

Re: impersonation and accessing remote folder

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.aspnet/2004-07/6277.html>

From: Jon L. Lovesky (jonl_at_willowrunfoods.com)

Date: 07/26/04

Date: Mon, 26 Jul 2004 14:23:19 -0400

Thanks for the feedback Scott. I am impersonating the user. The server and client in this case are both on the same domain, all Windows 2000. According to the referenced article, this means that kerberos is used. Is there any way to verify this?

Jon

"Scott Allen" <[bitmask@\[nospam\].fred.net](mailto:bitmask@[nospam].fred.net)> wrote in message news:5f23g05vql85uja9gfffdp4116v0aomd63@4ax.com...

> *Hi Jon:*

>

> *Are you impersonating or logging in the user? If it is impersonation, then NTLM does not support double-hop impersonations (meaning that once passed to the IIS server, the same credentials cannot be passed to a remote server to access a folder).*

>

> *One way around this is to use kerberos delegation:*

>

> <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vsent7/html/vxconaspnetdelegation.asp>

>

> --

> *Scott*

> <http://www.OdeToCode.com>

>

> *On Fri, 23 Jul 2004 16:47:19 -0400, "Jon L. Lovesky"*

> *<jonl@willowrunfoods.com> wrote:*

>

> *>Hello all,*

> >

> *>I am attempting to access a remote folder from an asp.net application (all*

> *>within the same domain). The application is configured for windows*

> *>authentication in IIS and the asp.net worker process runs as the local*

> *>ASP.NET account. When the application is about to access the remote folder I*

> *>switch the security context to the remote user programmatically, then switch*

> >it back afterwards. When I run the application on my local system where I
do
> >my development, it works fine. When I run the application from the
server,
> >access to the remote folder is denied. I have verified the security
context
> >switches to the remote user prior to accessing the remote folder and that
> >the user has been granted access to the folder, so I am not sure why
access
> >is denied. See the code below. This seems to be the accepted method to do
> >this, so what am I missing? Any suggestions are greatly appreciated.
> >
> >Jon
> >
> >
> >
> >Dim impersonationContext As
> >System.Security.Principal.WindowsImpersonationContext
> >
> >Dim currentWindowsIdentity As System.Security.Principal.WindowsIdentity
> >
> >currentWindowsIdentity = CType(HttpContext.Current.User.Identity,
> >System.Security.Principal.WindowsIdentity)
> >
> >impersonationContext = currentWindowsIdentity.Impersonate()
> >
> >'Access remote folder here and load a data table with file info
> >
> >impersonationContext.Undo()
> >
> >
> >
>