Solace JMS Integration with TIBCO ActiveMatrix BusinessWorks v5

Document Version 1.1

November 2015

This document is an integration guide for using Solace JMS as a JMS provider in TIBCO ActiveMatrix BusinessWorks 5.

TIBCO ActiveMatrix BusinessWorks is a standards-based integration backbone that includes an enterprise service bus (ESB) and Web services platform used to connect disparate applications and data with little to no programming. It provides an integrated services environment (ISE) for creating Web services and orchestrating process flows to improve the consistency and adaptability of both IT and business operations.

The Solace message router supports persistent and non-persistent JMS messaging with high throughput and low, consistent latency. Thanks to very high capacity and built-in virtualization, each Solace message router can replace dozens of software-based JMS brokers in multi-tenant deployments. Since JMS is a standard API, client applications connect to Solace like any other JMS broker so companies whose applications are struggling with performance or reliability issues can easily overcome them by upgrading to Solace's hardware.

Solace Systems®

Table of Contents

Solace Systems®

1 Overview

This document demonstrates how to integrate Solace Java Message Service (JMS) with TIBCO ActiveMatrix BusinessWorks for production and consumption of JMS messages. The goal of this document is to outline best practices for this integration to enable efficient use of both TIBCO BusinessWorks and Solace JMS.

The target audience of this document is developers using TIBCO BusinessWorks with knowledge of both TIBCO BusinessWorks and JMS in general. As such this document focuses on the technical steps required to achieve the integration. For detailed background on either Solace JMS or TIBCO BusinessWorks refer to the referenced documents below.

This document is divided into the following sections to cover the Solace JMS integration with TIBCO BusinessWorks:

- ○ Integrating with TIBCO BusinessWorks
- ○ Performance Considerations
- ○ Working with Solace High Availability
- ○ Debugging Tips
- ○ Advanced Topics including:
    - ○ Using SSL Communication

1.1 Related Documentation

These documents contain information related to the feature defined in this document

Document ID	Document Title	Document Source
[Solace-Portal]	Solace Developer Portal	http://dev.solacesystems.com
[Solace-JMS-REF]	Solace JMS Messaging API Developer Guide	http://dev.solacesystems.com/docs/solace-jms-api-developer-guide
[Solace-JMS-API]	Solace JMS API Online Reference Documentation	http://dev.solacesystems.com/docs/solace-jms-api-online-reference
[Solace-FG]	Solace Messaging Platform – Feature Guide	http://dev.solacesystems.com/docs/messaging-platform-feature-guide
[Solace-FP]	Solace Messaging Platform – Feature Provisioning	http://dev.solacesystems.com/docs/messaging-platform-feature-provisioning
[Solace-CLI]	Solace Message Router Command Line Interface Reference	http://dev.solacesystems.com/docs/cli-reference
[TIBCO BusinessWorks – REF]	TIBCO ActiveMatrix BusinessWorks reference documentation	Contact TIBCO Software Inc

Table 1 - Related Documents

2 Why Solace

Solace technology efficiently moves information between all kinds of applications, users and devices, anywhere in the world, over all kinds of networks. Solace makes its state-of-the-art data movement capabilities available via hardware and software "message routers" that can meet the needs of any application or deployment environment. Solace's unique solution offers unmatched capacity, performance, robustness and TCO so our customers can focus on seizing business opportunities instead of building and maintaining complex data distribution infrastructure.

Superior Performance

Solace's hardware and software messaging middleware products can cost-effectively meet the performance needs of any application, with feature parity and interoperability that lets companies start small and scale to support higher volume or more demanding requirements over time, and purpose-built appliances that offer 50-100x higher performance than any other technology for customers or applications that require extremely high capacity or low latency.

Robustness

Solace offers high availability (HA) and disaster recovery (DR) without the need for 3rd party products, and fast failover times no other solution can match. Distributing data via dedicated TCP connections ensures an orderly, well-behaved system under load, and patented techniques ensure that the performance of publishers and high-speed consumers is never impacted by slow consumers.

Simple Architecture

Modern enterprises run applications that demand many kinds of data movement such as persistent messaging, web streaming, WAN distribution and cloud-based communications. By supporting all kinds of data movement with a unified platform that can be deployed as a small-footprint software broker or high-capacity rack-mounted appliance, Solace lets architects design an end-to-end infrastructure that's easy to build applications for, integrate with existing technologies, secure and scale.

