

DevOps Sesión 12 (2022-03-21)

Documentación relacionada

- ./4-Topic 704 Configuration Management
- ./Material Curso Ansible/DO407-AUTOMATION WITH ANSIBLE I.pdf
- ./Material Curso Ansible/Introduccion Ansible.txt
- ./Material Curso Ansible/Curso Ansible 2020.pdf

ansible.cfg

- ./4-Topic 704 Configuration Management
- ./Material Curso Ansible/Curso Ansible 2020.pdf pag 54
- ./Laboratorios Ansible-playbook y Ad-doc.txt

- 4 secciones básicas
 - defaults
 - inventory
 - sudo_user
 - forks
 - timeout
 - log_path
 - nocows
 - privilege_escalation
 - become
 - become_method
 - become_user
 - ssh_connection
 - ssh_args
 - control_path
 - scp_if_ssh
 - colors

inventario

- mantener ficheros separados (para evitar errores) por entornos (buena práctica)
- usar **-i** para indicar el inventario
- palabra reservada **all**

ayuda

- ansible-doc -l
- ansible-doc copy
- ansible-doc -s copy

cowsay enable

```
yum install cowsay -y
sudo vi /etc/ansible/ansible.cfg
# 218: nocows = 0
# 224: cow_selection = random - otros animales
```

ansible-playbook

- --step

modules

- ./4-Topic 704 Configuration Management
- ./Material Curso Ansible/Curso Ansible 2020.pdf pag 141

copy

- https://docs.ansible.com/ansible/2.9/modules/copy_module.html
- [playbooks_resueltos/modulo_file.yml](#)

```
---
- name: Ejemplos de modulos de ficheros y openssl
  hosts: clientes
  remote_user: root
  tasks:
    - name: Crear un fichero con contenido especificado
      copy: content="Esto es un ejemplos desde master"
dest="/tmp/ejemplo-master.txt" backup=yes
    - name: Propiedades Fichero
      file: path="/tmp/ejemplo-master.txt" backup=yes mode="777"
owner="vagrant"
    - name: copia archivo /etc/hostname a remoto
      copy:
        src: /etc/hostname
        dest: /tmp
        owner: root
        group: root
        mode: '0644'
```

```
backup: yes
```

```
...
```

file

- mcedit (paquete mc)
- crea directorio **/var/log/journal** para persistir los logs del journal entre arranques

modulo_file_directorio.yml

```
---
- name: Ejemplos de modulos de ficheros y openssl
  hosts: clientes
  remote_user: root
  tasks:
    - name: Creacion de un directorio
      file:
        path: "/var/log/journal"
        state: directory
        owner: root
        group: systemd-journal
        mode: 2755
      notify: reiniciar_journald
  handlers:
    - name: reiniciar_journald
      service: name=systemd-journald state=restarted
...
```

- handlers: tareas que responden a una notificación enviada por otras tareas
 - **notify** le indica al **handler** que se tiene que ejecutar
 - si varias tareas llaman al mismo handler, solo se ejecuta una vez, después de la última tarea que lo llame.
 - puede haber 2 handlers que se llamen igual, con un solo **notify** se ejecutarán los dos

delete

modulo_file_delete.yml

```
---
- name: Ejemplos de modulos de ficheros y openssl
  hosts: clientes
  remote_user: root
  tasks:
    - name: Crear un fichero con contenido especificado
      copy: content="Estos2 es un ejemplos2 desde master" dest="/tmp/ejemplo-
master.txt" backup=yes
    - name: Propiedades Fichero
      file: path="/tmp/ejemplo-master.txt" state=absent backup=yes
...
```

locale

```

ansible all -a "timedatectl set-timezone Europe/Andorra"
ansible all -a "localectl"
ansible all -a "localectl set-locale LANG=es_ES.utf8"
ansible all -a "localectl set-keymap es"
ansible all -a "localectl"

ansible all -a "timedatectl"

ansible clientes -a "yum install httpd -y"

```

fetch

- recoge/descarga ficheros de los nodos, recreando la ruta de la ubicación (dentro de un directorio de máquina)

modulo_lineinfile.yml

```

---
- name: Ejemplo de modulo fetch
  hosts: clientes
  remote_user: root
  tasks:
    - name: Utilizando fetch para copia de seguridad
      fetch:
        src=/etc/hostname
        dest=/tmp/backup
    - name: Utilizando fetch para copia de seguridad con flat solo copia el
      fichero, el destino tiene que ser un fichero
      #fetch: src=/etc/passwd dest=/tmp/backup/passwd flat=yes
      fetch: src=/etc/passwd dest=/tmp/backup/
...

