МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

## Факультет информационных технологий и робототехники

Кафедра программного обеспечения информационных систем и технологий

Отчет по лабораторной работе № 5

по дисциплине:” Системное программирование”

на тему: ”Администрирование системы Linux ”

Вариант 6

Выполнили**:**  студент группы 10701222 Дяденькина Т.М.

Приняла**:** пр. Давыденко Н.В.

Минск 2024

**Цель работы**: Закрепить на практике основы администрирования системы Linux, изучить атрибуты файлов и права доступа к ним, освоить работу с файлами и каталогами.

**Задание**

Задание 1 – Создание пользователей

Создайте стандартного пользователя с помощью GUI. Пользователю присвойте имя члена вашей команды, но в имени должен присутствовать суфффикс \*GUI. Это нужно для проведения экспериментов. При присвоении имен действуйте по принципу: «Относитесь к именам переменных, как к именам детей своих». Пароль пользователя должен быть простым, например «123».

Просмотрите содержимое файла /etc/passwd, сравните атрибуты реальных пользователей и пользователя root. Результат подтвердите скриншотом. Дайте пояснения.

Создайте несколько стандартных пользователей (аккаунтов) посредсвам командной строки. При этом пользователям присвойте имена членов вашей команды, но в имени должен присутствовать суфффикс \*CL или \*CLI. Это нужно для того, чтобы отличить пользователей созданных при помощи графического и консольного интерфейсов. Создайте двух пользователей с одинаковыми простыми паролями. Создайте скелет и пользователя с шаблоном скилета в директории /home. Также попробуйте создать аккаунт реального пользователя без пороля.

При создании пользователей, которые нужны для экспериментов, назначайте им одинаковые пароли (это тоже эксперимент), такие чтобы не забыть. Например, «123». Небольшой совет. Имена пользователей пишите с маленьких букв, это поможет ускорить вашу дальнейшую работу. Разделителем в имени используйте нижнее подчеркивание «\_», тире «-» или заглавную букву.

Например, serhey\_stankevich, sergei-stankewich, sergStankevich.

Задание 2 – Изменение параметров паролей пользователей

Пароли должны быть простые и одинаковые. Это нужно для проведения экспериментов. Просмотрите содержимое файла /etc/passwd, сравните атрибуты реальных пользователей. Просмотрите содержимое файла /etc/shadow, сравните атрибуты паролей пользователей, особое внимание обратите на шифр пароля у пользователей с однаковым паролем. Измените периоды изменения паролей для пользователей. Внесите ограничения в следующие атрибуты: минимальный возраст пароля (сутки); максимальный возраст пароля (сутки); период предупреждения пароля; период бездействия пароля; дата истечения срока действия аккаунта.

Результат поддтвердите скриншотом. Дайте пояснения.

Задание 3 – Создание групп и работа с правами доступа к файлам

Создайте пользователя (аккаунт) с правами администратора. В каталогах /home/ администраторов и других пользователей создайте по несколько файлов с разными правами доступа. Создайте общую группу пользователей.

Выполните следующие требования:

1. Одного из пользователей перевести в группу shadow

2. Создать у каждого пользователей директорию с 2 файлами

3. Просмотреть текущие права доступа к файлам для всех пользователей

4. Каталог пользователя в группе shadow сделать доступным только в своей группе

5. Файлы второго пользователя сделать доступными только владельцам

6. Под админом назначить всем созданным файлам права только для чтения для всех пользователей

7. Пользователем в группе shadow лишить всех остальных пользователей права исполнять его файлы

8. Под админом назначить всем пользователям все права

9. Удалить пользователя, находящегося в группе shadow

Укажите является ли группа системной или создана пользвателем.

