

The Desktop Sharing Handbook

Brad Hards



The Desktop Sharing Handbook

Contents

1	Introduction	5
2	The Remote Frame Buffer protocol	6
3	Using Desktop Sharing	7
3.1	Desktop Sharing Main Window	7
3.1.1	Connection Details	7
3.1.2	Unattended Access	8
3.1.3	Transfer Login Information	8
3.1.4	Quit Desktop Sharing	8
3.2	Configuring Desktop Sharing	8
3.3	Connecting to Desktop Sharing	10
4	Credits and License	11

Abstract

Desktop Sharing is a server application that allows you to share your current session with a user on another machine, who can use a VNC client to view or even control the desktop.

Chapter 1

Introduction

Desktop Sharing is a server application that allows you to share your current session with a user on another machine, who can use a VNC client to view or even control the desktop.

You would typically use Desktop Sharing with the KDE VNC client, which is Remote Desktop Connection, since it closely matches the special features of Desktop Sharing.

Desktop Sharing doesn't require you to start a new X session - it can share the current session. This makes it very useful when you want someone to help you perform a task.

Please report any problems or feature requests to the KDE mailing lists or file a bug at <https://bugs.kde.org>.

Chapter 2

The Remote Frame Buffer protocol

This chapter provides a brief description of the Remote Frame Buffer protocol used by Desktop Sharing and by other compatible systems. If you are already familiar with Remote Frame Buffer, you can safely skip this chapter.

The high level implementation of a system using the Remote Frame Buffer protocol is known as Virtual Network Computer, or more often just as VNC.

Remote Frame Buffer (or RFB for short) is a simple protocol for remote access to graphical user interfaces. It works at the frame-buffer level, which roughly corresponds to the rendered screen image, which means that it can be applied to all windowing systems (including X11, Mac[®] OS and Microsoft[®] Windows[®]). Remote Frame Buffer applications exist for many platforms, and can often be freely re-distributed.

In the Remote Frame Buffer protocol, the application that runs on the machine where the user sits (containing the display, keyboard and pointer) is called the client. The application that runs on the machine where the framebuffer is located (which is running the windowing system and applications that the user is remotely controlling) is called the server. Desktop Sharing is the KDE server for the Remote Frame Buffer protocol. Remote Desktop Connection is the KDE client for the Remote Frame Buffer protocol.

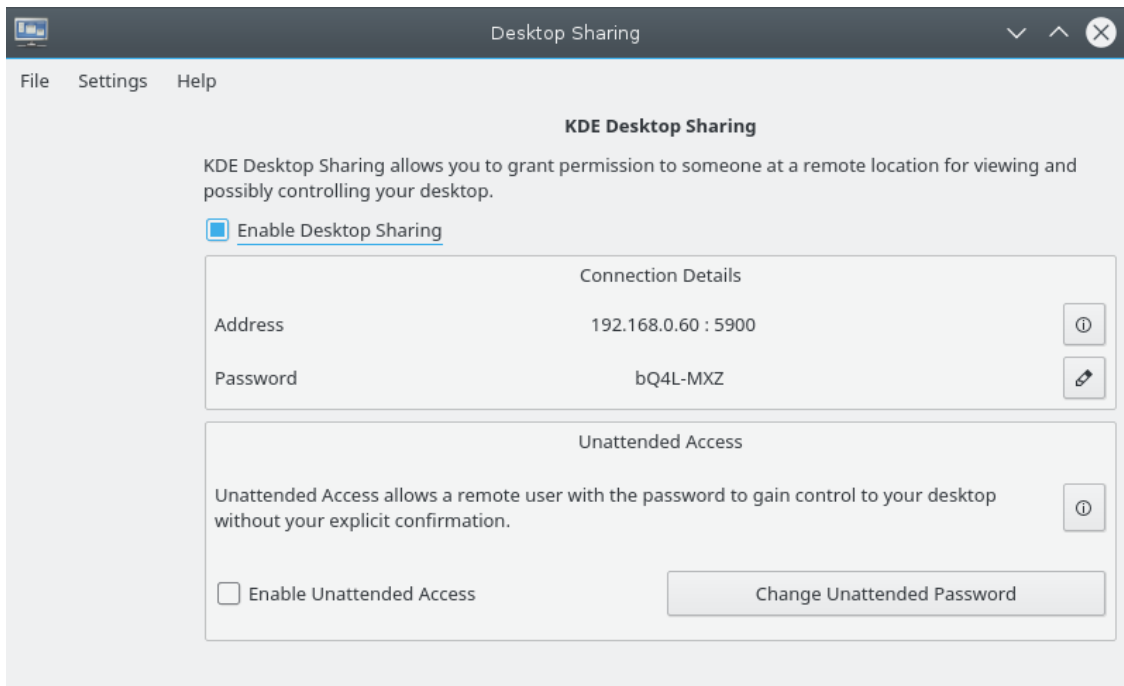
It takes a reasonable amount of network traffic to send an image of the framebuffer, so Remote Frame Buffer works best over high bandwidth links, such as a local area network. It is still possible to use Desktop Sharing over other links, but performance is unlikely to be as good.

