

Отчет по лабораторной работе №1

Дисциплина: Администрирование сетевых подсистем

Иванов Сергей Владимирович

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
3.1	Подготовка рабочего каталога	6
3.2	Развёртывание лабораторного стенда на ОС Windows	11
3.3	Внесение изменений в настройки внутреннего окружения виртуальной машины	14
4	Ответы на контрольные вопросы	16
5	Выводы	19

Список иллюстраций

3.1	Каталог проекта	6
3.2	Образ Rocky	6
3.3	Размещаем файлы	7
3.4	Размещаем файлы	7
3.5	Размещаем файлы	7
3.6	Размещаем файлы	8
3.7	Создание подкаталогов	8
3.8	Скрипт заглушка	9
3.9	Скрипт изменения названия	9
3.10	Скрипт изменения названия	10
3.11	Скрипт маршрутизации	10
3.12	Скрипт сетевых интерфейсов	11
3.13	Создание box-файла	11
3.14	box-файл	12
3.15	Регистрация образа	12
3.16	Запуск Server	12
3.17	Запуск Client	13
3.18	Подключение из консоли	13
3.19	Пользователь svivanov	13
3.20	Пользователь svivanov	13
3.21	Выключение машин	14
3.22	Фиксация изменений и логин	14
3.23	Фиксация изменений и логин	14

1 Цель работы

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

2 Задание

1. Сформировать box-файл с дистрибутивом Rocky Linux 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, добавив пользователя с правами администратора и изменив названия хостов
4. Скопировать необходимые для работы с Vagrant файлы и box-файлы виртуальных машин на внешний носитель. Используя эти файлы, мы можем попробовать развернуть виртуальные машины на другом компьютере.

3 Выполнение лабораторной работы

3.1 Подготовка рабочего каталога

Я выполнял работу в ОС Windows. Предварительно были установлены последние версии Packer, Vagrant и был обновлен VirtualBox.

Перед началом работы с Vagrant создаю каталог для проекта. C:\work\user_name\packer и C:\work\user_name\vagrant, где user_name — имя пользователя. (рис. 1).

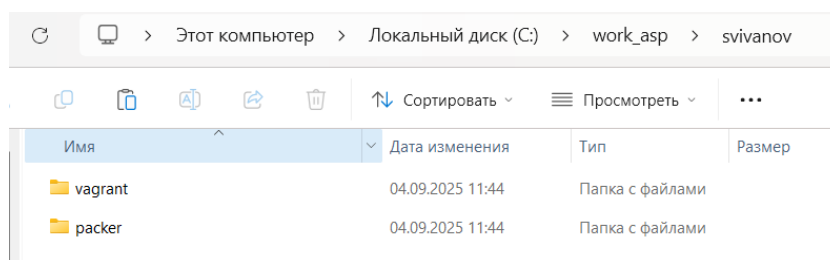


Рис. 3.1: Каталог проекта

В созданном рабочем каталоге в подкаталоге packer разместим образ варианта операционной системы Rocky Linux (Rocky-10.0-x86_64-minimal.iso) (рис. 2).

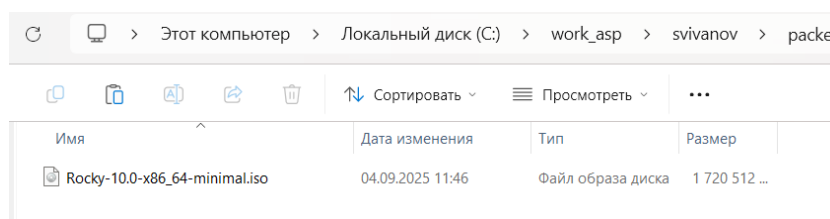


Рис. 3.2: Образ Rocky

В этом же рабочем каталоге разместим подготовленные заранее для работы с Vagrant файлы: в подкаталоге packer файл vagrant-rocky.pkr.hcl (рис. 3)

