Integrated Capture Points Guide

TIBCO—JMS Capture Point and SSL connection

12/19/2025

Contents

TIBCO—JMS Capture Point and SSL connection

This page provides an example of setting up an SSL connection between TIBCO and Interaction Server for JMS Capture Point.

Prerequisites

- TIBCO EMS Community Edition 10.2
- Interaction Server 9.0.010.07 with OpenJDK 16

One-way TLS connection

This section describes a sample configuration for setting up a one-way TLS connection.

Preparing a server certificate

1. Run the OpenSSL command:

   ```
   "/C=RU/ST=SPb/L=SPb/CN=${your_tibcohost_full_name_aka_fqdn}" -newkey rsa:2048 -keyout
   tibcoserver.key.pem -out tibcoserver.pem
   ```

   In the above command as well as in the subsequent commands given in this document, the following representation is used:

   - C=<your country code>
   - ST=<your state or region code>
   - L=<your city or location code>

   You must also replace **${your_tibcohost_full_name_aka_fqdn}** with a fully qualified domain name (FQDN) of the machine where TIBCO EMS resides.

2. Provide a password, for example `tibcoserver`, if prompted.
3. When the OpenSSL command completes, two files are generated: **tibcoserver.pem** and **tibcoserver.key.pem**. Copy the files to the machine where TIBCO EMS resides and ensure that the TIBCO process has access to these files.

Configuring the TIBCO server

1. Create a new server configuration file under the **/bin** directory, for example **ssl.conf**.
2. Specify the path to the two files that you obtained using the OpenSSL command as explained in the **Preparing a server certificate** section. Additionally, specify the password that you previously provided, for example `tibcoserver`.

```
listen = ssl://7243
ssl_server_identity = /server_machine/local/path/to/tibcoserver.pem
ssl_server_key = /server_machine/local/path/to/tibcoserver.key.pem
ssl_password = tibcoserver

# uncomment text below if you need debug details
# console_trace=DEFAULT,+SSL,+SSL_DEBUG

# uncomment text below if you need debug details
# log_trace=DEFAULT,+SSL,+SSL_DEBUG

logfile = tibco.log
```

> **Important**
>
> If you are using other parameters in your regular configuration file, move them into **ssl.conf**.

3. If you have an insecure port or other ports configured on your server, specify them using the `listen` parameter, for example:

```
listen = ssl://7243,tcp://7222
```

Configuring an SSL connection factory

The TIBCO clients must obtain initial connection parameters from pre-configured connection factories. To create a new entry, use the following steps:

1. Locate the **factories.conf** file under the **/bin** directory.
2. Create a new entry, for example:

```
[SSLQueueConnectionFactory]
  type                    = queue
  url                     = ssl:/${your_tibcohost_full_name_aka_fqdn}:7243
  ssl_verify_host         = enable

  #uncomment line below if the hostname in URL is different from that in the
certificate
  #ssl_expected_hostname   = my.tibco.expected.url

  ssl_trusted             = /server_machine/local/path/to/tibcoserver.pem
```

Ensure that **${your_tibcohost_full_name_aka_fqdn}** is same as the hostname that you used during the certificate generation. Otherwise, Interaction Server will not validate the server certificate.

You can also use `ssl_expected_hostname`.

Creating queues on TIBCO

Create the following queues by editing the **queues.conf** file in the TIBCO directory:

- inbound
- error
- processed
- notification

Creating a user on TIBCO

Create a user `genesys` with the password `tibcoclient`.

Configuring Interaction Server

1. In the Interaction Server settings, locate the **jvm-config** section. The section contains the **jvm-path** option. Using this option, specify the full path to the local **jvm.dll** file. For example:

   ```
   jvm-path=C:\Program Files\OpenJDK\jdk-16.0.2\bin\server\jvm.dll
   ```

   > **Important**
   > We recommend that you use JDK 11 or a higher version.

2. Locate the **jvm-options** section. Append the path of TIBCO libraries to the value of **-Djava.class.path**. For example:

   ```
   -Djava.class.path=lib\samples-9.0.0.jar;lib\ixn-java-aux.jar;lib\groovy-
   all-2.4.21.jar;lib\XmlTransformer\xercesImpl.jar;lib\XmlTransformer\xsltc.jar;lib\
   KafkaEventLogger\kafka-clients-3.1.0.jar;lib\KafkaEventLogger\
   KafkaEventLogger.jar;lib\KafkaEventLogger\slf4j-api-1.7.36.jar;lib\KafkaEventLogger\
   avro-1.11.1.jar;lib\KafkaEventLogger\jackson-core-2.13.4.jar;lib\KafkaEventLogger\
   jackson-databind-2.13.4.2.jar;lib\KafkaEventLogger\jackson-
   annotations-2.13.4.jar;C:\3rd party jars\tibco\tibemsd_sec.jar;C:\3rd party jars\
   tibco\tibjms.jar;C:\3rd party jars\tibco\tibjmsadmin.jar;C:\3rd party jars\tibco\
   tibjmsapps.jar;C:\3rd party jars\tibco\tibrvjms.jar;C:\3rd party jars\tibco\
   jms-2.0.jar;
   ```

   Ensure that the classpath contains all .jar files that are supplied with Interaction Server in the folder **lib** in the installation directory.

