Xdm:

The X Session Manager

Win Treese

Keith Packard

Chris Peterson

DRAFT 2 -- 21 September 1988 at 17:04

1. Why Xlogin?

Xlogin is a ''session manager'' for interactive user sessions using the X window system. Since using a windowing environment is not a simple extension of using a terminal, a more general notion of a ''user session'' is required. While users of UNIX workstations typically use one or more terminal emulators as well as other window-based applications, this is not the only paradigm for use. When a user logs into the workstation, an environment appropriate for her use should be created -- one which may or may not include terminal emulators. As a side effect, the additional overhead of supporting logins in *xterm* can be eliminated.

This document describes the general form of *Xlogin*. Appendix I contains descriptions of the configuration file formats for *Xlogin*. Appendix II contains sample configuration files for *Xlogin*. Appendix III describes some changes necessary for use at Project Athena, as well as a proposed configuration for use on Athena workstations.

2. Design Goals

The following overall goals were used in designing *Xlogin*:

1. *Xlogin* should support a simple set of defaults, yet also provide flexible configuration for users and system managers.
2. It should support virtually any set of X applications as a user configuration.
3. It should support a variety of X environments, including timesharing systems, workstations, and X-based terminals.
4. It should be able to manage aspects of a session surrounding the actual user interaction (*e.g.*, initializing and resetting a workstation).

3. What Xlogin Must Do

As a replacement for the standard *login* program in the X environment, *Xlogin* must minimally provide the following functions currently performed by *login*:

- Verify that logins have not been temporarily disallowed by the system manager (*i.e*, by creating */etc/nologin*.
- Authenticate the user who wishes to login. This includes verifying that the user is permitted to login on a particular machine.
- Set up the environment variables used by later processes.

In addition, *login* provides the following services, which will not be provided by *Xlogin*:

- Check if the user is over her disk quota and notify her if she is.
- Display the system message-of-the-day. This can be disabled by the existence of a file *.hushlogin* in the user's home directory.
- Checks for the existence of new mail for the user.

These functions can be performed outside the *Xlogin* program itself, using the startup procedures described below.

The additional flexibility of *Xlogin* makes it possible to perform site-specific changes to the login process.

4. Xlogin Configuration

The display servers managed by *Xlogin* are specified in the configuration file */etc/Xservers*. In addition, an X display can request that it be managed, as described below. Other configuration options are specified in */etc/Xdm-config*, which has the same format as an X resources file, though it is read directly. In general, the actual names of files described below are specified in *Xdm-config*.

Remote X displays may request that they be managed only for the duration of a single session; they are known as "transient" servers.

5. Display Management

Xlogin is a daemon started at boot time, typically near the end of */etc/rc*. For each entry in *Xservers*, *Xlogin* starts a server control program. In addition, a new server control program is started whenever a request is received on a well-known port. These requests come from X servers that wish to be managed (*e.g.*, an X terminal). The request consists of a single UDP packet. The format of the packet is specified below. An X server that no longer wishes to be managed may request that *Xlogin* cease doing so.

The server control programs run X servers for local displays. For remote displays, the server control program traps signals and sends termination notices to the remote server using another well-known port.

6. Session Management

Each server control program manages a "session" through the following cycle:

1. System resources are loaded from into the server from *Xresources* using *xrdb*.
2. A window is created to prompt for a username and password. In a later version, buttons or menus may be added for other other actions, such as invoking the system defaults rather than the user's personal configuration, or for providing "help." The display is controlled by xlogin resources (see Appendix I). Internally, the display is handled by the *Greet()* function.
3. After the user has supplied the username and password, *Xlogin* calls its internal function *Verify()*. This function can be customized with local authentication procedures if desired. *Verify()* also initializes several environment variables. They are:
   - HOME
   - SHELL

- PATH
- USER
- DISPLAY

4. The program *Xstartup* is executed if it exists. This allows optional system configuration to take place. The execution environment is not that of the user at this point, but it does include the USER and DISPLAY variables. If this program exits with a non-zero status, the user is notified and the session is reset as described below.
5. *Xlogin* then creates an execution environment with the user's variables.
6. A new process is created, and the real and effective UIDs of the process are changed to that of the user. The *Xsession* program is executed next. Typically, *Xsession* will execute the user's *~/.Xrc* file if it exists, providing some reasonable default action if it does not. Otherwise, the system default one (*Xsession*) is executed. In the future, it may be possible for a user to specify that the system default should be executed rather than the personal *.Xrc*. This ability may be useful if, for example, the user's *.Xrc* is not executing properly. If there is a non-executable *.Xrc* file, the system-wide default is executed in its place.
7. When the *.Xrc* program terminates (typically by having the window manager exit), the server control program resets (or optionally restarts) the X server.
8. The termination script *Xreset* is executed. The execution environment includes the USER and DISPLAY environment variables.
9. A new session is created for non-transient servers.

7. Information and Error Messages

During the course of a login, there may be messages indicating errors or status information that must be delivered to the user. These are displayed by creating a new window containing the message. The user indicates that she has read the message by clicking on the "OK" button on the window.

