Лабораторная работа №5

Анализ файловой системы Linux. Команды для работы с файлами и каталогами

Чемоданова Ангелина Александровна

Содержание

# 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: 2.1. 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. 2.2. В домашнем каталоге создайте директорию ~/ski.plases. 2.3. Переместите файл equipment в каталог ~/ski.plases. 2.4. Переименуйте файл ~/ski.plases/equipment в ~/ski.plases/equiplist. 2.5. Создайте в домашнем каталоге файл abc1 и скопируйте его в каталог ~/ski.plases, назовите его equiplist2. 2.6. Создайте каталог с именем equipment в каталоге ~/ski.plases. 2.7. Переместите файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment. 2.8. 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: 3.1. drwxr–r– … australia 3.2. drwx–x–x … play 3.3. -r-xr–r– … my\_os 3.4. -rw-rw-r– … feathers При необходимости создайте нужные файлы.
4. Проделайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды: 4.1. Просмотрите содержимое файла /etc/password. 4.2. Скопируйте файл ~/feathers в файл ~/file.old. 4.3. Переместите файл ~/file.old в каталог ~/play. 4.4. Скопируйте каталог ~/play в каталог ~/fun. 4.5. Переместите каталог ~/fun в каталог ~/play и назовите его games. 4.6. Лишите владельца файла ~/feathers права на чтение. 4.7. Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat? 4.8. Что произойдёт, если вы попытаетесь скопировать файл ~/feathers? 4.9. Дайте владельцу файла ~/feathers право на чтение. 4.10. Лишите владельца каталога ~/play права на выполнение. 4.11. Перейдите в каталог ~/play. Что произошло? 4.12. Дайте владельцу каталога ~/play право на выполнение.
5. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

# 3 Теоретическое введение

Файловая система в Linux состоит из фалов и каталогов. Каждому физическому носителю соответствует своя файловая система. Существует несколько типов файловых систем. Перечислим наиболее часто встречающиеся типы: – ext2fs (second extended filesystem); – ext2fs (third extended file system); – ext4 (fourth extended file system); – ReiserFS; – xfs; – fat (file allocation table); – ntfs (new technology file system). Некоторые команды для работы с файлами и каталогами. Для создания текстового файла можно использовать команду touch. Для просмотра файлов небольшого размера можно использовать команду cat. Для просмотра файлов постранично удобнее использовать команду less. Следующие клавиши используются для управления процессом просмотра: – Space — переход к следующей странице, – ENTER — сдвиг вперёд на одну строку, – b — возврат на предыдущую страницу, – h — обращение за подсказкой, – q — выход из режима просмотра файла. Команда head выводит по умолчанию первые 10 строк файла.

# 4 Выполнение лабораторной работы

Выполним все примеры, приведенные в первой части описания лабораторной работы. (рис. [1](#fig:001)) (рис. [2](#fig:002)).(рис. [3](#fig:003)).(рис. [4](#fig:004)).(рис. [5](#fig:005)).(рис. [6](#fig:006)).

