Using the Centre for Molecular and Biomolecular Informatics over an SSH Tunnel[*]

Aonghus Lawlor, Gavin McCullagh[†]
University College Dublin

May 7, 2004

Contents

1 Introduction

The Centre for Molecular and Biomolecular Informatics applications provide a set of software applications which may be run from the central database but displayed on the user's screen wherever in the world that may be. This is useful for several reasons:

- The application (eg *Quest*) need not be installed on the user's machine

[*] This document is available for download at `http://fiachra.ucd.ie/~gavin/`

[†] `www_gmc@fiachra.ucd.ie`

- The CPU power, disk space, etc is provided by the central database machine, so the user need not have a powerful computer.

Sadly, it's not necessarily so simple. The user may (and probably should) be on a firewalled network. Firewalls by their nature prevent this sort of remote application display. They can be configured to allow the connections through but in practice getting your firewall administrator to sort this out may be a lot of trouble and present security problems.

This document presents an alternative means, using an SSH *tunnel* which may be achieved entirely by an ordinary user. This method has the added bonus of encrypting all data in transit and of optionally compressing the data to speed it over slow networks.

We've attempted to keep things as non-technical as possible. However a more technical explanation of this is included, see Section 6.

2 Executive Summary

For the impatient, here is a quick summary before the tedious technical details. This will also hopefully serve as a quick guide after configuration is complete.

- Start and minimize a local Xserver
- Open an SSH connection to `http://www.cmbi.kun.nl/` with X11 Tunnelling noting the `DISPLAY`, see Section 4.1.2.
- Login on the website as normal.
- Tell it the `DISPLAY` (eg `localhost:13.0`).
- Start your process (eg *Quest*).
- Return to the X Server, type `term x` and `menu full`, see Fig 5.

3 Installing an X Server

The installation of an X Server is beyond the scope of this document. However, this technique has been tested using the free *Cygwin X Server*[1] for MS Windows^{TM}and XFree86[2] for Linux^{TM}. It has been reported also to work with the proprietary eXceed X server and should work with any X Server. Apple Macintosh^{TM}users should note that a native X Server exists within OS X[3].

Full detailed instructions for installation of the Cygwin X server can be found at this address.

`http://xfree86.cygwin.com/docs/ug/setup-cygwin-x-installing.html`

4 Setting up the SSH Tunnel

The setup of an SSH Tunnel requires installing an SSH client. There are many, though we tested with the free PuTTY [4] ssh client in MS Windows and OpenSSH[5] in Linux. Again, any SSH client capable of tunnelling X connections should be adequate including the `ssh.com`[6].

4.1 PuTTY

4.1.1 Configuring PuTTY

PuTTY has no installer per sé, you need only copy the little program onto your pc. Run it and a configuration window should appear, see Fig 2. You need to enter the hostname `http://www.cmbi.kun.nl` and select SSH as your protocol. You may wish to save this session by naming it in the "Saved Sessions" dialog box and clicking save. This will save you entering all this information every time.[7]

You must now enable "X11 forwarding". To do this select "Tunnels" near the bottom along the left pane. In the new right pane, select "Enable X11 Forwarding", see Fig 2.

The final two steps are optional. You may for convenience wish to set your username on the database (it's the same as for the website) under the "Auth" settings. If you are finding the programs are very slow you could also enable the compression option under the "SSH" settings. Go back into "sessions" and save the session after all of these changes.

4.1.2 Starting PuTTY

If you have saved a session, you can now simply launch PuTTY and double click the session which we called `www.cmbi.kun.nl_TUNNEL` here, see Fig 1. You will be prompted for your CMBI username & password. Now you

