РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

**Факультет физико-математических и естественных наук**

**Кафедра теории вероятностей и кибербезопасности**

ОТЧЕТ

по лабораторной работе № 1

Дисциплина «Администрирование сетевых подсистем»

*Тема «Подготовка лабораторного стенда»*

Студент: Щербак Маргарита Романовна

Ст. билет: 1032216537

Группа: НПИбд-02-21

**МОСКВА**

2023 г.

# Цель работы

Приобрести практические навыки установки Rocky Linux на виртуальную машину с помощью инструмента Vagrant.

# Задание

1. Сформировать box-файл с дистрибутивом Rocky Linux для VirtualBox.

2. Запустить виртуальные машины сервера и клиента, а также убедиться в их работоспособности.

3. Внести изменения в настройки загрузки образов виртуальных машин server и client, добавив пользователя с правами администратора и изменив названия хостов.

4. Скопировать необходимые для работы с Vagrant файлы и box-файлы виртуальных машин на внешний носитель.

# Выполнение работы

## 1. Формирование box-файла с дистрибутивом Rocky Linux для VirtualBox

В своей ОС установила последние версии Vagrant и VirtualBox. Для моей ОС Windows также понадобилось дополнительно установить Packer и FAR для удобства работы в терминале. Перед началом работы с Vagrant создала каталог \work\mrshcherbak для проекта, далее две папки: C:\work\ mrshcherbak\packer и C:\work\ mrshcherbak\vagrant (рис.1.1). В созданном рабочем каталоге разместила образ варианта операционной системы Rocky Linux: Rocky-9.2- x86\_64-minimal.iso — минимальный дистрибутив Rocky Linux. Разместила подготовленные заранее для работы с Vagrant файлы (рис.1.2 – рис.1.3):

– vagrant-rocky.pkr.hcl — специальный файл с описанием метаданных по установке дистрибутива на виртуальную машину;

– ks.cfg — определяет настройки для установки дистрибутива, которые пользователь обычно вводит вручную, в частности настройки языка интерфейса, языковые настройки клавиатуры, тайм-зону, сетевые настройки и т.п.; файл расположен в подкаталоге http;

– Vagrantfile — файл с конфигурацией запуска виртуальных машин — сервера и клиента.

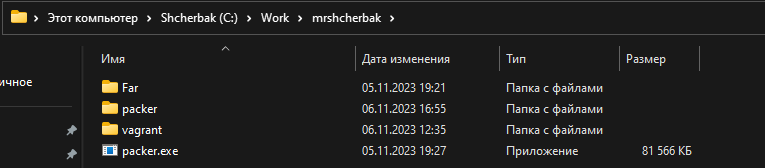


Рис.1.1. Создание каталогов

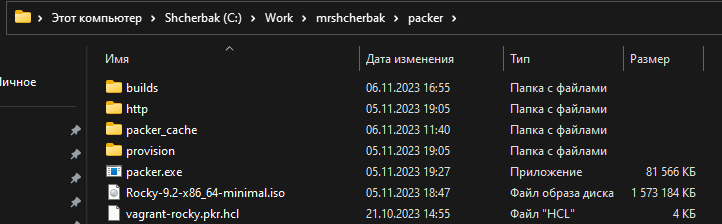


Рис.1.2. Файлы, необходимые для работы

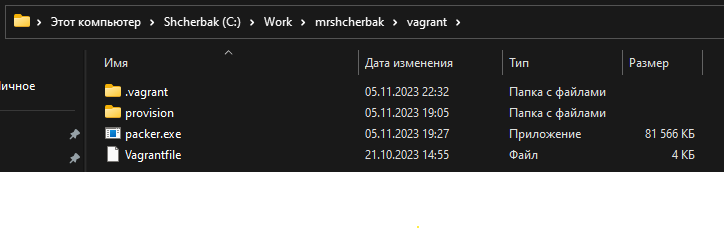


Рис.1.3. Файлы, необходимые для работы

Создала 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 (рис.1.4 – рис.1.5).

