Administering the EntireX XML/SOAP Listener

The EntireX XML/SOAP Listener is part of the EntireX XML/SOAP Runtime. It plugs the generated AAR file, including XMM files, into Web servers and so enables the EntireX XML/SOAP Runtime to send and receive XML documents using HTTP/HTTPS to/from a Web server. This component was formerly referred to as "XML Servlet".

This chapter covers the following topics:

- Introduction

- Configuring the XML/SOAP Listener

- XML/SOAP Listener with HTTP Basic Authentication and UsernameToken Authentication for EntireX Authentication

- Using Internationalization with the XML/SOAP Listener

Introduction

The EntireX XML/SOAP Listener requires a servlet-enabled Web server with an installation of Software AG Common Web Services Stack (WSS). See the separate Web Services Stack documentation. Client programs can access the XML/SOAP Runtime through HTTP/HTTPs interfaces provided in programming environments.

Configuring the XML/SOAP Listener

- Publishing the XML/SOAP Listener Initialization Parameters

- XML/SOAP Listener Initialization Parameters

- EntireX XML Init File

- External Configuration File for EntireX Web Services

Publishing the XML/SOAP Listener Initialization Parameters

The initialization parameters are set using the packaging wizard.

XML/SOAP Listener Initialization Parameters

Name in Web Services Wrapper	Parameter	Description
Default wait time	exx-default-waittime	Sets the value of the default wait time field to the argument (see `setDefaultWaittime` of class `BrokerService`).
Servlet internal sweep time	exx-sweeptime	Interval in which the servlet checks and frees unused resources. The default is 60 seconds.
Enable character reference	exx-use-characterreference	Enable/disable the character reference for the XML payload.
Behavior of non-conversation calls	exx-mep	The parameter indicates whether a non-conversational call is finalized with a logoff call to free Broker resource (default), or by means of timeout.

The default value for this parameter is "nonConv-with-logoff", which defines that a non-conversational call will finish with an additional logoff call (two calls per message). Set `exx-mep` to "nonConv-without-logoff" to specify that a non-conversational call will finish *without* logoff call (one call per message); Broker will clean up resources by means of timeout. |

EntireX XML Init File

The EntireX init file is generated by the packaging wizard that is called from the context menu of Software AG IDL or XMM files. It contains the XML/SOAP Listener initialization parameters.

External Configuration File for EntireX Web Services

Introduction

With an external configuration file you can redefine settings of some of the parameters for an EntireX Web service archive without modifying the EntireX Web service archive itself. This means you can use the same EntireX web service archive in different environments.

Using an External Configuration File

≫ **To use an external configuration file**

1. Define a name and a location for the external configuration file.

   In the parameter section of file *axis2.xml*, define a parameter "EntireX-XML-Listener" within a parameter "services". For the attribute "location" in parameter "services", specify an absolute or relative path to the external configuration file. File *axis2.xml* can be found in the *conf* directory or folder of the Web Services Stack Web application.

```
<parameter name="EntireX-XML-Listener">
   <parameter name="services" location="<path for file
                   overwriting settings of EntireX services>" />
</parameter>
```

Notes:

1. The path separator is a slash.
2. For determining the location of file *axis2.xml*, see *Configuration > Web Services Stack Runtime > Runtime Configuration* in the Web Services Stack documentation, also available under *http://documentation.softwareag.com > webMethods Product Line > Shared Services*.
3. The value of the location can contain operating system variables, for example `location="%USERPROFILE%/myconfig.xml"`.

2. Define services in the external configuration file.

   External configuration files are XML documents with a root element "serviceGroup". A service group is defined as a sequence of one or more services.

   To identify the service you are defining, specify an identifier for the attribute "name", for instance <service name= "service100">.

   To make common settings, that is, settings for all services, use an asterisk as identifier (<service name= "*">. Note that an individual setting can override a common setting.

   EntireX web service parameters that can be set are defined as subelements of "service". See the table below.

