domibusConnectorClient-4.0-RELEASE -

Administration- and User-Guide

Table of contents

[Table of contents 2](#_Toc511909880)

[1. Introduction 4](#_Toc511909881)

[1.1. Scope and Objective of this document 4](#_Toc511909882)

[1.2. The domibusConnector as a web application 4](#_Toc511909883)

[1.3. The domibusConnectorClient 4](#_Toc511909884)

[1.4. The gateway 4](#_Toc511909885)

[1.5. The domibus-connector-plugin 5](#_Toc511909886)

[2. Preconditions and technical requirements 6](#_Toc511909887)

[2.1. The domibusConnector distribution package 6](#_Toc511909888)

[2.2. Supported operating systems 7](#_Toc511909889)

[2.3. Java Runtime 7](#_Toc511909890)

[2.4. Database 7](#_Toc511909891)

[2.5. Web container 7](#_Toc511909892)

[2.6. Internet connection 8](#_Toc511909893)

[2.7. Technical specifications 8](#_Toc511909894)

[3. Database Installation 9](#_Toc511909895)

[3.1. Supported Database vendors 9](#_Toc511909896)

[3.2. New Database / Fresh Installation 9](#_Toc511909897)

[3.2.1. Using the scripts 9](#_Toc511909898)

[3.2.2. Using liquibase 9](#_Toc511909899)

[3.3. Database Upgrade 3.5 to 4.0 9](#_Toc511909900)

[3.3.1. Using the script 10](#_Toc511909901)

[3.3.2. Using liquibase 10](#_Toc511909902)

[3.4. Upgrade with Liquibase 10](#_Toc511909903)

[4. Configuration properties 12](#_Toc511909904)

[5. Certificate, Key-Stores and Truststores 13](#_Toc511909905)

[5.1. Connector Backend Key Store 14](#_Toc511909906)

[5.2. Connector Key Store 14](#_Toc511909907)

[5.3. Evidence Key Store 14](#_Toc511909908)

[5.4. Connector truststore 15](#_Toc511909909)

[5.5. TLS Key Store (System Key Store) 15](#_Toc511909910)

[6. Deployment 16](#_Toc511909911)

[7. Import of p-modes 17](#_Toc511909912)

[7.1. Import of a p-mode file 17](#_Toc511909913)

[7.2. DataTables 17](#_Toc511909914)

[8. Backend configuration 19](#_Toc511909915)

[8.1. Backend types 19](#_Toc511909916)

[8.1.1. Push/pull backend 19](#_Toc511909917)

[8.1.2. Push/push backend 19](#_Toc511909918)

[8.2. Adding the backend client keys to the Connector Backend Key Store 19](#_Toc511909919)

[8.3. Configuring the backend at the database 20](#_Toc511909920)

[8.3.1. DOMIBUS\_CONNECTOR\_BACKEND\_INFO 20](#_Toc511909921)

[8.3.2. DOMIBUS\_CONNECTOR\_BACK\_2\_S 20](#_Toc511909922)

[8.3.3. Example scripts 21](#_Toc511909923)

# Introduction

## Scope and Objective of this document

This document is a technical guide on how the domibusConnectorClient can give support in all its variants is to be integrated, installed and used to connect the domibusConnector web application and your national implementation.

The focus of this document is to give technical guidance on taking the decision on the variant of the domibusConnectorClient to be used to fit the needs of your national environment best. It also describes the functionalities of the domibusConnectorClient and gives an overview of preconditions to be met to integrate the domibusConnectorClient into your environment.

Therefore this guide addresses technical staffs that have knowledge of the national implementation in place and network infrastructure.

This guide document for the domibusConnectorClient only focuses on the client itself.

For more detailed information on the domibusConnector web application, please see the documentation shipped with the domibusConnector-4.0-RELEASE.

## The domibusConnector as a web application

Starting with version 4.0-RELEASE, the domibusConnector is on the technical basis of a web application.

This means, that the domibusConnector itself is a “ready-to-use” software component that only needs to be configured, set-up and deployed in a web container.

Once installed and configured properly, the domibusConnector should run on its own, besides some maintenance.

The domibusConnectorClient otherwise is delivered in different variants to support the connection between the domibusConnector web application and your national environment.

