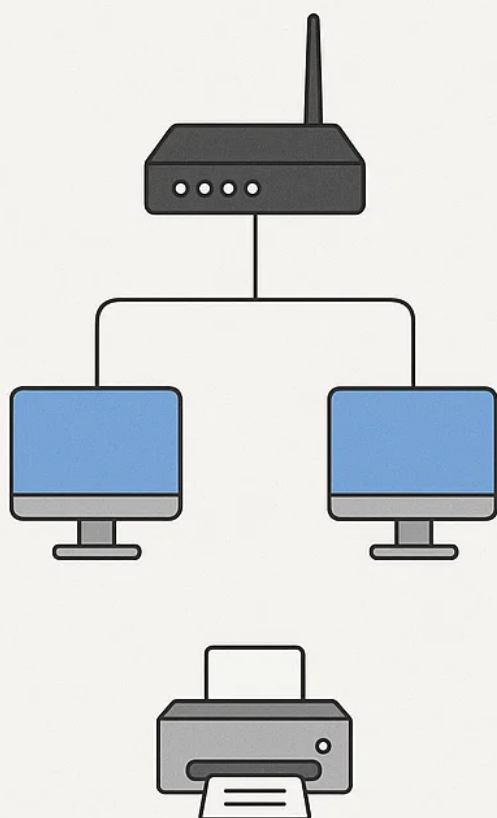
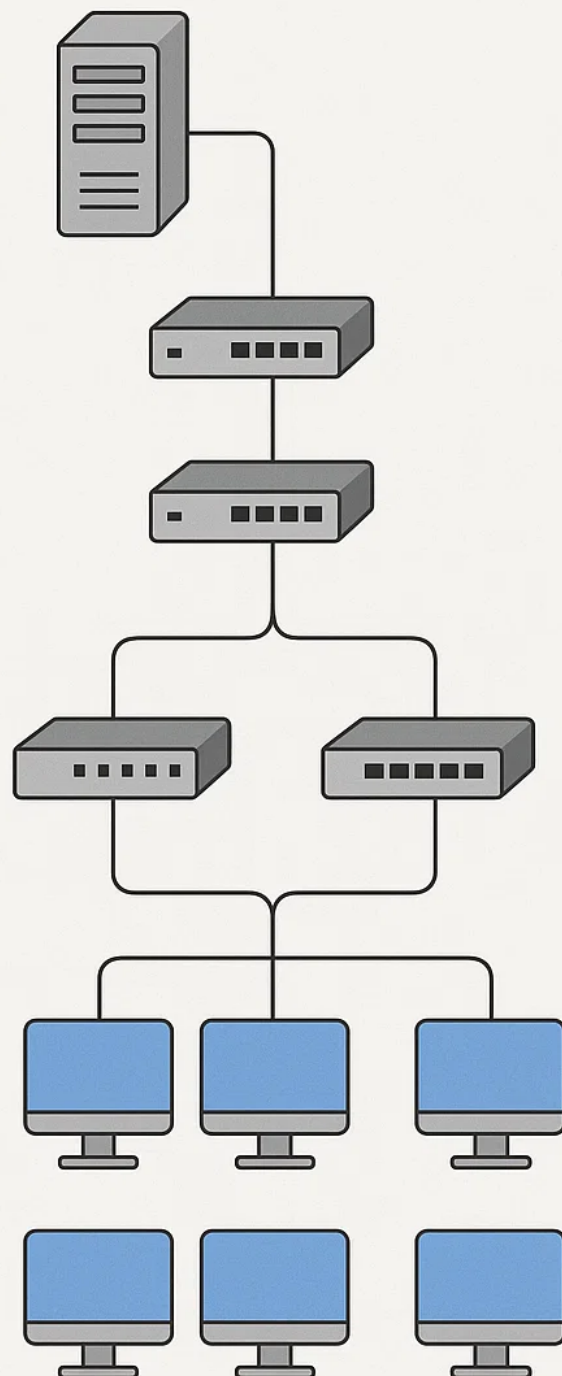


Serie: Creando tu Propia Red Local SOHO

Red pequeña



Red empresarial



 **Objetivo:** Aprender a configurar una red desde la simulación en **Packet Tracer** hasta su implementación en **Windows** y **Linux**.

Episodio 1: Introducción y Preparación

- ◆ **Contenido:**

- ✓ ¿Qué es una red SOHO y para qué se usa?
- ✓ Diferencias entre **red SOHO** y **red empresarial**.
- ✓ Explicación del **escenario de la tienda** (PCs, servidor, switch, router, impresora).
- ✓ Instalación y configuración básica de **Packet Tracer**.

 "Descarga Packet Tracer y prepárate para la práctica en el siguiente episodio."

Episodio 2: Creando la Red en Packet Tracer

- ◆ **Duración:** 7-10 min

- ◆ **Contenido:**

- ✓ Agregar y conectar dispositivos (PCs, switch, router, servidor, impresora).
- ✓ Asignación de direcciones IP.
- ✓ Configuración básica del **router** para acceso a Internet.
- ✓ Pruebas de conectividad (ping).

 "Prueba tu configuración y cuéntanos si lograste conectar todos los dispositivos."

Episodio 3: Configuración de Servidor y Compartición de Recursos en windows

- ◆ **Duración:** 7-10 min

- ◆ **Contenido:**

- ✓ Creación de un **servidor de base de datos** en Packet Tracer.
- ✓ Simulación de acceso desde las PCs al servidor.
- ✓ Configuración de una **impresora en red**.
- ✓ Prueba de impresión desde las PCs.

 "Intenta imprimir un documento y verifica si tu red funciona correctamente."

Episodio 4: Configuración de Servidor y Compartición de Recursos en Linux

- ♦ **Duración:** 7-10 min

- ♦ **Contenido:**

- ✓ Configurar IP estática en Linux (/etc/network/interfaces).
- ✓ Instalación de **MySQL Server** en Linux.
- ✓ Configuración de **CUPS** para impresora en red.
- ✓ Verificación de conectividad (ping, netstat).

 "Si completaste todos los videos, felicidades. Ahora intenta implementar esto en un entorno real."

Extras o Desafíos Adicionales (Opcionales)

- 🔥 **Episodio Extra:** Configuración de una VPN para acceso remoto.
 - 🔥 **Episodio Extra:** Seguridad en redes SOHO (Firewall, NAT, puertos).
-

Episodio 1: Introducción y Preparación

 **Objetivo:**

Aprender los fundamentos de una red SOHO (Small Office/Home Office), su utilidad y diferencias con redes empresariales. También prepararemos el entorno de trabajo instalando **Packet Tracer**, una herramienta de simulación de redes.

1. ¿Qué es una Red SOHO y para qué se usa?

Una red SOHO es una **red de área local (LAN)** diseñada para **pequeñas empresas u oficinas en casa**. Su propósito es conectar dispositivos como computadoras, impresoras y servidores, facilitando el **compartir archivos, acceso a internet y la comunicación interna**.

 **Ejemplos de usos comunes en negocios pequeños:**

- ✓ Conectar computadoras de empleados para compartir documentos.
 - ✓ Instalar una impresora en red para múltiples usuarios.
 - ✓ Configurar un servidor de base de datos para gestionar inventarios.
 - ✓ Establecer acceso remoto mediante VPN para trabajar desde casa.
-

2. Diferencias entre una Red SOHO y una Red Empresarial

Característica	Red SOHO	Red Empresarial
Tamaño	Pequeña (5-20 dispositivos)	Grande (cientos o miles de dispositivos)
Infraestructura	Equipos básicos (switch, router, impresora)	Servidores dedicados, routers avanzados, firewalls
Seguridad	Básica (contraseñas, NAT, firewall básico)	Avanzada (firewalls, VLANs, IDS/IPS, VPNs)
Costos	Bajos	Altos (equipos especializados y soporte IT)

📌 **Conclusión:** Una red SOHO es más sencilla y económica, ideal para negocios pequeños. En cambio, una red empresarial requiere mayor inversión y administración profesional.

3. Escenario del Proyecto: Tienda de Instrumentos Musicales

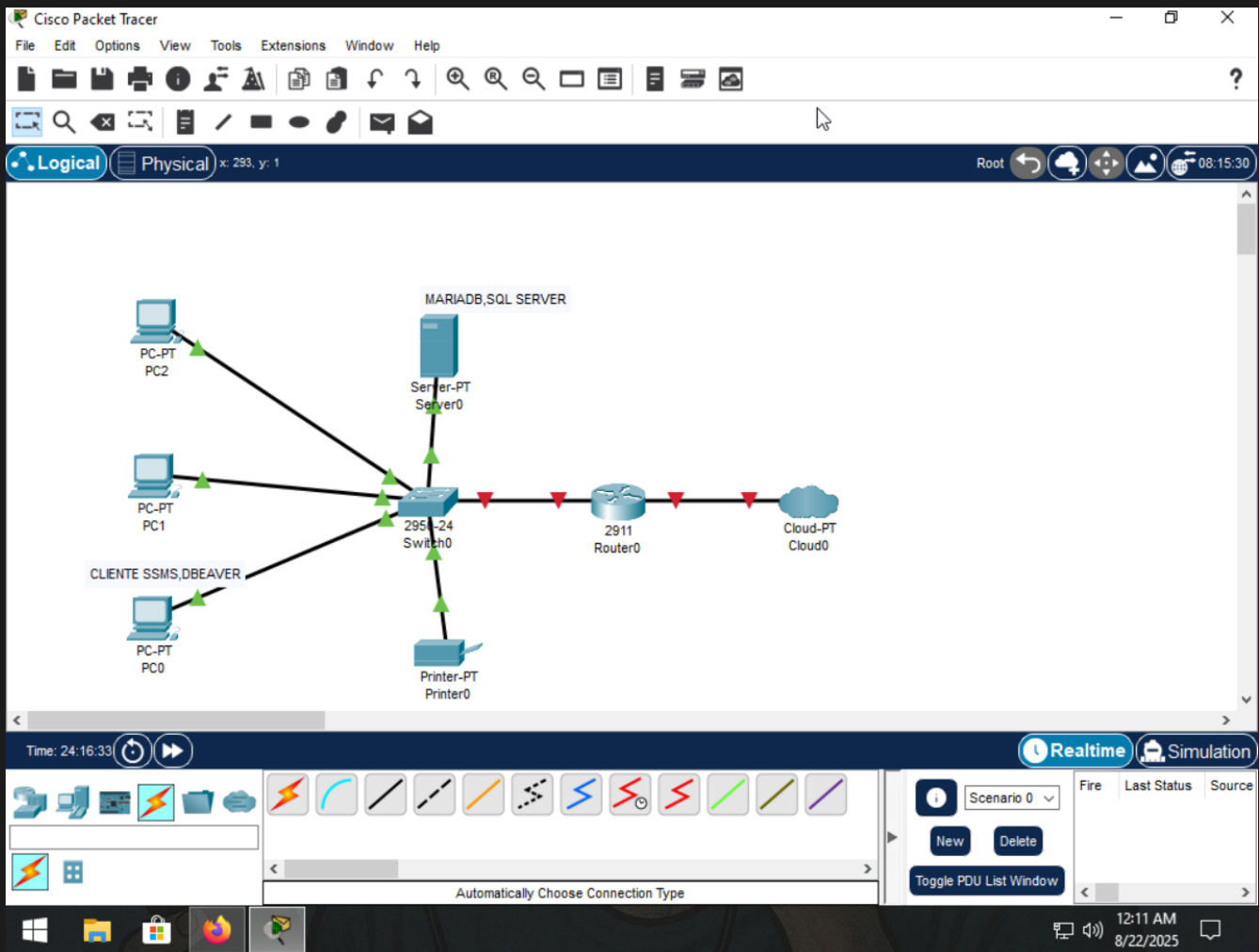
♦ Contexto:

Imagina que tienes una tienda de instrumentos musicales y necesitas configurar una red SOHO para conectar las computadoras de ventas con un servidor central, asegurando que todos puedan acceder a la base de datos de inventario y usar una impresora compartida.

♦ Dispositivos necesarios:

- ✓ 3 computadoras (para ventas, administración y caja).
- ✓ 1 servidor (almacenamiento de inventario y base de datos).
- ✓ 1 switch (para interconectar los dispositivos).
- ✓ 1 router (para proporcionar acceso a internet).
- ✓ 1 impresora en red (para generar facturas y documentos).

📌 **Objetivo del curso:** Simular esta red en Packet Tracer y luego replicarla en Windows y Linux.



4. Instalación y Configuración Básica de Packet Tracer

¿Qué es Packet Tracer? Packet Tracer es un software de simulación de redes desarrollado por Cisco que permite diseñar, probar y solucionar problemas en redes sin necesidad de hardware físico.

◆ Pasos para instalar Packet Tracer:

1. Descargar Packet Tracer:

- Ir al sitio oficial de Cisco: <https://www.netacad.com/>
- Registrarse en **Cisco Networking Academy** (gratuito).
- Descargar la versión más reciente según tu sistema operativo (Windows, Mac o Linux).

2. Instalar Packet Tracer:

- Ejecutar el instalador y seguir las instrucciones.

- Al abrir el programa, iniciar sesión con tu cuenta de **NetAcad(SKILL FOR ALL)**.

3. Configuración inicial:

- Familiarízate con la interfaz.
- Explora los diferentes dispositivos disponibles en la barra lateral.

-
- ✓ Descarga e instala **Packet Tracer**.
 - ✓ Familiarízate con la interfaz y las herramientas disponibles.
 - ✓ Prepara tu entorno para el siguiente episodio, donde crearemos la red en **Packet Tracer**.

