Lab 4: Creating and Mounting Filesystems

In this lab, you will create partitions and filesystems, making them accessible to your system. Each team will turn in answers to the questions in this lab

Step 1. Boot the system up into multi-user mode. You can use either the graphical or text console.

Create a DOS Filesystem

*In this part of the lab, you will create a DOS filesystem and use the **mtools** package of commands to access the floppy as a DOS floppy. The **mtools** commands allow you to perform DOS-like commands using DOS floppies; this is often convenient so that files can be transferred from a PC to UNIX or vice-versa.*

Step 2. Filesystems are created with the **mkfs** command. Insert the floppy and create a new DOS filesystem on the floppy:

```
# mkfs -t msdos /dev/fd0
```

Step 3. Copy a file from the UNIX filesystem to the newly formatted DOS floppy

```
# mcopy /etc/passwd a:
```

Step 4. List the contents of the DOS floppy, and look at the file's contents:

```
# mdir a:
# mtype a:passwd
```

Step 5. Format the DOS floppy and validate that it now has a new format:

```
# mformat a:
# mdir a:
```

Creating and using a Linux ext2 filesystem

*In this part of the lab, you will create a native **ext2** Linux filesystem on the floppy. Although you normally would not create this type of filesystem on a floppy, the procedure is the same for IDE or SCSI hard disks.*

Step 6. Create a new Linux **ext2** filesystem on the floppy. We need to specify the type as **ext2**. The **mkfs** command will destroy any existing data on the disk (or partition):

```
# mkfs –t ext2 /dev/fd0
mke2fs 1.18, 11-Nov-1999 for EXT2 FS 0.5b, 95/08/09
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
184 inodes, 1440 blocks
72 blocks (5.00%) reserved for the super user
First data block=1
1 block group
8192 blocks per group, 8192 fragments per group
184 inodes per group

Writing inode tables: done
Writing superblocks and filesystem accounting information: done
```

Step 7. To use the new filesystem, you need to mount it somewhere under root. Mount points are often created under directories such as /mnt. Create a mount point:

```
# mkdir -p /mnt/floppy
```

Step 8. Now mount the **ext2** filesystem onto the mount point:

```
# mount -t ext2 /dev/fd0 /mnt/floppy
```

Step 9. Now, the new **ext2** filesystem is ready for use. It can be accessed via the /mnt/floppy path.

```
# echo "Hello World" > /mnt/floppy/hello.txt
# cd /mnt/floppy
```

```
# cp /etc/passwd .
# mkdir newdir
# touch newdir/file1
# cd newdir
# touch file2
```

Step 10. Examine the /mnt/floppy filesystem:

```
# ls -RF /mnt/floppy
/mnt/floppy:
hello.txt  newdir/  passwd

/mnt/floppy/newdir:
file1  file2
# df -k /mnt/floppy
Filesystem           1k-blocks      Used Available Use% Mounted on
/dev/fd0                  1423         2      1421   1% /mnt/floppy
```

Step 11. Any mounted filesystem **must** be *un-mounted* before the device can be removed. Failure to do this **will** cause filesystem corruption. Also, any active process that has its working directory within this filesystem will cause the un-mount to fail. Try to un-mount the /mnt/floppy filesystem, and compare the error message below with any error messages you receive:

```
# umount /mnt/floppy
umount: /mnt/floppy: device is busy
```

Step 12. To successfully un-mount the floppy, your working directory must not be within the filesystem:

```
# cd /
# umount /mnt/floppy
```

Step 13. Swap floppies with your neighbor and insert the floppy into your systems drive. Try to look at the contents of **/mnt/floppy**:

```
# ls -l /mnt/floppy
```

There is nothing there! The filesystem has to be mounted before it can be used.

