

## Re: Active Directory query doesn't work...

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-04/msg03421.html>

---

- *From:* aziegler <[aziegler@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:aziegler@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Wed, 25 Apr 2007 01:40:02 -0700
- 

Thank you for the advice with the ADSI-group. They've told me a much easier way to do what I wanted (using WindowsIdentity).

Thanks anyway for your efforts.

"Willy Denoyette [MVP]" wrote:

"aziegler" <[aziegler@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:aziegler@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)> wrote in message [news:E31CDFEB-AB2A-4A04-B3B7-D05AFA87FF66@xxxxxxxxxxxxxxxxxxxx](mailto:news:E31CDFEB-AB2A-4A04-B3B7-D05AFA87FF66@xxxxxxxxxxxxxxxxxxxx)

Hello, everybody.

I'd like to do this: For a big program (a web service) I need information about the usergroups an active-directory-user is member of. To be more precise, I need to know if a particular user is in a particular group or not. This is my first Active-Directory-query in a C#-program, so it might look crude or primitive...well, it doesn't work anyway...

The interesting part of the code is this:

```
public bool GetADUserGroups(string userName, string gruppe)
{
    bool ergebnis = false;

    DirectoryEntry ebr = new
    DirectoryEntry("LDAP://DOMAINE.DO","DOMAIN USER","PASSWORD");

    /*(do I need a domain admin for this or is a standard domain user
    sufficient)*/

    DirectorySearcher search = new DirectorySearcher(ebr);

    /*(these are the many filter variants I tried. Except for the last one that
    is not a comment, all terminated with errors)*/

    //search.Filter = String.Format("(cn={0})", userName);
```

Re: Active Directory query doesn't work...

```
//search.Filter =  
String.Format("&(objectClass=user)(userprincipalname={0})", userName);  
  
//search.Filter = "&(objectClass=user)(userprincipalname=" +  
userName + ")";  
  
search.Filter = "(objectClass=user)";  
  
/*(the username has the format "firstname.lastname", just like the login  
name)*/  
  
search.PropertiesToLoad.Add("memberOf");  
search.PropertiesToLoad.Add("samAccountName");  
  
foreach (SearchResult table in search.FindAll())  
{  
int groupCount = table.Properties["memberOf"].Count;  
  
logger.LogInfo(table.Properties["samAccountName"].ToString());  
  
if (table.Properties["samAccountName"].ToString() == userName)  
{  
  
for (int i = 0; i < groupCount; i++)  
{  
  
logger.LogInfo(table.Properties["memberOf"][i].ToString());  
  
if (table.Properties["memberOf"][i].ToString() ==  
gruppe)  
{  
ergebnis = true;  
}  
}  
}  
}  
  
return ergebnis;  
}
```

So, I'm finally there where I don't have any more ideas. I'm still trying,  
but I'm feeling like any idea is a very long shot...

I'd be glad about any help you can provide. Many thanks in advance!

## Re: Active Directory query doesn't work...

Not really a C# question, you might get better responses when posting to the adsi NG, anyway, following is a snip that illustrates how you can get the groups a user belongs to.

```
// bind to the Global Catalog
string rootPath = "GC://domaine.do/DC=..., DC=...";
//or
string rootPath = "LDAP://domaine.do/DC=..., DC=...";
..
string userAccount = "someUser";
..
using (DirectoryEntry root = new DirectoryEntry(rootPath, "domainuser", "password",
AuthenticationTypes.FastBind))
{
using (DirectorySearcher ds = new DirectorySearcher(root))
{
SearchResult sr = null;
ds.Filter = "(SAMAccountName=" + userAccount + ")";
sr = ds.FindOne();
using (DirectoryEntry user = sr.GetDirectoryEntry())
{
PropertyCollection pcoll = user.Properties;
PropertyValueCollection memberOf = pcoll["memberOf"];
foreach (string cnGroup in memberOf)
{
ds.Filter = cnGroup.Substring(0, cnGroup.IndexOf(',));
sr = ds.FindOne();
using (DirectoryEntry group = sr.GetDirectoryEntry())
{
Console.WriteLine(group.Properties["SAMAccountName"].Value.ToString());
}
}
}
}
}
```

Willy.