Episodio 2: Creando la Red en Packet Tracer

Objetivo:

En este episodio, aprenderás a **crear tu propia red en Packet Tracer**, agregando dispositivos, asignando direcciones IP, configurando un router y realizando pruebas de conectividad. Este episodio te ayudará a poner en práctica todo lo aprendido en el primer episodio, llevando tu red a la vida.

1. Agregar y Conectar Dispositivos

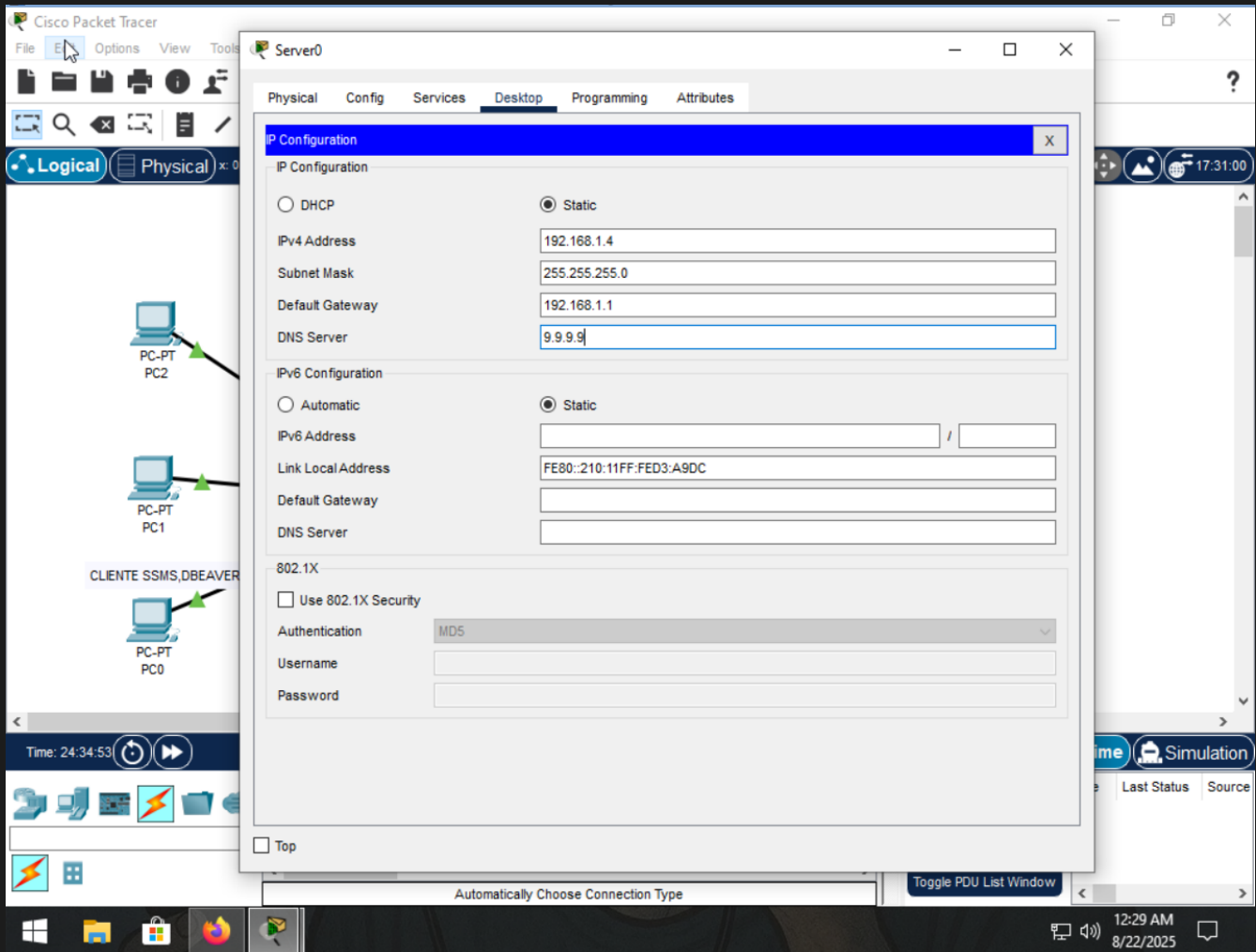
Comenzamos agregando los dispositivos esenciales para nuestra red:

- **PCs** para los usuarios.
- Un **switch** para conectar los PCs.
- Un **router** para gestionar el tráfico de la red.
- Un **servidor** para almacenar y gestionar los datos.
- Una **impresora** para que los usuarios puedan compartirla en la red.

2. Asignación de Direcciones IP con interfaz o línea de comandos.

Una vez conectados los dispositivos, es hora de asignar direcciones IP estáticas para que puedan comunicarse entre sí. Asignaremos direcciones IP de la siguiente manera:

- **PC1:** 192.168.1.2
- **PC2:** 192.168.1.3
- **Servidor:** 192.168.1.4
- **Router:** 192.168.1.1 en la interfaz de red interna.

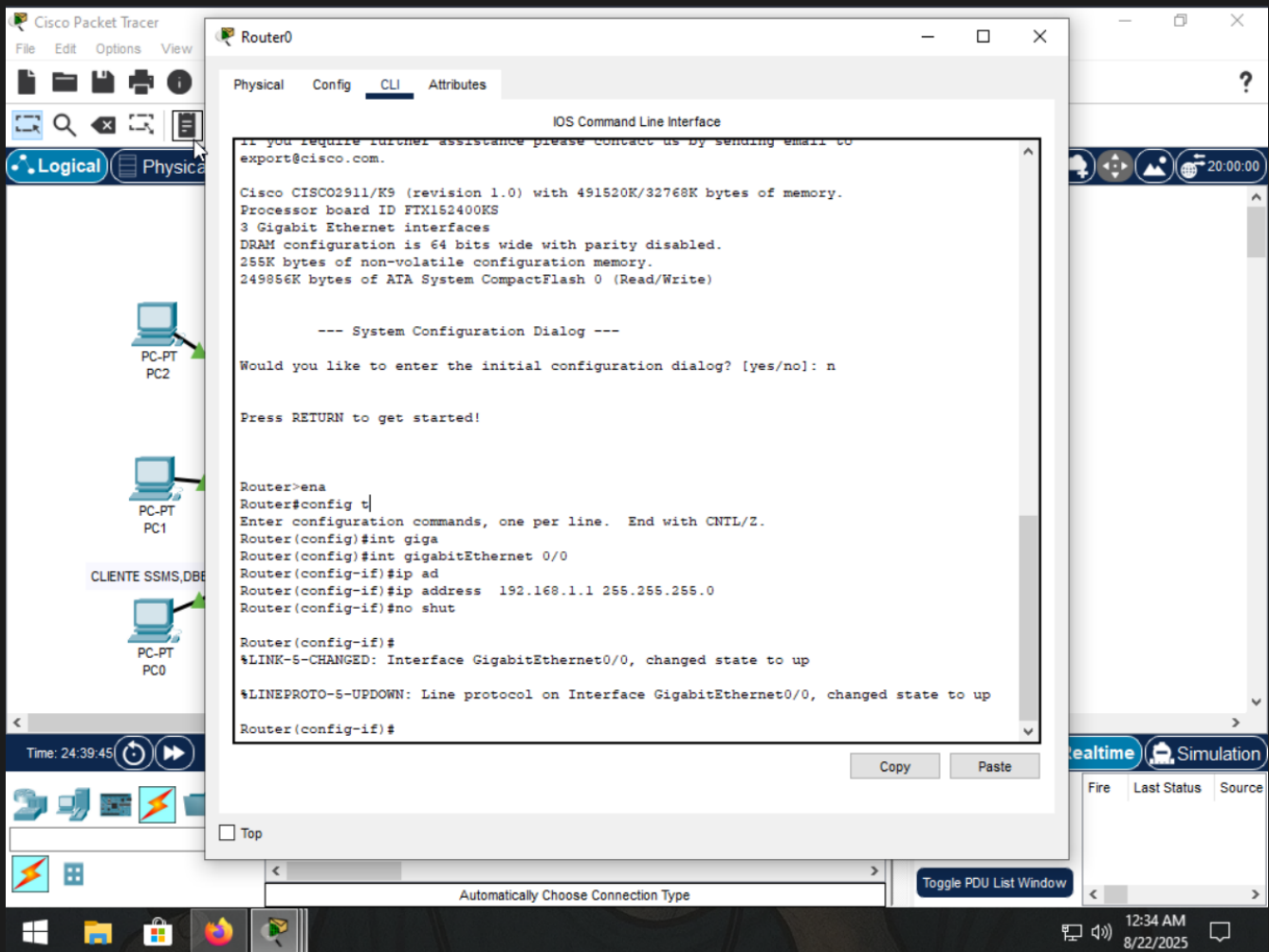


3. Configuración Básica del Router

A continuación, configuramos el **router** para gestionar el tráfico de la red. Usaremos los siguientes comandos en la interfaz del router para configurar la dirección IP de la interfaz interna del router y habilitarla para permitir la comunicación.

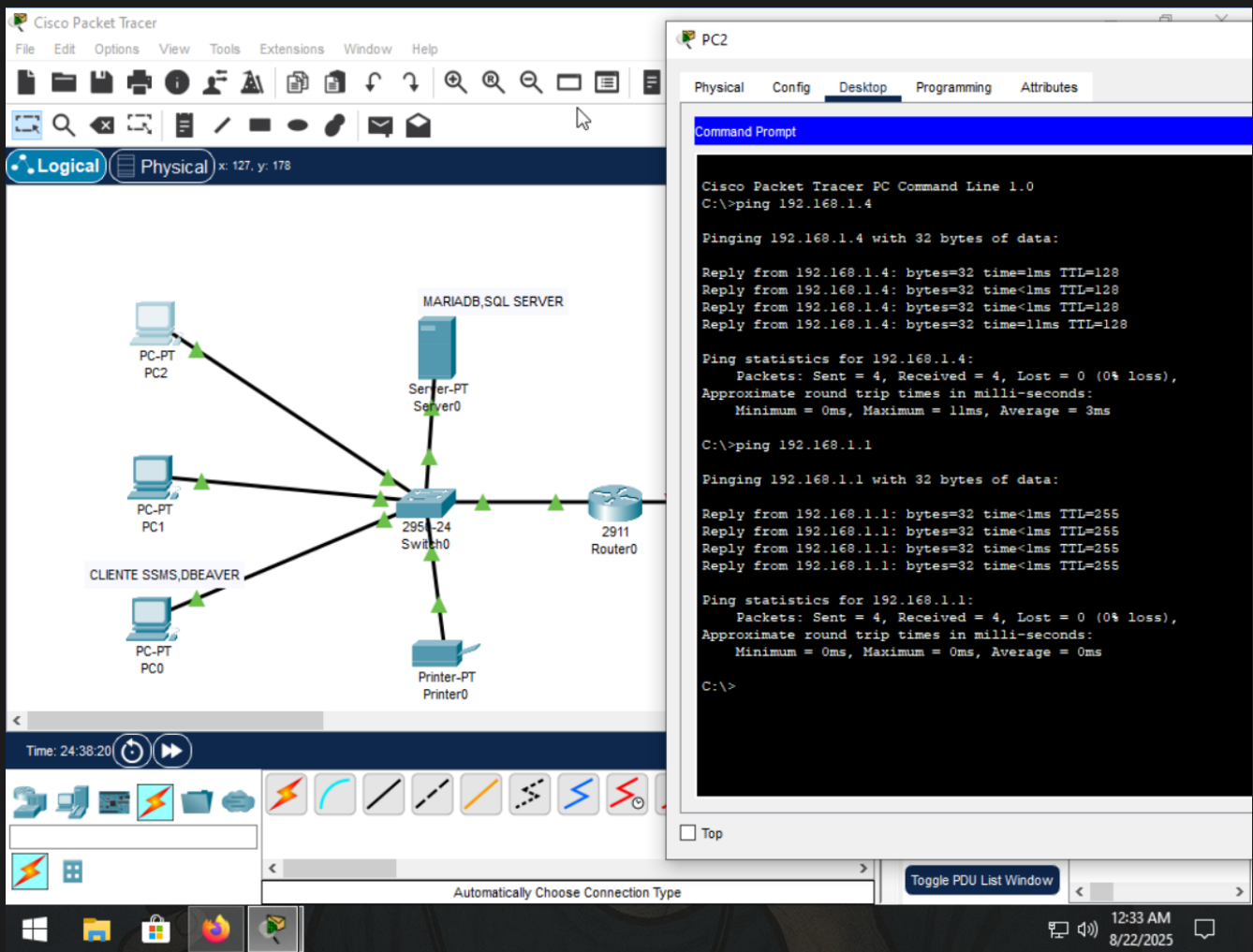
Comandos:

```
Router>ena
Router#config t
Router(config)#int gigabitEthernet 0/0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shut
```



4. Pruebas de Conectividad (Ping)

Finalmente, realizamos pruebas de conectividad entre los dispositivos utilizando el comando **ping**. Esto nos permitirá verificar que todos los dispositivos estén correctamente conectados y puedan comunicarse entre sí.



✓ Prueba tu configuración en Packet Tracer y asegúrate de que todos los dispositivos puedan comunicarse.

✓ Cuéntanos si lograste conectar todos los dispositivos correctamente. ¡Estamos aquí para ayudarte!

Paso 3: Configuración básica del router para acceso a Internet

🎯 Objetivo:

En este paso vamos a configurar el router no solo para la red local, sino también para que los dispositivos puedan **acceder a Internet** mediante **NAT (Network Address Translation)**. Esto permitirá que los dispositivos internos utilicen una sola dirección pública para comunicarse con el exterior.

