Linux XDMCP HOWTO

Thomas Chao

tomchao@lucent.com

Revision History

Revision v1.1 20 March 2001 Revised by: tc

This HOWTO describes how you can use the combination of X Display Manager (xdm, kdm and gdm) and XDMCP (X Display Manager Control Protocol) to provide the mechanism for an X−Terminal and a platform of cheap Remote X Apps solution. This document will be focusing on how to setup connection using XDMCP.

Table of Contents

1. Introduction....1
1.1. Disclaimer....1
1.2. Feedback....1

2. The Procedure....2
2.1. Before you begin, some backgrounds....2
2.2. Security Reminder....2
2.3. The System I use....2
2.4. Remote piece....3
2.5. Server Preparation....3
2.6. Steps to Complete the Procedures....3
2.7. Testing....5

3. Troubleshooting....6

4. XDMCP and GDM (Gnome Display Manager)....7

5. Additional References....8

6. Authors....9

7. Copyright Information....10

1. Introduction

XDMCP stands for "X Display Manager Control Protocol" and is a network protocol. It provides a mechanism for an X−Terminal. The X−Terminal will only run the X−Server and it will allow applications running on remote machine to be displayed on it.

Some of us running Linux (like me) are looking for the best parts of Linux. Among them is the ability to re−use old systems (like 486 CPUs) as a X−Terminal (with the Win32 apps; like Hummingbird's Exceed or X−Win32) to run Linux from any PC remotely. It is somehow very surprising that there aren't many documents on the internet which guide you step by step on how to set this up. This is how I come up with this document as a way to share my experience with all user. Essentially, by using X and XDMCP, you can create a cheap solution of a X− environment.

1.1. Disclaimer

No liability for the contents of this documents can be accepted. Use the concepts, examples and other content at your own risk. As this is a new edition of this document, there may be errors and inaccuracies, that may of course be damaging to your system. Proceed with caution, and although this is highly unlikely, the author(s) do not take any responsibility for that.

All copyrights are held by their by their respective owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark.

Naming of particular products or brands should not be seen as endorsements.

You are strongly recommended to take a backup of your system before major installation and backups at regular intervals.

1.2. Feedback

Feedback is most certainly welcome for this document. Without your submissions and input, this document wouldn't exist. Please send your additions, comments and criticisms to the following email address : <tomchao@lucent.com>.

---

2. The Procedure

This section details the procedures for setting up X−Terminal using XDMCP.

2.1. Before you begin, some backgrounds

Before you begin, it is better to have a basic understanding of how this works. (More details are at the Resources below and LDP HOWTO page)

The X server is usually started from the X Display Manager program (xdm, kdm and gdm. This document will use gdm as an example). It provides a nice and consistent interfaces for general users (X−based login, starting up a window manager, clock, etc.). X Display Manager manages a collection of X displays, which may be on the local host or remote servers.

When xdm runs, it is usually run as a local copy of X, also xdm can listen for requests from remote hosts over a network. For kdm (which comes with the KDE desktop), it is a replacement of xdm and configures the same way, except its files are in `/etc/X11/kdm`. The gdm (Gnome Display Manager) is a reimplementation of the xdm program. gdm has similar funtions to xdm and kdm, but was written from scratch and does not contain any original XDM / X Consortium code.

In the case of xdm, it offers display management in two different ways. It can manage X servers running on the local machine and specified in Xservers, and it can manage remote X servers (typically X terminals) using XDMCP (the XDM Control Protocol) as specified in the Xaccess file. (Courtesy of xdm man page).

- The XDM and Xterminal mini−HOWTO
- Linux Remote X Apps mini HOWTO A very good reference for Remote X in both theoretical and practical view.
- The Xterminal mini−HOWTO

2.2. Security Reminder

Using XDMCP is inherently insecure, therefore, most of the distributions shipped as its' XDMCP default turned off. If you must use XDMCP, be sure to use it only on a trusted networks, such as corporate network within a firewall. Unfortunely, XDMCP uses UDP, not TCP, therefore, it is not possible to forward XDMCP over SSH. Some people has success in X11 TCP/IP port Forwarding. Check this UC Berkeley Howto site for more info.

2.3. The System I use

I have tested the setup running a GNOME (gdm), as well as KDE (kdm) on Red Hat 6.0, 6.2 and Red Hat 7.0. The other I have tried on are on Caldera eDesktop 2.4, which is similar to RH's setup. I have not had a chance to test it on other Linux flavors (but plan to do so for Debian and Slackware in the future). If you have successfully setup one other than the Red Hat platform, please share it with me. I will add them into this document.