Simple Operations

Solace's solution features a shared administration framework for all kinds of data movement, deployment models and network environments so it's easy for IT staff to deploy, monitor, manage and upgrade their Solace-based messaging environment.

Cost Savings

Solace reduces expenses with high-capacity hardware, flexible software, and the ability to deploy the right solution for each problem. Solace's support for many kinds of messaging lets you replace multiple messaging products with just one, built-in HA, DR, WAN and Web functionality eliminate the need for third-party products.

3 Integrating with TIBCO BusinessWorks

This integration guide demonstrates how to configure TIBCO BusinessWorks to send and receive JMS messages using a JMS connection. Accomplishing this requires completion of the following steps. Additionally these sections outline some BusinessWorks considerations applicable to request/reply scenarios.

- ○ Step 1 – Configuration of the Solace Appliance.
- ○ Step 2 – TIBCO BusinessWorks – Connecting
- ○ Step 3 – TIBCO BusinessWorks – Sending Messages to Solace.
- ○ Step 4 – TIBCO BusinessWorks – Receiving Messages from Solace.
- ○ Step 5 – TIBCO BusinessWorks – Request/Reply with Solace JMS
- ○ Step 6 – TIBCO BusinessWorks – SOAP/JMS Request/Reply

3.1 Description of Resources Required

This integration guide will demonstrate creation of Solace resources and configuration of TIBCO BusinessWorks managed resources. This section outlines the resources that are created and used in the subsequent sections.

3.1.1 Solace Resources

The following Solace appliance resources are required.

Resource	Value	Description
Solace appliance IP:Port	<IP>:<Port>	The IP address and port of the Solace appliance message backbone. This is the address client's use when connecting to the Solace appliance to send and receive message. This document uses a value of __IP:PORT__.
Message VPN	Solace_BW_VPN	A Message VPN, or virtual message broker, to scope the integration on the Solace appliance.
Client Username	bw_user	The client username.
Client Password	bw_password	Optional client password.
Solace Queue	Q/requests	Solace destination of messages produced and consumed
JNDI Connection Factory	JNDI/CF/bw	The JNDI Connection factory for controlling Solace JMS connection properties
JNDI Queue Name	JNDI/Q/requests	The JNDI name of the queue used in the samples

Table 2 – Solace Configuration Resources

3.1.2 TIBCO BusinessWorks Configuration Resources

Resource	Value
JMS Connection [JMS Palette]	Solace.JndiTemplate
Global Variables	Relevant Global Variable values for Solace configuration resources, e.g. JMS URL, Username Password etc. Please refer to the above table

3.2   Step 1 – Configuring the Solace Appliance

The Solace appliance needs to be configured with the following configuration objects at a minimum to enable JMS to send and receive messages within TIBCO BusinessWorks.

- A Message VPN, or virtual message broker, to scope the integration on the Solace appliance.

- Client connectivity configurations like usernames and profiles

- Guaranteed messaging endpoints for receiving messages.

- Appropriate JNDI mappings enabling JMS clients to connect to the Solace appliance configuration.

For reference, the CLI commands in the following sections are from SolOS version 7.0 but will generally be forward compatible. For more details related to Solace appliance CLI see [Solace-CLI]. Wherever possible, default values will be used to minimize the required configuration. The CLI commands listed also assume that the CLI user has a Global Access Level set to Admin. For details on CLI access levels please see [Solace-FG] section "User Authentication and Authorization".

Also note that this configuration can also be easily performed using SolAdmin, Solace's GUI management tool. This is in fact the recommended approach for configuring a Solace appliance. This document uses CLI as the reference to remain concise.