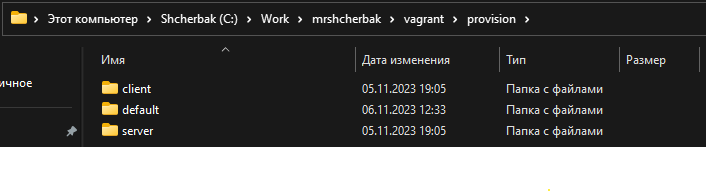


Рис.1.4. Создание каталогов

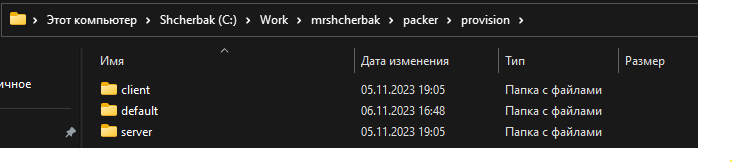


Рис.1.5. Создание каталогов

В каталогах default, server и client каждой папки разместила заранее подготовленный скрипт-заглушку 01-dummy.sh следующего содержания:

#!/bin/bash

echo "Provisioning script $0"

В каталоге default каждой папки разместила заранее подготовленный скрипт 01-user.sh по изменению названия виртуальной машины (рис.1.6).

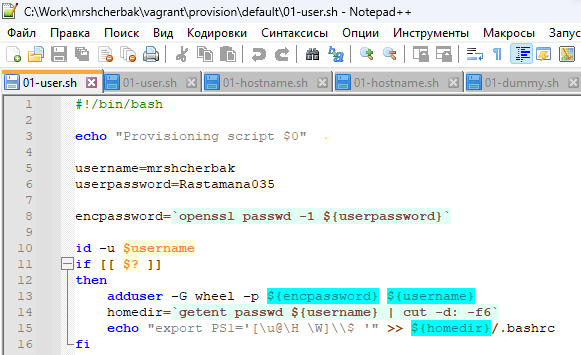


Рис.1.6. Содержимое файла 01-user.sh

В каталоге default каждой папки разместила заранее подготовленный скрипт 01-hostname.sh по изменению названия виртуальной машины (рис.1.7).

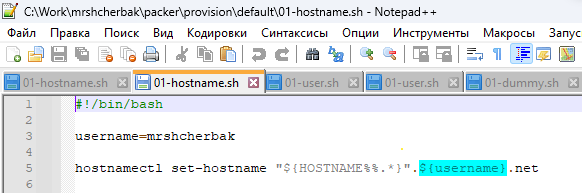


Рис.1.7. Содержимое файла 01-hostname.sh

## 2. Развёртывание лабораторного стенда на ОС Windows

Используя FAR, перешла в созданный рабочий каталог с проектом. В этом же каталоге размещён файл packer.exe. В командной строке ввела:

packer.exe init vagrant-rocky.pkr.hcl

packer.exe build vagrant-rocky.pkr.hcl

Так начнется автоматическая установка образа операционной системы Rocky Linux в VirtualBox и последующее формирование box-файла с дистрибутивом Rocky Linux для VirtualBox (рис.2.1 – рис.2.2). По окончании процесса в рабочем каталоге сформируется box-файл с названием vagrant-virtualbox-rocky-9-x86\_64.box (рис.2.3).

