Homework\_Lesson 12

**Изображение выглядит как текст, снимок экрана, документ, Шрифт

Автоматически созданное описание**

**Создадим директорию и скрипт с данными из примера**

Изображение выглядит как текст, Шрифт, снимок экрана, Графика

Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

**Запустим скрипт для генерации лог-файла, это создаст файл server.log с содержимым:**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

**Теперь, когда у нас есть лог-файл, мы напишем скрипт, который извлечет нужную информацию.**

**nano analyze\_logs.sh**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание**

* grep "status=200" $LOGFILE: Этот командой мы находим все строки в логах, где статус равен 200 (успешный вход).
* awk -F "ip=" '{print $2}': Используется для извлечения IP-адреса после ip=.
* awk -F " status" '{print $1}': Разделяет строку до статуса, чтобы получить только IP-адрес.
* sort | uniq: Сортирует и удаляет дублирующиеся значения, оставляя только уникальные IP-адреса.

Для обработки неудачных логинов (статус 403), мы используем аналогичный подход:

* grep "status=403" $LOGFILE: Находит строки с ошибкой 403.
* awk -F "user=" '{print $2}': Извлекает имя пользователя после user=.
* awk -F " ip=" '{print $1}': Разделяет строку на имя пользователя, получая только имя.
* sort | uniq: Сортирует и удаляет дубли, оставляя уникальные имена пользователей.

**Запускаем скрипт ./analyze\_logs.sh**

**Результат**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание