

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования
«Гомельский государственный университет
имени Франциска Скорины»

Факультет физики и информационных технологий

Кафедра общей физики

Управление службами

Отчет по лабораторной работе №5

Исполнитель
студент группы КИ-22:

Д.В.Скрежендевский

Проверил
ст. преподаватель:

В.В.Грищенко

Гомель 2025

Цель работы: Изучить основные принципы работы с процессами в операционных системах Windows и Linux.

1. Изучить основы применения программ и утилит, работающих с службами в операционных системах Linux и Windows.

2. Вывести список всех служб, работающих в обеих ОС при помощи соответствующих команд.

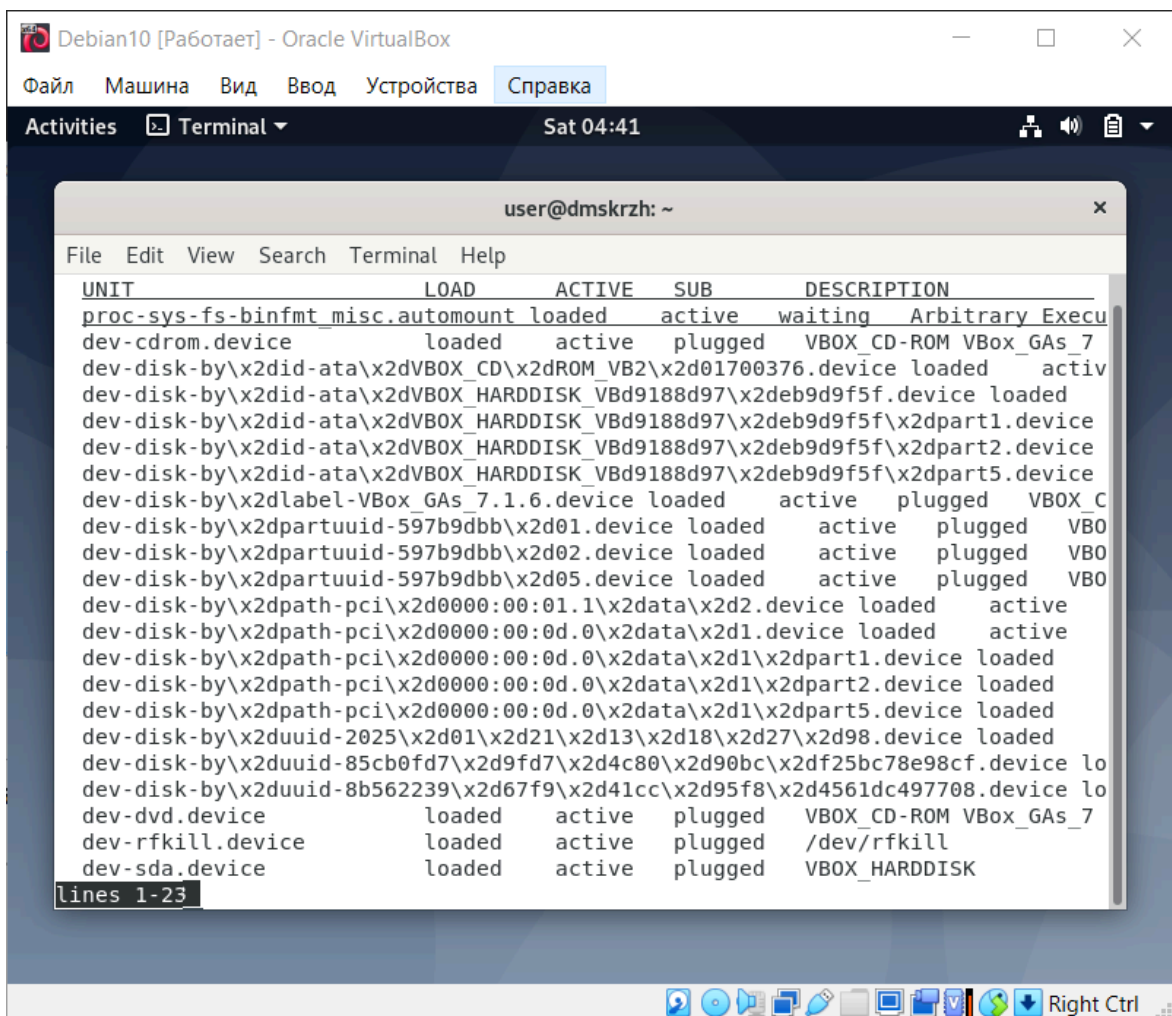


Рисунок 1 - Вывод списка всех служб в ОС Linux

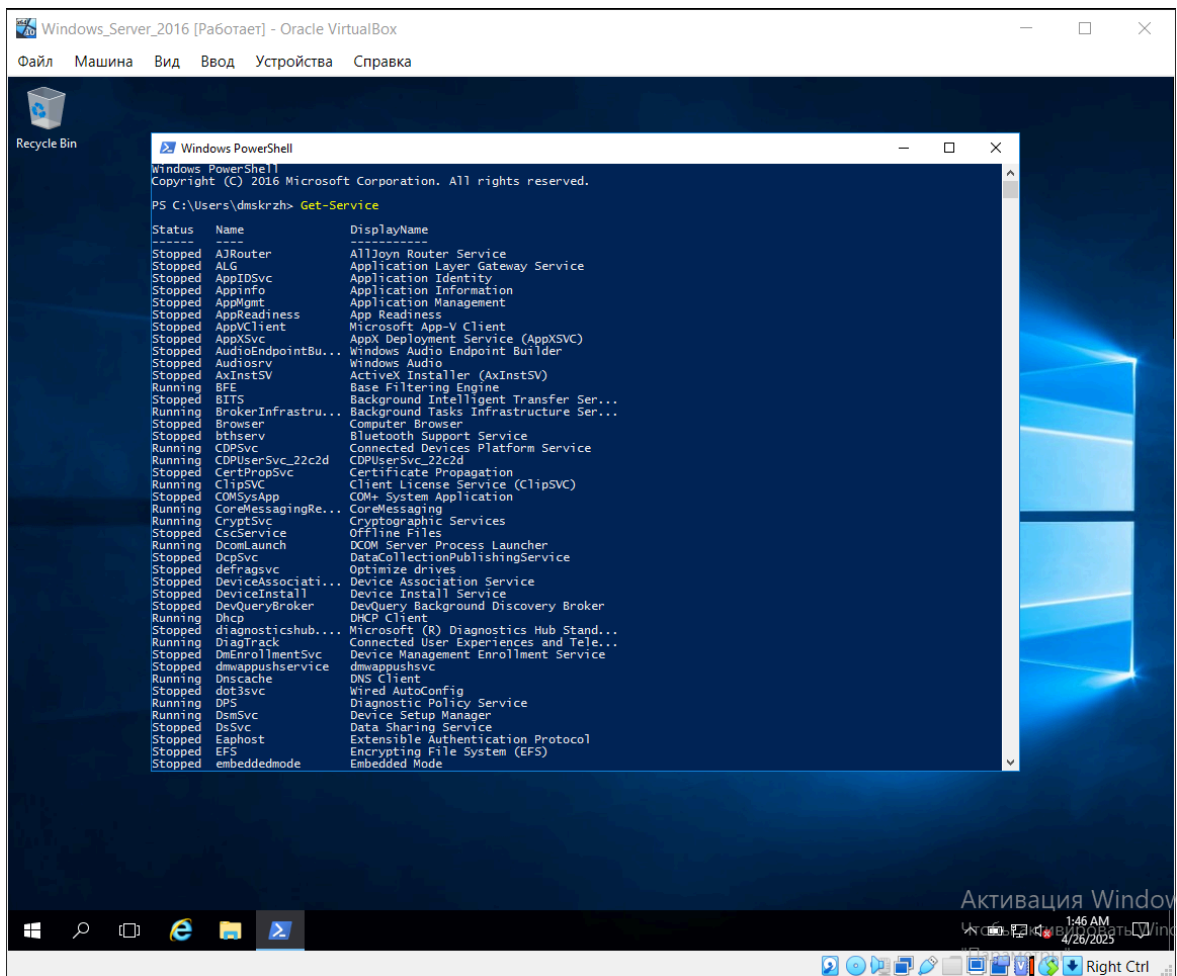


Рисунок 2 - Вывод списка всех служб в ОС Windows

3. Установите браузер Google в обеих ОС.