Messages which may be displayed include:

- Login failure.
- System message of the day.
- Quota warnings

This facility will also support messages that require a choice by the user. For example, the user may wish to terminate the login if certain abnormal conditions occur (*e.g.*, if her home directory is unavailable).

8. Acknowledgements

The authors would like to thank Dave Bundy of XPI, Bob Schleifler and Jim Fulton of the MIT X Consortium, and Jerry Saltzer, Dan Geer, Ralph Swick, and Jennifer Steiner of Project Athena for comments leading to this design. It's not their fault, however.

I. Configuration File Formats

9. Format of /etc/Xdm-config

This file is the only one with a hard-coded pathname. It has the format of a standard X resources file as used by the X Toolkit. Resources that may be specified include:

servers	List of servers to be managed (denoted *Xservers*).
errorLogFile	File for logging errors.
startup	Startup program (denoted *Xstartup*).
reset	Reset program (denoted *Xreset*).
session	System default session initialization (denoted *Xsession*).

10. Format of Xservers

/etc/Xservers contains a list of X servers to be managed, one per line. A '#' indicates the begining of a comment; the comment extends to the end of the line. If the '#' is enclosed in quotation marks, it is not treated as a comment delimiter.

The format of each line is

display keyword server-data

where:

display	An X display specification.
keyword	May be "secure," "insecure," "foreign," "transient," or "remove." The keywords cause the following actions:
	secure — Indicates a local server that may not be unmanaged by a "remove" request.
	insecure — Indicates a local server that may be unmanaged by a "remove" request.
	foreign — Indicates a remote server to be managed. A new session is begun each time the previous one terminates.
	transient — Indicates a remote server to be managed for a single session only.
	remove — Request that a server no longer be managed by *Xlogin*.
server-data	For "secure" and "insecure" servers, this is a command to be executed to start the server. It is usually simply the name of the server (*e.g.*, */usr/bin/X11/Xqdss*) and the display number, but it may include command line arguments. For "foreign" and "transient" servers, it is a "magic cookie" used for simple authentication of later management requests. The "remove" request requires a matching cookie before it will take any action.

11. Format of Xresources

Xresources is a standard X resources file in the format used by the X Toolkit. The resources for the *Xlogin* display begin with "xlogin."

12. Xstartup, Xreset, Xstdrc, Xrc

The files *Xstartup*, *Xreset*, *Xstdrc*, and the user's *.Xrc* are all executable files, typically shell scripts.

13. Format of Management Requests

A management request is a single line of text in the same format as the Xservers file.

The format of messages returned to remote servers has not been fully defined yet.

14. Resources

The following resources can be specified for *Xlogin*:

greetFont	Font used in the greeting message.
failFont	Font used for login failure messages.
greeting	Message displayed to welcome the user to the system.
fail	Message displayed if login fails.
namePrompt	Prompt string for username.
passwdPrompt	Prompt string for password.
greetColor	Color for greeting message.
failColor	Color for failure message.

In addition, other standard resources (*e.g.*, borderWidth) may be specified.

This list is subject to change.

II. Sample Xlogin Configuration

This appendix contains sample Xlogin configuration files.

15. /etc/Xdm-config

```
DisplayManager.servers: /usr/lib/X11/xdm/Xservers
DisplayManager.errorLogFile: /usr/lib/X11/xdm/xdm-errors
DisplayManager*resources: /usr/lib/X11/xdm/Xresources
DisplayManager*startup: /usr/lib/X11/xdm/Xstartup
DisplayManager*reset: /usr/lib/X11/xdm/Xreset
DisplayManager*session: /usr/lib/X11/xdm/Xsession
```

16. Xservers

```
:0 secure /usr/bin/X11/X
```

17. Xresources

```
xlogin*greetFont: vri-30
xlogin*failFont: fgi-20
xlogin*fail: login failed, please try again...
xlogin*greeting: Welcome to the X Window System
```

```
xlogin*namePrompt: Username:\ \
xlogin*passwdPrompt: Password:\ \
login*Font: vr-20
#ifdef COLOR
xlogin*greetColor: #f63
xlogin*failColor: red
xlogin*Foreground: black
xlogin*Background: #fdc
#endif
xlogin*borderWidth: 3
xlogin*borderColor: green
```

18. Xstartup

```
#!/bin/sh
/usr/local/bin/display-motd
```

19. Xsession

No -x in /bin/test

```
#!/bin/sh
STARTUP=$HOME/.Xrc
if [ -f $STARTUP ]; then
        if [ -x $STARTUP ]; then
                exec $STARTUP
        else
                exec /bin/sh $STARTUP
        fi
else
        uwm &
        exec xterm -geometry 80x24+270+272 -ls
fi
```

20. Xreset

```
xhost -
```

III. Configuration for Project Athena

This appendix discusses configuration issues particular to Project Athena workstations and users. It includes descriptions of changes to the standard parts of *Xlogin* as well as proposed configuration files for Athena use.

