



**Universidad Autónoma de Sinaloa**

**ADMINISTRACIÓN DE SISTEMAS**

**Nombre de la práctica:**

Tarea 2: Automatización y Gestión del Servidor DHCP

**Grupo:**

3-02

**Alumno:**

Montes Vázquez

Adrián Tadeo

**Profesor:**

Herman Geovany Ayala Zuñiga

**Repositorio de GitHub:**

<https://github.com/ADNTD1/Administracion-de-Sistemas-T2>

## **CONTROL DE VERSIONES:**

<b>V</b>	<b>Fecha</b>	<b>Descripción del Cambio</b>	<b>Autor</b>
<b>1.0</b>	<b>2026-02-08</b>	<b>Creación de la estructura base del documento en el repo</b>	<b>Alumno</b>
<b>1.1</b>	<b>2026-02-08</b>	<b>creacion del readme</b>	<b>Alumno</b>
<b>1.2</b>	<b>2026-02-09</b>	<b>Creacion del archivo de los entregables.</b>	<b>Alumno</b>

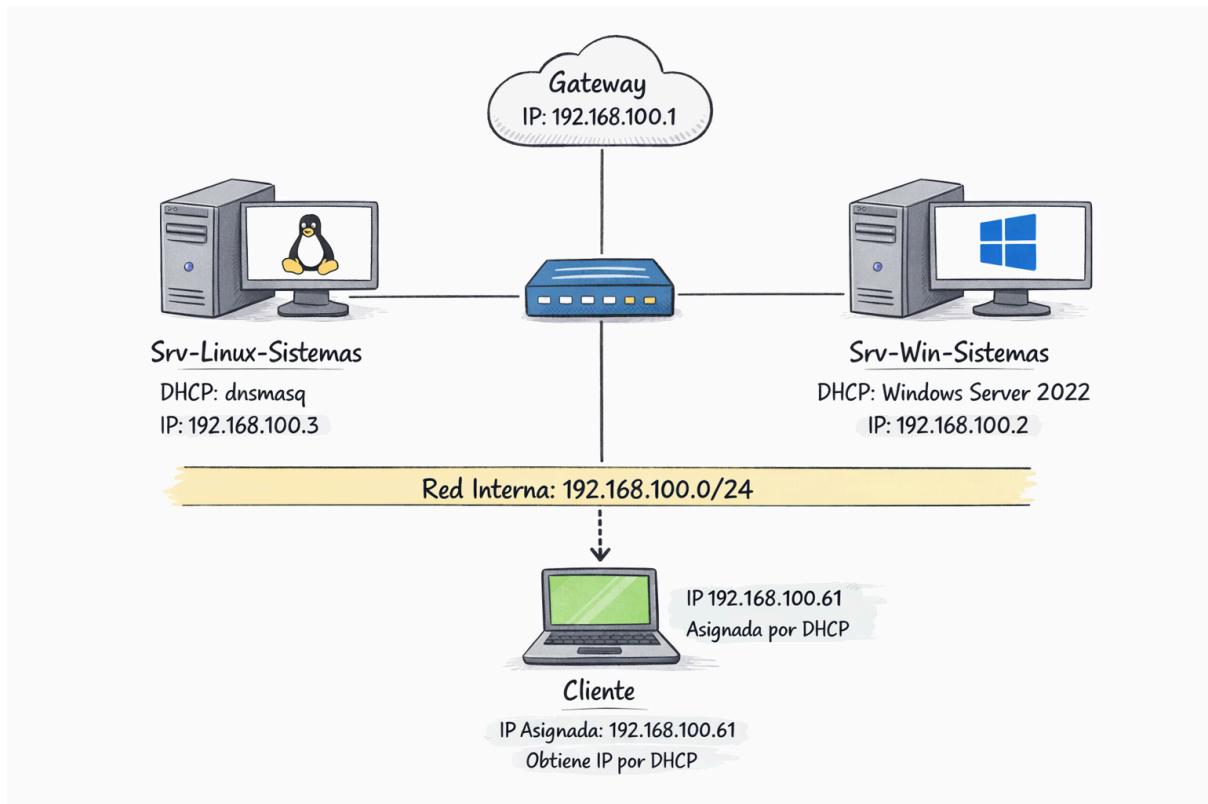
## **2. Introducción y Arquitectura de la Práctica**

