Лабораторная работа №1

Дисциплина: Администрирование сетевых подсистем

Жибицкая Евгения Дмитриевна

Содержание

# 1 Цель работы

Приобретение навыков установки Rocky Linux на виртуальную машину с помощью Vagrant.

# 2 Выполнение лабораторной работы

Перед непосредственным развертыванием машины необходимо подготовить пространство, установить различные программы и дополнения. Установим последнюю версию Vagrant, образ диска для Rocky, также нам понадобится Far и Packer(рис. 1).

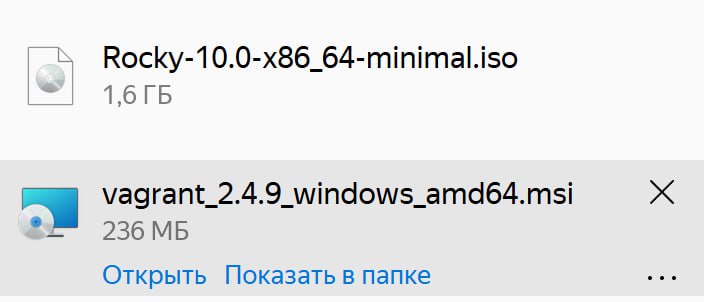


Рис. 1: Установка необходимых программ

Далее приступим к созданию рабочей структуры. Создадим каталоги для работы и разместим нужные подкаталоги: work - edzhibitskaya - packer и vagrant(рис. 2).

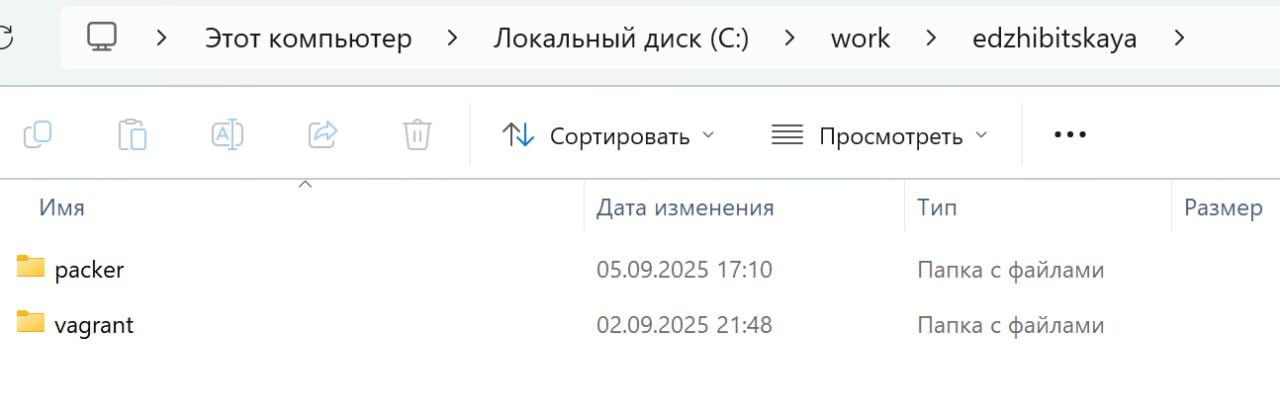


Рис. 2: Структура каталогов

В каталог packer добавляем образ диска, файл vagrant-rocky.pkr.hcl и подкаталог http с файлом ks.cfg, который определяет настройки для установки дистрибутива, которые пользователь обычно вводит вручную. Такжде добавляем туда packer.exe (рис. 3).

