

Re: WindowsPrincipal.IsInRole not working

Source: <http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.security/2003-09/0394.html>

From: K. Shier (*ks4hire_at_spamAtYourOwnRisk.yahoo.com*)

Date: 09/29/03

Date: Mon, 29 Sep 2003 11:43:11 -0400

"Joe Kaplan (MVP – ADSI)" <joseph.e.kaplan@removethis.accenture.com> wrote in message news:#tzVPN4gDHA.2960@TK2MSFTNGP11.phx.gbl...

> *That is an Active Directory question, but you should be able to change the type of the group unless it contains members that they current group cannot contain or if it is nested in another group that cannot contain a group of the new type. The Active Directory reference in MSDN explains all of this.*

well, although it pained me to do so, i just deleted and re-created them, since timely answers on how to edit them were not forthcoming. there were only a few anyway...

> *Reflection is cool, isn't it? :)*

yes – and one of those things we are using all the time without thinking about it. now that i know what it is, i'll think about it more, which might be a good way to start down the path of 'knowing when you should use it!' (i'd like to know how you arrived at the values of all the args you pass to .InvokeMember in that example... for now it's just one of those things i look at and accept at face value knowing that it comes from a 'higher source' =)

thanks again for the info!! =)

> *Joe K.*

> *"K. Shier" <ks4hire@spamAtYourOwnRisk.yahoo.com> wrote in message news:%23VsuHp3gDHA.3616@TK2MSFTNGP11.phx.gbl...*

> > *thanks Joe! i just checked that out and it looks like it is going to work.*

> >

> > *i appreciate the code snippet as i wouldn't even have known where to begin!*

> > *(until a couple minutes ago, my definition of 'reflection' was sitting at*

> *my*

> > *desk resting my chin on my hands thinking "I sure am screwed...")*

> >
> > *for some reason the network admin told me to create the security groups*
as
> > *'Domain Local' instead of 'Global'. but only groups that are Global*
show
> *up*
> > *in WindowsIdentity's _GetRoles method. i don't understand why, but*
that's
> > *only because i have no idea what the distinction is between a 'Global'*
and
> > *'Domain Local' group! (i have very limited experience administrating*
> *Win2k*
> > *nets.)*
> >
> > *FYI: i _am_ on framework 1.0, and it did occur to me briefly that*
> > *case-sensitivity might be the issue, but since the capitalization is*
> *correct*
> > *in my strings, i didn't give it any further thought. if i had known*
> *earlier*
> > *about the bug, i probably would have dwelled on capitalization forever,*
> > *thinking _that_ was the problem! =)*
> >
> > *now i just need to know if there is any way to reset the scope of the*
> > *security groups i've created to 'Global' or will i have to delete then*
> > *re-create them all? but i think that's a topic for another NG...*
> >
> > *THANKS AGAIN! =)*
> >
> > -----
> >
> > *"Joe Kaplan (MVP - ADSI)" <joseph.e.kaplan@removethis.accenture.com>*
wrote
> > *in message news:uRRZz3wgDHA.524@tk2msftngp13.phx.gbl...*
> > > *A couple of things:*
> > >
> > > *WindowsPrincipal definitely matches group names with "domain\group*
name"
> > > *where domain is either the domain or local machine name and group name*
> *is*
> > > *the samAccountName of the group. A bug in Framework 1.0 exists where*
> *role*
> > > *name checks are case sensitive for WindowsPrincipal.IsInRole. This is*
> > *fixed*
> > > *in 1.1, but it may be screwing you up.*
> > >
> > > *When I've been in a jam with these things, I've used reflection to*
call
> > *the*
> > > *private _GetRoles method of WindowsIdentity so that I can see the*
actual
> > > *array of role names. You shouldn't use this in production, but it is*

> > really
> > > helpful for debugging.
> > >
> > > (VB.NET code snippet)
> > >
> > > Dim id As WindowsIdentity = WindowsIdentity.GetCurrent()
> > >
> > > Dim idType As Type
> > > idType = GetType(WindowsIdentity)
> > > Dim result As Object = idType.InvokeMember("_GetRoles",
> > > BindingFlags.Static Or BindingFlags.InvokeMethod Or
> > > BindingFlags.NonPublic,
> > > Nothing, id, New Object() {id.Token}, Nothing)
> > > Dim roles() As String = DirectCast(result, String())
> > >
> > > HTH,
> > >
> > > Joe K.
> > >
> > >
> > >
> > > "K. Shier" <ks4hire@spamAtYourOwnRisk.yahoo.com> wrote in message
> > > news:uRTm7AugDHA.1648@TK2MSFTNGP09.phx.gbl...
> > > > VB.Net 2002, developing under Win2k pro, server is running Win2k
> > > > server
> > > >
> > > > i am trying to put some role-based security into my application.
the
> > > > roles
> > > > are based on permission Groups which i've already created on the
> > > > domain
> > > > controller. according to what i've read on the subject, a
> > > > WindowsPrincipal
> > > > should provide a direct mechanism to determine if the user is a
member
> > > > of
> > > > one of these Groups or not, but i can't get it to work.
> > > >
> > > >
> > > > here's a simplified excerpt from my prog:
> > > >
> > > > Public Sub SomeSecuredFeature()
> > > > Dim MyIdentity As System.Security.Principal.WindowsIdentity =
> > > > System.Security.Principal.WindowsIdentity.GetCurrent
> > > > Dim MyPrincipal As New
> > > > System.Security.Principal.WindowsPrincipal(MyIdentity)
> > > >
> > > > If MyPrincipal.IsInRole("DRS\DRS Administrators") Then
> > > > 'allow the user do something
> > > > Else
> > > > 'disallow

>>>> *End If*
>>>> *End Sub*
>>>>
>>>> *problem is, i can never get a True result for .IsInRole.*
>>>>
>>>> *the Group name is 'DRS Administrators' and according to AD Users And*
>>>> *Computers, i am a member of it!. the domain name is 'DRS', & the*
DC's
>>>> *machine name is 'DRSSERVER'*
>>>>
>>>> *i have tried a ludicrous number of permutations of what i *THINK**
the
>> *name*
>>>> *of the group might be: e.g. "DRS Administrators" (no machine/domain*
>>>> *qualifier) "DRSSERVER\DRS Administrators" (machine name instead of*
>> *domain*
>>>> *name), "\\DRS\DRS Administrators" (leading backslashes?!), and even*
>>>> *"drsserver.local/Users/DRS Administrators" (full AD object name). i*
>> *have*
>>>> *even tried matching the System groups like Administrators (which i*
am
>> *also*
>>>> *a*
>>>> *member of) and never get a match!*
>>>>
>>>> *seriously – i've tried every combination of these various formats*
for
>>>> *writing the group/role string, but no success...*
>>>>
>>>>
>>>> *what am i missing here?*
>>>>
>>>> *is there NOT a direct correlation between a DC's Groups and a .Net*
>>>> *WindowsPrincipal's 'Roles'? if not, how do i define the*
relationship
>>>> *between them?*
>>>>
>>>> *also, is there some method in .Net to simply query the DC and get a*
> *list*
>>> *of*
>>>> *all the Groups?*
>>>>
>>>>
>>>> *any insight would be much appreciated! THANKS! =)*
>>>>
>>>>
>>>
>>>
>>
>>
>>

microsoft.public.dotnet.security: Re: WindowsPrincipal.IsInRole not working

>>
>>
>>
>
>