

# **UNIVERSIDAD AUTONOMA DE SINALOA**

UNIDAD ACADEMICA FACULTAD DE INGENIERIA MOCHIS

LICENCIATURA EN INGENIERIA DE SOFTWARE



ASIGNATURA:

**Asministración de sistemas**

DOCENTE:

**Dr. Herman Geovany Ayala Zuñiga**

ALUMNO:

**Alberto Torres Chaparro**

GRUPO:

**3-01**

TAREA 1:

**Entorno de Virtualización e Infraestructura Base**

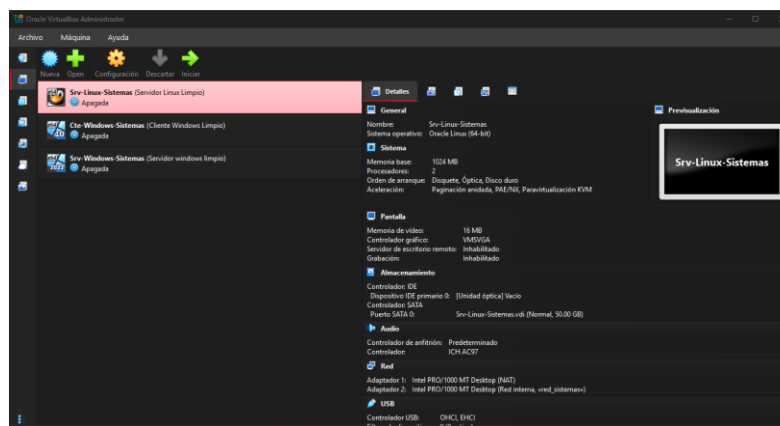
## Objetivo de la tarea

Configurar el entorno de trabajo que se usara durante el semestre en la materia de administración de sistemas.

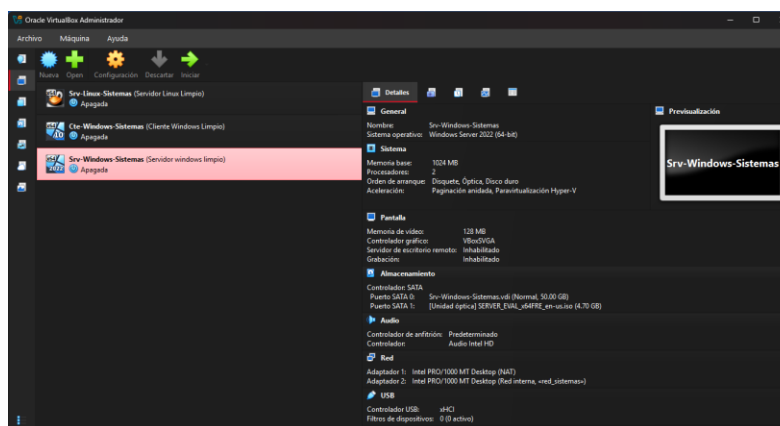
Para ello en el hipervisor VirtualBox configure 3 nodos

1. Servidor Linux
2. Servidor Windows
3. Cliente Windows

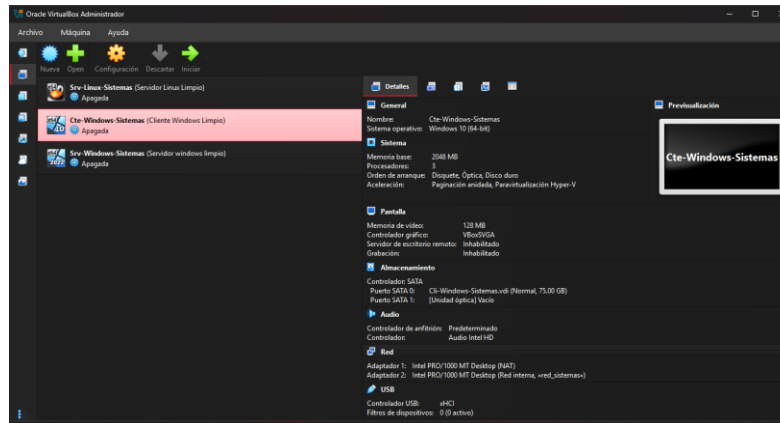
Las siguientes imágenes muestran la **configuración de hardware** de cada nodo en VirtualBox