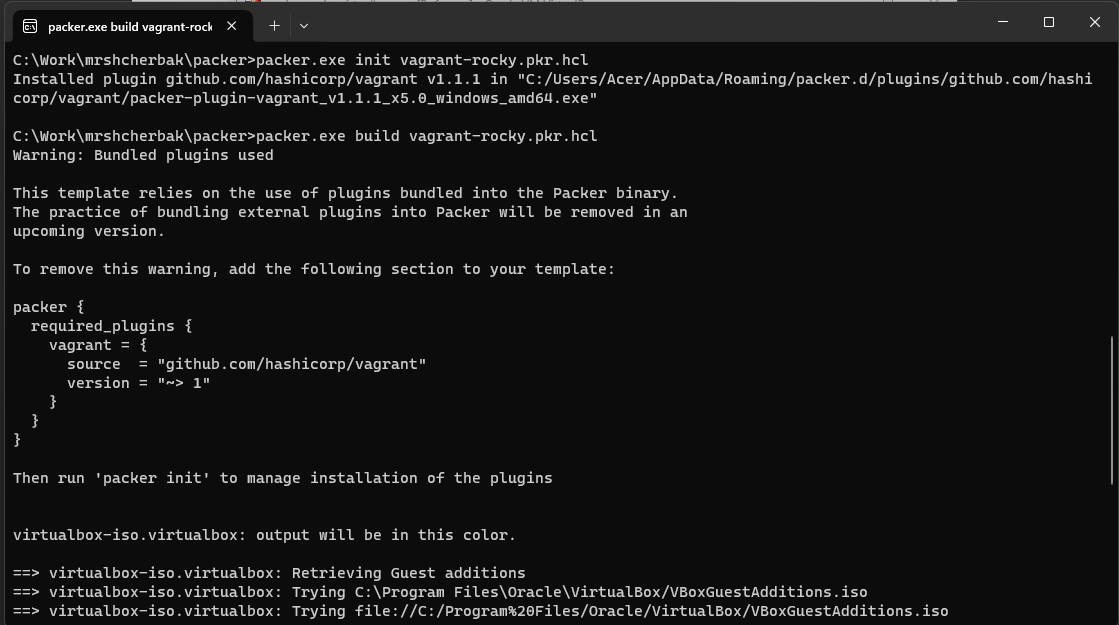


Рис.2.1. Выполнение команд

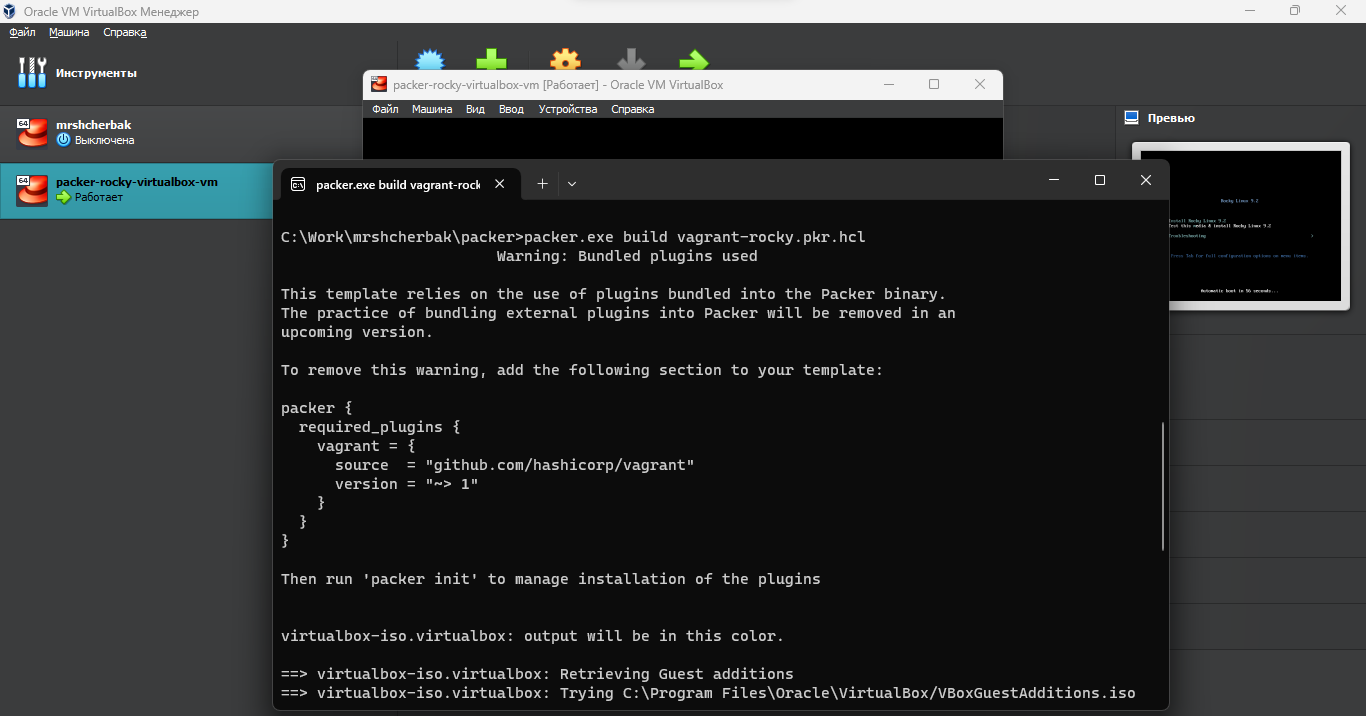


Рис.2.2. Автоматическая установка образа ОС Rocky Linux и последующее формирование box-файла

