Configuring User/Password Authentication for a Java Web Service

Release 650

HELP.BCWEBSERVICES_TUTORIALS

Copyright

Icons in Body Text

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Additional icons are used in SAP Library documentation to help you identify different types of information at a glance. For more information, see *Help on Help* → *General Information Classes and Information Classes for Business Information Warehouse* on the first page of any version of *SAP Library*.

Typographic Conventions

Type Style	Description
Example text	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
`Example text`	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example text	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
`EXAMPLE TEXT`	Keys on the keyboard, for example, `F2` or `ENTER`.

Configuring HTTP(S) User/Password Authentication for a Java Web Service

Task

This tutorial takes you through the development steps required for adding a user and password authentication to an existing Web service. A predefined Web service is used to check the credit limit of a customer (*CreditCheck*).

The Web service offers the simple function of checking the credit-worthiness of a customer using the customer number. If a customer number between 1001 and 1005 is passed, the credit limit of the customer is ascertained.

For further information about the Web service *CreditCheck,* refer to the tutorial Creating a Web Service.

Objectives

By the end of this tutorial, you will be able to:

✓ Create an HTTP(S) user/password authentication for an existing Web service.

Prerequisites

☐ Your SAP J2EE Engine is configured for the use of SSL.

You must install the SAP Java Cryptographic toolkit for this. With J2SE1.4, additional Sun Microsystems policy files are required; these must be copied to the directory *<java-home>\jre\lib\security*. For more information, refer to the section Deploying the SAP Java Cryptographic Toolkit.

Systems, Installations, and Authorizations

☐ You have installed the SAP NetWeaver Developer Studio and configured it for use of the SAP J2EE Engine.

☐ You can access the SAP J2EE Engine using the Visual Administrator.

Knowledge

☐ You can create and deploy a Web service based on an EJB.

Next Step:

Importing Projects [Page 6]

Importing Projects

The following Web Dynpro projects are available for this tutorial:

- EJB project *CreditLimitCheck* (on which the Web service is based)
- EAR project *CreditLimitCheckEAR* (which contains the Web service *CreditCheck*)

The projects are available in the file JAVA_SERVER_RAW.zip. To download the file, click here.

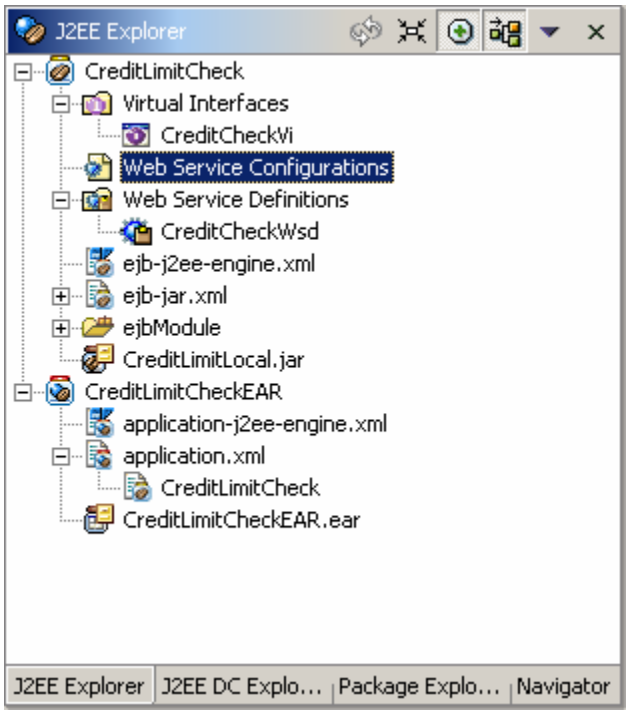
Procedure

Importing Projects into the SAP NetWeaver Developer Studio

1. Save the ZIP file to any directory on your local hard disk or directly in the work area of the SAP NetWeaver Developer Studio.
2. Unpack the ZIP file.
3. Start the SAP NetWeaver Developer Studio and open the Web service perspective.
4. Import the projects *CreditLimitCheck* and *CreditLimitCheckEAR* by choosing *File* → *Import* → *Existing Project into Workspace*.

