Knowledge Base

How to troubleshoot WMI-related issues in Windows XP SP2

PSS ID Number: 875605

Article Last Modified on 11/3/2004

The information in this article applies to:

- Microsoft Windows XP Professional Service Pack 2 (SP2)
- Microsoft Windows Management Instrumentation 1.5

SUMMARY

A number of security lockdown changes in Microsoft Windows XP Service Pack 2 (SP2) may cause problems with Windows Management Instrumentation (WMI), especially in remote scenarios. For example, Windows Firewall is enabled by default in Windows XP SP2. Also, DCOM restrictions in Windows XP SP2 are different from DCOM restrictions in earlier versions of Windows.

IN THIS TASK

Introduction

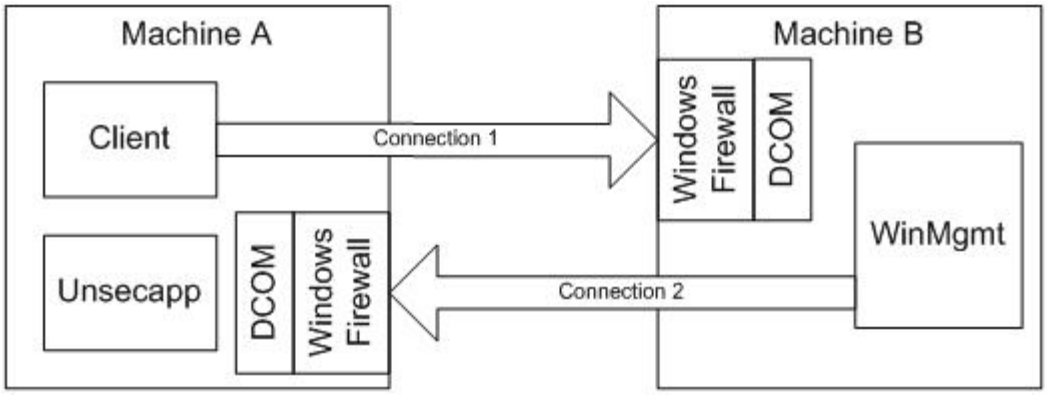
Because of security changes, you may receive "access denied" error messages when you access WMI in Microsoft Windows XP SP2. You may also have problems when you access a non-Windows XP SP2-based computer from a Windows XP SP2-based computer if you use an asynchronous query.

back to the top

Troubleshoot WMI-related issues in Windows XP SP2

When you troubleshoot WMI-related issues, first determine whether the issue is local or remote. To do this, try to access WMI locally to rule out network problems. If the problem occurs even when you access WMI locally, the problem is not related to security changes in Windows XP SP2.

If the problem does not occur when you access WMI locally, the issue may be related to Windows Firewall and to DCOM. When you perform a remote WMI operation from a computer A to computer B, a DCOM connection must be established from computer A to computer B. On computer B, both Windows Firewall and DCOM must be configured to allow the connection. If the WMI operation is synchronous or semi-synchronous, only one connection is required. However, if the WMI operation is asynchronous, another connection from computer B to computer A is required.



To establish connection 1 between computer A and computer B, follow these steps:

1. If Windows Firewall is enabled on computer B, enable the **Windows Firewall: Allow remote administration exception** setting. By default, Windows Firewall is enabled in Windows XP SP2.

   For more information about how to enable this setting, see the Allow for remote administration section.
2. If the user who is making the remote request is not an administrator, make sure that the user has DCOM Remote Launch permissions on computer B.

   For more information, see the Grant DCOM Remote Launch permissions section.

Connection 2 is only required when you use an asynchronous WMI operation. If you can, we recommend that you use a semi-synchronous operation instead. The performance effect is small, and a semi-synchronous operation allows the same functionality but does not require a reverse connection.

If you must use an asynchronous operation, follow these steps:

1. If Windows Firewall is enabled on computer A, open the DCOM port. By default, Windows Firewall is enabled in Windows XP SP2.

   For more information about how to open the DCOM port, see the Open the DCOM port section.
2. On computer A, add the client application to the Windows Firewall Exceptions list so that the reverse connection can be completed.

   The client application is frequently the Unsecapp.exe application. The Unsecapp.exe application is used to send results back to a client in a process that may not have permissions to be a DCOM service. Both scripting and the Microsoft .NET **System.Management** namespace rely on the Unsecapp.exe application to receive the results of asynchronous operations.

   For more information about how to add the client application to the Windows Firewall Exceptions list, see the Add the client application to the Windows Firewall Exceptions list section.
3. If the reverse connection is created as an anonymous connection, grant Remote Launch permissions in DCOM to the anonymous logon account on computer A. The reverse connection is created as an anonymous connection if one of the following conditions is true:
   - Computer B is a member of a workgroup.
   - Computer B is not in the same domain as computer A, and the domain of computer B is not a trusted domain.