⚙️ 1. Configuración de la Interfaz Interna del Router

Primero, asignamos la IP del **gateway** en la interfaz `GigabitEthernet0/0`, que conectará a la red interna.

****Comandos en la CLI del Router:****

```
Router> enable
Router# configure terminal
Router(config)# interface gigabitEthernet 0/0
Router(config-if)# ip address 192.168.1.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# exit
```

Esta IP será la ****puerta de enlace predeterminada**** para todos los dispositivos de la red local.

2. Configuración de la Interfaz Externa (Simulando el Acceso a Internet)

 Puedes conectar esta interfaz a otro router que simule el proveedor de Internet, o directamente a un "cloud" si estás usando ese recurso en Packet Tracer.

3. Configuración de NAT (Network Address Translation)

NAT permite que múltiples dispositivos de la red interna compartan una única IP pública para salir a Internet.

****Pasos:****

```
! =====
! CONFIGURACIÓN DE NAT PARA UNA PYME
! Objetivo: Permitir que todos los equipos internos (192.168.1.0/24)
! compartan una sola IP pública para acceder a Internet.
! =====
```

Router(config)# access-list 1 permit 192.168.1.0 0.0.0.255

! NOTA: se crea la lista 1 de acceso

Aquí "0.0.0.255" es el comodín (wildcard mask)

! y representa exactamente la misma red que 192.168.1.0 con máscara 255.255.255.0

! Es decir: todos los dispositivos desde 192.168.1.1 hasta 192.168.1.254

! En Cisco, en listas de acceso (ACL), se usa el comodín, no la máscara normal.

! Pero el resultado es el mismo: ¡toda la red Clase C interna de la PYME!

Router(config)# ip nat inside source list 1 interface gigabitEthernet 0/1 overload

! CONFIGURA el NAT dinámico con sobrecarga (PAT).

! CONFIGURA el PAT (Port Address Translation), también conocido como "NAT sobrecargado".

! Traduce todas las IPs de la lista 1 a la IP pública del interfaz gig0/1.

! "overload" = muchos dispositivos comparten una sola IP pública.

! Esto es lo más común en PYMES, porque el ISP solo da 1 IP pública.

Router(config)# interface gigabitEthernet 0/0

! ENTRA en la configuración del interfaz gigabitEthernet 0/0

! Este interfaz está conectado a la RED INTERNA (switch, servidores, PCs, etc.).

Router(config-if)# ip nat inside

! MARCA este interfaz como "interior" (inside).

! Todo tráfico que entre por aquí viene de la red local y debe traducirse con NAT.

Router(config-if)# exit

! SALE de la configuración del interfaz.

Router(config)# interface gigabitEthernet 0/1

! ENTRA en la configuración del interfaz gigabitEthernet 0/1

! Este interfaz está conectado al MODEM o al ISP (Internet).

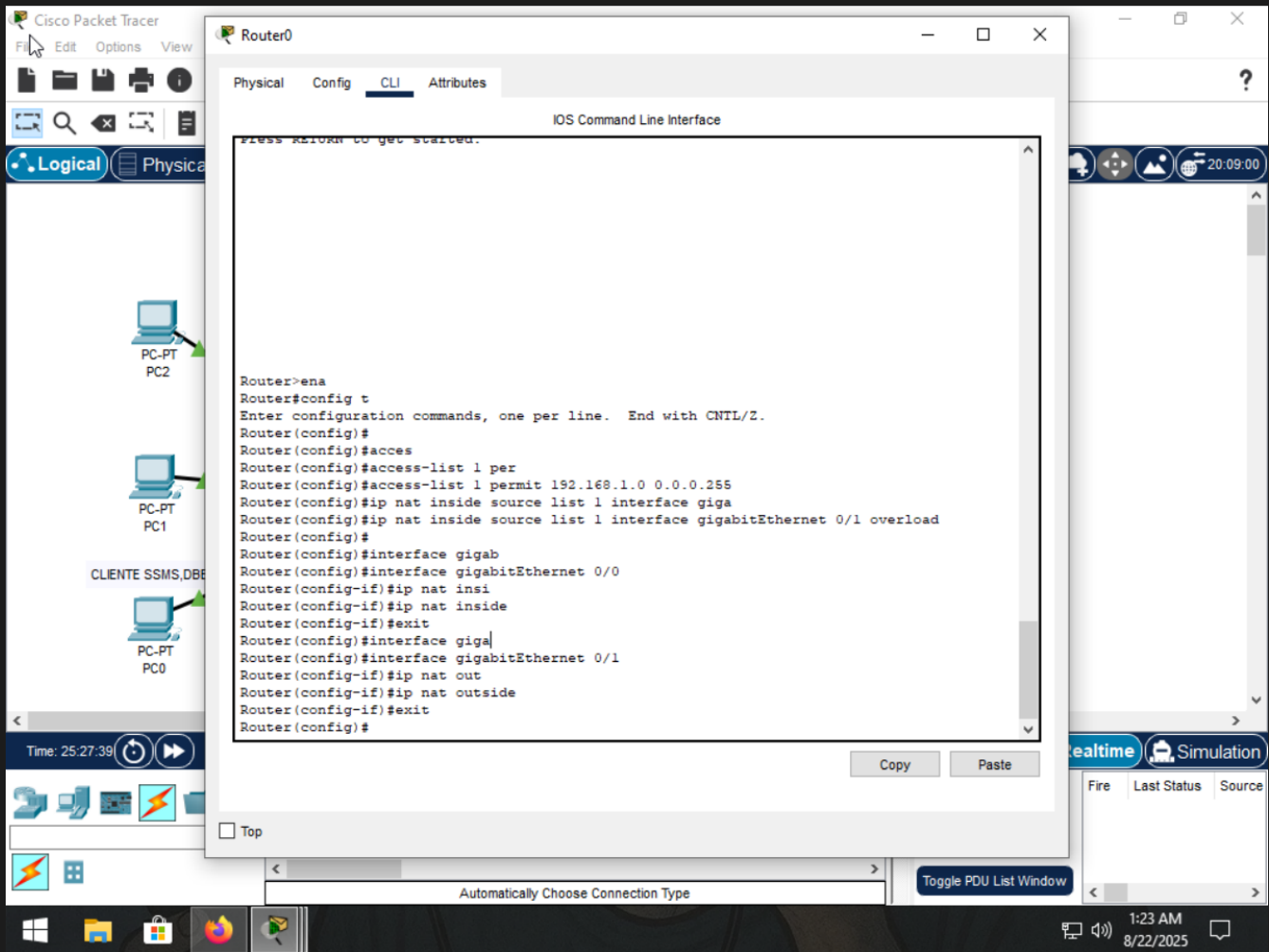
Router(config-if)# ip nat outside

! MARCA este interfaz como "exterior" (outside).

! Aquí sale el tráfico ya traducido, con la IP pública hacia Internet.

Router(config-if)# exit

! SALE de la configuración del interfaz.



!

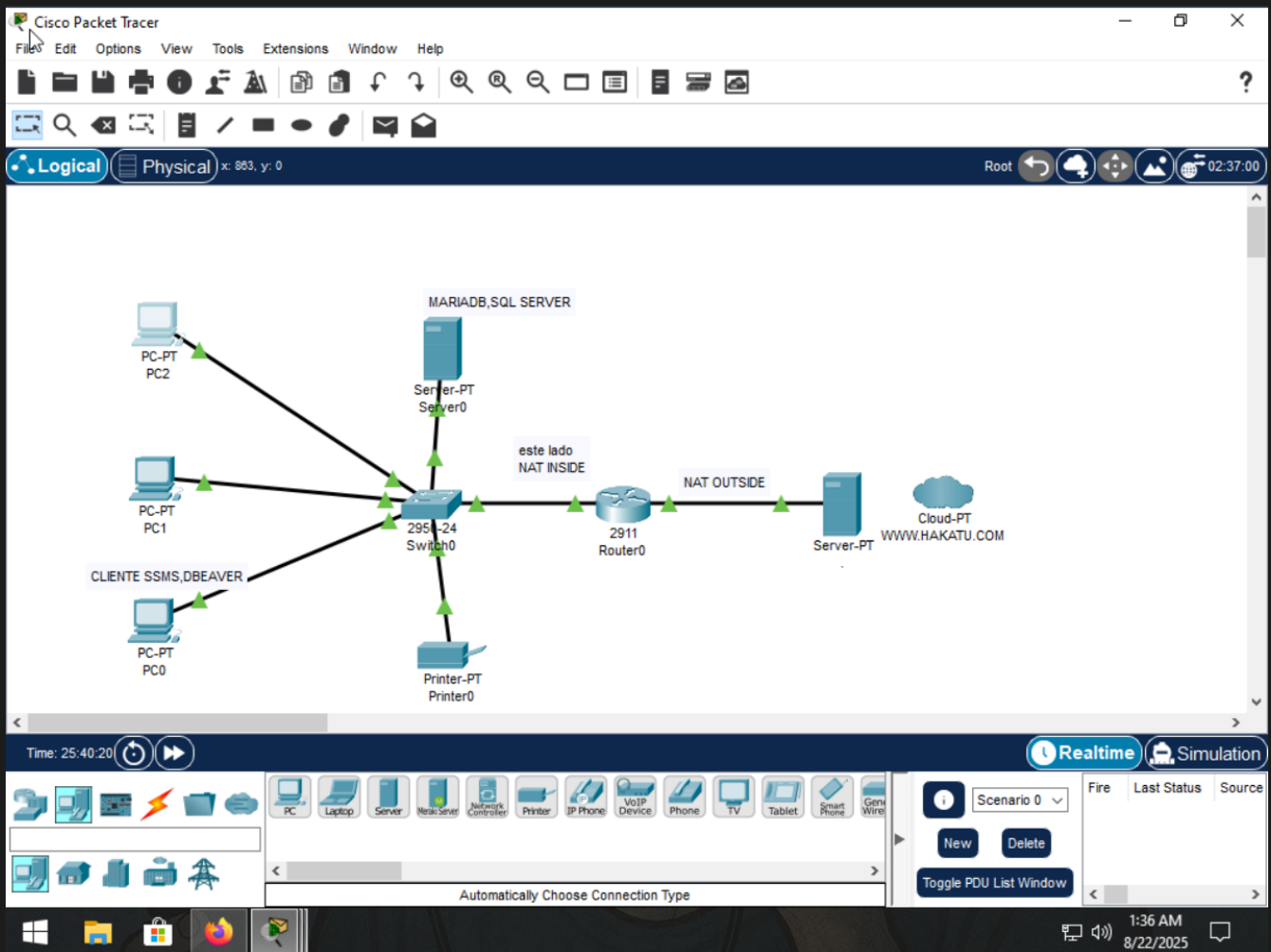
!  LISTO: El NAT está configurado.

! Ahora todos los equipos de la red 192.168.1.0 pueden navegar, imprimir, usar servidores, y acceder a Internet de forma segura y eficiente, usando una sola IP pública.

 Con esto, el router está listo para traducir las direcciones internas a la externa y permitir el tráfico hacia Internet.

4. Verificación de Conectividad a Internet

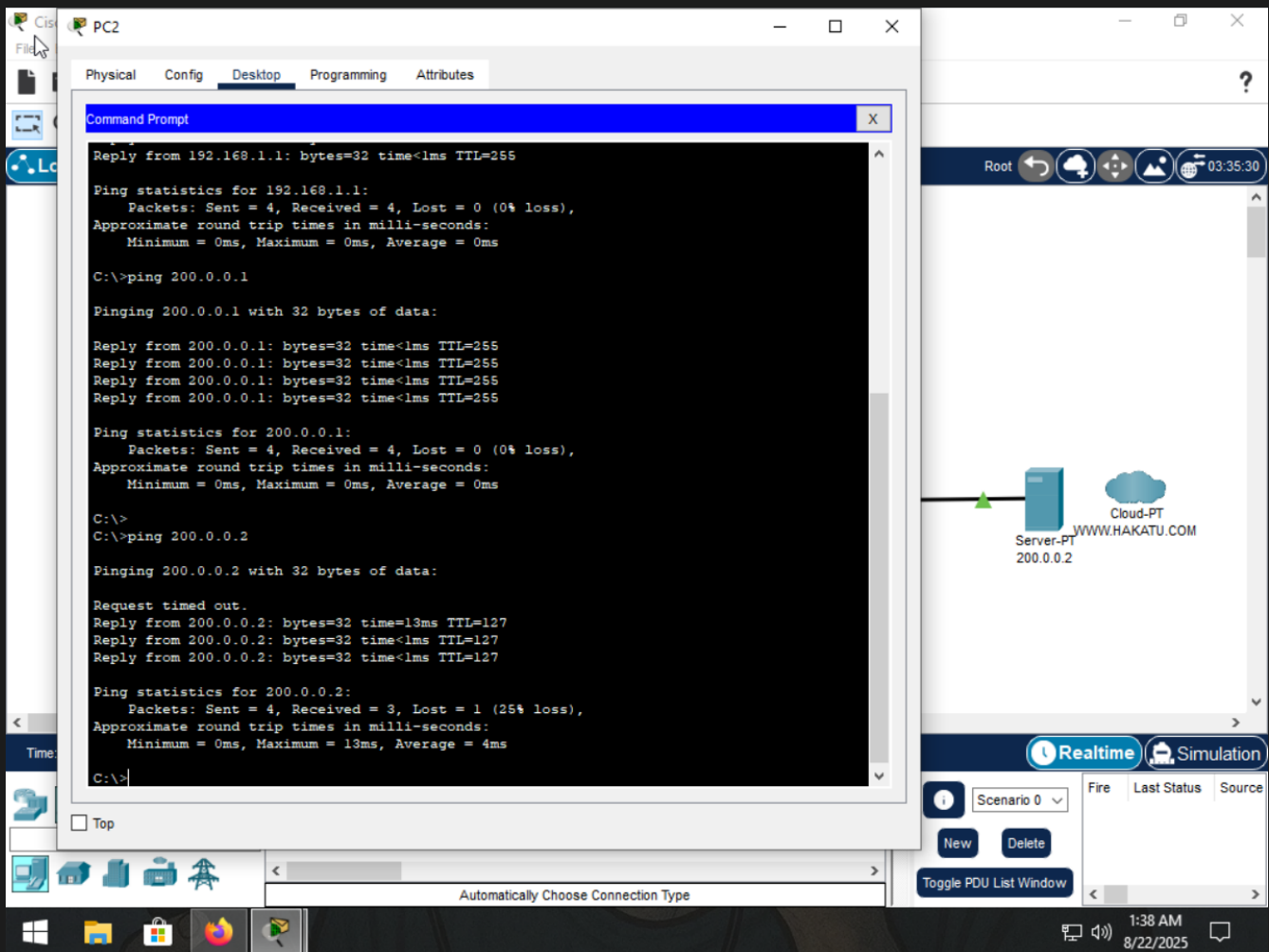
Pon un servidor conectado a la interfaz gigabitEthernet 0/1 con la ip 200.0.0.1 para el gateway y 200.0.0.2 para el servidor esto con el fin de simular la nube(opcional agregar un dominio en el server)



haz una prueba de ping desde un PC hacia una IP pública simulada (por ejemplo, 200.0.0.1 en la red del proveedor).

