

Re: NetGroupGetUsers() occasional failures

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.networks/2005-11/msg00271>

- *From:* "Eugene Gershnik" <gershnik@xxxxxxxxxxx>
 - *Date:* Thu, 17 Nov 2005 07:37:18 -0800
-

Dave Lowther wrote:

> My code has the domain controller name in the 1st param and works fine
> except on my PC where it fails once every 12 mins 20 secs. I'm
> calling the API once every 10 seconds, so it works 73 times, then
> fails once, then works 73 times etc.

Even if the code worked perfectly this kind of polling is a bad idea. Do you really expect groups to change so often? In any realistic organization polling every few hours should be more than enough. Enumerating groups is an expensive operation (especially on large domains) both in terms of server CPU time and in network traffic. Doing it very often (and I guess you will also do it from multiple machines) puts unnecessary strain on your DC and network.

Also remember that all Net APIs are remote RPC calls. As such you must **expect** them to fall occasionally. I am not saying that the symptoms you are seeing are not a result of some faulty software on your box but even if so it doesn't really matter. Any time you use Net API you should be prepared to get an error because of congested network, busy server etc. In such case your only choice is to retry a reasonable amount of times.

—

Eugene

<http://www.gershnik.com>

-
- *Follow-Ups:*
 - ◆ ***Re: NetGroupGetUsers() occasional failures***
 - ◇ *From:* Dave Lowther
 - *References:*
 - ◆ ***NetGroupGetUsers() occasional failures***
 - ◇ *From:* Dave Lowther

Re: NetGroupGetUsers() occasional failures

◆ **Re: NetGroupGetUsers() occasional failures**

◇ From: Dave Lowther

- Prev by Date: **Re: NetGroupGetUsers() occasional failures**
- Next by Date: **Re: Sent and Received Byte Count**
- Previous by thread: **Re: NetGroupGetUsers() occasional failures**
- Next by thread: **Re: NetGroupGetUsers() occasional failures**
- Index(es):
 - ◆ **Date**
 - ◆ **Thread**