Молдавский Государственный Университет Факультет Математики и Информатики Департамент Информатики

Работа по дисциплине

“ SMAI”

Тема работы N1: “ Основы Linux.”

Выполнил: студент группы I2302 Ciobanu Stanislav

Преподаватель: Dumitru Borș

Кишинев 2025

Раздел 1: Структура файловой системы

Файловая система Linux имеет иерархическую структуру, где корневой каталог / является самой высшей точкой. В отличии от файловой системы Windows, где каждый примонтированный том или диск фактически является самостоятельной структурой (Хотя там тоже можно делать сложные маунты), в Linux вся файловая система представляется единым целым, а примонтированные тома представляются, как директории.

Основные каталоги:

/bin – базовые бинарники

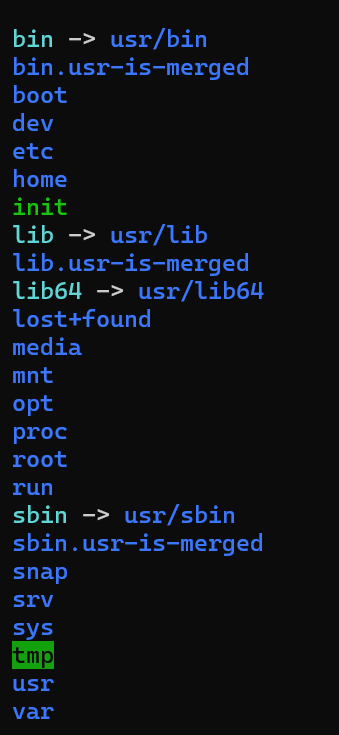
/etc - конфигурации

/home – домашние директории юзеров

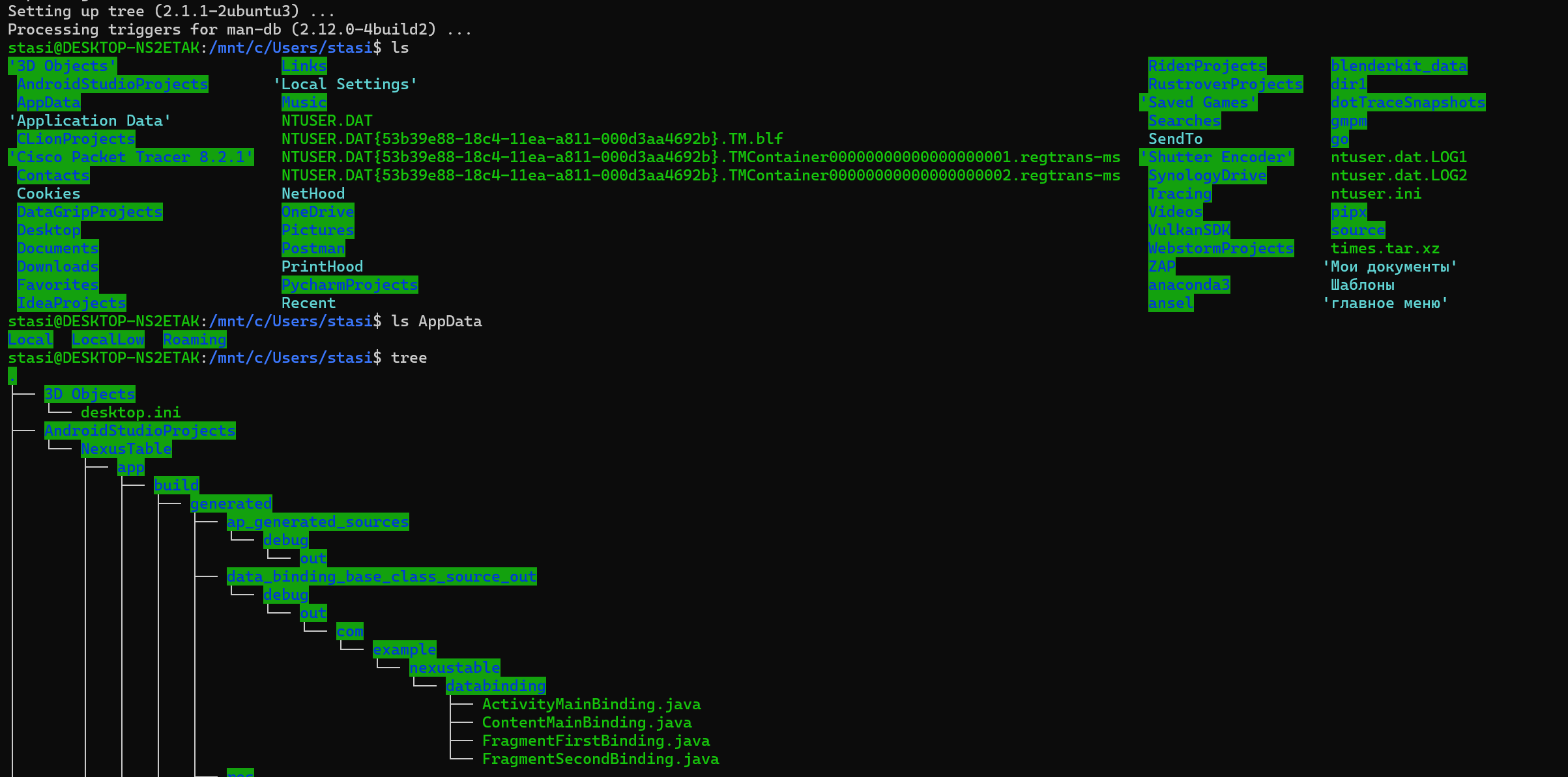
/var – временные данные

/usr – программы пользователей

Используя команду ls выводим на экран структуру файловой системы.



Команда ls /директория позволяет вывести содержимое конкретной директории, а команда tree целиком древовидную структуру со всеми вложениями



Раздел 2: Базовые команды Linux

Используем команды mkdir, touch, mv, rm и rmdir для создания, перемещения, копирования и удаления файлов и директорий.

