How to configure a Service Provider

This article describes how to configure a Web Service provider using the SAP ABAP SOA Manager (transaction SOAMANAGER). In order to provide a compact overview of the steps, this article does not cover all the available configuration options in detail. These details to be explained are left to more specific articles in the SDN. In this article, we give an overview and describe the configuration options that are essential to configure an ABAP WS service provider.

The result of a Web service configuration process is to define a so called binding. The W3C consortium defines a binding as follows (shortened): A binding defines message format and protocol details for operations and messages defined by a particular portType; a portType is defined as a set of abstract operations, where each operation refers to an input and output message.

The portType is often referred in the ABAP environment as the design time (DT) data of a Web Service. On ABAP WS provider side, the service definition is the representative of the DT data.

Several bindings can be grouped into one service implementing alternative ways for accessing a Web Service. A service must have at least one binding.

The binding information is used to support the configuration of WS consumers in the form of a WSDL document. Each configured ABAP WS provider configuration (a binding) can describe its settings via a WSDL document. A WSDL document describes a WS in an XML document format. It gets mostly used to pass the binding’s information to a WS consumer to support the setup of the consumer’s binding, which is referenced via a Logical Port.

Without a binding, it is not possible to call a Web Service. From the configuration perspective each binding represents a callable Web Service.

Prerequisites

To define a Web Service, a service definition gets used on ABAP systems. Each Web Service provider binding in ABAP is referring to a service definition as its basis. An ABAP service definition is practically a wrapper object keeping a link to the Web Service implementing class or function module and the DT configuration. The DT configuration is a collection of requirements for a Web Service which are already known at DT. It specify basic attributes (e.g. if operations are synchronously or asynchronously processed) and minimum requirements for a binding it has to keep for accessing the Web Service (e.g. the minimum authentication level required).

Procedure

To configure a Web Service provider, perform the following steps.

1. Start SOA Manager
   Use the transaction code SOAMANAGER to start the ABAP SOA Manager.

   The SOA Manager is an ABAP WebDynpro application. It is like each regular WD application using an ICF node for accessing
It. You may use transaction SICF to check that the ICF node for SOA Manager transaction (/sap/bc/webdynpro/sap/appl_soap_management) is active.

More information about this subject: SAP Note 1124553 and 1088717.

2. From the main screen of the SOA Manager, go to the Service Administration tab

3. Choose Web Service Configuration

4. Select Service Definition from the Search by menu, and specify a service definition

Here in this configuration example, we will use the service definition SRT_TEST_PROVIDER.

5. Select the Go button

The service definition is displayed in the Search Results

6. Select the row and choose Apply Selection

7. Existing services and bindings are displayed.
You may see multiple services each with at least one binding. If you have a service with multiple bindings, then the same service provides via different runtime configurations alternative ways for accessing the service.

8. Choose Create
A dialog box is displayed, where the identifying information about a new provider binding must get specified. You have to specify a service name, a binding name, and a service description.

The configuration view is displayed where the details for the new binding get specified. You may need to scroll down in the browser window to see this view completely.

At this stage, the new binding is not configured and has not yet been saved. Only default settings are selected where applicable. Note, that the security settings do not have default settings for the new binding and thus must be specified explicitly.

9. Go to the Provider Security tab
In this view the security settings on the new binding get specified.

The provider security settings displayed fall into two main categories:

- the communication security, i.e. settings dealing with the security while the message is on the network, and
- the authentication method to be used for identifying the user environment on the Web Service provider system.

For the communication security settings in this example, we use the ABAP NGAP (SAP_BASIS 8.x) codeline options on this UI view. You can see the following alternatives for the provider binding:
None: The HTTP protocol gets used, i.e. there is no encryption taking place on the WS messages on the network
SSL: The HTTPS protocol gets used (SSL = Secure Socket Layer), i.e. the HTTP message containing the WS message gets encrypted. This provides for integrity and privacy protection. SSL also provides server side authentication (the server identifies itself to the provider). The client side authentication (on connection establishment the consumer identifies itself to the provider) can be enforced by selecting the X.509 Client Certificate option in the Authentication section below. The combination of both provides a mutual authentication.

The following options are mainly implementations of OASIS security standards like WS Security and WS Secure Conversation.

- Symmetric Message Signature / Encryption: Secures the data exchange on the WS (SOAP) message level. The SOAP message will be signed and encrypted where both the signature and the encryption is calculated using a symmetric session key. This key is created by the consumer to be used for the complete request/response cycle. The symmetric key itself will be transferred in encrypted format using the provider’s public key of its Public-Private-Key-pair.
- Asymmetric Message Signature / Encryption: Secures the data exchange on the WS (SOAP) message level. The SOAP message will be signed and encrypted. The signature is computed using the consumer’s private asymmetric key. The encryption is calculated using a symmetric session key which is created by the consumer for the request to the provider. The provider will generate a new symmetric key for the response encryption. The symmetric key itself will be transferred in encrypted format using the provider’s asymmetric public key.