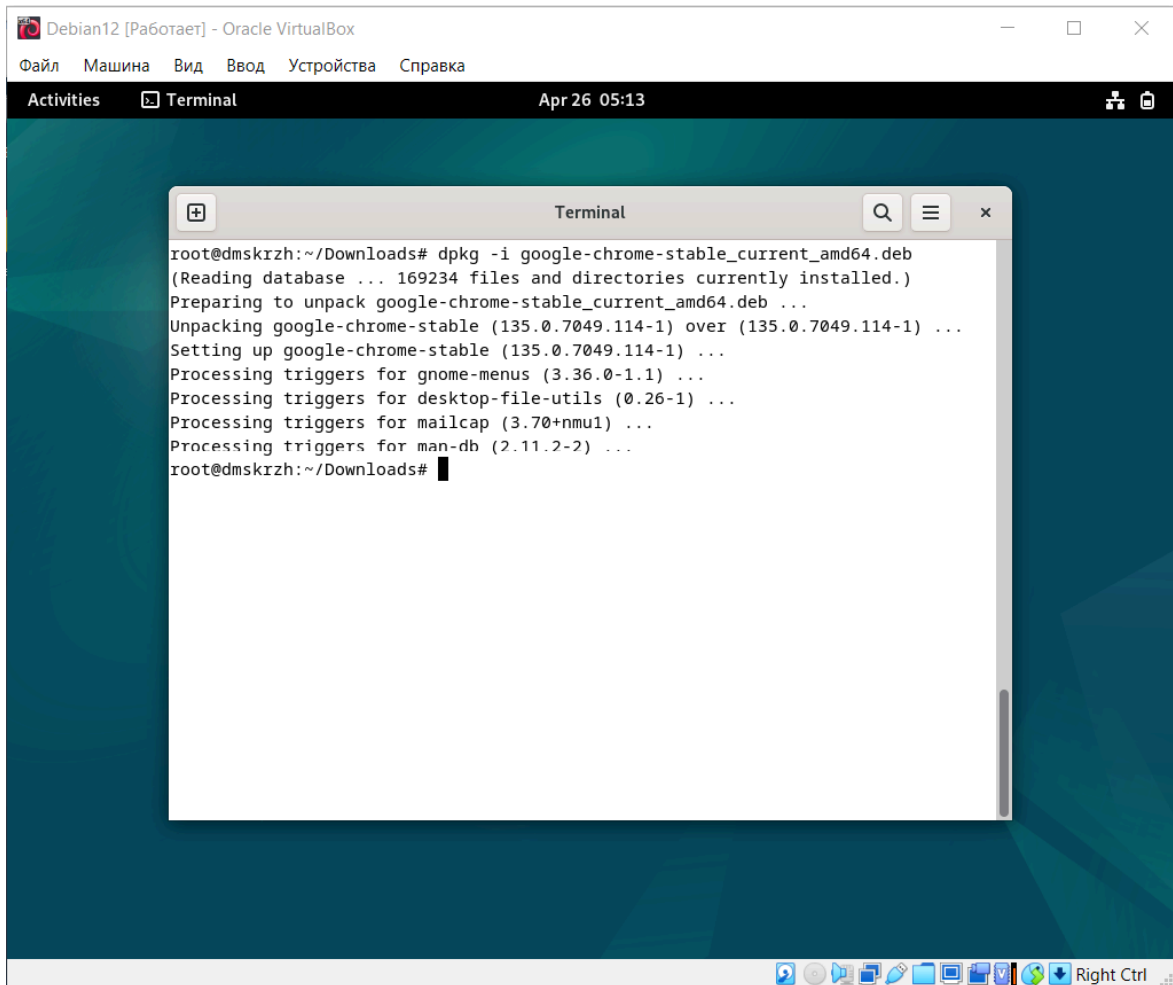


Рисунок 3 - Установка браузера в ОС Linux

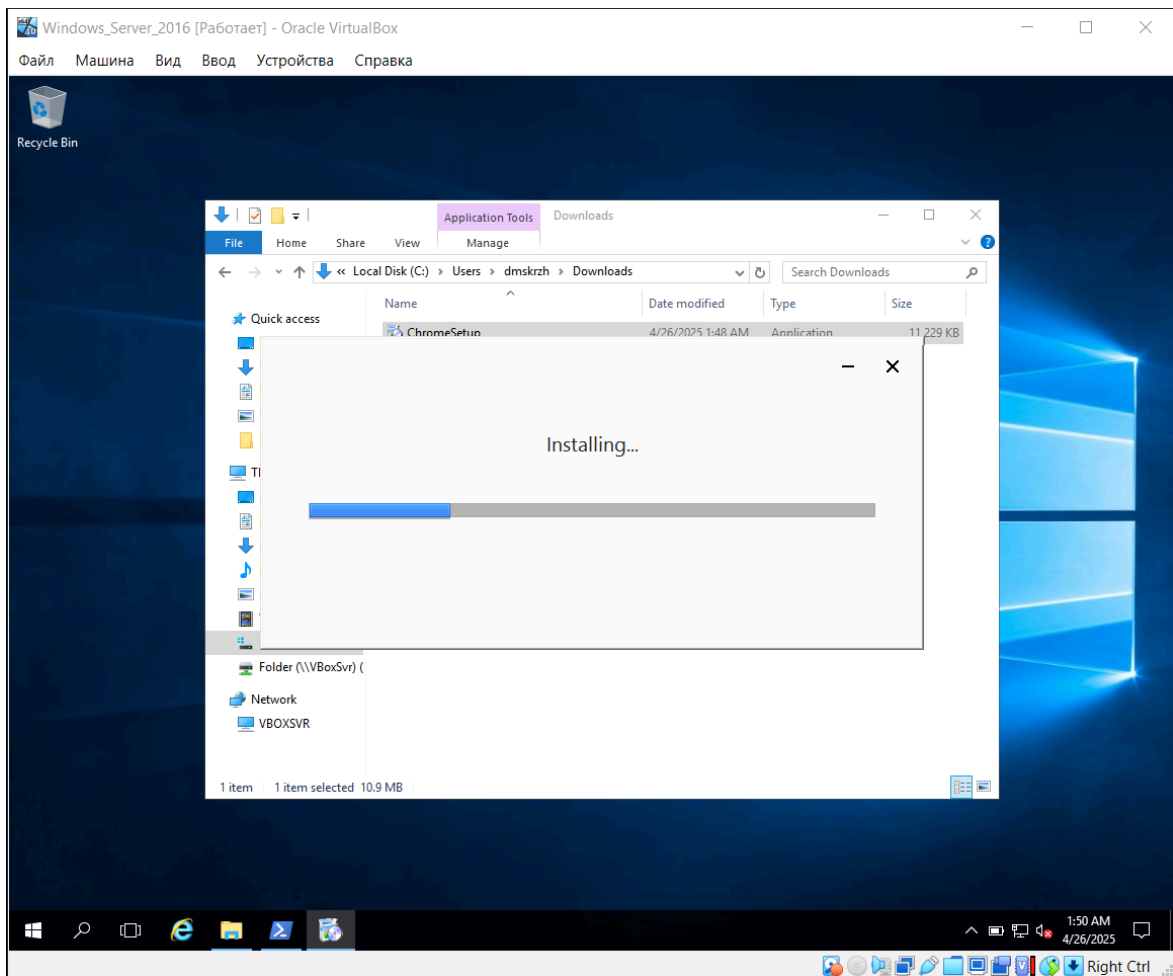


Рисунок 4 - Установка браузера в ОС Windows

4. Обнаружьте службу, отвечающую за автоматическое обновление Google, и выведите информацию о ней

В ОС Linux автоматическое обновление происходит при помощи репозитория.

