Please use the PDF Version of this Procedure if you wish take this procedure home.

Requirements - Remote Windows Desktop

- An ssh client or tunnel manager
  - Windows - puTTy
  - Linux - openssh, puTTY, gstm
  - Mac OSX - openssh, SSH Tunnel Manager
- The name of the target/remote computer. You can also use either Accelerator Terminal Servers *opswin1 opswin2*.
- You must be an Administrator or in the Remote Desktop Users group of the target/remote computer
- The target/remote computer must be Windows 7, Vista, or XP
- A ***Remote Desktop Client*** installed for your Operating System. If not, please refer to the last selection Remote Desktop Client Installation.

Quick Instructions

Windows - Quick Instructions

The PuTTy configuration settings are as follows:

```
Session->Hostname: login.jlab.org
SSH->Tunnels->Source Port: 3391
SSH->Tunnels->Destination: <computername>:3389
```

In Windows Remote Desktop Client:

```
Computer: localhost:3391
```

Linux - Quick Instructions

You can run the RDP Tunnel Script for Linux: rdptunnel and it will do everything for you.

Alternately, can do it from the command line:

Open an ssh tunnel:

```
ssh  -f -N -C -L 3391:<computername>:3389 login.jlab.org"
```

Start the rdesktop session:

```
rdesktop -g "1024x768 -a 16 <computername>
```

Mac OS X - Quick Instructions

Open an ssh tunnel:

```
ssh  -f -N -C -L 3391:<computername>:3389 login.jlab.org"
```

In Remote Desktop Client:

```
Computer: localhost:3391
```

Remote Desktop from On-Site

If you are on-site on one of the Jlab networks, you only need to configure the ***Remote Desktop Client*** with the hostname of the system to which you want to connect. You will also need to set the display resolution to something reasonable.

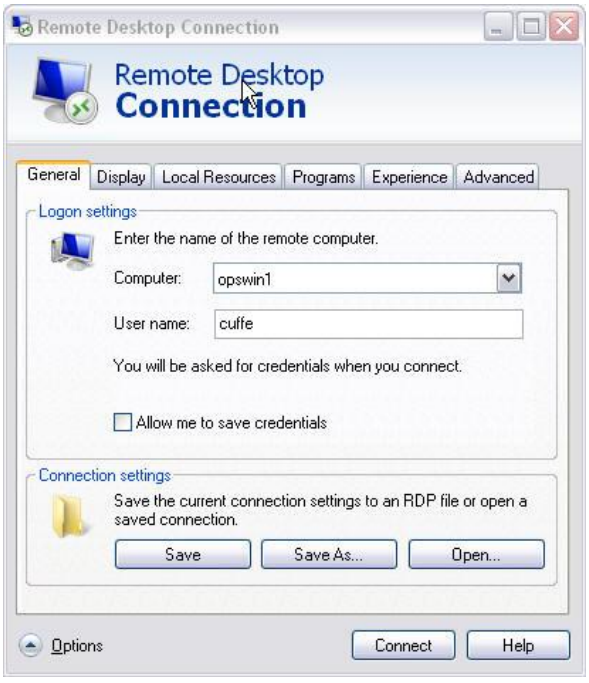
Windows - Remote Desktop from On-Site

1/ Start the Desktop Client - Windows

```
Start>>Programs>Accessories>>Remote Desktop Connection
```

