NAME

Xserver – X Window System server

SYNOPSIS

X [:displaynumber] [–option ...] [ttyname]

DESCRIPTION

X is the generic name for the X Window System server. It is frequently a link or a copy of the appropriate server binary for driving the most frequently used server on a given machine.

STARTING THE SERVER

The server is usually started from the X Display Manager program **xdm**(1). This utility is run from the system boot files and takes care of keeping the server running, prompting for usernames and passwords, and starting up the user sessions. It is easily configured for sites that wish to provide nice, consistent interfaces for novice users (loading convenient sets of resources, starting up a window manager, clock, and nice selection of terminal emulator windows).

Installations that run more than one window system will still need to use the **xinit**(1) utility. However, **xinit** is to be considered a tool for building startup scripts and is not intended for use by end users. Site administrators are **strongly** urged to use **xdm**, or build other interfaces for novice users.

When the X server starts up, it takes over the display. If you are running on a workstation whose console is the display, you cannot log into the console while the server is running.

NETWORK CONNECTIONS

The X server supports connections made using the following reliable byte-streams:

TCP/IP
: Listen on port 6000+*n*, where *n* is the display number.

Unix Domain
: Use **/tmp/.X11-unix/X***n* as the filename for the socket, where *n* is the display number.

DECnet
: Respond to connections to object **X$X***n*, where *n* is the display number. This is not supported in all environments.

OPTIONS

All X servers accept the following command line options:

–**a** *number*
: Set pointer acceleration (i.e. the ratio of how much is reported to how much the user actually moved the pointer). The default is 2.

–**ac**
: Disable host-based access control mechanisms. Enables access by any host, and permits any host to modify the access control list. Use with caution. This option exists primarily for executing remote test suites.

–**audit** *level*
: Set the audit trail level. The default is 1; only connection rejections are reported. Level 2 also reports successful connections and disconnects. Level 0 turns off the audit trail. Audit output is sent on the standard error output stream.

–**auth** *authorization-file*
Specify a file which contains a collection of authorization records used to authenticate access. See also **xdm**(1)

bc Disable certain kinds of error checking for bug compatibility with previous releases (e.g., to work around bugs in R2 and R3 xterms and toolkits). Deprecated.

–**bs** Disable backing store support on all screens.

–**c** Turn off key-click.

c *volume* Set key-click volume, range is 0-100.

–**cc** *class*
Set the visual class for the root window of color screens. Class numbers are as specified in the X protocol. Not implemented in all servers.

–**co** *filename*
Set name of RGB color database.

–**core** The server dumps core on fatal errors.

–**dpi** *resolution*
Set the resolution of the screen, in dots per inch. Use when the server cannot determine the screen size from the hardware. The default is 90.

–**f** *volume*
Set beep (bell) volume, range is 0-100. The default is 50.

–**fc** *cursorFont*
Set default cursor font.

–**fn** *font* Set default font.

–**fp** *fontPath*
Set the font search path. This path is a comma separated list of directories which the X server uses to search for font databases.

–**ep** *encodingPath*
Set the font encoding search path.

–**help** Print a usage message.

–**I** All following command line arguments are ignored.

–**logo** Turn on the X Window System logo display in the screen-saver. There is currently no way to change this from a client.

nologo Turn off the X Window System logo display in the screen-saver. There is currently no way to change this from a client.

–**p** *minutes*
Set screen-saver pattern cycle time in minutes. The default is 10 minutes.

–**pn** The server is to continue running if it fails to establish all of its well-known sockets, but establishes at least one.

–**r** Turn off auto-repeat.

r Turn on auto-repeat.

–s *minutes*
Set screen-saver timeout time in minutes. The default is 10 minutes.

–su Disable save under support on all screens.

–t *number*
Set pointer acceleration threshold in pixels (i.e. after how many pixels pointer acceleration should take effect). The default is 4.

–terminate
The server is to terminate instead of resetting.

–to *seconds*
Set default connection timeout in seconds. The default is 60 seconds.

–tst Disable all testing extensions (e.g., XTEST, XTrap, XTestExtension1).

tty*xx* ignored, for servers started the ancient way (from init).

v Set video-off screen-saver preference.

–v Set video-on screen-saver preference.

