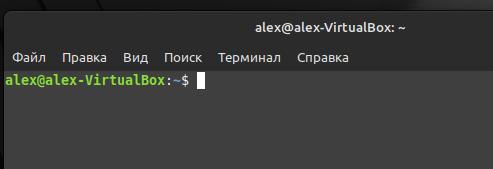
Лабораторная 8

**Основное задание**

1. Войти в систему с собственной учетной записью

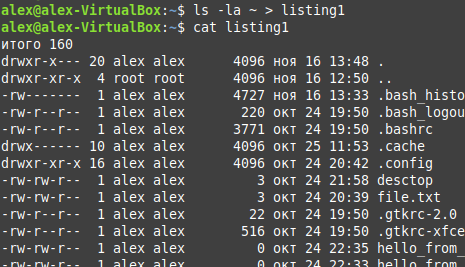


1. Создать в домашнем каталоге 2-3 файла произвольного содержания (имена файлов - u1, u2, u3).





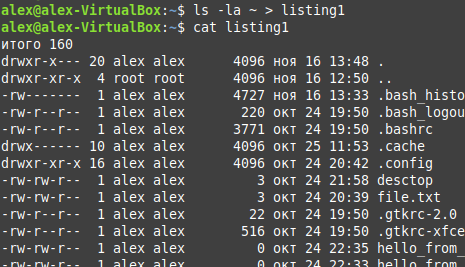
1. Получить развернутый список файлов домашнего каталога и сохранить его в файле listing1



Опция -l расшифровывается как "long format" и предоставляет более подробную информацию о файлах и каталогах. Она выводит каждый файл в формате колонок и включает дополнительные данные, такие как права доступа, владелец, группа, размер файла, дата последней модификации и имя файла.

Опция -a указывает ls выводить все файлы, включая скрытые. Файлы, начинающиеся с точки (например, .bashrc), считаются скрытыми в UNIX-подобных системах, и без этой опции они не отображаются при использовании команды ls.

1. Просмотреть файл listing1, обратив внимание на поля прав доступа, владельца и группы

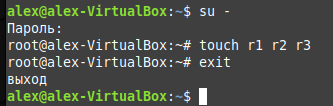


- rw-r--r-- - поля прав доступа, где rw- означает права доступа для владельца файла (read-write), а r-- и r-- означают права доступа для группы и других пользователей соответственно (read-only).

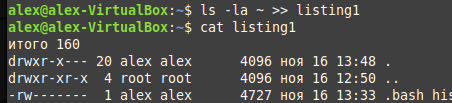
- alex - владелец файла.

- alex - группа файла.

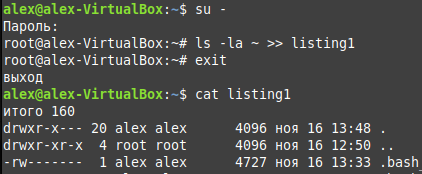
1. Повторить п. 2 от имени пользователя root в новом сеансе или по команде su (имена файлов - r1, r2, r3). Завершить сеанс root

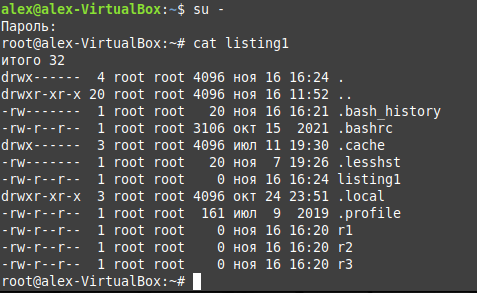


1. Повторить п.3, результат дописать в файл listing1



1. Открыть файл listing1 и сравнить права доступа для файлов, созданных от вашего имени и от имени суперпользователя

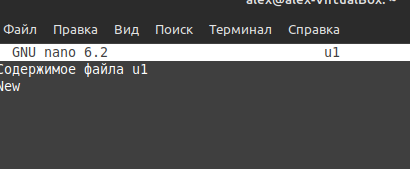




1. Изменить содержимое файлов, созданных вами и суперпользователем. Сохранить изменения

