

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

«Подготовка лабораторного стенда»

Дисциплина: Администрирование сетевых подсистем

Студент: Карташова А.С.

Группа: НФИбд-03-18

МОСКВА

2020 г.

Оглавление

Цель работы	2
Задачи.....	2
Ход работы	2
Внесение изменений в настройки внутреннего окружения виртуальной машины.....	10
Заключение.....	11
Контрольные вопросы.....	12

Цель работы

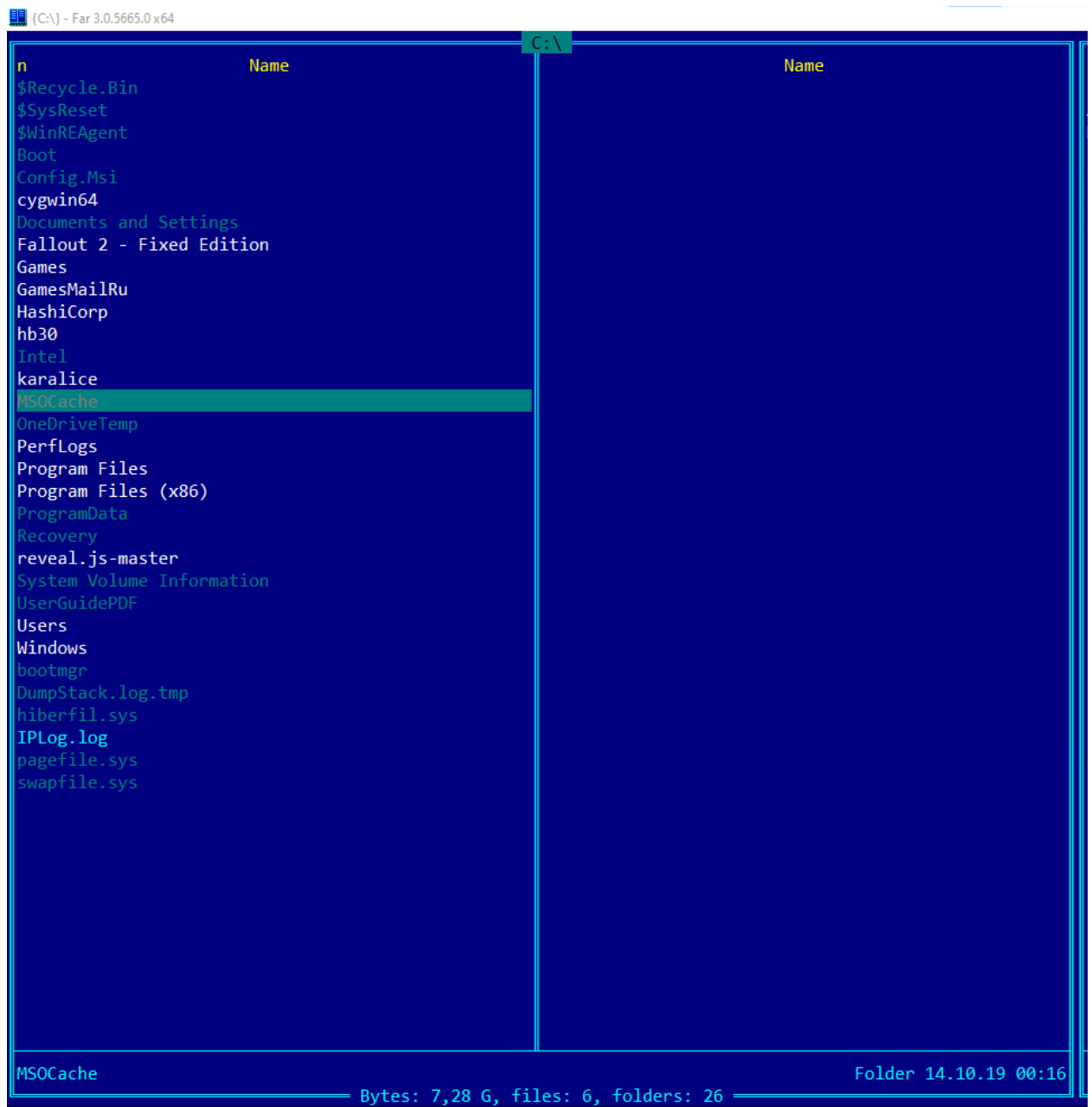
Целью данной работы является приобретение практических навыков установки CentOS на виртуальную машину с помощью инструмента Vagrant.