Result

Once you have imported the project into the SAP Developer Studio, the following structure is displayed in the *EJB Explorer*.



Next Step:

Adding a Web Service Configuration [Page 7]

Adding a Web Service Configuration

You can use a Web service configuration to specify the runtime properties of a Web service. At least one configuration is assigned to each Web service. When creating a proxy, a logical port is created for each configuration.

You will learn how to create a new Web service configuration - based on an existing Web service - and how to specify the settings for the activation of the user/password authentication via HTTP/HTTPS.

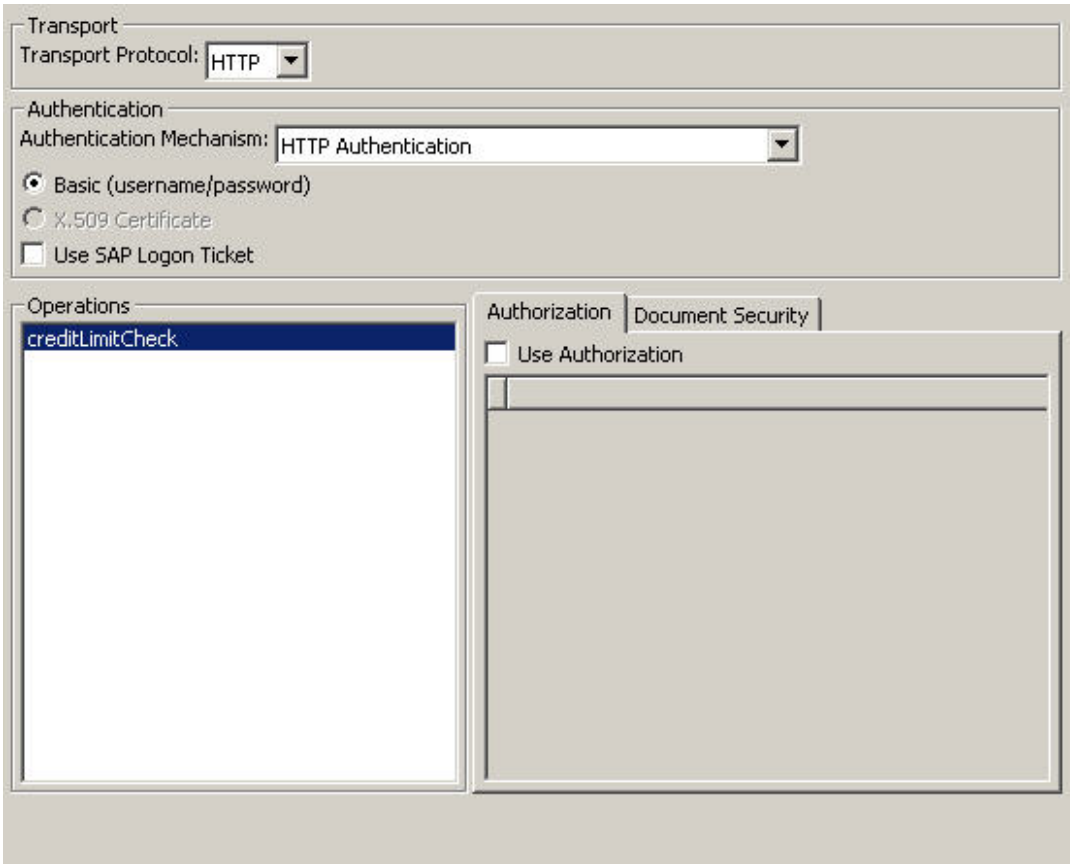
Prerequisites

- ☐ The structure of your projects *CreditLimitCheck* and *CreditLimitCheckEar* is displayed in the *EJB Explorer*.

Procedure

Creating and Configuring a Web Service Configuration for HTTP

1. To create a new configuration, open the *WS Deployment Descriptor Editor* by double-clicking the node *CreditLimitCheck - Web Service Configurations*. In the tree structure, select the entry *Web Service Configurations – CreditCheck,* and choose *Add.*
2. The dialog box *Web Services* appears. Enter the name of the new configuration (BASIC) in the field *Configuration Name* and choose *Finish*.
3. Select the node *Web Service Configurations – CreditCheck – BASIC – Security* to display the security settings of the new configuration.
4. Select **`HTTP`** as the *Transport Protocol* and **`HTTP Authentication`** as the *Authentication Mechanism*. **`Basic (username/password)`** is already selected as the default value. Accept the default value and save your settings with `CTRL-S`.



