Отчёт по лабораторной работе №5

дисциплина: Операционные системы

Студент: Зиязетдинов Алмаз Радикович

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

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.

2 Задание

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.
2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:

- Скопируйте файл /usr/include/sys/io.h в домашний каталог и назовите его equipment. Если файла io.h нет, то используйте любой другой файл в каталоге /usr/include/sys/ вместо него.
- В домашнем каталоге создайте директорию ~/ski.plases.
- Переместите файл equipment в каталог ~/ski.plases.
- Переименуйте файл ~/ski.plases/equipment в ~/ski.plases/equiplist.
- Создайте в домашнем каталоге файл abc1 и скопируйте его в каталог ~/ski.plases, назовите его equiplist2.
- Создайте каталог с именем equipment в каталоге ~/ski.plases.
- Переместите файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.
- Создайте и переместите каталог ~/newdir в каталог ~/ski.plases и назовите его plans.

3. Определите опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет: drwxr–r– australia drwx–x–x play -r-xr–r– my_os -rw-rw-r– feathers При необходимости создайте нужные файлы.
4. Проделайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:

- Просмотрите содержимое файла /etc/password.
- Скопируйте файл ~/feathers в файл ~/file.old.
- Переместите файл ~/file.old в каталог ~/play.
- Скопируйте каталог ~/play в каталог ~/fun.
- Переместите каталог ~/fun в каталог ~/play и назовите его games.
- Лишите владельца файла ~/feathers права на чтение.
- Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat?
- Что произойдёт, если вы попытаетесь скопировать файл ~/feathers?
- Дайте владельцу файла ~/feathers право на чтение.
- Лишите владельца каталога ~/play права на выполнение.
- Перейдите в каталог ~/play. Что произошло?
- Дайте владельцу каталога ~/play право на выполнение.

5. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

3 Выполнение лабораторной работы

Скопируем файл io.h в домашний каталог и назовём его equipment. Для этого воспользуемся командой cp и укажем путь к нашему файлу. Выполним проверку командой ls (рис. 3.1).

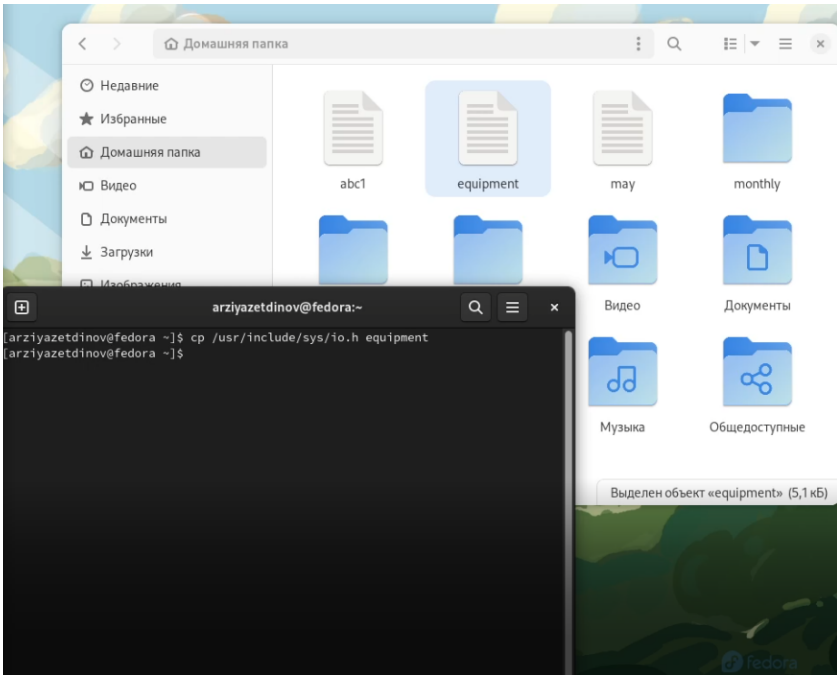


Рис. 3.1: Копирование файла io.h под названием equipment. Проверка.

В домашнем каталоге командой mkdir создаём директорию ski.plases. Выполняем проверку

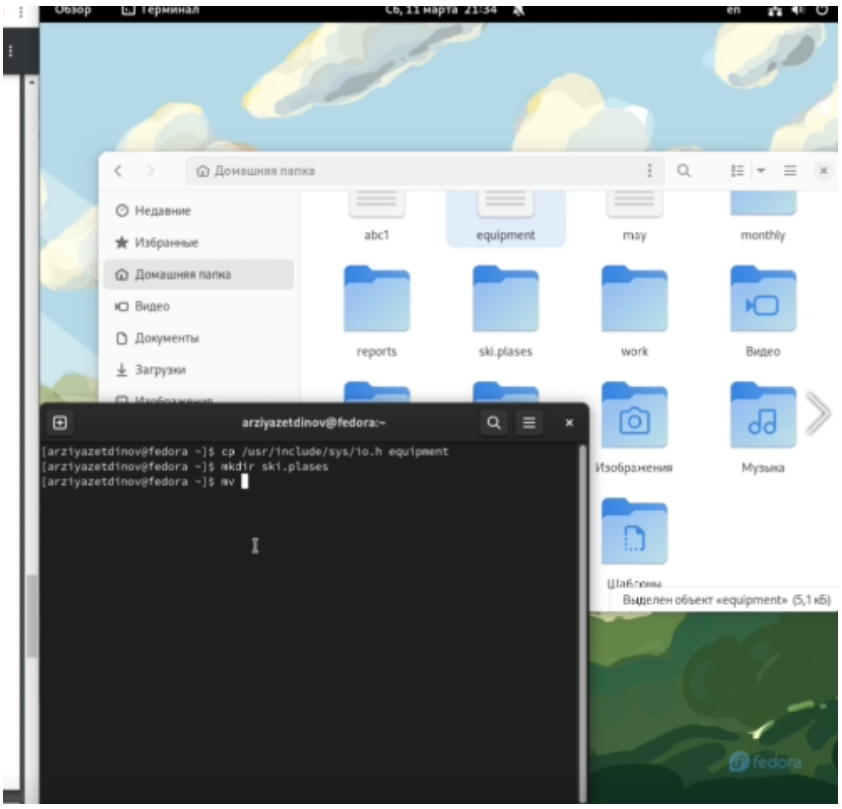


Рис. 3.2: Создание директории ski.plases

Перемещаем файл equipment в каталог ski.plases командой mv. Выполняем проверку (рис. 3).

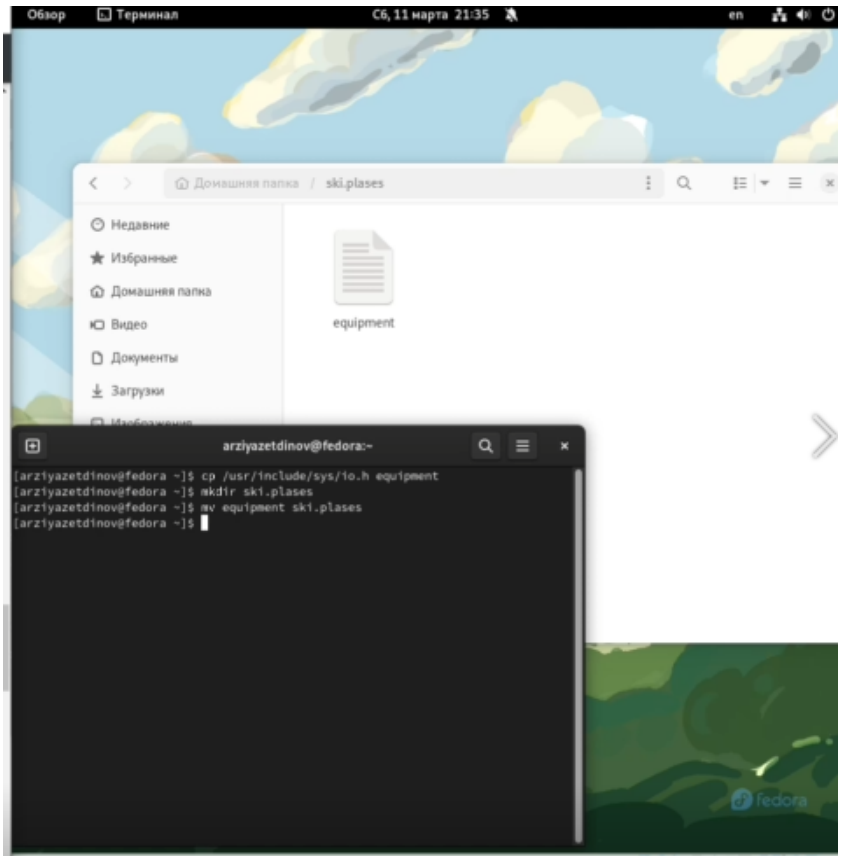


Рис. 3.3: Перемещение файла equipment в каталог ski.plases

Переименуем файл equipment, находящийся в катологе ski.plases в equiplist с помощью команды mv. Выполняем проверку (рис. 4).

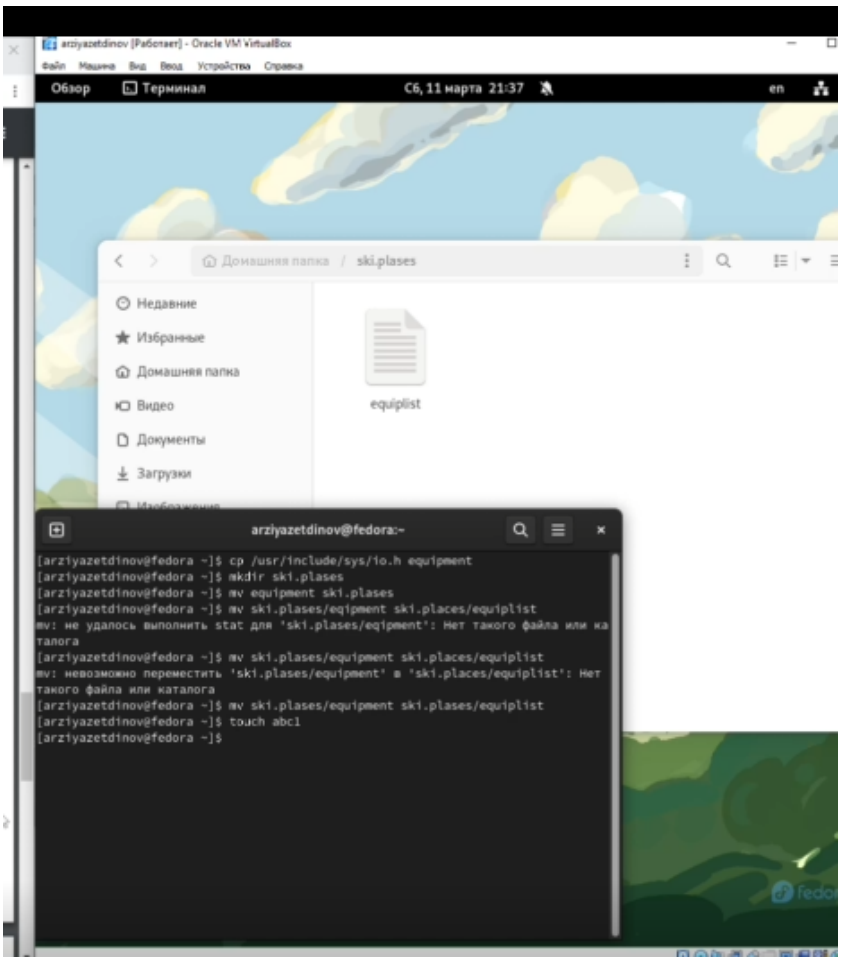


Рис. 3.4: Переименование файла equipment в equiplist

Создаём в домашнем каталоге файл abc1 командой touch и копируем его в каталог ski.plases под названием equiplist2. Выполняем проверку (рис. 5).

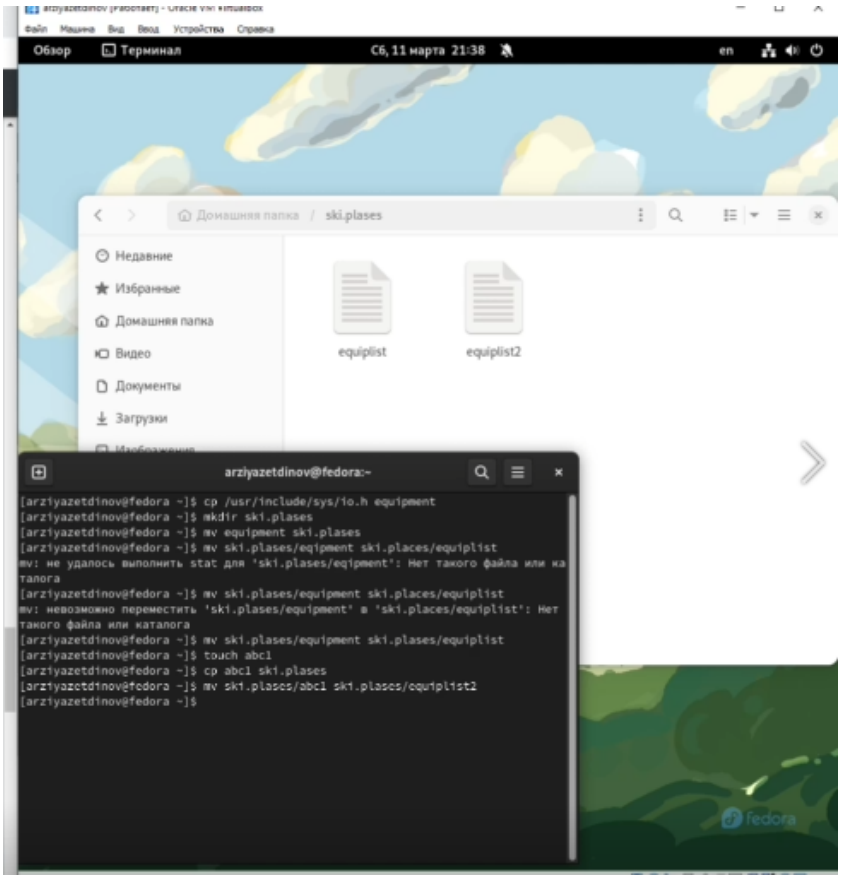


Рис. 3.5: Создание файла abc1 и копирование его в каталог ski.plases под названием equiplist2

Создаём каталог с именем equipment в каталоге ski.plases (рис. 6).