21. Greeting the User

The current *Xlogin* system is adequate in this respect. The addition of a "Register" button is planned to provide new users will the ability to obtain an Athena account.

22. Verifying the User

Login authentication is performed using Kerberos, and the KRBTKFILE environment variable is initialized. The user's home directory is attached. If the home directory cannot be attached, the user is queried on whether she wishes to proceed with a temporary home directory. If so, a temporary home directory is created in */tmp/usernmame* and the login procedure continues.

If the file */etc/nocreate* exists, only users listed in */etc/passwd* may login.

If the file */etc/noattach* exists, home directories are not attached, but are assumed to exist locally.

An */etc/passwd* entry is obtained from the Hesiod name service and stored into the file. The user's password is not encrypted for this. (Note: the only program that depends on the encrypted password being present is *ftp*. The *ftpd* daemon should be modified to use Kerberos for authentication. Using the current scheme is a particular problem on private workstations, since the password file entries are not updated now.)

23. Miscellaenous Configuration Issues

The following files must be on the workstation's local filesystem:

- *Xlogin*
- *Xlogin* configuration files listed in *Xdm-config*
- The X server
- Fonts for use in *Xlogin*
- *Xmessage* (the program for displaying messages)

The use of *.Xrc* by users needs to be described in appropriate documentation. In particular, note that the *Xsession* script below will attempt to exec *.Xrc* if it is executable or treat is as a Bourne shell script if it is not. The user initialization files that start X applications must also be modified.

Deactivation after remote logins must be worked out.

24. /etc/Xdm-config

```
DisplayManager.servers: /etc/Xservers
DisplayManager.errorLogFile: /site/usr/adm/Xdm-errors
DisplayManager*resources: /etc/Xresources
DisplayManager*startup: /etc/Xstartup
DisplayManager*reset: /etc/Xreset
DisplayManager*session: /etc/Xsession
```

25. Xservers

For the VAXstation II and 2000:

```
:0 secure /etc/Xqvss -fp /etc/Xfont:/usr/lib/X11/fonts
```

For the RT/PC:

```
:0 secure /etc/Xibm -fp /etc/Xfont:/usr/lib/X11/fonts
```

26. Xresources

```
xlogin*greetFont: vri-30
xlogin*failFont: fgi-20
xlogin*fail: login failed, please try again...
xlogin*greeting: Welcome to Project Athena
xlogin*namePrompt: Username:\ \
xlogin*passwdPrompt: Password:\ \
xlogin*Font: vr-20
```

27. Xstartup

```
#!/bin/sh

umask 22

/etc/athena/save_cluster_info

if [ -f /etc/clusterinfo.bsh ] ; then
        . /etc/clusterinfo.bsh
else
        Xmessage "Can't find library servers."
        exit 1
fi

/bin/athena/attach -h -n -o hard $SYSLIB $USRLIB
/usr/athena/access_off

if [ -f /etc/athena/activate.local ]; then
        /etc/athena/activate.local
fi

/usr/athena/display-quota-info
/usr/athena/display-motd
```

28. Xsession

```
#!/bin/sh
STARTUP=$HOME/.Xrc
if [ -f $STARTUP ]; then
        if [ -x $STARTUP ]; then
                exec $STARTUP
        else
                exec /bin/sh $STARTUP
        fi
else
        uwm &
        exec xterm -geometry 80x24+270+272 -ls
fi
```

29. Xreset

```
#!/bin/sh

umask 22

detach $USER
/usr/athena/kdestroy

# Flush all NFS uid mappings

/bin/athena/nfsid -p -a

if [ ! -s /etc/utmp ]; then
        # Turn off remote logins (if access_off is accessible)
        if [ -f /usr/athena/access_off ] ; then
                /usr/athena/access_off
        fi

        # Next, restore the password and group files

        /bin/cp /etc/passwd.local /etc/ptmp
        /bin/mv -f /etc/ptmp /etc/passwd
        if [ -f /etc/group.local ] ; then
                /bin/cp /etc/group.local /etc/gtmp
                /bin/mv -f /etc/gtmp /etc/group
        fi
fi

sleep 120

if [ ! -s /etc/utmp ]; then
        # Tell the Zephyr hostmanager to reset state
        if [ -f /etc/athena/zhm.pid ] ; then
                kill -HUP `cat /etc/athena/zhm.pid`
        fi

        # Then clean temporary areas
        # (including temporary home directories)
        /bin/rm -rf /tmp/
        if [ -f /usr/bin/find ] ;  then
                (cd /usr/tmp; \
                find . -mtime +4 -exec /bin/rm -f {} \; )
        fi


        # Delete any hashed password files
        /bin/rm -f /etc/passwd.{dir,pag}

        # Do any desired local clean-up
        if [ -f /etc/athena/deactivate.local ] ; then
                /etc/athena/deactivate.local
        fi
```

```
        # Perform an update if appropriate
        if [ -f /srvd/auto_update ] ; then
                /srvd/auto_update
        fi

        # Finally, detach all remote filesystems
        /bin/athena/detach -h -n -a
fi
```