Creating and Configuring a Web Service Configuration for HTTPS

1. Create a new Web service configuration as described in the first step of the section *Creating and Configuring a Web Service Configuration for HTTP*.
2. Name the configuration BASIC_SSL and confirm.

   Display the security settings of the new configuration.

   Select **HTTPS** as the *Transport Protocol* and **HTTP Authentication** as the *Authentication Mechanism*. Leave the remaining settings unchanged and save your settings.

Deploying a Modified Web Service

1. In the context menu for the project *CreditLimitCheck*, choose *Build EJB Archive*.
2. In the context menu for the project *CreditLimitCheckEAR*, choose *Build Application Archive*.
3. In the context menu of the node *CreditLimitCheckEAR – CreditLimitCheckEAR.ear*, call the function *Deploy to j2ee engine* and wait for the message in the format *hh:mm:ss [nnn] Finished Deployment [more]* displayed in the *Deploy Output View*.

Result

Based on the Web service *CreditCheck*, you created the new configuration *BASIC* or *BASIC_SSL* and made it accessible by deploying the Web service on the J2EE Engine. If the Web service is called using one of these configurations, a user/password authentication is required from the Web service consumer before the Web service can be used. HTTP and HTTPS are used for the data transfer.

The newly created configurations are displayed in the *WS Deployment Descriptor Editor* under the node *Web Service Configurations – CreditCheck*.

Choose the Web service *CreditCheck* in the Web Service Navigator. Open the home page via the context menu. Your changes are displayed:

Deploy-time features for Binding: BASICBinding		
Feature	Property	
	Name	*Value*
http://www.sap.com/webas/630/soap/features/authentication	AuthenticationMechanism	*HTTP*
	AuthenticationMethod	*BasicAuth*
	SupportsSSO2Authentication	*false*

Deploy-time features for Binding: BASIC_SSLBinding		
Feature	Property	
	Name	*Value*
http://www.sap.com/webas/630/soap/features/transportguarantee	TLSType	*SSL*
http://www.sap.com/webas/630/soap/features/authentication	AuthenticationMechanism	*HTTP*
	AuthenticationMethod	*BasicAuth*
	SupportsSSO2Authentication	*false*

Next Step:

Exporting the Certificate Used for SSL (Optional) [Page 9]

Exporting the Certificate Used for SSL (Optional)

When establishing a secure connection to a Web service client, the server sends a certificate for its authentication. Before the certificate can be checked for its trustworthiness by the Web service client, it must be exported from the J2EE Engine of the server for it then to be made available to the consumer of the service.

If the Web service and its client are located on the same SAP J2EE Engine, this step is not necessary.

Prerequisites

☐ The SSL provider of the used dispatcher was started.

Procedure

Exporting the Certificate

1. Start the Visual Administrator and log on to the SAP J2EE Engine that contains your Web service.
2. Select the node <*SID*> - *Server … - Services – SSL Provider,* where <SID> is the system ID of the J2EE Engine.
3. In the right window, activate the *Runtime* tab and double-click the used dispatcher in the list.
4. In the field *Configuration*, select the host/port configuration that is to be used for calling the Web service and activate the tab *Server Identity* in the bottom right window. Make a note of the text that appears in the field *Enabled Credentials.*
5. In the left window, select the node <*SID*> - *Server … - Services – Key Storage* and activate the tabs *Runtime* (top) and *Data* (bottom) in the right window.
6. In the list *Views*, select the entry *service_ssl* and in the list *Entries*, select the row with the noted key name and the extension **-cert**.
7. Choose *Export* in the area *Entry* (bottom right) and, in the dialog box that appears, save the certificate as a crt file in your file system.

Result

The certificate used by the server for the SSL handshake was exported into a crt file. It can now be used by a Web service client to check the identity of the called server.