

Отчёт по лабораторной работе №7

Дисциплина: Операционные системы

Верниковская Екатерина Андреевна

Содержание

1	Цель работы	6
2	Задание	7
3	Выполнение лабораторной работы	8
3.1	Копирование файлов и каталогов (пункт 1)	8
3.2	Перемещение и переименование файлов и каталогов (пункт 1) . .	9
3.3	Права доступа (пункт 1)	10
3.4	Анализ файловой системы (пункт 1)	12
3.5	Пункт 2	14
3.6	Пункт 3	15
3.7	Пункт 4	18
3.8	Пункт 5	21
3.9	Ответы на контрольные вопросы	24
4	Выводы	29
5	Список литературы	30

Список иллюстраций

3.1	Создание abc, may, april	8
3.2	Каталог monthly	8
3.3	Файл june	8
3.4	Каталог monthly.00	9
3.5	Копирование в каталог /tmp	9
3.6	April на july	9
3.7	Перемещение файла july	10
3.8	Переименование каталога	10
3.9	Перемещение каталога monthly.01	10
3.10	Переименование каталога monthly.01	10
3.11	Создание may + права на выполнение	11
3.12	Лишение прав на выполнение	11
3.13	Создание каталога monthly	11
3.14	Запрет на чтение	11
3.15	Создание файла abc1	12
3.16	Право записи	12
3.17	Просмотр файловых систем	12
3.18	Просмотр файла /etc/fstab	13
3.19	Использование команды df	13
3.20	Использование команды fsck	13
3.21	Копирование файла io.h	14
3.22	Создание каталога ski.plases	14
3.23	Перемещение файла equipment	14
3.24	Переименование файл equipment	14
3.25	Работа с файлом abc1	15
3.26	Создание каталога equipment	15
3.27	Перемещение файлов equiplist и equiplist2	15
3.28	Работа с каталогом newdir	15
3.29	Создание каталога australia	16
3.30	Изменение прав (1)	16
3.31	Создание каталога play	16
3.32	Изменение прав (2)	16
3.33	Создание файла my_os	17
3.34	Изменение прав (3)	17
3.35	Создание файла feathers	17
3.36	Изменение прав (4)	17
3.37	Просмотр файла passwd	18

3.38 Копирование файла feathers	18
3.39 Перемещение файла file.old	18
3.40 Копирование каталога play	19
3.41 Перемещение и переименование каталога fun	19
3.42 Лишение прав на чтение	19
3.43 Попытка просмотреть файл feathers	19
3.44 Добавление прав на чтение	20
3.45 Лишение прав на выполнение	20
3.46 Попытка перейти в каталог play	20
3.47 Добавление прав на выполнение	21
3.48 Информация о команде mount	21
3.49 Информация о команде fsck	22
3.50 Информация о команде mkfs	22
3.51 Информация о команде kill	23
3.52 Использование man	23

Список таблиц

1 Цель работы

Ознакомиться с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобрести практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

2 Задание

1. Выполнить все примеры, приведённые в первой части описания лабораторной работы.
2. Выполнить задания с командой `chmod`.
3. Выполнить задания с командой `man`.

3 Выполнение лабораторной работы

3.1 Копирование файлов и каталогов (пункт 1)

Создаём файл `abc1` и копируем его с названием `april` и `may` с помощью команд *touch* (создание файла) и *cp* (копирование файла) (рис. 3.1)

```
[eavernikovskaya@eavernikovskaya ~]$ touch abc1
[eavernikovskaya@eavernikovskaya ~]$ ls
abc1      katerok  Видео   Загрузки  Музыка   'Рабочий стол'
git-extended  LICENSE  Документы  Изображения  Общедоступные  Шаблоны
[eavernikovskaya@eavernikovskaya ~]$ cp abc1 april
[eavernikovskaya@eavernikovskaya ~]$ cp abc1 may
[eavernikovskaya@eavernikovskaya ~]$ ls
abc1      git-extended  LICENSE  Видео   Загрузки  Музыка   'Рабочий стол'
april     katerok       may      Документы  Изображения  Общедоступные  Шаблоны
```

Рис. 3.1: Создание `abc`, `may`, `april`

Создаём каталог `monthly` с помощью *mkdir* и копируем в него файлы `april` и `may` (рис. 3.2)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir monthly
[eavernikovskaya@eavernikovskaya ~]$ cp april may monthly
[eavernikovskaya@eavernikovskaya ~]$ ls monthly/
april  may
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.2: Каталог `monthly`

