# 8.5.3 File System Mounting Facts

Hard disk partitions, optical drives, USB drives, and other like devices in a Linux system must be mounted before they can be used.

This lesson covers the following topics:

- About mounting
- Managing and monitoring mountings
- Managing file system mountings
- Troubleshooting storage issues

## About Mounting

*Mounting* is the process of making a storage device accessible to users through the directory tree. The file system directory that is used to access the files on the device is called the *mount point*.

- Partitions and LVM logical volumes are represented by device files located in the **/dev** directory. However, these storage devices must be mounted before the data on them can be accessed.
- A storage device can be mounted in a directory in the file system. When accessing the directory in the file system, you are actually accessing the device mounted in that directory.
- You should mount storage devices in empty directories. Mounting a volume to a directory that contains data makes the data inaccessible.
- The **/mnt** and **/media** directories (depending on the system configuration) are directories that contain mount points specifically for external storage devices (e.g., CD-ROM drives, floppy drives, magnetic tape drives).

## Managing and Monitoring Mountings

The following files manage and monitor the file system mounting:

| File | Description |
|---|---|
| /etc/fstab | The **/etc/fstab** file identifies devices to mount each time the system boots. When the system boots, it automatically mounts the volumes identified in the file. The file contains entries with six fields that control how a device is mounted. The following is a typical fstab entry: |

| | |
|---|---|
| | /dev/sda3 /mnt/disk1 ext3 auto,ro,nosuid,users 0 1<br><br>An entry consists of the following variables, which are described below:<br><br>[*device_to_mount*] [*mount_point*] [*file_system_type*] [*options*] [*dump*] [*fsck*]<br><br>• *Device_to_mount* is the path to the device file or the label that describes the storage device to be mounted.<br>• *Mount_point* specifies where to mount the device. This is the directory where the data on the device can be accessed.<br>• *File_system_type* specifies the type of file system that has been created on the storage device.<br>• *Options* specify the additional options to be used when mounting the device. Multiple options are separated by commas.<br>  ◦ **sync** enables synchronous I/O. Changes are written to disk immediately. Usually used for removable storage devices. (**async** disables this function.)<br>  ◦ **async** enables asynchronous I/O. Changes are cached and then written when the device isn't busy. Usually used for non-removable devices such as hard drives. (**sync** disables this function.)<br>  ◦ **atime** updates the timestamp on each file's inode. (**noatime** disables this function.)<br>  ◦ **auto** allows the device to be mounted automatically when the system boots.<br>  ◦ **noauto** prevents the device from being mounted automatically when the system boots.<br>  ◦ **dev** allows block files to be read from the device. (**nodev** disables this function.)<br>  ◦ **exec** allows programs and script files in the file system to be run. (**noexec** disables this function.)<br>  ◦ **owner** identifies that only the device owner can mount the file system.<br>  ◦ **ro** mounts the storage device as read-only.<br>  ◦ **rw** mounts the storage device as read/write.<br>  ◦ **suid** allows the SUID bit to be set on files in the file system. (**nosuid** disables this function.)<br>  ◦ **nouser** allows only the root user to mount the file system.<br>  ◦ **users** allows any user to mount the file system.<br>  ◦ **defaults** uses the following default settings: rw, suid, dev, exec, auto, nouser, and async.<br>• *Dump* determines whether the file system needs to be dumped. If set to a value of **0**, it is assumed that the file system does not need to be dumped. If set to a value of **1**, the file system will be dumped.<br>• *fsck* determines the order in which to run **fsck** (file system check) during system boot. This field should always be set to a value of **1** for the device containing the root file system (/). All other file systems should be set to a value of **2**. |
| /etc/mtab | The **/etc/mtab** file tracks the currently mounted volumes on the system. |
| /procs/mounts | The **/procs/mounts** file contains entries for all currently mounted volumes on the system. |