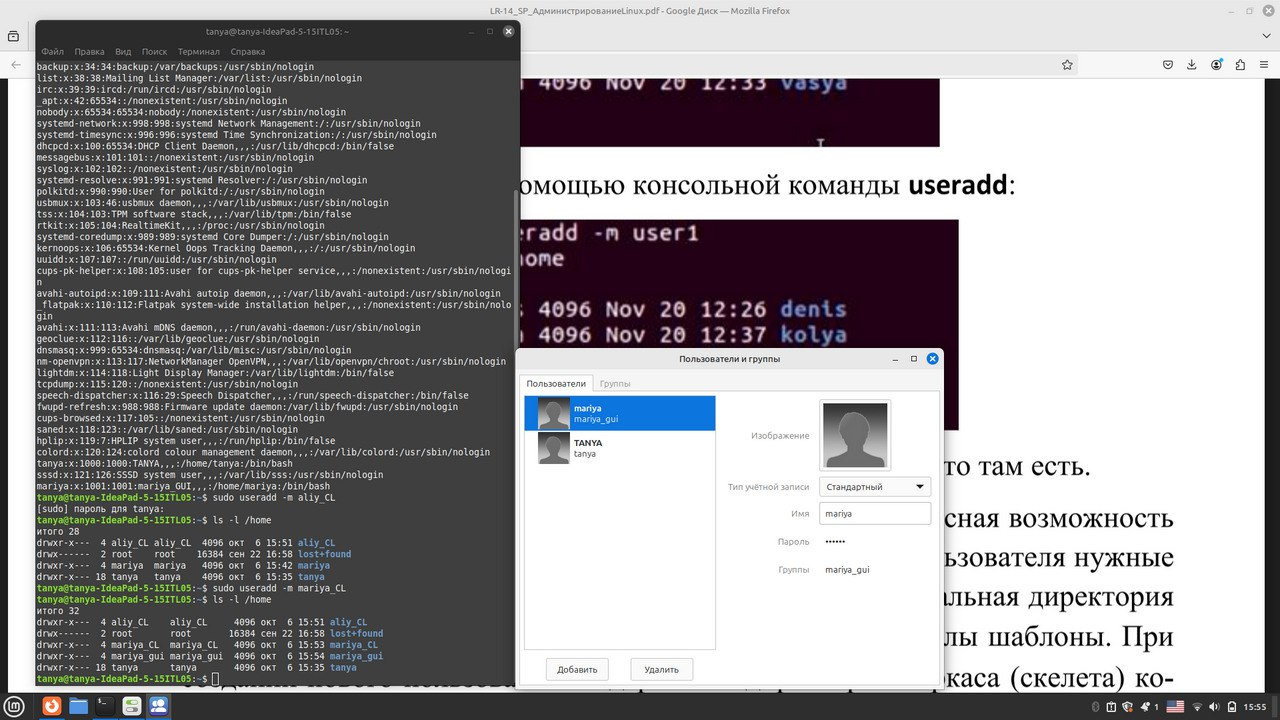
Получите индентификаторы пользователей и состояние активных пользователей системы. Проверьте содержимое файлов /etc/shadow, /etc/passwd,/etc/group.

