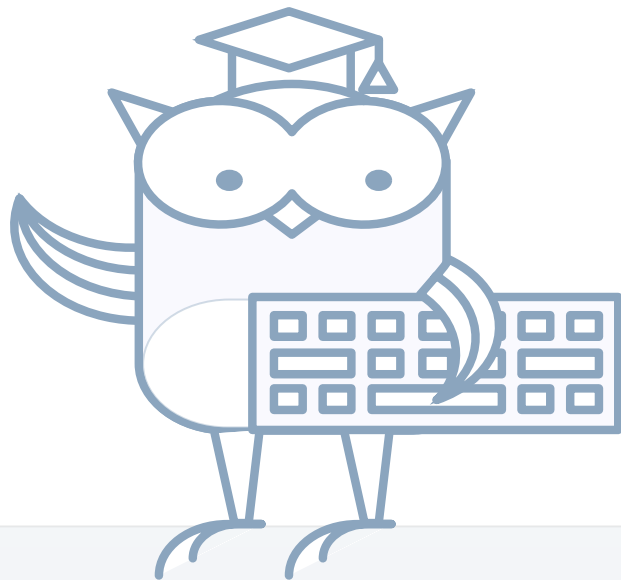




ОНЛАЙН-ОБРАЗОВАНИЕ

Vagrant Vagrantfile



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

```
  config.vm.define "vm-1" do |subconfig|
    subconfig.vm.box = "centos/7"
    subconfig.vm.hostname="vm-1"
    subconfig.vm.network :private_network, ip: "192.168.50.11"
    subconfig.vm.provider "virtualbox" do |vb|
      vb.memory = "1024"
      vb.cpus = "1"
    end
  end
```

```
  config.vm.define "vm-2" do |subconfig|
    subconfig.vm.box = "centos/7"
    subconfig.vm.hostname="vm-2"
    subconfig.vm.network :private_network, ip: "192.168.50.12"
    subconfig.vm.provider "virtualbox" do |vb|
      vb.memory = "1024"
      vb.cpus = "1"
    end
  end
```

```
config.ssh.insert_key = false
config.ssh.private_key_path = ['~/.vagrant.d/insecure_private_key', '~/.ssh/id_rsa']
config.vm.provision "file", source: "~/.ssh/id_rsa.pub", destination: "~/.ssh/authorized_keys"
```

```
end
```

На слайде ранее приведен пример vagrantfile для создания 2х вм на базе образа centos 7.
Для определения какие то параметров используется конструкция вида:

```
Vagrant.configure(2) do |config|  
end
```

Сначала всегда идет описание и потом оно закрывается при помощи end (Прямую аналогию можно провести с ruby)

Секция вида:

```
config.vm.define "vm-1" do |subconfig|  
  subconfig.vm.box = "centos/7"  
  subconfig.vm.hostname="vm-1"  
  subconfig.vm.network :private_network, ip: "192.168.50.11"  
  subconfig.vm.provider "virtualbox" do |vb|  
    vb.memory = "1024"  
    vb.cpus = "1"  
  end  
end
```

Служит для описание конкретной виртуальной машины. Путем комбинирования этих параметров можно создавать практически любые комбинации.

Дополнительные опции которые можно применить для конфигурирования Вы можете найти в офф документации.

При таком подходе автоматически отпадает необходимость в пробросе портов в ВМ, так как создается отдельная сеть. Трафик без проблем и ручных вмешательств маршрутизируется туда.

В упрощенном варианте конфигурация конкретной вм представлена следующей конструкцией:

```
config.vm.define "vm-1" do |subconfig|  
  
  end
```

Где действия по описанию вм должны начинаться с subconfig.

Секция вида:

```
config.ssh.insert_key = false
config.ssh.private_key_path = ['~/.vagrant.d/insecure_private_key', '~/.ssh/id_rsa']
config.vm.provision "file", source: "~/.ssh/id_rsa.pub", destination: "~/.ssh/authorized_keys"
```

Служит для добавления своих, созданных заранее ssh ключей. (Сделано исключительно для личного удобства ^_^)

При этом файл инвентаря для ansible может выглядеть следующим образом:

```
vm-1 ansible_host=192.168.50.11
vm-2 ansible_host=192.168.50.12
```

Сделано это для удобства провижининга. Провижининг VM я выделяю в отдельное действие. Но вам ничего не мешает добавить дополнительный блок такого вида в vagrantfile:

```
config.vm.provision "ansible" do |ansible|
  ansible.playbook = "provision.yml"
end
```

Где за провижининг отвечает заранее написанный плейбук provision.yml