|  |
| --- |
|  |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |   Институт Информационных технологий  Кафедра Инструментального и прикладного программного обеспечения  **ОТЧЕТ ПО ПРАКТИЧЕСКИМ РАБОТАМ №1**  **по дисциплине**  «Настройка и администрирование сервисного программного обеспечения»   |  |  | | --- | --- | | Выполнил:  Студент группы ИКБО-20-22 | Шумахер М. Е. | | Проверил: | Зарипов Е. А. |   МОСКВА 2024 г. |

**ОГЛАВЛЕНИЕ**

[ПРАКТИЧЕСКАЯ РАБОТА №1 3](#_Toc160934857)

[ПРАКТИЧЕСКАЯ РАБОТА №2 10](#_Toc160934858)

[ПРАКТИЧЕСКАЯ РАБОТА №3 26](#_Toc160934859)

[ПРАКТИЧЕСКАЯ РАБОТА №4 36](#_Toc160934860)

ПРАКТИЧЕСКАЯ РАБОТА №1

**Цель работы**: получить навыки по развертыванию приложения Docker.

**Задание на практическую работу:**

Установить Docker. Запустить Docker Dashboard. Разобраться с функционалом Docker Dashboard, ознакомившись с https://docs.docker.com/desktop/dashboard/ Для установки и запуска контейнеров Docker в ОС Windows запустить cmd (command line) и выполнить команду docker run имя\_контейнера. Например, docker run redis. Таким образом, в данном случае будет установлена СУБД redis. Далее перейти в Docker Dashboard, найти установленный и запущенный контейнер (в нашем случае redis) выбрать его и найти кнопку CLI (command line interface – командная строка для данного контейнера. Пиктограмма выглядит следующим образом: >\_ )

Ознакомиться с операциями, производимыми в консоли Docker и выполнить следующие действия:

1. Обратиться по адресу https://hub.docker.com/ и ознакомиться с приложениями доступными для работы с Docker.
2. Установить контейнер СУБД redis. Используя CLI контейнера, запустить клиент redis-cli.
3. Ознакомиться с документацией по командам redis. Документация на СУБД <https://redis.io/commands>.
4. В redis выполнить команды SET https://redis.io/commands/set и GET https://redis.io/commands/get Создать 5 ключей со значениями с помощью SET и прочитать ключи со значениями с помощью GET.
5. Получение значения по ключу и его замена на новое.
6. Добавление строки к уже существующему значению.
7. Добавление числа и изменение его значения.
8. Создание ключа со значением типа хеш-таблица
9. Работа со множествами. Задействовать команды SADD, SDIFF, SMOVE, SPOP, SUNION, SREM.
10. Работа с упорядоченными наборами. Задействовать команды ZADD, ZCOUNT, ZDIFF, ZPOPMAX, ZPOPMIN, ZUNION, ZMSCORE, ZLEXCOUNT.
11. Из документации выбрать любые не использовавшиеся ранее 5 команд и задействовать их в работе.

**Выполнение заданий**

Обратимся по адресу https://hub.docker.com и ознакомимся с приложениями доступными для работы с Docker, Рисунок 1.

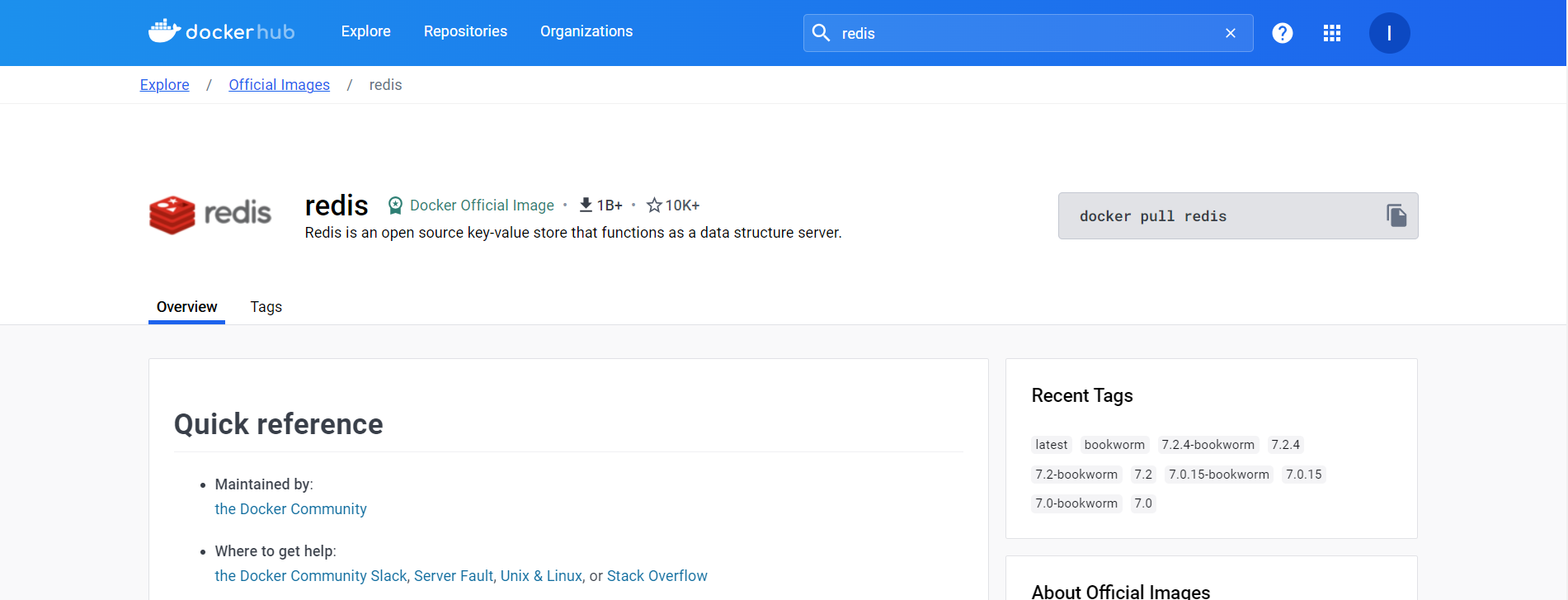


Рисунок 1 – Ознакомление с образом redis на Docker Hub

Установим контейнер СУБД redis, скачав при установке образ redis. Используя CLI контейнера, запустим клиент redis-cli, Рисунок 2.

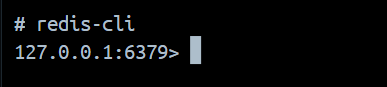


Рисунок 2 – Запуск redis-cli

Ознакомимся с документацией по командам redis из документации на https://redis.io/commands, Рисунок 3.



Рисунок 3 – внесение изменений

В redis-cli выполним команды SET и GET. Создаем 5 ключей со значениями с помощью SET и прочитываем ключи со значениями с помощью GET., Рисунок 4.

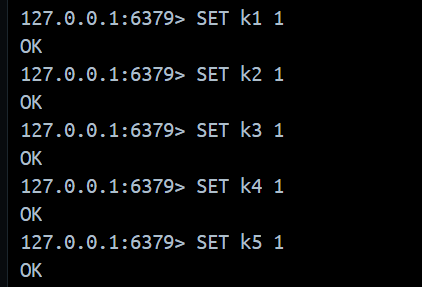


Рисунок 4 – создание 5 ключей со значениями и их чтение   
с помощью команд SET и GET

Получим значения по ключу и заменим его на новое значение через команду GETSET, Рисунок 5.

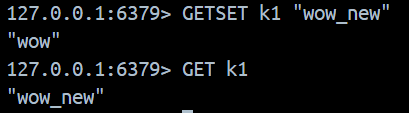


Рисунок 5 – получение значения и замена его на новое

Добавим строку к уже существующему значению через команду APPEND, Рисунок 6.

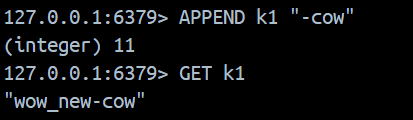


Рисунок 6 – добавление строки к существующему ключу

Добавим число и изменим его значение на 10, Рисунок 7.

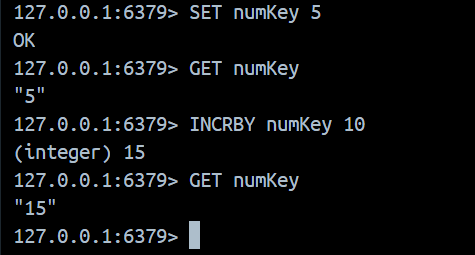


Рисунок 7 – добавление числа и изменение его значения на 5

Создадим ключ со значением типа хеш-таблица, Рисунок 8.

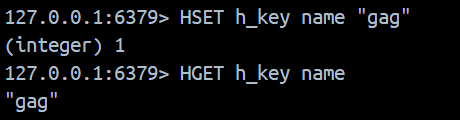


Рисунок 8 – создание ключа со значением типа хеш-таблица

Команды: SADD – добавление значения в множество, SDIFF – вывод разницы между множествами, SMOVE – перемещение элемента из одного множества в другой, SPOP – удаление и возвращение случайного элемента из множества, SUNION – возвращение объединённого множества, SREM – удаление элемента из множества. Задействуем данные команды для работы с множествами, Рисунок 9.

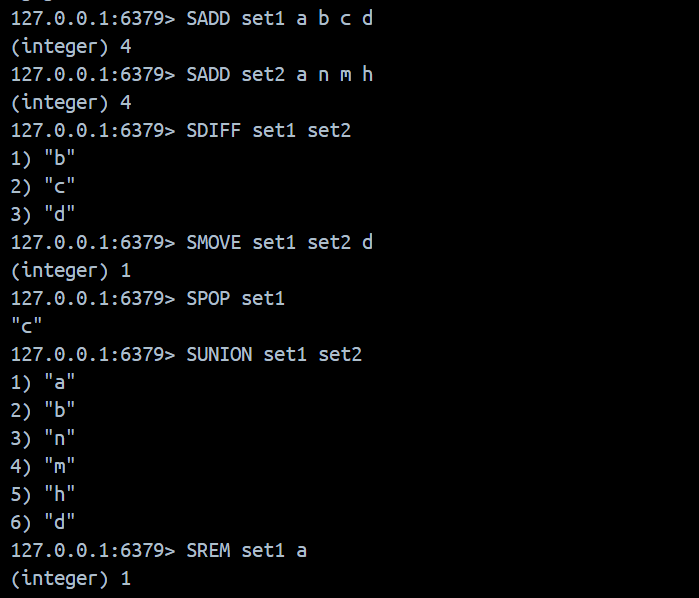
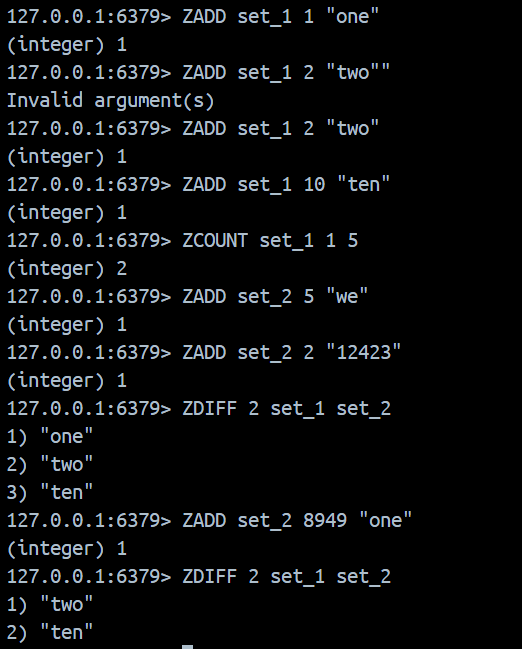


Рисунок 9 – Использование команд для работы с множествами

Команды: ZADD – добавление значения в набор, ZCOUNT – вывод кол-ва элементов на заданном диапазоне, ZDIFF – вывод разницы между двумя наборами, ZPOPMAX – удаление и возвращение элемента из набора с наивысшим приоритетом, ZPOPMIN – удаление и возвращение элемента из набора с низшим приоритетом, ZUNION – возвращает объединённый набор, ZMSCORE – возвращает приоритеты элементов в наборе, ZLEXCOUNT – возвращает количество элементов набора на заданном промежутке. Задействуем данные команды для работы с упорядоченными наборами, Рисунок 10.



