

MANUAL DE EJECUCIÓN Y CONFIGURACIÓN

Gestión y Despliegue de Sistemas Operativos con Ansible

INFORMACIÓN DEL PROYECTO

Curso: Sistemas Operativos

Ciclo: 6

Fecha: Noviembre 2025

Autores:

- Boris Quispe
- Jose Zuñiga

Docente:

Alex Roberto Villegas Cervera

Repositorio:

<https://github.com/kyrafka/ansible>

INTRODUCCIÓN

Este manual documenta el proceso de configuración y despliegue automatizado de una infraestructura de red basada en IPv6 utilizando Ansible como herramienta de Infraestructura como Código (IaC).

El proyecto implementa una arquitectura cliente-servidor que incluye:

- **Servidor Ubuntu:** Servicios de red (DNS, DHCP, Firewall, Samba, FTP, NFS)
- **Ubuntu Desktop:** Sistema cliente con usuarios diferenciados y permisos específicos
- **Windows 11:** Cliente Windows gestionado remotamente mediante WinRM

La gestión centralizada mediante Ansible permite automatizar la configuración, garantizar la consistencia entre sistemas y facilitar el mantenimiento de la infraestructura.

REQUISITOS PREVIOS

1. Infraestructura de Virtualización (VMware ESXi)

1.1 Instalación de ISOs

Antes de comenzar, es necesario cargar las imágenes ISO de los sistemas operativos en el datastore de ESXi:

- Ubuntu Server 24.04 LTS (amd64)
- Ubuntu Desktop 24.04 LTS (amd64)
- Windows 11 Home (x64)

1.2 Configuración de Red Virtual

Crear los siguientes switches virtuales en ESXi:

- **vSwitch0**: Red de gestión (acceso a Internet)
- **vSwitch1**: Red interna IPv6 (2025:db8:10::/64)

Configurar port groups:

- **VM Network**: Conectado a vSwitch0 (NAT/Bridge para Internet)
- **Internal Network**: Conectado a vSwitch1 (red aislada IPv6)

1.3 Creación de Máquinas Virtuales

El proyecto incluye scripts automatizados para la creación de VMs. Consultar:

- [playbooks/vms/create-vm-ubuntu-server.yml](#)
- [playbooks/vms/create-vm-ubuntu-desktop.yml](#)
- [playbooks/vms/create-vm-windows11.yml](#)

Especificaciones mínimas recomendadas:

VM	vCPU	RAM	Disco	Interfaces de Red
Ubuntu Server	2	4 GB	40 GB	2 (Internet + Interna)
Ubuntu Desktop	2	4 GB	40 GB	1 (Internas)
Windows 11	2	4 GB	60 GB	1 (Internas)

Nota: El servidor Ubuntu requiere dos interfaces de red:

- **ens33**: Conectada a VM Network (Internet) para descargar paquetes
- **ens34**: Conectada a Internal Network (red IPv6 interna)

⚙️ CONFIGURACIÓN INICIAL

2. Preparación del Servidor Ubuntu

2.1 Instalación de Dependencias

El servidor requiere acceso a Internet para instalar Ansible y dependencias:

```
# Actualizar repositorios
sudo apt update

# Instalar Ansible
sudo apt install -y ansible

# Instalar dependencias adicionales
sudo apt install -y python3-pip git
```

```
# Verificar instalación  
ansible --version
```

2.2 Configuración de Red Temporal

Durante la instalación inicial, configurar temporalmente la interfaz de Internet (ens33) para descargar paquetes. Una vez instalado Ansible, el playbook configurará automáticamente ambas interfaces.

2.3 Clonar Repositorio del Proyecto

```
git clone https://github.com/kyrafka/ansible.git  
cd ansible
```

3. Preparación de Windows 11

3.1 Configuración de WinRM

Windows 11 requiere la habilitación de WinRM (Windows Remote Management) para permitir la gestión remota mediante Ansible.

