

Informe Técnico de Propuesta de Sistema Operativo para un Laboratorio Académico y un Game Center

FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

Información del Proyecto

Curso: Sistemas Operativos

Profesor: Villegas Alex

Año: 2025

Ubicación: Lima, Perú

Autores

- **Quispe Chumbes Boris Santiago**
 - **Zúñiga Medina José Darío**
-

Índice

1. [Descripción del Proyecto](#)
 2. [Topología de Red](#)
 3. [Arquitectura del Sistema](#)
 4. [Servicios Implementados](#)
 5. [Gestión de Procesos y Servicios](#)
 6. [Administración de Usuarios y Permisos](#)
 7. [Automatización de Tareas](#)
 8. [Seguridad y Políticas](#)
 9. [Mantenimiento y Monitoreo](#)
 10. [Guía de Uso](#)
-

Descripción del Proyecto

Este proyecto implementa una infraestructura completa de red IPv6 para un laboratorio académico y game center, utilizando tecnologías de virtualización y automatización con Ansible.

Objetivos

- ☒ Implementar una red IPv6 pura (`2025:db8:10::/64`)
 - ☒ Configurar servicios de red esenciales (DNS, DHCP, Web)
 - ☒ Automatizar el despliegue con Ansible
 - ☒ Gestionar múltiples sistemas operativos (Linux, Windows, macOS)
 - ☒ Implementar seguridad con firewall y fail2ban
-

🌐 Topología de Red

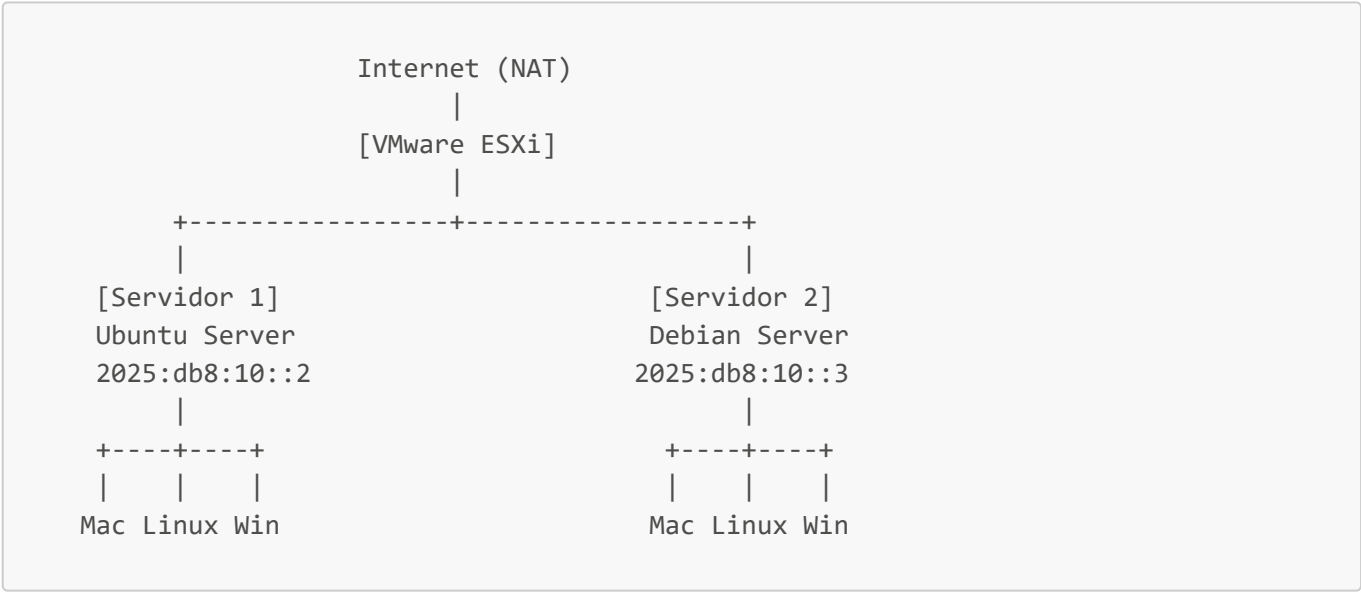
Servidor Gaming 1

- **Servidor Ubuntu** (Principal)
 - IP: 2025:db8:10::2
 - Servicios: DNS (BIND9), DHCPv6, Nginx, Firewall
- **Estaciones:**
 - macOS
 - Linux
 - Windows 11

