[2016-New Microsoft New Exam 70-513 VCE Files Free Instant Download (51-60)

2016 June Microsoft Official New Released 70-513 Q&As in GreatExam.com!

100% Free Download! 100% Pass Guaranteed!

No doubt that 70-513 exam is a worth challenging task but you should not feel hesitant against the confronting difficulties. GreatExam is supplying the new version of 70-513 VCE dumps now. Get a complete hold on 70-513 exam syllabus through GreatExam and boost up your skills. What's more, the 70-513 dumps are the latest. It would be great helpful to your 70-513 exam.

Following questions and answers are all new published by Microsoft Official Exam Center: (The full version is in the end of the article!!!)

QUESTION 51
Your company has an existing Windows Communication Foundation (WCF) service.
The following code segment is part of the service. (Line numbers are included for reference only.)

```
01  ServiceHost host = GetServiceHost();
02
host.Open();
```


You need to ensure that AJAX client applications can access the service.
Which code segment should you insert at line 02?

A.
```
NetTcpBinding binding = new NetTcpBinding();
ServiceEndpoint ep = host.AddServiceEndpoint(
 typeof(ICatalogService), binding, "ajax");
ep.Behaviors.Add(new WebHttpBehavior());
```

B.
```
WebHttpBinding binding = new WebHttpBinding();
ServiceEndpoint ep = host.AddServiceEndpoint(
 typeof(ICatalogService), binding, "ajax");
ep.Behaviors.Add(new WebScriptEnablingBehavior());
```

C.
```
NetTcpBinding binding = new NetTcpBinding();
ServiceEndpoint ep = host.AddServiceEndpoint(
 typeof(CatalogService), binding, "ajax");
ep.Behaviors.Add(new WebScriptEnablingBehavior());
```

D.
```
BasicHttpBinding binding = new BasicHttpBinding();
ServiceEndpoint ep = host.AddServiceEndpoint(
 typeof(CatalogService), binding, "ajax");
ep.Behaviors.Add(new WebScriptEnablingBehavior());
```


A. Option A
B. Option B
C. Option C
D. Option D

Answer: B

QUESTION 52
You need to modify a client application that consumes a Windows Communication Foundation (WCF) service.
The service metadata is no longer available.
You need to modify the previously generated proxy to include asynchronous calls to the service.
What should you do?

A. Update the service reference with the Generate asynchronous operations option.
B. Create a partial class for the previously generated proxy and include the new asynchronous methods.
C. Create a class with the same name as the previously generated proxy and add the new asynchronous methods.
Add the new class to a namespace that is different from the original proxy.
D. Create a class with the same name as the previously generated proxy and add the new asynchronous methods as partial methods.
Add the new class to a namespace that is different from the original proxy.

Answer: B

QUESTION 53
A Windows Communication Foundation (WCF) service handles online order processing for your company.
You discover that many requests are being made with invalid account numbers.
You create a class named AccountNumberValidator that has a method named Validate.
Before the message is processed, you need to validate account numbers with AccountNumberValidator and reject messages with invalid account numbers.
You create a new class that implements the IParameterInspector interface.
Which code segment should you use?

A.
```
public void AfterCall(string operationName, object[] outputs,
object returnValue, object correlationState)
{
String accountNumber = GetAccountNumber(outputs);
var validator = newAccountNumberValidator();
if(validator.Validate(accountNumber))
{
throw new FaultException();
}
}
public object BeforeCall(string operationName, object[]inputs)
{
return null;
}
```
B.
```
public void AfterCall(string operationName, object[] outputs,
object returnValue, object correlationState)
{
return;
}
public object BeforeCall(string operationName, object[] inputs)
{
return null;
}
```

C. public void AfterCall(string operationName, object[] outputs,
object returnValue, object correlationState)
{
String accountNumber = GetAccountNumber(outputs);
var validator = newAccountNumberValidator();
if(!validator.Validate(accountNumber))
{
return value = new FaultException();
}
}
public object BeforeCall(string operationName, object[]inputs)
{
return null;
}
D. public void AfterCall(string operationName, object[] outputs,
object returnValue, object correlationState)
{
return;
}
public object BeforeCall(string operationName, object[]inputs)
{
string accountNumber = GetAccountNumber(inputs);
var validator = newAccountNumberValidator();
if (!validator.Validate(accountNumber))
{
return new FaultException();
}
}

Answer: D

QUESTION 54
You are maintaining a Windows Communication Foundation (WCF) service that uses a custom username password class to authenticate clients with.
The service certificate is hosted in the deployment server store for trusted root certificate authorities and has a Subject value of TaxServiceKey.
Other service certificates hosted on the same server also use TaxServiceKey as a Subject value.
You need to ensure that the service identifies itself with a certificate whose subject name and distinguished names are TaxServiceKey.
Which code segment should you use?

