

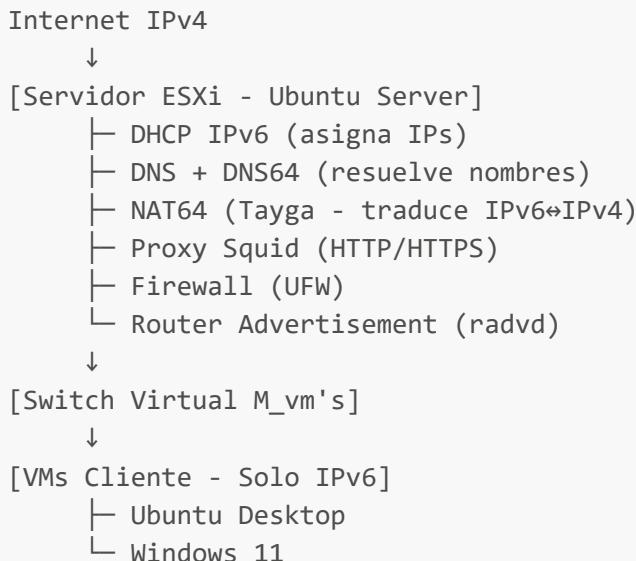
# Documentación Completa del Proyecto

## Objetivo del Proyecto

Crear una infraestructura de red **IPv6 únicamente** con:

- Servidor central que proporciona todos los servicios
- VMs cliente (Ubuntu y Windows) que solo usan IPv6
- Acceso a internet IPv4 mediante DNS64 + NAT64
- Gestión centralizada con Ansible

## Arquitectura



## Estructura del Proyecto

### Directorios Principales

```

ansible-gestion-despliegue/
  ├── playbooks/          # Playbooks de Ansible
  │   ├── infrastructure/ # Configuración del servidor
  │   ├── vms/             # Creación de VMs
  │   └── gaming/          # Instalación de juegos
  ├── roles/              # Roles de Ansible
  │   ├── dns_bind/        # Servidor DNS
  │   ├── dhcipv6/         # Servidor DHCP IPv6
  │   ├── nat64_tayga/     # NAT64 (Tayga)
  │   ├── firewall/        # Firewall (UFW)
  │   └── network/         # Configuración de red
  └── scripts/            # Scripts auxiliares

```

```

    └── virtualbox/      # Creación de VMs VirtualBox
        ├── run/         # Scripts de validación
        └── nat64/        # Scripts de NAT64
    └── group_vars/     # Variables de Ansible
        ├── all.yml       # Variables generales
        └── all.vault.yml # Contraseñas (encriptado)
    └── inventory.ini   # Inventario de hosts

```

## 🔧 Componentes del Servidor

### 1. DHCP IPv6 (ISC DHCP Server)

**Ubicación:** `roles/dhcpv6/`

**Qué hace:**

- Asigna direcciones IPv6 automáticamente a las VMs
- Rango: `2025:db8:10::100` a `2025:db8:10::200`
- Envía información de DNS y dominio
- Actualiza DNS dinámicamente (DDNS)

**Archivos clave:**

- `templates/dhcpd6.conf.j2` - Configuración principal
- `templates/isc-dhcp-server.j2` - Interfaces donde escucha
- `files/dhcp-ddns-update.sh` - Script de actualización DNS

**Configuración:**

```

# group_vars/all.yml
dhcp_range_start: "2025:db8:10::100"
dhcp_range_end: "2025:db8:10::200"

```

### 2. DNS + DNS64 (BIND9)

**Ubicación:** `roles/dns_bind/`

**Qué hace:**

- Resuelve nombres de dominio (`gamecenter.lan`)
- DNS64: Traduce direcciones IPv4 a IPv6 (prefijo `64:ff9b::/96`)
- Permite que clientes IPv6 accedan a sitios IPv4
- Actualización dinámica desde DHCP

**Archivos clave:**

- `templates/named.conf.j2` - Configuración principal
- `templates/named.conf.options.j2` - Opciones de DNS64

- `templates/named.conf.local.j2` - Zonas locales
- `templates/db.domain.j2` - Zona directa
- `templates/db.ipv6.reverse.j2` - Zona inversa

#### Ejemplo DNS64:

```
Cliente pregunta: google.com AAAA  
DNS responde: 64:ff9b::8efa:b92e (traducido de 142.250.185.46)
```

### 3. NAT64 (Tayga)

**Ubicación:** `roles/nat64_tayga/`

**Qué hace:**

- Traduce paquetes IPv6 a IPv4 y viceversa
- Permite que clientes IPv6 accedan a internet IPv4
- Crea interfaz virtual `nat64`
- Pool IPv4: 192.168.255.0/24

