



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА

Институт перспективных технологий и индустриального программирования (ИПТИП)
Кафедра индустриального программирования

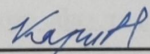
ОТЧЁТ ПО УЧЕБНОЙ ПРАКТИКЕ
Ознакомительная практика

приказ Университета о направлении на практику от «22» января 2024 г.
№ 385-С

Отчет представлен к
рассмотрению:

Студент группы
ЭФБО-09-23

«31» _мая_ 2024 г.

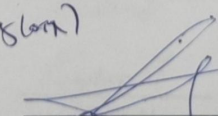

(подпись)

Калинин Н.В.

Отчет утвержден.
Допущен к защите:

Руководитель
практики от кафедры

«31» _мая_ 2024 г.

5 (6007)

(подпись)

Соловьев А.М.

Москва 2024



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА

Институт перспективных технологий и индустриального программирования (ИПТИП)
Кафедра индустриального программирования

ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ НА УЧЕБНУЮ ПРАКТИКУ
Ознакомительная практика

Студенту 1 курса учебной группы ЭФБО-09-23

Студент Калинин Никита Викторович

Место и время практики: РТУ МИРЭА, ИП, с 09 февраля 2024 г. по 31 мая 2024 г.

Должность на практике: студент

1. СОДЕРЖАНИЕ ПРАКТИКИ:

- 1.1. Изучить: использование и администрирование операционной системы Astra Linux.
- 1.2. Практически выполнить: задания на установку и использование операционной системы.
- 1.3. Ознакомиться: основные инструменты и технологии по использованию и администрированию операционной системы Astra Linux.

2. ДОПОЛНИТЕЛЬНОЕ ЗАДАНИЕ: в процессе прохождения практики следует выполнять задания в отведенные на них сроки.

3. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ: в процессе прохождения практики следует загружать результаты выполнения заданий в рабочую область в системе дистанционного обучения. В итоговый отчет следует поместить сборную форму каждого из заданий.

Руководитель практики от кафедры

«09» __ февраля ____ 2024 г.

(подпись)

(Соловьев А.М.)

Задание получил

«09» __ февраля ____ 2024 г.

(подпись)

(Калинин Н.В.)

СОГЛАСОВАНО:

Заведующий кафедрой:

«09» __ февраля ____ 2024 г.

(подпись)

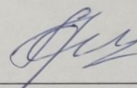
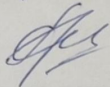
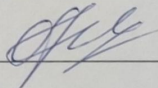
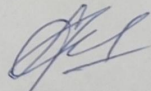
(Юдин А.В.)



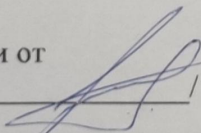
МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА

**РАБОЧИЙ ГРАФИК ПРОВЕДЕНИЯ
ОЗНАКОМИТЕЛЬНОЙ ПРАКТИКИ**

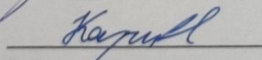
студент Калинин Н.В. 1 курса группы ЭФБО-09-23 очной формы обучения, обучающегося по направлению подготовки 09.03.02 «Информационные системы и технологии».

Неделя	Сроки выполнения	Этап	Отметка о выполнении
1-4	с 09.02.2024 по 09.03.2024	Подготовительный этап, включающий в себя организационное собрание	
5-8	с 09.02.2024 по 02.04.2024	Выполнение задания по практике в соответствии с выданным заданием	
9-12	с 02.04.2024 по 30.04.2024	Подготовка отчета по практике	
13-16	с 01.05.2024 по 31.05.2024	Защита отчета по практике у руководителя практики (представление отчета по практике к защите)	

Руководитель практики от
кафедры

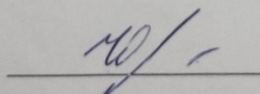
 / Соловьев А.М., доцент, к.т.н./

Обучающийся

 / Калинин Н.В./

Согласовано:

Заведующий кафедрой

 /Юдин А.В., д.э.н., к.ф-м.н./

Задание №1

Введение

Это и последующее задание будет выполняться в операционной системе Astra Linux. Если пункт выделен желтым, то это значит, что пункт необходимо зафиксировать скриншотом, либо записать необходимую информацию в отчет.

Для работы над заданием, если на вашем устройстве уже есть данная операционная система, можно пропустить все пункты, связанные с первичной установкой системы и прикрепить в оба раздела только скриншот из раздела «Информация о системе» (Пуск-Системные-Информация о системе, либо через поиск). В случае, если Вы планируете установить себе ОС напрямую, без использования виртуальной машины, Вам необходимо скачать образ на установочную флешку и продолжить работу с пункта «2. Загрузка образа и установка операционной системы».

1. Настройка виртуальной машины для будущей операционной системы

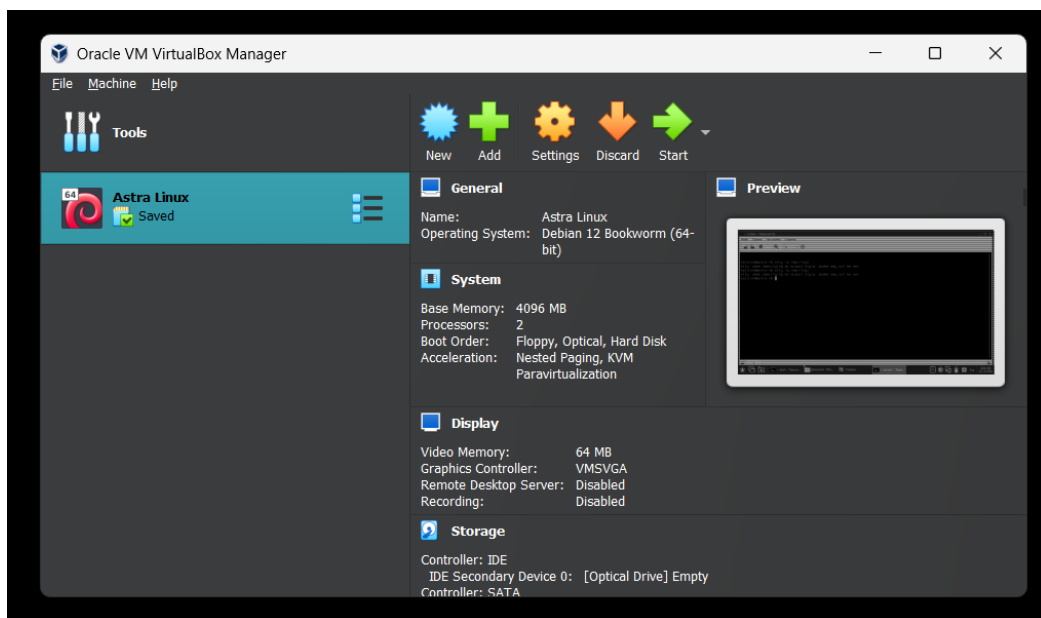
1. Скачайте и установите Virtual box.
2. Откройте VirtualBox.
3. Кликните на кнопку "Создать", расположенную в меню "Файл" в главном окне программы.
4. Задайте имя для вашей виртуальной машины, используя вашу фамилию на английском языке, и оставьте настройки для папки машины по умолчанию.

5. Выберите тип и версию операционной системы из доступных списков. Тип должен быть "Linux", а версия - "Debian 64-bit". Нажмите «Далее» для перехода к следующим настройкам.
6. Назначьте вашей виртуальной машине 4096 МБ оперативной памяти и нажмите «Далее».
7. Выберите пункт «Создать новый виртуальный жесткий диск», рекомендуется выбрать не менее 16ГБ. Нажмите «Далее».
8. Сделайте скриншот с описанием текущей виртуальной машиной, как в примере и прикрепите в область.
9. Нажмите на кнопку «Настроить».
10. Перейдите в раздел настроек "Дисплей". На вкладке "Экран" установите объем выделенной видеопамяти на уровне 64 МБ и более.
11. Перейдите в раздел «Общие» на вкладку "Дополнительно". Из выпадающего списка выберите "Двунаправленный" режим для следующих параметров:
 - Общий буфер обмена;
 - Функция Drag'n'Drop.

student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов) .

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git)

Скриншот шага №8



Отчет:

1. Программное обеспечение VirtualBox было успешно скачано и установлено на ваш компьютер.
2. После запуска VirtualBox, была инициирована создание новой виртуальной машины.
3. В качестве имени для виртуальной машины было использовано название операционной системы. Папка для хранения файлов машины оставлена по умолчанию.
4. Для созданной машины был выбран тип операционной системы Linux и версия Debian 64-bit.
5. Виртуальной машине было назначено 4096 МБ оперативной памяти для ее оптимальной работы.
6. Также было произведено создание нового виртуального жесткого диска с рекомендуемым минимальным размером в 16 ГБ.
7. Для подтверждения текущих настроек, был подготовлен и прикреплен скриншот интерфейса виртуальной машины.

8. Перейдя в раздел настройки "Дисплей", было произведено установление объема выделенной видеопамяти на уровне 64 МБ.
9. В заключительной части настроек, в разделе «Общие» на вкладке "Дополнительно" был настроен "Двунаправленный" режим для указанных параметров.

2. Загрузка образа и установка операционной системы

1. Скачайте ISO-образ операционной системы из СДО.
2. Нажмите «Запустить» виртуальную машину, которую Вы настроили и установили.
3. В всплывшем окне в поле «DVD» выберите образ, который Вы скачали (через проводник, в папке, куда скачали образ).
4. Выберите пункт «Графическая установка» (рекомендуется).
5. Примите условия пользовательского соглашения.
6. Нажимайте кнопку «Продолжить» в окне «Настройка клавиатуры» (только если не планируете их изменять).
7. В разделе «Настройка сети» оставьте имя компьютера «astra» и нажмите «Продолжить».
8. В окне «Настройка учетных записей пользователей и паролей» в качестве имени пользователя введите вашу фамилию на английском языке. Сделайте скриншот и разместите в рабочую область отчета. Нажмите «Продолжить».
9. В следующем окне задайте пароль 12345678 в обоих полях и нажмите «Продолжить».

10. В окне «Настройка времени» выберите Москва +00 - Москва и нажмите «Продолжить».
11. В окне «Разметка дисков» оставляем пункт «Авто - использовать весь диск» и нажимаем «Продолжить».
12. Установщик выберет диск, оставляем его по умолчанию и нажимаем «Продолжить».
13. В следующем окне оставляем пункт «Все файлы в одном разделе» и нажимаем «Продолжить».
14. В следующем окне оставляем «Закончить разметку и записать изменения на диск и нажимаем «Продолжить».
15. В следующем окне выбираем «Да» в выборе «Записать изменения на диск?» и нажимаем «Продолжить».
16. В окне «Установка базовой системы» в версии ядра выбираем то, которое установщик предложил по умолчанию и нажимаем «Продолжить».
17. В окне «Выбор программного обеспечения» выбираем все представленные пункт (некоторые из них не выбраны по умолчанию) и нажимаем «Далее».
18. После настройки в окне «Дополнительные настройки ОС» выбираем максимальный уровень защищенности «Смоленск» и нажимаем «Продолжить».
19. В следующем окне оставляем все настройки по умолчанию и нажимаем «Продолжить».
20. В следующем окне оставляем «Да» для системного загрузчика GRUB и нажимаем «Продолжить».
21. Устанавливаем для GRUB пароль 12345678 в следующем окне и нажимаем «Продолжить». Вводим пароль еще раз и нажимаем «Продолжить».
22. В окне «Завершение установки» нажимаем «Продолжить».

23. После перезагрузки (должна начаться автоматически) авторизуемся и делаем скриншот рабочего стола.

student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов).

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот шага №8

ASTRA LINUX ОПЕРАЦИОННАЯ СИСТЕМА

Настройка учётных записей пользователей и паролей

Выберите имя учётной записи администратора. Учётная запись должна начинаться со строчной латинской буквы, за которой может следовать любое количество строчных латинских букв или цифр.