Chapter 3

Using Desktop Sharing

3.1 Desktop Sharing Main Window

It is very easy to use Desktop Sharing - it has a simple interface, as shown in the screenshot below.



When you want to allow someone to access your desktop, you have to enable the checkbox **Enable Desktop Sharing**, which will start the server.

3.1.1 Connection Details

The **Address** contains the address of your computer and the port number, separated by a colon. The address is just a hint - you can use any address that can reach your computer. Desktop Sharing tries to guess your address from your network configuration, but does not always succeed in doing so. If your computer is behind a firewall it may have a different address or be unreachable for other computers.

You can change the port on the **Network** page in the configuration dialog.

The next field is prefilled with an automatically generated password. Click in the icon at the right of the field to change the password.

3.1.2 Unattended Access

Any remote user with the desktop sharing password will have to be authenticated. If unattended access is activated, and the remote user provides the password for unattended mode, desktop sharing access will be granted without explicit confirmation.

By default the password for this mode is empty, to change that click on the button and enter a password.

If unattended access is allowed, then you should probably specify a password.

If the machine is a server and you are using Desktop Sharing for remote administration, you probably want to use unattended access.

3.1.3 Transfer Login Information

Desktop Sharing has no invitation feature any more as in previous versions. So you have to transfer the login information yourself using email or a personal invitation.

If you cannot encrypt the email (or otherwise secure the link), sending a password by email is a very serious security risk, since anyone can read the password and address from the email as it passes over the network. This means that they can potentially take control of your machine.

If you cannot encrypt the email message, it may be better to use a personal invitation, telephone the person you are giving access to, verify the identity of that person, and provide the required information that way.

NOTE

Desktop Sharing uses the normal RFB password system, which does not transfer your password in the clear across the network. Instead, it uses a challenge-response system. This is reasonably secure, as long as the password is securely guarded.