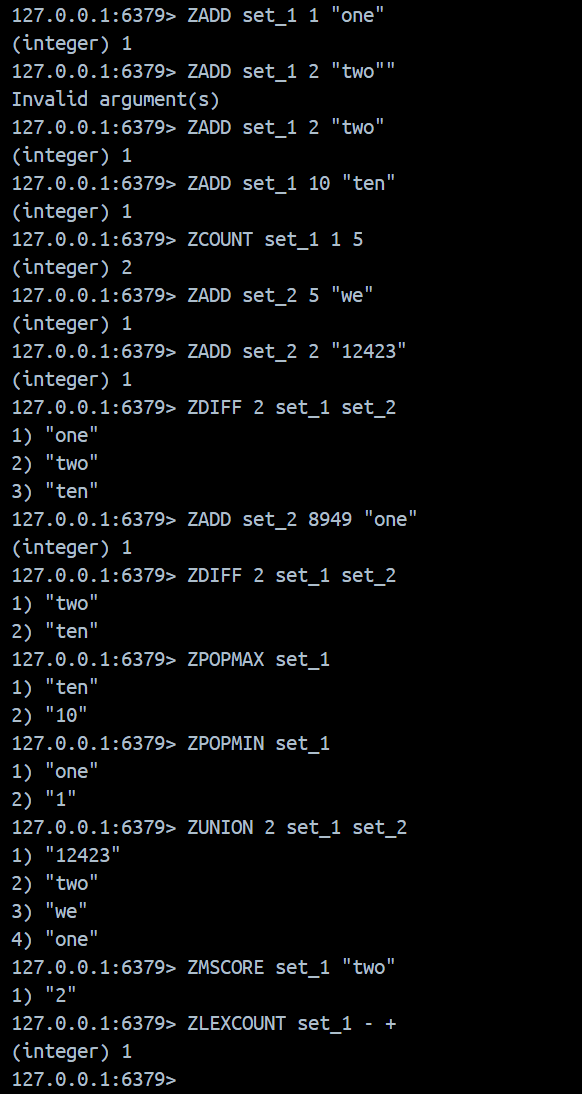


Рисунок 10 – Использование команд для работы с упорядоченными наборами

Выберем любые не использовавшиеся ранее 5 команд и задействуем их в работе, Рисунок 11.

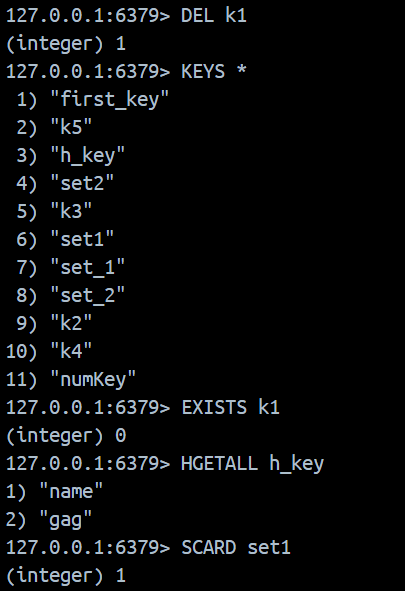


Рисунок 11 – Использование других 5-ти команд

**Вывод**: в результате выполнения работы были получены и освоены навыки работы с Docker и СУБД Redis.

ПРАКТИЧЕСКАЯ РАБОТА №2

**Цель работы**: получить навыки по развертыванию операционных систем Unix на основе Windows Subsystem для Linux в Windows 10.

**Задание на практическую работу:**

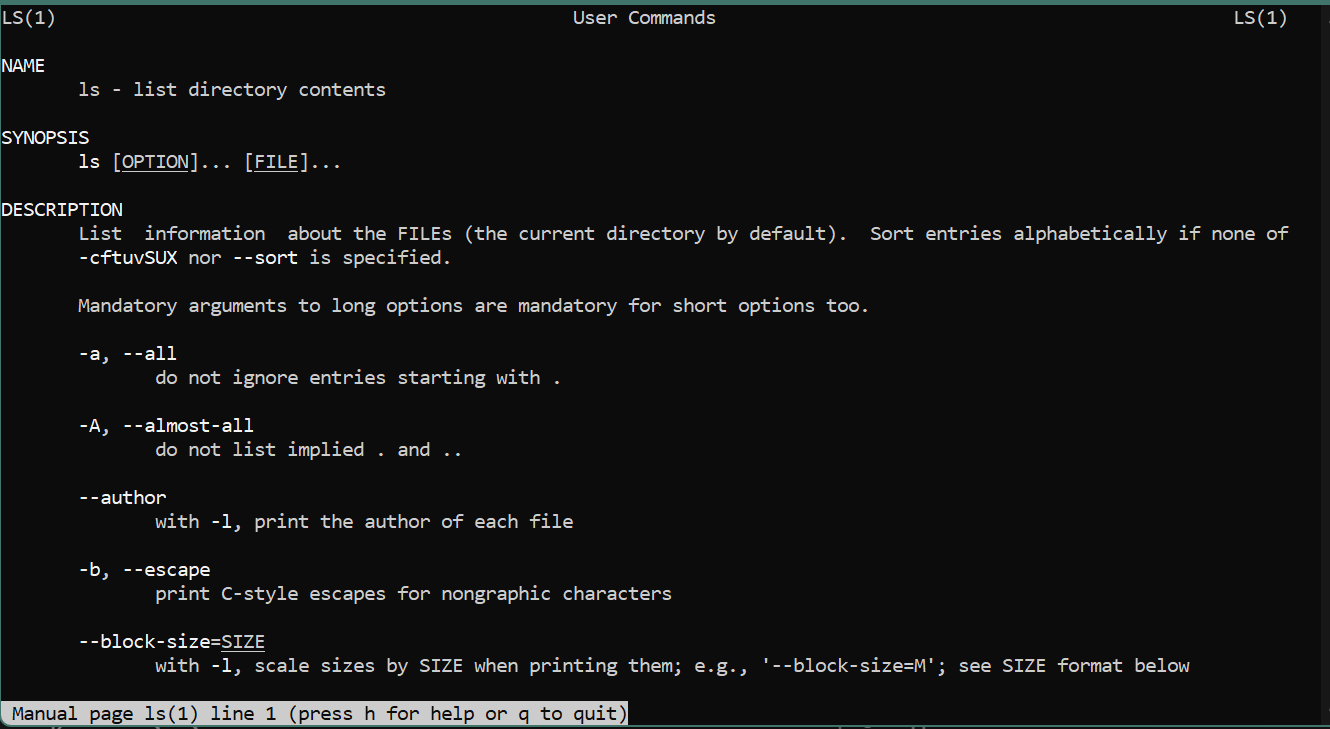
Установить Ubuntu.

Ознакомиться с операциями, производимыми в консоли установленной Linux системы и выполнить следующие команды:

1. Ознакомиться и выполнить команду man и опцию --help для любой команды
2. Ознакомиться и выполнить команду whatis
3. Ознакомиться и выполнить команду whereis Работа с текстом
4. Ознакомиться и выполнить команды more и less
5. Ознакомиться и выполнить команды head и tail
6. Ознакомиться и выполнить команду grep
7. Ознакомиться и выполнить команды sort
8. Ознакомиться и выполнить команды wc
9. Ознакомиться и выполнить команды diff
10. Отредактировать текстовый файл с помощью редакторов nano и vi Работа с процессами
11. Ознакомиться и выполнить команду kill
12. Ознакомиться и выполнить команду ps
13. Ознакомиться и выполнить команды top и htop. Посмотреть список запущенных процессов в операционной системе
14. Ознакомиться и выполнить команду time
15. Установить приложение используя команду sudo
16. Запустить приложение используя команду sudo
17. Ознакомиться и выполнить команду apt-get (с применением команды sudo)
18. Ознакомиться и выполнить команду date
19. Ознакомиться и выполнить команду uname
20. Ознакомиться и выполнить команду uptime
21. Ознакомиться и выполнить команду sleep
22. Ознакомиться и выполнить команды useradd, userdel и usermod
23. Ознакомиться и выполнить команду passwd
24. Ознакомиться и выполнить команду ip
25. Ознакомиться и выполнить команду ping
26. Ознакомиться и выполнить команду nethogs
27. Ознакомиться и выполнить команду traceroute
28. Ознакомиться и выполнить команду hostname
29. Ознакомиться и выполнить команду pwd
30. Ознакомиться и выполнить команду ls. Добавить параметры для этой команды -la. Выполнить эту же команду с параметрами -l и -a отдельно. Объяснить полученный результат
31. Ознакомиться и выполнить команду cd
32. Ознакомиться и выполнить команды mkdir и rmdir
33. Ознакомиться и выполнить команду rm
34. Ознакомиться и выполнить команду touch
35. Ознакомиться и выполнить команду cp
36. Ознакомиться и выполнить команду mv
37. Ознакомиться и выполнить команду locate
38. Ознакомиться и выполнить команду cat
39. Ознакомиться и выполнить команду df
40. Ознакомиться и выполнить команду du
41. Ознакомиться и выполнить команду tar
42. Ознакомиться и выполнить команды zip и unzip
43. Ознакомиться и выполнить команду chmod
44. Ознакомиться и выполнить команду chown
45. Ознакомиться и выполнить команду file
46. Ознакомиться и выполнить команду find
47. Ознакомиться и выполнить команду clear
48. Ознакомиться и выполнить команду halt
49. Ознакомиться и выполнить команду reboot
50. Ознакомиться и выполнить команды mount и umount

**Выполнение заданий**

Работа с документацией. Выполним команду man (предоставляет доступ к руководствам по использованию команд и программ) и опцию –help (отображает подсказку по командам) для любой команды, Рисунок 12.



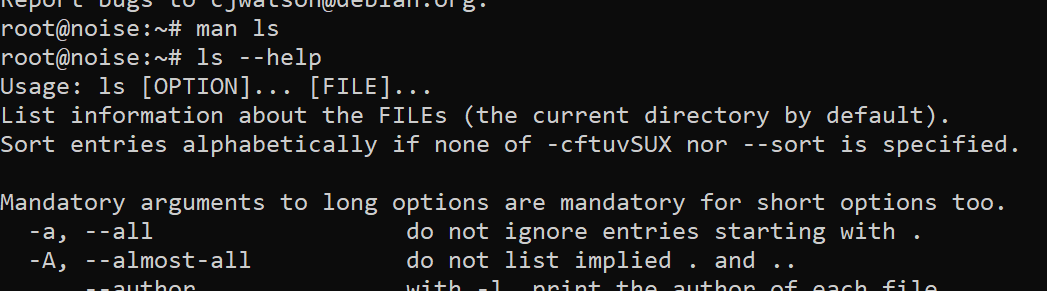


Рисунок 12 ­– Выполнение команды man и опции –help

Выполним команду whatis (выводит краткое описание команд), Рисунок 13.



