OPS235: Week 1

Hard Disks & Partitioning

Lab 1's Topics

- Lab Time
  - Hard Disk Basics / Hard Disk Names
  - Typical Directories in Linux File System
  - Linux File System (Types)
  - Tips for Performing Lab #1

  - Lab 1 Basic Commands (Throughout Lab)

Hard Disk Basics

- Hard disk made up of **platters**. Each side of platter have numbered concentric circles called **tracks** (starting from 0). Tracks are divided into **sectors** (starting from 1). These circles are magnetic tracks rearranged by read/write heads.

- Master Boot Record (**MBR**) on track0, sector1. Used to indicate # of partitions and pointer to specific partition / location to boot system.

- In Linux, loaders like *LILO* or *GRUB*, place this info on MBR; otherwise, Linux will not boot properly.

Hard Disk Basics

- A **partition** is a virtual drive inside a hard drive.

- There are many <u>advantages</u> in creating separate partitions including: separation of OS from programs, grouping program files, multi-boot systems, database efficiency (file sizes).

- **BIOS** limits the number of partitions on a single hard drive:

  - Maximum of 4 **primary partitions**.
  - **Extended Partition** (a container for up to 16 logical partitions). To create more than 4 partitions, need to create at least one extended partition.

Hard Disk Basics

- All devices stored in the **/dev** directory (including hard disks). Good to study typical directories in a Linux file system...

- Hard drives begin with **hd** or **sd** depending on type of hard drive (eg. IDE or SCSI & SATA). A letter denotes each hard disk, and a following number denotes the partition number.   **/dev/sda1**, **/dev/sda2**, **/dev/hdb5**

- Unlike windows where your make reference to drives, in Linux all drives (and corresponding partitions) are files. There is more flexibility to mount different drives / partitions for different purposes: For example **/**, **/home**, **/opt**, etc...

5

Linux File Systems

- All operating systems create file systems to store data (programs files, directory files, regular files, etc).

- A system administrator needs to not only know which file system to create (eg. Linux, Windows, Unix), but which type of file system is best suited for their needs.

- There are several types of file systems in Linux:
  - **Ext2** – Can set size of blocks to speed up data transfer.
  - **Ext3** – Journalling system to record activity of file system. In case of a system crash, activity can be read from journal to correct problems.
  - **Ext4** – Supports large volume sizes, faster system file checking, backwards compatible with ext3

Notes Before Performing Lab #1

- Extra Paper → condense into lab notes.

- Even perform operations in WARNING sections if you are asked to do so...

- Don't worry about making mistakes. Lab is designed to have you make mistakes and learn.

- Adopt trouble-shooting techniques if problems. Try to refer to previous "investigations" to see what is wrong.

- If taking too much time, call instructor or lab monitor.

- Try to record as much as possible in lab notes, since this will be resource for quizzes, term tests, and final exam.

System Admin: Hard Drives

- How to log in as **Superuser**?
- What is Difference between **su** and **su -**
- What is purpose of **/root** directory?
- What is purpose of **whoami** and **pwd**?
- When should you use above commands?

- What is a **mount-point**?
- What is the purpose of **mount** / **umount** commands?
- What is the purpose of **udevd**?
- What various uses (syntax) of **mount** command?
- What steps must be performed when using **mount** command?
- How to verify that you have set up correct mount point?

Reality Check

- We want to practice manually mounting devices:

  - Why do we want to mount devices?

  - Steps to <u>manually</u> mount devices (after boot-up) (List correct order of steps).

  - Why mount for <u>different</u> file systems? (What type is common for USB keys?)

  - Why do we need to kill **udevd** first before manually mounting devices? What are the consequences for not killing *udevd* process?

  - How to determine device name if udevd killed?

  - What common error message do you get when trying to mount drives? How to fix those errors?

System Admin: Partitions

- What is the command to create a **partition**?
    - What are the various options (command line, internal)?
    - What could prevent partition from being created?
    - How to create partitions for other Operating Systems?
    - How to create a file system in a partition?

- What is the purpose of **mkfs** command?
    - What is purpose of **lost+found** directory?
    - What happens if **umount** while in mount-point?
    - What is the purpose of a **volume label**? How to make?
    - What command provides information regarding device's file system? What information is available?

System Admin: Images

- Saving **Images** on a File System
  - Purpose of Saving images (i.e. practical uses)?
  - How to create and save an image on file system?
  - How to mount saved image?
  - Can contents in mounted images be saved for future mounting?
  - What are problems regarding Copying images?
  - What is purpose of backing-up MBR?
  - Command to backup MBR?

Lab1 Completion

- When you have finished your lab, shrink information on lab1 sheet (lab log book).

- Follow lab instructions to have tasks completed for instructor to view on your computer.

- Call over your instructor to verify results from your computer and your lab1 log book
(If successful, instructor will "sign-off")