



Universidad Autónoma de Sinaloa

Facultad de Ingeniería Los Mochis

Materia:

“Administración de sistemas”

Tarea 1:

“Entorno de Virtualización e Infraestructura Base”

Profesor:

Ayala Zuñiga Herman Geovany

Integrante:

Laurean Acosta Luis Donaldo

Fecha:

05/01/2026

Tabla de historial de cambios (GitHub)

Versión	Fecha	Autor	Descripción del Cambio
v1.0	05/02/2026	Laurean Luis	Commit Inicial: Creación de scripts base (.sh y .ps1).
v1.1	05/02/2026	Laurean Luis	Fix: Corrección de lógica de colores en PowerShell y permisos en Bash.
v1.2	05/02/2026	Laurean Luis	Update: Incluimos un README

Tabla de direccionamiento

Nodo	Función	Versión	Hostname	IP
Nodo 1	Servidor NixOs	nixos-minimal-25.11.4882	Srv-Linux-Sistemas	192.168.10.1
Nodo 2	Windows Server	SERVER_EVAL_x64FRE_es-es	Srv-Win-Sistemas	192.168.10.2
Nodo 3	Cliente Win 10	Windows10 Pro	Cli-Win10-Sistemas	192.168.10.3

Introducción

Objetivo:

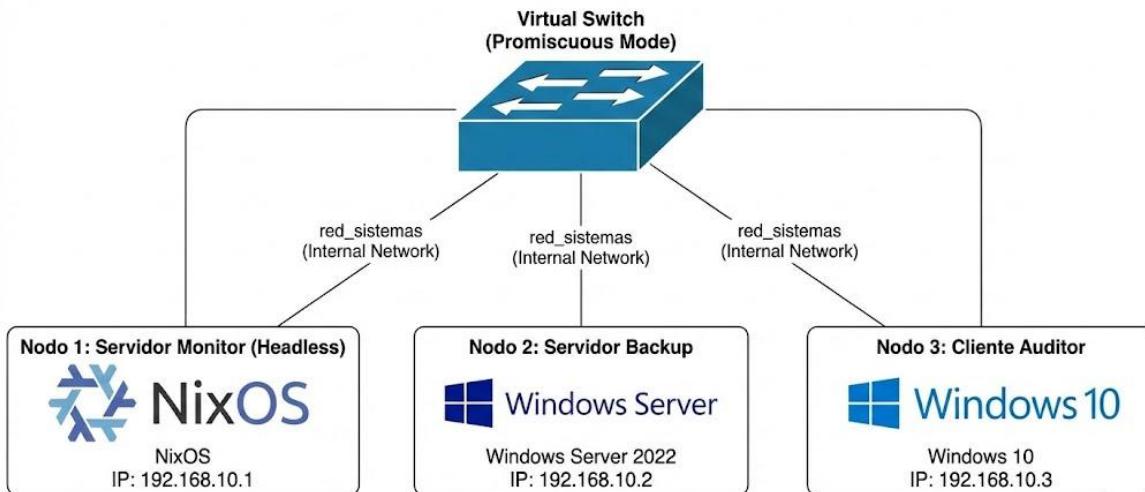
Automatizar la verificación de conectividad y latencia en una infraestructura de red heterogénea (Linux/Windows), asegurando la comunicación bidireccional entre tres nodos críticos (Monitor, Backup y Cliente) mediante scripts personalizados.

Diagrama de Topología:

Se implementó una arquitectura de red interna aislada (`red_sistemas`) conectada a un switch virtual en modo promiscuo.

