Лабораторная работа №5

Операциооные системы

Пашаев Юсиф Юнусович

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 Выполнение лабораторной работы

1. Выполним примеры, приведённые в первой части описания лабораторной работы.



Figure 1: Выполнение примеров

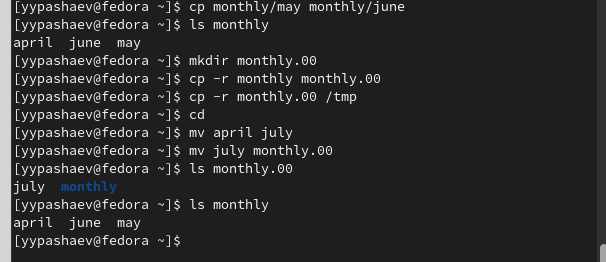


Figure 2: Выполнение примеров

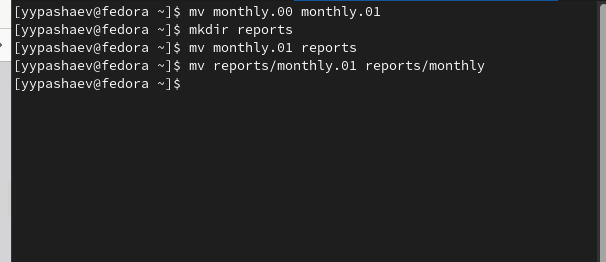


Figure 3: Выполнение примеров

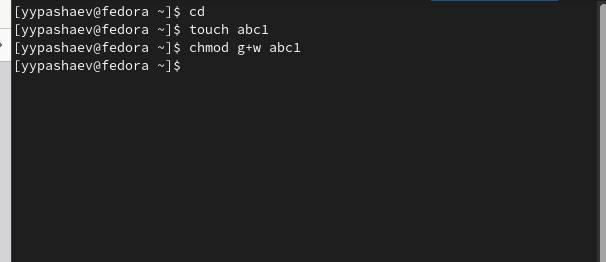


Figure 4: Выполнение примеров

1. В домашнем каталоге создаем директорию ski.plases. и перемещаем в него файл equipment. Переименовываем файл equipment в equiplist. После этого создаем в домашнем каталоге файл abc1 и копируем его в каталог ski.plases. и переименовываем в equiplist2. Создаем каталог с именем equipment в каталоге ski.plases. Перемещаем файлы equiplist и equiplist2 в каталог equipment. Создаем и перемещаем каталог newdir в каталог ski.plases и называем его plans.

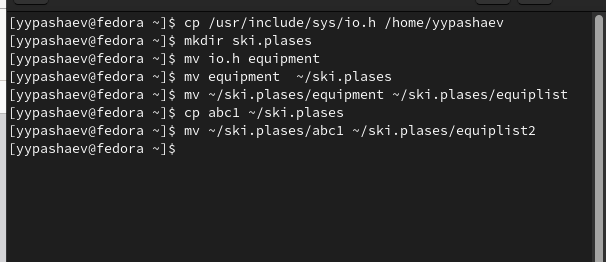


Figure 5: Работа с каталогами

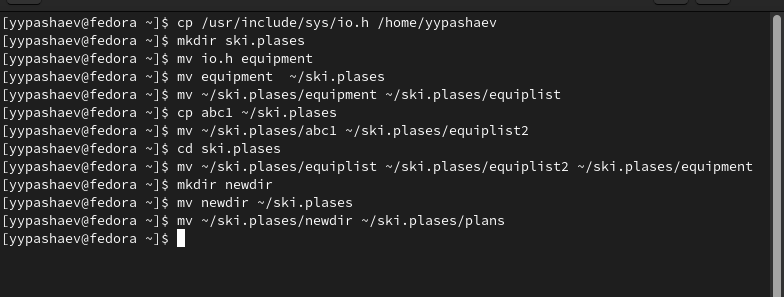


Figure 6: Работа с каталогами

3.Определю опции команды chmod, необходимые для того, чтобы присвоить перечис- ленным ниже файлам выделенные права доступа

