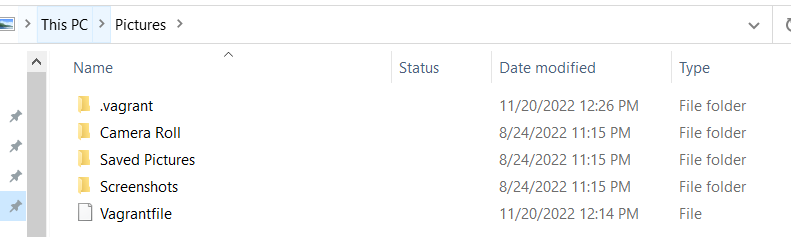
# Khởi tạo vagrant

Ban đầu nếu ko có Vagrantfile ta sẽ khởi tạo vagrant bằng lệnh

vagrant init bento/ubuntu-18.04

tức là n sẽ pull image ubuntu-18.04 về để làm image chạy máy ảo,

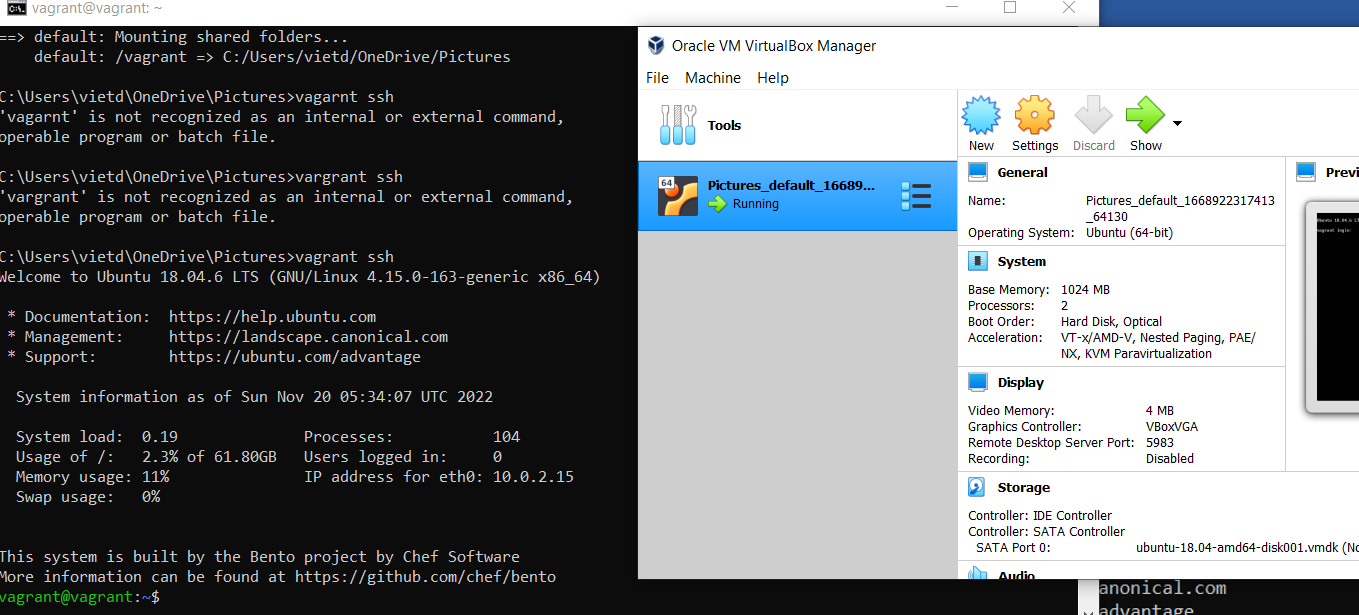
khi chạy xog n sẽ tạo ra 1 file tên Varantfile



vagrant up --provider=virtualbox

khởi chạy máy ảo vagrant

nhớ bật tính năng hyber-V



Oke n đã lên

Lưu ý 1 số version của virtualbox hỗ trợ việc chạy vagrant

## Câu lệnh

* Khởi tạo 1 vagrantfile mặc định

vagrant init bentos/ubuntu-18.04

* khởi chạy vagrant qua virtualbox

vagrant up --provider=virtualbox

* stop máy ảo, shutdown

vagrant halt

* restart máy ảo

vagrant reload

* ssh vào máy ảo

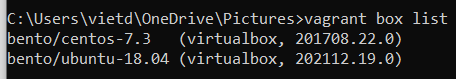
vagrant ssh

* add thêm VM

vagrant box add bento/centos-7.3

* list

vagrant box list



* xóa VM

vagrant remove bento/centos-7.3

* xóa VM

vagrant destroy

## config file vagrant

Guest: máy khách – máy virtualbox

Host: máy chủ vật lý localhost - window

Config cho server nginx

config.vm.network "forwarded\_port", guest: 80, host: 8080, host\_ip: "127.0.0.1"

chỉ cho phép IP localhost window access vào, nếu muốn config để mọi thằng access vào

config.vm.network "forwarded\_port", guest: 80, host: 8080, host\_ip: "0.0.0.0"