Задачи

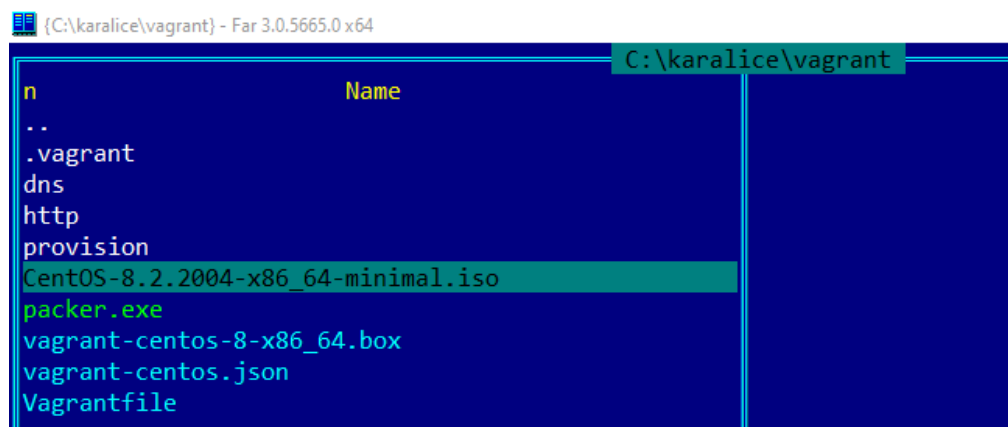
1. Сформировать box-файл с дистрибутивом CentOS для VirtualBox
2. Запустить виртуальные машины сервера и клиента и убедиться в их работоспособности
3. Внести изменения в настройки загрузки образов виртуальных машин server и client, добавить пользователя с правами администратора и изменить названия хостов
4. Скопировать необходимые для работы с Vagrant файлы и box-файлы виртуальных машин на внешний носитель.

Ход работы

Перед началом работы с Vagrant создайте каталог karalice\vagrant для проекта

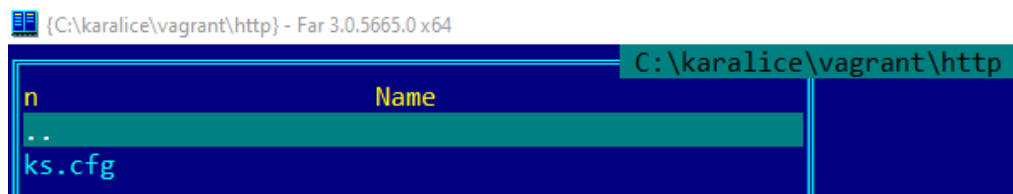
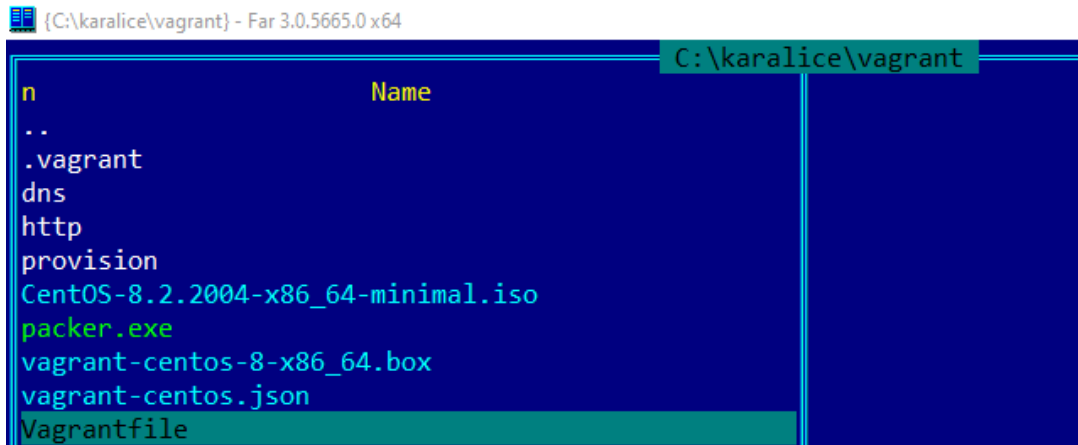


В созданном рабочем каталоге разместим образ операционной системы CentOS 8

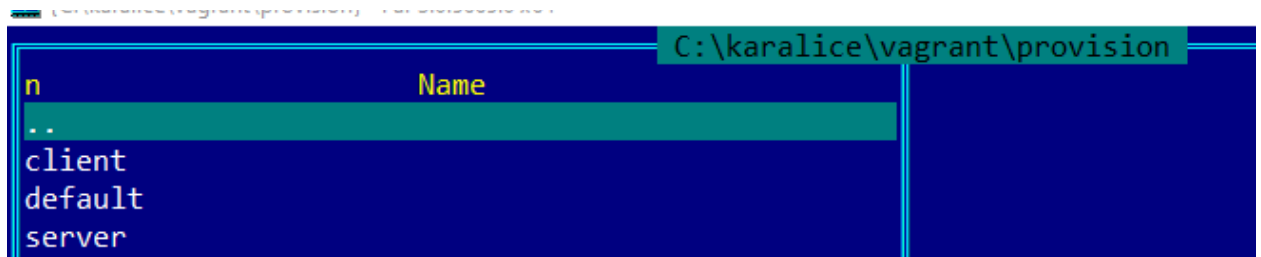


В этом же каталоге разместим подготовленные заранее для работы с Vagrant файлы:

- vagrant-centos.json
- ks.cfg
- Vagrantfile



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.