Nodo#1 Servidor Linux (Oracle Linux Server 10.1)



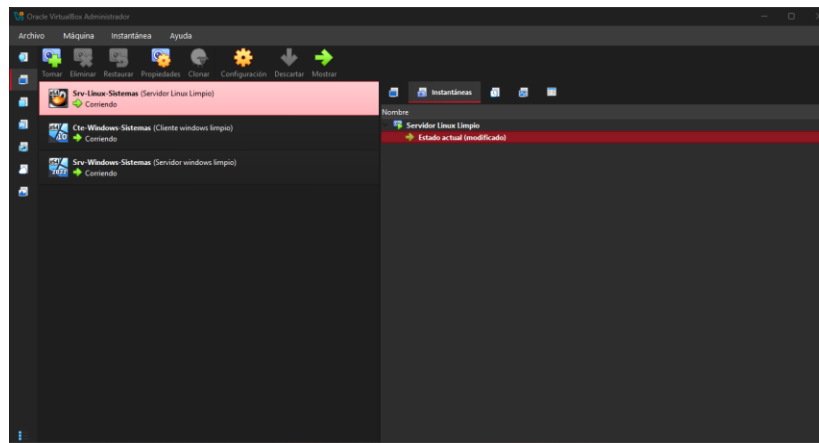
Nodo #2 Servidor Windows (Windows Server 2022)



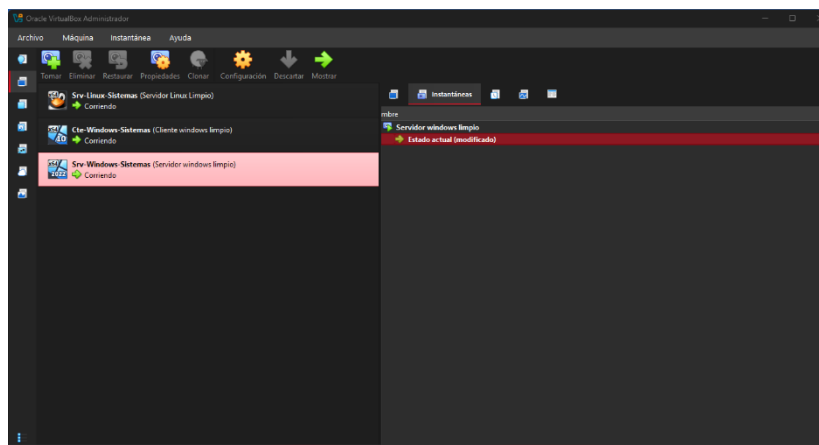
Nodo #3 Cliente Windows (Windows 10)

## Snapshots que guardan un estado “Limpio” de cada nodo

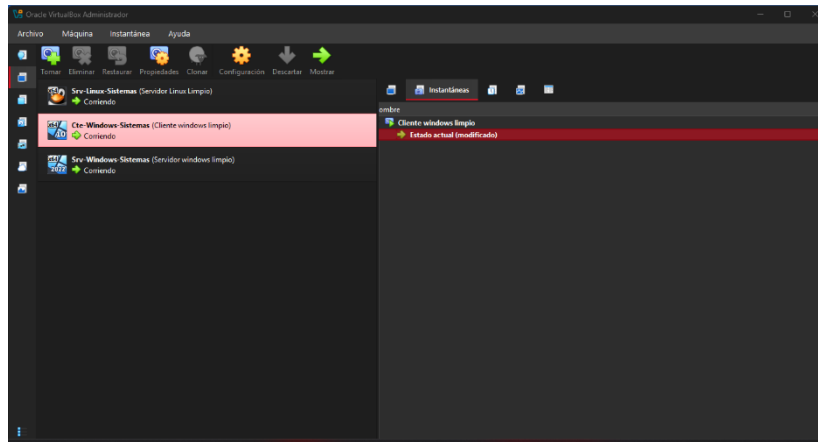
Estas snapshots se tomaron una vez que se instaló el sistema operativo en cada nodo



