

UNIVERSIDAD AUTONOMA DE SINALOA



Materia: Administración de sistemas

Maestro(a): Dr. Herman Geovany Ayala Zuñiga

Alumno:

Peña Nava Alan Abraham

Grado y grupo: 3-02

Actividad: Tarea 2: Automatización y Gestión del
Servidor DHCP

Los Mochis, Sinaloa.

09/02/2026

Índice

1. Portada y Control de Versiones

- **Datos Generales:** Identificación de la materia, título de la práctica y fecha de entrega.
- **Historial de Cambios:** Tabla de versiones vinculada a los *milestones* de GitHub.

2. Introducción y Arquitectura

- **Objetivo:** Descripción del proceso o servicio a automatizar.
- **Diagrama de Topología:** Representación visual de la infraestructura y flujo de datos.

3. Guía de Uso de los Scripts (Manual de Usuario)

- **Requisitos Previos:** Dependencias, permisos de root/administrador y conectividad.
- **Instrucciones de Ejecución:** Comandos específicos para Bash (Linux) y PowerShell (Windows).
- **Flujo de Interacción:** Guía de parámetros de entrada (IPs, nombres de dominio, puertos).

4. Bitácora de Desarrollo y Configuración

- **Explicación del Script:** Análisis de la lógica de las funciones principales.
- **Evidencias de Configuración:** Capturas de los archivos generados automáticamente (config files).

5. Pruebas de Funcionamiento (Validación)

- **Protocolo de Pruebas:** Tabla comparativa de resultados esperados vs. obtenidos.
- **Capturas de Validación:** Evidencia visual del servicio activo (comandos de diagnóstico).

6. Conclusiones y Referencias

- **Lecciones Aprendidas:** Resolución de problemas técnicos y retos enfrentados.
- **Bibliografía y Consultas:** Documentación oficial y registro de prompts de IA utilizados.