В каталоге `monthly` копируем файл `may` с именем `june` (рис. 3.3)

```
[eavernikovskaya@eavernikovskaya ~]$ cp monthly/may monthly/june
[eavernikovskaya@eavernikovskaya ~]$ ls monthly/
april  june  may
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.3: Файл `june`

Копируем каталог `monthly` в каталог с именем `monthly.00` введя `cp -r` (рис. 3.4)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir monthly.00
[eavernikovskaya@eavernikovskaya ~]$ cp -r monthly monthly.00
[eavernikovskaya@eavernikovskaya ~]$ ls monthly.00/
monthly
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.4: Каталог `monthly.00`

Далее копируем каталог `monthly.00` в каталог `/tmp` (рис. 3.5)

```
[eavernikovskaya@eavernikovskaya ~]$ cp -r monthly.00 /tmp
[eavernikovskaya@eavernikovskaya ~]$ mc

[eavernikovskaya@eavernikovskaya tmp]$ ls
monthly.00
sddm-auth-586fd89c-8cc8-4a9b-99eb-d5fccb68a614
sddm--RTTRTN
systemd-private-f2406086692c4b13bb76af9b840f1934-chronyd.service-3wGaxr
systemd-private-f2406086692c4b13bb76af9b840f1934-dbus-broker.service-FengZn
systemd-private-f2406086692c4b13bb76af9b840f1934-ModemManager.service-LvD3u3
systemd-private-f2406086692c4b13bb76af9b840f1934-polkit.service-eUJWEq
systemd-private-f2406086692c4b13bb76af9b840f1934-rtkit-daemon.service-yY7zeS
systemd-private-f2406086692c4b13bb76af9b840f1934-systemd-hostnamed.service-ovB2JV
systemd-private-f2406086692c4b13bb76af9b840f1934-systemd-logind.service-a0bkT9
systemd-private-f2406086692c4b13bb76af9b840f1934-systemd-oomd.service-nBtZPC
systemd-private-f2406086692c4b13bb76af9b840f1934-systemd-resolved.service-LKMC85
systemd-private-f2406086692c4b13bb76af9b840f1934-upower.service-Q7WmVQ
[eavernikovskaya@eavernikovskaya tmp]$
```

Рис. 3.5: Копирование в каталог `/tmp`

3.2 Перемещение и переименование файлов и каталогов (пункт 1)

В домашнем каталоге меняем название файла `april` на `july` с поомщью `mv` (рис. 3.6)

```
[eavernikovskaya@eavernikovskaya ~]$ mv april july
[eavernikovskaya@eavernikovskaya ~]$ ls
abc1      july      LICENSE  monthly  Видео     Загрузки  Музыка    'Рабочий стол'
git-extended katerok  may      monthly.00  Документы  Изображения  Общедоступные  Шаблоны
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.6: `April` на `july`

Перемещаем файл `july` в каталог `monthly.00` используя команду `mv` (рис. 3.7)

```
[eavernikovskaya@eavernikovskaya ~]$ mv july monthly.00
[eavernikovskaya@eavernikovskaya ~]$ ls monthly.00/
july  monthly
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.7: Перемещение файла july

Переименовываем каталог monthly.00 в monthly.01 (рис. 3.8)

```
[eavernikovskaya@eavernikovskaya ~]$ mv monthly.00 monthly.01
[eavernikovskaya@eavernikovskaya ~]$ ls
abc1      katerok  may      monthly.01  Документы  Изображения  Общедоступные  Шаблоны
git-extended  LICENSE  monthly  Видео      Загрузки   Музыка       'Рабочий стол'
```

Рис. 3.8: Переименование каталога

После создаём каталог с названием reports и перемещаем в него каталог monthly.01 (рис. 3.9)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir reports
[eavernikovskaya@eavernikovskaya ~]$ mv monthly.01 reports
[eavernikovskaya@eavernikovskaya ~]$ ls reports
monthly.01
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.9: Перемещение каталога monthly.01

В каталоге reports переименовываем каталог monthly.01 в monthly (рис. 3.10)

```
[eavernikovskaya@eavernikovskaya ~]$ mv reports/monthly.01 reports/monthly
[eavernikovskaya@eavernikovskaya ~]$ ls reports/
monthly
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.10: Переименование каталога monthly.01

3.3 Права доступа (пункт 1)