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

```
#!/bin/bash
```

```
echo "Provisioning script $0"
```

```
(C:\karalice\vagrant) - Far 3.0.5665.0 x64
C:\karalice\vagrant\provision\default
n      Name
..
01-dummy.sh
01-hostname.sh
01-user.sh
```

```
C:\karalice\vagrant\provision\server
n      Name
..
dhcp
dns
firewall
http
mail
mysql
nfs
ntp
ssh
01-dummy.sh
```

```
C:\karalice\vagrant\provision\client
n      Name
..
ntp
01-dummy.sh
01-routing.sh
mail.sh
```

```
edit 01-dummy.sh - Far 3.0.5665.0 x64
C:\karalice\vagrant\provision\client\01-dummy.sh
#!/bin/bash

echo "Provisioning script $0"
```

В каталоге default разместим заранее подготовленный скрипт 01-user.sh по изменению названия виртуальной машины

```
edit 01-user.sh - Far 3.0.5665.0 x64
C:\karalice\vagrant\provision\default\01-user.sh
#!/bin/bash

echo "Provisioning script $0"

username=askartashova
userpassword=123456

encpassword=`openssl passwd -1 ${userpassword}`

id -u $username
if [[ $? ]]
then
    adduser -G wheel -p ${encpassword} ${username}
    homedir=`getent passwd ${username} | cut -d: -f6`
    echo "export PS1='[\u@\H \W]\$ '" >> ${homedir}/.bashrc
fi
```

В каталоге default разместим заранее подготовленный скрипт 01-hostname.sh по изменению названия виртуальной машины.

```
edit 01-hostname.sh - Far 3.0.5665.0 x64
C:\karallice\vagrant\provision\default\01-hostname.sh
#!/bin/bash

echo "Provisioning script $0"
username=askartashova
hostnamectl set-hostname "${HOSTNAME%%.*}.${username}.net"
```

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

packer.exe build vagrant-centos.json

```
(C:\kartashova_alice\vagrant) - Far 3.0.5665.0 x64
centos-8: Transaction check succeeded.
centos-8: Running transaction test
centos-8: Transaction test succeeded.
centos-8: Running transaction
centos-8:   Preparing           :                               1/1
centos-8:   Installing          : http-2.2.0-6.el8.x86_64         1/3
centos-8:   Installing          : tmux-2.7-1.el8.x86_64             2/3
centos-8:   Running scriptlet: tmux-2.7-1.el8.x86_64             2/3
centos-8:   Installing          : mc-1:4.8.19-9.el8.x86_64         3/3
centos-8:   Running scriptlet: mc-1:4.8.19-9.el8.x86_64         3/3
centos-8:   Verifying          : mc-1:4.8.19-9.el8.x86_64         1/3
centos-8:   Verifying          : tmux-2.7-1.el8.x86_64             2/3
centos-8:   Verifying          : http-2.2.0-6.el8.x86_64         3/3
centos-8: Installed products updated.
centos-8:
centos-8: Installed:
centos-8:  http-2.2.0-6.el8.x86_64  mc-1:4.8.19-9.el8.x86_64  tmux-2.7-1.el8.x86_64
centos-8:
centos-8: Complete!
centos-8: Removed /etc/systemd/system/default.target.
centos-8: Created symlink /etc/systemd/system/default.target → /usr/lib/systemd/system/graphical.target.
centos-8: Image Provisioned!
==> centos-8: Gracefully halting virtual machine...
==> centos-8: Preparing to export machine...
centos-8: Deleting forwarded port mapping for the communicator (SSH, WinRM, etc) (host port 3368)
==> centos-8: Exporting virtual machine...
centos-8: Executing: export packer-centos-vm --output builds\packer-centos-vm.ovf --manifest --vsys 0 --description CentOS 8.2 (build 2004) --version 8.2.2004
==> centos-8: Cleaning up floppy disk...
==> centos-8: Deregistering and deleting VM...
==> centos-8: Running post-processor: vagrant
==> centos-8 (vagrant): Creating a dummy Vagrant box to ensure the host system can create one correctly
==> centos-8 (vagrant): Creating Vagrant box for 'virtualbox' provider
centos-8 (vagrant): Copying from artifact: builds\packer-centos-vm-disk001.vmdk
centos-8 (vagrant): Copying from artifact: builds\packer-centos-vm.mf
centos-8 (vagrant): Copying from artifact: builds\packer-centos-vm.ovf
centos-8 (vagrant): Renaming the OVF to box.ovf...
centos-8 (vagrant): Compressing: Vagrantfile
centos-8 (vagrant): Compressing: box.ovf
centos-8 (vagrant): Compressing: metadata.json
centos-8 (vagrant): Compressing: packer-centos-vm-disk001.vmdk
centos-8 (vagrant): Compressing: packer-centos-vm.mf
Build 'centos-8' finished after 59 minutes 17 seconds.

==> Wait completed after 59 minutes 17 seconds

==> Builds finished. The artifacts of successful builds are:
--> centos-8: 'virtualbox' provider box: vagrant-centos-8-x86_64.box

C:\kartashova_alice\vagrant>packer.exe2
1Help      2UserIn    3View      4Edit      5Copy      6RenMov    7WkFold    8Delete    9ConfMn    10Quit
```

для начала автоматической установки образа операционной системы CentOS в VirtualBox и последующего формирования box-файла с дистрибутивом CentOS для VirtualBox.