Configure the Capture Point object

1. Under the Interaction Server installation directory, locate the folder **CapturePointTemplates** and its contents (files) with names starting with **JMSCapturePoint**. If you are using desktop Configuration Manager, import the Application Template using the .apd file. If you are using GAX, import the template using the .xml file.
2. Create a new application based on the imported template.

3. Copy the file **tibcoserver.pem** to the Interaction Server machine.
4. Set the following options:

```
[jms-additional-context-attributes]
com.tibco.tibjms.naming.ssl_debug_trace=true
com.tibco.tibjms.naming.ssl_trace=true
com.tibco.tibjms.naming.security_protocol=ssl
com.tibco.tibjms.naming.ssl_password=tibcoclient
com.tibco.tibjms.naming.ssl_trusted_certs=/IXN machine/local/path/to/tibcoserver.pem
java.naming.security.credentials=tibcoclient
java.naming.security.principal=genesys

[settings]
copy-original-properties-in-reply=false
error-queue-name=error
inbound-queue-name=inbound
include-ids-in-duplicate-error=false
jms-connection-factory-lookup-name=SSLQueueConnectionFactory
jms-initial-context-factory=com.tibco.tibjms.naming.TibjmsInitialContextFactory
jms-provider-url=ssl://${your_tibcohost_full_name_aka_fqdn}:7243
notification-queue-name=notification
processed-queue-name=processed
```

   You can leave other options as they are.

Verifying your setup

1. Start TIBCO using the command:

```
tibemsd.exe -ssl_trace -ssl_debug_trace -config ssl.conf
```

   Ensure that TIBCO reports **Server is ready**.

2. Open the Interaction Server's log file and check the following lines:

```
Std 23213 Capture point 'Tibco': started session on queue 'notification'
 Std 23213 Capture point 'Tibco: started session on queue 'inbound'
```

   These two records indicate that Interaction Server successfully connected to the TIBCO EMS.

Two-way TLS connection (Mutual TLS)

This section describes a sample configuration for setting up a mutual TLS connection.

Prerequisite

A one-way TLS connection is already configured using the instructions given in the One-way TLS connection section.

Preparing a client certificate

1. Run the OpenSSL command:

```
openssl.exe req -x509 -days 365 -subj "/C=RU/ST=SPb/L=SPb/CN=
 ${your_tibcohost_full_name_aka_fqdn} " -newkey rsa:2048 -keyout tibcoclient.key.pem
 -out tibcoclient.pem
```

You must replace **${your_tibcohost_full_name_aka_fqdn}** with a fully qualified domain name (FQDN) of machine where TIBCO EMS resides.

2. Provide a password, for example `tibcoclient`, if prompted.
3. When the OpenSSL command completes, two files are generated: **tibcoclient.pem** and **tibcoclient.key.pem**. Copy **tibcoclient.pem** to the machine where TIBCO server resides and ensure that the TIBCO process has access to it.
4. Combine the private and public keys into a single file:

```
openssl pkcs12 -export -in tibcoclient.old.pem -inkey tibcoclient.old.key.pem -out
 tibcoclient.p12
```

5. Provide a password, for example `tibcoclient`, if prompted.
6. When the command completes, a single file **tibcoclient.p12** is generated. Copy this file to both the Interaction Server and TIBCO machines.

Updating the TIBCO EMS Server configuration file

Add the following lines to the **ssl.conf** file under the /bin directory:

```
ssl_server_trusted =/Tibco/machine/local/path/to/tibcoclient.pem
ssl_require_client_cert = enable
```

Updating the connection factory

Update **factories.conf** with the following line:

```
ssl_identity = /Tibco/machine/local/path/to/tibcoclient.p12
```

Updating the Capture Point object

Add the following options:

```
[jms-additional-context-attributes]
com.tibco.tibjms.naming.ssl_identity=/IXN machine/local/path/to/tibcoclient.p12
[settings]
username=genesys
password=tibcoclient
```

Verifying your setup

1. Start TIBCO using the command:

```
tibemsd.exe -ssl_trace -ssl_debug_trace -config ssl.conf
```

Ensure that TIBCO reports **Server is ready**.

2. Open the Interaction Server's log file and check the following lines:

```
Std 23213 Capture point 'Tibco': started session on queue 'notification'
 Std 23213 Capture point 'Tibco: started session on queue 'inbound'
```

These two records indicate that Interaction Server successfully connected to the TIBCO EMS with mutual TLS.