

Лабораторная работа №5

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

Копирование файлов и каталогов. Команда cp

С самого начала копируем файл `/usr/include/sys/io.h` в домашний каталог:

```
yvegorova@dk6n62 ~ $ cp /usr/include/sys/io.h equipment
```

Figure 1: Копирование файла в домашний каталог.

Создаем в домашнем каталоге файл `abc1` и копируем его в каталог `~/ski.plases`, называем его `equiplist2`:

```
yvegorova@dk6n62 ~ $ cp abc1 ski.plases  
yvegorova@dk6n62 ~ $ mv ski.plases/abc1 ski.plases/equiplist2
```

Figure 2: Копирование файла в каталог и изменение названия.

Копируем файл `~/feathers` в файл `~/file.old`:

```
yvegorova@dk6n62 ~ $ cp feathers file.old
```

Figure 3: Копирование файла в другой файл.

Перемещение файлов и каталогов. Команда mv

Перемещаем файл equipment в каталог ~/ski.places:

```
yvegorova@dk6n62 ~ $ mv equipment ski.places
```

Figure 5: Перемещение файла в каталог.

Перемещаем файлы ~/ski.places/equiplist и equiplist2 в каталог ~/ski.places/equipment:

```
yvegorova@dk6n62 ~ $ mv ski.places/equiplist ski.places/equipment  
yvegorova@dk6n62 ~ $ mv ski.places/equiplist2 ski.places/equipment
```

Figure 6: Перемещение файлов в каталог.

Перемещаем файл ~/file.old в каталог ~/play:

```
yvegorova@dk6n62 ~ $ mv file.old play
```

Figure 7: Перемещение файла в каталог.

Изменение имени файла и каталога. Команда mv

Переименовываем файл ~/ski.plases/equipment в ~/ski.plases/equiplist:

```
yvegorova@dk6n62 ~ $ cd  
yvegorova@dk6n62 ~ $ mv ski.plases/equipment ski.plases/equiplist  
yvegorova@dk6n62 ~ $
```

Figure 8: Переименовывание файла.

Перемещаем каталог ~/fun в каталог ~/play и называем его games:

```
yvegorova@dk6n62 ~ $ mv fun play  
yvegorova@dk6n62 ~ $ mv play/fun play/games  
yvegorova@dk6n62 ~ $ chmod u-r feathers  
yvegorova@dk6n62 ~ $ chmod u+r feathers  
yvegorova@dk6n62 ~ $ chmod u-r play  
yvegorova@dk6n62 ~ $ cd play  
yvegorova@dk6n62 ~/play $ cd  
yvegorova@dk6n62 ~ $ chmod u-x play  
yvegorova@dk6n62 ~ $ cd play  
yvegorova@dk6n62 ~/play $ cd  
yvegorova@dk6n62 ~ $ chmod u+x play  
yvegorova@dk6n62 ~ $ chmod u-x play  
yvegorova@dk6n62 ~ $ chmod u+r play
```

Figure 9: Переименовывание файла.

Определяем опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет.

Лишаем владельца файла `~/feathers` права на чтение.