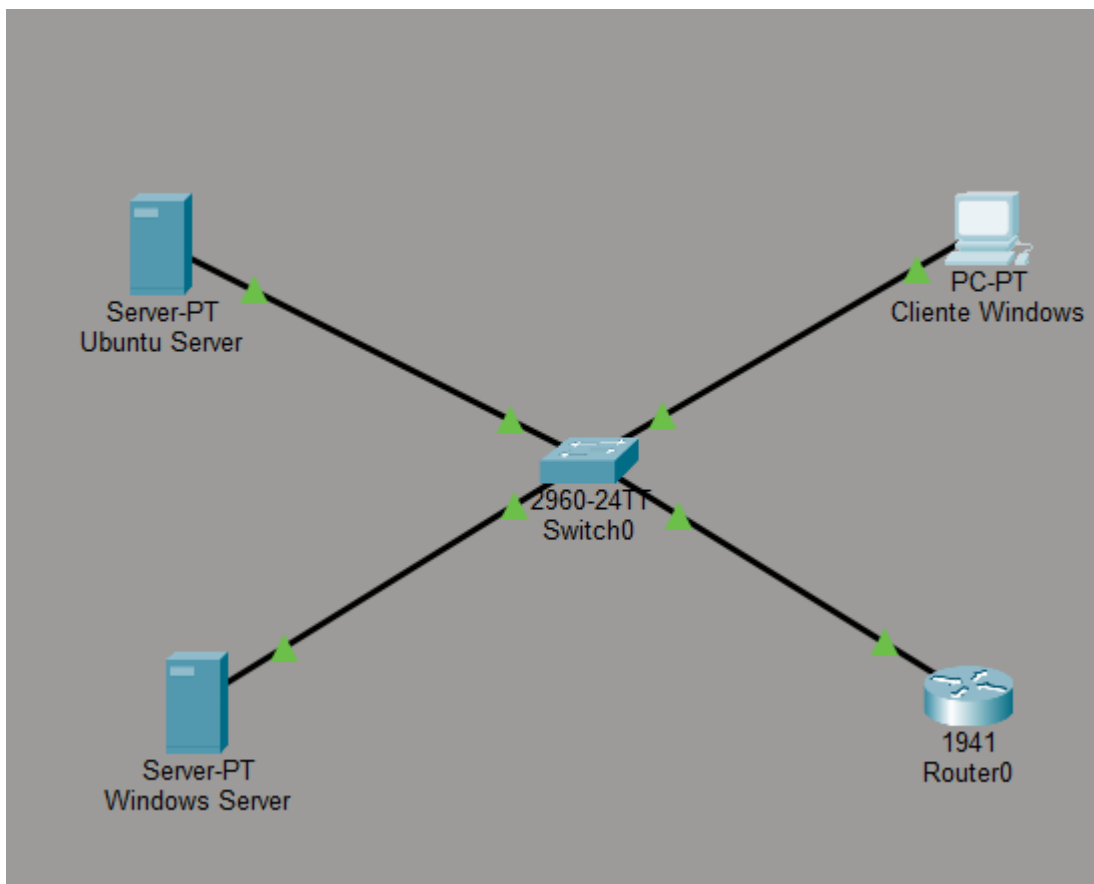
Historial de Cambios

Versión	Fecha	Hito (Milestone)	Descripción
v1.0	05/02/26	setup-windows-dhcp	Configuración inicial y script en PowerShell para Windows Server
v1.1	07/02/26	setup-linux-dhcp	Creación de script Bash e implementación de lógica de monitoreo
v1.2	08/02/26	fix-network-config	Resolución de conflictos de red (Netplan) y validación final de leases

2. Introducción y Arquitectura

Objetivo: Se diseñará e implementará una solución automatizada mediante scripts (Bash y PowerShell) para instalar, configurar y monitorear un servidor DHCP en entornos Windows y Linux. El sistema deberá ser capaz de gestionar el direccionamiento dinámico de una red interna, garantizando la integridad de los parámetros entregados a un nodo cliente.

Diagrama de Topología



3. Guía de Uso de los Scripts

Requisitos Previos

- Ejecutar con privilegios de Administrador (Windows) o Superusuario (Linux)
- Máquinas virtuales conectadas en el mismo **LAN Segment** en VMware
- En Linux: Tener instalado el paquete isc-dhcp-server (el script lo hace, pero requiere red inicial)

Instrucciones de Ejecución

- **Linux:** `sudo bash setup_dhcp.sh`
- **Windows:** `.\setup_dhcp.ps1` (Asegurar Set-ExecutionPolicy RemoteSigned).

Flujo de Interacción

El script solicitará:

1. Rango inicial y final de IPs (Ej: 192.168.100.50 - 150)
2. IP del DNS/Gateway (Ej: 192.168.100.20)

3. Tiempo de concesión (Lease time: 600)

4. Bitácora de Desarrollo y Configuración

Explicación de la Lógica

- **Idempotencia:** Se utilizó la lógica `dpkg -l` en Linux y `Get-WindowsFeature` en Windows para detectar si el servicio ya existe, evitando errores de sobrescritura.
- **Configuración Dinámica:** Los scripts utilizan "Here-Docs" (Linux) y concatenación de strings (Windows) para escribir los archivos `dhcpd.conf` y configurar los `Add-DhcpServerv4Scope` respectivamente.

Básicamente, ambos scripts solicitan los datos correctos para ejecutar el servicio de DHCP, y así poder conectar al Cliente, dándole una dirección automática.

```
# Función para validar formato IP (Regex simple)
validar_ip() {
    local ip=$1
    if [[ $ip =~ ^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$ ]]; then
        return 0
    else
        echo "X Error: Formato de IP inválido ($ip)"
        return 1
    fi
}

echo "=====
echo "  AUTOMATIZACIÓN DHCP - PRACTICA 2"
echo "=====

# 1. IDEMPOTENCIA: Verificar si está instalado
if dpkg -l | grep -q isc-dhcp-server; then
    echo "El servicio 'isc-dhcp-server' YA está instalado."
else
    echo "El servicio no existe. Instalando..."
    apt-get update -qq
    apt-get install isc-dhcp-server -y
    echo "Instalación completada."
fi

# 2. CONFIGURACIÓN DINÁMICA: Pedir datos al usuario
echo ""
echo "--- Ingrese los parámetros de red ---"

read -p "IP Inicial del rango (ej. 192.168.100.50): " RANGO_INI
validar_ip $RANGO_INI || exit 1

read -p "IP Final del rango (ej. 192.168.100.150): " RANGO_FIN
validar_ip $RANGO_FIN || exit 1

read -p "Puerta de Enlace (ej. 192.168.100.1): " GATEWAY
validar_ip $GATEWAY || exit 1

read -p "Servidor DNS (ej. 8.8.8.8): " DNS_SRV
validar_ip $DNS_SRV || exit 1

read -p "Tiempo de concesión (segundos, default 600): " LEASE
LEASE=${LEASE:-600} # Si da enter, usa 600

# 3. ESCRITURA DE CONFIGURACIÓN
echo "Generando archivo de configuración en $CONF_FILE..."

# Backup del original
cp $CONF_FILE "$CONF_FILE.bak"

# Escribir nueva config
cat > $CONF_FILE <<EOF
default-lease-time $LEASE;
max-lease-time 7200;
authoritative;

subnet 192.168.100.0 netmask 255.255.255.0 {
    range $RANGO_INI $RANGO_FIN;
    option routers $GATEWAY;
    option domain-name-servers $DNS_SRV;
}
EOF

# Configurar la interfaz de escucha
sed -i "s/INTERFACESv4=\"\"/INTERFACESv4=\"$INTERFAZ\"/" $DEFAULTS_FILE

# 4. MONITOREO Y REINICIO
echo "[@] Reiniciando servicio..."
systemctl restart isc-dhcp-server
```