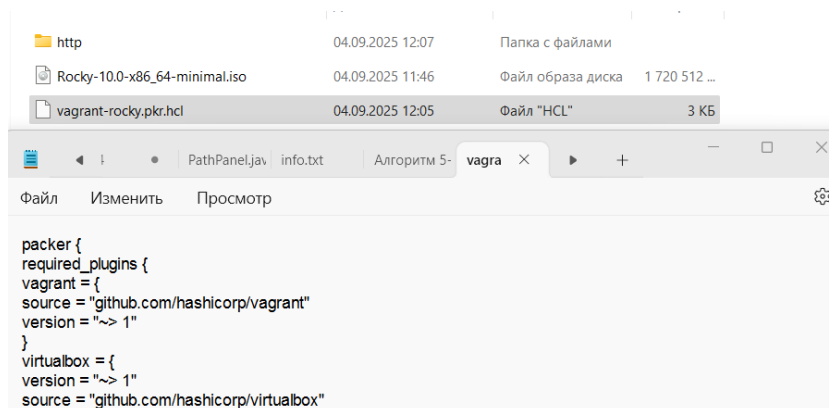


Рис. 3.3: Размещаем файлы

В подкаталоге packer подкаталог http с файлом ks.cfg (рис. 4)

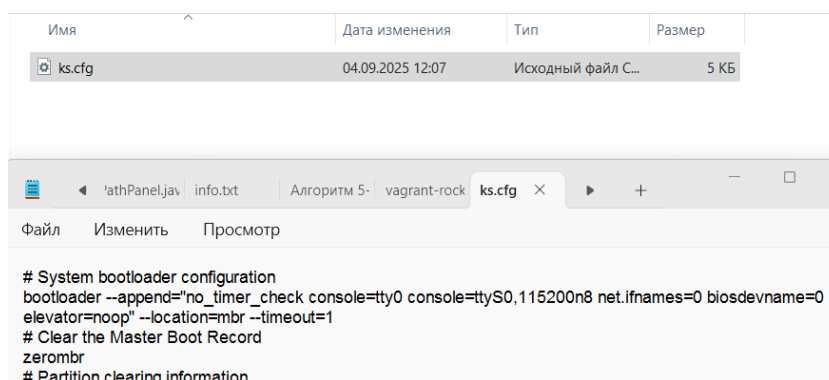


Рис. 3.4: Размещаем файлы

В подкаталоге vagrant файл Vagrantfile (рис. 5)

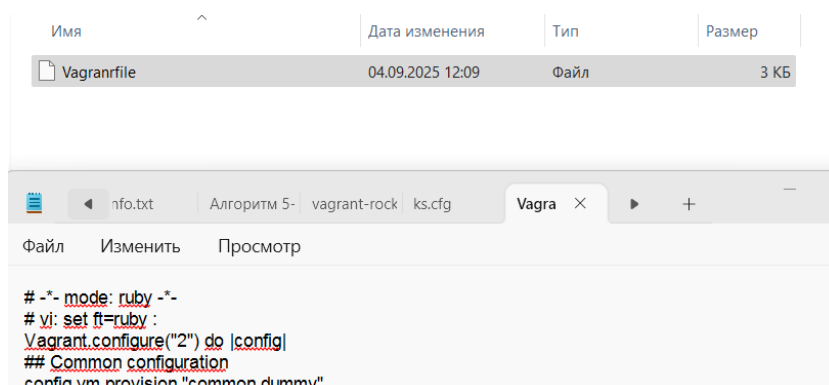


Рис. 3.5: Размещаем файлы

В подкаталоге vagrant файл Makefile (рис. 6)

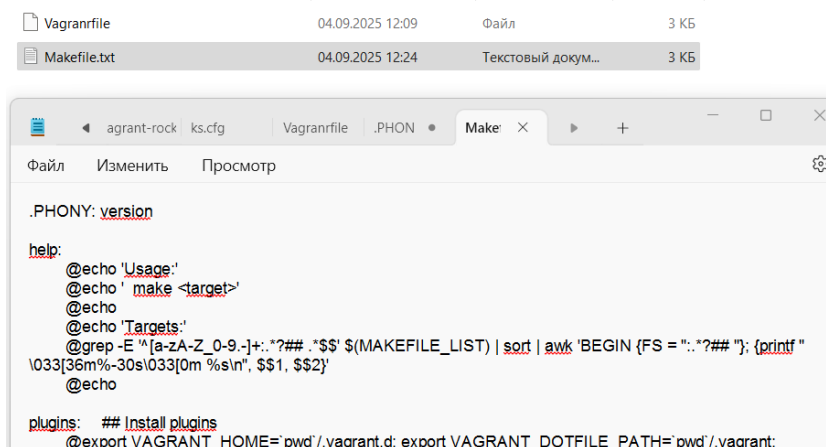


Рис. 3.6: Размещаем файлы

В этом же рабочем каталоге в подкаталоге vagrant создадим каталог provision с подкаталогами default, server и client, в которых будут размещаться скрипты, изменяющие настройки внутреннего окружения базового образа виртуальной машины, сервера или клиента соответственно. (рис. 7)