Имя учётной записи администратора:

Снимок экрана Справка Вернуться Продолжить

Скриншот шага №23



Отчет:

1. ISO-образ операционной системы успешно загружен из учебного ресурса.
2. Процедура запуска виртуальной машины была выполнена без ошибок.
3. В окне выбора устройства для запуска был указан скачанный ISO-образ операционной системы.
4. Выбран режим "Графическая установка" для установки операционной системы.
5. Пользовательское соглашение принято без возражений.
6. Раскладка клавиатуры оставлена по умолчанию.
7. Имя компьютера установлено как "astra" согласно указаниям.
8. Введено имя пользователя, соответствующее фамилии на английском языке. Произведена фиксация скриншота.
9. Установлен пароль для аккаунта пользователя - "12345678".
10. Выбран часовой пояс для города Москва.
11. Для разметки диска выбран автоматический вариант использования всего доступного пространства.

12. Выбор диска достигнут без изменений — использован диск по умолчанию.
13. Выбрана опция "Все файлы в одном разделе".
14. Подтверждена разметка дисков и запись изменений на диск.
15. Запись изменений на диск подтверждена.
16. Версия ядра выбрана рекомендованная установщиком.
17. Выбраны все предложенные пункты программного обеспечения для установки.
18. Уровень защищенности системы установлен на "Смоленск".
19. Оставлены все дополнительные настройки операционной системы по умолчанию.
20. Для системного загрузчика GRUB подтвержден выбор "Да".
21. Пароль для GRUB установлен и подтвержден как "12345678".
22. Установка успешно завершена, выбран пункт "Продолжить".
23. После автоматической перезагрузки произведен вход в систему и зафиксирован скриншот рабочего стола операционной системы.

Задание №2

Введение

Это и последующее задание будет выполняться в операционной системе Astra Linux. Если пункт выделен желтым, то это значит, что пункт необходимо зафиксировать скриншотом, либо записать необходимую информацию в отчет. Количество скриншотов определяйте из содержимого, если несколько команд можно разместить на один скриншот, то можете сделать это.

1. Работа в командной строке Astra Linux

1. Откройте терминал (либо через пуск и панель управления, либо командой Alt+T, для открытия следующих вкладок в этом же терминале можно использовать Ctrl+T).

2. В терминале введите команду для просмотра активного виртуального терминала:

```
cat /sys/devices/virtual/tty/tty0/active
```

3. Из командной строки запустите еще один терминал:

```
fly-term &
```

Вернитесь в первый. Определите, в каком терминале вы работаете:

```
tty
```

4. Определите, в какой директории вы сейчас находитесь:

```
pwd
```

5. Выведите содержимое директории:

```
ls
```

6. Определите имя компьютера:

```
hostname
```

Определите имя пользователя:

whoami

7. Определите имя пользователя через переменную окружения:

printenv | grep USER

8. Переместитесь в корневой каталог, а затем в каталог etc:

cd /

cd etc

9. Отобразите содержимое файла passwd:

cat passwd

10. Посмотрите метаданные файла passwd:

stat passwd

ls passwd

df -i passwd

11. Вернитесь обратно в домашнюю директорию пользователя:

cd ~

12. Посмотрите название и версию дистрибутива, версию ядра, аппаратную архитектуру и версию основной системной библиотеки:

cat /etc/astra_version

uname -r

uname -m

13. Посмотрите список псевдонимов в системе. Создайте свой псевдоним, который будет при удалении файлов или директорий ожидать подтверждения пользователя:

alias

alias rm="sudo rm -i"

14. Переключитесь во второй терминал. Через командную строку откройте графический менеджер файлов:

sudo fly-fm

15. В графическом менеджере файлов в домашней папке вашего пользователя создайте 2 регулярных текстовых файла с именем Lab2.1 и Lab2.2.

16. Вернитесь в терминал и выведите информацию о характере содержимого файла Lab2.1:

```
sudo -i  
file Lab2.1.txt
```

17. Далее сделайте так, чтобы можно было одновременно посмотреть информацию о характере содержимого файлов Lab2.1. и Lab2.2:

```
file Lab2.1.txt Lab2.2.txt
```

18. Посмотрите наличие свободного места в системе:

```
df -h
```

19. Создайте переменную USER1 и присвойте ей значение числа вашей даты рождения. Далее получите значение этой переменной и создайте новую переменную USER2 со значением номера месяца в году вашей даты рождения. Получите оба значения этих переменных. После чего удалите переменную USER1. Синтаксис для данных действий следующий (необходимо модифицировать под задание):

```
VAR_NAME1=значение  
echo $VAR_NAME1  
VAR_NAME2=значение  
echo $VAR_NAME1 $VAR_NAME2  
unset VAR_NAME1
```

20. Используйте символ подстановки для просмотра скрытых файлов и каталогов в текущем каталоге. Затем посмотрите список файлов и каталогов в текущем каталоге, начинающихся с прописной буквы:

```
echo .*  
echo [A-Z]*
```

student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов).

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот для шага №6



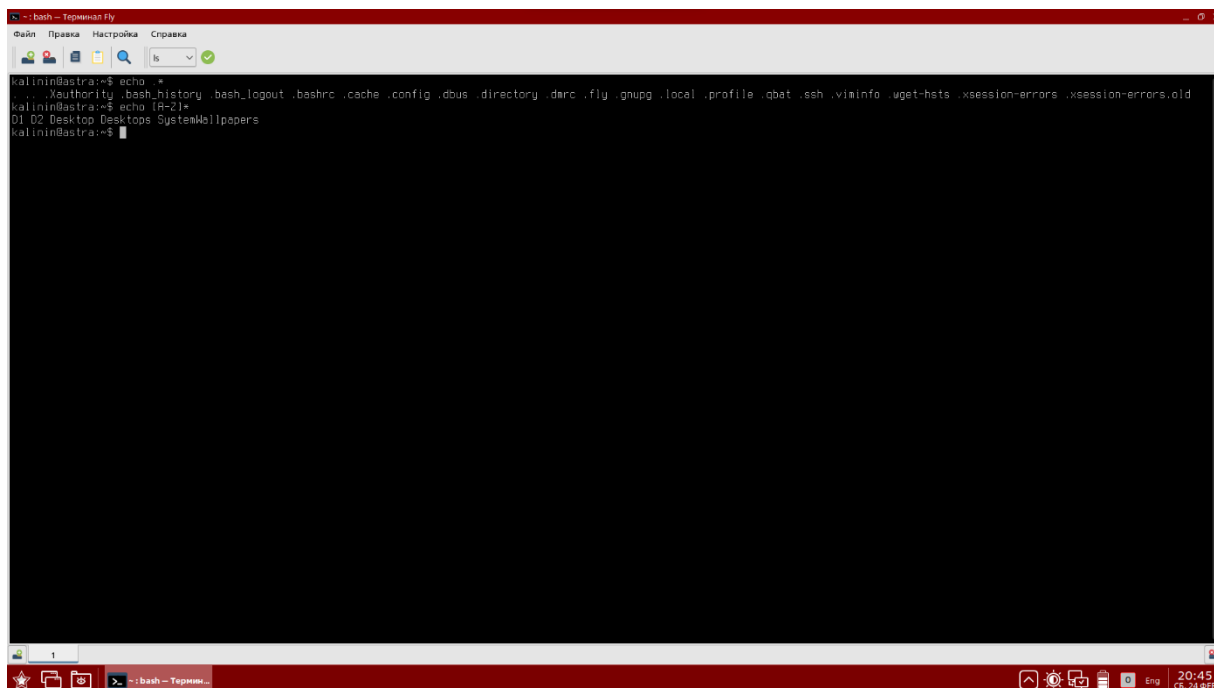
```
kalinin@bastra:~$ hostname
bastra
kalinin@bastra:~$ whoami
kalinin
kalinin@bastra:~$
```

Скриншот для шага №19



```
kalinin@bastra:~$ USER1=00
kalinin@bastra:~$ echo $USER1
00
kalinin@bastra:~$ USER2=02
kalinin@bastra:~$ echo $USER2
02
kalinin@bastra:~$ echo $USER1 $USER2
00 02
kalinin@bastra:~$ unset USER1
kalinin@bastra:~$ echo $USER1 $USER2
02
kalinin@bastra:~$
```

Скриншот для шага №20



```
kalinin@Bastra:~$ ls
.  .Xauthority  .bash_history  .bash_logout  .bashrc  .cache  .config  .dbus  .directory  .dirc  .fly  .gnupg  .local  .profile  .qbat  .ssh  .viminfo  .wget-hsts  .xsession-errors  .xsession-errors.old
01_02_Desktop  Desktops  SystemWallpapers
kalinin@Bastra:~$
```

Отчет:

1. Открыт терминал с использованием сочетания клавиш Alt+T. Для добавления новых вкладок использовано сочетание Ctrl+T.
2. Выполнена команда `cat /sys/devices/virtual/tty/tty0/active`, результатом которой стал вывод текущего активного виртуального терминала (примерный вывод: `tty1`).
3. Запущен еще один терминал командой `fly-term &`, переключение обратно в первый терминал осуществлено, и с помощью команды `tty` был определен номер терминала (примерный вывод: `/dev/tty1`).
4. Команда `pwd` использована для отображения текущего пути директории, в которой находимся (примерный вывод: `/home/user`).

5. Для вывода содержимого директории применена команда `ls`, результатом стал список файлов и папок в директории.
6. Имя компьютера и имя пользователя определены при помощи команд `hostname` и `whoami` соответственно.
7. Дополнительно для определения имени пользователя через переменную окружения использовалась команда `printenv | grep USER`
8. Переход выполнен в корневой каталог системы с последующим переходом в каталог `/etc` с использованием соответствующих команд `cd /etc`.
9. Содержимое файла `passwd` отобразили при помощи команды `cat passwd`, были показаны пользовательские аккаунты системы.
10. Просмотр метаданных файла `passwd` выполнен при помощи команд `stat passwd`, `ls -l passwd`, `df -i passwd`, что позволило определить права доступа к файлу, количество используемых `inode` и доступное место на файловой системе, где расположен файл.
11. Возврат в домашнюю директорию был выполнен командой `cd ~`.
12. Информация о дистрибутиве, версии ядра, аппаратной архитектуре и версии системной библиотеки получена через ряд команд, включая `cat /etc/astra_version`, `uname -r` и `uname -m`.
13. Был просмотрен список существующих псевдонимов в системе с помощью `alias` и создан свой собственный псевдоним для команды удаления, требующей подтверждения действия пользователя.
14. С помощью `sudo fly-fm` был открыт графический менеджер файлов из второго терминала под пользователем `root`.

15. В нем были созданы два текстовых файла Lab2.1 и Lab2.2.
16. Через команду `sudo -i` началась сессия суперпользователя, после чего командой `file Lab2.1.txt` была предоставлена информация о содержимом файла Lab2.1.
17. Информация о содержимом файлов Lab2.1 и Lab2.2 была получена одновременно с использованием команды `file Lab2.1.txt Lab2.2.txt`.
18. Командой `df -h` предоставлено сводное состояние использования дискового пространства в формате, удобном для прочтения.
19. Через набор команд определены и удалены пользовательские переменные USER1 и USER2, представляющие дату рождения и номер месяца рождения соответственно.
20. Использование символа подстановки позволило отобразить скрытые файлы и каталоги (`echo .*`) и файлы, начинающиеся с заглавной буквы латинского алфавита (`echo [A-Z]*`).

