

Отчёт по лабораторной работе №7

По теме: Анализ файловой структуры UNIX. Команды для работы с файлами и каталогами

Выполнил: Чубаев Кирилл Евгеньевич, НММбд-01-24

Содержание

1	Цель работы.....	1
2	Выполнение лабораторной работы	1
3	Вывод.....	7
4	Контрольные вопросы.....	7

Список иллюстраций

Figure 1: Выполнение примеров.....	2
Figure 2: Работа с каталогами	3
Figure 3: Настройка прав доступа	4
Figure 4: Файл /etc/passwd	5
Figure 5: Работа с файлами и правами доступа	6
Figure 6: Чтение команд	6

1 Цель работы

Целью данной лабораторной работы является 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, по проверке использования диска и обслуживанию файловой системы.

2 Выполнение лабораторной работы

1. Я выполнил примеры, приведённые в первой части описания лабораторной работы.

```

kechubaev@kechubaev:~$ touch hey.md
kechubaev@kechubaev:~$ cp hey.md may
kechubaev@kechubaev:~$ mkdir monthly
kechubaev@kechubaev:~$ cp hey.md april
kechubaev@kechubaev:~$ cp april may monthly/
kechubaev@kechubaev:~$ cp monthly/may monthly/june
kechubaev@kechubaev:~$ ls monthly/
april  june  may
kechubaev@kechubaev:~$ mkdir monthly.00
kechubaev@kechubaev:~$ cp -r monthly monthly.00/
kechubaev@kechubaev:~$ cp -r monthly.00/ /tmp
kechubaev@kechubaev:~$ mv april july
kechubaev@kechubaev:~$ mv july monthly.00/
kechubaev@kechubaev:~$ ls monthly.00/
july  monthly
kechubaev@kechubaev:~$ mv monthly.00/ monthly.01
kechubaev@kechubaev:~$ mkdir reports
kechubaev@kechubaev:~$ mv monthly.01/ reports/
kechubaev@kechubaev:~$ mv reports/monthly.01/ reports/monthly
kechubaev@kechubaev:~$ touch may
kechubaev@kechubaev:~$ ls -l may
-rw-rw-r-- 1 kechubaev kechubaev 0 Mar 29 16:56 may
kechubaev@kechubaev:~$ chmod u+x may
kechubaev@kechubaev:~$ ls -l may
-rwxrw-r-- 1 kechubaev kechubaev 0 Mar 29 16:56 may
kechubaev@kechubaev:~$ chmod u-x may
kechubaev@kechubaev:~$ ls -l may
-rw-rw-r-- 1 kechubaev kechubaev 0 Mar 29 16:56 may
kechubaev@kechubaev:~$ cd
kechubaev@kechubaev:~$ mkdir monthly
mkdir: cannot create directory 'monthly': File exists
kechubaev@kechubaev:~$ chmod g-r, o-r monthly
chmod: invalid mode: 'g-r,'
Try 'chmod --help' for more information.
kechubaev@kechubaev:~$ chmod g-r, o-r monthly/
chmod: invalid mode: 'g-r,'
Try 'chmod --help' for more information.
kechubaev@kechubaev:~$ chmod g-r,o-r monthly/
kechubaev@kechubaev:~$ touch abc1
kechubaev@kechubaev:~$ chmod g+w abc1

```

Figure 1: Выполнение примеров

2. Потом я начал выполнять задания с каталогами:

2.1. Я скопировал файл /usr/include/sys/io.h в домашний каталог и переименовал его equipment. Такого нет, взял другой файл.

2.2. - 2.5. В домашнем каталоге создал директорию ski.plases. и переместил в него файл equipment. Переименовал файл equipment в equiplist. После этого создал в домашнем каталоге файл abc1 и скопировал его в каталог ski.plases. и переименовал в equiplist2.

2.6. - 2.7. Далее я создал каталог с именем equipment в каталоге ski.plases. Затем переместил файлы equiplist и equiplist2 в каталог equipment.