Создаём файл `may` с правом выполнения для владельца. Задать права можно с помощью команды `chmod u+x may` (рис. 3.11)

```
[eavernikovskaya@eavernikovskaya ~]$ touch may
[eavernikovskaya@eavernikovskaya ~]$ ls -l may
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:24 may
[eavernikovskaya@eavernikovskaya ~]$ chmod u+x may
[eavernikovskaya@eavernikovskaya ~]$ ls -l may
-rwxr--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:24 may
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.11: Создание may + права на выполнение

После лишаем владельца файла may прав на выполнения введя *chmod u-x may* (рис. 3.12)

```
[eavernikovskaya@eavernikovskaya ~]$ chmod u-x may
[eavernikovskaya@eavernikovskaya ~]$ ls -l may
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:24 may
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.12: Лишение прав на выполнение

Создаём каталог monthly с запретом на чтение для членов группы и всех остальных пользователей. Это можно сделать введя команды *chmod g-r monthly* и *chmod o-r monthly* (рис. 3.13), (рис. 3.14)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir monthly
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:10 abc1
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 132 мар 3 21:29 git-extended
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 мар 12 19:21 katerok
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:24 may
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:35 monthly
```

Рис. 3.13: Создание каталога monthly

```
[eavernikovskaya@eavernikovskaya ~]$ chmod g-r monthly
[eavernikovskaya@eavernikovskaya ~]$ chmod o-r monthly
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:10 abc1
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 132 мар 3 21:29 git-extended
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 мар 12 19:21 katerok
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:24 may
drwx--x--x. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:35 monthly
```

Рис. 3.14: Запрет на чтение

Далее создаём файл `abc1` с правом записи для членов группы `chmod g+w abc1` (рис. 3.15), (рис. 3.16)

```
[eavernikovskaya@eavernikovskaya ~]$ touch abc1
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya  0 мар 18 17:37 abc1
```

Рис. 3.15: Создание файла `abc1`

```
[eavernikovskaya@eavernikovskaya ~]$ chmod g+w abc1
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-rw-r--. 1 eavernikovskaya eavernikovskaya  0 мар 18 17:37 abc1
```

Рис. 3.16: Право записи

3.4 Анализ файловой системы (пункт 1)

Для просмотра используемых в операционной системе файловых систем можно используем командой `mount` без параметров (рис. 3.17)

```
[eavernikovskaya@eavernikovskaya ~]$ mount
/dev/sda3 on / type btrfs (rw,relatime,seclabel,compress=zstd:1,space_cache=v2,subvolid=257,subvol=/root)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=243964,mode=755,inode64)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nsdelegate,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
efivarfs on /sys/firmware/efi/efivars type efivarfs (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=398740k,nr_inodes=819200,mode=755,inode64)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
```

Рис. 3.17: Просмотр файловых систем

Далее с помощью команды `cat /etc/fstab` просматриваем файл `/etc/fstab` чтобы определить смонтированные файловые системы в ОС (рис. 3.18)

```
[eavernikovskaya@eavernikovskaya ~]$ cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Tue Feb 20 11:01:43 2024
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=9b7af885-0e46-4cab-bd10-b82d536598b0 / btrfs subvol=root,compress=zstd:1 0 0
UUID=83ae1bd3-481c-412d-972a-0b0f28647cad /boot ext4 defaults 1 2
UUID=C439-825A /boot/efi vfat umask=0077,shortname=winnt 0 2
UUID=9b7af885-0e46-4cab-bd10-b82d536598b0 /home btrfs subvol=home,compress=zstd:1 0 0
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.18: Просмотр файла /etc/fstab

Потом используем команду *df* для определения объёма свободного пространства на файловой системе (рис. 3.19)

```
[eavernikovskaya@eavernikovskaya ~]$ df
Файловая система 1К-блоков  Использовано  Доступно  Использовано%  Смонтировано в
/dev/sda3          82221056    14014428  67175316         18% /
devtmpfs           4096          0        4096          0% /dev
tmpfs              996848        3412    993436          1% /dev/shm
efivarfs           256          172        80         69% /sys/firmware/efi/efivars
tmpfs              398740        1176    397564          1% /run
tmpfs              996848          4    996844          1% /tmp
/dev/sda3          82221056    14014428  67175316         18% /home
/dev/sda2          996780        245600    682368         27% /boot
/dev/sda1          613160        19492    593668          4% /boot/efi
work              166171216    143078280  23092936         87% /media/sf_work
tmpfs              199368         100    199268          1% /run/user/1000
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.19: Использование команды df

Далее с помощью команды *fsck* проверяем целостность файловых систем (рис. 3.20)