Servidor Gaming 2

- **Servidor Debian**
 - Servicios: Secundario/Backup
- **Estaciones:**
 - macOS
 - Linux
 - Windows 11

Diagrama de Red



🏗️ Arquitectura del Sistema

Tecnologías Utilizadas

Componente	Tecnología	Versión
Virtualización	VMware ESXi	7.0+
Automatización	Ansible	2.15+
Servidor DNS	BIND9	9.18+

Componente	Tecnología	Versión
Servidor DHCP	isc-dhcp-server	4.4+
Servidor Web	Nginx	1.24+
Firewall	UFW + fail2ban	-
Sistema Base	Ubuntu Server	24.04 LTS

Estructura del Proyecto

```
ansible-gestion-despliegue/
├── roles/
│   ├── common/           # Configuración base
│   ├── network/          # Red IPv6 y radvd
│   ├── dns_bind/         # Servidor DNS
│   ├── dhcpv6/           # Servidor DHCPv6
│   ├── http_web/         # Servidor web Nginx
│   ├── firewall/         # UFW y fail2ban
│   └── storage/          # Gestión de almacenamiento
├── playbooks/            # Playbooks de Ansible
├── scripts/
│   ├── run/              # Scripts de ejecución
│   ├── diagnostics/      # Scripts de diagnóstico
│   └── setup/            # Scripts de instalación
├── inventory/            # Inventario de hosts
└── group_vars/           # Variables de configuración
```

🔑 Servicios Implementados

1. DNS (BIND9)

- **Dominio:** gamecenter.local
- **Zona directa:** Resolución de nombres a IPs
- **Zona inversa:** Resolución de IPs a nombres
- **Registros configurados:**
 - gamecenter.local → 2025:db8:10::2
 - servidor.gamecenter.local → 2025:db8:10::2
 - www.gamecenter.local → CNAME a servidor
 - web.gamecenter.local → CNAME a servidor

2. DHCPv6

- **Rango de IPs:** 2025:db8:10::10 - 2025:db8:10::FFFF
- **Asignación dinámica** con DUID
- **Configuración automática** de DNS y dominio
- **SLAAC desactivado** para control centralizado

3. Servidor Web (Nginx)

- **Puerto:** 80 (HTTP)
- **Página de bienvenida** personalizada
- **Acceso por nombre:** `http://gamecenter.local`
- **Headers de seguridad** configurados

4. Firewall y Seguridad

- **UFW:** Firewall con reglas específicas
- **fail2ban:** Protección contra ataques de fuerza bruta
- **Puertos abiertos:**
 - 22/tcp (SSH con rate limiting)
 - 53/tcp+udp (DNS)
 - 80/tcp (HTTP)
 - 546-547/udp (DHCPv6)



Gestión de Procesos y Servicios

Linux (Servidor Ubuntu/Debian)

Herramientas Clave

- `top`, `htop` - Monitoreo en tiempo real
- `ps aux` - Lista de procesos
- `systemctl` - Gestión de servicios
- `journalctl` - Logs del sistema
- `ss`, `netstat` - Puertos y conexiones

Comandos Esenciales

```
# Monitorización
top                                # Ver CPU/RAM
ps aux --sort=-%cpu | head -n 20  # Top procesos por CPU
ps aux --sort=-%mem | head -n 20  # Top procesos por memoria

# Gestión de servicios
systemctl status nombre_servicio  # Ver estado
sudo systemctl restart nombre_servicio # Reiniciar
sudo systemctl enable nombre_servicio # Habilitar al inicio
sudo systemctl disable nombre_servicio # Deshabilitar

# Logs
sudo journalctl -u nombre_servicio --since "2 hours ago"
sudo journalctl -p err -b          # Errores del boot actual
sudo journalctl -f                  # Seguir logs en tiempo real

# Puertos y conexiones
```

```
ss -tulnp                                # Ver puertos abiertos
sudo ss -tulnp | grep :80                # Ver quién usa puerto 80
```

Ejemplo: Reiniciar Nginx

```
sudo systemctl restart nginx
sudo systemctl status nginx --no-pager
sudo journalctl -u nginx -n 50
```

Comportamiento ante Cuelgue de Servicio

1. Ver estado: `systemctl status servicio`
2. Revisar logs: `journalctl -u servicio -n 200`
3. Reiniciar: `sudo systemctl restart servicio`
4. Si persiste: `sudo reboot` (con aviso previo)