2.8. Создал и переместил каталог newdir в каталог ski.plases и называем его plans.

```
kechubaev@kechubaev:~$ cp /usr/include/linux/sysinfo.h ~
kechubaev@kechubaev:~$ mv sysinfo.h equipment
kechubaev@kechubaev:~$ mkdir ski.plases
kechubaev@kechubaev:~$ mv equipment ski.plases
kechubaev@kechubaev:~$ mv equipment ski.plases/
mv: cannot stat 'equipment': No such file or directory
kechubaev@kechubaev:~$ mv ski.plases/equipment ski.plases/equiplist
kechubaev@kechubaev:~$ cp abc1 ski.plases/equiplist2
kechubaev@kechubaev:~$ cd ski.plases
kechubaev@kechubaev:~/ski.plases$ mkdir equipment
kechubaev@kechubaev:~/ski.plases$ mv equiplist equipment/
kechubaev@kechubaev:~/ski.plases$ mv equiplist2 equipment/
kechubaev@kechubaev:~/ski.plases$ cd
kechubaev@kechubaev:~$ mkdir newdir
kechubaev@kechubaev:~$ mv newdir/ ski.plases/
kechubaev@kechubaev:~$ mv ski.plases/newdir ski.plases/plans
```

Figure 2: Работа с каталогами

3. Затем я определил опции команды chmod, необходимые для того, чтобы присвоить файлам из хода работы нужные права доступа.

```

kechubaev@kechubaev:~$ mkdir australia play
kechubaev@kechubaev:~$ touch my_os feathers
kechubaev@kechubaev:~$ chmod 744 australia/
kechubaev@kechubaev:~$ chmod 711 play/
kechubaev@kechubaev:~$ chmod 544 my_os
kechubaev@kechubaev:~$ chmod 664 feathers
kechubaev@kechubaev:~$ ls -l
total 92
-rw-rw-r-- 1 kechubaev kechubaev    0 Mar 29 16:58 abc1
drwxr--r-- 2 kechubaev kechubaev 4096 Mar 29 17:04 australia
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar 23 21:27 bin
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar 24 15:46 Desktop
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar  9 00:35 Documents
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar 24 17:16 Downloads
-rw-rw-r-- 1 kechubaev kechubaev    0 Mar 29 17:04 feathers
-rw-rw-r-- 1 kechubaev kechubaev    7 Mar 29 16:23 hello.md
-rw-rw-r-- 1 kechubaev kechubaev    0 Mar 29 16:50 hey.md
-rw-rw-r-- 1 kechubaev kechubaev 18657 Mar 15 21:01 LICENSE
-rw-rw-r-- 1 kechubaev kechubaev    0 Mar 29 16:56 may
drwx-wx--x 2 kechubaev kechubaev 4096 Mar 29 16:52 monthly
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar  9 00:35 Music
-r-xr--r-- 1 kechubaev kechubaev    0 Mar 29 17:04 my_os
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar  9 00:35 Pictures
drwx--x--x 2 kechubaev kechubaev 4096 Mar 29 17:04 play
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar  9 00:35 Public
drwxrwxr-x 3 kechubaev kechubaev 4096 Mar 29 16:56 reports
drwxrwxr-x 4 kechubaev kechubaev 4096 Mar 29 17:03 ski.places
drwx----- 10 kechubaev kechubaev 4096 Mar 15 22:32 snap
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar  9 00:35 Templates
-rw-rw-r-- 1 kechubaev kechubaev   670 Mar 15 22:10 texput.log
drwxr-xr-x 2 kechubaev kechubaev 4096 Mar  9 00:35 Videos
drwxrwxr-x 5 kechubaev kechubaev 4096 Mar 24 15:45 work

```

Figure 3: Настройка прав доступа

4. После этого я начал работу с файлами и правами доступа:

4.1. Просмотрел содержимое файла /etc/passwd:

```

kechubaev@kechubaev:~$ cat /etc/paswd
cat: /etc/paswd: No such file or directory
kechubaev@kechubaev:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
systemd-timesync:x:996:996:systemd Time Synchronization:/:/usr/sbin/nologin
dhcpcd:x:100:65534:DHCP Client Daemon,,,:/usr/lib/dhcpcd:/bin/false
messagebus:x:101:101::/nonexistent:/usr/sbin/nologin
syslog:x:102:102::/nonexistent:/usr/sbin/nologin
systemd-resolve:x:991:991:systemd Resolver:/:/usr/sbin/nologin
uidd:x:103:103::/run/uidd:/usr/sbin/nologin
usbmux:x:104:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
tss:x:105:105:TPM software stack,,,:/var/lib/tpm:/bin/false
systemd-oom:x:990:990:systemd Userspace OOM Killer:/:/usr/sbin/nologin
kernoops:x:106:65534:Kernel Oops Tracking Daemon,,,:/usr/sbin/nologin
whoopsie:x:107:109::/nonexistent:/bin/false
dnsmasq:x:999:65534:dnsmasq:/var/lib/misc:/usr/sbin/nologin
avahi:x:108:111:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
tcpdump:x:109:112::/nonexistent:/usr/sbin/nologin
sssd:x:110:113:SSSD system user,,,:/var/lib/sss:/usr/sbin/nologin
speech-dispatcher:x:111:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false
cups-pk-helper:x:112:114:user for cups-pk-helper service,,,:/nonexistent:/usr/sbin/nologin

