LIS901a

Lecture 0

Introductory Lecture

Thomas Krichel

2001–09–23

Structure

1. Course objectives
2. Introduction to Unix/Linux
3. Introduction to putty and wotan
4. practical arrangements

Reading

nothing

Objectives

1. learning about digital libraries
2. digital library building
3. jumping the quantum jump
4. from click computing to instruction computing

Implementation

1. realize that the learning curve is steep
2. use an environment without click
3. jumping the quantum jump
4. from click computing to instruction computing

now ask the students, Thomas

Why UNIX

There are free versions of UNIX around.

It is ideal as a server machine.

It is ideal as a multi-user machine where files can be shared between users. In particular, the instructure can intervene to sort out a problem of the student.

Why not UNIX

UNIX is terse. It is not user-friendly. It is expert-friendly.

It is widely used esp. in the academic world. It remains the operating system of most digital library systems that I know of. It is not likely to be replaced quickly.

UNIX history: Once there was Multics

Big project started in 64 to create the ultimate computer operating system. Participants where GE, Bell Labs and MIT. A lot of programs each capable of doing a lot of things.

Bell Labs dropped out of the project in 1969.

UNIX created in opposition to Multics at Bell Labs. Programs in UNIX would only be good a one or two things. But different programs would interoperate well. The output of one program would be the input of another.

UNIX History

The first version of UNIX was created in 1969 by Kenneth Thompson and Dennis Ritchie, system engineers at AT&T's Bell Labs. In 1977 it was first made commercially available.

At the same time a team from the University of California at Berkeley was working to improve UNIX. In 1977 it released the first Berkeley Software Distribution, which became known as BSD.

Meanwhile the AT&T version was developing in different ways. In 1983 Sun Microsystems produced a UNIX workstation. System V appeared, directly descended from the original AT&T UNIX and the prototype of the more widely used variant today.

What makes the success of UNIX

- Keringham and Ritchie also developed the C programming language. UNIX is written in C.
- Internet software implemented for BSD. UNIX became O/S of choice for networked machines.
- Bell Labs and UCB had liberal usage policies for the code.

GNU

stands for "GNU is Not UNIX". It is an attempt founded by Richard M. Stallman to write a "free" replacement for, and improvement on, UNIX.

free != $0

It's free speech rather than free beer.

The GNU public license

Users are allowed to use the program and to modify its source code. They are prohibited to distribute a modified version without distributing the source code, because that would restrict the freedom of others to make changes.

Source code is the plain computer code—usually written in C, or one of its derivatives—that is the basis for the executable program that is used by the end user.

Such software is now known as "open source software".

Linux

Initially, Linux is a free implementation of a UNIX kernel, written by Linus Thorvalds for the Intel hardware architecture. More generally, it is the packaging of that kernel with the GNU utilities and other free software to produce a completely operational computer.

There are many ways of doing that packaging. There are called distributions.

We use Debian/GNU linux. This is a more strictly free distribution for the die-hard hacker.

The credo of this course.

This course aims to help building a "free" digital library.

This is not a trivial task. It needs a lot of skill. This course will teach you the skill. It will take you onto the road of geekdom.

More importantly, it shows the bridge between open source to open library.

Junk hardware

wotan.liu.edu 148.4.2.231

Wotan is the chief god of the German legend.

It is a pentium with 166 MHz (I think). It has 2 times 32 Mega of RAM. 2 times 1.7 Giga of disk space.

But it has good software that does not require the resources that Mickeysoft products require.

users

- SuperUser, account name “root”
- special users
- ordinary users, such as you and me. They have a home directory **/home/***username*.

now create accounts for the students, Thomas

1st basic component: the kernel

a program that always run on the machine and that talks to the hardware of the machine.

On a linux machine, it is contained in the file **/vmlinuz**

2nd basic component: the shell

The shell is a command line interpreter. It reads you commands, and passes them on to the kernel.

There are various shells

- bourne shell **/bin/sh**
- korn shell **/bin/ksh**
- bourne again shell **/bin/bash**
- C shell **/bin/csh**
- exTended C shell **/bin/tcsh**

In Linux, the default shell is **/bin/bash**, invoked by **/bin/sh**. It is the only shell that we will be using.

Flags

Commands accept arguments and flags. Flags modify the way a command operates. Arguments give the object of a command.

Traditionally, flags are of the form *–letter*.

Example **`ls -l`** *file*

Such flags may be concatenated. Thus **`ls -la`** *file* is the same as **`ls -l -a`** *file*. Geeks like to be terse.

GNU utilities also accept a long format of options, for example **`man --apropos`** *string*. Note the double minus.

getting help

1. **`man`** *program*

displays paged information about the command or program *command*.

Leave the data display with “q”.

2. **`info`** *program*

opens a simple hypertext system with information about the program *program*.

Leave the data display with “Control-x Control-c”. Some program do not have info pages.

homework

install and operate putty at home

find a geeky way to use the “date” or “cal” program and show it to your classmate next week.

practical arrangements

teach on Thanksgiving

Finally

He's a real UNIX Man
Sitting in his UNIX LAN
Making all his UNIX plans
For nobody.

Knows the blocksize from du(1)
Cares not where /dev/null goes to
Isn't he a bit like you
And me?

UNIX Man, please listen(2)
My lpd(8) is missin'
UNIX Man
The wo-o-o-orld is at(1) your command.

He's as wise as he can be
Uses lex and yacc and C
UNIX Man, can you help me At all?

UNIX Man, don't worry
Test with time(1), don't hurry
UNIX Man
The new kernel boots, just like you had planned.

He's a real UNIX Man
Sitting in his UNIX LAN
Making all his UNIX plans For nobody ...
Making all his UNIX plans For nobody.