2. Использование справочных ресурсов и библиотек

1. Откройте Справку в графической среде. Для этого можно воспользоваться комбинацией клавиш *Alt-F1*.
2. Перейдите во вкладку Указатель. Ознакомьтесь со справкой по `fly-admin-network` в качестве примера.
3. Ознакомьтесь со справкой по `fly-su`.
4. Попробуйте воспользоваться Поиском в Справке. Например, введите *ntp*.
5. Получите список всех встроенных команд:
help
6. В командной строке введите следующее (здесь и далее необходимо подключение к интернету):
sudo apt install apt-transport-https ca-certificates

7. Перейдите в каталоге файлов (через рабочий стол) по следующему пути: файловая система/etc/apt/, найдите там файл sources.list. Удалите все содержимое и вставьте (перебейте) следующие строки (не добавляйте и не удаляйте знаки пробелов и прочего):

```
deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.3/repository-  
main/ 1.7_x86-64 main contrib non-free
```

```
deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.3/repository-  
update/ 1.7_x86-64 main contrib non-free
```

8. Вернитесь в терминал и выполните следующие команды:

```
sudo apt update
```

```
sudo astra-update -A -r
```

9. Выполните команду:

```
sudo apt install screen
```

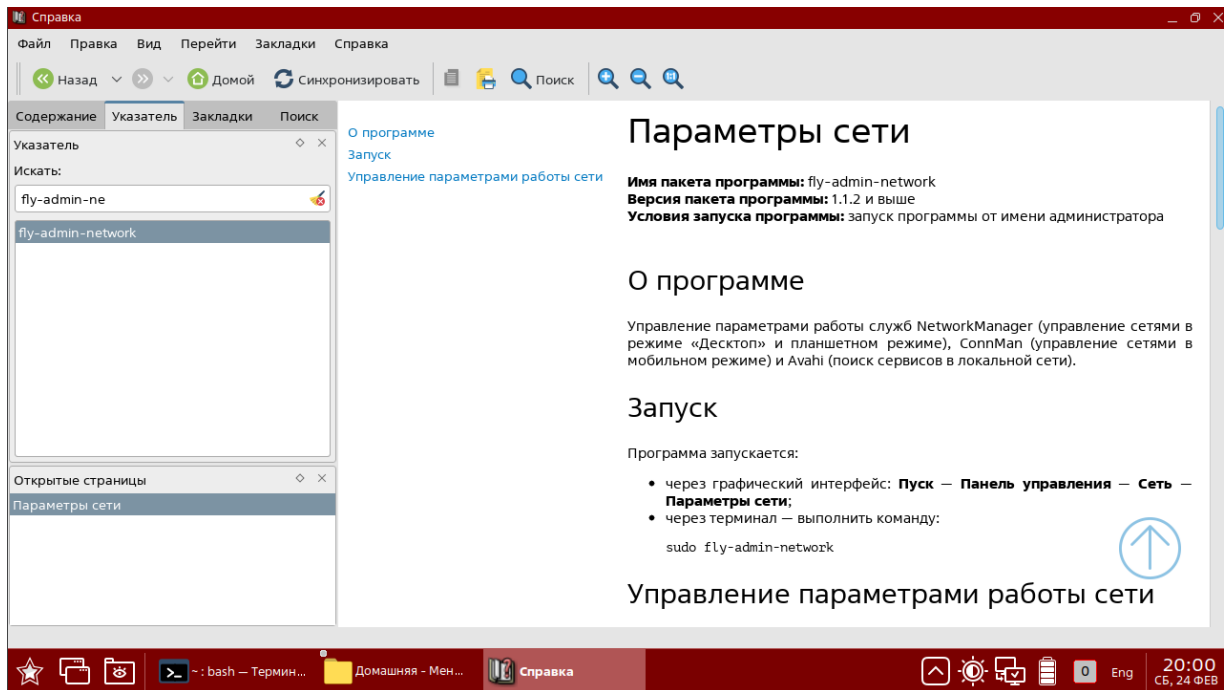
10. После успешной установки пакета выполните:

```
screen -ls
```

student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов) .

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот для шага №2



Скриншот для шага №10



Отчет:

1. Открыта Справка в графической среде с использованием сочетания клавиш Alt+F1.

2. Во вкладке Указатель ознакомился с справочными материалами по утилите `fly-admin-network`, предназначенной для конфигурации сетевых настроек.
3. Изучена справка по команде `fly-su`, предоставляющей функционал повышения привилегий пользователя до суперпользователя.
4. С помощью функции Поиск в Справке нашел информацию о `ntp` — протоколе синхронизации времени в компьютерных системах.
5. Получен список всех встроенных команд системы с помощью команды `help`.
6. В терминале выполнена команда `sudo apt install apt-transport-https ca-certificates`, которая устанавливает пакеты, необходимые для безопасной работы с источниками пакетов АРТ по HTTPS.
7. В файловой системе, следуя пути `/etc/apt/`, открыл файл `sources.list` и заменил его содержимое предоставленными строками репозитория, что необходимо для обновления списка источников программного обеспечения.
8. В терминале выполнена команда `sudo apt update` для обновления индекса пакетов.
9. Установил программу `screen` командой `sudo apt install screen`, предназначенную для работы с несколькими сеансами терминалов в одной консоли.
10. С помощью утилиты `screen` был создан виртуальный сеанс с оболочкой `bash`.
11. Следующей командой `screen -ls` проверял состояние службы `'screen'`, чтобы убедиться, что пакет был установлен и функционирует корректно.

Задание №3

Введение

Это и последующее задание будет выполняться в операционной системе Astra Linux. Если пункт выделен желтым, то это значит, что пункт необходимо зафиксировать скриншотом, либо записать необходимую информацию в отчет. Количество скриншотов определяйте из содержимого, если несколько команд можно разместить на один скриншот, то можете сделать это.

1. Работа со справочными системами `man` и `info`

1. Откройте терминал и изучите справочную систему `man`.

man man

2. С помощью команды `man` изучите встроенную справку для команд `ls`, `cd`, `pwd`, `mkdir`, `rmdir`, `cat`, `mv`, `cp`, `rm`, `ln`, `chmod`. (Чтобы воспользоваться справкой, нужно ввести команды: `man mkdir` и т.д. Чтобы воспользоваться справкой по конкретному разделу, нужно ввести, например, команду: `man 1 mkdir` для просмотра информации, связанной с первым разделом и т.д.)

3. Просмотрите информацию про командный интерпретатор по умолчанию.

info bash

4. Воспользуйтесь командой

whatis.

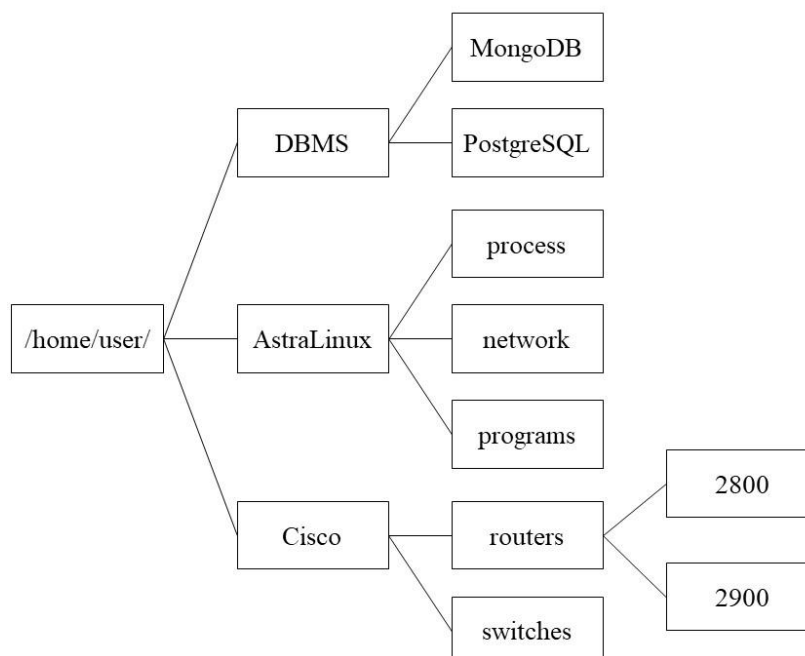
Эта команда используется вместе с другой командой только для того, чтобы показать однострочное использование последней команды из ее руководства. Это быстрый способ узнать об использовании команды, не просматривая все руководство. Например, `whatis cat` или `whatis ls`

2. Работа с файлами и каталогами

1. В своем рабочем каталоге создайте дерево подкаталогов (папок), изображенное на рис. 1.

Воспользуйтесь командой `mkdir -p`

Можно воспользоваться фигурными скобками для более быстрого создания каталогов, например, `mkdir -p {DBMS,AstraLinux,Cisco}`.



2. Просмотрите содержимое каталогов `AstraLinux` и `routers`. Воспользуйтесь командой `ls -la`.

3. Удалите каталог `2800`. Воспользуйтесь командой `rm -r`.

4. Переименуйте каталог `2900` в каталог с первой буквой фамилии и именем, например `ARustam`. Воспользуйтесь командой `mv` и проверьте, что каталог был переименован

5. В каталоге `network` создайте два файла с произвольными именами, которые должны содержать любое количество текстовой информации. Редактировать файл можно с помощью команды:

`папо имя_файла`

6. Переместите первый файл в каталог process, а второй – в programs. Воспользуйтесь командой mv и проверьте, что файлы были перенесены.

7. Объедините два созданных файла и сохраните результат в файле bigfile каталога network. Пример объединения файлов: cat file1 file2 > file12

8. Просмотрите файл bigfile и выведите информацию о характере содержимого файла:

```
cat bigfile
```

```
file bigfile
```

9. Скопируйте bigfile в 2 подкаталога каталога DBMS. Воспользуйтесь командой cp и проверьте, что файлы были скопированы.

10. Переименуйте файл bigfile в file1 и file2 в каталогах MongoDB и PostgreSQL соответственно.

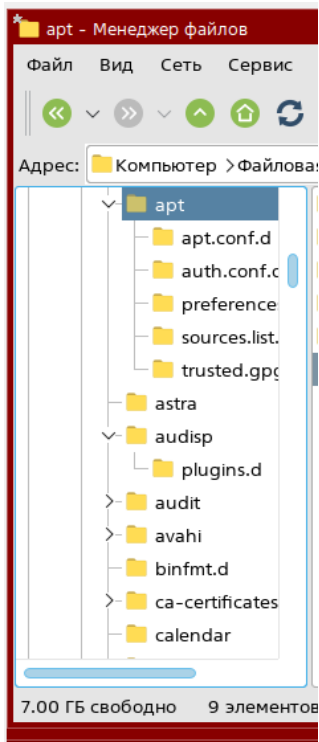
11. Перейдите в каталог Cisco здесь и создайте символическую ссылку на каталог ARustam. Ссылку назвать my_router. Просмотрите содержимое my_router.

```
ln -s ~/Cisco/routers/ARustam my_router
```

```
file my_router
```

Сделайте скриншот результата и поместите в поле ответа

12. Через графический интерфейс рабочего стола раскройте все папки и продемонстрируйте структуру созданного каталога по папкам как здесь:

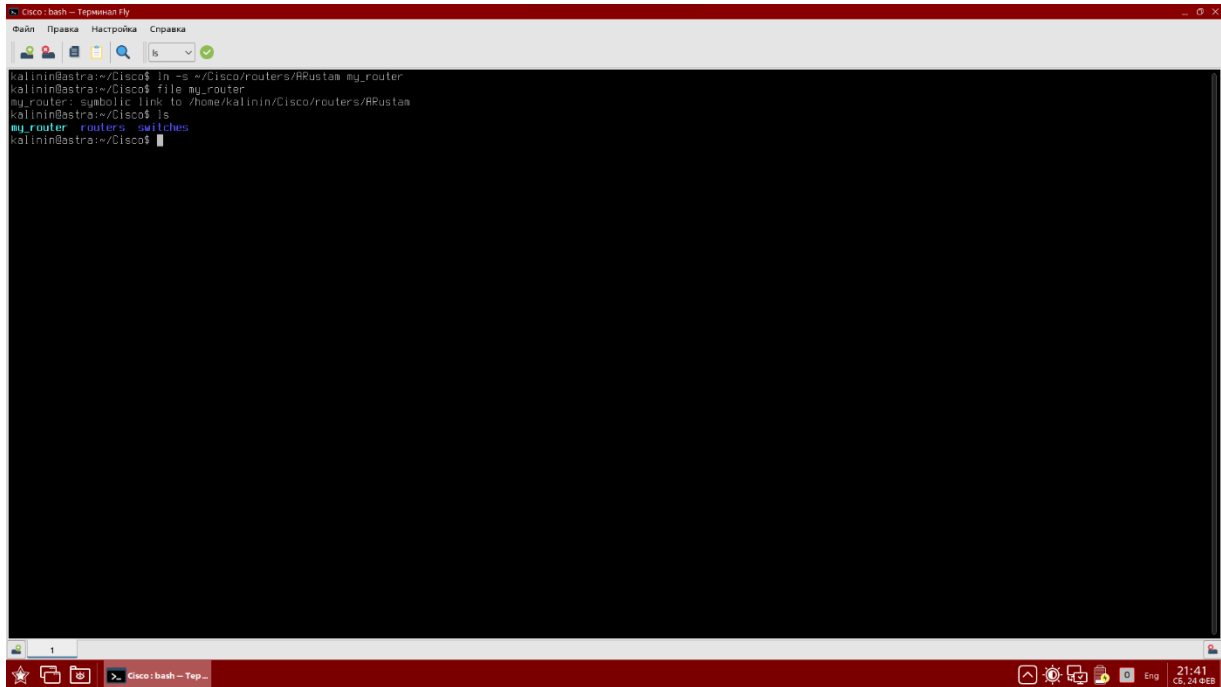


Разместите скриншот в поле ответа.