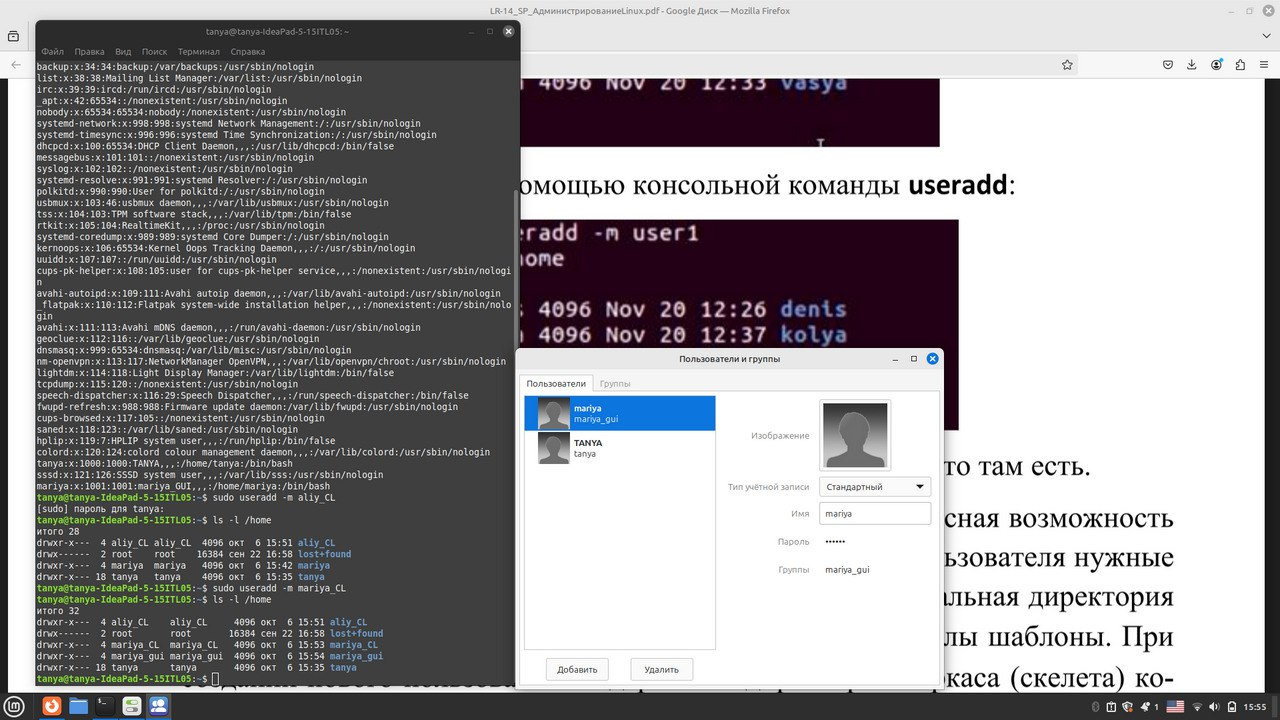
Некоторые файлы защитите липким битом. Попробуйте удалить созданные файлы из других аккаунтов администраторов и простых пользователей. Опишите какой получили результат.