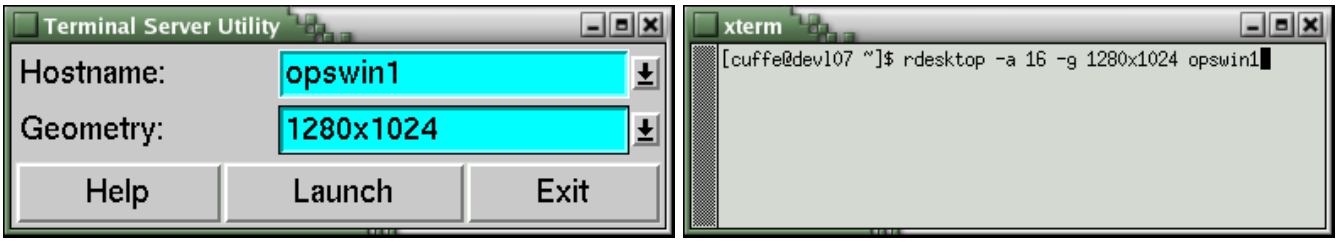
2/ Configure the Client

- On the ***General*** tab, fill in "Computer:" with your desktop hostname or use one the Accelerator Terminal Servers ***opswin1*** or ***opswin2***.
- On the ***Experience*** tab, select the proper speed based on your connection. If you are onsite, this should be set to "LAN (10 Mbps or higher)
- Set the display resolution to something reasonable or just leave it at "Full Screen."



Linux - Remote Desktop from On-Site

You can connect to a remote windows system from linux using a locally developed program called ***tslaunch*** or from the command line invoking rdesktop directly. Once again, you can specify you own desktop or by using one of the Accelerator Terminal Servers *opswin1 opswin2*.



Linux with tslaunch

You can invoke tslaunch from an xterm simply by typing "tslaunch" or you can select it from the Ops Menu (Windows Terminal Server). Once the gui is up, you need only specify the hostname and the geometry. If you need more options, you should use the command line utility described in the next section.

Linux with Command Line rdesktop

You can invoke rdesktop directly from the command line. You can get a full overview by reading the man page (man rdesktop), but the most common options are color depth (-a) and geometry (-g)

```
Examples:

     rdesktop -a 16 -g 1024x768 sftcuffe

     rdesktop -a 16 -g 1280x1024 opswin1

     rdesktop -a 16 -g 800x600 opswin2

[cuffe@devl07 ~]$ rdesktop -h

rdesktop: A Remote Desktop Protocol client.

Version 1.6.0. Copyright (C) 1999-2008 Matthew Chapman.

See http://www.rdesktop.org/ for more information.


Usage: rdesktop [options] server[:port]

   -u: user name

   -d: domain

   -s: shell

   -c: working directory

   -p: password (- to prompt)

   -n: client hostname

   -k: keyboard layout on server (en-us, de, sv, etc.)
```

```
-g: desktop geometry (WxH)

-f: full-screen mode

-b: force bitmap updates

-L: local codepage

-A: enable SeamlessRDP mode

-B: use BackingStore of X-server (if available)

-e: disable encryption (French TS)

-E: disable encryption from client to server

-m: do not send motion events

-C: use private colour map

-D: hide window manager decorations

-K: keep window manager key bindings

-S: caption button size (single application mode)

-T: window title

-N: enable numlock syncronization

-X: embed into another window with a given id.

-a: connection colour depth

-z: enable rdp compression

-x: RDP5 experience (m[odem 28.8], b[roadband], l[an] or hex nr.)

-P: use persistent bitmap caching

-r: enable specified device redirection (this flag can be repeated)

-0: attach to console

-4: use RDP version 4

-5: use RDP version 5 (default)
```