Nodo #1

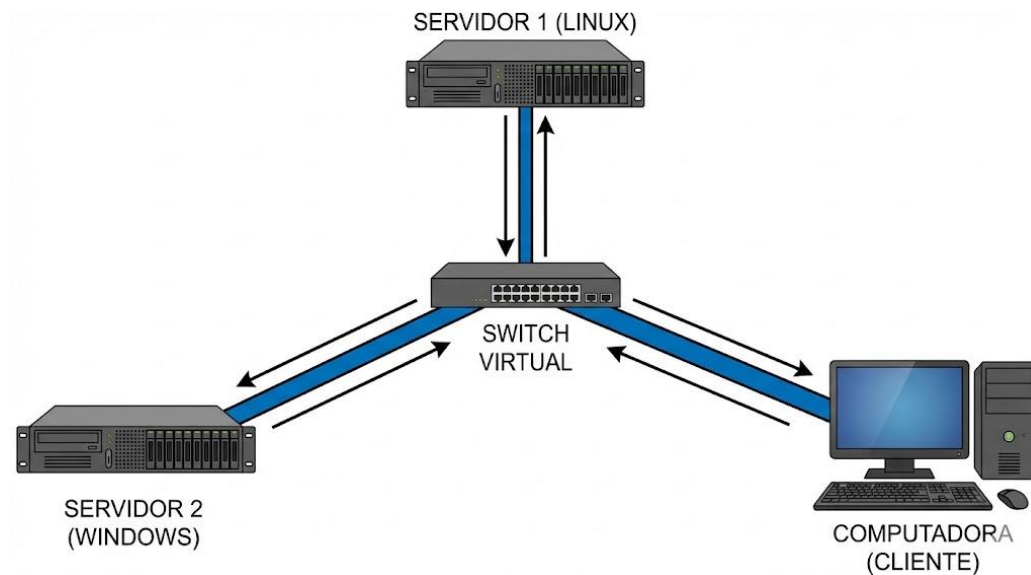


Nodo #2



Nodo #3

## Configuración de la red entre los nodos



La topología de red es de tipo **Estrella**. Los tres nodos (Oracle Linux, Windows Server y Windows 10) están conectados a un **switch virtual** dentro del segmento de red **192.168.10.0/24**, permitiendo comunicación bidireccional directa entre cualquiera de los puntos.

## Tabla de direcciones IP

No. Nodo	Sistema operativo	Nombre del equipo	dirección IP	Mascara de subred
1	Oracle Linux	srv-linux	192.168.10.10	255.255.255.0
2	Windows Server	srv-windows	192.168.10.20	255.255.255.0
3	Windows 10	cte-windows	192.168.10.30	255.255.255.0

Para la configuración de red de Oracle Linux Server, me apoye de la GUI de instalación siguiendo estos pasos:

- Se accedió al panel de **Configuración de Red** (Wired Settings).
- En la pestaña **IPv4**, se cambió el método de "Automático (DHCP)" a **"Manual"**.
- Se definieron los parámetros:
  - **Dirección:** 192.168.10.10
  - **Máscara de Red:** 255.255.255.0 (o prefijo /24).

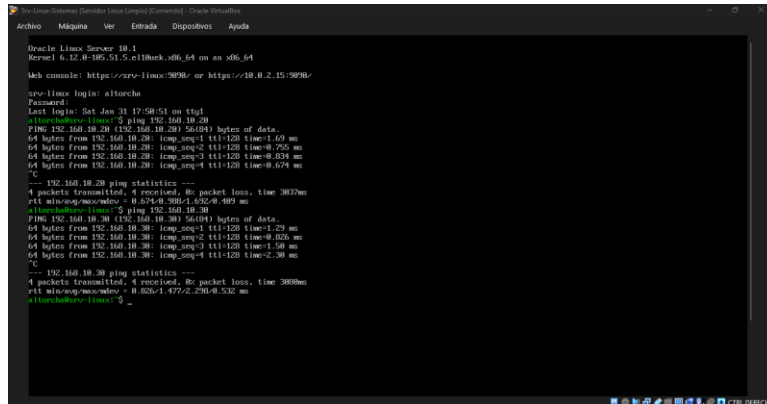
Se reinició la interfaz para aplicar los cambios persistentes.

