

Отчёт по лабораторной работе №1

Дисциплина: Администрирование сетевых подсистем

Ибрахим Мухсейн Алькамаль

Содержание

1 Цель работы	5
2 Выполнение лабораторной работы	6
3 Развёртывание лабораторного стенда на ОС Windows	12
4 Внесение изменений в настройки внутреннего окружения виртуальной машины	17
5 Выводы	21
6 Контрольные вопросы:	22
7 Список литературы	24

Список иллюстраций

2.1	Структура рабочего каталога с подкаталогами packer и vagrant	6
2.2	Содержимое каталога packer с ISO-образом и HCL-файлом	6
2.3	Файл vagrant-rocky.pkr.hcl с описанием переменных и плагинов	7
2.4	Файл ks.cfg с параметрами автоматической установки Rocky Linux	8
2.5	Файл Vagrantfile с конфигурацией виртуальной машины server	9
2.6	Структура каталога provision с подкаталогами default, server и client	9
2.7	Скрипты-заглушки 01-dummy.sh в каталогах default, server и client	10
2.8	Скрипты-заглушки 01-dummy.sh в каталогах default, server и client	10
2.9	Скрипты-заглушки 01-dummy.sh в каталогах default, server и client	10
2.10	Скрипт 01-user.sh для создания пользователя и настройки окружения	10
2.11	Скрипт 01-hostname.sh для установки доменного имени хоста	11
2.12	Скрипт 02-forward.sh для включения IP-forwarding и masquerading	11
2.13	Скрипт 01-routing.sh для настройки маршрутизации клиента	11
3.1	Выполнение команд packer.exe init и packer.exe build	12
3.2	Сформированный box-файл vagrant-virtualbox-rocky-9-x86_64.box	13
3.3	Регистрация box-файла в Vagrant	13
3.4	Запуск виртуальной машины server через vagrant up	13
3.5	Запуск виртуальной машины client через vagrant up	14
3.6	Графический вход в систему на виртуальной машине server	14
3.7	Графический вход в систему на виртуальной машине client	15
3.8	Подключение по SSH к server и переход к пользователю alkamal	15
3.9	Подключение по SSH к client и переход к пользователю alkamal	15
3.10	Остановка виртуальных машин server и client	16
4.1	Фрагмент Vagrantfile с блоками common user и common hostname	17
4.2	Повторный запуск server с ключом –provision	18
4.3	Повторный запуск client с ключом –provision	18
4.4	Графический вход под пользователем alkamal на сервере	19
4.5	Графический вход под пользователем alkamal на клиенте	19
4.6	Проверка SSH-подключения и отображения приглашения пользователя	20

Список таблиц

1 Цель работы

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

2 Выполнение лабораторной работы

В рабочем каталоге ОС Windows созданы подкаталоги C:\work\alkamal\packer и C:\work\alkamal\vagrant, предназначенные для подготовки box-файла Vagrant (рис. 2.1). В каталоге packer размещены образ Rocky-9.4-x86_64-minimal.iso, исполняемый файл packer.exe, файл конфигурации vagrant-rocky.pkr.hcl и подкаталог http, что соответствует требованиям к структуре проекта (рис. 2.2).

