Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«ТОМСКИЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И

РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

Кафедра комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС)

LINUX, DOCKER, ОКРУЖЕНИЕ ДЛЯ РАЗРАБОТКИ

Отчет по лабораторной работе №1

по дисциплине «Системное программирование»

Студент гр. 718-1

\_\_\_\_\_\_\_\_ Прозорова Е. А.

«\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022

Принял

М. н. с. ИСИБ

\_\_\_\_\_\_\_\_ Калинин Е. О.

«\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022

Томск 2022

1 Введение

Цель работы: ознакомление с основными командами ОС Unix/Linux, возможностями Docker для контейнеризации окружения программного обеспечения и его зависимостей, средств разработки и сборки ПО и создание Dockerfile.

Вариант: 17.

2 Ход работы

По заданию практической работы были выполнены команды для обновления пакетов и обновления индекса пакетов (рисунок 2.1).

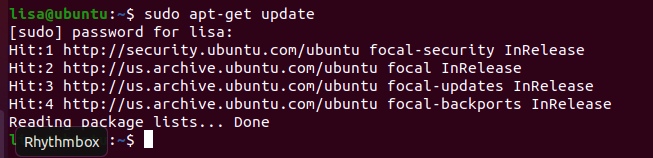


Рисунок 2.1 – Обновление индекса пакетов

После были установлены пакеты, которые позволяют использовать пакеты по HTTPS (рисунок 2.2).

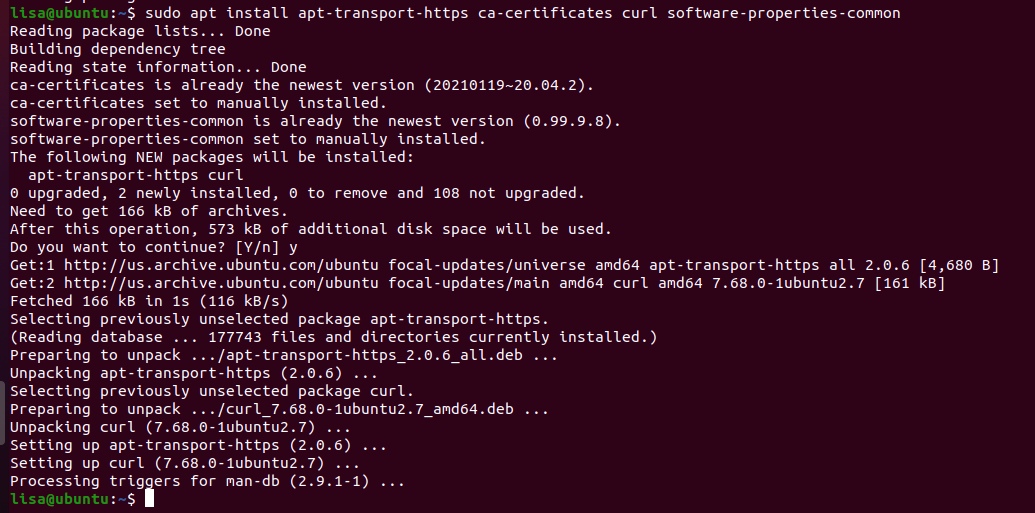


Рисунок 2.2 – Установка пакетов

Далее был добавлен ключ GPG официального репозитория Docker (рисунок 2.3)

https://sun9-69.userapi.com/impf/tXAFJcDE1lbUBGtCeb0d4sFQcSc9Zn4bhyfFLw/X1zH-pWKpCY.jpg?size=837x60&quality=96&sign=eddb721a708da1e562871d4264f9d605&type=album

Рисунок 2.3 – Добавление ключа GPG

Также был добавлен репозиторий Docker в список источников пакетов APT (рисунок 2.4).