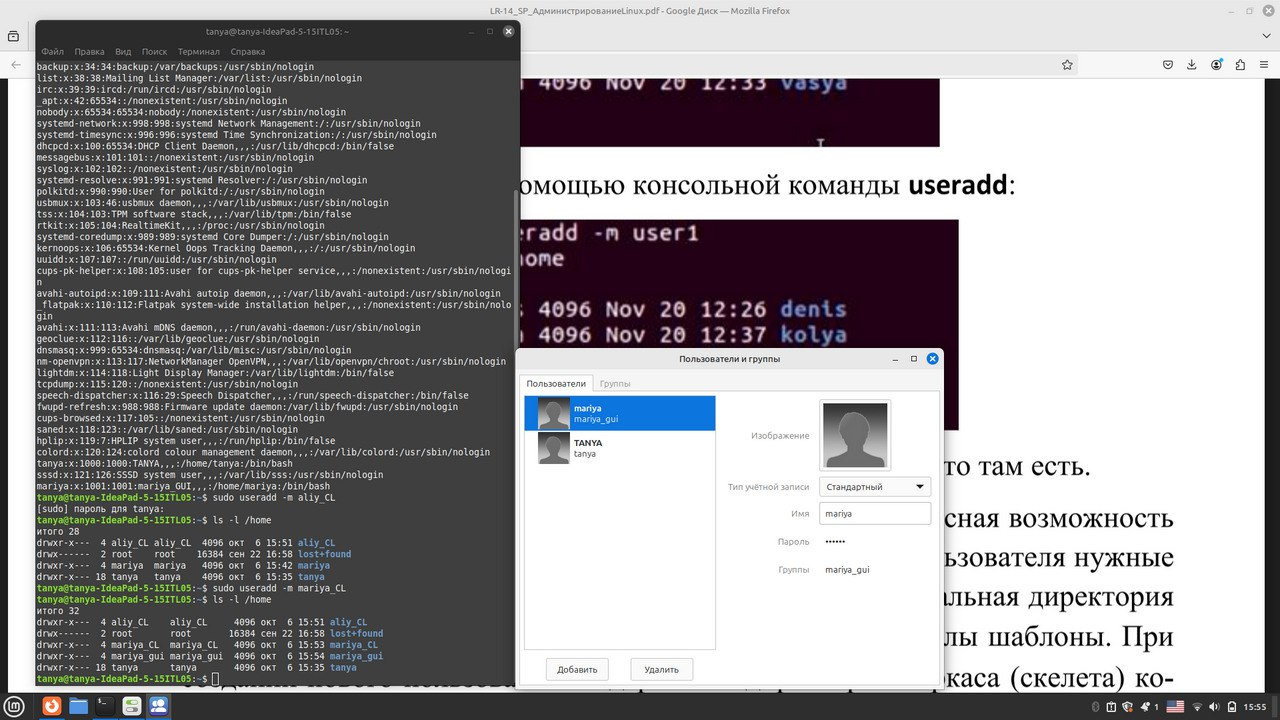
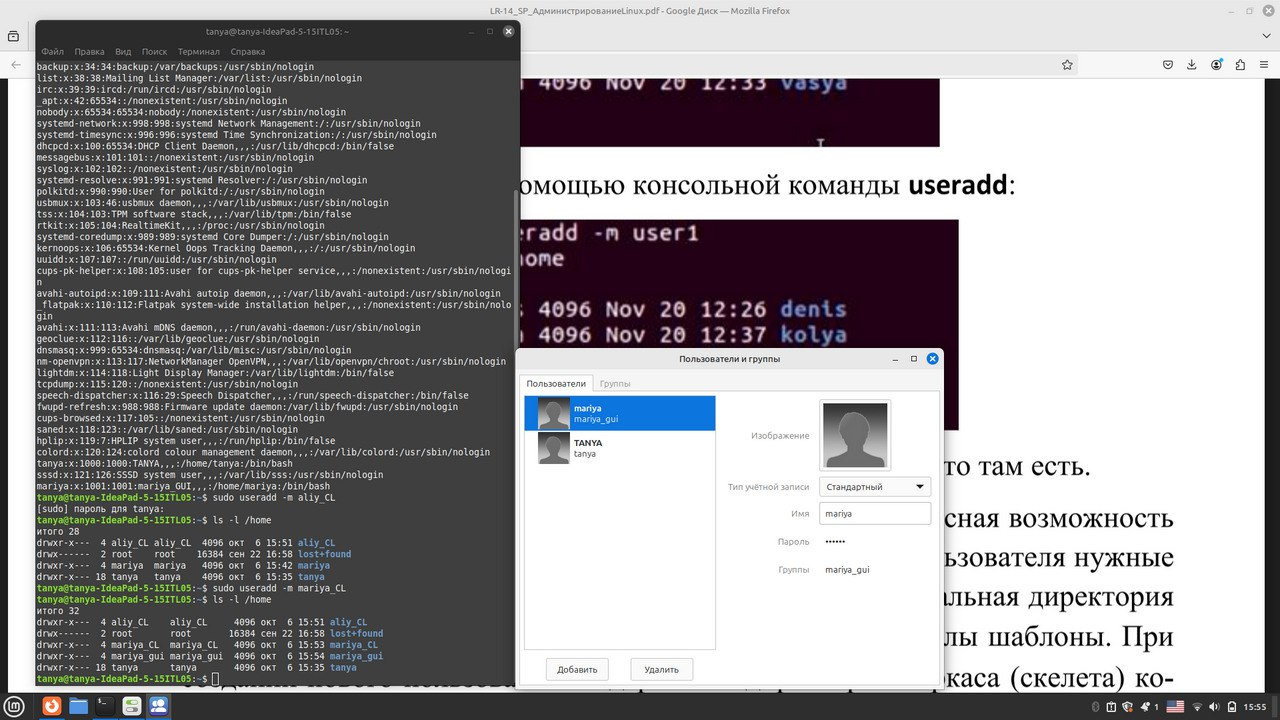
Создайте жесткую ссылку, перенесите эту ссылку в пространство другого пользователя и откройте ее в сеансе этого пользователя, затем присвойте жесткой ссылке одну группу и откройте ссылку из пользователя этой группы. Измените владельца и группу жесткой ссылки и посмотрите, как изменились атрибуты основного файла.