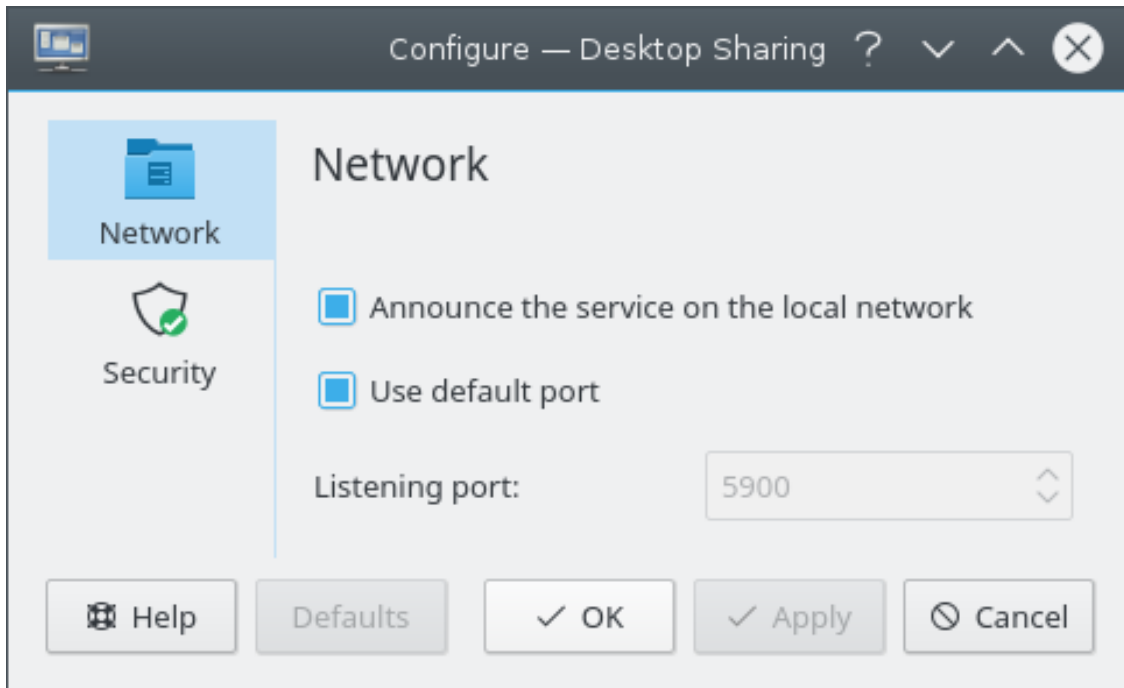
3.1.4 Quit Desktop Sharing

If you close the Desktop Sharing main window by clicking on the window close icon or using the shortcut **Alt+F4** the server keeps running, which is indicated by an icon in the system tray. To stop Desktop Sharing either use **File** → **Quit** in the main window or right click on the icon in the system tray and select **Quit**.

3.2 Configuring Desktop Sharing

In addition to the main Desktop Sharing interface shown and described above, you can also control Desktop Sharing using the **Configure...** on the Desktop Sharing main window. The Desktop Sharing configuration has two pages, as shown in the screenshot below:

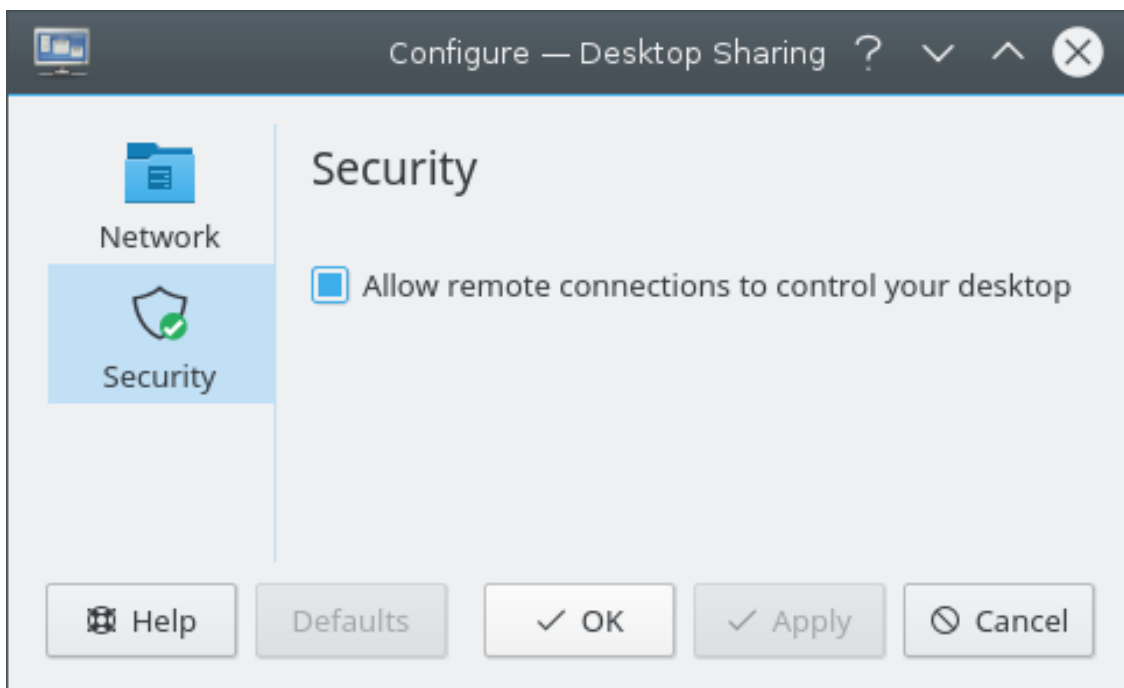
The **Network** page allows control over the port that Desktop Sharing uses, as shown below.