Para la configuración de red de **Windows Server**, seguí los siguientes pasos:

1. **Ingreso a la herramienta:** Desde **PowerShell**, se ejecutó el comando: **sconfig**
2. **Selección del Menú de Red:** Se seleccionó la Opción 8 "Network Settings".
3. **Identificación del Adaptador:** El sistema listó las tarjetas de red disponibles. Se seleccionó el Índice del adaptador conectado a la Red Interna.
4. **Configuración de la Dirección IP:** Se seleccionó la sub-opción 1 ("Set Network Adapter Address" / "Establecer dirección de adaptador de red"). Se eligió el tipo "S" (Static / Estática).
5. **IP Address:** Se ingresó 192.168.10.20. Subnet Mask: Se ingresó 255.255.255.0
6. **Default Gateway:** Se dejó en blanco ya que es una red aislada.

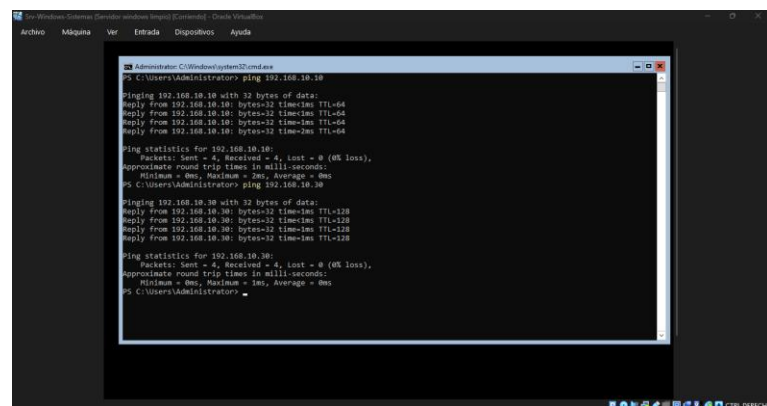
## Ping entre los nodos

Para comprobar que los nodos están en red se utilizó el comando **ping** en cada nodo seguido de la dirección IPv4 de los nodos



```
Oracle Linux Server 8.1
Kernel 6.12.0-105.51.el1.elrepo.x86_64 on an s390x
lib console: https://orv-linux-7090/ or https://10.0.2.15:7090/
orv-linux login: aliorcha
Password:
Last login: Sat Jan 31 17:50:51 on tttyl
PING 192.168.10.20 (192.168.10.20) 56(84) bytes of data:
64 bytes from 192.168.10.20: icmp_seq=1 ttl=120 time=1.63 ms
64 bytes from 192.168.10.20: icmp_seq=2 ttl=120 time=0.725 ms
64 bytes from 192.168.10.20: icmp_seq=3 ttl=120 time=0.634 ms
64 bytes from 192.168.10.20: icmp_seq=4 ttl=120 time=0.674 ms
^C
--- 192.168.10.20 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3037ms
rtt min/avg/max/mdev = 0.675/0.902/1.632/0.409 ms
aliorcha@orv-linux:~$ ping 192.168.10.30
PING 192.168.10.30 (192.168.10.30) 56(84) bytes of data:
64 bytes from 192.168.10.30: icmp_seq=1 ttl=120 time=1.29 ms
64 bytes from 192.168.10.30: icmp_seq=2 ttl=120 time=0.826 ms
64 bytes from 192.168.10.30: icmp_seq=3 ttl=120 time=1.58 ms
64 bytes from 192.168.10.30: icmp_seq=4 ttl=120 time=2.30 ms
^C
--- 192.168.10.30 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3000ms
rtt min/avg/max/mdev = 0.826/1.477/2.30/0.532 ms
aliorcha@orv-linux:~$
```

