

Ansible - Gestión y Despliegue GameCenter

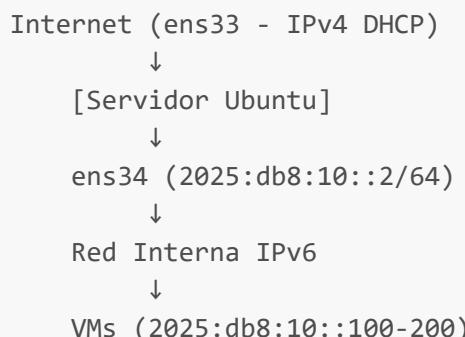
Proyecto de Ansible para configurar un servidor Ubuntu con servicios de red IPv6 (DNS, DHCP, Firewall, NFS) para un centro de juegos.

🎮 Descripción

Este proyecto automatiza la configuración completa de un servidor Ubuntu que actúa como:

- **Gateway IPv6** con NAT66
- **Servidor DNS** (BIND9)
- **Servidor DHCPv6** (ISC DHCP)
- **Firewall** (UFW + fail2ban)
- **Servidor NFS** (almacenamiento compartido)

Ἑ Arquitectura



Red: 2025:db8:10::/64

Dominio: gamecenter.local

Servidor: 2025:db8:10::2

VMs (DHCP): 2025:db8:10::100 a 2025:db8:10::200

📋 Requisitos

- Ubuntu Server 22.04 o superior
- Python 3.8+
- Ansible 2.15+
- Dos interfaces de red (ens33 para internet, ens34 para red interna)

🚀 Instalación rápida

1. Clonar el repositorio

```
git clone <https://github.com/kyrafka/ansible.git>
cd ansible-gestion-despliegue
```

2. Activar entorno virtual de Ansible

```
source activate-ansible.sh
```

3. Configurar contraseña

```
# Crear archivo de contraseña
echo "ubuntu123" > .vault_pass
chmod 600 .vault_pass
```

4. Ejecutar playbook completo

```
./run.sh
```

O ejecutar roles individuales:

```
./run.sh common      # Configuración base
./run.sh network    # Red IPv6 y NAT66
./run.sh dns         # Servidor DNS
./run.sh dhcp        # Servidor DHCPv6
./run.sh firewall   # Firewall y seguridad
./run.sh storage     # NFS y almacenamiento
```

(Roles)

Roles disponibles

Rol	Descripción	README
common	Configuración base del sistema	Ver
network	Red IPv6, NAT66, interfaces	Ver
dns_bind	Servidor DNS (BIND9)	Ver
dhcpv6	Servidor DHCPv6	Ver
firewall	UFW + fail2ban	Ver
storage	NFS y monitoreo de discos	Ver

🔧 Configuración

Variables principales

Edita `group_vars/all.yml`:

```
network_config:  
  ipv6_network: "2025:db8:10::/64"  
  ipv6_gateway: "2025:db8:10::1"  
  server_ipv6: "2025:db8:10::2"  
  domain_name: "gamecenter.local"  
  dhcp_range_start: "2025:db8:10::100"  
  dhcp_range_end: "2025:db8:10::200"
```

Variables sensibles

Edita `group_vars/all.vault.yml`:

```
vault_sudo_password: "ubuntu123"  
vault_ubuntu_password: "ubuntu123"
```

Para encriptar:

```
./encrypt-vault.sh
```

📘 Uso

Ejecutar todo el playbook

```
ansible-playbook site.yml --connection=local --become --vault-password-file  
.vault_pass -e "ansible_become_password={{ vault_sudo_password }}"
```

O simplemente:

```
./run.sh
```

Ejecutar un rol específico

```
./run.sh [rol]
```

Ejemplos:

```
./run.sh firewall      # Solo firewall  
./run.sh dns           # Solo DNS  
./run.sh network       # Solo red
```