# Managing File System Mountings

Use the following commands to manage the file system mounting:

| Command | Description | Example |
|---|---|---|
| **mount /dev/[*device*] [*mountpoint*]** | Mount a volume or device. Common mount options:<br>• **-a** mounts all file systems listed in the **/etc/fstab** file.<br>• **-r, ro** mounts the volume as read only.<br>• **-w, rw** mounts the volume as read/write.<br>• **-t** specifies the volume type (If you mount an ext3 file system without the **-t**, the system recognizes it as an ext2 file system).<br>• **-o loop** mounts an ISO file. | • **mount -a** reads the **/etc/fstab** file and mounts all volumes listed (except those with the **noauto** option).<br>• **mount -rt reiserfs /dev/sdc1 /mnt/reis** mounts the sdc1 device using the Reiser file system read-only to the **/mnt/reis** mount point.<br>• **mount -t iso9660 /dev/sr0 /media/cdrom** mounts an optical disc device to the **/media/cdrom** mount point.<br>• **mount -wt vfat /dev/fd0 /mnt/floppy** mounts the fd0 device with the VFAT file system as read/write to the floppy mount point. |
| **mount** | View the currently mounted volumes on the system. | • **mount /etc/mtab** displays the contents of the **/etc/mtab** file. |
| **df** | View which file systems are mounted to specific mount points. | |
| **umount [*device*]**<br>**umount [*mountpoint*]** | Unmount a volume or device from the system. If a "disk is busy" error message is displayed when unmounting a device:<br>• Make sure the current working directory is not in that file system.<br>• Close any open files located on that file system. | • **umount /dev/sdc1** unmounts the sdc1 device.<br>• **umount /mnt/reis** unmounts the device on the /mnt/reis mount point.<br>• **umount /dev/sr0** unmounts the optical disc device.<br>• **umount /mnt/cdrom** unmounts the device on the /mnt/cdrom mount point (most likely an optic |

# Troubleshooting Storage Issues

Many storage issues center around the physical aspects of the storage device. Other issues involve operating system configurations. A good strategy for troubleshooting storage issues are to start checking the simplest system features and then move to the more complex features. Many problems are resolved outside of Linux. Here are a few general items to check before moving to the Linux operating system.

- Determine whether the storage device has power. If the device is external to the computer, ensure it's plugged in and turned on. A power light is also a good indicator.
- If the drive has platters (it's not a solid state device), determine that the drive is spinning. You should be able to hear the head moving across the platters.
- Determine if the drive is recognized by the BIOS/UEFI. If not, the problem could be a hardware issue.

The following table list some issues that are detected and resolved with Linux.

| Storage Issue | Description | Resolution Techniques |
|---|---|---|
| Degraded storage | Disk volumes, especially RAID sets may show a status of degraded. This means that I/O errors have been detected on a region of the disk. This can occurs when one disk in a RAID set is offline or is unable to provide the proper RAID redundancy. | Repair or replace the disk volume. |
| Missing devices or missing volumes | A storage device with hardware problems may not show in the /dev directory. Or, the partitions on a storage device might be missing. | The **fdisk -lu** and **parted -l** commands may give some clues to the problem. |
| Missing mount point | Missing mount points indicate that a partition is not mounted. Often, this happens after a reboot. | Ensure that the proper entry has been added to the **/etc/fstab** file. |
| Disk performance | A slow disk can dramatically affect the total performance of a Linux system, especially if the system partition is on the slow disk. | The **atop** command can be used to monitor disk I/O stats. This may give an indication of a slow disk. |
| Resource exhaustion | A full volume, especially if it is the system volume can cause Linux to freeze or crash. | Use the **df** command to monitor disk space. |
| Adapters | Physical storage devices are attached to storage adapters. Example are SCSI, RAID, SATA host bus adapters (HBAs). Other HBAs, like a Fiber Channel adapter can connect to storage area networks. | Use **/sys/class/scsi_host#/scan** to scan for storage devices connected to the SCSI adapter. |

| | | |
|---|---|---|
| Storage integrity | Bad sectors and bad blocks on a storage devices can corrupt data. | Use the **badblocks** command to scan for bad sectors and blocks. |