

EPAM University Programs
DevOps external course
Module 2 Virtualization and Cloud Basic
TASK 2.2

Danylenko Victor - Homework

ЧАСТИНА 1. РОБОТА З VIRTUALBOX

1. Перший запуск VirtualBox та віртуальної машини (VM).

1.1 Ознайомитись зі структурою керівництва користувача VirtualBox [1]

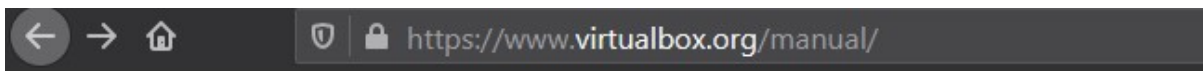


Table of Contents

[Preface](#)

[1. First Steps](#)

[1.1. Why is Virtualization Useful?](#)

[1.2. Some Terminology](#)

[1.3. Features Overview](#)

[1.4. Supported Host Operating Systems](#)

[1.4.1. Host CPU Requirements](#)

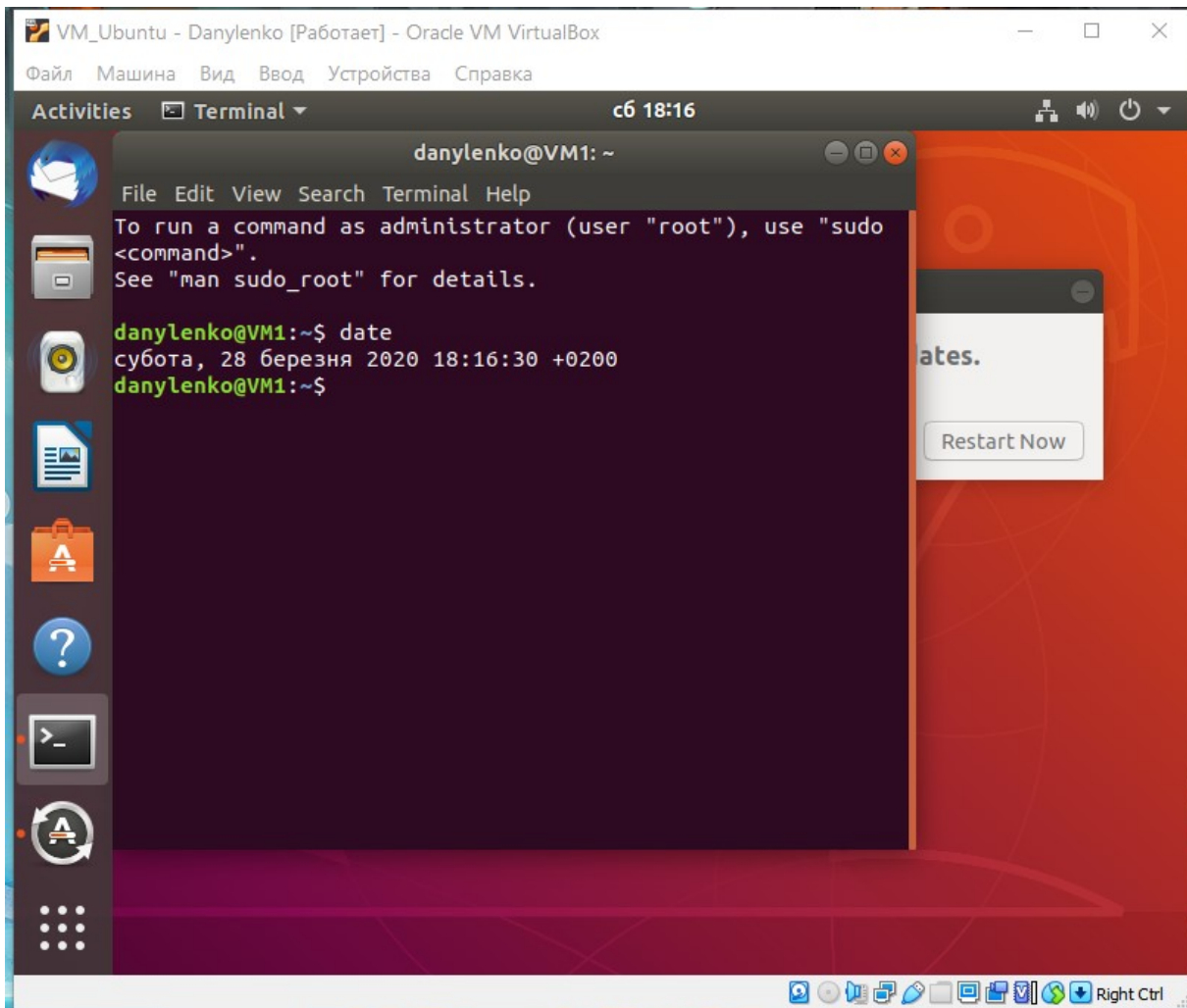
[1.5. Installing Oracle VM VirtualBox and Extension Packs](#)

[1.6. Starting Oracle VM VirtualBox](#)

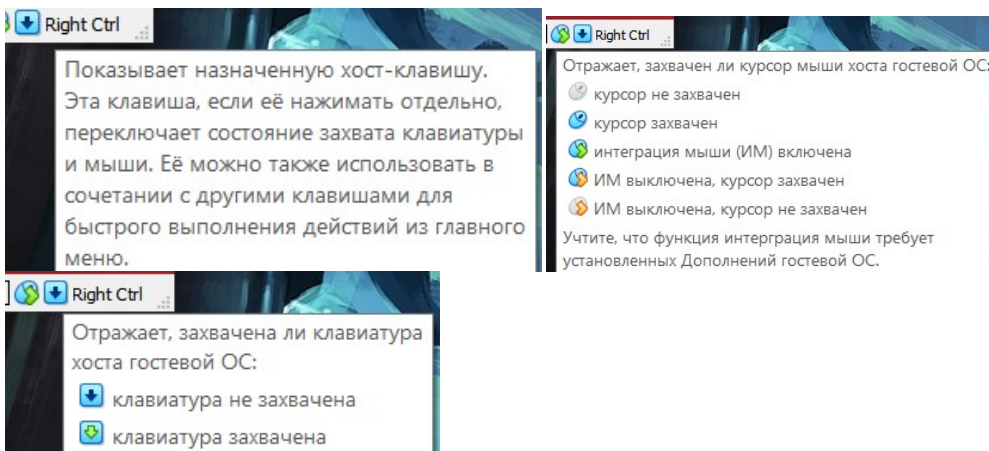
1.2 З офіційного сайту VirtualBox [2] завантажити останню стабільну версію VirtualBox відповідно до хостової операційної системи (ОС), що встановлена на робочому місці студента. Для ОС Windows файл може називатися, наприклад, VirtualBox-6.0.12-133076-Win.exe. Провести інсталяцію VirtualBox.

