



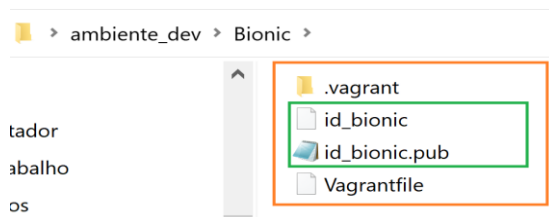
Pastas compartilhadas – Synced folder

Provisionamento é deixar a máquina provisionando com recursos necessários para a determinada função dela. Acontece que **muitas das vezes esses determinados recursos não precisam somente ser instalados, eles também precisam ser configurados**.

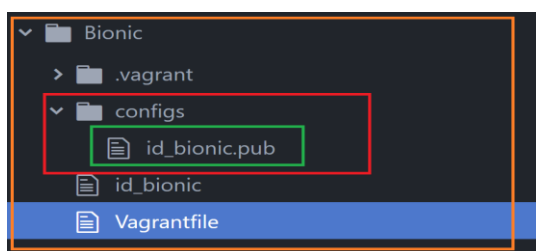
Levando em consideração essa configuração seria bom que pudéssemos levar arquivos da **máquina local(host)** para a **máquina virtual(guest)**, na verdade a gente pode! Através dos **Synced folder/Pastas compartilhadas**, ou seja, pastas que existem na **máquina host**, mas que a **máquina guest** conhece. Quando a **máquina virtual** sobe, dá pra ver a **menção a essas pastas**:

```
lucas@DESKTOP-AST3N3N-MINGW64 ~/Desktop/ambiente_dev/Bionic
$ vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
=> default: Checking if box 'hashicorp/bionic64' version '1.0.282' is up to date...
=> default: Clearing any previously set forwarded ports...
=> default: Clearing any previously set network interfaces...
=> default: Preparing network interfaces based on configuration...
default: Adapter 1: nat
default: Adapter 2: bridged
=> default: Forwarding ports...
default: 80 (guest) => 8090 (host) (adapter 1)
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: Warning: Connection reset. Retrying...
=> 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: 6.0.10
default: VirtualBox Version: 6.1
=> default: Configuring and enabling network interfaces...
=> default: Mounting shared folders...
=> default: /vagrant => C:/Users/lucas/Desktop/ambiente_dev/Bionic
=> default: Machine already provisioned. Run 'vagrant provision' or use the '--provision'
=> default: flag to force provisioning. Provisioners marked to run always will still run.
```

Um bom exemplo para usar as pastas compartilhadas pode ser a configuração das **chaves de SSH criadas**, onde meu **host** e **equipe de desenvolvimento** vai possuir a **chave privada** enquanto a **máquina virtual** vai ter a **chave pública**, dessa maneira **permitindo conexões SSH através das chaves**:



Por **boa prática** é bom organizar o que é de configuração, então vou jogar a **chave pública** pra uma **pasta de configurações**:



Agora é necessário mapear que essa pasta de **configs** é uma **pasta shared/synced** e já que o vagrant é infra como código, obviamente esse mapeamento vai ser via **Vagrantfile**, vou aproveitar e **já tirar**

a pasta **vagrant dos compartilhados**, já que eu não quero e nem preciso que a **máquina virtual** a veja.

A propriedade de **synced folders** recebe a **pasta a ser mapeada**, como você quer que ela apareça na **máquina virtual** e um **parâmetro booleano** caso ela seja desabilitada:

```
Vagrant.configure("2") do |config|
  config.vm.box = "hashicorp/bionic64"
  config.vm.network "forwarded_port", guest: 80, host: 8090
  config.vm.network "public_network", ip: "192.168.100.41", bridge:

  config.vm.provision "shell", inline: "echo hello world >> hello.t

  config.vm.synced_folder "./configs", "/configs"
  config.vm.synced_folder ".", "/vagrant", disabled: true
end
```