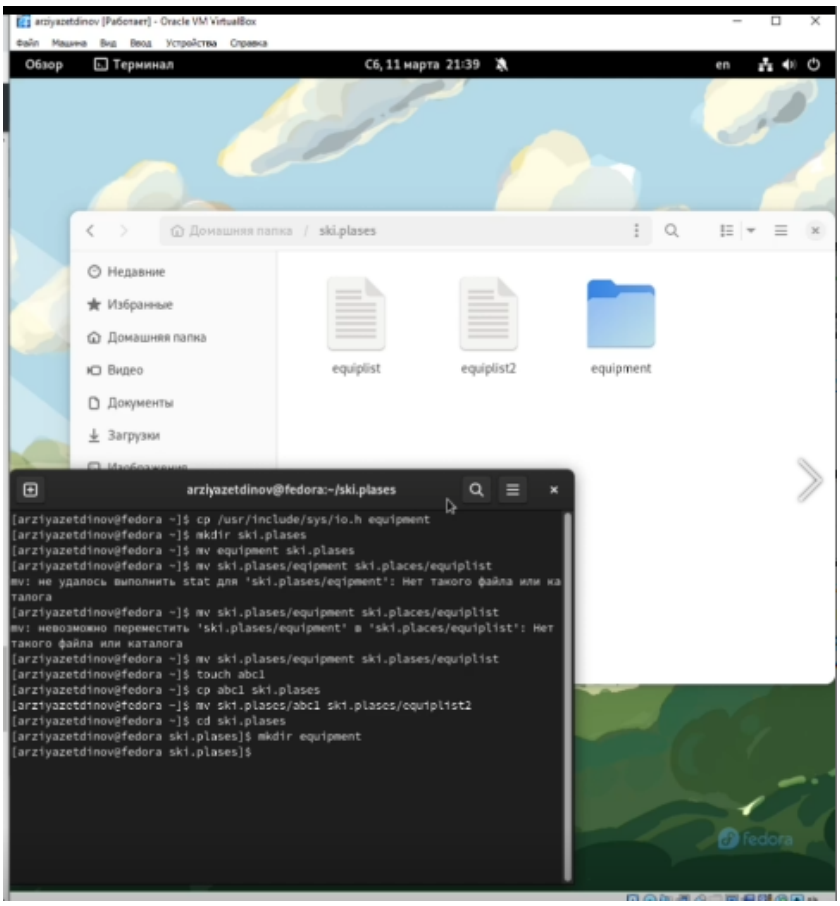


Рис. 3.6: Создание каталога equipment в каталоге ski.plases

Перемещаем файлы equiplist и equiplist2 из каталога ski.plases в подкаталог equipment (рис. 7).

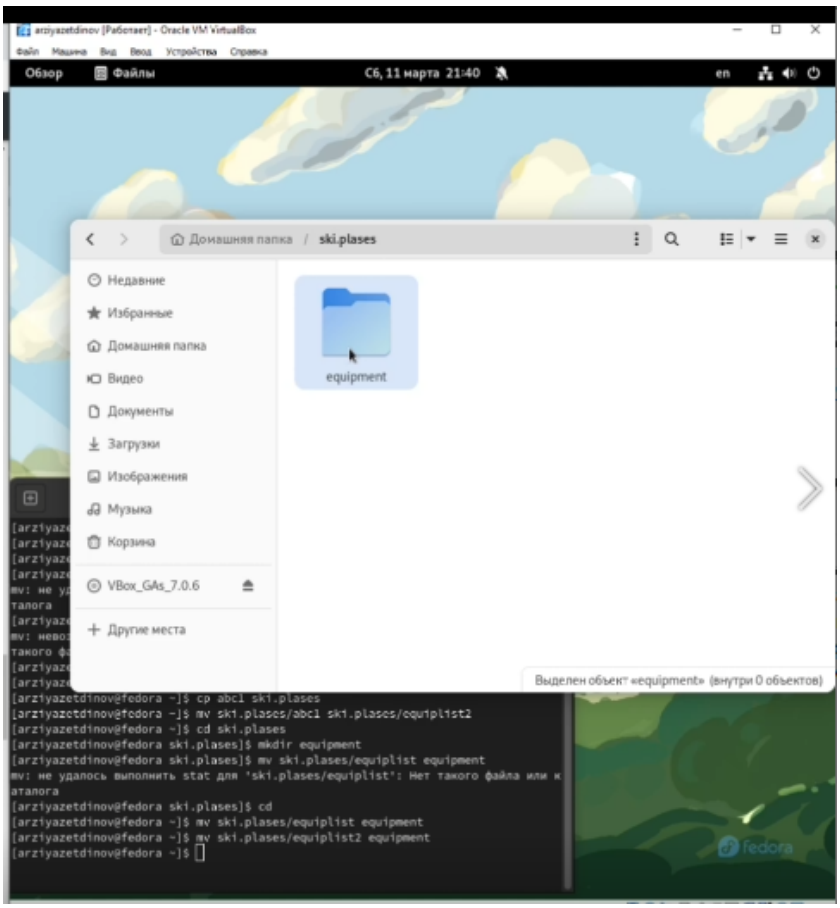


Рис. 3.7: Перемещение файлов в подкаталог equipment

Создаём каталог newdir. Далее перемещаем его в каталог ski.plases под названием plans (рис. 8).