A. HostInstance Credentials.ServiceCertificate SetCertificate(
StoreLocation.LocalMachine, StoreName.My,
x509FindType.FindBySubjectName, "CN="TaxServiceKey");
B. HostInstance Credentials.ServiceCertificate SetCertificate(
StoreLocation.LocalMachine, StoreName.AuthRoot,
x509FindType.FindBySubjectName, "CN="TaxServiceKey");
C. HostInstance Credentials.ServiceCertificate SetCertificate(
StoreLocation.LocalMachine, StoreName.My,
x509FindType.FindByDistinguishedName, "CN="TaxServiceKey");

D. HostInstance Credentials.ServiceCertificate SetCertificate(
StoreLocation.LocalMachine, StoreName.Root,
x509FindType.FindByDistinguishedName, "CN="TaxServiceKey");

Answer: D

QUESTION 55
A Windows Communication Foundation (WCF) client configuration file contains the following XML segment in the system.serviceModel element.
<client>
<endpoint address="net.tcp://server/ContosoService"
binding="netTcpBinding"
contract="Contoso.IContosoService"
name="netTcp"/>
<endpoint address="net.pipe://localhost/ContosoService"
binding="netNamedPipeBinding"
contract="Contoso.IContosoService"
name="netPipe" />
</client>
You need to create a channel factory that can send messages to the endpoint listening at net.pipe://localhost/ContosoService.
Which code segment should you use?

A. ChannelFactory<Contoso.IContoso> factory =
new ChannelFactory<Contoso.IContoso>("Contoso.IContoso");
B. ChannelFactory<Contoso.IContoso> factory =
new ChannelFactory<Contoso.IContoso>("netNamedPipeBinding");
C. ChannelFactory<Contoso.IContoso> factory =
new ChannelFactory<Contoso.IContoso>("netPipe");
D. ChannelFactory<Contoso.IContoso> factory =
new ChannelFactory<Contoso.IContoso>(
"net.pipe//localhost/ContosoService");

Answer: C

QUESTION 56
A self-hosted Windows Communication Foundation (WCF) service uses a secure HTTP binding with a custom principal permission mode.
The binding requires users to provide their Windows logon credentials.
You need to retrieve the identity of the caller.
What are two possible properties you can use to achieve this goal? (Each correct answer presents a complete solution Choose two)

A. Thread.CurrentPrincipal.Identity.Name
B. HttpContext.Current.User.Identity.Name
C. ServiceSecurityContext.Current.PrimaryIdentity.Name
D. OperationContext.Current.ServiceSecurityContext.PrimaryIdentity.Name

Answer: CD

QUESTION 57
You are developing a Windows Communication Foundation (WCF) service that does not operate on a duplex channel.

You find that operations do not start until all previous operations have finished.
The service hosting code contains the following lines.
var service = new WarehouseService();
var host = new ServiceHost(service);
You need to ensure that new operations do not wait for previous operations to finish.
Which attribute should you use to decorate the service?

A. [ServiceBehavior(InstanceContextMode=InstanceContextMode.Single, ConcurrencyMode=ConcurrencyMode.Multiple)]
B. [CallbackBehavior(ConcurrencyMode=ConcurrencyMode.Multiple)]
C. [ServiceBehavior(InstanceContextMode=InstanceContextMode.Single, ConcurrencyMode=ConcurrencyMode.Single)]
D. [ServiceBehavior(InstanceContextMode=InstanceContextMode.Single, ConcurrencyMode=ConcurrencyMode.Reentrant)]

Answer: A
Explanation:
Only Concurrency Mode. Multiple gives as singleton service with support of multiple requests

QUESTION 58
Your Windows Communication Foundation (WCF) client application uses HTTP to communicate with the service.
You need to enable message logging and include all security information such as tokens and nonces in logged messages.
What should you do?