Parameter	Description
`<exx-brokerID>`	The broker ID to use.
`<exx-service>`	The service name is the triple set of server class/server name/service.
`<exx-userID>`	The user ID specified here is used for calling the broker.
`<exx-password>`	The user ID specified here is used for calling the broker.
`<exx-password-encryption>`	Specifies how the password is encrypted in the configuration file. Encryption is performed automatically when the configuration file is read for the first time.
`<exx-use-security>`	Possible values: true|false.
`<exx-encryption-level>`	Possible values: 0|1|2.
`<exx-rpc-userID>`	The RPC user ID specified here is used for Natural Security
`<exx-rpc-password>`	The RPC Password specified here is used for Natural Security.
`<exx-natural-security>`	Enable/Disable the Natural Security (true|false).
`<exx-natural-library>`	The Natural library to use.

Example of an External Configuration File

```
<?xml version="1.0" encoding="utf-8" ?>
<serviceGroup>
    <!--    Optional section for all EntireX services  -->
    <service name="*">
       <exx-brokerID>host:1234</exx-brokerID>
       <exx-service>RPC/SRV1/CALLNAT</exx-service>
    </service>

    <!-- service100 overwrites the service address -->
    <service name="service100">
       <exx-service>RPC/SRV2/CALLNAT</exx-service>
    </service>

    <!-- service101 adds library setting -->
    <service name="service101">
       <exx-natural-library>MYLIB</exx-natural-library>
    </service>
</serviceGroup>
```

XML/SOAP Listener with HTTP Basic Authentication and UsernameToken Authentication for EntireX Authentication

The XML/SOAP Listener allows you to use the user credentials from the incoming request by means of Basic Authentication or `UsernameToken`. The same credentials are used for EntireX Broker authentication and (Natural) RPC Server authentication. This means you need to make some settings for the EntireX Web service in Web Service Wizard and Configuration Editor.

Note:
`UsernameToken` is part of WS-Security. See WS-Security `UsernameToken` Specification. See also *Example: Setting up an EntireX Client to Consume a Secured Web Service* in the IDL Extractor for WSDL documentation.

The priority of credentials settings is as follows:

1. `exx-userID`, `exx-password`, `exx-rpc-userID`, `exx-rpc-password` (highest priority)

2. `UsernameToken`

3. Basic Authentication (lowest priority)

≫ **To use the XML/SOAP Listener with Basic Authentication and `UsernameToken` Authentication**

