



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Integrated Capture Points Guide

TIBCO—SSL for JMS Capture Point

12/19/2025

TIBCO—SSL for JMS Capture Point

In general, configuring an SSL connection consists of the following major steps:

1. Prepare the certificates.
2. Configure the JMS provider to operate in SSL mode.
3. Configure the options in Interaction Server's **jvm-options** section and add required JARs to the class path.
4. Configure the JMS Capture Point.

Configuring a capture point to use SSL (TIBCO example)

Important

This example assumes that:

- An instance of TIBCO Enterprise Message Service is configured and operating with a JMS Capture Point, without SSL.
- TIBCO EMS 6.0 is running on a host named **tibcohost**.
- OpenSSL is present.

The first several steps involve configuring the TIBCO EMS:

1. Use OpenSSL to generate the following certificates:
 - a. Generate a server certificate: `openssl req -x509 -days 365 -subj "/C=US/ST=California/L=Daly City/CN=tibcohost.genesyslab.com" -newkey rsa:2048 -keyout tibcoserver.key.pem -out tibcoserver.pem`
Note that the PEM password in this example is **tibcoserver**.
 - b. Generate a client certificate: `openssl req -x509 -days 365 -subj "/C=US/ST=California/L=Daly City/CN=tibcohost.genesyslab.com" -newkey rsa:2048 -keyout tibcoclient.key.pem -out tibcoclient.pem`
Note that the PEM password in this example certificate is **tibcoclient**.
 - c. Export the generated certificate and the key into a client identity: `openssl pkcs12 -export -in tibcoclient.pem -inkey tibcoclient.key.pem -out tibcoclient.p12`
2. Configure TIBCO properties:
 - a. New configuration file: this example assumes that the relevant certificates are copied into the folder **/opt/tibco/ems/6.0/samples/certs/**. Prepare a new TIBCO configuration file **tibemsd_ssl.conf**

based on **tibemsd.conf** by adding or modifying the following lines:

```
listen = ssl://7243
ssl_require_client_cert = enabled
ssl_server_identity = /opt/tibco/ems/6.0/samples/certs/tibcoserver.pem
ssl_server_key = /opt/tibco/ems/6.0/samples/certs/tibcoserver.key.pem
ssl_password = tibcoserver
ssl_server_trusted = /opt/tibco/ems/6.0/samples/certs/tibcoclient.pem
```

- b. Update factories configuration: In **factories.conf**, configure the following factory (or add a factory with a new name):

```
[SSLQueueConnectionFactory]
type = queue
url = ssl://tibcohost.genesyslab.com:7243
ssl_identity = //opt/tibco/ems/6.0/samples/certs/tibcoclient.p12
ssl_trusted = //opt/tibco/ems/6.0/samples/certs/tibcoserver.pem
```

- c. Use the TIBCO EMS Administration tool to create a new user: `tcp://localhost:7222> create user genesys password=tibcoclient`

Important

The user password must be exactly the same as the PEM password for the example client certificate. Note the following excerpt from the TIBCO EMS User's Guide (Chapter 18): "Because connection factories do not contain the **ssl_password** (for security reasons), the EMS server uses the password that is provided in the **create connection** call for user authentication. If the **create connection** password is different from the **ssl_password**, the connection creation will fail."

- d. Restart TIBCO with the new configuration: `tibemsd -config "{Path to tibemsd_ssl.conf}/tibemsd_ssl.conf"`
3. Configure Interaction Server options: Add the following TIBCO EMS jars to the **-Djava.class.path** option in the **jvm-options** section: **jms.jar**, **tibjms.jar**, **tibcrypt.jar**, **slf4j-simple-1.4.2.jar**, **slf4j-api-1.4.2.jar**.
4. Configure the JMS Capture Point:
- a. In the `settings` section, set options as follows:
- `jms-connection-factory-lookup-name=SSLQueueConnectionFactory`
This option points to a new connection factory.
 - `jms-provider-url=ssl://tibcohost.genesyslab.com:7243`
The provider URL now points to a secure port.
 - `password=tibcoclient`
 - `username=genesys`
The username and password correspond to those of the newly created TIBCO client.
- b. In the **jms-additional-context-attributes** section, set options as follows:
- `com.tibco.tibjms.naming.security_protocol=ssl`
 - `com.tibco.tibjms.naming.ssl_enable_verify_host=true`
 - `com.tibco.tibjms.naming.ssl_enable_verify_hostname=false`
 - `com.tibco.tibjms.naming.ssl_identity={Local path to certificates}\tibcoclient.p12`
-

- `com.tibco.tibjms.naming.ssl_password=tibcoclient`
 - `com.tibco.tibjms.naming.ssl_trusted_certs={Local path to certificates}\tibcoserver.pem`
 - `java.naming.security.credentials=tibcoclient`
 - `java.naming.security.principal=genesys`
- The following two options can be added for debugging:
- `com.tibco.tibjms.naming.ssl_debug_trace=true`
 - `com.tibco.tibjms.naming.ssl_trace=true`