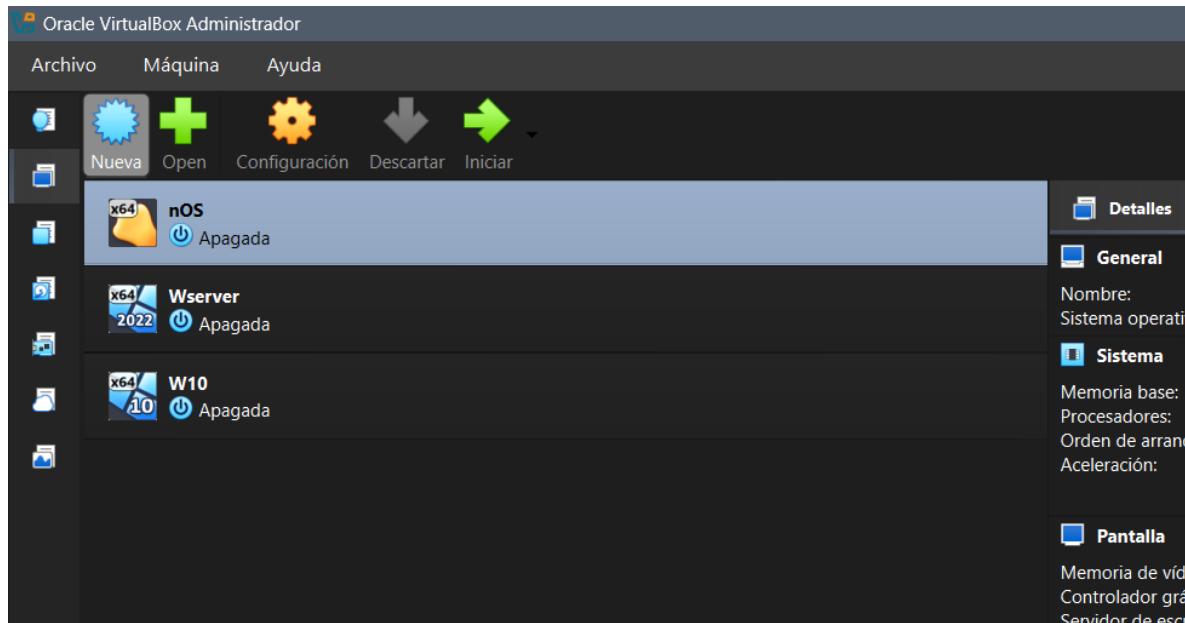
- **Nodo 1 (NixOS):** 192.168.10.1 (Servidor Monitor - Headless)
- **Nodo 2 (Win Server):** 192.168.10.2 (Servidor Backup)
- **Nodo 3 (Win 10):** 192.168.10.3 (Cliente Auditor)

Diagrama de Topología de Red Virtual: `red_sistemas`



Guía uso de los scripts (Manual de Usuario)

Captura de las 3 máquinas virtuales ya preconfiguradas listas para la practica a realizar:



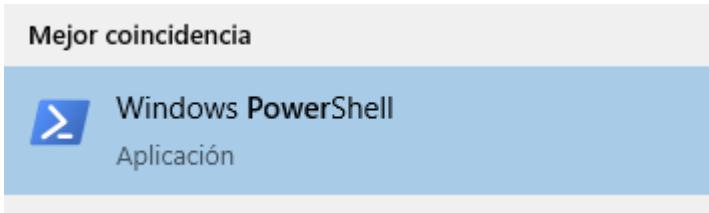
Requisitos Previos:

- **General:** Las máquinas virtuales deben estar encendidas y conectadas a la red **red_sistemas**.
- **Windows 10 y Server:** PowerShell con permisos de Administrador y Política de Ejecución configurada (**Set-ExecutionPolicy Bypass**).
- **Linux:** Permisos de ejecución en el archivo (**chmod +x**).

Instrucciones de Ejecución:

A. En Windows Server(Sevidor):

1. Abrir PowerShell como Administrador.



2. Navegar a la carpeta del script.

```
PS C:\Users\Administrador> Set-Content -Path "C:\tarea1_diagnostico.ps1" -Value @"
>> Write-Host "===== -ForegroundColor Cyan
>> Write-Host " DIAGNOSTICO - SRV-WINDOWS " -ForegroundColor Cyan
>> Write-Host "===== -ForegroundColor Cyan
>> Write-Host "HOSTNAME: $(hostname)"
>> Write-Host "IP INTERNA: $((Get-NetIPAddress -InterfaceAlias '*Ethernet 2*' -AddressFamily IPv4).IPAddress)"
>> Write-Host "Disco: $($((Get-Volume C).SizeRemaining / 1GB) GB Libres"
>> Write-Host "===== -ForegroundColor Cyan
>> pause
>> "@
PS C:\Users\Administrador>
```