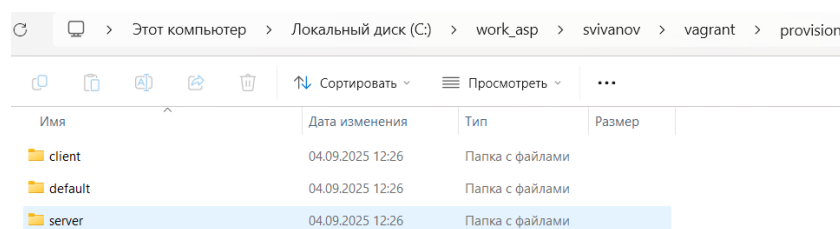


Рис. 3.7: Создание подкаталогов

В каталогах default, server и client разместим заранее подготовленный скрипт-заглушку 01-dummy.sh (рис. 8)

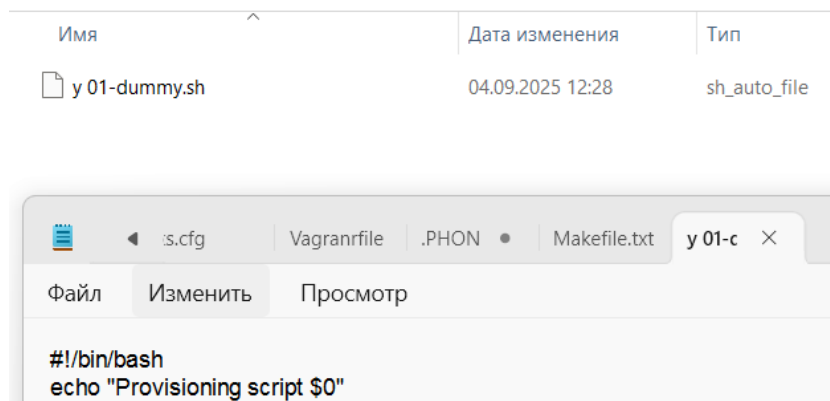


Рис. 3.8: Скрипт заглушка

В каталоге default разместим заранее подготовленный скрипт 01-user.sh по изменению названия виртуальной машины. (рис. 9)

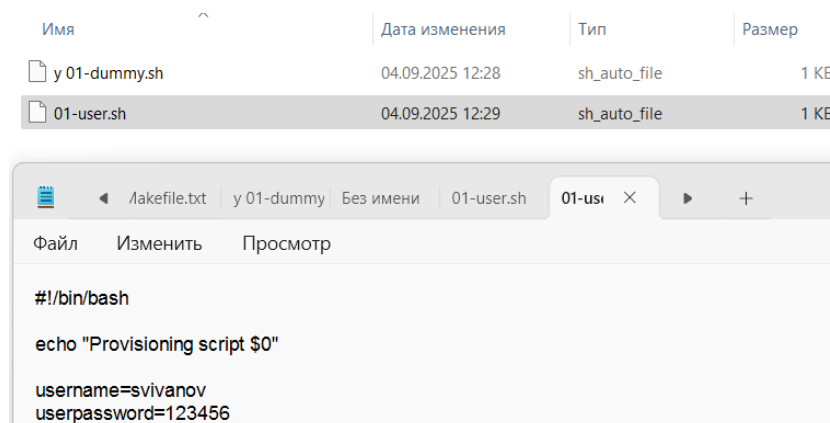


Рис. 3.9: Скрипт изменения названия

В каталоге default разместим заранее подготовленный скрипт 01-hostname.sh по изменению названия виртуальной машины. (рис. 10)