```

- flat: no recrea la ruta de la ubicación del fichero y lo deja directamente en el directorio que le hemos indicado

lineinfile

se utiliza para la configuración de ficheros, podremos asegurarnos de que una línea en particular esté en un archivo, o reemplace una línea existente usando una expresión regular, antes o después de la expresión, que este presente o que no este presente.

modulo_lineinfile.yml

```

---
- name: Ejemplo de modulo fetch
  hosts: clientes
  remote_user: root
  tasks:
    - name: Utilizando modulo lineinfile para modificar la configuracion de
      selinux

```

```

    lineinfile: path=/etc/sysconfig/selinux regexp="^SELINUX="
line="SELINUX=disabled"
- name: Eliminar una linea de un fichero, en este caso del arvhivo sudoers
la linea de %wheel
    lineinfile: dest=/etc/sudoers state=absent regexp="^%wheel"
- name: Añadir una linea de configuracion al archivo de apache httpd.conf
    lineinfile:
      path: /etc/httpd/conf/httpd.conf
      regexp: '^Listen '
      insertafter: '^#Listen '
      line: 'Listen 8080'
- name: Añadir una linea despues de la expresion buscada en un archivo
    lineinfile:
      path: /etc/services
      regexp: '^# port for http'
      insertbefore: '^www.*80/tcp'
      line: '# port for http by default'

- name: touch a file, using symbolic modes to set the permissions
(equivalent to 0644)
    file:
      path: /tmp/testfile
      state: touch
      mode: "u=rw,g=r,o=r"

- name: Add a line to a file if it does not exist, without passing regexp
    lineinfile:
      path: /tmp/testfile
      line: '192.168.1.99 foo.lab.net foo'
...

```

install docker

; playbook_centos_install_docker.yaml

```

---
- name: Install docker Centos
  gather_facts: No
  hosts: clientes
  user: root

  tasks:
    - name: Install yum utils
      yum:
        name: yum-utils
        state: latest

    - name: Install device-mapper-persistent-data
      yum:
        name: device-mapper-persistent-data
        state: latest

    - name: Install lvm2

```

```
  yum:
    name: lvm2
    state: latest

- name: Add Docker repo
  get_url:
    url: https://download.docker.com/linux/centos/docker-ce.repo
    dest: /etc/yum.repos.d/docer-ce.repo
    become: yes

- name: Enable Docker Edge repo
  ini_file:
    dest: /etc/yum.repos.d/docer-ce.repo
    section: 'docker-ce-edge'
    option: enabled
    value: 0
    become: yes

- name: Enable Docker Test repo
  ini_file:
    dest: /etc/yum.repos.d/docer-ce.repo
    section: 'docker-ce-test'
    option: enabled
    value: 0
    become: yes

- name: Install Docker
  package:
    name: docker-ce
    state: latest
    become: yes

- name: Install Docker Compose
  package:
    name: docker-compose
    state: latest
    become: yes

- name: Start Docker service
  service:
    name: docker
    state: started
    enabled: yes
    become: yes

- name: Add user vagrant to docker group
  user:
    name: vagrant
    groups: docker
    append: yes
    become: yes

- name: Utilizando modulo lineinfile para modificar la configuracion de
  selinux
  lineinfile: path=/etc/sysconfig/selinux regexp="^SELINUX="
```

```
line="SELINUX=disabled"

- name: Stop and disabled firewalld
  service:
    name: firewalld
    state: stopped
    enabled: no
...
```

when_facts

- ./4-Topic 704 Configuration Management
- ./Material Curso Ansible/Curso Ansible 2020.pdf pag 68
- ./Specific Distribution ansible facts.txt
- <https://techviewleo.com/list-of-ansible-os-family-distributions-facts/>