mkdir – создание директории

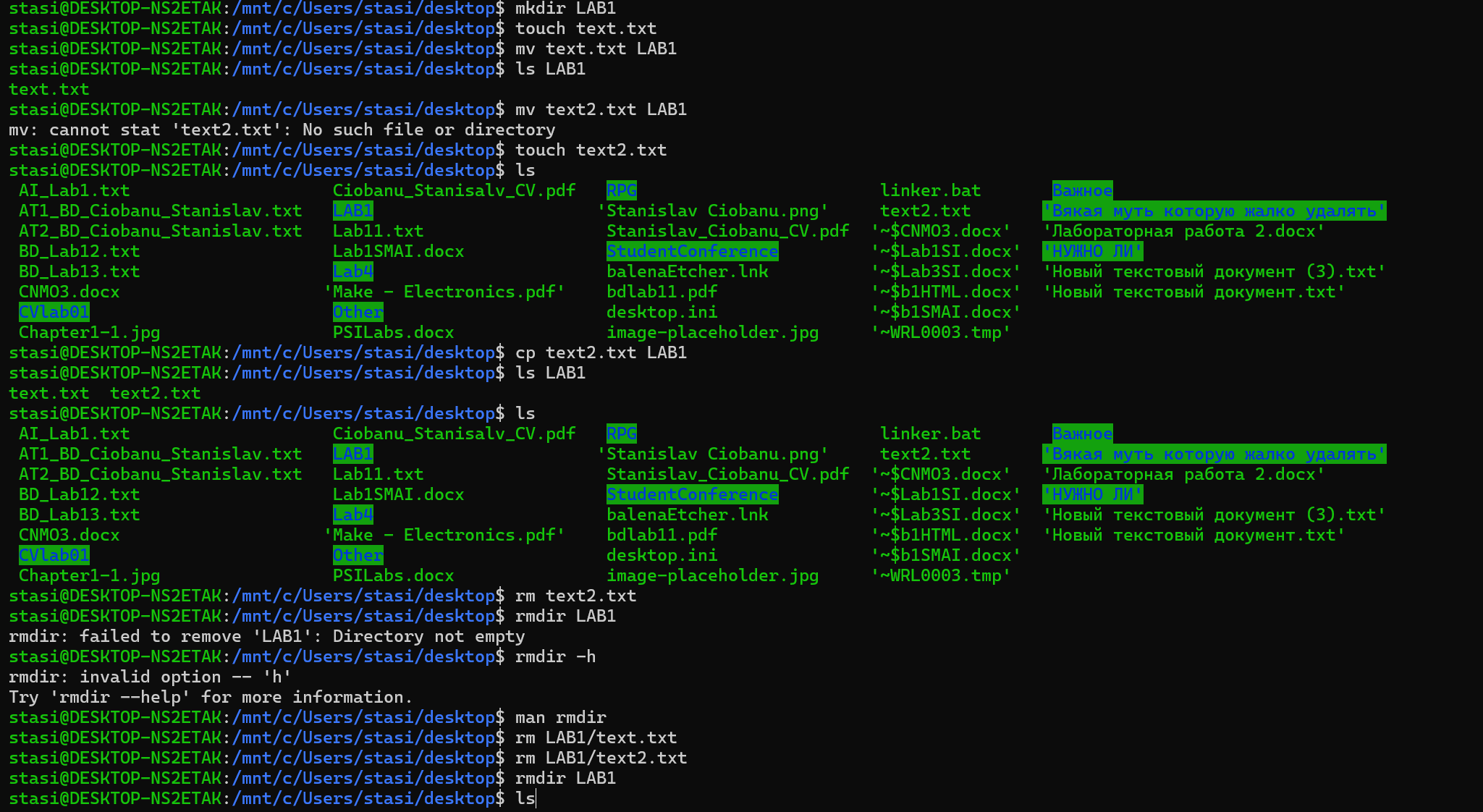
rmdir – удаление директории

touch – создание файла

mv – перемещение файла

rm – удаление файла

cp – копирование файла или директории



Раздел 3: Управление пользователями и правами

В Linux каждый файл и процесс имеет владельца и группу. Доступ определяется правами:

r (read) – чтение,

w (write) – запись,

x (execute) – выполнение.

Права задаются для трёх категорий:

u – владелец (user),

g – группа (group),

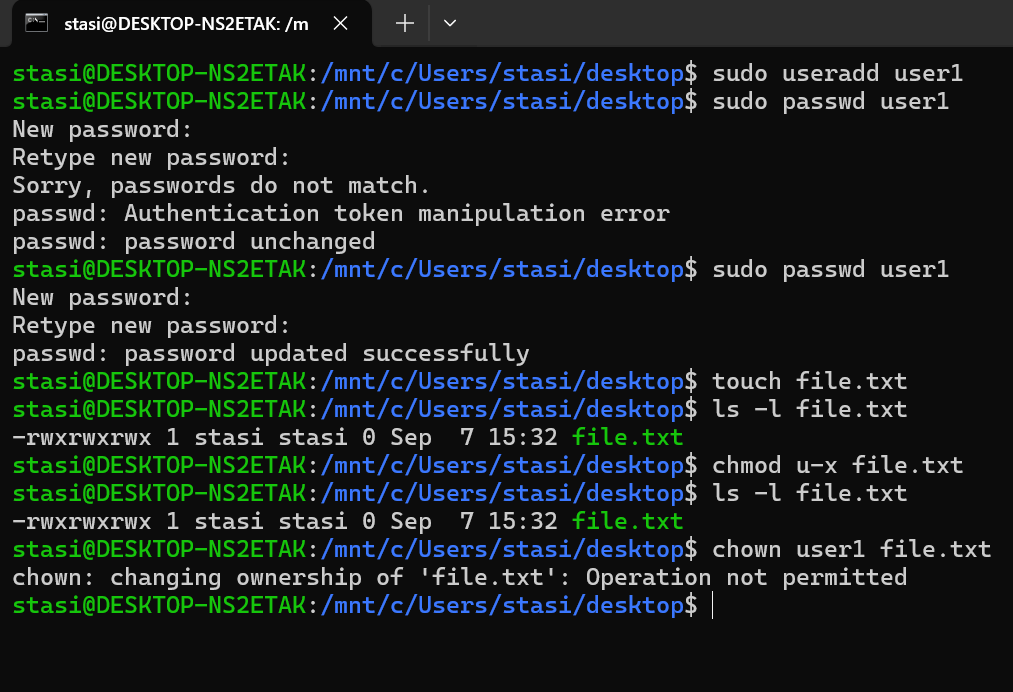
o – остальные пользователи (others).

Для управления правами используются команды:

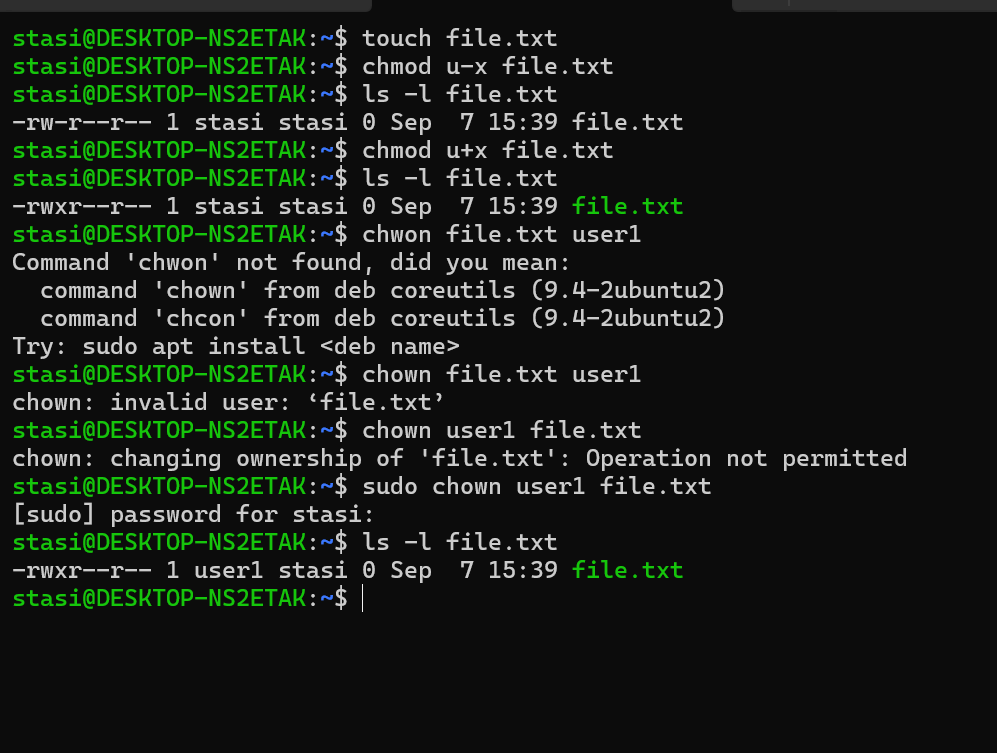
chmod – изменение прав доступа.

