

# Re: General questions about LDAP, GC and access permissions

---

*Source:*

[http://www.tech-archive.net/Archive/Windows/microsoft.public.windows.server.active\\_directory/2007-07/msg00460](http://www.tech-archive.net/Archive/Windows/microsoft.public.windows.server.active_directory/2007-07/msg00460)

---

- *From:* "Joe Kaplan" <[joseph.e.kaplan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:joseph.e.kaplan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Wed, 11 Jul 2007 13:29:09 -0500
- 

Gotcha, that makes sense. If a one time snapshot is all you need, then don't bother with sync.

The reason I mentioned it is that sometimes things that look like they need to be one time deals actually do need to be able to stay in sync, but you don't find out until later because they didn't realize that it was important when they were first designing it.

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>

---  
"UncleRedz" <[UncleRedz@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:UncleRedz@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)> wrote in message [news:03CA2938-B5EA-4F2F-9CA9-44B8DB40E845@xxxxxxxxxxxxxxxxxxxx](mailto:news:03CA2938-B5EA-4F2F-9CA9-44B8DB40E845@xxxxxxxxxxxxxxxxxxxx)

Oh, and to clarify. The administrator only needs a snapshot of all available users and groups that CAN be added to the DB at the time that the administrator is managing the system. The administrator manually have to choose which users and groups to add to the DB. So there's no need to synchronize the DB with the AD.

"UncleRedz" wrote:

Task 1 was done in no time using the WindowsIdentity.Groups property. As for task 2, I think I've reached an acceptable level by using the GC.

As I only need to store the SID in the DB in order to be able to associate information with it, there's no need to sync the DB with AD. So I'm done with that task as well.

Re: General questions about LDAP, GC and access permissions

Cheers,  
UncleRedz

"Joe Kaplan" wrote:

Yes, that is the fully nested membership too, so you don't need to do any LDAP queries to discover the nesting. You are basically done with task 1.

Talk 2 is harder. I'd suggest you use a product to do that like Microsoft's MIIS. It is designed to sync various directories and can automate the task of moving the users and groups into SQL and keeping them in sync.

If you want to do this programmatically, use DirSync. This is represented in .NET 2.0+ with the DirectorySynchronization class which is available from the DirectorySearcher. We cover this in more detail in our book in ch 5 and have some code samples available on our book's website (link below).

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>

—  
"UncleRedz" <UncleRedz@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>  
wrote in message  
[news:D7FC31E1-E3EB-4693-9B15-944C453B8207@xxxxxxxxxxxxxxxxxxxx](mailto:news:D7FC31E1-E3EB-4693-9B15-944C453B8207@xxxxxxxxxxxxxxxxxxxx)

"UncleRedz" wrote:

Don't forget  
that when

Re: General questions about LDAP, GC and access permissions

you are  
using  
Windows  
auth,  
Windows  
itself  
will  
calculate a  
user's group  
membership  
in the user's  
logon  
token. It  
is  
best  
to not try to  
get the  
user's group  
membership  
via LDAP if  
Windows  
is  
going  
to do it for  
you.

Well, this sound most  
interesting, if the  
information that can be  
gained  
is  
enough, then this would be  
the easiest solution. Do you  
have any  
pointers  
to  
where I should look in order  
to get the memberships from  
the token?

Well, this is embarrassing, found the groups  
right in the  
WindowsIdentity...  
in plain sight, couldn't be any easier.

Cheers,  
UncleRedz

Re: General questions about LDAP, GC and access permissions