3.2.1   Creating a Message VPN

This section outlines how to create a message-VPN called "Solace_BW_VPN" on the Solace appliance with authentication disabled and 2GB of message spool quota for Guaranteed Messaging. This message-VPN name is required in TIBCO BusinessWorks configuration when connecting to the Solace messaging appliance. In practice appropriate values for authentication, message spool and other message-VPN properties should be chosen depending on the end application's use case.

```
(config)# create message-vpn Solace_BW_VPN
(config-msg-vpn)# authentication
(config-msg-vpn-auth)# user-class client
(config-msg-vpn-auth-user-class)# basic auth-type none
(config-msg-vpn-auth-user-class)# exit
(config-msg-vpn-auth)# exit
(config-msg-vpn)# no shutdown
(config-msg-vpn)# exit
(config)#
(config)# message-spool message-vpn Solace_BW_VPN
(config-message-spool)# max-spool-usage 2000
(config-message-spool)# exit
(config)#
```

3.2.2   Configuring Client Usernames & Profiles

This section outlines how to update the default client-profile and how to create a client username for connecting to the Solace appliance. For the client-profile, it is important to enable guaranteed messaging for JMS messaging and transacted sessions if using transactions.

The chosen client username of "bw_user" will be required by TIBCO BusinessWorks when connecting to the Solace appliance.

```
(config)# client-profile default message-vpn Solace_BW_VPN
(config-client-profile)# message-spool allow-guaranteed-message-receive
(config-client-profile)# message-spool allow-guaranteed-message-send
(config-client-profile)# message-spool allow-transacted-sessions
(config-client-profile)# exit
(config)#
(config)# create client-username bw_user message-vpn Solace_BW_VPN
(config-client-username)# acl-profile default
(config-client-username)# client-profile default
(config-client-username)# no shutdown
(config-client-username)# exit
(config)#
```

3.2.3   Setting up Guaranteed Messaging Endpoints

This integration guide shows receiving messages within TIBCO BusinessWorks from a single JMS Queue. For illustration purposes, this queue is chosen to be an exclusive queue with a message spool quota of 2GB matching quota associated with the message VPN. The queue name chosen is "Q/requests".

```
(config)# message-spool message-vpn Solace_BW_VPN
(config-message-spool)# create queue Q/requests
(config-message-spool-queue)# access-type exclusive
(config-message-spool-queue)# max-spool-usage 2000
(config-message-spool-queue)# permission all delete
(config-message-spool-queue)# no shutdown
(config-message-spool-queue)# exit
(config-message-spool)# exit
(config)#
```

3.2.4   Setting up Solace JNDI References

To enable the JMS clients to connect and look up the Queue destination required by TIBCO BusinessWorks, there are two JNDI objects required on the Solace appliance:

- ○   A connection factory: JNDI/CF/bw

- ○   A queue destination: JNDI/Q/requests

They are configured as follows:

```
(config)# jndi message-vpn Solace_BW_VPN
(config-jndi)# create connection-factory JNDI/CF/bw
(config-jndi-connection-factory)# property-list messaging-properties
(config-jndi-connection-factory-pl)# property default-delivery-mode persistent
(config-jndi-connection-factory-pl)# exit
(config-jndi-connection-factory)# property-list transport-properties
(config-jndi-connection-factory-pl)# property direct-transport false
(config-jndi-connection-factory-pl)# property "reconnect-retry-wait" "3000"
(config-jndi-connection-factory-pl)# property "reconnect-retries" "20"
(config-jndi-connection-factory-pl)# property "connect-retries-per-host" "5"
(config-jndi-connection-factory-pl)# property "connect-retries" "1"
(config-jndi-connection-factory-pl)# exit
(config-jndi-connection-factory)# exit
(config-jndi)#
(config-jndi)# create queue JNDI/Q/requests
(config-jndi-queue)# property physical-name Q/requests
(config-jndi-queue)# exit
(config-jndi)#
(config-jndi)# no shutdown
(config-jndi)# exit
(config)#
```

3.3 Step 2 – TIBCO BusinessWorks – Connecting

TIBCO Designer is the IDE used for configuring TIBCO BusinessWorks applications. The same can be used for testing integration with Solace JMS.

For more details you may refer to [TIBCO BusinessWorks – REF]. This can be accessed from the TIBCO Designer Help Menu under the JMS link. Setting up TIBCO BusinessWorks requires two steps to be completed. First, the Solace JMS libraries must be copied to TIBCO BusinessWorks. Then TIBCO BusinessWorks must be configured correctly. See the following sections for details.

3.3.1 Install the Solace JMS libraries in TIBCO BusinessWorks

Solace JMS libraries need to be added to the classpath of TIBCO BusinessWorks. BusinessWorks provides a common location for "third party client libraries" (tpcl) that need to be included in the classpath. This is the folder named **tpcl**. When the TIBCO BusinessWorks engine starts, all the jar files in the tpcl directory are automatically loaded in its classpath.