chown – смена владельца.

usermod, adduser, deluser – управление пользователями.



Так как я работаю в wsl и нахожусь в примонтированной NTFS директории, тут не действуют классические Linux права доступа. Командой cd перейдём к корневой директории юзера



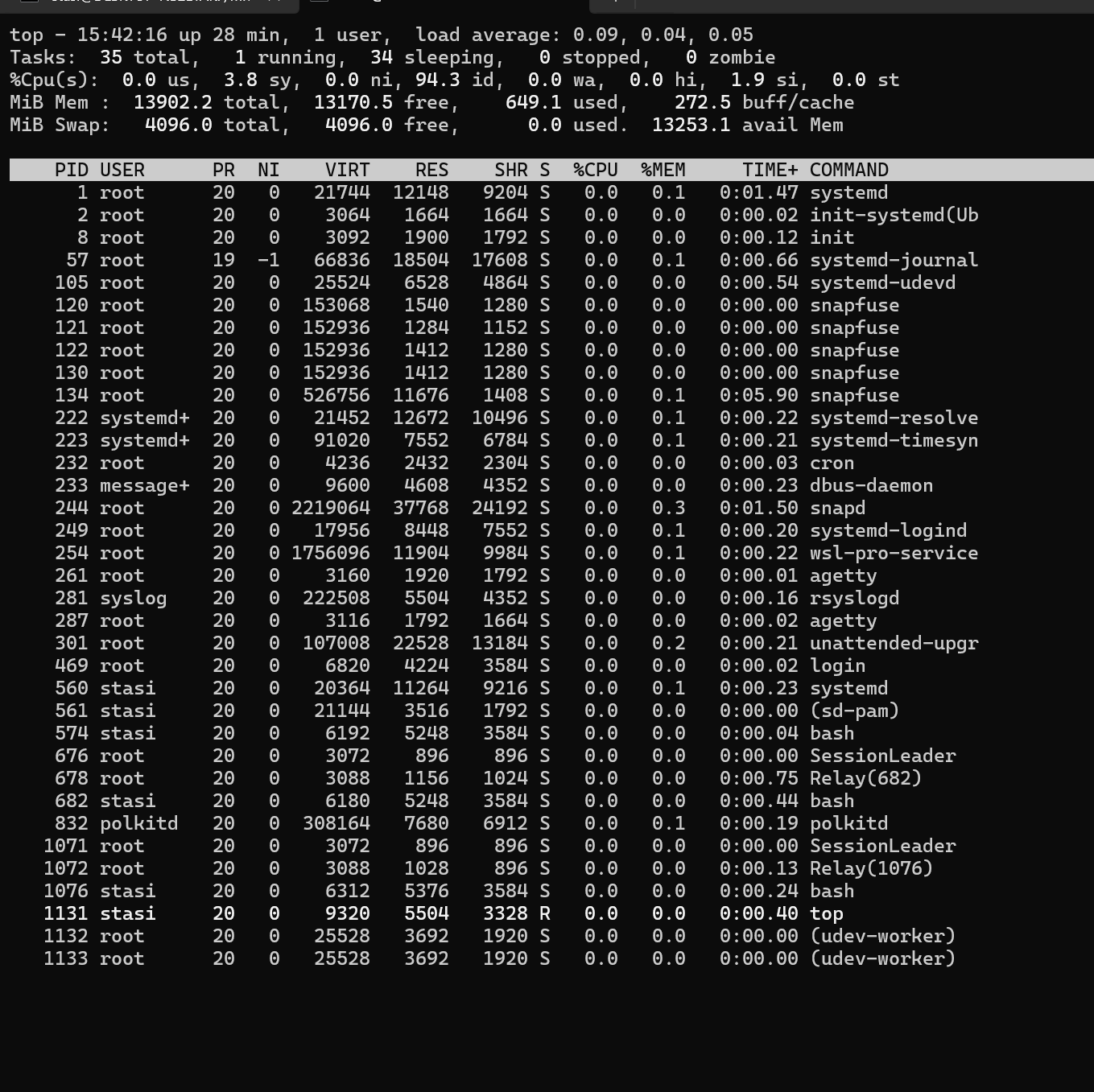
Создадим файл file.txt и сменим права на его исполнение. Для этого применим chmod с параметром u-x (user - execute). Командой ls -l проверим что права изменились.