Для регистрации образа виртуальной машины в vagrant в командной строке введем:

```
vagrant box add centos8 vagrant-centos-8-x86_64.box
```

```
For help on any individual command run `vagrant COMMAND -h`

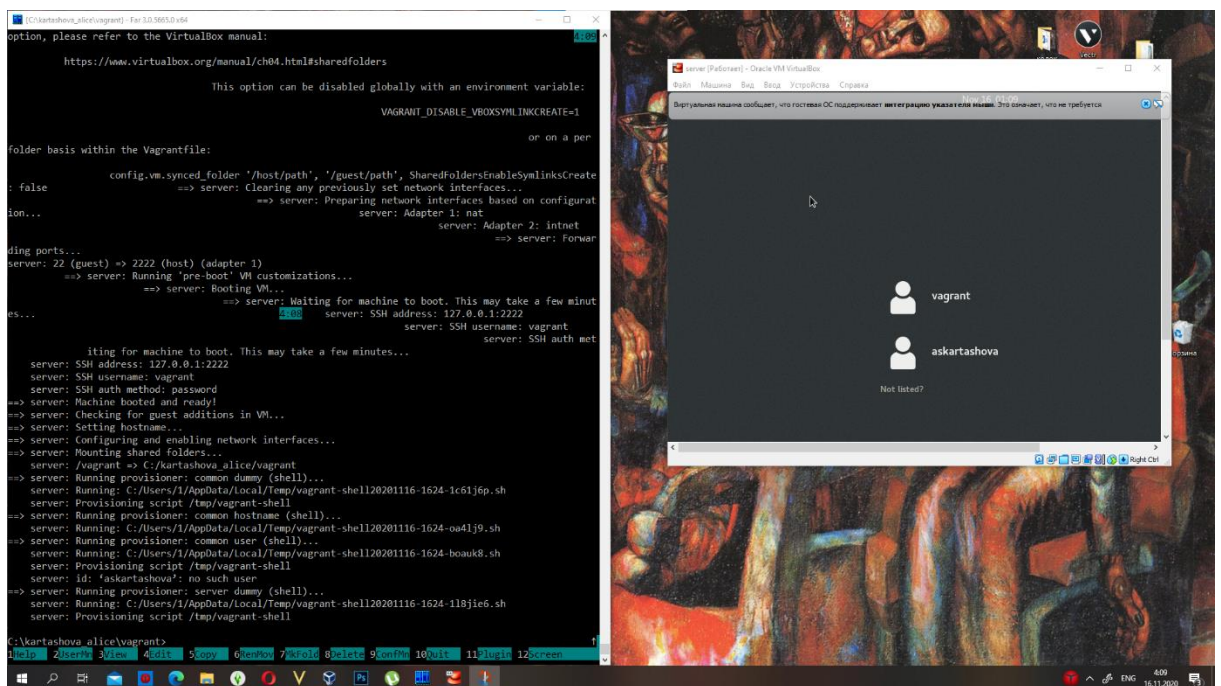
Additional subcommands are available, but are either more advanced
or not commonly used. To see all subcommands, run the command
`vagrant list-commands`.
--[no-]color           Enable or disable color output
--machine-readable     Enable machine readable output
-v, --version          Display Vagrant version
--debug               Enable debug output
--timestamp           Enable timestamps on log output
--debug-timestamp      Enable debug output with timestamps
--no-tty              Enable non-interactive output

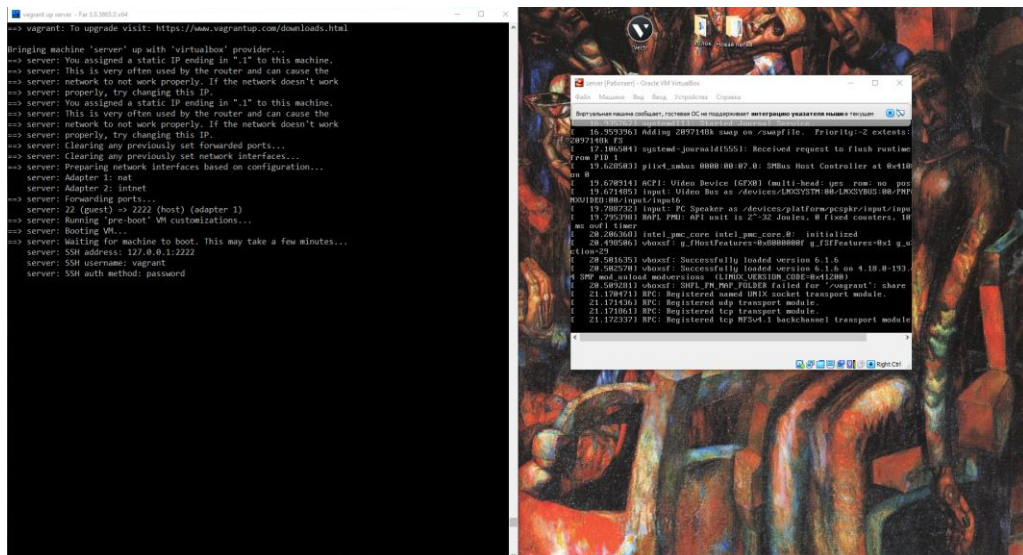
C:\kartashova_alice\vagrant>vagrant box add centos8 vagrant-centos-8-x86_64.box
==> box: Box file was not detected as metadata. Adding it directly...
==> box: Adding box 'centos8' (v0) for provider:
       box: Unpacking necessary files from: file://C:/kartashova_alice/vagrant/vagrant-centos-8-x86_64.box
       box:
==> box: Successfully added box 'centos8' (v0) for 'virtualbox'!

C:\kartashova_alice\vagrant>2.
```