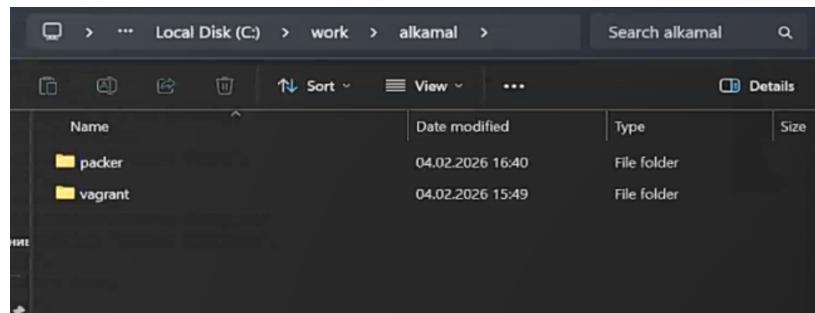


Рисунок 2.1: Структура рабочего каталога с подкаталогами packer и vagrant

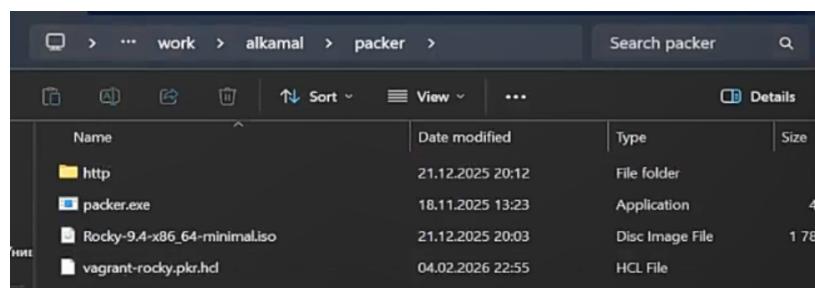


Рисунок 2.2: Содержимое каталога packer с ISO-образом и HCL-файлом

Файл `vagrant-rocky.pkr.hcl` содержит блок `packer` с разделом `required_plugins` (`vagrant`, `virtualbox`), а также переменные, определяющие версию дистрибутива, размер диска, контрольную сумму ISO-образа, архитектуру и параметры SSH (рис. 2.3). В подкаталоге `http` размещён файл `ks.cfg`, задающий автоматизированные параметры установки системы: загрузчик, очистку разделов, язык, сетевые настройки (DHCP), создание пользователя и `post`-настройки (рис. 2.4).



```
C: > work > alkamal > packer > vagrant-rocky.pkr.hcl
 1  packer {
 2    required_plugins {
 3      vagrant = {
 4        source = "github.com/hashicorp/vagrant"
 5        version = "~> 1"
 6      }
 7      virtualbox = {
 8        version = "~> 1"
 9        source = "github.com/hashicorp/virtualbox"
10      }
11    }
12  }
13  variable "artifact_description" {
14    type = string
15    default = "Rocky 9.4"
16  }
17  variable "artifact_version" {
18    type = string
19    default = "9.4"
20  }
21  variable "disk_size" {
22    type = string
23    default = "61440"
24  }
25  variable "iso_checksum" {
26    type = string
27    default = "ee3ac97fdffab58652421941599902012179c37535aece76824673105169c4a2"
28  }
29  variable "iso_checksum_type" {
30    type = string
31    default = "sha256"
32  }
33  variable "iso_url" {
34    type = string
35    default = "Rocky-9.4-x86_64-minimal.iso"
36  }
37  variable "redhat_platform" {
38    type = string
39    default = "x86_64"
40  }
41  variable "redhat_release" {
42    type = string
43    default = "9"
44  }
45  variable "ssh_password" {
46    type = string
47    default = "vagrant"
```

Рисунок 2.3: Файл `vagrant-rocky.pkr.hcl` с описанием переменных и плагинов

```

C: > work > alkamal > packer > http > ks.cfg
 1 # System bootloader configuration
 2 bootloader --append="no_timer_check console=tty0 console=ttyS0,115200n8net.ifnames=0
 3 # Clear the Master Boot Record
 4 zerombr
 5 # Partition clearing information
 6 clearpart --all
 7 # Reboot after installation
 8 reboot
 9 # Use text mode install
10 text
11 # Keyboard layouts
12 keyboard --vkeymap=us,ru --xlayouts='us,ru'
13 # System language
14 lang en_US.UTF-8
15 # Network information
16 network --bootproto=dhcp --device=link --activate
17 # System authorization information
18 authselect select ssd with-sudo with-mkhomedir --force
19 authselect apply-changes
20 # Root password
21 rootpw vagrant
22 user --name=vagrant --password=vagrant
23 firstboot --disable
24 # Do not configure the X Window System
25 #skipx
26 # System services
27 services --enabled="NetworkManager,sshd,chronyd"
28 # System timezone
29 timezone UTC --utc
30 user --name=vagrant --password=vagrant
31 # Disk partitioning information
32 part / --fstype="xfs" --size=10239
33 %post
34 # configure swap to a file
35 fallocate -l 2G /swapfile
36 chmod 600 /swapfile
37 mkswap /swapfile
38 echo "/swapfile none swap defaults 0 0" >> /etc/fstab
39 # sudo
40 echo "%vagrant ALL=(ALL) NOPASSWD: ALL" > /etc/sudoers.d/vagrant
41 chmod 0440 /etc/sudoers.d/vagrant
42 # Fix for https://github.com/CentOS/sig-cloud-instance-build/issues/38
43 cat > /etc/sysconfig/network-scripts/ifcfg-eth0 << EOF
44 DEVICE="eth0"
45 BOOTPROTO="dhcp"
46 ONBOOT="yes"
47 TYPE="Ethernet"

```

Рисунок 2.4: Файл ks.cfg с параметрами автоматической установки Rocky Linux

В каталоге `vagrant` размещён файл `Vagrantfile`, определяющий конфигурацию виртуальной машины `server`: имя `box` `rocky9`, сетевой интерфейс с IP `192.168.1.1`, параметры `VirtualBox` (RAM, CPU, VRAM, графический контроллер) и подключение provision-скриптов (рис. 2.5). Также создан каталог `provision` с подкаталогами `default`, `server` и `client`, предназначенными для разделения сценариев настройки окружения (рис. 2.6).

```

C: > work > alkamal > vagrant > Vagrantfile
1  # -*- mode: ruby -*-
2  # vi: set ft=ruby :
3  Vagrant.configure("2") do |config|
4    ## Common configuration
5    config.vm.provision "common dummy",
6      type: "shell",
7      preserve_order: true,
8      path: "provision/default/01-dummy.sh"
9    config.vm.provision "common hostname",
10   type: "shell",
11   preserve_order: true,
12   run: "always",
13   path: "provision/default/01-hostname.sh"
14   config.vm.provision "common user",
15     type: "shell",
16     preserve_order: true,
17     path: "provision/default/01-user.sh"
18
19   ## Server configuration
20   config.vm.define "server", autostart: false do |server|
21     server.vm.box = "rocky9"
22     server.vm.hostname = 'server'
23     server.vm.boot_timeout = 1440
24     server.ssh.insert_key = false
25     server.ssh.username = 'vagrant'
26     server.ssh.password = 'vagrant'
27     server.vm.network :private_network,
28       ip: "192.168.1.1",
29     virtualbox_intnet: true
30     server.vm.provision "server dummy",
31       type: "shell",
32       preserve_order: true,
33       path: "provision/server/01-dummy.sh"
34     server.vm.provider :virtualbox do |v|
35       v.linked_clone = true
36     # Customize the amount of memory on the VM
37     v.memory = 1024
38     v.cpus = 1
39     v.name = "server"
40     # Display the VirtualBox GUI when booting the machine
41     v.gui = true
42     # Set the video memory to 12Mb
43     v.customize ["modifyvm", :id, "--vram", "12"]
44     v.customize ["modifyvm", :id, "--natdnshostresolver1", "on"]
45     v.customize ["modifyvm", :id, "--clipboard", "bidirectional"]
46     v.customize ["modifyvm", :id, "--draganddrop", "bidirectional"]
47     v.customize ["modifyvm", :id, "--graphicscontroller", "vmsvga"]
48