Ping 200.0.0.2

- ✓ Si el ping responde, ¡tienes acceso simulado a Internet en tu red!
- ✗ Si no, revisa la configuración de NAT y las conexiones entre routers.



✓ ¿Todo Listo?

📡 Verifica que los dispositivos puedan salir a Internet a través del router.

¿Tu red está funcionando correctamente?

¿Tienes dudas con la NAT o la conexión al "ISP"?

💬 Escribenos o deja tu comentario. ¡Te ayudamos a resolverlo!

📺 Episodio 3: Configuración de Servidor y Compartición de Recursos

🎯 Objetivo

En este episodio aprenderás a conectar tu red de Packet Tracer con un servidor SQL real alojado en una máquina virtual, para que las PCs puedan acceder a datos y compartir recursos como en un entorno empresarial.

1. Introducción

En los episodios anteriores:

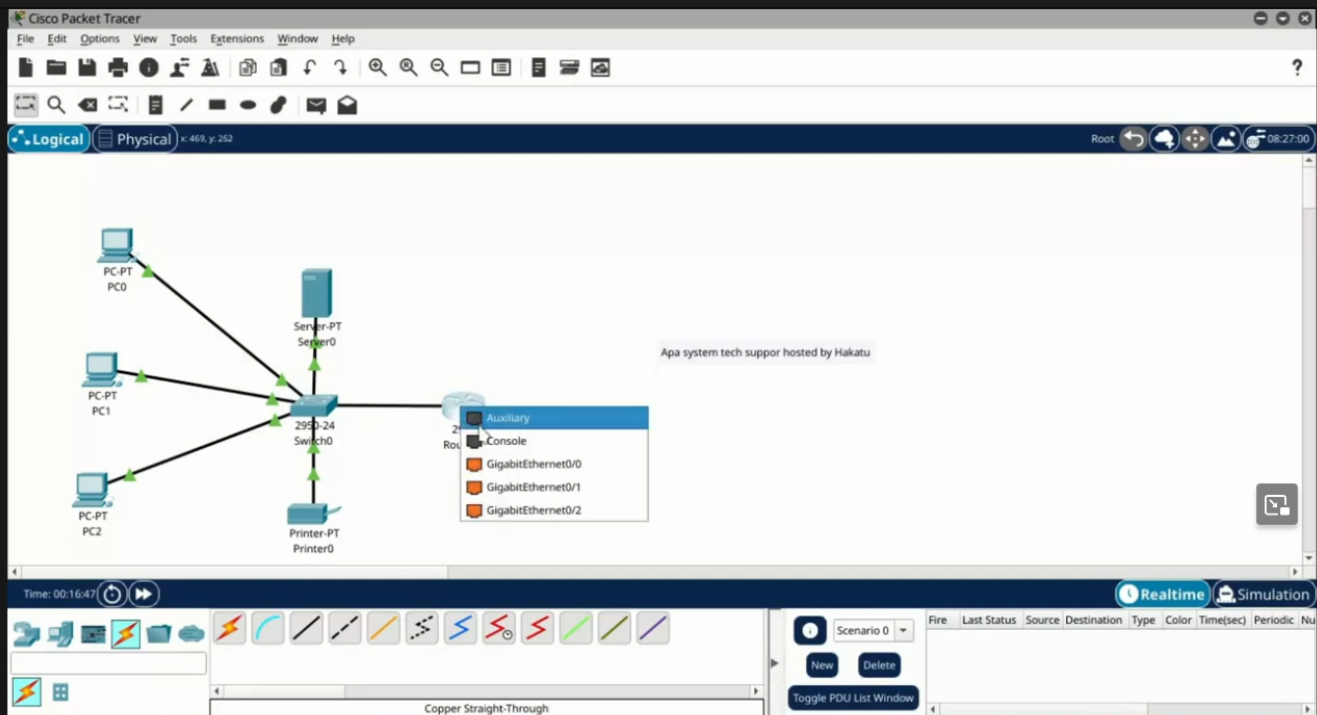
- Montamos la topología básica en Packet Tracer.
- Asignamos IPs y configuramos el router.
- Preparamos un servidor simulado, impresoras y PCs en la red.

Ahora vamos a dar el salto al mundo real conectando la red simulada a un servidor SQL en máquina virtual, para que las PCs clientes puedan acceder a datos reales.

2. Preparar el Entorno

Requisitos previos:

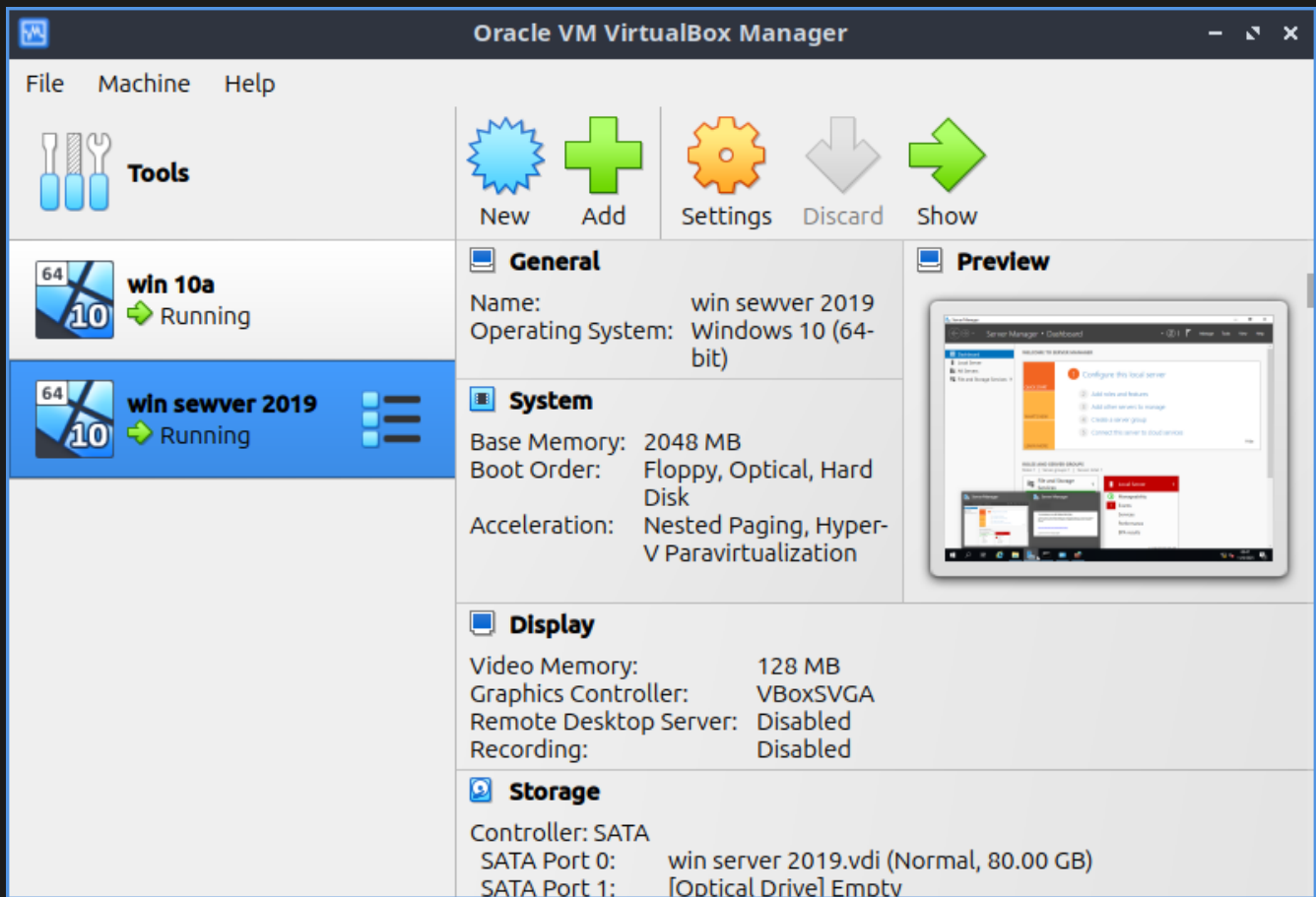
- Topología de Packet Tracer terminada con servidor, PCs y router.



- Software de virtualización (VirtualBox, Qemu, VMware o Hyper-V).
- ISO de Windows Server 2019 o Windows 10.

3. Instalación del Servidor SQL en la Máquina Virtual

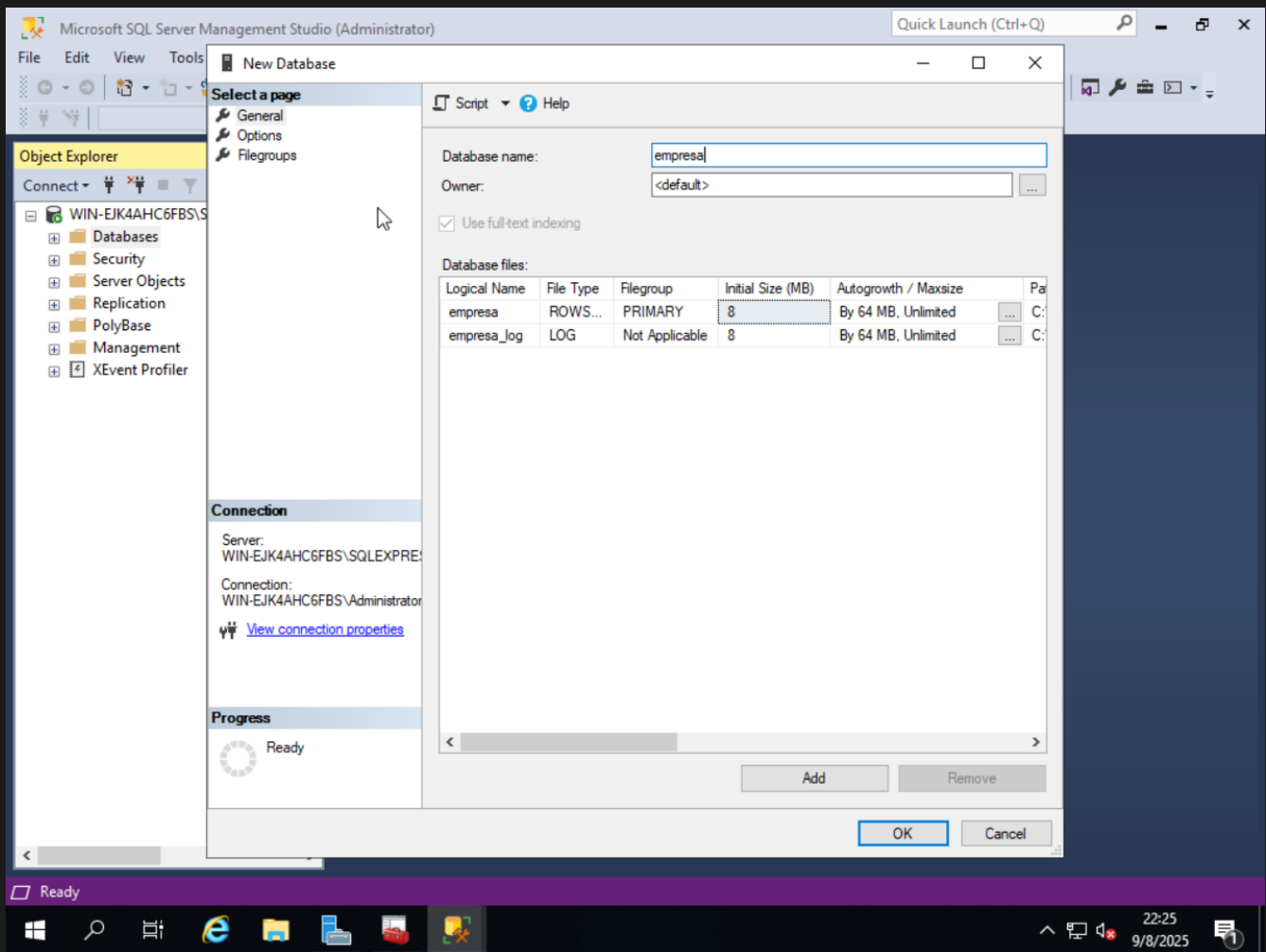
1. Instalar el SO en la VM (Windows Server o Windows 10).