Step 14. Before using a filesystem, it should be checked using the **fsck** command, which will validate the integrity of the filesystem, and fix any inconsistencies that may exist. Using a corrupted filesystem can crash the system, or worse, cause incorrect data to be used. Once the filesystem comes up as **clean** it is safe to be mounted.

```
# fsck –t ext2 /dev/fd0
Parallelizing fsck version 1.18 (11-Nov-1999)
e2fsck 1.18, 11-Nov-1999 for EXT2 FS 0.5b, 95/08/09
/dev/fd0: clean, 11/184 files, 41/1440 blocks
```

Step 15. Mount the filesystem on the floppy with the command below:

```
# mount /mnt/floppy
```

Step 16. Compare the command above with the command in Step 8 - notice only a single argument above was given - the mount point. The **-t type** argument and the device argument **/dev/fd0** was not supplied this time. This works because the **mount** command consults the file **/etc/fstab**, which defines how filesystems should be mounted (see **man fstab** for the format of this file). There must already be an entry in **/etc/fstab** for the floppy device:

```
# grep floppy /etc/fstab
  /dev/fd0                /mnt/floppy             auto    noauto,owner    0 0
```

Step 17. The **mount** command without any arguments will output the mounted filesystems:

```
# mount
/dev/hde2 on / type ext2 (rw)
none on /proc type proc (rw)
/dev/hde4 on /home type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
```

```
/dev/fd0 on /mnt/floppy type vfat (rw,nosuid,nodev)
```

Step 18. Examine the filesystem the same as you did in Step 10 to see if everything looks correct. If not, give the owner of the floppy a good Doh! sound. In either case, un-mount floppy. Notice that you can use either the mount point name, or the device name. This time, we've used the device name **/dev/fd0**. Eject the floppy and return it to the owner.

```
# umount /dev/fd0
```

Adding Additional Swap Space

Sometimes it is necessary to add additional swap while the system is running. In this part of the lab, you will configure and add additional swap space to the system.

Step 19. The **mkswap** command sets up a swap area on a device or a file:

```
# mkswap /dev/fd0
Setting up swapspace version 1, size = 1470464 bytes
```

Step 20. Use the **swapon** command to add the pre-configured swap space to the system:

```
# swapon /dev/fd0
```

Step 21. The **-s** option to **swapon** shows the system's swap space (your output will differ). What is the device name for primary your swap partition? How much additional swap space was added to the system?

```
# swapon -s
Filename                        Type            Size    Used    Priority
/dev/hde5                       partition       136512  0       -1
/dev/fd0                        partition       1432    0       -3
```

Step 22. The **swapoff** command removes the swap space from the system:

```
# swapoff /dev/fd0
```

Examining and Partitioning Disks

*IDE & SCSI drives need to be partitioned before a filesystem can be created. **fdisk** is used to manipulate partition information.*

Step 23. Use the **mount** command to determine the device name of your hard disk where the / filesystem is located. In the output below, the / filesystem is on the device /dev/hde2 (hde2 refers to only the second partition of this device, and not the entire device). The device entry that specifies the entire drive would be /dev/hde.

```
# mount
/dev/hde2 on / type ext2 (rw)
none on /proc type proc (rw)
/dev/hde4 on /home type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/fd0 on /mnt/floppy type vfat (rw,nosuid,nodev)
```

Step 24. You are going to use the **fdisk** command to examine the disk - you can easily corrupt the partition tables and the data on the disk; please follow the instructions below. Start the **fdisk** command, supplying as an argument the correct disk name you found above. The **p** command in **fdisk** will print out the partition table. Type **p** and hit Enter:

```
# fdisk /dev/hde
Command (m for help): p

Disk /dev/hde: 255 heads, 63 sectors, 4982 cylinders
Units = cylinders of 16065 * 512 bytes

   Device Boot    Start       End    Blocks   Id  System
/dev/hde1           656      1930  10241437+  1c  Hidden Win95 FAT32 (LBA)
/dev/hde2   *         1       638   5124703+  83  Linux
/dev/hde3           639       655    136552+   5  Extended
/dev/hde4          1931      3205  10241437+  83  Linux
/dev/hde5           639       655    136521   82  Linux swap

Partition table entries are not in disk order
```

Step 25. Examine your output and determine the following. When you are done, use the **q** key to quit **fdisk**.

Q1. The device name of the disk? _____

Q2. The number of disk partitions? _____

Q3. The size in bytes of each partition?_____

Q4. The meaning of the Id field (hint: use the **l** (lowercase L) command in **fdisk**)?_____

Q5. The meaning of Start and End?_____

Q6. The meaning of the *?_____