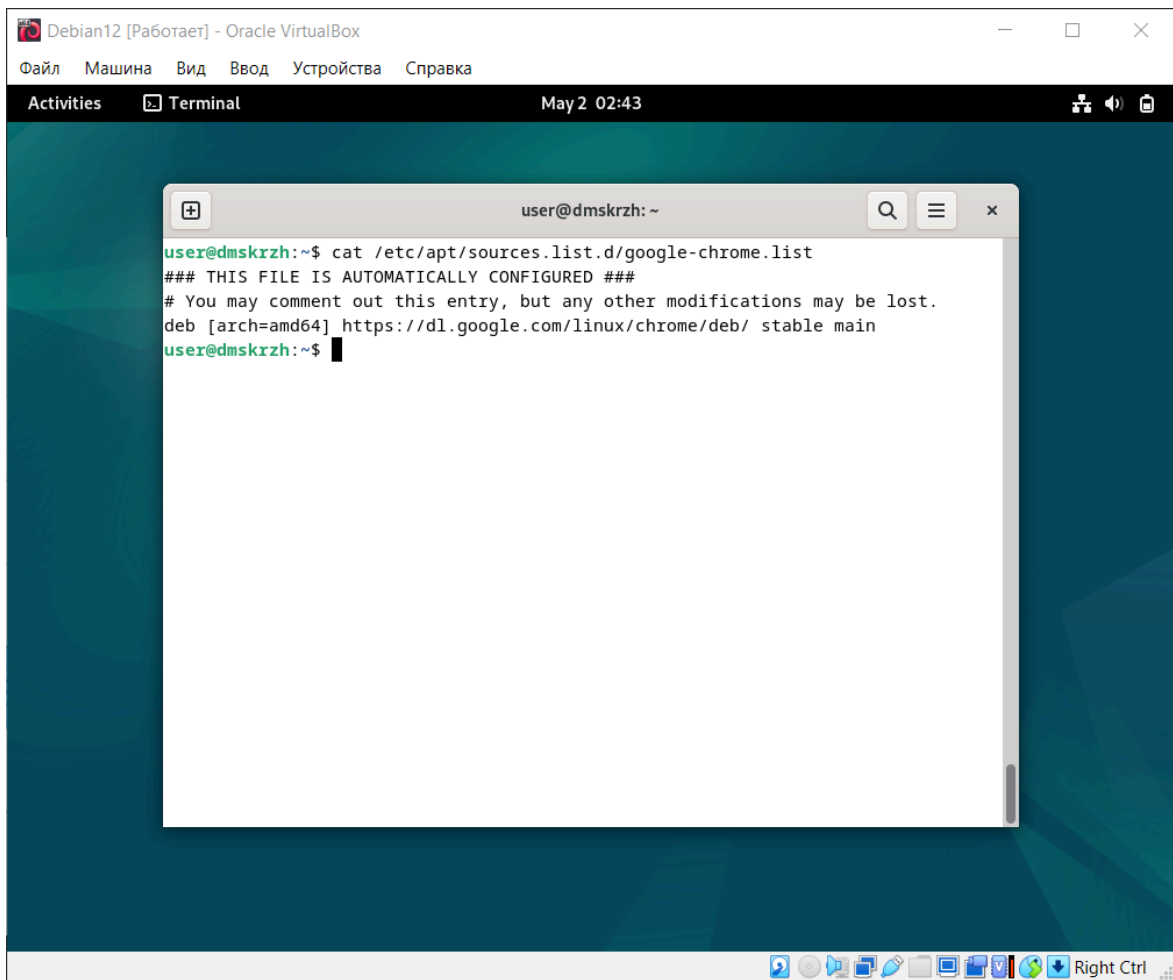


Рисунок 5 - Обнаружение репозитория в ОС Linux

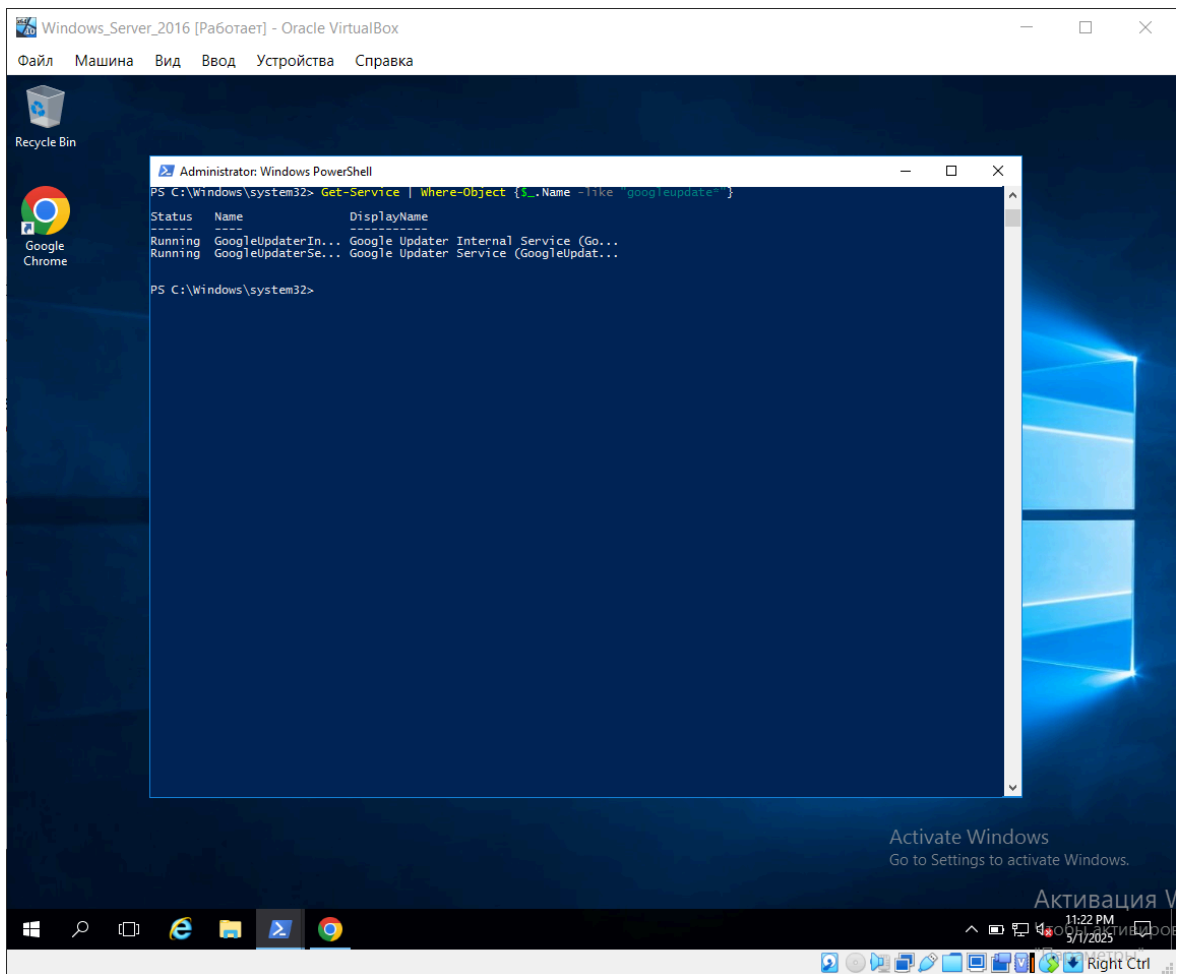


Рисунок 6 - Обнаружение службы в ОС Windows

5. Остановите данную службу в обоих ОС.

