OPS235: Week 5

Lab4: Investigations 1 – 3

Lab 4 – Part 1 - Topics

Lab Time

- **Manually Mounting / Un-mounting Partitions**
  - Purpose
  - Command-Line
  - How to automate (/etc/fstab)

- **Adding PVs to existing VGs**
  - Purpose
  - Graphical
  - Command-Line

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...

System Admin: Hard Drives

- What is a **mount-point**?
- What is the purpose of **mount** / **umount** commands?
- 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?

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?

- How to format a hard drive partition?

Creating Physical Volumes

- As mentioned, LVM allows for scalability: for example, adding hard disks to allow file systems to grow.

- We have demonstrated how to extend Logical volumes via Graphical and Command Line methods.

- We only have 1 removable hard disk, but we can use LVM to "simulate" adding another Physical Volume to an existing Volume Group. This would allow extra space to "grow a file-system" if required. **In other words, we can create a "virtual partition" for practice**.

- **What is the method to add a PV to an existing VG? (graphical method, Command Line Method)**.