### **2.1 Objetivo**

El objetivo de esta práctica es diseñar e implementar una solución automatizada mediante scripts en Bash y PowerShell para instalar, configurar y monitorear un servidor DHCP en entornos Linux (NixOs) y Windows seever.

La solución permite gestionar de forma dinámica el direccionamiento IP de una red interna, garantizando que los clientes reciban correctamente los parámetros de red, tales como dirección IP, puerta de enlace y tiempo de concesión, dns y gateway.

## DIAGRAMA DE TOPOLOGIA DE RED:



# **MANUAL DE USUARIO:**

## **Requisitos de Linux:**

- Tener habilitado el servicio DHCP desde configuration.nix
- Usuario con privilegios root
- Interfaz de red configurada desde el hipervisor

## Requisitos de Windows Server:

- Ip estatica configurada
- tener rol de administrador
- interfaz de red configurada desde el hipervisor.

## **Instrucciones de ejecución:**

Linux:

- Posicionarse en el directorio donde se encuentra el script
- ejecutarlo con: `sudo bash dhcp_linux_nixos.sh`

Windows:

- Posicionarse en el directorio donde se encuentra el script
- ejecutarlo con: `.\dhcp_windows.ps1`

## **Flujo de Interacción**

Durante la ejecución, los scripts solicitan al administrador:

- Nombre del ámbito (Scope).
- Dirección IP inicial.
- Dirección IP final.
- Puerta de enlace (Gateway).

Servidor DNS .

Tiempo de concesión (Lease Time).

La ejecución de este script se mira de esta manera con la terminal pidiéndote los parámetros necesarios:

```
Scope:red_sistemas
IP inicial:192.168.100.50
IP final:192.168.100.150
Lease time:24h
Gateway192.168.100.1
Dns: 192.168.100.3
building the system configuration...
updating GRUB 2 menu...
activating the configuration...
setting up /etc...
reloading user units for adrian...
restarting sysinit-reactivation.target
the following new units were started: NetworkManager-dispatcher.service
Done. The new configuration is /nix/store/my5d44yddjirz7a2fuksbbkcy45m
● dnsmasq.service - Dnsmasq Daemon
  Loaded: loaded (/etc/systemd/system/dnsmasq.service; enabled; pre
    Active: active (running) since Sun 2026-02-08 19:41:47 CST; 2h 16
  TriggeredBy: 222E1f-4C29241be918a294491e71504
```

### Explicación del script:

El script solicita el Scope y se valida si existe, si si existe se solicita la IP inicial que dentro hace uso de una función que está en otro archivo la cual valida su formato haciendo uso de regex, si es válida pregunta la siguiente IP, después se solicita el Lease time y dentro se convierte a un formato de segundos para que se pueda aplicar a la configuración, después de eso se imprime en pantalla el status del servicio, en la captura se observa que está Active (running).

Con esa salida en pantalla se puede saber que el script se ejecutó correctamente.

## 4. Bitácora de Desarrollo y Configuración

### Explicación del Script

#### -Verificación de instalación:

El script valida la existencia del servicio DHCP antes de instalarlo, garantizando idempotencia.

#### -Configuración dinámica:

Se solicitan parámetros al usuario para crear el scope y las opciones DHCP.  
Activación del servicio:

El ámbito DHCP se activa explícitamente para permitir la asignación de IPs.

-Monitoreo:

Scripts independientes consultan el estado del servicio y las concesiones activas.

