

Paso 1: Crear un inventario

Crea un archivo llamado `inventory.yml` con el siguiente contenido:

all:

children:

database:

hosts:

dbserver:

ansible_host: 192.168.1.10

ansible_user: ec2-user

ansible_ssh_private_key_file: /home/ec2-user/my-key.pem

Paso 2: Ver el inventario

Ejecuta el siguiente comando para inspeccionar el inventario:

ansible-inventory --list

- ¿Cuáles son los grupos de hosts disponibles?
- ¿El inventario muestra la IP (`ansible_host`) del `dbserver`?

Paso 3: Inspeccionar variables de un host sin conectarse

Usa el siguiente comando para verificar la información almacenada en el inventario:

ansible-inventory --list | jq '._meta.hostvars["dbserver"]'

- ¿Qué información muestra sobre el `dbserver`?
- ¿Ves algo como `ansible_default_ipv4.address`? (Debería no aparecer).

Paso 4: Obtener información del host en tiempo de ejecución (setup)

ansible dbserver -m setup

- ¿Ahora aparece `ansible_default_ipv4.address`?
- ¿Qué información nueva se obtuvo respecto a `ansible-inventory`?

Paso 5: Usar debug para ver una variable específica

ansible dbserver -m debug -a "var=ansible_default_ipv4.address"

- ¿La salida es diferente a setup?
- ¿Cómo podemos usar esto dentro de un playbook?

Paso 6: Crear un pequeño playbook

Crea un archivo `playbook.yml` con el siguiente contenido:

- hosts: database

gather_facts: yes

tasks:

- name: Mostrar la dirección IP del host

debug:

msg: "La dirección IP del servidor es {{ ansible_default_ipv4.address }}"

Ejecuta:

ansible-playbook -i inventory.yml playbook.yml

- ¿Qué diferencia hay entre usar debug en un playbook y en la línea de comandos?
- ¿El resultado cambia si `gather_facts: no`?