

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

## ОТЧЕТ

### ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

*дисциплина:* Администрирование сетевых подсистем

Студент: Бакулин Никита 1032201747

Группа: НПИбд-01-20

МОСКВА

2022 г.

## **Постановка задачи**

1. Сформируйте box-файл с дистрибутивом Rocky Linux для VirtualBox
2. Запустите виртуальные машины сервера и клиента и убедитесь в их работоспособности.
3. Внесите изменения в настройки загрузки образов виртуальных машин server и client, добавив пользователя с правами администратора и изменив названия хостов.

## **Выполнение работы**

1.
  - 1.1. Перед началом работы с Vagrant создайте каталог для проекта.
  - 1.2. В созданном рабочем каталоге разместите образ варианта операционной системы Rocky Linux
  - 1.3. В этом же каталоге разместите подготовленные заранее для работы с Vagrant файлы
  - 1.4. В этом же каталоге создайте 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
  - 1.5. В каталогах default, server и client разместите заранее подготовленный скриптзаглушку 01-dummy.sh
  - 1.6. В каталоге default разместите заранее подготовленный скрипт 01-user.sh
  - 1.7. В каталоге default разместите заранее подготовленный скрипт 01-hostname.sh
  - 1.8. Используя FAR, перейдите в созданный вами рабочий каталог с проектом. В этом же каталоге должен быть размещён файл packer.exe. В командной строке введите packer.exe build vagrant-rocky.pkr.hcl для начала автоматической установки образа операционной системы Rocky Linux в VirtualBox и последующего формирования box-файла с дистрибутивом Rocky Linux для VirtualBox. По окончании процесса в рабочем каталоге сформируется box-файл с названием vagrant-virtualbox-rocky-9-x86\_64.box.

| D:\Desktop\Documents\labs\packer        |      | D:\Desktop\Documents\labs\vagrant   |      |
|---|------|-------------------------------------|------|
| n                                       | Name | n                                   | Name |
| ..                                      |      | ..                                  |      |
| http                                    |      | provision                           |      |
| packer_cache                            |      | Makefile                            |      |
| Makefile                                |      | Vagrantfile                         |      |
| packer.exe                              |      |                                     |      |
| Rocky-9.0-20220805.0-x86_64-minimal.iso |      |                                     |      |
| vagrant-rocky.pkr.hcl                   |      |                                     |      |
| vagrant-rocky.pkr.hcl                   |      | Vagrantfile                         |      |
| 4938 01.11.22 12:02                     |      | 2472 01.11.22 12:14                 |      |
| Bytes: 1 53 6, files: 4, folders: 2     |      | Bytes: 4 92 K, files: 2, folders: 1 |      |

Рисунок 1

```
D:\Desktop\Documents\labs\packer>packer.exe init vagrant-rocky.pkr.hcl
Installed plugin github.com/hashicorp/vmware v1.0.7 in "C:/Users/bakul/AppData/Roaming/packer.d/plugins/github.com/hashicorp/vmware/packer-plugin-vmware_v1.0.7_x5.0_windows_amd64.exe"
```