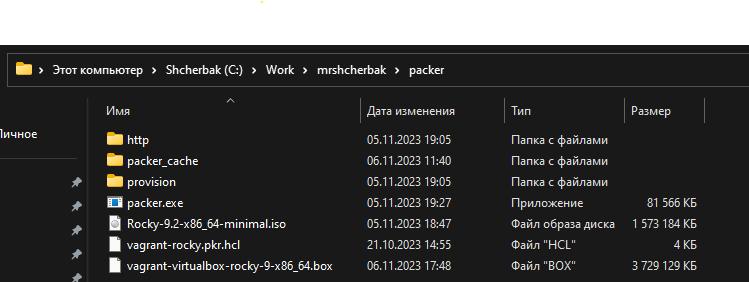


Рис.2.3. Появился сформированный box-файл

Для регистрации образа виртуальной машины в vagrant в командной строке ввела: vagrant box add rocky9 vagrant-virtualbox-rocky-9-x86\_64.box (рис.2.4).

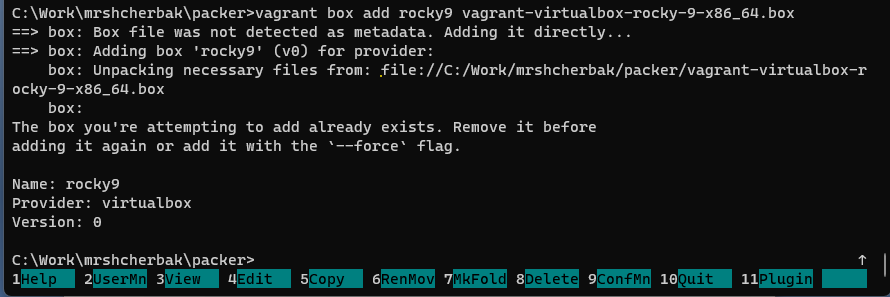


Рис.2.4. Выполнение команды

Для запуска виртуальной машины Server ввела в консоли: vagrant up server (рис.2.5).

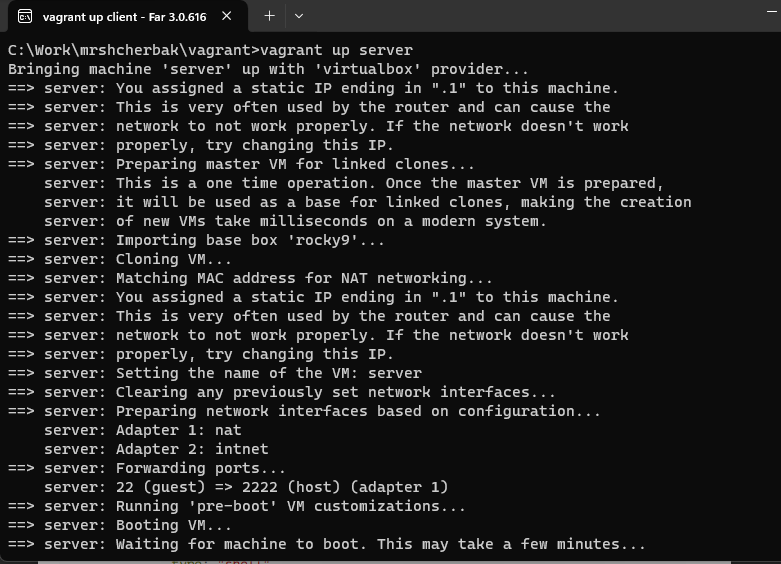


Рис.2.5. Выполнение команды

Для запуска виртуальной машины Client ввела в консоли: vagrant up client (рис.2.6).

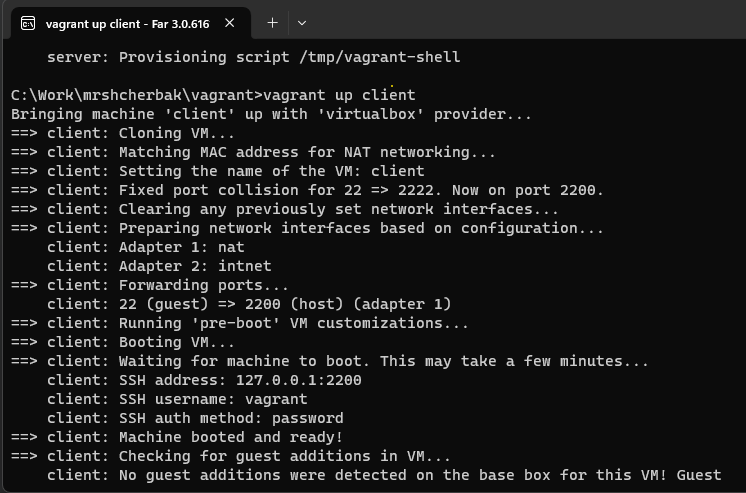


Рис.2.6. Выполнение команды

Убедилась, что запуск обеих виртуальных машин прошёл успешно. Корректно выключила виртуальные машины.