   For more information, see the Grant DCOM Remote Launch permissions section.
4. Make the reverse connection as secure as possible. For more information, visit the following Microsoft Developer Network (MSDN) Web site:

   http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/wmi/setting_security_on_an_asynchronous_call.asp

back to the top

Allow for remote administration

1. Click **Start**, click **Run**, type `gpedit.msc`, and then click **OK**.
2. Under **Console Root**, expand **Computer Configuration**, expand **Administrative Templates**, expand **Network**, expand **Network Connections**, expand **Windows Firewall**, and then click **Domain Profile**.

3. Right-click **Windows Firewall: Allow remote administration exception**, and then click **Properties**.
4. Click **Enabled**, and then click **OK**.

back to the top

Grant DCOM Remote Launch permissions

1. Click **Start**, click **Run**, type `DCOMCNFG`, and then click **OK**.
2. In the **Component Services** dialog box, expand **Component Services**, expand **Computers**, and then expand **My Computer**.
3. On the toolbar, click the **Configure My Computer** button.

   The **My Computer** dialog box appears.
4. In the **My Computer** dialog box, click the **COM Security** tab.
5. Under **Launch and Activate Permissions**, click **Edit Limits**.
6. In the **Launch Permission** dialog box, follow these steps if your name or your group does not appear in the **Groups or user names** list:
   a. In the **Launch Permission** dialog box, click **Add**.
   b. In the **Select Users, Computers, or Groups** dialog box, add your name and the group in the **Enter the object names to select** box, and then click **OK**.
7. In the **Launch Permission** dialog box, select your user and group in the **Group or user names** box. In the **Allow** column under **Permissions for User**, select **Remote Launch**, and then click **OK**.

back to the top

Open the DCOM port

Before you enable ports in Windows Firewall, make sure that the **Windows Firewall: Allow local port exceptions** setting in Group Policy is enabled. To do this, follow these steps:

1. Click **Start**, click **Run**, type `gpedit.msc`, and then click **OK**.
2. Under **Console Root**, expand **Computer Configuration**, expand **Administrative Templates**, expand **Network**, expand **Network Connections**, expand **Windows Firewall**, and then click **Domain Profile**.
3. Right-click **Windows Firewall: Allow local port exceptions**, and then click **Properties**.
4. Click **Enabled**, and then click **OK**.

Note You can also use the **Windows Firewall: Define port exceptions** setting to configure local port exceptions.

The DCOM port is TCP 135. To open the DCOM port, follow these steps:

1. Click **Start**, and then click **Control Panel**.
2. Double-click **Windows Firewall**, and then click the **Exceptions** tab.
3. Click **Add Port**.
4. In the **Name** box, type `DCOM_TCP135`, and then type `135` in the **Port number** box.
5. Click **TCP**, and then click **OK**.
6. Click **OK**.

Note You can also type the following command at a command prompt to open a port:

netsh firewall add portopening *[TCP/UDP][Port][Name]*

back to the top

Add a client application to the Windows Firewall Exceptions list

Before you define program exceptions in Windows Firewall, make sure that the **Windows Firewall: Allow local program exceptions** setting in Group Policy is enabled:

1. Click **Start**, click **Run**, type `gpedit.msc`, and then click **OK**.
2. Under **Console Root**, expand **Computer Configuration**, expand **Administrative Templates**, expand **Network**, expand **Network Connections**, expand **Windows Firewall**, and then click **Domain Profile**.
3. Right-click **Windows Firewall: Allow local program exceptions**, and then click **Properties**.
4. Click **Enabled**, and then click **OK**.

Note You can also use the **Windows Firewall: Define program exceptions** setting to configure local program exceptions.

To add a client application to the Windows Firewall Exceptions list, follow these steps:

1. Click **Start**, and then click **Control Panel**.
2. Double-click **Windows Firewall**, and then click the **Exceptions** tab.
3. Click **Add Program**.
4. Locate the application that you want to add, and then click **OK**.
5. Click **OK**.

Note You can also type the following command at a command prompt to add a program to the Windows Firewall Exception list:

netsh firewall add allowedprogram *[<Path>\ProgramName] [ENABLE/DISABLE]*

back to the top

Example

When you try to use the System Information tool, Msinfo32.exe, to connect to a remote a computer that is running Microsoft Windows XP SP2, you receive the following error message:
The connection to *computer name* could not be established. Check to see that the network path name is correct, that you have sufficient permission to access Windows Management Instrumentation, and that Windows Management Instrumentations is installed on the computer.
Note In this message, *computer name* is a placeholder.

To work around this problem, follow the steps that are mentioned in the Allow for remote administration section.

back to the top

REFERENCES

For additional information, visit the following Microsoft Web sites:

http://www.microsoft.com/downloads/details.aspx?FamilyID=4454e0e1-61fa-447a-bdcd-499f73a637d1

For additional information, click the following article number to view the article in the Microsoft Knowledge Base:

875357 Troubleshooting Windows Firewall settings in Windows XP Service Pack 2

back to the top

Additional query words: Windows Firewall WINXP SP2 WMI DCOM

Keywords: kbtshoot kbinfo KB875605
Technology: kbAudDeveloper kbWinXPPro kbWinXPProSearch kbWinXPProSP2 kbWinXPSearch kbWMI150 kbWMISearch

---