–wm Force the default backing-store of all windows to be WhenMapped; an inexpensive way of getting backing-store to apply to all windows.

–x *extension*
Load the specified extension at init. Not supported in most implementations.

You can also have the X server connect to **xdm**(1) using XDMCP. Although this is not typically useful as it does not allow **xdm** to manage the server process, it can be used to debug XDMCP implementations, and serves as a sample implementation of the server side of XDMCP. For more information on this protocol, see the *X Display Manager Control Protocol* specification. The following options control the behavior of XDMCP.

–query *host-name*
Enable XDMCP and send Query packets to the specified host.

–broadcast
Enable XDMCP and broadcast BroadcastQuery packets to the network. The first responding display manager will be chosen for the session.

–indirect *host-name*
Enable XDMCP and send IndirectQuery packets to the specified host.

–port *port-num*
Use an alternate port number for XDMCP packets. Must be specified before any –query, –broadcast or –indirect options.

–once Terminate the server after one session.

–class *display-class*
XDMCP has an additional display qualifier used in resource lookup for display-specific options. This option sets that value, by default it is "MIT-Unspecified" (not a very useful value).

–displayID *display-id*

Yet another XDMCP specific value, this one allows the display manager to identify each display so that it can locate the shared key.

Many servers also have device-specific command line options. See the manual pages for the individual servers for more details.

SECURITY

The X server implements a simple authorization protocol, MIT-MAGIC-COOKIE-1 which uses data private to authorized clients and the server. This is a rather trivial scheme; if the client passes authorization data which is the same as that held by the server, it is allowed connection access. This scheme is worse than the host-based access control mechanisms in environments with unsecure networks as it allows any host to connect, given that it has the private key. But in many environments, this level of security is better than the host-based scheme as it provides access control on a per-user instead of per-host basis.

In addition, the server provides support for a DES-based authorization scheme, SUN-DES-1, using Sun's Secure RPC. It involves encrypting data with the X server's public key. See the *Solaris X Window System Developers Guide* for more information.

The authorization data is passed to the server in a private file named with the −**auth** command line option. Before accepting connections after a reset or when the server is starting, it reads this file. If this file contains authorization records, the local host is not allowed access to the server; only clients which send one of the authorization records contained in the file in the connection setup information are allowed access. See the **xauth**(1) manual page for a description of the binary format of this file.

The X server also uses a host-based access control list for deciding whether or not to accept connections from clients on a particular machine. If no other authorization mechanism is being used, this list initially consists of the host on which the server is running as well as any machines listed in the file **/etc/X*n*.hosts**, where *n* is the display number of the server. The file contains either an Internet hostname (e.g. expo.lcs.mit.edu) or a DECnet hostname in double colon format (e.g. hydra::). Each hostname must be newline separated with no leading or trailing whitespace. For example:

```
joesworkstation
corporate.company.com
star::
bigcpu::
```

Users add or remove hosts from this list and enable or disable access control using the **xhost** command from the same machine as the server.

The X protocol intrinsically does not have any notion of window operation permissions or place any restrictions on what a client can do; if a program can connect to a display, it has full run of the screen. Sites that have better authentication and authorization systems (such as Kerberos) might wish to make use of the hooks in the libraries and the server to provide additional security models.

SIGNALS

The X server handles the following signals:

SIGHUP
: Close all existing connections, free all resources, and restore server defaults. This signal is sent by the display manager whenever the user's primary application (usually an **xterm**(1) or window manager) exits to force the server to clean up and prepare for the next user.

SIGTERM
: The server exits cleanly.

SIGUSR1
: This signal is used quite differently from either of the above. When the server starts, determines if SIGUSR1 is set to SIG_IGN. If this is true the server sends a SIGUSR1 to its parent process after it has set up the various connection schemes. **Xdm** uses this feature to recognize when the server is ready for client connections (see **xdm**(1)).

FONTS

Fonts are normally stored as individual files across various directories. The X server can obtain fonts from directories and/or from font servers. The list of directories and font servers the X server uses when trying to open a font is controlled by the *font path*. Although most sites will choose to have the X server start up with the appropriate font path (see the −**fp**), This path can be overridden using the **xset**(1) program.