****Aqui é tudo navegação em pasta, então o "." representa a pasta origem do arquivo, daí você pode ir mapeando com "/" a partir do "." ****

```
lucas@DESKTOP-A5T3N3N MINGW64 ~/Desktop/ambiente_dev/Bionic
$ vagrant reload
==> default: Attempting graceful shutdown of VM...
==> default: Checking if box 'hashicorp/bionic64' version '1.0.282' is up to date...
==> default: Clearing any previously set forwarded ports...
==> default: Preparing network interfaces based on configuration...
==> default: Adapter 1: nat
==> default: Adapter 2: bridged
==> default: Forwarding ports...
==> default: 80 (guest) => 8090 (host) (adapter 1)
==> 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: 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: 6.0.10
==> default: VirtualBox Version: 6.1
==> default: Configuring and enabling network interfaces...
==> default: Mounting shared folders...
==> default: /configs => C:/Users/lucas/Desktop/ambiente_dev/Bionic/configs
==> default: Machine already provisioned. Run 'vagrant provision' or use the '--provision'
==> default: flag to force provisioning. Provisioners marked to run always will still run.
```

Agora que a **chave pública** está em uma **pasta compartilhada**, eu posso acessar ela dentro da minha **máquina virtual** e copia-la para a pasta de **"/.ssh/authorized_keys"** de forma **AUTOMÁTICA**, toda vez que eu iniciar essa **máquina virtual**.

É bem simples fazer isso, é só usar o **provisionamento do shell** para programar que a **máquina virtual pegue a chave pública da pasta compartilhada** e **faça uma saída na pasta de ssh** no arquivo de **chaves autorizadas**:

```
Vagrant.configure("2") do |config|
  config.vm.box = "hashicorp/bionic64"
  config.vm.network "forwarded_port", guest: 80, host: 8090
  config.vm.network "public_network",
    ip: "192.168.100.41", bridge: "Realtek PCIe GbE Family Controller #2"

  config.vm.provision "shell",
    inline: "cat /configs/id_bionic.pub >> /.ssh/authorized_keys"

  config.vm.synced_folder "./configs", "/configs"
  config.vm.synced_folder ".", "/vagrant", disabled: true
end
```

Pra garantir que esse código como infra está funcionando legal eu vou **destruir a máquina virtual** e **subir ela de novo**, ela deve copiar a **chave pública para a pasta de chaves autorizadas da VM** e eu **devo poder me conectar com a chave privada**:

```
lucas@DESKTOP-A5T3N3N MINGW64 ~/Desktop/ambiente_dev/Bionic
$ vagrant destroy
   default: Are you sure you want to destroy the 'default' VM? [y/N] y
==> default: Forcing shutdown of VM...
==> default: Destroying VM and associated drives...
```

```
lucas@DESKTOP-A5T3N3N MINGW64 ~/Desktop/ambiente_dev/Bionic
$ vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Importing base box 'hashicorp/bionic64'...
==> default: Matching MAC address for NAT networking...
==> default: Checking if box 'hashicorp/bionic64' version '1.0.282' is up to date...
==> default: Setting the name of the VM: Bionic_default_1644449261754_42839
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
   default: Adapter 1: nat
   default: Adapter 2: bridged
==> default: Forwarding ports...
   default: 80 (guest) => 8090 (host) (adapter 1)
   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: Warning: Connection reset. Retrying...
   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: 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:
   default: Guest Additions Version: 6.0.10
   default: VirtualBox Version: 6.1
==> default: Configuring and enabling network interfaces...
==> default: Mounting shared folders...
   default: /configs => C:/Users/lucas/Desktop/ambiente_dev/Bionic/configs
==> default: Running provisioner: shell...
   default: Running: inline script
```

Em tese devo poder me conectar usando a **chave privada** agora:

```
lucas@DESKTOP-A5T3N3N MINGW64 ~/Desktop/ambiente_dev/Bionic
$ ssh -i id_bionic vagrant@192.168.100.41
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-58-generic x86_64)

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

System information as of Wed Feb  9 23:59:46 UTC 2022

System load:  0.0               Processes:    88
Usage of /:   2.5% of 61.80GB    Users logged in: 0
Memory usage: 11%              IP address for eth0: 10.0.2.15
Swap usage:   0%                IP address for eth1: 192.168.100.41

0 packages can be updated.
0 updates are security updates.

vagrant@vagrant:~$ |
```