CIS 18A
Introduction to Linux / Unix

History and Background

De Anza College
Instructor: Clare Nguyen

Part 1

Topics:
- Intro to Unix / Linux
- History

What is Linux / Unix?

- Linux / Unix is an *operating system*
- A computer can be divided into 3 main functional layers:
  - *Applications*: the highest functional layer. This layer includes software programs that run for a specific purpose.
    Example: this power point presentation, an email tool, a web browser.
  - *Operating System (OS)*: the middle functional layer. The OS controls the hardware and supports the applications running above it.
    Example of popular OS: Linux, Windows, Mac OS, Android, iOS
  - *Hardware*: physical devices that make up the computer.
    Example hardware include: keyboard, monitor, hard disk.
- The main tasks of an operating system:
  - Interacts with the user
  - Manages how applications run
  - Handles data storage
  - Controls hardware peripherals

Why Linux / Unix?

Linux are used for many different purposes in a wide range of applications:

- The internet runs on Linux servers.
- Movie special effects are created on Linux machines.
- Banking and financial transactions are done on Linux computers.
- Video games are developed on Linux platforms.
- Military computing are done on Linux machines.
- Government records, such as taxes, are kept on Linux computers.
- University engineering and computer departments run classes on Linux computers.

History of Linux / Unix (1 of 4)

- Unix was officially announced to the world in 1969
- Before Unix, computers had operating systems that are:
  - hardware specific: Changing computers meant re-writing or buying all the software so they could work with the new hardware.
  - single tasking: Only one job could be done at a time on the computer, everyone had to sign up for a timeslot on the computer.
  - single user: Sharing data meant making a tape of your data and giving it to another user.
- Multics:
  - In the 1960s, scientists at AT&T Bell Labs, MIT, and GE worked together to create a new time-sharing operating system. Time-sharing is the new concept that will eventually lead to a multitasking, multi-user OS later.
  - However the Multics project became too large. It was difficult to manage and the project ended in 1969.
- Unix:
  - Some of the developers from Bell Labs used their knowledge gained from Multics to create an even more progressive operating system, which they originally named Unics, a play on the word Multics.
  - Eventually the spelling of Unics changed to Unix.

History of Linux / Unix (2 of 4)

- Characteristics of Unix
  - *Multi-tasking*: the OS appears to do multiple tasks at the same time by switching between all the tasks and giving its time to each task one at a time (also known as *time-sharing*).
  - *Multi-user*: the OS can support many users logging in at the same time, as well as one user logging in multiple times at once.
  - *Portable*: the OS is written in a high-level language (the C language) and therefore can be run on different hardware platform.
- University involvement
  - In 1969 Bell Labs released Unix, for free, to the Engineering and Computer Science departments at different universities.
  - The universities embraced Unix and their graduate students began work to develop more features by adding to the source code. One such university was the University of California, Berkeley.
  - Eventually there were 2 major versions of Unix:
    Unix System V (developed by AT&T)
    Unix BSD (developed by UC Berkeley)

History of Linux / Unix (3 of 4)

- Industry involvement
  - The graduate students eventually went on to found successful start up companies such as Sun Microsystem, SGI, etc. and brought Unix to the high tech industry in the late 1970s and early 1980s.
  - The success of the start up companies caused many established companies such as IBM, Microsoft, HP, etc. to also adopt Unix.
  - Unix became the OS platform for many companies in the high tech industry.
  - One serious side effect of corporate involvement was that Unix became copyrighted and had a high price attached to it. It was partly because of this high price that Microsoft, with its less expensive DOS operating system, became the operating system of choice when desktop computers became widely available to the general public.
- Linux
  - Fed up with the copyrighted source code trend, the Free Software Foundation was formed to establish the open source movement. This movement believes that software should be open for anyone to work on and be free of charge.
  - Supported by the open source movement, in 1991 Linus Torvalds created a Unix operating system that is meant to run on a desktop computer. He named it Linux.