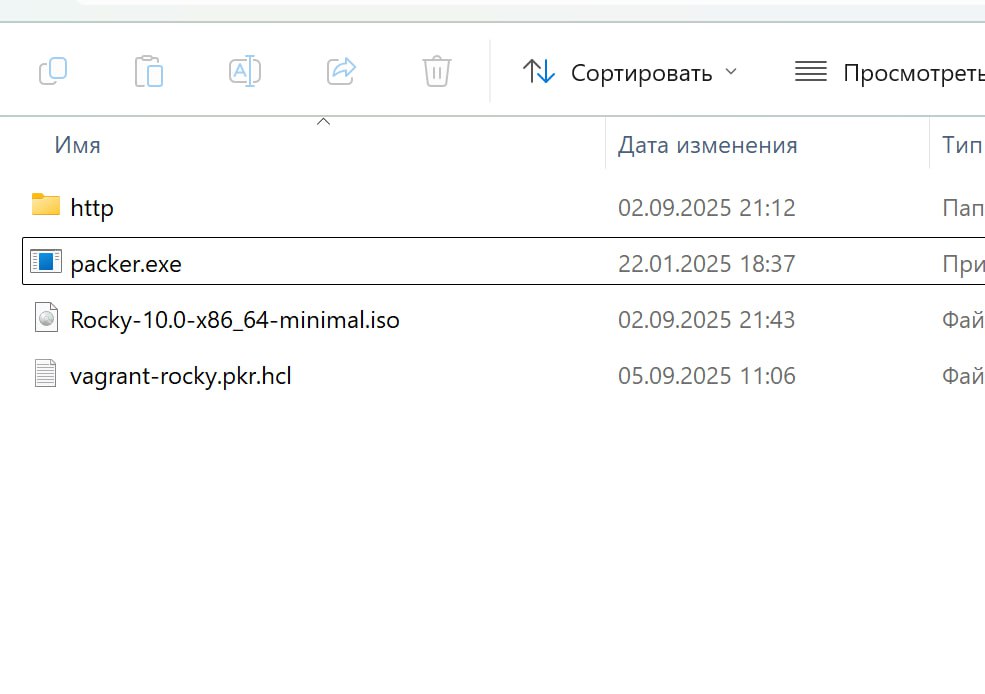


Рис. 3: Структура packer

В каталоге vagrant размещаем vagrantfile, создаем каталог provision с подкаталогами default, server и client, в которых будут размещаться скрипты, изменяющие настройки внутреннего окружения базового (общего) образа виртуальной машины, сервера или клиента соответственно. Размещаем там заранее подготовленный скриптзаглушку 01-dummy.sh(рис. 4).

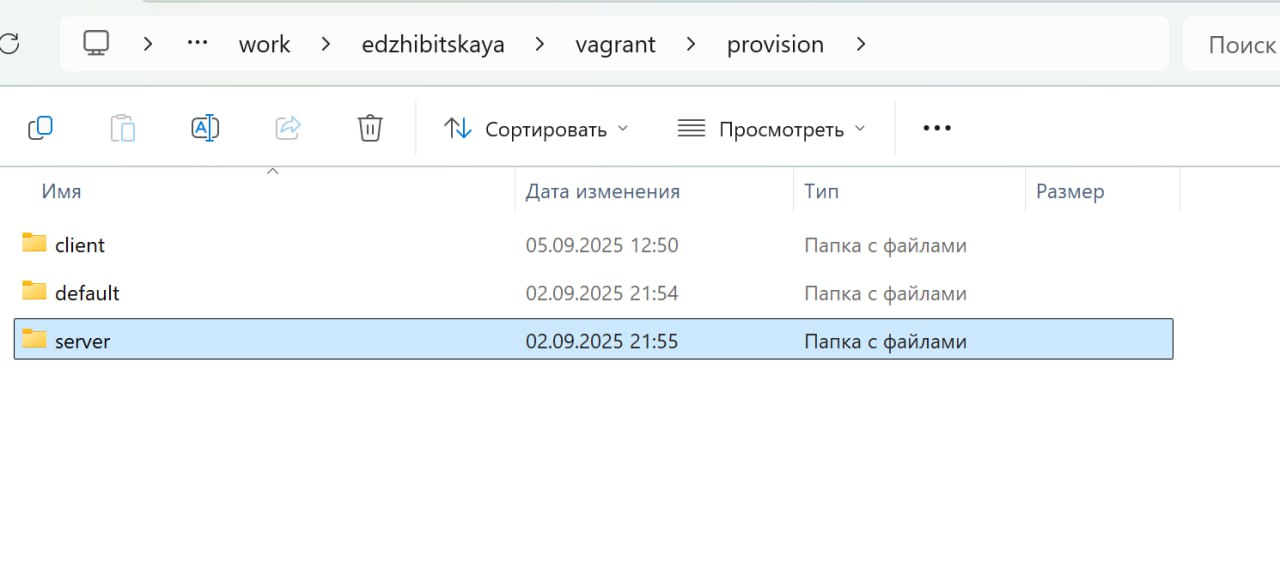


Рис. 4: Файлы vagrant

В созданных файлах размещаем данные нам скрипты, меняем в файле vagrant-rocky.pkr.hcl значение чек-суммы, версию Rocky(рис. 5).