# 3. Внесение изменений в настройки внутреннего окружения виртуальной машины

Для отработки созданных скриптов во время загрузки виртуальных машин убедилась, что в конфигурационном файле Vagrantfile до строк с конфигурацией сервера имеется запись, выделенная красным цветом (рис.3.1).

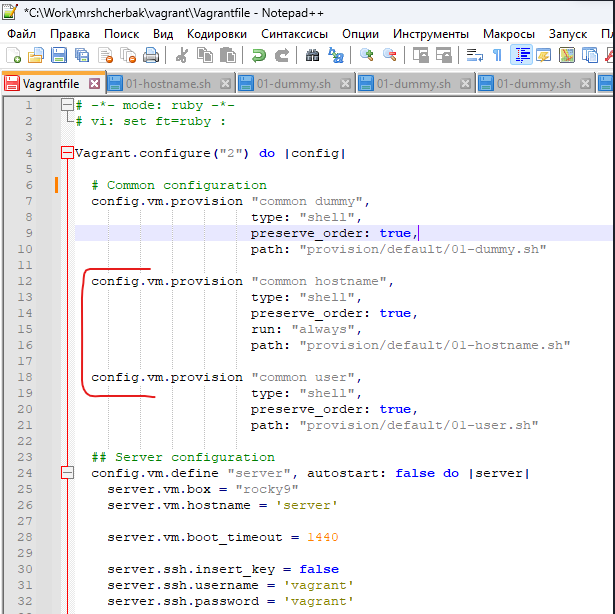


Рис.3.1. Содержимое файла Vagrantfile

Зафиксировала внесённые изменения для внутренних настроек виртуальных машин, введя в терминале:

vagrant up server --provision

vagrant up client –provision

Выполнение команд представлено на рис.3.2 и рис.3.3.