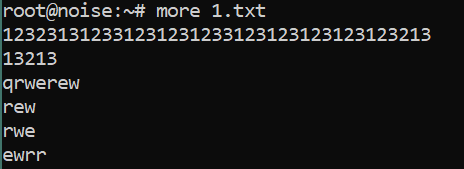
Рисунок 13 ­– Выполнение команды whatis

Выполним команду whereis (позволяет найти расположение двоичных файлов, файлов исходного кода и файлов справочной страницы для команды), Рисунок 14.



Рисунок 14­ – Выполнение команды whereis

Работа с текстом. Выполним команды more (позволяет просматривать содержимое текстовых файлов постранично) и less (усовершенствованная версия команды more), Рисунок 15.



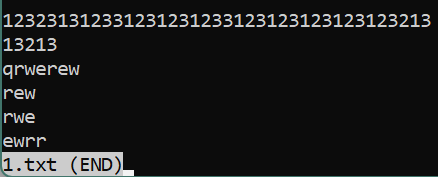


Рисунок 15 ­– Выполнение команд more, less

Выполним команды head (вывод первых строк из файлов) и tail (вывод последних строк из файлов), Рисунок 16.

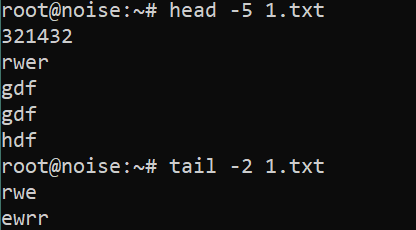


Рисунок 16 ­– Выполнение команды

Выполним команду grep (находит на вводе строки, отвечающие заданному регулярному выражению, и выводит их), Рисунок 17.

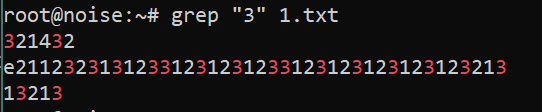


Рисунок 17 ­– Выполнение команды grep

Выполним команду sort (сортирует строки в файле и выводит их на консоль), Рисунок 18.

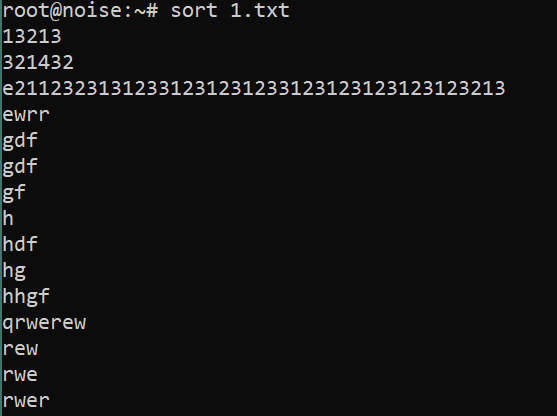


Рисунок 18 ­– Выполнение команды sort

Выполним команду wc (подсчитывает число строк, слов и байт в файлах, указанных в параметре), Рисунок 19.



Рисунок 19 ­– Выполнение команды wc

Выполним команду diff (выводит разницу в файлах), Рисунок 20.

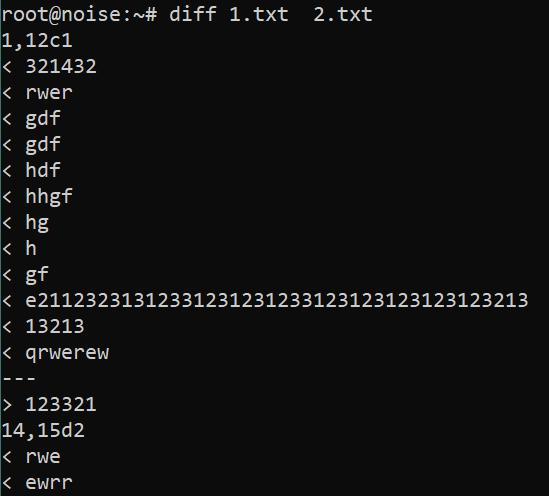
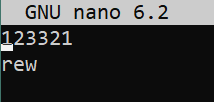


Рисунок 20 ­– Выполнение команды diff

Отредактируем текстовый файл с помощью редакторов nano и vi, Рисунок 21.



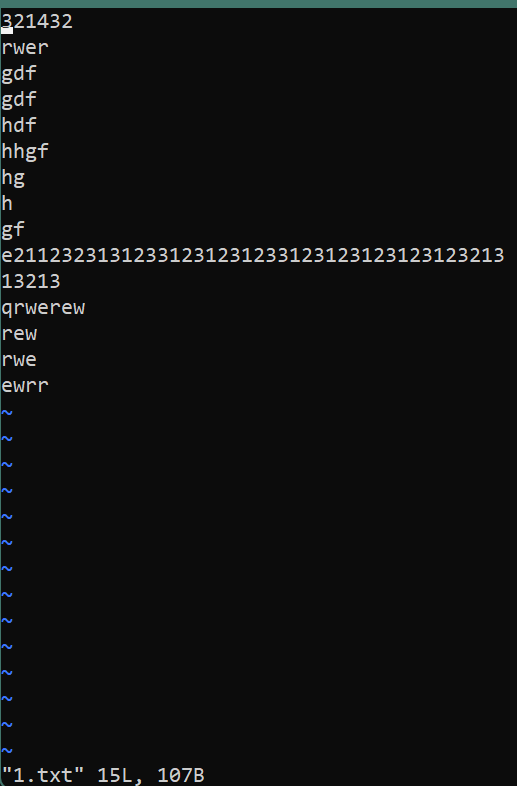


Рисунок 21 ­– Выполнение команд nano, vi

Работа с процессами. Выполним команду kill (завершение работы процесса), Рисунок 22.



Рисунок 22 ­– Выполнение команды kill

Выполним команду ps (вывод всех процессов системы), Рисунок 23.

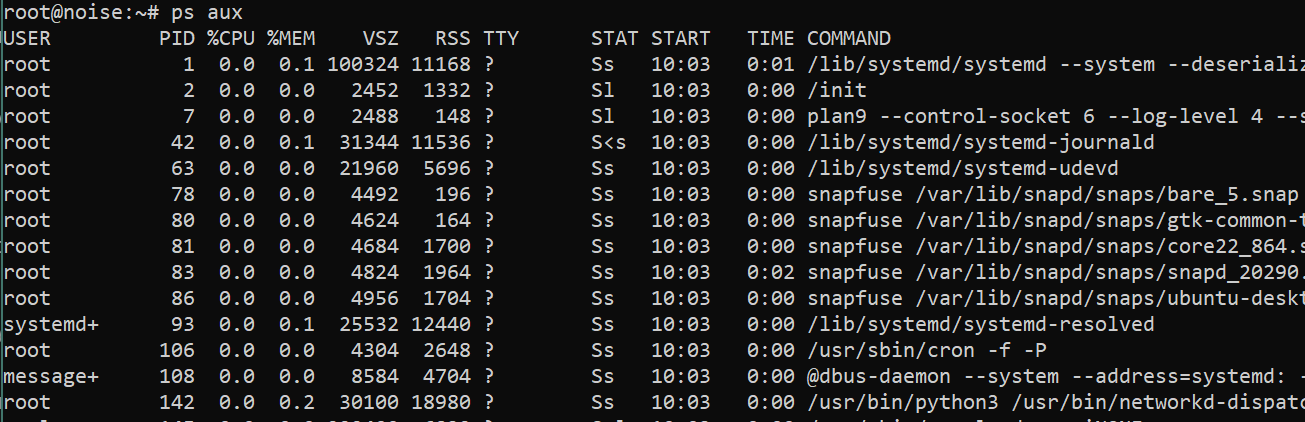


Рисунок 23 ­– Выполнение команды ps

Выполним команды top (позволяет пользователям отслеживать процессы и использование системных ресурсов) и htop и просмотрим список запущенных процессов в операционной системе, Рисунок 24.

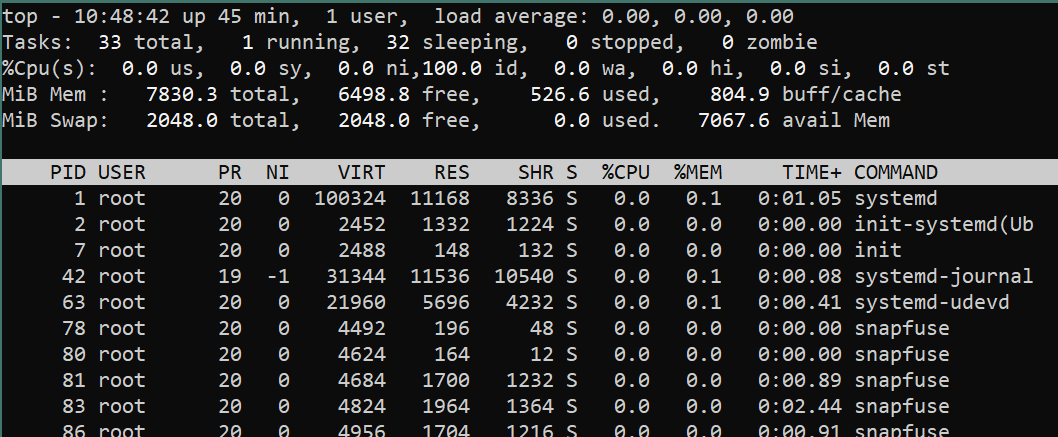
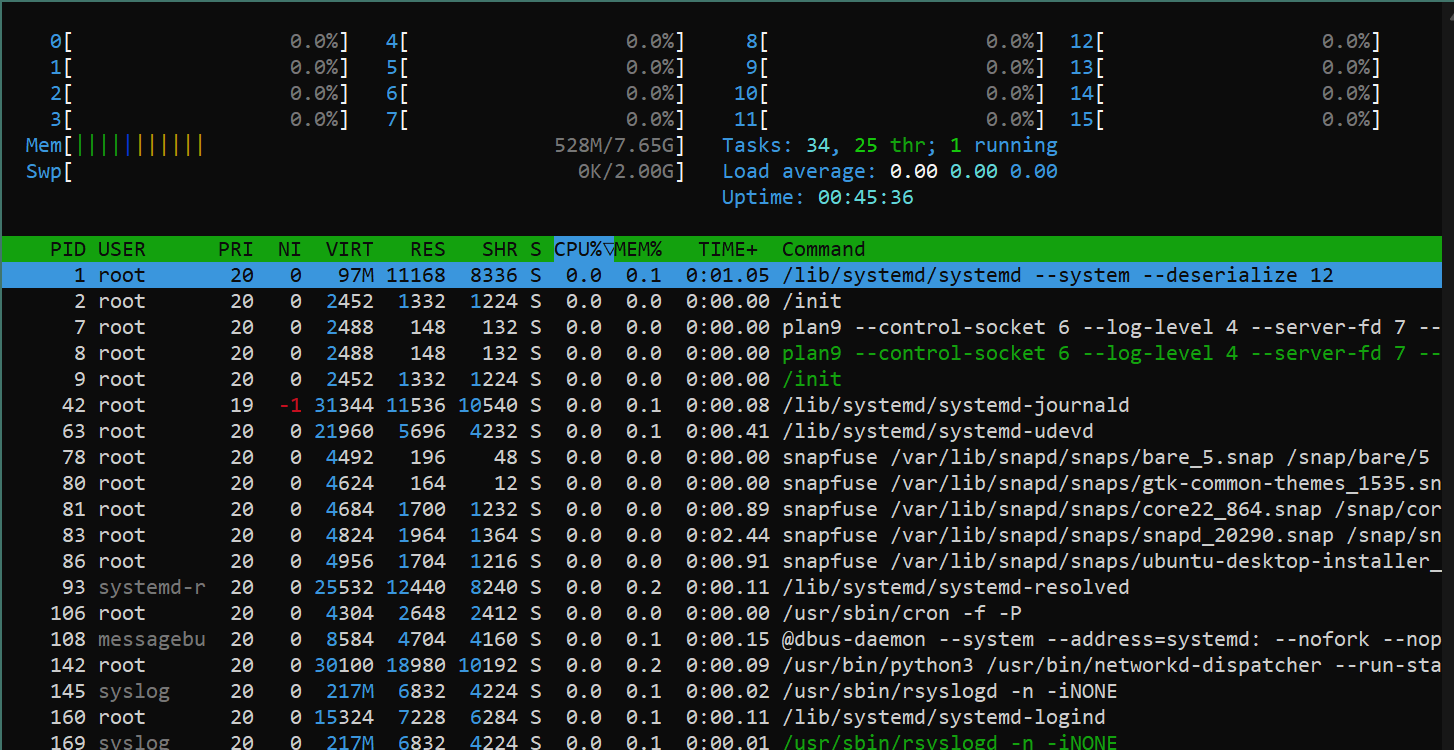
 

Рисунок 24 ­– Выполнение команды top, htop

Выполним команду time (запускает заданную пользователем команду и после этого выводит информацию о времени ее выполнения), Рисунок 25.