### Del Nodo #1 a los Nodos #2 y #3



```
PS C:\Users\Administrator> ping 192.168.10.10

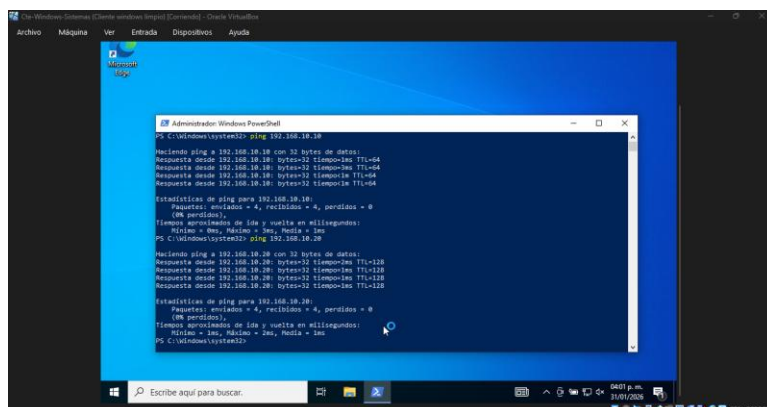
Pinging 192.168.10.10 with 32 bytes of data:
Reply from 192.168.10.10: bytes=32 time=1ms TTL=64
Reply from 192.168.10.10: bytes=32 time=1ms TTL=64
Reply from 192.168.10.10: bytes=32 time=1ms TTL=64
Reply from 192.168.10.10: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.10.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 0ms
PS C:\Users\Administrator> ping 192.168.10.30

Pinging 192.168.10.30 with 32 bytes of data:
Reply from 192.168.10.30: bytes=32 time=1ms TTL=128
Reply from 192.168.10.30: bytes=32 time=1ms TTL=128
Reply from 192.168.10.30: bytes=32 time=1ms TTL=128
Reply from 192.168.10.30: bytes=32 time=1ms TTL=128

Ping statistics for 192.168.10.30:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
PS C:\Users\Administrator>
```

### Ping del Nodo #2 a los Nodos #1 y #3



```
PS C:\Windows\system32> ping 192.168.10.10

Haciendo ping a 192.168.10.10 con 32 bytes de datos:
Respuesta desde 192.168.10.10: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.10.10: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.10.10: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.10.10: bytes=32 tiempo=1ms TTL=64

Estadísticas de ping para 192.168.10.10:
    Paquetes enviados = 4, recibidos = 4, perdidos = 0
    (0% pérdidas),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 1ms, Medio = 1ms
PS C:\Windows\system32> ping 192.168.10.20

Haciendo ping a 192.168.10.20 con 32 bytes de datos:
Respuesta desde 192.168.10.20: bytes=32 tiempo=1ms TTL=128
Respuesta desde 192.168.10.20: bytes=32 tiempo=1ms TTL=128
Respuesta desde 192.168.10.20: bytes=32 tiempo=1ms TTL=128
Respuesta desde 192.168.10.20: bytes=32 tiempo=1ms TTL=128

Estadísticas de ping para 192.168.10.20:
    Paquetes enviados = 4, recibidos = 4, perdidos = 0
    (0% pérdidas),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 1ms, Medio = 1ms
PS C:\Windows\system32>
```