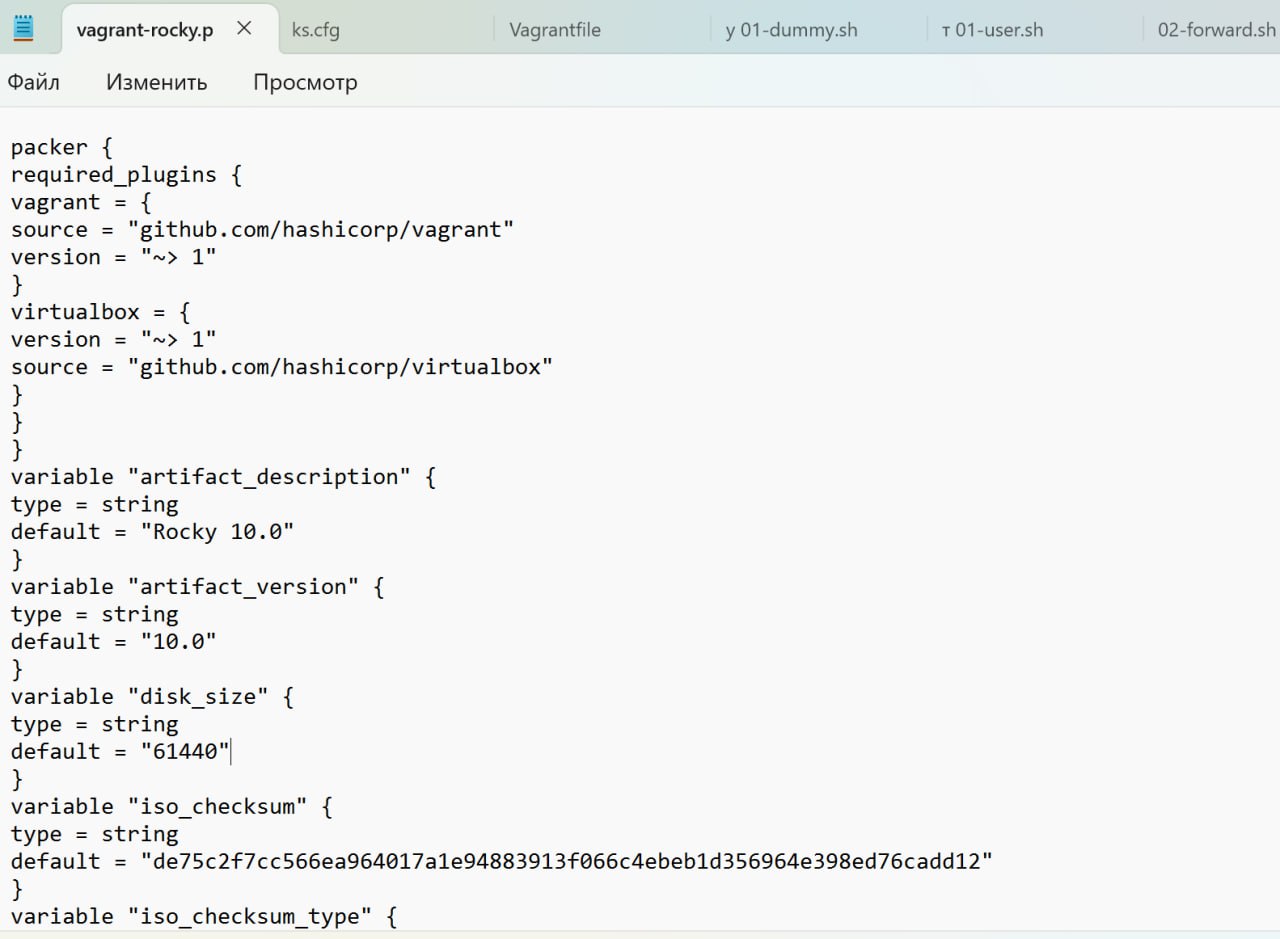


Рис. 5: Скрипты

Далее переходим к непосредственно развертыванию лабораторного стенда. Открываем терминал и с помощью far или же просто командной строки переходим в каталог, создаем машину используя команды

packer.exe init vagrant-rocky.pkr.hcl packer.exe build vagrant-rocky.pkr.hcl

Запускается установка образа операционной системы(рис. 6) и (рис. 7).