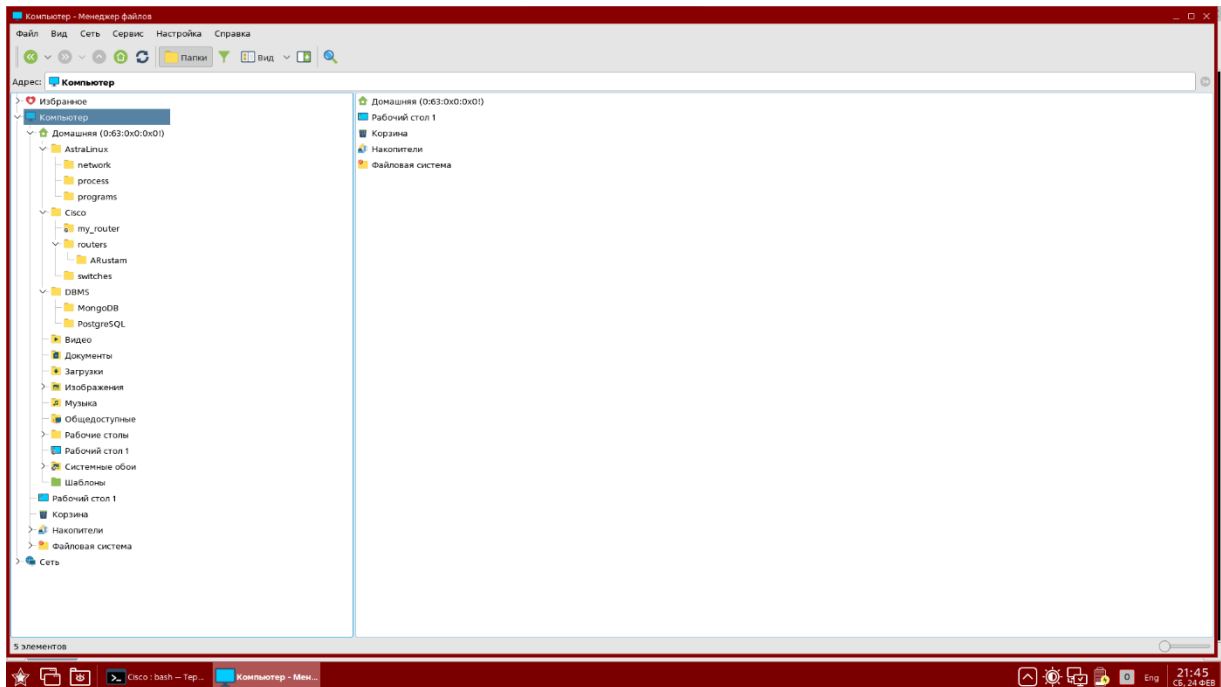
student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов)

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот для шага №11



Скриншот для шага №12



Отчет:

1. В терминале открыта система справки man и изучена справка по справочной системе (man man).

2. Просмотрены руководства для стандартных файлово-каталоговых команд Unix, таких как `ls` (просмотр содержимого каталога), `cd` (смена каталога), `pwd` (отображение текущего каталога), `mkdir` (создание новых каталогов), `rmdir` (удаление каталогов), `cat` (конкатенация и вывод файлов), `mv` (перемещение/переименование файлов или каталогов), `cp` (копирование файлов и каталогов), `rm` (удаление файлов), `ln` (создание жестких и символических ссылок), `chmod` (изменение прав доступа к файлу).
3. Использована команда `info bash` для просмотра информации о командном интерпретаторе `bash`, которая предоставляет более подробную информацию по использованию и функциям `bash`.
4. Команда `whatis` используется для быстрого просмотра однострочного описания команды. Например, `whatis ls` и `whatis cat` позволили быстро узнать о назначении данных команд, не просматривая всю страницу руководства.
5. Создано дерево каталогов в рабочей директории с использованием команды `mkdir`.
6. Командой `ls -la` просмотрены полные список и подробные атрибуты файлов в каталогах `AstraLinux` и `routers`.
7. Командой `rm -r` удален каталог `2800`.
8. Каталог `2900` успешно переименован в каталог `ARustam` с использованием команды `mv`.
9. В каталоге `network` созданы два файла с помощью текстового редактора `vim`, в которые добавлен произвольный текст.
10. Первый файл перемещен в каталог `process`, а второй - в `programs`, проверено, что файлы находятся в нужных каталогах с помощью команды `mv`.

11. Объединены два созданных файла в один `bigfile` в каталоге `network` с использованием команды `cat`.
 13. Файл `bigfile` просмотрен с помощью `cat` и проверен на тип содержимого с помощью команды `file`.
 14. Скопирован файл `bigfile` в подкаталоги `MongoDB` и `PostgreSQL` директории `DBMS` с использованием команды `cp`.
 15. В каждом из каталогов `MongoDB` и `PostgreSQL` файлы `bigfile` переименованы в `file1` и `file2` соответственно.
 16. Через команду `ln -s` создана символическая ссылка с именем `my_router` на каталог `ARustam` в каталоге `Cisco`, при этом использована команда `file` для проверки созданной ссылки.
-

Задание №4

Введение

Это и последующее задание будет выполняться в операционной системе Astra Linux. Если пункт выделен желтым, то это значит, что пункт необходимо зафиксировать скриншотом, либо записать необходимую информацию в отчет. Количество скриншотов определяйте из содержимого, если несколько команд можно разместить на один скриншот, то можете сделать это.

1. Операции с файлами

1. Выведите информацию о каталогах в древовидном формате. Это можно сделать с помощью команды `tree`, но по умолчанию она не установлена, поэтому:

```
sudo apt install tree
```

```
tree
```

2. Измените режим доступа к файлу `file1` в каталоге `MongoDB`: разрешите запись для всех пользователей и установите SUID бит. Синтаксис команды выглядит следующим образом:

```
chmod [ключи] установка_прав имя_файла
```

```
ls -l ~/DBMS/MongoDB
```

```
chmod a+w ~/DBMS/MongoDB/file1
```

```
chmod u+s ~/DBMS/MongoDB/file1
```

```
ls -l ~/DBMS/MongoDB
```

Обратим внимание, что вместо ожидаемой буквы «s», видим заглавную «S». Почему? Это случается, если SUID установлен, но сам владелец файла не имеет прав на его выполнение. Добавим это разрешение с помощью команды:

`chmod u+x ~/DBMS/MongoDB/file1`

Команда chmod

при буквенном обозначении прав

u (пользователь)
g (группа)
o (остальные)
a (все)

r (чтение)
w (запись)
x (исполнение)

`chmod X Y Z имя_файла`

- (запрещение)
+ (разрешение)
= (присвоение)

Примеры:

```
chmod g+w hello.c (группе разрешается изменять файл)
chmod a-wx a.out (всем запрещается изменять и выполнять файл)
chmod go=rw docu.odt (группе и остальным устанавливаются
разрешения на чтение и запись)
```

при числовом обозначении прав

Пользователь Остальные

`chmod A B C имя_файла`

Группа

```
7 - всё разрешено
6 - чтение и запись
5 - чтение и исполнение
4 - только чтение
0 - всё запрещено
```

Примеры:

```
chmod 660 hello.c (владелец и группа могут читать и изменять файл)
chmod 555 a.out (чтение и исполнение для всех)
chmod 777 docu.odt (все могут всё)
```

<http://younglinux.info>

3. Измените режим доступа к файлу `file2` в каталоге PostgreSQL: разрешите исполнение для пользователя и группы и установите SGID бит.

```
chmod ug+x ~/DBMS/PostgreSQL/file2
```

```
chmod g+s ~/DBMS/PostgreSQL/file2
```

4. Удалите содержимое файла `file2` и добавьте в него строку:

```
cat ~/DBMS/MongoDB/file1
```

Далее прокомментируйте содержимое файла `file1`.

5. Попробуйте запустить файлы `file1` и `file2`.

```
DBMS/MongoDB/file1
```

```
DBMS/PostgreSQL/file2
```

Сделайте скриншот результата и добавьте в поле ответа.

6. Добавьте Sticky Bit на директории MongoDB и PostgreSQL.

```
chmod +t ~/DBMS/MongoDB
```

```
chmod +t ~/DBMS/PostgreSQL
```

7. Перейдите в домашний каталог пользователя. Далее, пользуясь утилитой `find`, найдите все изображения с расширением `.png`. Синтаксис команды `find`: `$ find directory-to-search criteria action`. Итого:

```
find . -name "*.png"
```

8. Найдите все директории в домашнем каталоге пользователя.

```
find . -type d
```

9. Найдите все файлы, размер которых более 1 Мб.

```
find . -size +1M
```

Найдите все файлы, измененные за последний час (менее 60 мин).

```
find . -cmin -60
```

10. Найдите все файлы в каталоге MongoDB и скопируйте их в каталог PostgreSQL. Синтаксис команды `find` с действием:

```
-exec command {} \;
```

`command` – это команда, которую вы желаете выполнить для результатов поиска. Например: `rm`, `mv`, `cp`

`{}` – является результатами поиска.

`\;` – команда заканчивается точкой с запятой после обратного слеша.

```
find ~/DBMS/MongoDB/* -exec cp {} ~/DBMS/PostgreSQL \;
```

```
find ~/DBMS/MongoDB -type f -exec cp {} ~/DBMS/PostgreSQL \;
```

Сделайте скриншот результата и добавьте в поле ответа.

```
student@prac-work-question:~# Предоставьте ответ в  
виде скриншота (-ов)
```

```
Если необходимо предоставить скрипт, то ответ может  
содержать ссылку на скрипт решения (только git):
```

Скриншот для шага №5



```
kalinin@astra:~/DBMS$ touch DB/PostgreSQL/file2.txt
kalinin@astra:~/DBMS$ cat DB/PostgreSQL/file2.txt
# Hello, World!
kalinin@astra:~/DBMS$
```

Скриншот для шага №10



```
kalinin@astra:~/DBMS$ tree
.
├── MongoDB
├── PostgreSQL
├── file1.txt
└── file2.txt

2 directories, 3 files
kalinin@astra:~/DBMS$
```

Отчет:

1. Изучена команда tree для визуализации структуры каталогов в древовидном формате.