```
[eavernikovskaya@eavernikovskaya ~]$ fsck /dev/sda1
fsck from util-linux 2.39.3
fsck.fat 4.2 (2021-01-31)
open: Permission denied
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.20: Использование команды fsck

3.5 Пункт 2

Копируем файл `/usr/include/sys/io.h` в домашний каталог и называем его `equipment` (рис. 3.21)

```
[eavernikovskaya@eavernikovskaya ~]$ cp /usr/include/sys/io.h equipment
[eavernikovskaya@eavernikovskaya ~]$ ls
abc1      git-extended  LICENSE  monthly  Видео  Загрузки  Музыка  'Рабочий стол'
equipment katerok       may      reports  Документы  Изображения  Общедоступные  Шаблоны
```

Рис. 3.21: Копирование файла `io.h`

В домашнем каталоге создаём директорию `~/ski.places` (рис. 3.22)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir ski.places
[eavernikovskaya@eavernikovskaya ~]$ ls
abc1      git-extended  LICENSE  monthly  ski.places  Документы  Изображения  Общедоступные  Шаблоны
equipment katerok       may      reports  Видео  Загрузки  Музыка  'Рабочий стол'
```

Рис. 3.22: Создание каталога `ski.places`

Перемещаем файл `equipment` в каталог `~/ski.places` (рис. 3.23)

```
[eavernikovskaya@eavernikovskaya ~]$ mv equipment ski.places/
[eavernikovskaya@eavernikovskaya ~]$ ls ski.places/
equipment
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.23: Перемещение файла `equipment`

В каталоге `ski.places` переименовываем файл `equipment` в `equiplist` (рис. 3.24)

```
[eavernikovskaya@eavernikovskaya ~]$ mv ski.places/equipment ski.places/equiplist
[eavernikovskaya@eavernikovskaya ~]$ ls ski.places/
equiplist
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.24: Переименование файл `equipment`

В домашнем каталоге создаём файл `abc1` и копируем его в каталог `~/ski.places` с название `equiplist2` (рис. 3.25)

```
[eavernikovskaya@eavernikovskaya ~]$ touch abc1
[eavernikovskaya@eavernikovskaya ~]$ cp abc1 ski.places/equiplist2
[eavernikovskaya@eavernikovskaya ~]$ ls ski.places/
equiplist  equiplist2
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.25: Работа с файлом abc1

Создаём каталог с именем equipment в каталоге ~/ski.places (рис. 3.26)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir ski.places/equipment
[eavernikovskaya@eavernikovskaya ~]$ ls ski.places/
equiplist  equiplist2  equipment
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.26: Создание каталога equipment

Перемещаем файлы equiplist и equiplist2 в каталог ~/ski.places/equipment (рис. 3.27)

```
[eavernikovskaya@eavernikovskaya ~]$ mv ski.places/equiplist ski.places/equiplist2 ski.places/equipment/
[eavernikovskaya@eavernikovskaya ~]$ ls ski.places/equipment/
equiplist  equiplist2
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.27: Перемещение файлов equiplist и equiplist2

Создаём каталог ~/newdir и перемещаем его с названием plans в каталог ~/ski.places (рис. 3.28)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir newdir
[eavernikovskaya@eavernikovskaya ~]$ mv newdir/ ski.places/plans
[eavernikovskaya@eavernikovskaya ~]$ ls ski.places/
equipment  plans
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.28: Работа с каталогом newdir

3.6 Пункт 3

Создаём каталог с названием australia и лишаем членов группы и других пользователей прав на выполнение (рис. 3.29), (рис. 3.30)


```
[eavernikovskaya@eavernikovskaya ~]$ mkdir australia
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
```

Рис. 3.29: Создание каталога australia

```
[eavernikovskaya@eavernikovskaya ~]$ chmod g-x australia
[eavernikovskaya@eavernikovskaya ~]$ chmod o-x australia
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
```

Рис. 3.30: Изменение прав (1)

Создаём каталог с названием play и лишаем членов группы и других пользователей прав на чтение (рис. 3.31), (рис. 3.32)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir play
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya   132 мар  3 21:29 git-extended
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya    0 мар 12 19:21 katerok
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:24 may
drwx--x--x. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:35 monthly
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:03 play
```

Рис. 3.31: Создание каталога play

```
[eavernikovskaya@eavernikovskaya ~]$ chmod g-r play
[eavernikovskaya@eavernikovskaya ~]$ chmod o-r play
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya   132 мар  3 21:29 git-extended
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya    0 мар 12 19:21 katerok
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:24 may
drwx--x--x. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:35 monthly
drwx--x--x. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:03 play
```