My server hardware is an IBM PC clone running an Intel Pentium II 400 MHz with 128 MB memory and 30 MB ATA−66 Hard Drive. (I found out that 486 PC and my other Pentium 100 MHz PC runs this just fine). I use a 3COM 10/100 Fast Ethernet (3C509B) NIC.

2.4. Remote piece

I use the Hummingbird Exceed 6.x (with Service Pack) and have tested them on Windows 98 SE, Windows NT 4.0 and Windows 2000 Pro. I found out that another popular choice are X−Win32 and VNC. However, there are many open−source apps as well as commercial one available.

2.5. Server Preparation

To prepare your X Server for XDMCP session, you need to make sure the following are properly installed:

1. Install your Linux OS. In my case, I installed Red Hat 6.2 (Custom Installation). I also tried on RH 7.0.
2. Setup your Networking. To test it out, **ping** and **telnet** are good comamnds to use to determine if your network works.
3. Setup X. Do *not* setup with a resolution higher than what the remote users are able to use for their display. Test the X Server by typing either **startx** or **telinit 5**. Make sure X is running properly.
4. Creates the necessary user accounts (and associated groups) for user who will access via the X−Terminal.

2.6. Steps to Complete the Procedures

These are steps I used to setup the Xserver for accepting XDMCP requests:

1. Modify `/etc/rc.d/init.d/xfs` and make the following changes. Change all (this is where the Font Server port):

   ```
   daemon xfs -droppriv -daemon -port -1
   ```

   to:

   ```
   daemon xfs -droppriv -daemon -port 7100
   ```

   In RH 7.0, you do not need to do this, since by default, it is, for security enhancement, not listening to TCP port any longer! If you need to setup default font server to use, do it in /etc/X11/fs/config and add the setting there.

2. In `/etc/X11/xdm/Xaccess`, change (this allow all hosts to connect):

   ```
   #*     # any host can get a login window
   ```

   to:

   ```
   *      # any host can get a login window
   ```

xdm usually run as a local copy of X and can listen for requests from remote hosts over a network. xdm reads its configuration files `/etc/X11/xdm/xdm-config` for all configuration and log files that xdm uses. For kdm, it is a replacement of xdm and configures the same way, except its files are in `/etc/X11/kdm`. It is worth noting that the **Xsession** file is what runs your environment.

The gdm (Gnome Display Manager) is a reimplementation of the well known xdm. gdm has similar funtions to xdm and kdm, gdm is the Gnome Display Manager, and its configuration files are found in `/etc/X11/gdm/gdm.conf`. The `gdm.conf` file contains sets of variables and many options for gdm, and the Sessions directory contains a script for each session option; each script calls `/etc/X11/xdm/Xsession` with the appropriate option.

3. I use the gdm as default and use gdm login window to switch between KDE and GNOME. Edit `/etc/X11/gdm/gdm.conf`. This activates XDMCP, causing it to listen to the request. Change this:

```
[xdmcp]
Enable=0
```

to:

```
Enable=1
```

Make sure "**Port=177**" is at the end of this block.

4. Now edit `/etc/inittab` and change the following line:

```
id:3:initdefault:
```

to:

```
id:5:initdefault:
```

Before changing this line, you can use the **telinit** command to test prior to modifying the line. Use either **telinit 3** to set to level 3, or **telinit 5** to set to level 5, graphics mode (you can issue this command on the second machine that telnets into this server).

5. Make sure the proper security of the file `/etc/X11/xdm/XServers` set to 444 (chmod 444).
6. Locate `/etc/X11/xdm/Xsetup_0` and **chmod 755** this file.
7. Edit the `XF86Config` file in `/etc/X11` and change the line, if you are using RH 6.2:

```
FontPath    "unix:-1"
```

to:

```
FontPath    "unix:7100"
```

8. (You do not have to make this change. You can keep the default setting, but this is what I use. If you are not sure, leave this alone.) Add this line to the end of `/etc/inittab`:

```
x:5:respawn:/usr/bin/gdm
```

You are now ready to run a test.