History of Linux / Unix (4 of 4)

- Linux today
  - Used in many applications, from embedded systems to servers to supercomputers.
  - Many people around the world maintain and enhance Linux, since it is an open source product.
  - Different packages of Linux are called *distributions*, and can be downloaded and installed for free.
- Linux vs. Unix
  - Unix is proprietary to each company that uses the Unix platform. Some examples are IBM AIX, HP UX, SGI IRIX, Sun Solaris, DEC Ultrix. These OS are all Unix based and follow the same, uniform Unix standard called POSIX.
  - Linux is the same way, it follows the same POSIX standard. But Linux is open source and is free.
  - All utilities covered in this class follow this POSIX standard and will work with both the proprietary Unix and with Linux. Therefore for the rest of this class, "Linux" will be used to shorten the full name "Linux / Unix".

Accessing Linux

- To access a Linux system you need to have an account, with a user ID and a password.
- A *user ID* is also called a *log in ID* or *user name*
  - It is assigned to each user by the system administrator.
  - There is a unique user ID for each user in the system.
  - A user may have more than one user ID.
  - Depending on your organization, you may be allowed to make up your own user ID or you may be assigned a specific user ID.
  - You cannot change the user ID yourself, it must be changed by a system administrator.
- A *password* is also required to log in to the system
  - Depending on how the system is set up, you may be able to create your own password or you may be given a password.
  - You can change your password any time by using a utility. Depending on how it is set up, some systems require you to change your password on a regular basis for security reasons.
  - Typically there are rules for the password that you choose. At school, your password must be a combination of letters and numbers or punctuation marks.

Part 2

Topics:
- Logging in
- Users
- Main components
- Types of users
- ssh
- sftp

Logging in to voyager – The Basics

- voyager is the name of the school Linux system.
  If you are registered for the class, you have a voyager account.
  You can log in to the voyager server from the classroom, or from the CIS main lab, or from any computer where you have Internet access.

- voyager is set up for a **c**ommand **l**ine **i**nterface (CLI).
  This means your interaction with voyager will be by typing one command line at a time and seeing the system output on screen as text.

- The opposite of a command line interface is a GUI, or **g**raphical **u**ser **i**nterface.
  In a GUI environment, you can use the mouse, have multiple windows and menu bars. (Microsoft Windows is an example)

- Linux has a GUI package that is equivalent to Microsoft's or Apple's GUI.
  However, we Linux users are more technically sophisticate. We use the command line interface by choice because it's more efficient; or because sometime it's the only option, such as when working in an embedded environment or when doing basic networking with a server (or when you are a CIS 18A student at De Anza).

Logging in to Voyager - Preparation

- To set up your local system (your computer) to log in to voyager:
  - If your local computer is a Windows system, you need 2 applications that can communicate between Windows and Linux.
    - Download the *winscp416setup.zip* file from the class website, under the *putty and winscp* link.
    - Follow the instructions at the link on the class website to extract the *putty.exe* and *winscp.exe* files from the zip file. I recommend that you extract both files to your desktop, or some place that's easy to access on your computer.
  - If your local computer is a Mac or Linux machine, you can use the command window of your computer to log in to voyager. You will not need winscp or putty.

- To create a voyager account:
  - If you already have a CIS Windows account set up:
    You don't need to do anything, your Windows user id and password are the same as your voyager user id and password.
  - If you don't have a CIS Windows account:
    Go to https://reports.fhda.edu/php/stuact.php to fill out the online request for a CIS computer account.
    When asked to enter your choice of user id, please do not use your De Anza student id number. The reason will be obvious when we start to use the system in the next module.

Logging in to Voyager – The Process (1 of 2)

- To log in to voyager from a Windows system
  - Once you have your user ID and password set up, at any time you can click to open *putty* and type voyager.deanza.edu as the hostname.
  - Use your user ID and password to log in. Note: *nothing* will show up on screen when you type in your password (unlike the * that appears with the other inferior OS).
  - Be careful typing in your user ID and password. If you tried unsuccessfully 3 times in a row, the system will lock your account because it assumes that someone is trying to enter the system illegally. If this does happen, let me know.
  - You know you've successfully logged in to the system if you see the prompt: [yourUserID@voyager directoryName] $
    This is known as the shell prompt.
- Once you've successfully logged in, the first utility to learn is how to log out: type exit and hit the enter key.
- If you typically will log in from a Windows system, you should still read the next slide about ssh, the Linux utility that is used by putty to help you log in.

