

Отчет по лабораторной работе № 1

Подготовка лабораторного стенда.

Бызова Мария Олеговна, НПИбд-01-23

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
3.1	Подготовка рабочего каталога	7
3.2	Развёртывание лабораторного стенда на ОС Windows	9
3.3	Внесение изменений в настройки внутреннего окружения виртуальной машины	11
4	Ответы на контрольные вопросы.	13
5	Выводы	15

Список иллюстраций

3.1	Созданный каталог проекта	7
3.2	Каталог packer	7
3.3	Каталог vagrant	8
3.4	Каталог provision	8
3.5	Каталог default	8
3.6	Каталог server	9
3.7	Каталог client	9
3.8	packer.exe init vagrant-rocky.pkr.hcl	9
3.9	packer.exe build vagrant-rocky.pkr.hcl	9
3.10	Регистрация образа виртуальной машины	10
3.11	Запуск виртуальной машины Server	10
3.12	Запуск виртуальной машины Client	10
3.13	Подключение к серверу из консоли	10
3.14	Отключение виртуальной машины сервер	11
3.15	Отключение виртуальной машины клиент	11
3.16	Фиксирование внесённых изменений	11
3.17	Фиксирование внесённых изменений	11
3.18	Отключение виртуальной машины сервер	12
3.19	Отключение виртуальной машины клиент	12

Список таблиц

1 Цель работы

Целью данной работы является приобретение практических навыков установки Rocky Linux на виртуальную машину с помощью инструмента Vagrant.

2 Теоретическое введение

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.

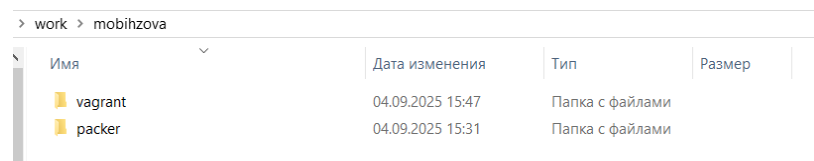
С проектом Vagrant и документацией по этому инструментальному средству можно ознакомиться на сайте <https://www.vagrantup.com>.

Основные понятия Vagrant: – провайдер (provider) — система виртуализации, с которой работает Vagrant (например, VirtualBox, VMWare и т.п.); – 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.

3 Выполнение лабораторной работы

3.1 Подготовка рабочего каталога

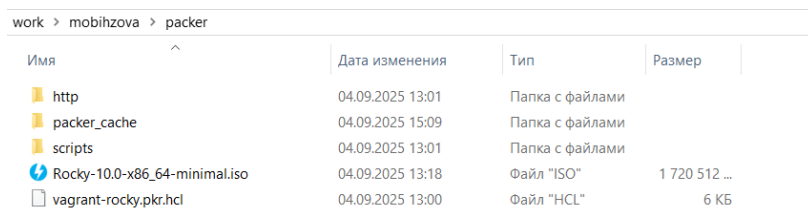
Перед началом работы с Vagrant я создала каталог для проекта(рис. 3.1).



Имя	Дата изменения	Тип	Размер
vagrant	04.09.2025 15:47	Папка с файлами	
packer	04.09.2025 15:31	Папка с файлами	

Рисунок 3.1: Созданный каталог проекта

В созданном рабочем каталоге в подкаталоге packer разместила образ варианта операционной системы Rocky Linux (рис. 3.2).



Имя	Дата изменения	Тип	Размер
http	04.09.2025 13:01	Папка с файлами	
packer_cache	04.09.2025 15:09	Папка с файлами	
scripts	04.09.2025 13:01	Папка с файлами	
Rocky-10.0-x86_64-minimal.iso	04.09.2025 13:18	Файл "ISO"	1 720 512 ...
vagrant-rocky.pkr.hcl	04.09.2025 13:00	Файл "HCL"	6 КБ

Рисунок 3.2: Каталог packer

В этом же рабочем каталоге разместила подготовленные заранее для работы с Vagrant файлы: в подкаталоге packer файл vagrant-rocky.pkr.hcl, в подкаталоге packer подкаталог http с файлом ks.cfg, в подкаталоге vagrant файл Vagrantfile. (рис. 3.3).

provision	04.09.2025 13:04	Папка с файлами	
Vagrantfile	04.09.2025 15:57	Файл	4 КБ

Рисунок 3.3: Каталог vagrant

В этом же рабочем каталоге в подкаталоге vagrant 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. 3.4).

work > mobihzova > vagrant > provision			
Имя	Дата изменения	Тип	Разме
client	04.09.2025 13:04	Папка с файлами	
default	04.09.2025 13:05	Папка с файлами	
server	04.09.2025 13:05	Папка с файлами	

Рисунок 3.4: Каталог provision

В каталогах default, server и client разместила заранее подготовленный скрипт-заглушку 01-dummy.sh

В каталоге default разместила заранее подготовленный скрипт 01-user.sh

В каталоге default разместила заранее подготовленный скрипт 01-hostname.sh (рис. 3.5).