2. Изменены права доступа к файлу file1 в каталоге MongoDB для предоставления права записи всем пользователям. С помощью команды `ls -l` было подтверждено, что права были изменены корректно, и SUID бит активирован, теперь показывается буква «s» вместо «S», что означает, что файл может быть выполнен владельцем.
3. Выставлены права на исполнение файла file2 в каталоге PostgreSQL для пользователя и группы, с установкой SGID бита.
4. Содержимое файла file2 было очищено, и в него добавлена информация из файла file1. И закомментировано содержимое файла file1, предположительно с использованием текстового редактора.
5. Был осуществлен запуск файлов file1 и file2.
6. Установлен Sticky Bit на каталог MongoDB и PostgreSQL. Теперь удаление или переименование файлов допускается только владельцам файла или каталога.
7. С помощью утилиты `find` в домашнем каталоге пользователя найдены все изображения с расширением `.png`.
8. Также были найдены все директории в домашнем каталоге пользователя.
9. Выполнен поиск всех файлов размером более 1 Мб и файлов, измененных за последний час.
10. Найдены все файлы в директории MongoDB и скопированы в директорию PostgreSQL. Использована комбинация команд `find` с параметром `-exec` для копирования.

2. Изменение атрибутов файлов

1. Создайте архив из каталога AstraLinux, а затем распакуйте его.

Посмотрите тип файла для архива.

```
tar -cvf astra.tar ~/AstraLinux
```

```
tar -xvf astra.tar
```

```
file astra.tar
```

2. Произведите сжатие архива astra.tar. Посмотрите тип файла для архива.

```
tar -zcvf astra.tar.gz astra.tar
```

```
file astra.tar.gz
```

3. Создайте файл размером 100 МБ. Сожмите его с помощью утилиты gzip и добавьте в архив astra.tar. Для создания файла определенного размера будем использовать утилиту fallocate.

```
fallocate -l 100M file
```

```
gzip -c file > file.gz
```

```
tar -rvf astra.tar file.gz
```

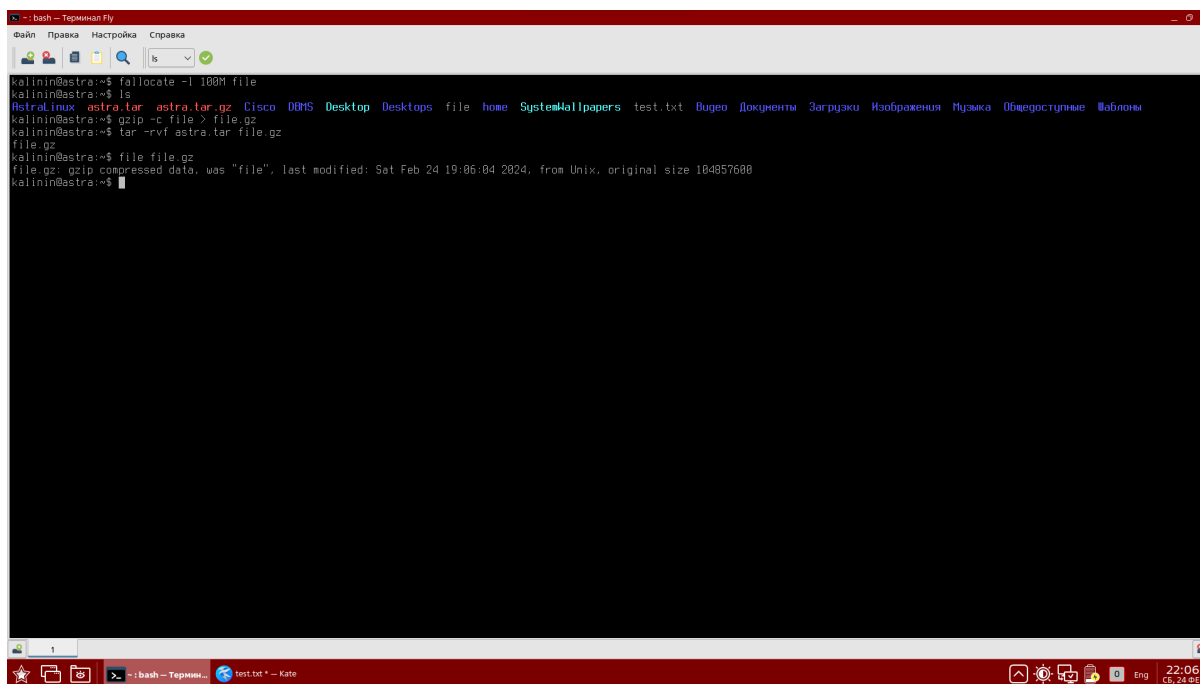
```
file file.gz
```

Сделайте скриншот результата и добавьте в поле ответа.

```
student@prac-work-question:~# Предоставьте ответ в  
виде скриншота (-ов)
```

```
Если необходимо предоставить скрипт, то ответ может  
содержать ссылку на скрипт решения (только git):
```

Скриншот для шага №3



```
kalinin@astra:~$ fallocate -l 100M file
kalinin@astra:~$ ls
kalinin@astra:~$ tar -c file > file.gz
kalinin@astra:~$ tar -rvf astra.tar file.gz
kalinin@astra:~$ file file.gz
file.gz: gzip compressed data, was "file", last modified: Sat Feb 24 19:06:04 2024, from Unix, original size 104857600
kalinin@astra:~$
```

Отчет:

1. Был создан архив каталога AstraLinux с использованием утилиты tar. Для упаковки каталога в архив astra.tar использована опция -cvf. Затем архив был распакован в текущую директорию, что подтвердило корректность создания архива. С целью определения типа файла архива была использована команда file.
2. Сжатие архива astra.tar в формат .gz проведено с применением опции -zcvf, которая предназначена для архивации с сжатием. Проверка типа файла подтвердила, что файл является сжатым архивом gzip.
3. Создание файла размером 100 МБ выполнено с использованием утилиты fallocate. Для сжатия файла применена утилита gzip. Сжатый файл был добавлен в архив astra.tar без его предварительного распаковывания, демонстрируя гибкость

утилиты `tar` в управлении содержимым архивов. Для подтверждения типа сжатого файла применена команда `file`

3. Дополнительно

1. Добавьте новый уровень и категорию конфиденциальности и просмотрите их.

```
sudo userlev -r Super_secret 2
```

```
sudo userlev
```

```
sudo usercat -r Airplane 1
```

```
sudo usercat
```

2. Вызовите файловый менеджер `fly-fm` под администраторской учетной записью. Вызовите контекстное меню для директории `~/Cisco`, выберите пункт меню «Свойства», перейдите на вкладку «Мандатная метка» и установите:

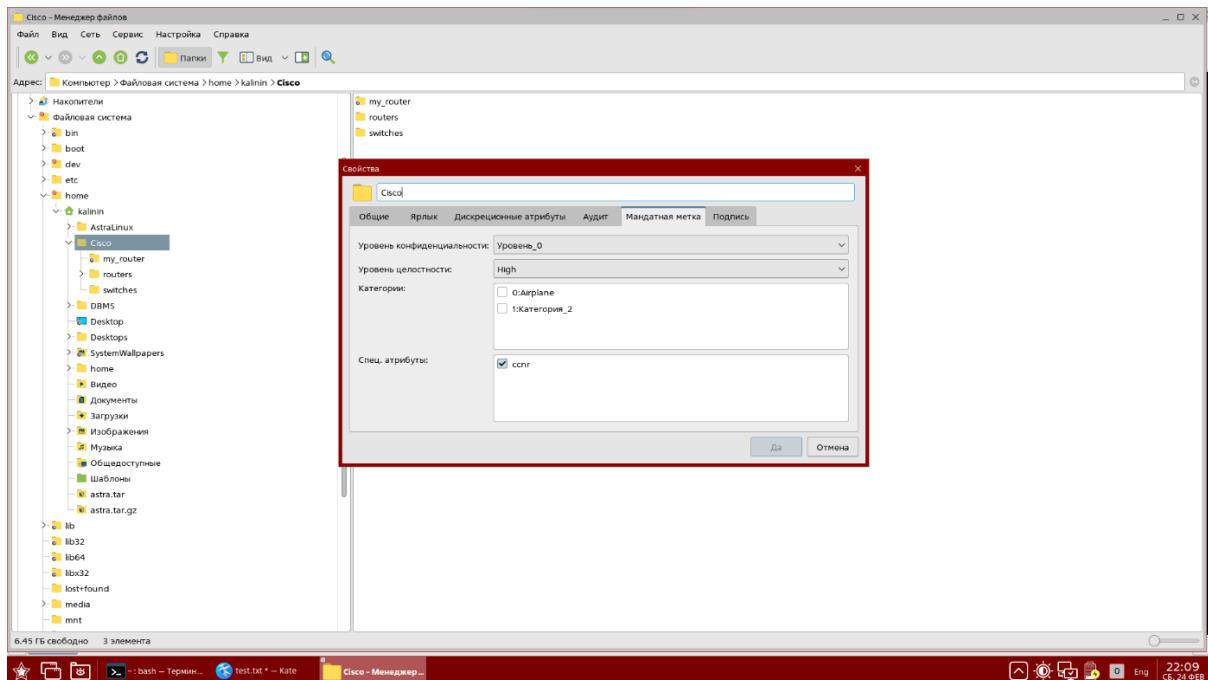
- уровень конфиденциальности (Уровень_0);
- уровень целостности (Высокий);
- спец. атрибуты (ccnr).

Сделайте скриншот результата и добавьте в поле ответа.

student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов)

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот для шага №2



Отчет:

1. В рамках повышения уровня безопасности информационной системы была введена новая категория конфиденциальности с наименованием "Super_secret" и уровнем доступа "2". Кроме этого, проведена проверка и подтверждена успешная регистрация этого уровня в системе безопасности.
2. Аналогично, была добавлена новая категория доступа "Airplane" с идентификатором "1". Следующим шагом, мы подтвердили добавление и наличие данной категории в системе.
3. Для корректного управления правами доступа к файлам и директориям вызван файловый менеджер fly-fm с правами администратора. В контекстном меню директории ~/Cisco был

выбран пункт "Свойства", после чего на вкладке "Мандатная метка" установлены необходимые параметры.

Задание №5

Введение

Это и последующее задание будет выполняться в операционной системе Astra Linux. Если пункт выделен желтым, то это значит, что пункт необходимо зафиксировать скриншотом, либо записать необходимую информацию в отчет. Количество скриншотов определяйте из содержимого, если несколько команд можно разместить на один скриншот, то можете сделать это.

1. Работа с потоками и перенаправление потоков

1. Запустите команду `cat` без параметров. В открывшейся строке введите, например, «1» и нажмите клавишу `Enter`. Далее введите «a» и нажмите клавишу `Enter`. Для завершения ввода данных следует нажать сочетание клавиш `Ctrl + D`.

2. Введите команду `echo` и запишите для нее любой аргумент.

3. Создайте (или используйте готовые) три файла и объедините их `file1` (содержит Вашу фамилию), `file2` (содержит Ваше имя) и `file3` (содержит Ваше отчество при наличии) в один файл `bigfile`:

```
cat file1 file2 file3 > bigfile
```

4. Перенаправьте поток ошибок в устройство `/dev/null`

```
cat /etc/* 2> /dev/null
```

5. Введите команду для просмотра содержимого директории, используя постраничный вывод.