2. Descargar e instalar SQL Server (puede ser SQL Server 2008 Express o SQL Server 2019 Express).

3. Abrir SQL Server Management Studio (SSMS) y:

- Crear una base de datos de prueba (por ejemplo, **empresa**).



- Añadir una tabla sencilla (**empleados**, **productos**, etc.).

4. Activar el acceso remoto en SQL Server:

- Habilitar el protocolo TCP/IP y Name Pipes en el SQL Server Configuration Manager.

✓ Resumen final: Conexión a SQL Server Express desde cliente

1. Problema inicial

- Desde el servidor: funcionaba con **.** o **localhost**.
- Desde cliente (Win 10): no conectaba con IP\SQLEXPRESS.

2. Solución encontrada

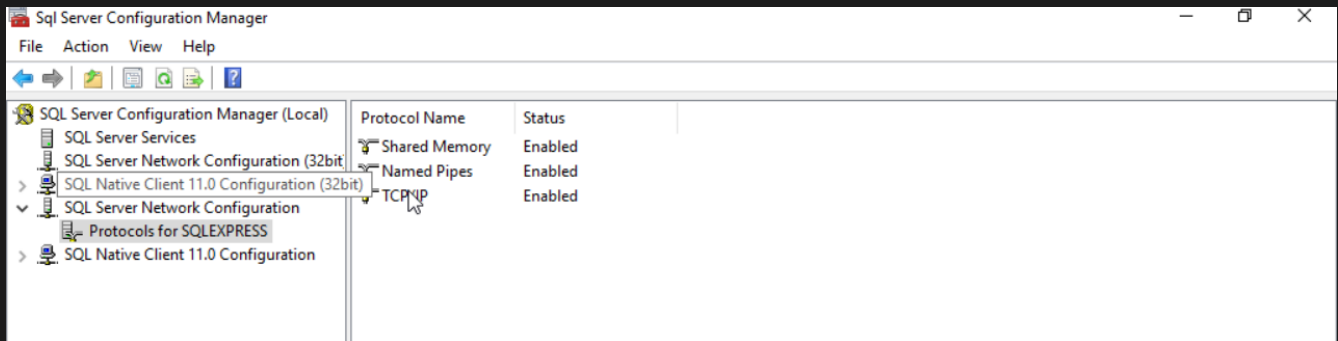
- ✓ Usar solo la IP del servidor (sin **\SQLEXPRESS**).
- Ejemplo: **192.168.1.100** → funciona.

3. ¿Por qué?

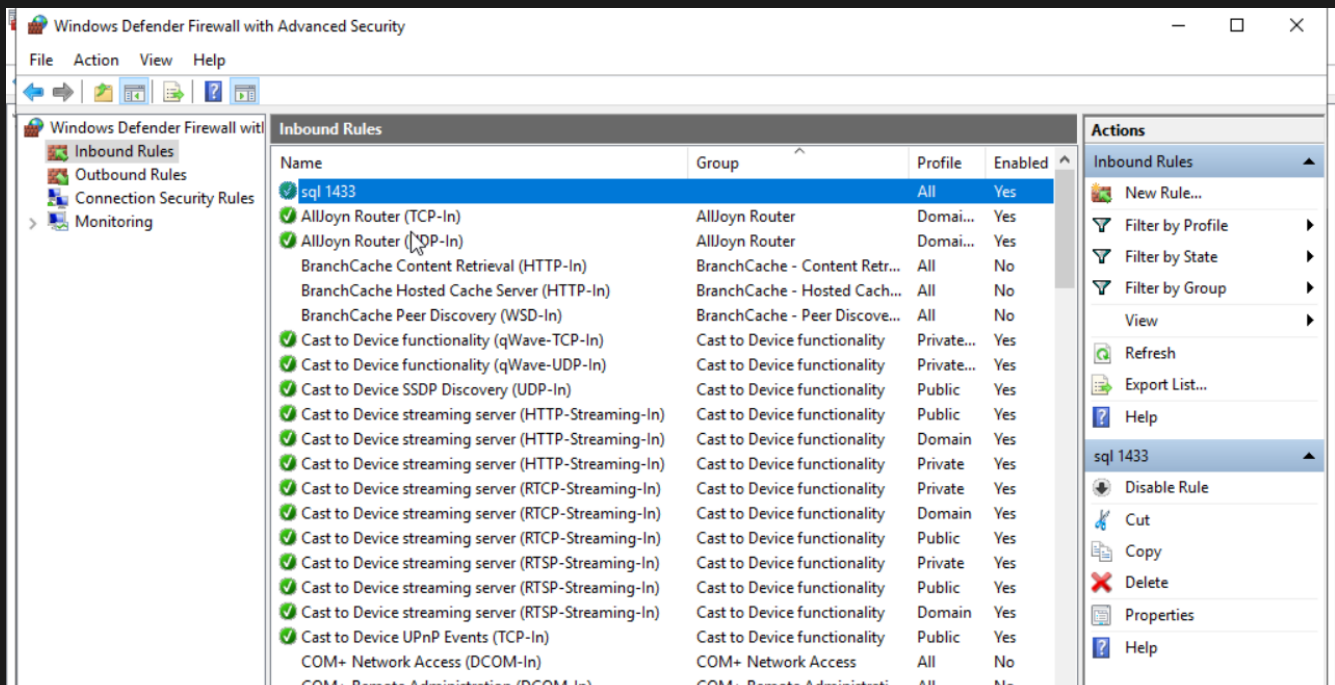
- La instancia no responde correctamente al nombre **SQLEXPRESS** desde fuera.
- Falta el servicio SQL Server Browser o está mal configurado.
- El cliente no puede "descubrir" la instancia nombrada.

4. Requisitos mínimos para que funcione

- ☒ TCP/IP habilitado en SQL Server Configuration Manager.
 - ☒ Puerto 1433 asignado en **IPAll**.
 - ☒ Firewall del servidor: puerto 1433 abierto (TCP).
 - ☒ Servicio SQL Server reiniciado después de cambios.
 - ☒ Máquinas en la misma red (VirtualBox: modo Puente recomendado).
5. Conexión desde SSMS (cliente)
- Nombre del servidor: **IP_DEL_SERVIDOR**
(ej: **192.168.1.100**)
 - Autenticación: SQL Server Authentication
 - Usuario: el creado en SQL (ej: **miusuario**)
 - Contraseña: la definida
6. No es necesario
- Usar **\SQLEXPRESS** si ya funciona con solo la IP.
 - Tener SQL Server Browser activo (aunque ayuda si usas instancias nombradas).
7. Consejo clave
- Si solo tienes una instancia, usa la IP + puerto 1433 → es más simple y directo.



- Configurar reglas de firewall para permitir el puerto 1433.

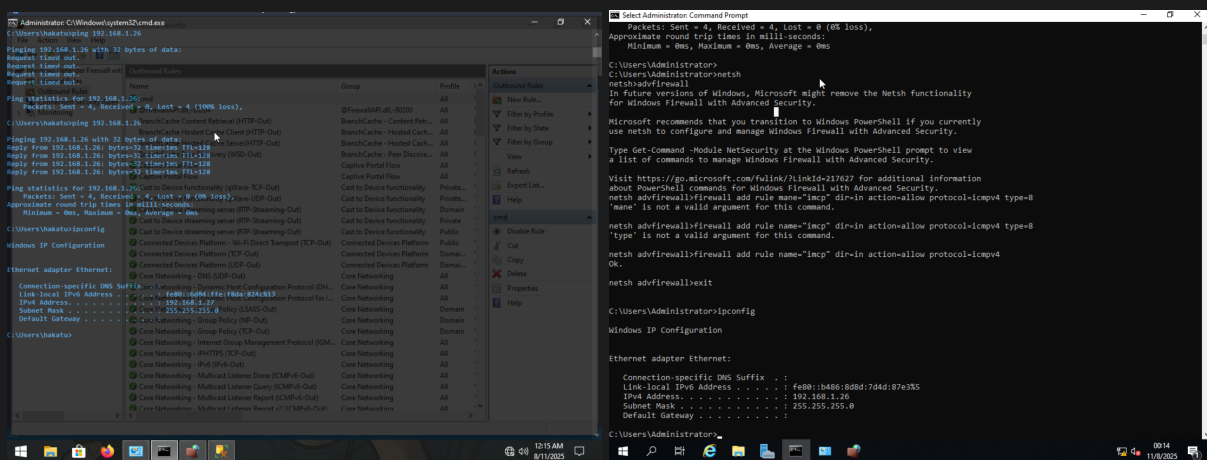


4. Integrar Packet Tracer con la VM

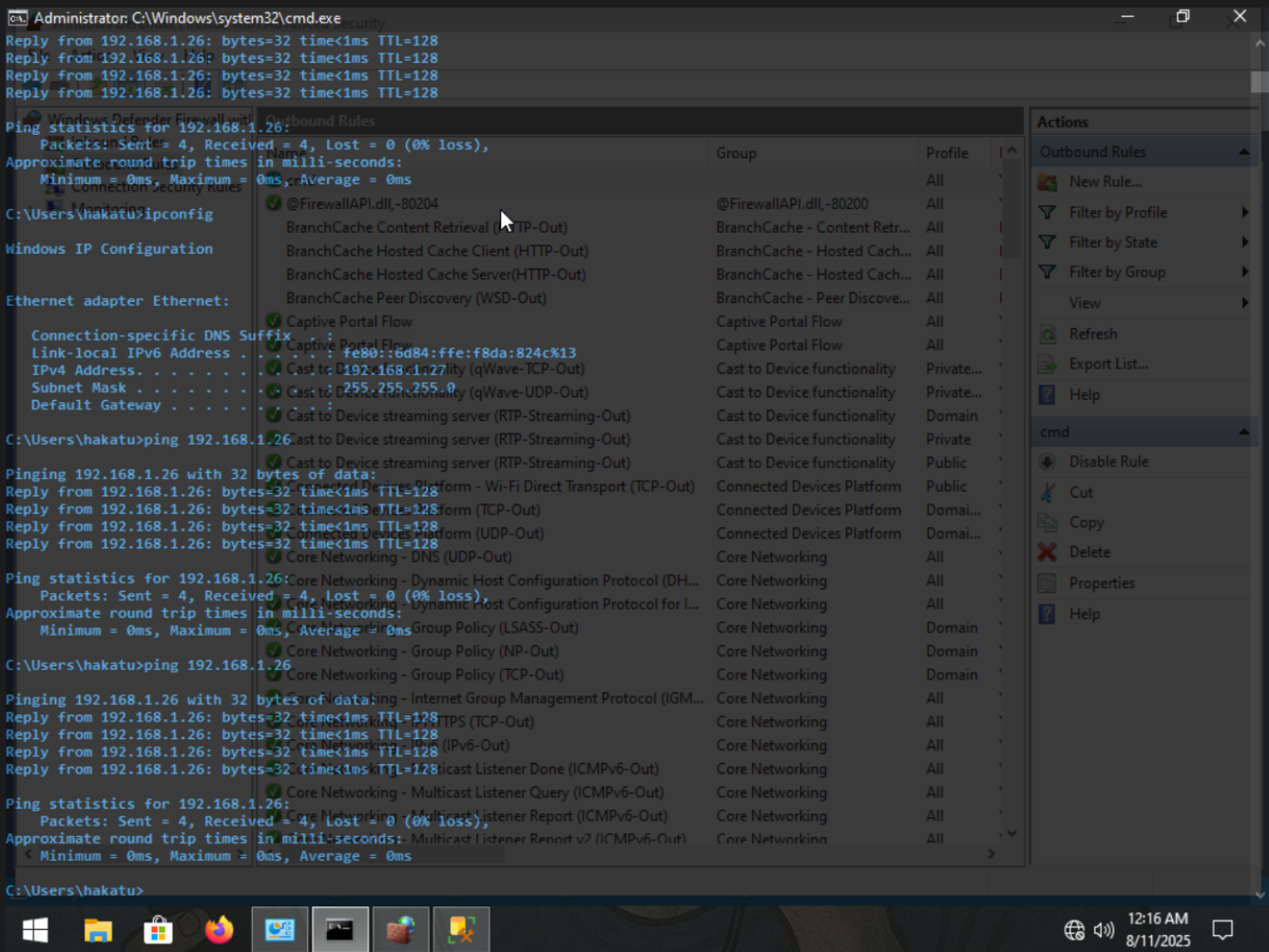
💡 **Nota:** Packet Tracer no conecta directamente con máquinas reales, así que vamos a simular la red completa en virtualización, o usar un adaptador puente para unir la VM y el host.

Pasos:

1. Asigna a la VM una IP estática en el mismo rango que tu red de Packet Tracer (por ejemplo, **192.168.1.27**).
2. Configura las PCs virtuales clientes (Windows 10 o Windows Server) en la misma red.