modulo_when_facts.yaml

```
---
- hosts: clientes
  user: root

  tasks:

    - name: restart apache en debian
      service:
        name: apache2
        state: started
        enabled: yes
      when: ansible_distribution == 'Debian' or ansible_distribution == 'Ubuntu'

    - name: restart httpd en centos
      service:
        name: httpd
        state: started
        enabled: yes
      when: ansible_distribution == 'CentOS' or ansible_distribution == 'Red Hat Enterprise Linux'
...
```

```
---
- hosts: clientes
  user: root

  tasks:
    - name: Install apache para debian
      apt:
        name: {{ item }}
```

```

    state: latest
  with_items:
    - apache2
    - php
  when: ansible_distribution == 'Debian' or ansible_distribution == 'Ubuntu'

- name: Install httpd para centos
  yum:
    name: {{ item }}
    state: latest
  with_items:
    - httpd
    - httpd-devel
    - php*
    - mariadb*
  when: ansible_distribution == 'CentOS' or ansible_distribution == 'Red Hat Enterprise Linux'

- name: restart apache en debian
  service:
    name: apache2
    state: started
    enabled: yes
  when: ansible_distribution == 'Debian' or ansible_distribution == 'Ubuntu'

- name: restart httpd en centos
  service:
    name: httpd
    state: started
    enabled: yes
  when: ansible_distribution == 'CentOS' or ansible_distribution == 'Red Hat Enterprise Linux'
...

```

modulo_comandos_script.yml

loop

- https://docs.ansible.com/ansible/latest/user_guide/playbooks_loops.html
- **with_items** se eliminará este formato, usar **loop**

```

- name: paquetes
  yum:
    name: "{{ item }}"
    state: present
  loop:
    - httpd
    - php*

```

expect

- ./4-Topic 704 Configuration Management
- ./module expect ansible cambiar password a usuarios linux.txt
- https://docs.ansible.com/ansible/latest/modules/expect_module.html

- librerías/dependencias necesarias:

```
ansible cliente12 -a "yum install pexpect -y"
ansible cliente12 -a "yum -y install python-pip -y"
ansible cliente12 -a "pip install --upgrade pexpect"
```

modulo_expect.yml

```
---
- name: Ejecuta un comando y responde a las solicitudes
  hosts: cliente12
  remote_user: root
  tasks:
    # - name: Para ejecutar este modulo tenemos que instalar el paquete pexpect
    #   yum: name=pexpect state=latest

    - name: Case insensitive password string match
      expect:
        command: passwd vagrant
        responses:
          (?i)password: "vagrant"
...

```

- The question, or key, under responses is a python regex match. Case insensitive searches are indicated with a prefix of ?i.

docker

- ./4-Topic 704 Configuration Management
- ./Material Curso Ansible/Curso Ansible 2020.pdf pag 193
- https://docs.ansible.com/ansible/2.9/modules/docker_container_module.html
- Laboratorio ansible docker-network.txt linea 71

; /etc/ansible/hosts

```
...
[docker]
192.168.33.11 ansible_python_interpreter=/usr/bin/python3
...

```

```
ansible docker -a "yum -y install python2-pip -y"
ansible docker -a "pip install docker-py"

ansible docker -a "docker run -dtiP --name ansible-web nbrown/nginxhello"
```

docker-network.yml

```
---
- name: Crear red y conectar el contenedor ansible-web
  hosts: docker
  remote_user: root
  tasks:
    - name: crear red llama da ansible
      docker_network:
        name: ansible
        state: present
        connected:
          - ansible-web
...

```

```
ansible docker -a "docker network list"
ansible docker -m shell -a "docker network inspect ansible | grep -w Name"
```

volumenes

playbook_modulo_cloud_docker2.yml

```
---
- name: Playbook docker con volumenes, mapeos de puertos y mensajes con
  pushbullet
  hosts: docker
  remote_user: root
  tasks:
    #- pip: name=pushbullet.py

    - name: Creacion de un directorio /web en los servidores de docker para
      persistir un volumen
      file:
        path: "/web"
        state: directory
        owner: root
        group: root
        mode: 0755

    - name: crear un fichero con contenido específico
      copy: content="Web en docker-Ansible" dest="/web/index.html" backup=yes

# Reinicio de servicios de docker
# - name: Restart docker
#   action: service name=docker state=restarted

# Example action to start service docker, if not running
- service:

```

```
    name: docker
    state: started

- name: pull image
  docker_image:
    name: agarciáf/intranet
    # state: present

- name: create docker container
  docker_container:
    name: intranet2
    image: agarciáf/intranet
    ports:
      - "8081:80"
    volumes:
      - "/web:/var/www/html"
    state: started

...

```

kubernetes

- ./4-Topic 704 Configuration Management/Deploy kubernetes con ansible
- https://docs.ansible.com/ansible/2.9/modules/k8s_module.html

roles

- fraccionar un playbook en diferentes ficheros para facilitar su implementación

pushbullet/pushover

- notificaciones a los teléfonos a través de app

From:
<https://matebcn.eu/> - miguel angel torres egea

Permanent link:
<https://matebcn.eu/info:cursos:pue:devops2022:s12>

Last update: 21/03/2022 21:47

