

## SECTION 4 Manage the Linux File System

In this section of the workbook, you learn how to do the following:

- “Configure Partitions on Your Hard Drive” on 4-2

In this exercise, you practice creating partitions and file systems on them with YaST and **fdisk**. You also use command line tools to create file systems.

- “Manage File Systems from the Command Line” on 4-9

In this exercise you practice to manage file systems from the command line.

- “Create Logical Volumes” on 4-13

In this exercise, you learn how to administer LVM with YaST.

## **Exercise 4-1    *Configure Partitions on Your Hard Drive***

In this exercise, you practice creating partitions and file systems on them using YaST and fdisk. You also use command line tools to create file systems.

In the first part of this exercise, use YaST to create the following partitions and file systems:

- An extended partition using the remaining disk space.
- One logical partition with a size of 500 MB, an ext2 file system, and a mountpoint of /apps.
- One logical partition with a size of 1 GB, a Reiser file system, and a mountpoint of /srv.

In the second part of this exercise, use fdisk to create the following partitions:

- One partition of the partition type Win95/FAT32 with a size of 500 MB
- Two partitions with the partition type Linux and a size of 1 GB and 2 GB, respectively

After this exercise, there should still be space left on the hard disk to create additional partitions. If disk space is limited, use smaller values for the partitions than those given here to make sure there is empty space for later exercises.

In the third part, you create file systems on the partitions you created in Part II, using the applicable options for mkfs:

- Create a FAT32 file system on /dev/hda7 (or /dev/sda7, depending on your hardware).
- Create an ext2 file system on /dev/hda8 (or /dev/sda8, depending on your hardware).
- Create a Reiser file system with a file system size of 625 MB on /dev/hda9 (or /dev/sda9, depending on your hardware).

**Detailed Steps to Complete this Exercise:**

- Part I: Create Partitions and File Systems with YaST
- Part II: Partition Manually with fdisk

**Part I: Create Partitions and File Systems with YaST**

To create partitions and file systems with YaST, do the following:

1. Open a terminal window, **su -** to root (password novell), and enter **yast2 disk**  
A warning message appears.
2. Continue by selecting **Yes**.  
After a few moments the Expert Partitioner dialog appears.
3. If there is no extended partition yet, create an extended partition with YaST:
  - a. Create a new partition by selecting **Create**.
  - b. Make sure **Extended Partition** is selected; then select **OK**.  
A **Create an Extended Partition** dialog appears.
  - c. Make sure that the values in the fields for first and last cylinder of the extended partition comprise the whole remaining disk space. Then select **OK**.  
You are returned to the Expert Partitioner dialog, with the new partition listed as a new entry for your hard disk.
4. Create a new ext2 partition with YaST:
  - a. Create a new partition by selecting **Create**.
  - b. Configure a new logical partition by entering or selecting the following:
    - File system: **Ext2**
    - End (cylinder): **+500M**
    - Mount Point: **/apps**
  - c. When you finish, confirm the partition definition by selecting **OK**.

You are returned to the Expert Partitioner dialog where the new partition is added to the list.

The asterisk (\*) after the mount point signifies that the file system is not mounted right now (see explanation in the help text to the left).

5. Create a partition for the directory /srv:
  - a. Select the same hard disk as before; then select **Create**.
  - b. Configure a new logical partition by entering or selecting the following:
    - File system: **Reiser**
    - End (cylinder): **+1G**
    - Mount Point: **/srv**
  - c. Add the /srv partition by selecting **OK**.

You are returned to the Expert Partitioner dialog where the new partition is added to the list.

6. Add the new partitions to the hard drive by selecting **Apply**.

A dialog appears asking if you really want to execute the changes.

Continue by selecting **Finish**.

This executes the changes and closes the Expert Partitioner dialog. If you select **Apply** instead of **Finish**, the changes are executed and you are returned to the Expert Partitioner dialog.

7. Verify creation of the new partition for /apps:
  - a. In the terminal window where you are logged in as root, verify that the new partition is mounted by entering **mount**.

You should see the following line:

```
/dev/hda5 on /apps type ext2 (rw)
```

- b. Verify that the appropriate entry was added to the /etc/fstab for the new partition by entering the following:

**cat /etc/fstab**

You should see the following:

```
/dev/hda5 /apps ext2 acl,user_xattr 1 2
```

This entry makes sure that when the system boots, the new file system is mounted.