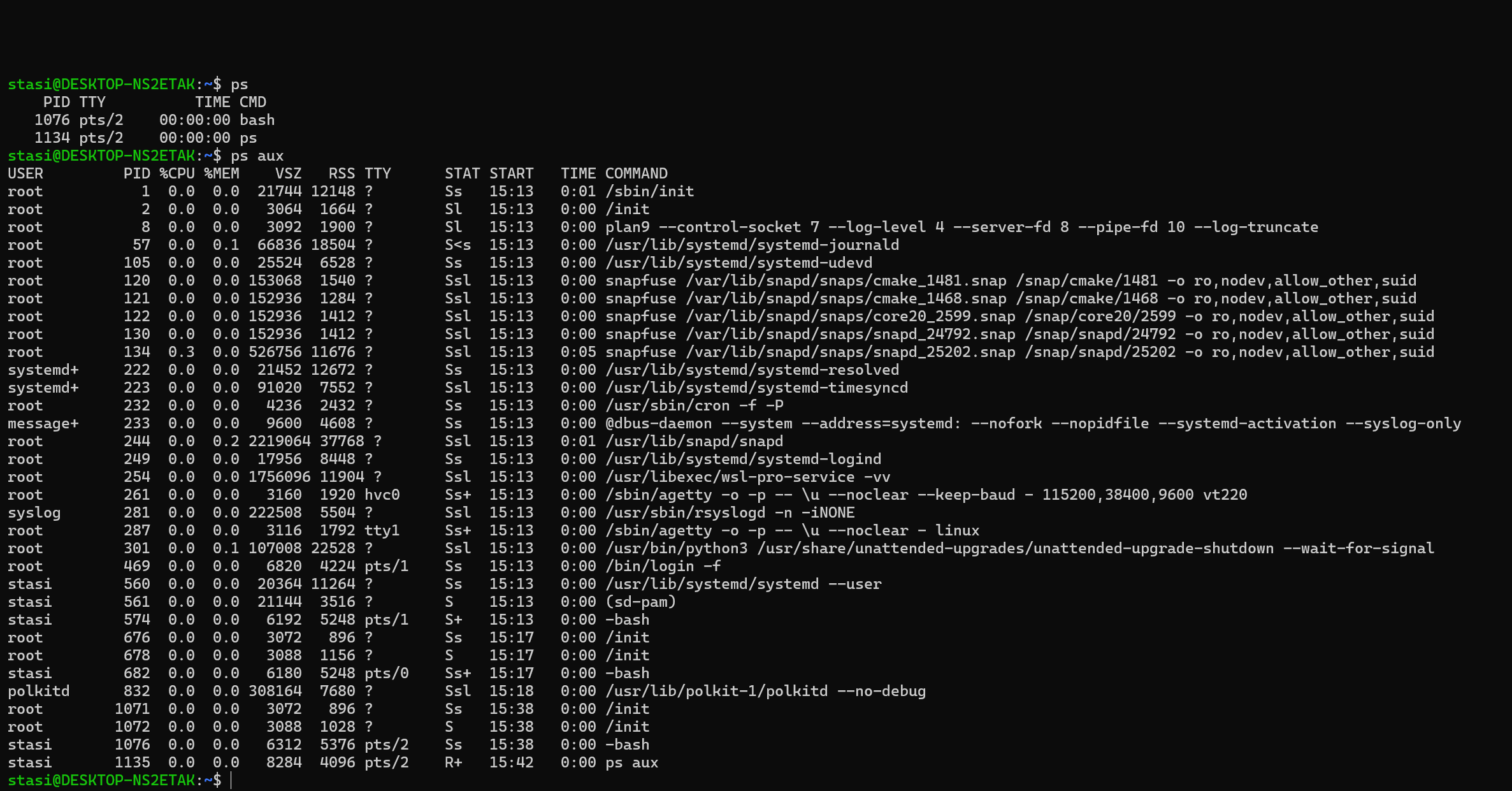
Чтобы вернуть права можно аналогично прописать chmod u+x. Аналогично могут быть изменены права на чтение (r) и запись (w)

Командой chown сменим владельца файла. Тут важно уточнить что делать это нужно под root правами, поэтому применяем sudo и вводим пароль.