> work > mobihzova > vagrant > provision > default			
Имя	Дата изменения	Тип	Размер
01-dummy.sh	04.09.2025 13:04	Исходный файл SH	1 КБ
01-hostname.sh	04.09.2025 13:06	Исходный файл SH	1 КБ
01-user.sh	04.09.2025 13:06	Исходный файл SH	1 КБ

Рисунок 3.5: Каталог default

В каталоге server разместила заранее подготовленный скрипт 02-forward.sh (рис. 3.6).

> work > mobihzova > vagrant > provision > server				
Имя	Дата изменения	Тип	Размер	
01-dummy.sh	04.09.2025 13:04	Исходный файл SH	1 КБ	
02-forward.sh	04.09.2025 13:05	Исходный файл SH	1 КБ	

Рисунок 3.6: Каталог server

В каталоге client разместила заранее подготовленный скрипт 01-routing.sh (рис. 3.7).

> work > mobihzova > vagrant > provision > client				
Имя	Дата изменения	Тип	Размер	
01-dummy.sh	04.09.2025 13:04	Исходный файл SH	1 КБ	
01-routing.sh	04.09.2025 13:04	Исходный файл SH	1 КБ	

Рисунок 3.7: Каталог client

3.2 Развёртывание лабораторного стенда на ОС

Windows

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

packer.exe init vagrant-rocky.pkr.hcl (рис. 3.8).

```
Far Manager, version 3.0.6446.0 x86
Copyright © 1996-2000 Eugene Roshal, Copyright © 2000-2025 Far Group
C:\work\mobihzova\packer>packer.exe init vagrant-rocky.pkr.hcl
```

Рисунок 3.8: packer.exe init vagrant-rocky.pkr.hcl

packer.exe build vagrant-rocky.pkr.hcl (рис. 3.9).

```
C:\work\mobihzova\packer>packer.exe build vagrant-rocky.pkr.hcl
```

Рисунок 3.9: packer.exe build vagrant-rocky.pkr.hcl

Для регистрации образа виртуальной машины в vagrant в командной строке ввела (рис. 3.10).

```
vagrant box add rocky9 vagrant-virtualbox-rocky-9-x86_64.box
```

```
C:\work\mobihzova\packer>vagrant box add rocky10 vagrant-virtualbox-rockylinux10-x86_64.box
==> box: Box file was not detected as metadata. Adding it directly...
==> box: Adding box 'rocky10' (v0) for provider: (amd64)
      box: Unpacking necessary files from: file:///C:/work/mobihzova/packer/vagrant-virtualbox-rockylinux10-x86_64.box
      box:
==> box: Successfully added box 'rocky10' (v0) for '(amd64)'!
```

Рисунок 3.10: Регистрация образа виртуальной машины

Для запуска виртуальной машины Server ввела в консоли `vagrant up server` (рис. 3.11).

```
C:\work\mobihzova\packer>vagrant up server
```

Рисунок 3.11: Запуск виртуальной машины Server

Для запуска виртуальной машины Client ввела в консоли `vagrant up client` (рис. 3.12).

```
C:\work\mobihzova\vagrant>vagrant up client
```

Рисунок 3.12: Запуск виртуальной машины Client

Убедилась, что запуск обеих виртуальных машин прошёл успешно, залогинилась под пользователем `vagrant` с паролем `vagrant` в графическом окружении.

Подключилась к серверу из консоли: `vagrant ssh server`. Ввела пароль `vagrant` (рис. 3.13).

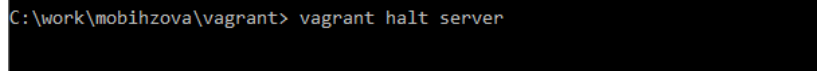
```
C:\work\mobihzova\vagrant>vagrant ssh server
```

Рисунок 3.13: Подключение к серверу из консоли

Перешла к пользователю `mobihzova`. Отлогинилась. Проделала то же самое для клиента.

Выключила обе виртуальные машины

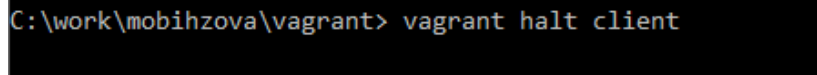
`vagrant halt server` (рис. 3.14).



```
C:\work\mobihzova\vagrant> vagrant halt server
```

Рисунок 3.14: Отключение виртуальной машины сервер

`vagrant halt client` (рис. 3.15).