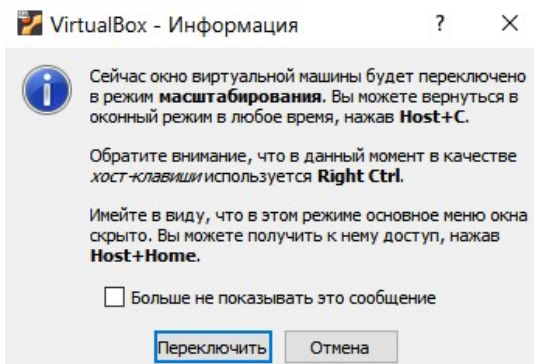
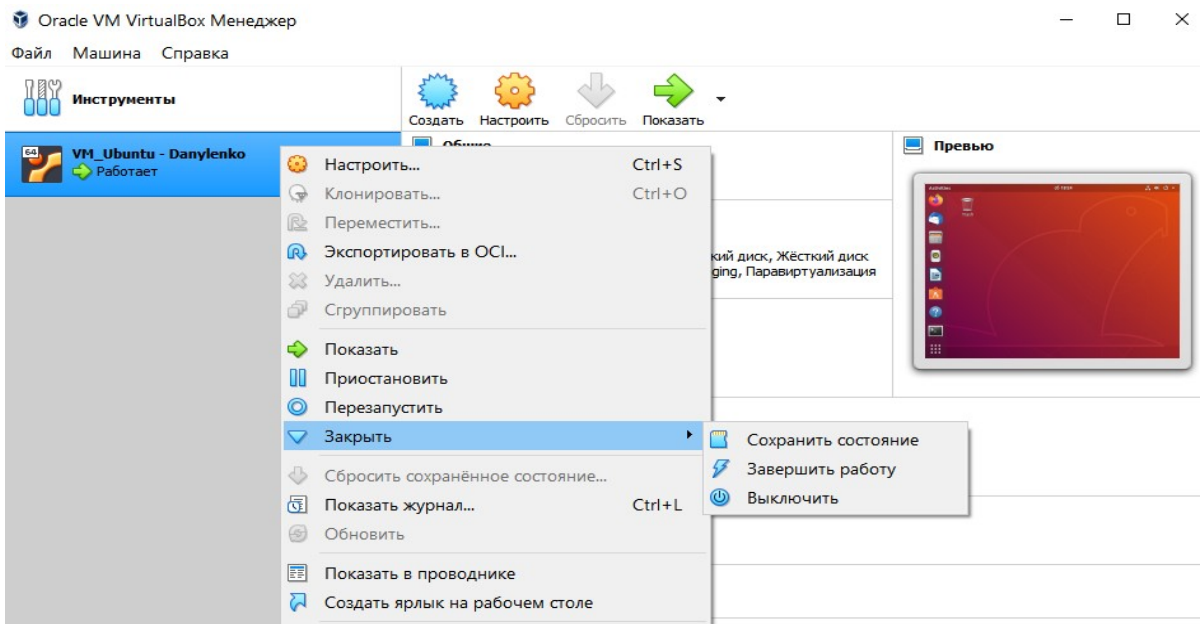
1.2 Завантажити з офіційного сайту останню стабільну версію образу ОС Ubuntu Desktop або Ubuntu Server [3].

1.3 Створити VM1 та провести інсталяцію ОС Ubuntu користуючись інструкціями [1, п.1.7].
Ім'я машини задати як «ім'я хостової машини»_«прізвище студента»

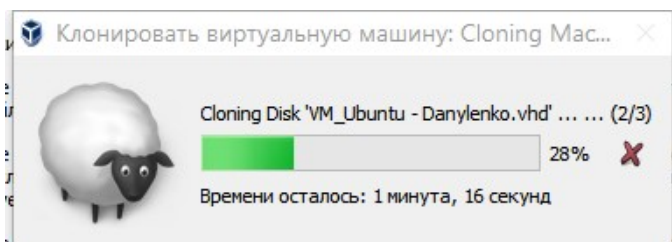


1.4 Ознайомитись з можливостями керування VM1 – запуск, зупинка, перезавантаження, збереження стану, використання Host key та комбінацій клавіш, захват миші та ін. [1, п.1.8].

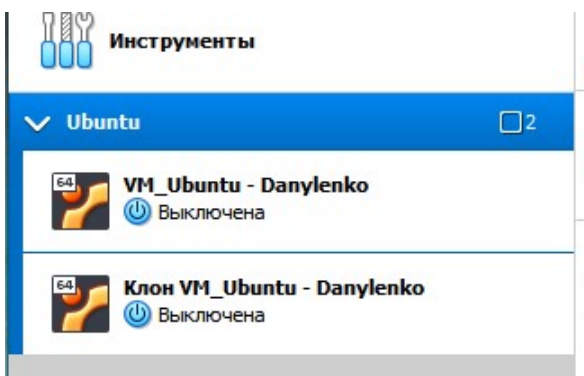




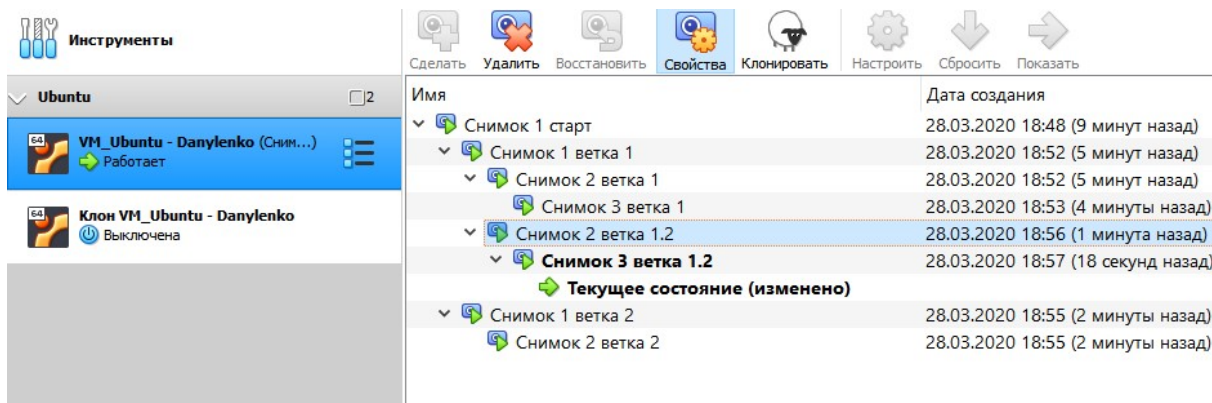
1.5 Клонувати існуючу VM1, створивши VM2 [1, п.1.13].



1.6 Створити групу з двох VM: VM1, VM2 та вивчити функції, що відносяться до груп [1, п.1.9].

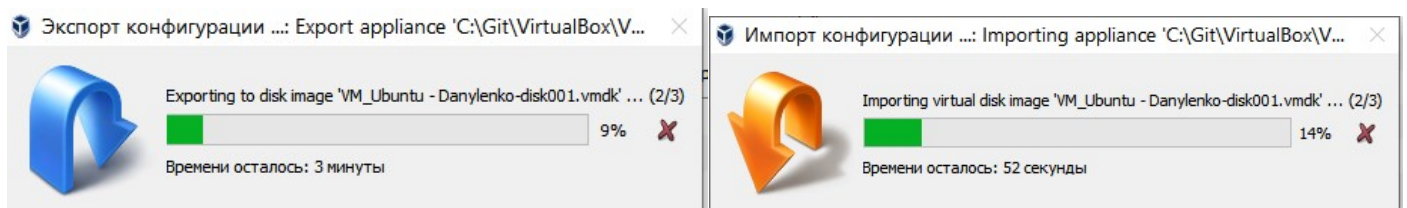


1.7 Для VM1 змінюючи її стан, зробити кілька різних знімків, утворивши розгалужене дерево знімків [1, п.1.10].