Для запуска виртуальной машины Server введем в консоли

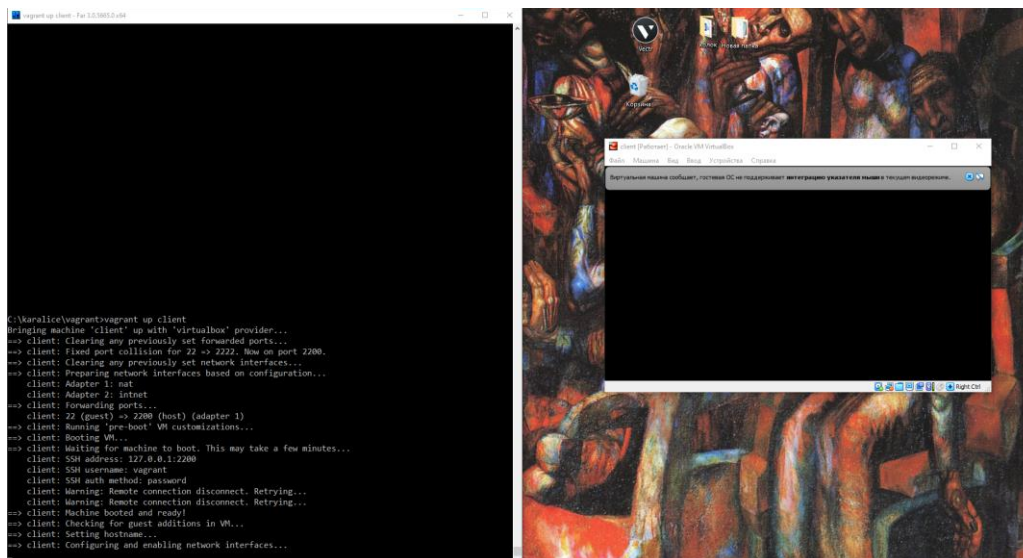
```
vagrant up server
```





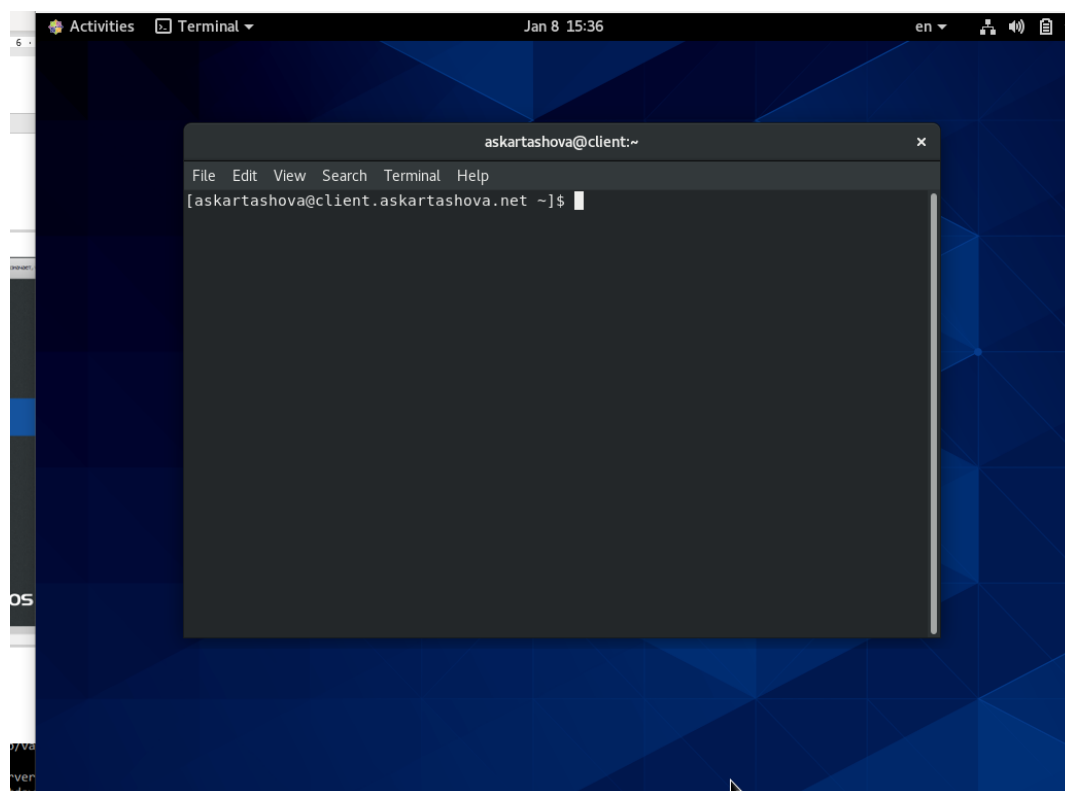
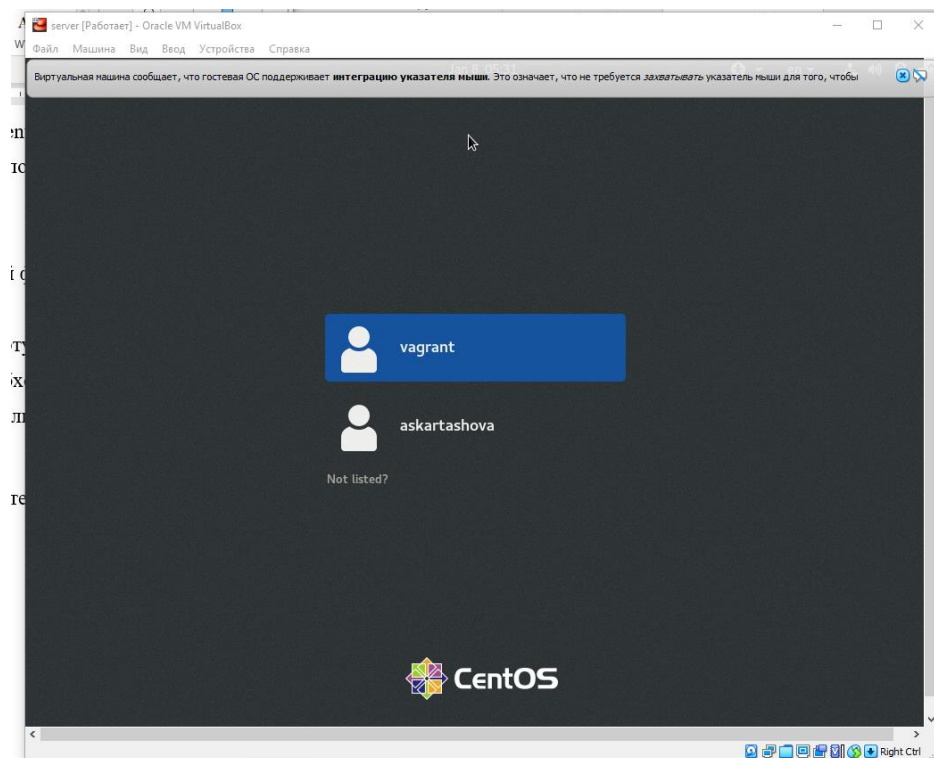
Для запуска виртуальной машины Client введем в консоли