По команде man читаем описание команд mount, mkfs, fsck, kill:

```

MOUNT(8)                                     System Administration                                MOUNT(8)

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-t type] [-o options] device mountpoint

    mount [-t type] [-o options] [-W options] device mountpoint

    mount [-t type] [-o options] [-W options] device mountpoint

    mount --bind|--rbind|--move source1:target1[:source2:target2[:...]] mountpoint

DESCRIPTION
    All files accessible in a Unix system are arranged in one big tree, the file hierarchy, rooted at /. These files can be spread out over several devices. The mount command serves to
    attach the filesystem found on some device to the big file tree. Conversely, the umount(8) command will detach it again. The filesystem is used to control how data is stored on the
    device or accessed in a virtual way by network or other services.

    The standard form of the mount command is:

        mount -t type device dir

    This tells the kernel to attach the filesystem found on device (which is of type type) at the directory dir. The option -t type is optional. The mount command is usually able to
    detect a filesystem. The root permissions are necessary to mount a filesystem by default. See section "Non-interactive mounts" below for more details. The previous contents (if any) and
    owner and mode of dir become irrelevant, and as long as this filesystem remains mounted, the pathname dir refers to the root of the filesystem on device.

    If only the directory or the device is given, for example:

        mount /dir

    then mount looks for a mountpoint (and if not found then for a device) in the /etc/fstab file. It's possible to use the --target or --source actions to avoid ambiguous interpretation
    of the given argument. For example:

        mount --target /mountpoint

    The same filesystem may be mounted more than once, and in some cases (e.g., network filesystems) the same filesystem may be mounted on the same mountpoint multiple times. The mount
    command does not implement any policy to control this behavior. All behavior is controlled by the kernel and it is usually specific to the filesystem driver. The exception is "all",
    in this case already mounted filesystems are ignored (see --all below for more details).

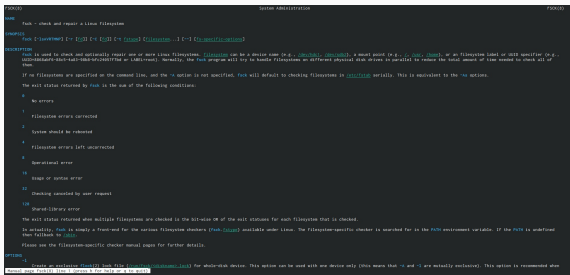
    Listing the mounts
    The listing mode is maintained for backward compatibility only.

    For more robust and customizable output use findmnt(8), especially in your scripts. Note that control characters in the mountpoint name are realized with '\'.

    (mount) does mounting like a daemon & for doing so it has root

```

Figure 10: Описание команды mount.



Команда man

```

# Create a file named file1
$ echo "Hello World" > file1

# Build a Linux file system
$ mkfs -t ext4 /dev/sda1

# Mount the file system
$ mount /dev/sda1 /mnt

# Verify the file system
$ ls -l /mnt
total 4
-rw-r--r-- 1 root root 12 B Jan 1 12:34 file1

# Unmount the file system
$ umount /mnt

# Create a file named file2
$ echo "Hello World" > file2

# Build a Linux file system
$ mkfs -t ext4 /dev/sda2

# Mount the file system
$ mount /dev/sda2 /mnt2

# Verify the file system
$ ls -l /mnt2
total 4
-rw-r--r-- 1 root root 12 B Jan 1 12:34 file2

# Unmount the file system
$ umount /mnt2

```

Figure 12: Описание команды kill.

```
kill(1)                                user Commands                                kill(1)

NAME
    kill - send a signal to a process.

SYNOPSIS
    kill [action] pid...

DESCRIPTION
    The default signal for kill is TERM. Use -l or -L to list available signals. Particularly useful signals include KILL, SIGKILL, SIGSTOP, SIGCONT, and SIG. Alternative signals may be specified in three ways: -s, -signal, or -sig.
    Negative PID values may be used to choose which process groups; see the PIDs column in the command output. A PID of -1 is special: it indicates all processes except the kill process itself and init.

OPTIONS
    -s, --signal=signal
        Send signal to every pid listed.

    -signal=signal
    -s, --signal=signal
        Specify the signal to be sent. The signal can be specified by using name or number. The behavior of signals is explained in signal(7) manual page.

    -s, --signal=signal
        Use signalname() rather than kill(2) and the value argument is used to specify an integer to be sent with the signal. If the receiving process has installed a handler for this signal using the SA_SIGINFO flag to sigprocmask(2), then it can obtain this data via the si_value field of the siginfo_t structure.

    -l, --list [table]
        List signal names. This option has optional argument, which will convert signal number to signal name, or other way round.

    -L, --table
        List signal names in a nice table.

NOTES
    Your shell (command line interpreter) may have a built-in kill command. You may need to run the command described here as /usr/bin/kill to avoid the conflict.

EXAMPLES
    kill -9 1
        Kill all processes you can kill.

    kill -1 1
        Translate number 1 into a signal name.

    kill -L
        List the available signal choices in a nice table.

    kill -s SIGSTOP 1
        Send the default signal, SIGTERM, to all those processes.

SEE ALSO
    kill(2), killall(1), nice(1), ps(1), renice(1), signal(7), signalname(3), shell(1)

CONTRIBUTORS
    This command is part of the GNU system. The GNU system is Linux-specific.
    Copyright 1989 by Richard M. Stallman. All rights reserved. See the GNU manual.
```

Figure 13: Описание команды mkfs.