

SBClient and Microsoft Windows Terminal Server

(Including Citrix Server)

Contents

1. Introduction
2. SBClient Compatibility Information
3. SBClient Terminal Server Installation Instructions
4. Resolving Performance Problems
5. General Notes

1. Introduction

Microsoft Windows Terminal Services extends the model of distributed computing by allowing PCs to operate in a server-based computing environment. When running Terminal Services, client applications and data processing/data storage operations are executed on the server. Terminal Services is a component of Windows 2003/2008 Server.

Applications and user desktops are transmitted over the network via remote desktop protocol (RDP) and displayed via remote desktop client software. Similarly, print streams, keyboard input, and mouse clicks are also transmitted over the network between the server and the remote desktop client software. Each user logs on to the Terminal Server environment and sees only his or her individual session. Each session is managed transparently by the server operating system and is independent of any other client sessions.

Citrix MetaFrame is a third-party add-on layer to Microsoft Windows Terminal Services from Citrix Systems, Inc. It incorporates the Citrix Independent Computing Architecture (ICA) protocol and provides extended capabilities for client devices, network connections, and local system resources.

To clarify the terminology, Terminal Server is a multi-user server core that provides the ability to host multiple, simultaneous client sessions on Windows 2003/2008. Terminal Server architecture is the foundation for add-on products such as the Citrix MetaFrame environment. Citrix MetaFrame environment is not possible without first enabling and licensing the server core to support Terminal Services. Any application that is supported in the Terminal Server environment should work unchanged in the add-on Citrix environment. Support for Citrix-specific issues should be resolved with Citrix technical support before contacting U2 technical support.

2. SBClient Compatibility Information

SBClient 6 has been tested on Microsoft Windows 2003/2008 with Terminal Server Role enabled.

The following functionality has been verified to work as expected:

- Installation and licensing
- Connectivity configuration to local and remote hosts
- GUI components ActiveX + HTML+ SBX
- Smart Query and output to Word or Excel
- File uploads and downloads using PD.LO/PD.LI
- Connecting SBClient as a published application
- Device licensing
- SBCOM
- Scripts
- Saving .sbc sessions on the Windows desktop

File uploads and downloads using PD.LO/PD.LI

With Citrix MetaFrame, it is possible to specify the local hard drive from which the client connection is initiated, as well as the disk drive on the server. Check the drive letter assignment in "My Computer" from the ICA client.

Note: You can expect a slow transfer rate if you specify the local hard drive because the data must first be transferred to the Citrix server and then to the local hard drive.

SBClient as a published application

Microsoft Windows Terminal Services can provide users with access to a single published application when access to a full Microsoft Windows desktop is not required. SBClient can work as a published application in this context.

You can set up SBClient as a published application in the Citrix Published Application Manager on the server. To set up a new published application, use the **Define the Application** dialog box. Make sure to specify the full path of 'sbclient.exe'. It is best to click the Browse button and select the 'sbclient.exe' by browsing to it. This automatically puts the full path of both Command Line and Working Directory in the proper place.

The default paths are as follows:

Command Line: C:\U2\SBClient6\sbopen\BIN\sbclient.exe

Working Directory: C:\U2\SBClient6\sbopen\BIN

When you open Citrix Program Neighborhood on the client PC, you will find SBClient as an option to start as a published application. If not, add it as such.

Saving .sbc sessions on the Windows desktop

You cannot copy .sbc sessions from one user to another because of ownership and permissions. In fact, we do not recommend using .sbc session files in the Terminal Server environment because certain information is hard-coded when the session file is saved, and this can lead to problems.

3. SBClient Terminal Server Installation Instructions

3.1 Important Notes

- Terminal Services needs to be in Application mode or Installation mode before installing SBClient. Please refer to documentation on installation of shared applications in a Terminal Server environment for the specific Terminal Services versions you are installing.
- Due to various limitations, it is not possible to uninstall SBClient or its user accounts from Microsoft Windows Terminal Services. You must uninstall manually by deleting all related items in the registry, the Program Files directory, and the Documents and Settings directory for each user.

3.2 Installation

Installing SBClient 6 on Microsoft Windows Terminal Services is now significantly easier than in previous versions.

1. Log on to Microsoft Windows Terminal Services as Administrator. Ensure that Terminal Services is in Application or Installation mode before continuing with the installation.
2. Do not run the SBClient setup.exe file directly from its source location. Refer to your Windows Terminal Server documentation to launch the installation.

3. Do not select the option to run SBClient Manager automatically. Using this option produces the message "TU is an invalid account" on sites where the user logs on to multiple machines.
4. After installation is complete, the Terminal Services clients do not need to perform any further installation steps. They need only start SBClient by clicking on the SBClient icon (shortcut) in the SBClient program group, which is available to all users. The first time a user runs SBClient, the necessary installation steps are performed automatically with no user intervention.

4. Resolving Performance Problems

If you should experience performance problems with SBClient on Microsoft Windows Terminal Services, please check the following:

- Ensure that all of your users' session configurations are set up for the file transfer protocol of Network Copy (if your host is on Windows) or FTP (if your host is on UNIX). From the SBClient menu, select Setup > File Transfer and change the default setting of SBZ to Network Copy (for Windows) or FTP (for UNIX). Save this change by selecting File > Save Configuration.

You can also experiment with two registry settings, as detailed below. The location of the registry entry depends on whether you are running a 32-bit or 64-bit version of Windows.

32-bit Windows:

HKEY_LOCAL_MACHINE\SOFTWARE\Rocket Software\SBClient

64-bit Windows:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Rocket Software\SBClient

- Sleptime (default value 10). The Sleptime value determines the number of milliseconds an SBClient thread will wait before checking the socket for incoming data.
- Polltime (default value 0). The Polltime value determines the amount of time SBClient will poll the socket looking for incoming data. The time is calculated by $1/\text{Polltime}$, so a Polltime of 1 will poll for 1 second, a Polltime of 10 will poll for 100 milliseconds, and a Polltime of 0 will result in no polling.

5. General Notes

- Always enable Terminal Services on a server installed on an NTFS file system partition, as NTFS provides the security necessary for users in a multi-session environment.
- Terminal Services provides two options when closing a client: disconnect or log off.
 - Disconnect - Using disconnect leaves the session running on the server, meaning you can reconnect to the server and resume the session. If you are running a task on the server, such as a time-consuming database query, you can start the task and disconnect from the session. Later on, you can log on to the server again, re-enter the session, and either resume the task or check the results. This is especially helpful when operating over a remote access connection on a long-distance toll line. Network administrators can limit the amount of time a disconnected session is allowed to be left on the server. Check with your administrator for information about limits to the length of sessions.
 - Logging Off - Ending the session is known as logging off, which completely terminates the session running on the server. Any applications running within the session are closed and unsaved changes made to open files are lost. The next time the user logs on to the server, a new session is created.
 - **Note:** When users leave disconnected sessions running, they may receive the error message “TU is not a valid account” when reconnecting and launching SBClient. This occurs when Terminal Services cannot determine the original session to reconnect to, and creates a second session with the same user credentials. The second session cannot access the shared SBClient application files and will fail. The resolution is to have the system administrator log off all sessions with this user’s credentials, or wait for the appropriate session timeout to occur.