Рис. 3.32: Изменение прав (2)

Создаём файл с названием my_os. Владельцу файла добавляем права на выполнение и лишаем его прав на запись (рис. 3.33), (рис. 3.34)


```
[eavernikovskaya@eavernikovskaya ~]$ touch my_os
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya  132 мар  3 21:29 git-extended
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya    0 мар 12 19:21 katerok
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:24 may
drwx--x--x. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:35 monthly
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:05 my_os
```

Рис. 3.33: Создание файла my_os

```
[eavernikovskaya@eavernikovskaya ~]$ chmod u-w my_os
[eavernikovskaya@eavernikovskaya ~]$ chmod u+x my_os
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya  132 мар  3 21:29 git-extended
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya    0 мар 12 19:21 katerok
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:24 may
drwx--x--x. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:35 monthly
-r-xr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:05 my_os
```

Рис. 3.34: Изменение прав (3)

Создаём файл с названием feathers и добавляем членам группы права на запись (рис. 3.35), (рис. 3.36)

```
[eavernikovskaya@eavernikovskaya ~]$ touch feathers
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:40 feathers
```

Рис. 3.35: Создание файла feathers

```
[eavernikovskaya@eavernikovskaya ~]$ chmod g+w feathers
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
-rw-rw-r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:40 feathers
```

Рис. 3.36: Изменение прав (4)

3.7 Пункт 4

Просматриваем содержимое файла `/etc/passwd` с помощью команды `cat` (рис. 3.37)

```
[eavernikovskaya@eavernikovskaya ~]$ cat /etc/passwd
root:x:0:0:Super User:/root:/bin/bash
bin:x:1:1:bin:/bin:/usr/sbin/nologin
daemon:x:2:2:daemon:/sbin:/usr/sbin/nologin
adm:x:3:4:adm:/var/adm:/usr/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/usr/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/usr/sbin/nologin
operator:x:11:0:operator:/root:/usr/sbin/nologin
games:x:12:100:games:/usr/games:/usr/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/usr/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/usr/sbin/nologin
dbus:x:81:81:System Message Bus:/usr/sbin/nologin
tss:x:59:59:Account used for TPM access:/usr/sbin/nologin
```

Рис. 3.37: Просмотр файла `passwd`

Копируем файл `~/feathers` в файл `~/file.old` (рис. 3.38)

```
[eavernikovskaya@eavernikovskaya ~]$ cp feathers file.old
[eavernikovskaya@eavernikovskaya ~]$ ls
abc1      file.old  LICENSE  my_os    ski.plases  Загрузки  Общедоступные
australia git-extended  may      play     Видео       Изображения  'Рабочий стол'
feathers  katerok    monthly  reports  Документы  Музыка       Шаблоны
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.38: Копирование файла `feathers`

Перемещаем файл `file.old` в каталог `~/play` (рис. 3.39)

```
[eavernikovskaya@eavernikovskaya ~]$ mv file.old play/
[eavernikovskaya@eavernikovskaya ~]$ ls play
file.old
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.39: Перемещение файла `file.old`

Копируем каталог `~/play` с именем `~/fun` (рис. 3.40)

```
[eavernikovskaya@eavernikovskaya ~]$ cp -r play/ fun/
[eavernikovskaya@eavernikovskaya ~]$ ls
abc1      fun      LICENSE  my_os    ski.plases  Загрузки  Общедоступные
australia git-extended may      play      Видео      Изображения 'Рабочий стол'
feathers  katerok  monthly  reports   Документы  Музыка     Шаблоны
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.40: Копирование каталога play

После перемещаем каталог ~/fun в каталог ~/play и называем его games (рис. 3.41)

```
[eavernikovskaya@eavernikovskaya ~]$ mv fun/ play/games
[eavernikovskaya@eavernikovskaya ~]$ ls play
file.old  games
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.41: Перемещение и переименование каталога fun

Лишаем владельца файла feathers прав на чтение введя *chmod u-r feathers* (рис. 3.42)

```
[eavernikovskaya@eavernikovskaya ~]$ chmod u-r feathers
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 18:02 australia
--w-rw-r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 18:40 feathers
```

