

Задание 1: Просмотреть содержимое директорий /etc, /proc, /home. Посмотреть пару произвольных файлов в /etc.

Просмотр содержимого директорий возможно несколькими способами. При запуске соединения по SSH изначально приглашение оболочки находится в разделе /home/meshd . Для перехода в директорию /home воспользуюсь вводом

```
cd ..
```

Таким образом перемещаюсь на директорию вверх в директорию /home.
Для перехода в директории /proc и /etc команды используются соответственно

```
cd /proc  
cd /etc
```

Для просмотра файлов в директории /etc использовал команды cat и vi

```
vi timezone  
cat subgid
```

Задание 2: Выяснить, для чего предназначена команда cat. Используя данную команду, создать два файла с данными, а затем объединить их. Просмотреть содержимое созданного файла. Переименовать файл, дав ему новое имя.

Для просмотра функций cat можно воспользоваться командой

```
man cat
```

В выводе указывается объединение, создание, и стандартный вывод в терминал.
Для создания файлов используются команды

```
cat > test1
```

После чего в терминале вводятся данные в файл test 1
Ввод данных прерывается командой Ctrl+D

```
cat > test2
```

После чего в терминале вводятся данные в файл test 1
Ввод данных прерывается командой Ctrl+D

Для объединения файлов используется команда

```
cat test2 >> test1
```

В результате данные файла test2 записываются в конец файла test1

Для переименования воспользуюсь командой

```
mv test1 > testing
```

В результате файл test1 "перезаписывается" в файл testing.

Задание 3: Создать несколько файлов. Создать директорию, переместить файл туда. Удалить все созданные в этом и предыдущем задании директории и файлы.

Для создание директории используется команда

mkdir test

Создается директория test в /home/meshd

Так же для создания вложенной директории с помощью абсолютного пути воспользуюсь командой

mkdir -p /home/meshd/test/1/2/3/4

Что позволяет создать вложенные директории с флагом -p

Для удаления созданных файлов и директорий использовал

rm -r testing

rm -r test2

Для удаления директории "4" подходит команда при условии пустой директории

rmdir 4

Для удаления вложенных директорий используется команда

rm -r test/

В данном случае удаляются и вложенные директории и файлы.

Задание 4: В ОС Linux скрытыми файлами считаются те, имена которых начинаются с символа ".". Сколько скрытых файлов в вашем домашнем каталоге? (Использовать конвейер. Подсказка: для подсчета количества строк можно использовать wc.)

Для подсчёта количества скрытых файлов в домашней директории использовал команду

ls -Al | cut -d ' ' -f 1 | grep rw- | wc -l

Где в конвейер были запущены команды: cut, grep, wc. Необходимо было для разбивки вывода команды "ls" по пробелам, выбрав первое поле, далее из этого поля командой "grep" были отсеены скрытые директории, далее командой wc было посчитано количество линий на выводе конвейера.

ls -lad .* | grep ^- | wc -l

ls -AF | grep '^\. ' | grep -v '/' | wc -l

find . -maxdepth 1 -type f -iname ".*" -ls | wc -l

Задание 5: Попробовать вывести с помощью cat содержимое всех файлов в директории /etc. Направить ошибки в отдельный файл в вашей домашней директории. Сколько файлов, которые не удалось посмотреть, оказалось в списке?

Для выполнения данной задачи воспользуюсь данной командой:

```
cat /etc/* > test1 2>> test2
```

В созданный файл test1 будет записана информация из файлов в директории /etc. В файл test2 будут записаны ошибки, по которым данные не смогли быть записаны в test1. На деле получается, что в файл test2 будут записаны ошибки, что определенный файл является директорией.

```
cat test2 | wc -l
```

Данная команда выведет количество ошибок(строк) записанных в файл test2 при создании файла test1. (ручной пересчёт совпал).

*Примечание: я больше чем уверен, что данное задание мною выполнено не верно. Вписывать значения полученные от других участников курса - считаю не честным.

Задание 6: Запустить в одном терминале программу, а в другом терминале посмотреть PID процесса и остановить с помощью kill, посылая разные типы сигналов. Что происходит?

Для опытов был запущен FireFox. Через команду

```
top
```

отслеживая процессы в реальном времени нахожу PID процесса FireFox командами (два раза запуск-kill)

```
kill -9 PID
```

```
kill -15 PID
```

Разницы в вариантах завершения для FireFox нет.

Подобный вариант на других приложениях приводит к одинаковому результату.

При использовании команды

```
kill -15 PID
```

```
kill -9 PID
```

в терминале через подключение по SSH кроме отображения Terminated в консоле сервера - других действий нет.