

# Лабораторная работа №3

## Управление файловой системой ОС Red Hat Enterprise Linux 4.

**Цель работы:** Получить базовые знания управления файловой системой ОС RHEL 4 и практические навыки монтирования, преобразования, восстановления различных типов файловых систем и устранения неполадок.

### Краткие теоретические сведения:

#### 1. Особенности файловой системы Linux

Работа с файлами и устройствами хранения данных в ОС Linux отличается от привычной ОС Windows. Есть файлы и иерархическая структура каталогов, но в системе Linux нет букв, обозначающих диски. В Linux есть только одна файловая структура. Она начинается от корня (/), и все локальные файловые системы, все локальные устройства и все удаленные системы представлены в этой структуре подкаталогами.

```
/
|-- bin
|-- boot
|-- dev
|-- etc
|-- mnt
|-- root
|-- sbin
|-- tmp
|-- usr
| |-- X11R6
| | |-- bin
| | |-- include
| | |-- lib
| | |-- man
| | `-- share
|-- bin
```

Когда Linux загружается впервые, он строит эту файловую структуру на основе информации из [/etc/fstab](#) файла. Там, где Windows присваивает разделам жесткого диска и другим устройствам хранения данных буквы, Linux определяет им каталоги в корневой файловой структуре. Структуру иерархии можно полностью настраивать и менять на лету.

Слово, обозначающее добавление устройства к файловой системе – монтирование ([mounting](#)). Linux автоматически монтирует корневую (/) файловую систему. Отдельно может присутствовать файловая система [/boot](#), в которой расположены загрузочные файлы ядра. Linux также монтирует некоторые особые файловые системы. Область свопинга не показывается как часть файловой системы, но управляется ядром.

Другие файловые системы, такие как сменный носитель или удаленные файловые системы необходимо монтировать вручную. Монтируя файловую систему, вы должны знать правильный путь, чтобы сослаться на него из Linux и иметь пустой каталог, чтобы использовать его как [точку монтирования \(mount point\)](#). Для сменных носителей Linux, как правило, создает точки монтирования во время инсталляции. В Red Hat Linux устройство чтения компакт-дисков предполагается монтировать к каталогу [/media/cdrecorder](#). Это означает, что когда вы вставляете CD в устройство CDRом, вы вводите команду:

```
mount /dev/cdrom /media/cdrecorder (если не прописан в /etc/fstab)
```

CD добавляется в файловую систему и устройство чтения CD блокируется так, что случайно оно не откроется. Для доступа к содержимому компакт-диска просто используйте каталог [/media/cdrecorder](#). Когда вы закончите работу с CD, можно будет удалить его из файловой системы при помощи команды:

```
umount /media/cdrecorder
```

Каталог будет очищен, а устройство чтения CD разблокировано. Точно также следует поступать и с остальными сменными носителями информации, такими как дискета:

```
/media/floppy
```

Если выполнить [mount](#) без аргументов, то будут показаны файловые системы, подмонтированные в настоящий момент.

## 2. Стандартные имена устройств в Linux

Linux не видит разницы между файлами и устройствами, все файлы устройств находятся в каталоге `/dev`.

Рассмотрим стандартные IDE-диски. Всего в компьютере с интерфейсом IDE может быть установлено 4 устройства - Primary Master, Primary Slave, Secondary Master, Secondary Slave. В Linux это соответствует именам `/dev/hda`, `/dev/hdb`, `/dev/hdc`, `/dev/hdd`.

Разделы на дисках нумеруются: `/dev/hda1` - первый раздел на Primary Master.

Всего может быть 4 первичных раздела.

Например, у вас имеется 3 раздела - загрузочный для Windows, Linux Native, Linux Swap.

В Linux это будет представляться так:

`/dev/hda1` - Windows

`/dev/hda2` - Linux Native

`/dev/hda3` - Linux Swap

В Windows - только C:

## 3. Монтирование Windows файловой системы

Когда вы устанавливаете Linux, вы создаете корневую файловую систему. Под корневой файловой системой будем понимать раздел, на который вы установили Linux.

Чтобы работать с информацией, расположенной на разделах Windows, вам нужно их примонтировать, т.е. подключить к файловой системе. **Точка монтирования** - это всего лишь каталог, через который будет происходить обращение к другим файловым системам. Обычно другие разделы подключаются к подкаталогам каталога `/mnt (/media)`. Например, если Windows-раздел примонтирован к подкаталогу `win` каталога `/mnt`, то вы можете промотреть его содержимое командой

```
ls /mnt/win
```

## 4. Устранение неполадок.

Если в смонтированном Windows-разделе вместо русских имен файлов сплошные «???????» необходимо в опциях монтирования указать используемую кодировку для файловой системы `vfat`:

- `codepage=866,iocharset=koi8-r`
- `codepage=866,iocharset=cp1251`
- `iocharset=utf8`

```
mount -t vfat iocharset=utf8 /dev/hda1 /mnt/win
```

Для NTFS вместо `iocharset=%имя_кодировки%` следует использовать `nls=%имя_кодировки%`. Файловая система NTFS, используемая в Windows линейки NT (включая Windows 2000, Windows XP и более старшие версии), является полностью закрытой разработкой. По этой причине создание NTFS-драйвера сопряжено с целым рядом проблем. Однако в большинстве популярных дистрибутивов у вас будет возможность использовать NTFS-разделы без каких-либо сложностей. Монтируются они так же, как и в случае с FAT32, только вместо `vfat` нужно указать `ntfs`.