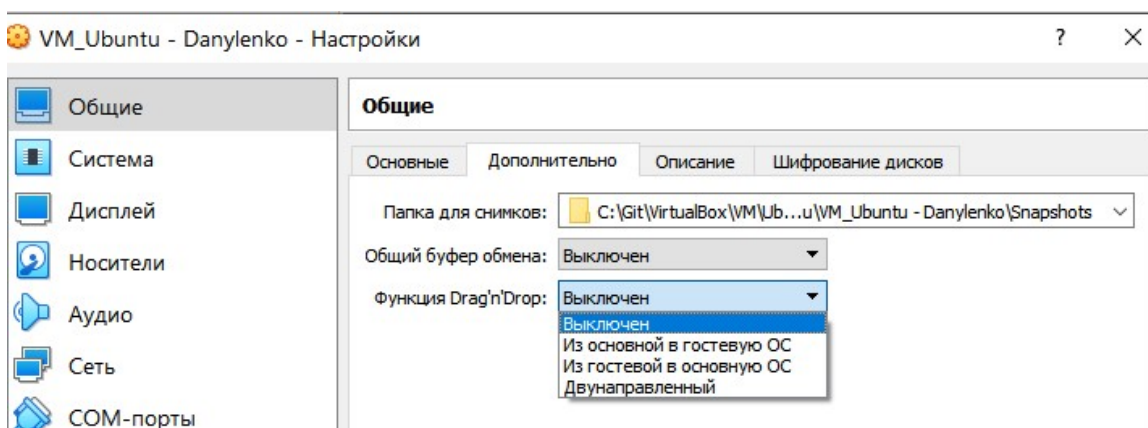
1.8 Зробити експорт VM1, файл *.ova зберегти на мережному диску зі спільним доступом. На цьому ж диску обрати файл *.ova, що створений іншим студентом та імпортувати його [1, п.1.14].

Tested import export without network shared folders



2. Конфігурація віртуальних машин

2.1 Вивчити можливості налаштування VM (загальні налаштування, системні параметри, дисплей, зберігання, аудіо, мережі тощо).



Общие

Система

Дисплей

Носители

Аудио

Сеть

Общие

Основные

Дополнительно

Описание

Шифрование дисков

☒ Включить шифрование дисков
Алгоритм шифрования дисков: Не менять
Введите новый пароль:
Подтвердите новый пароль:

Система

Материнская плата

Процессор

Ускорение

Основная память: 1024 МБ
4 МБ 8192 МБ

Порядок загрузки:

☒ Гибкий диск

☒ Оптический диск

☒ Жёсткий диск

☐ Сеть

Чипсет: PIIX3

Манипулятор курсора: USB планшет

Дополнительные возможности:

☒ Включить I/O APIC

☐ Включить EFI (только специальные ОС)

☒ Часы в системе UTC

Система

Материнская плата

Процессор

Ускорение

Процессор(ы): 1 ЦП
1 ЦП 8 ЦП

Предел загрузки ЦПУ: 100%
1% 100%

Дополнительные возможности:

☐ Включить PAE/NX

☐ Включить Nested VT-x/AMD-V

Система

Материнская плата

Процессор

Ускорение

Интерфейс паравиртуализации: По умолчанию

Аппаратная виртуализация: ☒ Включить Nested Paging

Дисплей

Экран

Удаленный доступ

Запись

Видеопамять: 256 МБ
0 МБ 256 МБ

Количество мониторов: 1
1 8

Коефициент масштабирования: Все мониторы
Мин Макс 100%

Графический контроллер: VMSVGA

Ускорение: ☐ Включить 3D-ускорение

Дисплей

Экран

Удаленный доступ

Запись

☐ Включить сервер удалённого доступа
Порт сервера: 3389
Метод аутентификации: Нет авторизации
Время ожидания аутентификации: 5000
Дополнительные возможности: ☐ Многопользовательский удалённый доступ

Дисплей

Экран Удаленный доступ Запись

☐ Включить запись

Режим записи: Только видео

Путь к файлу: C:\Git\VirtualB...tu - Danylenko\VM_Ubuntu - Danylenko.webm

Размер кадра: 1024 x 768 (4:3) 1024 768

Скорость записи: 1 кадр/сек 30 кадр/сек 25 кадр/сек

Качество видео: низкое среднее высокое 512 кбит/сек

Качество аудио: низкое среднее высокое

Около 18MB за 5 минут видео

Дисплей: ☒ Дисплей 1

Носители

Носители

- Контроллер: IDE
 - Пусто
 - Контроллер: SATA
 - VM_Ubuntu - Danylenko.vhd

Атрибуты

Имя: IDE

Тип: PIIX4

☒ Кэширование ввода/вывода

- AHCI (SATA)
- PIIX4 (Default IDE)
- PIIX3 (IDE)
- ICH6 (IDE)
- LsiLogic (Default SCSI)
- BusLogic (SCSI)
- I82078 (Floppy)
- LsiLogic SAS (SAS)
- USB
- NVMe (PCIe)
- virtio-scsi