### Ping del Nodo #3 a los Nodos #1 y #2

## Script de diagnóstico para Linux (check\_status.sh)

```
#!/bin/bash
# Script de Diagnostico Basico para linux
# Proposito: Validar identidad, version y almacenamiento del nodo

echo "=====
echo "      DIAGNOSTICO DE LINUX"
echo "=====

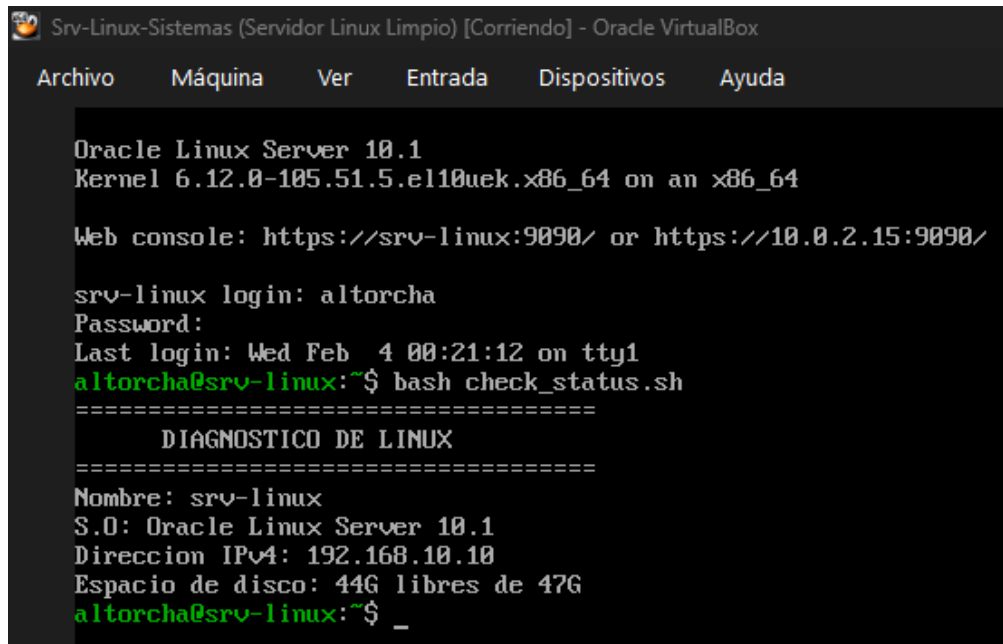
# 1. Identificacion del Nodo
# Muestra el nombre del equipo para confirmar en que Nodo estamos
echo "Nombre: $(hostname)"

# 2. Version del sistema operativo
# Busca la linea "PRETTY_NAME" en la info del sistema
echo "S.O: $(grep PRETTY_NAME /etc/os-release | cut -d '"' -f2)"

# 3. Direccion IP (Red interna)
# host name -I de todas las IPs. Usamos awk '{print $2}' para tomar la segunda IP
# ya que la primera suele ser la NAT y la segunda es la red interna (en este caso)
echo "Direccion IPv4: $(hostname -I | awk '{print $2}')"

# 4. Revisa la particion Raiz, salta el encabezado y muestra solo el espacio libre y el total
echo "Espacio de disco: $(df -h / | awk 'NR==2 {print $4 " libres de " $2}')"

```



The screenshot shows a terminal window titled "Srv-Linux-Sistemas (Servidor Linux Limpio) [Corriendo] - Oracle VirtualBox". The terminal output displays the results of the `check_status.sh` script. It shows the system is Oracle Linux Server 10.1 with kernel 6.12.0-105.51.5.el10uek.x86\_64. The user `altorcha` is logged in. The script output is as follows:

```
Oracle Linux Server 10.1
Kernel 6.12.0-105.51.5.el10uek.x86_64 on an x86_64

Web console: https://srv-linux:9090/ or https://10.0.2.15:9090/

srv-linux login: altorcha
Password:
Last login: Wed Feb  4 00:21:12 on tty1
altorcha@srv-linux:~$ bash check_status.sh
=====
      DIAGNOSTICO DE LINUX
=====
Nombre: srv-linux
S.O: Oracle Linux Server 10.1
Direccion IPv4: 192.168.10.10
Espacio de disco: 44G libres de 47G
altorcha@srv-linux:~$ _
```