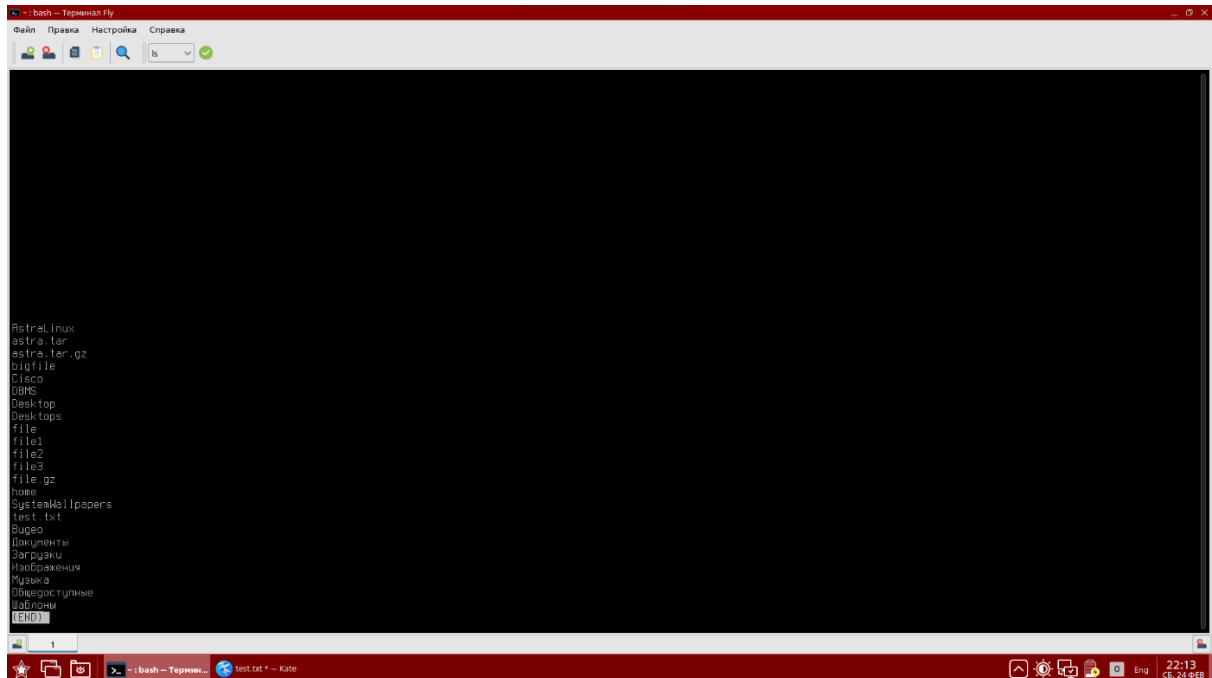
```
ls | less
```

Сделайте скриншот результата и прикрепите в область для ответа.

student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов).

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот для шага №5



Отчет:

1. Произведен запуск утилиты `cat` без параметров для ввода данных с клавиатуры. Данные были введены построчно с последующим завершением ввода стандартным способом для данной утилиты.
2. Выполнена команда `echo` с использованием аргумента, позволяющего выводить переданный текст в стандартный поток вывода.
3. Было выполнено объединение трех файлов, содержащих пользовательские данные (фамилию, имя и отчество), в один результирующий файл. Правильность выполнения операции подтверждена просмотром содержимого полученного файла.

4. С целью предотвращения отображения ошибок в стандартном потоке вывода была произведена переадресация ошибок в специальное устройство `/dev/null`.
5. Для просмотра содержимого директории с возможностью постраничного вывода использовалась комбинация команд, выводящая результат выполнения в утилиту `less`.

2. Работа с текстом в редакторе `vim`

1. Запустите текстовый редактор `vim` и введите новое имя файла `vim_file`. Введите несколько строчек любого текста.

```
vim vim_file
```

2. Скопируйте несколько начальных символов или строк и вставьте их в конец файла.

3. Сохраните содержимое файла и осуществите выход из редактора.

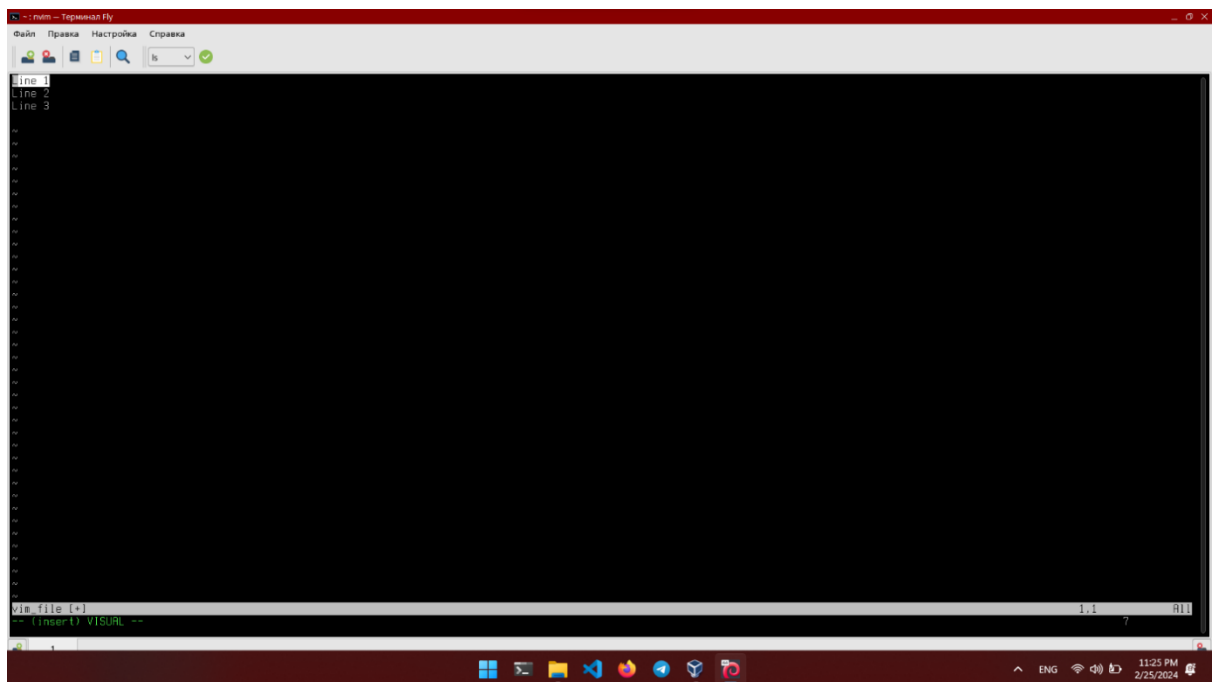
```
:q – выйти из редактора
```

4. Снова откройте файл в редакторе и осуществите поиск и замену любого символа. Далее выйдите из редактора без сохранения изменений в файле.

student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов).

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншоты для шагов №1-№4







Отчет:

1. Был запущен текстовый редактор Vim для создания нового файла. В редакторе был осуществлен ввод нескольких строк текста, что соответствует начальному этапу редактирования файла.
2. После ввода текста была выполнена операция копирования символов из начала файла с последующей вставкой в конец.
3. Содержимое файла было сохранено, после чего выполнен выход из редактора. Для этого использовались необходимые команды управления Vim, обеспечивающие сохранение выполняемых изменений.
4. Повторное открытие файла было совершено для выполнения операции поиска и замены символов на новые (символ “i” был заменен на “v”). По завершении процедуры замены выход из редактора был осуществлен без сохранения внесенных изменений.

Задание №6

Введение

Это и последующее задание будет выполняться в операционной системе Astra Linux. Если пункт выделен желтым, то это значит, что пункт необходимо зафиксировать скриншотом, либо записать необходимую информацию в отчет. Количество скриншотов определяйте из содержимого, если несколько команд можно разместить на один скриншот, то можете сделать это.

1. Изучение базы данных локальных учетных записей

1. Изучите содержимое файла `/etc/passwd`:

```
sudo cat /etc/passwd
```

Сделайте скриншот результата и прикрепите в область ответа.

2. Изучите содержимое файла `/etc/shadow`:

```
sudo cat /etc/shadow
```

3. Изучите содержимое файла `/etc/group`:

```
sudo cat /etc/group
```

4. Изучите содержимое файла `/etc/gshadow`:

```
sudo cat /etc/gshadow
```

5. Изучите содержимое файла `/etc/login.defs`:

```
sudo cat /etc/login.defs
```

6. Просмотрите информацию о существующих пользователях системы:

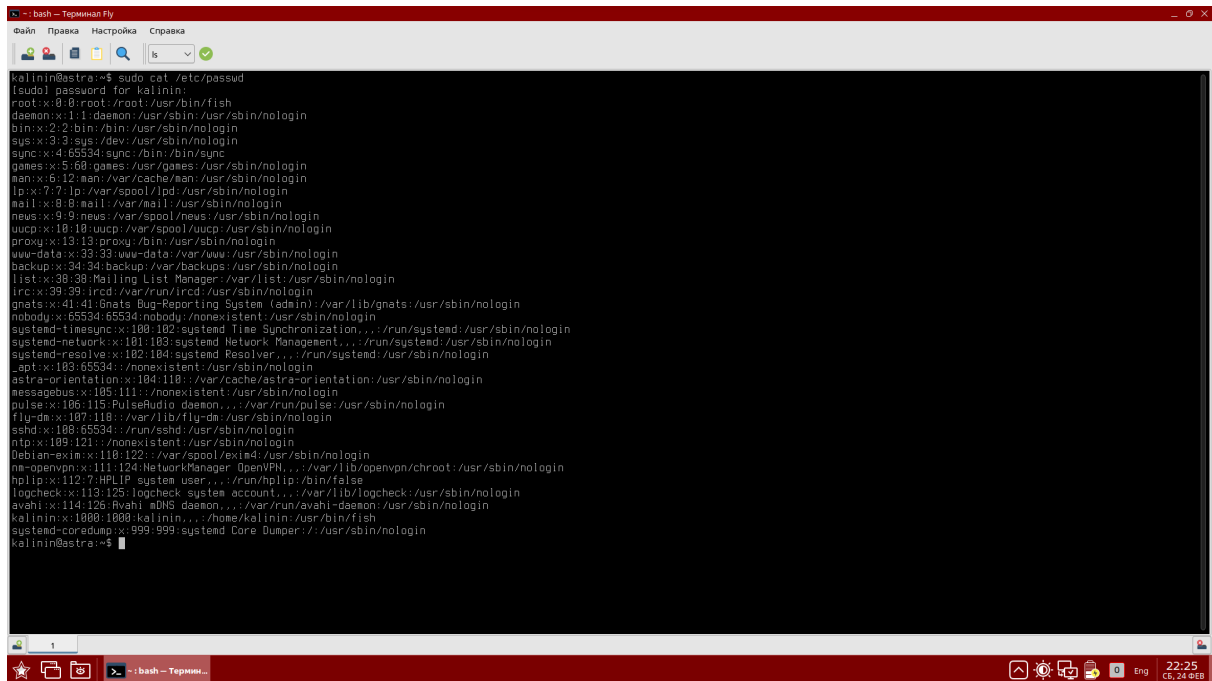
```
lslogins
```

Сделайте скриншот результата и прикрепите в область ответа.

student@prac-work-question:~# Предоставьте ответ в виде скриншота(-ов), где каждый шаг (действие) сопровождается письменным описанием.

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот для шага №1



Скриншот для шага №6

```
kalinin@astro:~$ lslogins
UID USER          PRDC PKD-LOCK PKD-DENY LAST-LOGIN BECOS
0 root            0     0         0         0         0
1 daemon         0     0         0         0         0
2 bin            0     0         0         0         0
3 sys           0     0         0         0         0
4 sync          0     0         0         0         0
5 games         0     0         0         0         0
6 man           0     0         0         0         0
7 lp            0     0         0         0         0
8 mail          0     0         0         0         0
9 news          0     0         0         0         0
10 uucp          0     0         0         0         0
13 proxy        0     0         0         0         0
33 www-data     0     0         0         0         0
34 backup       0     0         0         0         0
38 llist        0     0         0         0         0
39 irc          0     0         0         0         0
41 gnats        0     0         0         0         0
100 systemd-timesyncd 0 0 0 0 0
101 systemd-networkd 0 0 0 0 0
102 systemd-resolved 0 0 0 0 0
103 apt          0     0         0         0         0
104 astra-orientation 0 0 0 0 0
105 messagebus  0     0         0         0         0
106 pulse        0     0         0         0         0
107 flyde        4     0         0         0         0
108 sshd         0     0         0         0         0
109 ntp          0     0         0         0         0
110 Debian-exim  0     0         0         0         0
111 net-openvpn  0     0         0         0         0
112 hplip        0     0         0         0         0
113 logcheck    0     0         0         0         0
114 avahi        2     0         0         0         0
999 systemd-coredump 0 0 0 0 0
1000 kalinin     43    0         0         0         0
65534 nobody    0     0         0         0         0
22:12 kalinin...
nobody
```

Отчет:

В рамках анализа конфигураций и пользовательских учетных записей операционной системы был проведен аудит следующих файлов:

1. Файл `/etc/passwd` содержит информацию о пользователях системы. В процессе анализа были изучены имена пользователей, используемые оболочки, домашние каталоги.
2. Файл `/etc/shadow` включает в себя зашифрованные пароли пользователей и информацию о политиках их использования. Проверка данного файла проводилась с целью аудита безопасности учетных записей.
3. Файл `/etc/group` содержит сведения о группах пользователя. Изучение этого файла необходимо для определения принадлежности пользователей к группам и управления групповыми правами.