[1] See `http://www.cygwin.com/xfree/`

[2] See `http://www.xfree86.org`

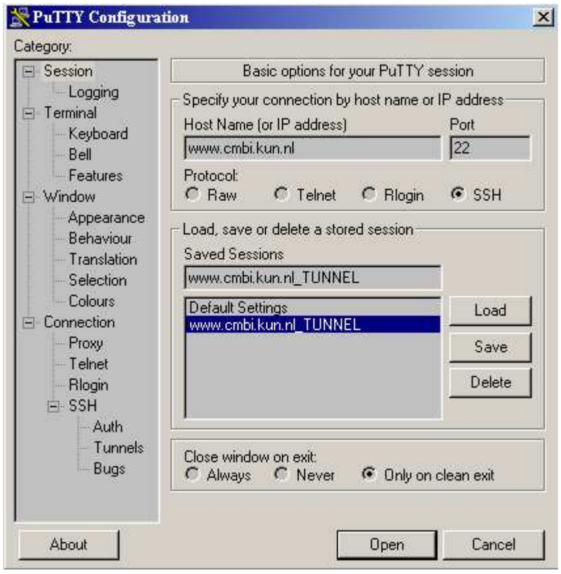
[3] See `http://www.apple.com/downloads/macosx/apple/x11formacosx.html`

[4] See `http://www.chiark.greenend.org.uk/~sgtatham/putty/`

[5] See `http://openssh.org/`

[6] See `http://ssh.com/`

[7] It is also possible to create a desktop link direct to this profile by adding to the shortcut target eg `putty.exe -load <SESSION_NAME>`



PuTTY Configuration

Basic options for your PuTTY session

Specify your connection by host name or IP address

Host Name (or IP address): www.cmbi.kun.nl

Port: 22

Protocol: ○ Raw ○ Telnet ○ Rlogin ● SSH

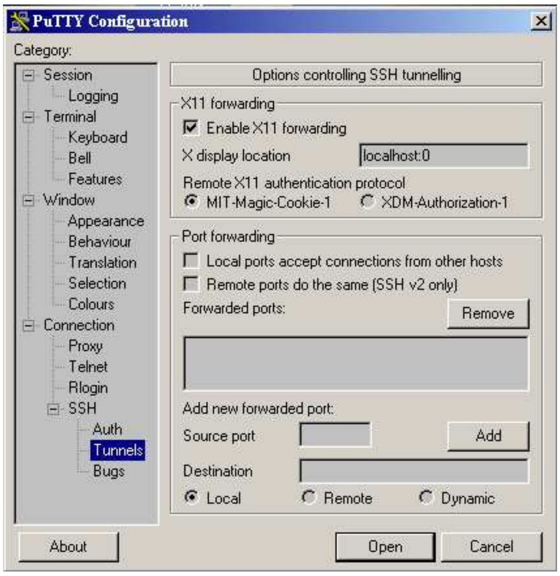
Load, save or delete a stored session

Saved Sessions: www.cmbi.kun.nl_TUNNEL

Default Settings
www.cmbi.kun.nl_TUNNEL

Close window on exit: ○ Always ○ Never ● Only on clean exit

Figure 1: Putty Configuration: hostname, protocol and sessions



PuTTY Configuration

Options controlling SSH tunnelling

X11 forwarding

☑ Enable X11 forwarding

X display location: localhost:0

Remote X11 authentication protocol: ● MIT-Magic-Cookie-1 ○ XDM-Authorization-1

Port forwarding

☐ Local ports accept connections from other hosts

☐ Remote ports do the same (SSH v2 only)

Forwarded ports:

Add new forwarded port:

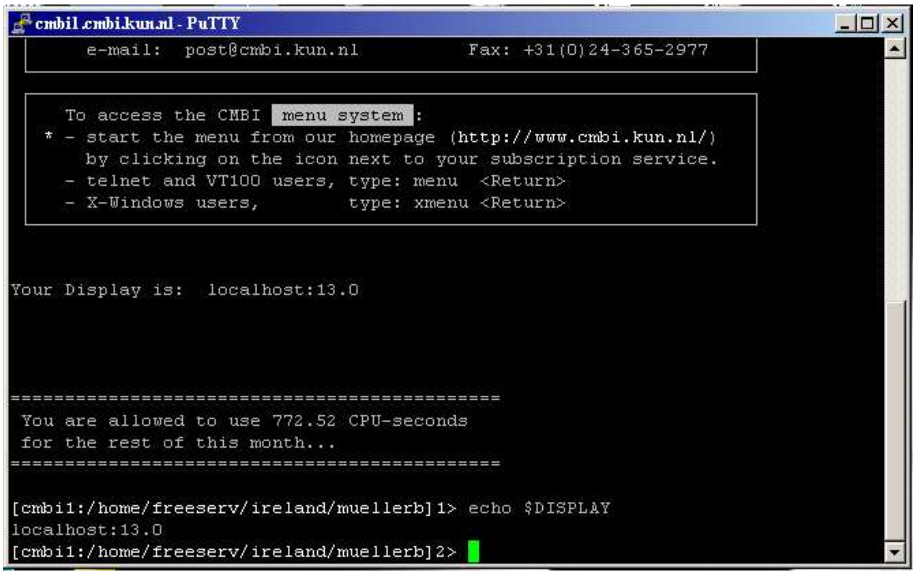
Source port:

Destination:

● Local ○ Remote ○ Dynamic

Figure 2: Putty Configuration: Enable X11 Forwarding

must type `echo $DISPLAY` (see Fig 3) to find out the tunnel display name (eg `localhost:13.0`). This information will be needed on the website.



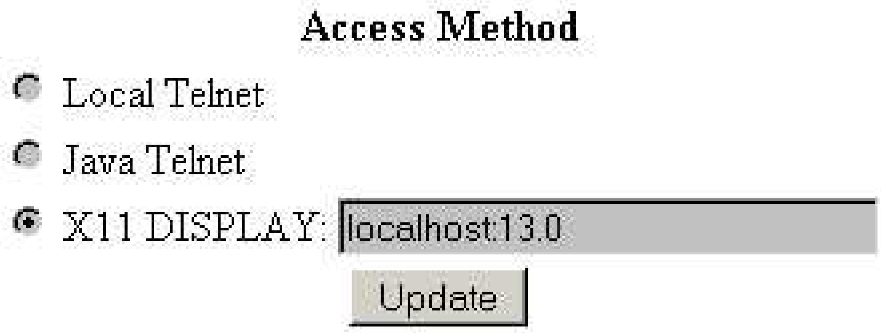
```
      e-mail:  post@cmbi.kun.nl          Fax: +31(0)24-365-2977

    To access the CMBI menu system :
  * - start the menu from our homepage (http://www.cmbi.kun.nl/)
      by clicking on the icon next to your subscription service.
    - telnet and VT100 users, type: menu  <Return>
    - X-Windows users,        type: xmenu <Return>


Your Display is:  localhost:13.0



=============================================
 You are allowed to use 772.52 CPU-seconds
 for the rest of this month...
=============================================

[cmbi1:/home/freeserv/ireland/muellerb]1> echo $DISPLAY
localhost:13.0
[cmbi1:/home/freeserv/ireland/muellerb]2>
```

Figure 3: The Running Tunnel: finding out your X Display address

5 Starting the application

So now you've an X Server and an SSH tunnel you're ready to get things going. In this order:

1. Start your X Server and minimize it's window.
2. Start the ssh tunnel (eg PuTTY) and note the `DISPLAY`.
3. Go to the website `http://www.cmbi.kun.nl/`, click "login menu". Login as usual.
4. The website will now ask for your "Access Method". Choose "X Display" and type the `DISPLAY` we noted above, see Fig 4.
5. Now execute the process you want through the website (eg *Quest*).
6. Bring up the minimized X Server window. It should now have a text program running in it.
7. As you normally would type `term x` and `menu full`[8] and your application should appear, see Figs 5,6.



Access Method

○ Local Telnet

○ Java Telnet

● X11 DISPLAY: localhost:13.0

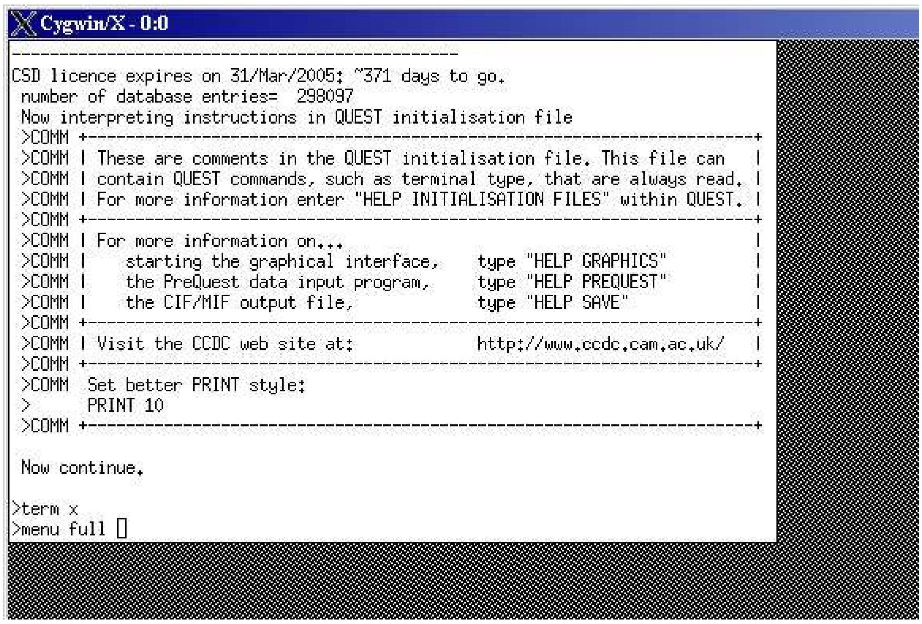
Figure 4: The Website: setting your Access Method after login

6 X Servers, Clients and Tunnels . . . Oh My!

6.1 X

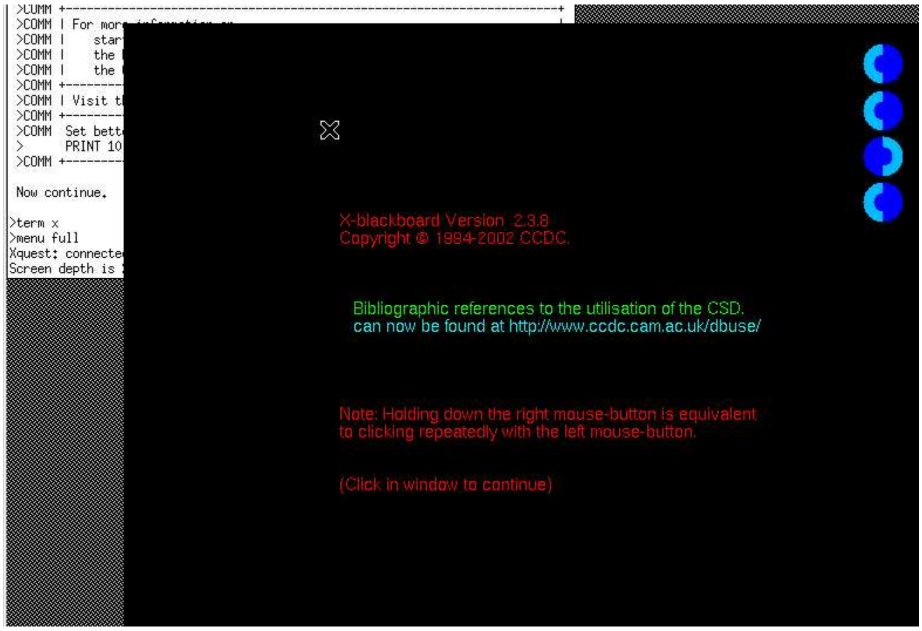
The technology used to provide the remote display feature is called simply "X" and has been the standard on UNIX systems for decades. The approach is simple enough. The display at which the user sits is called the *X Server* as it provides