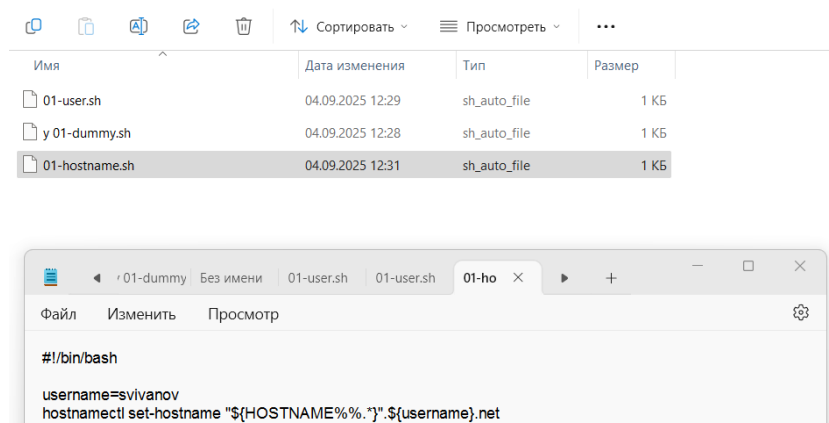


Рис. 3.10: Скрипт изменения названия

В каталоге server разместим заранее подготовленный скрипт 02-forward.sh. Этот скрипт обеспечивает корректную маршрутизацию ip-адресов между сервером и клиентом. (рис. 11)

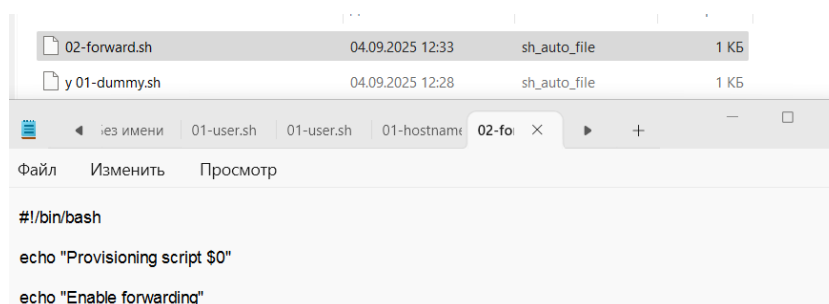


Рис. 3.11: Скрипт маршрутизации

В каталоге client разместим заранее подготовленный скрипт 01-routing.sh. Этот скрипт обеспечивает корректную работу сетевых интерфейсов клиента. (рис. 12)

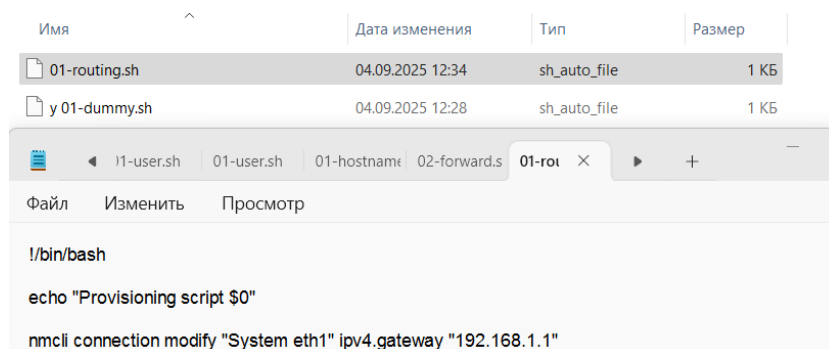


Рис. 3.12: Скрипт сетевых интерфейсов

3.2 Развёртывание лабораторного стенда на ОС Windows

Перейдем в созданный рабочий каталог с проектом. В этом же каталоге размещён файл `packer.exe`. В командной строке введём `packer.exe init vagrant-rocky.pkr.hcl` `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 сформировался `box`-файл с названием `vagrant-virtualbox-rocky-9-x86_64.box`. (рис. 13, 14)

