Отчет по лабораторной работе № 1

Подготовка лабораторного стенда.

Бызова Мария Олеговна, НПИбд-01-23

Содержание

# 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. 1](#fig-001)).

|  |
| --- |
| Рисунок 1: Созданный каталог проекта |

В созданном рабочем каталоге в подкаталоге packer разместила образ варианта операционной системы Rocky Linux ([рис. 2](#fig-002)).

|  |
| --- |
| Рисунок 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](#fig-003)).

|  |
| --- |
| Рисунок 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. 4](#fig-004)).

|  |
| --- |
| Рисунок 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. 5](#fig-005)).

|  |
| --- |
| Рисунок 5: Каталог default |

В каталоге server разместила заранее подготовленный скрипт 02-forward.sh ([рис. 6](#fig-006)).

|  |
| --- |
| Рисунок 6: Каталог server |

В каталоге client разместила заранее подготовленный скрипт 01-routing.sh ([рис. 7](#fig-007)).

|  |
| --- |
| Рисунок 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. 8](#fig-008)).

|  |
| --- |
| Рисунок 8: packer.exe init vagrant-rocky.pkr.hcl |

packer.exe build vagrant-rocky.pkr.hcl ([рис. 9](#fig-009)).

|  |
| --- |
| Рисунок 9: packer.exe build vagrant-rocky.pkr.hcl |

Для регистрации образа виртуальной машины в vagrant в командной строке ввела ([рис. 10](#fig-010)).

vagrant box add rocky9 vagrant-virtualbox-rocky-9-x86\_64.box

|  |
| --- |
| Рисунок 10: Регистрация образа виртуальной машины |

Для запуска виртуальной машины Server ввела в консоли vagrant up server ([рис. 11](#fig-011)).

|  |
| --- |
| Рисунок 11: Запуск виртуальной машины Server |

Для запуска виртуальной машины Client ввела в консоли vagrant up client ([рис. 12](#fig-012)).

|  |
| --- |
| Рисунок 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. 13](#fig-013)).

|  |
| --- |
| Рисунок 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. 14](#fig-014)).

|  |
| --- |
| Рисунок 14: Отключение виртуальной машины сервер |

vagrant halt client ([рис. 15](#fig-015)).

|  |
| --- |
| Рисунок 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. 16](#fig-016)).

|  |
| --- |
| Рисунок 16: Фиксирование внесённых изменений |

vagrant up client –provision ([рис. 17](#fig-017)).

|  |
| --- |
| Рисунок 17: Фиксирование внесённых изменений |

Залогинилась на сервере и клиенте под созданным пользователем.

Выключила виртуальные машины.

vagrant halt server ([рис. 18](#fig-018)).

|  |
| --- |
| Рисунок 18: Отключение виртуальной машины сервер |

vagrant halt client ([рис. 19](#fig-019)).

|  |
| --- |
| Рисунок 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: vargrant up
* Вход: vagrant ssh
* Остановка маишны: vagrant halt
* Удаление собранной машины: vagrant destroy

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

• Файл vagrant-rocky.pkr.hcl

Блок packer 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 сборщики 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 установки 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, видеопамяти, имя хоста, настройки VBoxAddtions.

• Makefile

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

# 5. Выводы

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