Mac - Remote Desktop from On-Site

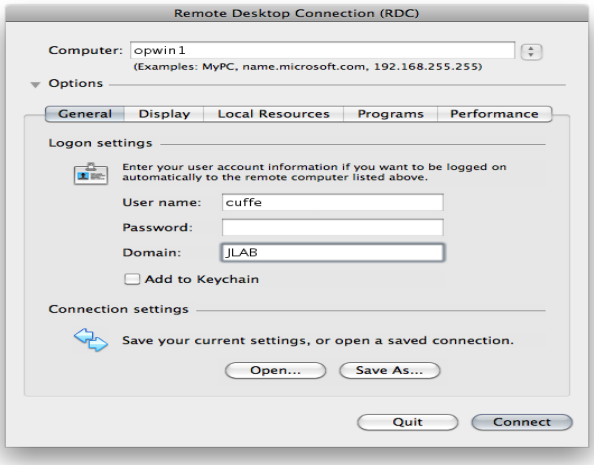
1/ Start the Desktop Client - Mac

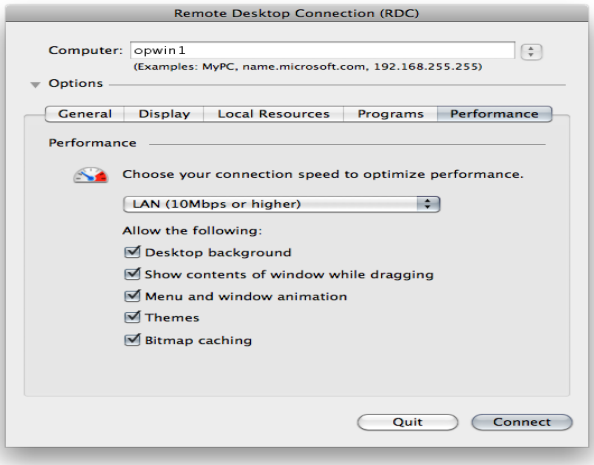
```
Finder>>Applications>>Remote Desktop Connection>>Remote Desktop Connection
```

2/ Configure the Client

- On the ***General*** tab, fill in "Computer:" with your desktop hostname or use one the Accelerator Terminal Servers ***opswin1*** or ***opswin2***.

- On the ***Performance*** tab, select the proper speed based on your connection. If you are on-site, this should be set to "LAN (10 Mbps or higher)
- Set the display resolution to something reasonable or just leave it at "Full Screen."





Remote Desktop from Off-Site (Home or Travel)

The procedure to to connect to a remote windows desktop differs slightly when your are off-site because you must use ***ssh*** to create a tunnel to your system through the jlab firewall. This will be discuss in detail for each platform.

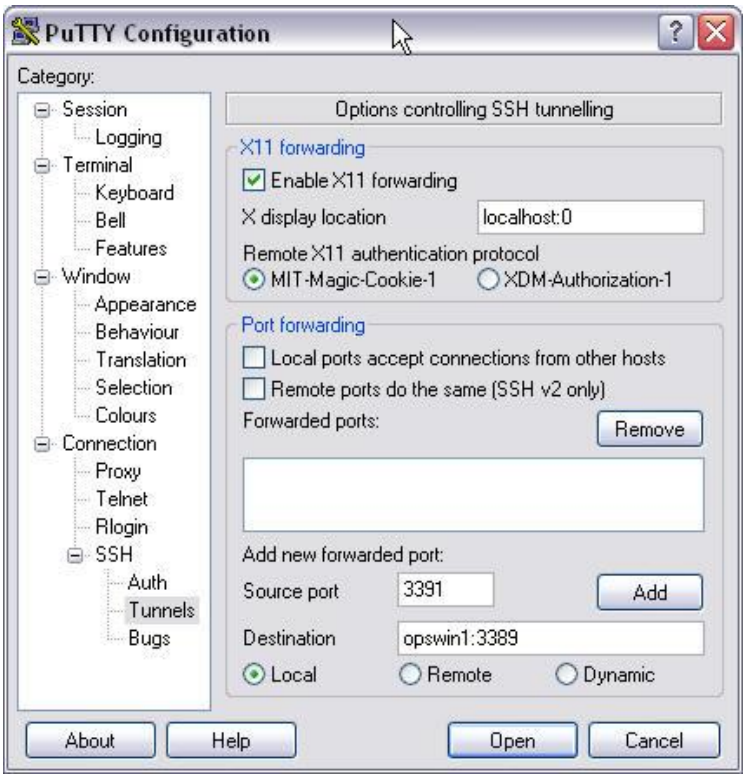
Windows - Remote Desktop from Off-Site

Create an SSH Tunnel using puTTy

1/ Open a new puTTy window

2/ Under "Category" on the left, expand "SSH" and scroll down and click "Tunnels"

- Click "X11"
- On the right, check "Enable X11 forwarding"
- On the right under "Source port" type "3391"
- Under "Destination" type ":3389" replacing with the Target/Remote Computer's Name.
- Now click "Add" and a new entry will appear in the "Forwarded Ports" box.