## EVIDENCIAS DE CONFIGURACION:

Linux:

Archivo de configuracion del sistema operativo (configuration.nix)

```
#Servicio de DHCP en la interfaz ens37

services.dnsmasq = {
    enable = true;

    settings = {
        interface = "ens37";
        bind-interfaces = true;
    };
}
```

Salida de los servicios después de haber descargado el Feature en windows:

```
PS C:\Users\Administrator\Desktop\Scripts> notepad create_scope.ps1
PS C:\Users\Administrator\Desktop\Scripts> Get-DhcpServerv4Scope
ScopeId      SubnetMask      Name      State      StartRange      EndRange      LeaseDuration
-----      -----      -----      -----      -----      -----      -----
192.168.100.0  255.255.255.0  Scope-Red-S...  Active  192.168.100.50  192.168.100.150  8.00:00:00
```

## PRUEBAS DE FUNCIONAMIENTO;

Entrada	Salida Esperada	Salida Obtenida
Cliente solicita IP	IP dentro del rango	192.168.100.X
Scope activo	Servicio DHCP operativo	Confirmado
Release / Renew	Nueva concesión	Exitosa

Windows - Linux Cliente:

Prueba de la asignacion de la IP solicitada desde el cliente (NixOS):

```
[adrian@nixos:~/Escritorio/Scripts]$ nano test_dhcp.sh
[adrian@nixos:~/Escritorio/Scripts]$ ./test_dhcp.sh
eliminando la ip...
El dispositivo «ens37» ha sido desconectado correctamente.
Renovando ip...
El dispositivo «ens37» se activó correctamente con «ddf1bcf5-0895-34c3-8020-e5ca65794620»

nueva ip:
Ip: 192.168.100.61/24
gateway: 192.168.100.1
```

Prueba del monitoreo de red:

```
Concesiones activas:

IPAddress      ClientId      HostName AddressState LeaseExpiryTime
-----        -----
192.168.100.61 00-0c-29-7a-0c-6a nixos    Active       2/17/2026 2:34:04 PM
```

**Linux Server - Linux Cliente:**

Prueba de la asignacion de la ip a traves del servicio:

```
[adrian@nixos:~]$ ip route
default via 192.168.100.1 dev ens37 proto dhcp src 192.168.100.61 metric 100
192.168.100.0/24 dev ens37 proto kernel scope link src 192.168.100.61 metric 100
```

Prueba del monitor de red:

```
[adrian@Srv-Linux-Sisremas:~/Scripts/DHCP]$ ./dhcp_monitor.sh
estado del servicio:
active
concesiones activas:
IP: 192.168.100.61 Host: nixos

[adrian@Srv-Linux-Sisremas:~/Scripts/DHCP]$
```

## **6. Conclusiones y Referencias**

### **Lecciones Aprendidas**

- Importancia de la idempotencia en scripts de automatización.
- Diferencias entre administración declarativa (NixOS) e imperativa (Windows).
- Manejo de validaciones internas del servicio DHCP en Windows Server.
- Uso de scripts de monitoreo para diagnóstico en tiempo real.

### **Conclusión:**

Con esta practica se practica mucho el uso de Bash y Shell, ademas de comprender como se interactua desde un cliente a un servidor DHCP, El hecho de configurar Nix Os desde ls configuracion del sistema fue mas compleja que solo descargar un paquete como se configura en windows, pero el manejo de todo es mas libre que el de windows con instrucciones menos complejas para el desarrollo de los scripts en este sistema operativo.

## **BIBLIOGRAFIA:**

**DHCP Server Overview (Windows Server)**

<https://learn.microsoft.com/en-us/windows-server/networking/technologies/dhcp/>

**DHCP Server Cmdlets in Windows PowerShell**

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

**NixOS**

**NixOS Manual (Stable)**

<https://nixos.org/manual/nixos/stable/>

**IAs utilizadas:**

**ChatGPT: para la configuracion de NixOS y para la creacion de los scripts, ademas de informacion relacionada con errores del dns en la configuracion de el servicio en Windows**