Unix/Programming Tools for Client Server

Michael B. Spring
Department of Information Science and Telecommunications
University of Pittsburgh
Pittsburgh, Pa 15260

Easy or Hard

• This course is either very easy or very hard. If you don’t know and use the tools available you will spend a lot of time spinning your wheels
• The following will make this course easier
  • Understand how to connect to Unix and work with the Common Desktop Environment (CDE)
  • Read and understand man pages and documentation
  • Know how to use relevant Unix utilities
  • Use a full featured editor for coding
  • Use an IDE for compilation and debugging
• The actual scope of what you need to know is greater than can be covered here, but this is a start
Overview

- A basic connections
- The Common Desktop Environment
- Unix help via man pages
- Selected Unix utilities
- Editors – nedit
- IDEs – workshop
Making a connection (3 ways)

• Login to a Unix workstation running CDE (best)
  • You can use all the graphical tools available to work
• Telnet from a PC (worse)
  • Use command line Unix
  • Use Unix line editors to work on files or use PC tools to create or manipulate files and FTP the files back and forth
• Turn a PC into a X-terminal and telnet (ok)
  • This allows all the benefits of a workstation login, but it can be slow

A PC telnet session

• Run telnet
  • On the start menu of an MS windows machine, there is a menu item called “Run”
  • Select this item and type “telnet” in the text box
  • When the DOS window opens with the telnet prompt, type “open” and the name of the machine to connect to:
    open paradox.sis.pitt.edu
  • When the machine responds, login to the remote host
  • When done, “exit” from the remote machine, and close the DOS window
A Basic telnet window

Microsoft (R) Windows 2000 (TM) Version 5.00 (Build 2195)
Welcome to Microsoft Telnet Client
Telnet Client Build 5.00.95203.1
Escape Character is 'CTRL+]'
Telnet Client> open paradox.sis.pitt.edu

More on the telnet window
A Couple Commands in Telnet

Common Desktop Environment
Using a Workstation

- A workstation provides the most natural access to Unix resources
  - Login at the login screen
  - Start at least one xterm session
  - Run any graphical applications you wish
  - Tailor your environment

Login
The Basics of the CDE

• Look at how to navigate the various desktops
• Learn how to add your own menu items
• Learn enough about X resources to tailor your environment

Main Screen
The Desktop Control Panel

The Menus

<table>
<thead>
<tr>
<th>Workspace Menu</th>
<th>Applications</th>
<th>Local Programs</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applications</td>
<td>❌ Application Manager</td>
<td>Adobe Suite...</td>
<td>☑ AnswerBook2</td>
</tr>
<tr>
<td>Local Applications</td>
<td>❌ Audio Control</td>
<td>Emacs...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Cards</td>
<td>❌ Audio and Video</td>
<td>FrameMaker...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Files</td>
<td>❌ Calculator</td>
<td>Ftptool...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Folders</td>
<td>❌ Calendar</td>
<td>GhostView...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Help</td>
<td>❌ Icon Editor</td>
<td>Mathematica...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Hosts</td>
<td>❌ Image Viewer</td>
<td>Mptool...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Links</td>
<td>❌ OW Clock</td>
<td>Nedit...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Mail</td>
<td>❌ Snapshot</td>
<td>Netscape...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Tools</td>
<td>❌ Text Editor</td>
<td>Star Office...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td>Windows</td>
<td>❌ Voice Note</td>
<td>X3270...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Xarchie...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Xcoral...</td>
<td>☑ Congratulations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Xv...</td>
<td>☑ Congratulations</td>
</tr>
</tbody>
</table>

Note: The menus contain various applications and tools for managing the desktop environment.
**Using a PC as an Xserver**

- Install Lan Workplace Pro or other X-window Server software on your PC
- Set your terminal type and any other configuration parameters you wish
- Establish either an XDM or Telnet connection to a Unix system
- Set the DISPLAY environment variable if it is not done automatically
- Start the editors, tools, and other X applications as if you were at a workstation
- Run other commands you need at the command line
Workplace Pro Startup