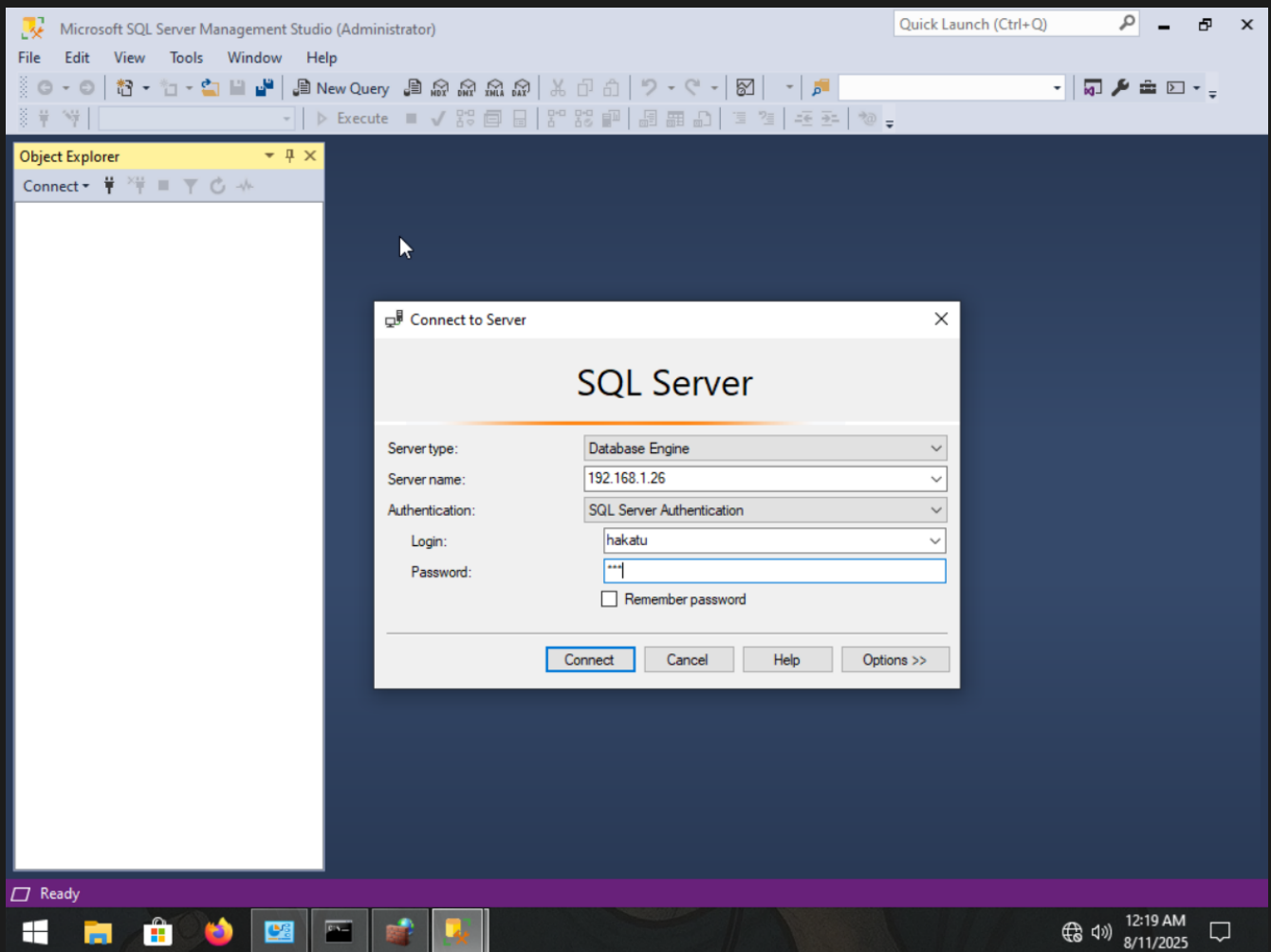


3. Comprueba conectividad con un ping desde las PCs cliente hacia el servidor SQL.



5. Conexión de las PCs al Servidor SQL

1. En la PC cliente, abre SQL Server Management Studio.
2. En Servidor, escribe la IP del servidor SQL (ejemplo: **192.168.1.26**).
3. Usa el usuario y contraseña que configuraste en SQL Server.



Haz una consulta de prueba:

```
SELECT * FROM productos;
```

4. Verifica que los datos se muestren correctamente.

SQLQuery1.sql - 192.168.1.26.empresa (hakatu (54)) - Microsoft SQL Server Management Studio (Administrator) Quick Launch (Ctrl+Q)

File Edit View Query Project Tools Window Help

empressa

Object Explorer

Connect

192.168.1.26 (SQL Server 15.0.2000 - hakatu (54))

Databases

System Databases

Database Snapshots

empresa

Security

Server Objects

Replication

PolyBase

Management

XEvent Profiler

SQLQuery1.sql - 192...presa (hakatu (54))

```
create table Productos(  
    ProductoId INT Primary key identity,  
    Nombre nvarchar(100),  
    Precio DECIMAL(10,2),  
    cantidad INT  
);  
  
insert into Productos values('INSTALACIONES',33,3);  
  
SELECT * FROM Productos;
```

100 %

Results Messages

	ProductoId	Nombre	Precio	cantidad
1	1	hakatu	33.00	3
2	2	soho	33.00	3
3	3	INSTALACIONES	33.00	3

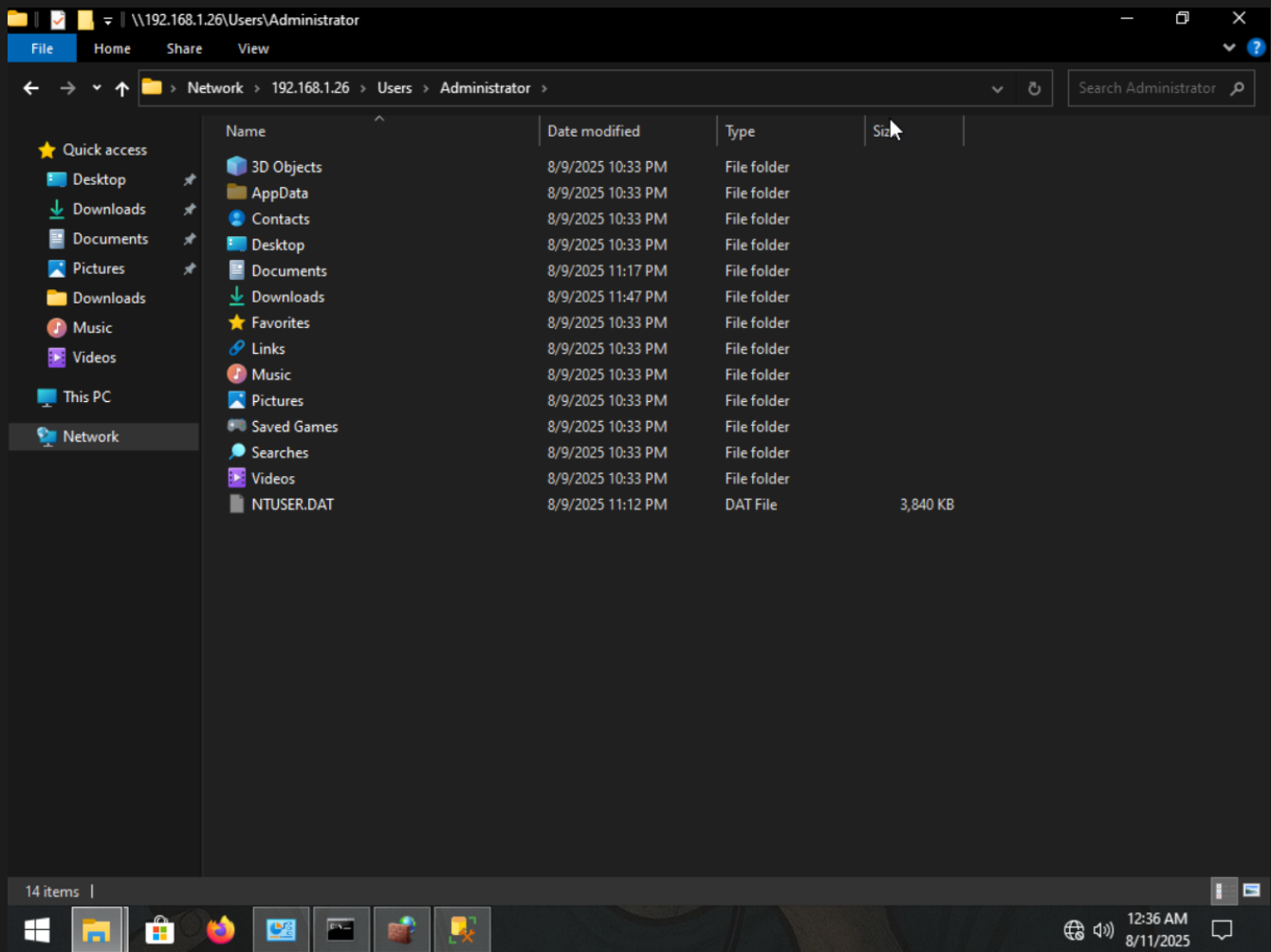
Query executed successfully. | 192.168.1.26 (15.0 RTM) | hakatu (54) | empresa | 00:00:00 | 3 rows

Ready Ln 13 Col 1 Ch 1 INS

12:31 AM 8/11/2025

6. Comprobación de Recursos Compartidos

- Si en episodios anteriores configuraste impresoras en red, intenta imprimir un documento desde la PC cliente.
- Verifica que la PC y el servidor se vean mutuamente en el Explorador de Windows.



✓ Tarea

- Comprueba la conexión entre PCs y servidor SQL.
- Haz una consulta y guarda el resultado.
- Si algo falla, revisa IPs, firewall y configuración de acceso remoto en SQL Server.

📌 Mensaje final

Con esto, ya tienes una red integrada que combina simulación (Packet Tracer) y servidores reales en máquinas virtuales, logrando un laboratorio de práctica muy similar a un entorno de trabajo real. En el próximo episodio, podríamos llevarlo al siguiente nivel con aplicaciones cliente-servidor que consuman la base de datos.

Perfecto 🙌 Aquí tienes el contenido de la guía del **Episodio 4** en texto plano para que lo copies directamente (es el mismo que te generé en el PDF, solo que en formato editable):

Episodio 4: Configuración Real en Linux

♦ **Objetivo:** Configurar un servidor Linux con IP estática, MariaDB, cliente de db(dbeaver) , y CUPS para simular un entorno real de PyME.

✓ 1. Configuración de IP estática en Linux

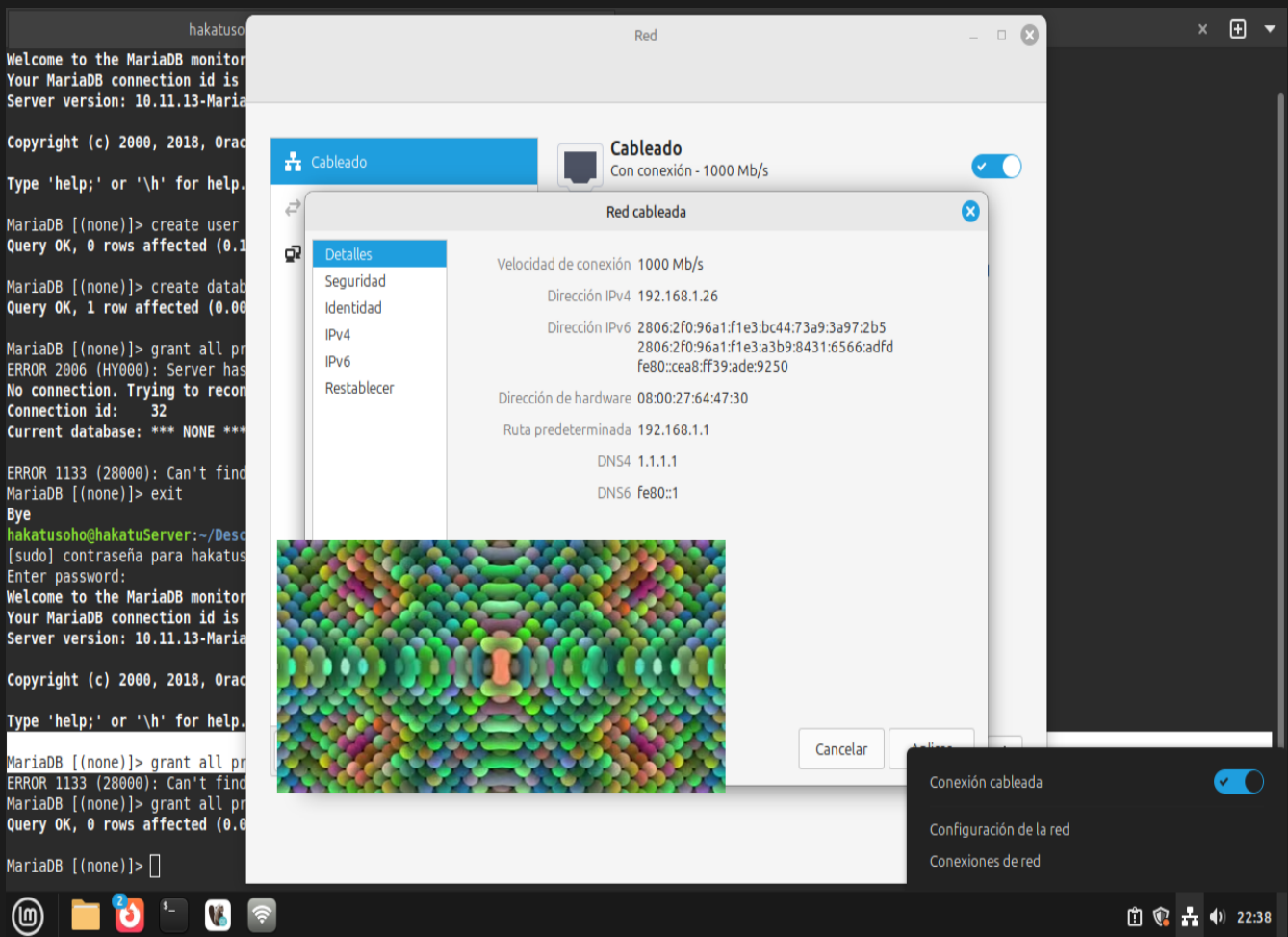
Abrimos la interfaz gráfica y ponemos la configuración de red:

Ejemplo de configuración del servidor:

```
address 192.168.1.26
netmask 255.255.255.0
gateway 192.168.1.1
dns-nameservers 8.8.8.8
```

Ejemplo de configuración del cliente:

```
address 192.168.1.267
netmask 255.255.255.0
gateway 192.168.1.1
dns-nameservers 8.8.8.8
```



Reiniciar red desde consola o interfaz gráfica:

```
sudo systemctl restart networking
```

Verificar conectividad:

```
ping 8.8.8.8
```

✓ 2. Instalación de MariaDB Server

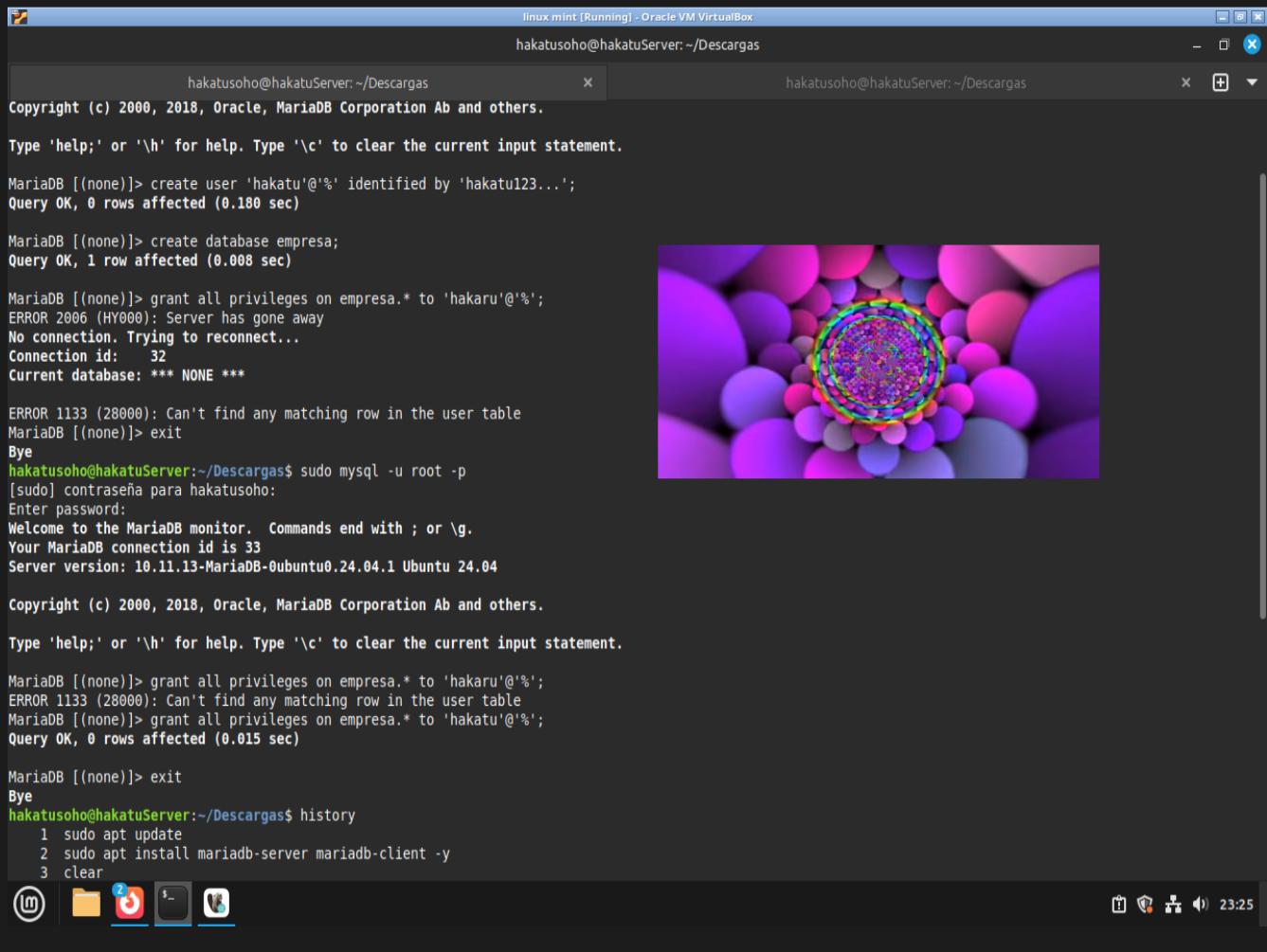
MariaDB es el reemplazo gratuito y abierto de MySQL. instalarlo desde línea de comando o ubuntu software tienda de aplicaciones.

```
sudo apt update
```

```
sudo apt install mariadb-server mariadb-client -y
```

Entrar al cliente:

`sudo mysql -u root -p`



The screenshot shows a terminal window titled "linux mint [Running] - Oracle VM VirtualBox" with a sub-header "hakatusoho@hakatuServer: ~/Descargas". The terminal displays the following sequence of commands and outputs:

```
hakatusoho@hakatuServer: ~/Descargas
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create user 'hakatu'@'%' identified by 'hakatu123...';
Query OK, 0 rows affected (0.180 sec)

MariaDB [(none)]> create database empresa;
Query OK, 1 row affected (0.008 sec)

MariaDB [(none)]> grant all privileges on empresa.* to 'hakaru'@'%';
ERROR 2006 (HY000): Server has gone away
No connection. Trying to reconnect...
Connection id: 32
Current database: *** NONE ***

ERROR 1133 (28000): Can't find any matching row in the user table
MariaDB [(none)]> exit
Bye
hakatusoho@hakatuServer:~/Descargas$ sudo mysql -u root -p
[sudo] contraseña para hakatusoho:
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 33
Server version: 10.11.13-MariaDB-0ubuntu0.24.04.1 Ubuntu 24.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> grant all privileges on empresa.* to 'hakaru'@'%';
ERROR 1133 (28000): Can't find any matching row in the user table
MariaDB [(none)]> grant all privileges on empresa.* to 'hakatu'@'%';
Query OK, 0 rows affected (0.015 sec)

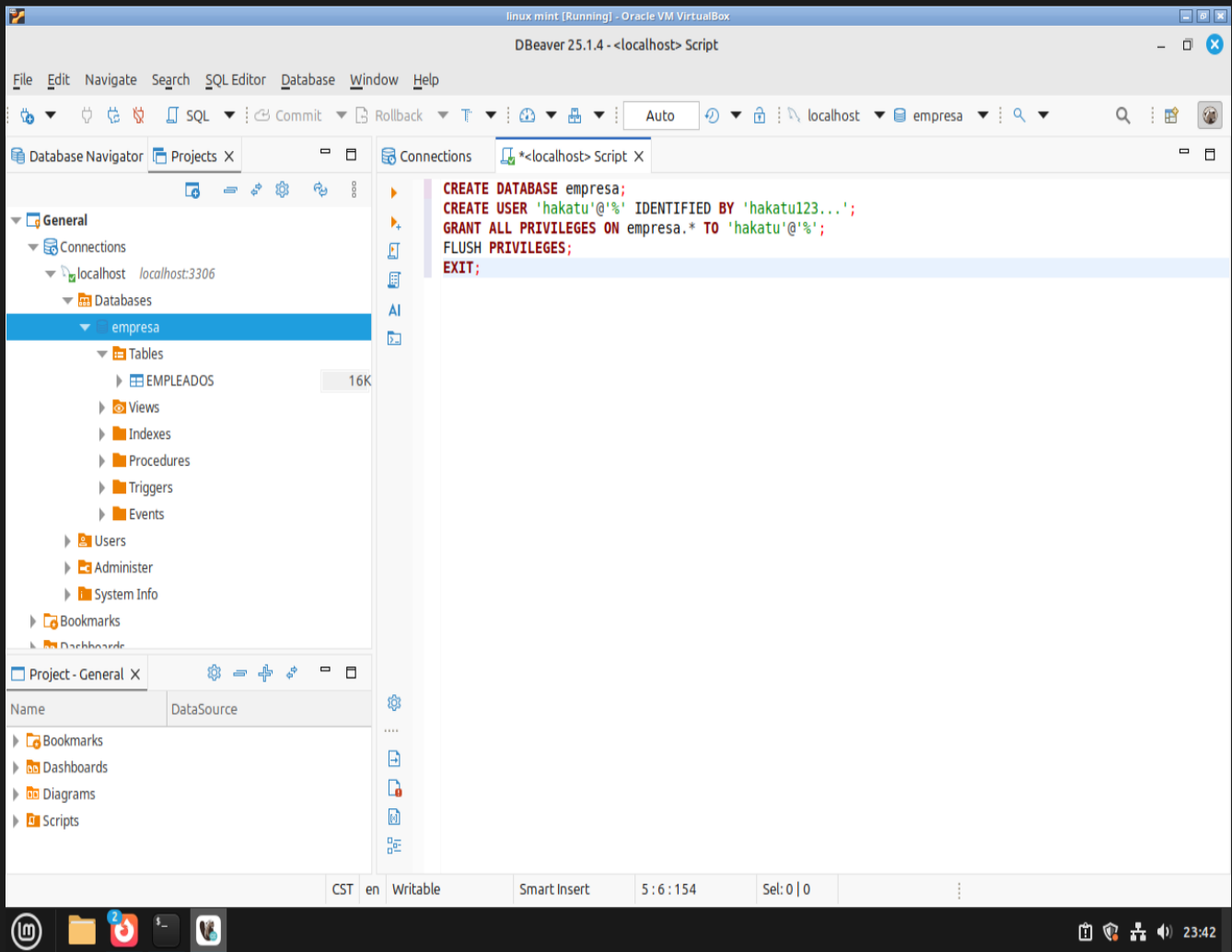
MariaDB [(none)]> exit
Bye
hakatusoho@hakatuServer:~/Descargas$ history
1 sudo apt update
2 sudo apt install mariadb-server mariadb-client -y
3 clear
```

On the right side of the terminal, there is a colorful fractal image with a central circular pattern of dots in various colors (red, green, blue, yellow) surrounded by concentric rings of dots.

✓ 3. Creación de base de datos y usuario

Dentro del cliente MariaDB en el servidor en la terminal:

```
CREATE DATABASE empresa;
CREATE USER 'hakatu'@'%' IDENTIFIED BY '1234';
GRANT ALL PRIVILEGES ON empresa.* TO 'hakatu'@'%';
FLUSH PRIVILEGES;
EXIT;
```



Verificar que MariaDB escuche en todas las interfaces:

```
sudo nano /etc/mysql/mariadb.conf.d/50-server.cnf
```

Buscar la línea:

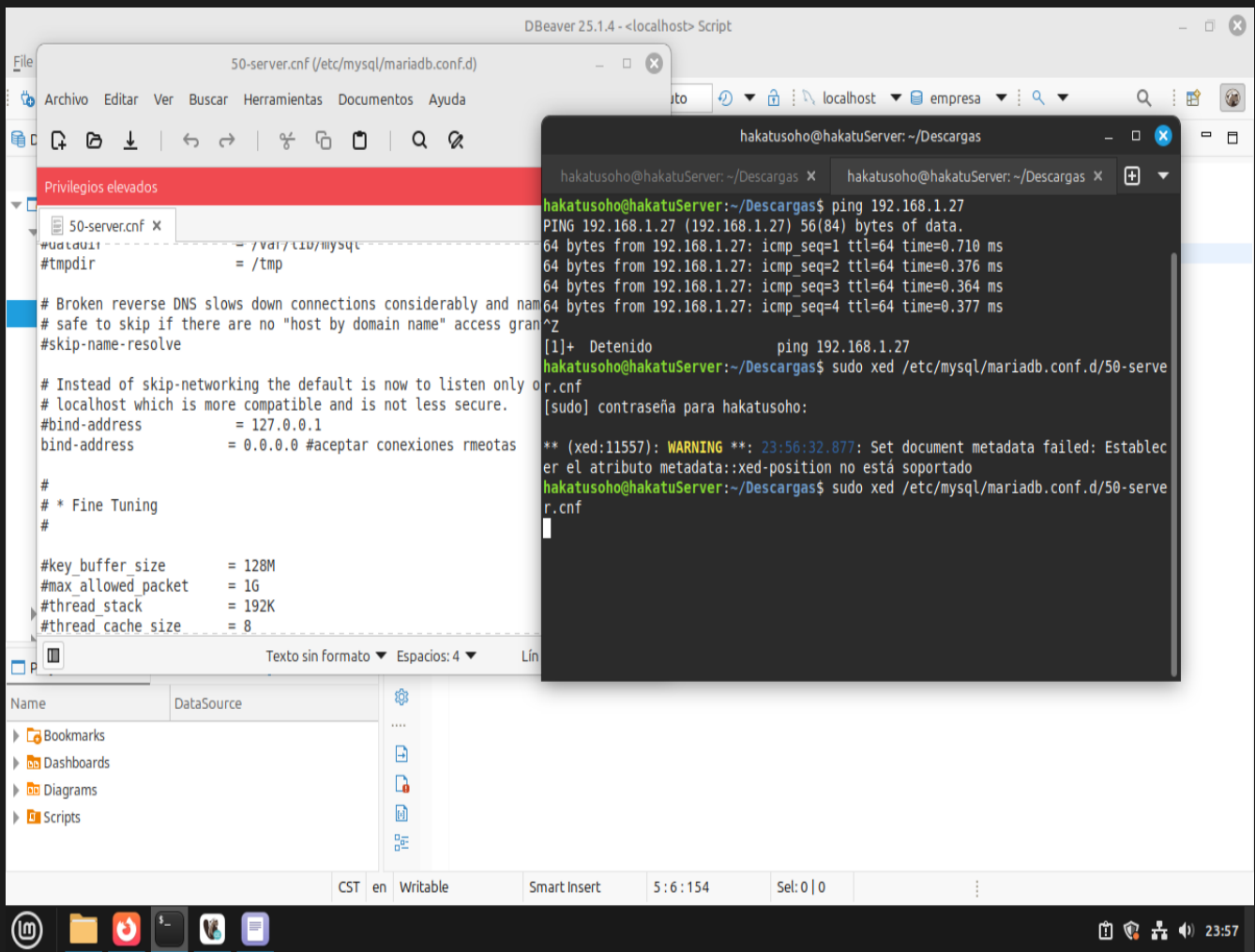
```
bind-address = 127.0.0.1
```

Y cambiarla por:

```
bind-address = 0.0.0.0
```

Reiniciar MariaDB:

```
sudo systemctl restart mariadb
```



✓ 4. Instalación de cliente de DB (servidor y cliente)