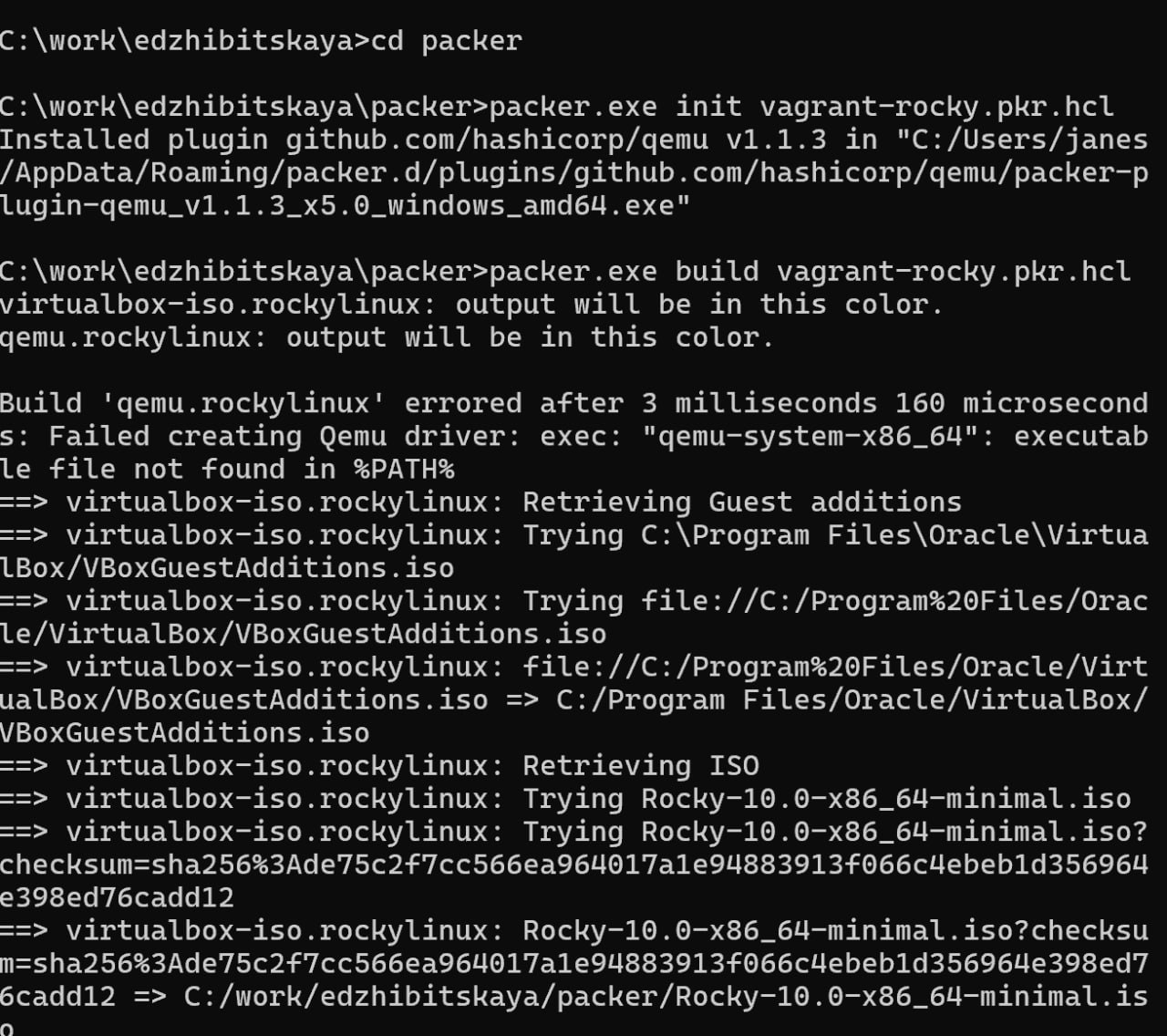


Рис. 6: Установка OC. Терминал

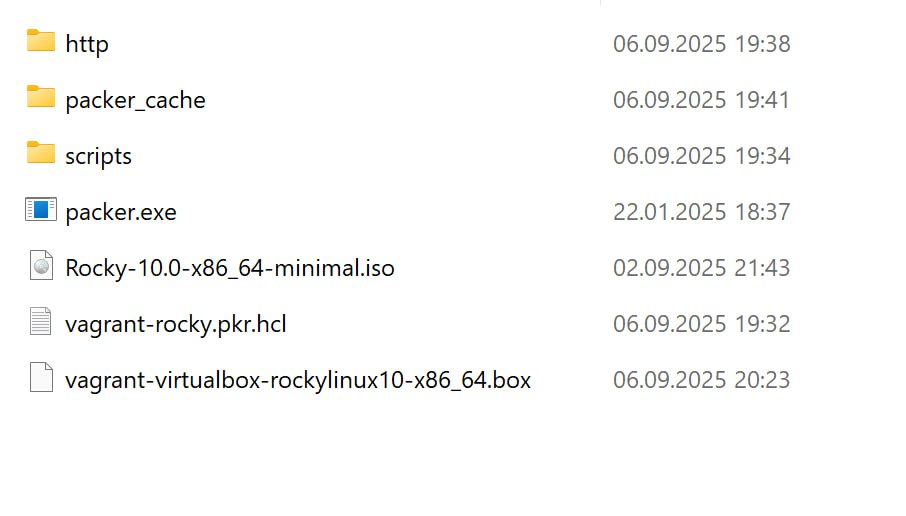


Рис. 7: Установка OC

Для регистрации в Vagrant вводим команду vagrant box add rocky10 vagrant-virtualbox-rocky-10-x86\_64.box(рис. 8).

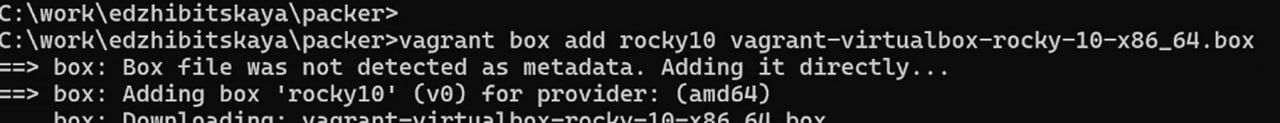


Рис. 8: Регистрация в vagrant

Далее запускаем ВМ Сервер и Клиент(рис. 9, рис. 10), проверяем, что есть возможность перейти к нашему созданному ранее пользователю и выключаем машину(рис. 11).Для корректной отработки скриптов убеждаемся, что необходимый код есть в Vagrantfile(рис. 12). Еще раз логинимся и смотрим, что скрипты отрабатываются(рис. 13, рис. 14),. Предварительно фиксируем изменения командами vagrant up server/сclient –provision(рис. 15).