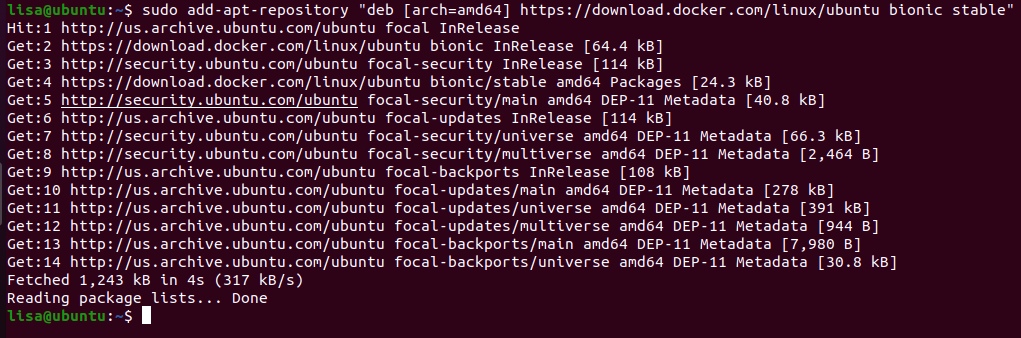


Рисунок 2.4 – Добавление Docker-репозитория

По заданию лабораторной работы была обновлена база данных пакетов информацией о пакетах Docker из вновь добавленного репозитория (рисунок 2.5).

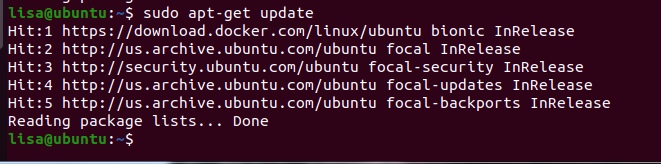


Рисунок 2.5 – Обновление базы данных пакетов

После была проведена проверка, что Docker установлен из репозитория Docker, а не из репозитория по умолчанию Ubuntu (рисунок 2.6).

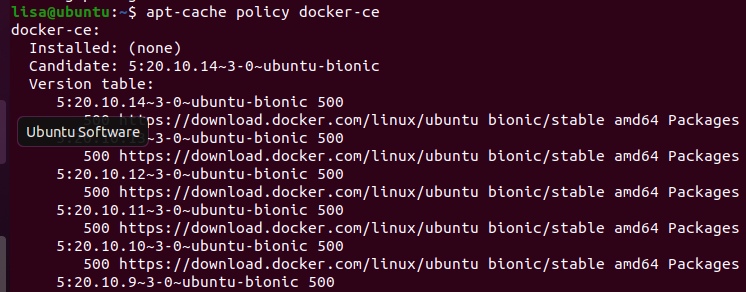


Рисунок 2.6 – Проверка репозитория

Далее был установлен Docker (рисунок 2.7).

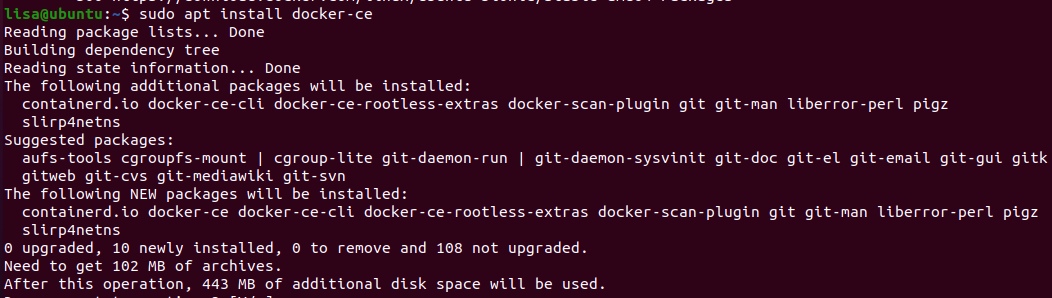


Рисунок 2.7 – Установка Docker

Также была выполнена проверка наличия процесса Docker (рисунок 2.8).