## Script de diagnóstico para Windows (check\_status.ps1)

```
# Script de Diagnostico basico para windows
# Proposito: Validar identidad, version y almacenamiento del nodo

Write-Host "=====
DIAGNOSTICO DE WINDOWS
=====

# 1. Identificacion del Nodo
# Usa la version de entorno del sistema para mostrar el Hostname
Write-Host "Nombre: $env:COMPUTERNAME"

# 2. Version del Sistema Operativo
# Utiliza CIM para extraer el nombre comercial de la version de Windows

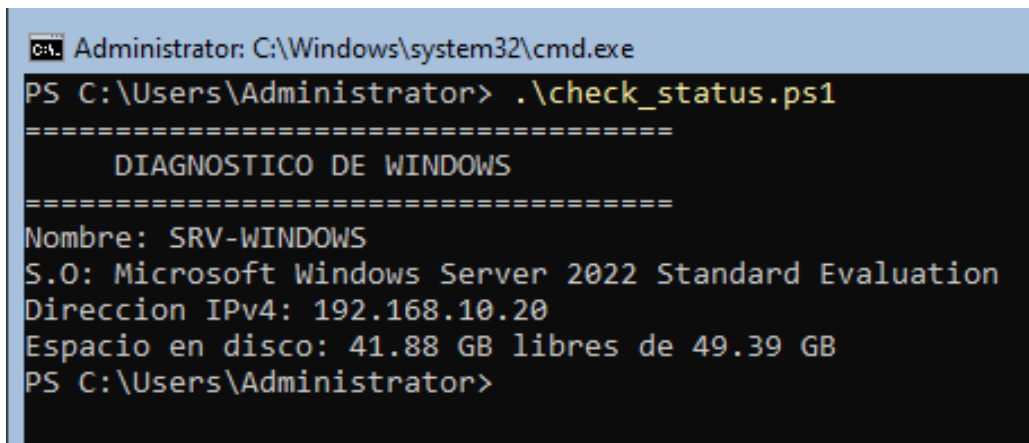
$os = (Get-CimInstance Win32_OperatingSystem).Caption
Write-Host "S.O: $os"

# 3. Direccion IP (filtrada)
# Busca todas las direcciones IPv4 pero filtra solo la que empieza con "192."
# Para asegurar que mostramos la IP de la red interna y no otras interfaces.

$ip = Get-NetIPAddress -AddressFamily IPv4 | Where-Object {$_.IPAddress -like "192.*"} |
Select-Object -ExpandProperty IPAddress
Write-Host "Direccion IPv4: $ip"

# 4. Calculo de Almacenamiento
# Obtiene el objeto del disco C y realiza una operacion matematica para
# convertir los bytes a Gigabytes redondeando a 2 decimales

$disk = Get-Volume -DriveLetter C
$free = [math]::Round($disk.SizeRemaining / 1GB, 2)
$total = [math]::Round($disk.Size / 1GB, 2)
Write-Host "Espacio en disco: $free GB libres de $total GB"
```



```
C:\Windows\system32\cmd.exe
PS C:\Users\Administrator> .\check_status.ps1
=====
DIAGNOSTICO DE WINDOWS
=====
Nombre: SRV-WINDOWS
S.O: Microsoft Windows Server 2022 Standard Evaluation
Direccion IPv4: 192.168.10.20
Espacio en disco: 41.88 GB libres de 49.39 GB
PS C:\Users\Administrator>
```



## Conclusiones

La realización de esta tarea de la materia de administración de sistemas me dejó una experiencia fundamental para comprender la interoperabilidad entre plataformas Linux y Windows, permitiendo no solo la automatización de diagnósticos mediante *scripting* en Bash y PowerShell, sino también la validación de competencias críticas en entornos de servidores. Un aspecto distintivo de la práctica fue la operación en infraestructuras carentes de interfaz gráfica lo cual eliminó la dependencia de herramientas visuales de escritorio y exigió un dominio estricto de la línea de comandos para la gestión de archivos y configuraciones, replicando las condiciones de optimización de recursos exigidas en los centros de datos reales. Durante el desarrollo, fue necesario investigar y aplicar soluciones a bloqueos de seguridad nativos, como la restricción de ejecución de scripts en Windows que requirió la modificación de políticas, así como la gestión avanzada del control de versiones en Git, implementando técnicas de *squash* para mantener un historial de desarrollo profesional.

## Bibliografía y Referencias

- **Oracle Corporation.** (2024). *Oracle Linux 9 System Administration Guide*. Recuperado de la documentación oficial de Oracle.
- **Microsoft Corporation.** (2024). *about\_Execution\_Policies - PowerShell*. Microsoft Learn. Recuperado de [learn.microsoft.com](https://learn.microsoft.com).
- **Git SCM.** (2024). *Git Tools - Rewriting History*. Recuperado de [git-scm.com/book](https://git-scm.com/book).
- **OpenAI & Google DeepMind.** (2026). *Asistencia técnica mediante Inteligencia Artificial Generativa*. (Modelos utilizados para la consulta de sintaxis Bash/PowerShell y depuración de errores de Git).