Примените разные варианты изменения атрибутов доступа файла и каталогов. Попробуйте совершить разные операции с этими файлами от имени других пользователей. Активно используйте команду su.

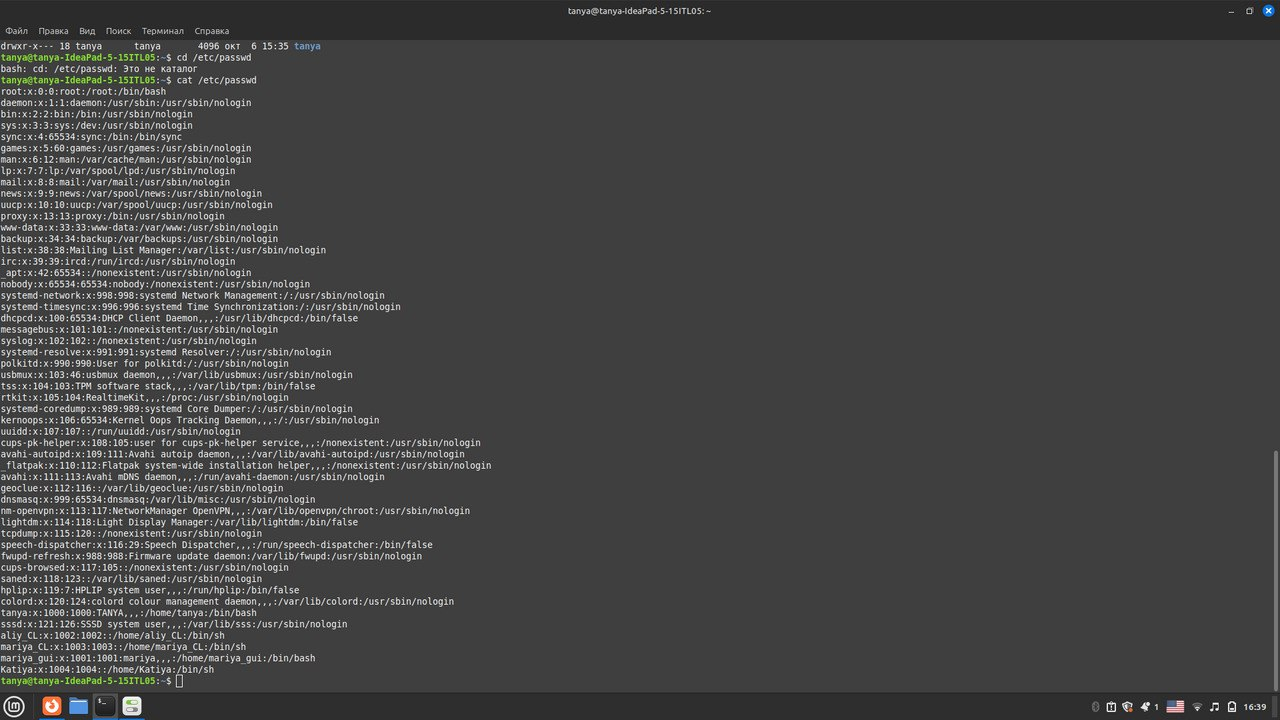
**Выполнение**

Cl and GUI user

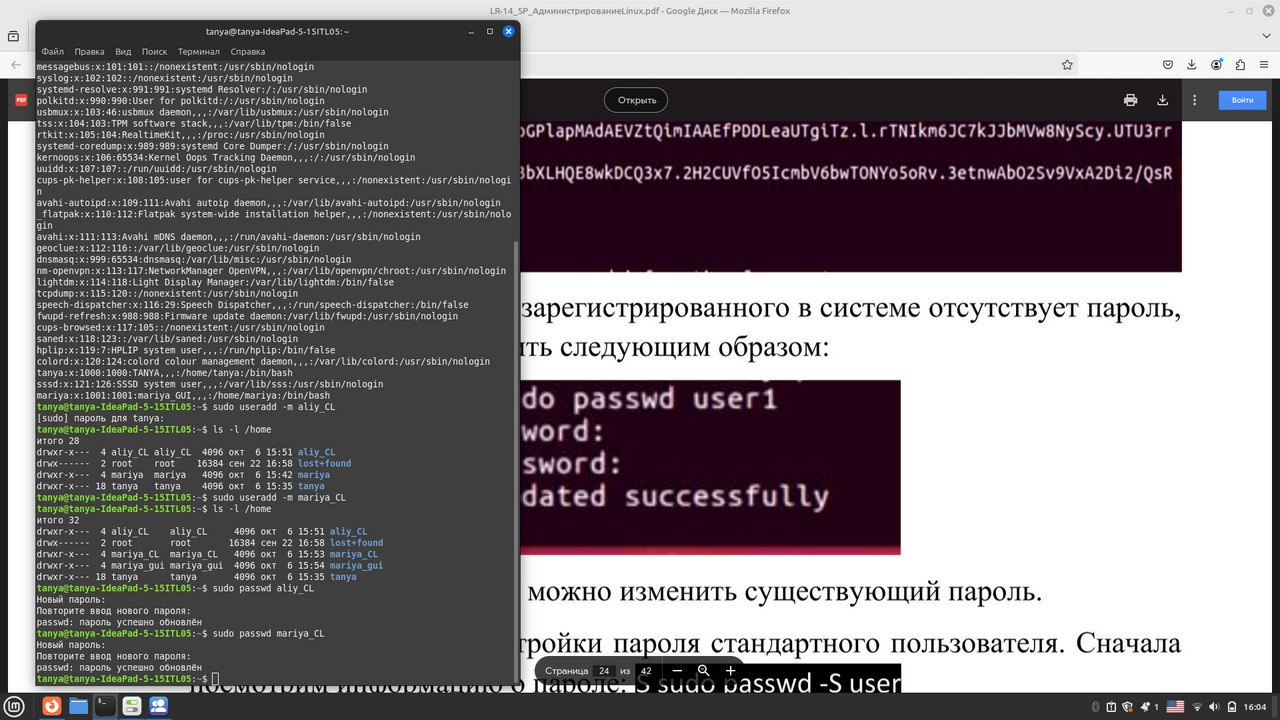




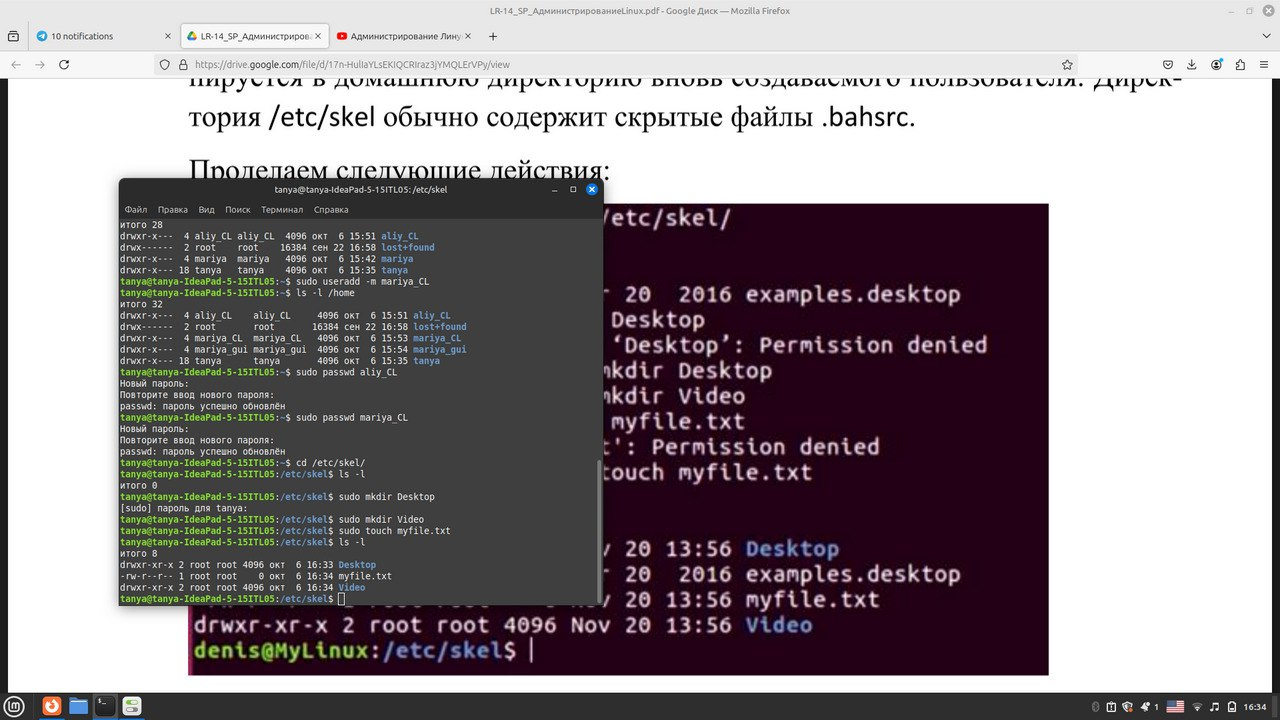
Просмотрите содержимое файла /etc/passwd



