



Konstantin Androsov, Fotolia

Supporting Windows terminal clients from Linux with xrdp

# TERMINAL BOSS

Xrdp helps your Windows terminal clients connect to Linux. **BY KEN HESS**

**W**indows-to-Linux remote desktop connectivity is often bandwidth intensive, insecure, and difficult to configure. What if a remote Windows user could establish a connection to a Linux system as easily as connecting to a Windows system – and using the same application used to connect to Windows?

Xrdp is an open source implementation of RDP (Remote Desktop Protocol), the protocol used with Windows Terminal Services for native Windows desktop connectivity. The xrdp package provides RDP functionality, along with an X server capable of accepting connections from rdesktop [1] and Windows Terminal Server clients.

Once connected and authenticated to the Linux RDP server, the remote user is presented a Linux graphical desktop. The best part is you don't have to run an X server on your Windows computer or export your Linux display to the Windows computer.

Setting up the RDP server on Linux is easy and only takes a few minutes. For

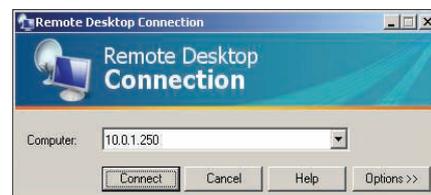
the new Linux user or Windows Administrator, you'll need the following:

- a standard Linux user account,
- console or remote access to the Linux server,
- access to the root user account.

Also, you'll need to navigate the Linux filesystem, use some basic DOS-like commands, and understand basic network connectivity. It also helps if you

**Table 1: Linux Runlevels**

0	Halt
1	Single-user mode
2	Multiuser, without NFS
3	Full multiuser mode
4	Unused
5	X11 (GUI)



**Figure 1: Remote Desktop Connection window.**



Figure 2: Enter your Linux system User name and Password.

know a little about RDP and optimizing client settings.

## Installation and Setup

First, download the xrdp source package [2], then unzip and untar the package:

```
$ tar zxvf xrdp-0.4.1.tar.gz
```

All source files are now in the *xrdp-0.4.1* directory. Then you need to change into that directory and compile:

```
cd xrdp-0.4.1
$ make
```

Use *sudo* or *su* to become root and run:

```
# make install
```

The executable files, scripts, and libraries install to */usr/local/xrdp*, and the configuration files install to */etc/xrdp*.

The documentation is a bit unclear as to what to do next. Although I decided against changing any parameters in the *sesman.ini* and *xrdp.ini* files under */etc/*

*xrdp* and approached the situation as if I were a Windows administrator, assuming things would work out of the box without a lot of tweaking, it didn't take long for this Windows approach to frustrate me, so I began poking around in the */usr/local/xrdp* directory a bit and speaking via email with one of the xrdp developers.

Eventually, I discovered the answers myself. Once the system is installed, you need to issue the following commands as root:

```
# cd /usr/local/xrdp
# cp xrdp_control.sh \
/etc/init.d/xrdp_control
# chkconfig --add xrdp_control
# chkconfig xrdp_control on
# service xrdp_control start
```

The *xrdp\_control.sh* script controls xrdp start, stop, and restart functionality. When I copied the script to the */etc/init.d* directory, I changed its name because there, the *.sh* is superfluous and typically not used.

The *chkconfig* command provides a way of maintaining, updating, and querying information for system services by alleviating the need to manually manipulate these services.

To create a new startup service, copy its control script to */etc/init.d*, as shown above, then use *chkconfig* to add the service. The line *chkconfig xrdp\_control on* defines the runlevels for the service (Table 1). For instance, *# chkconfig -level 35 xrdp\_control on* sets *xrdp\_control* to run in runlevels 3 and 5. The final command starts the services *sesman* and *xrdp*.

To connect to your Linux computer's RDP service from a Windows computer, select *Start | Programs | Accessories | Communications | Remote Desktop Connection*. (On some Windows computers, this is under *Start | Programs | Accessories | Remote Desktop Connection*). Enter the name or IP address of your Linux computer and click *Connect*, as shown in Figure 1.

The RDP client opens a new dialog, as shown in Figure 2, prompting you for a *User name* and *Password*. This username and password corresponds to your user account on the Linux server system.

Enter your valid Linux username and password, and click *OK*. Even if you join your Linux computers to an Active Directory domain, you'll need to add each user to the Linux systems you want to access via RDP.