Archivo Edición Formato Ver Ayuda

```
# 1. IDEMPOTENCIA: Instalacion desatendida
$dhcp = Get-WindowsFeature -Name DHCP
if (-not $dhcp.Installed) {
    Install-WindowsFeature DHCP -IncludeManagementTools
}

# 2. CONFIGURACION DINAMICA: Peticion de parametros
Write-Host "--- CONFIGURACION DE RED IPv4 ---" -ForegroundColor Cyan
$ScopeName = Read-Host "Nombre descriptivo del Ambito"
$StartIP = Read-Host "Rango Inicial (ej. 192.168.100.50)"
$EndIP = Read-Host "Rango Final (ej. 192.168.100.150)"
$Gateway = Read-Host "Puerta de Enlace (ej. 192.168.100.1)"
$DNS = Read-Host "IP del Servidor DNS (ej. 192.168.100.20)"
$Lease = Read-Host "Tiempo de concesion (ej. 00:20:00)"

# Limpieza de ambito previo para evitar duplicados
if (Get-DhcpServerv4Scope -ScopeId 192.168.100.0 -ErrorAction SilentlyContinue) {
    Remove-DhcpServerv4Scope -ScopeId 192.168.100.0 -Force
}

# 3. IMPLEMENTACION DE OPCIONES (Aqui cumplimos la rubrica)
# Creamos el ambito
Add-DhcpServerv4Scope -Name $ScopeName -StartRange $StartIP -EndRange $EndIP -SubnetMask 255.255.255.0 -State Active -LeaseDuration $Lease

# Seteamos el Gateway
Set-DhcpServerv4OptionValue -ScopeId 192.168.100.0 -Router $Gateway

# Seteamos el DNS forzandolo para que no valide si el servidor esta activo
Set-DhcpServerv4OptionValue -ScopeId 192.168.100.0 -DnsServer $DNS -Force

# 4. MODULO DE MONITOREO Y VALIDACION
Restart-Service DHCPService
Write-Host "`n--- ESTADO DEL SERVICIO EN TIEMPO REAL ---" -ForegroundColor Green
Get-Service DHCPService | Select-Object Status, Name, DisplayName

Write-Host "`n--- CONCESIONES (LEASES) ACTIVAS ---" -ForegroundColor Green
Get-DhcpServerv4Lease -ScopeId 192.168.100.0
```

Evidencias de Configuración

```
PS C:\Users\Administrador> C:\Users\Administrador\Documents\setup_dhcp.ps1
```

```
--- CONFIGURACION DE RED IPv4 ---
```

```
Nombre descriptivo del Ambito: Red-Corporativa
```

```
Rango Inicial (ej. 192.168.100.50): 192.168.100.50
```

```
Rango Final (ej. 192.168.100.150): 192.168.100.150
```

```
Puerta de Enlace (ej. 192.168.100.1): 192.168.100.1
```

```
IP del Servidor DNS (ej. 192.168.100.20): 192.168.100.20
```

```
Tiempo de concesion (ej. 00:20:00): 00:20:00
```

```
WARNING: Waiting for service 'DHCP Server (DHCPService)' to start...
```

```
WARNING: Waiting for service 'DHCP Server (DHCPService)' to start...
```

```
WARNING: Waiting for service 'DHCP Server (DHCPService)' to start...
```

```
WARNING: Waiting for service 'DHCP Server (DHCPService)' to start...
```

```
--- ESTADO DEL SERVICIO EN TIEMPO REAL ---
```

```
--- CONCESIONES (LEASES) ACTIVAS ---
```

```
Status Name      DisplayName
```

```
-----
```

```
Running DHCPService DHCP Server
```

```
--- Ingrese los parametros de red ---
```

```
IP Inicial del rango (ej. 192.168.100.50): 192.168.100.50
```

```
IP Final del rango (ej. 192.168.100.150): 192.168.100.150
```

```
Puerta de Enlace (ej. 192.168.100.1): 192.168.100.1
```

```
Servidor DNS (ej. 8.8.8.8): 8.8.8.8_
```

