# Ejercicio DevOps: Crear un rol en Ansible para instalar Apache y PHP en Ubuntu y Rocky Linux

### **6** Objetivo

Este ejercicio te guiará paso a paso en la creación de un rol de Ansible llamado apache\_srv que instalará Apache y PHP en sistemas basados en **Ubuntu** y **Rocky Linux** (o cualquier distribución con dnf). El rol será modular, reutilizable y compatible con buenas prácticas profesionales de Ansible.

El objetivo es que el rol:

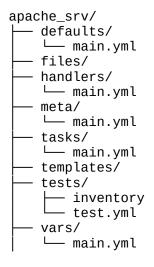
- Use módulos totalmente calificados (ansible.builtin.dnf, etc.)
- Instale paquetes específicos según la distribución
- Use variables para los nombres de paquetes
- Use handlers para reiniciar Apache si es necesario
- Sea invocado desde un playbook limpio y claro

## **X** Paso 1: Crear el rol

Ejecuta el siguiente comando para crear la estructura base del rol:

ansible-galaxy role init apache\_srv

Esto creará la siguiente estructura de carpetas:



#### 📂 Paso 2: Editar archivos del rol



Define los paquetes según la distribución:

```
# apache_srv/vars/main.yml
apache_packages:
   RedHat:
        - httpd
        - php
   Debian:
        - apache2
        - php
```

#### 🐞 tasks/main.yml

Aquí se definen las tareas principales que instalarán Apache y PHP.

```
# apache_srv/tasks/main.yml
- name: Establecer lista de paquetes según el sistema operativo
  set_fact:
    selected_packages: "{{ apache_packages['RedHat'] if
ansible_facts['os_family'] == 'RedHat' else apache_packages['Debian'] }}"
- name: Instalar paquetes en sistemas RedHat
 ansible.builtin.dnf:
    name: "{{ selected_packages }}"
    state: present
 when: ansible_facts['os_family'] == 'RedHat'
 notify: Reiniciar Apache
- name: Instalar paquetes en sistemas Debian
 ansible.builtin.apt:
   name: "{{ selected_packages }}"
    state: present
   update_cache: true
 when: ansible_facts['os_family'] == 'Debian'
 notify: Reiniciar Apache
```

#### 🙇 handlers/main.yml

Define un handler para reiniciar Apache según la distribución.

```
# apache_srv/handlers/main.yml
- name: Reiniciar Apache
   ansible.builtin.service:
    name: "{{ 'httpd' if ansible_facts['os_family'] == 'RedHat' else
'apache2' }}"
   state: restarted
   enabled: true
```

#### Paso 3: Crear el playbook principal

En la raíz del proyecto (fuera del rol), crea el archivo site.yml:

# site.yml

- name: Instalar Apache y PHP con rol apache\_srv

hosts: webservers

become: true

roles:

- apache\_srv

# Paso 4: Inventario de prueba (opcional) En clase no es necesario

Puedes crear un archivo de inventario llamado hosts.ini:

[webservers]
192.168.56.101 ansible\_user=ubuntu
192.168.56.102 ansible\_user=rocky

# Paso 5: Ejecutar el playbook

Una vez tengas todo listo, ejecuta el playbook con:

ansible-playbook -i hosts.ini site.yml

# 🚹 ¿Qué hace cada archivo?

Archivo Función

tasks/main.yml Ejecuta la instalación y notifica cambios handlers/main.yml Reinicia Apache cuando hay cambios

site.yml Playbook principal que usa el rol

hosts.ini Inventario de prueba (puede variar según tu entorno)

Define los paquetes a instalar, diferenciando por sistema

# Resultado esperado

Al ejecutar el playbook:

vars/main.yml

- Detectará si el host es RedHat/Rocky o Ubuntu
- Instalará Apache y PHP usando dnf o apt respectivamente

• Reiniciará Apache al finalizar la instalación