Windows 11 (Estaciones)

Herramientas Clave

- Administrador de tareas (Task Manager)
- `tasklist` - Lista de procesos
- PowerShell (`Get-Process`, `Get-Service`)
- `services.msc` - Gestión de servicios
- `eventvwr.msc` - Visor de eventos

Comandos PowerShell

```
# Ver procesos top CPU
Get-Process | Sort-Object CPU -Descending | Select-Object -First 10

# Ver procesos top memoria
Get-Process | Sort-Object WS -Descending | Select-Object -First 10

# Gestión de servicios
Get-Service -Name "Spooler"
Restart-Service -Name "Spooler"
Stop-Service -Name "Spooler"
Start-Service -Name "Spooler"

# Ver eventos críticos
Get-EventLog -LogName System -EntryType Error -Newest 50
```

Principios y Convenciones

- **Nombres de cuenta:** rol_area_num (ej: alumno_redes_01, tec_soporte_01)
- **No usar cuentas admin** para tareas diarias
- **Roles definidos:**
 - Estudiante/Jugador
 - Staff/Técnico
 - Administrador

Linux - Gestión de Usuarios

```
# Crear grupo
sudo groupadd alumnos

# Crear usuario
sudo useradd -m -s /bin/bash -G alumnos nombre_usuario
sudo passwd nombre_usuario

# Cambiar propietario y permisos
sudo chown usuario:grupo /ruta/recurso
sudo chmod 750 /ruta/recurso

# ACLs (permisos avanzados)
sudo setfacl -m u:usuario:rwX /ruta/carpeta
getfacl /ruta/carpeta
```

Ejemplo Completo

```
# Crear usuario para jugador
sudo groupadd jugadores
sudo useradd -m -s /bin/bash -G jugadores pepe
sudo passwd pepe

# Crear directorio personal
sudo mkdir -p /srv/games/pepe
sudo chown pepe:jugadores /srv/games/pepe
sudo chmod 750 /srv/games/pepe
```

Compartir Recursos (Samba)

Configuración en `/etc/samba/smb.conf`

```
[games]
    path = /srv/games
    browseable = yes
    read only = no
```

```
valid users = @jugadores
create mask = 0750
directory mask = 0750
```

Agregar Usuario Samba

```
sudo smbpasswd -a pepe
```

Conectar desde Windows

```
net use Z: \\192.168.1.10\games /user:pepe contraseña
```

Automatización de Tareas

Linux - Cron

Editar Crontab

```
crontab -e      # Usuario actual
sudo crontab -e # Root
```

Ejemplos de Tareas

```
# Limpiar /tmp cada día a las 02:00
0 2 * * * /usr/bin/find /tmp -mindepth 1 -mtime +1 -delete

# Backup diario a las 03:00
0 3 * * * /usr/local/bin/backup_rsync.sh

# Actualizar sistema semanalmente (domingos 04:00)
0 4 * * 0 /usr/bin/apt update && /usr/bin/apt -y upgrade >> /var/log/apt-
upgrade.log 2>&1
```

Script de Backup (/usr/local/bin/backup_rsync.sh)

```
#!/bin/bash
SRC="/srv/data/"
DEST="/mnt/backup/data/"
LOG="/var/log/backup_rsync.log"
```

```
rsync -a --delete --exclude='tmp/' $SRC $DEST >> $LOG 2>&1
```

```
sudo chmod +x /usr/local/bin/backup_rsync.sh
```

Windows - Task Scheduler

Script de Limpieza (**limpieza.bat**)

```
@echo off
del /q /f C:\Windows\Temp\*
del /q /f %temp%\*
echo Limpieza completada >> C:\logs\limpieza.log
```

PowerShell Backup (**C:\scripts\backup.ps1**)

```
$source = "C:\Users\Public\Documents"
$dest = "\\192.168.1.10\backup\PC01"

New-Item -ItemType Directory -Path $dest -Force
robocopy $source $dest /MIR /FFT /R:3 /W:5 /LOG:C:\scripts\logs\robocopy-PC01.log
```

Seguridad y Políticas

Contraseñas

- **Longitud mínima:** 12 caracteres
- **Complejidad:** Mayúsculas, minúsculas, números y símbolos
- **Cambio:** Cada 90 días para administradores
- **Prohibido:** Cuentas compartidas

Actualizaciones

Linux

```
# Actualización manual
sudo apt update && sudo apt upgrade -y

# Actualización automática (cron semanal)
0 4 * * 0 /usr/bin/apt update && /usr/bin/apt -y upgrade >> /var/log/apt-
upgrade.log 2>&1
```


Windows

- Programar Windows Update fuera de horario pico
- Mejor control manual en game centers
- Actualizaciones en madrugada

Firewall

Linux (UFW)

```
# Habilitar UFW
sudo ufw enable

# Reglas básicas
sudo ufw allow from 192.168.1.0/24 to any port 22 proto tcp
sudo ufw allow 53/tcp
sudo ufw allow 53/udp
sudo ufw allow 80/tcp
sudo ufw allow 139,445/tcp # Samba

# Ver estado
sudo ufw status verbose
```

Windows

- Configurar reglas en Windows Defender Firewall
- Permitir solo puertos necesarios
- Bloquear tráfico entrante por defecto

Antivirus

- **Windows:** Windows Defender + análisis semanales
- **Linux:** ClamAV (opcional)
- Mantener firmas actualizadas

Mantenimiento y Monitoreo

Checklist Diario

- ☐ Verificar estado del servidor (`top`, `df -h`)
- ☐ Revisar logs de errores (`journalctl -p err -n 100`)
- ☐ Comprobar backups diarios
- ☐ Verificar disponibilidad de servicios
- ☐ Revisar tickets/incidencias

Checklist Semanal

- ☐ Aplicar actualizaciones de seguridad

- ☐ Escaneo antivirus completo
- ☐ Limpieza de logs grandes
- ☐ Probar restauración de archivos desde backup
- ☐ Revisar uso de disco

Checklist Mensual

- ☐ Revisión de cuentas inactivas
- ☐ Limpieza profunda de discos
- ☐ Pruebas de rendimiento
- ☐ Revisión de permisos

Checklist Trimestral

- ☐ Prueba completa de restauración desde backup
- ☐ Revisión de políticas de contraseñas
- ☐ Inventario de hardware
- ☐ Revisión física de equipos

Guía de Uso

Instalación Inicial

```
# 1. Clonar repositorio
git clone <url-repositorio>
cd ansible-gestion-despliegue

# 2. Configurar entorno Ansible
bash scripts/setup/setup-ansible-env.sh --auto

# 3. Activar entorno
source activate-ansible.sh

# 4. Configurar inventario
nano inventory/hosts.ini

# 5. Ejecutar playbook completo
ansible-playbook site.yml
```

Scripts Disponibles

Ejecución de Servicios

```
bash scripts/run/run-network.sh      # Configurar red
bash scripts/run/run-dns.sh          # Configurar DNS
bash scripts/run/run-dhcp.sh         # Configurar DHCP
bash scripts/run/run-web.sh          # Configurar Nginx
```

```
bash scripts/run/run-firewall.sh      # Configurar firewall
bash scripts/run/run-all-services.sh # Ejecutar todo
```

Validación

```
bash scripts/run/validate-network.sh # Validar red
bash scripts/run/validate-dns.sh     # Validar DNS
bash scripts/run/validate-dhcp.sh    # Validar DHCP
bash scripts/run/validate-web.sh     # Validar web
```

Diagnóstico

```
bash scripts/diagnostics/diagnose-dns.sh      # Diagnóstico DNS
bash scripts/diagnostics/test-dns-records.sh  # Probar registros DNS
```

Capturas de Pantalla

Topología de Red

 Topología

Panel de Administración

 Panel

Página Web

 Web

Notas Adicionales

Procedimiento ante Incidentes

1. **Descripción:** Recoger reporte (quién, qué, cuándo)
2. **Impacto:** ¿Afecta a todos o solo a una máquina?
3. **Contención:** Aislar máquina/red si es necesario
4. **Diagnóstico:** Revisar logs, procesos, recursos
5. **Mitigación:** Reinicio, restaurar backup, aplicar parche
6. **Recuperación:** Volver a servicio normal
7. **Postmortem:** Documentar causa raíz y prevención

Template de Reporte de Incidente

Fecha/hora:
Reportado por:
Afectados:
Síntomas:
Acciones tomadas:
Resultado:
Recomendaciones:

Referencias

- [Documentación de Ansible](#)
- [BIND9 Documentation](#)
- [Nginx Documentation](#)
- [Ubuntu Server Guide](#)

Licencia

Este proyecto es parte de un trabajo académico para el curso de Sistemas Operativos.

Última actualización: Noviembre 2025