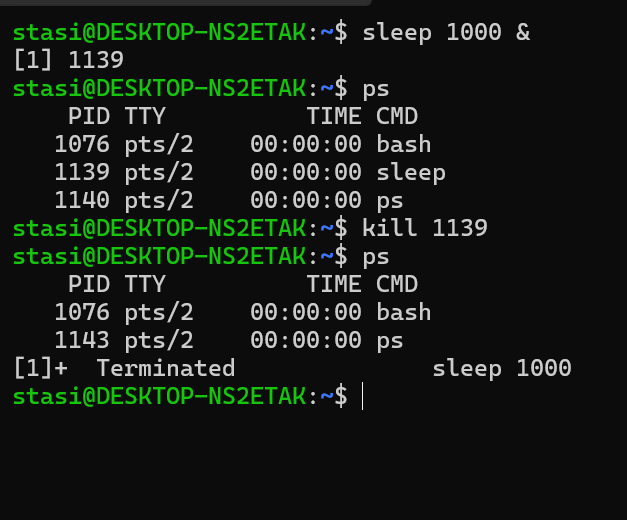
Раздел 4: Процессы в Linux

Управление процессами является одной из важнейших вещей в любой операционной системе. В Linux для того чтобы ознакомиться со списком действующих процессов, вызываем утилиту ps aux (Или утилиту top если нам нужно динамически следить за процессами). Если нам необходимо увидеть список активных процессов, можно ввести просто ps





Создадим процесс sleep. Он отображается в активных процессах. Для его уничтожения используем команду kill. Проверяем что процесс действительно прекратился



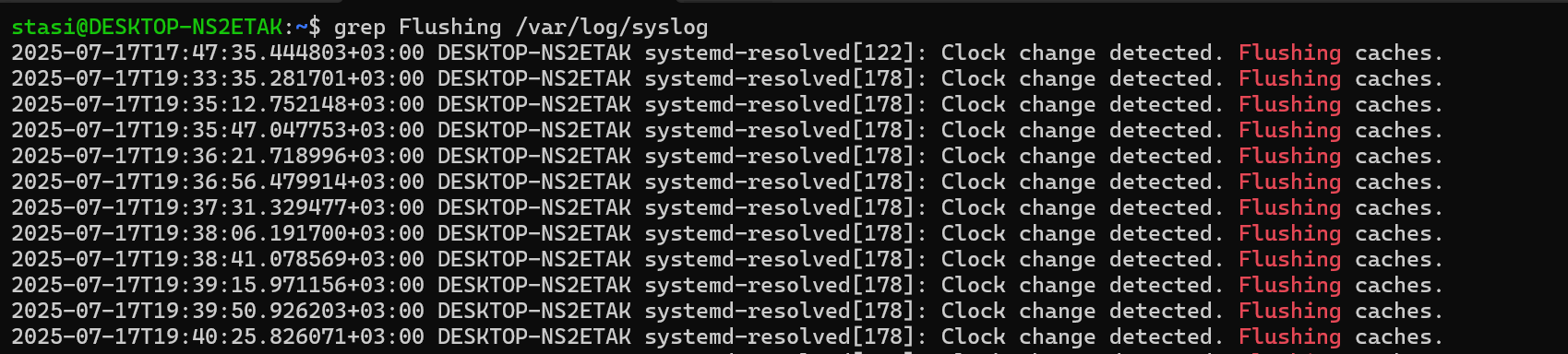
Раздел 5: Обработка текста с помощью grep, sed, awk

