

Aula 2 - IAC com Vagrant e Ansible

Neste repositório, vamos aprender a usar o Vagrant e o Ansible para criar e gerenciar máquinas virtuais. Também vamos aprender a usar o Ansible para fazer o deploy do site "mundo invertido"

Este documento também está disponível em formato PDF (docs/README.pdf) e HTML (docs/README.html) para que você possa visualizá-lo offline.

Tabela de conteúdos

- [Pré-requisitos](#)
- [Passo a passo](#)
- [Erros conhecidos](#)
- [Saiba mais](#)

Pré-requisitos

- Instalação do VirtualBox
 - <https://www.virtualbox.org/wiki/Downloads>
- Instalação do Vagrant
 - https://developer.hashicorp.com/vagrant/downloads?product_intent=vagrant
- Instalação do Visual Studio Code
 - <https://code.visualstudio.com>

Passo a passo

1. Comece fazendo o clone do repositório:

```
git clone https://gitlab.com/dvp2025-2/aula-2-iac-com-vagrant-e-ansible.git
cd aula-2-iac-com-vagrant-e-ansible
```

[!NOTE] Se você não tem o Git instalado ou não sabe usá-lo, sem problema algum, você pode simplesmente fazer o [download do repositório](#) e descompactá-lo em sua pasta/diretório de trabalho ou na pasta/diretório de seu usuário

2. Crie uma pasta/diretório para o projeto (Ex.: [aula-2-iac-com-vagrant-e-ansible](#)) em sua pasta/diretório de trabalho ou na pasta/diretório de seu usuário

Se preferir, fazer pelo terminal:

```
cd seu_diretorio_de_trabalho
mkdir aula-2-iac-com-vagrant-e-ansible
cd aula-2-iac-com-vagrant-e-ansible
```

3. Se você estiver no Windows Explorer, clique com o botão direito do mouse sobre a pasta/diretório criada e selecione "Open in Terminal"
4. Já dentro do terminal, execute o seguinte comando:

```
vagrant init
```

O comando irá gerar um arquivo chamado **Vagrantfile**

5. Execute o seguinte comando para editar o arquivo **Vagrantfile**:

```
code .
```

6. Inclua as seguintes informações no arquivo **Vagrantfile**:

```
# -*- mode: ruby -*-
# vi: set ft=ruby :

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

  config.vm.network "forwarded_port", guest: 80, host: 8080

  config.vm.provider "virtualbox" do |vb|
    vb.memory = "1024"
    vb.cpus = 1
    vb.name = "nginx - webserver"
  end

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

[!NOTE] Esse arquivo Vagrantfile é o arquivo declarativo que informa ao Vagrant como deve ser a máquina virtual que será criada e como deve ser configurada, usando o Ansible para fazer a configuração

7. Agora vamos criar o arquivo de playbook do Ansible:

```
code playbook.yml
```

[!NOTE] Esse arquivo playbook.yml é o arquivo declarativo que informa ao Ansible como deve ser o estado desejado do sistema, ou seja, o que o Ansible deve fazer para configurar a

máquina virtual e garantir que ela esteja sempre neste estado

8. Agora inclua as seguintes informações no arquivo `playbook.yml`:

```
---
- hosts: all
  become: yes
  tasks:
    - name: Atualiza o cache do apt
      apt:
        update_cache: yes
      tags:
        - packages

    - name: Instala o Nginx
      apt:
        name: nginx
        state: present
      tags:
        - packages

    - name: Copia a página web para o diretório do Nginx
      copy:
        src: files/
        dest: /var/www/html
        owner: www-data
        group: www-data
        mode: '0644'
      notify:
        - Reiniciar Nginx

  handlers:
    - name: Reiniciar Nginx
      service:
        name: nginx
        state: restarted
```

9. Agora podemos iniciar o provisionamento da máquina virtual:

```
vagrant up
```

10. Assim que a máquina virtual for provisionada, podemos acessar o site pelo navegador:

`http://localhost:8080`

O resultado esperado é esse:



11. **[Desafio Opcional]** Tente usar o módulo `template` do Ansible para copiar o arquivo `index.html` e fazer uma alteração nele ao invés do módulo `copy`

[!TIP] Comece criando um arquivo chamado `index.html.j2` no diretório `files`:

```
code files/index.html.j2
```

[!TIP] Segue a documentação do módulo `template`:

https://docs.ansible.com/ansible/latest/collections/ansible/builtin/template_module.html

12. **[Desafio Opcional]** Se você terminou de configurar o módulo `template`, você tem pedir para o Vagrant realizar o provisionamento novamente:

```
vagrant provision
```

13. **[Desafio Opcional]** Acesse o site novamente e verifique se a alteração foi aplicada:
<http://localhost:8080>

14. Se você quiser desprovisionar a máquina virtual, execute o seguinte comando:

```
vagrant destroy
```

Erros conhecidos

No **Windows**, caso você receba este erro do VirtualBox:

[!CAUTION] **VT-x is not available. (VERR_VMX_NO_VMX)**

Significa que o **Hyper-V** está habilitado e configurado como virtualizador padrão no Windows, pois ele e o VirtualBox não podem coexistir.

Para resolver isso, você precisa desabilitá-lo, para isso siga estes passos no Terminal:

```
bcdedit /set hypervisorlaunchtype off
```

Depois de desabilitá-lo, reinicie o computador e tente novamente.

Saiba mais

- [Explorando módulos do Ansible](#)
- [Documentação dos módulos do Ansible](#)
- [Documentação do Ansible](#)
- [Documentação do Vagrant](#)
- [Documentação do VirtualBox](#)