In addition, you can specify via checkboxes, if you want to switch on secure conversation and extended header and signature protection.

- Secure Conversation: If Secure Conversation is selected in addition to another communication security option, the SOAP message will be signed and encrypted using a symmetric session key. This key will be used for multiple request/response cycles till the Secure Conversation session gets terminated, e.g. on a termination of a WS Reliable Message sequence. The chosen communication security in together with the chosen authentication method will be used to establish the Secure Conversation session.
- Extended header and signature protection: This will provide additional means of protecting SOAP message like Signature Encryption, Signature confirmation and SOAP Header encryption

For the authentication settings, you have the following options available to specify the authentication method or methods to be used.

- No Authentication: This specifies that your service is considered to be a public WS. The authentication is done with a fixed user account specified as additional input on choosing this option. The user account will be stored in the binding’s ICF node as service user.
- Transport Channel Authentication: The authentication is done by means of the transport channel. You specify a User ID/Password, a X.509 client certificate or a SAP assertion ticket authentication. The authentication information for User ID/Password and SAP assertion ticket authentication will be transferred as http header.

Using User ID and Password Authentication
X.509 SSL Client Certificate only works in combination with Communication Security SSL where authentication is performed inside the SSL Layer.

- Message Authentication: The authentication is done in the WS message itself as defined in the WS-Security standard. The authentication information is transferred within the SOAP security header. You can specify these kinds of message authentication:
  - User ID/password (Username Token),
  - X.509 certificate (X509 Token), or
  - Single Sign On using SAML (SAML Token).

In case you selected option Single Sign On using SAML as message based authentication method, this table shows what subject confirmation method the provider will expect. This is particularly important when calling a SAP Web Service with a non-SAP consumer. The SAML subject confirmation method mainly depends on the chosen Communication Security setting.

<table>
<thead>
<tr>
<th>Communication Security Setting</th>
<th>Sender-Vouches</th>
<th>Holder-of-Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>None (HTTP)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSL (HTTPS, Transport Channel Security)</td>
<td>✔️</td>
<td></td>
</tr>
<tr>
<td>Asymmetric Message Signature/Encryption</td>
<td>✔️</td>
<td></td>
</tr>
<tr>
<td>SSL + Secure Token Service Issuer selected</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Symmetric Message Signature/Encryption</td>
<td></td>
<td>✔️</td>
</tr>
</tbody>
</table>

The communication security setting Symmetric Message Signature/Encryption always mean Holder-of-Key subject confirmation. All other
settings default to Sender-Vouches.
If you however selected SSL in conjunction with a Token Issuer in section Secure Token Service Settings, the Holder-of-Key subject confirmation method will be expected by the provider.

For more information about Single Sign-On mechanisms and the distinction between Transport Channel authentication and Message authentication consult the following SAP documentation link:
Single Sign-On for Web Services
You can find configuration examples via this link:
Configuration Examples for AS ABAP

The security features do not run "out-of-the-box", i.e. you have to prepare the ABAP system to use the security mechanisms offered in this view. Especially the message based authentication need preparation via the report WSS_SETUP to run properly. Refer to the SAP security online documentation for more details on the setup.

You need to select at least one option from the authentication method section before the service can be saved. You may even select multiple methods, so the bindings serves different authentications simultaneously.

You cannot mix transport level and message based authentication methods within one binding. If you want to make the service available for both transport level and message level authentication, you have to specify different bindings for the service.

These settings constitute the minimum security settings required to call this Web Service.

The settings permitted in the provider security settings are based on the security settings of the service definition specified in ABAP Development Workbench.

11. Save the binding
When the binding gets saved, it is displayed in the list of configurations belonging to the selected service definition. The new binding is now accessible under its access URL by WS consumers. Also it is now possible that this binding is providing a WSDL document to describe itself.
12. To display the Access URL for the binding, go to the Transport Settings tab

The access URL path is displayed in the Calculated Access URL field in the Transport Settings tab. You can also define an alternative URL path for the calls in case your provider can only use a fixed access path. The access path that you set in the field Alternative Access Path adds one additional, alternative access path to the Web Service. Due to the restrictions of the ICF framework, that alternative access path can only be used once in the system (in NGAP per SAP client).
Result

You have now configured a Web Service provider (aka. "Web Service"), and this provider can now be called by a consumer. For the Web Service a WSDL document is also now available; the WSDL-URL can be seen via the Overview tab and, if maintained, you can test the Web Service via a testpage of the SAP J2EE engine.

More information:
How to find Web Service WSDL in SOA Manager
Web Service Testing Using the J2EE Server Web Service Navigator

More Information

SAP Note 2175422 - Web service provider configuration in transaction SOAMANAGER [Video]
SAP Note 1124553 - Inactive ICF services of Web Service runtime
SAP Note 1319507 - Overview: Analysis of ABAP Web Service Configuration