

## Mais Vagrant: Ambientes

Vamos dar uma olhada no nosso `Vagrantfile`:

```
Vagrant.configure("2") do |config|
  config.vm.box = "hashicorp/precise32"

  config.vm.define :web do |web_config|
    web_config.vm.network "private_network", ip: "192.168.50.10"
  end
end
```

Repare que definimos uma VM com o nome `web`. Esse nome pode ser utilizado na linha de comando:

```
vagrant up web
```

e para parar a VM:

```
vagrant halt web
```

Como temos apenas uma VM definida podemos omitir esse nome (o que fizemos até agora), mas nada impede declarar outras VMs no mesmo arquivo. Veja o exemplo:

```
Vagrant.configure("2") do |config|
  config.vm.define :web do |web_config|
    web_config.vm.box = "hashicorp/precise32"
    web_config.vm.network "private_network", ip: "192.168.50.10"
  end

  config.vm.define :prod do |prod_config|
    prod_config.vm.box = "centos/7"
    prod_config.vm.network "private_network", ip: "192.168.50.11"
  end
end
```

Repare que definimos mais uma VM com o nome `prod`. Ela é baseada no centos e tem um outro IP. Para inicializá-la basta usar o nome na linha de comando:

```
vagrant up prod
```

Também podemos chamar as duas VMs:

```
vagrant up web prod
```

ou apenas usar `vagrant up` para subir todas as VMs declaradas, mas tenha cuidado pois agora somos **obrigado** de definir o nome na hora de se conectar com `ssh` :

```
vagrant ssh prod
```

*Obs: Lembrando que o CentOS não vem com Puppet pre-instalado.*