Workplace Configuration
Workplace Configuration

The PC X Display/Multiple Windows
Man Pages

man
the Basic Unix Help Facility

- It provides formatted help pages on every utility, system call, and API function
- For functions, the man page defines the arguments, purpose, and returns
- The man on man that tells you how to search, move around, and otherwise use man pages
- The man pages are in sections, so be careful, you may need to use the section switch to get the right section
- There is a -k switch, which lists all relevant man pages
**man on signal**

```
#include <signal.h>

void (*signal)(int sig, void (*disp)(int))(int);
void (*sigset)(int sig, void (*disp)(int))(int);
int sighold(int sig);
int sigrelease(int sig);
int sigignore(int sig);
```

**SYNOPSIS**

```
Standard C Library Functions
```

**NAME**

- signal
- sigset
- sighold
- sigrelease
- sigignore
- sigpause

Simplified signal management for application processes

**Example**

```
#include <signal.h>

void (*signal)(int sig, void (*disp)(int))(int);
void (*sigset)(int sig, void (*disp)(int))(int);
int sighold(int sig);
int sigrelease(int sig);
int sigignore(int sig);
```
xman

a Graphical Interface to man

- Allows for easier browsing
  - Index and pages open simultaneously
  - Recognition versus recall
  - Allows model of “sections”
- Allows easier manipulation
  - Scrolling back and forth
- Does not allow within page search
More Things to Do

• Read a couple include files
  • stdio.h
  • socket.h

• Read a complete standard – maybe something simple like http or smtp

• Read the man pages for:
  • the shell you use
  • grep
  • cc

Unix Utilities
Selected Unix Utilities & Commands

• There are literally 100’s of Unix utilities and commands that are useful to know. Only a few are covered here:
  • File transfer -- ftp
  • File searching -- grep
  • Process control -- ps, kill, &, bg, fg, and tee.
  • File viewing – more, tail
  • File location – which and knowledge of file structure i.e. usr, etc, spool, var, tmp
  • Shortcuts and accelerators – history, tab, and ln

ftp

• ftp stands for file transfer protocol
  • It was one of the first network programs in Unix
  • The ftp works through a client/server program set
  • The host system runs a server called ftpd
  • The client runs a client program normally called ftp
  • Some systems allow restricted “anonymous” ftp
• PC based interfaces tend to hide the underlying ftp commands, Unix ftp is normally command line
  • ftp does automatic conversion of line endings as needed
  • To turn this feature off, use binary mode. To turn it on, use ascii mode
Common ftp Commands

• get will get a single file
• put will put a single file
• binary specifies that files are to be moved byte for byte
  • ascii mode does pc to unix conversion of line endings
• prompt turns off prompting for mget and mput
  • mget will get multiple files
  • mput will put multiple files
  • hash prints a hash mark for each block moved
• Commands like cd and pwd operate on the remote system
• “! Command” allows a command to be executed locally from within ftp
grep

- The General Regular Expression Program or grep is a program to search streams
  - allows the user to search a standard I/O stream for terms or patterns
  - Allows the numbering of lines
  - Allows non-matches to be displayed
  - Allows counts to be developed
  - Helps with voluminous data streams
Process Control Commands

- ps can be used to list processes
  - `ps -ef` lists all processes with full info
- To run a process in the background, type the command followed by an `&`
  - `xman&` runs xman in the background returning the prompt to the user
- When you forget the ampersand, type `^Z` which will stop the process
  - Typing `bg` will then move the process to the background
  - If you stop a process (`^Z`) but do not move it to the background, the application will not accept input
File Commands

- When you execute a command, you will occasionally find what results is not what you expected. The command “which” allows you to identify the executable you are running – e.g. which nedit
- Knowing the basic Unix file structure is also very helpful – see next two slides
- At the command line you can page through a file or look at the beginning or ending using the commands more, head, and tail

Hierarchical File System

- Must have a root directory – /
- A user’s home directory is ~username
Important Directories

• Some standard directories include
  • /bin – user binaries
  • /sbin – binaries for system administration
  • /dev & /etc – device files system admin data files
  • /home – user file systems
  • /spool & /tmp – temp files for printing and other things
  • /usr – binary files (unix system resources)
    • /usr/lib – libraries for the linker
    • /usr/include – most of the include files
    • /usr/man – the man pages main location