vagrant up client



Убедитесь, что запуск обеих виртуальных машин прошёл успешно.

выключим виртуальные машины.



```
client: Provisioning script /tmp/vagrant-shell
C:\karalice\vagrant>vagrant halt server
==> server: Attempting graceful shutdown of VM...

C:\karalice\vagrant>vagrant halt client
==> client: Attempting graceful shutdown of VM...

C:\karalice\vagrant>
1Help 2UserMn 3View 4Edit 5Copy 6RenMov 7MkFold 8Delete 9ConfMn 10Quit 11Plugin 12Screen ↑
```

Внесение изменений в настройки внутреннего окружения виртуальной машины

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

```
6  # Common configuration
7  config.vm.provision "common dummy",
8    type: "shell",
9    preserve_order: true,
10   path: "provision/default/01-dummy.sh"
11
12  config.vm.provision "common hostname",
13    type: "shell",
14    preserve_order: true,
15    run: "always",
16    path: "provision/default/01-hostname.sh"
17
18  config.vm.provision "common user",
19    type: "shell",
20    preserve_order: true,
21    path: "provision/default/01-user.sh"
22
```

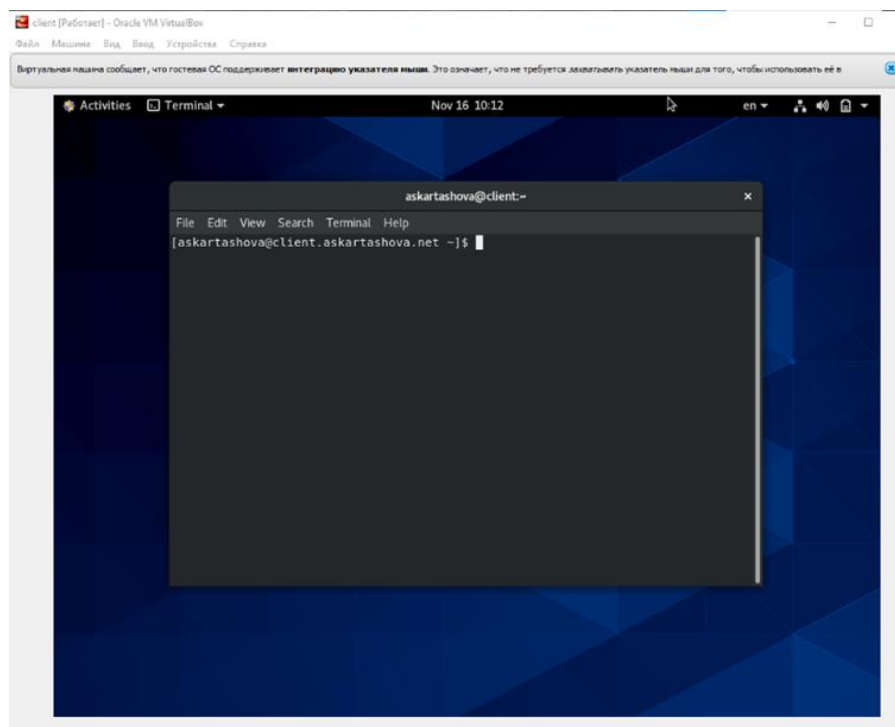
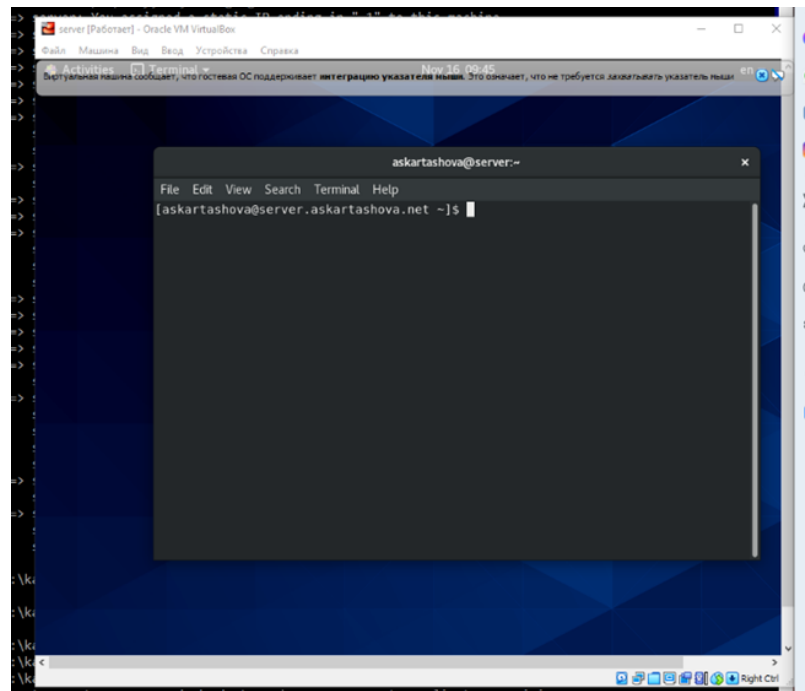
Зафиксируйте внесённые изменения для внутренних настроек виртуальных машин, введя в терминале:

vagrant up server --provision

vagrant up client --provision

Залогиньтесь на сервере и клиенте под созданным пользователем.

Убедитесь, что в терминале приглашение отображается в виде `user@server.askartashova.net` на сервере и в виде `user@client.askartashova.net` на клиенте.



Выключим виртуальные машины.

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

Заключение

Мы приобрели навыки установки CentOS на виртуальную машину

с помощью инструмента Vagrant.

Контрольные вопросы

Для чего предназначен Vagrant?

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

Что такое box-файл? В чём назначение Vagrantfile?

- box-файл — сохранённый образ виртуальной машины с развёрнутой в ней операционной системой, 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.

Приведем описание и примеры вызова основных команд Vagrant.

– `vagrant help` — вызов справки по командам Vagrant;

```
C:\karalice\vagrant>vagrant help
Usage: vagrant [options] <command> [<args>]

    -h, --help                Print this help.