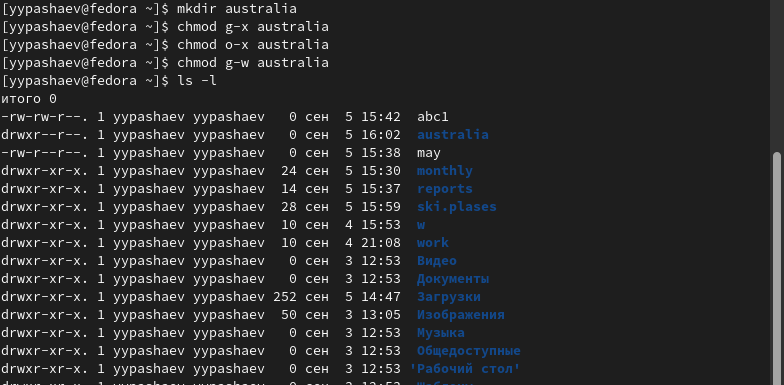


Figure 7: Настройка прав доступа

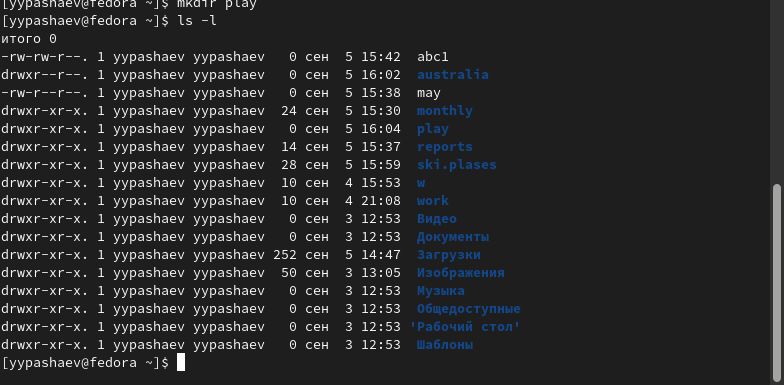


Figure 8: Настройка прав доступа

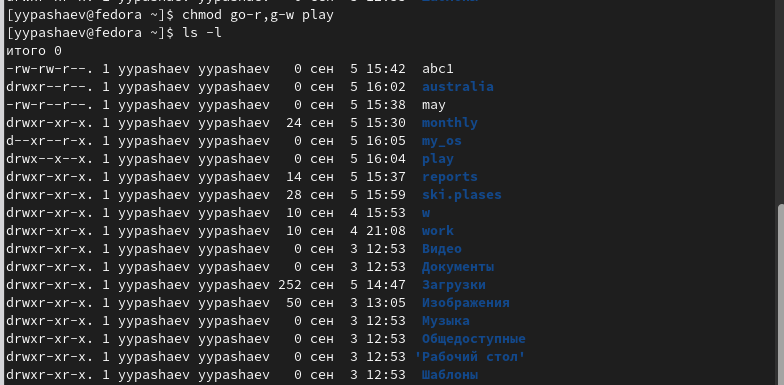


Figure 9: Настройка прав доступа

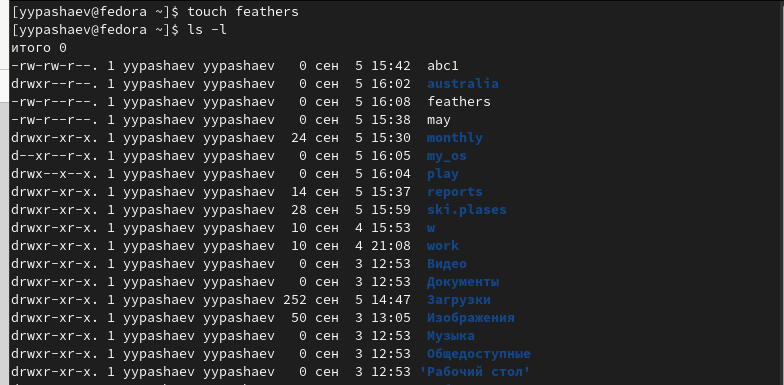


Figure 10: Настройка прав доступа

1. 4.2 - 4.12. Выполним все указанные действия по перемещению файлов и ката- логов (рис. [11](#fig:011))