Logging in to Voyager – The Process (2 of 2)

- To log in to voyager from a Linux system or a Mac
  - You need to use the utility ssh (for **s**ecure **sh**ell).
  - ssh allows a user to log in remotely to another system. The term *secure* refers to the fact that data are encrypted when transferred between the local and remote systems.
  - ssh runs from a command window on Linux or the Mac.
  - Format: ssh –l login_id   system_name
    -l for **l**ogin, this tells ssh that login_id is the user id for the remote system.
    system_name is the remote system name.

    For example, to log in to voyager, the command line is:
    ssh –l your_voyager_user_id   voyager.deanza.edu
  - When the connection to the remote system is established, ssh will prompt for a password, and when the password is verified, the command line prompt will appear.
  - To disconnect: exit

Main components of Linux (1 of 2)

The kernel, the file system and utilities, and the shell are the main components of the Linux OS

Hardware
Kernel
File system and Utilities
Shell — user
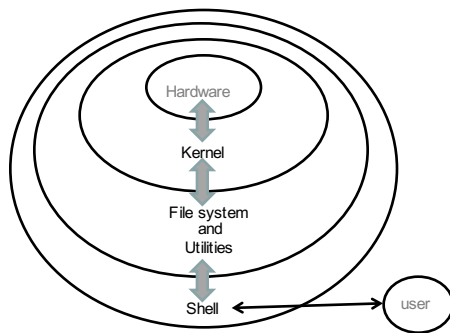
Main components of Linux (2 of 2)

- *Kernel*: the core of the OS
  - only one kernel can run per system
  - gets loaded into the main memory when the system powers up
  - manages resources so that multitasking can take place
- *File system*: for data storage
  - data and external devices are organized into files
  - files are organized into directory structures
- *Shell*: how the user interfaces with the OS
  - the shell interprets what the user types on the keyboard so it can be run by the system
  - there can be more than one shell running at a time
  - the shell also coordinates multiple commands in a file and acts as a programming language
- *Utilities*: tools to help a user do work on the system
  - also called commands
  - there are many utilities, each utility is supposed to do one task and do it well
  - they are program executables that are run by the user to do specific tasks, such as copy a file, send email, communicate with another user, check system resources, etc.

Types of Linux Users

- A *user* is a person who has a Linux account and can log in to the Linux system.
- When a user creates a file, the user is the owner of the file and the term 'user' also means owner (of the file).
- There are 3 main types of users:
  - Regular users use the Linux system as a platform to do their work. For example, on a Linux server there can be a game developer writing code for games, a QA person testing software, a student writing a paper. They are all regular users who use Linux utilities to do their work.
  - System programmers write and maintain code that makes up the kernel, core system libraries, core utilities, and system daemons.
  - System administrators set up, maintain, and update Linux systems. It is the sys admin's job to keep the system running so that all users can do their jobs.
- For this class, we are regular Linux users.

File Transfer

- One common utility for transferring files from one system to a remote system is sftp (for **s**ecure **f**ile **t**ransfer **p**rotocol)
- To transfer files between Windows and Linux
  - Use the tool winscp, which is a wrapper program for sftp.
  - Open winscp, then enter voyager.deanza.edu at the address, then your voyager userID and password.
  - winscp will use sftp to connect to voyager, and will bring up a 2 panel window. One panel is your home directory on Windows, the other is your home directory on voyager.
  - To transfer file, you select a file and drag and drop in the proper panel.
- To transfer files between a Linux or Mac system
  - Use sftp in a command window.
  - Type on the command line: sftp username@voyager.deanza.edu
  - You will be prompted for the password.
  - When you successfully log in to voyager, use
    - mput filename to transfer the file from the local system to voyager
    - mget filename to transfer the file from voyager to the local system

Next stop: Basic Utilities