**Archivos clave:**

- `templates/tayga.conf.j2` - Configuración de Tayga
- `tasks/main.yml` - Instalación y configuración

**Flujo de tráfico:**

```
Cliente IPv6 → 64:ff9b::8.8.8.8  
↓  
Tayga traduce → 8.8.8.8 (IPv4)  
↓  
Internet IPv4
```

### 4. Proxy Squid

**Ubicación:** Scripts manuales

**Qué hace:**

- Proxy HTTP/HTTPS en puerto 3128
- Cachea contenido web
- Mejora velocidad de navegación
- Filtra contenido (opcional)

**Configuración en clientes:**

```
HTTP Proxy: 2025:db8:10::2
Port: 3128
```

## 5. Firewall (UFW)

**Ubicación:** `roles/firewall/`

**Qué hace:**

- Controla acceso a servicios
- Permite solo puertos necesarios
- Bloquea SSH para usuarios no admin
- Protege el servidor

**Puertos abiertos:**

- 22 (SSH) - Solo para administrador
- 53 (DNS) - TCP/UDP
- 547 (DHCP IPv6) - UDP
- 3128 (Proxy Squid) - TCP

## 6. Router Advertisement (radvd)

**Ubicación:** `roles/network/`

**Qué hace:**

- Anuncia la red IPv6 a los clientes
- Envía información de gateway
- Configura autoconfiguración SLAAC (deshabilitada)
- Indica que deben usar DHCP

## Configuración de VMs Cliente

**Ubuntu Desktop**

**Scripts:**

- `scripts/configure-vm-ipv6-only.sh` - Configuración completa
- `scripts/create-users.sh` - Crea usuarios (auditor, gamer01)
- `scripts/setup-gamer-theme.sh` - Tema gaming
- `scripts/setup-auditor-theme.sh` - Tema profesional

**Usuarios:**

1. **administrador** - Admin completo, SSH permitido
2. **auditor** - Solo lectura, acceso a logs

### 3. **gamer01** - Usuario estándar para juegos

#### Configuración de red:

- DHCP IPv6 habilitado
  - DNS: 2025:db8:10::2
  - Ruta NAT64: 64:ff9b::/96
  - Proxy: http://[2025:db8:10::2]:3128
- 

## Windows 11

#### Scripts:

- `scripts/virtualbox/crear-vm-windows11-vbox.ps1` - Crea VM
- `scripts/create-windows-users.ps1` - Crea usuarios

#### Usuarios:

1. **Tu usuario** - Administrador
2. **auditor** - Usuario estándar
3. **gamer01** - Usuario estándar

#### Configuración de red:

- DHCP IPv6 habilitado
  - DNS: 2025:db8:10::2
  - Proxy manual en navegador
- 

## ✍ Archivos de Configuración

### inventory.ini

Define los hosts donde Ansible ejecuta tareas.

```
[servers]
localhost ansible_connection=local

[ubuntu_desktops]
# VMs Ubuntu Desktop

[windows_desktops]
# VMs Windows
```

### group\_vars/all.yml

Variables globales del proyecto.

#### Secciones principales:

- **network\_config:** Red IPv6, gateway, DHCP
  - **dns\_config:** Dominio, servidores DNS
  - **users:** Usuarios del sistema
  - **dhcp6\_config:** Tiempos de lease DHCP
- 

## group\_vars/all.vault.yml

Contrasen as encriptadas con Ansible Vault.

### Contiene:

- Contrasen as de usuarios
- Claves DDNS
- Credenciales ESXi
- Contrasen as de servicios

### Desencriptar:

```
ansible-vault edit group_vars/all.vault.yml
```

## 🚀 Playbooks Principales

### playbooks/infrastructure/setup-complete-infrastructure.yml

Configura TODO el servidor de una vez.

### Incluye:

1. Configuraci n de red
2. DHCP IPv6
3. DNS + DNS64
4. Firewall
5. NAT64 (opcional)

### Ejecutar:

```
ansible-playbook -i inventory.ini playbooks/infrastructure/setup-complete-infrastructure.yml
```

### playbooks/infrastructure/playbook-dns.yml

Solo configura DNS.

### Tareas:

- Instala BIND9
  - Configura zonas DNS
  - Habilita DNS64
  - Configura DDNS
- 

## playbooks/infrastructure/playbook-dhcp.yml

Solo configura DHCP.