Рисунок 2

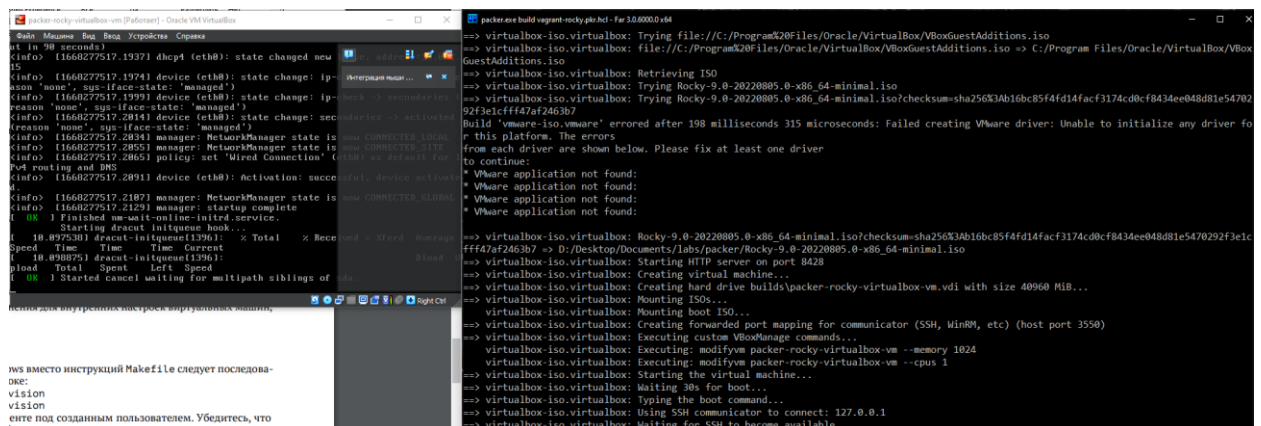


Рисунок 3

- 1.9. Для регистрации образа виртуальной машины в vagrant в командной строке введите `vagrant box add rocky9 vagrant-virtualbox-rocky-9-x86_64.box`
- 1.10. Для запуска виртуальной машины Server введите в консоли vagrant `up server`
- 1.11. Для запуска виртуальной машины Client введите в консоли vagrant `up client`

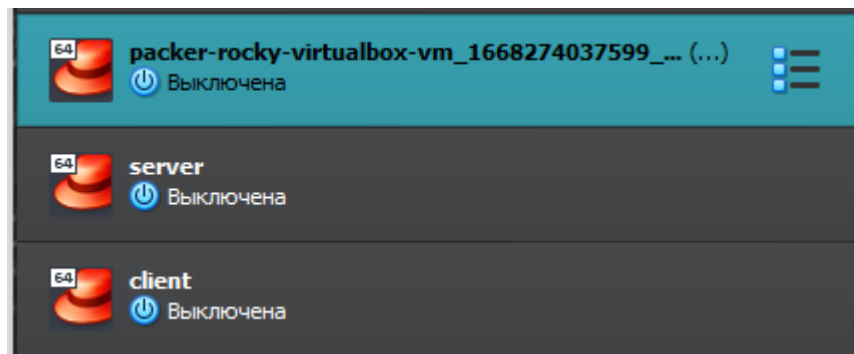


Рисунок 4

- 1.12. Убедитесь, что запуск обеих виртуальных машин прошёл успешно.  
Корректно выключите виртуальные машины.

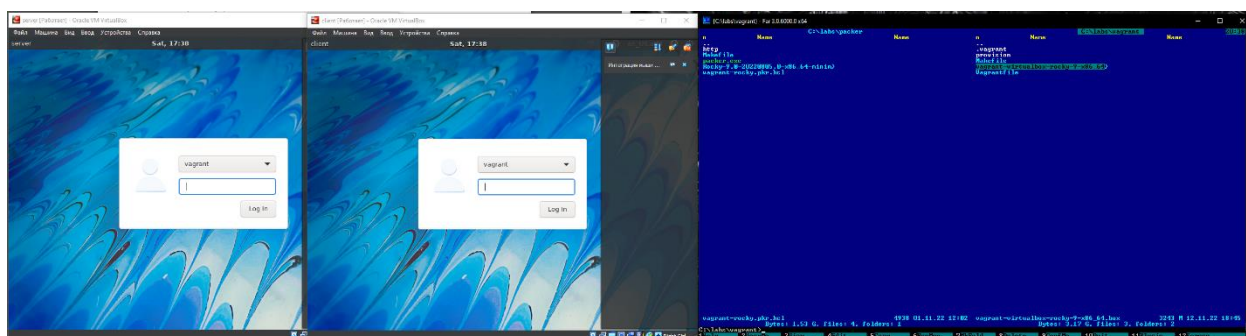


Рисунок 5

2.