Рис. 3.42: Лишение прав на чтение

Теперь пытаемся просмотреть файл feathers с помощью команды *cat*, но у нас это не получится, так как мы лишены прав на чтение (рис. 3.43)

```
[eavernikovskaya@eavernikovskaya ~]$ cat feathers
cat: feathers: Отказано в доступе
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.43: Попытка просмотреть файл feathers

Возвращаем владельцу файла feathers права на чтение введя *chmod u+r feathers* (рис. 3.44)

```
[eavernikovskaya@eavernikovskaya ~]$ chmod u+r feathers
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
-rw-rw-r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:40 feathers
```

Рис. 3.44: Добавление прав на чтение

Лишаем владельца каталога play прав на выполнение введя *chmod u-x play* (рис. 3.45)

```
[eavernikovskaya@eavernikovskaya ~]$ chmod u-x play
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:02 australia
-rw-rw-r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:40 feathers
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 132 мар  3 21:29 git-extended
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya    0 мар 12 19:21 katerok
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
-rw-r--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:24 may
drwx--x--x. 1 eavernikovskaya eavernikovskaya    0 мар 18 17:35 monthly
-r-xr--r--. 1 eavernikovskaya eavernikovskaya    0 мар 18 18:05 my_os
drw---x--x. 1 eavernikovskaya eavernikovskaya   26 мар 18 18:47 play
```

Рис. 3.45: Лишение прав на выполнение

Пытаемся перейти в каталог play, но у нас это не получится, так как мы лишены прав на выполнение (т.е. лишены возможности сделать каталог play текущим) (рис. 3.46)

```
[eavernikovskaya@eavernikovskaya ~]$ cd play/
bash: cd: play/: Отказано в доступе
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.46: Попытка перейти в каталог play

Возвращаем владельцу каталога play права на выполнение введя *chmod u+x play* (рис. 3.47)

```

[eavernikovskaya@eavernikovskaya ~]$ chmod u+x play
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:56 abc1
drwxr--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 18:02 australia
-rw-rw-r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 18:40 feathers
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 132 мар 3 21:29 git-extended
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 мар 12 19:21 katerok
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:24 may
drwx--x--x. 1 eavernikovskaya eavernikovskaya 0 мар 18 17:35 monthly
-r-xr--r--. 1 eavernikovskaya eavernikovskaya 0 мар 18 18:05 my_os
drwx--x--x. 1 eavernikovskaya eavernikovskaya 26 мар 18 18:47 play

```

Рис. 3.47: Добавление прав на выполнение

3.8 Пункт 5

С помощью команды *man* смотрим информацию о командах *mount*, *fsck*, *mkfs*, *kill* (рис. 3.48), (рис. 3.49), (рис. 3.50), (рис. 3.51), (рис. 3.52)

```

MOUNT(8)                                     System Administration                                MOUNT(8)
NAME
    mount - mount a filesystem
SYNOPSIS
    mount [-h|-V]

    mount [-l] [-t fstype]

    mount -a [-ffnrsvw] [-t fstype] [-O optlist]

    mount [-fnrsvw] [-o options] device|mountpoint

    mount [-fnrsvw] [-t fstype] [-o options] device mountpoint

    mount --bind|--rbind|--move olddir newdir

    mount --make-[shared|slave|private|unbindable|rshared|rslave|rprivate|runbindable] 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 provided in a virtual way by network or other services.

    The standard form of the mount command is:

    mount -t type device dir

```

Рис. 3.48: Информация о команде mount

```
FSCK(8)                                     System Administration                                     FSCK(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype] [filesystem...] [--] [fs-specific-options]

DESCRIPTION
    fsck is used to check and optionally repair one or more linux filesystems. filesystem can be a device name (e.g.,
    /dev/hdc1, /dev/sdb2), a mount point (e.g., /, /usr, /home), or a filesystem label or UUID specifier (e.g.,
    UUID=8868abf6-88c5-4a83-98b8-bfc24b57f7bd or LABEL=root). Normally, the fsck program will try to handle filesystems
    on different physical disk drives in parallel to reduce the total amount of time needed to check all of them.

    If no filesystems are specified on the command line, and the -A option is not specified, fsck will default to
    checking filesystems in /etc/fstab serially. This is equivalent to the -As options.

    The exit status returned by fsck is the sum of the following conditions:

    0      No errors

    1      Filesystem errors corrected

    2      System should be rebooted
```