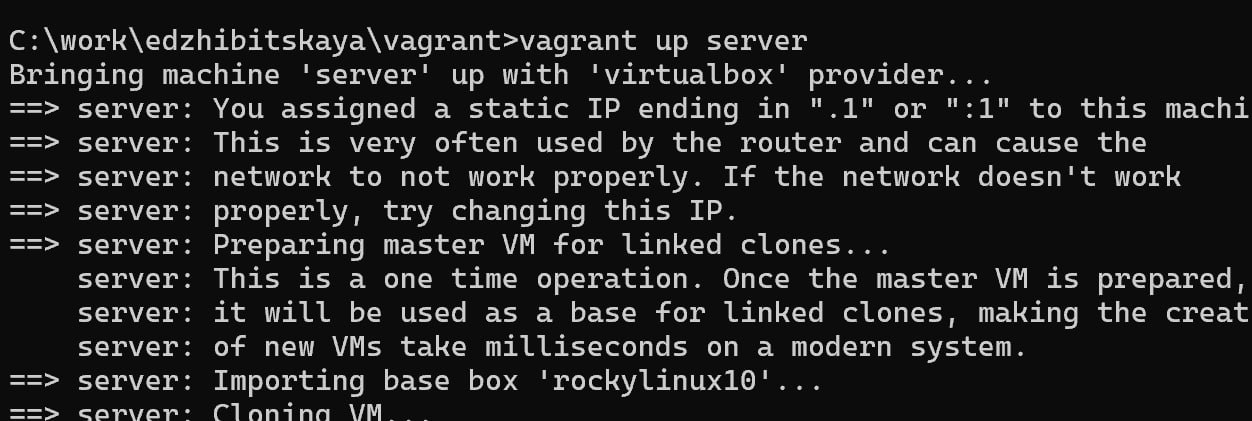


Рис. 9: Запуск

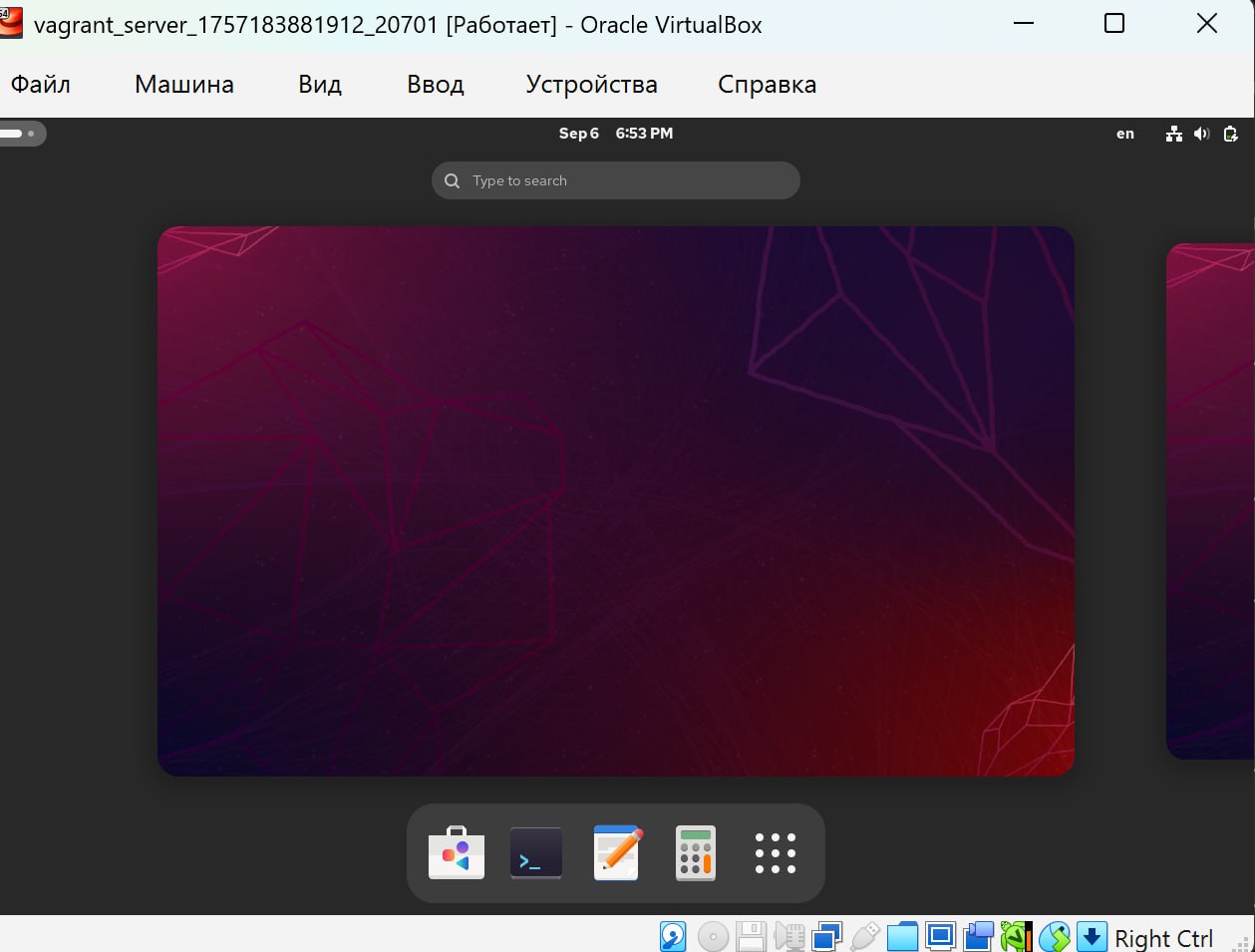


Рис. 10: Запуск

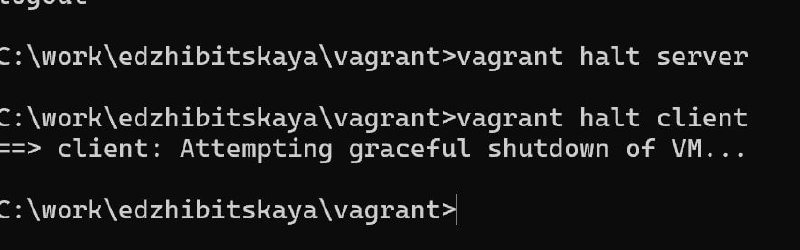


Рис. 11: Завершение работы

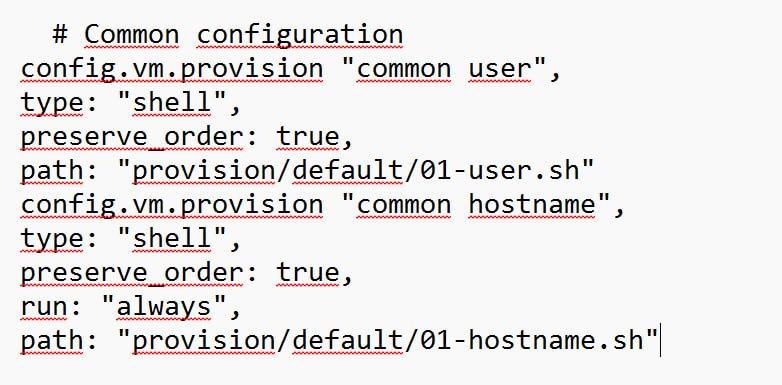


Рис. 12: Проверка скрипта

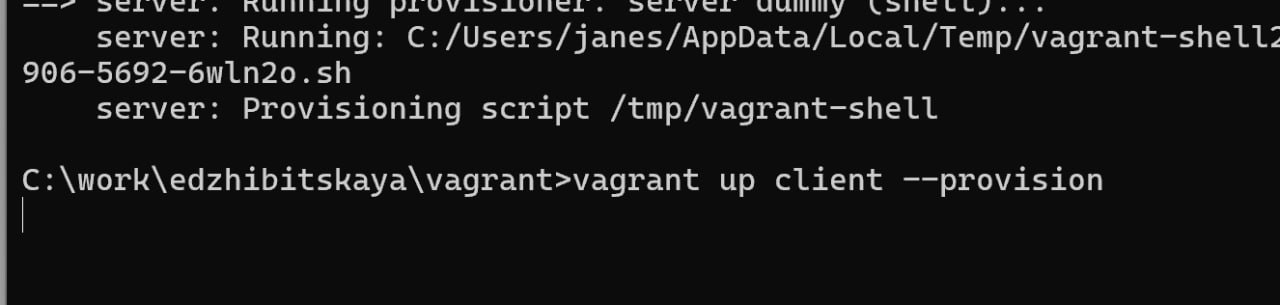


Рис. 13: Проверка

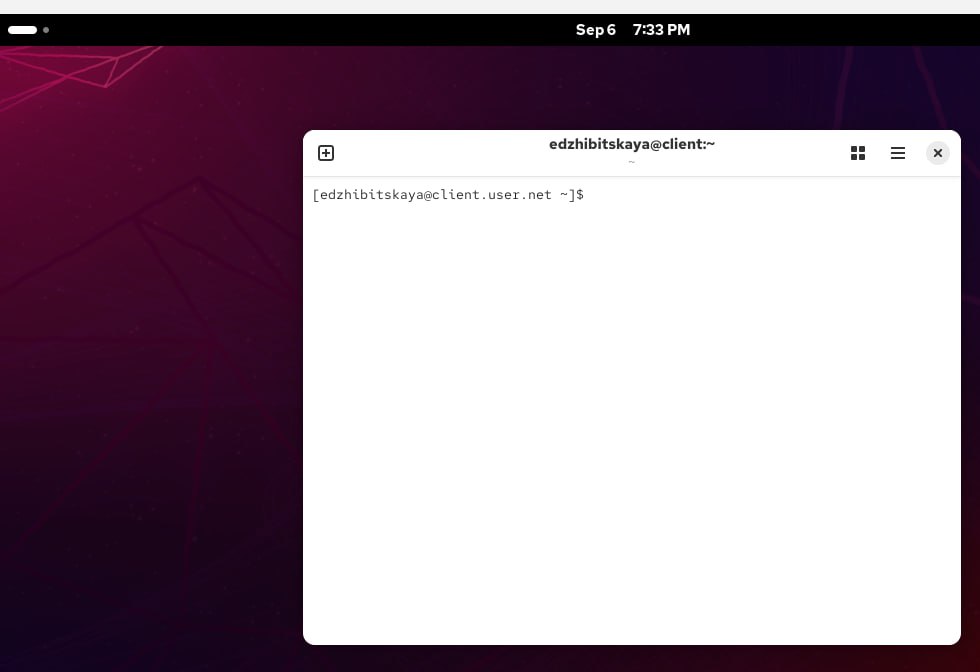


Рис. 14: Проверка

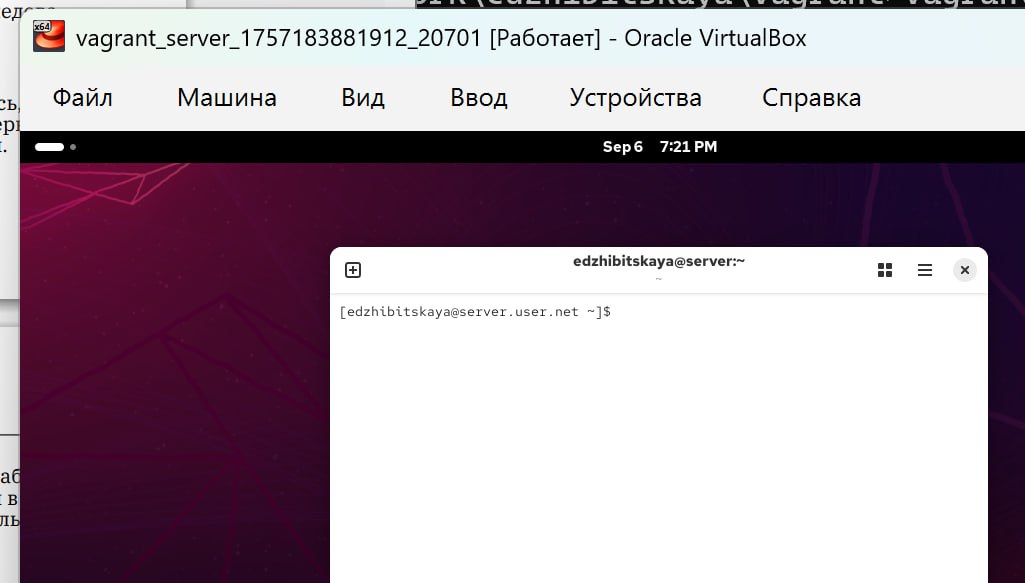


Рис. 15: vagrant up server/сclient –provision

# 3 Контрольные вопросы

1. Для чего предназначен Vagrant?

Он предназначен для создания и управления средами виртуальных машин в одном процессе.

1. Что такое box-файл? В чём назначение Vagrantfile?