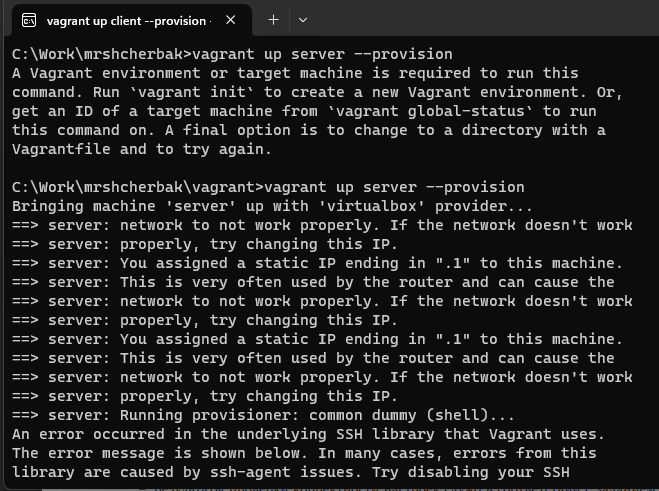


Рис.3.2. Выполнение команды vagrant up server --provision

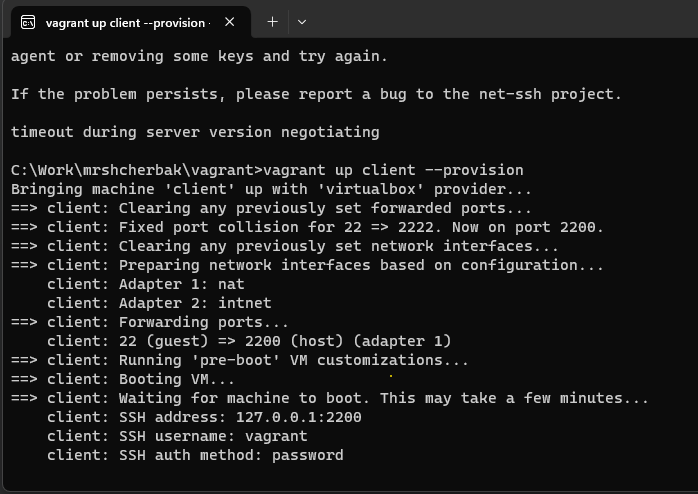


Рис.3.3. Выполнение команды vagrant up client --provision

Залогинилась на сервере и клиенте под созданным пользователем. Убедилась, что в терминале приглашение отображается в виде mrshcherbak@server.mrshcherbak.net на сервере и в виде mrshcherbak@client.mrshcherbak.net на клиенте (рис.3.4 – рис.3.5). Выключила виртуальные машины.

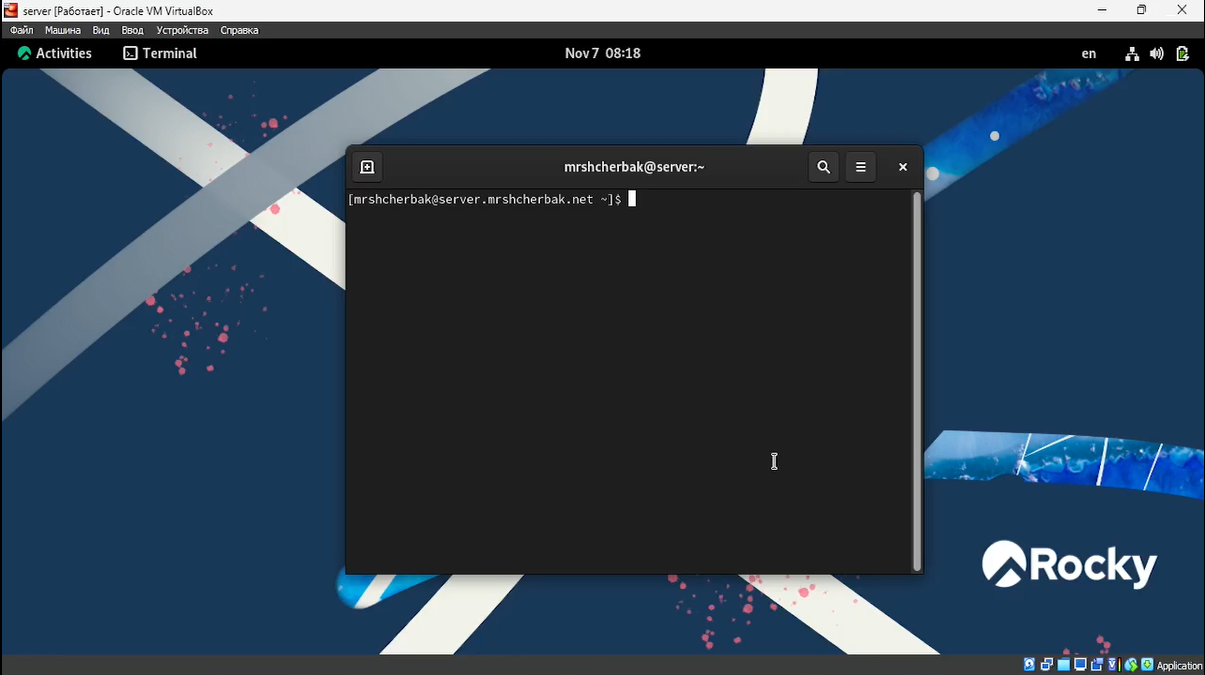


Рис.3.4. На сервере приглашение отображается в виде mrshcherbak@server.mrshcherbak.net

