

# Actividad Práctica: Despliegue y Verificación de un Entorno Ansible con Docker

Cristóbal Suárez Abad

Optativa

2º ASIR

## 1. Objetivo de la Actividad

El objetivo de esta práctica es desplegar un laboratorio de automatización con **Ansible** utilizando **Docker Compose**. Se levantará una infraestructura virtualizada que consta de un **Nodo de Control** (donde se ejecuta Ansible) y tres **Nodos Gestionados** (servidores destino).

Al finalizar la actividad, deberás ser capaz de:

- Interpretar un archivo `docker-compose.yml` para entornos de laboratorio.
- Acceder al contenedor de control.
- Verificar la conectividad SSH y la configuración del inventario de Ansible.

## 2. Descripción del Escenario

A partir del archivo `LaboratorioAnsible.zip`, se generará la siguiente arquitectura:

- **Red:** `ansible-net` (Bridge) para la comunicación interna entre contenedores.
- **Nodo de Control (`ansible-control`):**
  - Basado en Ubuntu 22.04.
  - Tiene instalado Ansible, cliente SSH y herramientas de red.
  - Contiene un proyecto pre-configurado copiado en la ruta `/ansible/proyecto2`.
- **Nodos Gestionados (`node1, node2, node3`):**
  - Basados en Ubuntu 22.04 con servidor SSH habilitado.
  - Tienen un usuario pre-creado llamado `ansible` con la contraseña `password`.
  - Python instalado (requisito para que Ansible funcione).

### 3. Instrucciones de Despliegue

#### Paso 1: Preparación del Entorno

1. Descarga y descomprime el archivo [LaboratorioAnsible.zip](#) en tu máquina local.
2. Abre una terminal y navega hasta la carpeta descomprimida [LaboratorioAnsible](#).

#### Paso 2: Construcción y Arranque

Ejecuta el siguiente comando para construir las imágenes y levantar los contenedores en segundo plano:

Bash

```
docker-compose up -d --build
```

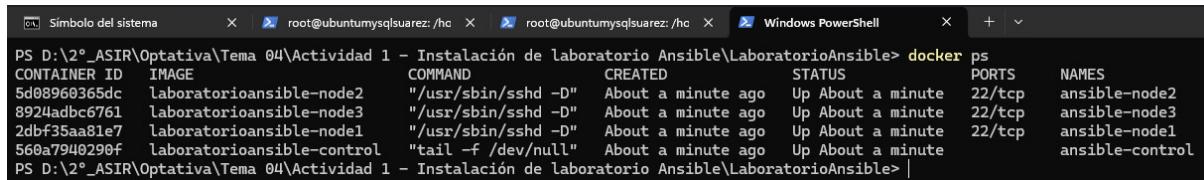


```
PS D:\2º_ASIR\Optativa\Tema 04\Actividad 1 - Instalación de laboratorio Ansible\LaboratorioAnsible> docker-compose up -d
time="2025-12-17T10:54:48+01:00" level=warning msg="D:\\2º_ASIR\\Optativa\\Tema 04\\Actividad 1 - Instalación de laboratorio , it will be ignored, please remove it to avoid potential confusion"
[+] Building 2.1s (7/15)
=> [internal] load local bake definitions
=> => reading from stdin 2.50kB
=> [node2 internal] load build definition from Dockerfile
=> => transferring dockerfile: 604B
=> [control internal] load build definition from Dockerfile
=> => transferring dockerfile: 413B
=> [node2 internal] load metadata for docker.io/library/ubuntu:22.04
=> [control internal] load .dockerignore
=> => transferring context: 2B
=> [node2 internal] load .dockerignore
=> => transferring context: 2B
=> [node3 1/4] FROM docker.io/library/ubuntu:22.04@sha256:104ae83764a5119017b8e8d6218fa0832b09df65aae7d5a6de29a85d813da2fb
=> => resolve docker.io/library/ubuntu:22.04@sha256:104ae83764a5119017b8e8d6218fa0832b09df65aae7d5a6de29a85d813da2fb
=> => sha256:7e49dc6156b0b532730614d83a65ae5e7ce61e966b0498703d333b4d03505e4f 0B / 29.54MB
=> [control internal] load build context
=> => transferring context: 613B

-> [concierto] resolving provenance for metadata file
[+] Running 10/10
✓ laboratorioansible-node3           Built
✓ laboratorioansible-control         Built
✓ laboratorioansible-node1           Built
✓ laboratorioansible-node2           Built
✓ Network laboratorioansible ansible-net   Created
✓ Volume "laboratorioansible ansible_workspace" Created
✓ Container ansible-node2          Started
✓ Container ansible-node3          Started
✓ Container ansible-node1          Started
✓ Container ansible-control        Started
PS D:\2º_ASIR\Optativa\Tema 04\Actividad 1 - Instalación de laboratorio Ansible\LaboratorioAnsible> |
```