Сеть

Адаптер 1 Адаптер 2 Адаптер 3 Адаптер 4

☒ Включить сетевой адаптер

Тип подключения: NAT

Имя: NAT

Дополнительно

Тип адаптера: Сетевая мост

Неразборчивый режим: Внутренняя сеть

MAC-адрес: 080027121BD5

☒ Подключить кабель

Проброс портов

COM-порты

Порт 1 Порт 2 Порт 3 Порт 4

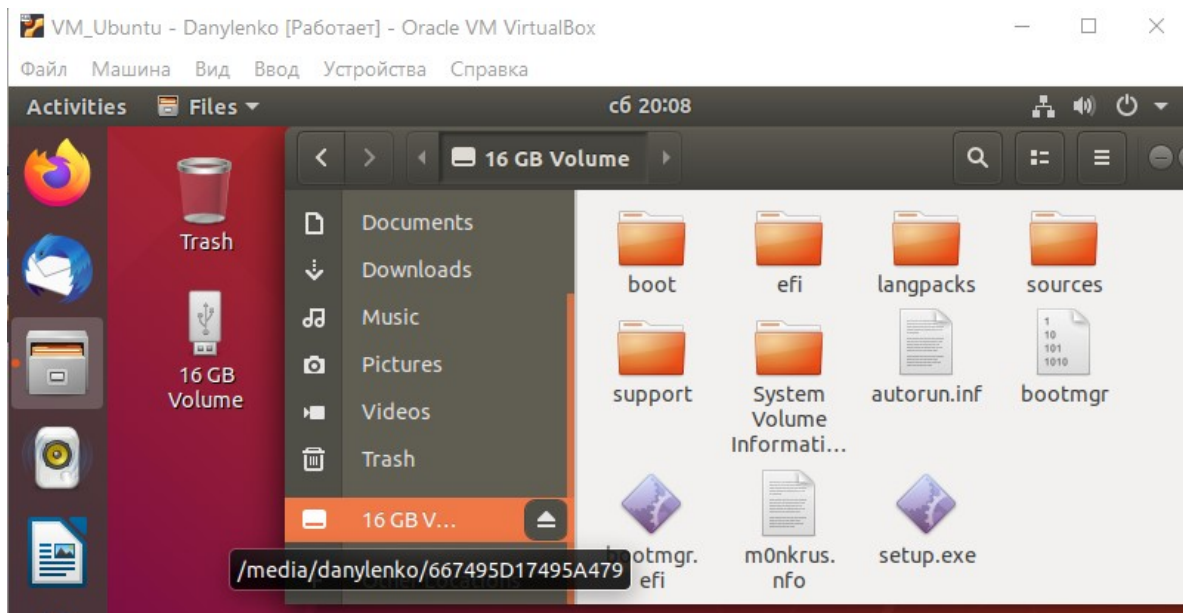
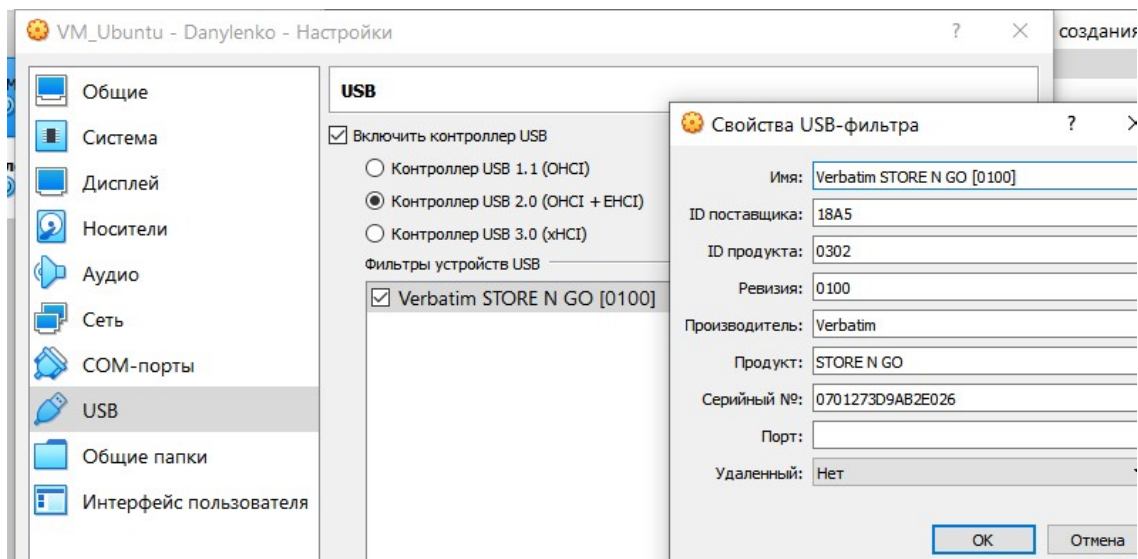
☒ Включить последовательный порт

Номер порта: COM1 Прерывание: 4 Порт В/В: 0x3F8

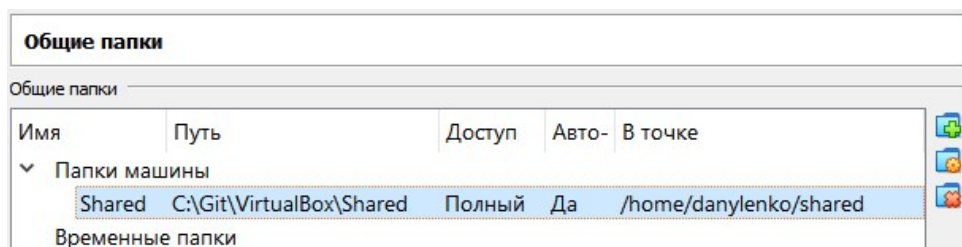
Режим порта: Отключен

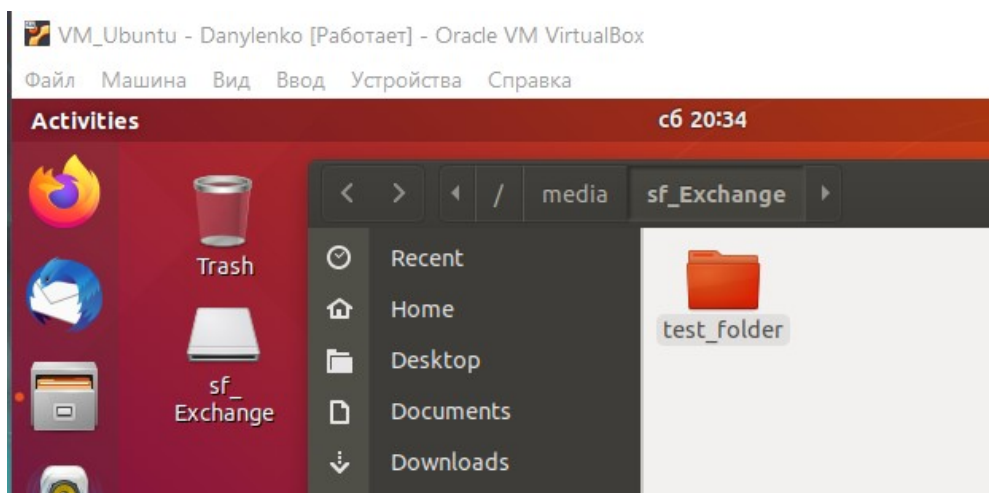
Путь/адрес: Хост-канал

2.2 Провести налаштування USB для підключення USB-портів хостової машини до VM [1, п.3.11].



2.3 Провести налаштування спільної папки для обміну даними між віртуальною машиною та хостом [1, п.4.3].





2.4 Провести налаштування різних режимів роботи мережі для VM1, VM2. Перевірити наявність зв'язку між VM1, VM2, Host, Internet для різних режимів роботи мережі. Для цього можна використати команду ping. Скласти відповідну таблицю можливих зав'язків.

