WCF - SERVICE BINDING

WCF service binding is a set of several elements in which each element defines the way the service is communicating with the client. A transport element and a message encoding element are the two most vital components of each binding. In this chapter, we will discuss various WCF service bindings that are commonly used.

Basic Binding

Basic binding is offered by the BasicHttpBinding class. It uses the HTTP protocol to transport and represent a WCF service as an ASP.NET web service *ASMXwebservice*, so that old clients who use ASMX web services can consume the new services conveniently.

Basic binding is set as default binding in a WCF web service enabled by Silverlight and is a standard binding for communications in web service style. It does not support reliable messaging.

Given below is a code snippet depicting the default settings for basic binding.

```
<basicHttpBinding>

    <binding name="basicHttpBindingDefaults" allowCookies="false"
        bypassProxyOnLocal="false" hostNameComparisonMode="StrongWildcard"
        maxBufferPoolSize="524288" maxBufferSize="65536"
        maxReceivedMessageSize="65536" messageEncoding="Text" proxyAddress=""
        textEncoding="utf-8" transferMode="Buffer" useDefaultWebProxy="true"
        closeTimeout="00:01:00" openTimeout="00:01:00" receiveTimeout="00:10:00"
        sendTimeout="00:01:00">

        <readerQuotas maxArrayLength="16384" maxBytesPerRead="4096"
            maxDepth="32"
            maxNameTableCharCount="16384" maxStringContentLength="8192"/>

        <security mode="None">
            <transport clientCredentialType="None" proxyCredentialType="None" realm=""/>
            <message algorithmSuite="Basic256" clientCredentialType="UserName" />
        </security>

    </binding>

</basicHttpBinding>
```

The above default settings have their obvious limitations, as the message size is limited and there is no security mode. However, customization of basic binding solves this problem like the one below.

```
<basicHttpBinding>

    <binding name="basicHttpSecure" maxBufferSize="100000"
maxReceivedMessageSize="100000">

        <readerQuotas maxArrayLength="100000" maxStringContentLength="100000"/>
        <security mode="TransportWithMessageCredential" />

    </binding>

</basicHttpBinding>
```

Web Service Binding

Web Service *WS* binding is provided by the WSHttpBinding class. It is quite similar to basic binding and uses the same protocols for transport, but offers several WS–* specifications such as WS–Reliable Messaging, WS–Transactions, WS–Security, and many more. In a nutshell, WSHttpBinding equals to the sum of basicHttpBinding and WS–* specifications. Given below is a code snippet

depicting the default settings for WS Binding:

```xml
<wsHttpBinding>

    <binding name="wsHttpBindingDefaults" allowCookies="false"
        bypassProxyOnLocal="false" closeTimeout="00:01:00"
        hostNameComparisonMode="StrongWildcard"
        maxBufferPoolSize="524288" maxReceivedMessageSize="65536"
        messageEncoding="Text" openTimeout="00:01:00"
        receiveTimeout="00:10:00" proxyAddress="" sendTimeout="00:01:00"
        textEncoding="utf-8" transactionFlow="false"
        useDefaultWebProxy="true" >

        <readerQuotas maxArrayLength="16384" maxBytesPerRead="4096"
            maxDepth="32" maxNameTableCharCount="16384"
            maxStringContentLength="8192"/>

        <reliableSession enabled="false" ordered="true"
            inactivityTimeout="oo:10:00" />

        <security mode="Message">

            <message algorithmSuite="Basic256" clientCredentialType="Windows"
                esatalishSecurityContext="true"
                negotiateServiceCredential="true" />

            <transport clientCredentialType="Windows"
                proxyCredentialType="None" realm=""/>

        </security>

    </binding>

</wsHttpBinding>
```

IPC Binding

IPC binding makes use of named pipe and is offered by the netNamedPipeBinding class. This is the fastest binding and the most secure one amidst all the available bindings. Although message-level security is not supported here, messages are secure by default because of a robust transport security. Given below is the code snippet depicting the default settings for IPC binding:

```xml
<netNamedPipeBinding>

    <binding name="netPipeDefaults" closeTimeout="00:01:00"
        hostNameComparisonMode="StrongWildcard" maxBufferPoolSize="524288"
        maxBufferSize="65536" maxConnections="10"
        maxReceivedMessageSize="65536" openTimeout="00:01:00"
        receiveTimeout="00:10:00" sendTimeout="00:01:00" transactionFlow="false"
        transactionProtocol="OleTransactions" transferMode="Buffered">

        <readerQuotas maxArrayLength="16384" maxBytesPerRead="4096"
            maxDepth="32" maxNameTableCharCount="16384"
            maxStringContentLength="8192"/>

        <security mode="Transport">
        </security>

    </binding>

</netNamedPipeBinding>
```

Other Types of Service Bindings

- **TCP Binding** - Provided by the NetTCPBinding class, this binding makes use of the TCP protocol for communication within the same network and does the message encoding in binary format. This binding is considered as the most reliable in contrast to others.

- **WS Dual Binding** - This type of binding is more like WSHttpBinding with the only exception that it facilitates bidirectional communication, i.e., messages can be sent and received by both clients and services. It is offered by the WSDualHttpBinding class.

- **Web binding** - Web binding is designed to represent WCF services in the form of HTTP requests by the use of HTTP-GET, HTTP-POST, etc. It is offered by the WebHttpBinding class and is used commonly with social networks.

- **MSMQ Binding** - It is offered by the NetMsmqBinding class and is used to provide solutions in case the service processes a message at a distinct time than that sent by the client. MSMQ binding makes use of MSMQ for transportation and provides support to detached message queued. MSMQ is an implementation for message queuing offered by Microsoft.

- **Federated WS Binding** - It is a specific form of WS binding and offers support for federated security. It is offered by the WSFederationHttpBinding class.

- **Peer Network Binding** - Offered by the NetPeerTCPBinding class, it is mainly used in file sharing systems. It uses TCP protocol but makes use of peer networking as transport. In this networking, each machine $node$ acts as a client and a server to the other nodes. Peer network binding is used in file sharing systems like torrent.

- **MSMQ Integration Binding** - Offered by the MsmqIntegrationBinding class, it helps communicate with existing systems that communicate via MSMQ $Microsoft Message Queuing$.

Apart from these, it is also possible to create custom bindings. However, since it is possible to tweak the configuration properties of each WCF binding, the need for creating custom bindings arises rarely.

Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js