```

Figure 4: Файл /etc/passwd

4.2 - 4.12. Выполнил все указанные действия по перемещению файлов и каталогов:

```

kechubaev@kechubaev:~$ cp feathers file.old
kechubaev@kechubaev:~$ mv file.old play
kechubaev@kechubaev:~$ mkdir fun
kechubaev@kechubaev:~$ cp -R play/ fun
kechubaev@kechubaev:~$ mv fun/ play/games
kechubaev@kechubaev:~$ chmod i-r feathers
chmod: invalid mode: 'i-r'
Try 'chmod --help' for more information.
kechubaev@kechubaev:~$ chmod u-r feathers
kechubaev@kechubaev:~$ cat feathers
cat: feathers: Permission denied
kechubaev@kechubaev:~$ cp feathers feathers2
cp: cannot open 'feathers' for reading: Permission denied
kechubaev@kechubaev:~$ chmod u+r feathers
kechubaev@kechubaev:~$ chmod -x play/
kechubaev@kechubaev:~$ cd play/
bash: cd: play/: Permission denied
kechubaev@kechubaev:~$ chmod +x play/

```

Figure 5: Работа с файлами и правами доступа

4.7. Если посмотреть файл feathers командой cat, то будет отказано в доступе.

4.8. Если скопировать файл feathers то у не получится это сделать, так как я ограничил себя в доступе для чтения.

5. Потом я прочитал инструкцию (man) по командам mount, fsck, mkfs, kill и кратко их охарактеризовал, приведя примеры.

```

kechubaev@kechubaev:~$ man mount
kechubaev@kechubaev:~$ man fsck
kechubaev@kechubaev:~$ man mkfs
kechubaev@kechubaev:~$ man kill

```

Figure 6: Чтение команд

- 1) mount - монтирование файловой системы к общему дереву каталогов. Для размонтирования используется команда umount.
- 2) fsck (проверка файловой системы) – это утилита командной строки, которая позволяет выполнять проверки согласованности и интерактивное исправление в одной или нескольких файловых системах Linux. Она использует программы, специфичные для типа файловой системы, которую она проверяет. Вы можете использовать команду fsck для восстановления поврежденных файловых систем в ситуациях, когда система не загружается или раздел не может быть смонтирован.

- 3) Буквы в mkfs значке означают “make file system” (создать файловую систему). Команда обычно используется для управления устройствами хранения в Linux. Вы можете рассматривать mkfs как инструмент командной строки для форматирования диска в определенной файловой системе.
- 4) Системный вызов kill может быть использован для отправки какого-либо сигнала какому-либо процессу или группе процесса.

3 Вывод

В ходе данной лабораторной работы я ознакомился с файловой системой Linux, её структурой, именами и содержанием каталогов. Научился совершать базовые операции с файлами, управлять правами их доступа для пользователя и групп. А также я ознакомился с анализом файловой системы. К тому же, я получил базовые навыки по проверке использования диска и обслуживанию файловой системы.

4 Контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу.

Ответ: Ext2FS (расширенная файловая система номер два). Многие годы ext2 была файловой системой по умолчанию в GNU/Linux. Ext2 заменила собой Extended File System (вот откуда появилось “Second” в названии). В “новой” файловой системе были исправлены некоторые проблемы, а также убраны ограничения. Отличная стабильность, комплексные инструментальные средства для спасения удаленных файлов, очень долгое время перезагрузки после аварии, есть вероятность частичной или полной потери данных после аварии. Одним из главных недостатков “традиционных” файловых систем, подобных Ext2FS, является низкая сопротивляемость к резким системным сбоям (сбой питания или авария программного обеспечения)

Ext3 (Расширенная файловая система номер три) - является наследником файловой системы Ext2FS. Ext3 совместима с Ext2, но обладает одной новой и очень интересной особенностью – запись. Процесс сохранения объекта происходит прежде чем запись в журнал. В результате мы получаем всегда последовательную файловую систему. Это приводит к тому, что при появлении проблем, проверка и восстановление происходят очень быстро. Время, потраченное на то, чтобы проверить файловую систему таким образом, пропорционально его фактическому использованию и не больше его размера.

ReiserFS (Это тоже журналируемая файловая система подобно Ext3FS, но их внутренняя структура радикально отличается. В ReiserFS используется концепция бинарных деревьев (binary-tree), позаимствованная из программного обеспечения баз данных.