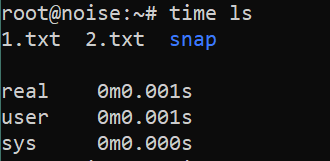


Рисунок 25 ­– Выполнение команды time

Команды пользователя. Установим приложение используя команду sudo, Рисунок 26.

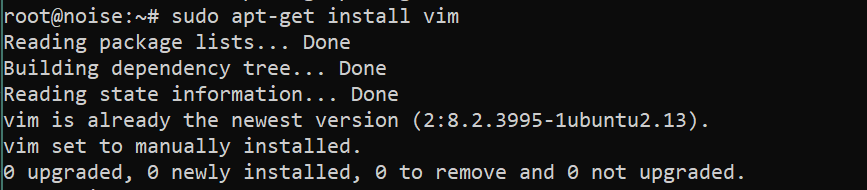


Рисунок 26 ­– Выполнение команды sudo apt-get

Запустим приложение используя команду sudo(рис. 27.

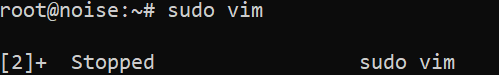


Рисунок 27 ­– Выполнение команды sudo vim

Выполним команду apt-get, Рисунок 28.

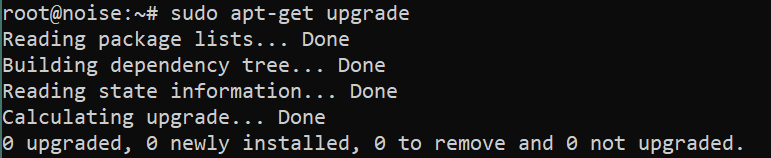


Рисунок 28 ­– Выполнение команды apt-get

Выполним команду date, Рисунок 29.



Рисунок 29 ­– Выполнение команды date

Выполним команду uname, Рисунок 30.



Рисунок 30 ­– Выполнение команды uname

Выполним команду uptime, Рисунок 31.



Рисунок 31 ­– Выполнение команды uptime

Выполним команду sleep, Рисунок 32.



Рисунок 32 ­– Выполнение команды sleep

Управление пользователями. Выполним команды useradd, userdel и usermod, passwd, Рисунок 33.

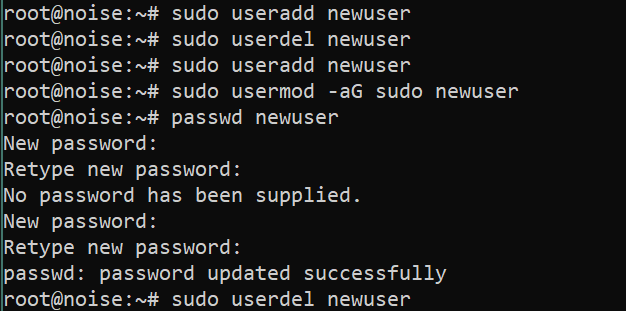


Рисунок 33 ­– Выполнение команд управления пользователями

Выполним команду ip, Рисунок 34.

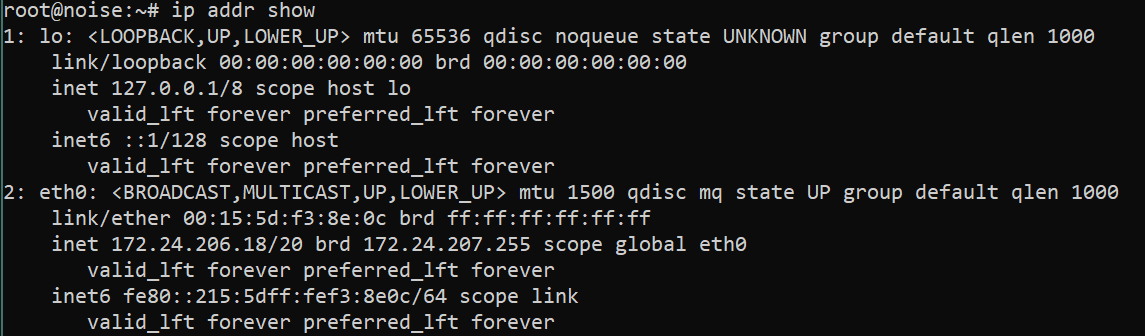


Рисунок 34 ­– Выполнение команды ip

Выполним команду ping, Рисунок 35.

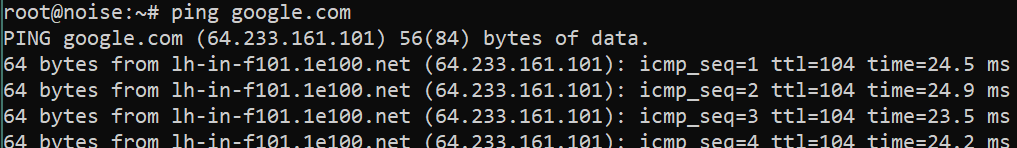


Рисунок 35 ­– Выполнение команды ping

Выполним команду nethogs (просмотр состояния сетей), Рисунок 36.

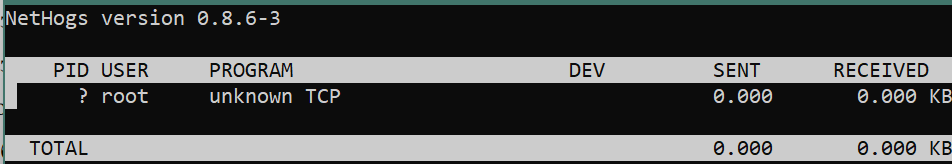


Рисунок 36 ­– Выполнение команды nethogs

Выполним команду traceroute (отправка пакетов на url), Рисунок 37.

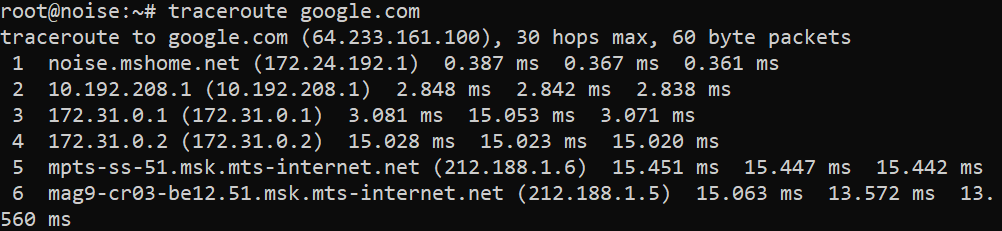


Рисунок 37 ­– Выполнение команды traceroute

Выполним команду hostname, Рисунок 38.



Рисунок 38 ­– Выполнение команды hostname

Выполним команду pwd, Рисунок 39.



Рисунок 39 ­– Выполнение команды pwd

Выполним команду ls. Добавить параметры для этой команды -la. Выполнить эту же команду с параметрами -l и -a отдельно, Рисунок 40. Объяснить полученный результат:

-la выводит длинный лист файлов (также начинающихся на .)

-l выводит длинный лист файлов (все, кроме начинающихся на .)

-a выводит названия файлов (также начинающихся на ..

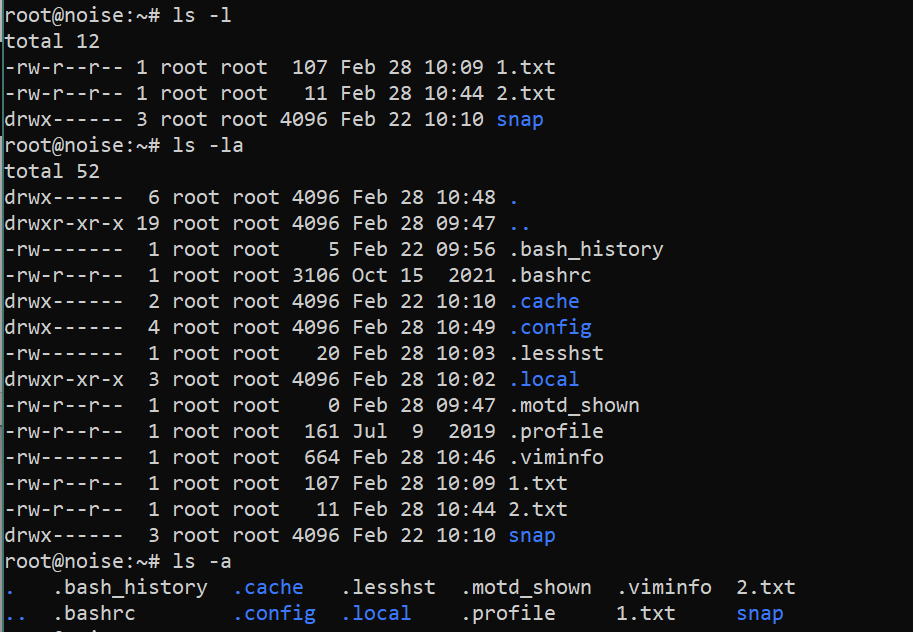


Рисунок 40 ­– Выполнение команд ls

Выполним команду cd, Рисунок 41.

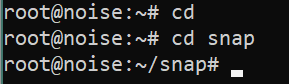


Рисунок 41 ­– Выполнение команды cd

Выполним команды mkdir и rmdir, rm, cp, mv, touch, Рисунок 42.

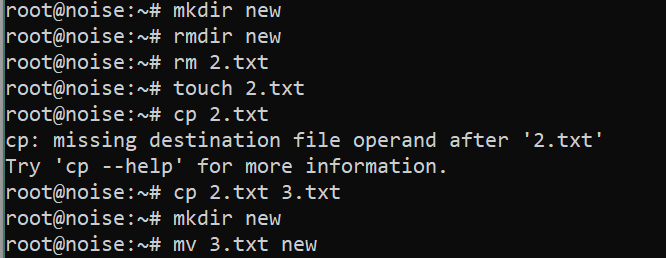


Рисунок 42 ­– Выполнение команд mkdir и rmdir, rm, cp, mv, touch

Выполним команду locate, Рисунок 43.

