

Re: Getting GROUPS from Active Directory by inputing an AD username

Source:

<http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.framework.aspnet.security/2007-02/msg00117.html>

- *From:* "Joe Kaplan" <joseph.e.kaplan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Sun, 25 Feb 2007 20:29:58 -0600
-

That sounds like the reason. Ask your admins or use ldp.exe to do a rootDSE query to see what the domain and forest functional levels are.

It sounds like you'll need to do an LDAP query, so better check out that tokenGroups code. :)

Joe K.

--

Joe Kaplan-MS MVP Directory Services Programming
Co-author of "The .NET Developer's Guide to Directory Services Programming"
<http://www.directoryprogramming.net>

--

"Patrick.O.Ige" <naijacoder@xxxxxxxxxxxx> wrote in message
<news:uCB3qyTWHHA.1212@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

Yeah Joe..

I tried you idea but 'm getting error SSPI status code error:
SEC_E_NO_S4U_PROT_SUPPORT

"The Kerberos subsystem encountered an error. A service for user protocol request was made against a domain controller which does not support service for a user."

My only guess for now is that our domain is still on WINDOWS 2000.

Could that be the reason and do you have any idea what might be wrong?

Thanks in Advance

Patrick.

"Joe Kaplan" <joseph.e.kaplan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message
<news:eA8AhVwVHHA.388@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

Patrick, the protocol transition code would look something like this:

```
WindowsIdentity wi = new WindowsIdentity("user@xxxxxxxxxxxx");  
IdentityReferenceCollection groups = wi.Groups;
```

The userPrincipalName is the user's logon name. You would prompt them

Re: Getting GROUPS from Active Directory by inputing an AD username

for that. If they supply the name in a different format, you would have to translate it to the UPN.

For the tokenGroups sample, the key is to get a DirectoryEntry object that is bound to the user. You might do this by prompting for the user name and then using a DirectorySearcher to find the user object in AD. Then, use that result to build the DirectoryEntry. Reading more in ch 10 may be of assistance here.

Joe K.

Joe Kaplan—MS MVP Directory Services Programming
Co—author of "The .NET Developer's Guide to Directory Services Programming"
<http://www.directoryprogramming.net>

"Patrick.O.Ige" <naijacoder@xxxxxxxxxxxx> wrote in message
<news:uOWcasvVHHA.4828@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

Thanks Joe for th reply.
What i have done from what Dominick adviced from
<http://www.leastprivilege.com/GettingAllGroupsForAWindowsAccountInNET20.aspx>
was to use
WindowsIdentity id =
(WindowsIdentity)HttpContext.Current.User.Identity
and then passed it to the grtGroups(id)
But with that i'm not passing the Logon user. i want to pass
in
grtGroups(logon_user).So that a user can come in and then
inputs a
username
and then retrieve their AD GROUPS.
But i don't really get what you wrote about using the
"userPrincipalName
as the only parameter"
Also i looked at using the tokenGroups method listed below
what would i
have to do to pass logon_user
Thanks in Advance

```
StringBuilder sb = new StringBuilder();

//we are building an '|' clause
sb.Append("|");

foreach (byte[] sid in user.Properties["tokenGroups"])
{
//append each member into the filter
sb.AppendFormat(
"(objectSid={0})", BuildFilterOctetString(sid));
```

Re: Getting GROUPS from Active Directory by inputing an AD username

```
}  
  
//end our initial filter  
sb.Append("");  
  
DirectoryEntry searchRoot = new DirectoryEntry(  
"LDAP://DC=domain,DC=com,  
null,  
null,  
AuthenticationTypes.Secure  
);  
  
using (searchRoot)  
{  
//we now have our filter, we can just search for the groups  
DirectorySearcher ds = new DirectorySearcher(  
searchRoot,  
sb.ToString() //our filter  
);  
  
using (SearchResultCollection src = ds.FindAll())  
{  
foreach (SearchResult sr in src)  
{  
//Here is each group now...  
Console.WriteLine(  
sr.Properties["samAccountName"][0]);  
}  
}  
}
```

"Joe Kaplan"
<joseph.e.kaplan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in
message
news:OCAxpCvVHHA.5092@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

There are two options for this that I would
consider using:

If your AD is 2003 native mode and the
machine that your app is running
on is 2003 or higher, you can use protocol
transition to generate a
WindowsIdentity for a user and get their
groups. Use the constructor
on WindowsIdentity that takes the

Re: Getting GROUPS from Active Directory by inputing an AD username

Just use
plain
WindowsAuthentication
- you can
get all
groups from
the
WindowsIdentity
that hangs
off
Context.User...

<http://www.leastprivilege.com/GettingAllGroupsForAWindowsAccount.aspx>

Dominick
Baier
(<http://www.leastprivilege.com>)

Developing
More
Secure
Microsoft
ASP.NET
2.0
Applications
(<http://www.microsoft.com/mspress/books/9989.asp>)

I
used
the
WindowsTokenRoleProvider
and
i
was
able
to
input
my
username
and
i
retrieved
all
the
GROUPS
i
belong
to
on

Re: Getting GROUPS from Active Directory by inputing an AD username

my
PC.
I'm
thinking
of
doing
the
same
but
against
Active
Directory.
How
can
i
do
the
same
against
AD?
Will
i
have
to
use
"AuthorizationStoreRoleProvider"
and
install
Azman?
Or
iare
they
any
other
ways?
I
have
used
ActiveDirectoryMembershipProvider
before
with
my
treeview
for
securitytimming
can
i
use
that?
Thanks
in

Re: Getting GROUPS from Active Directory by inputing an AD username

Re: Getting GROUPS from Active Directory by inputing an AD username

Advance