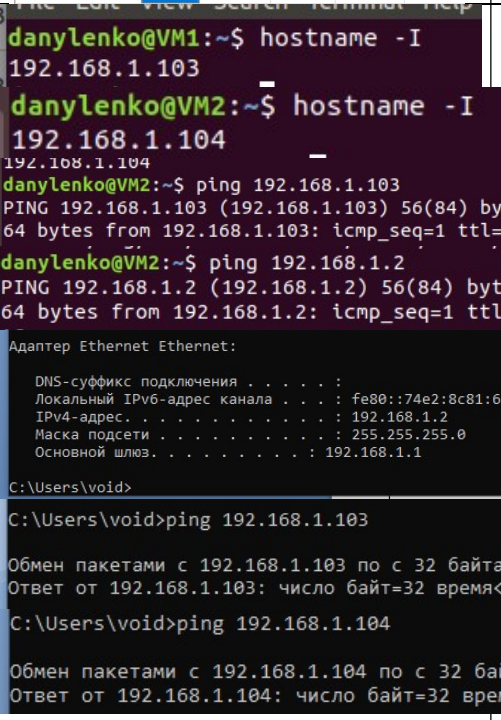
Table from instruction manual:

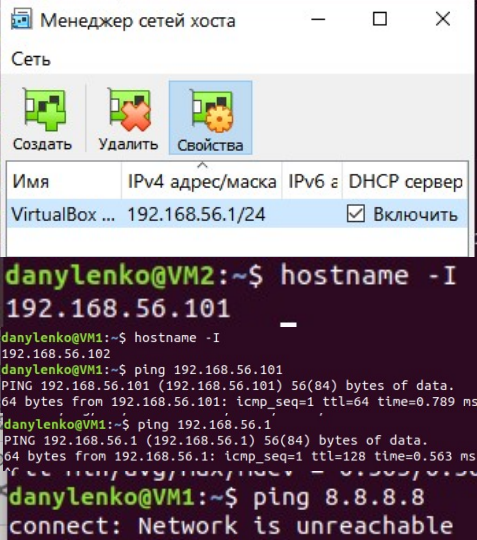
Table 6.1. Overview of Networking Modes

Mode	VM→Host	VM←Host	VM1↔VM2	VM→Net/LAN	VM←Net/LAN
Host-only	+	+	+	–	–
Internal	–	–	+	–	–
Bridged	+	+	+	+	+
NAT	+	Port forward	–	+	Port forward
NATservice	+	Port forward	+	+	Port forward

My setup and checks table:

Network type	Host <> VM	скриншоты	VM1<>VM2	Internet<>VM
NAT	<p>HOST>VM NOT direct</p> <p>Access to VM ports is possible with Port Forward on HOST ports</p> <p>VM > HOST YES on Gateway IP</p> <pre>danylenko@VM1:~\$ ping 10.0.2.2 PING 10.0.2.2 (10.0.2.2) 56(84) bytes of data: 64 bytes from 10.0.2.2: icmp_seq=1 ttl=64 time=0.000000s 64 bytes from 10.0.2.2: icmp_seq=2 ttl=64 time=0.000000s</pre>		<p>NOT direct</p> <p>Access to ports is possible with Port Forward through HOST ←</p>	<p>VM> Internet YES</p> <p>Internet>VM to ports with Port Forward though Host</p>

			←																			
NAT network	Same as just NAT, but PortForward setup is inside Nat Network setup	 <table><thead><tr><th>Имя</th><th>Протокол</th><th>Адрес хоста</th><th>Порт хоста</th><th>Адрес гостя</th><th>Порт гостя</th></tr></thead><tbody><tr><td>Rule 1</td><td>TCP</td><td></td><td>2222</td><td>10.0.2.15</td><td>22</td></tr><tr><td>Rule 2</td><td>TCP</td><td></td><td>2221</td><td>10.0.2.4</td><td>22</td></tr></tbody></table>	Имя	Протокол	Адрес хоста	Порт хоста	Адрес гостя	Порт гостя	Rule 1	TCP		2222	10.0.2.15	22	Rule 2	TCP		2221	10.0.2.4	22	YES	VM> Internet YES Internet>VM to ports with Port Forward though Host
Имя	Протокол	Адрес хоста	Порт хоста	Адрес гостя	Порт гостя																	
Rule 1	TCP		2222	10.0.2.15	22																	
Rule 2	TCP		2221	10.0.2.4	22																	
Bridged	YES		YES	YES																		
Internal	NO		YES	NO																		

Host Virtual Adapter Network	YES		YES	NO
---------------------------------------	-----	--	-----	----

3. Робота з CLI через VBoxManage.

3.1 Запустити командний рядок cmd.exe.

3.2 Вивчити призначення та виконати основні команди VBoxManage list, showvminfo, createvm, startvm, modifyvm, clonevm, snapshot, controlvm [1, п.8].

```
C:\Program Files\Oracle\VirtualBox>VBoxManage list vms
"VM_Ubuntu - Danylenko" {434bab3a-6aa5-438e-8d3c-db64084360a8}
"Клон VM_Ubuntu - Danylenko" {579d3bfc-fb9f-4b00-921e-a688cca26fe7}
"VM1Danylenko.TEST" {590f1e27-0c5f-4377-8f42-16604ffd150e}
"VM2Danylenko.TEST" {d721c010-fd63-41d5-9875-71c64af8983c}
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage list runningvms
"Клон VM_Ubuntu - Danylenko" {579d3bfc-fb9f-4b00-921e-a688cca26fe7}
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage showvminfo "VM_Ubuntu - Danylenko"
Name: VM_Ubuntu - Danylenko
Groups: /Ubuntu
Guest OS: Ubuntu (64-bit)
UUID: 434bab3a-6aa5-438e-8d3c-db64084360a8
Config file: C:\Git\VirtualBox\VM\Ubuntu\VM_Ubuntu - Danylenko\VM_Ubuntu - Danylenko.vbox
Snapshot folder: C:\Git\VirtualBox\VM\Ubuntu\VM_Ubuntu - Danylenko\Snapshots
Log folder: C:\Git\VirtualBox\VM\Ubuntu\VM_Ubuntu - Danylenko\Logs
Hardware UUID: 434bab3a-6aa5-438e-8d3c-db64084360a8
Memory size: 1024MB
Page Fusion: disabled
VRAM size: 256MB
CPU exec cap: 100%
HPET: disabled
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage unregistervm "VM_Ubuntu - Danylenko" --delete
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage createvm --name "VM_Ubuntu - Danylenko" --ostype Ubuntu_64 --default --register
Virtual machine 'VM_Ubuntu - Danylenko' is created and registered.
UUID: ff90847e-3c48-43e9-9a27-c9b61fb68d4e
Settings file: 'C:\Git\VirtualBox\VM\VM_Ubuntu - Danylenko\VM_Ubuntu - Danylenko.vbox'
```



```
C:\Program Files\Oracle\VirtualBox>VBoxManage modifyvm "VM_Ubuntu - Danylenko" --name VM1
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage modifyvm VM1 --groups /Ubuntu
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage startvm VM1  
Waiting for VM "VM1" to power on...  
VM "VM1" has been successfully started.
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage controlvm VM1 savestate  
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage snapshot "Клон VM_Ubuntu - Danylenko" take Snapshot1 --description="Starting snapshot "  
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%  
Snapshot taken. UUID: a41404b7-337f-4ef3-bc42-6b20f9697925
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage snapshot "Клон VM_Ubuntu - Danylenko" list  
Name: Snapshot1 (UUID: a41404b7-337f-4ef3-bc42-6b20f9697925)  
Description:  
Starting snapshot  
Name: Snapshot2 (UUID: 0f9b4e28-c620-4cd0-a296-60fbe0738dcb)  
Description:  
Snapshot2  
Name: Snapshot3 (UUID: e037457d-2144-43f1-a2ea-277843c54dbb) *  
Description:  
Snapshot3
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage controlvm "Клон VM_Ubuntu - Danylenko" poweroff  
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage snapshot "Клон VM_Ubuntu - Danylenko" restore Snapshot2  
Restoring snapshot 'Snapshot2' (0f9b4e28-c620-4cd0-a296-60fbe0738dcb)  
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
```

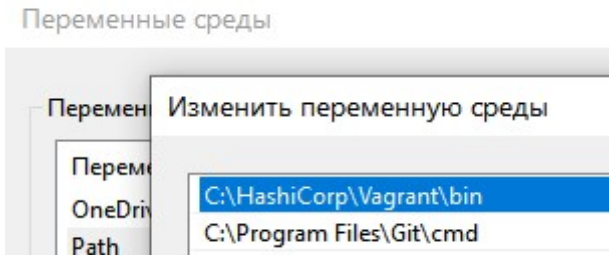
```
C:\Program Files\Oracle\VirtualBox>VBoxManage snapshot "Клон VM_Ubuntu - Danylenko" delete Snapshot1  
Deleting snapshot 'Snapshot1' (a41404b7-337f-4ef3-bc42-6b20f9697925)  
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
```

```
C:\Program Files\Oracle\VirtualBox>VBoxManage clonevm "Клон VM_Ubuntu - Danylenko" --name="VM1_Ubuntu - Danylenko" --register  
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%  
Machine has been successfully cloned as "VM1_Ubuntu - Danylenko"
```

ЧАСТИНА 2. РОБОТА 3 VAGRANT

1. Завантажити необхідну версію Vagrant відповідно інструкціям [5] та відповідно до хостової операційної системи (ОС), що встановлена на робочому місці студента. Для ОС Windows файл може називатися, наприклад, vagrant_2.2.0_x86_64.msi. Провести інсталяцію

Vagrant. Перевірити наявність шляху до Vagrant bin у змінній Path (My computer -> Properties -> Advanced system settings-> Advanced -> Environment Variables).



2. Запустіть powershell. Створіть папку «прізвище студента» (англійською мовою). В цьому прикладі створимо папку `vagrant_test`. Далі заходимо в папку.

3. Проведемо ініціалізацію оточення з вказівкою боксу Vagrant за замовчуванням: `init hashicorp/precise64`

```
Каталог: C:\git

Mode                LastWriteTime         Length Name
----                -
d-----          3/28/2020   9:01 PM             danylenko

PS C:\git> cd .\danylenko
PS C:\git\danylenko> vagrant init hashicorp/precise64
A 'Vagrantfile' has been placed in this directory. You are now
ready to 'vagrant up' your first virtual environment! Please read
the comments in the Vagrantfile as well as documentation on
'vagrantup.com' for more information on using Vagrant.
```

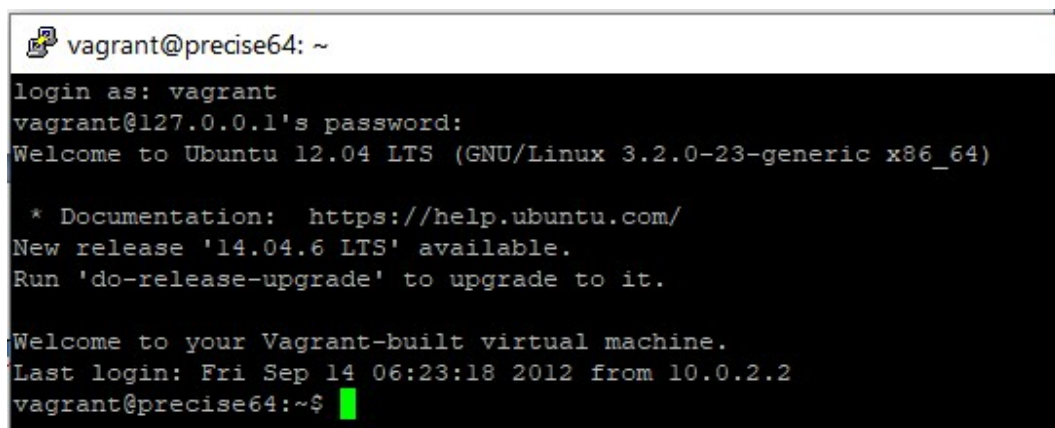
4. Запускаємо `vagrant up` та спостерігаємо за повідомленнями під час завантаження та запуску VM.


```

PS C:\git\danylenko> vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
=> default: Box 'hashicorp/precise64' could not be found. Attempting to find and install...
=> default: Box Provider: virtualbox
=> default: Box Version: >= 0
=> default: Loading metadata for box 'hashicorp/precise64'
=> default: URL: https://vagrantcloud.com/hashicorp/precise64
=> default: Adding box 'hashicorp/precise64' (v1.1.0) for provider: virtualbox
=> default: Downloading: https://vagrantcloud.com/hashicorp/boxes/precise64/versions/1.1.0/providers/virtualbox.box
=> default: Download redirected to host: vagrantcloud-files-production.s3.amazonaws.com
=> default: Successfully added box 'hashicorp/precise64' (v1.1.0) for 'virtualbox'!
=> default: Importing base box 'hashicorp/precise64'...
=> default: Matching MAC address for NAT networking...
=> default: Checking if box 'hashicorp/precise64' version '1.1.0' is up to date...
=> default: Setting the name of the VM: danylenko_default_1585422362240_90959
Vagrant is currently configured to create VirtualBox synced folders with
the 'SharedFoldersEnableSymlinksCreate' option enabled. If the Vagrant
guest is not trusted, you may want to disable this option. For more
information on this option, please refer to the VirtualBox manual:
https://www.virtualbox.org/manual/ch04.html#sharedfolders
This option can be disabled globally with an environment variable:
VAGRANT_DISABLE_VBOXSYMLINKCREATE=1
or on a per folder basis within the Vagrantfile:
config.vm.synced_folder '/host/path', '/guest/path', SharedFoldersEnableSymlinksCreate: false
=> default: Clearing any previously set network interfaces...
=> default: Preparing network interfaces based on configuration...
=> default: Adapter 1: nat
=> default: Forwarding ports...
=> default: 22 (guest) => 2222 (host) (adapter 1)
=> default: Booting VM...
=> default: Waiting for machine to boot. This may take a few minutes...
=> default: SSH address: 127.0.0.1:2222
=> default: SSH username: vagrant
=> default: SSH auth method: private key
=> default: Vagrant insecure key detected. Vagrant will automatically replace
=> default: this with a newly generated keypair for better security.
=> default: Inserting generated public key within guest...
=> default: Removing insecure key from the guest if it's present...
=> default: Key inserted! Disconnecting and reconnecting using new SSH key...
=> default: Machine booted and ready!
=> default: Checking for guest additions in VM...
=> default: The guest additions on this VM do not match the installed version of
=> default: VirtualBox! In most cases this is fine, but in rare cases it can
=> default: prevent things such as shared folders from working properly. If you see
=> default: shared folder errors, please make sure the guest additions within the
=> default: virtual machine match the version of VirtualBox you have installed on
=> default: your host and reload your VM.
=> default: Guest Additions Version: 4.2.0
=> default: VirtualBox Version: 6.1
=> default: Mounting shared folders...
=> default: /vagrant => C:/Git/danylenko
PS C:\git\danylenko>

```

5. Підключаємося до VM за допомогою програми PuTTY (завантажити можна з [6]), використовуючи SSH, IP-адресу та порт що вказані вище (127.0.0.1:2222). За замовчуванням login – vagrant та password також vagrant.



```

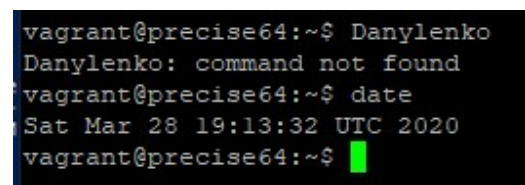
vagrant@precise64: ~
login as: vagrant
vagrant@127.0.0.1's password:
Welcome to Ubuntu 12.04 LTS (GNU/Linux 3.2.0-23-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
New release '14.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Welcome to your Vagrant-built virtual machine.
Last login: Fri Sep 14 06:23:18 2012 from 10.0.2.2
vagrant@precise64:~$

```

6. Зафіксуйте дату та час, виконавши команду date



```

vagrant@precise64:~$ Danylenko
Danylenko: command not found
vagrant@precise64:~$ date
Sat Mar 28 19:13:32 UTC 2020
vagrant@precise64:~$

```

7. Зупиніть та видаліть створену VM.

```

PS C:\git\danylenko> vagrant halt
==> default: Attempting graceful shutdown of VM...
PS C:\git\danylenko> vagrant destroy
default: Are you sure you want to destroy the 'default' VM? [y/N] y
==> default: Destroying VM and associated drives...
PS C:\git\danylenko>

```

9. Створити власний Vagrant box, використовуючи інструкції [8] та вимоги, що визначає викладач або обирає студент.

```

==> VM_Ubuntu - Danylenko: Exporting VM...
PS C:\Git\danylenko> vagrant package --base 'VM_Ubuntu - Danylenko' --output Ubuntu_18.04.4_Desktop_Danylenko
==> VM_Ubuntu - Danylenko: Attempting graceful shutdown of VM...
VM_Ubuntu - Danylenko: Guest communication could not be established! This is usually because
VM_Ubuntu - Danylenko: SSH is not running, the authentication information was changed,
VM_Ubuntu - Danylenko: or some other networking issue. Vagrant will force halt, if
VM_Ubuntu - Danylenko: capable.
==> VM_Ubuntu - Danylenko: Forcing shutdown of VM...
==> VM_Ubuntu - Danylenko: Exporting VM...
==> VM_Ubuntu - Danylenko: Compressing package to: C:/Git/danylenko/Ubuntu_18.04.4_Desktop_Danylenko
PS C:\Git\danylenko>
PS C:\Git\danylenko> vagrant box add Ubuntu_18.04.4_Desktop_Danylenko --name 'ubuntu_18.04_danylenko'
==> box: Box file was not detected as metadata. Adding it directly...
==> box: Adding box 'ubuntu_18.04_danylenko' (v0) for provider:
box: Unpacking necessary files from: file://C:/Git/danylenko/Ubuntu_18.04.4_Desktop_Danylenko
box:
==> box: Successfully added box 'ubuntu_18.04_danylenko' (v0) for 'virtualbox'!
PS C:\Git\danylenko> vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Importing base box 'ubuntu_18.04_danylenko'...
==> default: Matching MAC address for NAT networking...
==> default: Setting the name of the VM: danylenko_default_1585501144072_62366
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
default: Adapter 1: nat
==> default: Forwarding ports...
default: 22 (guest) => 2222 (host) (adapter 1)
==> default: Booting VM...
==> default: Waiting for machine to boot. This may take a few minutes...
default: SSH address: 127.0.0.1:2222
default: SSH username: vagrant
default: SSH auth method: private key
default:
default: Vagrant insecure key detected. Vagrant will automatically replace
default: this with a newly generated keypair for better security.
default:
default: Inserting generated public key within guest...
default: Removing insecure key from the guest if it's present...
default: Key inserted! Disconnecting and reconnecting using new SSH key...
==> default: Machine booted and ready!
==> default: Checking for guest additions in VM...
==> default: Mounting shared folders...
default: /vagrant => C:/Git/danylenko
PS C:\Git\danylenko>

```

```

Last login: Sun Mar 29 19:33:05 2020 from 192.168.1.2
vagrant@VM1:~$ date
неділя, 29 березня 2020 20:06:45 +0300
vagrant@VM1:~$ Danylenko

```

8. Створити тестову середу з двох серверів, використовуючи інструкції [7]. Параметри серверів задаються викладачем або обираються самостійно студентом.

According to instruction, picked myself these characteristics of vagrant machines:

```

BRIDGE_NET="192.168.1."
INTERNAL_NET="172.17.17."
DOMAIN="TEST"
servers=[
{
:hostname => "VM1Danylenko." + DOMAIN,
:ip => BRIDGE_NET + "200",
:ip_int => INTERNAL_NET + "1",
:ram => 1000

```



```

},
{
:hostname => "VM2Danylenko." + DOMAIN,
:ip => BRIDGE_NET + "201",
:ip_int => INTERNAL_NET + "2",
:ram => 2000,
:hdd_name => "db1_hdd.vdi",
:hdd_size => "5000"
}
]

```

Output of Vagrant:

```

PS C:\git\danylenko> vagrant up
Bringing machine 'VM1Danylenko.TEST' up with 'virtualbox' provider...
Bringing machine 'VM2Danylenko.TEST' up with 'virtualbox' provider...
==> VM1Danylenko.TEST: Importing base box 'ubuntu_18.04_danylenko'...
==> VM1Danylenko.TEST: Matching MAC address for NAT networking...
==> VM1Danylenko.TEST: Setting the name of the VM: VM1Danylenko.TEST
==> VM1Danylenko.TEST: Clearing any previously set network interfaces...
==> VM1Danylenko.TEST: Preparing network interfaces based on configuration...
VM1Danylenko.TEST: Adapter 1: nat
VM1Danylenko.TEST: Adapter 2: bridged
VM1Danylenko.TEST: Adapter 3: intnet
==> VM1Danylenko.TEST: Forwarding ports...
VM1Danylenko.TEST: 22 (guest) => 2222 (host) (adapter 1)
==> VM1Danylenko.TEST: Running 'pre-boot' VM customizations...
==> VM1Danylenko.TEST: Booting VM...
==> VM1Danylenko.TEST: Waiting for machine to boot. This may take a few minutes...
VM1Danylenko.TEST: SSH address: 127.0.0.1:2222
VM1Danylenko.TEST: SSH username: vagrant
VM1Danylenko.TEST: SSH auth method: password
VM1Danylenko.TEST:
VM1Danylenko.TEST: Inserting generated public key within guest...
VM1Danylenko.TEST: Removing insecure key from the guest if it's present...
VM1Danylenko.TEST: Key inserted! Disconnecting and reconnecting using new SSH key...
==> VM1Danylenko.TEST: Machine booted and ready!
==> VM1Danylenko.TEST: Checking for guest additions in VM...
==> VM1Danylenko.TEST: Setting hostname...
==> VM1Danylenko.TEST: Configuring and enabling network interfaces...
==> VM2Danylenko.TEST: Importing base box 'ubuntu_18.04_danylenko'...
==> VM2Danylenko.TEST: Matching MAC address for NAT networking...
==> VM2Danylenko.TEST: Setting the name of the VM: VM2Danylenko.TEST
==> VM2Danylenko.TEST: Fixed port collision for 22 => 2222. Now on port 2200.
==> VM2Danylenko.TEST: Clearing any previously set network interfaces...
==> VM2Danylenko.TEST: Preparing network interfaces based on configuration...
VM2Danylenko.TEST: Adapter 1: nat
VM2Danylenko.TEST: Adapter 2: bridged
VM2Danylenko.TEST: Adapter 3: intnet
==> VM2Danylenko.TEST: Forwarding ports...
VM2Danylenko.TEST: 22 (guest) => 2200 (host) (adapter 1)
==> VM2Danylenko.TEST: Running 'pre-boot' VM customizations...
==> VM2Danylenko.TEST: Booting VM...
==> VM2Danylenko.TEST: Waiting for machine to boot. This may take a few minutes...
VM2Danylenko.TEST: SSH address: 127.0.0.1:2200
VM2Danylenko.TEST: SSH username: vagrant
VM2Danylenko.TEST: SSH auth method: password
VM2Danylenko.TEST: Warning: Connection aborted. Retrying...
VM2Danylenko.TEST:
VM2Danylenko.TEST: Inserting generated public key within guest...
VM2Danylenko.TEST: Removing insecure key from the guest if it's present...
VM2Danylenko.TEST: Key inserted! Disconnecting and reconnecting using new SSH key...
==> VM2Danylenko.TEST: Machine booted and ready!
==> VM2Danylenko.TEST: Checking for guest additions in VM...
==> VM2Danylenko.TEST: Setting hostname...
==> VM2Danylenko.TEST: Configuring and enabling network interfaces...
PS C:\git\danylenko>

```

VM1:

```

Last login: Sun Mar 29 20:58:14 2020 from
vagrant@VM1Danylenko:~$ date
неділя, 29 березня 2020 20:58:04 +0300
vagrant@VM1Danylenko:~$ hostname -I
10.0.2.15 192.168.1.200 172.17.17.2
vagrant@VM1Danylenko:~$

```

VM2 and connection pings

```
vagrant@VM2Danylenko:~$ date
неділя, 29 березня 2020 20:58:07 +0300
vagrant@VM2Danylenko:~$ hostname -I
10.0.2.15 192.168.1.201 172.17.17.3
vagrant@VM2Danylenko:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=51 time=37.0 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=51 time=36.7 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 36.790/36.914/37.038/0.124 ms
vagrant@VM2Danylenko:~$ ping 192.168.1.200
PING 192.168.1.200 (192.168.1.200) 56(84) bytes of data.
64 bytes from 192.168.1.200: icmp_seq=1 ttl=64 time=0.385 ms
64 bytes from 192.168.1.200: icmp_seq=2 ttl=64 time=0.700 ms
64 bytes from 192.168.1.200: icmp_seq=3 ttl=64 time=0.571 ms
^C
--- 192.168.1.200 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2022ms
rtt min/avg/max/mdev = 0.385/0.552/0.700/0.129 ms
vagrant@VM2Danylenko:~$ ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=128 time=0.740 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=128 time=0.339 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=128 time=0.364 ms
^C
--- 192.168.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.339/0.481/0.740/0.183 ms
vagrant@VM2Danylenko:~$ ping 172.17.17.2
PING 172.17.17.2 (172.17.17.2) 56(84) bytes of data.
64 bytes from 172.17.17.2: icmp_seq=1 ttl=64 time=0.833 ms
64 bytes from 172.17.17.2: icmp_seq=2 ttl=64 time=0.489 ms
^C
--- 172.17.17.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.489/0.661/0.833/0.172 ms
vagrant@VM2Danylenko:~$
```

Pings from Host, VMs are accessible though bridged adapters.

```
C:\Users\void>ping 192.168.1.200

Обмен пакетами с 192.168.1.200 по с 32 байтами данных:
Ответ от 192.168.1.200: число байт=32 время<1мс TTL=64
Ответ от 192.168.1.200: число байт=32 время<1мс TTL=64

Статистика Ping для 192.168.1.200:
    Пакетов: отправлено = 2, получено = 2, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 0 мсек, Среднее = 0 мсек
Control-C
^C
C:\Users\void>ping 192.168.1.201

Обмен пакетами с 192.168.1.201 по с 32 байтами данных:
Ответ от 192.168.1.201: число байт=32 время<1мс TTL=64
Ответ от 192.168.1.201: число байт=32 время<1мс TTL=64

Статистика Ping для 192.168.1.201:
    Пакетов: отправлено = 2, получено = 2, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 0 мсек, Среднее = 0 мсек
Control-C
^C
C:\Users\void>ping 172.17.17.3

Обмен пакетами с 172.17.17.3 по с 32 байтами данных:
Превышен интервал ожидания для запроса.

Статистика Ping для 172.17.17.3:
    Пакетов: отправлено = 1, получено = 0, потеряно = 1
    (100% потерь)
Control-C
^C
C:\Users\void>
```



```
==> VM2Danylenko.TEST: Configuring and enabling network interfaces...
PS C:\git\danylenko> vagrant halt
==> VM2Danylenko.TEST: Attempting graceful shutdown of VM...
==> VM1Danylenko.TEST: Attempting graceful shutdown of VM...
PS C:\git\danylenko> vagrant destroy
   VM2Danylenko.TEST: Are you sure you want to destroy the 'VM2Danylenko.TEST' VM? [y/N] y
==> VM2Danylenko.TEST: Destroying VM and associated drives...
   VM1Danylenko.TEST: Are you sure you want to destroy the 'VM1Danylenko.TEST' VM? [y/N] y
==> VM1Danylenko.TEST: Destroying VM and associated drives...
```

ПЕРЕЛІК ПОСИЛАНЬ

1. Oracle VM VirtualBox.User Manual <https://www.virtualbox.org/manual/>
2. Офіційна сторінка VirtualBox <https://www.virtualbox.org/>
3. Сторінка завантаження Ubuntu <https://ubuntu.com/download>
4. Сторінка документації Vagrant <https://www.vagrantup.com/docs/index.html>
5. Сторінка з інструкціями щодо інсталяції Vagrant
<https://www.vagrantup.com/docs/installation/index.html>
6. Сторінка завантаження PuTTY <https://www.putty.org/>
7. Робота з vagrantfile <http://sysadm.pp.ua/linux/sistemyvirtualizacii/vagrantfile.html>
8. Створення власного Vagrant box <http://sysadm.pp.ua/linux/sistemyvirtualizacii/vagrant-box-creation.html>