To add users, first create a group for all RDP users:

```
# groupadd rdpusers
# useradd -g rdpusers ajones
# passwd ajones
```

If the user already exists on the system, the *usermod* command adds *ajones* to the *rdpusers* group:

```
# usermod -G rdpusers ajones
```

Although not specifically required for access, a special user group simplifies administration for that group. Enter your username and password, then click *OK*.

The screen shown in Figure 3 appears, which is the Session Manager Connection Log showing the negotiation between client and server. Note the Session

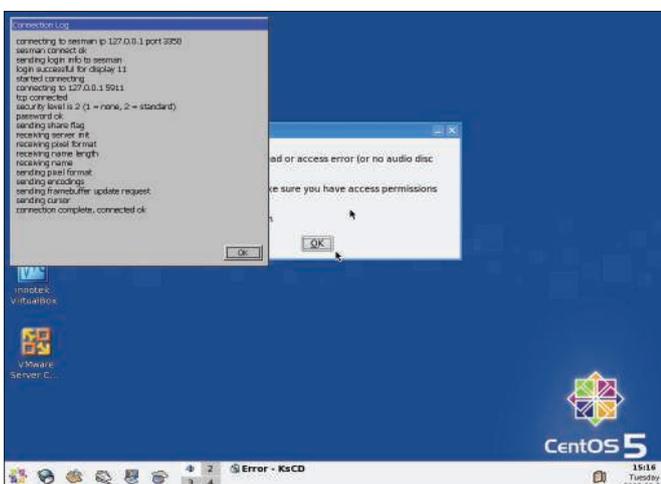


Figure 3: The xrdp Session Manager Connection Log.

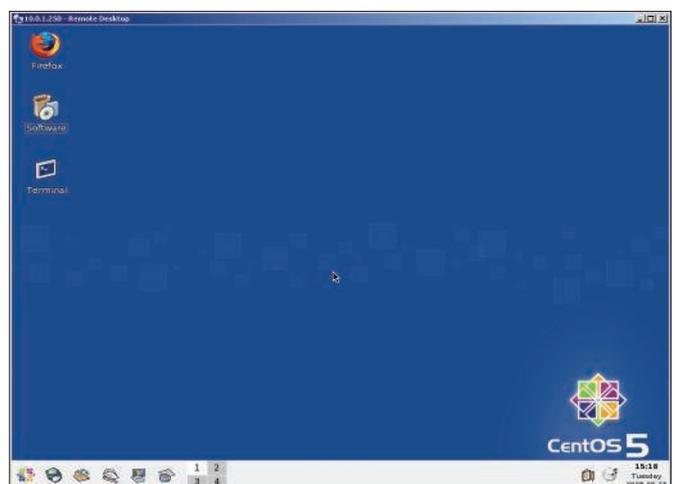


Figure 4: A desktop connection open for business.

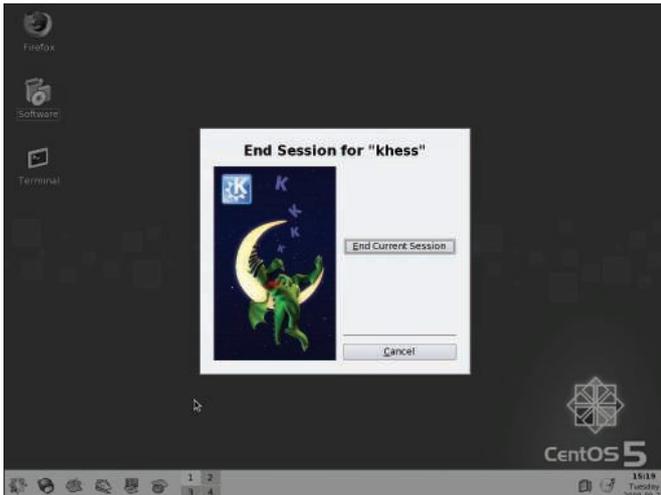


Figure 5: VNC's End Session dialog.