In order to install Solace JMS, copy the Solace JMS libraries (all the jar files under the Solace JMS API distribution) under %TRA_ HOME%/tpcl/x.x/lib folder, where x/x is the TIBCO product version.

3.3.2 Configure TIBCO BusinessWorks

In order to establish a connection with the Solace appliance, it is necessary to create a new JMS Connection Resource. To create a new JMS Connection from the JMS Palette, follow these steps:

1. Drag and Drop a new JMS Connection Resource from the JMS Palette to your project under any folder (e.g. Shared Resources).

2. Select the JMS Connection, and configure the JMS Connection Properties as outlined below: in the provided Input boxes. Italics added to provide further Solace specific context for individual properties.

Name: SolaceJMSConnection *(or anything as per your project's naming conventions)*
Description: *Optional*
User Name: **bw_user** *(This is the value of the Solace Client Username created above)*
Password: **bw_password** *(blank if no password was set while creating the Solace Client Username)*
Auto-generated Client ID: **Checked**
Client ID: **Leave Blank**
SSL: **Unchecked**
Use JNDI for Connection Factory: **Checked**
Use Shared JNDI Configuration: **Unchecked**

JNDI Context Factory: **com.solacesystems.jndi.SolJNDIInitialContextFactory** *(type, if it doesn't appear in the drop down)*
JNDI Context URL: **smf://<solace ip>:<port>** (note that this is the Solace Data Plane IP, not the management plane IP)
JNDI User Name: **bw_user** *(This is the value of the Solace Client Username created above)*
JNDI Password: **bw_password** *(blank if no password was set while creating the Solace Client Username)*
JNDI Context Factory: **JNDI/CF/BW** *(this is the name of the connection factory created in Solace*

3. Click Advanced. Configure the following property values in the Advanced Tab in the provided input boxes.

Topic Connection Factory: **JNDI/CF/bw** *(this is the connection factory you created in Solace above. The same name can be used for both Topic and Queue Connection Factories)*
Queue Connection Factory: **JNDI/CF/bw** *(this is the connection factory you created in Solace above. The same name can be used for both Topic and Queue Connection Factories)*

JNDI Properties: *Click on the Plus (+) icon on the right side to add a new JNDI Property as follows:*
Name   *Type*   *Value*
Solace_JMS_VPN   **string**   **Solace_BW_VPN**

This above is the name of the Solace VPN (partition) as created above

4. Click on Test Connection. You should get a Connection Successful Message

Your Solace JMS Connection configuration is complete. You can now use the Solace JMS Connection in any JMS Palette resources, such as sending and receiving messages.

If you don't get a Connection Successful Message, see Section 6 Debugging Tips for Solace JMS API Integration for some suggestions on how to troubleshoot common errors when connecting to the Solace appliance.

Note that you can also configure all the JMS Connection Properties as "Global Variables" and refer to them from the JMS Connection Pallete. Using Global Variables is a TIBCO BusinessWorks best practice, as it allows you to deploy your BusinessWorks application from one environment to another, by simply updating Global Variable values avoiding the need for code modification.

3.4   Step 3 – TIBCO BusinessWorks – Sending Messages to Solace

To send a message to a Solace destination follow these steps

○ Create a Process in any folder (e.g. Processes)

○ Add the JMS Queue Sender or the JMS Topic Publisher Resource to the Process

○ Configure the JMS Queue Sender or JMS Topic Publisher Resource to **use the Solace JMS Connection Resource** (as created above) to send the messages.

○ **Configure the Queue JNDI Name** (or Topic JNDI Name) as configured above (i.e. **JNDI/Q/requests)**

○ Complete and Test the process via Tester. You can monitor the Queue receiving messages via SolAdmin/Solace CLI, or can consume from the queue.

3.5   Step 4 – TIBCO BusinessWorks – Receiving Messages from Solace

To receive a message to a Solace a queue or a topic follow these steps

○ Create a Process in any folder (e.g. Processes).

○ Add the JMS Queue Receiver or the JMS Topic Subscriber Resource to the Process as a Started Resource

○ Configure the JMS Queue Receiver or JMS Topic Subscriber Resource to **use the Solace JMS Connection Resource** (as created above) to send the messages.