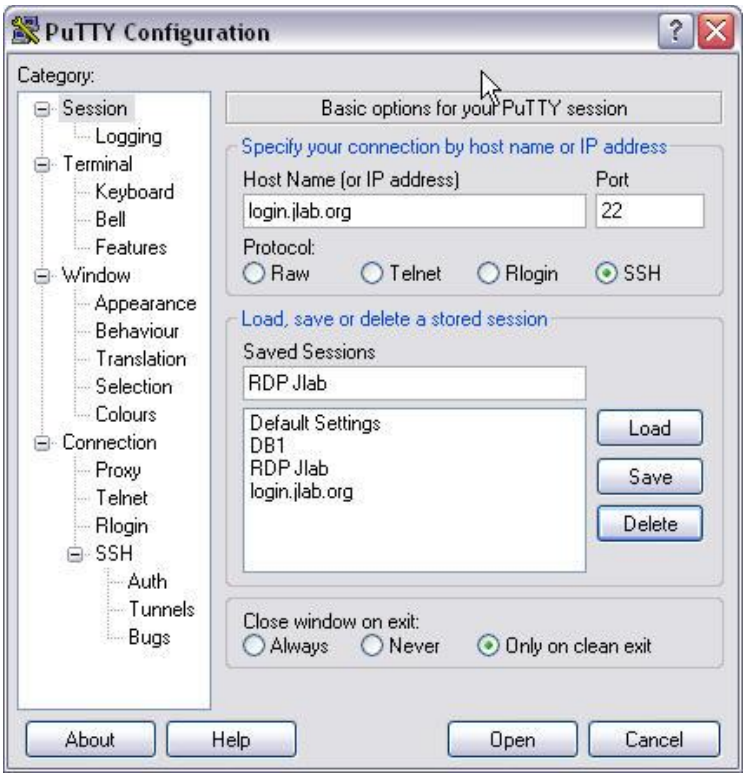


3/ Under "Category" Click "Session"

- On the right under "Host Name (or IP address)" type "login.jlab.org" (without the quotes)
- On the right under "Saved Sessions" type the name you want to use "RDP Jlab" (without the quotes)
- Click "Save" and "RDP Jlab" should appear in the list below "Saved Sessions"

4/ Login by double-clicking "RDP Jlab" under "Saved Sessions"

- Log in with your CUE username and password
- You can now minimize or otherwise forget about puTTy



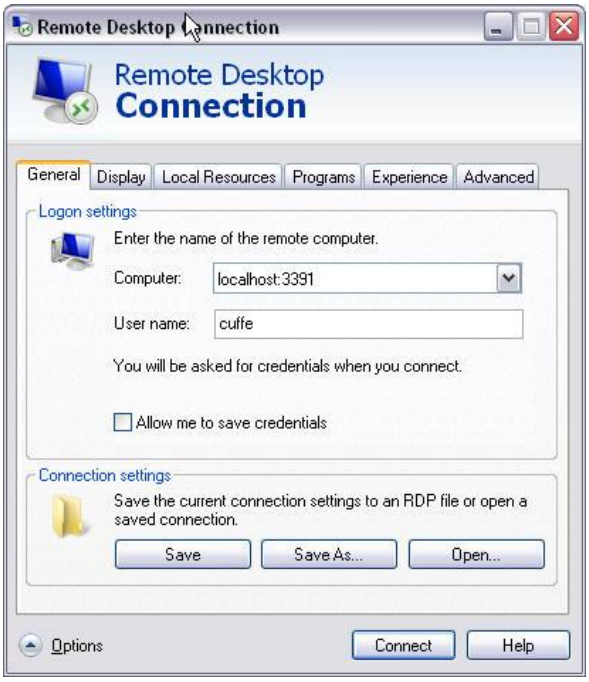
Start the Remote Desktop Client - Windows

1/ Start the Desktop Client

```
Start>>Programs>Accessories>>Remote Desktop Connection
```

2/ Configure the Client

- On the ***General*** tab, fill in "Computer:" with "localhost:3391.
- On the ***Experience*** tab, select the proper speed based on your connection.
- Set the display resolution to something reasonable or just leave it at "Full Screen."



Linux - Remote Desktop from Off-Site

Note that you can run the RDP Tunnel Script for Linux: rdptunnel and it will do everything for you. You just need to copy it to your local computer and execute it with the hostname of the windows machine to which you want to connect.