8. The current content of the directory `/srv` is no longer visible, as it is used as mount point for the partition `/dev/hda6` (or `/dev/sda6`, depending on your hardware). To copy the content to the new partition, do the following:
  - a. Unmount `/dev/hda6` (or `/dev/sda6`, depending on your hardware) by entering:  
**umount /srv**
  - b. Mount the partition `/dev/hda6` (or `/dev/sda6`, depending on your hardware) under `/mnt` by entering:  
**mount /dev/hda6 /mnt**
  - c. Move the content of `/srv` to `/mnt` :  
**mv /srv/\* /mnt**
  - d. Unmount `/mnt` and mount `/dev/hda6` (or `/dev/sda6`, depending on your hardware) again, using the entries in `/etc/fstab`:  
**umount /mnt**  
**mount -a**
  - e. Verify that the files you moved are available again under `/srv` by entering  
**ls /srv/**

## Part II: Partition Manually with fdisk

To partition manually with `fdisk`, do the following:

1. From the command line, start the utility `fdisk` on the first IDE hard disk on your server by entering **fdisk /dev/hda** (if your computer uses SATA or SCSI disks, enter **fdisk /dev/sda**). Depending on the number of cylinders on your disk a message is displayed that the number of cylinders is above 1024, which might cause problems under certain circumstances.
2. View the current partition table in `fdisk` by entering **p**.

Notice that there are 5 partitions (hda1, hda2, hda3, hda5 and hda6).

3. Create a new 500MB Win95 FAT32 logical partition as the next partition in the extended partition by doing the following:
  - a. Create a new partition by entering **n**.
  - b. Enter **l** (lower case L) for logical.
  - c. Accept the default first cylinder by pressing **Enter**.
  - d. Indicate the partition size by entering **+500M**.
  - e. Change the partition type to Win95 FAT32 by entering **t** (for type, the highest partition number from the range offered, and then **b** (for Win95/FAT32).
  - f. Verify the new partition configuration by entering **p**.  
Notice that a new hda7 partition has been added to the table.
4. Create 2 more logical partitions with the partition type of Linux (the default) by doing the following:
  - a. Create a new partition by entering **n**; enter **l** (lower case L); then accept the default first cylinder by pressing **Enter**.
  - b. Indicate the partition size by entering **+1G**.
  - c. Create a new partition by entering **n**; enter **l** (lower case L); then accept the default first cylinder by pressing **Enter**.
  - d. Indicate the partition size by entering **+2G**.
  - e. Verify the new partition configuration by entering **p**.  
Notice that 2 new partitions have been added to the partition table.
5. Write the new partition table to your hard drive and exit fdisk by entering **w**.
6. View the current partition table used by the kernel by entering:  
**cat /proc/partitions**
7. To be able to access the new partitions, the kernel has to update its partition table stored in memory. Do one of the following:
  - Reboot the system by entering **reboot**.

- Have the kernel update its partition table by entering **partprobe**.
8. View again the partition table used by the kernel by entering:  
**cat /proc/partitions**

### Part III: Manage File Systems from the Command Line

To manage file systems from the command line, do the following:

1. From the GNOME desktop, open a terminal window; then **su -** to root (password **novell**).
2. Create the following file systems (depending on your hardware use **sda** instead of **hda** in the following steps):

- a. Create a new FAT32 file system on **/dev/hda7** and give it the label “**data1**” by entering the following (make sure you don’t have a typo when specifying the device in the command; there won’t be a warning message, the command is executed immediately):

```
mkfs.msdos -n data1 /dev/hda7
```

A message such as **mkfs.msdos 2.11 (12 Mar 2005)** confirms the file system creation.

- b. Create a new ext2 file system on **/dev/hda8** with verbose output by entering the following (make sure you don’t have a typo when specifying the device in the command; there won’t be a warning message, the command is executed immediately):

```
mkfs -t ext2 -v /dev/hda8
```

Notice that by adding the option **-v**, you received extensive information about the new file system.

- c. Create a new Reiser file system on **/dev/hda9** that is only 625 MB by entering the following:

```
mkreiserfs /dev/hda9 160000
```

A warning message appears indicating that all data will be lost on **/dev/hda9**.