○ **Configure the Queue JNDI Name** (or Topic JNDI Name) as configured above (i.e. **JNDI/Q/requests)**

○ Complete and Test the process via Tester. You can test by sending messaged to the Queue via the Sender process created in 2.4, or you can also monitor the "bind count" on the queue via SolAdmin/Solace CLI.

3.6   Step 5 – TIBCO BusinessWorks – Request/Reply with Solace JMS

The steps outlined in Section 3.4 and 3.5 can be followed to realize a Request Reply pattern with TIBCO BusinessWorks, by using the JMS Queue Requestor and JMS Topic Requestor palettes.

However, it is noted that in some versions of BusinessWorks, the default "Reply To" destination generated by BusinessWorks violates the JMS Specification and won't work.

In such cases (or even otherwise), overriding the "Reply To" destination under Advanced settings of the JMS Queue/Topic Requestor with any unique string solves the problem.

3.7   Step 6 – TIBCO BusinessWorks – SOAP/JMS Request/Reply

SOAP/JMS works seamlessly with Solace JMS. The configured values outlined in Section 3.4 and 3.5 can be used in relevant Input Boxes as supplied by the SOAP and Services Palettes of TIBCO Designer.

However, just like in Request Reply, it is noted that in some versions of BusinessWorks, the default "Reply To" destination generated by BusinessWorks violates the JMS Specification and won't work.

In such cases (or even otherwise), overriding the "Reply To" destination under Advanced settings of the JMS Queue/Topic Requestor with any unique string solves the problem.

4 Performance Considerations

3.1 JMS Publisher Throughput - Round Trip Time Dependency

In JMS, the maximum throughput possible to be achieved by a publisher is a function of the Round Trip Time between the TIBCO BusinessWorks Server and the JMS Broker.

The JMS1.1 specification mandates that a JMS API sending persistent messages must wait for explicit ACKs from the JMS Broker for the message it published, before it can send the next message. There is no support for windowed or asynchronous publishing.

This limits the throughput of a given instance of a JMS Publisher.

For example, if you round trip time between the TIBCO BusinessWork Server and the Solace appliance is 10ms, the maximum JMS publish rate will be 1000ms/10ms = 100 messages per second. Even though a Solace appliances can handle hundred of thousands of persistent messages per second, the per publisher JMS rate is limited as shown in the example.

In order to determine trip times, it is possible to do a ping test to the Solace appliance at the applicable payload size. Or alternatively the round trip time can be shown on the Solace appliance cli by showing a client's connections using wide mode or via SolAdmin under a client's details by double clicking the client connection to get the full details.

Therefore, in order to effectively scale the throughput when using TIBCO BusinessWorks it may be required to use more than one JMS publisher. TheSolace appliances can support thousands of such publishers so this is not a limitation but a requirement to properly configure TIBCO BusinessWorks to take advantage of multiple publishers.

Development Note: Often when developers develop against a local installation of TIBCO EMS, the RTT limitation is not obvious due to the message broker being on the same server as the TIBCO BusinessWorks application. However in production, when EMS is not local, and will also have a RTT similar to the Solace appliance, the same throughput restriction appears. This is because the RTT is a function of the network latency and not the message broker.

It's important to design JMS applications, including TIBCO BusinessWorks applications keeping in mind the above restriction.

Solace Systems®

5 Working with Solace High Availability (HA)

The [Solace-JMS-REF] section "Establishing Connection and Creating Sessions" provides details on how to enable the Solace JMS connection to automatically reconnect to the standby appliance in the case of a HA failover of a Solace appliance. By default Solace JMS connections will reconnect to the standby appliance in the case of an HA failover.

In general the Solace documentation contains the following note regarding reconnection:

Note: When using HA redundant appliances, a fail-over from one appliance to its mate will typically occur in under 30 seconds, however, applications should attempt to reconnect for at least five minutes.

