

## Re: Utter madness!

**Source:**

<http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.framework.aspnet.security/2004-07/0187.html>

---

**From:** Joe Kaplan \((MVP - ADSI)\) (*joseph.e.kaplan\_at\_removethis.accenture.com*)

**Date:** 07/14/04

Date: Wed, 14 Jul 2004 10:46:09 -0500

It is just Windows security stuff. Whether or not it is obscure is debatable, but it sure helps to understand this stuff. Keith Brown has a great online book at [www.pluralsight.com](http://www.pluralsight.com) called Windows Security for .NET Developers that tells you what you need to know.

You can get a trusted connection back to SQL server. Just change your ASP.NET account (either processModel or app pool identity depending on version of IIS) to a domain account and make sure you have impersonation disabled. Then you are using SSPI to connect to SQL with a specific account.

If the requirement is to use WIA and have those credentials be used to authenticate with SQL server on a different box on the network, then Kerberos delegation is required. This is enabled either per machine or per user in Active Directory. Like I said, I'd avoid using this approach unless you absolutely need to because you are using specific per user security features in SQL Server as it hurts scalability and makes your life much more complicated.

Joe K.

"Paul Mason" <[masonp@cancer.bham.ac.uk](mailto:masonp@cancer.bham.ac.uk)> wrote in message [news:eoBinFbaEHA.2576@TK2MSFTNGP10.phx.gbl...](mailto:news:eoBinFbaEHA.2576@TK2MSFTNGP10.phx.gbl...)

>

> *Hi Joe,*

>

> *I tried using impersonation in one application and for the amount of connections we generate (300 max) it created a lot of extra work. I'll stick to forms authentication for now.*

>

> *I do find it odd that it's so easy to identify a domain authenticated user (through the WindowsIdentity object) and yet it's so difficult for it to then pass this onto SQL server.*

>

> *If it's going to be "tricky" to get a trusted connection to my SQL box working without having IIS installed on the same box, then it's not worth doing. Most people need something that's straightforward and reliable.*

Re: Utter madness!

>  
> *I will have a root around, but if it requires the level of in-depth*  
> *knowledge of an obscure technology that you're hinting at then I doubt*  
I'll  
> *take it any further...*  
>  
> *Cheers...P*  
>  
> *"Joe Kaplan (MVP – ADSI)" <joseph.e.kaplan@removethis.accenture.com> wrote*  
> *in message news:u5\$mm8aaEHA.384@TK2MSFTNGP10.phx.gbl...*  
>> *Lots of people run SQL on other boxes. There is no reason why you can't*  
> *do*  
>> *this. However, certain authentication scenarios are harder in that set*  
> *up.*  
>>  
>> *The issue of passing Windows credentials to SQL server can get tricky if*  
> *it*  
>> *is on a different box on the network. If it is your expectation that*  
you  
>> *will log on to SQL using web logged on user's credentials and you are*  
> *using*  
>> *Windows Integrated Authentication, then you will need to learn some*  
stuff  
>> *about Kerberos delegation to make this work. This is discussed ad*  
nauseum  
>> *in this group and you will find many pointers here with a Google search.*  
>>  
>> *However, there are many reasons why you would not want to use the user's*  
>> *credentials to connect to SQL but instead would want to use some kind of*  
>> *service account. One of the primary reasons is that you'll get better*  
>> *scalability if you use one set of credentials as you can use connection*  
>> *pooling. Another reason is that you can avoid the whole Kerberos*  
> *delegation*  
>> *thing that way. To do the service account approach, you have three*  
> *typical*  
>> *approaches: change the process account for ASP.NET to a domain account,*  
>> *impersonate a specific domain account or put your data access code in a*  
> *COM+*  
>> *component and configure it to use a specific domain identity via COM+.*  
> *All*  
>> *have good points and bad points.*  
>>  
>> *Joe K.*  
>>  
>> *"Paul Mason" <masonp@cancer.bham.ac.uk> wrote in message*  
>> *news:eNB9QkaaEHA.3512@TK2MSFTNGP12.phx.gbl...*  
>>>  
>>> *I think i've been getting my groups mixed up.*  
>>>  
>>> *I've been trying to get my intranet system to authenticate to SQL*  
server

> > > (2K) using a trusted connection for some time and have had to wait  
until  
> > we  
> > > upgraded to Active directory for kerberos to start working (I'm not  
100%  
> > > sure it's kerberos so bear with me).  
> > >  
> > > Now I've hit the final brick wall which means this isn't ever gonna  
> happen  
> > > in the current setup. It finally twigged (dropped like a tonne of  
lead  
> > more  
> > > like) when I read in the help :  
> > >  
> > > "If your application runs on a Windows-based intranet, you might be  
able  
> > > to  
> > > use Windows integrated security for database access. Integrated  
security  
> > > requires:  
> > > a.. That SQL Server be running on the same computer as IIS..... "  
> > > I can't believe that someone from MS actually wrote this. Are they  
> > > mad?...IIS and SQL server on the same machine....hackers paradise!  
> Appart  
> > > from being plain dangerous, it's bad networking practice, bad  
> programming  
> > > practice...it's just bad.  
> > >  
> > > Does anyone know if they are actually going to write something  
> useful...or  
> > > are we stuck with forms authentication forever!?! Not that I'm  
> > complaining.  
> > >  
> > > Cheers...P  
> > >  
> > >  
> > >  
> >  
> >  
>  
>