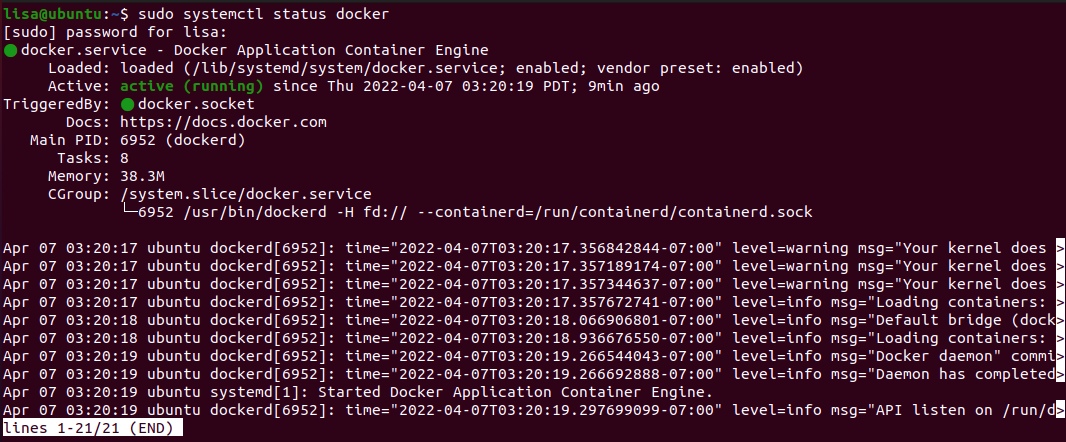


Рисунок 2.8 – Проверка работы Docker

По заданию лабораторной работы пользователь был добавлен в группу Docker (рисунок 2.9).

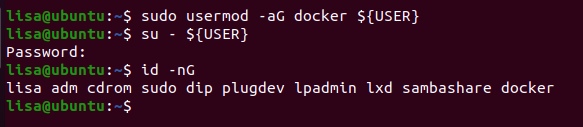


Рисунок 2.9 – Добавление пользователя в группу docker

После был скачен образ ubuntu (рисунок 2.10).

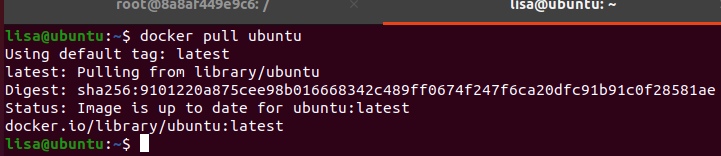


Рисунок 2.10 – Скачивание образа

Далее был просмотрен список скаченных образов (рисунок 2.11).

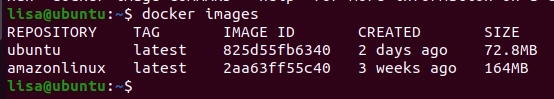


Рисунок 2.11 – Список скаченных образов

Также была проверена возможность запуска образа ubuntu (рисунок 2.12).

https://sun9-19.userapi.com/impf/V93jcv-sIoSLqB5mUi6uB46uVDoaHaM8774dBQ/GhnRqMdU7kI.jpg?size=434x51&quality=96&sign=8e03d8081883af3aa797b47cc8ffb020&type=album

Рисунок 2.12 – Запуск контейнера

По заданию лабораторной работы был создан каталог, в котором будут хранится dockerfile и файл с скриптом (рисунок 2.13).



Рисунок 2.13 – Создание каталога

После этого был написан скрипт по варианту и написан dockerfile (рисунки 2.14 – 2.15). Код программы представлен в приложении А.

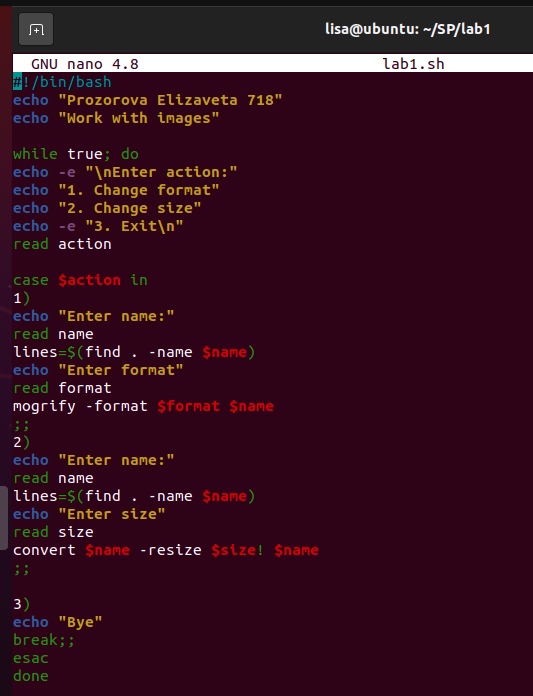


Рисунок 2.14 – Написанный скрипт

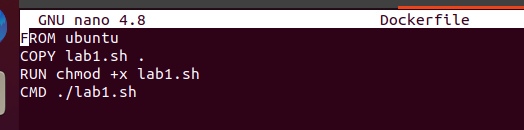


Рисунок 2.15 – Написанный Dockerfile

Далее был собран образ и запущен контейнер (рисунки 2.16 – 2.17).