JFS (журналируемая файловая система). JFS была разработана и использовалась IBM. Вначале JFS была закрытой системой, но недавно IBM решила открыть доступ для движения свободного программного обеспечения. Внутренняя структура JFS близка к ReiserFS. Средняя стабильность, нет комплексных инструментальных средств для спасения удаленных файлов, очень быстрая перезагрузка после аварии, очень хорошее восстановление данных после аварии.

2. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры. Ответ:

- Загрузочный блок занимает первый блок файловой системы. Только корневая файловая система имеет активный загрузочный блок, хотя место для него резервируется в каждой файловой системе.
- Суперблок располагается непосредственно за загрузочным блоком и содержит самую общую информацию о ФС (размер ФС, размер области индексных дескрипторов, их число, список свободных блоков, свободные индексные дескрипторы и т. д.). Суперблок всегда находится в оперативной памяти. Различные версии ОС Unix способны поддерживать разные типы файловых систем. Поэтому у структуры суперблока могут быть варианты (сведения о свободных блоках, например, часто хранятся не как список, а как шкала бит), но суперблок всегда располагается за загрузочным блоком. При монтировании файловой системы в оперативной памяти создается копия ее суперблока. Все последующие операции по созданию и удалению файлов влекут изменения копии суперблока в оперативной памяти. Эта копия периодически записывается на магнитный диск. Обычно причиной повреждения файловой системы является отключение электропитания (или зависание ОС) в тот момент, когда система производит копирование суперблока из оперативной памяти на магнитный диск.
- Область индексных дескрипторов содержит описатели файлов (inode). С каждым файлом связан один inode, но одному inode может соответствовать несколько файлов. Binode хранится вся информация о файле, кроме его имени. Область индексных дескрипторов имеет фиксированный формат и располагается непосредственно за суперблоком. Общее число описателей и, следовательно, максимальное число файлов задается в момент создания файловой системы. Описатели нумеруются натуральными числами. Первый описатель используется ОС для описания специального файла (файла «Плохих блоков»). То есть поврежденные блоки раздела рассматриваются ОС как принадлежащие к специальному файлу и поэтому считаются «занятыми». Второй – описывает корневой каталог файловой системы.

- В области данных расположены как обычные файлы, так и файлы каталогов (в том числе корневой каталог). Специальные файлы представлены в ФС только записями в соответствующих каталогах и индексными дескрипторами специального формата, т. е. места в области памяти не занимают.
3. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе? Ответ: Команда `cat` - позволяет вывести на экран содержимое любого файла, однако в таком виде эта команда практически не используется. Если файл слишком большой, то его содержимое пролистается на экране, а Вы увидите только последние строки файла. С помощью этой команды можно комбинировать и объединять копии файлов, а также создавать новые файлы. Если набрать просто в командной строке `cat` и нажать Enter, то можно вводить (и соответственно видеть) текст на экране. Повторное нажатие клавиши Enter удвоит строку и позволит начать следующую. Когда текст набран, следует одновременно нажать клавиши Ctrl и d.
4. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы? Ответ: Некорректность файловой системы может возникать:
- В результате насильственного прерывания операций ввода-вывода, выполняемых непосредственно с диском.
 - В результате нарушения работы дискового кэша. Кэширование данных с диска предполагает, что в течение некоторого времени результаты операций ввода-вывода никак не сказываются на содержимом диска — все изменения происходят с копиями блоков диска, временно хранящихся в буферах оперативной памяти (в этих буферах оседают данные из пользовательских файлов и служебная информация файловой системы, такая как каталоги, индексные дескрипторы, списки свободных, занятых и поврежденных блоков и т. п.)
5. Как создаётся файловая система? Ответ: Общее дерево файлов и каталогов системы Linux формируется из отдельных “ветвей”, соответствующих различным физическим носителям. В UNIX нет понятия “форматирования диска” (и команды форматирования), а используется понятие “создание файловой системы”. Когда мы получаем новый носитель, например, жесткий диск, мы должны создать на нем файловую систему. То есть каждому носителю ставится в соответствие отдельная файловая система. Чтобы эту файловую систему использовать для записи в нее файлов, надо ее вначале подключить в общее дерево каталогов (“смонтировать”). Вот и получается, что можно говорить о монтировании файловых систем или о монтировании носителей (с созданными на них файловыми системами). Например, создается файловая система типа `ext2fs`. Создание файловой системы типа `ext2fs` подразумевает создание в данном разделе на диске суперблока, таблицы индексных

дескрипторов и совокупности блоков данных. Делается все это все с помощью команды `mkfs`. В простейшем случае достаточно дать эту команду в следующем формате:

`[root]# mkfs -t ext2 /dev/hda5`, где `/dev/hda5` надо заменить указанием на соответствующее устройство или раздел. Например, если вы хотите создать файловую систему на диске, то команда примет вид:

`[root]# mkfs -t ext2 /dev/fd0`

После выполнения команды `mkfs` в указанном разделе будет создана файловая система `ext2fs`. В новой файловой системе автоматически создается один каталог с именем `lost+found`. Он используется в экстренных случаях программой `fsck`, поэтому не удаляйте его. Для того, чтобы начать работать с новой файловой системой, необходимо подключить ее в общее дерево каталогов, что делается с помощью команды `mount`. В качестве параметров команде `mount` надо, как минимум, указать устройство и “точку монтирования”. Точкой монтирования называется тот каталог в уже существующем и известном системе дереве каталогов, который будет теперь служить корневым каталогом для подключаемой файловой системы. После монтирования файловой системы в каталог `/mnt/disk2` прежнее содержимое этого каталога станет для вас недоступно до тех пор, пока вы не размонтируете вновь подключенную файловую систему. Прежнее содержимое не уничтожается, а просто становится временно недоступным. Поэтому в качестве точек монтирования лучше использовать пустые каталоги (заранее заготовленные).

6. Дайте характеристику командам, которые позволяют просмотреть текстовые файлы. Ответ: Для просмотра небольших файлов удобно пользоваться командой `cat`. Формат команды: `cat имя-файла`

Для просмотра больших файлов используйте команду `less` — она позволяет осуществлять постраничный просмотр файлов (длина страницы соответствует размеру экрана). Формат команды: `less имя-файла`

Для управления процессом просмотра можно использовать следующие управляющие клавиши: - `Space` — переход на следующую страницу, - `ENTER` — сдвиг вперёд на одну строку, - `b` — возврат на предыдущую страницу, - `h` — обращение за подсказкой, - `q` — выход в режим командной строки.

Для просмотра начала файла можно воспользоваться командой `head`. По умолчанию она выводит первые 10 строк файла. Формат команды: `head [-n] имя-файла`, где `n` — количество выводимых строк.

Команда `tail` выводит несколько (по умолчанию 10) последних строк файла. Формат команды: `tail [-n] имя-файла`, где `n` — количество выводимых строк.

7. Приведите основные возможности команды `cp` в Linux. Ответ: Копирование отдельных файлов Для копирования файла следует использовать утилиту `cp` с аргументами, представленными путями к исходному и целевому файлам.

Копирование файлов в другую директорию В том случае, если в качестве пути к целевому файлу используется путь к директории, исходные файлы будут скопированы в эту целевую директорию.

Команда `cp -r` Для копирования директорий целиком следует использовать команду `cp -r` (параметр `-r` позволяет осуществлять рекурсивное копирование всех файлов из всех поддиректорий).

Копирование множества файлов в директорию Вы также можете использовать утилиту `cp` для копирования множества файлов в одну директорию. В этом случае последний аргумент (аргумент, указывающий на цель) должен быть представлен путем к директории.

Команда `cp -i` Для предотвращения перезаписи существующих файлов в ходе использования утилиты `cp` следует использовать параметр `-i` (для активации интерактивного режима копирования).

8. Назовите и дайте характеристику командам перемещения и переименования файлов и каталогов. Ответ: Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов. Формат команды `mv`: `mv [-опции] старый_файл новый_файл` Примеры:
- Переименование файлов в текущем каталоге. Изменить название файла `april` на `july` в домашнем каталоге: `cd mv april july`
 - Перемещение файлов в другой каталог. Переместить файл `july` в каталог `monthly.00`: `mv july monthly.00 ls monthly.00` Результат: `april july june may`. Если необходим запрос подтверждения о перезаписи файла, то нужно использовать опцию `i`.
 - Переименование каталогов в текущем каталоге. Переименовать каталог `monthly.00` в `monthly.01` `mv monthly.00 monthly.01`
 - Перемещение каталога в другой каталог. Переместить каталог `monthly.01` в каталог `reports`: `mkdir reports mv monthly.01 reports`
 - Переименование каталога, не являющегося текущим. Переименовать каталог `reports/monthly.01` в `reports/monthly`: `mv reports/monthly.01 reports/monthly`
9. Что такое права доступа? Как они могут быть изменены? Ответ: Права доступа — совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носителям, процессам и другим ресурсам). Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора. Формат команды: `chmod режим имя_файла` Режим (в формате команды) имеет

следующие компоненты структуры и способ записи: = установить право -
лишить права + дать право r чтение w запись x выполнение u (user) владелец
файла g (group) группа, к которой принадлежит владелец файла o (others) все
остальные В работе с правами доступа можно использовать их цифровую
запись (восьмеричное значение) вместо символьной