Web Service Notes

This document will contain notes on the issues in the design of the web service.

# Operations we will provide.

The service can have offer to the client operations that provides access to data. The following is a list of some of the operations.

* Clients order list.
* Product list.

Other information gathering steps follow:

* Map the data to be sent in the message.

# Protection.

The Internet is a potentially hostile environment, and you must treat all communications passing over it with the utmost suspicion. This section deals with the security model available for our use. Based on the selection there will be administration work on both the server and client.

A service must provide a means for a user to identify herself and then prove that identity. Protection for an HTTP service is either at the transport or message level.

## Windows

### Transport

In this mode, message security is provided by using HTTPS . You must configure SSL for the service by using a certificate. The client authenticates the service by using the service’s SSL certificate.

### Message

In this mode, the service uses SOAP message-level security to encrypt the message. The service must have a certificate installed, and the client uses the public key from the service’s certificate to perform the encryption. The service can send the certificate containing its public key at the start of the message exchange, or an administrator can install the service certificate on the client computer before the client application (in which case you must specify how to locate the service certificate in the client certificate store by adding a service behavior using the <serviceCredentials> element to the client configuration file). You will learn more about this in Chapter 5.

See chapter 4 in Wcf Step by Step. This chapter has information on the tracing and examining of the messages, very good.

The following shows the bindings used for a service.

BasicHttpBinding--This binding conforms to the WS-I Basic Profile 1.1 (for maximum backward compatibility with older Web services and client applications). It can use the HTTP and HTTPS transport protocols and encodes messages as XML text. Use this binding to build services that are compatible with client applications previously developed to access ASMX-based Web services.

WS2007HttpBinding--This binding conforms to the WS-\* specifications that support distributed transactions and secure, reliable sessions. It supports the HTTP and HTTPS transport protocols. Messages can be encoded as XML text or by using the Message Transmission Optimization Mechanism (MTOM). MTOM is an efficient encoding mechanism for transporting messages that contain binary data. You will learn more about MTOM in Chapter 13. See the exercise starting on pg 142 “Protecting an HTTP service at the Message Level”.

The following are my notes on the tests I was able to perform following along in the book.

The following example uses the port 8000 in the book WCF Step by Step

page 75-76 shows you how to reserve http port 8000.

To reserve port 8000 enter the following cmd for the TCP test

netsh http add urlacl url=http://+:8000/ user=UserName UserName for me john

And to remove the port reservation enter the following

netsh http delete urlacl url=http://+:8000/

================ HTTPS test =========================

To configure the SSL Certificate

Configure the HTTP endpoint with an SSL Certificate page 137-142

makecert -sr LocalMachine -ss My -n CN=HTTPS-Server -sky exchange -sk HTTPS-Key

Once the certificate is created you can view it using MMC and note the thumbprint

you need to enter that in the following command, pg 138.

netsh http add sslcert ipport=0.0.0.0:8000 certhash=‎7d57a094ab19560e7e8c63f66f5f3e6ee02b6012 appid={0293AE3D-9B2E-42C4-B3A9-B9B57C4A16B1}

Note: Enter the thumbprint with no spaces. You can use guidgen in the tools folder to create the appid key.

delete the entry above and enter the new https one

netsh http add urlacl url=https://+:8000/ user=john UserName for me john

To view all of the reservations

netsh http show urlacl

netsh http show sslcert

**Important pg 138** Certificates that you create by using the makecert utility should not be used in a production environment as they are not certified by a verifiable certification authority. Remember that the service uses this certificate to prove its identity. The client must be able to trust that this certificate was created by a reliable source that can verify the veracity of the service. When deploying a production service, you should obtain your certificates from a recognized certification authority, such as **VeriSign** or **Thawte**. Alternatively, you can use **Active Directory Certificate Services**, which enables an enterprise to generate its own certificates.

**Warning pg 141** When a client application receives a certificate from a server, the WCF runtime attempts to ascertain that the certificate is valid and that the authority that issued it is trusted. The WCF runtime will fail this check when using the certificate that you have just installed. You can force the WCF runtime to override this check and allow the certificate to be used.

## 

## 

## Over the internet

There are three approaches for handling the security over the internet.

#### Authenticating over the internet

* Configure IIS to handle HTTPS, see the steps starting on pg 167.
* Use the IIS Manager console to create the test certificate much like the one from chapter 4 but for the internet you do not use makecert. The same warnings apply to this certificate as well, pg 167.
* Create the WCF Web application, pg 169.
* Create the ASP.Net web site to host the service, pg 170.
* Import the code, pg 170.
* Configure the WCF service activation and bindings, pg 171.
* I added back the Service.svc file.
* I had to turn on HTTP-Activation in Programs Features in Windows 8.
* I had to set aspNetCompatibilityEnabled to false in the configuration file.

Each of the following methods use different approaches to handle the client security. I am not going through each.

### Authenticating Users and Services in an Internet Environment

Create the ASP.NET Membership Provider (to store a list of users and their credentials in a SQL Server database) together with the ASP .NET Role Provider (to associate users with roles). Alternatively, you can use the Authorization Store Role Provider to record users and roles in XML files.

### Authenticating and Authorizing Users by Using Certificates

Using a public key infrastructure (PKI). PKI is based on pairs of keys (a key is a long sequence of random numbers): a public key that you can use to encrypt messages, and a private key that you can use to decrypt them again. Use 128 bits or more.

You can send a third party a copy of your public key. The third party can encrypt their messages using this key and transmit them to you. You can decrypt these messages using your private key.

You can request a pair of keys in a certificate from a certification authority, or CA such as **VeriSign** or **Thawte**. Alternatively, you can use **Active Directory Certificate Services**. The certificate contains a public key and a private key (you usually have to pay for this service) . The certificate also contains other bits of identity information about you and about the CA itself.

### 

### 

### Authenticating Service Messages by Using a Certificate

Using the HTTPS protocol with a service gives a client application a reasonable degree of confidence that communications with the service are secure. You implement message-level security with mutual authentication in place of using transport-level security.

Exercises on page 196.