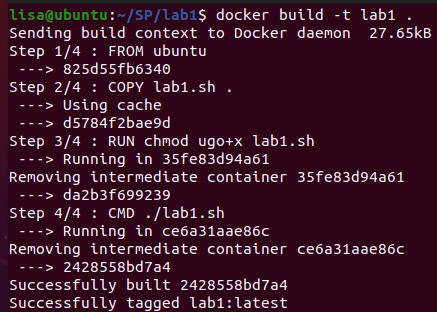


Рисунок 2.16 – Сборка образа

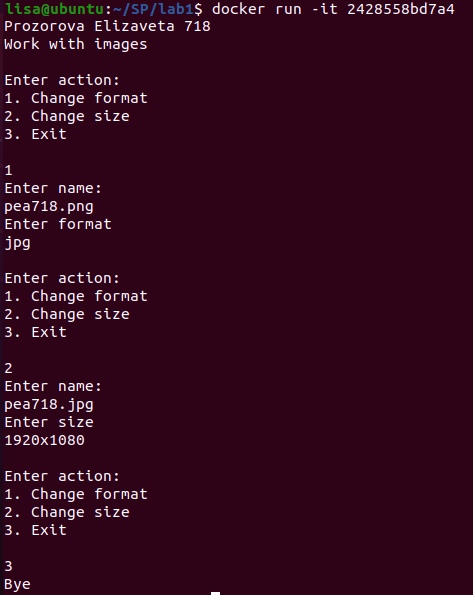


Рисунок 2.17 – Запуск контейнера

Также на github был создан репозиторий (рисунок 2.18).

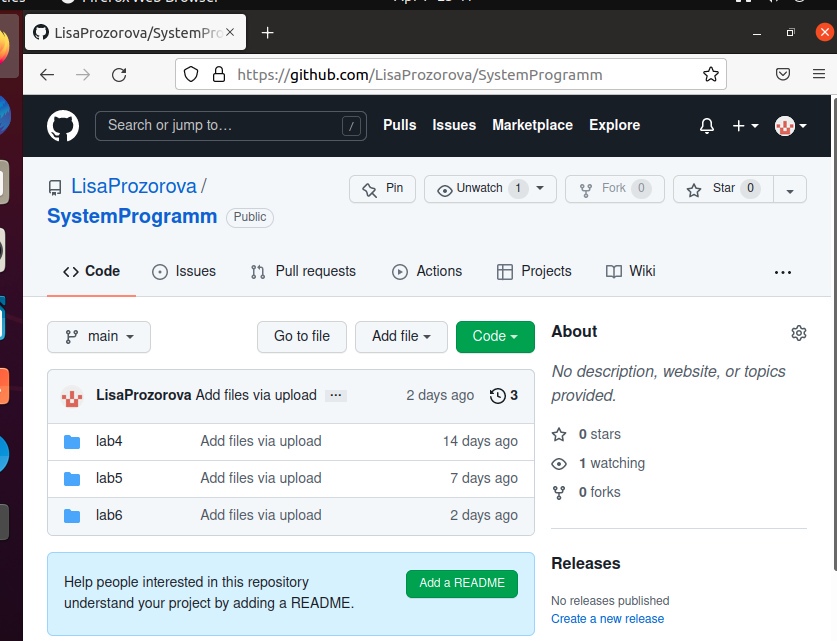


Рисунок 2.18 – Репозиторий github

Далее были подготовлены файлы, которые после были загружены на github (рисунки 2.19 – 2.20).