It is advised though, that the domibusConnector web application is already properly installed and configured in your environment before starting with the integration of the domibusConnectorClient.

## The domibusConnectorClient

The domibusConnector web application offers different interfaces that can be used to approach its functionalities. Those interfaces can be used directly, if intended. A detailed description of the interfaces offered by the domibusConnector web application can be found in the documentation shipped with the domibusConnector-4.0-RELEASE.

To close the missing link between your own implementation of the e-CODEX use cases and the services of the domibusConnector, also a domibusConnectoClient has been implemented to support the connection to the domibusConnector.

The domibusConnectorClient is distributed in the following different variants that are described in detail in this document:

* domibusConnectorClient-4.0-RELEASE-Libraries
* domibusConnectorClient-4.0-RELEASE-Standalone

# Preconditions and technical requirements

This chapter describes what requirements have to be fulfilled to use the functionalities of the domibusConnectorClient. It also lists some technical specifications of the domibusConnectorClient to give a more detailed insight.

## Supported operating systems

The domibusConnectorClient is a software product, that was completely implemented using the JAVA programming language.

As JAVA is by definition a platform independent environment, every operating system with a proper JAVA installation should fit the needs of setting up/ integrate the domibusConnectorClient.

During implementation and testing phase of the domibusConnectorClient, it was tested and installed on the following environments:

* Microsoft Windows 7
* Linux
* IBM AIX

## Java Runtime

As the domibusConnectorClient is a JAVA application, it also requires a proper installation of a Java Runtime to be able to run the software.

The recent version 4.0-RELEASE of the domibusConnectorClient was implemented and compiled with an Oracle JDK jdk-8u161. So at least this version or above should be in place to avoid incompatibilities.

## Technical specifications

For your information the main frameworks and technologies the domibusConnectorClient was implemented with is listed here.

* Java 8 (Oracle jdk-8u161)
* Spring framework 4.3.12.RELEASE
* Spring-boot 1.5.8.RELEASE
* Apache CXF 3.2.1
* Apache Maven 3

## The domibusConnectorClient distribution package

To get started, you first need to download and extract the distribution package.

The domibusConnectorClient distribution package is placed on the e-CODEX Nexus repository server at:

https://secure.e-codex.eu/nexus/content/groups/public/eu/domibus/connector/client/domibusConnectorClientDistribution/4.0-RELEASE /

To get access to the distribution packages, you need to identify via authentication at the Nexus server.

The following distribution packages can be found there:

* domibusConnectorClient-4.0-RELEASE-Libraries
* domibusConnectorClient-4.0-RELEASE-Standalone

### domibusConnectorClient-4.0-RELEASE-Standalone

This client replaces the domibusConnector-Standalone prior to version 4.0-RELEASE. It is a completely self-running application that runs without having any other implementation in place.

The domibusConnectorClient-Standalone interoperates with the file system to receive and send messages from and to the domibusConnector.

It can also be started using a graphical user interface (GUI) to support reading and sending messages. This GUI also supports in setting the configuration.

The contents of the domibusConnectorClient-4.0-RELEASE-Standalone distribution package are the following:

|  |  |
| --- | --- |
| **File/directory** | **Description** |
| /bin (directory) | This directory contains the application JAR file “domibusConnectorClientRunnable.jar”. |
| /conf (directory) | This directory contains all the properties that need to be configured.  For more details see chapter [Configuration properties](#_Configuration_properties). |
| /documentation (directory) | Contains this guide. |
| /lib (directory) | All Java libraries that are needed to run the domibusConnectorClient-Standalone besides the “domibusConnectorClientRunnable.jar” can be found in this folder. |
| /messages | There are two subfolders underneath:   * “outgoing” * “incoming”   Both are empty folders where the default settings point to with the purpose to store received messages in “incoming” and to search for new messages in “outgoing”.  For more details see chapter [domibusConnectorClient-Standalone](#_domibusConnectorClient-Standalone) |
| DomibusConnectorClient.bat | This is a startup script to initialize the domibusConnectorClient-Standalone. It is built to run in MS Windows environments to initialize the application properties, set the Java Runtime and build the classpath. Running this script starts the domibusConnectorClient-Standalone in console mode. No GUI is started. |
| DomibusConnectorClient.sh | This is a startup script to initialize the domibusConnectorClient-Standalone. It is built to run in Unix-compatible environments to initialize the application properties, set the Java Runtime and build the classpath. Running this script starts the domibusConnectorClient-Standalone in console mode. No GUI is started. |
| DomibusConnectorClientGUI.bat | The same as “DomibusConnectorClient.bat”, but it additionally starts the GUI as well. |
| DomibusConnectorClientGUI.sh | The same as “DomibusConnectorClient.sh”, but it additionally starts the GUI as well. |