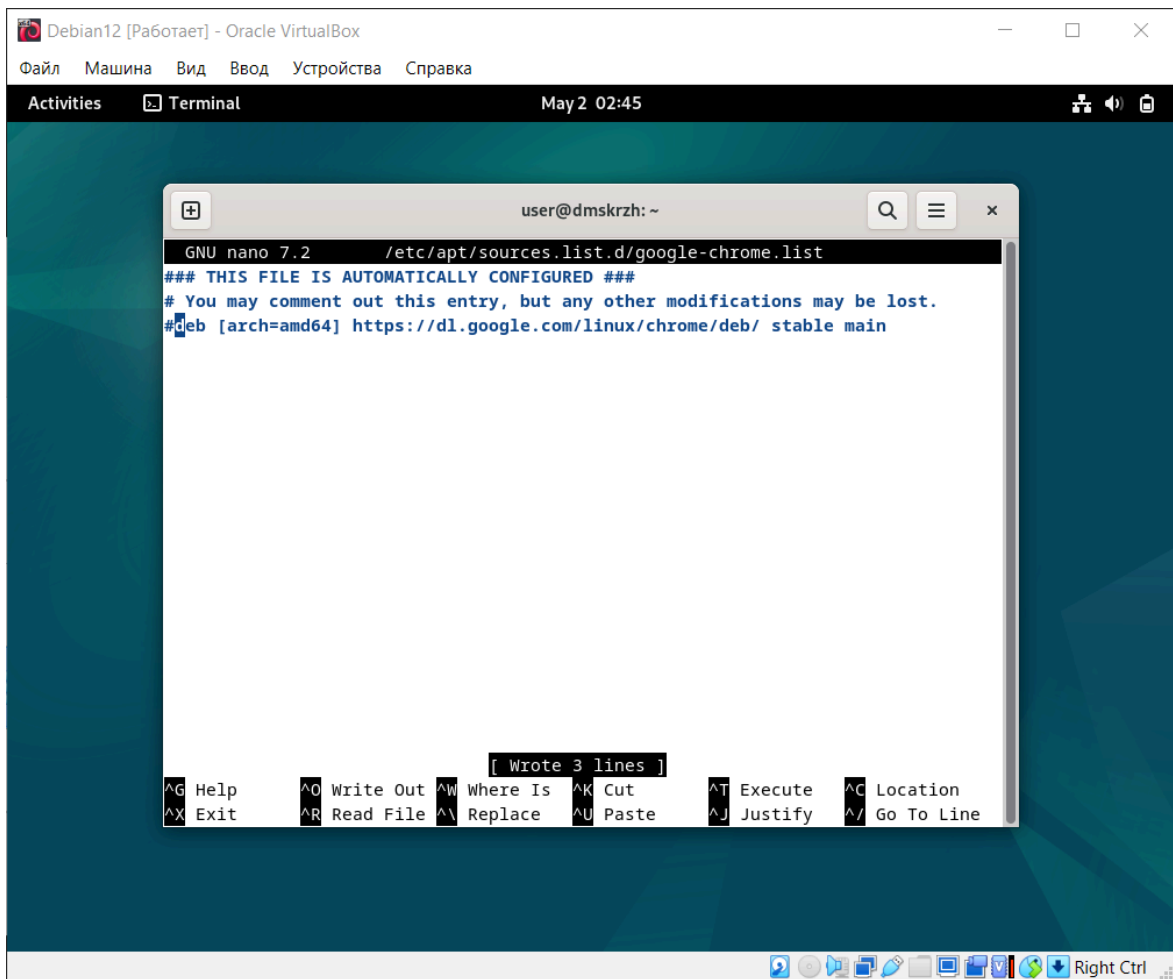


Рисунок 7 - Исключение репозитория из списка в ОС Linux