## Paso 3: Validación de Contenedores

Asegúrate de que los cuatro contenedores estén corriendo correctamente:



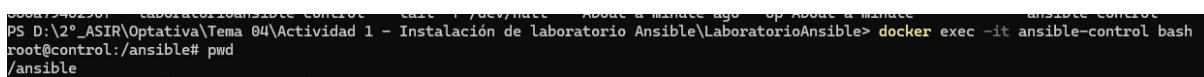
```
PS D:\2º_ASIR\Optativa\Tema 04\Actividad 1 - Instalación de laboratorio Ansible\LaboratorioAnsible> docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
5d08960365dc laboratorioansible-node2 "/usr/sbin/sshd -D" About a minute ago Up About a minute 22/tcp ansible-node2
8924adbc6761 laboratorioansible-node3 "/usr/sbin/sshd -D" About a minute ago Up About a minute 22/tcp ansible-node3
2dbf35aa81e7 laboratorioansible-node1 "/usr/sbin/sshd -D" About a minute ago Up About a minute 22/tcp ansible-node1
560a7940290f laboratorioansible-control "tail -f /dev/null" About a minute ago Up About a minute 22/tcp ansible-control
PS D:\2º_ASIR\Optativa\Tema 04\Actividad 1 - Instalación de laboratorio Ansible\LaboratorioAnsible> |
```

## 4. Ejecución y Pruebas de Ansible

### Paso 4: Acceso al Nodo de Control

Para ejecutar comandos de Ansible, necesitas entrar en la terminal del contenedor de control

docker exec -it ansible-control bash

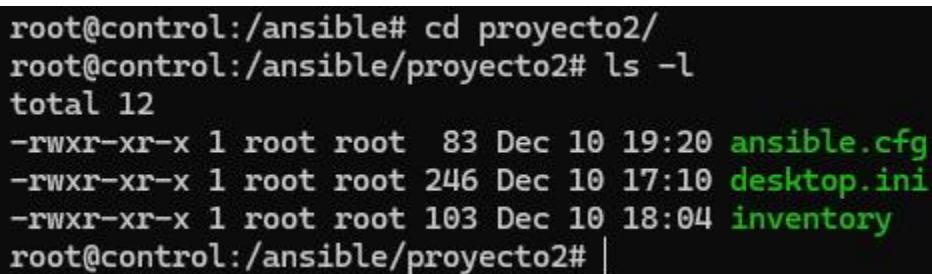


```
PS D:\2º_ASIR\Optativa\Tema 04\Actividad 1 - Instalación de laboratorio Ansible\LaboratorioAnsible> docker exec -it ansible-control bash
root@control:/ansible# pwd
/ansible
```

### Paso 5: Ubicación del Proyecto

El **Dockerfile** del nodo de control ha copiado los archivos de configuración. Navega al directorio del proyecto (observa que dentro del contenedor se ha nombrado como **proyecto2**):

```
Bash
cd /ansible/proyecto2
ls -l
```



```
root@control:/ansible# cd proyecto2/
root@control:/ansible/proyecto2# ls -l
total 12
-rwxr-xr-x 1 root root 83 Dec 10 19:20 ansible.cfg
-rwxr-xr-x 1 root root 246 Dec 10 17:10 desktop.ini
-rwxr-xr-x 1 root root 103 Dec 10 18:04 inventory
root@control:/ansible/proyecto2# |
```

Deberías ver dos archivos clave:

1. **ansible.cfg**: Archivo de configuración que define el inventario y deshabilita la comprobación de llaves de host (`host_key_checking = False`).

```
root@control:/ansible/proyecto2# cat ansible.cfg
[defaults]
inventory = inventory
host_key_checking = False
remote_user = ansibleroot@control:/ansible/proyecto2# |
```

2. **inventory**: Archivo que lista los tres nodos (`node1`, `node2`, `node3`) bajo el grupo `[webservers]` y define las credenciales de conexión (`ansible_user` y `ansible_password`).

```
remote_user = ansibleroot@control:/ansible/proyecto2# cat inventory
[webservers]
node1
node2
node3

[webservers:vars]
ansible_password=password
ansible_user=ansibleroot@control:/ansible/proyecto2# |
```

## Paso 6: Prueba de Conectividad (Ping)

Ahora realizaremos la prueba de fuego. Ejecutaremos un comando *ad-hoc* de Ansible utilizando el módulo `ping` para verificar que el nodo de control puede "hablar" con los nodos gestionados.

Ejecuta el siguiente comando:

Bash

```
ansible all -m ping
```

**Resultado Esperado:** Deberías recibir una respuesta `SUCCESS` en color verde para los tres nodos, similar a esto:

JSON

```
node1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
... (y lo mismo para node2 y node3)
```

```
root@control:/ansible/proyecto2# ansible all -m ping
node3 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
node1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
node2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
root@control:/ansible/proyecto2# |
```