The functionalities of the domibusConnectorClient-Standalone and how to install and configure it is described in detail in chapter [domibusConnectorClient-Standalone](#_domibusConnectorClient-Standalone).

### domibusConnectorClient-4.0-RELEASE-Libraries

This is the distribution package holding all other variants of the domibusConnectorClient.

Its contents are the following:

|  |  |
| --- | --- |
| **File/directory** | **Description** |
| /conf (directory) | This directory contains all the properties that need to be configured.  For more details see chapter [Configuration properties](#_Configuration_properties). |
| /documentation (directory) | Contains this guide. |
| /libraries/domibusConnectorClientLibrary-4.0-RELEASE.jar | This is an integrate able library that can be used to be embedded into an already implemented application. It can also be a basis for new developments as well. |
| /libraries/domibusConnectorClientScheduler-4.0-RELEASE.jar | This library is an extension of the domibusConnectorClientLibrary. It enhances the functionality of the library with time triggered jobs that can be configured to run the functionalities of the library automatically triggered. |
| /libraries/domibusConnectorClientWebLib-4.0-RELEASE.jar | If your national application is a web application that runs inside of a web service container, the domibusConnectorClientWebLib offers the opportunity, to start a web service itself for the delivery of messages from the domibusConnector. The advantage of this variant is, that no jobs need to be triggered, as the connection between the domibusConnector and the client work as a push/push web service. |
| /libraries/domibusConnectorClient35Library-4.0-RELEASE.jar | This library only addresses implementers that had previous versions of the domibusConnector framework prior to 4.0-RELEASE in place. It offers access to the functionalities of the new domibusConnectorClient by using the old interfaces that were in place up to version 3.5.1 of the domibusConnector framework. All of those interfaces are marked as deprecated though. |
| /libraries/domibusConnectorClientWSLink-4.0-RELEASE.jar | This is a library all previous libraries depend on. It initializes the web service client that connects to the domibusConnector backend web service. |
| /libraries/domibusConnectorAPI-4.0-RELEASE.jar | This is a library all previous libraries depend on. It holds the interfaces of the domibusConnector. |

All the variants listed above as libraries, how they work and how they can be installed/integrated are described in separate chapters of this document.

# Certificate and Key-Store

To ensure the highest reasonable level of security, the domibusConnector uses web service security on different levels. The connection between the domibusConnectorClient and the domibusConnector underlies the OASIS ws-security standard. A detailed description of the standard can be found here:

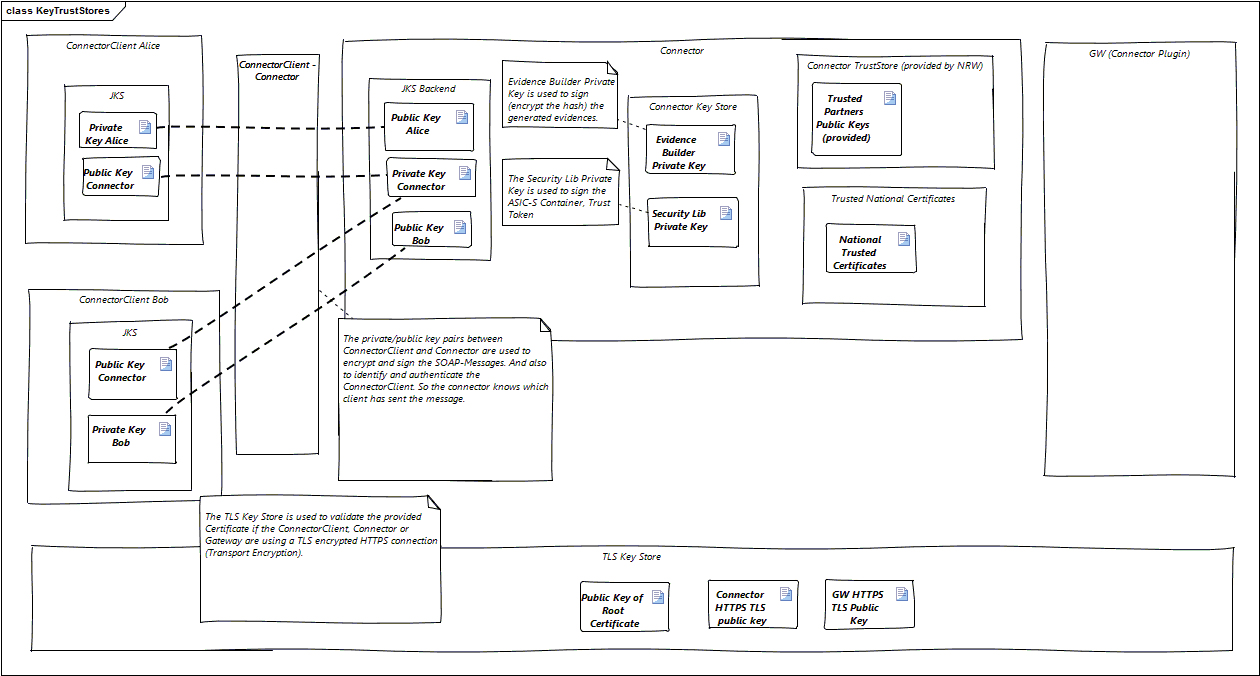
<https://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf>

Every backend client that connects to the domibusConnector needs to sign and encrypt messages.

For this purpose, the domibusConnectorClient needs a certificate to fulfil the following conditions:

* Authenticate the domibusConnectorClient at the backend of the domibusConnector.
* Sign the body and header of SOAP messages that are sent to the domibusConnector.
* Encrypt the body and header of SOAP messages that are sent to the domibusConnector.
* Decrypt the body and header of SOAP messages received from the domibusConnector.
* Verify the signature of the body and header of SOAP messages received from the domibusConnector.

The following graphic shows the environment specifications of the used certificates and stores:



## Certificate for the domibusConnectorClient

Every client needs a certificate to sign and decrypt the messages sent/received by/to the connector.

The type of certificate used must be compatible to be loaded into a Java-Keystore (JKS). During testphase certificates with RSA algorithm were used. A minimum keysize of 2048 is recommended.

The common name (CN) of the client certificate must match the configured backend name at the domibusConnector.

## Public Key of the domibusConnector

To be able to encrypt messages to the domibusConnector and to verify the signature of messages from the domibusConnector, the public key of the domibusConnector backend keystore is required. Details on the backend keystore and configuration of a backend client can be found in the “domiubsConnecto\_InstallationGuide.pdf” distributed together with the domibusConnector installed.

## domibusConnectorClient keystore

Both, the private key of the domibusConnectorClient certificate described above and the public key of the domibusConnector backend certificate configured on the domibusConnector the client should connect to, need to be added to a keystore.

The domibusConnectorClient therefore needs a Java-Keystore containing those keys.

This keystore needs to be configured in the “connector-client.properties” described in chapter [Configuration properties](#_Configuration_properties).

# Configuration properties

To give the domibusConnector the missing links about your environment, some properties have to be set in a property file.

Usually this is called “connector.properties”.

Also, the possibility is given to adopt the logging configuration. This gives the opportunity to control where logs are written at and what to log.

Example properties and an empty property file, as well as an example for logging configuration can be found in the distribution package at “documentation/properties”.

The properties in those file are all well described on what is expected there.

The variants on how to include the properties into your web server environment is dependent on what product you have in place.

For the web server products Apache Tomcat and BEA Weblogic this is described exemplarily in the Chapter [Deployment](#_Deployment).

# domibusConnectorClientLibrary

# domibusConnectorClientScheduler

# domibusConnectorClient-Standalone

# domibusConnectorClient35Library