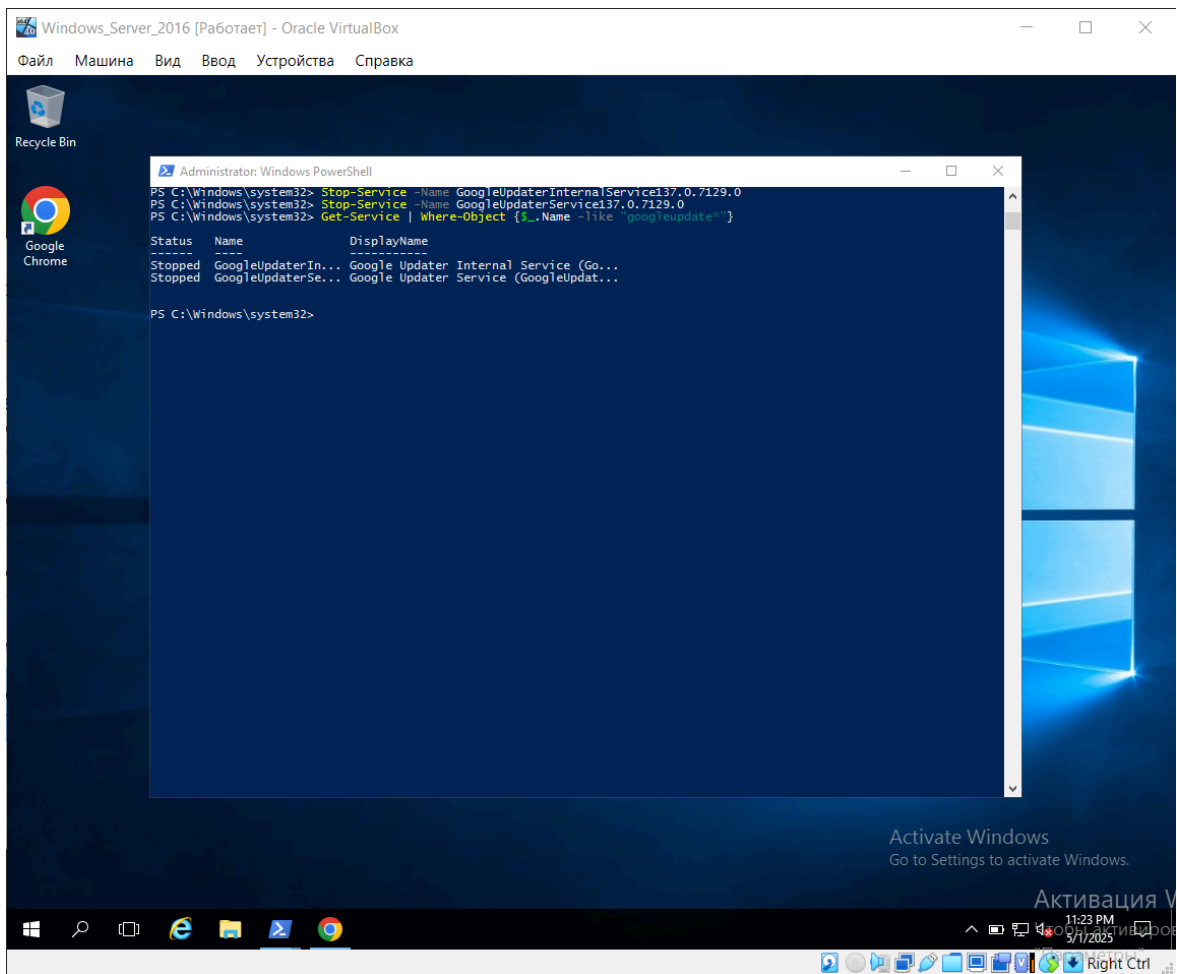
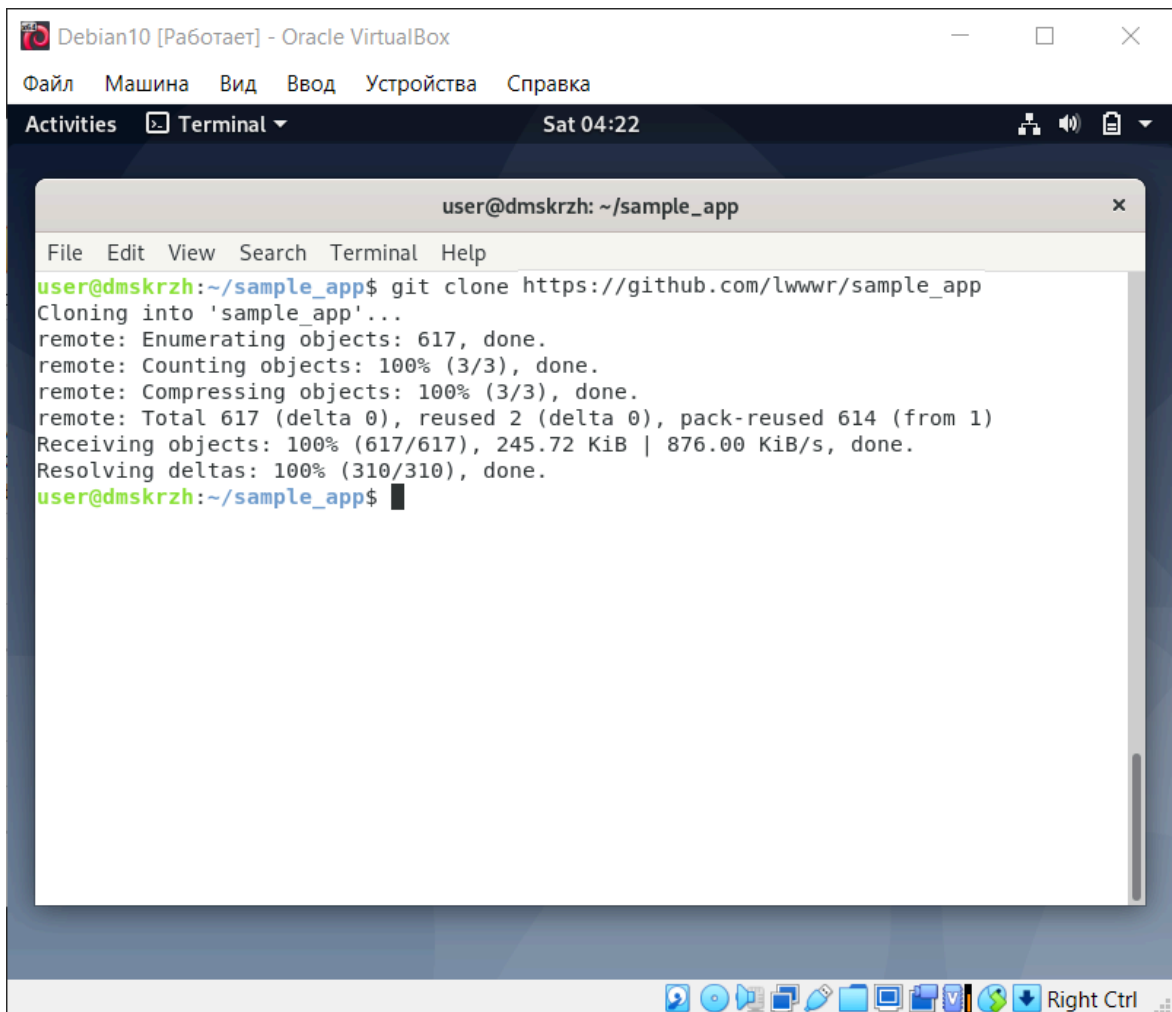


