**Министерство науки и высшего образования Российской Федерации**
федеральное государственное автономное образовательное учреждение высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

# Отчет
по лабораторной работе *№1*
по дисциплине «**Инфраструктура как код**»

**Автор:**

**Зыонг Тхи Хуэ линь**

**Факультет: ИТИП**

**Группа: M34091**


Санкт-Петербург

2024

# Цель работы

Запустить приложение при помощи ansible, научиться писать плейбуки и пользоваться group vars

**Docker-образ:** https://hub.docker.com/repository/docker/timurbabs/django

Универсальный файл для вагранта: Vagrantfile

# Рекомендуемая документация

- Инфраструктура как код #1: понятие инфраструктурного кода (Лекция)
- Инфраструктура как код #2: Знакомство с Ansible (Лекция)
- Как поднимать виртуальные машины в Vagrant
- Документация Ansible
- Подборка по Ansible

# Ход работы

Установить Vagrant

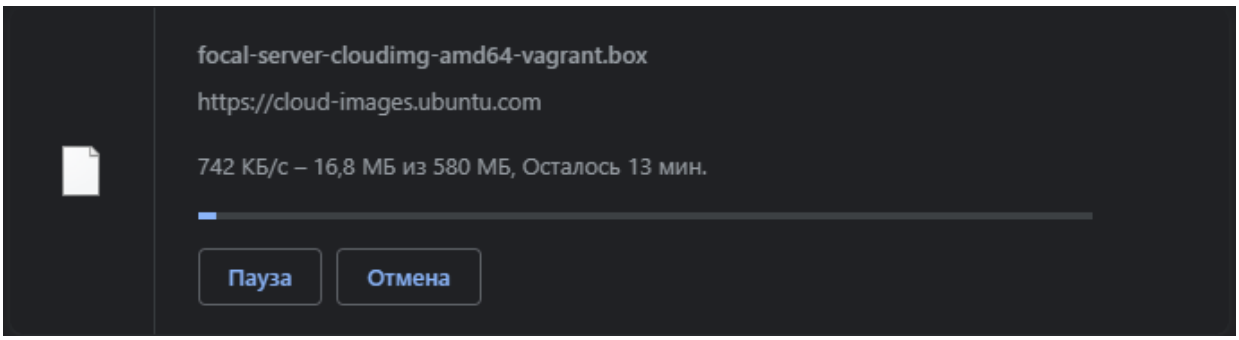
```
Microsoft Windows [Version 10.0.22631.4317]
(c) Microsoft Corporation. All rights reserved.

C:\Users\k60du>vagrant --version
Vagrant 2.4.1
```

Создать новую папку и выполнить: *vagrant init*

Modify Vagrantfile (copy content from exists file on google-drive)

```
Vagrant.configure("2") do |config|
  config.vm.box = "ubuntu/focal64"


  # config.ssh.private_key_path = "~/.ssh/id_rsa"

  SERVERS = 3
  BRIDGE = "Intel(R) Wi-Fi 6 AX200 160MHz"

  def create_host(config, hostname, ip)
    config.vm.define hostname do |host|
      # Cấu hình mạng private với IP được cung cấp
      host.vm.network "private_network", ip: ip
      # Cấu hình mạng public nếu cần kết nối cầu nối
      host.vm.network "public_network", bridge: BRIDGE
      # Thiết lập hostname cho máy
      host.vm.hostname = hostname

      host.vm.provision "shell", inline: <<-SHELL
        apt-get update
        apt-get install -y python3-minimal openssh-server
        systemctl enable ssh
        systemctl start ssh
      SHELL
      yield host if block_given?
    end
  end

  (1..SERVERS).each do |machine_id|
    create_host(config, "srv#{machine_id}", "192.168.56.#{200+machine_id}")
  end

end
```

dowload vagrant-box



Add box to vagrant

```
C:\Users\k60du\devOps-itmo-sem7>vagrant box add focal-server-cloudimg-amd64-vagrant.box --name ubuntu/focal64
==> box: Box file was not detected as metadata. Adding it directly...
==> box: Adding box 'ubuntu/focal64' (v0) for provider:
    box: Unpacking necessary files from: file://C:/Users/k60du/devOps-itmo-sem7/focal-server-cloudimg-amd64-vagrant.box
    box:
The box you're attempting to add already exists. Remove it before
adding it again or add it with the `--force` flag.

Name: ubuntu/focal64
Provider: virtualbox
Version: 0
```

Vagrant up

```
C:\Users\k60du\devOps-itmo-sem7>vagrant up
Bringing machine 'srv1' up with 'virtualbox' provider...
Bringing machine 'srv2' up with 'virtualbox' provider...
Bringing machine 'srv3' up with 'virtualbox' provider...
==> srv1: Clearing any previously set forwarded ports...
==> srv1: Clearing any previously set network interfaces...
==> srv1: Preparing network interfaces based on configuration...
    srv1: Adapter 1: nat
    srv1: Adapter 2: hostonly
    srv1: Adapter 3: bridged
==> srv1: Forwarding ports...
    srv1: 22 (guest) => 2222 (host) (adapter 1)
==> srv1: Running 'pre-boot' VM customizations...
==> srv1: Booting VM...
==> srv1: Waiting for machine to boot. This may take a few minutes...
    srv1: SSH address: 127.0.0.1:2222
    srv1: SSH username: vagrant
    srv1: SSH auth method: private key
==> srv1: Machine booted and ready!
==> srv1: Checking for guest additions in VM...
    srv1: The guest additions on this VM do not match the installed version of
    srv1: VirtualBox! In most cases this is fine, but in rare cases it can
    srv1: prevent things such as shared folders from working properly. If you see
    srv1: shared folder errors, please make sure the guest additions within the
    srv1: virtual machine match the version of VirtualBox you have installed on
    srv1: your host and reload your VM.
    srv1:
    srv1: Guest Additions Version: 6.1.32
    srv1: VirtualBox Version: 6.0
==> srv1: Setting hostname...
==> srv1: Configuring and enabling network interfaces...
==> srv1: Mounting shared folders...
    srv1: /vagrant => C:/Users/k60du/devOps-itmo-sem7
==> srv1: Machine already provisioned. Run `vagrant provision` or use the `--provision`
```

**I use SRV2 as a place to install ansible and run ansible playbook.**

Vagrant ssh srv2

```
C:\Users\k60du\devOps-itmo-sem7>vagrant ssh srv2
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-113-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon Oct 21 16:34:12 UTC 2024

  System load:  0.06              Users logged in:         0
  Usage of /:   5.5% of 38.71GB   IPv4 address for enp0s3: 10.0.2.15
  Memory usage: 22%               IPv4 address for enp0s8: 192.168.56.202
  Swap usage:   0%                IPv4 address for enp0s9: 192.168.50.245
  Processes:    125

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

235 updates can be applied immediately.
182 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

New release '22.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


Last login: Thu Oct 17 19:03:50 2024 from 10.0.2.2
vagrant@srv2:~$
```

Install ansible

```
vagrant@srv2:~$ ansible --version
ansible 2.9.6
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/vagrant/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 3.8.10 (default, Mar 15 2022, 12:22:08) [GCC 9.4.0]
```

Add Hosts

```
  GNU nano 4.8
srv1 ansible_ssh_host=192.168.56.201 ansible_user=vagrant ansible_password=vagrant
srv2 ansible_ssh_host=192.168.56.202 ansible_user=vagrant ansible_password=vagrant
srv3 ansible_ssh_host=192.168.56.203 ansible_user=vagrant ansible_password=vagrant

[app]
srv1

[docker]
srv1
```

Ping app

```
vagrant@srv2:~/iac/lab1$ ansible -i hosts -m ping app
srv1 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

Write playbook to install docker (*docker-playbook.yaml*)

```
  GNU nano 4.8                                                    docker-playbook.yaml
---
- name: Install Docker on Ubuntu
  hosts: app
  become: true
  vars:
    ansible_user: "vagrant"
  tasks:
    - name: Update the apt package cache
      apt:
        update_cache: yes

    - name: Add Docker API repository
      apt_repository:
        repo: "deb [arch=amd64] https://download.docker.com/linux/ubuntu {{ ansible_distribution_release }} stable"
        state: present

    - name: Update the apt package cache again (after adding Docker repo)
      apt:
        update_cache: yes

    - name: Install Docker
      apt:
        name: docker-ce
        state: present

    - name: Add user to docker group (optional)
      user:
        name: "{{ ansible_user }}"
        groups: docker
        append: yes

    - name: Start Docker service
      systemd:
        name: docker
        state: started
        enabled: yes

    - name: Install pip3
      apt:
        name: python3-pip
        state: present

    - name: Install Docker SDK for python
      pip:
        name: docker
        state: present
```

Ansible-playbook –I hosts docker-playbook.yaml

```
vagrant@srv2:~/iac/lab1$ ansible-playbook -i hosts docker-playbook.yaml

PLAY [Install Docker on Ubuntu] ***********************************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************************************************
ok: [srv1]

TASK [Update the apt package cache] *******************************************************************************************************************
changed: [srv1]

TASK [Add Docker API repository] **********************************************************************************************************************
ok: [srv1]

TASK [Update the apt package cache again (after adding Docker repo)] **********************************************************************************
changed: [srv1]

TASK [Install Docker] *********************************************************************************************************************************
changed: [srv1]

TASK [Add user to docker group (optional)] ************************************************************************************************************
changed: [srv1]

TASK [Start Docker service] ***************************************************************************************************************************
ok: [srv1]

TASK [Install pip3] ***********************************************************************************************************************************
changed: [srv1]

TASK [Install Docker SDK for python] ******************************************************************************************************************
changed: [srv1]

PLAY RECAP ********************************************************************************************************************************************
srv1                       : ok=9    changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Clone and Run repo Django:

run with docker:

```
vagrant@srv2:~/iac/lab1$ cat app-docker-playbook.yaml
- name: "Run Django Application"
  hosts: "app"
  become: true
  tasks:
    - name: "Run Django Application"
      docker_container:
        name: django-app
        image: timurbabs/django:latest
        state: started
        ports:
          - "8000:8000"
```

```
vagrant@srv2:~/iac/lab1$ ansible-playbook -i hosts app-docker-playbook.yaml

PLAY [Run Django Application] *************************************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************************************************
ok: [srv1]

TASK [Run Django Application] *************************************************************************************************************************
changed: [srv1]

PLAY RECAP ********************************************************************************************************************************************
srv1                       : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
vagrant@srv1:~$ docker ps
CONTAINER ID   IMAGE                     COMMAND                  CREATED         STATUS         PORTS                    NAMES
d8a398d2bb15   timurbabs/django:latest   "/bin/sh -c 'gunicor…"   5 seconds ago   Up 4 seconds   0.0.0.0:8000->8000/tcp   django-app
```