Common commands:
    autocomplete    manages autocomplete installation on host
    box             manages boxes: installation, removal, etc.
    cloud           manages everything related to Vagrant Cloud
    destroy         stops and deletes all traces of the vagrant machine
    global-status   outputs status Vagrant environments for this user
    halt            stops the vagrant machine
    help            shows the help for a subcommand
    init            initializes a new Vagrant environment by creating a Vagrantfile
    login
    package         packages a running vagrant environment into a box
    plugin          manages plugins: install, uninstall, update, etc.
    port            displays information about guest port mappings
    powershell      connects to machine via powershell remoting
    provision        provisions the vagrant machine
    push            deploys code in this environment to a configured destination
    rdp             connects to machine via RDP
    reload          restarts vagrant machine, loads new Vagrantfile configuration
    resume          resume a suspended vagrant machine
    snapshot        manages snapshots: saving, restoring, etc.
    ssh             connects to machine via SSH
    ssh-config      outputs OpenSSH valid configuration to connect to the machine
    status          outputs status of the vagrant machine
    suspend         suspends the machine
    up              starts and provisions the vagrant environment
    upload          upload to machine via communicator
    validate        validates the Vagrantfile
    version         prints current and latest Vagrant version
    winrm           executes commands on a machine via WinRM
    winrm-config    outputs WinRM configuration to connect to the machine
```

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

– vagrant box add — подключение box-файла к Vagrant

```
C:\kartashova_alice\vagrant>vagrant box add centos8 vagrant-centos-8-x86_64.box
==> box: Box file was not detected as metadata. Adding it directly...
==> box: Adding box 'centos8' (v0) for provider:
box: Unpacking necessary files from: file:///C:/kartashova_alice/vagrant/vagrant-centos-8-x86_64.box
box:
==> box: Successfully added box 'centos8' (v0) for 'virtualbox'!
```