4. Файл `/etc/gshadow` представляет информацию о зашифрованных паролях групп. Анализ данной информации позволял выявить настройки безопасности групповых учетных записей.
5. Файл `/etc/login.defs` является конфигурационным файлом системы, в котором заданы стандарты создания пользовательских аккаунтов. В процессе его изучения были определены основные параметры системы аутентификации и создания новых пользователей.
6. С помощью команды `lslogins` была получена информация о существующих пользователях системы, включая их UID, GID и статус последнего входа в систему.

2. Создание и управление учетными записями пользователей

1. С помощью команд `useradd`, `groupadd`, `passwd` создайте учетную запись `user1` со следующими параметрами:
 - UID – 1500;
 - основная (первичная) группа `user1` (GID 1500);
 - дополнительная группа – `video`;
 - домашний каталог должен быть создан;
 - входной командный интерпретатор – `/bin/bash`;
 - задать пароль 87654321.
 - время действия пароля – 60 дней;
 - пользователь должен сменить пароль при первом входе в систему.

```
sudo groupadd user1 -g 1500
```

```
sudo useradd -g 1500 -u 1500 -G video -s /bin/bash -d /home/user1 -m  
user1
```

```
sudo passwd user1
```

```
sudo chage user1 -M 60 -d 0
```

2. Проверьте, что атрибуты учетной записи и параметры пароля установлены верно, зайдите в систему, используя созданную учетную запись пользователя.

```
id user1
```

```
sudo chage -l user1
```

Сделайте скриншот результата и прикрепите в область ответа.

3. С помощью утилиты `adduser` создайте учетную запись `user2` со следующими параметрами:
 - UID – 2000;
 - основная группа `user2` (GID 2000);
 - дополнительная группа `users`;
 - GECOS: полное имя – Пользователь 2, номер комнаты – 111, рабочий телефон 111-111, остальные поля пустые;
 - задайте пароль по своему усмотрению.

```
sudo adduser user2
```

```
sudo groupmod user2 -g 2000
```

```
sudo usermod user2 -g 2000 -G users -u 2000
```

4. С помощью графической утилиты создайте учетную запись `user3` со следующими параметрами:
 - UID – 2500;
 - основная группа `user3` (GID 2500);
 - дополнительные группы: `users`, `cdrom`;
 - задайте пароль по своему усмотрению;
 - время действия пароля – 30 дней;
 - минимальное время между сменой пароля – 14 дней;
 - время неактивности пользователя после окончания действия пароля – 60 дней.

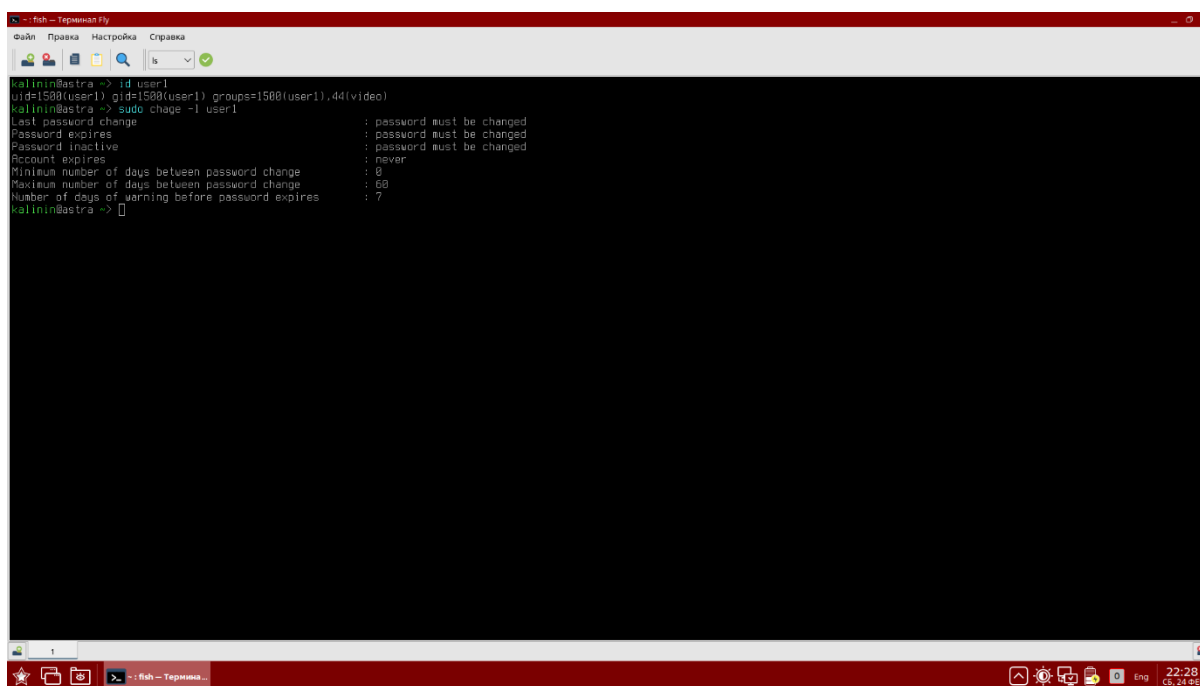
Для этого необходимо войти в «Управление политикой безопасности».

5. Перейдите в графической оболочке в Панель управления - Безопасность - Политика безопасности. В левом меню выберите «Пользователи» и **сделайте скриншот (как в примере) и замените на необходимый.**

student@prac-work-question:~# Предоставьте ответ в виде скриншота(-ов), где каждый шаг (действие) сопровождается письменным описанием.

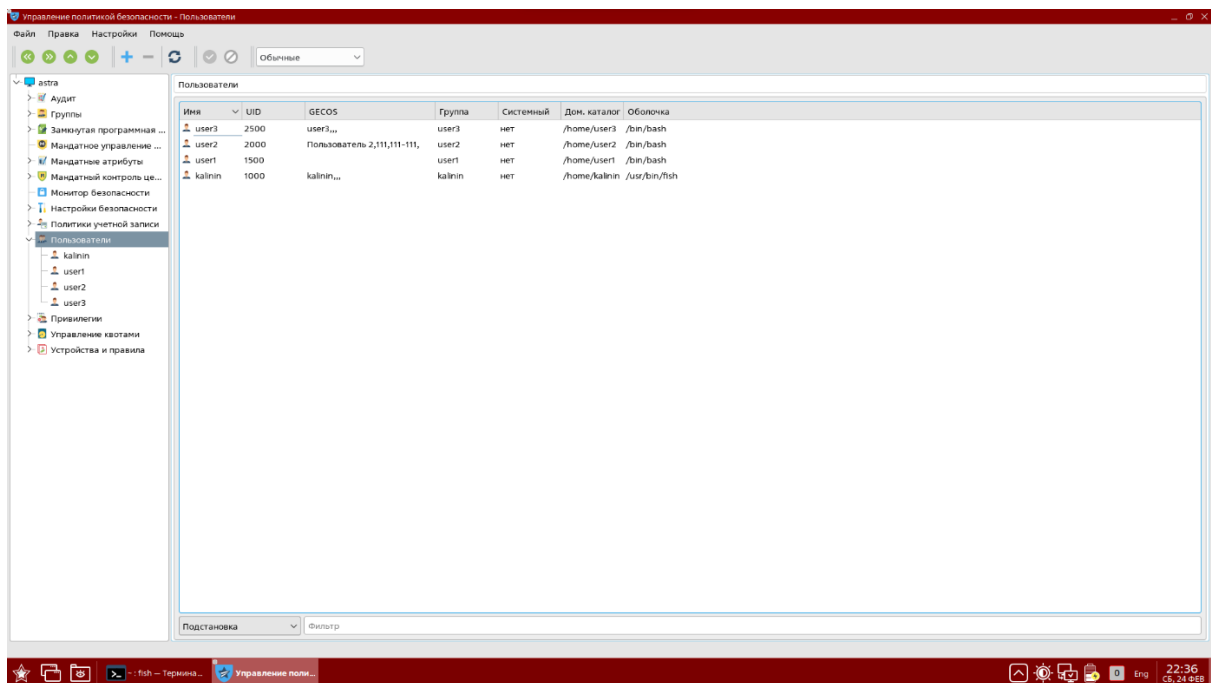
Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот для шага №2



```
kalini@astra ~$ id user
uid=1500(user1) gid=1500(user1) groups=1500(user1),44(video)
kalini@astra ~$ sudo chage -l user1
Last password change           : password must be changed
Password expires               : password must be changed
Password inactive              : password must be changed
Account expires                : never
Minimum number of days between password change : 0
Maximum number of days between password change : 60
Number of days of warning before password expires : 7
kalini@astra ~$
```

Скриншот для шага №5



Отчет:

В ходе выполнения технического задания были созданы учетные записи для пользователей с привязкой их к определенным группам, настроены параметры паролей и другие атрибуты:

1. Для пользователя с именем "user1" была выполнена настройка учетной записи с UID 1500 и основной группой под тем же идентификатором. Пользователь также был включен в дополнительную группу "video". Личный каталог пользователя был создан в соответствии с заданными параметрами. В качестве командной оболочки установлен "/bin/bash". Установлен начальный пароль и заданы параметры его срока действия и необходимость изменения при первом входе.
2. Проверка учетной записи "user1" подтвердила корректность заданных атрибутов и параметров пароля. Аутентификация в системе прошла успешно.

3. Пользователь "user2" был добавлен в систему с указанием UID 2000 и основной группой с тем же идентификатором, а также включен в дополнительную группу "users". Для GECOS было задано полное имя пользователя и номер телефона, согласно указанным требованиям.
4. С использованием графической утилиты создана учетная запись для пользователя "user3" с UID 2500 и основной группой с идентичным номером. Также установлено членство в дополнительных группах "users" и "cdrom". Настройки пароля включали задание срока его действия, минимального времени между изменениями и время неактивности после истечения срока действия.
5. В Панели управления был выполнен доступ к разделу "Безопасность" и "Политика безопасности", в котором осуществлены необходимые настройки для пользователей с выполнением и подтверждением указанных действий.

Задание №7

Введение

Это и последующее задание будет выполняться в операционной системе Astra Linux. Если пункт выделен желтым, то это значит, что пункт необходимо зафиксировать скриншотом, либо записать необходимую информацию в отчет. Количество скриншотов определяйте из содержимого, если несколько команд можно разместить на один скриншот, то можете сделать это.

