using System.ServiceModel;

namespace WCFTestImpersonation1

{

[ServiceContract]

public interface IService1

{

[OperationContract]

string getIdentity ();

}

}

using System;

using System.Security.Principal;

using System.ServiceModel;

namespace WCFTestImpersonation1

{

public class Service1 : IService1

{

[OperationBehavior(Impersonation = ImpersonationOption.Allowed)]

public string getIdentity()

{

string Error = "";

try

{

WindowsIdentity callerWindowsIdentity = ServiceSecurityContext.Current.WindowsIdentity;

return "Thread Identity: " + callerWindowsIdentity.Name + " | Thread Identity level: " + callerWindowsIdentity.ImpersonationLevel + " | Token:" + callerWindowsIdentity.Token.ToString();

}

catch (Exception es)

{

Error = es.ToString();

}

return Error;

}

}

}

Web Config del Web Service

<?xml version="1.0"?>

<configuration>

<appSettings>

<add key="aspnet:UseTaskFriendlySynchronizationContext" value="true" />

</appSettings>

<system.web>

<compilation debug="true" targetFramework="4.5.2" />

<httpRuntime targetFramework="4.5.2"/>

</system.web>

<system.serviceModel>

<bindings>

<wsHttpBinding>

<binding name="securingWSHttpBinding">

<security mode ="Message">

<transport clientCredentialType="Windows" />

<message establishSecurityContext="true" />

</security>

</binding>

</wsHttpBinding>

</bindings>

<behaviors>

<serviceBehaviors>

<behavior name="ConfigurationServiceBehavior">

<!-- set the values below to false before deployment -->

<serviceMetadata httpGetEnabled="true" httpsGetEnabled="true"/>

<!-- Set to false before deployment to avoid disclosing exception information -->

<serviceDebug includeExceptionDetailInFaults="true"/>

</behavior>

</serviceBehaviors>

</behaviors>

<protocolMapping>

<add binding="basicHttpsBinding" scheme="https" />

</protocolMapping>

<serviceHostingEnvironment aspNetCompatibilityEnabled="true" multipleSiteBindingsEnabled="true" />

<services>

<service name="WCFTestImpersonation1.Service1"

behaviorConfiguration="ConfigurationServiceBehavior">

<endpoint binding="wsHttpBinding"

bindingConfiguration="securingWSHttpBinding"

contract="WCFTestImpersonation1.IService1" />

</service>

</services>

</system.serviceModel>

<system.webServer>

<modules runAllManagedModulesForAllRequests="true"/>

<!--

To browse web app root directory during debugging, set the value below to true.

Set to false before deployment to avoid disclosing web app folder information.

-->

<directoryBrowse enabled="true"/>

</system.webServer>

</configuration>

**LA LLAMADA**

Service1Client client = new Service1Client();

client.ClientCredentials.Windows.AllowedImpersonationLevel =

System.Security.Principal.TokenImpersonationLevel.Impersonation;

client.ClientCredentials.Windows.ClientCredential.UserName = "tdal6392@sa.convergys.com";

client.ClientCredentials.Windows.ClientCredential.Password = "Levitico@06";

client.ClientCredentials.Windows.ClientCredential.UserName = "ccec3554@sa.convergys.com";

client.ClientCredentials.Windows.ClientCredential.Password = "Convergys109";

string ss = client.getIdentity();

Elimine en el Proyecto de llamada la parte del identity, no es necesario

<endpoint address="http://localhost:61336/Service1.svc" binding="wsHttpBinding"

bindingConfiguration="WSHttpBinding\_IService1" contract="ServiceReference1.IService1"

name="WSHttpBinding\_IService1">

<identity>

<userPrincipalName value="tdal6392@sa.convergys.com" />

</identity>

</endpoint>