

# Playbooks y primeros pasos

Un playbook es una especie de plantilla que nos permite ejecutar entornos complejos dentro de ansible sin necesidad de intervención humana. De esta forma podemos automatizar múltiples procesos de manera muy sencilla. Se utilizan ficheros de tipo YAML. Por eso es tan importante aprender la estructura y su sintaxis.

Aunque hay distintas secciones dentro de un **playbook** podemos decir que están compuestos de tareas que se pueden combinar en un componente denominado **play** y a su vez varios plays conforman un Playbook.

Tenemos varias acciones con las que se puede trabajar dentro de un playbook. Destacamos las siguientes:

- **Hosts:** Máquinas donde se ejecuta.
- **Vars:** Variables del play.
- **Tasks:** Lista de tareas a ejecutar en el play.
- **Handlers:** Tareas que se ejecutan solo ante algún cambio.
- **Roles:** Permiten importar recursos. Cargan recursos de forma ordenada.

## Crear nuestro primer playbook

Podemos ejecutar un playbook de prueba, más adelante explicaremos en detalle cual es el funcionamiento del mismo.

```
---
- name: Primer playbook
  hosts: srv1

  tasks:
    - name: Instalar los paquetes zip y unzip
      ansible.builtin.apt:
        name:
          - zip
          - unzip
        state: present

    - name: Crear un fichero /home/user/ejemplo con permisos 777
      ansible.builtin.file:
        path: /home/user/ejemplo
        state: touch
        owner: root
        group: root
        mode: "0777"

    - name: Añadir la cadena hola al fichero /home/user/ejemplo
      ansible.builtin.shell: |
        if test "$(tail -1 /home/user/ejemplo)" != "hola"
        then
          echo "hola" >> /home/user/ejemplo
        fi
```

```
- name: Añadir un fichero con contenido
  ansible.builtin.copy:
    dest: /home/user/copia
    content: |
      Hola que tal
      Espero que bien.
      Un saludo
      Martin
```

Si ejecutamos el playbook podemos ver que se lanza con éxito.

```
ansible@debian64:~/practicas/pr01$ ansible-playbook playbook.yaml

PLAY [Primer playbook]
*****
*****
***** TASK [Gathering Facts]
*****
*****
***** ok: [srv1]

TASK [Instalar los paquetes zip y unzip]
*****
*****
***** changed: [srv1]

TASK [Crear un fichero /home/user/ejemplo con permisos 777]
*****
*****
***** changed: [srv1]

TASK [Añadir la cadena hola al fichero /home/user/ejemplo]
*****
*****
***** changed: [srv1]

TASK [Añadir un fichero con contenido]
*****
*****
***** ok: [srv1]

PLAY RECAP
*****
*****
*****   srv1           : ok=5      changed=3      unreachable=0      failed=0
      skipped=0      rescued=0      ignored=0
```

Si volvemos a ejecutar podemos apreciar que

```
ansible@debian64:~/practicas/pr01$ ansible-playbook playbook.yaml

PLAY [Primer playbook]
*****
***** TASK [Gathering Facts]
*****
***** ok: [srv1]

TASK [Instalar los paquetes zip y unzip]
*****
***** ok: [srv1]

TASK [Crear un fichero /home/user/ejemplo con permisos 777]
*****
***** changed: [srv1]

TASK [Añadir la cadena hola al fichero /home/user/ejemplo]
*****
***** changed: [srv1]

TASK [Añadir un fichero con contenido]
*****
***** ok: [srv1]

PLAY RECAP
*****
***** : ok=5      changed=2      unreachable=0      failed=0
skipped=0    rescued=0    ignored=0
```

El estado "changed" en Ansible indica que una tarea ha realizado un cambio en el sistema remoto comparado con el estado previo. En la primera ejecución donde se reporta "changed=3" significa que tres tareas realizaron cambios efectivos porque el estado inicial no coincidía con lo declarado en el playbook (por ejemplo, se instalaron paquetes, se creó/modificó un archivo y se añadió texto al archivo).

En la segunda ejecución, el hecho de que "changed" sea 2 significa que sólo dos tareas observaron que tenían que modificar algo: probablemente la tarea de instalación de paquetes ya no hizo cambios porque los paquetes ya estaban instalados; sin embargo, la tarea de crear el archivo con ciertos permisos y la de añadir la cadena "hola" sí hicieron cambios, por ejemplo, si los permisos se modificaron nuevamente o si la condición para añadir la cadena se cumplió otra vez. La tarea que copia un archivo sin cambios aparece como "ok" porque no modificó nada.

## Playbook con diferentes PLAY

Puedo tener en un mismo playbook diferentes plays para tareas distintas.

```
---
```

```
- name: Ejecución de tareas para srv1
  hosts: srv1

  tasks:
    - name: Crear el directorio /tmp/directorio
      ansible.builtin.file:
        path: /tmp/directorio
        state: directory
        mode: "0755"

    - name: Cambiar permisos y propietario
      ansible.builtin.file:
        path: /tmp/directorio
        owner: ansible
        group: ansible
        mode: "0777"

    # - name: Borrar el directorio anterior
    #   ansible.builtin.file:
    #     path: /tmp/directorio
    #     state: "absent"

- name: Ejecución de tareas para srv2
  hosts: srv2

  tasks:
    - name: Crear el grupo smr2
      ansible.builtin.group:
        name: smr2
        state: present

    - name: Crear el usuario martin con password cifrada, home y shell
      ansible.builtin.user:
        name: martin
        group: smr2
        state: present
        home: /home/martin
        shell: /bin/bash
        password:
          "#$6$WfguojIdWi5o8V8W$jx9rm9Z2ABaSXY8AHj0tFAw01.u4MGG8iAve303PyXztGTB0oS8ak0S5Mfs3L
          wasMFVuEqXrIcumR9n2RiKTf."
        create_home: yes

    # - name: Eliminar el usuario martin y su home
    #   ansible.builtin.user:
    #     name: martin
    #     state: absent
    #     remove: yes

    # - name: Eliminar el grupo smr2
```

```
#    ansible.builtin.group:  
#        name: smr2  
#        state: absent
```

La ejecución es la siguiente.

```
ansible@debian64:~/practicas/pr01$ ansible-playbook playbook2.yaml  
  
PLAY [Ejecución de tareas para srv1]  
*****  
*****  
  
TASK [Gathering Facts]  
*****  
*****  
ok: [srv1]  
  
TASK [Crear el directorio /tmp/directorio]  
*****  
*****  
changed: [srv1]  
  
TASK [Cambiar permisos y propietario]  
*****  
*****  
changed: [srv1]  
  
PLAY [Ejecución de tareas para srv2]  
*****  
*****  
  
TASK [Gathering Facts]  
*****  
*****  
ok: [srv2]  
  
TASK [Crear el grupo smr2]  
*****  
*****  
changed: [srv2]  
  
TASK [Crear el usuario martin con password cifrada, home y shell]  
*****  
*****  
changed: [srv2]  
  
PLAY RECAP  
*****  
*****  
srv1 : ok=3      changed=2      unreachable=0      failed=0  
skipped=0    rescued=0     ignored=0
```

```
srv2 : ok=3    changed=2    unreachable=0    failed=0
skipped=0   rescued=0   ignored=0
```

## Ejercicios para repasar los conceptos anteriores

Se propone la [Tarea 2](#) para repasar los conceptos vistos hasta este punto.