

Re: Impersonation (pour Michel)

Source:

<http://www.tech-archive.net/Archive/SharePoint/microsoft.public.sharepoint.windowsservices/2004-09/1812.html>

From: David McKenzie (*DavidMcKenzie_at_bonbon.net*)

Date: 09/29/04

Date: Wed, 29 Sep 2004 13:02:07 -0500

Hi Greg,

Sorry , I have not checked the ng this week

Anyways – I found that I had to add the impersonated administrator to the root site (where the site templates are stored) as well as the parent of your new site (if they are not the same)

I am not certain if it was required for the impersonated admin to be a 'site collection administrator', but i have that assigned as well.

"Greg Christie" <grey_christ@yahoo.com> wrote in message news:9792070.0409240804.db3819b@posting.google.com...

> *Wei-Dong Xu or David Mackenzie,*

>

> *I desperately need help from whoever of you has my answer. I have
> tried to implement impersonation code in my web part over and over
> again, and though the impersonation code runs and seems to get a
> context, it appears that the impersonation doesn't actually take. In
> other words, the code still runs under the context of the logged-in
> user.*

>

> *I'm losing hair rapidly here! Any thoughts on other settings that I
> should be looking at or some simple thing I'm missing?*

>

> *I have the web.config set to Full trust, and I still keep getting a
> request for login 3X when I try to hit this WSS site as a reader even
> when that reader has access to the site. My code tries to iterate
> through the hierarchy of subsites, upon which it fails for my poor
> "Reader" user.*

>

>

> *Here's the code below...*

>

> using System;
> using System.Configuration;
> using System.ComponentModel;
> using System.Security.Principal;

microsoft.public.sharepoint.windowsservices: Re: Impersonation (pour Michel)

```
> using System.Runtime.InteropServices;
> using System.Web.UI;
> using System.Web.UI.WebControls;
> using System.Xml.Serialization;
> using Microsoft.SharePoint;
> using Microsoft.SharePoint.Utilities;
> using Microsoft.SharePoint.WebPartPages;
> using Microsoft.SharePoint.WebControls;
> using System.Reflection;
> using System.Security;
>
> namespace WRF_CustomWebParts
> {
>     /// <summary>
>     /// Description for WRF_AvailableSiteList.
>     /// </summary>
>     [ToolboxData("<{0}:WRF_AvailableSiteList
> runat=server></{0}:WRF_AvailableSiteList>"),
>     XmlRoot(Namespace="WRF_CustomWebParts")]
>     public class WRF_AvailableSiteList :
>     Microsoft.SharePoint.WebPartPages.WebPart
>     {
>
>         #region Impersonation functions and API Declarations
>
>         protected static WindowsIdentity CreateIdentity(string User, string
> Domain,
> string Password)
>         {
>             // The Windows NT user token.
>             IntPtr tokenHandle = new IntPtr(0);
>
>             const int LOGON32_PROVIDER_DEFAULT = 0;
>             const int LOGON32_LOGON_NETWORK = 3;
>
>             tokenHandle = IntPtr.Zero;
>
>             // Call LogonUser to obtain a handle to an access token.
>             bool returnValue = LogonUser(User, Domain, Password,
> LOGON32_LOGON_NETWORK, LOGON32_PROVIDER_DEFAULT,
> ref tokenHandle);
>
>             if (false == returnValue)
>             {
>                 int ret = Marshal.GetLastWin32Error();
>                 throw new Exception("LogonUser failed with error code: " + ret);
>             }
>
>             System.Diagnostics.Debug.WriteLine("Created user token: " +
> tokenHandle);
>
>             //The WindowsIdentity class makes a new copy of the token.
>             //It also handles calling CloseHandle for the copy.
>             WindowsIdentity id = new WindowsIdentity(tokenHandle);
>             CloseHandle(tokenHandle);
>             return id;
>         }
>
>         [DllImport("advapi32.dll", SetLastError=true)]
>         private static extern bool LogonUser(String lpszUsername, String
> lpszDomain, String lpszPassword,
> int dwLogonType, int dwLogonProvider, ref IntPtr phToken);
```

microsoft.public.sharepoint.windowsservices: Re: Impersonation (pour Michel)

```
>
> [DllImport("kernel32.dll", CharSet=CharSet.Auto)]
> private extern static bool CloseHandle(IntPtr handle);
>
> #endregion
>
> public WRF_AvailableSiteList()
> {
> this.Title = "Available Site List";
> }
>
> /// <summary>
> /// Render this Web Part to the output parameter specified.
> /// </summary>
> /// <param name="output"> The HTML writer to write out to </param>
> protected override void RenderWebPart(HtmlTextWriter output)
> {
> /* Create the logon objects
> string userName = this.Context.User.Identity.Name.ToString();
>
> try
> {
> /* Get the appropriate security context to run under.
>
> /* -----
> -----
> string impUser =
> ConfigurationSettings.AppSettings["ImpersonationUser"].ToString();
> string impDomain =
> ConfigurationSettings.AppSettings["ImpersonationDomain"].ToString();
> string impPwd =
> ConfigurationSettings.AppSettings["ImpersonationPwd"].ToString();
>
> // The Windows NT user token.
> WindowsImpersonationContext oWIC = CreateIdentity(impUser,
> impDomain, impPwd).Impersonate();
>
>
> /* -----
> -----
>
> /* Get the current web level
> SPWeb oMasterWeb = SPControl.GetContextWeb(this.Context);
>
> /* Write out the title bar
> output.Write("<table><tr><td>" +
> SPEncode.HtmlEncode(oMasterWeb.Title) + " Client Matter
> Sites</td></tr>");
>
> /* Get the site html to display
> output.Write(GetSiteHTML(oMasterWeb, userName, ""));
>
> /* Write out the bottom part.
> output.Write("</table>");
>
> /* Write out the line at the bottom of the web part.
> output.Write("<TABLE id='Table1' cellSpacing='0' cellPadding='0'
> width='100%' border='0'><TR><TD class='ms-partline' colSpan='2'><IMG
> height='1' alt=' ' src='/_layouts/images/blank.gif'
> width='1'></TD></TR><TR><TD class='ms-addnew' style='PADDING-BOTTOM:
> 3px'><IMG alt=' ' src='/_layouts/images/rect.gif'>&nbsp;  <A
> class='ms-addnewx' id='idHomePageNewEvent' onclick='alert(\"This
```