Для пользователей Red Hat Linux создатели этого дистрибутива не включают в его комплект драйвер NTFS, поэтому вам потребуется самим скачать его и установить.

## 5. Автоматизация процесса монтирования.

Связь между устройством и его точкой монтирования настраивается в файле `/etc/fstab`. Его можно редактировать вручную или оставить программе администрирования. Вот пример `/etc/fstab`:

<code>/dev/hda5</code>	<code>/</code>	<code>ext3</code>	<code>defaults</code>	<code>1 1</code>
<code>/dev/hda2</code>	<code>/boot</code>	<code>ext3</code>	<code>exec,dev,uid,rw</code>	<code>1 2</code>
<code>/dev/hda6</code>	<code>swap</code>	<code>swap</code>	<code>defaults</code>	<code>0 0</code>
<code>/dev/scdo</code>	<code>/mnt/cdrom</code>	<code>auto</code>	<code>ro,noauto,exec</code>	<code>0 0</code>
<code>none</code>	<code>/dev/pts</code>	<code>devpts</code>	<code>id=5,mode=620</code>	<code>0 0</code>
<code>none</code>	<code>/proc</code>	<code>proc</code>	<code>defaults</code>	<code>0 0</code>
<code>none</code>	<code>/dev/shm</code>	<code>tmpfs</code>	<code>defaults</code>	<code>0 0</code>

Каждая строка представляет монтируемую файловую систему. Первая колонка определяет устройство, которое должно быть подмонтировано. Во второй колонке находится точка монтирования – местоположение данного устройства в файловой системе. Третья колонка определяет тип файловой системы. В четвертой содержатся опции, показывающие как эта файловая система будет обрабатываться. В последней колонке написаны флаги, относящиеся к файловой системе. Первая цифра, 0 или 1, показывает, должна ли система

Лабораторная работа №3. Файловая система Red Hat Enterprise Linux 4.

копироваться при помощи команды `dump` (это нужно для системных резервных копий). Вторая цифра может быть 0, 1 или 2, она показывает порядок, в котором файловая система должна быть проверена при загрузке. 0 – не должна проверяться вовсе. 1 – должна проверяться первой и использоваться как корневая (/). Для всех остальных систем ставится 2.

В файле `fstab`, приведенном выше, корневая файловая система расположена на первом IDE жестком диске, в пятом разделе, первый логический диск в расширенном разделе. Файловая система `/boot`, где находятся файлы запуска ядра, расположена на первом IDE жестком диске, во втором первичном разделе. Пространство свопинга расположено на первом IDE жестком диске, в шестом разделе, втором логическом диске в расширенном разделе. Другие перечисленные файловые системы показывают свое устройство, как `"none"`. Мы осветим это вкратце. А сейчас давайте сконцентрируемся на физических дисках.

Опции в четвертой колонке будут меняться в зависимости от типа файловой системы. В приведенном выше примере / и `/boot` подмонтированы с опциями `"default"`. Это означает, что они монтируются автоматически, доступны для чтения и записи с асинхронным I/O (вводом/выводом). Только `root` может монтировать и отмонтировать устройства, но пользователи могут выполнять бинарники. Для `/mnt/cdrom`, однако, опции другие. Он не монтируется автоматически и будет открыт только для чтения. Пользователи смогут выполнять скрипты и программы в этой файловой системе.

Опция	Значение
<code>auto</code>	Файловая система может монтироваться командой <code>mount</code> с опцией <code>-a</code>
<code>defaults</code>	Использовать набор опций, задаваемый по умолчанию: <code>rw, suid, dev, exec, auto, nouser, async</code>
<code>noauto</code>	Файловая система может монтироваться только явно. Опция <code>-a</code> не приведет к монтированию файловой системы
<code>exec</code>	Разрешает выполнение двоичных файлов
<code>remount</code>	Позволяет перемонтировать уже смонтированную файловую систему. Обычно используется для изменения опций монтирования файловой системы, особенно для того, чтобы расширить права доступа (вместо прав только на чтение установить права на чтение/запись)
<code>ro</code>	Монтирует файловую систему только на чтение
<code>rw</code>	Монтирует файловую систему для чтения и записи
<code>user</code>	Позволяет непривилегированному пользователю монтировать файловую систему. Для таких пользователей монтирование всегда выполняется с опциями <code>noexec, nosuid, nodev</code>
<code>nodev</code>	Файлы байт-ориентированных и блок-ориентированных устройств в файловой системе не интерпретируются как специальные файлы
<code>nouser</code>	Запрещает непривилегированному пользователю монтировать файловую систему

Разделы в Linux работают в общем так же как и в Windows. Консольная команда `fdisk` используется для создания и управления разделами. При выполнении этой команды вы должны указать ей устройство. Чтобы посмотреть доступные устройства, воспользуйтесь командой `fdisk -l`.