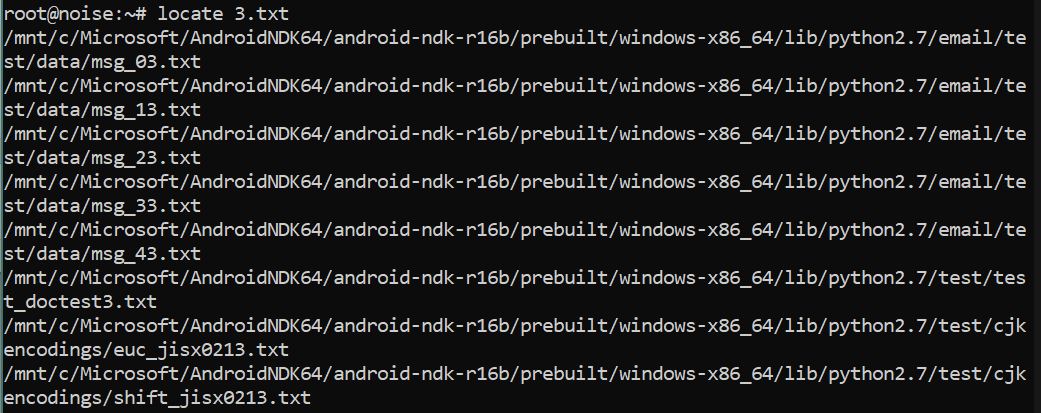


Рисунок 43 ­– Выполнение команды locate

Выполним команду cat, df, Рисунок 44.

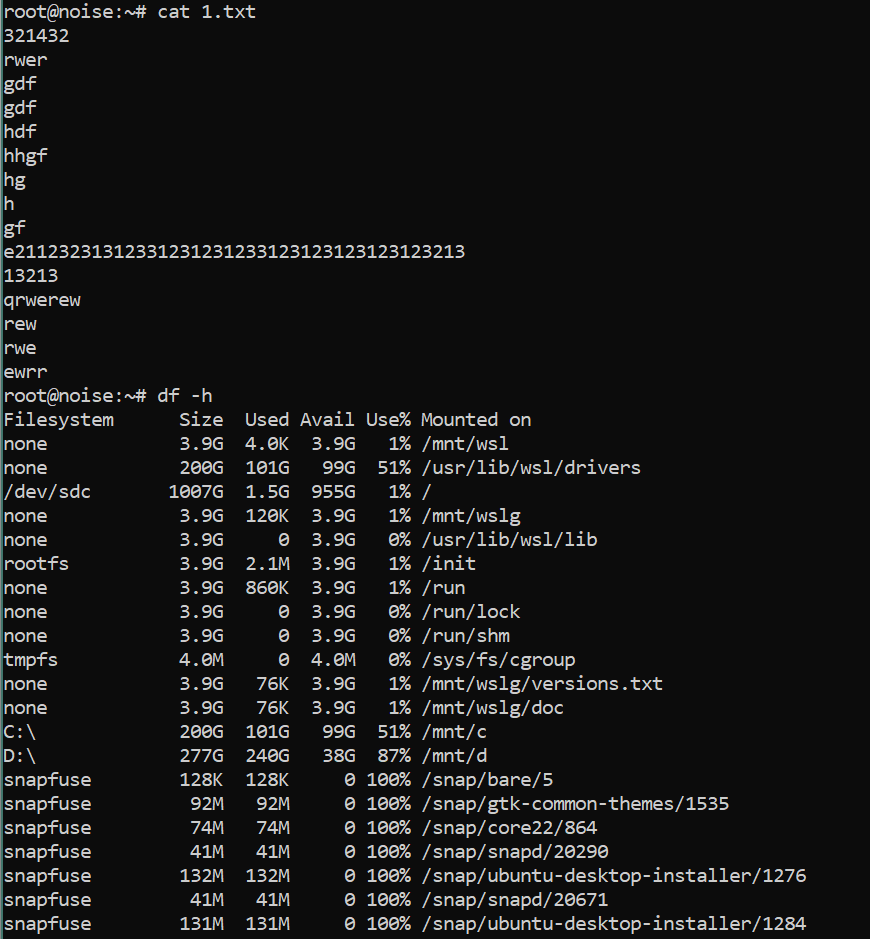


Рисунок 44 ­– Выполнение команд cat, df

Выполним команду du, Рисунок 45.

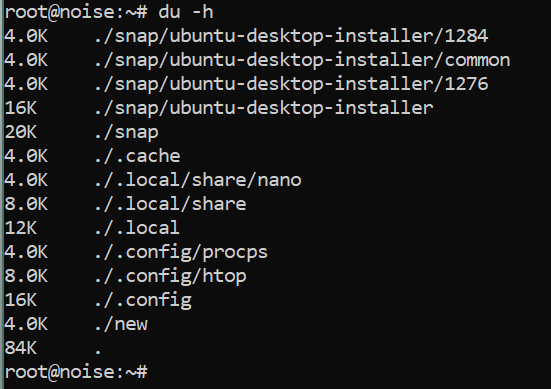


Рисунок 45 ­– Выполнение команды du

Выполним команду tar, Рисунок 46.



Рисунок 46 ­– Выполнение команды tar

Выполним команды zip и unzip, chmod, chown, Рисунок 47.

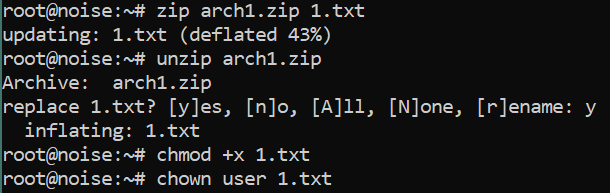
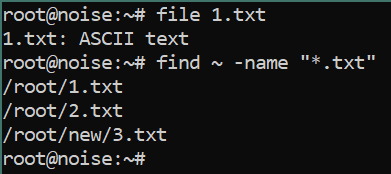


Рисунок 47 ­– Выполнение команд zip и unzip, chmod, chown

Выполним команды file, find, Рисунок 48.



Рисуно 48 – Выполнение команд file, find

Выполним команду clear, выполним команду halt (принудительное завершение работы процессора), Рисунок 49.

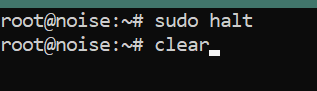


Рисунок 49 ­– Выполнение команды halt, clear

Выполним команду reboot, Рисунок 50.



Рисунок 50 ­– Выполнение команды reboot

Выполним команды mount и umount, Рисунок 51.

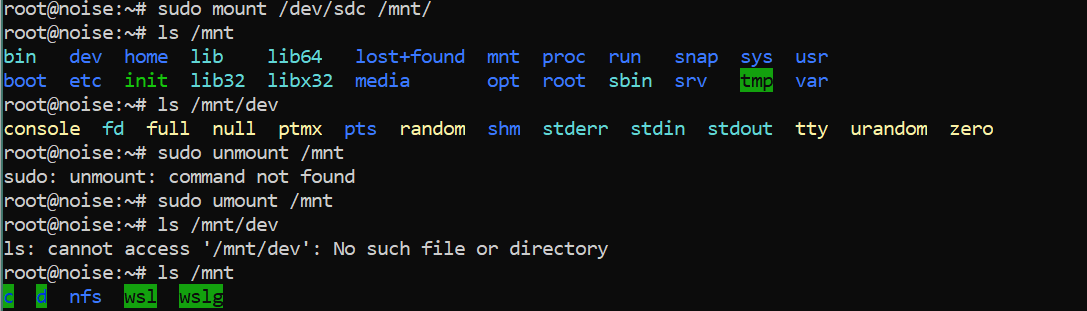


Рисунок 51 ­– Выполнение команд mount и umount

**Вывод:** в результате выполнения работы были получены и освоены навыки работы с WSL и основными командами Linux.

ПРАКТИЧЕСКАЯ РАБОТА №3

**Цель работы:** получить навыки по запуску Linux-приложений в Windows 10, а также запуску Windows-приложений из ОС Linux.

**Задание на практическую работу:**

В Windows 10 появились три классические консольные UNIX утилиты curl.exe, tar.exe, ssh.exe.

curl --version

ssh -V

tar --version

ver – вывести версию Windows

Ознакомиться и выполнить команды в командной строке Windows 10. Для выполнения работы понадобится командный интерпретатор cmd. После запуска cmd с помощью команды cd переместиться в удобный каталог для работы (там, где удобнее хранить временные файлы, например, c:\temp, например, cd c:\temp. Все выполняемые команды проверить на работоспособность и привести в отчете. В cmd работает поиск по комбинации CTRL-F.

1. Проверить версию команды curl. Например, curl –version
2. Открыть 10 URL различных ресурсов и ознакомиться с полученными данными. Например, curl mail.ru curl rambler.ru curl google.com
3. Осуществить загрузку файла с помощью команды curl с оригинальным именем, опция -O. Скачать одной командой 3 файла с опцией -О.
4. Осуществить загрузку с 3 ftp-серверов (например, производителей компьютерного оборудования HP, Asus, MSI, Lenovo, Dell) файлы (например, драйверы. Например, curl -O ftp://ftp.hp.com:21/pub/softpaq/sp40001/sp40001.cva Возможно, для ftp потребуется указывать логин, пароль Например, curl -u anonymous:anonymous -O ftp://yourftpserver/yourfile.tar.gz anonymous – стандартный логин и пароль для анонимного входа на ftpсервер
5. Осуществить загрузку файла с помощью команды curl и сохраните файл под другим именем, опция -o. Скачайте три файла с опцией -o для сохранения файлов с определенным именем
6. Сохранить какое-либо изображение используя опцию -OL . Сохраните 3 изображения используя опцию -OL (можно сохранить каждое изображение по отдельности)
7. Посмотреть для 5 URL какие файлы cookie загружаются на компьютер. Например, curl --cookie-jar cnncookies.txt https://www.cnn.com/index.html -O. Просмотреть сформированные cookie файлы можно с помощью команды type. Например, type cnncookies.txt
8. Выполнить команду curl с опцией -L. Сравнить вывод без этой опции. Объяснить полученный результат вывода. Например, curl -L google.com
9. Получить заголовки сайтов 10 URL командой curl, опция -I
10. Составить (а при наличии тестового сервера и отправить) POST-запросы используя команду curl, в виде списка имя=значение, и в виде JSON, для 3 URL, опция -d Например, curl -d "param1=test1&param2=test2" http://test.com (ключ=значение) curl -d '{"param1":"test1","param2":"test2"}' \http://www.test.com (JSON)
11. Использовать авторизацию для 3 URL, опция -u. Например, curl -u <https://testurl.com/entry>
12. Перенаправить запрос с localhost на URL (эквивалентно прописыванию в /etc/hosts), сделать для 3 URL. Для каждого URL свое перенаправление отдельной командой., опция –resolve. Например, curl --resolve www.test.com:80:localhost <http://www.test.com/>
13. Вывести время работы команды для конкретного URL. Повторить команду 3 раза на одном URL. Выполнить такую же операцию для 5 различных URL, опция -w Например, curl -w "%{time\_total}\n" -o /dev/null -s [www.mail.ru](http://www.mail.ru)
14. Для 3 URL осуществить имитацию определенного браузера, опция -A. Может понадобиться в случае, если удаленный сервер может быть настроен так, чтобы блокировать пользовательский агент (браузер) или возвращать различное содержимое в зависимости от устройства посетителя и браузера Например, curl -A "Mozilla/5.0 (X11; Linux x86\_64; rv:60.0) Gecko/20100101 Firefox/60.0" <https://getfedora.org/>
15. Отправить файлы cookie при скачивании дистрибутива. По умолчанию при запросе ресурса с помощью curl файлы cookie не отправляются и не сохраняются. Чтобы отправить файлы cookie на сервер, необходимо использовать опцию -b за которой следует имя файла, содержащего файлы cookie, или строку. Например, чтобы загрузить rpm-файл Oracle Java JDK jdk-10.0.2\_linux-x64\_bin.rpm необходимо передать файл cookie с именем oraclelicense со значением a curl -L -b "oraclelicense=a" -O <http://download.oracle.com/otnpub/java/jdk/10.0.2+13/19aef61b38124481863b1413dce1855f/jdk10.0.2_linux-x64_bin.rpm>
16. Выполнить команду по загрузке rpm-файла Oracle Java JDK jdk10.0.2\_linux-x64\_bin.rpm без отправки cookie. Сравнить скачанные файлы и объяснить полученный результат
17. Самостоятельно выбрать ранее не использовавшиеся 5 опций команды curl. Выполнить команды и объяснить полученный результат.