Create an SSH Tunnel using puTTY, gstm or command line.

There are several tools available to create an SSH tunnel on Linux. You can use puTTY, gstm (gnome ssh tunnel manager) or just the command line.

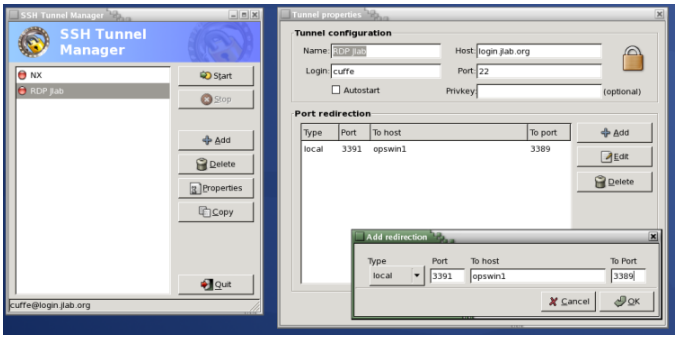
SSH Tunnel with puTTy

Follow the directions outlined for puTTy previously.

SSH Tunnel with gstm

1/ Invoke "gstm" buy issuing the command gstm from an xterm.

2/ Click "Add" to add a new Tunnel



- Under "Name" put anything you want or "RDP Jlab"
- Under "Login: put your username
- Under "Host: put login.jlab.org
- Leave "Port" as 22
- Now click "Add" and a new box will appear to add a "Forwarded Port."
    - Under "Port: put 3391
    - Under "To Host: put the of the system to which you want to connect
    - Under "To Port: put 3389
    - Click "OK" to add the new forwarded port
- Click "OK" to add a the new Tunnel

4/ Login by double-clicking "RDP Jlab" or by Clicking "Start"

- Log in with your CUE username and password
- You can now minimize or otherwise forget about gstm

SSH Tunnel from Linux Command Line

1/ Open an ssh connection using the ssh command. Note that you should use the name of the system to which you want to connect for .

```
ssh  -f -N -C -L 3391:<computername>:3389 login.jlab.org
```

```
Examples:

       ssh  -f -N -C -L 3391:opswin1:3389 login.jlab.org

       ssh  -f -N -C -L 3391:sftcuffe:3389 login.jlab.org
```

2/ Kill any left over ssh tunnels when you are done.

```
ps -ef | grep ssh | grep 3391

kill -9 PID
```

Start the Remote Desktop Client - Linux

You invoke rdesktop directly from the command line. Note that you will be connecting to "localhost:3391" (The port you forwarded in the previous step) rather than the hostname of the computer. You can get a full overview by reading the man page (man rdesktop), but the most common options are color depth (-a) and geometry (-g)

```
Examples:

     rdesktop -a 16 -g 1024x768 localhost:3391

     rdesktop -a 16 -g 1280x1024 localhost:3391

     rdesktop -a 16 -g 800x600 localhost:3391

[cuffe@devl07 ~]$ rdesktop -h

rdesktop: A Remote Desktop Protocol client.

Version 1.6.0. Copyright (C) 1999-2008 Matthew Chapman.

See http://www.rdesktop.org/ for more information.


Usage: rdesktop [options] server[:port]

   -u: user name

   -d: domain

   -s: shell

   -c: working directory

   -p: password (- to prompt)

   -n: client hostname

   -k: keyboard layout on server (en-us, de, sv, etc.)

   -g: desktop geometry (WxH)

   -f: full-screen mode

   -b: force bitmap updates

   -L: local codepage

   -A: enable SeamlessRDP mode

   -B: use BackingStore of X-server (if available)

   -e: disable encryption (French TS)
```

```
-E: disable encryption from client to server

-m: do not send motion events

-C: use private colour map

-D: hide window manager decorations

-K: keep window manager key bindings

-S: caption button size (single application mode)

-T: window title

-N: enable numlock syncronization

-X: embed into another window with a given id.

-a: connection colour depth

-z: enable rdp compression

-x: RDP5 experience (m[odem 28.8], b[roadband], l[an] or hex nr.)

-P: use persistent bitmap caching

-r: enable specified device redirection (this flag can be repeated)

-0: attach to console

-4: use RDP version 4

-5: use RDP version 5 (default)
```

Mac - Remote Desktop from Off-Site

Note that you can run the RDP Tunnel Script for Mac: mactunnel.command and it will do everything for you. You just need to copy it to your local computer (Right click and select "Save As") and set it to have execute permissions(chmod +x mactunnel.command). You can also create an alias on you dock for quick launch.

Create an SSH Tunnel using stm or command line.

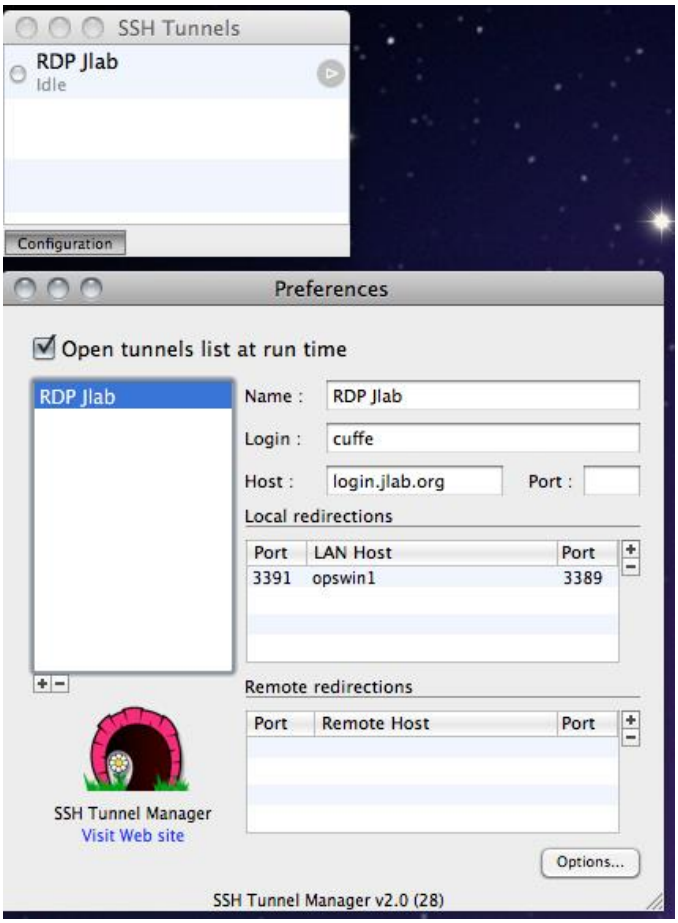
There are several tools available to create an SSH tunnel on a Mac. You can use SSH Tunnel Manager or just the command line.

SSH Tunnel with SSH Tunnel Manager

1/ Invoke "stm" through Finder>Applications>SSH Tunnel Manager

2/ Click "Configuration" and then the plus sign under the list of tunnels box to add a new Tunnel.

- Under "Name" put anything you want or "RDP Jlab"
- Under "Login: put your username
- Under "Host: put login.jlab.org
- Leave "Port" as 22
- Now click the plus sign for "Local redirections"
    - Under "Port: put 3391
    - Under "LAN Host: put the of the system to which you want to connect
    - Under Second "Port: put 3389
- Click "the window X" to close the Configuration dialogue.



4/ Login by double-clicking "RDP Jlab" or by Clicking the play button.