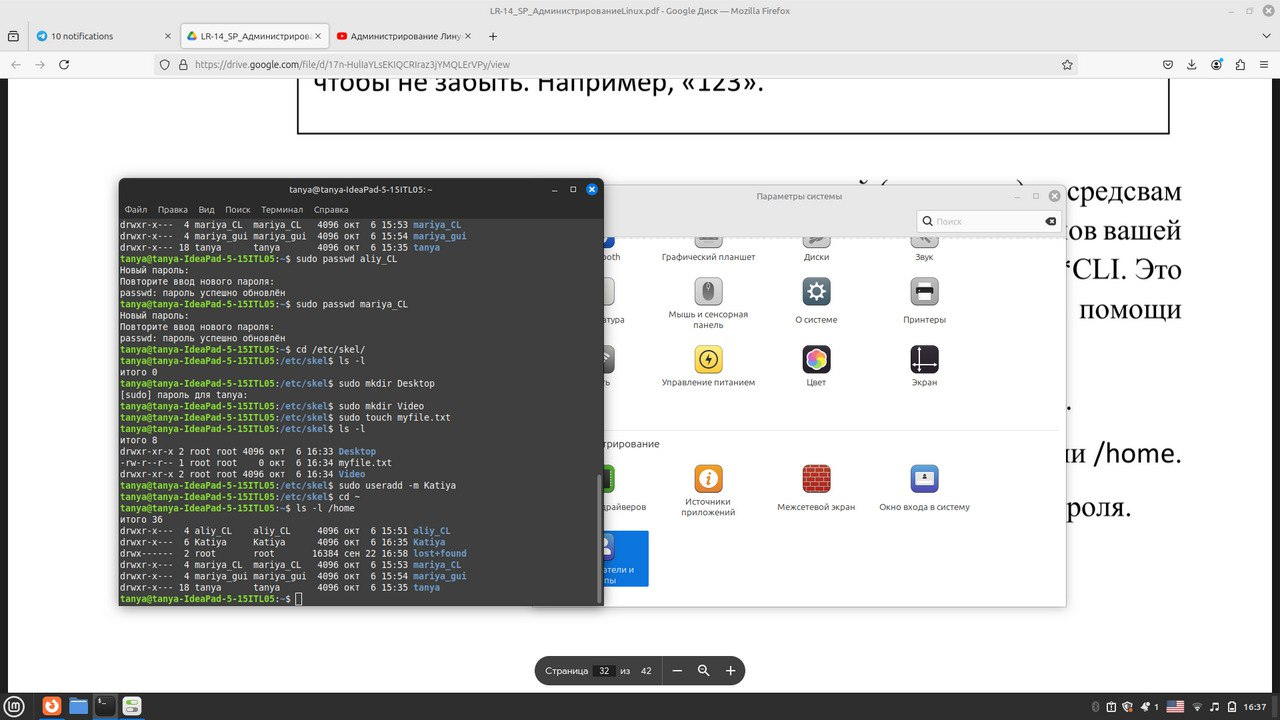
Password



Skel



New user



**Контрольные вопросы**

1. Опишите отличия команд su и sudo.

Ответ:

Команды `su` и `sudo` в операционных системах на базе Unix/Linux используются для выполнения команд с повышенными привилегиями, но они имеют разные механизмы и область применения. Вот основные отличия между ними:

1. Основное назначение:

- `su` (Substitute User): Позволяет переключаться на другого пользователя в системе, чаще всего на пользователя с правами суперпользователя (root). Когда вы выполняете `su`, вы обычно будете запрашивать пароль для целевого пользователя (по умолчанию root). Если вы хотите переключиться на root, вам нужно будет ввести пароль root.

- `sudo` (Superuser Do): Позволяет пользователям выполнять команды от имени суперпользователя или другого пользователя, без необходимости переключаться на него. Для выполнения команды с помощью `sudo` достаточно ввести свой собственный пароль, если ваша учетная запись имеет соответствующие права в файле конфигурации `/etc/sudoers`.

2. Настройки доступа:

- `su`: Доступ предоставляется пользователям, которые знают пароль целевого пользователя. Обычно это означает что нужно знать пароль root для выполнения команд от его имени.

- `sudo`: Доступ контролируется через файл `/etc/sudoers`, где администраторы могут настраивать, какие пользователи или группы могут выполнять какие команды. Таким образом, `sudo` позволяет более детализированный контроль над привилегиями. Например, один пользователь может иметь право на выполнение только определённых команд с привилегиями root, в то время как другой может иметь доступ ко всем командам.

3. Уровень безопасности:

- `su`: По умолчанию, требует ввода пароля целевого пользователя, что может представлять риск, если пользователи обладают знаниями о пароле.

- `sudo`: Работает на основе принципа "наименьших привилегий". Пользователи могут выполнять только те команды, на которые у них есть разрешение, и не могут входить в систему под учетной записью root (в большинстве случаев). Это улучшает безопасность и отслеживание действий пользователей.

4. Временные метки и учет действий:

- `su`: Обычно не ведёт учёт истории команд, выполненных при использовании.

- `sudo`: Записывает все команды, выполненные через него, в системный лог (обычно `/var/log/auth.log` или аналогичный), что позволяет отслеживать и анализировать действия пользователя.

5. Интерфейс:

- `su`: После успешного входа в систему командная оболочка переключается на оболочку целевого пользователя, и вы работаете от его имени, пока не выйдете из него.

- `sudo`: Позволяет выполнять одну команду с повышенными привилегиями, после чего ваша сессия возвращается к обычному пользователю.

Заключение:

В общем, хотя обе команды позволяют выполнять действия с повышенными правами, `sudo` предоставляет более безопасный и контролируемый способ управления доступом, подходит для сценариев, где требуется ограниченный доступ к определённым командам, и обеспечивает возможность аудита действий пользователя. `su` же используется для полноценного переключения на другого пользователя и подразумевает большее доверие к знанию паролей.