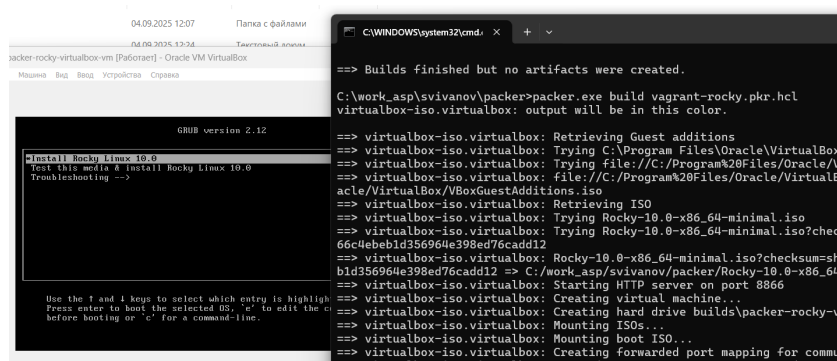


Рис. 3.13: Создание `box`-файла

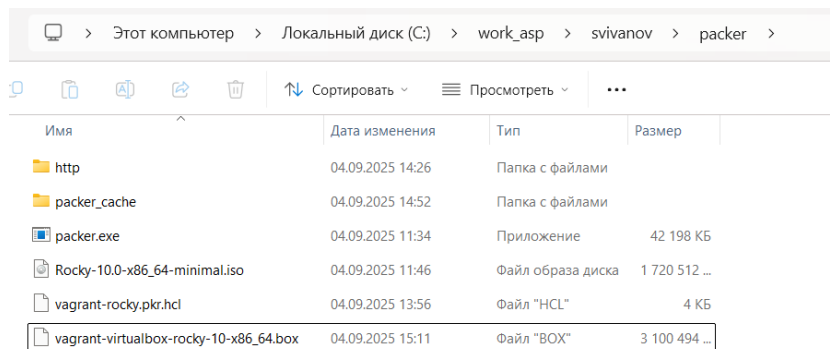


Рис. 3.14: box-файл

Для регистрации образа виртуальной машины в vagrant в командной строке введем `vagrant box add rocky9 vagrant-virtualbox-rocky-9-x86_64.box` (рис. 15)

```
C:\work_asp\svivanov\packer>vagrant box add rocky9 vagrant-virtualbox-rocky-10-x86_64.box
==> box: Box file was not detected as metadata. Adding it directly...
==> box: Adding box 'rocky9' (v0) for provider: (amd64)
box: Unpacking necessary files from: file://C:/work_asp/svivanov/packer/vagrant-virtualbox-rocky-10-x86_64.box
==> box: Successfully added box 'rocky9' (v0) for '(amd64)'!
C:\work_asp\svivanov\packer>
```

Рис. 3.15: Регистрация образа

Запустим виртуальную машину Server, введя `make server-up` (рис. 16)

```
C:\work_asp\svivanov\packer>cd C:/work_asp/svivanov/vagrant
C:\work_asp\svivanov\packer>vagrant up server
Bringing machine 'server' up with 'virtualbox' provider...
==> server: You assigned a static IP ending in ".1" or ":1" to this machine. This is very often used by the router and can cause the network to not work properly. If the network doesn't work properly, try changing this IP.
==> server: Preparing master VM for linked clones...
server: This is a one time operation. Once the master VM is prepared, it will be used as a base for linked clones, making the creation of new VMs take milliseconds on a modern system.
==> server: Importing base box 'rocky9'...
==> server: Cloning VM...
==> server: Matching MAC address for NAT networking...
==> server: You assigned a static IP ending in ".1" or ":1" to this machine. This is very often used by the router and can cause the network to not work properly. If the network doesn't work properly, try changing this IP.
==> server: Setting the name of the VM: server
```