C:\kartashova_alice\vagrant>2

1Help 2UserMn 3View 4Edit 5Copy 6RenMov 7kFold

-vagrant global-status -показывает информации о всех известных Vagrant средах на этой машине

```
C:\karalice\vagrant>vagrant box list
centos8 (virtualbox, 0)

C:\karalice\vagrant>vagrant global-status
id      name  provider  state  directory
-----
72bdd44 client virtualbox poweroff C:/karalice/vagrant
8fceb39 server virtualbox poweroff C:/karalice/vagrant

The above shows information about all known Vagrant environments
on this machine. This data is cached and may not be completely
up-to-date (use "vagrant global-status --prune" to prune invalid
entries). To interact with any of the machines, you can go to that
directory and run Vagrant, or you can use the ID directly with
Vagrant commands from any directory. For example:
"vagrant destroy 1a2b3c4d"
```

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

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

– vagrant reload — перезагрузка виртуальной машины;

```

C:\karalice\vagrant>vagrant reload client
==> client: Attempting graceful shutdown of VM...
==> client: Forcing shutdown of VM...
==> client: Clearing any previously set forwarded ports...
==> client: Fixed port collision for 22 => 2222. Now on port 2200.
==> client: Clearing any previously set network interfaces...
==> client: Preparing network interfaces based on configuration...
      client: Adapter 1: nat
      client: Adapter 2: intnet
==> client: Forwarding ports...
      client: 22 (guest) => 2200 (host) (adapter 1)
==> client: Running 'pre-boot' VM customizations...
==> client: Booting VM...
==> client: Waiting for machine to boot. This may take a few minutes...
      client: SSH address: 127.0.0.1:2200
      client: SSH username: vagrant
      client: SSH auth method: password
==> client: Machine booted and ready!
==> client: Checking for guest additions in VM...
==> client: Setting hostname...
==> client: Configuring and enabling network interfaces...
==> client: Mounting shared folders...
      client: /vagrant => C:/karalice/vagrant
==> client: Machine already provisioned. Run `vagrant provision` or use the `--provision`
==> client: flag to force provisioning. Provisioners marked to run always will still run.
==> client: Running provisioner: common hostname (shell)...
      client: Running: C:/Users/1/AppData/Local/Temp/vagrant-shell20210108-14836-u9brfs.sh
      client: Provisioning script /tmp/vagrant-shell

```

– vagrant halt — остановка и выключение виртуальной машины;

```

      client: Provisioning script /tmp/vagrant-shell
C:\karalice\vagrant>vagrant halt server
==> server: Attempting graceful shutdown of VM...

C:\karalice\vagrant>vagrant halt client
==> client: Attempting graceful shutdown of VM...

C:\karalice\vagrant>
1Help 2UserMn 3View 4Edit 5Copy 6RenMov 7WkFold 8Delete 9ConfMn 10Quit 11Plugin 12Screen ↑

```

– vagrant provision — настройка внутреннего окружения имеющейся виртуальной машины

Дайте построчные пояснения содержания файлов vagrant-centos.json, ks.cfg, Vagrantfile.

Описание Vagrantfile

Первые две строки указывают на режим работы с Vagrantfile и на использование языка Ruby.

Vagrant.configure(2) do |config| - войдем в главную конфигурацию vagrant версии 2. Цикл do, заменяет конструкцию Vagrant.configure на config.

Параметры в config.vm изменяют конфигурацию машины, которой управляет Vagrant.

config.vm.provision - настраивает средства обеспечения на машине, чтобы программное

обеспечение могло быть автоматически установлено и настроено при создании машины.

`config.vm.hostname` (строка) - Имя хоста, которое должен иметь компьютер. По умолчанию ноль. Если ноль, Vagrant не будет управлять именем хоста. Если установлено в виде строки, имя хоста будет установлено при загрузке. Если установлено, Vagrant обновит `/etc/hosts` на гостевой машине с настроенным именем хоста.

Vagrantfile — C:\karalice\vagrant — Atom

File Edit View Selection Find Packages Help



Welcome Guide

Vagrantfile

```
1  # -*- mode: ruby -*-
2  # vi: set ft=ruby :
3
4  Vagrant.configure("2") do |config|
5
6    # Common configuration
7    config.vm.provision "common user",
8      type: "shell",
9      preserve_order: true,
10     path: "provision/default/01-user.sh"
11
12    config.vm.provision "common hostname",
13      type: "shell",
14      preserve_order: true,
15      run: "always",
16      path: "provision/default/01-hostname.sh"
17
18    config.vm.provision "common user",
19      type: "shell",
20      preserve_order: true,
21      path: "provision/default/01-user.sh"
22
23    # Server configuration
24
25    config.vm.define "server", autostart: false do |server|
26      server.vm.box = "centos8"
27      server.vm.hostname = 'server'
28
29      server.ssh.insert_key = false
30      server.ssh.username = 'vagrant'
31      server.ssh.password = 'vagrant'
32
33      server.vm.network :private_network, ip: "192.168.1.1", virtualbox____intnet: true
34
```