2.7. Testing

To test if your XDMCP with X Server is ready to accept connections, do these steps. I find it easier using the X Server and another machine to test:

1. (Though you don't need to; it doesn't hurt...) Reboot the machine (I am assuming you are running level 5).
2. Make sure the Graphical login page comes up. Make sure the display resolution and mouse work. Log in from the console to see if the local access is OK. If OK, do not log off.
3. Setup Hummingbird Exceed to either query this machine (using the IP address or fully qualified DNS name) or set to use XDMCP−Broadcast and try to connect to the X server. You should see the X Session come up and the login screen appear.

---

3. Troubleshooting

- If X cannot come up and is broken:

  If X is broken and the connection fails, most of the time it has this error messages:

```
_ FontTransSocketUNIXConnect: Can't connect: errno = 111
failed to set dafault font path 'unix:-1'
Fatal server error:
could not open default font 'fixed'
```

  This is likely due to xfs not finding the correct port for the Font Server (again, if you are running RH 6.2). To resolve this, check steps 1 and 7 above. Make sure the configuration are pointing to (port) 7100 and make sure you have the following fonts installed (if not re−install the XFree86 font packages):

```
FontPath  "/usr/lib/X11/fonts/75dpi/"
FontPath  "/usr/lib/X11/fonts/misc/"
FontPath  "/usr/lib/X11/fonts/CID"
FontPath  "/usr/lib/X11/fonts/Speedo"
FontPath  "/usr/lib/X11/fonts/100dpi"
```

  Use the command **startx** (on local) to restart the X server (or use **telinit 5** to switch the run−level).

- If Exceed has no respond:

  In this case, most likely your xdm (or gdm, depending upon which is used in `/etc/inittab`) is not starting correctly. Issue the command: **ps −ef | grep gdm** (or **ps −ef | grep xdm** if xdm is used).

  If the process is not running, check the steps on the setup above (make sure there are no typo's and that the correct path is given). Restart X using the command **telinit 5**.

---

4. XDMCP and GDM (Gnome Display Manager)

The following is taken from the Gnome Display Manager Reference Manual:

GDM also supports the X Display Manager Protocol (XDMCP) for managing remote displays. GDM listens to UDP port 177 and will repond to QUERY and BROADCAST_QUERY requests by sending a WILLING packet to the originator. GDM can also be configured to honor INDIRECT queries and present a host chooser to the remote display. GDM will remember the user's choice and forward subsequent requests to the chosen manager. GDM only supports the MIT−MAGIC−COOKIE−1 authentication system. Little is gained from the other schemes, and no effort has been made to implement them so far. Since it is fairly easy to do denial of service attacks on the XDMCP service, GDM incorporates a few features to guard against attacks. Please read the XDMCP reference section below for more information.

Even though GDM tries to outsmart potential attackers, it is still adviced that you block UDP port 177 on your firewall unless you really need it. GDM guards against DoS attacks, but the X protocol is still inherently insecure and should only be used in controlled environments. Even though your display is protected by cookies the XEvents and thus the keystrokes typed when entering passwords will still go over the wire in clear text. It is trivial to capture these. You should also be aware that cookies, if placed on an NFS mounted directory, are prone to eavesdropping too.

---

5. Additional References

Some additional references on this subject include:

- Your local xdm man page.
- Your local gdm man page.
- Configuring XDM
- xdmcp/udp
- XDMCP Documentation
- Should you be running XDMCP?
- X Window System Terminals
- A second way of using XDM
- Accessing Xterms from Windows
- How to download and install X−Win32
- How to install X−Win32 (another ref.)
- Taming the X Display Manager
- GNOME Display Manager

---

6. Authors

Current: Thomas Chao, Lucent Technologies

7. Copyright Information

This document is copyrighted (c) 2000, 2001 Thomas Chao and is distributed under the terms of the Linux Documentation Project (LDP) license, stated below.

Unless otherwise stated, Linux HOWTO documents are copyrighted by their respective authors. Linux HOWTO documents may be reproduced and distributed in whole or in part, in any medium physical or electronic, as long as this copyright notice is retained on all copies. Commercial redistribution is allowed and encouraged; however, the author would like to be notified of any such distributions.

All translations, derivative works, or aggregate works incorporating any Linux HOWTO documents must be covered under this copyright notice. That is, you may not produce a derivative work from a HOWTO and impose additional restrictions on its distribution. Exceptions to these rules may be granted under certain conditions; please contact the Linux HOWTO coordinator at the address given below.

In short, we wish to promote dissemination of this information through as many channels as possible. However, we do wish to retain copyright on the HOWTO documents, and would like to be notified of any plans to redistribute the HOWTOs.

If you have any questions, please contact <linux-howto@metalab.unc.edu>