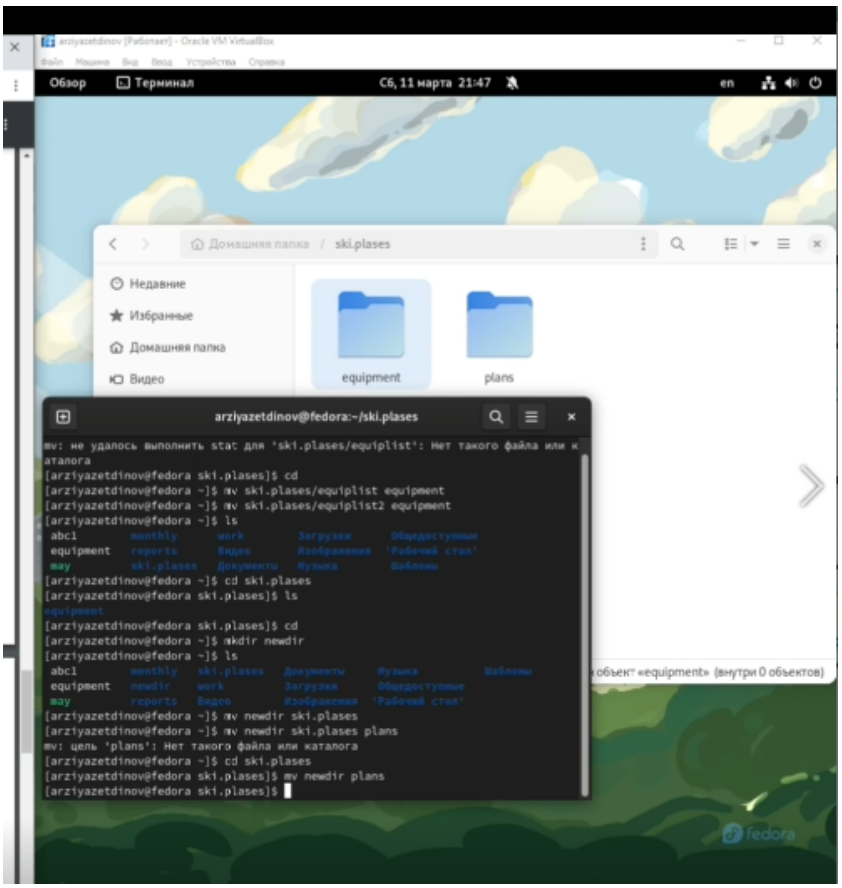


Рис. 3.8: Создание каталога newdir и перемещение его в каталог ski.plases под названием plans

Создаём 2 каталога (australia и play) и 2 файла (my_os и feathers) (рис. 9).

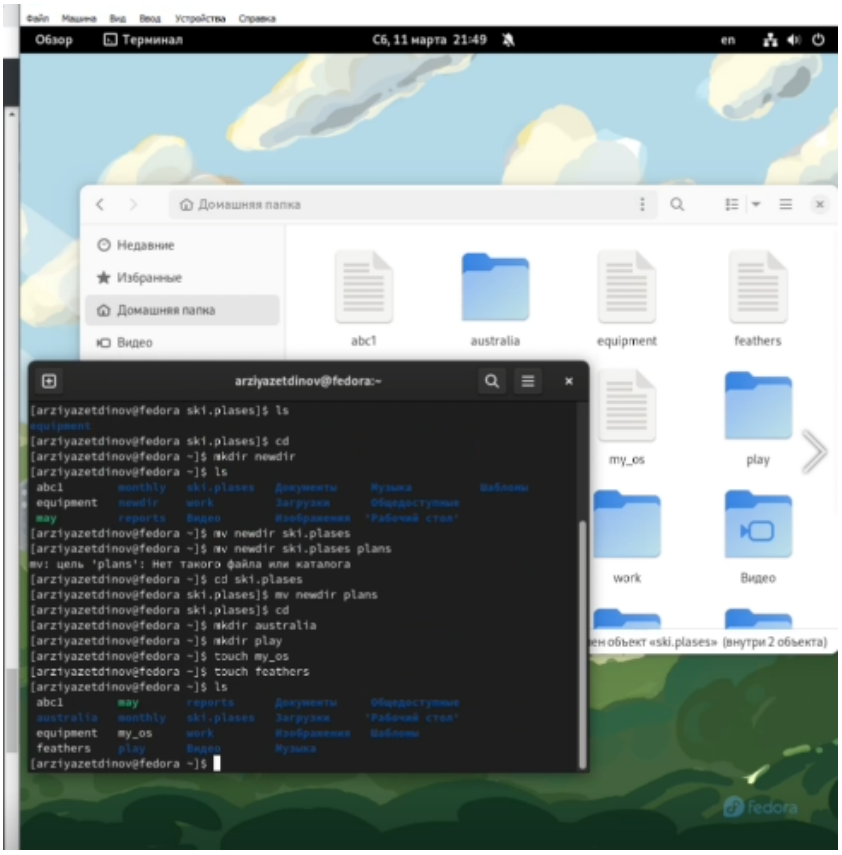


Рис. 3.9: Создание каталогов и файлов

Далее присвоим каждому из каталогов и файлов определённые права доступа (рис.10)

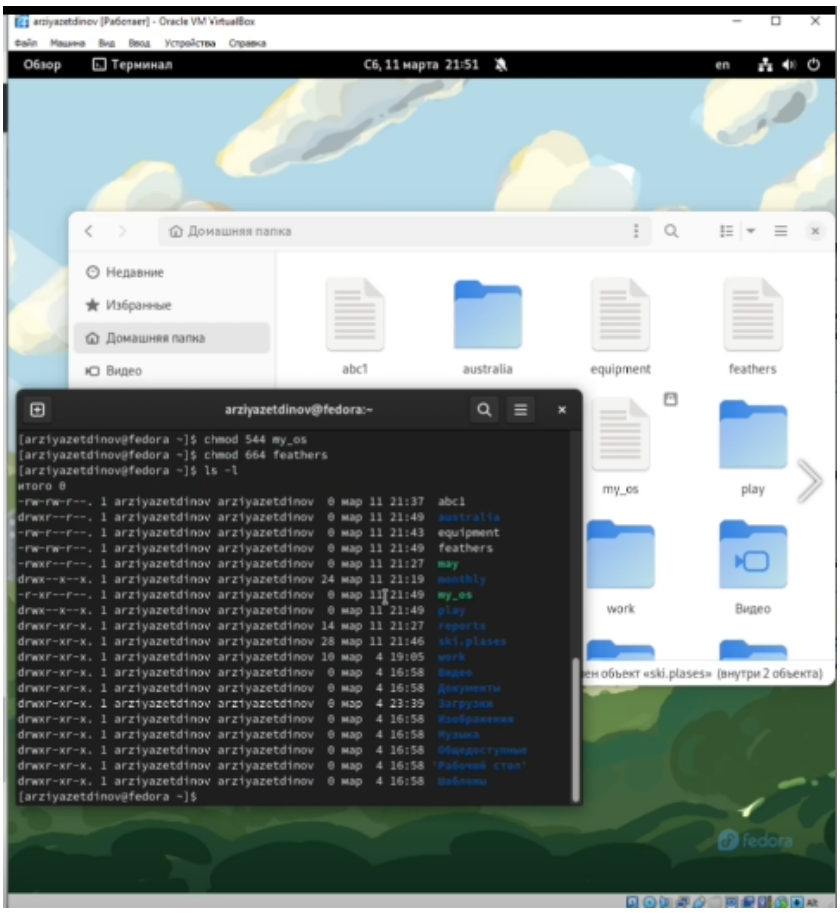


Рис. 3.10: Присваивание прав доступа

Просматриваем содержимое файла passwd с помощью команды cat (рис. 11).

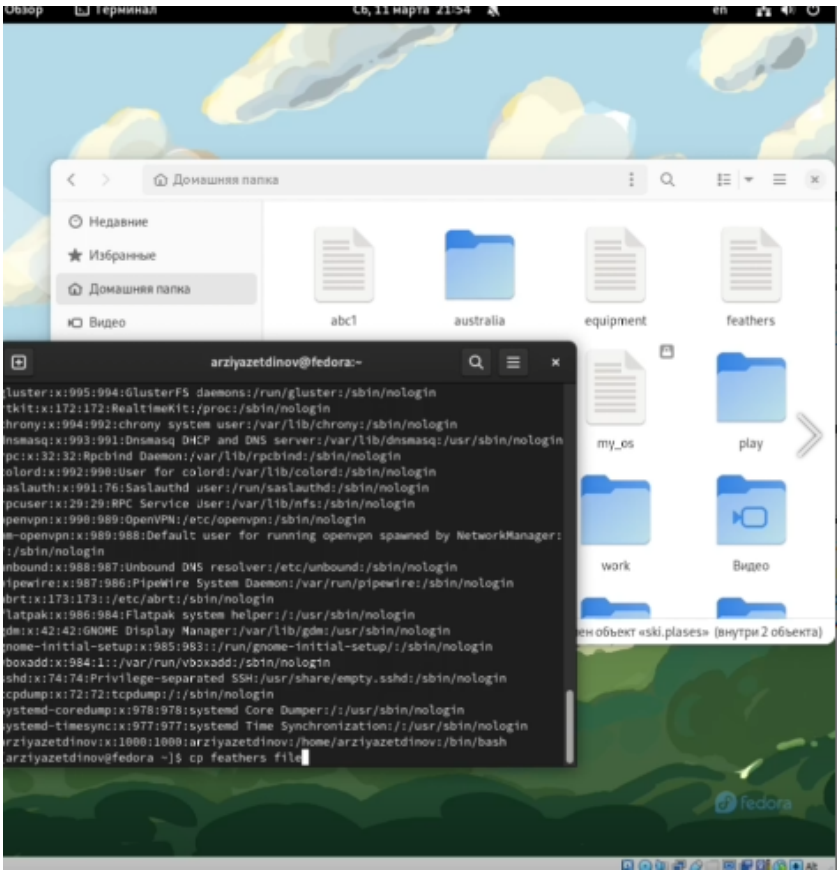


Рис. 3.11: Просмотр содержимого файла passwd

Скопируем файл feathers в file.old (рис. 12).

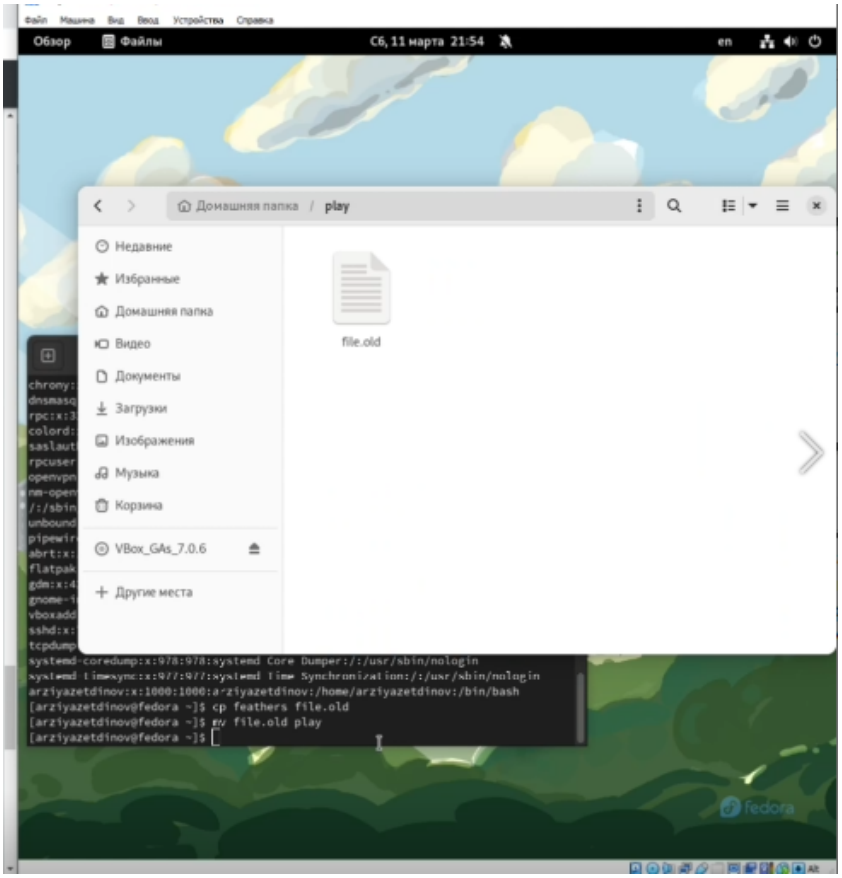


Рис. 3.12: Копирование файла feathers в file.old

Переместим файл file.old в каталог play (рис. 13).

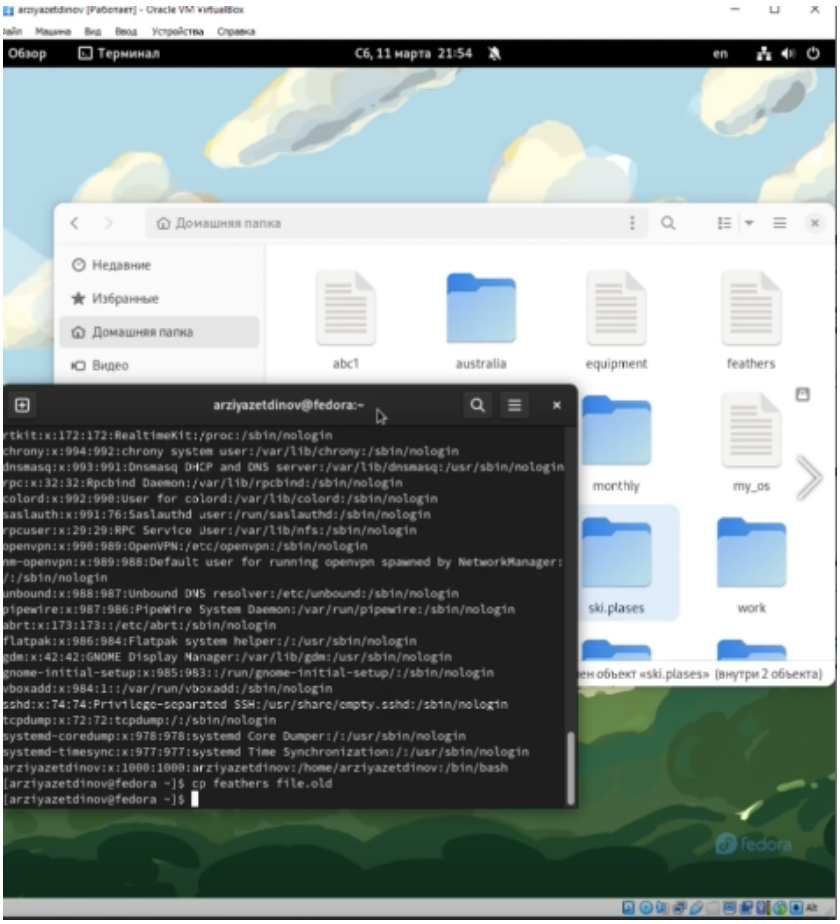


Рис. 3.13: Перемещение файла file.old в каталог play

Скопируем каталог play в каталог fun (рис. 14).

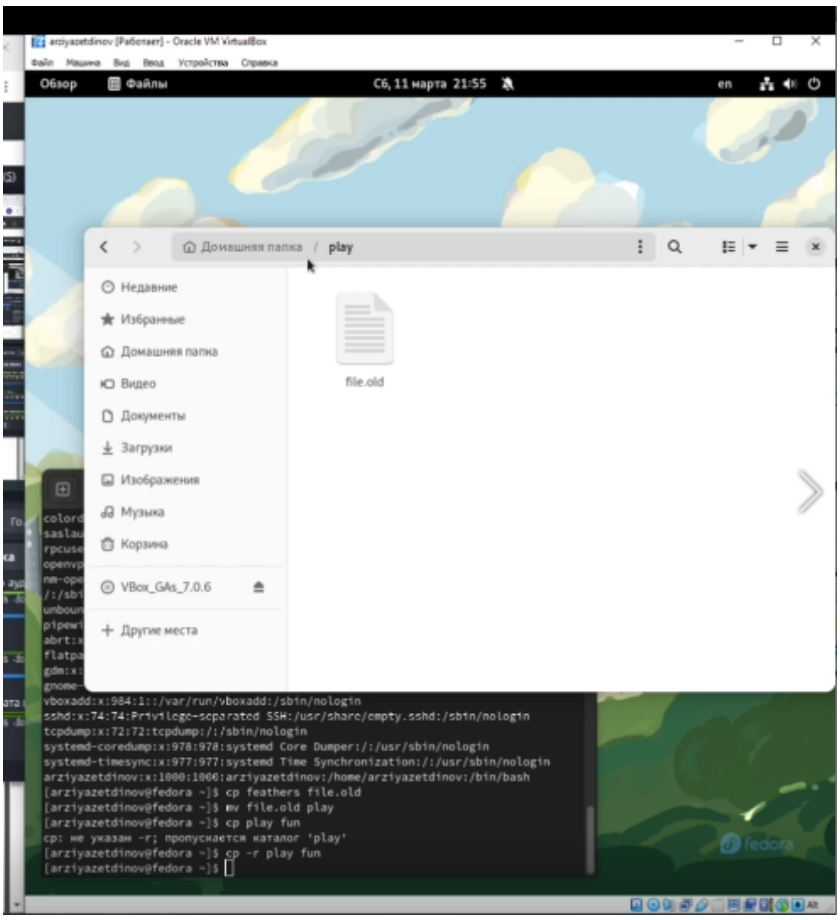


Рис. 3.14: Копирование каталога play в каталог fun

Переместим каталог fun в каталог play и назовём его games (рис. 15).

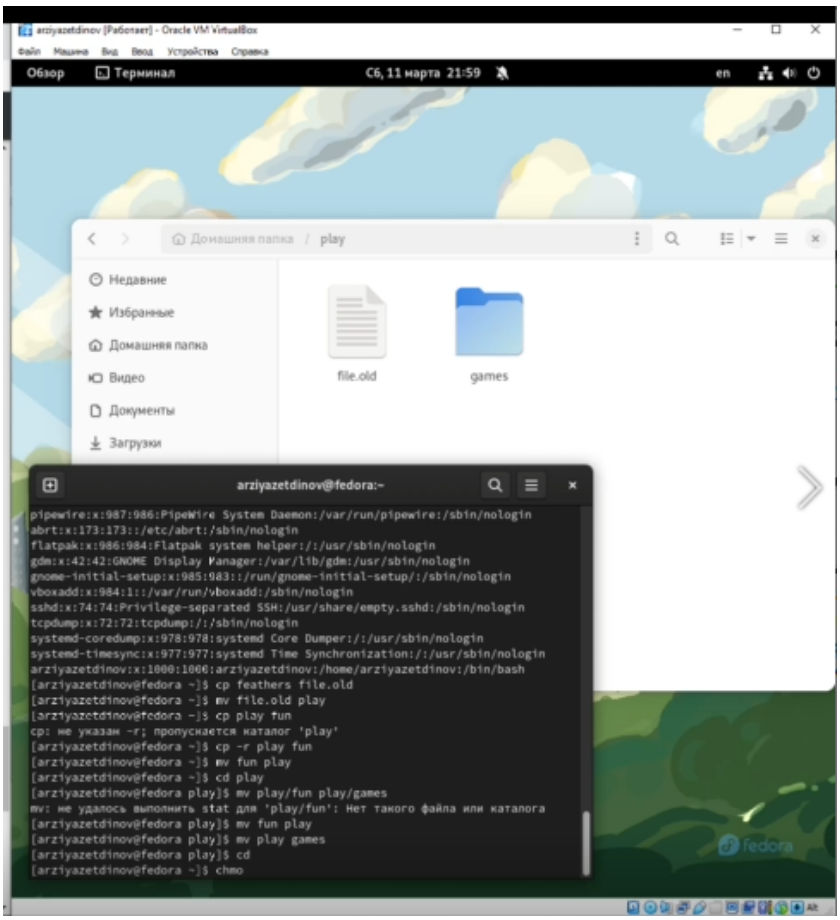


Рис. 3.15: Перемещение каталога fun с изменением название на games

Лишаем владельца файла права на чтение. При попытке просмотреть файл мы получаем отказ в доступе, такой же отказ мы получаем при попытке скопировать этот файл. В конце возвращаем владельцу файла право на чтение (рис. 16).

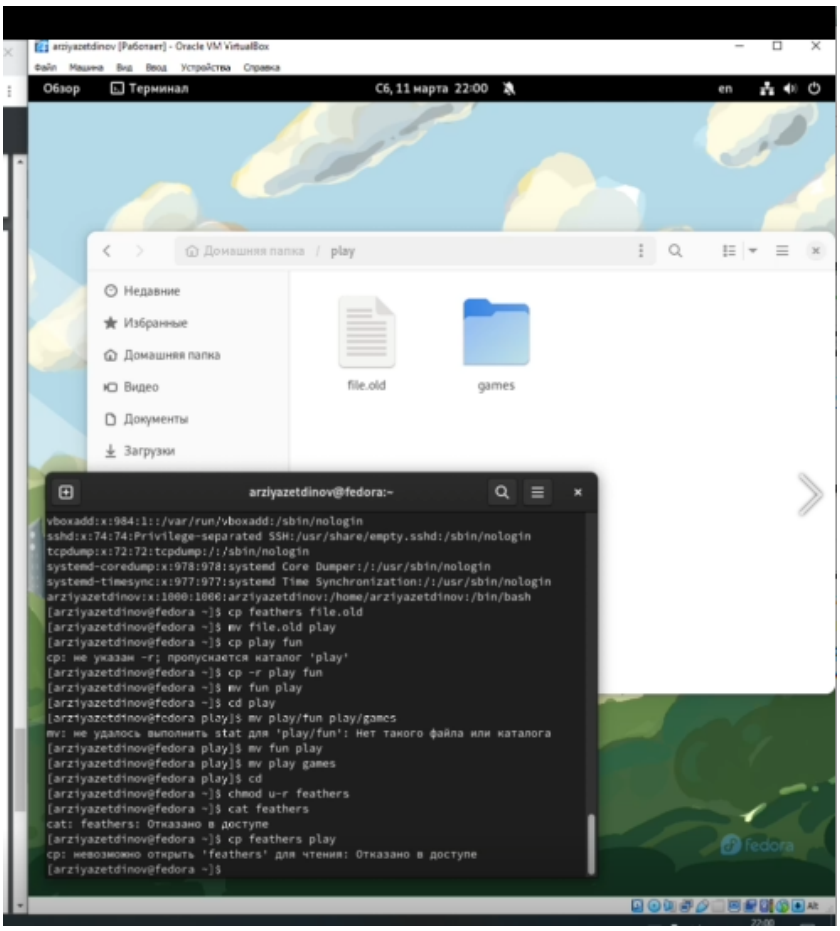


Рис. 3.16: Лишение права на чтение. Попытки чтения и копирования файла. Возвращение права на чтение

Лишаем владельца каталога play права на выполнение. При попытке перейти в этот каталог мы получаем отказ в доступе. Возвращаем владельцу каталога право на выполнение (рис. 17).

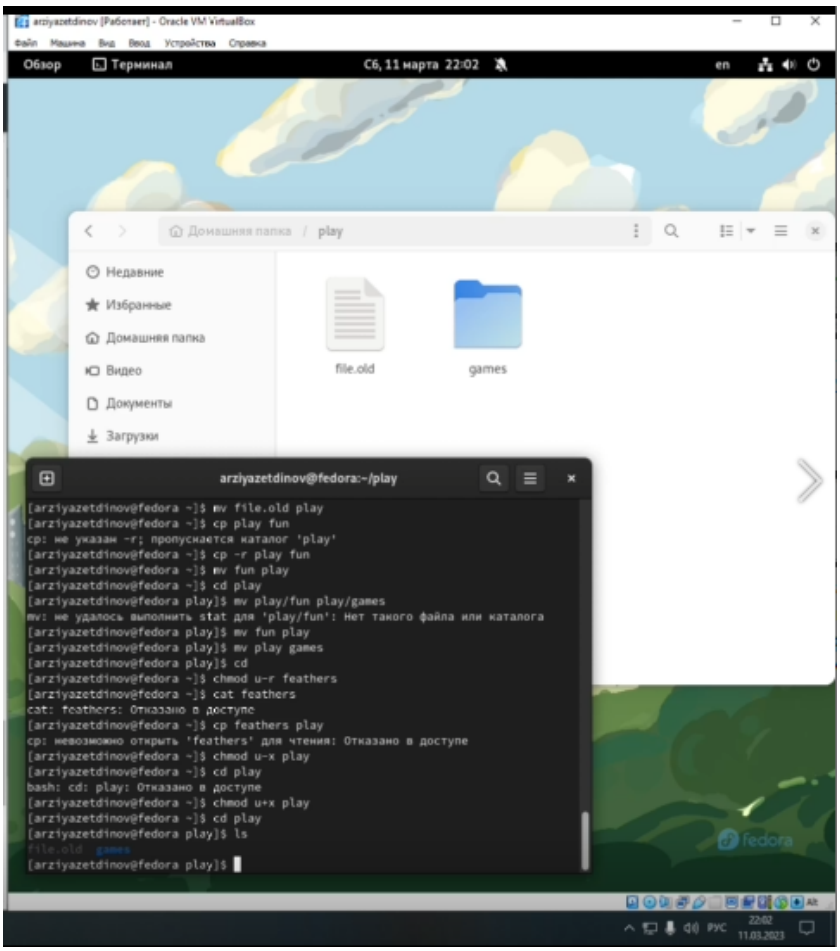


Рис. 3.17: Лишение права на выполнение. Попытка перехода в каталог. Возвращение права на выполнение

Прочитаем с помощью команды man следующие команды: mount, fsck, mkfs, kill. Кратко охарактеризуем эти команды.

Для просмотра используемых в операционной системе файловых систем используется команда mount (рис. 18).

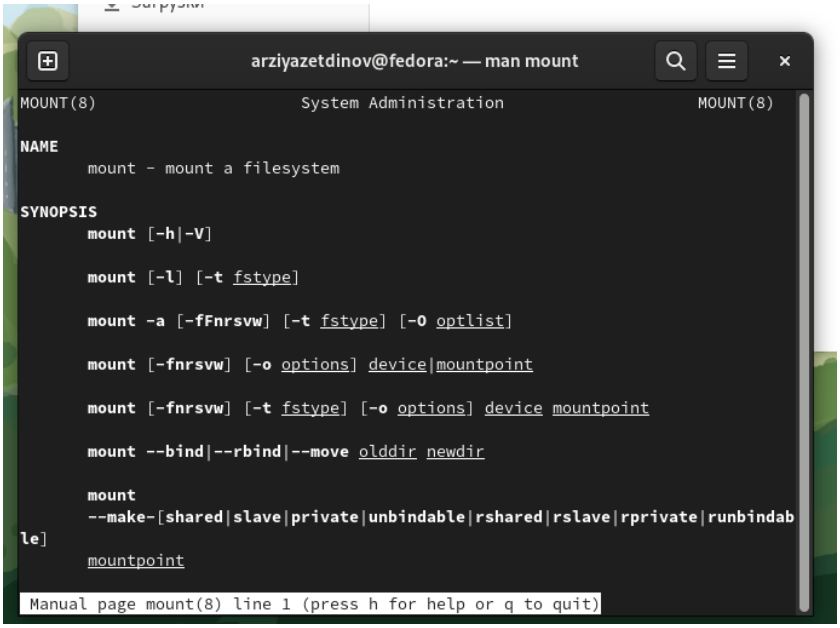


Рис. 3.18: man mount

С помощью команды fsck можно проверить (а в ряде случаев восстановить) целостность файловой системы (рис. 19).

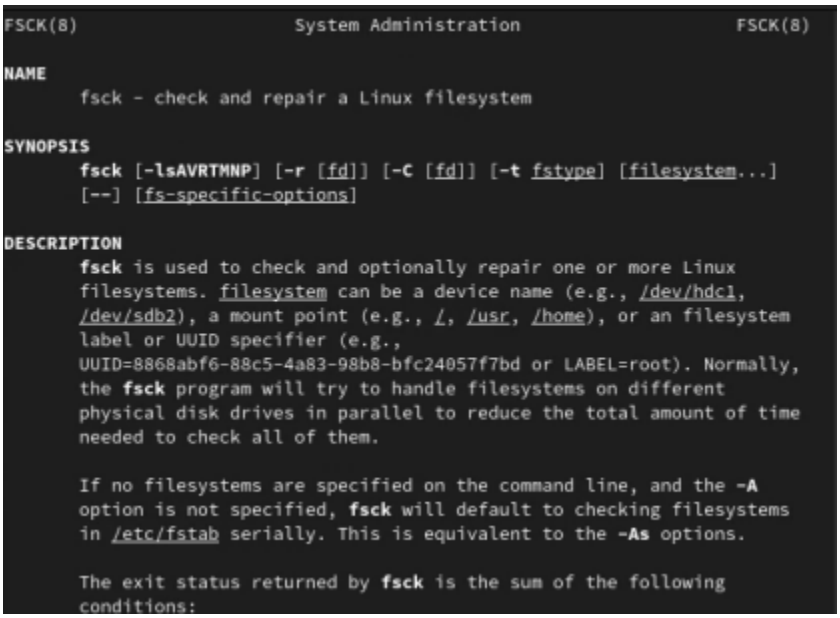


Рис. 3.19: man fsck

mkfs используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента filesys для файловой системы может выступать или название устройства (например, /dev/hda1, /dev/sdb2) или точка монтирования (например, /, /usr, /home) (рис. 20).

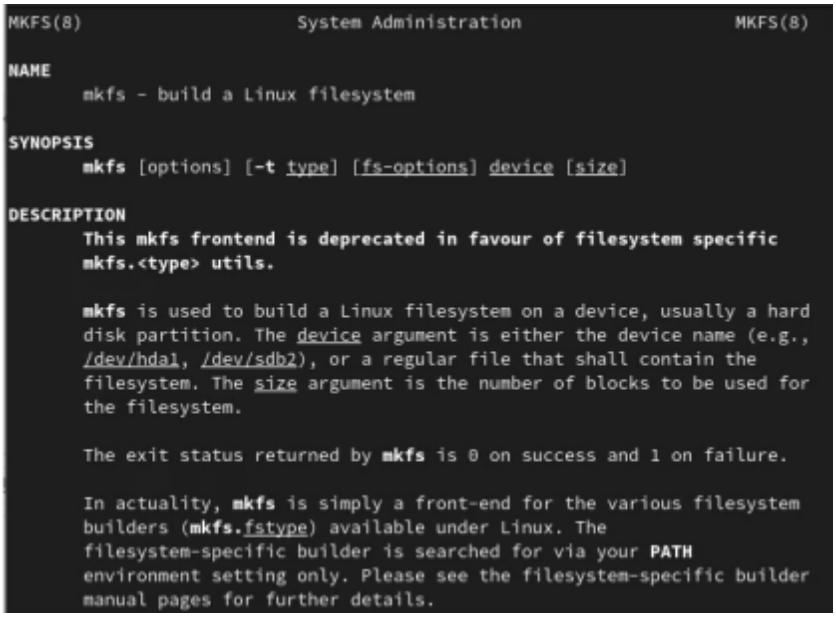
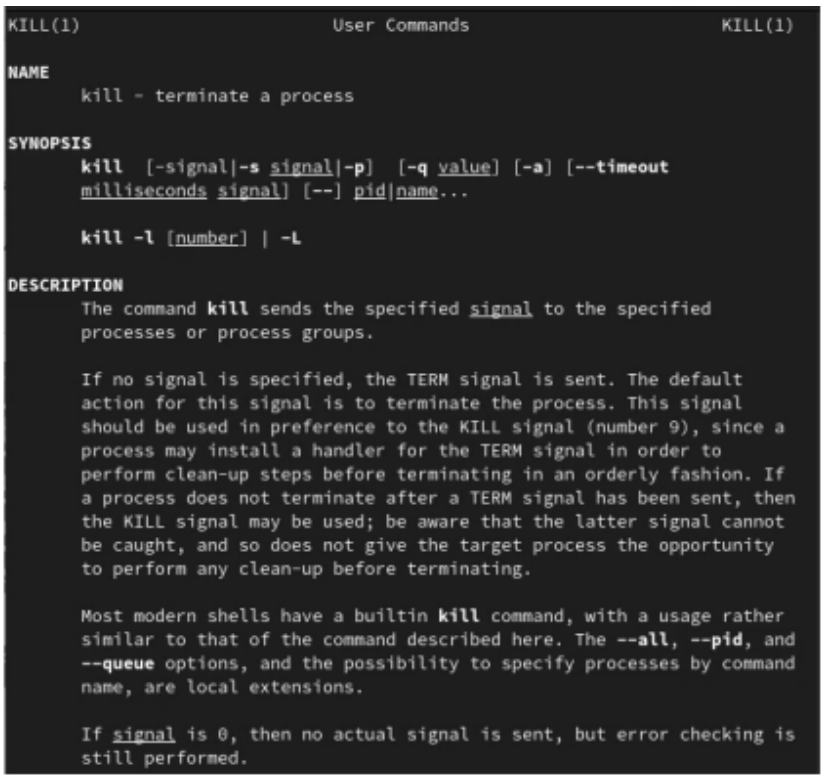


Рис. 3.20: man mkfs

Утилита kill отправляет сигнал процессу(-ам), указанному с помощью каждого из операндов идентификатор_процесса. По умолчанию утилита kill отправляет сигнал SIGTERM, но эту настройку по умолчанию можно переопределить путем определения имени сигнала для отправки (рис. 21).



```
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. 3.21: man kill

4 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 экзабайта.

JFS или Journaled File System была разработана в IBM для AIX UNIX и использовалась в качестве альтернативы для файловых систем ext. Сейчас она используется там, где необходима высокая стабильность и минимальное потребление ресурсов. При разработке файловой системы ставилась цель создать максимально эффективную файловую систему для многопроцессорных компьютеров. Также как и ext, это журналируемая файловая система, но в журнале хранятся только метаданные, что может привести к использованию старых версий файлов после сбоев.

ReiserFS - была разработана намного позже, в качестве альтернативы ext3 с улучшенной производительностью и расширенными возможностями. Она была разработана под руководством Ганса Райзера и поддерживает только Linux.

Из особенностей можно отметить динамический размер блока, что позволяет упаковывать несколько небольших файлов в один блок, что предотвращает фрагментацию и улучшает работу с небольшими файлами. Еще одно преимущество - в возможности изменять размеры разделов на лету. Но минус в некоторой нестабильности и риске потери данных при отключении энергии. Раньше ReiserFS применялась по умолчанию в SUSE Linux, но сейчас разработчики перешли на Btrfs.

XFS - это высокопроизводительная файловая система, разработанная в Silicon Graphics для собственной операционной системы еще в 2001 году. Она изначально была рассчитана на файлы большого размера, и поддерживала диски до 2 Терабайт. Из преимуществ файловой системы можно отметить высокую скорость работы с большими файлами, отложенное выделение места, увеличение разделов на лету и незначительный размер служебной информации.

XFS - журналируемая файловая система, однако в отличие от ext, в журнал записываются только изменения метаданных. Она используется по умолчанию в дистрибутивах на основе Red Hat. Из недостатков - это невозможность уменьшения размера, сложность восстановления данных и риск потери файлов при записи, если будет неожиданное отключение питания, поскольку большинство данных находится в памяти.

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 не 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?

mkfs - позволяет создать файловую систему Linux.

6. Дайте характеристику командам для просмотра текстовых файлов.

Cat - выводит содержимое файла на стандартное устройство вывода

7. Приведите основные возможности команды cp в Linux.

Cp – копирует или перемещает директорию, файлы.

8. Приведите основные возможности команды mv в Linux.

Mv - переименовать или переместить 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.

5 Выводы

В ходе выполнения лабораторной работы мы ознакомились с файловой системой Linux, её структурой, именами и содержанием каталогов. 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.