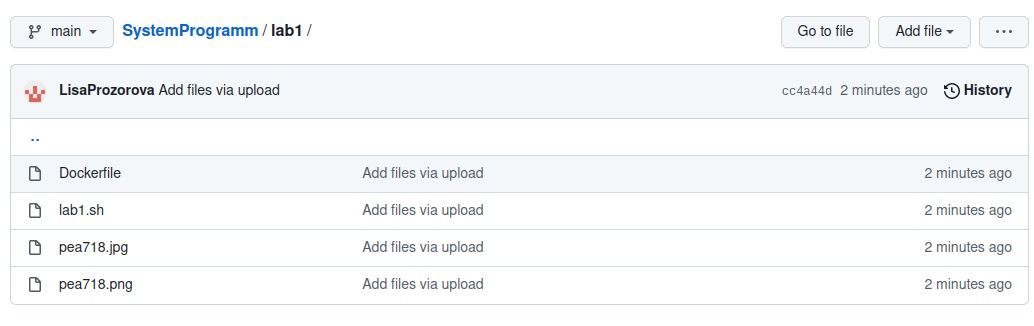


Рисунок 2.19 – Файлы на github

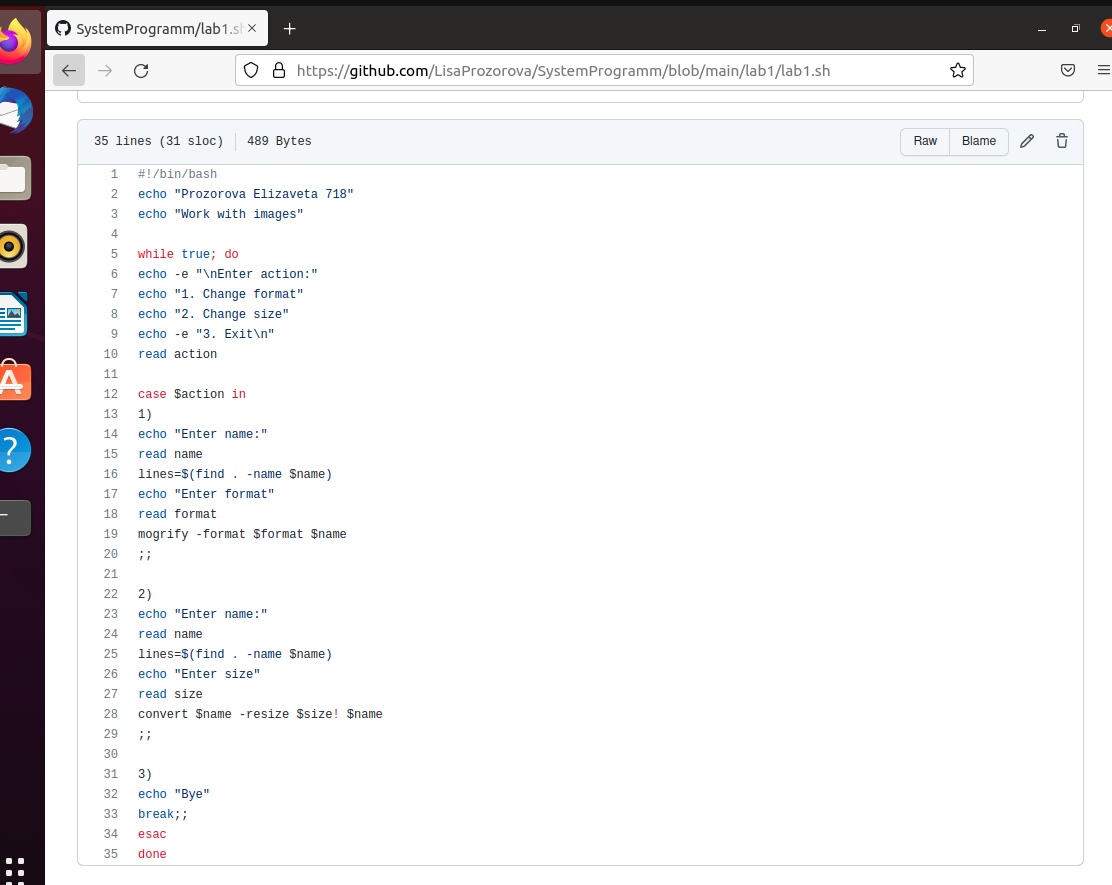


Рисунок 2.20 – Код основной программы на github

3 Заключение

По заданию лабораторной работы была изучены основные команды ОС Unix/Linux, возможности Docker для контейнеризации окружения программного обеспечения и его зависимостей, средств разработки и сборки ПО и был создан Dockerfile.

Приложение А

(Код программы)

#!/bin/bash

echo "Prozorova Elizaveta 718"

echo "Work with images"

while true; do

echo -e "\nEnter action:"

echo "1. Change format"

echo "2. Change size"

echo -e "3. Exit\n"

read action

case $action in

1)

echo "Enter name:"

read name

lines=$(find . -name $name)

echo "Enter format"

read format

mogrify -format $format $name

;;

2)

echo "Enter name:"

read name

lines=$(find . -name $name)

echo "Enter size"

read size

convert $name -resize $size! $name

;;

3)

echo "Bye"

break;;

esac

done

Приложение Б

(Ссылка на репозиторий)

Ссылки на репозиторий gitlab:

https://github.com/LisaProzorova/SystemProgramm