

Re: Role Providers Cache

Source:

<http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.framework.aspnet.security/2006-08/msg00180.html>

- *From:* Tom <Tom@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Fri, 18 Aug 2006 14:49:02 -0700
-

Ok think you answered all my question thanks a lot. Looks like we have to do our own cache using the cache management API. Also have to make sure we want to manage session. Some issue on how production support wants to set up in production. We have a farm but not sure if they want to have a cache Server. But our reverse proxies do have some sort of sticky sessions.

Thanks
Tom

"Joe Kaplan (MVP – ADSI)" wrote:

I'm not sure about the max size of the cookie, but you could look that up.

It seems like if you want to use the built-in stuff, the session-based caching would give you a good security blanket here, with the possible downside of less scalability and more consideration required in a server farm environment with out of process session state.

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>

--

"Tom" <Tom@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message
<news:CA624428-350F-4AED-9067-18A412AFFD80@xxxxxxxxxxxxxxxxxxxx>

Thanks for the reply.

The provider we need to write may be a short term once the Web site with over 200 functions (applications) is rewritten from VB6/ASP to .NET. The current solution uses a DLL to access all the security roles in a Novell EDIR. Once the WEB site if fully .NET we may have some flexibility to use a Auth Manager Role Provider or use the SQL Role provider (moving the roles to

Re: Role Providers Cache

SQL). But while in transition it seems like the best approach is to use a role provider using the DLL that as the source for the authorization.

We

have no way around the 200+ roles since each company(1200+) using the web

site wants to secure the application for their user. Each company has there

own data update and reporting function.

So my main concern was on the cache feature and the cookie was worried we may hit the limit on the cookie size and wonder if it worth caching the data

in cookies if the data size get large.

Was thinking if we needed to cache the results that the session would be better ?

Or maybe no cache at all ? Or Cookie but how large is too large ?

Thanks

Tom

"Joe Kaplan (MVP – ADSI)" wrote:

What was the question you were trying to ask? It sounds like you know the options pretty well.

If you are trying to avoid using session state (always a good idea in my mind unless it is unavoidable), the cookie does sound like a good idea,

but

it could be really big. You could create some sort of compression type

of

thing where you have a big bitmap to represent the roles and then just

store

the values in the cookie as the representative byte array of the roles.

Of

course, then you'd need some code to read and write the cookie in order

to

turn those back into friendly representations internally.

Joe K.

Re: Role Providers Cache

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

—
"Tom" <Tom@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in
message
<news:82C8D131-8E7B-482C-A852-192B8E9DFD73@xxxxxxxxxxxxxxxxxxxx>

If you do not cache role provider to Cookies
with the
cacheRolesInCookie="True" in your role
provider configuration, will it
default to cache to Session. We may have
over 200 roles for some
users
and
I am concerned about the size of the cookie
but would like the role
cached
if
possible. We will be writing our own role
provider but would like to
do
as
little as possible.

Thanks
Tom