- Log in with your CUE username and password
- You can now minimize or otherwise forget about stm

SSH Tunnel from Mac Command Line

1/ Open an ssh connection using the ssh command. Note that you should use the name of the system to which you want to connect for .

```
ssh  -f -N -C -L 3391:<computername>:3389 login.jlab.org"
```

```
Examples:

      ssh  -f -N -C -L 3391:opswin1:3389 login.jlab.org

      ssh  -f -N -C -L 3391:sftcuffe:3389 login.jlab.org
```

2/ Kill any left over ssh tunnels when you are done.

```
ps -ef | grep ssh | grep 3391

kill -9 PID
```

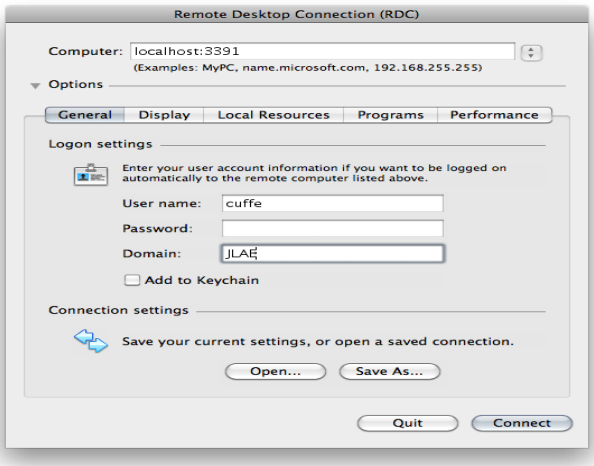
Start the Remote Desktop Client - Mac

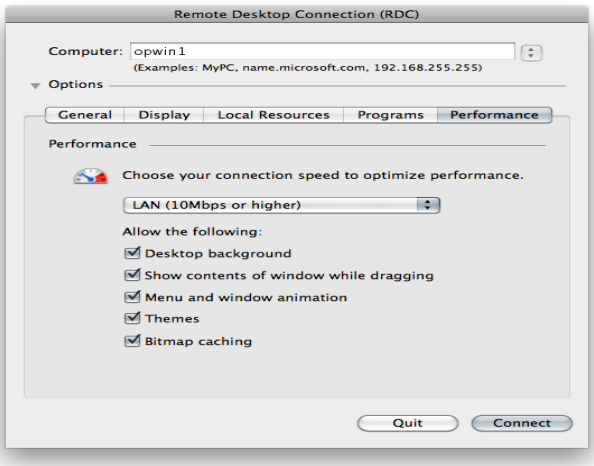
1/ Start the Desktop Client -Mac

```
Finder>>Applications>>Remote Desktop Connection>>Remote Desktop Connection
```

2/ Configure the Client

- On the ***General*** tab, fill in "Computer:" with localhost:3391.
- On the ***Performance*** tab, select the proper speed based on your connection.
- Set the display resolution to something reasonable or just leave it at "Full Screen."





Remote Desktop Client Installation

Windows - Client Installation

Windows XP, Vista and Windows 7 come with the Remote Desktop Client installed. You launch it from:

```
Start>>Programs>Accessories>>Remote Desktop Connection
```

The Remote Desktop Client is not supported on other versions of Windows.

Linux - Client Installation

The ***Remote Desktop Client*** on Linux is provided by an Open Source Project called rdesktop. It is usually installed by default in most popular linux distributions. If not, you should be able to use your package manager to install it. If no pre-built packages are available, you can compile it from source code. The source code can be found at http://www.rdesktop.org.

If you are using an Accelerator maintained Linux system. rdesktop is installed and already in you path.

Installing from Package Manager

```
On Redhat or CentOS 4.XX:

    sudo up2date rdesktop

On Redhat or CentOS 5.XX:

    sudo yum install rdesktop
```

```
On Ubuntu:

     sudo apt-get install rdesktop
```

Building from Source

The following gives a rough outline of the build process. You should tailor this to your own system.

```
cd /tmp

wget "http://prdownloads.sourceforge.net/rdesktop/rdesktop-1.6.0.tar.gz?download"

tar -xzvf rdesktop-1.6.0.tar.gz

cd rdesktop-1.6.0


./configure

make

make install


Alternately, if you put extra software somewhere else, you could do something like:


./configure  --prefix=/opt/rdesktop/1.6

make

make install

cd /opt/rdesktop/1.6

ln -s 1.6 pro

ln -s pro/bin .

ln -s pro/man .

ln -s /opt/rdesktop/bin/rdesktop /opt/bin
```

Mac - Client Installation

You can download the Mac client from Microsoft at Remote Desktop Connection Client 1.0.3 for Mac. The installation instructions can be found on the download page.