A. In the application configuration file, add the IogKnownPii attribute to the message logging diagnostics source and set the value of the attribute to true.
Generate the ContosoService class using the Add Service Reference wizard.
Add a reference to System.ServiceModel.Routing.dll.
Add the following code segment:
ContosoService client = new ContosoService();
SoapProcessingBehavior behavior = new SoapProcessingBehavior();
behavior.ProcessMessages = true;
client.Endpoint.Behaviors.Add(behavior);
B. In the application configuration file, add the following XML segment to the system.serviceModel configuration section group.
<diagnostics>
<messageLogging LogMessagesAtTransportLevel="true"
LogEntireMessage="true" />
</diagnostics>
C. In the machine configuration file, add the following XML segment to the system.serviceModel configuration section.
<machineSettings enableLoggingKnownPii="true" />
Generate the ContosoService class using the Add Service Reference wizard.
Add the following code segment.
ContosoService client = new ContosoService();
client.Endpoint.Behaviors.Add(new CallbackDebugBehavior(true));
D. In the machine configuration file, add the following XML segment to the system.serviceModel configuration section.
<machineSettings enableLoggingKnownPii="true" />
In the application configuration file, add the IogKnownPii attribute to the message logging

diagnostics source and set the value of the attribute to true.
In the application configuration file, add the following XML segment to the
system.serviceModel configuration section group.
<diagnostics>
<messageLogging LogMessagesAtTransportLevel="true"/>
</diagnostics>

Answer: D

QUESTION 59
You are moving a Windows Communication Foundation (WCF) service into production.
You need to be able to monitor the health of the service.
You only want to enable all performance counter instances exposed by the ServiceModelService 4.0.0.0 counter group.
Which element should you add to the system.serviceModel section in the application configuration file?

A. <diagnostics performanceCounters="ServiceOnly" />
B. <diagnostics wmiProviderEnabled="true" performanceCounters="Off" />
C. <diagnostics performanceCounters="All" />
D. <diagnostics wmiProviderEnabled="true" />

Answer: A

QUESTION 60
A Windows Communication Foundation (WCF) solution uses the following contract to share a message across its clients. (Line numbers are included for reference only.)

```
01 [ServiceContract]
02 public interface ITeamMessageService
03 {
04   [OperationContract]
05   string GetMessage();
06
07   [OperationContract]
08   void PutMessage(string message);
09 }
```


The code for the service class is as follows.

```
10 public class TeamMessageService : ITeamMessageService
11 {
12   Guid key = Guid.NewGuid();
13   string message = "Today's Message";
14   public string GetMessage()
15   {
16     return string.Format("Message:{0}. Key:{1}",
         message, key);
17   }
18
19   public void PutMessage(string message)
20   {
21     this.message = message;
22   }
23 }
```


The service is self-hosted.
The hosting code is as follows.

```
24 ServiceHost host =
     new ServiceHost(typeof(TeamMessageService));
25 BasicHttpBinding binding =
     new BasicHttpBinding(BasicHttpSecurityMode.None);
26 host.AddServiceEndpoint(
     "MyApplication.ITeamMessageService", binding,
     "http://localhost:12345");
27 host.Open();
```


You need to ensure that all clients calling GetMessage will retrieve the updated string if the message is updated by any client calling PutMessage.
What should you do?

A. Add the following attribute to the TeamMessageService class, before line 10.
[ServiceBehavior(InstanceContextMode =
InstanceContextMode.Single)]
B. Add the following attribute to the TeamMessageService class, before line 10.
[ServiceBehavior(InstanceContextMode =
InstanceContextMode.PerSession)]
Then change the binding definition on the service at line 25, and on the client to the following
WSHttpBinding binding = new WSHttpBinding(SecurityMode.None);
binding.ReliableSession.Enabled = true;
C. Pass a service instance to the instancing code in line 24, as follows.
ServiceHost host = new ServiceHost(new TeamMessageService());
D. Redefine the message string in line 13, as follows
static string message = "Today's Message";
Then change the implementation of PutMessage in lines 19-22 to the following
public void PutMessage(string message)
{
TeamMessageServiceMessage.PutMessage;
}

Answer: A

GreatExam is one of the leading exam preparation material providers. We have a complete range of exams offered by the top vendors. You can download 70-513 dumps in PDF format on GreatExam.com. Comparing with others', our 70-513 exam questions are more authoritative and complete. What's more, the 70-513 prepare material are the latest. We ensure you pass the 70-513 exam easily.

2016 Microsoft 70-513 exam dumps (All 341 Q&As) from GreatExam:

http://www.greatexam.com/70-513-exam-questions.html [100% Exam Pass Guaranteed!!!]