

Universidade Presbiteriana Mackenzie
Ciência da Computação
Laboratório de Engenharia de Software

**Projeto - Parte 1 - Infraestrutura como Código e
Gerenciamento de Configuração**

Turma: 6N11
TIA: 41829611
Nome: Gabriel Coelho da Cunha

Ferramenta escolhida: Ansible

O que é?

Ansible é uma ferramenta de automação de TI. Permite configurar sistemas, colocar software em produção e orquestrar tarefas de TI mais avançadas como colocar software em produção de forma contínua ou manter aplicações atualizadas sem tempo ocioso.

Vantagens

Os principais objetivos de Ansible são simplicidade e facilidade de uso. Com forte foco em segurança e confiança, utilizando um mínimo de partes móveis, usando OpenSSH para comunicação e ainda uma linguagem construída com manutenção e compreensão por humanos em mente, YAML.

Componentes

Modules

Ansible conecta-se às máquinas e transmite modules a elas. A maioria dos modules aceitam parâmetros que descrevem qual o estado desejável do sistema. Quando os módulos estão presentes na máquina, Ansible os executa utilizando SSH por padrão e os remove quando terminado.

Module utilities

Quando muitos modules usam o mesmo código, Ansible armazena essas funções como module utilities para minimizar duplicação e manutenção.

Plugins

Plugins aumentam a funcionalidade principal do Ansible. Enquanto modules executam no sistema destino em processos separados (normalmente, em sistemas remotos), plugins executam na máquina de controle dentro do processo Ansible.

Inventory

Por padrão, Ansible representa a máquina que administra em um arquivo que colocam todas as máquinas em grupos definidos pelo usuário.

Playbooks

Playbooks podem orquestrar várias partes da topologia de infraestrutura, com controle detalhado de quantas máquinas devem ser configurados num momento.

O caminho de busca do Ansible

O caminho de busca determina onde cada arquivo do Ansible (modules, module utilities, plugins, inventory, playbooks) serão utilizados a cada vez que um playbook executa.

Explicação do código

Vagrantfile

```
# Determina o provisionador como script de terminal
db_config.vm.provision "shell", path: "boot.sh"
# Determina o provisionador como ansible na propria maquina
db_config.vm.provision "ansible_local" do |ansible|
  # Determina arquivo de provisionamento
  ansible.playbook = "db/playbook.yml"
  # Determina grupo de hosts disponiveis
  ansible.limit = "#{db_config.vm.hostname}, localhost"
  # Determina que provisionamento deve ser feito pelo usuario root
  ansible.become = true
```

Nessa configuração se trata da máquina “db”, mas foi feita a mesma configuração para cada máquina.

boot.sh

```
#!/bin/bash
# Atualizar repositórios
sudo apt-get update -qq
# Se dependências do ansible não estiverem instaladas, instalar
dpkg -l \
| grep -qw software-properties-common \
|| sudo DEBIAN_FRONTEND=noninteractive apt-get install -y -qq software-properties-common
dpkg -l \
| grep -qw python-software-properties \
|| sudo DEBIAN_FRONTEND=noninteractive apt-get install -y -qq python-software-properties
```

db/playbook.yml

```
---
- name: Configurar Banco de Dados
  hosts: db

  tasks:
    - name: instalar pacote mysql-server
      apt:
        name: mysql-server
        state: present # se ja estiver instalado, nao tenta instalar novamente
        update_cache: yes # atualiza repositórios de pacotes antes de tentar instalar
        cache_valid_time: 3600 # atualiza repositórios somente se a ultima vez que foram atualizados faz mais de 3600 segundos
        # se ocorrer erro ao utilizar aplicacao apt, tenta novamente no maximo 100 vezes
        register: apt_action
        retries: 100
        until: apt_action is success or ('Failed to lock apt for exclusive operation' not in apt_action.msg and '/var/lib/dpkg/lock' not in apt_action.msg)

    # copia arquivo da maquina host para a guest
    - name: assegurar presença do arquivo /etc/mysql/conf.d/allow_external.cnf
      copy:
        src: allow_ext.cnf # arquivo fonte na maquina host
        dest: /etc/mysql/conf.d/allow_external.cnf # destino na maquina guest
        owner: mysql
        group: mysql
        mode: '0644'
      # notifica o handler reiniciar mysql
      notify:
        - reiniciar mysql

    - name: assegurar que servico mysql esteja rodando
      service:
        name: mysql
        state: started # servico deve estar iniciado
        enabled: yes # servico deve estar habilitado

    # roda os comandos no terminal da vm
    - name: criar loja_schema
      shell: mysql -uroot loja_schema || mysqladmin -uroot create loja_schema

    # roda os comandos no terminal da vm
    - name: remover usuario anonimo
      shell:
        cmd: >
        mysql -u' ' ||
        mysql -uroot -e
        "DELETE FROM mysql.user
        WHERE user='';
        FLUSH PRIVILEGES"

    # roda os comandos no terminal da vm
    - name: criar loja_user
      shell:
        cmd: >
        mysql -uloja -plojasecret loja_schema ||
        mysql -uroot -e
        "GRANT ALL PRIVILEGES ON
        loja_schema.* TO 'loja'@'%
        IDENTIFIED BY 'lojasecret';"

  handlers:
    - name: reiniciar mysql
      service:
        name: mysql
        state: restarted # servico deve ser reiniciado
```

web/playbook.yml

```
---
- name: Configurar Web
  hosts: web
  # variaveis para utilizar em templates
  vars:
    db_host: "192.168.33.10"
    db_schema: "loja_schema"
    db_user: "loja"
    db_password: "lojasecret"

  tasks:
    - name: instalar pacotes mysql-client e tomcat7
      apt:
        pkg:
          - mysql-client
          - tomcat7
        state: present # se ja estiver instalado, nao tenta instalar novamente
        update_cache: yes # atualiza repositórios de pacotes antes de tentar instalar
        cache_valid_time: 3600 # atualiza repositórios somente se a última vez que foram atualizados faz mais de 3600 segundos
        # se ocorrer erro ao utilizar aplicacao apt, tenta novamente no maximo 100 vezes
        register: apt_action
        retries: 100
        until: apt_action is success or ('Failed to lock apt for exclusive operation' not in apt_action.msg and '/var/lib/dpkg/lock' not in apt_action.msg)

    # copia arquivo da maquina host para a guest
    - name: assegurar presença do arquivo /var/lib/tomcat7/conf/.keystore
      copy:
        src: .keystore # arquivo fonte na maquina host
        dest: /var/lib/tomcat7/conf/.keystore # destino na maquina guest
        owner: root
        group: tomcat7
        mode: '0640'
        # notifica o handler reiniciar tomcat7
      notify:
        - reiniciar tomcat7

    # copia arquivo da maquina host para a guest
    - name: assegurar presença do arquivo /var/lib/tomcat7/conf/server.xml
      copy:
        src: server.xml # arquivo fonte na maquina host
        dest: /var/lib/tomcat7/conf/server.xml # destino na maquina guest
        owner: root
        group: tomcat7
        mode: '0644'
        # notifica o handler reiniciar tomcat7
      notify:
        - reiniciar tomcat7

    # copia arquivo da maquina host para a guest
    - name: assegurar presença do arquivo /etc/default/tomcat7
      copy:
        src: tomcat7 # arquivo fonte na maquina host
        dest: /etc/default/tomcat7 # destino na maquina guest
        owner: root
        group: root
        mode: '0644'
        # notifica o handler reiniciar tomcat7
      notify:
        - reiniciar tomcat7

    - name: assegurar que servico tomcat7 esteja rodando
      service:
        name: tomcat7
        state: started # servico deve estar iniciado
        enabled: yes # servico deve estar habilitado

    # cria diretorio
    - name: assegurar presença do diretorio /tmp/cargo/installs
      file:
        path: /tmp/cargo/installs # caminho completo do diretorio
        state: directory
        mode: '0755'

    # copia arquivo da maquina host para a guest
    - name: assegurar presença do arquivo /tmp/cargo/installs/jetty-distribution-8.1.14.v20131031.tar.gz
      copy:
        src: jetty-distribution-8.1.14.v20131031.tar.gz # arquivo fonte na maquina host
        dest: /tmp/cargo/installs/jetty-distribution-8.1.14.v20131031.tar.gz # destino na maquina guest
        owner: root
        group: root
        mode: '0644'

    # cria diretorio
    - name: assegurar presença do diretorio /home/vagrant/.m2
      file:
        path: /home/vagrant/.m2 # caminho completo do diretorio
        state: directory
        mode: '0755'

    # copia arquivo da maquina host para a guest
    - name: assegurar presença do arquivo /home/vagrant/.m2/settings.xml
      copy:
        src: settings.xml # arquivo fonte na maquina host
        dest: /home/vagrant/.m2/settings.xml # destino na maquina guest
        owner: root
        group: root
        mode: '0644'

    # copia arquivo da maquina host para a guest alterando os nomes das variaveis pelos valores
    - name: assegurar presença do arquivo /var/lib/tomcat7/conf/context.xml
      template:
        src: context.xml # arquivo fonte na maquina host
        dest: /var/lib/tomcat7/conf/context.xml # destino na maquina guest
        owner: root
        group: tomcat7
        mode: '0644'
      notify:
        - reiniciar tomcat7

    # copia arquivo da maquina host para a guest
    - name: assegurar presença do arquivo /var/lib/tomcat7/webapps/devopsnpratca.war
      copy:
        src: devopsnpratca.war # arquivo fonte na maquina host
        dest: /var/lib/tomcat7/webapps/devopsnpratca.war # destino na maquina guest
        owner: tomcat7
        group: tomcat7
        mode: '0644'
      notify:
        - reiniciar tomcat7

  handlers:
    - name: reiniciar tomcat7
      service:
        name: tomcat7
        state: restarted # servico deve ser reiniciado
```