In section 3.2.4 Setting up Solace JNDI References, the Solace CLI commands correctly configured the required JNDI properties to reasonable values. These commands are repeated here for completeness.

```
config)# jndi message-vpn Solace_BW_VPN
(config-jndi)# create connection-factory JNDI/CF/bw
(config-jndi-connection-factory)# property-list transport-properties
(config-jndi-connection-factory-pl)# property "reconnect-retry-wait" "3000"
(config-jndi-connection-factory-pl)# property "reconnect-retries" "20"
(config-jndi-connection-factory-pl)# property "connect-retries-per-host" "5"
(config-jndi-connection-factory-pl)# property "connect-retries" "1"
(config-jndi-connection-factory-pl)# exit
(config-jndi-connection-factory)# exit
(config-jndi)# exit
(config)#
```

6 Debugging Tips for Solace JMS API Integration

The key component for debugging integration issues with the Solace JMS API is the API logging that can be enabled. How to enable logging in the Solace API is described below.

6.1 How to enable Solace JMS API logging

Solace supports log4j logging. To enable logging for Solace APIs in TIBCO BusinessWorks the log4j logging has to be enabled for BusinessWorks. Please refer to the TIBCO Documentation for details on how to enable log4j.

Once log4j logging has been enabled, the following logging appenders can be added to control Solace API logging:

```
log4j.category.com.solacesystems.jms=INFO
log4j.category.com.solacesystems.jcsmp=INFO
```

6.2 Common Issues Connecting to the Solace appliance

The follow is a list of common problems when encountered when creating a TIBCO BusinessWorks JMS Connection Resource.

Error Details	Resolution
com.solacesystems.jndi.SolJNDIInitialContextFactory not found	It means that the Solace libraries are not in the correct folder/path. Make sure that you have added the ALL the Solace JMS and JCSMP jar files in the correct **tpcl** lib folder.
VPN Name shut down	This means that the VPN Name that you are using is not enabled or configured. Note that if you don't configure the VPN name on the Advanced page, Designer will try and connect to the "default" VPN, which might not be enabled, or might not have the Username defined in it.
Client User Name shut down	This means that the user that you are using was either not found or not enabled. This could also mean that the VPN name is wrong, and you are trying to connect with a username that does not exist in the given VPN. Make sure that the correct Solace VPN name and user name are configured
Connection Factory errors	Make sure that the correct Connection Factory is configured. Also make sure that on the Solace appliance, the VPN is configured to allow JNDI lookups. (By default it's disabled).

Table 3 – Common Connectivity Issues

If your problem still persists then please contact Solace Systems Support for further assistance.

7 Advanced Topics

7.1 Using SSL Communication

This section outlines how to update the Solace appliance and TIBCO BusinessWorks configuration to switch the client connection to using secure connections with the Solace appliance. For the purposes of illustration, this section uses a server certificate on the Solace appliance and basic client authentication. It is possible to configure Solace JMS to use client certificates instead of basic authentication. This is done using configuration steps that are very similar to those outlined in this document. The [Solace-FP] and [Solace-JMS-REF] outline the extra configuration items required to switch from basic authentication to client certificates.

To change TIBCO BusinessWorks from using a plain text connection to a secure connection, first the Solace appliance configuration must be updated as outlined in Section 7.1.1 and the Solace JMS configuration within TIBCO BusinessWorks must be updated as outlined in Section 7.1.2.

7.1.1 Configuring the Solace Appliance

To enable secure connections to the Solace appliance, the following configuration must be updated on the Solace appliance.

- ○ Server Certificate
- ○ TLS/SSL Service Listen Port
- ○ Enable TLS/SSL over SMF in the Message VPN

The following sections outline how to configure these items.

7.1.1.1 Configure the Server Certificate

Before, starting, here is some background detail on the server certificate required by the Solace appliance. This is from the [Solace-FP] section "Setting a Server Certificate"

> To enable the exchange of information through TLS/SSL-encrypted SMF service, you must set the TLS/SSL server certificate file that the Solace appliance is to use. This server certificate is presented to a client during the TLS/SSL handshakes. A server certificate used by an appliance must be an x509v3 certificate and it must include a private key. The server certificate and key use an RSA algorithm for private key generation, encryption and decryption, and they both must be encoded with a Privacy Enhanced Mail (PEM) format.
>
> The single server certificate file set for the appliance can have a maximum chain depth of three (that is, the single certificate file can contain up to three certificates in a chain that can be used for the certificate verification).