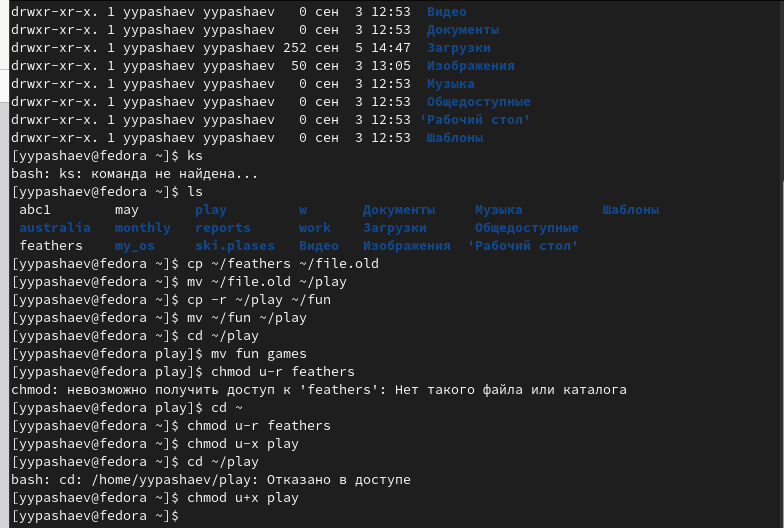


Figure 11: Работа с файоами и правами доступа

1. Прочитаем man по командам mount, fsck, mkfs, kil (рис. [12](#fig:012))

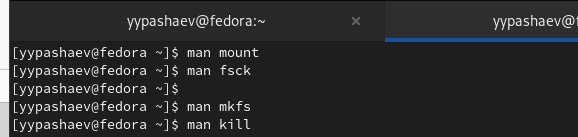


Figure 12: прочитаем man

# 3 Выводы

В ходе данной работы мы ознакомились с файловой системой Linux, её структу- рой, именами и содержанием каталогов. Научились совершать базовые операции с файлами, управлять правами их доступа для пользователя и групп. Ознако- мились с Анализом файловой системы. А также получили базовые навыки по проверке использования диска и обслуживанию файловой системы.

# 4 Контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёст- ком 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 убра- ны 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 фай- ловой 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 про- верить 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 15 бинарных деревьев (binary-tree), позаимствованная из программного обеспече- ния баз данных. JFS (журналируемая файловая система). JFS была разработана и использовалась IBM. Вначале JFS была закрытой системой, но недавно IBM решила открыть до- ступ для движения свободного программного обеспечения. Внутренняя структура JFS близка к ReiserFS. Средняя стабильность, нет комплексных инструменталь- ных 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, свобод- ные индексные дескрипторы и т. д.). Суперблок всегда находится в оперативной памяти. Различные версии ОСUnix способны поддержи- вать 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 распола- гается 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 записыва- ется 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 соответ- ствовать несколько файлов. Вinode хранится вся информация о файле, кроме его имени. Область индексных дескрипторов имеет фиксиро- ванный формат и располагается непосредственно за суперблоком. Об- щее число описателей и, следовательно, максимальное число файлов задается в момент создания файловой системы. Описатели нумеру- ются натуральными числами. Первый описатель используется ОС для описания специального файла (файла «Плохих блоков»). То есть по- врежденные 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 ката- логов (в том числе корневой каталог). Специальные файлы представле- ны 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 фай- ловой 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 объеди- нять 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 храня- щихся в буферах оперативной памяти (в этих буферах оседают данные из пользовательских файлов и служебная информация файловой си- стемы, такая как каталоги, индексные дескрипторы, списки свободных, занятых и поврежденных блоков и т. п.)
5. Как создаётся файловая система? Ответ: Общее дерево файлов и катало- гов 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, жест- кий 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, созда- ется 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 дискете, то команда примет вид: [root]# mkfs -t ext2 /dev/fd0 После выполнения команды mkfs в указанном разделе будет создана фай- ловая 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 монтирова- ния”. Точкой монтирования называется тот каталог в уже существующем и известном системе дереве каталогов, который будет теперь служить корне- вым 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 мон- тирования лучше использовать пустые каталоги (заранее заготовленные).
6. Дайте характеристику командам, которые позволяют просмотреть тексто- вые 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 управ- ляющие 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 умол- чанию 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 пу- ти 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 переименова- ния файлов и каталогов. Ответ: Команды mv и mvdir 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 фай- ла 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 ката- лог monthly.00 в monthly.01 mv monthly.00 monthly.01 • Перемещение каталога в другой каталог. Переместить каталог monthly.01в 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 досту- па — совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носите- лям, 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 администрато- ра. 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 # Список литературы{.unnumbered}