

Re: Remote call to COM impersonating another user

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2008-01/msg00790.html>

- *From:* "Willy Denoyette [MVP]" <willy.denoyette@xxxxxxxxxxx>
 - *Date:* Tue, 8 Jan 2008 17:42:28 +0100
-

"JCav" <jcavanaugh@xxxxxxxxxxx> wrote in message
[news:4783a140\\$0\\$28841\\$4c368faf@xxxxxxxxxxxxxxxxxxxx](mailto:news:4783a140$0$28841$4c368faf@xxxxxxxxxxxxxxxxxxxx)

I'm new to this, so bear with me. I seem to be missing something.

When I call LogonUser, it fails, I think because the domain I need to log into is not available from the machine I run this from. When I use the local domain it works fine – I become the other user when I impersonate him. This is how far I got before the original post. Is there a call that sends this information to the server and tells it to do this? As I said, this works with whatever JIntegra does it.

You don't have to send this information to the server, it's the role of COM to authenticate the client and pass the security context to the server.

When you call CoInitializeSecurity, specifying DynamicCloaking (or StaticCloaking) very early in the process, COM will automatically pass the impersonation token of the client to the server, the server will use this token when impersonating (the server needs to call CoImpersonateClient for this).

What you need to take care of is that the token passed is an impersonating token, so be carefull when calling LogonUser, the token sent must be an impersonation token not a direct token.

That means that you'll have to specify a "batch" or "interactive" logon type when calling LogonUser, before calling Impersonate. Another option is to use a "network" logon type and call "DuplicateToken" before using the duplicated token in the Impersonate call.

Willy.

.