**Выполнение задания**

Проверим версию команды curl, Рисунок 52.

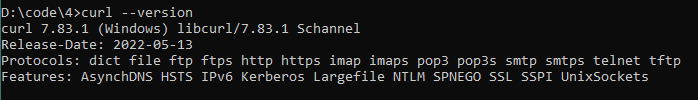


Рисунок 52 – Проверка версии curl

Откроем 10 URL различных ресурсов и ознакомимся с полученными данными, Рисунок 53.

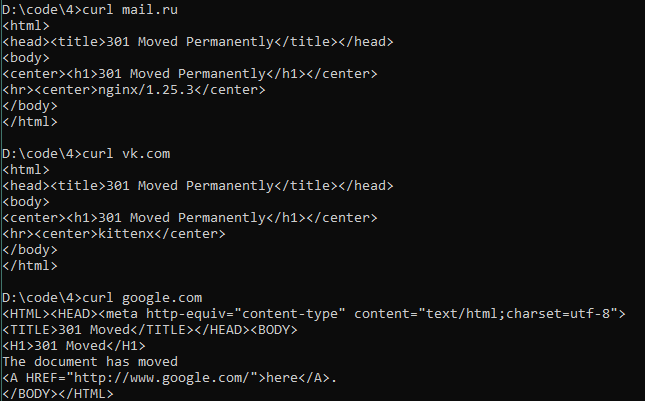


Рисунок 53 – Открытие 10 URL

Данное действие было повторено еще на 7 URL. Осуществим загрузку файла с помощью команды curl с оригинальным именем, опция -O. Скачаем одной командой 3 файла с опцией -О, Рисунок 54.

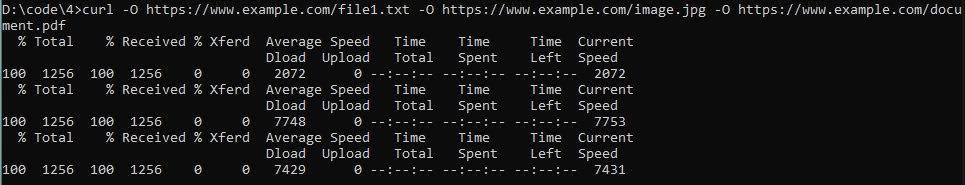


Рисунок 54 – Осуществление загрузки файла с помощью команды curl

Осуществим загрузку файлов с 3 разных ftp-серверов, Рисунок 55.

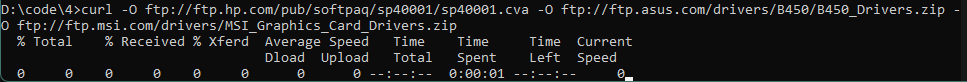


Рисунок 55 – Осуществление загрузки файлов с ftp-серверов

Осуществим загрузку файла с помощью команды curl и сохраним файл под другим именем, опция -o. Скачаем три файла с опцией -o для сохранения файлов с определенным именем, Рисунок 56.

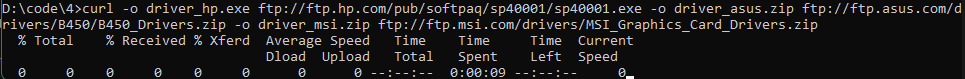


Рисунок 56 – Осуществление загрузки файлов с опцией -o

Сохраним какое-либо изображение используя опцию -OL. Сохраним 3 изображения используя опцию -OL, Рисунок 57.

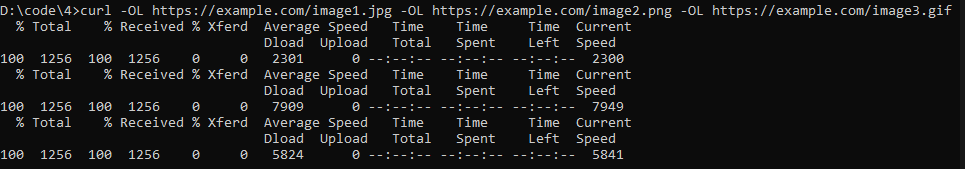


Рисунок 57 – Осуществление загрузки изображений с опцией -OL

Посмотрим для 5 URL какие файлы cookie загружаются на компьютер, Рисунок 58.

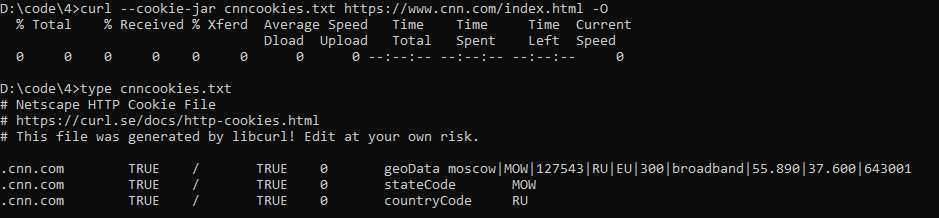


Рисунок 58 – Просмотр cookie файлов на компьютере

Данное действие было сделано еще на 4 URL. Выполним команду curl с опцией -L. Сравним вывод без этой опции, Рисунок 59-60. Объяснение: Опция -L позволяет следовать перенаправлениями, если они имеются. Без неё команда выводит только первоначальный ответ, а с ней – ответ после всех перенаправлений.

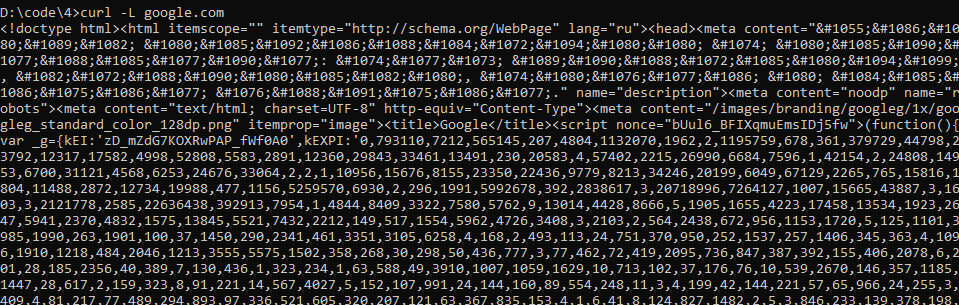


Рисунок 59 – Выполнение команды curl с опцией -L

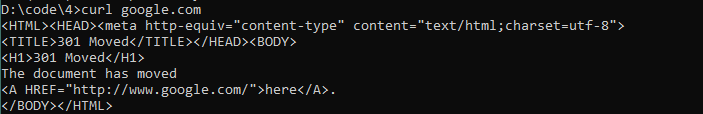


Рисунок 60 – Выполнение команды curl без опции -L

Получим заголовки сайтов 10 URL командой curl, опция -I, Рисунок 61.

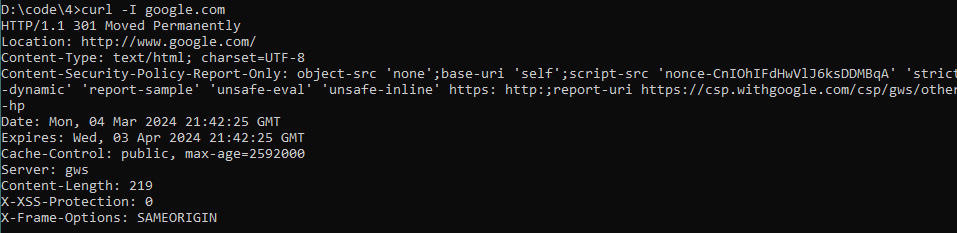


Рисунок 61 – Выполнение команды curl с опцией -I

Данное действие повторим еще 9 раз с другими URL. Составим POST-запросы используя команду curl, в виде списка имя=значение, и в виде JSON, для 3 URL, опция -d, Рисунок 62.

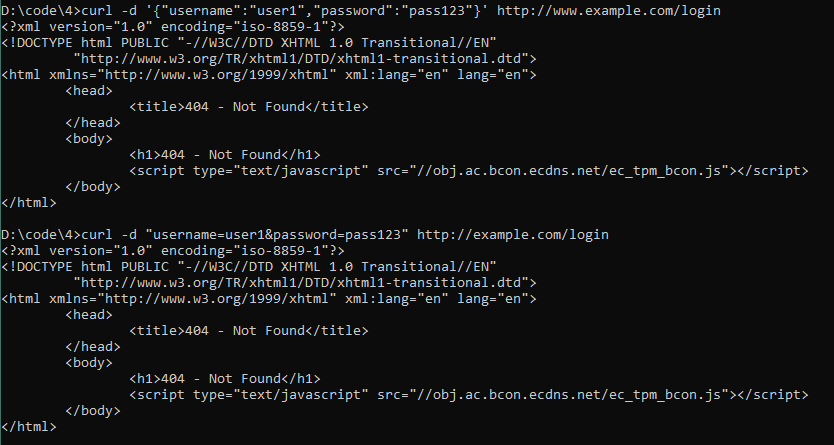


Рисунок 62 – Выполнение команды curl с POST запросом

Проделаем эти действия еще с 2 URL. Используем авторизацию для 3 URL, опция -u, Рисунок 63.

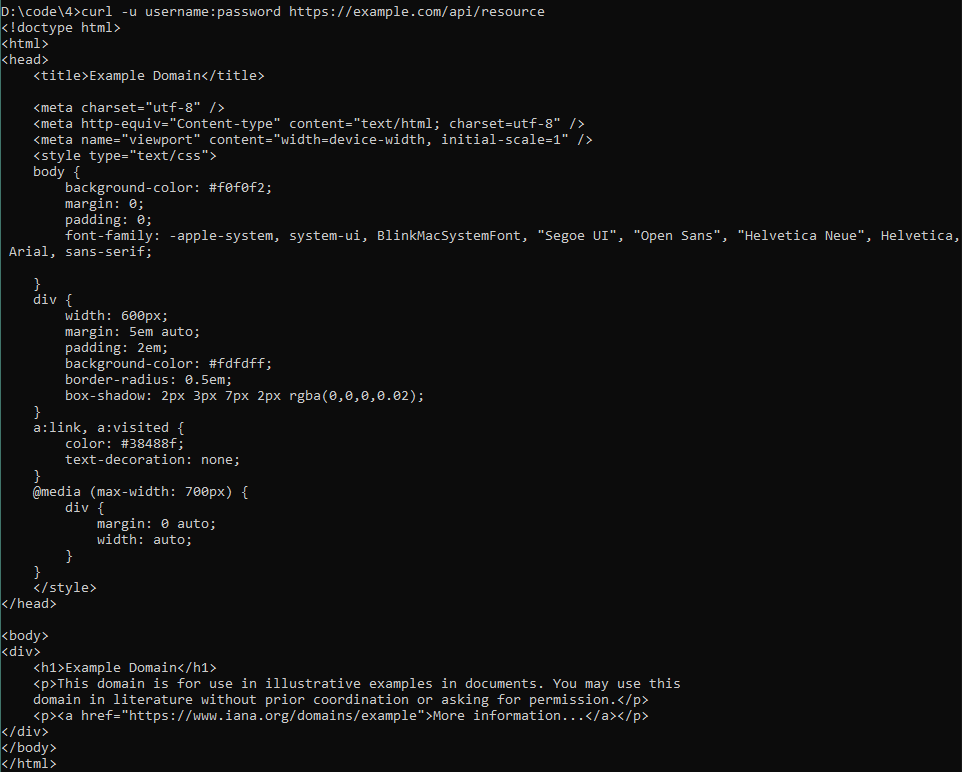


Рисунок 63 – Выполнение команды curl с опцией -u

Проделаем это действие еще с 2 URL. Выведем время работы команды для конкретного URL. Повторим команду 3 раза на одном URL, Рисунок 64. Выполним такую же операцию для 5 различных URL, опция -w, Рисунок 65.