```

Рисунок 2.5: Файл Vagrantfile с конфигурацией виртуальной машины server

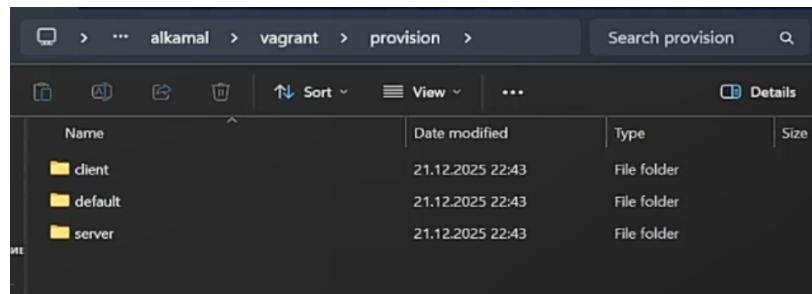
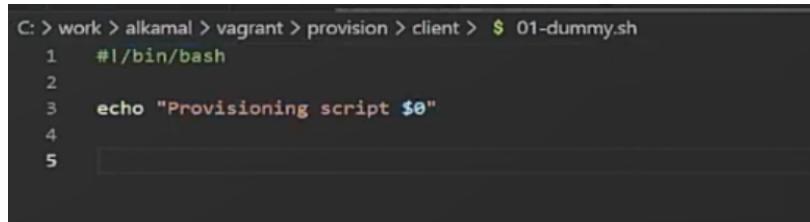


Рисунок 2.6: Структура каталога provision с подкаталогами default, server и client

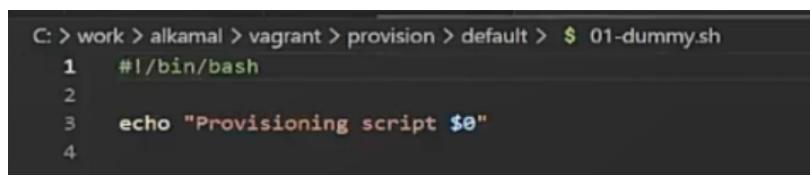
В подкаталогах `default`, `server` и `client` размещены скрипты-заглушки `01-dummy.sh`, выводящие сообщение о выполнении provisioning (рис. 2.9). В каталоге `default` дополнительно размещены скрипты `01-user.sh`

(создание пользователя alkamal, добавление в группу wheel, настройка PS1) и 01-hostname.sh (установка FQDN вида *.alkamal.net) (рис. 2.10, рис. 2.11). В каталоге server размещён скрипт 02-forward.sh, активирующий IP-маршрутизацию и masquerading через firewalld, а в каталоге client – скрипт 01-routing.sh, настраивающий шлюз 192.168.1.1 и параметры интерфейсов через nmcli (рис. 2.12, рис. 2.13).



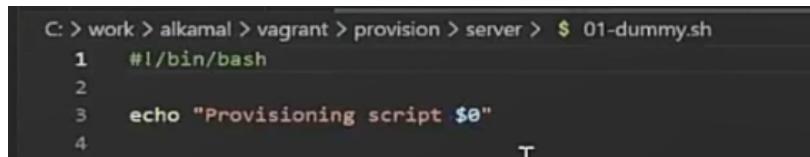
```
C: > work > alkamal > vagrant > provision > client > $ 01-dummy.sh
1  #!/bin/bash
2
3  echo "Provisioning script $0"
4
5
```

Рисунок 2.7: Скрипты-заглушки 01-dummy.sh в каталогах default, server и client



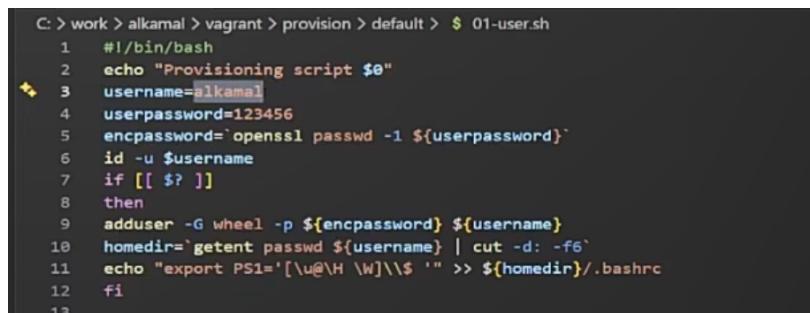
```
C: > work > alkamal > vagrant > provision > default > $ 01-dummy.sh
1  #!/bin/bash
2
3  echo "Provisioning script $0"
4
```

Рисунок 2.8: Скрипты-заглушки 01-dummy.sh в каталогах default, server и client



```
C: > work > alkamal > vagrant > provision > server > $ 01-dummy.sh
1  #!/bin/bash
2
3  echo "Provisioning script $0"
4
```

Рисунок 2.9: Скрипты-заглушки 01-dummy.sh в каталогах default, server и client



```
C: > work > alkamal > vagrant > provision > default > $ 01-user.sh
1  #!/bin/bash
2  echo "Provisioning script $0"
3  username=alkama]
4  userpassword=123456
5  encpassword=`openssl passwd -1 ${userpassword}`
6  id -u $username
7  if [[ $? ]]
8  then
9    adduser -G wheel -p ${encpassword} ${username}
10   homedir=`getent passwd ${username} | cut -d: -f6`
11   echo "export PS1='[\u@\H \W]\$\'' >> ${homedir}/.bashrc
12 fi
```

Рисунок 2.10: Скрипт 01-user.sh для создания пользователя и настройки окружения

```
C: > work > alkamal > vagrant > provision > default > $ 01-hostname.sh
1  #!/bin/bash
2
3  username=alkamal
4  hostnamectl set-hostname "${HOSTNAME%.*}}.${username}.net
```

Рисунок 2.11: Скрипт 01-hostname.sh для установки доменного имени хоста

```
C: > work > alkamal > vagrant > provision > server > $ 02-forward.sh
1  #!/bin/bash
2  echo "Provisioning script $0"
3  echo "Enable forwarding"
4  echo "net.ipv4.ip_forward = 1" > /etc/sysctl.d/90-forward.conf
5  sysctl -w net.ipv4.ip_forward=1
6  echo "Configure masquerading"
7  firewall-cmd --add-masquerade --permanent
8  firewall-cmd --reload
9  restorecon -vR /etc
10
```