The default font path for the X server includes eight directories:

/usr/openwin/lib/X11/fonts/misc
: This directory contains many miscellaneous bitmap fonts that are useful on all systems. It contains a family of fixed-width fonts, a family of fixed-width fonts from Dale Schumacher, several Kana fonts from Sony Corporation, two JIS Kanji fonts, two Hangul fonts from Daewoo Electronics, two Hebrew fonts from Joseph Friedman, the standard cursor font, two cursor fonts from Digital Equipment Corporation, and cursor and glyph fonts from Sun Microsystems. It also has various font name aliases for the fonts, including **fixed** and **variable**.

/usr/openwin/lib/X11/fonts/Speedo
: This directory contains outline fonts for Bitstream's Speedo rasterizer. A single font face, contributed by Bitstream, Inc., in normal, bold, italic, and bold italic, is provided.

/usr/openwin/lib/X11/fonts/75dpi
: This directory contains bitmap fonts contributed by Adobe Systems, Inc., Digital Equipment Corporation, Bitstream, Inc., Bigelow and Holmes, and Sun Microsystems, Inc. for 75 dots per inch displays. An integrated selection of sizes, styles, and weights are provided for each family.

/usr/openwin/lib/X11/fonts/100dpi
: This directory contains 100 dots per inch versions of some of the fonts in the *75dpi* directory.

/usr/openwin/lib/X11/fonts/F3
: This directory contains scalable outlie fonts for the F3 format. 57 typefaces are

present.

/usr/openwin/lib/X11/fonts/F3bitmaps
This directory contains bitmaps in various point sized for the 57 F3 fonts.

/usr/openwin/lib/X11/fonts/Xt+
This directory contains fonts used by OLIT.

/usr/openwin/lib/X11/fonts/Type1
This directory contains outline Adobe Type1 fonts.

Font databases are created by executing the **mkfontdir**(1) program in the directory containing the compiled versions of the fonts (the **.pcf** files). Whenever fonts are added to a directory, **mkfontdir** should be executed so that the server can find the new fonts. If **mkfontdir** is not run, the server will not find any fonts in the directory.

DIAGNOSTICS

Too numerous to list them all. If run from **init**(1M), errors are typically logged in the file **/usr/adm/X*msgs**,

FILES

/etc/X*.hosts	Initial access control list
/usr/openwin/lib/X11/fonts/misc	Bitmap font directory
/usr/openwin/lib/X11/fonts/75dpi	Bitmap font directory
/usr/openwin/lib/X11/fonts/100dpi	Bitmap font directory
/usr/openwin/lib/X11/fonts/Speedo	Outline font directory
/usr/openwin/lib/X11/fonts/F3	F3 outline font directory
/usr/openwin/lib/X11/fonts/F3bitmaps	Bitmap font directory
/usr/openwin/lib/X11/fonts/Xt+	OLIT font directory
/usr/openwin/lib/X11/fonts/Type1	Outline font directories
/usr/openwin/lib/X11/fonts/PEX	PEX font directories
/usr/openwin/lib/X11/rgb.txt	Color database
/tmp/.X11-unix/X*	Unix domain socket
/usr/adm/X*msgs	Error log file

SEE ALSO

X11(7), **bdftopcf**(1), **mkfontdir**(1), **xauth**(1), **xdm**(1), **xhost**(1), **xinit**(1), **xset**(1), **xsetroot**(1), **xterm**(1), **Xsun**(1) *X Window System Protocol, Definition of the Porting Layer for the X v11 Sample Server, Strategies for Porting the X v11 Sample Server, Godzilla's Guide to*

Porting the X V11 Sample Server

BUGS

The option syntax is inconsistent with itself and **xset**(1).

The acceleration option should take a numerator and a denominator like the protocol.

If **X** dies before its clients, new clients won't be able to connect until all existing connections have their TCP TIME_WAIT timers expire.

The color database is missing a large number of colors.

COPYRIGHT

AUTHORS

The sample server was originally written by Susan Angebranndt, Raymond Drewry, Philip Karlton, and Todd Newman, from Digital Equipment Corporation, with support from a large cast. It has since been extensively rewritten by Keith Packard and Bob Scheifler, from MIT.