Getting around and issuing commands

• Most students use Unix crudely because they don’t know better. Consider the following four things that can be done.
  • Most shells, including the default bash shell, have history. Cursor up to see commands you have already issued – allowing you to easily reissue the command.
  • History lines can be edited and the modified line run.
  • If you type the first few letters of a directory, filename, or executable and hit the tab key, Unix will complete the word or show the options.
  • Unix allows links (ln) that can make moving to a given directory as easy as typing just a couple letters.
A Couple GUI utilities

• Netscape is the default browser on Unix. Hotjava also is installed on solaris
• Netscape can be used:
  • For viewing web sites (netscape)
  • Editing webpages (netscape -c)
  • Reading mail and keeping address books (netscape -m)
• The mail tools used on Unix are multiple and all are imap and pop compliant
  • pine exists as a command line mail tool
  • mailtool is the default solaris mailtool
  • netscape provides a powerful mail capability

Netscape Navigator (netscape)
Netscape Messenger (netscape –m)

Subject: URL of 2779 project  
Date: Mon, 17 Dec 2001 14:49:53 -0500 (EST)  
From: Ming Luo <mimo@pitt.edu>  
CC: mingluo@pitt.edu, mingluo@pitt.edu  
To: Ming Luo <mimo@pitt.edu>, mingluo@pitt.edu  

Dr. Spring  
Sorry for forgetting to tell you the URL of our spider project.  
The URL is http://www.cs.pitt.edu/~minluo/databases/2779/FindSpider.html  
Ming Luo

Acrobat Reader (acroread)
Editors

- There are a variety of editors
  - vi and pico
  - emacs and xemacs
  - nedit
- The line editors are vi (universal), pico (very simple), emacs (very powerful).
- The graphical editors are nedit (powerful but simple) and xemacs (very powerful and not too hard to learn)
Nedit as an editor

• You must be in an X Window System environment to run nedit
• It is relatively simple to use and has a little built-in help
• It is important to keep in mind:
  • Preferences are at two levels, immediate and saved
  • Really useful for programming – syntax, braces, include files, indents, etc.

Nedit Preferences Menu
Nedit File Menu

Some Important nedit Features

- If you select a line number from a compiler error listing, you can select “go to line number” under search.
- If you select an include file line, selecting “go to selected” under file to open the file.
- If you place the cursor after a parenthesis or brace, the corresponding brace will be highlighted.
- The window can be split to see the beginning and end of a very long procedure.
- Line numbers can be added to the display.
- Syntax is graphically highlighted.
Integrated Development Environments – IDEs

• Writing, compiling and linking complex applications is difficult.
• Integrated development environments make it easier.
• Sun has two IDEs, which appear to be moving toward integration and which are constantly changing names. They are:
  • workshop (for C, C++, Fortran, etc.)
  • forte (for Java)
• This presentation briefly introduces workshop
Workshop Functionality

• Workshop provides four main functions from our point of view
  • The ability to construct a project which allows a set of files to be identified and manages the make process
  • An integrated editor
  • A debugger that includes graphical breakpoints and the ability to debug multiple processes
  • A facility for developing GUIs in a WYSIWYG fashion

Using Workshop

• To start workshop, type workshop at the command prompt
• On the file menu, use new project and the project wizard to set up a first project
• After creating a simple program – with a loop or something, build it.
• After it is built, choose to debug it and put a breakpoint just before the loop
• Choose a variable to display, and then step through the program a line at a time to see what happens.
Workshop Startup

Welcome to Sun WorkShop

Select this to create a new project from existing source files.

Create a New Project
Build a Simple Program
Create an Empty Project
Open an Existing Project

Show this dialog at startup
Show Tip of the Day

Next
Finish
Cancel

Select Project Type

Select type of application:

- Executable
- Static Library/Archive
- Shared Library
- Standalone Application
- Complex Application
- Java Application
- Native Application
- Windows Application
- Web Application
- Web Application with Web Start Support
- No Source Code

The Executable Application Wizard creates a project that:

- Builds a single executable file or a set of source files
- Links to the Solaris Link Library
- Supports Java 1.0 Console Application

Back

Project Wizard
Build Window

Debug Menu
debug controls

dump window

editor window

create watch

watch window