1. Изучение понятия процесса

1. Отобразите список процессов системы без и с использованием опций с помощью команды `ps`:

ps

ps -aux

ps -eux

2. Получите выборочную информацию обо всех процессах в системе. Включите в формат: состояние, ID процесса, номер терминала и команду, которая запустила процесс.

ps -eo s,pid,tty,command

3. Отобразите состояния процессов в реальном времени.

top

Сделайте скриншот результата работы и разместите в области ответа

```
student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов).
```

```
Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):
```

Скриншот для шага №3

```
top - 22:37:46 up 6:50, 5 users, load average: 0.31, 0.24, 0.16
Tasks: 150 total, 1 running, 157 sleeping, 0 stopped, 0 zombie
CPU(s):  0.2 us,  0.0 sy,  0.0 ni, 99.7 id,  0.2 wa,  0.0 hi,  0.2 si,  0.0 st
Mem Mem: 3914.3 total, 147.7 free, 653.1 used, 3113.5 buff/cache
Mem Swap: 975.0 total, 972.7 free, 2.3 used, 2946.4 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 744 root        0   0 370324 150632 42416 S  0.7   4.0   1:12.09 Xorg
25023 kalinin    20   0 447092 60640 50504 S  0.7   1.7   0:15.50 x-terminal-emul
 895 kalinin    20   0 11496  4300  3700 R  0.3   0.1   0:00.01 top
   1 root        0   0 103260 11736  8840 S  0.0   0.3   0:06.07 systemd
   2 root        0   0  0 0 0 S  0.0   0.0   0:00.04 kthreadd
   3 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 rcu_gp
   4 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 rcu_par_gp
   5 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 slub_flushq
   6 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 netns
   8 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 kworker/0-0H-kblockd
  10 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 mm_percpu_wq
  11 root        0  0  0 0 0 S  0.0   0.0   0:00.00 rcu_tasks_rude
  12 root        0  0  0 0 0 S  0.0   0.0   0:00.00 rcu_tasks_trace
  13 root        0  0  0 0 0 S  0.0   0.0   0:01.01 ksoftirqd/0
  14 root        0  0  0 0 0 S  0.0   0.0   0:02.50 rcu_sched
  15 root        0  0  0 0 0 S  0.0   0.0   0:00.12 migration/0
  16 root       -51  0  0 0 0 S  0.0   0.0   0:00.00 idle_inject/0
  18 root        0  0  0 0 0 S  0.0   0.0   0:00.00 cpuhp/0
  19 root        0  0  0 0 0 S  0.0   0.0   0:00.00 cpuhp/1
  20 root       -51  0  0 0 0 S  0.0   0.0   0:00.00 idle_inject/1
  21 root        0  0  0 0 0 S  0.0   0.0   0:00.57 migration/1
  22 root        0  0  0 0 0 S  0.0   0.0   0:03.02 ksoftirqd/1
  24 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 kworker/1-0H-events_highpri
  25 root        0  0  0 0 0 S  0.0   0.0   0:00.03 kdevtmpfs
  26 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 inet_frag_wq
  27 root        0  0  0 0 0 S  0.0   0.0   0:00.11 kauditd
  28 root        0  0  0 0 0 S  0.0   0.0   0:00.01 khungtaskd
  29 root        0  0  0 0 0 S  0.0   0.0   0:00.00 adm_reaper
  30 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 writeback
  31 root        0  0  0 0 0 S  0.0   0.0   0:01.51 kcompactd0
  32 root        0  0  0 0 0 S  0.0   0.0   0:00.00 kmsd
  33 root        0  0  0 0 0 S  0.0   0.0   0:00.00 khugepaged
  80 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 kintegrityd
  81 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 kblockd
  82 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 blkcg_punt_bio
  83 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 tpm_dev_wq
  84 root        0 -20  0 0 0 S  0.0   0.0   0:00.00 ata_sff
```

Отчет:

В рамках запрошенной задачи были проведены следующие операции для оценки состояния процессов операционной системы:

1. Отображение списка процессов системы было выполнено с использованием стандартных инструментов без применения опций для предоставления общего списка процессов, а также с использованием опций для расширенного отображения информации, включая всех пользователей, использование памяти, времени CPU и информацию о процессах, запущенных не из терминала.
2. Для получения выборочной информации обо всех процессах в системе, в формат вывода было включено состояние процесса, его идентификатор (PID), номер терминала и команду, которая запустила процесс. Это позволило оценить текущее состояние и ресурс, привязанный к каждому процессу.

3. Состояния процессов в реальном времени были изучены с использованием инструмента, который показывает динамический обзор процессов системы и их состояние, с анализом использования CPU и памяти.

2. Управление процессами

1. Выведите список процессов пользователя root, используя команду ps. С помощью утилиты для подсчета количества строк посчитайте, сколько процессов принадлежит этому пользователю. Перенаправьте список процессов root в файл procl.

```
ps -u root
```

```
ps -u root | wc -l
```

```
ps -u root > procl
```

```
less procl
```

2. Запустите браузер firefox в фоновом режиме в первом терминале. Во втором терминале запустите утилиту top в фоновом режиме.

```
firefox &
```

```
top &
```

3. В первом терминале найдите первый запущенный вами процесс с помощью команды ps и перенаправьте вывод этой команды на ввод команды grep. Определите ID процесса и номер терминала.

```
ps | grep firefox
```

4. Во втором терминале по идентификатору процесса найдите запущенный вами процесс, связанный с утилитой top.


```
ps | grep {ID_процесса}
```

5. Вернитесь в первый терминал и найдите поддерево для процесса firefox, изучите список составляющих его процессов.

```
pstree
```

6. Измените приоритет для процесса, порожденного программой firefox. Сделайте его равным 10. Подтвердите, что приоритет изменился.

```
ps | grep firefox
```

```
renice -n 10 {номер процесса firefox}
```

```
ps -o s,pid,tty,command,nice | grep firefox | grep 10
```

Сделайте скриншот результата работы и разместите в области ответа

7. Завершите все процессы вашего пользователя с помощью команды *killall*.

```
killall -u {имя_текущего_пользователя}
```

student@prac-work-question:~# Предоставьте ответ в виде скриншота (-ов) .

Если необходимо предоставить скрипт, то ответ может содержать ссылку на скрипт решения (только git):

Скриншот для шага №6

```
kalinin@astra:~$ firefox &
[1] 2093
kalinin@astra:~$ (Parent 2093, Main Thread) WARNING: Theme parsing error: gtk.css:3916:13: Not using units is deprecated. Assuming 'px': 'glib warning', file /builds/AstraOS/buildsystem/astra-build-fixes/ci_firefox/firefox-120.0.1-build1/toolkit/xre/nsSigHandlers.cpp:187
((firefox:2093): Gtk-WARNING **: 22:44:30.658: Theme parsing error: gtk.css:3916:13: Not using units is deprecated. Assuming 'px'.
(Parent 2093, Main Thread) WARNING: Theme parsing error: gtk.css:3916:13: Not using units is deprecated. Assuming 'px': 'glib warning', file /builds/AstraOS/buildsystem/astra-build-fixes/ci_firefox/firefox-120.0.1-build1/toolkit/xre/nsSigHandlers.cpp:187
((firefox:2093): Gtk-WARNING **: 22:44:30.735: Theme parsing error: gtk.css:3916:13: Not using units is deprecated. Assuming 'px'.
(Parent 2093, Main Thread) WARNING: Theme parsing error: gtk.css:3916:13: Not using units is deprecated. Assuming 'px': 'glib warning', file /builds/AstraOS/buildsystem/astra-build-fixes/ci_firefox/firefox-120.0.1-build1/toolkit/xre/nsSigHandlers.cpp:187
((firefox:2093): Gtk-WARNING **: 22:44:30.765: Theme parsing error: gtk.css:3916:13: Not using units is deprecated. Assuming 'px'.
(Parent 2093, Main Thread) WARNING: Theme parsing error: gtk.css:3916:13: Not using units is deprecated. Assuming 'px': 'glib warning', file /builds/AstraOS/buildsystem/astra-build-fixes/ci_firefox/firefox-120.0.1-build1/toolkit/xre/nsSigHandlers.cpp:187
((firefox:2093): Gtk-WARNING **: 22:44:30.785: Theme parsing error: gtk.css:3916:13: Not using units is deprecated. Assuming 'px'.
kalinin@astra:~$ libEGL warning: OR12: failed to authenticate
ATTENTION: default value of option mesa_gthread overridden by environment
ps | grep firefox(Parent 2093, Main Thread) WARNING: Failed to call GetIdleTime(): GDBus.Error:org.freedesktop.DBus.Error.ServiceUnknown: The name org.gnome.Mutter.IdleMonitor was not provided by any .service files
: 'glib warning', file /builds/AstraOS/buildsystem/astra-build-fixes/ci_firefox/firefox-120.0.1-build1/toolkit/xre/nsSigHandlers.cpp:187
** (firefox:2093): WARNING **: 22:44:37.392: Failed to call GetIdleTime(): GDBus.Error:org.freedesktop.DBus.Error.ServiceUnknown: The name org.gnome.Mutter.IdleMonitor was not provided by any .service files

2093 pts/0    00:00:03 firefox
kalinin@astra:~$ ps | grep firefox
2093 pts/0    00:00:03 firefox
kalinin@astra:~$ renice -n 10 2093
2093 (process ID) old priority 0, new priority 10
kalinin@astra:~$ ps -o s,pid,ttty,command,nice | grep firefox | grep 10
S    2093 pts/0    /usr/lib/firefox/firefox      10
kalinin@astra:~$
```

Отчет:

1. Проведено отображение списка процессов пользователя root. Число запущенных процессов было определено с использованием утилиты для подсчета строк. Полученный список процессов был сохранен в файл с именем "proc1" для последующего анализа. Использование команды просмотра содержимого файла позволило проверить правильность сохраненных данных.
2. В первом терминале запущен браузер в фоновом режиме. Во втором терминале запущено утилита для динамической отображения процессов также в фоновом режиме.
3. После запуска браузера был выполнен поиск первого запущенного процесса, что позволило получить его идентификатор и номер терминала, что критически важно для управления процессами.

4. Используя данные идентификатор процесса в первом терминале, был выполнен поиск соответствующего процесса утилиты во втором терминале. Это был ключевой шаг для валидации исполняемых операций.
5. Была изучена структура поддерева процессов для приложения браузера, что обеспечило ценную информацию о всех связанных процессах.
6. Изменение приоритета выполнения процесса, ассоциированного с браузером, было проведено успешно, чему был дан подтверждающий отчет с указанием изменившегося приоритетного уровня.
7. Все процессы текущего пользователя были корректно завершены с использованием команды, отправляющей сигналы завершения соответствующим процессам, осуществив очистку рабочей среды.

Отчет

Прохождение ознакомительной практики направлено на получение базовых навыков в области администрирования операционных систем, в частности Astra Linux. Практика включает изучение командной строки, работы с файловой системой, а также основ управления процессами и учетными записями.

В рамках профессиональной практики были усвоены следующие компетенции:

1. Владение техниками создания и настройки виртуальных машин, что включало установку операционных систем и их конфигурацию согласно потребностям пользователя.

2. Освоение базового набора команд командной строки с целью эффективной навигации по файловой системе, а также управления файлами и каталогами.

3. Приобретение практического опыта использования справочных систем `man` и `info`, позволяющих получить подробную информацию о работе команд и их параметрах.

4. Оттачивание навыков работы с файловой системой через команды `cp`, `mv`, `rm` и другие, позволяющие осуществлять копирование, перемещение и удаление файлов.

5. Углубление знаний в области настройки прав доступа к файлам с использованием команд `chmod` и `chown`, а также познакомились с расширенными командами для управления атрибутами файлов.

6. Изучение и практика работы в текстовом редакторе `vim`, в том числе овладение его модами, базовыми функциями редактирования и способами сохранения документов.

8. Развитие умений в области управления учетными записями: поиск информации в локальной базе данных, создание, модификация и удаление пользовательских аккаунтов, а также управление группами пользователей.

9. Освоение концепции процессов в операционной системе и приобретение навыков мониторинга и управления процессами при помощи команд `ps`, `top`, `kill` и других, а также выполнение заданий, связанных с управлением приоритетами процессов и их корректным завершением.

По завершении ознакомительной практики были успешно освоены базовые концепции администрирования операционных систем семейства GNU/Linux на примере Astra Linux. Установлены и закреплены начальные навыки работы с виртуальными машинами, командной строкой, файловой системой и управлением пользователями и процессами. Полученные знания и навыки лягут в фундамент будущей профессиональной деятельности в сфере ИТ.