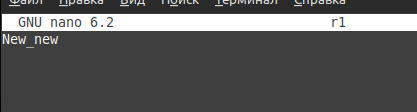
вами





суперпользователем





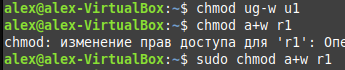
1. Открыть сеанс root



1. Перейти в каталог /home/ваша\_учетная\_запись



1. Изменить права доступа к файлам u1 и r1 следующим образом:
   * u1: запретить запись для владельца и группы
   * r1: разрешить запись для всех



ug-w: Это комбинация параметров, которая удаляет права на запись у владельца и у группы для указанного файла

u - означает владельца -g - означает группу

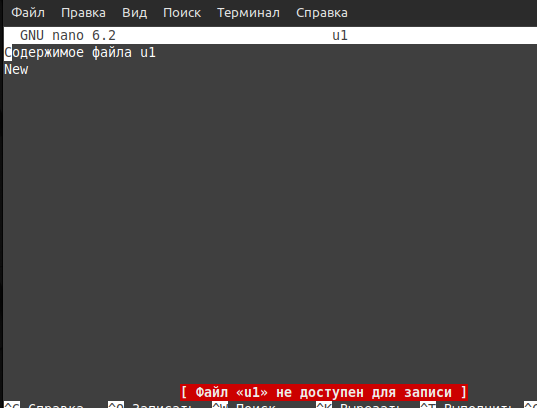
w - убирает (отменяет) права на запись. Знак - означает удаление права.

a+w: Это комбинация параметров, которая добавляет права на запись для всех пользователей для указанного файла или каталога.

a - означает все (all). В данном случае, это означает добавление прав для всех пользователей.

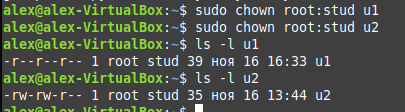
+w - добавляет права на запись. Знак + означает добавление права.

1. Переключиться в сеанс пользователя и изменить содержимое файлов u1 и r1. Сохранить изменения

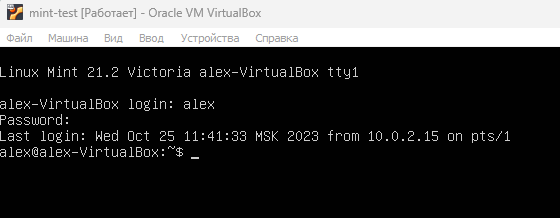
 u1 теперь недоступен для записи

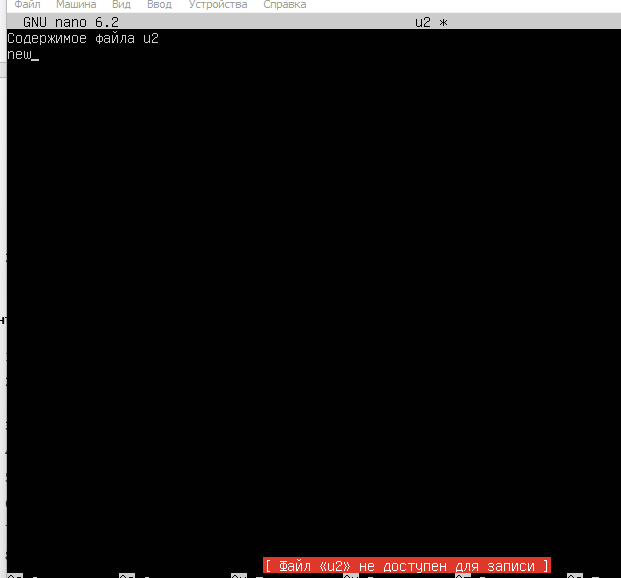
 r1 теперь доступен для записи

1. Перейти в сеанс root и изменить владельца файлов u1 и u2 на root, а группу - на stud

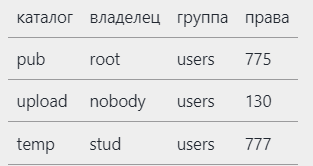


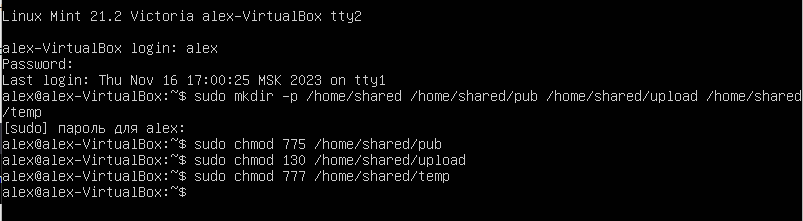
1. Из tty1 попробовать изменить файл u2





1. Из tty2 создать каталоги /home/shared, home/shared/pub, /home/shared/upload, /home/shared/temp. Установить на них следующие права:





Команда chmod используется для изменения прав доступа к файлам или каталогам в Linux. Права доступа состоят из трех групп: владелец файла , группа и остальные пользователи

Числовые значения для установки прав доступа определяются как комбинация следующих прав:

- 4: чтение (read)

- 2: запись (write)

- 1: выполнение (execute)

В команде chmod, когда мы устанавливаем права доступа, используются суммы этих числовых значений для каждой из групп прав (user, group, others).