The **Announce service on the local network** checkbox controls whether Desktop Sharing announces the service over the local network using Service Location Protocol. This is normally a good idea, but only works really well with a Service Location Protocol aware client, such as Remote Desktop Connection.

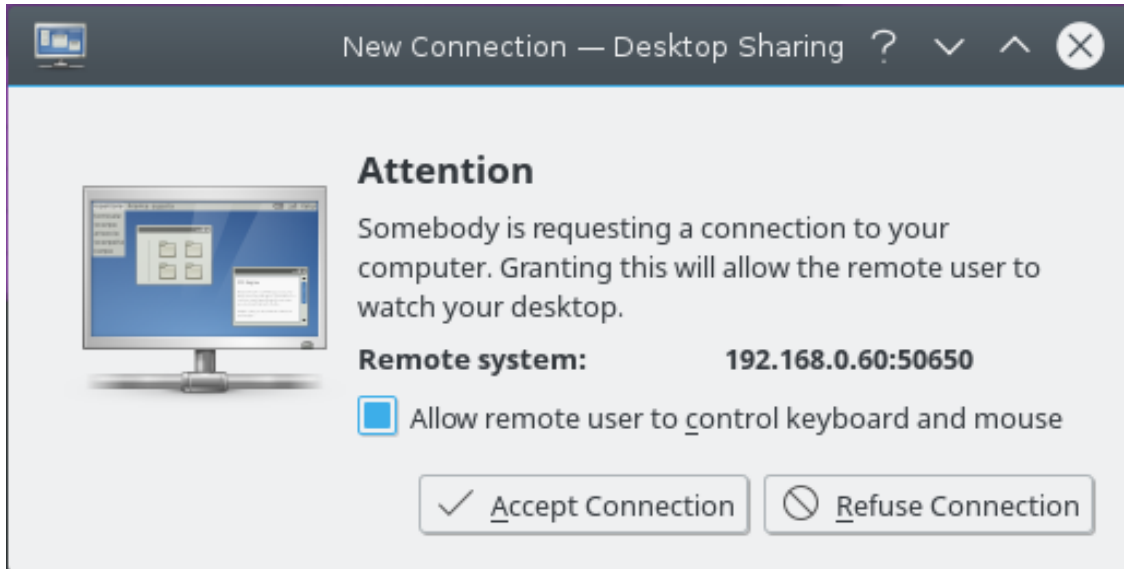
If you select the **Use default port** checkbox, then Desktop Sharing will locate a suitable port. If you deselect this checkbox, you can specify a particular port. Specifying a particular port may be useful if you are using port-forwarding on the firewall. Note that if Service Location Protocol is turned on, this will automatically deal with identifying the correct port.

The **Security** page allows you to configure whether the person connecting to the Desktop Sharing server can control the desktop, or only view.



3.3 Connecting to Desktop Sharing

When someone connects to Desktop Sharing on your machine, you will get a pop-up notification that looks like the following screenshot, unless you are accepting unattended access without confirmation.



If you **Accept Connection**, the client can proceed to authenticate, which requires the correct password for a login. If you **Refuse Connection**, then the attempt to connect will be terminated.

The **Allow remote user to control keyboard and mouse** check box determines whether this client can only observe, or can take control of your machine.

Chapter 4

Credits and License

Desktop Sharing

Program copyright 2002 Tim Jansen tim@tjansen.de

Contributors:

- Ian Reinhart Geiser geiseri@kde.org

Documentation Copyright (c) 2003 Brad Hards bradh@frogmouth.net

This documentation is licensed under the terms of the [GNU Free Documentation License](#).

This program is licensed under the terms of the [GNU General Public License](#).