- 2.1. Для отработки созданных скриптов во время загрузки виртуальных машин убедитесь, что в конфигурационном файле Vagrantfile до строк с конфигурацией сервера имеется следующая запись:

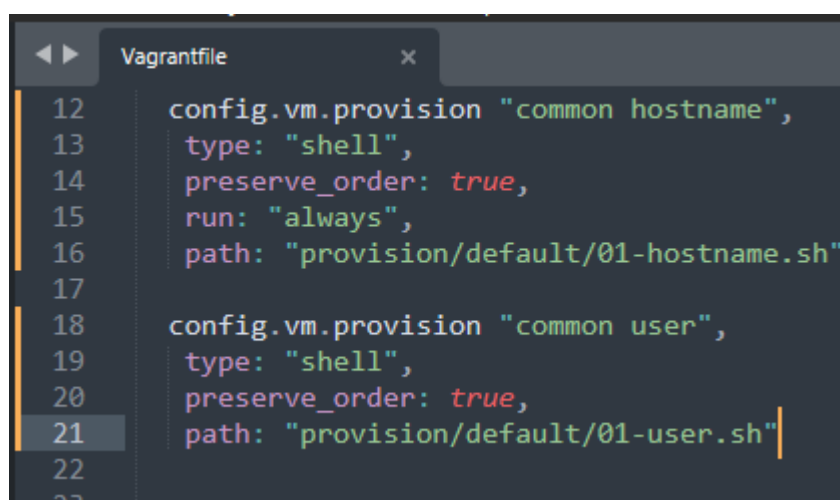


Рисунок 6

- 2.2. Зафиксируйте внесённые изменения для внутренних настроек виртуальных

машин, введя в терминале `vagrant up server --provision` `vagrant up client --provision`

2.3. Залогиньтесь на сервере и клиенте под созданным пользователем. Убедитесь, что в терминале приглашение отображается в виде `user@server.user.net` на сервере и в виде `user@client.user.net` на клиенте, где вместо `user` указан ваш логин. 4. Выключите виртуальные машины

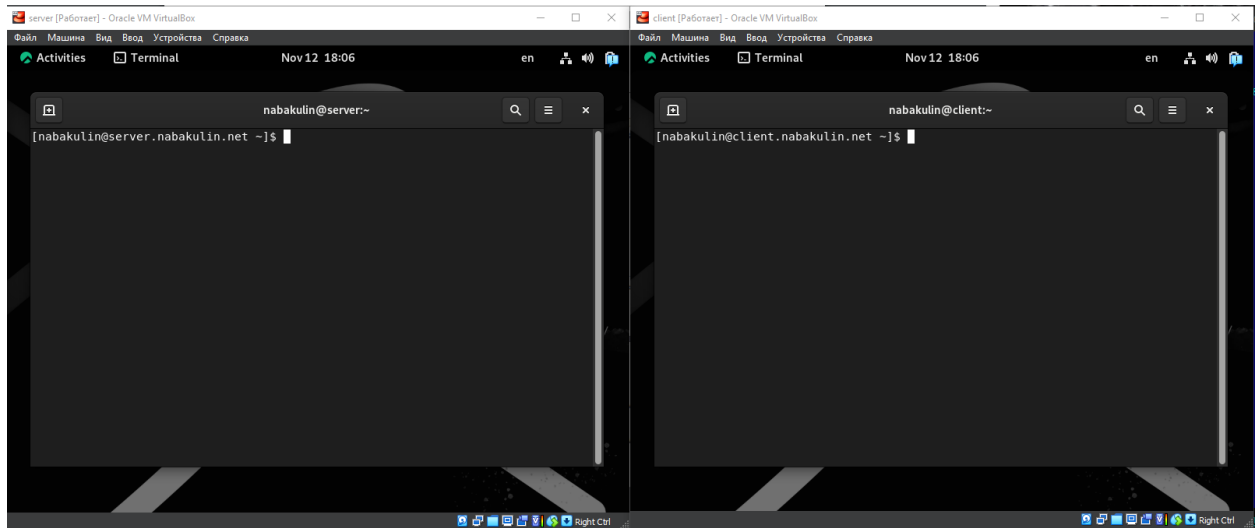


Рисунок 7

2.4. После выключения виртуальных машин скопируйте необходимые для работы с Vagrant файлы и box-файлы виртуальных машин на внешний носитель или в другой каталог вашей ОС

### Контрольные вопросы

1. Для чего предназначен Vagrant?

Для создания и конфигурирования виртуальной среды разработки (обертка над по виртуализации)

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

Формат пакета для окружений Vagrant, конфигурирование окружения производится в Vagrantfile.