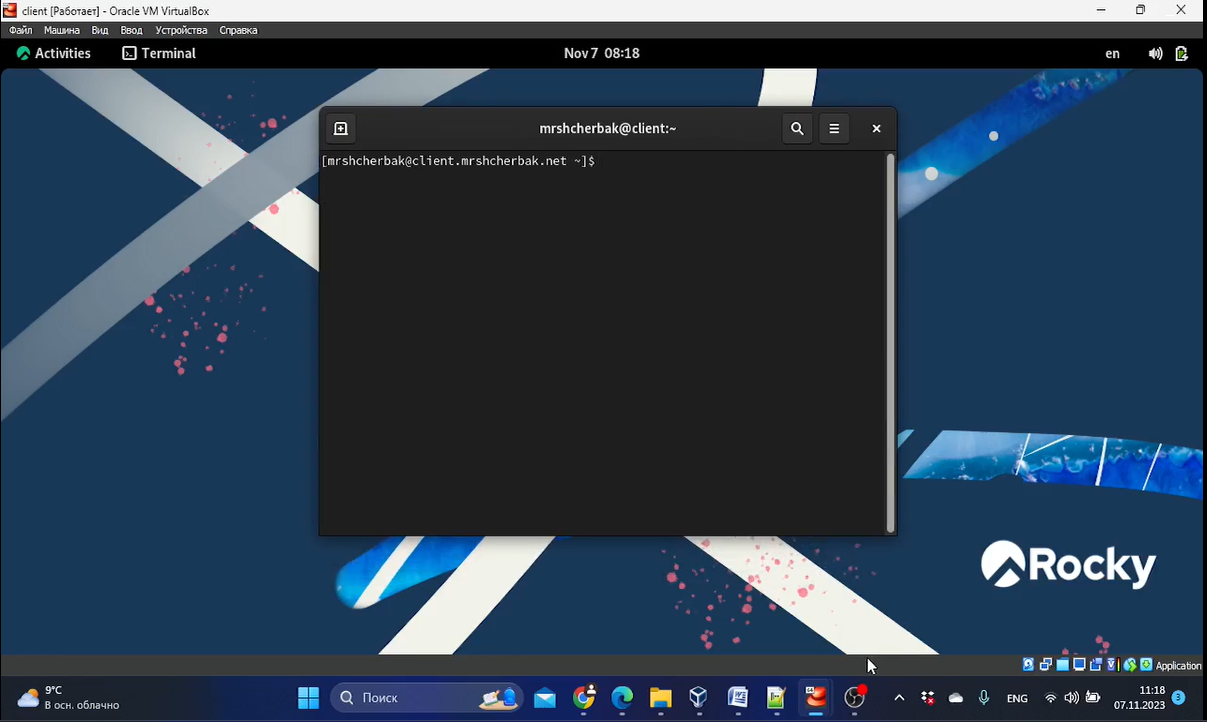


Рис.3.5. На клиенте приглашение отображается в виде mrshcherbak@client.mrshcherbak.net

После выключения виртуальных машин скопировала необходимые для работы с Vagrant файлы и box-файлы виртуальных машин в другой каталог (рис.3.6).

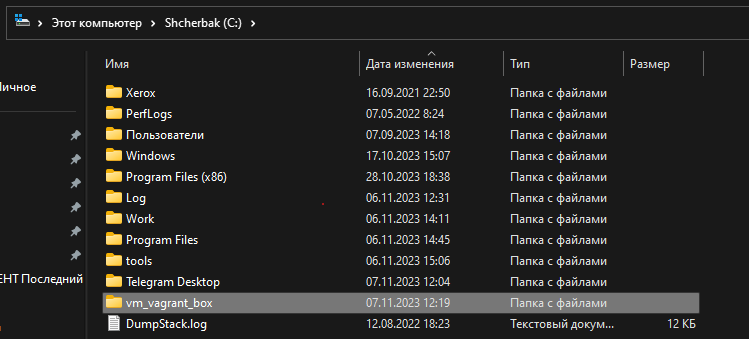


Рис.3.6. Копия необходимых файлов в другой каталог

**Вывод:** таким образом, в ходе л/р №1 я приобрела практические навыки установки Rocky Linux на виртуальную машину с помощью инструмента Vagrant.

# Контрольные вопросы

1. Для чего предназначен Vagrant?

Vagrant — представляет собой инструмент для создания и управления средами виртуальных машин в одном рабочем процессе. Этот инструмент, по сути, позволяет автоматизировать процесс установки на виртуальную машину как основного дистрибутива операционной системы, так и настройки необходимого в дальнейшем программного обеспечения.

2. 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 - это файл конфигурации для инструмента Vagrant. Он используется для определения параметров и настроек виртуальной среды, в том числе выбора базового box-образа, настройки сети и других параметров виртуальной машины. Vagrantfile позволяет описывать всю конфигурацию в одном месте, что упрощает управление виртуальными окружениями и делает процесс воспроизводимым.

3. Приведите описание и примеры вызова основных команд Vagrant.