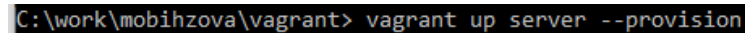
```
C:\work\mobihzova\vagrant> vagrant halt client
```

Рисунок 3.15: Отключение виртуальной машины клиент

3.3 Внесение изменений в настройки внутреннего окружения виртуальной машины

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

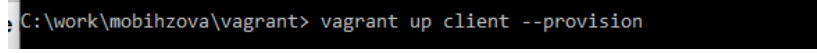
`vagrant up server --provision` (рис. 3.16).



```
C:\work\mobihzova\vagrant> vagrant up server --provision
```

Рисунок 3.16: Фиксирование внесённых изменений

`vagrant up client --provision` (рис. 3.17).



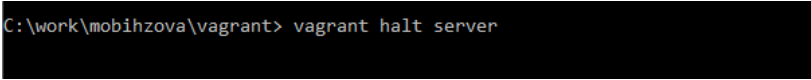
```
C:\work\mobihzova\vagrant> vagrant up client --provision
```

Рисунок 3.17: Фиксирование внесённых изменений

Залогинилась на сервере и клиенте под созданным пользователем.

Выключила виртуальные машины.

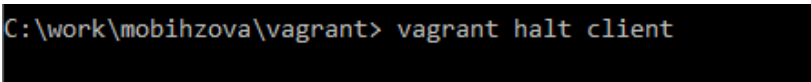
`vagrant halt server` (рис. 3.18).



```
C:\work\mobihzova\vagrant> vagrant halt server
```

Рисунок 3.18: Отключение виртуальной машины сервер

vagrant halt client (рис. 3.19).



```
C:\work\mobihzova\vagrant> vagrant halt client
```

Рисунок 3.19: Отключение виртуальной машины клиент

После выключения виртуальных машин скопировала необходимые для работы с Vagrant файлы и box-файлы виртуальных машин на внешний носитель.

4 Ответы на контрольные вопросы.

1. Для чего предназначен Vagrant? Vagrant используется для создания и конфигурирования виртуальной среды разработки, при чем создаваемое окружение можно легко повторять и переносить.
2. Что такое box-файл? В чём назначение Vagrantfile? Boxфайл содержит образ виртуальной машины, Vagrantfile, в котором дается описание машины и информация, как настроить и подготовить ее к работе.
3. Приведите описание и примеры вызова основных команд Vagrant.
 - Создание и запуск виртуальной машины: `vagrant up`
 - Вход: `vagrant ssh`
 - Остановка машины: `vagrant halt`
 - Удаление собранной машины: `vagrant destroy`
4. Дайте строчные пояснения содержания файлов `vagrant-rocky.pkr.hcl`, `ks.cfg`, `Vagrantfile`, `Makefile`

- Файл `vagrant-rocky.pkr.hcl`

Блок `racker` устанавливает, что для работы необходимы версии `vagrant` и `VirtualBox` не ниже 1 (`version = «> 1»`). Затем идут блоки `variable`, где задаются переменные, которые будут использоваться в работе скрипта, например имя ВМ, версия, размер дискового пространства, архитектура процессора и т. д. Блок `source` задает конфигурацию сборки

с возможностью переиспользования. В нашем случае задаются параметры сборки виртуальной машины в VirtualBox, какой образ использовать, сколько выделить оперативной памяти, ядер процессора. Последний блок build описывает сам процесс сборки. Здесь указаны скрипты, которые будут запущены: настройка каталогов, установка необходимых для работы утилит.

- ks.cfg

В этом файле мы задаем настройки для установки дистрибутива, которые обычно выбираются пользователем вручную при установке дистрибутива. Определяем системный язык, необходимые раскладки клавиатуры (русская и английская), логин и пароль root-пользователя, настраиваем swap.

- Vagrantfile

Описываем конфигурацию запуска виртуальных машин сервера и клиента: количество оперативной памяти, видеопамати, имя хоста, настройки VBox-Additions.

- Makefile

Содержат скрипты для программы make, упрощающие работу с vagrant. Содержит следующие цели: addbox, client-destroy, client-halt, client-provision, client-up, plugins, server-destroy, server-halt, server-provision, server-up, каждая из которых вызывает утилиту vagrant с соответствующими параметрами.

5 Выводы

В рамках лабораторной работы познакомились с инструментом Vagrant и подготовили лабораторный стенд.