3. Приведите описание и примеры вызова основных команд Vagrant

1. – `vagrant help` — вызов справки по командам Vagrant;
2. – `vagrant box list` — список подключённых к Vagrant box-файлов;
3. – `vagrant box add` — подключение box-файла к Vagrant;
4. – `vagrant destroy` — отключение box-файла от Vagrant и удаление его из виртуального окружения;
5. окружения;

6. – `vagrant init` — создание «шаблонного» конфигурационного файла
  7. `Vagrantfile` для его последующего изменения;
  8. – `vagrant up` — запуск виртуальной машины с использованием инструкций по запуску
  9. из конфигурационного файла `Vagrantfile`;
  10. – `vagrant reload` — перезагрузка виртуальной машины;
  11. – `vagrant halt` — остановка и выключение виртуальной машины;
  12. – `vagrant provision` — настройка внутреннего окружения имеющейся виртуальной
  13. машины (например, добавление новых инструкций (скриптов) в ранее созданную
  14. виртуальную машину);
  15. – `vagrant ssh` — подключение к виртуальной машине через `ssh`.
4. Дайте построчные пояснения содержания файлов `vagrant-rocky.pkr.hcl`, `ks.cfg`, `Vagrantfile`, `Makefile`.

- a. Настройка версии `racker` и плагинов

```
packer {
  required_version = ">= 1.7.0"
  required_plugins {
    vmware = {
      version = ">= v1.0.0"
      source  = "github.com/hashicorp/vmware"
    }
  }
}
```

- b. Настройки источника - `virtualbox`

```
source "virtualbox-iso" "virtualbox" {
  boot_command = ["<esc>", "<wait><esc><esc>", "linux",
    "inst.ks=http://{{.HTTPIP}}:{{.HTTPPort}}/ks.cfg biosdevname=0 net.ifnames=0", "<enter>"]
  boot_wait    = "30s"
  disk_size    = "${var.disk_size}"
  export_opts  = ["--manifest", "--vsys", "0", "--description", "${var.artifact_description}",
    "--version", "${var.artifact_version}"]
  guest_additions_path = "VBoxGuestAdditions.iso"
  guest_os_type = "RedHat_64"
  hard_drive_interface = "sata"
  http_directory = "${path.root}/http"
  iso_checksum   = "${var.iso_checksum_type}:${var.iso_checksum}"
  iso_url        = "${var.iso_url}"
  output_directory = "builds"
  shutdown_command = "sudo -S /sbin/halt -h -p"
  shutdown_timeout = "5m"
  ssh_password    = "${var.ssh_password}"
  ssh_username    = "${var.ssh_username}"
  ssh_port        = 22
  ssh_pty         = true
  ssh_timeout     = "60m"
  vboxmanage      = ["modifyvm", "${.Name}", "--memory", "1024", ["modifyvm", "${.Name}"],
    "--cpus", "1"]
  virtualbox_version_file = ".vbox_version"
  vm_name              = "packer-rocky-virtualbox-vm"
}
```

- c. Настройки сборки

```

build {
    sources = ["source.virtualbox-iso.virtualbox"]

    provisioner "shell" {
        execute_command = "echo 'packer'|{{ .Vars }} sudo -S -E bash '{{ .Path }}'"
        inline           = ["sleep 30",
            "sudo dnf -y install epel-release",
            "sudo dnf -y groupinstall 'Development Tools'",
            "sudo dnf -y install kernel-devel",
            "sudo dnf -y install dkms",
            "sudo mkdir /tmp/vboxguest",
            "sudo mount -t iso9660 -o loop /home/vagrant/VBoxGuestAdditions.iso /tmp/vboxguest",
            "cd /tmp/vboxguest",
            "sudo ./VBoxLinuxAdditions.run",
            "cd /tmp",
            "sudo umount /tmp/vboxguest",
            "sudo rmdir /tmp/vboxguest",
            "rm /home/vagrant/VBoxGuestAdditions.iso",
            "sudo dnf -y install lightdm",
            "sudo dnf -y groupinstall 'Server with GUI'",
            "sudo dnf install -y mc htop tmux",
            "sudo systemctl set-default graphical.target",
            "echo Image Provisioned!"]
    }

    post-processor "vagrant" {
        compression_level = "6"
        output              = "vagrant-virtualbox-rocky-${var.redhat_release}-${var.redhat_platform}.box"
    }
}

```