Рис. 3.49: Информация о команде fsck

```
MKFS(8)                                     System Administration                                     MKFS(8)

NAME
    mkfs - build a Linux filesystem

SYNOPSIS
    mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
    This mkfs frontend is deprecated in favour of filesystem specific mkfs.<type> utils.

    mkfs is used to build a linux filesystem on a device, usually a hard disk partition. The device argument is either
    the device name (e.g., /dev/hda1, /dev/sdb2), or a regular file that shall contain the filesystem. The size argument
    is the number of blocks to be used for the filesystem.

    The exit status returned by mkfs is 0 on success and 1 on failure.

    In actuality, mkfs is simply a front-end for the various filesystem builders (mkfs.fstype) available under Linux.
    The filesystem-specific builder is searched for via your PATH environment setting only. Please see the
    filesystem-specific builder manual pages for further details.

OPTIONS
    -t, --type type
        Specify the type of filesystem to be built. If not specified, the default filesystem type (currently ext2) is
        used.

    fs-options
        Filesystem-specific options to be passed to the real filesystem builder.

    -V, --verbose
        Produce verbose output, including all filesystem-specific commands that are executed. Specifying this option
        more than once inhibits execution of any filesystem-specific commands. This is really only useful for testing.
```

Рис. 3.50: Информация о команде mkfs

```
KILL(1)                                User Commands                                KILL(1)

NAME
    kill - terminate a process

SYNOPSIS
    kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds signal] [--] pid/name...

    kill -l [number] | -L

DESCRIPTION
    The command kill sends the specified signal to the specified processes or process groups.

    If no signal is specified, the TERM signal is sent. The default action for this signal is to terminate the process. This signal should be used in preference to the KILL signal (number 9), since a process may install a handler for the TERM signal in order to perform clean-up steps before terminating in an orderly fashion. If a process does not terminate after a TERM signal has been sent, then the KILL signal may be used; be aware that the latter signal cannot be caught, and so does not give the target process the opportunity to perform any clean-up before terminating.

    Most modern shells have a builtin kill command, with a usage rather similar to that of the command described here. The --all, --pid, and --queue options, and the possibility to specify processes by command name, are local extensions.

    If signal is 0, then no actual signal is sent, but error checking is still performed.
```