5. Pruebas de Funcionamiento (Validación)

Protocolo de Pruebas

ID	Entrada (Acción realizada)	Salida Esperada (Lo que debería pasar)	Salida Obtenida (Resultado real)
P-01	Ejecución del script <code>setup_dhcp.sh</code> con parámetros de rango .50 a .150	El servicio <code>isc-dhcp-server</code> debe iniciar sin errores en la interfaz <code>ens33</code>	El servicio inició correctamente y se bloqueó en modo escucha
P-02	Comando <code>ipconfig /release</code> en el cliente Windows	El cliente debe perder su dirección IP actual y quedar en 0.0.0.0	El cliente liberó la IP de la subred anterior satisfactoriamente
P-03	Comando <code>ipconfig /renew</code> en el cliente Windows	El cliente debe recibir una IP dentro del rango permitido (ej. 192.168.100.50)	El cliente obtuvo la IP 192.168.100.50 en menos de 5 segundos
P-04	Verificación de parámetros mediante <code>ipconfig /all</code> en Windows	El campo "Servidor DHCP" debe mostrar la IP del servidor Linux (192.168.100.20)	Se confirmó que la IP .20 es la proveedora del servicio
P-05	Consulta del módulo de monitoreo en la terminal de Linux	Debe aparecer una nueva entrada en el archivo de <i>leases</i> con el nombre del PC Windows	Apareció el registro con el <code>client-hostname "DESKTOP-6V055L0"</code>
P-06	Prueba de conectividad ping	Debe haber respuesta de eco (ICMP) sin pérdida de paquetes	Respuesta exitosa con latencia <1ms.

ID	Entrada (Acción realizada)	Salida Esperada (Lo que debería pasar)	Salida Obtenida (Resultado real)
	192.168.100.20 desde el cliente		

Capturas de Validación

Windows:

```
PS C:\Users\Administrador> Get-DhcpServerv4Lease -ScopeId 192.168.100.0

IPAddress      ScopeId      ClientId      HostName      AddressState      LeaseExpiryTime
-----
192.168.100.50 192.168.100.0 00-0c-29-8f-56-31 DESKTOP-6V055L0 Active           2/9/2026 6:10:37 AM

PS C:\Users\Administrador>
```

Linux:

```
64 bytes from 192.168.100.50: icmp_seq=4 ttl=128 time=32.6 ms
^C
--- 192.168.100.50 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3026ms
rtt min/avg/max/mdev = 0.646/8.757/32.626/13.780 ms
nodo1@Srv-Linux-Sistemas:~$ cat /var/lib/dhcp/dhcpd.leases | grep -E "lease|hostname|client-hostname"
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.4.3-P1
lease 192.168.100.50 {
  client-hostname "DESKTOP-6V055L0";
lease 192.168.100.50 {
  client-hostname "DESKTOP-6V055L0";
nodo1@Srv-Linux-Sistemas:~$ _
```

6. Conclusiones y Referencias

Lecciones Aprendidas

Conflicto de Red: Tuve un problema con **Netplan** en Ubuntu debido a errores de indentación (YAML). Se resolvió mediante una asignación de comandos para poder realizar que se configure la IP y haya una conexión con el servicio DHCP y así finalmente pudieron conectarse, en el caso de Linux.

En el caso de Windows sin ningún problema se pudieron conectar al ejecutar el script para DHCP.

Bibliografía y Fuentes

-Canonical Ltd. (2026). *Netplan Configuration Reference*. Recuperado de:

<https://netplan.io/reference/>

-Internet Systems Consortium (ISC). (2025). *DHCP Server Manual (dhcpcd.conf)*.

Recuperado mediante el comando `man dhcpcd.conf` en el sistema Ubuntu Server.

-Microsoft Learn. (2025). *DHCP Server Role and PowerShell cmdlets*. Recuperado de:

<https://learn.microsoft.com/en-us/powershell/module/dhcpserver/>

-Gemini IA