* box-файл (или Vagrant Box) — сохранённый образ виртуальной машины с развёрнутой в ней операционной системой; по сути, box-файл используется как основа для клонирования виртуальных машин с теми или иными настройками;
* Vagrantfile — конфигурационный файл, написанный на языке Ruby, в котором указаны настройки запуска виртуальной машины.

1. Приведите описание и примеры вызова основных команд Vagrant.

– vagrant help — вызов справки по командам Vagrant;

– vagrant box list — список подключённых к Vagrant box-файлов;

– vagrant box add — подключение box-файла к Vagrant;

– vagrant destroy — отключение box-файла от Vagrant и удаление его из виртуального окружения;

– vagrant init — создание «шаблонного» конфигурационного файла Vagrantfile для его последующего изменения;

– vagrant up — запуск виртуальной машины с использованием инструкций по запуску из конфигурационного файла Vagrantfile;

– vagrant reload — перезагрузка виртуальной машины;

– vagrant halt — остановка и выключение виртуальной машины;

– vagrant provision — настройка внутреннего окружения имеющейся виртуальной машины (например, добавление новых инструкций (скриптов) ранее созданную виртуальную машину);

– vagrant ssh — подключение к виртуальной машине через ssh.

# 4 Выводы

В ходе работы были получены навыки по работе с Vagrant и установке с его помощью Rocky Linux на виртуальную машину.

# Список литературы

[ТУИС](https://esystem.rudn.ru/course/view.php?id=9060)