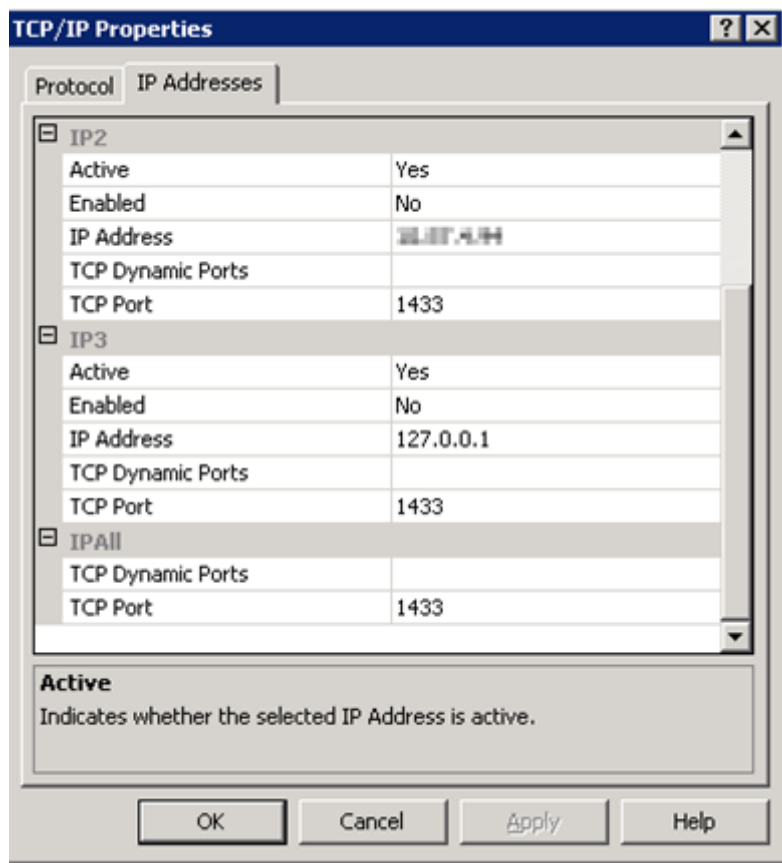


ON WHICH PORT IS SQL SERVER LISTENING ON?

It is a well known fact that whenever SQL Server starts it binds itself with the **IP Address** of the Server and it starts listening on a port assigned to it. In **SQL Server 2005** and above, the port on which SQL Server should listen on is configured using the **SQL Server Configuration Manager**.



The values defined in *TCP Port* under *IPAll* section is the port on which **SQL Server** is configured to listen on.

At times when the **SQL Server** instance is starting up, due to some reasons it will not be able to bind with the port assigned which is usually logged in the **SQL Server Error Log**.

```
server SuperSocket Info: Bind failed on TCP port 1433
```

On this error usually **SQL Server** will start listening on **Named Pipes**. Restarting the **SQL Server** usually helps **SQL Server** to listen on the configured port.

How to check on which port **SQL Server** is listening on at a given point in time? The first place to look for is the **SQL Server Error Log**. When **SQL Server** starts up this information is logged in it.

```

2010-02-16 20:58:43.780 Server      A self-generated certificate was successfully loaded for encryption.
2010-02-16 20:58:43.810 Server      Server is listening on [ 'any' <ipv4> 1433].
2010-02-16 20:58:43.810 Server      Server local connection provider is ready to accept connection on [
2010-02-16 20:58:43.810 Server      Server local connection provider is ready to accept connection on [

```

But on my client's instances the **SQL Server Error Log** is recycled every night. If a server was restarted a week ago, it will be a little time consuming to find out the **Error Log** number and then reading it. The other way to get this information is through the netstat command.

The following command can be used to get the port number of **SQL Server**.

```
netstat -ano | find /i "ProcessIDofSQLServer"
```


The definition of the parameters used for the netstat command is as below.

- -a : Displays all active TCP connections and the **TCP** and UDP ports on which the computer is listening.
- -n : Displays active **TCP** connections, however, addresses and port numbers are expressed numerically and no attempt is made to determine names.
- -o : Displays active **TCP** connections and includes the process ID (**PID**) for each connection. You can find the application based on the PID on the Processes tab in **Windows Task Manager**

In this example, the **SQL Server Process ID** is 2276. The netstat command would be:

```
netstat -ano | find /i "2276"
```

The output of this command will be:



```

C:\Documents and Settings\...>netstat -ano |find /i "2276"
TCP        0.0.0.0:1433          0.0.0.0:*           LISTENING       2276
TCP        0.0.0.0:1434          0.0.0.0:*           LISTENING       2276
TCP        0.0.0.0:1655          0.0.0.0:*           ESTABLISHED     2276

```

The second column in the output refers to *Local Address* and the third column refers to the Foreign Address. The entry after the IP address which begins with a ":" is the port number on which **SQL Server** is listening on. In the above example it is **1433**.

The corresponding entry in the third column i.e. the Foreign Address, has the details of the IP Address and port number of the client which has connected to the **SQL Server** on the port in the *Local Address*. Whenever a client connects to **SQL Server**, one connection which has Local Address and Foreign Address is **ESTABLISHED**.

The rows which have a status of **LISTENING**, correspond to the ports on which **SQL Server** is listening. In the above example it is listening on ports 1433 and 1434 (for **DAC**).

Hence a quick look at the *Local Address* column will tell on which port **SQL Server** is listening on. Please feel free share if you are following some other methods to find out the port details.