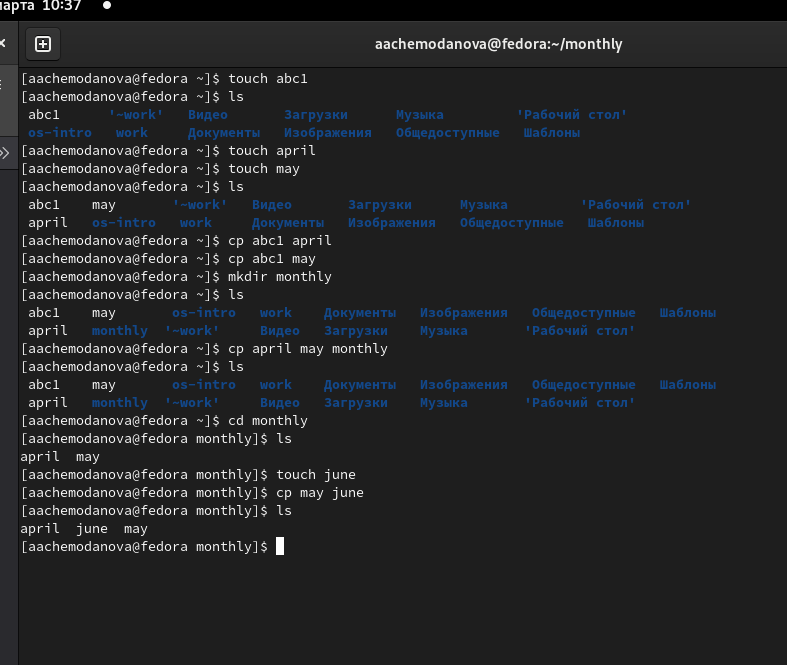


Figure 1: Выполнение примеров

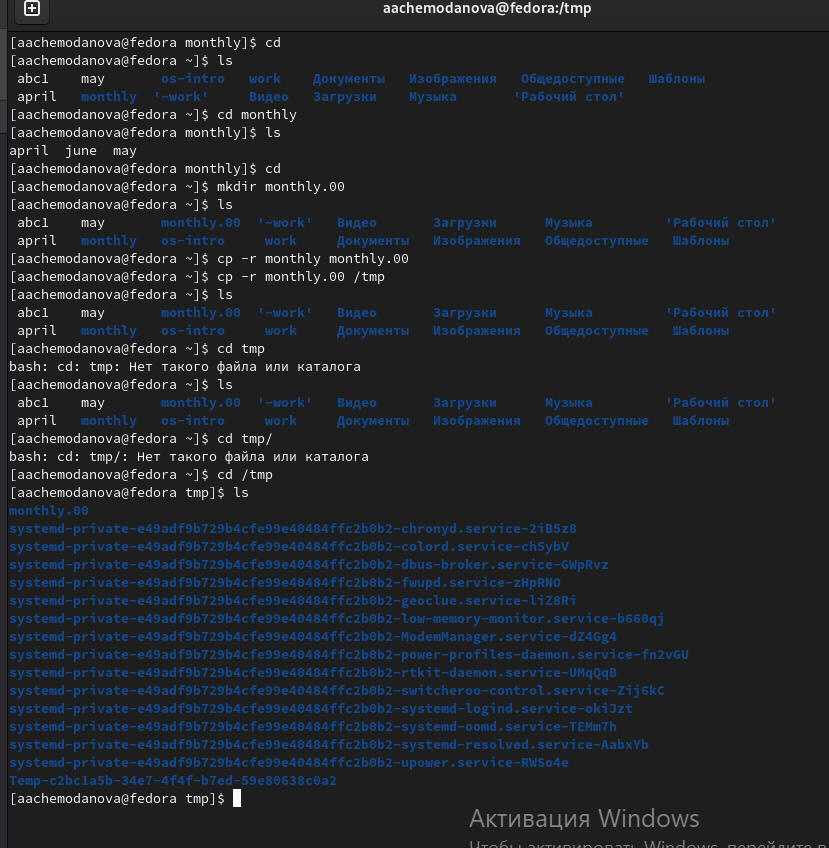


Figure 2: Выполнение примеров

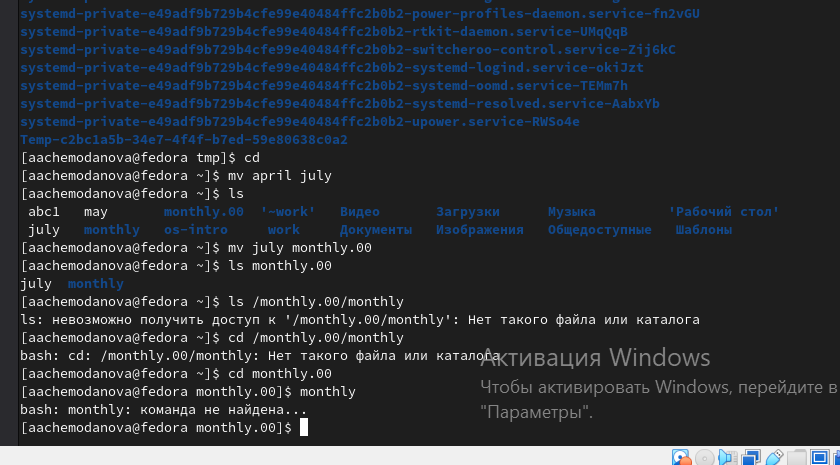


Figure 3: Выполнение примеров

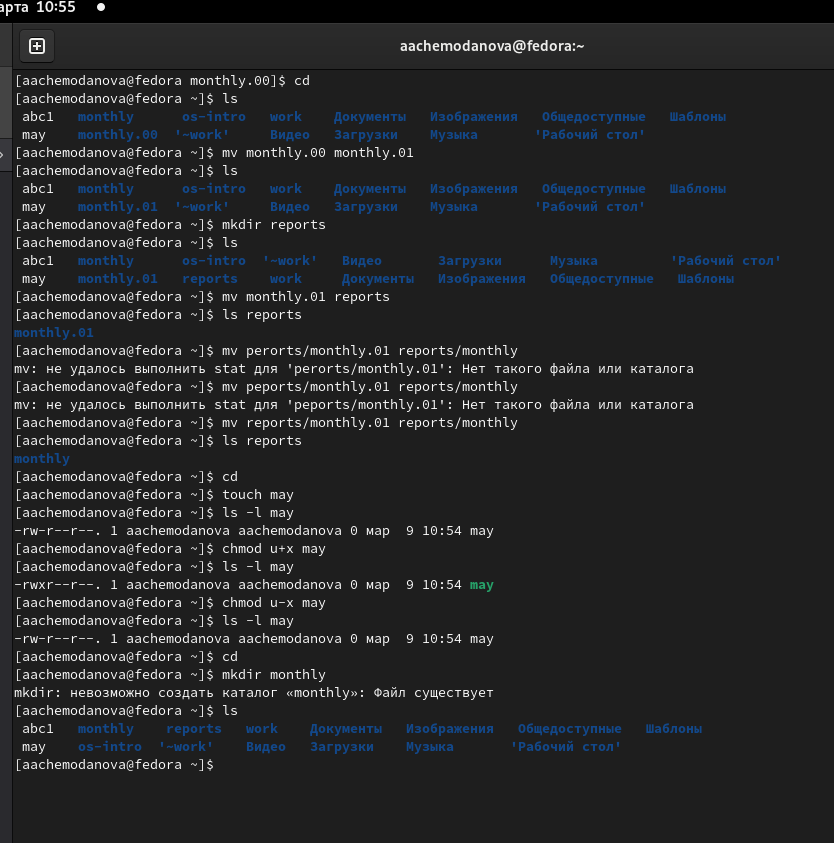


Figure 4: Выполнение примеров

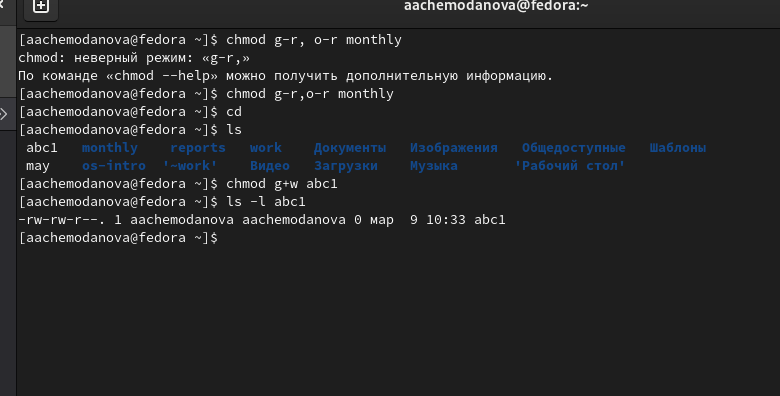


Figure 5: Выполнение примеров

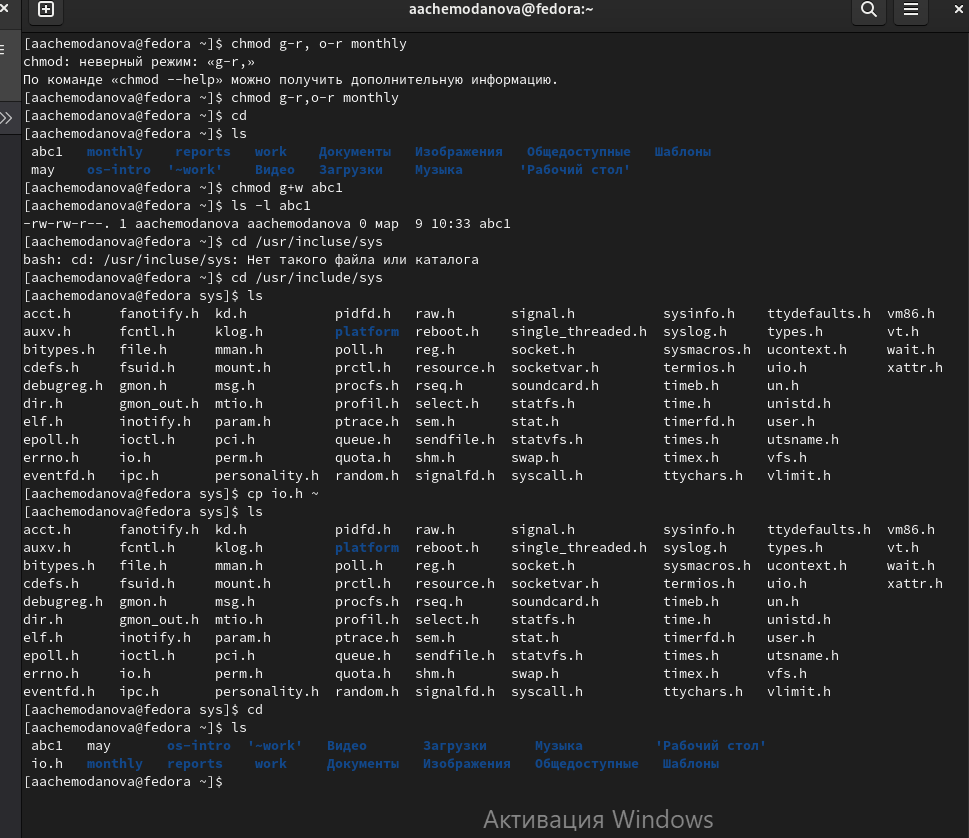


Figure 6: Выполнение примеров

Скопируем файл /usr/include/sys/io.h в домашний каталог при помощи команды cp io.h ~. (рис. [7](#fig:007)).

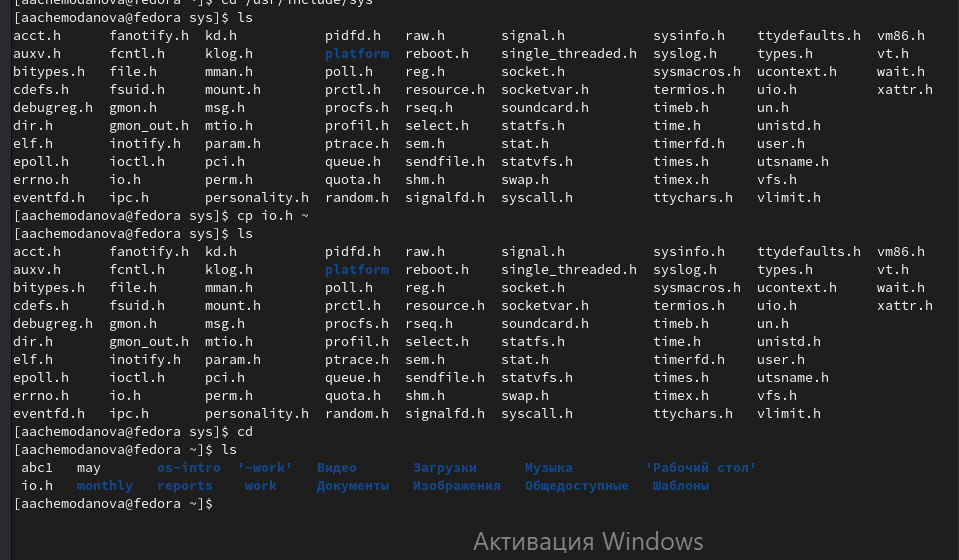


Figure 7: Копирование файла

И назовем его equipment при помощи mv io.h equipment. В домашнем каталоге создадим директорию ~/ski.plases при помощи mkdir. Переместим файл equipment в каталог ~/ski.plases при помощи mv. Переименуем файл ~/ski.plases/equipment в ~/ski.plases/equiplist при помощи mv. Создадиим в домашнем каталоге файл abc1 и скопируйте его в каталог ~/ski.plases при помощи mv, назовите его equiplist2 при помощи mv. Создадим каталог с именем equipment в каталоге ~/ski.plases при помощи mkdir. Переместим файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment при помощи mv. Создадим при помощи mkdir и переместим каталог ~/newdir в каталог ~/ski.plases при помощи mv и назовем его plans при помощи mv.(рис. [8](#fig:008)).

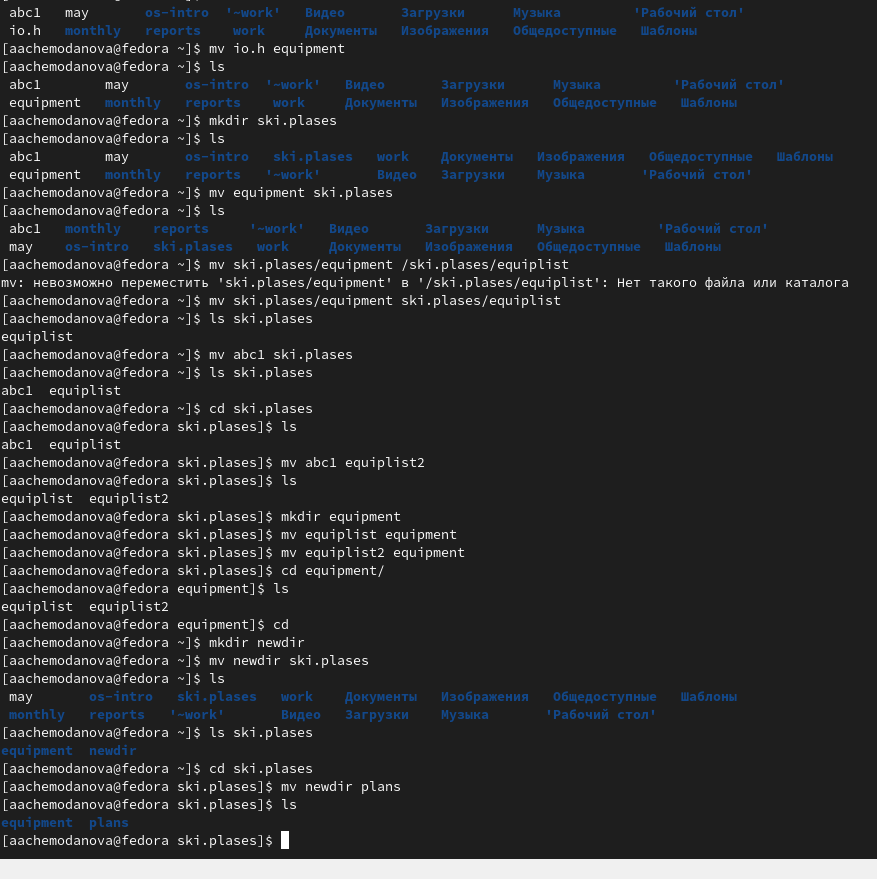


Figure 8: Полное выполнение второго задания

Теперь присвоим определенным файлам выделенные права доступа. (рис. [9](#fig:009))(рис. [10](#fig:010)).

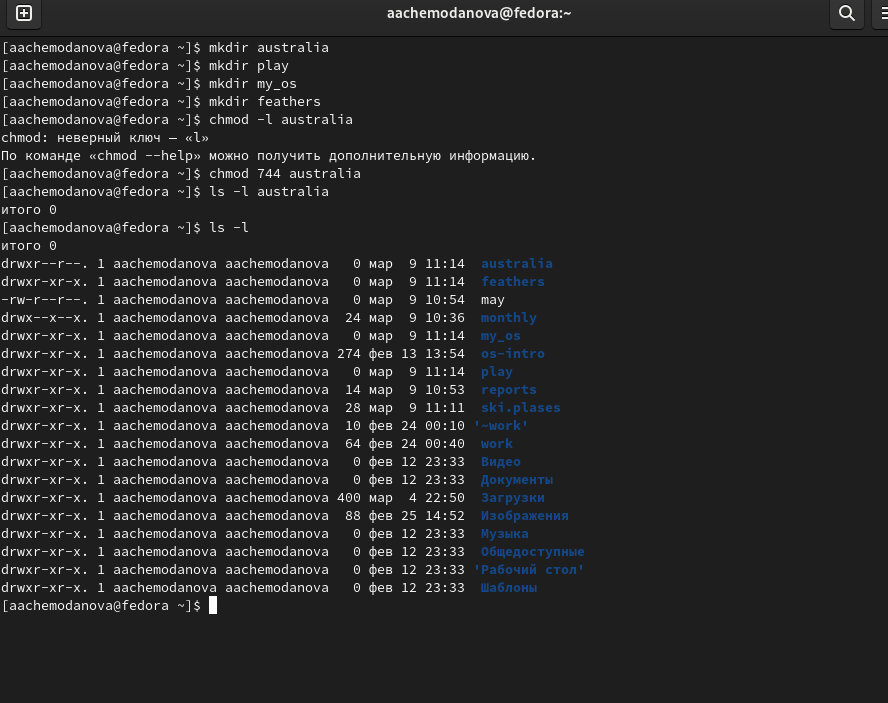


Figure 9: Присвоение прав australia

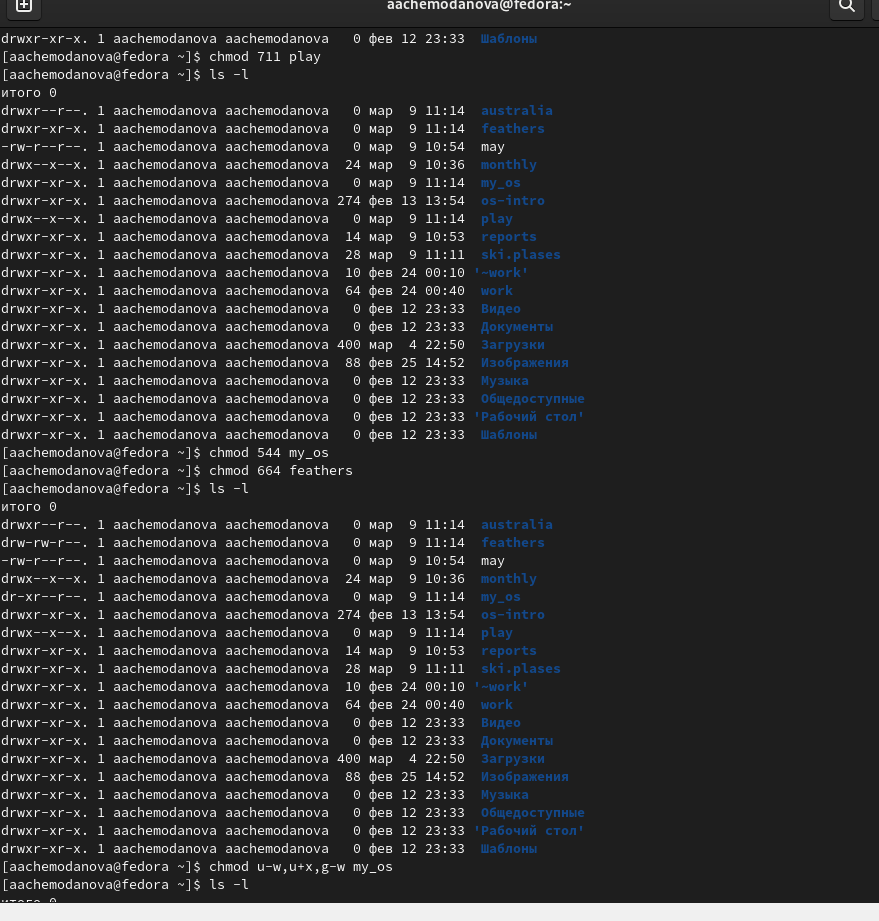


Figure 10: Присвоение прав play, my\_os, feathers

Просмотрим содержимое файла /etc/password при помощи команды cat, но у меня не нашлось такого файла. (рис. [11](#fig:011)).

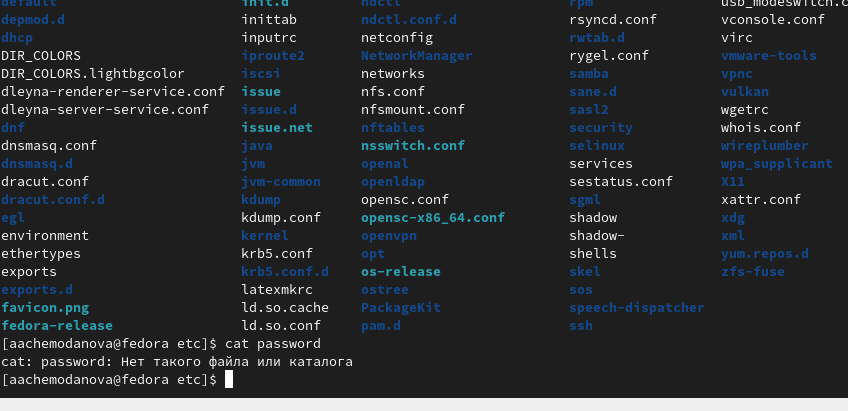


Figure 11: Файл password

Скопируем файл ~/feathers в файл ~/file.old при помощи cp. Переместим файл ~/file.old в каталог ~/play при помощи mv. Скопируем каталог ~/play в каталог ~/fun cp -r. Переместим каталог ~/fun в каталог ~/play mv и назовем его games mv. (рис. [12](#fig:012)).

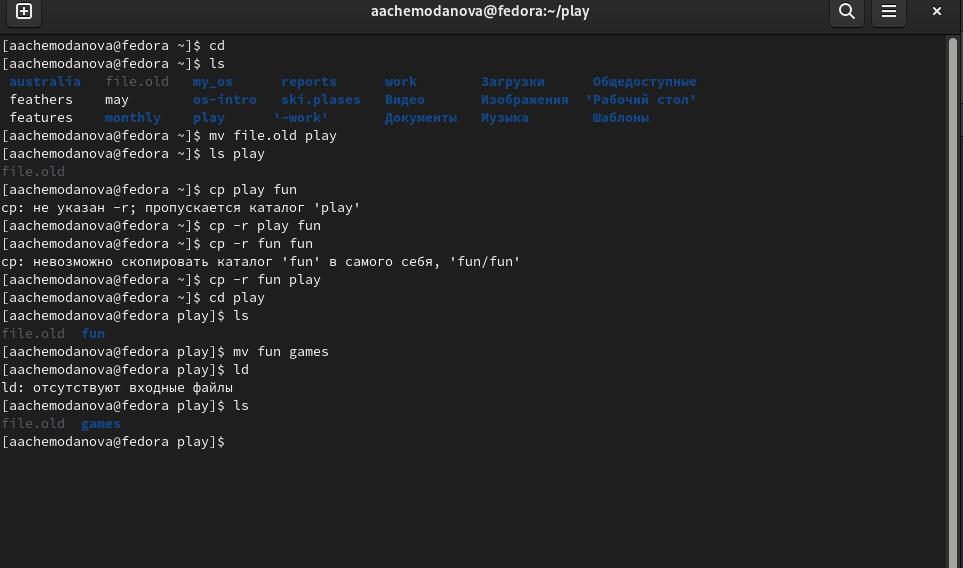


Figure 12: Работа с каталогами play и fun

Лишим владельца файла ~/feathers права на чтение при помощи chmod u-r.(рис. [13](#fig:013)).

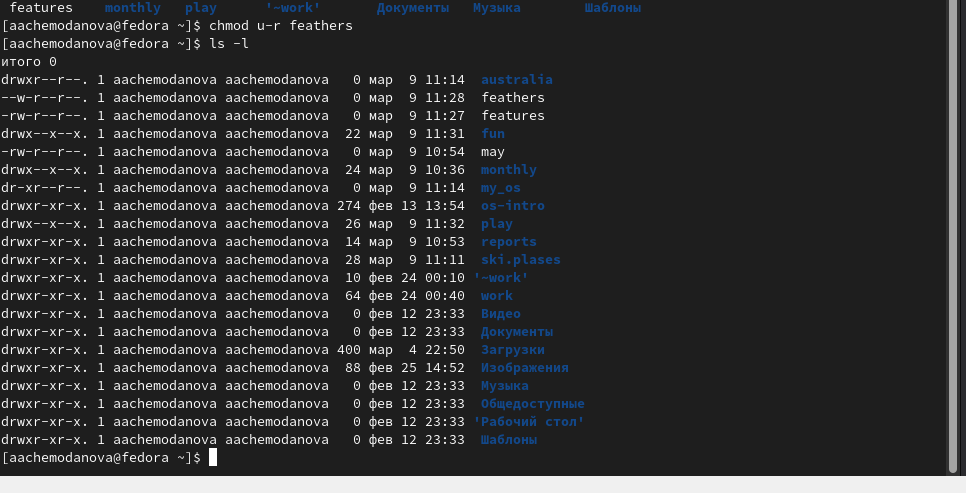


Figure 13: Лишение владельца файла ~/feathers права на чтение

Попытаемся просмотреть файл ~/feathers командой cat, но у нас не получится. Попытаемся скопировать файл ~/feathers при помощи ср. Дадим владельцу файла ~/feathers право на чтение. (рис. [14](#fig:014)).

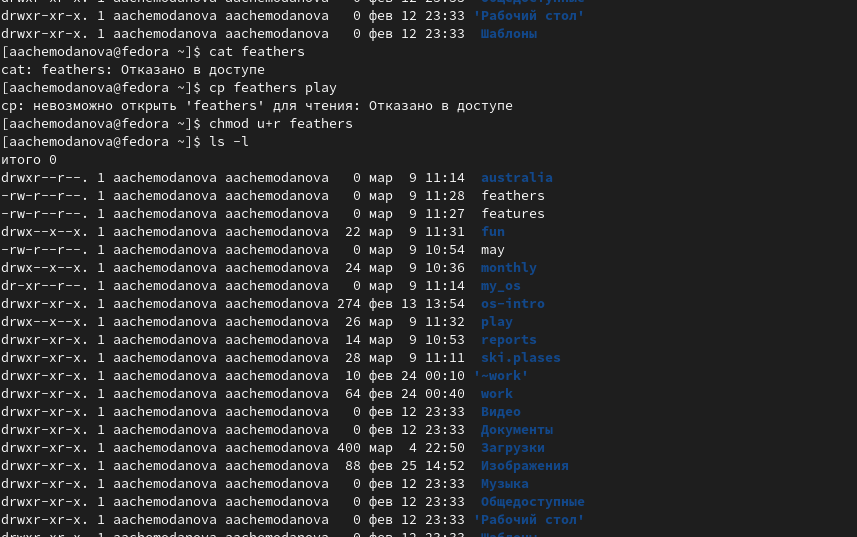


Figure 14: Взаимодействия с файлом ~/feathers

Лишим владельца каталога ~/play права на выполнение. Попробуем перейти в каталог ~/play, у нас не получится. Дадим владельцу каталога ~/play право на выполнение. (рис. [15](#fig:015)).

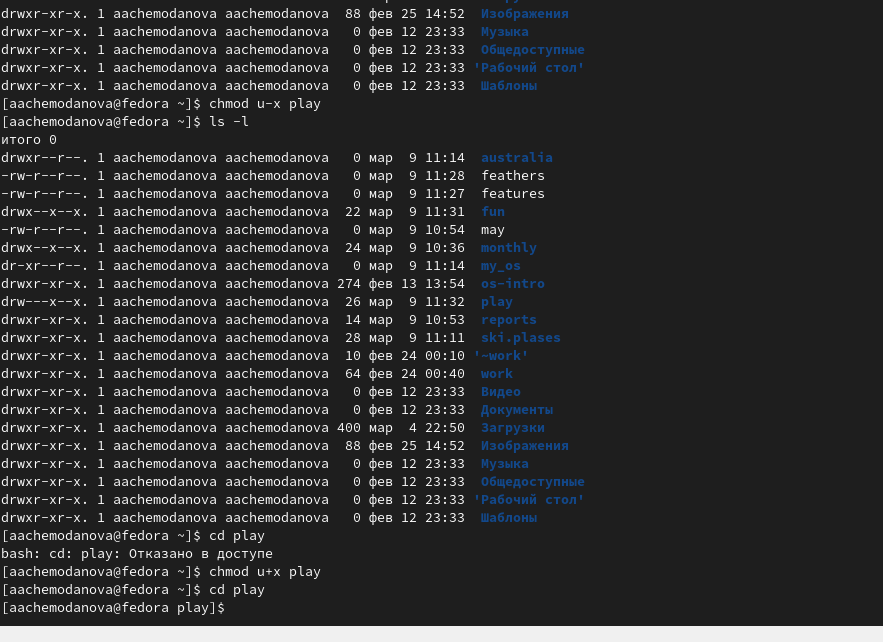


Figure 15: Взаимодействия с каталогом ~/play

Прочитаем man по командам mount, fsck, mkfs, kill и кратко их охарактеризуем. mount - команда для монтирования файловой системы в Linux. Команда позволяет присоединить хранящиеся на разных носителях файлы к общему дереву каталогов. fsck - команда, которая позволяет проверять согласованность и интерактивное исправление в одной или нескольких файловых систсемах. То есть она проверяет файловую систему на наличие ошибок и нерешенных проблем. mkfs - команда, использующаяся для управления устройствами хранения файлов в Linux. Она может создать файловую систему на некотором устройстве, например, в разделе жесткого диска. kill - команда, которая посылает сигналы процессам по их идентификаторам. Обычно исользуется для устранения процессов. (рис. [16](#fig:016))

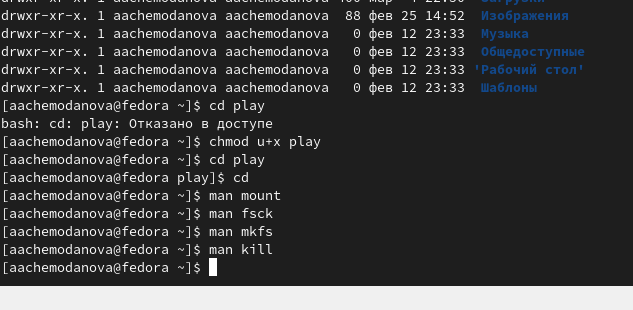


Figure 16: man по командам mount, fsck, mkfs, kill

# 5 Выводы

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.

# 6 Контрольные вопросы

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

С помощью команды df –T, мы можем познакомиться с файловыми системами · Файловая система devtmpfs была разработана для решения проблемы с доступностью устройств во время загрузки. Ядро создает файлы устройств по мере надобности, а также уведомляет менеджер udevd о том, что доступно новое устройство. После получения такого сигнала менеджер udevd не создает файлы устройств, а выполняет инициализацию устройства и отправляет уведомление процессу. Кроме того, он создает несколько символических ссылок в каталоге /dev для дальнейшей идентификации устройств. · Tmpfs — временное файловое хранилище в Unix . Предназначена для монтирования файловой системы, но размещается в ОЗУ вместо физического диска. Подобная конструкция является RAM диском.Все данные в Tmpfs являются временными, в том смысле, что ни одного файла не будет создано на жёстком диске. После перезагрузки все данные, содержащиеся в Tmpfs, будут утеряны. · Fourth extended file system , сокр. ext4, или ext4fs — журналируемая ФС, используемая в ОС с ядром Linux. Основана на ФС ext3, ранее использовавшейся по умолчанию во многих дистрибутивах GNU/Linux. · Распределенная сетевая файловая система AFS (Andrew File System) внедрена с целью создания единого файлового пространства пользователей при работе на различных фермах или рабочих станциях ОИЯИ, где домашний каталог пользователя определен как каталог в -AFS. Использование системы AFS позволяет пользователям осуществлять прямой доступ к файловому пространству других организаций, где эта система используется и где пользователь имеет регистрацию в AFS.

1. Приведите общую структуру файловой системы и дайте характеристику каждой ди- ректории первого уровня этой структуры.

Все каталоги можно разделить на две группы: для статической (редко меняющейся) информации – /bin, /usr и динамической (часто меняющейся) информации – /var, /tmp. Исходя из этого администраторы могут разместить каждый из этих каталогов на собственном носителе, обладающем соответствующими характеристиками. · Корневой каталог. Корневой каталог / является основой любой ФС UNIX. Все остальные каталоги и файлы располагаются в рамках структуры (дерева), порождённой корневым каталогом, независимо от их физического местонахождения. · /bin. В этом каталоге находятся часто употребляемые команды и утилиты системы общего пользования. Сюда входят все базовые команды, доступные даже если была примонтирована только корневая файловая система. Примерами таких команд являются:Ls,cp и т.д. · /boot. Директория содержит всё необходимое для процесса загрузки операционной системы: программу-загрузчик, образ ядра операционной системы и т.п.. · /dev. Каталог содержит специальные файлы устройств, являющиеся интерфейсом доступа к периферийным устройствам. Наличие такого каталога не означает, что специальные файлы устройств нельзя создавать в другом месте, просто достаточно удобно иметь один каталог для всех файлов такого типа. · /etc. В этом каталоге находятся системные конфигурационные файлы. В качестве примеров можно привести файлы /etc/fstab, содержащий список монтируемых файловых систем, и /etc/ resolv.conf, который задаёт правила составления локальных DNS-запросов. Среди наиболее важных файлов – скрипты инифиализации и деинициализации системы. В системах, наследующих особенности UNIX System V, для них отведены каталоги с /etc/rc0.d по /etc/rc6.d и общий для всех файл описания – /etc/inittab. · /home (необязательно). Директория содержит домашние директории пользователей. Её существование в корневом каталоге не обязательно и её содержимое зависит от особенностей конкретной UNIX-подобной операционной системы. · /lib. Каталог для статических и динамических библиотек, необходимых для запуска программ, находящихся-в директориях/bin,/sbin. · /mnt. Стандартный каталог для временного монтирования файловых систем – например, гибких и флэш-дисков, компакт-дисков и т. п. · /root (необязательно). Директория содержит домашюю директорию суперпользователя. Её существование в корневом каталоге не обязательно. · /sbin. В этом каталоге находятся команды и утилиты для системного администратора. Примерами таких команд являются: route, halt, init и др. Для аналогичных целей применяются директории /usr/sbin и /usr/local/sbin. · /usr. Эта директория повторяет структуру корневой директории – содержит каталоги /usr/ bin, /usr/lib, /usr/sbin, служащие для аналогичных целей. Каталог /usr/include содержит заголовочные файлы языка C для всевозможные библиотек, расположенных в системе. · /usr/local является следующим уровнем повторения корневого каталога и служит для хранения программ, установленных администратором в дополнение к стандартной поставке операционной системы. · /usr/share хранит неизменяющиеся данные для установленных программ. Особый интерес представляет каталог /usr/share/doc, в который добавляется документация ко всем установленным программам. · /var, /tmp. Используются для хранения временных данных процессов – системных и пользовательских соответственно.

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

С помощью команды cd мы переходим в каталог, в котором находится файл. С помощью less мы открываем этот файл.

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

Основные причины нарушения целостности файловой системы: · Из-за прерывания операций ввода-вывода выполняемых непосредственно с диском; · Сбоя питания; · Краха ОС; · Нарушения работы дискового КЭШа; Устранение поврежденных файлов:В большинстве случаев, проверка файловой системы способна обнаружить и выполнить ремонт такой ошибки автоматически, и после завершения процесс начальной загрузки продолжится как обычно. Если проблема файловой системы более серьезна, проверка файловой системы не может решить проблему автоматически. В этом случае процесс надо будет запустить вручную.

1. Как создаётся файловая система?

Обычно при установке Linux создание файловых систем - компетенция инсталлятора, который осуществляет его с некоторыми опциями по умолчанию. Изменить характеристики, определенные для файловой системы при ее создании, невозможно без повторного выполнения этого процесса . Файловая система Ext2fs может быть создана любой из следующих команд - /sbin/mke2fs, / sbin/mkfs, /sbin/mkfs.ext2 с указанием файла устройства в качестве аргумента. Для создания XFS -mkfs.xfs (из пакета xfsprogs). Для создания файловой системы ext3fs -mke2fs с опцией j. Файловая система ReiserFS - /sbin/mkreiserfs из пакета reiserfsprogs.

1. Дайте характеристику командам для просмотра текстовых файлов.

Для просмотра небольших файлов -cat. (cat имя-файла) · Для просмотра больших файлов-less . (less имя-файла) · Для просмотра начала файла-head. По умолчанию она выводит первые 10 строк файла. (head [-n] имя-файла), n — количество выводимых строк. · Команда tail . выводит несколько (по умолчанию 10) последних строк файла. (tail [-n] имя-файла),n — количество выводимых строк.

1. Приведите основные возможности команды cp в Linux.

При помощи команды cp осуществляется копирование файлов и каталогов (cp[-опции] исходный\_файл целевой\_файл) Возможности команды ср: · копирование файла в текущем каталоге · копирование нескольких файлов в каталог · копирование файлов в произвольном каталоге · опция i в команде cp поможет избежать уничтожения информации в случае, если на место целевого файла вы поставите имя уже существующего файла(т.е. система попросит подтвердить, что вы хотите перезаписать этот файл) · Команда cp с опцией r (recursive) позволяет копировать каталоги вместе с входящими в них файлами и каталогами.

1. Приведите основные возможности команды mv в Linux.

Команды mv и mvdir предназначены для перемещения и переименования файлов и каталогов. (mv [-опции] старый\_файл новый\_файл) Для получения предупреждения перед переписыванием файла стоит использовать опцию i.

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

Права доступа - совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носителям, процессам и другим ресурсам) установленных правовыми документами или собственником, владельцем информации. Для изменения прав доступа к файлу или каталогу используется команда-chmod.( chmod режим имя\_файла) Права доступа к файлу может поменять только владелец и администратор. Режим (в формате команды) имеет следующую структуру 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 · (others) все остальные.