3. Dar permisos:

```
PS C:\Users\Administrador> Set-ExecutionPolicy Bypass -Scope Process
Cambio de directiva de ejecución
La directiva de ejecución te ayuda a protegerte de scripts en los que no confías. Si cambias dicha directiva, podrías exponerte a los riesgos de seguridad descritos en el tema de la Ayuda about_Execution_Policies en https://go.microsoft.com/fwlink/?LinkID=135170. ¿Quieres cambiar la directiva de ejecución?
[S] Sí [O] Sí a todo [N] No [T] No a todo [U] Suspender [?] Ayuda (el valor predeterminado es "N"): S
PS C:\Users\Administrador>
```

4. Ejecutar el comando:

```
PS C:\Users\Administrador> C:\tarea1_diagnostico.ps1
=====
DIAGNOSTICO - SRV-WINDOWS
=====
HOSTNAME: Srv-Win-Sistemas
IP INTERNA: 192.168.10.2
Disco: 39.1935348510742 GB Libres
=====
Presione Entrar para continuar...:
```

Descripción: Hostname Srv-Win-Sistemas, IP estática 192.168.10.1 y Script de diagnóstico configurados.

B. En NixOS (Servidor):

1. Abrir terminal.

```
[root@Srv-Linux-Sistemas:~]#
```

2. Navegar por la carpeta del script:

```
[root@Srv-Linux-Sistemas:~]# nano tarea1_diagnostico.sh
```

```
GNU nano 8.7                                     tarea1_diagn
#!/bin/sh

echo "====="
echo "  DIAGNOSTICO - SRV-LINUX"
echo "====="
echo ""
echo "HOSTNAME: $(hostname)"
echo "IP INTERNA: $(ip addr show enp0s8 | grep 'inet ' | awk '{print $2}')"
echo "DISCO: $(df -h / | awk 'NR==2 {print $2 " TOTAL / " $4 " LIBRE"}')"
echo "=====
```

3. Ahora le damos permiso al script:

```
[root@Srv-Linux-Sistemas:~]# chmod +x tarea1_diagnostico.sh
```

4. Ejecutamos el script:

```
[root@Srv-Linux-Sistemas:~]# ./tarea1_diagnostico.sh
```

```
[root@Srv-Linux-Sistemas:~]# ./tarea1_diagnostico.sh
=====
  DIAGNOSTICO - SRV-LINUX
=====

HOSTNAME: Srv-Linux-Sistemas
IP INTERNA: 192.168.10.1/24
DISCO: 20G TOTAL / 17G LIBRE
=====
```

Descripción: Hostname Srv-Linux-Sistemas, IP estática 192.168.10.1 y Script de diagnóstico configurados.

C. En Windows 10 (Cliente):

1. Abrir PowerShell como adm:



2. Abrir la carpeta del script:

```
Presione Entrar para continuar...:
PS C:\Windows\system32> Set-Content -Path "C:\diagnostico_cliente.ps1" -Value @"
>> '$pingServer = Test-Connection -ComputerName 192.168.10.2 -Count 1 -Quiet
>> '$pingLinux = Test-Connection -ComputerName 192.168.10.1 -Count 1 -Quiet
>> Write-Host "======" -ForegroundColor Yellow
>> Write-Host " DIAGNOSTICO - CLIENTE - WINDOWS 10 " -ForegroundColor Yellow
>> Write-Host "======" -ForegroundColor Yellow
>> Write-Host "EQUIPO: $(hostname)"
>> Write-Host "IP LOCAL (RED INTERNA): $((Get-NetIPAddress -InterfaceAlias "Ethernet 2" -AddressFamily IPv4).IPAddress)"
>> Write-Host -Nonewline "ESTADO NODO 2 (Win-Srv): "
>> if ('$pingServer') { Write-Host "CONECTADO" -ForegroundColor Green } else { Write-Host "DESCONECTADO" -ForegroundColor Red }
>> Write-Host -Nonewline "ESTADO NODO 1 (NixOS-Srv): "
>> if ('$pingLinux') { Write-Host "CONECTADO" -ForegroundColor Green } else { Write-Host "DESCONECTADO" -ForegroundColor Red}
>> Write-Host "======" -ForegroundColor Yellow
>> pause
>> @"
```

3. Damos permiso:

Set-ExecutionPolicy Bypass

4. Ejecutamos el script:

```
PS C:\Windows\system32> Set-ExecutionPolicy Bypass -Scope Process -Force
PS C:\Windows\system32> C:\diagnostico_cliente.ps1
=====
    DIAGNOSTICO - CLIENTE - WINDOWS 10
=====
EQUIPO: Cli-Win-Sistemas
IP LOCAL (RED INTERNA): 192.168.10.3
ESTADO NODO 2 (Win-Srv): CONECTADO
ESTADO NODO 1 (NixOS-Srv): CONECTADO
=====
Presione Entrar para continuar...:
```

Descripción: Hostname Cli-Win-Sistemas, IP estática 192.168.10.3 y Script de diagnóstico de confirmación de conexión a los 2 nodos exitoso.

Pruebas de Funcionamiento (Validación)

Protocolo de Pruebas:

Entrada (Acción)	Salida Esperada	Salida Obtenida	Estado
Ping a 192.168.10.1	Respuesta TTL=64 / 128	Respuesta desde 192.168.10.1 bytes=32...	Éxito
Ping a 192.168.10.2	Respuesta TTL=128	Respuesta desde 192.168.10.2 bytes=32...	Éxito
Ejecución Script .ps1	Mensajes en VERDE	[Ver captura abajo]	Éxito

Capturas de validación:

```
PS C:\Windows\system32> Set-ExecutionPolicy Bypass -Scope Process -Force
PS C:\Windows\system32> C:\diagnostico_cliente.ps1
=====
      DIAGNOSTICO - CLIENTE - WINDOWS 10
=====
EQUIPO: Cli-Win-Sistemas
IP LOCAL (RED INTERNA): 192.168.10.3
ESTADO NODO 2 (Win-Srv): CONECTADO
ESTADO NODO 1 (NixOS-Srv): CONECTADO
=====
Presione Entrar para continuar...:
```

```
PS C:\Windows\system32> ping 192.168.10.1

Haciendo ping a 192.168.10.1 con 32 bytes de datos:
Respuesta desde 192.168.10.1: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.10.1: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.10.1: bytes=32 tiempo<1ms TTL=64
Respuesta desde 192.168.10.1: bytes=32 tiempo<1ms TTL=64

Estadísticas de ping para 192.168.10.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
                (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 1ms, Media = 0ms
PS C:\Windows\system32> ping 192.168.10.2

Haciendo ping a 192.168.10.2 con 32 bytes de datos:
Respuesta desde 192.168.10.2: bytes=32 tiempo=1ms TTL=128
Respuesta desde 192.168.10.2: bytes=32 tiempo<1ms TTL=128
Respuesta desde 192.168.10.2: bytes=32 tiempo<1ms TTL=128
Respuesta desde 192.168.10.2: bytes=32 tiempo<1ms TTL=128

Estadísticas de ping para 192.168.10.2:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
                (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 1ms, Media = 0ms
```

Conclusiones y Referencias

Lecciones Aprendidas (Solución de Problemas):

Conflicto de Direcciones MAC (Capa de Enlace): Se detectó que la interfaz NAT y la interfaz de Red Interna compartían la misma dirección MAC en el Nodo 3, causando que la configuración de IP estática se aplicara al adaptador incorrecto (WAN en lugar de LAN).

Solución: Se identificó la interfaz correcta mediante **Get-NetAdapter**, se eliminó la configuración errónea con **Remove-NetIPAddress** y se asignó la **IP 192.168.10.3** a la interfaz física correcta, logrando conectividad inmediata.

Bibliografía y Herramientas

- **Manuales:** NixOS Manual (<https://nixos.org/manual/nixos/stable/#sec-installation-booting>).
- **Herramienta de IA:** Google Gemini 3 Pro (Asistencia en depuración de scripts PowerShell, diagnóstico de conflictos de direcciones MAC y resolución de problemas de drivers en VirtualBox).
- **Repositorio GitHub:** [LuisD48/adm-sistemas-prac1](https://github.com/LuisD48/adm-sistemas-prac1)