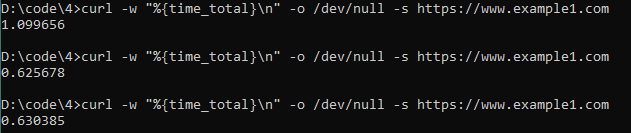


Рисунок 64 – Выполнение команды на вывод времени работы

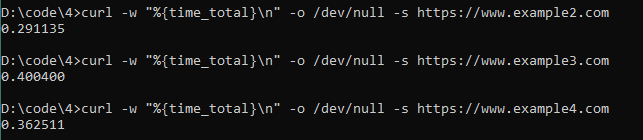


Рисунок 65 – Выполнение команды на вывод времени работы

Команда была проверена еще на 2 URL. Для 3 URL осуществить имитацию определенного браузера, опция -A, Рисунок 66.

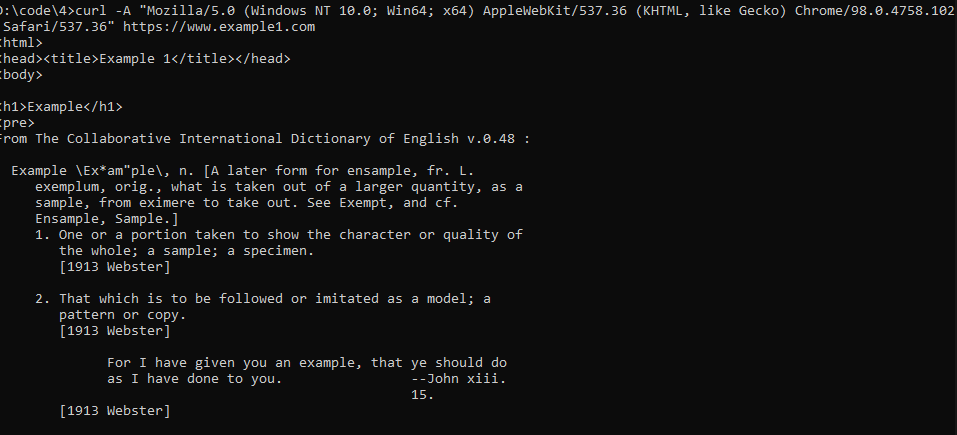


Рисунок 66 – Выполнение команды на имитацию определенного браузера

Проделаем эту команду еще на 2 URL. Отправим файлы cookie при скачивании дистрибутива, Рисунок 67. По умолчанию при запросе ресурса с помощью curl файлы cookie не отправляются и не сохраняются. Чтобы отправить файлы cookie на сервер, необходимо использовать опцию -b за которой следует имя файла, содержащего файлы cookie, или строку.

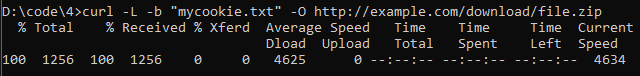


Рисунок 67 – Выполнение команды на отправку файлов cookie

Выполним команду по загрузке rpm-файла Oracle Java JDK jdk10.0.2\_linux-x64\_bin.rpm без отправки cookie, Рисунок 68.

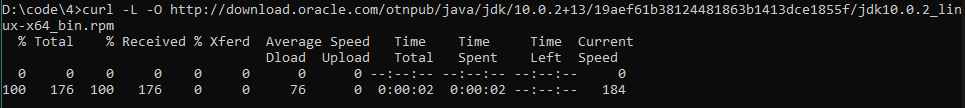


Рисунок 68 – Загрузка rpm-файла без отправки cookie

Теперь, поскольку вы загрузили тот же файл без отправки файла cookie, он, вероятно, будет идентичен по содержанию и размеру. Однако, если сервер Oracle использует механизмы аутентификации на основе файла cookie, отсутствие файла cookie может привести к различному поведению сервера и, возможно, к различиям в содержании загруженного файла.

Самостоятельно выберем ранее не использовавшиеся 5 опций команды curl, Рисунок 69-73.

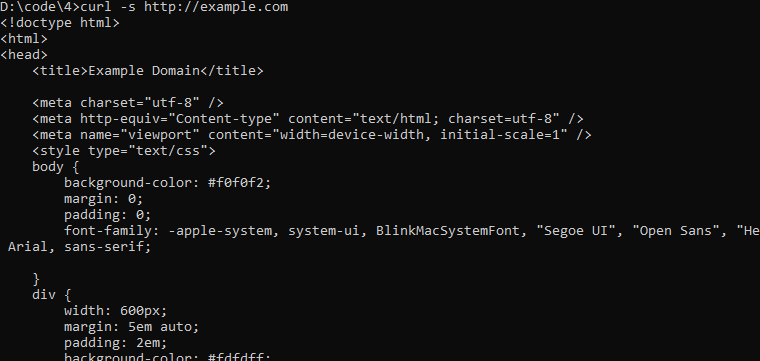


Рисунок 69 – Выполнение команды curl -s

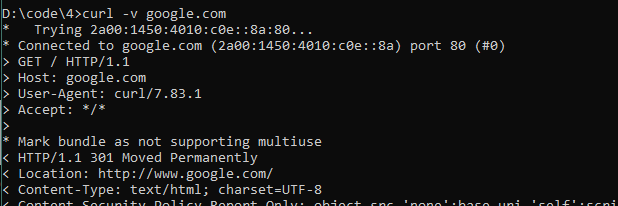


Рисунок 70 – Выполнение команды curl -v

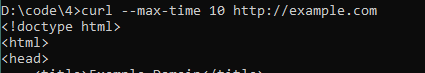


Рисунок 71 – Выполнение команды curl –max-time



Рисунок 72 – Выполнение команды curl –upload-file



Рисунок 73 – Выполнение команды curl -D

**Вывод:** в результате выполнения работы были получены навыки по запуску Linux-приложений в Windows 10, а также запуску Windows-приложений из ОС Linux.

ПРАКТИЧЕСКАЯ РАБОТА №4

**Цель работы:** получить навыки по запуску Linux-приложений в Windows 10, а также запуску Windows-приложений из ОС Linux.

**Задание на практическую работу:**

Ознакомиться и выполнить команды в командной строке Windows 10, а также повторить действия в WSL2 (в Linux. Для выполнения работы понадобится командный интерпретатор cmd. После запуска cmd с помощью команды cd переместиться в удобный каталог для работы (там, где удобнее хранить временные файлы, например, c:\temp, например, cd c:\temp. Все выполняемые команды проверить на работоспособность и привести в отчете. В cmd работает поиск по комбинации CTRL-F. Команда SSH (Windows и Linux)

Возможна такая ситуация, что ключи команды SSH в Windows и Linux могут отличаться.

Выполняя работу, предполагается, что работа ведется с файлами и директорией, содержащей готовый к загрузке на удаленный сервер созданный сайт.

Рекомендуется результат выполнения каждого задания сохранять в отдельную директорию (в заданиях, где копируются файлы и директории), например z1, z2 и т.д.

1. Проверить версию команды ssh

2. Подключиться к localhost (в среде Windows 10. Таким образом, находясь физически на этой машине (Windows 10) можно осуществить имитацию работы с удаленной машиной (localhost), которая также будет этой же машиной. Подключиться, можно указав localhost или как вариант ip-адрес машины 127.0.0.1 или адрес локальной машины, который можно получить, выполнив команду ipconfig и посмотрев IPv4-адрес (будет такого вида 192.168.0.45 или 192.168.145 или 10.0.1.45 или 10.1.1.45.

3. Используя команду tar –version проверьте версию архиватора tar. Используя команду tar /? посмотрите краткую справку. С помощью команды tar --help посмотрите подробную справку. Заархивируйте несколько файлов в архив. Разархивируйте полученный архив. Заархивируйте директорию. Разархивируйте полученный архив.

4. С помощью команды scp осуществить копирование трех файлов (каждый файл отдельно) через ssh c Windows 10 машины на localhost. Этой же командой и в этой же среде скопировать директорию. Скопируйте архив, а затем разархивируйте архив на новом месте.

5. Используя WSL2 (среду Linux), подключиться посредством SSH к машине Windows 10. Если SSH (клиент или сервер) в Linux не установлен, следует осуществить установку.

6. С помощью команды scp осуществить копирование трех файлов (каждый файл отдельно) через ssh из среды Linux на Windows 10 машину. Этой же командой и в этой же среде скопировать директорию. Скопируйте архив, а затем разархивируйте архив на новом месте.

7. С помощью команды sftp осуществить подключение к localhost (в нашем случае это будет remote host по отношению к нашей локальной машине. Ознакомиться с командами sftp, используя команду help. Вывести версию sftp командой version. С помощью команды get загрузить с remote host на localhost три файла. С помощью команды put отправить с локальной машины на remote host три файла. Убедиться по всем командам, что копирование прошло корректно. В sftp доступна команда ls.

8. Осуществить подключение из WSL2 (Linux), используя команду sftp к Windows 10. Выполнить sftp-команды put и get (переслать три файла) по направлению Windows 10.

9. Осуществить подключение из Windows 10 используя команду sftp к WSL2 (Linux. Выполнить sftp-команды put и get (переслать три файла) по направлению WSL2 (Linux)

10. Самостоятельно выбрать ранее не использовавшиеся 2 опции каждой из команд scp, sftp, tar. Выполнить команды в среде Linux и Windows 10 с этими опциями и объяснить полученный результат.

11. Попробуйте скопировать файл со следующего ресурса (или любого другого удаленного доступного для работы ресурса) командами scp, sftp. ftp://ftp.hp.com:21/pub/softpaq/sp40001/sp40001.cva Объясните полученный результат по отношению к команде curl. Попробуйте скопировать этот же файл командой ftp (в среде Linux и Windows 10)

**Выполнение задания**

Проверим версию команды ssh, Рисунок 74.



Рисунок 74 – Проверка команды ssh

Подключимся к localhost (в среде Windows 10), Рисунок 75.



Рисунок 75 – Подключение к localhost

Используя команду tar –version проверим версию архиватора tar, через команду tar /? посмотрим краткую справку. С помощью команды tar --help посмотрим подробную справку. Далее заархивируем несколько файлов в архив и разархивируем полученный архив, заархивируем директорию и разархивируем полученный архив, Рисунок 76-77.

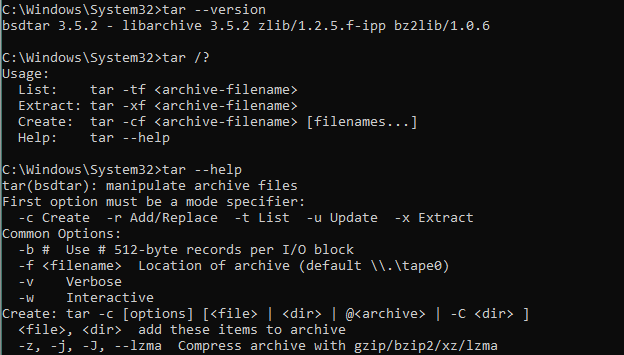


Рисунок 76 – Выполнение команды tar

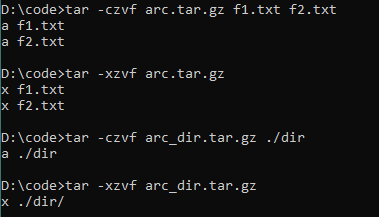


Рисунок 77 – Выполнение команды tar

С помощью команды scp осуществим копирование трех файлов (каждый файл отдельно) через ssh c localhost на Windows 10 машину. Этой же командой и в этой же среде скопируем директорию. Скопируем архив, а затем разархивируем его на новом месте, Рисунок 78-79.

