



Minicurso

Introdução à Infraestrutura

como Código (IaC) com

Ansible

Marcos Vilela

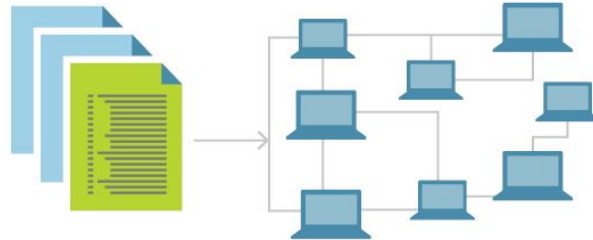


Sumário

- Apresentação
- O que é IaC
- O que é Automação
- O que é o Ansible
- Arquitetura do Ansible
- Playbook e inventário
- Roles e Handlers
- Principais módulos
- Benefícios
- Instalando o Ansible
- Na prática
- Construindo o playbook
- Executando playbooks
- Boas práticas
- Referências

O que é IaC

- Abordagem para gerenciar e provisionar infraestrutura de TI através de código, ao invés de processos manuais.



- Nesta abordagem, arquivos de configuração que incluem as especificações da infraestrutura são criados, facilitando a edição e a distribuição das suas configurações.



O que é IaC

- **Reprodutibilidade:** Infraestrutura definida em arquivos de código, permitindo a criação e replicação de ambientes de forma consistente.
- **Gerenciamento de Versões:** Código de infraestrutura versionado, facilitando o controle de alterações e o rollback.
- **Automação:** Processos de provisionamento e configuração automatizados, reduzindo erros humanos e aumentando a eficiência.



Princípios do IaC

Idempotência: execução repetida não altera o estado desejado.

Garante que a aplicação da mesma configuração várias vezes resulte no mesmo estado de infraestrutura sem efeitos colaterais indesejados.

Imutabilidade: substituição de recursos em vez de modificação direta.

Ela elimina todos os desvios de configuração e torna ainda mais fácil manter a consistência entre os ambientes de teste e implementação.

Declaratividade: definição do estado desejado, não dos passos para alcançá-lo.

Principais ferramentas



puppet



CHEF™



Terraform



HashiCorp

Packer



HashiCorp

Vagrant



CloudFormation



Google

Cloud Deployment
Manager



O que é Automação

É a utilização de ferramentas e scripts para executar tarefas repetitivas de forma automática.

Casos de uso:

- Orquestração de tarefas.
- Configuração de software.
- Provisionamento de servidores.
- Deploy de aplicações.

Benefícios:

- Redução de erros humanos.
- Aumento da velocidade e eficiência.
- Liberação de tempo para tarefas mais estratégicas.
- Maior escalabilidade e confiabilidade.



Níveis de automação

Níveis de automação:

- Tarefas simples: scripts para tarefas pontuais.
- Fluxos de trabalho complexos: orquestração de múltiplas tarefas.
- Orquestração de sistemas: gerenciamento completo de infraestrutura.

Entrega contínua: automação do ciclo de vida do software.

Cultura DevOps: Automação, colaboração e feedback contínuo.



O que é o Ansible


É uma ferramenta de automação de código aberto para provisionamento, gerenciamento de configuração e orquestração de aplicações.



ANSIBLE

O Ansible, como ferramenta de TI, foi desenvolvido por Michael DeHaan em 2012. Em 2013, DeHaan, Timothy Gerla e Saïd Ziouani fundaram a Ansible, Inc. para patrocinar e dar suporte comercial ao Ansible. Em 2015, a Red Hat adquiriu o Ansible.






Ele é escrito em Python e pode ser usado em qualquer operação administrativa que antes era realizada manualmente.

Sintaxe YAML, facilitando a criação e compreensão dos playbooks.

Agente-less, não requer instalação de agentes nos nós gerenciados, utilizando SSH para comunicação. Porta SSH deve estar disponível.



Licença Pública Geral GNU (GPLv3) - É uma licença de software livre que permite que usuários usem, modifiquem e distribuam software sem restrições.

Produto comercial é o Ansible Automation Platform, e tem a versão open source: AWX.

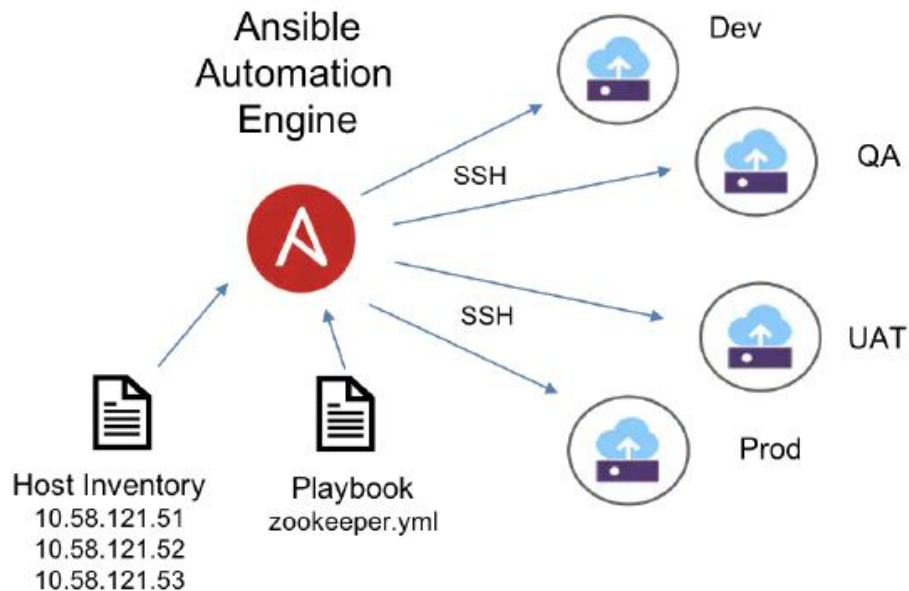
O Ansible tem as seguintes vantagens:

- Fácil aprendizado
- Não precisa de agents
- Grande comunidade: suporte e recursos abundantes
- Multiplataforma: Gerencia diversos sistemas operacionais
- Agnóstico de Nuvem

Desvantagens:

- O Ansible não tem noção de estado
- Não possui recursos avançados como execução paralela ou orquestração

Arquitetura do Ansible



Control Node: Máquina onde o Ansible é instalado e de onde os playbooks são executados.

Managed Nodes: Servidores ou dispositivos que são gerenciados pelo Ansible.

Inventário: Arquivo que lista os managed nodes e seus respectivos grupos.

Playbooks: Arquivos YAML que definem as tarefas a serem executadas nos managed nodes.

Módulos: Componentes que executam ações específicas nos managed nodes.

Arquitetura do Ansible



layout de diretórios

```
inventories/  
  production/  
    hosts                # inventory file for production servers  
  
  staging/  
    hosts                # inventory file for staging environment  
  
site.yml  
webservers.yml  
dbservers.yml  
  
roles  
  common/  
  webtier/  
  monitoring/  
  fooapp/
```

Inventário



hosts.yaml

```
mail.example.com  
192.168.0.1
```

```
[webservers]  
foo.example.com  
192.168.2.1
```

```
[dbservers]  
one.example.com  
192.168.3.1
```

Playbook



```
playbook.yaml

---
- name: Intro to Ansible Playbooks
  hosts: all

  tasks:
    - name: Copy file hosts with permissions
      ansible.builtin.copy:
        src: ./hosts
        dest: /tmp/hosts_backup
        mode: '0644'

    - name: Add the user 'bob'
      ansible.builtin.user:
        name: bob
        become: yes
        become_method: sudo
```



Roles e Handlers

Roles: Permitem organizar os playbooks em estruturas reutilizáveis.

Handlers: Tarefas que são executadas apenas quando notificadas por outras tarefas.

Modularidade: Roles e handlers facilitam a criação de playbooks mais complexos e reutilizáveis.



Roles e Handlers

layout de diretórios para role

```
roles/  
  webserver/  
    tasks/  
      main.yml  
    templates/  
      index.html.j2  
  vars/  
    main.yml  
  handlers/  
    main.yml
```

main.yml

```
- name: Instalar o servidor web  
  apt:  
    name: apache2  
    state: present  
  
- name: Copiar o arquivo index.html  
  template:  
    src: index.html.j2  
    dest: /var/www/html/index.html  
    notify: restart apache
```

main.yml

```
- name: restart apache  
  service:  
    name: apache2  
    state: restarted
```



Principais módulos

ping: Verifica a conectividade com os managed nodes.

command: Executa comandos arbitrários nos managed nodes.

apt: Gerencia pacotes no Debian/Ubuntu.

yum: Gerencia pacotes no CentOS/RHEL.

service: Gerencia serviços nos managed nodes.

user: Gerencia usuários e grupos nos managed nodes.

file: Gerencia arquivos e diretórios nos managed nodes.

template: Cria arquivos a partir de templates.