1. Select an IDL file or XMM file.

2. Choose **Web Service > Generate Web Service...**.

3. Disable check box **Use Defaults**.
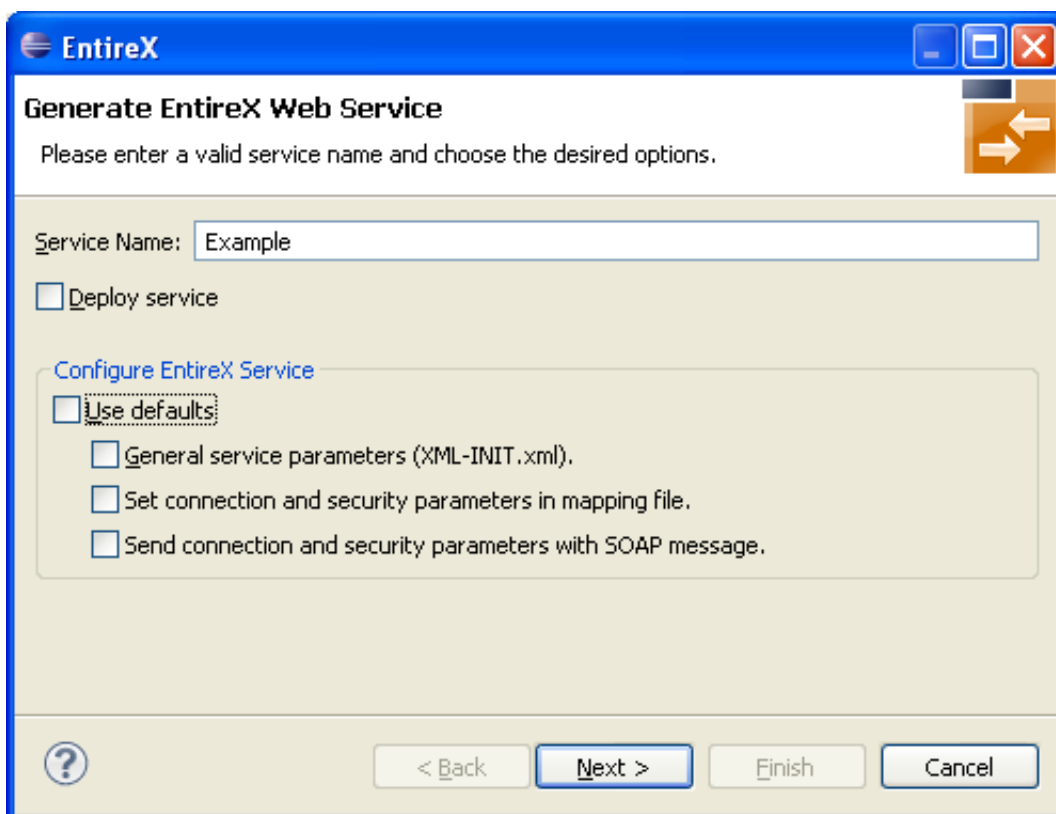
4. Enable at least **General service parameters...**.
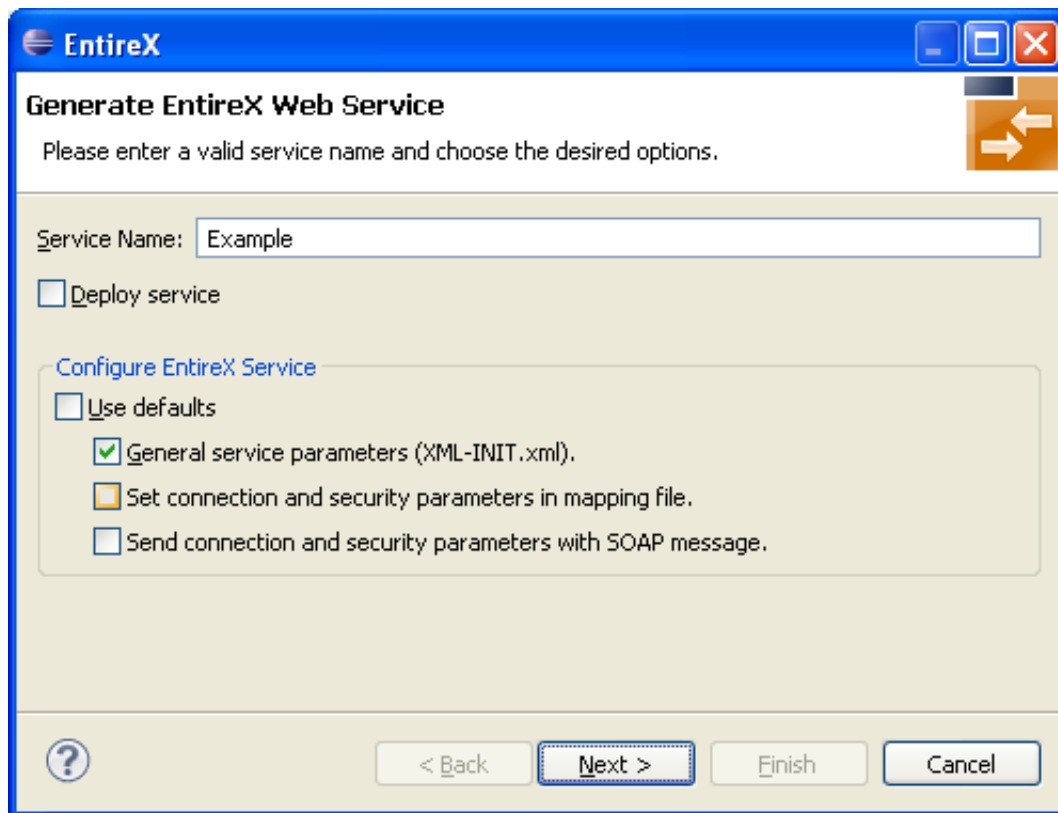
5. If using EntireX Security or Natural Security, enable **Set connection and security...** too.
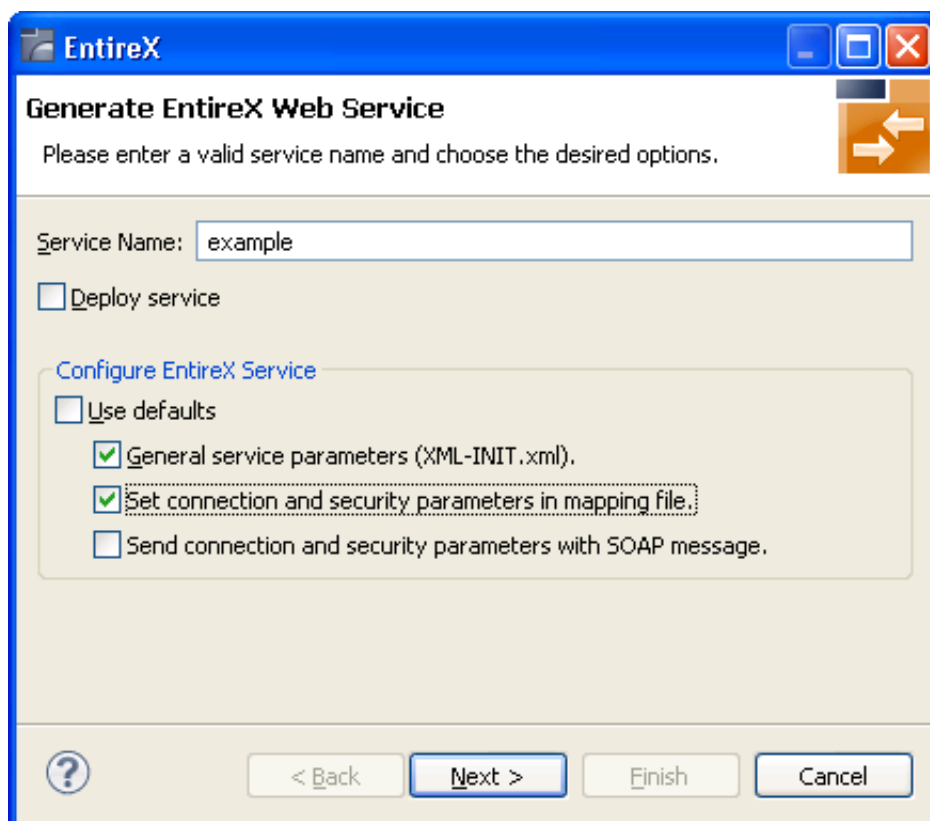
6. Press **Next**.

7. Enable the required authentication. In this example, both possibilities of web service authentication are enabled.

8. Press **Next**.

9. The page with XMM settings appears if it was selected before (step 5). Enable the required security (EntireX Security and/or Natural Logon).

10. Press **Next** and follow the wizard.

11. After generating the web service archive (extension "aar"), open the generated AAR file with the Configuration Editor (e.g. with double click).

For more information on the Configuration Editor see *Configuring Web Services*.

Using Internationalization with the XML/SOAP Listener

The XML/SOAP Listener supports both conversion and translation. See *Internationalization with EntireX* for more information.