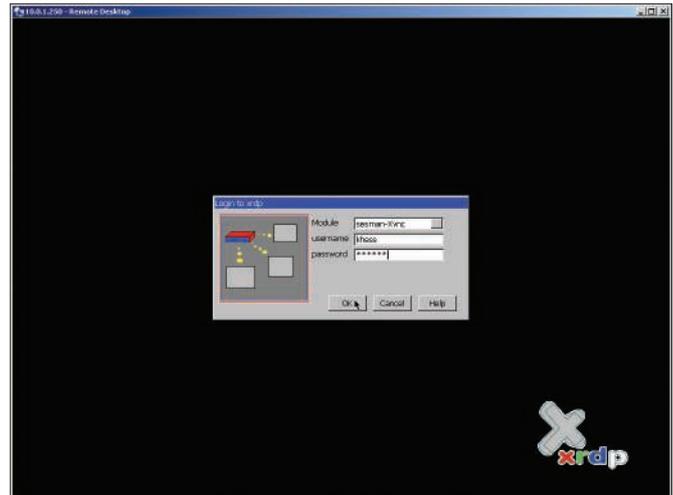


Figure 6: The xrdp login screen.

Manager first connects to the RDP port, then to the VNC port to present the desktop to you. Figure 4 shows my CentOS 5 desktop.

When disconnecting from a Linux RDP session, you'll see the screen shown in Figure 5. Notice that the only two actions available are *End Current Session* and *Cancel*, which is a standard feature of VNC remote connections.

## How It Works

The Linux server runs *xrdp* and *sesman*, listening for RDP connections. Once a Windows Terminal Server client at-

tempts a connection to the RDP server, the server and client negotiate an encryption level; keys and server capabilities are exchanged.

The client chooses the session color depth and screen resolution.

If the user enters a username and password in the client software, authentication begins; otherwise, the user selects the module from a drop-down list and enters a username and password at the xrdp login screen, as shown in Figure 6.

The *libvnc* module loads, and a TCP connection is made to *127.0.0.1* or the IP

Address specified in the */etc/xrdp/xrdp.ini* file.

The user's login credentials, screen resolution, and color depth pass to *sesman* for authentication.

If *sesman* finds a running session with the specified resolution and color depth, it returns that display to the user; otherwise, it starts a new *xvnc* instance with the user's desktop specifications.

## Summary

Xrdp is easy to install, configure, customize, and use. To work with this capable product, you don't need years of Linux experience. With nothing to install on the Windows side, you can't go wrong with xrdp.

Although I haven't performed any extensive network speed or bandwidth analysis with this product, because it's RDP, you can be sure that performance begins to suffer as the number of simultaneous users approaches the two dozen mark. However, I really like xrdp, and it is now part of my standard cross-platform arsenal. ■

## Xrdp FAQ

Some useful facts about xrdp:

**Does RDP provide a remote control session?**

No, it is a remote terminal session, just like a Windows Terminal Server client session.

**Why don't I have to run an X server on Windows and export my display?**

Because the Linux server supplies a desktop session through the VNC protocol, you won't need to run an X server.

**Do I need a VNC server to use xrdp?**

If you aren't going to use X11rdp, you will need a VNC server; however, you can only connect using one of those two protocols. On the Windows computer, you do not need a VNC client or VNC server.

**If I want to use X11rdp, should I compile it or use the binary version from xrdp.org?**

You can do either. If you are lucky, the binary will work for you because recompiling X is time consuming and painful, and it can cause your X server to be de-optimized, or worse, not to work at all. You'll

find the binary at <http://server1.xrdp.org/temp>.

**What are the prerequisites for installing xrdp?**

You need *make*, *gcc*, PAM headers, and OpenSSL headers.

**Are my Windows-to-Linux RDP sessions encrypted?**

Yes, that is the RDP standard. Available encryption includes 40-bit one-way, 40-bit two-way, and 128-bit two-way for low, medium, and high encryption, respectively.

**I notice you use the KDE desktop. Is that my only option?**

No, you may run any window manager you like, but it must exist on the Linux system and set up in the */usr/local/xrdp/startwm.sh* file. Move the window manager you want to the top of the file, and issue a force-reload to xrdp as root with:

```
# service xrdp_control ↵
force-reload.
```

Don't run the *startwm.sh* file from the command line.

## INFO

[1] Rdesktop: <http://www.rdesktop.org>

[2] Xrdp: <http://xrdp.sourceforge.net>

## THE AUTHOR

Ken is a freelance technical writer and journalist. He covers a variety of open source topics, including Linux, databases, and virtualization. Reach him via his website at [www.kenhess.com](http://www.kenhess.com). Thanks to Jay Sorg of xrdp and Matt Chapman of the rdesktop project for their help with this article.