* ánh xạ folder máy khách - máy virtualbox

config.vm.synced\_folder "./html/", "/var/www/html/"

* config cấu hình

   config.vm.provider "virtualbox" do |vb|

     # Display the VirtualBox GUI when booting the machine

     vb.gui = false # giao diện đồ họa

     # Customize the amount of memory on the VM:

     vb.memory = "1024"

   end

* cài đặt phần mềm khi run VM

   config.vm.provision "shell", inline: <<-SHELL

     apt-get update

     apt-get install -y apache2

   SHELL

end

Vagrantfile

Vagrant.configure("2") do |config|

  config.vm.define "web-server"

  config.vm.hostname = "web-server"     # tên hostname của máy ảo

  config.vm.box = "bento/ubuntu-18.04"

  config.vm.network "forwarded\_port", guest: 80, host: 8080, host\_ip: "127.0.0.1"

   config.vm.synced\_folder "./html/", "/var/www/html/"

   config.vm.provider "virtualbox" do |vb|

     vb.name = "web-server1"          # tên của con VM được tạo trong virutalbox

     vb.gui = false # giao diện đồ họa

     # Customize the amount of memory on the VM:

     vb.memory = "1024"

   end

   config.vm.provision "shell", inline: <<-SHELL

     apt-get update

     apt-get install -y apache2

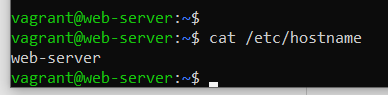
   SHELL

   config.vm.provision "shell", run: "always", inline: <<-SHELL

  echo "hello vagrant"

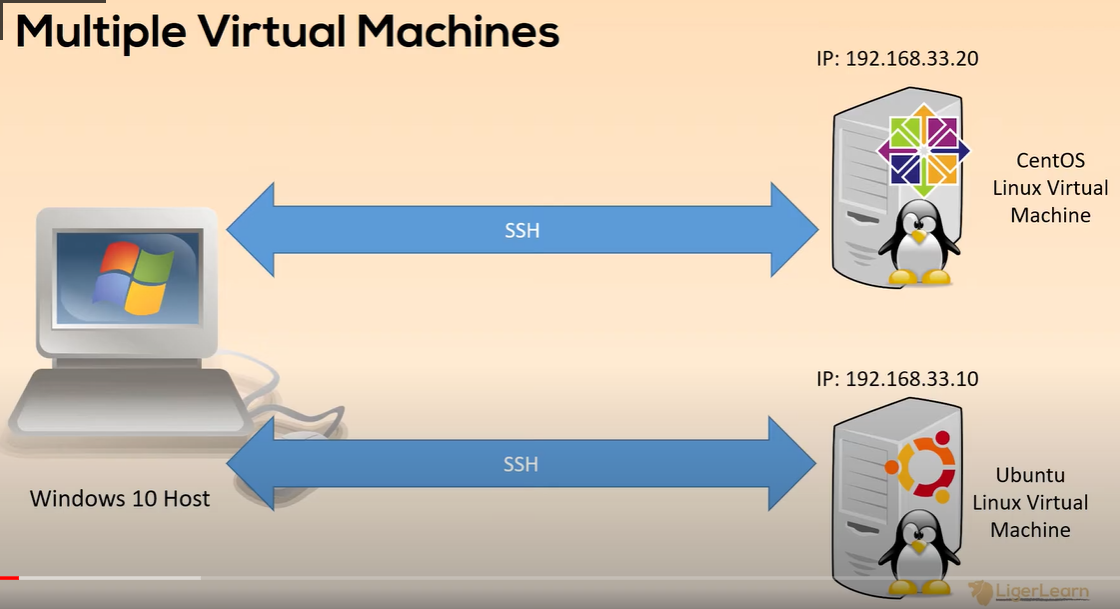
   SHELL

end



# Multiple VMs and Private Networks

Bài này tôi sẽ tạo 2 máy ảo dùng 2 image khác nhau



Vagrantfile

Vagrant.configure("2") do |config|

  config.vm.define "ubuntu-vm"  do  |vm1|       # tên VM

  vm1.vm.hostname = "ubuntu-vm"

  vm1.vm.box = "bento/ubuntu-18.04"

  vm1.vm.network "private\_network", ip: "192.168.10.10"

   vm1.vm.provider "virtualbox" do |vb|         # tên vitualbox

     vb.name = "ubuntu-vm"

     vb.gui = false

     vb.memory = "1024"

   end

  end

  config.vm.define "centos-vm"  do  |vm2|

  vm2.vm.hostname = "centos-vm"

  vm2.vm.box = "bento/centos-7.3"

  vm2.vm.network "private\_network", ip: "192.168.10.20"

   vm2.vm.provider "virtualbox" do |vb|

     vb.name = "centos-vm"