grep – достаточно мощная команда для поиска текста

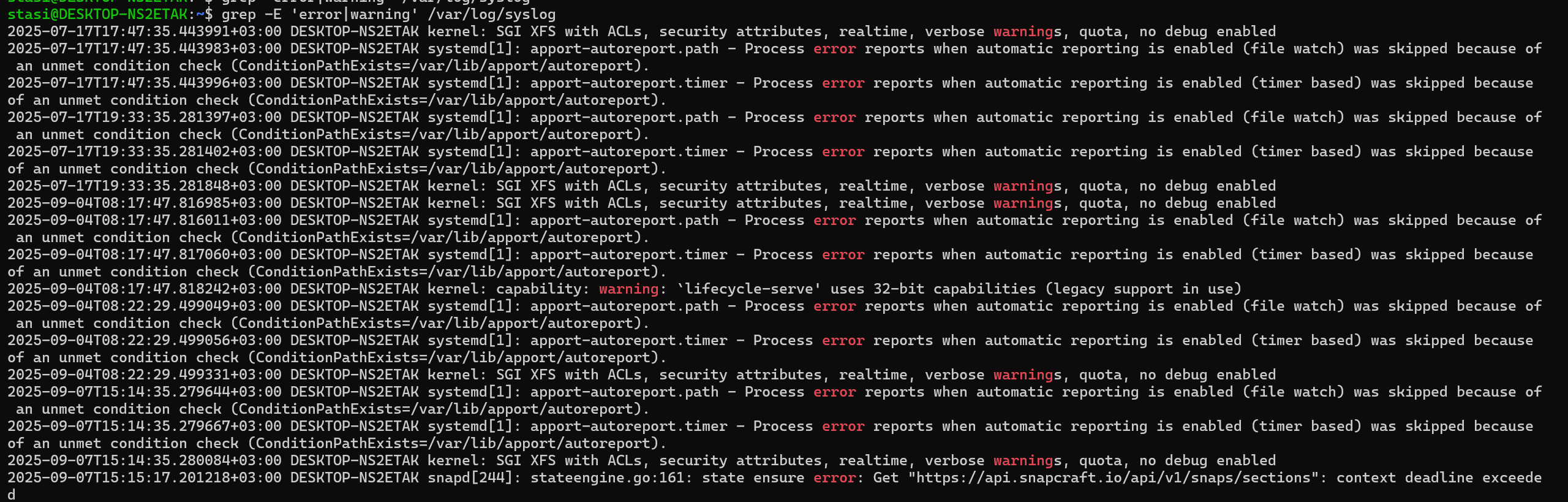
sed – предназначена для работы с потоками текстовых данных

awk – предназначен для анализа текстовых данных

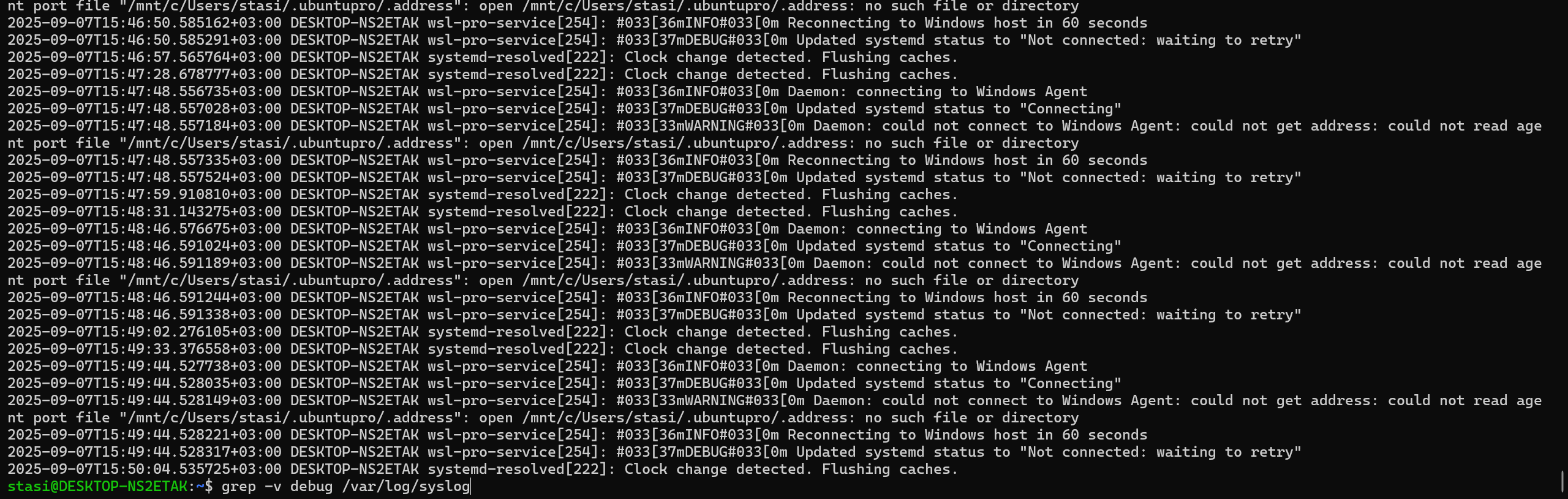
Попробуем применить данные команды. Найдём в файле /var/log/syslog строки в которых есть слов Flushing