Рисунок 8 - Остановка службы в ОС Windows

6. Для приложения исходный код которого, размещен в репозитории https://github.com/lwwwr/sample_app настройте его запуск как службы.



The screenshot shows a terminal window titled "user@dmskrzh: ~/sample_app" within an Oracle VM VirtualBox environment. The terminal output shows the successful cloning of the repository "sample_app" from GitHub. The command executed is "git clone https://github.com/lwwwr/sample_app". The output includes progress details for enumerating, counting, and compressing objects, as well as the total size of the repository (617 objects, 245.72 KiB) and the download speed (876.00 KiB/s).

```
user@dmskrzh: ~/sample_app
File Edit View Search Terminal Help
user@dmskrzh:~/sample_app$ git clone https://github.com/lwwwr/sample_app
Cloning into 'sample_app'...
remote: Enumerating objects: 617, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 617 (delta 0), reused 2 (delta 0), pack-reused 614 (from 1)
Receiving objects: 100% (617/617), 245.72 KiB | 876.00 KiB/s, done.
Resolving deltas: 100% (310/310), done.
user@dmskrzh:~/sample_app$
```

Рисунок 9 - Загрузка исходного кода

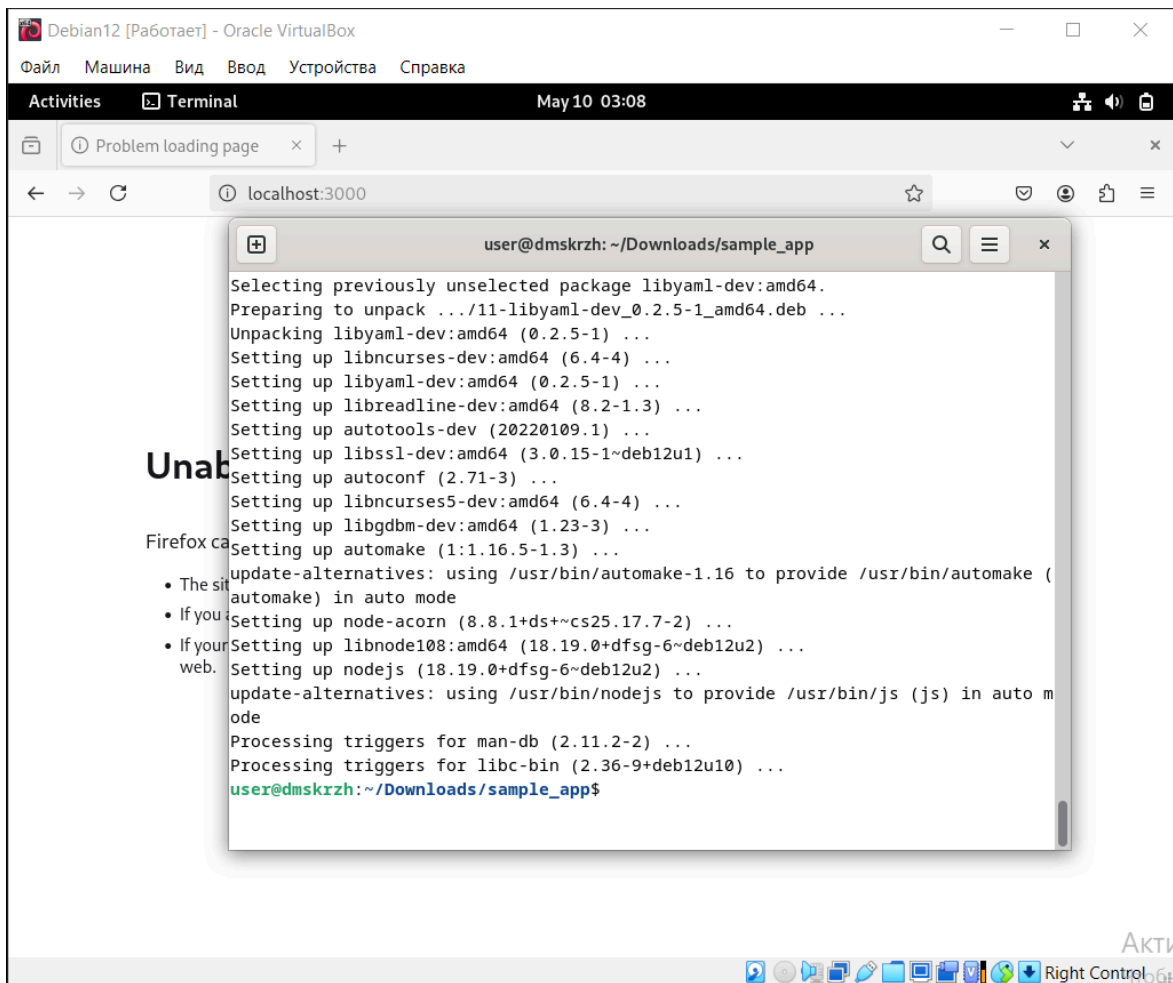


Рисунок 10 - Установка зависимостей

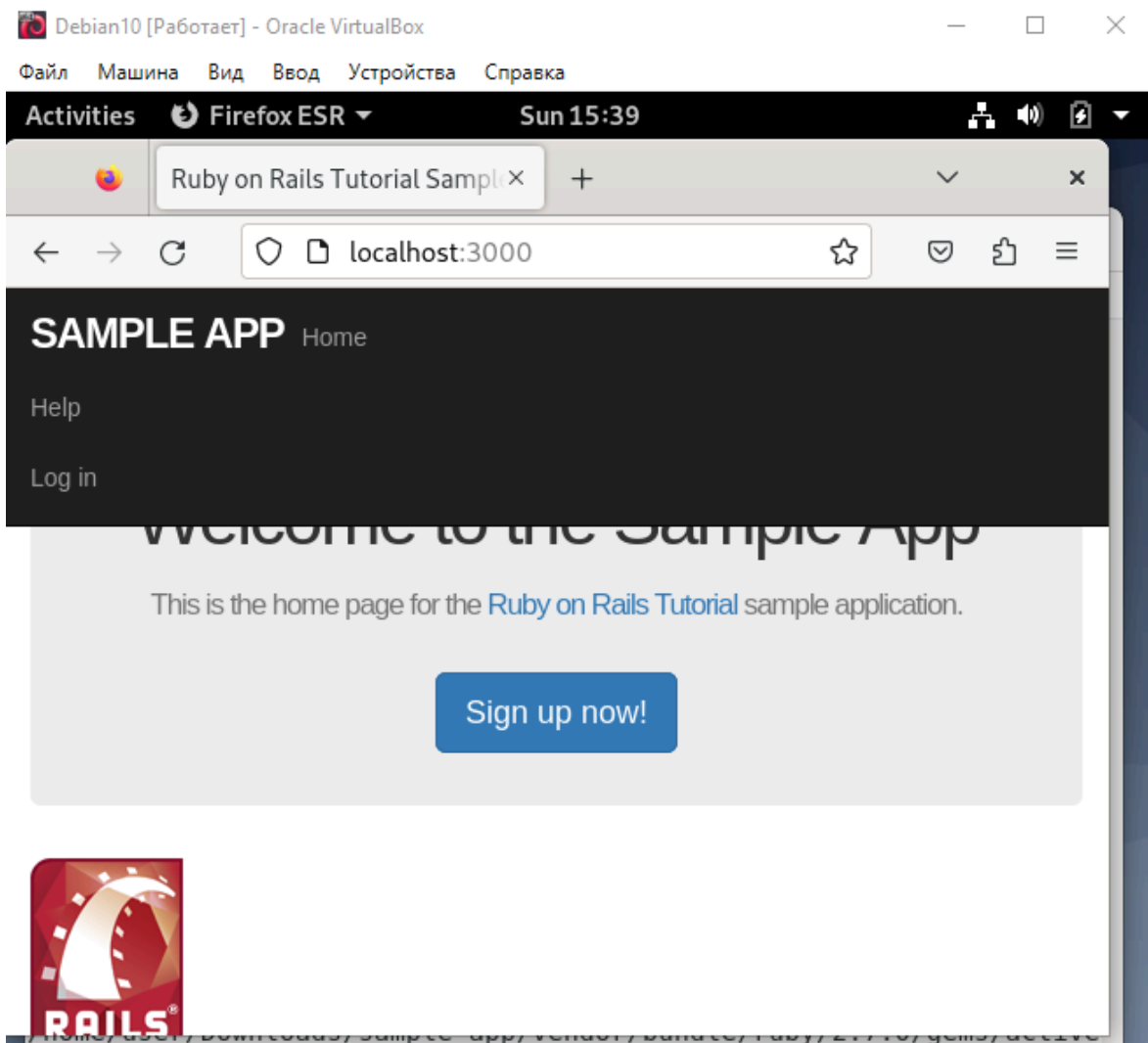
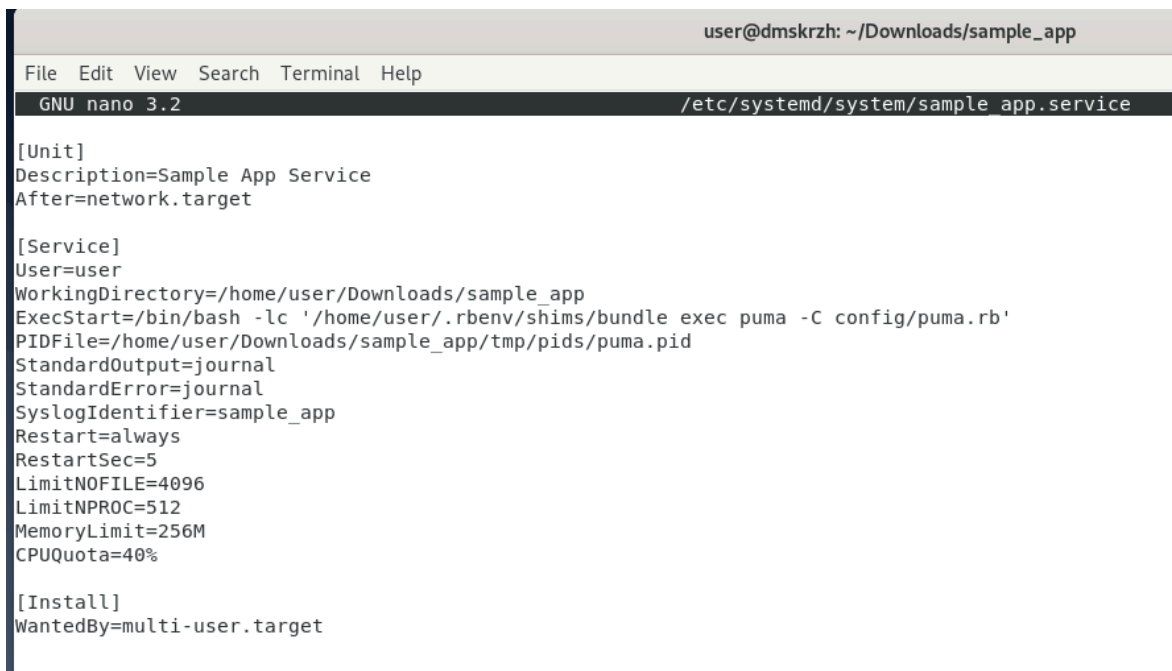


Рисунок 11 - Запущенное приложение

6.1 Определите процесс управления приложением с помощью systemd.