- d. Continue by entering **y**.
3. Create the directories **data1**, **data2** and **data3** under **/export/** by using **mkdir -p /export/data{1,2,3}**.
4. Verify that the directories were created by entering **ls -l /export**.
5. As root, using an editor of your choice, for instance **vi** in a terminal window, add entries to the file **/etc/fstab** for the new file systems:
  - a. Open the file **/etc/fstab** in an editor.
  - b. At the end of the file **fstab**, add the following entries:

```
/dev/hda7 /export/data1 vfat defaults 1 2  
/dev/hda8 /export/data2 ext2 defaults 1 2  
/dev/hda9 /export/data3 reiserfs defaults 1 2
```

You must include an empty line at the end of the file, otherwise the mount command cannot read the file.  
These entries make sure that the **hda7**, **hda8**, and **hda9** partitions are mounted when starting or rebooting the system.
  - c. When you finish, save **/etc/fstab** (when using **vi**, press **Esc**, then enter **:wq**).
6. From the terminal window, mount all of the new file systems and re-read the **/etc/fstab** file by entering **mount -a**.
7. View the information about the mounted file systems by entering the following 3 commands:

```
mount  
cat /proc/mounts
```

**(End of Exercise)**

## **Exercise 4-2     Manage File Systems from the Command Line**

In this exercise you practice to manage file systems from the command line.

In the previous exercise, you created various partitions and file systems. If you used `/dev/sda` previously, replace `/dev/hda` by `/dev/sda` in this exercise as well.

In the first part of this exercise, you run `e2fsck` on the `ext2` file system you created on `/dev/hda5`, which is mounted on `/apps`.

In the second part of the exercise, you convert the partition `/dev/hda8` to an `ext3` file system by adding a journal; also add the label `/export/data2` to it.

Then resize the Reiser file system on `/dev/hda9` to use the entire partition and not just 625 MB.

### **Detailed Steps to Complete this Exercise:**

- Part I: Run `e2fsck`
- Part II: Customize the File Systems

#### **Part I: Run `e2fsck`**

To run `e2fsck`, do the following:

1. Unmount the file system on `/dev/hda5` (or `/dev/sda5`, depending on your hardware) by entering **umount /apps**.
2. Verify that the file system is no longer mounted by entering **mount**.

The `/dev/hda5` partition is not displayed.

3. Start a file system check on `hda5` running in verbose mode with an automatic response of yes to prompts by entering the following:

```
e2fsck -f -y -v /dev/hda5
```

4. Mount the /apps file system again by entering **mount /apps**.
5. Verify that the file system is mounted by entering **mount**.

## Part II: Customize the File Systems

To customize the file systems, do the following:

1. Modify the partition /dev/hda8:
  - a. From the terminal window, unmount /dev/hda8 and view details about the ext2 file system on /dev/hda8 by entering the following:  
**umount /dev/hda8 ; dumpe2fs /dev/hda8 | more**  
Notice the block size and the file system state.
  - b. Give the ext2 file system the volume name /export/data2 while the file system is unmounted by entering the following:  
**tune2fs -L /export/data2 /dev/hda8**  
Naming a file system can be useful in system rescue situations when the /etc/fstab is not available. It is common practice to use this naming convention.
  - c. Verify that the file system now has a volume name by entering **dumpe2fs /dev/hda8 | less**.
  - d. Add a journal to the file system (making it an ext3 file system) by entering **tune2fs -j /dev/hda8**.
  - e. Verify that the file system now contains a journal by entering **dumpe2fs /dev/hda8 | less**.
  - f. Mount /dev/hda8 again by entering **mount /dev/hda8**.
  - g. View information about the mounted file systems by entering **mount**.  
Notice that the file system is still mounted as an ext2 file system.
  - h. Unmount the partition /dev/hda8 again by entering **umount /dev/hda8**.

- i. Verify that the file system state is clean by entering **dumpe2fs /dev/hda8 | less**.
  - j. Edit the file `/etc/fstab` to change the file system type from `ext2` to `ext3`, as in the following:  

```
/dev/hda8 /export/data2 ext3 defaults 1 2
```

Save the file.
  - k. From the command line, re-read `/etc/fstab` and mount the partition as an `ext3` file system by entering **mount -a**.
  - l. Verify the change by entering **mount**.
  - m. Unmount the partition `/dev/hda8` again by entering **umount /export/data2**.
  - n. Mount the partition as an `ext2` file system manually by entering the following:  
**mount -t ext2 /dev/hda8 /export/data2**
  - o. Verify that the file system is mounted without a journal (as an `ext2` file system) by entering **mount**.  
As you can see, `ext3` is backward compatible with `ext2`.
  - p. Remount `/dev/hda8` as an `ext3` file system and verify the change by entering the following 3 commands:  
**umount /export/data2**  
**mount -a**  
**mount**
2. Modify the partition `/dev/hda9`:
    - a. View the size of the partition `/dev/hda9` by entering **df -h**.
    - b. Unmount `dev/hda9`  
**umount /export/data3**.
    - c. While the partition is unmounted, add a label to the file system of `/export/ data3` by entering the following:  
**reiserfstune -l /export/data3 /dev/hda9**
    - d. Resize the partition to consume the entire partition by entering **resize\_reiserfs /dev/hda9**.

When no size is specified, the file system is resized to use all available space on the partition. Increasing the size of the ReiserFS is also possible when the file system is mounted.

- e. Remount the partition by entering **mount -a**.
- f. View the size of the partition by entering **df -h**.  
The size is no longer 625 MB, but should be 1 GB or more depending on the size of your partition.
- g. Unmount the partition to run a file system check on it by entering **umount /export/data3**.
- h. Run a check on the file system on /dev/hda9 by entering the following:  
**reiserfsck -y /dev/hda9**
- i. Remount all file systems by entering **mount -a**.

***(End of Exercise)***

### **Exercise 4-3    Create Logical Volumes**

In this exercise, you learn how to administer LVM with YaST.

In the first part of this exercise, use YaST to create two physical volumes (PV) with a size of 1 GB each. Add them to a volume group (VG) named projects. Within the volume group, add two logical volumes named pilot (750MB) and production (750MB), to be mounted under /projects/pilot and /projects/production, respectively.

In the second part of the exercise, increase the size of the logical volume production to the maximum space available within the volume group.

#### **Detailed Steps to Complete this Exercise:**

- Part I: Create LVM Physical Volumes, a Volume Group, and Logical Volumes
- Part II: Resize an LVM Volume

#### **Part I: Create LVM Physical Volumes, a Volume Group, and Logical Volumes**

To create LVM Physical Volumes, a Volume Group, and Logical Volumes, do the following:

1. Start **YaST**, enter the root password (**novell**), and select **System > Partitioner**.  
Acknowledge the warning message by selecting **Yes**.  
The Expert Partitioner dialog appears.
2. Create a new LVM partition by doing the following:
  - a. Select **Create**.
  - b. Select **Do not format**; then select or enter the following:
    - File system ID: **0x8E Linux LVM**

- End (cylinder): **+1GB**
  - c. Save the partition definition by selecting **OK**.
- 3. Create another 1GB LVM partition by repeating step 2.  
You should now have two 1GB LVM partitions.
- 4. Select the **LVM** button and enter the following in the **Create a Volume Group** dialog:
  - Volume Group Name: **projects**
  - Physical Extent Size: **4M**
- 5. Continue by selecting **OK**.
- 6. Add each Linux LVM physical volume to the volume group **projects** by selecting each physical volume (such as **/dev/hda10**) and then selecting **Add Volume**. Then select **Next**.

The **Logical Volume Manager: Logical Volumes** dialog appears.

- 7. Add a logical volume **pilot** within the **projects** volume group:
  - a. Select **Add**.  
A **Create Logical Volume** dialog appears.
  - b. Enter or select the following:
    - Format (file system): **Reiser**
    - Logical volume name: **pilot**
    - Size: **750 MB**
    - Mount Point: **/projects/pilot**
  - c. Save the logical volume definition by selecting **OK**.
- 8. Add a logical volume **production** within the **projects** volume group:
  - a. Select **Add**.  
A **Create Logical Volume dialog** appears.
  - b. Enter or select the following:
    - Format (File system): **Reiser**
    - Logical volume name: **production**

- Size: **750 MB**
  - Mount Point: **/projects/production**
- c. Save the logical volume definition by selecting **OK**.
9. Save the changes by selecting **Next**.  
You are returned to the Expert Partitioner.
10. In the Expert Partitioner select **Apply**.  
A message appears; accept the changes by selecting **Finish**.
11. From a terminal window, **su -** to root (password **novell**).
12. View the new LVM file systems by entering the following:  
**df -h**  
Notice the size of these new file systems.
13. View the device names and mount locations by entering  
**cat /etc/fstab**.

## Part II: Resize an LVM Volume

To resize a LVM Volume, do the following:

1. From the YaST Control Center, select **System > LVM**.  
The LVM dialog appears.
2. From the **Logical volumes** list select **/dev/projects/production**; then select **Edit**.  
The Edit Logical Volume dialog appears.  
Notice the volume size.
3. Select the **max** button.  
Notice that the size changes to the maximum space available.
4. Continue by selecting **OK**.
5. Save the changes by selecting **Finish**; then confirm the notification by selecting **OK**.

6. From the terminal window, view the new size of production by entering  
**df -h.**
7. Close all open windows.

***(End of Exercise)***