Ejecutar en PowerShell como Administrador:

```
# Habilitar WinRM  
Enable-PSRemoting -Force -SkipNetworkProfileCheck  
  
# Configurar autenticación básica  
Set-Item -Path WSMan:\localhost\Service\Auth\Basic -Value $true  
Set-Item -Path WSMan:\localhost\Service\AllowUnencrypted -Value $true  
  
# Configurar firewall  
New-NetFirewallRule -Name "WinRM-HTTP" -DisplayName "WinRM HTTP" `  
    -Enabled True -Direction Inbound -Protocol TCP -LocalPort 5985  
  
# Configurar red como privada  
Get-NetConnectionProfile | Set-NetConnectionProfile -NetworkCategory Private  
  
# Reiniciar servicio  
Restart-Service WinRM  
  
# Verificar configuración  
winrm get winrm/config
```

Scripts disponibles:

- scripts/windows/setup-winrm-remote.ps1 (PowerShell)

- `scripts/windows/setup-winrm-simple.bat` (Batch)

3.2 Verificar Conectividad IPv6

Asegurar que Windows 11 obtiene una dirección IPv6 mediante DHCP del servidor:

```
ipconfig | findstr "IPv6"
```

4. Preparación de Ubuntu Desktop

4.1 Configuración de Red

Ubuntu Desktop debe configurarse para obtener dirección IPv6 automáticamente mediante DHCP del servidor.

Verificar conectividad:

```
ip -6 addr show  
ping6 -c 4 2025:db8:10::2
```

4.2 Instalación de Cliente Samba (Opcional)

Para pruebas de conectividad a recursos compartidos:

```
sudo apt install -y cifs-utils smbclient
```

📋 SISTEMAS A CONFIGURAR

Una vez completados los requisitos previos, proceder con la configuración automatizada de:

1. **Servidor Ubuntu** - Servicios de red (DNS, DHCP, Firewall, Samba, FTP, NFS)
2. **Ubuntu Desktop** - Cliente Linux con usuarios y permisos diferenciados
3. **Windows 11** - Cliente Windows con gestión remota mediante Ansible

1 CONFIGURAR SERVIDOR UBUNTU

1.1 Verificar Requisitos Previos

Antes de ejecutar el playbook, verificar:

```
# Verificar que Ansible está instalado  
ansible --version  
  
# Verificar conectividad a Internet (para descargar paquetes)  
ping -c 2 8.8.8.8  
  
# Verificar que el repositorio está clonado  
ls -la ~/ansible
```

1.2 Ejecutar Playbook Principal

```
cd ~/ansible  
ansible-playbook site.yml --connection=local --become --ask-become-pass
```

Nota: El playbook solicitará la contraseña de sudo. Este proceso puede tardar entre 10-15 minutos dependiendo de la velocidad de Internet.

Servicios configurados automáticamente:

- **DNS (BIND9)**: Resolución de nombres para la red interna (gamecenter.lan)
- **DHCP IPv6**: Asignación automática de direcciones IPv6 a clientes
- **Firewall (UFW)**: Reglas de seguridad para servicios permitidos
- **fail2ban**: Protección contra intentos de acceso no autorizado
- **NFS**: Servidor de archivos para sistemas Linux
- **Samba**: Servidor de archivos compatible con Windows
- **FTP (vsftpd)**: Servidor FTP para transferencia de archivos

1.3 Verificar Servicios Configurados

```
bash scripts/diagnostics/show-server-config.sh
```

Verificación manual de servicios críticos:

```
# DNS  
sudo systemctl status bind9  
dig @localhost gamecenter.lan AAAA  
  
# DHCP  
sudo systemctl status isc-dhcp-server6  
  
# Firewall  
sudo ufw status verbose  
  
# Samba  
sudo systemctl status smbd
```

```
sudo smbstatus  
  
# FTP  
sudo systemctl status vsftpd
```

2 CONFIGURAR UBUNTU DESKTOP

EJECUTAR EN UBUNTU DESKTOP:

A. Configurar Usuarios y Temas

```
bash scripts/client/setup-users-and-themes.sh
```

Crea:

- administrador (sudo completo)
- auditor (sudo limitado)
- gamer01 (sin sudo)