6.2 Описание сервиса должно включать описание основных операций, ведение журнала, пользователя, от имени которого будет запускаться приложение, рабочий каталог и каталог, который будет содержать файл с PID процесса.



```
user@dmskrzh: ~/Downloads/sample_app
File Edit View Search Terminal Help
GNU nano 3.2 /etc/systemd/system/sample_app.service

[Unit]
Description=Sample App Service
After=network.target

[Service]
User=user
WorkingDirectory=/home/user/Downloads/sample_app
ExecStart=/bin/bash -lc '/home/user/.rbenv/shims/bundle exec puma -C config/puma.rb'
PIDFile=/home/user/Downloads/sample_app/tmp/pids/puma.pid
StandardOutput=journal
StandardError=journal
SyslogIdentifier=sample_app
Restart=always
RestartSec=5
LimitNOFILE=4096
LimitNPROC=512
MemoryLimit=256M
CPUQuota=40%

[Install]
WantedBy=multi-user.target
```

Рисунок 10 - Создание unit-файла

6.3 Настройте ограничения на используемые системные ресурсы. Приложение должно запускаться с помощью команды `systemctl`.

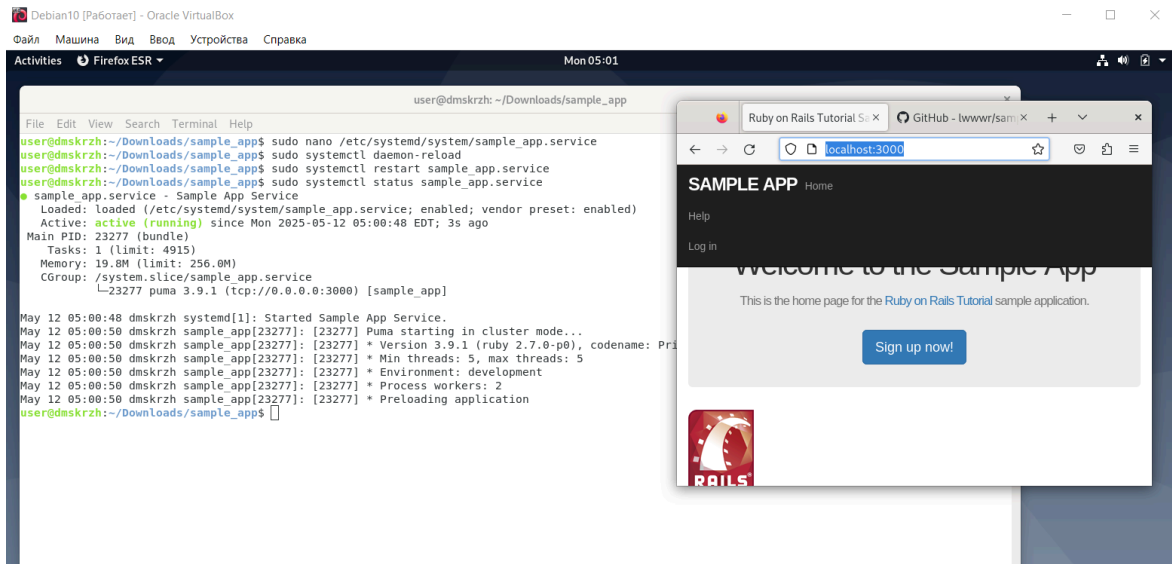


Рисунок 11 - Перезагрузка демона и запуск службы

Контрольные вопросы

1. Что такое служба ОС?

Служба – программа, запускаемая операционной системой и работающая в фоновом режиме без прямого взаимодействия с пользователем.

2. Зачем нужны службы в ОС?

Службы в ОС работают в фоновом режиме, автоматически управляя ресурсами, обеспечивая стабильность, безопасность и поддержку системных функций без участия пользователя.

3. Что такое юнит в Linux?

Юниты – это файлы конфигурации, хранящие информацию о службе, устройстве или сокете.

4. Какие юниты доступны в Ubuntu?

- `service` – программа, которая управляет работой других служб;
- `socket` – сокет для взаимодействия процессов (определенных службой);
- `device` – конфигурационный файл, который содержит инструкции для обработки списка устройств;
- `mount` – файл, содержащий информацию о точке монтирования файловой системы;

- automount – файл, содержащий информацию о списке устройств, которые должны быть смонтированы автоматически;
- target – файл, указывающий на группу служб, которые требуется выполнить этой единице;
- snapshot – файл, указывающий на другие единицы и восстанавливающий состояние ранее запущенных служб.

5. Инструмент инициализации systemctl.

Systemctl – инструмент системы инициализации systemd, предназначенный для управления службами в ОС Ubuntu Linux.

6. Основные параметры команды systemctl.

- start NAME – Запустить службу
- stop NAME – Остановить службу
- restart NAME – Перезапустить службу
- reload NAME – Обновить конфигурацию службы
- status NAME – Проверить статус службы
- is-active NAME – Проверить, работает ли служба
- is-failed NAME – Проверить, завершилась ли служба с ошибкой
- enable NAME – Включить автозапуск
- disable NAME – Отключить автозапуск
- is-enabled NAME – Проверить, включён ли автозапуск
- daemon-reload – Обновить конфигурацию systemd после изменения файла
- list-units – Список активных служб
- list-unit-files – Список файлов конфигурации служб
- list-dependencies NAME – Показать зависимости службы
- kill NAME – Завершить процесс службы

7. Как запустить или остановить службу в Ubuntu.

Для запуска и остановки служб в ОС Ubuntu используются следующие команды:

```
sudo systemctl start service_name
sudo systemctl stop service_name
```

8. Назначение служб в Windows.

Службы выполняют ключевые функции, необходимые для работы серверов и рабочих станций.

9. Графический инструмент для управления службами в Windows.

Основным инструментом для администрирования служб в графическом режиме на локальном компьютере является утилита *Службы*.

10. Основные командлеты PowerShell для управления службами Windows.

- Get-Service – Получить список всех служб
- Start-Service – Запустить службу
- Stop-Service – Остановить службу

- Restart-Service – Перезапустить службу
- Set-Service – Настроить службу
- New-Service – Создать службу

11. Как запустить или остановить службу в Windows?

Для запуска и остановки служб в ОС Windows используются следующие команды:

```
Start-Service -Name Service_name -PassThru  
Stop-Service -Name Service_name -Force
```