[8] NB you may have to place the mouse over the text window



```
------------------------------------------------
CSD licence expires on 31/Mar/2005; ~371 days to go.
 number of database entries=  298097
 Now interpreting instructions in QUEST initialisation file
 >COMM +------------------------------------------------------------------+
 >COMM | These are comments in the QUEST initialisation file. This file can   |
 >COMM | contain QUEST commands, such as terminal type, that are always read. |
 >COMM | For more information enter "HELP INITIALISATION FILES" within QUEST. |
 >COMM +------------------------------------------------------------------+
 >COMM | For more information on...                                          |
 >COMM |    starting the graphical interface,    type "HELP GRAPHICS"        |
 >COMM |    the PreQuest data input program,     type "HELP PREQUEST"        |
 >COMM |    the CIF/MIF output file,             type "HELP SAVE"            |
 >COMM +------------------------------------------------------------------+
 >COMM | Visit the CCDC web site at:             http://www.ccdc.cam.ac.uk/   |
 >COMM +------------------------------------------------------------------+
 >COMM  Set better PRINT style:
 >      PRINT 10
 >COMM +------------------------------------------------------------------+

 Now continue.

>term x
>menu full
```

Figure 5: The X Server: the initial text interface



```
 >COMM +------------------------------------------------------------------+
 >COMM | For mor
 >COMM |    star
 >COMM |    the
 >COMM |    the
 >COMM +--------
 >COMM | Visit t
 >COMM +--------
 >COMM  Set bett
 >      PRINT 10
 >COMM +--------

 Now continue.

>term x
>menu full
Xquest: connecte
Screen depth is
```

X-blackboard Version 2.3.8
Copyright © 1994-2002 CCDC.

Bibliographic references to the utilisation of the CSD.
can now be found at http://www.ccdc.cam.ac.uk/dbuse/

Note: Holding down the right mouse-button is equivalent
to clicking repeatedly with the left mouse-button.

(Click in window to continue)

Figure 6: The X Server: the initial text interface

the physical display hardware. The central database which runs the applications (eg *Quest*) is the *X Client*.

The user starts an X server, logs into the website and tells it the address of his X Display eg `helge.ucd.ie:0.0`. They then initiate a process such as *Quest*. The central database starts *Quest* and tells it the address of the physical display to use (`helge.ucd.ie:0.0`). Now *Quest* connects [9] over the network to the user's computer and it's display appears on their X server.

6.2 SSH Tunnels

Without over-complicating things, the above process may be blocked by a firewall. As a result an alternative route for the information transfer must be provided. This process is sometimes called *tunnelling* and one means is using SSH , the *Secure SHell*.

The user logs into the remote machine (in our case the database) over SSH with the additional "Tunnel X Requests" option. This creates an encrypted channel within SSH which may be used for the above transaction. Once the tunnel is initiated you need only change the display address from directly accessing the X server (eg `helge.ucd.ie:0.0`) to use the SSH tunnel (usually eg `localhost:11.0` [10].

7 Corrections

Any comments, corrections, additions etc, please send to: `www_gmc@fiachra.ucd.ie`.

8 Acknowledgements

- Phil Bradley & Dr. Helge Mueller-Bunz for proof-reading.
- Dr. Helge Mueller-Bunz and Dr. Grace Morgan for their patience while we figured this stuff out.
- The XFree86 Project[11]
- The Cygwin Project[12]
- The OpenSSH Team[13]
- Simon Tatham for PuTTY[14]

[9] It is this connection which the firewall may block. An X connection is initiated by the Database to the user. Firewalls are generally configured to allow incoming *replies* only (eg requested webpages, pop3 email, etc). This connection appears to the firewall as an *initiation* not a reply and as such gets blocked

[10] How does this solve the problem? As we said, the firewall blocks connection initiations to the user's machine. The SSH channel is initiated by the user, so all data travelling along it is a reply. By tunneling all X display information over this pre-initiated SSH channel the firewall only has to allow replies.

[11] See `http://www.xfree86.org/`

[12] See `http://www.cygwin.com/http://www.cygwin.com/xfree/`

[13] See `http://www.openssh.org/`

[14] See `http://www.chiark.greenend.org.uk/~sgtatham/`