Запустите `fdisk` с устройством и вы получите короткое приглашение. Ввод `"m"` даст вам меню команд. Текущая таблица разделов отобразится по нажатию `"p"`. Вы можете создавать, удалять и изменять тип существующих разделов. `"l"` покажет вам полный список доступных типов разделов. Свои изменения в таблицу разделов вносите с помощью `"w"`, закрывайте программу или выходите без сохранения используя `"q"`. Некоторые изменения вступят в силу незамедлительно. Некоторые могут потребовать перезагрузки системы.

## 6. Использование файловых ресурсов удаленных хостов

Многие файловые ресурсы подключаются через сетевые системы. Наиболее распространенными являются NFS (Network File System – сетевая файловая система) и SMB ресурсы. Для доступа к этим файловым системам необходимо знать имя хоста и имя «расшаренного» ресурса. Для определения того, какие файловые системы экспортируются удаленной системой, используются следующие команды:

- для NFS:  
`showmount -e имя_удаленной_системы`
- для SMB:  
`smbclient -L имя_удаленной_системы -N`

Когда имя хоста и имя «расшаренного» ресурса известны, то используют следующие команды для подключения сетевых файловых систем к дереву локальной файловой системы.

- для NFS:  
`mount имя_удаленного_хоста:/shared/dir /mnt/имя_ntf_ресурса`

Лабораторная работа №3. Файловая система Red Hat Enterprise Linux 4.

- для SMB:

```
mount //имя_удаленного_хоста/share /mnt/имя_samba_ресурса
```

## 7. Типы файловых систем.

Linux может обслуживать любой тип файловой системы, о котором знает ядро. Изрядное их количество компилируется по умолчанию, а новые можно добавлять. Наиболее интересные из них:

<a href="#">ext2</a>	Стандартная файловая система Linux
<a href="#">ext3</a>	Стандартная файловая система Linux с добавленным протоколированием
<a href="#">vfat</a>	Файловая система Microsoft Fat32
<a href="#">jfs</a>	Журналируемая файловая система IBM
<a href="#">reiserfs</a>	Еще одна популярная журналируемая файловая система

Журналируемые файловые системы помогают уберечь данные при непредвиденных выключениях системы. Если том выключается неотмонтированным, то могут остаться незаконченные процессы и файлы в промежуточном состоянии. При использовании обычной файловой системы потребуется полная проверка, что может занять немало времени для больших объемов. Журналируемая файловая система хранит корректную запись каждого ввода информации на диск за период времени, например пять секунд. Когда том не отмонтирован аккуратно, файловая система просто откатывается назад к последнему нормальному состоянию, которое она запомнила. И то, что могло бы восстанавливаться минут двадцать теперь поднимается за несколько секунд!

## 8. Форматирование разделов.

Созданные разделы форматируются подходящей версией команды [mkfs](#). У файловых систем есть свои собственные версии [mkfs](#), например, [mkfs.ext2](#) или [mkfs.ext3](#). Эти вспомогательные скрипты позволят создать файловую систему, просто указав на раздел. Вот несколько примеров:

- [mkfs.ext2 /dev/hda3](#)
- [mkfs.ext3 /dev/sdb1](#)

Существуют различные продвинутые параметры, чтобы влиять на то, как будет форматироваться раздел, но в общем случае вполне подходят значения по умолчанию. Отформатированный раздел может быть смонтирован в корень (/). Файловую систему следует отмонтировать перед переформатированием.

### Ход работы:

1. Ознакомьтесь с содержимым файла [/etc/fstab](#)
2. Примонтируйте различные сменные носители:
  - [Дискету](#)
  - [CD-ROM](#)
  - [Флеш накопитель](#)
3. Изучите также другие программы файловой системы

Есть несколько программных средств для просмотра состояния дисков и файловых систем.

[df](#)

[df](#) означает "[disk free](#)". Эта команда сообщает, сколько места на диске используется и доступно в подмонтированной файловой системе. Полезные ключи:

### Проверка дискового пространства

[df -l](#)

Ограничивает список до локальных файловых систем; по умолчанию показываются еще и удаленные

[du](#)

[du](#) означает "[disk usage](#)" (использование диска). Команда выводит, сколько дискового пространства используется определенными файлами и для каждого подкаталога (если аргумент – каталог) Полезные ключи:

### Проверка использования диска

[du -a](#)

Показывает счетчики для всех файлов, не только каталогов

[du -h](#)

При отображении размера файла использует дружелюбный для человека вывод (в Mb, Kb), а не отображать его в

Лабораторная работа №3. Файловая система Red Hat Enterprise Linux 4.

**Проектирование и администрирование компьютерных сетей. Олейник М.А.**

байтах.