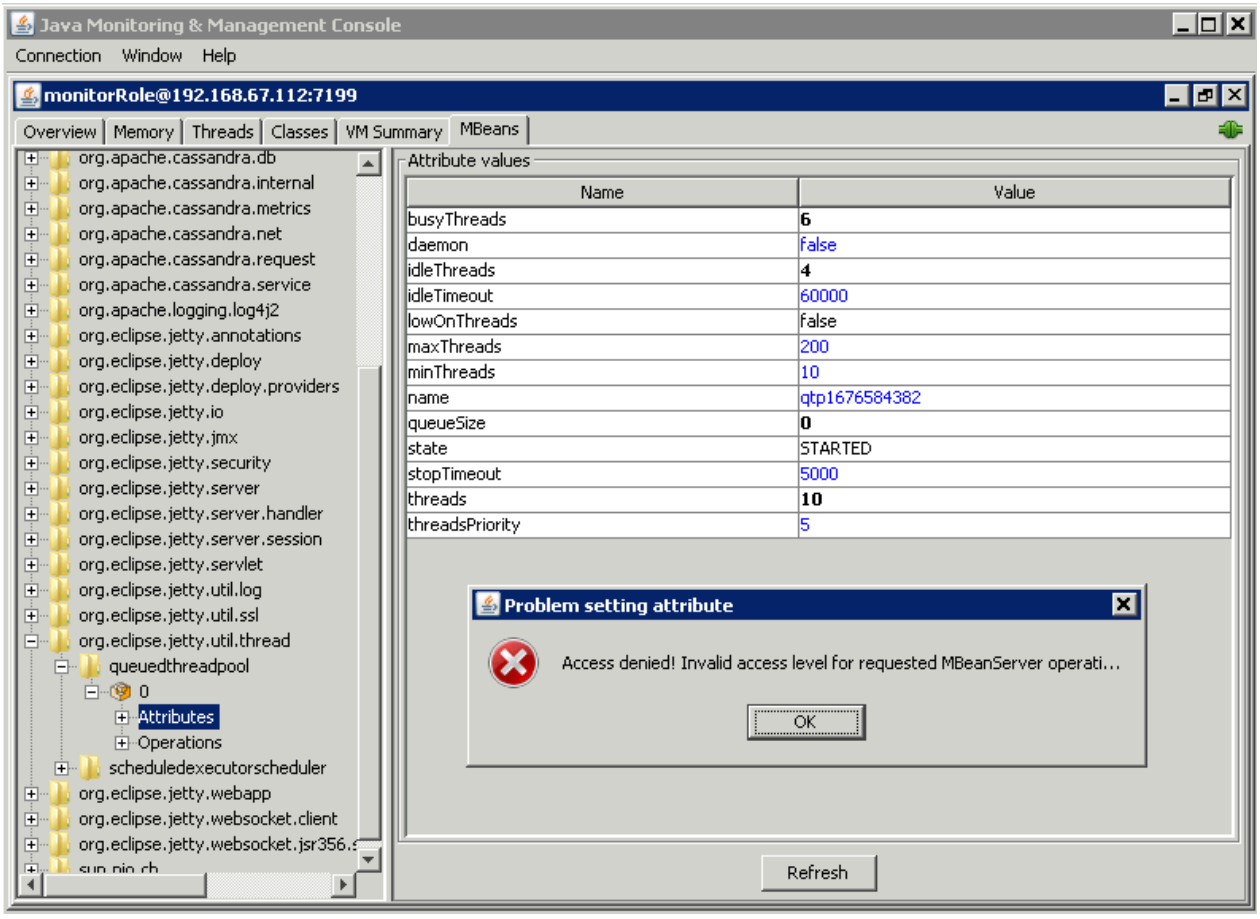
After enabling remote JMX, you can test your authentication using the procedure below.

Testing Remote JMX Authentication

1. Start the application server.
2. Run JConsole or another JMX visual tool and log in with read-only credentials (**monitorRole** by default).



3. Go to **Mbeans**. Expand **org.eclipse.jetty.util.thread** and the **queuethreadpool** attributes. Try to change the **maxThreads**. You should see an access denied error.



4. Repeat the previous steps with read-write access (**controlRole** by default) and verify you do *not* see an access denied error.

Important

As an additional security measure, you can add an SSL certificate to prevent JMX passwords from being passed as plain text. For details, see Enabling remote JMX with password authentication and SSL.