

# Sql Server Connection Irritations (again)

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.adonet/2006-05/msg00475.html>

---

- *From:* "Shawn B." <[leabre@xxxxxxxxx](mailto:leabre@xxxxxxxxx)>
  - *Date:* Wed, 17 May 2006 13:12:13 -0700
- 

We recently migrated from .NET framework 1.x to 2.0 with a code base that is 4 years old. The application is a mix between Windows Services, Web Application, and some internal WinForms tools. We never had a problem using framework 1.x and even the same code that fails in framework 2.0, pasted into a 1.x project succeeds every time. This is a C# 2005 project.

All our servers are Windows Server 2003 and all our SQL Server databases are SQL Server 2000 (we do not have an installation of 2005 on our network). We recently started receiving the following error message:

A connection was successfully established with the server, but then an error occurred during the pre-login handshake. When connecting to SQL Server 2005, this failure may be caused by the fact that under the default settings SQL Server does not allow remote connections. (provider: TCP Provider, error: 0 - A request to send or receive data was disallowed because the socket is not connected and (when sending on a datagram socket using a sendto call) no address was supplied.)

We have been attempting to troubleshoot this for 3 weeks now with minimal results. This problem happens during the SqlConnection.Open() command and always returns instantly as far we can tell. We have done the following:

- \* Added "np:" (named pipes) prefix to the connection string. This usually solves the problem but our production SQL Servers are seperated over the network from the application servers via VLAN and so named pipes is not an option, must be TCP/IP.
- \* Specified "tcp:x.x.x.x, xxxx;" in the connection string to no effect.
- \* Toggled various other connection string settings such as pooling, integrated security, and so on, to no effect in endless combination.
- \* Tested our application builds in DEBUG and RELEASE modes. DEBUG mode always fails while RELEASE build always succeeds. There is nothing in our project/solution settings that we changed apart from the defaults when created, and there is nothing in our code that distinguishes from RELEASE and DEBUG mode.

## Sql Server Connection Irritations (again)

\* Ported our c# code to VB.NET to see what happens and the problem happens exactly the same in VB.NET.

\* Ported our c# code to VS.NET 2003 and it succeeds.

\* Disabled the inclusion of VSHOST.EXE in our builds. This had a significant impact. On some client machines the problem dissappeared, on others it intensified, in others, it reduced the occurance but not completely.

We have toggled (in various combinations) the network adapter card CHECKSUM settings (on/off) to no appearant effect.

We have many database servers in production. When the user logs in, they always go to the same database which authenticates them and then returns with a connection string that they should use throughout the rest of their session. The first authentication connection NEVER fails but the second connection, where they are actually in the application, fails most of the time. These connections happen on the same server, BTW, just different databases. The assigned connectionstring never changes. Sometimes it works, sometimes it doesn't. In framework 1.x, it ALWAYS works.

According to this article from Siraj Lala at the Data Programmability team blog for data access in ADO.NET, <http://blogs.msdn.com/dataaccess/archive/2005/06/09/426957.aspx> the new format of an exception for ADO.NET 2.0 as the following:

```
<sqlclient message> (provider:<SNIX provider>, error: <SNIX error code> –  
<SNIX error message>)
```

where SNIX is the SQL network transport later used by the SqlClient <namespace>. The exception will be thrown by the data provider via the generic DbException object. In our case, we are recieving the SqlException exception directly, which derives from DbException. This indicates something happening with the data provider.

The <SNIX error message> portion of our error message is:

A request to send or receive data was disallowed because the socket is not connected and (when sending on a datagram socket using a sendto call) no address was supplied.

This is a socket message: WSAENOTCONN (10057):  
<<http://kb.globalscape.com/article.aspx?id=10140>>

This leads me to believe that something is happening internally with the SQL client data provider for Framework 2.0, primarily because it happens with the same code in VB.NET 2005 and C# 2005.

Our connection string is of the form: "Data Source=192.168.1.x,1433;Initial Catalog=databaseName;User ID=userName;Password=password;"

## Sql Server Connection Irritations (again)

Any help tracking this source of the problem and potential solutions would be greatly appreciated.

Thanks,  
Shawn