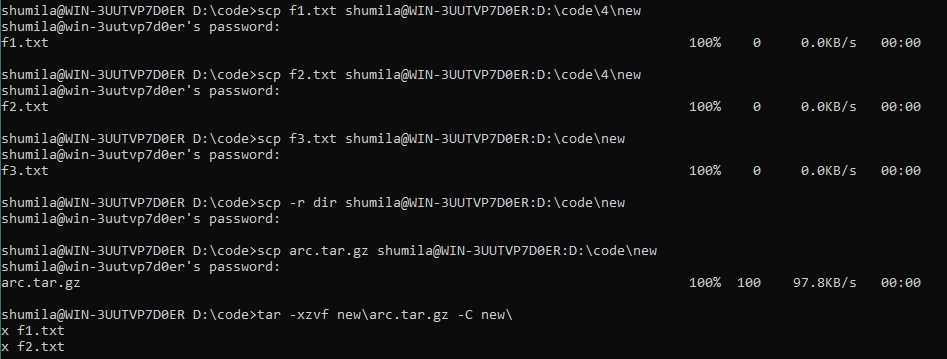


Рисунок 78 – Выполнение команды scp

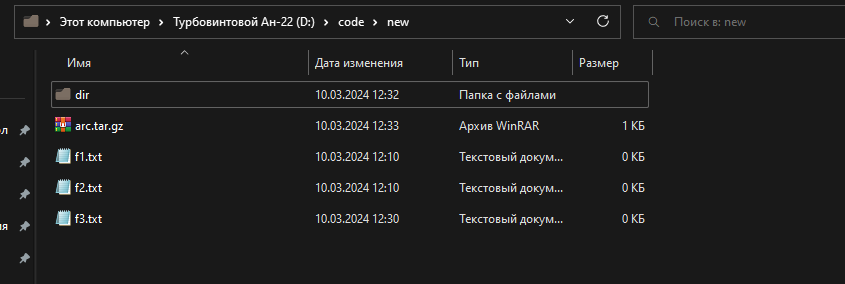


Рисунок 79 – Проверка выполнения команды scp

Используя WSL2 (среду Linux), подключимся посредством SSH к машине Windows 10, Рисунок 80.



Рисунок 80 – Подключение по ssh в WSL к Win10

С помощью команды scp осуществим копирование трех файлов (каждый файл отдельно) через ssh из среды Linux на Windows 10 машину. Этой же командой и в этой же среде скопируем директорию. Скопируем архив, а затем разархивируем архив на новом месте, Рисунок 81-83.

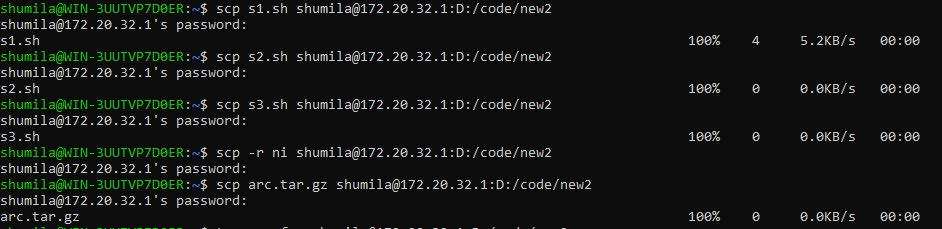


Рисунок 81 – Выполнение scp на WSL в Win10



Рисунок 82 – Выполнение scp на WSL в Win10

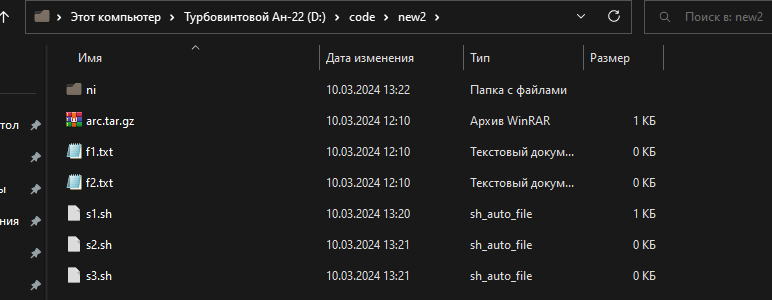


Рисунок 83 – Проверка выполнения scp на WSL в Win10

С помощью команды sftp осуществим подключение к localhost (в нашем случае это будет remote host по отношению к нашей локальной машине. Ознакомимся с командами sftp, используя команду help. Выведем версию sftp командой version. С помощью команды get загрузим с remote host на localhost три файла. С помощью команды put отправим с локальной машины на remote host три файла, Рисунок 84-85.

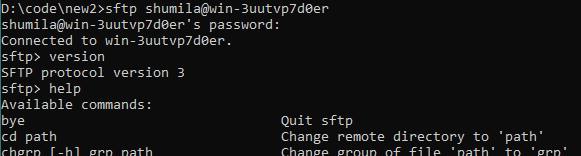


Рисунок 84 – Выполнение команды sftp

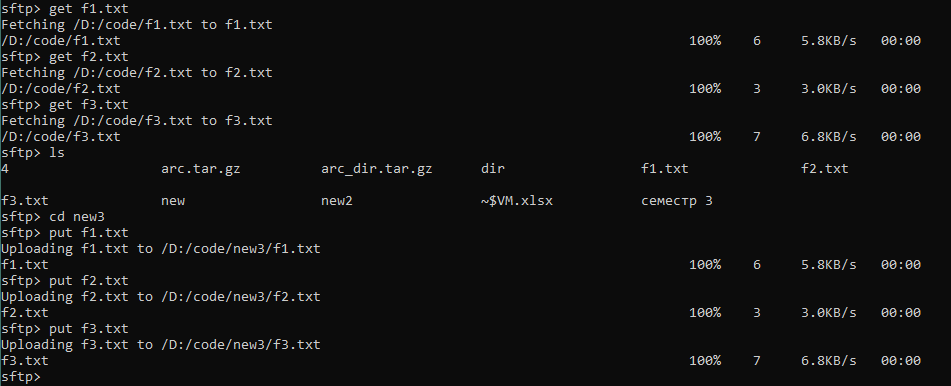


Рисунок 85 – Выполнение команды sftp

Осуществим подключение из WSL2 (Linux), используя команду sftp к Windows 10. Выполним sftp-команды put и get (переслать три файла) по направлению Windows 10, Рисунок 86-87.

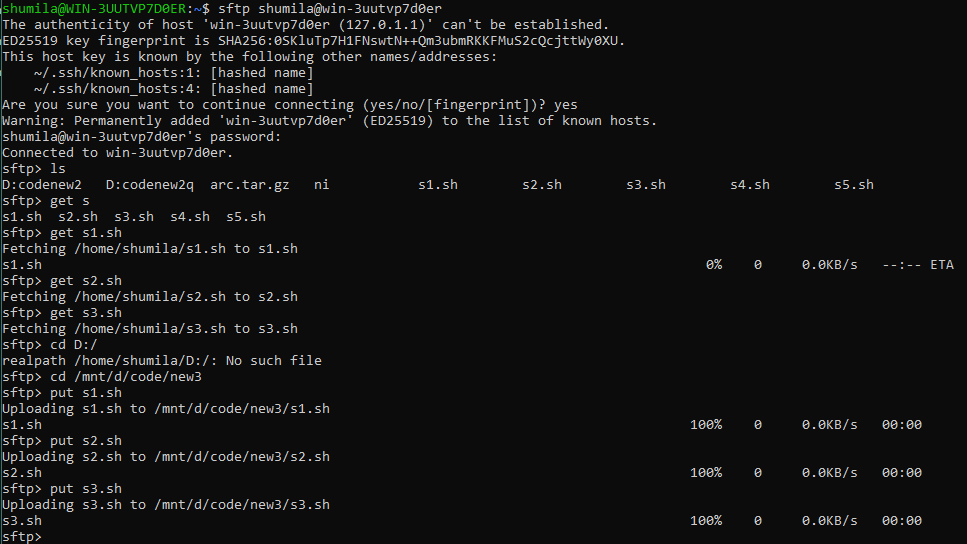


Рисунок 86 – Выполнение команд get и put через sftp

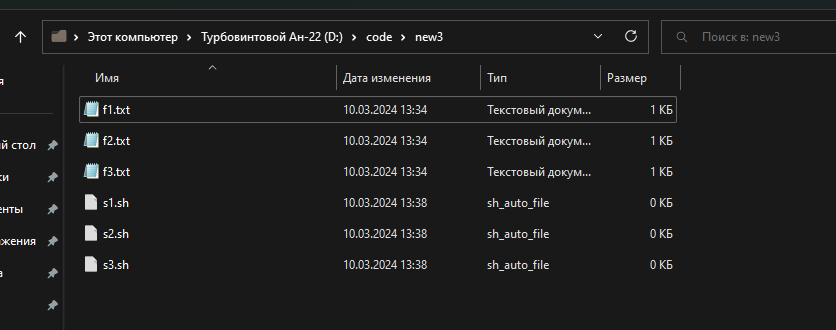


Рисунок 87 – Проверка выполнения команды get

Осуществим подключение из Windows 10 используя команду sftp к WSL2 (Linux. Выполним sftp-команды put и get (переслать три файла) по направлению WSL2 (Linux), Рисунок 88.



Рисунок 88 – Выполнение команд get и put через sftp

Выберем ранее не использовавшиеся 2 опции каждой из команд scp, sftp, tar. Выполним команды в среде Linux и Windows 10 с этими опциями, Рисунок 89-91.

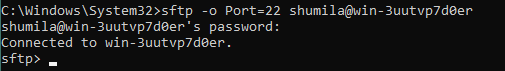


Рисунок 89 – Выполнение команд

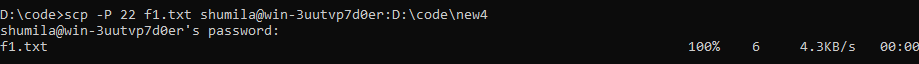


Рисунок 90 – Выполнение команды



Рисунок 91 – Выполнение команды

Попробуем скопировать файл со следующего ресурса (ftp://ftp.hp.com:21/pub/softpaq/sp40001/sp40001.cva) командами scp, sftp. Объясните полученный результат по отношению к команде curl. Попробуйте скопировать этот же файл командой ftp (в среде Linux и Windows 10), Рисунок 92.

Команды scp и sftp предназначены для копирования файлов по протоколу SSH (Secure Shell), который обычно работает на порте 22. Однако, файл находится на FTP-сервере, который использует протокол FTP (File Transfer Protocol), работающий на порту 21. Попытка использовать scp или sftp для копирования файла с FTP-сервера не удастся, так как эти команды предназначены для работы с SSH, а не с FTP.



Рисунок 92 – Получение файла с ресурса

**Вывод**

В результате выполнения работы были получены навыки по запуску Linux-приложений в Windows 10, а также запуску Windows-приложений из ОС Linux. Разница между командами:

Curl - команда используется для передачи данных по сети с использованием различных протоколов, таких как HTTP, HTTPS, FTP и многих других.

Scp - команда используется для копирования файлов между локальной машиной и удаленным сервером через защищенное соединение SSH.

Sftp - команда также используется для передачи файлов между локальной машиной и удаленным сервером, но она работает через протокол SFTP.

Ftp - команда используется для передачи файлов между локальной машиной и удаленным сервером через протокол FTP. В отличие от SCP и SFTP, FTP не обеспечивает шифрование данных, поэтому он менее безопасен для передачи конфиденциальной информации.