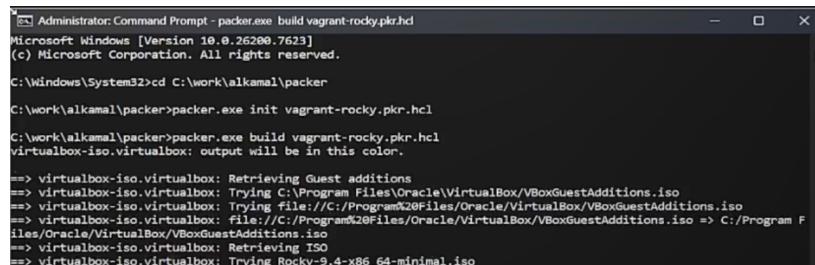
Рисунок 2.12: Скрипт 02-forward.sh для включения IP-forwarding и masquerading

```
C: > work > alkamal > vagrant > provision > client > $ 01-routing.sh
1  !/bin/bash
2  echo "Provisioning script $0"
3  nmcli connection modify "System eth1" ipv4.gateway "192.168.1.1"
4  nmcli connection up "System eth1"
5  nmcli connection modify eth0 ipv4.never-default true
6  nmcli connection modify eth0 ipv6.never-default true
7  nmcli connection down eth0
8  nmcli connection up eth0
9
```

Рисунок 2.13: Скрипт 01-routing.sh для настройки маршрутизации клиента

3 Развёртывание лабораторного стенда на ОС Windows

В рабочем каталоге C:\work\alkamal\packer выполнены команды packer.exe init vagrant-rocky.pkr.hcl и packer.exe build vagrant-rocky.pkr.hcl, в результате чего произведена автоматическая установка Rocky Linux в VirtualBox и сформирован box-файл (рис. 3.1). По завершении сборки в каталоге создан файл vagrant-virtualbox-rocky-9-x86_64.box, подтверждающий корректное формирование образа для Vagrant (рис. 3.2).



```
Administrator: Command Prompt - packer.exe build vagrant-rocky.pkr.hcl
Microsoft Windows [Version 10.0.26290.7623]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>cd C:\work\alkamal\packer
C:\work\alkamal\packer>packer.exe init vagrant-rocky.pkr.hcl
C:\work\alkamal\packer>packer.exe build vagrant-rocky.pkr.hcl
virtualbox-iso.virtualbox: output will be in this color.

==> virtualbox-iso.virtualbox: Retrieving Guest additions
==> virtualbox-iso.virtualbox: Trying C:\Program Files\Oracle\VirtualBox\VBoxGuestAdditions.iso
==> virtualbox-iso.virtualbox: Trying file:///C:/Program%20Files/Oracle/VirtualBox/VBoxGuestAdditions.iso
==> virtualbox-iso.virtualbox: file:///C:/Program%20Files/Oracle/VirtualBox/VBoxGuestAdditions.iso => C:/Program F
iles/Oracle/VirtualBox/VBoxGuestAdditions.iso
==> virtualbox-iso.virtualbox: Retrieving ISO
==> virtualbox-iso.virtualbox: Trying Rocky-9.4-x86_64-minimal.iso
```

Рисунок 3.1: Выполнение команд packer.exe init и packer.exe build

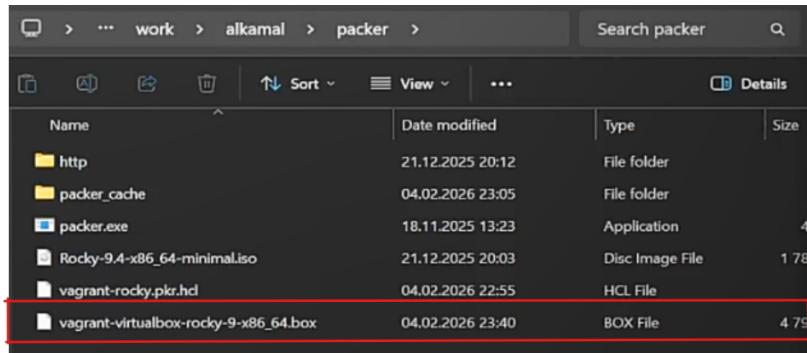


Рисунок 3.2: Сформированный box-файл vagrant-virtualbox-rocky-9-x86_64.box

Далее выполнена регистрация образа в Vagrant командой `vagrant box add rocky9 vagrant-virtualbox-rocky-9-x86_64.box`, после чего box добавлен под именем `rocky9` (рис. 3.3). Запуск виртуальной машины Server осуществлён командой `vagrant up server`, выполнена инициализация сетевых интерфейсов, проброс портов и запуск provisioning-сценариев (рис. 3.4).

```
C:\work\alkamal\packer>vagrant box add rocky9 vagrant-virtualbox-rocky-9-x86_64.box
--> box: Box was not detected as metadata. Adding it directly...
--> box: Adding box 'rocky9' (v8) for provider: (amd64)
--> box: Unpacking necessary files from: file:///C:/work/alkamal/packer/vagrant-virtualbox-rocky-9-x86_64.box
--> box:
```

Рисунок 3.3: Регистрация box-файла в Vagrant

```
C:\work\alkamal\vagrant>vagrant up server
Bringing machine 'server' up with 'virtualbox' provider...
--> server: You assigned a static IP ending in ".1" or ":1" to this machine.
--> server: This is very often used by the router and can cause the
--> server: network to not work properly. If the network doesn't work
--> server: properly, try changing this IP.
--> server: You assigned a static IP ending in ".1" or ":1" to this machine.
--> server: This is very often used by the router and can cause the
--> server: network to not work properly. If the network doesn't work
--> server: properly, try changing this IP.
--> server: Clearing any previously set forwarded ports...
--> server: Clearing any previously set network interfaces...
--> server: Preparing network interfaces based on configuration...
--> server: Adapter 1: nat
--> server: Adapter 2: intnet
--> server: Forwarding ports...
--> server: 22 (guest) => 2222 (host) (adapter 1)
--> server: Running 'pre-boot' VM customizations...
--> server: Booting VM...
--> server: Waiting for machine to boot. This may take a few minutes...
--> server: SSH address: 127.0.0.1:2222
--> server: SSH username: vagrant
--> server: SSH auth method: password
--> server: Machine booted and ready!
--> server: Checking for guest additions in VM...
--> server: Setting hostname...
--> server: Configuring and enabling network interfaces...
--> server: Mounting shared folders...
--> server: C:/work/alkamal/vagrant => /vagrant
--> server: Machine already provisioned. Run 'vagrant provision' or use the '--provision'
--> server: flag to force provisioning. Provisions marked to run always will still run.
--> server: Running provisioner: common hostname (shell)...
--> server: Running: C:/Users/EBRAHIM/AppData/Local/Temp/vagrant-shell120260206-26712-p9lxpn.sh
```

Рисунок 3.4: Запуск виртуальной машины server через `vagrant up`

Запуск виртуальной машины Client выполнен командой `vagrant up client`, произведена настройка сетевых адаптеров и подключение через SSH

(рис. 3.5). В графическом интерфейсе VirtualBox обе виртуальные машины успешно загружены, выполнен вход под пользователем vagrant с паролем vagrant (рис. 3.6, рис. 3.7).

```
C:\work\alkamal>vagrant>vagrant up client
Bringing machine `client` up with 'virtualbox' provider...
--> client: Cloning VM...
--> client: Matching MAC address for NAT networking...
--> client: Setting the name of the VM: client ...
--> 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: No guest additions were detected on the base box for this VM! Guest
--> client: additions are required for forwarded ports, shared folders, host only
--> client: networking, and more. If SSH fails on this machine, please install
--> client: the guest additions and repackage the box to continue.
--> client: This is not an error message; everything may continue to work properly.
-->> client: Setting hostname...
-->> client: Configuring and enabling network interfaces...
-->> client: Mounting shared folders...
-->> client: C:/work/alkamal/vagrant => /vagrant
Vagrant was unable to mount VirtualBox shared folders. This is usually
because the filesystem 'vboxsf' is not available. This filesystem is
made available via the VirtualBox Guest Additions and kernel module.
Please verify that these guest additions are properly installed in the
guest. This is not a bug in Vagrant and is usually caused by a faulty
Vagrant box. For context, the command attempted was:
mount -t vboxsf -o uid=1000,gid=1000,_netdev vagrant /vagrant
The error output from the command was:
mount: can't find /vagrant in /etc/fstab
```

Рисунок 3.5: Запуск виртуальной машины client через vagrant up

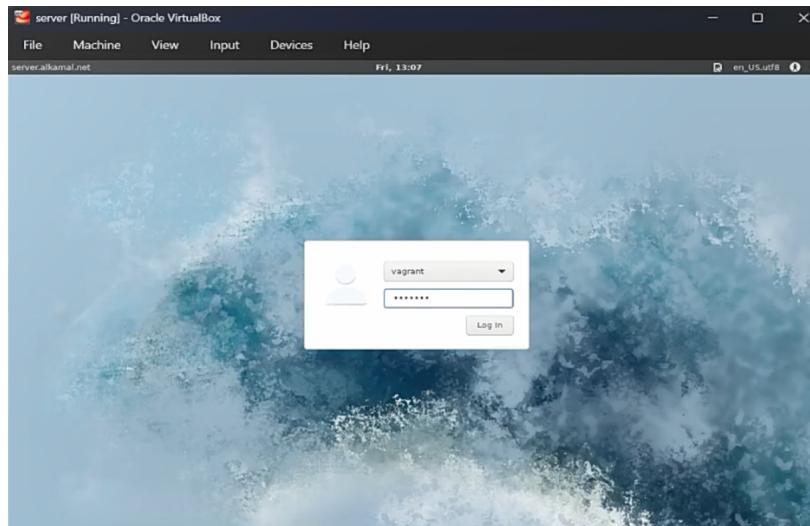


Рисунок 3.6: Графический вход в систему на виртуальной машине server

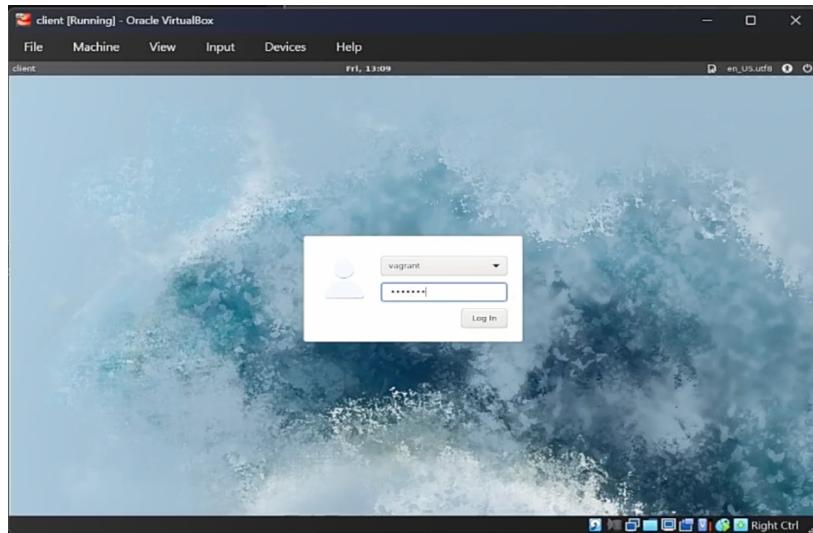


Рисунок 3.7: Графический вход в систему на виртуальной машине client

Подключение к серверу выполнено командой `vagrant ssh server`, после ввода пароля осуществлён переход к пользователю `alkamal` командой `su - alkamal` и последующий выход из сеанса (рис. 3.8). Аналогичные действия выполнены для клиента с использованием команды `vagrant ssh client` (рис. 3.9).

```
C:\work\alkamal\vagrant>vagrant ssh server
==> server: The machine you're attempting to SSH into is configured to use
==> server: password-based authentication. Vagrant can't script entering the
==> server: password for you. If you're prompted for a password, please enter
==> server: the same password you have configured in the Vagrantfile.
vagrant@127.0.0.1's password:
Last login: Fri Feb  6 13:07:21 2026
[vagrant@server ~]$ su - alkamal
Password:
[alkamal@server alkamal.net ~]$ logout
[vagrant@server ~]$ exit
logout
```

Рисунок 3.8: Подключение по SSH к server и переход к пользователю alkamal

```
C:\work\alkamal\vagrant>vagrant ssh client
==> client: The machine you're attempting to SSH into is configured to use
==> client: password-based authentication. Vagrant can't script entering the
==> client: password for you. If you're prompted for a password, please enter
==> client: the same password you have configured in the Vagrantfile.
vagrant@127.0.0.1's password:
Last login: Fri Feb  6 13:29:46 2026
[vagrant@client ~]$ su - alkamal
Password:
[alkamal@client alkamal.net ~]$ logout
[vagrant@client ~]$ su - alkamal
Password:
Last login: Fri Feb  6 13:31:20 UTC 2026 on pts/0
[alkamal@client alkamal.net ~]$ logout
[vagrant@client ~]$ exit
logout
```

Рисунок 3.9: Подключение по SSH к client и переход к пользователю alkamal

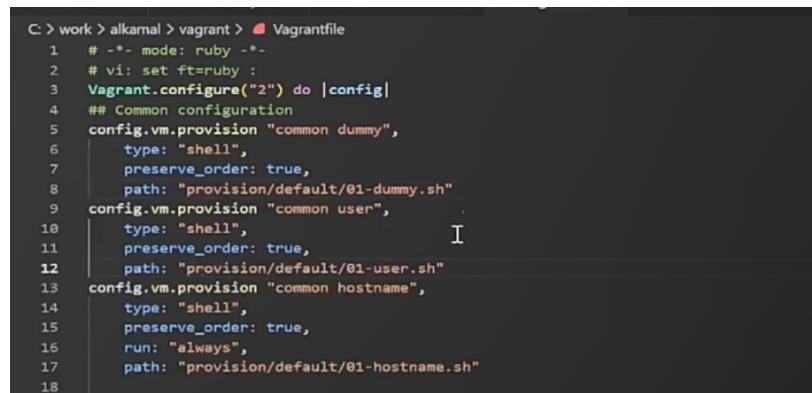
Завершение работы лабораторного стенда выполнено командами `vagrant halt server` и `vagrant halt client`, что обеспечило корректное выключение обеих виртуальных машин (рис. 3.10).

```
C:\work\alkamal\vagrant>vagrant halt server  
--> server: Attempting graceful shutdown of VM...  
C:\work\alkamal\vagrant>vagrant halt client  
--> client: Attempting graceful shutdown of VM...
```

Рисунок 3.10: Остановка виртуальных машин `server` и `client`

4 Внесение изменений в настройки внутреннего окружения виртуальной машины

В конфигурационном файле `Vagrantfile` перед разделом с описанием виртуальной машины `server` добавлены provisioning-блоки `common user` и `common hostname`, обеспечивающие выполнение скриптов `01-user.sh` и `01-hostname.sh` при загрузке ВМ (рис. 4.1). Это гарантирует создание пользователя `alkamal` и установку доменного имени вида `*.alkamal.net`.



```
C: > work > alkamal > vagrant > Vagrantfile
1  # -*- mode: ruby -*-
2  # vi: set ft=ruby :
3  Vagrant.configure("2") do |config|
4    ## Common configuration
5    config.vm.provision "common dummy",
6      type: "shell",
7      preserve_order: true,
8      path: "provision/default/01-dummy.sh"
9    config.vm.provision "common user",
10      type: "shell",
11      preserve_order: true,
12      path: "provision/default/01-user.sh"
13    config.vm.provision "common hostname",
14      type: "shell",
15      preserve_order: true,
16      run: "always",
17      path: "provision/default/01-hostname.sh"
18
```

Рисунок 4.1: Фрагмент `Vagrantfile` с блоками `common user` и `common hostname`

Для применения изменений выполнены команды `vagrant up server --provision` и `vagrant up client --provision`, в результате чего повторно запущены provisioning-сценарии и зафиксированы внутренние

настройки виртуальных машин (рис. 4.2, рис. 4.3). В процессе выполнения на сервере зафиксировано существование пользователя alkamal, что подтверждает корректную работу скрипта создания пользователя.

```
C:\work\alkalmal>vagrant up server --provision
Bringing machine 'server' up with 'virtualbox' provider...
=> server: You assigned a static IP ending in ".1" or ":1" to this machine.
=> server: This is very often used by the router and can cause the
=> server: network to not work properly. If the network doesn't work
=> server: properly, try changing this IP.
=> server: You assigned a static IP ending in ".1" or ":1" to this machine.
=> server: This is very often used by the router and can cause the
=> server: network to not work properly. If the network doesn't work
=> server: properly, try changing this IP.
=> server: Cleaning any previously set forwarded ports...
=> server: Clearing any previously set network interfaces...
=> server: Preparing network interfaces based on configuration...
=> server: Adapter 1: nat
=> server: Adapter 2: intnet
=> server: Forwarding ports...
=> server: 22 (guest) => 2222 (host) (adapter 1)
=> server: Running 'pre-boot' VM customizations...
=> server: Booting VM...
=> server: Waiting for machine to boot. This may take a few minutes...
=> server: SSH address: 127.0.0.1:2222
=> server: SSH username: vagrant
=> server: SSH auth method: password
=> server: Machine booted and ready!
=> server: Checking for guest additions in VM...
=> server: Setting hostname...
=> server: Configuring and enabling network interfaces...
=> server: Mounting shared folders...
=> server: C:/work/alkalmal/vagrant => /vagrant
=> server: Running provisioner: common_dummy (shell)...
=> server: Running: C:/Users/EBRAHIMI/AppData/Local/Temp/vagrant-shell20260206-12488-1ljrnr.sh
=> server: Provisioning script /tmp/vagrant-shell
=> server: Running provisioner: common_user (shell)...
=> server: Running: C:/Users/EBRAHIMI/l/AppData/Local/Temp/vagrant-shell20260206-12488-52bd7.e7.sh
=> server: Provisioning script /tmp/vagrant-shell
=> server: 1901
=> server: adduser: user 'alkalmal' already exists
=> server: Running provisioner: common_hostname (shell)...
=> server: Running: C:/Users/EBRAHIMI/l/AppData/Local/Temp/vagrant-shell20260206-12488-vbwn21.sh
=> server: Running provisioner: server_dummy (shell)...
=> server: Running: C:/Users/EBRAHIMI/l/AppData/Local/Temp/vagrant-shell20260206-12488-z51ko.sh
=> server: Provisioning script /tmp/vagrant-shell
```

Рисунок 4.2: Повторный запуск server с ключом –provision

```
[root@vagrant ~]# vagrant up client --provision
Bringing machine 'client' up with 'virtualbox' provider...
-> 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: Predefining network interfaces based on configuration...
-> Client: Adapter 1: nat
-> Client: Adapter 2: intnet
-> Client: Forwarding ports...
  (guest) 22/tcp (host) (adapter 1)
-> Client: Running 'pre-boot' VM customizations...
-> Client: Booting VM...
-> Client: Waiting for machine to boot. This may take a few minutes...
  SSH address: 127.0.0.1:2200
  SSH auth method: password
  client: Warning: Remote connection disconnect. Retrying...
  client: Warning: Remote connection disconnect. Retrying...
-> Client: Machine booted successfully. Ready!
-> Client: Checking for guest additions in VM...
-> Client: Setting hostname...
-> Client: Configuring and enabling network interfaces...
-> Client: Copying files back from host...
  C:/work/vagrant/vagrant -> /vagrant
-> Client: Running provisioner: common_dummy (shell)...
  Client: Running: C:/Users/EBRAHIMI/AppData/Local/Temp/vagrant-shell20260206-21276-7lb3n9.sh
-> Client: Running provisioner: common_user (shell)...
  Client: Running: C:/Users/EBRAHIMI/AppData/Local/Temp/vagrant-shell20260206-21276-y7parv.sh
  Client: Provisioning script '/tmp/vagrant-shell'
  Client: 100%|██████████| 0/0 [00:00<00:00, 00.00 MiB/s]
-> Client: Address 'user' 'okamal' already exists
  Client: Running provisioner: common_hostname (shell)...
  Client: Running: C:/Users/EBRAHIMI/AppData/Local/Tem/vagrant-shell20260206-21276-kiwsr9.sh
-> Client: Running provisioner: common_ip (shell)...
  Client: Running: C:/Users/EBRAHIMI/AppData/Local/Tem/vagrant-shell20260206-21276-62xg2e.sh
  Client: Provisioning script '/tmp/vagrant-shell'
-> Client: Running provisioner: client_routing (shell)...
  Client: Running: C:/Users/EBRAHIMI/AppData/Local/Tem/vagrant-shell20260206-21276-d3ryhn.sh
-> Client: Error: Connection activation failed: No such file or directory
  Client: Provisioning script '/tmp/vagrant-shell'
  Client: Error: Failed to modify connection 'System eth1': ipv4.gateway: gateway cannot be set if there are no addresses configured
  Client: Error: Connection activation failed: IP configuration could not be reserved [no available address, timeout etc]
  Client: Hint: use journalctl -xe NM_CONNECTION=9c92fa7d-g6ec-96c-b4d-8a7c7c69c04 + NM_DEVICE=eth1 to get more details
s.  Client: Connection eth0 successfully deactivated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/0)
  Client: Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/7)
```

Рисунок 4.3: Повторный запуск client с ключом -provision

После применения настроек выполнен вход в графическом интерфейсе под

пользователем alkamal на сервере и клиенте (рис. 4.4, рис. 4.5). При подключении по SSH командой `vagrant ssh` выполнен переход к пользователю `alkamal` через `su - alkamal`, при этом приглашение терминала отображается в формате `alkamal@server.alkamal.net` и `alkamal@client.alkamal.net`, что подтверждает корректную настройку hostname и пользовательского окружения (рис. 4.6).

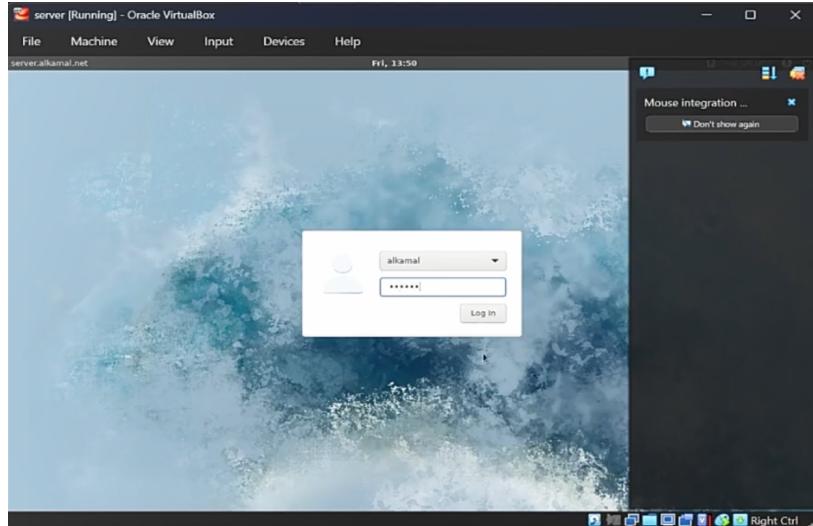


Рисунок 4.4: Графический вход под пользователем alkamal на сервере

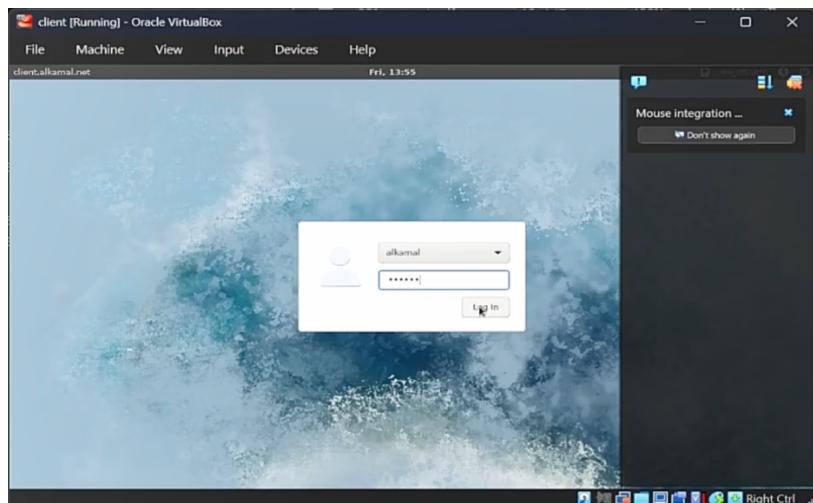


Рисунок 4.5: Графический вход под пользователем alkamal на клиенте

```
C:\work\alkamal\vagrant>vagrant ssh server
=> server: The machine you're attempting to SSH into is configured to use
=> server: password-based authentication. Vagrant can't script entering the
=> server: password for you. If you're prompted for a password, please enter
=> server: the same password you have configured in the Vagrantfile.
vagrant@127.0.0.1's password:
Last login: Fri Feb  6 13:30:28 2026 from 10.0.2.2
[vagrant@server ~]$ su - alkamal
Password:
su: Authentication failure
[vagrant@server ~]$ su - alkamal
Password:
Last login: Fri Feb  6 13:50:24 UTC 2026 on :0
Last failed login: Fri Feb  6 13:55:32 UTC 2026 on pts/1
There was 1 failed login attempt since the last successful login.
[alkamal@server.alkamal.net ~]$ logout
[vagrant@server ~]$ exit
logout

C:\work\alkamal\vagrant>vagrant ssh client
=> client: The machine you're attempting to SSH into is configured to use
=> client: password-based authentication. Vagrant can't script entering the
=> client: password for you. If you're prompted for a password, please enter
=> client: the same password you have configured in the Vagrantfile.
vagrant@127.0.0.1's password:
Last failed login: Fri Feb  6 13:54:52 UTC 2026 from 10.0.2.2 on ssh:notty
There were 432 failed login attempts since the last successful login.
Last login: Fri Feb  6 13:31:07 2026 from 10.0.2.2
[vagrant@client ~]$ su - alkamal
Password:
Last login: Fri Feb  6 13:55:11 UTC 2026 on :0
[alkamal@client.alkamal.net ~]$ logout
[vagrant@client ~]$ exit
logout
```

Рисунок 4.6: Проверка SSH-подключения и отображения приглашения пользователя

5 Выводы

В ходе работы выполнена автоматическая сборка box-файла Rocky Linux с использованием Packer и его регистрация в Vagrant. Развёрнуты виртуальные машины server и client, применены provisioning-скрипты для создания пользователя и настройки hostname. Подтверждена корректная работа SSH-доступа и сетевых параметров. Лабораторный стенд в ОС Windows успешно подготовлен к дальнейшей работе.

6 Контрольные вопросы:

1. Для чего предназначен Vagrant? – Это инструмент для создания и управления средами виртуальных машин в одном рабочем процессе. Он 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, 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. Приведите описание и примеры вызова основных команд Vagrant. `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 от Vagrant 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`. `Vagrantfile` - Первые две строки указывают на режим работы с `Vagrantfile` и использование языка Ruby. Затем идёт цикл `do`, заменяющий конструкцию `Vagrant.configure` далее по тексту на `config`. Стока `config.vm.box = «BOX_NAME»` задаёт название образа (box-файла) виртуальной машины (обычно выбирается из официального репозитория). Стока `config.vm.hostname = «HOST_NAME»` задаёт имя виртуальной машины. Конструкция `config.vm.network` задаёт тип сетевого соединения и может иметь следующие назначения: – `config.vm.network «private_network»`, `ip: «xxx.xxx.xxx.xxx»` – адрес из внутренней сети; – `config.vm.network «public_network»`, `ip: «xxx.xxx.xxx.xxx»` – публичный адрес, по которому виртуальная машина будет доступна; – `config.vm.network «private_network»`, `type: «dhcp»` – адрес, назначаемый по протоколу DHCP. Стока `config.vm.define «VM_NAME»` задаёт название виртуальной машины, по которому можно обращаться к ней из `Vagrant` и `VirtualBox`. В конце идёт конструкция, определяющая параметры провайдера, а именно запуск виртуальной машины без графического интерфейса и с выделением 1 ГБ памяти.

7 Список литературы

1. GNU Bash Manual. — 2019. — URL: <https://www.gnu.org/software/bash/manual/> (дата обр. 13.09.2021).
2. GNU Make Manual. — 2016. — URL: <http://www.gnu.org/software/make/manual/> (visited on 09/13/2021).
3. Powers S. *Vagrant Simplified [Просто о Vagrant]* / Пер.: А. Панин // Библиотека сайта ruslinux.net. — 2015. — URL: <http://rus-linux.net/MyLDP/vm/vagrant-simplified.html> (visited on 09/13/2021).
4. Vagrant Documentation. — URL: <https://www.vagrantup.com/docs> (visited on 09/13/2021).
5. Купер М. *Искусство программирования на языке сценариев командной оболочки.* — 2004. — URL: https://www.opennet.ru/docs/RUS/bash_scripting_guide/ (дата обр. 13.09.2021).