С Vagrant можно работать, используя следующие основные команды:

* 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 отVagrant 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) в ранее созданную виртуальную машину);
* vagrant ssh — подключение к виртуальной машине через ssh.

4. Дайте построчные пояснения содержания файлов vagrant-rocky.pkr.hcl, ks.cfg, Vagrantfile, Makefile.

* vagrant-rocky.pkr.hcl — специальный файл с описанием метаданных по установке дистрибутива на виртуальную машину.

packer блок: определяет зависимость от плагина Vagrant и его версии.

variable блоки: определяют переменные, которые будут использоваться в конфигурации Packer. Например, artifact\_description, artifact\_version, и так далее.

source "virtualbox-iso" "virtualbox" блок: определяет источник для создания виртуальной машины с использованием образа VirtualBox.

build блок: определяет, какие источники использовать и какие провижинеры и пост-процессоры применять.

provisioner "shell" блок: используется для выполнения команд внутри виртуальной машины. В данном случае, устанавливает необходимое ПО и конфигурирует систему, также производится установка VirtualBox Guest Additions.

post-processor "vagrant" блок: определяет пост-процессор для создания Vagrant box файла. Указывает уровень сжатия и имя выходного файла.

* ks.cfg — определяет настройки для установки дистрибутива, которые пользователь обычно вводит вручную, в частности настройки языка интерфейса, языковые настройки клавиатуры, тайм-зону, сетевые настройки и т.п.;

bootloader блок: указывает параметры загрузчика, такие как аргументы ядра и местоположение загрузчика.

zerombr и clearpart блоки: очищают информацию о разделах для установки системы на "чистый" диск.

reboot блок: задает перезагрузку после завершения установки.

text блок: устанавливает режим установки в текстовом формате.

keyboard, lang блоки: устанавливают раскладку клавиатуры и язык системы.

network блок: конфигурирует сетевые параметры для автоматического получения IP-адреса по DHCP.

authselect и rootpw блоки: выбирают метод аутентификации и устанавливают пароль для пользователя root.

user блок: создает пользователя vagrant с паролем vagrant.

services блок: включает необходимые службы, такие как NetworkManager, sshd, chronyd.

timezone блок: устанавливает временную зону системы.

part блок: определяет раздел / с файловой системой XFS и размером 10 ГБ.

%post блок: содержит команды, которые будут выполнены после завершения установки.

Настроены swap-файл, sudo, сетевой интерфейс, sshd, SELinux, и другие параметры.

%packages блок: содержит список пакетов, которые будут установлены. Например, bash-completion, bzip2, chrony и т.д.

%addon com\_redhat\_kdump блок: отключает Kdump (причина "disable").

* Vagrantfile — файл с конфигурацией запуска виртуальных машин — сервера и клиента.

Vagrant.configure("2") блок: задает версию конфигурации Vagrant.

Общая конфигурация:

config.vm.provision блоки определяют три шага для общей конфигурации (01-dummy.sh, 01-hostname.sh, 01-user.sh).

Конфигурация сервера (server):

config.vm.define определяет виртуальную машину с именем "server".

Указаны параметры виртуальной машины, такие как имя box, имя хоста, таймаут загрузки, настройки SSH и сетевые параметры.

server.vm.provision блок определяет шаг для конфигурации сервера (01-dummy.sh).

server.vm.provider блок определяет параметры провайдера (VirtualBox) для сервера.

Конфигурация клиента (client):

config.vm.define определяет виртуальную машину с именем "client".

Указаны параметры виртуальной машины, такие как имя box, имя хоста, таймаут загрузки, настройки SSH и сетевые параметры.

client.vm.provision блок определяет два шага для конфигурации клиента (01-dummy.sh, 01-routing.sh).

client.vm.provider блок определяет параметры провайдера (VirtualBox) для клиента.

* Makefile — набор инструкций для программы make по работе с Vagrant.

help: выводит справку о доступных командах и их использовании.

plugins: устанавливает Vagrant-плагин (vagrant-vbguest).

addbox: добавляет построенный Vagrant box (vagrant-virtualbox-rocky-9-x86\_64.box) в Vagrant.

up: поднимает все виртуальные машины (сервер и клиент).

server-up и client-up: поднимают только сервер или только клиента.

server-provision и client-provision: поднимают и (выполняют дополнительные настройки) сервер или клиента.

server-destroy и client-destroy: уничтожают (останавливают и удаляют) сервер или клиента.