Рис. 3.16: Запуск Server

Запустим виртуальную машину Client, введя `make client-up` (рис. 17)

```
C:\work_asp\svivanov\ 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. N...
==> client: Clearing any previously set network interfaces...
==> client: Preparing network interface based on the host...
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 minute or two...
client: SSH address: 127.0.0.1:2200
client: SSH username: vagrant
client: SSH auth method: password

[ OK ] Started upower.service - Daemon for power management.
[ 11.222490 ] NET: Registered PF_INET protocol family
[ OK ] Started NetworkManager.service - Network Manager.
[ OK ] Finished dhcpcd.service - Builds and all new kernel modules through DKMS.
[ OK ] Started firewalld.service - Firewall - dynamic firewall daemon.
[ OK ] Reached target network-pre.target - Preparation for Network.
Starting NetworkManager.service - Network Manager...
[ OK ] Started systemd-hostnamed.service - Hostname Service.
Starting NetworkManager-dispatcher.service - Network Manager Dispatcher Service...
[ 13.131040 ] e1880: eth0 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[ 13.167717 ] e1880: eth1 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[ OK ] Started NetworkManager-dispatcher.service - Network Manager Dispatcher Service.
[ OK ] Started NetworkManager.service - Network Manager.
[ OK ] Reached target network.target - Network.
Starting NetworkManager-wait-online.service - Network Manager Wait Online...
Starting cups.service - CUPS Scheduler...
Starting sshd.service - OpenSSH server daemon...
Starting systemd-user-sessions.service - Permit User Sessions...
Starting tuned.service - Dynamic System Tuning Daemon...
[ OK ] Finished systemd-user-sessions.service - Permit User Sessions.
```

Рис. 3.17: Запуск Client

Подключимся к серверу из консоли: `vagrant ssh server` Введем пароль `vagrant`. (рис. 18)

```
C:\work_asp\svivanov\ vagrant>vagrant ssh server
==> server: The machine you're attempting to SSH into is configured with password-based authentication. Vagrant can't script password-based authentication.
==> server: password for you. If you're prompted for a password, please type:
vagrant@127.0.0.1's password:
Last login: Thu Sep 4 12:36:25 2025
vagrant@server:~$
```

Рис. 3.18: Подключение из консоли

Перейдем к пользователю `svivanov`: (рис. 19) `su - user`

```
vagrant@server:~$ su - svivanov
Password:
Last login: Чт сен 4 12:53:53 UTC 2025 on pts/0
svivanov@server:~$ logout
vagrant@server:~$
```

Рис. 3.19: Пользователь svivanov

Отлогинимся и выполним тоже самое для клиента. (рис. 20)

```
vagrant@client:~$ su - svivanov
Password:
Last failed login: Чт сен 4 12:58:49 UTC 2025 on pts/0
There were 2 failed login attempts since the last successful login.
svivanov@client:~$
```

Рис. 3.20: Пользователь svivanov

Выключим обе виртуальные машины: `vagrant halt server` `vagrant halt client` (рис. 21)

```

PS C:\Users\lserg> cd C:/work_asp/svivanov/vagrant
PS C:\work_asp\svivanov\vagrant> vagrant halt server
==> server: Attempting graceful shutdown of VM...
PS C:\work_asp\svivanov\vagrant> vagrant halt client
==> client: Attempting graceful shutdown of VM...
PS C:\work_asp\svivanov\vagrant> |

```

Рис. 3.21: Выключение машин

3.3 Внесение изменений в настройки внутреннего окружения виртуальной машины

Зафиксируем внесённые изменения для внутренних настроек виртуальных машин, введя в терминале: `vagrant up server --provision vagrant up client --provision` Залогинимся на сервере и клиенте под созданным пользователем. Убедимся, что в терминале приглашение отображается в виде `user@server.user.net` на сервере и в виде `user@client.user.net` на клиенте. (рис. 22, 23)

```

vagrant@server:~$ su - svivanov
Password:
Last login: Чт сен  4 13:14:57 UTC 2025 on tty2
[svivanov@server.svivanov.net ~]$
[svivanov@server.svivanov.net ~]$
[svivanov@server.svivanov.net ~]$

```

Рис. 3.22: Фиксация изменений и логин

```

PS C:\work_asp\svivanov\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: Thu Sep  4 13:44:18 2025 from 10.0.2.2
vagrant@client:~$ su - svivanov
Password:
Last login: Чт сен  4 13:57:41 UTC 2025 on tty2
[svivanov@client.svivanov.net ~]$

```

Рис. 3.23: Фиксация изменений и логин

После выключения виртуальных машин скопировал необходимые для работы

с Vagrant файлы и box-файлы виртуальных машин на внешний носитель.

4 Ответы на контрольные вопросы

1. Для чего предназначен Vagrant?

Это инструмент для создания и управления средами виртуальных машин в одном рабочем процессе.

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

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. 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`.

- `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 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.
- `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.

Строка `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.

5 Выводы

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