To configure the server certificate, first copy the server certificate to the Solace appliance. For the purposes of this example, assume the server certificate file is named "mycert.pem".

```
# copy sftp://[<username>@]<ip-addr>/<remote-pathname>/mycert.pem /certs
<username>@<ip-addr>'s password:
#
```

Then set the server certificate for the Solace appliance.

```
(config)# ssl server-certificate mycert.pem
(config)#
```

7.1.1.2 Configure TLS/SSL Service Listen Port

By default, the Solace appliance accepts secure messaging client connections on port 55443. If this port is acceptable then no further configuration is required and this section can be skipped. If a non-default port is desired, then follow the steps below. Note this configuration change will disrupt service to all clients of the Solace appliance and should therefore be performed during a maintenance window when this client disconnection is acceptable. This example assumes that the new port should be 55403.

```
(config)# service smf
(config-service-smf)# shutdown
All SMF and WEB clients will be disconnected.
Do you want to continue (y/n)? y
(config-service-smf)# listen-port 55403 ssl
(config-service-smf)# no shutdown
(config-service-smf)# exit
(config)#
```

7.1.1.3 Enable TLS/SSL within the Message VPN

By default within Solace message VPNs both the plain-text and SSL services are enabled. If the Message VPN defaults remain unchanged, then this section can be skipped. However, if within the current application VPN, this service has been disabled, then for secure communication to succeed it should be enabled. The steps below show how to enable SSL within the SMF service to allow secure client connections from TIBCO BusinessWorks.

```
(config)# message-vpn Solace_BW_VPN
(config-msg-vpn)# service smf
(config-msg-vpn-service-smf)# ssl
(config-msg-vpn-service-ssl)# no shutdown
(config-msg-vpn-service-ssl)# exit
(config-msg-vpn-service-smf)# exit
(config-msg-vpn-service)# exit
(config-msg-vpn)# exit
(config)#
```

7.1.2 Configuring TIBCO BusinessWorks

Follow the steps as described in section 3.3, to create a new JMS Connection. The following modifications to the configuration should be done:

1. Include the following line in the **designer.tra** and the **bwengine.tra** files. These files are located in the bin folders of the relevant product under TIBCO_HOME. This line can included after the Common Variables section.

```
java.property.TIBCO_SECURITY_VENDOR j2se
```

2. Modify the Connection Properties as outlined below. The RED text highlights the items that are specifically required for SSL connectivity.

Name: SolaceJMSConnection *(or anything as per your project's naming conventions)*
Description: *Optional*
User Name: **bw_user** *(This is the value of the Solace Client Username created above)*
Password: **bw_password** *(blank if no password was set while creating the Solace Client Username)*
Auto-generated Client ID: **Checked**
Client ID: **Leave Blank**
SSL: **Unchecked (SSL will be configured in the advanced tab)**
Use JNDI for Connection Factory: **Checked**
Use Shared JNDI Configuration: **Unchecked**

JNDI Context Factory: **com.solacesystems.jndi.SolJNDIInitialContextFactory** *(type, if it doesn't appear in the drop down)*
JNDI Context URL: **smfs://<solace ip>:<port>** (note that this is the Solace Data Plane IP, not the management plane IP. Note that the protocol is smfs and not smf for using SSL)
JNDI User Name: **bw_user** *(This is the value of the Solace Client Username created above)*
JNDI Password: **bw_password** *(blank if no password was set while creating the Solace Client Username)*
JNDI Context Factory: **JNDI/CF/BW** *(this is the name of the connection factory created in Solace*

3. Modify the Connection Properties as below

Topic Connection Factory: **JNDI/CF/bw** *(this is the connection factory you created in Solace above. The same name can be used for both Topic and Queue Connection Factories)*
Queue Connection Factory: **JNDI/CF/bw** *(this is the connection factory you created in Solace above. The same name can be used for both Topic and Queue Connection Factories)*

JNDI Properties: *Click on the Plus (+) icon on the right side to add a new JNDI Property as follows:*
Name *Type* *Value*
Solace_JMS_VPN **string** **Solace_BW_VPN**
Solace_JMS_SSL_TrustStore **string** **C:\truststore.jks** *(the path to your keystore)*
Solace_JMS_SSL_TrustStorePassword **string** **<trust store password>**

The root CA of the appliance's certificate is installed in the trust store

4. Click on Test Connection. You should get a Connection Successful Message.

5. Repeat the steps above for sending and receiving messages. No special configuration is required.

6. You can verify from SolAdmin whether the connection is over SSL or not as shown in the below screenshot: