KillTest

質量更高　服務更好

學習資料

http://www.killtest.net

一年免費更新服務

Exam : **70-554(C#)**

Title : UPGRADE:MCSD MS.NET Skills to MCPD Entpse App Dvlpr Pt2

Version : DEMO

1. A client application calls a Web service named Math. Both the client application and Math are configured with a Web Services Enhancements (WSE) 3.0 policy named Secure to validate anonymous access for certificate security. A Web reference to the Math Web service is added to the client application's project using Microsoft Visual Studio 2005. When the client application is built and executed, a SoapException exception is thrown with the following message.

The security requirements are not met because the security headeris not included in the incoming message.

You need to ensure that the application runs without throwing the SoapException exception. What should you do?

A. Add the following attribute to the Math proxy class definition.[Microsoft.Web.Services3.Policy("Secure")]

B. Set the Name property for the WebServiceBindingAttribute attribute on the Math proxy class definition to MathWseSoap.

C. Add the following attribute to the Math proxy class definition.[Microsoft.Web.Services3. Policy("anonymousForCertificateSecurity")]

D. Modify the Math proxy class so that it derives from the following protocol.System.Web.Services. Protocols.SoapHttpClientProtocol

Answer: A

2. An application uses a Web service named DynamicImage. DynamicImage encrypts all response messages and then compresses these response messages.

The application uses a class named Contoso.Decrypt to decrypt the response messages and it uses a class named Contoso.Decompress to decompress the response messages. Both classes can be used as SOAP extensions. You need to ensure that the response messages from the DynamicImage Web service methods are decompressed and decrypted properly. What should you do?

A. Add the following XML to the <webServices> section of your App.config file.<soapExtensionTypes><add type="Contoso.Decompress, Version=1.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"
priority="1" group="0"/><add type="Contoso.Decrypt,
Version=1.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" priority="2" group="1"/></soapExtensionTypes>

B. Add the following XML to the <webServices> section of your App.config file.<soapExtensionTypes><add type="Contoso.Decompress, Version=1.0.0.0,
Culture=neutral,
PublicKeyToken=31bf3856ad364e35"

priority="2" group="0"/><add type="Contoso.Decrypt, Version=1.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" priority="1" group="0"/></soapExtensionTypes>

C. Add the following XML to the <webServices> section of your App.config file.<soapExtensionTypes><add type="Contoso.Decompress, Version=1.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" priority="2" group="1"/><add type="Contoso.Decrypt, Version=1.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" priority="1" group="0"/></soapExtensionTypes>

D. Add the following XML to the <webServices> section of your App.config file.<soapExtensionTypes><add type="Contoso.Decompress, Version=1.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" priority="1" group="1"/><add type="Contoso.Decrypt, Version=1.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" priority="2" group="0"/></soapExtensionTypes>

Answer: A

3. A Web service named Math uses Web Service Enhancements (WSE) 3.0 policy assertions that are defined in the policy configuration file to implement security. The Web service authenticates access to the Web methods using the UsernameOverTransportAssertion policy assertion. A client application must call a

Web method on the Math Web service. When the client application is executed, it throws an InvalidOperationException exception that displays the following text.

Unable to determine client token to use. Client token type

requested was 'Microsoft.Web.Services3.Security.Tokens.UsernameToken'.

The following code is used to invoke the Web method. (Line numbers are included for reference only.)

01 Math ws = new Math();

02 ws.SetPolicy("Secure");

03 int result = ws.Add(3, 4);

You need to ensure that the Web methods can be invoked without causing the client application to throw the InvalidOperationException exception. What should you do?

A. Add the following code between lines 02 and 03.ws. Credentials = System.Net.CredentialCache.DefaultCredentials;

B. Add the following code between lines 02 and 03.ws.Credentials = new System.Net.NetworkCredentials(userId, password);

C. Add the following code between lines 02 and 03.ws.SetServiceCredential<UsernameToken>(new

UsernameToken(userId, password));

D. Add the following code between lines 02 and 03.ws.SetClientCredential<UsernameToken>(new UsernameToken(userId, password));

Answer: D

4. You write an XML Web service. The XML Web service contains Web methods that return large amounts of non-sensitive public data. The data is transferred over the Internet. You need to be able to detect if the data was tampered with during transit. The implementation must be configurable at run time.

Your solution must minimize the impact on the performance of the Web service. What should you do?

A. Configure the virtual directory to require the Secure Sockets Layer (SSL) protocol.

B. Apply Web Services Enhancements (WSE) 3.0 security to the Web service that is configured to use an X.509 certificate with the Sign and Encrypt protection level.

C. Apply Web Services Enhancements (WSE) 3.0 security to the Web service that is configured to use an X.509 certificate with the Sign-Only protection level.

D. Configure the virtual directory that hosts the XML Web service to use basic authentication.

Answer: C

5. You are writing a Web service application that uses Web Services Enhancements (WSE) 3.0 policies. The Web service request and response data must be signed. Routing occurs between the client and server computers, and uses the Action SOAP header of the SOAP messages. You need to ensure that the SOAP signature is not invalidated when the SOAP message is routed. What should you do?

A. In the policy file, set the signatureOptions attribute to IncludeAddressing.

B. In the policy file, set the signatureOptions attribute to IncludeSoapBody.

C. Sign the SOAP request with a UserName token that includes a password to allow the SOAP message to be signed again during routing.

D. Sign a SOAP request that encapsulates the initial SOAP request.

Answer: B

6. You are creating a Windows-based application that allows users to store photographs remotely by using a Web service. The Web service is configured to use Web Services Enhancements (WSE) 3.0, and it has the following configuration XML in its Web.config file.

```
<microsoft.web.services3>
  <messaging>
    <mtom serverMode="always"/>
  </messaging>
</microsoft.web.services3>
```

You need to configure the application to ensure that a photograph can be sent to the Web service. What

should you do?

