Xrfb Client-Server Setup(old)

Xrfb Client Setup

Configuring your client

To configure your Xrfb client you'll need the **address** and the **password** for your Xrfb server, both of which should be obtainable from your Systems Administrator. Once you have this information you're ready to complete the setup:

1. Select **Cobalt > General Preferences**.
2. In the text box next to **Xrfb Server Host Address** enter your Xrfb **server address**.
3. The **Xrfb Server Port Number** is set by default to 5899. In most cases, there should be no need to change this.
4. In the text box next to **Xrfb Server Host Address** enter your Xrfb **server password**.
5. Click **Enter**. Once you're finished you may close the **General Preferences** window.

Xrfb Server Setup

Server Requirements

- i386 32-bit architecture (Athlon or Pentium III or better)
- Recent Linux 2.6 kernel
- 1.5 GB free space
- Dedicated IP address, reachable from the clients
- Firewall opened to allow connections to TCP ports 5899 and 5910-5999 from the clients
- Root access required

To build your own Xrfb jail, the server machine needs to be

- Running Fedora Core 4 Linux
- 2.5 GB free space
- Must be root to build the jail

Step 1: Download the Installation Kit

The last Xrfb (aka Totoro) built by Henry Strickland and Ryan Seto under contract and supervision of the University of Wisconsin for use with the Dormouse version of Croquet.

- Read ME
- xrfb-documentation.txt
- ERATTA-1.txt
- xrfb-LICENSE.txt
- xrfb-all-changes.cs
- XrfbJail.tgz
- xrfb-build-README.txt
- Xrfb-build.tgz

Step 2: Create a jail.

The Xrfb server runs in a chroot jail. You may either use the Jail we provide (XrfbJail.tgz) or you may build your own (using xrfb-build.tgz).

If you use XrfbJail.tgz, as root, chdir to the directory where you want to place the jail, and untar it:

```
su root

cd /somewhere

tar xzf ..../...XrfbJail.tgz
```

In that example, the jail will be named "/somewhere/XrfbJail". (We'll continue to use /somewhere as the example location; change it to your actual location.)

To build your own jail, follow the instruction in the READ_ME file inside xrfb-build.tgz

Step 3: Set the password

Choose a password containing at least six (preferably many more) letters and numbers. Do not use spaces or other punctuation!

Put it in the file /somewhere/XrfbJail/home/guest/XrfbPasswd:

```
echo MyReallyHardToGuessS3cr3t > /somewhere/XrfbJail/home/guest/XrfbPasswd
```

This password will need to be shared with all clients using the server.

Step 4: Run the daemon

To run it in the foreground, enter:

```
sh /somewhere/XrfbJail/XrfbDaemon.sh
```

Diagnostics will be sent to the console. To kill it, use ^C.

To run it in background, add the "-d" flag:

```
sh /somewhere/XrfbJail/XrfbDaemon.sh -d
```

You may put that line in /etc/rc.local to have it restart automatically when the server reboots.

Step 5: Manage the daemon

Every Croquet client can manage the daemon; it is a shared resource, like much of Croquet.

1. From within squeak, file-in the xrfb-all-changes.cs file, if you haven't already done so.
2. From within croquet, start a "Dormouse" (also known as "Hedgemine") session. From the menu, choose **Tools >> Xrfb Desktops Control Panel**.
3. From the control panel, use **Configure...** to set the correct IP address, port number (should be 5899), and password (set in step 2).
4. Click **OK**.

These **management buttons** are availble in the control panel:

List Desktops

Every desktop running on the server will be listed, as will every client connected to it.

Examples:

listing 10 583928446372263745 http://google.com

A desktop server process (named Xvnc) is listening for connections on port 5910 (add 5900 to the desktop number 10 to get ths port). It is browsing http://google.com . The large arbitray number 583928446372263745 is a desktop id that is kept in the model object in Croquet. It tells us whether the client is viewing the same or different models.

connected 10 10.33.44.55:33459

A client on IP address 10.33.44.55 is connected from client port 33459 to the desktop server at port 5910.

Servers listening on desktop numbers with connections are in use.

Clean up

Servers may be left running with no connections because they might be used again.   This speeds up closing & re-opening a Croquet world. However, you may also shut down all servers with no connections using the "Clean up" button. This is not usually necessary, since unconnected servers are automatically cleaned up whenever a new server is allocated.

Kill All Desktops!

This is an emergency shutdown for all running desktops; you probably don't need to use it. Any clients connected will be aborted; they will see their desktop portals go light green.

The XrfbDaemon itself is NOT shut down. New management and desktop requests will be handled.

To repair client connections that are aborted (due to "Kill All Desktops!", server reboots, network disconnections, etc.), closing croquet (saving the world) and restarting the croquet world. Fresh desktop connections will be made to view the saved models.

KNOWN BUGS

Copy&paste clones of windows are prone to freezing up, apparently due to neither smalltalk nor the free unix Xvnc (vncserver) processes being multithreaded enough. I've seen the non-smalltalk linux vncviewer also lock up, apparently because the Xvnc server is in deadlock with a Croquet RFB process.
Workaround: Close croquet (saving the world), close squeak, restart squeak, and rejoin the world.

We can't find any way to find out when our XrfbMorph becomes garbage (e.g. when the entire Croquet is closed), so we can abort the client connection. This only wastes resources; it's not usually noticable. (When the portals are explicitly closed by clicking on the floating halo "X", we do intercept the message and abort the client.)
Workaround: ignore it, or same as above.

#back to top