B. Verificar Usuarios

```
bash scripts/client/mostrar-usuarios-grupos.sh
```

C. Verificar Particiones

```
bash scripts/client/mostrar-particiones.sh
```

D. Probar Samba y FTP

```
bash scripts/client/test-samba-ftp.sh
```

3 CONFIGURAR WINDOWS 11

EJECUTAR EN EL SERVIDOR:

A. Verificar Requisitos Previos

Antes de configurar Windows, asegurar que:

1. WinRM está habilitado en Windows (ver sección 3.1 de Requisitos Previos)

2. Windows tiene conectividad IPv6 con el servidor
3. El inventario de Ansible tiene la IP correcta de Windows

```
# Verificar inventario  
cat inventory/windows.ini  
  
# Verificar conectividad  
ping6 -c 2 2025:db8:10::4f # Reemplazar con la IP de Windows
```

B. Probar Conexión WinRM

```
bash scripts/server/test-windows-connection.sh
```

Nota: Si la conexión falla, verificar:

- WinRM está activo en Windows: `Get-Service WinRM`
- Puerto 5985 está abierto: `Get-NetFirewallRule -Name "WinRM-HTTP"`
- La IP en `inventory/windows.ini` es correcta

C. Configurar Windows Remotamente

```
bash scripts/server/configure-windows.sh
```

O ejecutar playbook directamente:

```
ansible-playbook -i inventory/windows.ini playbooks/configure-windows.yml
```

Crea:

- Usuario `dev`
- Usuario `cliente`
- Carpetas `C:\Compartido` y `C:\Dev`
- Firewall configurado

D. Verificar Configuración de Windows

```
bash scripts/server/mostrar-windows-config.sh
```

Verificación manual en Windows (PowerShell):

```
# Ver usuarios creados
Get-LocalUser | Format-Table Name, Enabled

# Ver carpetas creadas
Get-ChildItem C:\ | Where-Object {$_.Name -match 'Compartido|Dev' }

# Ver reglas de firewall
Get-NetFirewallRule | Where-Object {$_.DisplayName -match 'WinRM|ICMPv6' }
```

TABLA DE SCRIPTS

Script	Sistema	Ejecutar en	Función
site.yml	Servidor	Servidor	Configurar servicios
scripts/client/setup-users-and-themes.sh	Ubuntu Desktop	Ubuntu Desktop	Crear usuarios
scripts/client/mostrar-usuarios-grupos.sh	Ubuntu Desktop	Ubuntu Desktop	Ver usuarios
scripts/client/mostrar-particiones.sh	Ubuntu Desktop	Ubuntu Desktop	Ver particiones
scripts/client/test-samba-ftp.sh	Ubuntu Desktop	Ubuntu Desktop	Probar Samba/FTP
scripts/server/configure-windows.sh	Windows	Servidor	Configurar Windows
scripts/server/mostrar-windows-config.sh	Windows	Servidor	Ver config Windows

ORDEN DE EJECUCIÓN

1. **Servidor:** ansible-playbook site.yml --connection=local --become --ask-become-pass
2. **Servidor:** bash scripts/diagnostics/show-server-config.sh
3. **Ubuntu Desktop:** bash scripts/client/setup-users-and-themes.sh
4. **Ubuntu Desktop:** bash scripts/client/test-samba-ftp.sh
5. **Servidor:** bash scripts/server/configure-windows.sh
6. **Servidor:** bash scripts/server/mostrar-windows-config.sh

REFERENCIAS

- Documentación oficial de Ansible: <https://docs.ansible.com/>
- Repositorio del proyecto: <https://github.com/kyrafka/ansible>

- BIND9 Documentation: <https://bind9.readthedocs.io/>
 - ISC DHCP Server: <https://www.isc.org/dhcp/>
-

NOTAS FINALES

Este manual documenta la configuración completa de una infraestructura de red IPv6 gestionada mediante Ansible. Todos los componentes están versionados y pueden ser replicados en diferentes entornos.

Para soporte o consultas, revisar la documentación en el repositorio del proyecto.

Proyecto: Gestión y Despliegue de Sistemas Operativos

Curso: Sistemas Operativos - Ciclo 6

Fecha: Noviembre 2025

Autores: Boris Quispe, Jose Zuñiga

Docente: Alex Roberto Villegas Cervera