A. Add the following XML to the application's App.config file.<microsoft.web.services3> <messaging> <mtom serverMode="always" />
</messaging></microsoft.web.services3>

B. Add the following XML to the application's App.config file.<microsoft.web.services3> <messaging> <mtom serverMode="never" />
</messaging></microsoft.web.services3>

C. Add the following XML to the application's App.config file.<microsoft.web.services3> <messaging> <mtom clientMode="Off" />
</messaging></microsoft.web.services3>

D. Add the following XML to the application's App.config file.<microsoft.web.services3> <messaging> <mtom clientMode="On" />
</messaging></microsoft.web.services3>

Answer: D

7. A Windows Forms application calls in to a Web service named SensitiveData. The project has a Web reference named SensitiveDataWS. The code uses a class of type SensitiveDataWS.Service. SensitiveDataWS.Service is a proxy to the Web service.

An administrator reports that users running the client application receive a SoapHeaderException exception with the following message text: "Security requirements are not satisfied because the security header is not present

in the incoming message". You discover that the Web Services Enhancements (WSE) 3.0 policy file for the Web service was changed to require the encryption of SOAP messages.

You acquire the X.509 certificate that is used for encryption in the Web service. You need to ensure that the Windows Forms application meets the new security requirements of the Web service. What should you do?

A. In each Web method, assign an instance of the X509Certificate class that is initialized with the acquired X.509 certificate, to the ClientCertificates property of the SensitiveDataWS.Service class.

B. Apply WSE security to the Windows Forms project that is configured to use the acquired X.509 certificate with the Sign and Encrypt protection level.Renew the Web reference and modify the code to use the new SensitiveDataWS.ServiceWse class.

C. Apply WSE security to the Windows Forms project that is configured to use the acquired X.509 certificate with the Sign-Only protection level.Renew the Web reference and modify the code to use the new SensitiveDataWS.ServiceWse class.

D. In the Windows Forms project, create a custom class that inherits from the SoapHeader attribute class

with a public property of type X509Certificate.Initialize the public property with the acquired X.509 certificate.Apply the created attribute to code in the client application that calls the Web service.Initialize the attribute with the name of the custom class.

Answer: B

8. You write a client application that uses a Web service. The Web service is hosted in a Windows service application. The Web service can be called only by using TCP. You are provided with the proxy that is enabled by Web Services Enhancements (WSE) 3.0. The proxy was generated by adding a Web reference by using

Microsoft Visual Studio 2005. You need to ensure that the proxy calls the Web service correctly. Which two actions should you perform? (Each correct answer presents part of the solution. Choose two.)

A. On the Destination property of the proxy, set the TransportAddress property to the TCP address of the hosted Web service.

B. Set the Action object in the SOAP request's addressing headers to the name of the Web service method you want to call.

C. Change the URL of the proxy to the TCP address of the hosted Web service.

D. Set the Action object in the SOAP request's addressing headers to the TCP address of the Web service you want to call.

Answer: BC

9. You are creating a Windows-based application that allows users to store photographs remotely by using a Web service. Both the Web service and the application are configured to use Web Services Enhancements (WSE) 3.0. You need to configure the application to ensure that a photograph can be sent to the Web service. What should you do?

A. Write the following code segment in the application.

ImageServiceWse serviceProxy = new ImageServiceWse();byte[] img = GetImageBytes ();serviceProxy.RequestSoapContext.Add(imageName, img);serviceProxy.SaveImage(null, imageName);

B. Write the following code segment in the application.

ImageServiceWse serviceProxy = new ImageServiceWse();byte[] img = GetImageBytes ();serviceProxy.SaveImage(img, imageName);

C. Write the following code segment in the application.

ImageService serviceProxy = new ImageService();serviceProxy.SoapVersion = SoapProtocolVersion.

Soap12;byte[] img = GetImageBytes();serviceProxy.SaveImage(img, imageName);

D. Write the following code segment in the application.

ImageService serviceProxy = new

ImageService();serviceProxy.UserAgent = "Microsoft WSE 3.0";byte[] img

= GetImageBytes();serviceProxy.SaveImage(img, imageName);

Answer: B

10. A Web service application uses Web Services Enhancements (WSE) 3.0. A class named RejectUnknownActorFilter is derived from the SoapFilter class. The RejectUnknownActorFilter class throws a SoapException exception if the request contains an unexpected actor. A class defines a policy assertion as follows.

(Line numbers are included for reference only.)

```
01   public class RequireActorAssertion : PolicyAssertion {
02       public override SoapFilter
03           CreateClientInputFilter(FilterCreationContext context) {
04           return null;
05       }
06       public override SoapFilter
07         CreateClientOutputFilter(FilterCreationContext context) {
08         return null;
09       }
10       public override SoapFilter
11           CreateServiceInputFilter(FilterCreationContext context) {
12           return null;
13       }
14       public override SoapFilter
15           CreateServiceOutputFilter(FilterCreationContext context) {
16           return null;
17       }
18   }
```

You need to ensure that the Web service rejects any SOAP request that contains an unexpected actor. Your code must minimize the server resources used to process the request. What should you do?

A. Replace line 04 with the following line of code.return new RejectUnknownActorFilter();

B. Replace line 08 with the following line of code.return new RejectUnknownActorFilter();

C. Replace line 12 with the following line of code.return new RejectUnknownActorFilter();

D. Replace line 16 with the following line of code.return new RejectUnknownActorFilter();

Answer: C