

Re: ActiveDirectory authentication – more issues

Source:

<http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.framework.aspnet.security/2006-03/msg00301.html>

- *From:* David Thielen <thielen@xxxxxxxxxxxxxx>
 - *Date:* Tue, 21 Mar 2006 05:27:06 -0800
-

ok – thanks

—

thanks – dave
david_at_windward_dot_net
<http://www.windwardreports.com>

"Dominick Baier [DevelopMentor]" wrote:

Because it is using a different protocol – as simple as that.

Only IWA results in a WindowsIdentity. FormsAuthentication results in a FormsIdentity.

In the case of the AD Provider, LDAP is used to verify credentials – IsInRole does NOT hit the Active Directory – the roles are empty by default with the AD provider – you have to stack a role provider on top of that – there is no AD role provider – so often a combination for AD membership and AzMan roles is used.

To create a WindowsIdentity you have to create a token – you could do that yourself – you have username/password – but this has to be done on every single request – so you would have to cache/store the credentials of the user on the web server – nothing i would recommend.

so to cut a long story short –

you want an automatically generated WindowsIdentity – use IIS authentication for all other auth methods you get a Forms/Generic Identity.

Dominick Baier – DevelopMentor
<http://www.leastprivilege.com>

Hi;

Re: ActiveDirectory authentication – more issues

I believe you that it works this way. But I am curious as to why for this one use case.

User is prompted (forms) for username & password. username/password are authenticated via ActiveDirectory and IsInRole hits ActiveDirectory. This means the user's username/password in ActiveDirectory were passed to AD and verified in AD.

Why can't it at that point create a WindowsPrincipal/Identity? It has the user and has authenticated them. It seems to me that it would be legit at that point to issue the credentials. And this would then handle the case of a domain user using firefox or oasis.

"Dominick Baier [DevelopMentor]" wrote:

hi,

ok...

1) this can be mapped in web.config – both formats are supported. See in visual studio help for all variations

e.g, attributeMapUsername="SAMAccountName"

uses only the username without domain

2) no –you are doing forms authentication. NTLM would be IIS authentication and <authentication mode="Windows" /> would be set.

Then you cannot use the membership providers

3) no – see 2

4) see 2. You could maybe use Protocol Transition (only for domain accounts, only on w2k3, only in w2k3 domains) to get a token or use the Win32 LogonUser API (needs to store the password on the server – not recommended).

5) still no idea

have you ordered the book already?

Dominick Baier – DevelopMentor

Re: ActiveDirectory authentication – more issues

<http://www.leastprivilege.com>

Hi;

Ok, I have ActiveDirectory authentication working but have a couple of issues:

- 1) My username must be dave@xxxxxxxxxxxxxxx – it does not take windward\dave – why?
- 2) The authentication type is shown as forms – shouldn't it be NTLM?
- 3) Since I'm running from a computer on the domain and using IE, shouldn't it handle this automatically?
- 4) I do not get a WindowsIdentity but instead a FormsIdentity. I need a WindowsIdentity so I can do impersonation. How do I get that?
- 5) Context.User.IsInRole() returns false for groups I am a member of such as "windward\\Domain Users" – why?