Рис. 3.51: Информация о команде kill

```
[eavernikovskaya@eavernikovskaya ~]$ man mount
[eavernikovskaya@eavernikovskaya ~]$ man fsck
[eavernikovskaya@eavernikovskaya ~]$ man mkfs
[eavernikovskaya@eavernikovskaya ~]$ man kill
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.52: Использование man

С помощью команды *man* я прочитала описание каждой из приведённой в задании команды:

1. **mount**: Используется для монтирования файловых систем в определенные точки монтирования в операционной системе Linux.
2. **fsck**: Проверяет и исправляет целостность файловой системы, обнаруживая и исправляя ошибки на диске.
3. **mkfs**: Создает новую файловую систему на указанном устройстве.
4. **kill**: Используется для отправки сигналов процессам в Linux, что может привести к завершению процесса.

3.9 Ответы на контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу.
 - Ext2, Ext3, Ext4 или Extended Filesystem - это стандартная файловая система для Linux. Она была разработана еще для Minix. Она самая стабильная из всех существующих, кодовая база изменяется очень редко и эта файловая система содержит больше всего функций. Версия ext2 была разработана уже именно для Linux и получила много улучшений. В 2001 году вышла ext3, которая добавила еще больше стабильности благодаря использованию журналирования. В 2006 была выпущена версия ext4, которая используется во всех дистрибутивах Linux до сегодняшнего дня. В ней было внесено много улучшений, в том числе увеличен максимальный размер раздела до одного экзбайта.
 - Btrfs или B-Tree File System - это совершенно новая файловая система, которая сосредоточена на отказоустойчивости, легкости администрирования и восстановления данных. Файловая система объединяет в себе очень много новых интересных возможностей, таких как размещение на нескольких разделах, поддержка подтомов, изменение размера на лету, создание мгновенных снимков, а также высокая производительность. Но многими пользователями файловая система Btrfs считается нестабильной. Тем не менее, она уже используется как файловая система по умолчанию в OpenSUSE и SUSE Linux.
2. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры.
 - / — root каталог. Содержит в себе всю иерархию системы;
 - /bin — здесь находятся двоичные исполняемые файлы. Основные общие команды, хранящиеся отдельно от других программ в системе (прим.: pwd,

- ls, cat, ps);
- /boot — тут расположены файлы, используемые для загрузки системы (образ initrd, ядро vmlinuz);
 - /dev — в данной директории располагаются файлы устройств (драйверов). С помощью этих файлов можно взаимодействовать с устройствами. К примеру, если это жесткий диск, можно подключить его к файловой системе. В файл принтера же можно написать напрямую и отправить задание на печать;
 - /etc — в этой директории находятся файлы конфигураций программ. Эти файлы позволяют настраивать системы, сервисы, скрипты системных демонов;
 - /home — каталог, аналогичный каталогу Users в Windows. Содержит домашние каталоги учетных записей пользователей (кроме root). При создании нового пользователя здесь создается одноименный каталог с аналогичным именем и хранит личные файлы этого пользователя;
 - /lib — содержит системные библиотеки, с которыми работают программы и модули ядра;
 - /lost+found — содержит файлы, восстановленные после сбоя работы системы. Система проведет проверку после сбоя и найденные файлы можно будет посмотреть в данном каталоге;
 - /media — точка монтирования внешних носителей. Например, когда вы вставляете диск в дисковод, он будет автоматически смонтирован в директорию /media/cdrom;
 - /mnt — точка временного монтирования. Файловые системы подключаемых устройств обычно монтируются в этот каталог для временного использования;
 - /opt — тут расположены дополнительные (необязательные) приложения. Такие программы обычно не подчиняются принятой иерархии и хранят свои файлы в одном подкаталоге (бинарные, библиотеки, конфигурации);

- /proc — содержит файлы, хранящие информацию о запущенных процессах и о состоянии ядра ОС;
- /root — директория, которая содержит файлы и личные настройки суперпользователя;
- /run — содержит файлы состояния приложений. Например, PID-файлы или UNIX-сокеты;
- /sbin — аналогично /bin содержит бинарные файлы. Утилиты нужны для настройки и администрирования системы суперпользователем;
- /srv — содержит файлы сервисов, предоставляемых сервером (прим. FTP или Apache HTTP);
- /sys — содержит данные непосредственно о системе. Тут можно узнать информацию о ядре, драйверах и устройствах;
- /tmp — содержит временные файлы. Данные файлы доступны всем пользователям на чтение и запись. Стоит отметить, что данный каталог очищается при перезагрузке;
- /usr — содержит пользовательские приложения и утилиты второго уровня, используемые пользователями, а не системой. Содержимое доступно только для чтения (кроме root). Каталог имеет вторичную иерархию и похож на корневой;
- /var — содержит переменные файлы. Имеет подкаталоги, отвечающие за отдельные переменные. Например, логи будут храниться в /var/log, кэш в /var/cache, очереди заданий в /var/spool/ и так далее.

3. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе?

Монтирование тома.

4. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы?

Отсутствие синхронизации между образом файловой системы в памяти и ее данными на диске в случае аварийного останова может привести к появлению следующих ошибок:

- Один блок адресуется несколькими mode (принадлежит нескольким файлам).
- Блок помечен как свободный, но в то же время занят (на него ссылается onode).
- Блок помечен как занятый, но в то же время свободен (ни один inode на него не ссылается).
- Неправильное число ссылок в inode (недостаток или избыток ссылающихся записей в каталогах).
- Несовпадение между размером файла и суммарным размером адресуемых inode блоков.
- Недопустимые адресуемые блоки (например, расположенные за пределами файловой системы).
- “Потерянные” файлы (правильные inode, на которые не ссылаются записи каталогов).
- Недопустимые или неразмещенные номера inode в записях каталогов.

5. Как создаётся файловая система?

Команда `mkfs` - позволяет создать файловую систему Linux.

6. Дайте характеристику командам для просмотра текстовых файлов.

- `Cat` - выводит содержимое файла на стандартное устройство вывода.
- Выполнение команды `head` выведет первые 10 строк текстового файла.
- Выполнение команды `tail` выведет последние 10 строк текстового файла.
- Команда `tac` - это тоже самое, что и `cat`, только отображает строки в обратном порядке.

- Для того, чтобы просмотреть огромный текстовый файл применяются команды для постраничного просмотра. Такие как `more` и `less`.

7. Приведите основные возможности команды `cp` в Linux.

`Ср` – копирует или перемещает директорию, файлы.

8. Приведите основные возможности команды `mv` в Linux.

`Мv` - переименовывает или перемещает файл или директорию

9. Что такое права доступа? Как они могут быть изменены?

Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.

4 Выводы

В ходе выполнения лабораторной работы мы ознакомились с файловой системой Linux, её структурой, именами и содержанием каталогов. А также приобрели практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

5 Список литературы

Не пользовалась сайтами.