

Re: Getting logged in user from a service?

Re: Getting logged in user from a service?

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-06/msg00804.html>

- *From:* "Willy Denoyette [MVP]" <willy.denoyette@xxxxxxxxxxx>
 - *Date:* Tue, 5 Jun 2007 21:17:54 +0200
-

"Larry Smith" <no_spam@xxxxxxxxxxx> wrote in message
news:OSBya85pHHA.4188@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

news:e3dxMO5pHHA.4532@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

CLR and the .NET Framework is all about. If you are coding directly against the OS services (that is, by directly calling WIN32 Api's) you have to consider a lot of things at "development" time, things like – is the API available on the *target* machine? – What are the security constraints, what privileges are there required to call these API when running as say "Local Service"? Can the API access a remote server instance? Most of these things are taken care of by the framework and it's underlying services, whatever these are, and in this particular case the underlying service is native WMI in top of Win32.

I don't see how using .NET Framework exempts you from worrying about security constraints, privileges, etc. It might automatically enable a held privilege in your token, that's about it.

No, the system.Management classes (and this is what we are talking about here) and WMI makes it possible to call OS services without YOU having the need to run with these elevated privileges.

Can you cite an example since this appears to defy standard Windows security (if I understand you correctly).

Re: Getting logged in user from a service?

Not at all, WMI is client/server based using DCOM, you call a service and the service executes the service call, when WMI needs to "enable" a privilege (note that I said 'enable'), it's up to the caller to ask the service to enable the required (whatever this one may be) privilege, the user doesn't need to know the "privilege" required, WMI know which one as it's stored in it's metabase.

In the exceptional case (there are only a few) that a call requires a privilege that is not held by the WMI account (say "Network Service"), then it's up to the caller to run as a more privileged user (or get a stronger logon token) and ask WMI to impersonate when executing the service call.

Willy.

.