Principais plugins

winm: Para gerenciamento de hosts Windows

docker: Para interagir com containers Docker

Callback Plugins:

default: A saída padrão do Ansible.

json: Saída em formato JSON.

Benefícios



Agilidade: Automatiza tarefas repetitivas, acelerando o provisionamento e a configuração da infraestrutura.

Consistência: Garante que a infraestrutura seja configurada de forma consistente em todos os ambientes.

Confiabilidade: Reduz erros humanos e aumenta a confiabilidade da infraestrutura.

Escalabilidade: Facilita o gerenciamento de grandes ambientes.

Simplicidade: Sintaxe YAML legível por humanos, facilitando o aprendizado e a utilização.



Instalando o Ansible

```
shell

$ sudo apt update
$ sudo apt install software-properties-common
$ sudo add-apt-repository --yes --update ppa:ansible/ansible
$ sudo apt install ansible
```



Prática

Ambiente local para simulação

<https://github.com/marcosvile/iac-escola-de-verao-2025.git>



Boas práticas

Versionamento: Utilize um sistema de controle de versões (ex: Git) para gerenciar seus playbooks.

Testes: Teste seus playbooks em ambientes de testes antes de executá-los em produção.

Modularidade: Utilize roles e handlers para criar playbooks reutilizáveis.

Documentação: Documente seus playbooks para facilitar a compreensão e a manutenção.

https://docs.ansible.com/ansible/2.8/user_guide/playbooks_best_practices.html#top-level-playbooks-are-separated-by-role



Conclusão

Ansible: Ferramenta poderosa para automatizar o gerenciamento da sua infraestrutura.

Benefícios: Agilidade, consistência, confiabilidade, escalabilidade e simplicidade.

Próximos Passos: Explore a documentação oficial, participe da comunidade e pratique com o Ansible para aprimorar suas habilidades.

Extras

```
output

$ ansible-playbook -u vagrant -i inventories/hosts playbook.yaml --check

PLAY [Playbook]
*****

TASK [Gathering Facts]
*****
ok: [192.168.3.111]

TASK [nginx : Instala pacotes de dependências para o NGINX]
*****
ok: [192.168.3.111]

TASK [nginx : Importa chave de assinatura do repositório]
*****
changed: [192.168.3.111]

...

RUNNING HANDLER [nginx : Inicia o NGINX]
*****
fatal: [192.168.3.111]: FAILED! => changed=false
  msg: 'Could not find the requested service nginx: host'

PLAY RECAP
*****
192.168.3.111      : ok=10   changed=8   unreachable=0   failed=1   skipped=0
rescued=0   ignored=0
```

Extras



`ansible-doc`: Permite acessar a documentação de módulos diretamente do terminal.

`check mode`: Simula a execução de um playbook sem realizar alterações nos hosts.

Comando: `ansible-playbook --check playbook.yml`

`--tags`: Executa apenas as tarefas marcadas com tags específicas.

Exemplo: `ansible-playbook playbook.yml --tags database`

`ansible --version`: Exibe a versão do Ansible instalada.

Executa um playbook Ansible.

`ansible-playbook <nome_do_playbook.yml>`

Flags

```
-u vagrant
-i inventories/hosts
--private-key ~/.ssh/iac-dev
-vvv
```



Referências

RED HAT. (2024). O que é Infraestrutura como Código (IaC)?. Red Hat. Disponível em: <https://www.redhat.com/pt-br/topics/automation/what-is-infrastructure-as-code-iac>. Acesso em: 16 maio 2024.

ANSIBLE. (2024). Introdução ao inventário. Ansible. Disponível em: https://docs.ansible.com/ansible/latest/inventory_guide/intro_inventory.html. Acesso em: 16 maio 2024.

ANSIBLE. (2024). Melhores práticas de playbooks. Ansible. Disponível em: https://docs.ansible.com/ansible/2.8/user_guide/playbooks_best_practices.html#top-level-playbooks-are-separated-by-role. Acesso em: 16 maio 2024.

RED HAT. (2023). A evolução de Ansible: empoderando as pessoas e a tecnologia. Red Hat. Disponível em: <https://www.redhat.com/pt-br/blog/evolution-ansible-empowering-people-and-technology>. Acesso em: 16 maio 2024.

Obrigado



https://dev.to/marcos_vilela



<https://github.com/marcosvile/>



<https://www.linkedin.com/in/marcosvile/>