monitor/playbook.yml

```
---
- name: Configurar Monitoramento
  hosts: monitor

  tasks:
    # roda os comandos no terminal da vm
    - name: configurar sistema para adquirir ultima versao do nagios
      shell:
        cmd: >
          echo "Package: nagios*
          Pin: release n=raring
          Pin-Priority: 990" | sudo tee /etc/apt/preferences.d/nagios
          &&
          echo "deb http://old-releases.ubuntu.com/ubuntu raring main" |
          sudo tee /etc/apt/sources.list.d/raring.list

    - name: instalar pacotes nagios3 e expect-dev
      apt:
        pkg:
          - nagios3
          - expect-dev
        state: present # se ja estiver instalado, nao tenta instalar novamente
        update_cache: yes # atualiza repositórios de pacotes antes de tentar instalar
        cache_valid_time: 3600 # atualiza repositórios somente se a ultima vez que foram atualizados faz mais de 3600 segundos
        # se ocorrer erro ao utilizar aplicacao apt, tenta novamente no maximo 100 vezes
        register: apt_action
        retries: 100
        until: apt_action is success or ('Failed to lock apt for exclusive operation' not in apt_action.msg and '/var/lib/dpkg/lock' not in apt_action.msg)

    # roda os comandos no terminal da vm
    - name: configurar postfix
      shell:
        cmd: >
          echo "set postfix/mailname monitor.lojavirtualdevops.com.br" |
          debconf-communicate
          &&
          echo "set postfix/main_mailer_type Internet Site" |
          debconf-communicate

    # copia arquivo da maquina host para a guest
    - name: assegurar presenca do arquivo nagios.exp
      copy:
        src: nagios.exp # arquivo fonte na maquina host
        dest: /etc/nagios3/nagios.exp # destino na maquina guest
        owner: root
        group: root
        mode: '0770' # permissao de leitura, escrita e execucao para dono e grupo

    - name: configurar senha do nagiosadmin
      shell: /etc/nagios3/nagios.exp # executar script gerado manualmente uma vez utilizando autoexpect do pacote expect-dev

    # copia arquivo da maquina host para a guest
    - name: assegurar presenca do arquivo /etc/nagios3/conf.d/loja_virtual.cfg
      copy:
        src: loja_virtual.cfg # arquivo fonte na maquina host
        dest: /etc/nagios3/conf.d/loja_virtual.cfg # destino na maquina guest
        owner: root
        group: root
        mode: '0644'

    # notifica o handler reiniciar nagios3
    notify:
      - reiniciar nagios3

  handlers:
    - name: reiniciar nagios3
      service:
        name: nagios3
        state: restarted # servico deve ser reiniciado
```

Passos para instalação

1. Instalar “Vagrant” e “VirtualBox” na máquina host.
2. Adicionar a imagem inicial “hashicorp/precise32” utilizando o comando “vagrant box add hashicorp/precise32” no terminal.
3. Extrair o conteúdo do arquivo “projeto.zip” como na arvore de diretórios a seguir:

```
.
├── boot.sh
├── db
│   ├── allow_ext.cnf
│   └── playbook.yml
├── monitor
│   ├── loja_virtual.cfg
│   ├── nagios.exp
│   └── playbook.yml
├── Vagrantfile
└── web
    ├── context.xml
    ├── devopsnapratica.war
    ├── jetty-distribution-8.1.14.v20131031.tar.gz
    ├── .keystore
    ├── playbook.yml
    ├── server.xml
    ├── settings.xml
    └── tomcat7

3 directories, 15 files
```


Passos para execução e utilização

1. Executar, no terminal, o comando “vagrant up” no diretório onde o arquivo “Vagrantfile” estiver.
2. Após o término do provisionamento das máquinas “db”, “web” e “monitor”, acessar no navegador os endereços:
 - a. <http://192.168.33.12:8080/devopsnapratica/> para ver a loja funcional.
 - b. <https://192.168.33.12:8443/devopsnapratica/admin/> com usuario e senha “admin” para ver estrutura da loja.
 - c. <http://192.168.33.14/nagios3/> com usuario “nagiosadmin” e senha “secret” para acessar o serviço de monitoramento.

Conclusão

Ansible, a ferramenta de gerenciamento de configuração utilizada que usa a linguagem YAML, me proporcionou a atender os requisitos de configuração neste projeto de maneira razoavelmente fácil e rápida de progredir chegando ao deploy da aplicação.

O progresso rápido na configuração é devido em grande parte a extensa documentação disponibilizada no site da Ansible.

Referências

<https://docs.ansible.com/ansible/latest/index.html>

https://docs.ansible.com/ansible/latest/dev_guide/overview_architecture.html