### Tareas:

- Instala ISC DHCP Server
  - Configura rango de IPs
  - Configura DDNS
  - Habilita servicio
- 

## playbooks/enable-nat64.yml

Configura NAT64 (Tayga).

### Tareas:

- Instala Tayga
  - Crea interfaz nat64
  - Configura rutas
  - Habilita forwarding
  - Configura iptables
- 

## Scripts Útiles

### scripts/diagnose-nat64.sh

Diagnostica problemas de NAT64.

### Verifica:

- Tayga corriendo
  - Interfaz nat64 existe
  - Rutas configuradas
  - Forwarding habilitado
  - Reglas de firewall
- 

### scripts/fix-tayga.sh

Repara Tayga si falla.

### Hace:

- Detiene Tayga
  - Limpia interfaz
  - Reinicia servicio
  - Configura rutas
- 

## **scripts/configure-vm-ipv6-only.sh**

Configura VM Ubuntu para IPv6.

### **Hace:**

- Cambia máscara /128 a /64
  - Agrega ruta NAT64
  - Crea servicio systemd
  - Verifica conectividad
- 

## Seguridad

### **SSH**

- Solo usuario **administrador** puede conectarse
- Otros usuarios bloqueados
- Configurado en **/etc/ssh/sshd\_config**

### **Firewall**

- UFW habilitado
- Solo puertos necesarios abiertos
- Reglas específicas por servicio

### **Usuarios**

- Contraseñas en vault encriptado
  - Permisos mínimos necesarios
  - Separación de roles (admin/auditor/cliente)
- 

## Pruebas y Validación

### **Verificar DHCP**

```
sudo systemctl status isc-dhcp-server6
sudo cat /var/lib/dhcp/dhcpd6.leases
```

### **Verificar DNS**

```
dig @localhost gamecenter.lan AAAA  
dig @localhost google.com AAAA # Debe devolver 64:ff9b::....
```

## Verificar NAT64

```
ping6 -c 3 64:ff9b::8.8.8.8
```

## Verificar Proxy

```
curl -x http://[2025:db8:10::2]:3128 http://google.com
```

## Monitoreo (Futuro)

### Prometheus

- Recolecta métricas del servidor
- CPU, RAM, disco, red
- Alertas automáticas

### Grafana

- Dashboards visuales
- Gráficas en tiempo real
- Histórico de métricas

## Backup (Futuro)

### Automático

- Backup diario de configuraciones
- Backup semanal de datos
- Retención de 30 días

### Manual

```
# Backup de configuraciones  
tar -czf backup-config.tar.gz /etc/bind /etc/dhcp /etc/squid  
  
# Backup de datos  
rsync -av /srv/games /backup/
```

## 🎮 Samba (Futuro)

### Carpetas Compartidas

- `/srv/games` - Juegos compartidos
- `/srv/documentos` - Documentos
- `/srv/backups` - Backups de VMs

### Acceso

- Windows: `\\"2025:db8:10::2\games`
  - Linux: `smb://[2025:db8:10::2]/games`
- 

## 🔧 Troubleshooting

### VM no recibe IP

1. Verificar DHCP corriendo
2. Verificar interfaz correcta
3. Renovar DHCP: `sudo dhclient -6 -r && sudo dhclient -6`

### No hay internet

1. Verificar DNS64: `dig @2025:db8:10::2 google.com AAAA`
2. Verificar NAT64: `ping6 64:ff9b::8.8.8.8`
3. Verificar forwarding: `cat /proc/sys/net/ipv4/ip_forward`

### Proxy no funciona

1. Verificar Squid: `sudo systemctl status squid`
  2. Verificar puerto: `sudo netstat -tlnp | grep 3128`
  3. Configurar en navegador manualmente
- 

## 📋 Referencias

- **BIND9:** <https://www.isc.org/bind/>
  - **ISC DHCP:** <https://www.isc.org/dhcp/>
  - **Tayga:** <http://www.litech.org/tayga/>
  - **Squid:** <http://www.squid-cache.org/>
  - **Ansible:** <https://docs.ansible.com/>
- 

## ✓ Checklist de Implementación

- Servidor ESXi configurado
- Red IPv6 funcionando
- DHCP asignando IPs
- DNS resolviendo nombres

- DNS64 traduciendo IPv4
  - NAT64 funcionando
  - Proxy Squid activo
  - Firewall configurado
  - VM Ubuntu Desktop con internet
  - VM Windows 11 con internet
  - Samba configurado
  - Backup automático
  - Monitoreo activo
- 

**Última actualización:** 2025-11-18 **Versión:** 1.0 **Autor:** Proyecto Ansible Gaming Center