Таким образом

775 для /home/shared/pub:

- 7 для владельца (user) - чтение, запись и выполнение (4+2+1 = 7)

- 7 для группы (group) - чтение, запись и выполнение (4+2+1 = 7)

- 5 для остальных (others) - только чтение и выполнение (4+1 = 5)

130 для /home/shared/upload:

- 1 для владельца (user) - только выполнение (1)

- 3 для группы (group) - чтение и выполнение (2+1 = 3)

- 0 для остальных (others) - без прав доступа (0)

777 для `/home/shared/temp`:

- 7 для владельца (user) - чтение, запись и выполнение (4+2+1 = 7)

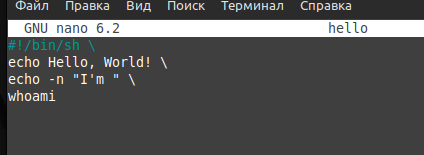
- 7 для группы (group) - чтение, запись и выполнение (4+2+1 = 7)

- 7 для остальных (others) - чтение, запись и выполнение (4+2+1 = 7)

**Задания для самостоятельного выполнения**

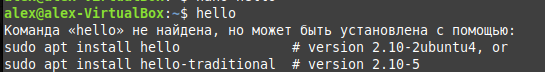
1. В текущей папке создать файл hello следующего содержания

|  |  |
| --- | --- |
| 1  2  3  4 | #!/bin/sh \  echo Hello, World! \  echo -n "I'm " \  whoami |





1. Выполнить следующие действия и проанализировать результаты:
   * набрать в командной строке имя файла hello и нажать Enter



* + набрать в командной строке sh hello и нажать Enter



1. Разберитесь, как в Linux можно организовать специальные каталоги-файлообменники - такие, что создавать в них файлы могут любые пользователи, а удалять - только владельцы файлов (тот, кто их создал).

В Linux для этой задачи можно использовать права доступа

Создайте общий каталог: sudo mkdir /shared\_directory

- Установите права доступа, чтобы пользователи могли создавать файлы: sudo chmod 1777 /shared\_directory

- 1 - права на выполнение для всех (чтобы можно было зайти в директорию)

- 777 - чтение, запись и выполнение для всех (это позволит всем пользователям создавать файлы)

- s - установка флага "sticky bit", который делает каталог файлообменником

- Пользователи могут создавать файлы в этой директории, но они будут владельцами этих файлов.

**Контрольные вопросы**

1. Зачем у файла нужны атрибуты доступа?

Атрибуты доступа определяют, как пользователи и программы могут взаимодействовать с файлами. Они контролируют, кто может читать, записывать и исполнять файлы.

1. Для каких трех категории пользователей задаются права доступа каждого конкретного файла?

Права доступа к файлам в UNIX-подобных системах определяются для трех категорий пользователей:

- Владелец файла (user)

- Группа пользователей, к которой принадлежит файл (group)

- Остальные пользователи, не владеющие файлом (others)

1. Для чего в UNIX-подобных системах используются пользовательские группы?

Пользовательские группы позволяют организовать пользователей в группы для более удобного управления доступом к файлам и ресурсам системы. Группа позволяет давать общие права доступа к файлам для нескольких пользователей.

1. Какие действия с файлами регламентируются правами доступа?

Права доступа определяют, можно ли читать, записывать или выполнять файлы. Это ограничивает действия, такие как чтение содержимого, запись в файл или его исполнение.

1. На какой системе счисления основывается числовое представление прав доступа?

Числовое представление прав доступа основано на восьмеричной системе счисления. В каждой из трех категорий (владелец, группа, остальные) используются три бита для представления различных прав доступа

1. Чем отличается назначение прав доступа к простому файлу и к директории?

Права доступа к файлу определяют, что можно делать с самим файлом (читать, записывать, исполнять). Права доступа к директории управляют доступом к файлам внутри нее: создание, удаление и изменение файлов.

1. Кто может менять права доступа определенного файла?

Владелец файла и администратор системы (обычно пользователь root) могут менять права доступа определенного файла.

1. Каково назначение специальных битов прав доступа?

Специальные биты прав доступа, такие как sticky bit и setgid/setuid, предоставляют дополнительные возможности. Например, sticky bit применяется к директориям для управления удалением файлов, созданных другими пользователями, а setuid/setgid устанавливают права на исполнение файлов с правами владельца/группы.