Verificar servicios

```
# DNS  
dig @localhost server.gamecenter.local AAAA  
  
# DHCP  
systemctl status isc-dhcp-server6  
  
# Firewall  
sudo ufw status verbose  
  
# NFS  
showmount -e localhost  
  
# Red  
ip -6 addr show  
ip6tables -t nat -L -v
```

🛠 Scripts útiles

Script	Descripción
run.sh	Ejecutar playbook completo o rol específico
activate-ansible.sh	Activar entorno virtual de Ansible
encrypt-vault.sh	Encriptar variables sensibles
/usr/local/bin/logs	Monitorear logs del sistema
/usr/local/bin/fw-monitor	Monitorear firewall

📊 Monitoreo

Logs centralizados

```
# Ver todos los logs  
/usr/local/bin/logs  
  
# Logs específicos  
tail -f /var/log/dns/queries.log  
tail -f /var/log/dhcp/dhcpd6.log  
tail -f /var/log/security/fail2ban.log
```

Estado de servicios

```
systemctl status named          # DNS
systemctl status isc-dhcp-server6 # DHCP
systemctl status ufw             # Firewall
systemctl status fail2ban         # Fail2ban
systemctl status nfs-server       # NFS
```

🔒 Seguridad

Firewall

- **Política:** Denegar todo por defecto
- **SSH:** Rate limiting (máx 5 intentos en 10 min)
- **Fail2ban:** Baneo automático de IPs maliciosas
- **Puertos abiertos:** 22 (SSH), 53 (DNS), 546-547 (DHCP), 21000-21010 (FTP)

Contraséñas

- Almacenadas en `group_vars/all.vault.yml`
- Encriptadas con Ansible Vault
- Contraseña del vault en `.vault_pass` (no subir a git)

📁 Estructura del proyecto

```
.
├── roles/                  # Roles de Ansible
│   ├── common/              # Configuración base
│   ├── network/              # Red IPv6
│   ├── dns_bind/             # DNS
│   ├── dhcipv6/              # DHCP
│   ├── firewall/             # Firewall
│   └── storage/              # NFS
├── group_vars/              # Variables globales
│   ├── all.yml               # Variables públicas
│   └── all.vault.yml         # Variables sensibles
├── inventory/              # Inventario de hosts
│   └── hosts.ini
├── site.yml                 # Playbook principal
└── run.sh                   # Script de ejecución
└── README.md                # Este archivo
```

🔧 Troubleshooting

El DHCP no arranca

```
# Ver errores
journalctl -u isc-dhcp-server6 -n 50
```

```
# Verificar sintaxis  
dhcpd -6 -t -cf /etc/dhcp/dhcpd6.conf  
  
# Verificar interfaz  
ip -6 addr show ens34
```

El DNS no resuelve

```
# Ver logs  
tail -f /var/log/dns/queries.log  
  
# Probar resolución  
dig @localhost server.gamecenter.local AAAA  
  
# Verificar zonas  
named-checkzone gamecenter.local /etc/bind/zones/db.gamecenter.local
```

El firewall bloquea todo

```
# Ver reglas  
sudo ufw status numbered  
  
# Permitir IP temporalmente  
sudo ufw allow from 2025:db8:10::10  
  
# Deshabilitar temporalmente  
sudo ufw disable
```

✍ Documentación adicional

- [ARQUITECTURA.md](#) - Arquitectura detallada del proyecto
- [CORRECCIONES-APLICADAS.md](#) - Historial de correcciones
- [CHANGELOG.md](#) - Registro de cambios

🤝 Contribuir

1. Fork el proyecto
2. Crea una rama (`git checkout -b feature/nueva-funcionalidad`)
3. Commit tus cambios (`git commit -am 'Agregar nueva funcionalidad'`)
4. Push a la rama (`git push origin feature/nueva-funcionalidad`)
5. Abre un Pull Request

📄 Licencia

Este proyecto está bajo la licencia Apache 2.0. Ver [LICENSE.txt](#) para más detalles.

◆ Autor

Proyecto desarrollado para la gestión automatizada de un centro de juegos con servicios de red IPv6.