

## Re: Passthrough authentication w/ SQL trusted connection

**Source:**

<http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.framework.aspnet.security/2004-03/0355.html>

---

**From:** Ken Schaefer (*kenREMOVE\_at\_THISadOpenStatic.com*)

**Date:** 03/25/04

Date: Thu, 25 Mar 2004 12:40:50 +1100

a) if you do this, you will lose the benefits of connection pooling, as a separate connection will be used for each security context (each user account will have it's own pool). So, this solution will not scale to a large number of users. It's OK if you have a small number of users

b) the problem is double-hop authentication. When using IWA, the webserver does not have the user's password. It just gets a token from the DC, but the token does not have permission to logon to network resources.

Options:

a) if you are using a Windows 2000 Domain, you can enable delegation. This allows the IIS server to impersonate the Windows account, and logon to the backend SQL Server. You need to use Kerberos authentication for this (not NTLM v2)

b) if you are using a Windows 2003 Domain, when you enable constrained delegation, you can use Protocol Transition. This allows the user to authenticate using any of a number of mechanisms to the IIS server (eg Digest, or NTLM), and the webserver can still get an Kerberos token to logon to the SQL Server.

Here are a few articles to get you started:

**IMPORTANT:**

Read chapter 12 from the Building Secure ASP.Net Application Book – it has very good information about building scalable, secure ASP.Net applications (eg using a trusted subsystem model):

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetsec/html/secnetlpMSDN.asp?frame=true>

<http://support.microsoft.com/?id=319723>

INF: SQL Server 2000 Kerberos support including SQL Server virtual servers on server clusters

<http://support.microsoft.com/default.aspx?scid=kb;en-us;810572>

HOW TO: Configure an ASP.NET Application for a Delegation Scenario