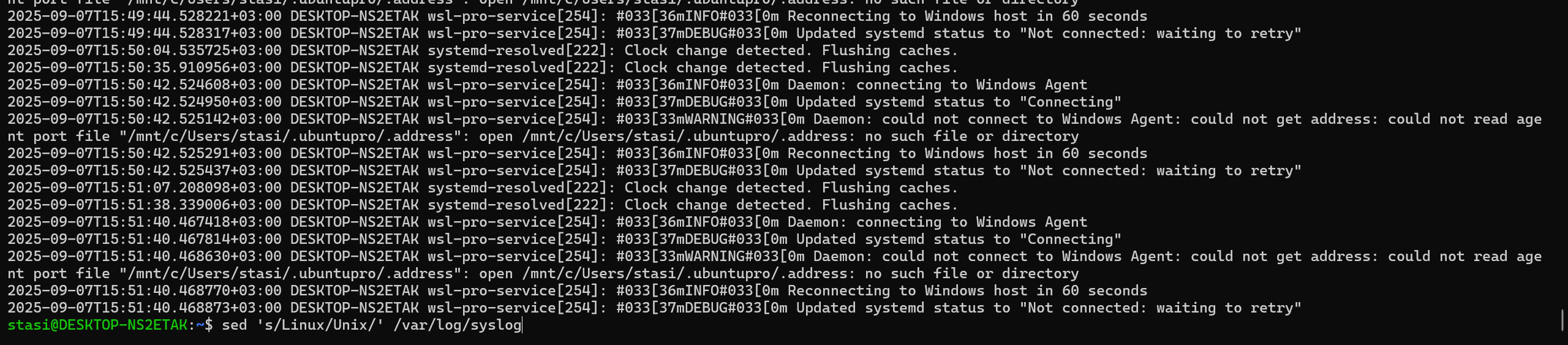
Применим небольшое выражение ‘error|warning’ для того чтобы вывести все строки, в которых встречается error или warning, предварительно применив флаг -E



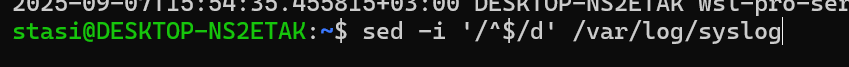
Чтобы найти строки в которых нет слова debug, применим флаг -v



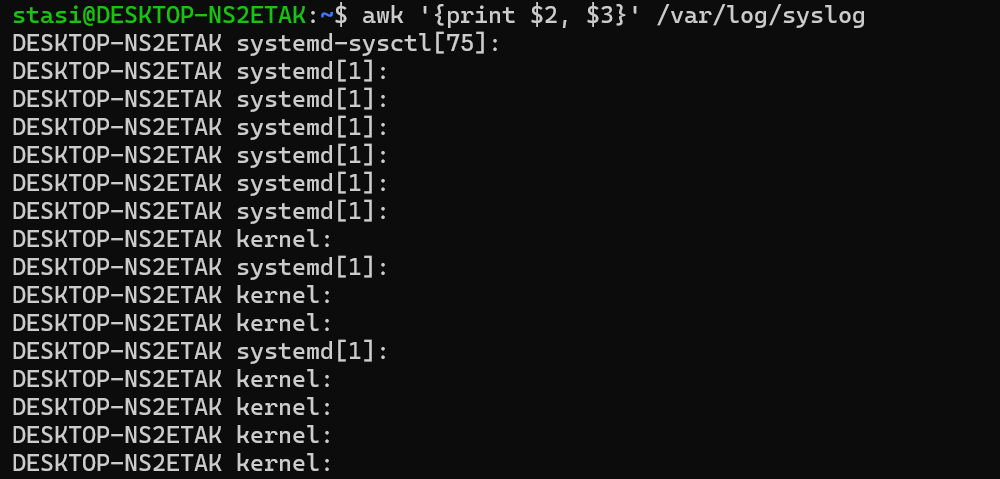
Командой sed ‘s/Linux/Unix’ мы можем заменить все слова Linux на Unix. В данном примере я не стал сохранять файл



При следующем использовании команды sed я применил флаг -i для того чтобы удалить все пустые строки (^$ соответствует всем пустым строкам, d - delete)



Используем awk чтобы вывести 2 и 3 столбцы



А также применяем выражение ‘{$3 > 100}’ для выборки строк, где значение в 3 столбце больше 100



Допустим у нас есть CSV файл такого содержания

id,name,age,city

1,Alex,25,London

2,Maria,30,Berlin

3,John,22,Paris

4,Anna,30,London

Тогда для того чтобы вывести все строки, где возраст больше 30, применим -

awk -F, '$3 == 30 {print}' data.csv

Флаг -F необходим для указания разделителя (, в нашем случае)

Заключение:

Мой опыт операционной системы Linux, как обычного пользователя не был очень успешным. Я ставил archlinux, ubuntu`, mint и fedora, но так и не нашёл в них замены windows. Из-за open source натуры эти системы слабо пригодны для повседневного пользования.

По части devops Linux незаменим. Он отлично подходит для развертывания приложений (В том числе из-за огромного количества дистрибутивов и открытого исходного кода).