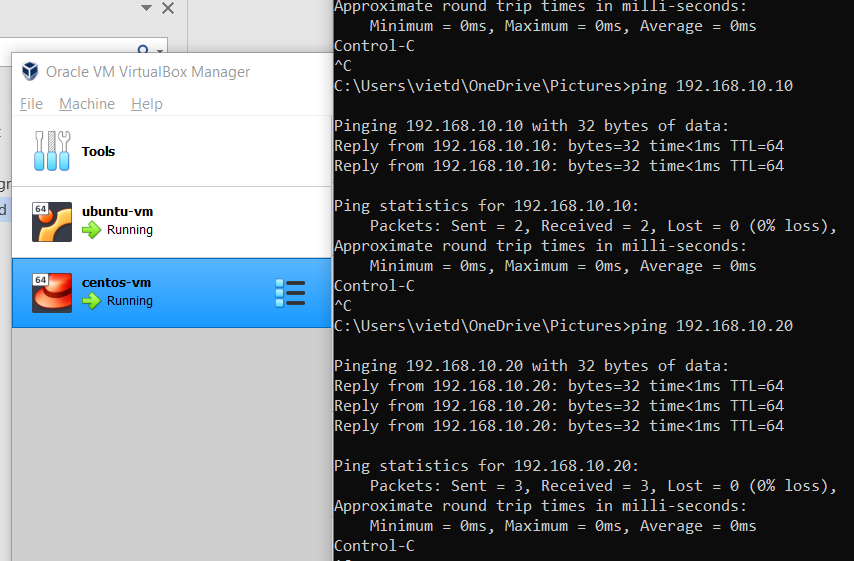
     vb.gui = false

     vb.memory = "1024"

   end # end vb

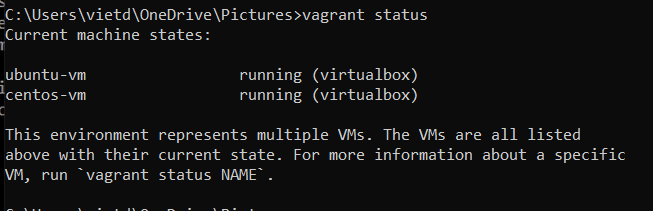
  end # end vm2

end # end config



Oke

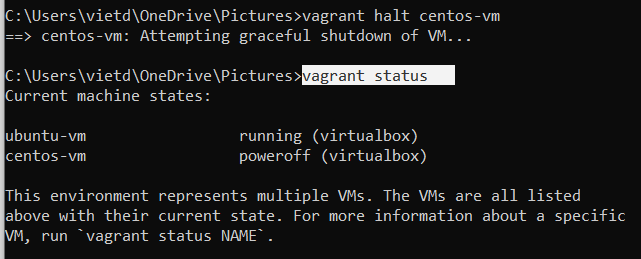
vagrant status



2 vm đang chạy

* stop 1 VM

vagrant halt centos-vm



* start lại VM

vagrant up centos-vm

* xóa 1 VM

vagrant destroy ubuntu-vm

## Multi VM

Vagrant.configure("2") do |config|

    (1..3).each do |i|

        if i <= 3

            config.vm.define "fnb-linux-node-#{i}" do |node|

                node.vm.box = "bento/ubuntu-20.04"

                node.vm.hostname = "fnb-linux-node-#{i}"

                # Bridged network

                node.vm.network "private\_network", ip:"192.168.58.#{i}"

                # Provider-specific configuration

                node.vm.provider "vmware\_workstation" do |vb|

                    # Customize the amount of memory on the VM

                    vb.memory = "1500"

                    # Specify machine name

                    #vb.name = "fnb-linux-node-#{i}"

                end

            end

        end

    end

end

Vagrant.configure("2") do |config|

  config.vm.define "win"  do  |vm3|

  vm3.vm.hostname = "win"

  vm3.vm.box = "StefanScherer/windows\_2016"

  vm3.vm.communicator = "winrm"

  vm3.winrm.username = "vagrant"

  vm3.winrm.password = "vagrant"

  vm3.vm.network "private\_network", ip: "192.168.58.6"

   vm3.vm.provider "virtualbox" do |vb, override|

     vb.name = "win"

     vb.gui = false

     vb.customize ["modifyvm", :id, "--memory", 2048]

     vb.customize ["modifyvm", :id, "--cpus", 2]

   end

  end

end

BOX\_IMAGE = "centos/8"

MEMORY = "2048"

CPU = "2"

Vagrant.configure("2") do |config|

  config.vm.define "node1" do |subconfig|

    subconfig.vm.box = BOX\_IMAGE

    subconfig.vm.hostname = "node1"

  subconfig.vm.network :private\_network, ip: "192.168.56.111"

  subconfig.vm.provider :virtualbox do |vb|

          vb.customize ["modifyvm", :id, "--memory", MEMORY]

          vb.customize ["modifyvm", :id, "--cpus", CPU]

  end

  end

  #Install some additional packages

  config.vm.provision "shell", inline: <<-SHELL

  sudo su

  sed -i 's/PasswordAuthentication no/PasswordAuthentication yes/g' /etc/ssh/sshd\_config

    systemctl restart sshd.service

  yum -y install nano

  yum -y install net-tools

  SHELL

end

disable firewall

Set-NetFirewallProfile -Enabled False