

SUN/129.5

Starlink Project
Starlink User Note 129.5

M. J. Bly, P. M. Allan,
B. K. McIlwrath
3 February 2000

XDISPLAY
Setting remote X windows
Version 2.1

Contents

1	Introduction	1
2	The problem	1
3	The preferred solution	1
4	The Xdisplay alternative	1
5	What can go wrong?	2
5.1	Authorisation	2
5.2	Multiple Hops	2
5.3	Inappropriate Use	3

1 Introduction

This note describes how to use the XDISPLAY utility on Starlink systems that are running X windows. It makes no attempt to describe the X windows system. It assumes that you have some familiarity with using X windows, although not to any great depth.

2 The problem

If you are using X windows purely on a workstation, then there is no problem. Graphical output created by X will appear automatically on your screen. The problem arises if you are logged on to an X client other than the machine that you are sat in front of (the X server).

Suppose that you are sat in front of a SPARCstation called `r1ssp0`, but that you are remotely logged into `r1saxps`, a Compaq Alpha. To get X output from a program run on `r1saxps` to be displayed on the screen of `r1ssp0`, you need to type

```
% setenv DISPLAY r1ssp0.bnsc.rl.ac.uk:0
```

This is easy to forget and too much to type every time you do a remote login.

3 The preferred solution

Starlink now recommends that users use the *Secure Shell* login command (`ssh` or its alias `slogin`) between machines which support it (all the Starlink Unix computers and even Microsoft Windows systems using additional software – see your Site Manager). This uses sophisticated means to authenticate that the remote machine is the correct one and encrypts all data traffic over the network.

An additional benefit is that `ssh` on Unix automatically creates an encrypted-channel X11 device on the remote computer, points the `DISPLAY` environment variable towards this and sets up local X11 authentication to allow remote access from this machine. The setting of `DISPLAY` must be left under the control of `ssh` to use this facility and it is important to check login files (such as `.login`) to ensure that these do not alter this environment variable.

4 The Xdisplay alternative

If `ssh` is **NOT** being used the `xdisplay` procedure is available on all Starlink machine to wrap up what you need to type to get X output from the machine that you are logged into back to your X server.

To get X output drawn on your X server from a remote client, just type `xdisplay` before running the program. If you type `xdisplay` when you are not logged into a remote system, then nothing will happen and the output will still appear on your screen.

What XDISPLAY does is to see where you logged in from and executes an appropriate command to point the X output back to you.

5 What can go wrong?

There are a few things that might stop XDISPLAY from working as you intended. These are problems with security, multi-hop logins and inappropriate use.

5.1 Authorisation

If you are logged into the Compaq Alpha `rlsaxps` from the SUN Sparc `rlssp0`, and you type `xdisplay`, then it is possible that you will get a message such as:

```
Xlib: Client is not authorized to access server
```

To fix this, you need to make sure that the client is authorized to send X output to the server. If you do not know how to do this, consult your site manager.

5.2 Multiple Hops

A problem that is not so easily overcome, but is probably less likely to occur, is that of multi-hop logins.

If you are using X-terminal `xacc` with an X session on Compaq Alpha `rlsaxps`, and you log into Compaq Alpha `rlsaxp2`, and then type `xdisplay` on `rlsaxp2`, XDISPLAY thinks you are on `rlsaxps` and will point the display back to `rlsaxps`. This will fail because `rlsaxps` is a server machine and does not have an X console.

A somewhat more embarrassing possibility is that you have logged in from one workstation (A) to a second (B), and thence to a third machine (C). If you then type `xdisplay` on C it will successfully point the X output back to B if the security setting allows it. The trouble is that you are sat in front of A wondering where the window has gone and the user sat in front of B is wondering why a random window has suddenly appeared, and worse, possibly trashed what was displayed on a pre-existing graphics window!

Making XDISPLAY automatically cope with multi-hop logins verges on the impossible, but there is a simple manual override available. If you type

```
xdisplay <nodename>
```

where `<nodename>` is the name of the computer or X-terminal that you want the display to appear on, then the display will appear on that X-server, regardless of the route that you used to log in.

5.3 Inappropriate Use

Note that you can only use the `xdisplay` command interactively. You cannot put it in a shell script as `xdisplay` is actually an alias that uses the shell's command line recall features.

If you need to use `XDISPLAY` in a file there are two possibilities. If you want to have `XDISPLAY` work out where you logged in from, put:

```
% source /star/etc/xdisplay
```

in your file. This will generally be your `.login` file. If you want to explicitly set the node that output will appear on, then simply set the environment variable `DISPLAY`, *e.g.*:

```
% setenv DISPLAY adam4.bnsc.rl.ac.uk:0
```

which is all that `XDISPLAY` does anyway.

If you give `XDISPLAY` the explicit node name, it is also possible to specify the transport mechanism to use. On Unix systems, you can select the transport to be used by the case of the node name. If it is purely in upper case, then DECnet will be used. Anything else will cause TCP/IP to be used, which is normally what is required.

Finally, if you try to use `XDISPLAY` in an inappropriate way then you probably will not get what you expected. For example, if you log into a computer from a VT type terminal connected to a TCP/IP terminal server, then `XDISPLAY` will quite happily point your X output to the terminal server. It is only when you come to run an X application that you will see the problem.