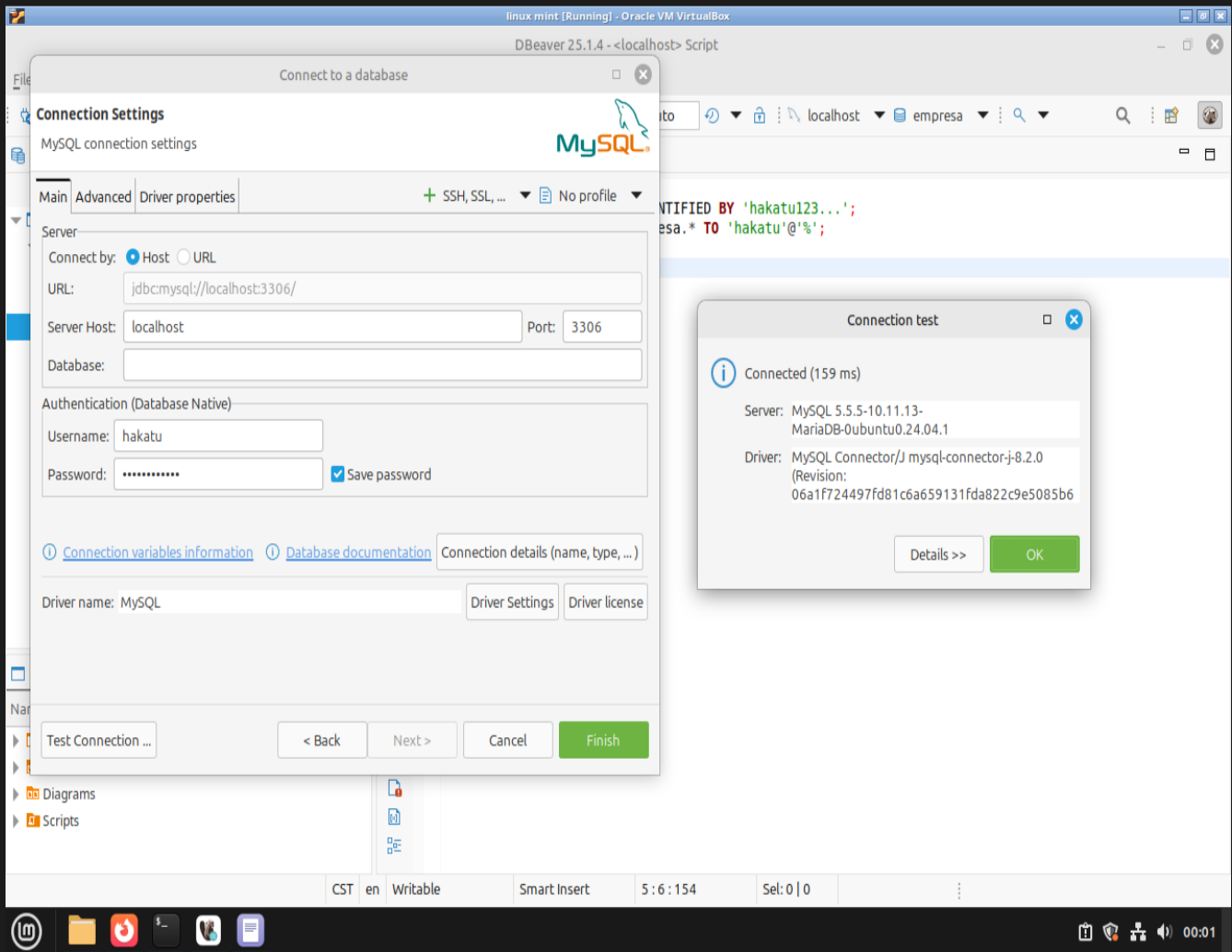
En **ambas máquinas** (Linux servidor y Linux cliente):

Descarga dbeaver e instalar el paquete .deb:

En el cliente abrir menu database opción new database connection(mysql):

- **Host:** 192.168.1.26
- **User:** hakatu
- **Password:** hakatu123...

Probar la conexión.



Crear tabla de prueba desde el cliente dbeaver :

USE empresa;

CREATE TABLE empleados (

id INT AUTO_INCREMENT PRIMARY KEY,

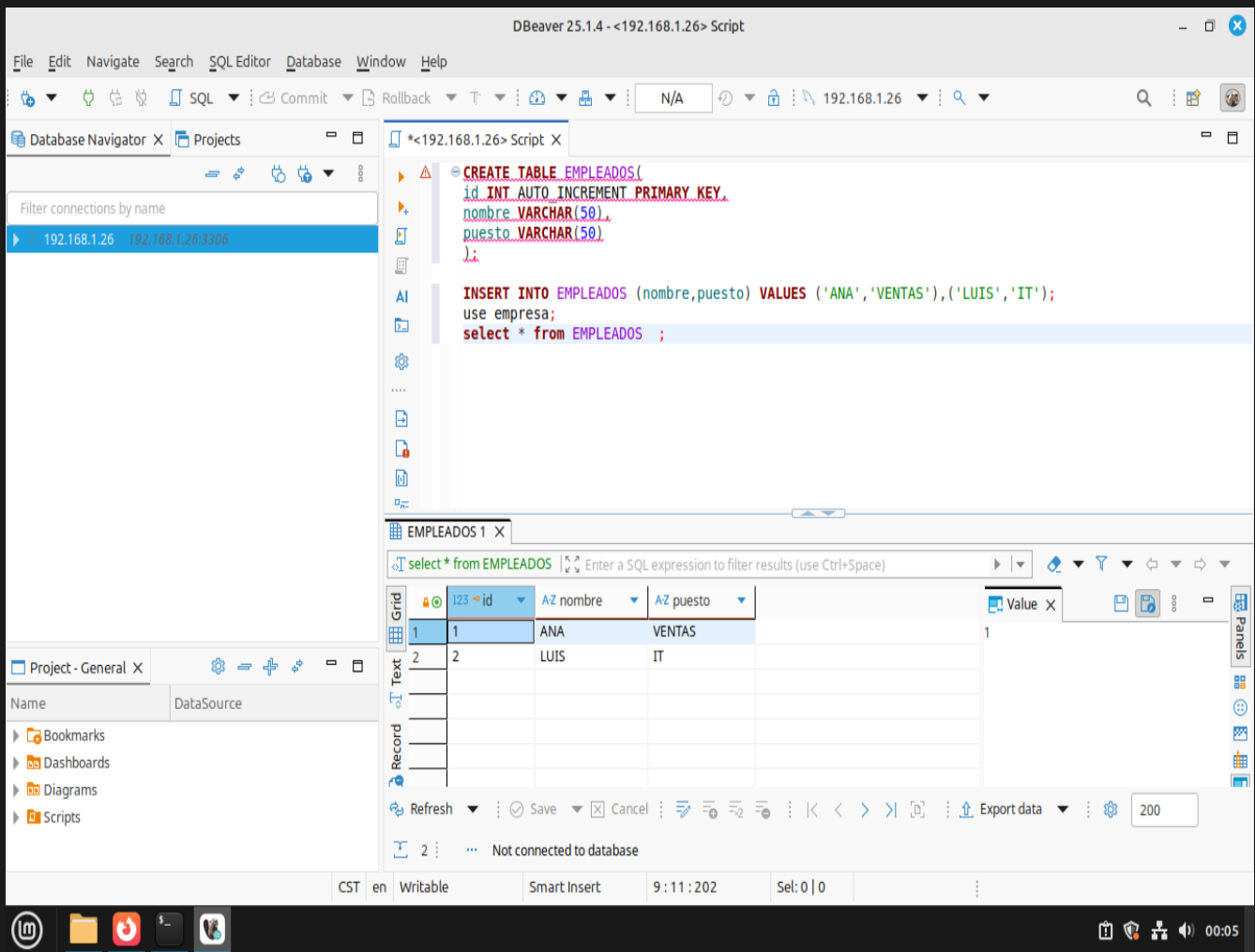
nombre VARCHAR(50),

puesto DECIMAL(10,2)

);

INSERT INTO empleado (nombre,puesto) VALUES ('ANA','VENTAS'),('LUIS','IT');

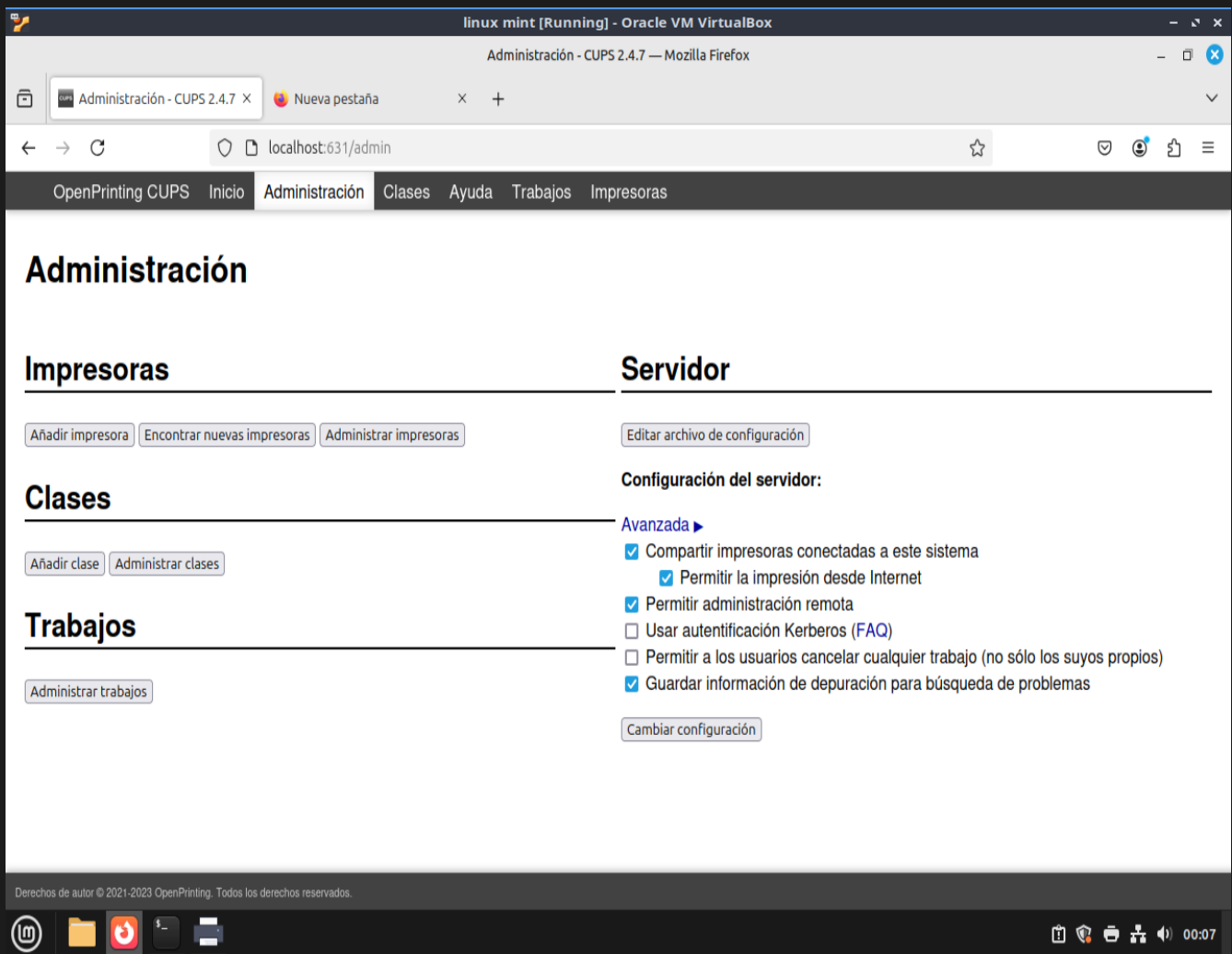
SELECT * FROM productos;



✓ 5. Configuración de CUPS para impresora en red

✓ Instalación de CUPS desde la interfaz gráfica en Lubuntu/Ubuntu

Una vez iniciado el sistema, **CUPS (Common Unix Printing System)** fue instalado desde la **interfaz gráfica** utilizando la **tienda de aplicaciones de Ubuntu (Ubuntu Software)**, como una forma sencilla y accesible para usuarios que no desean usar la línea de comandos. Damos permisos para compartir en red en el menú administración.



O puedes utilizar los comandos.

Instalar CUPS en el servidor:

```
sudo apt install cups -y
```

Dar permisos a usuarios:

```
sudo usermod -aG lpadmin $USER
```

Editar configuración:

```
sudo nano /etc/cups/cupsd.conf
```

En la sección `<Location />` y `<Location /admin>` permitir acceso desde la red:

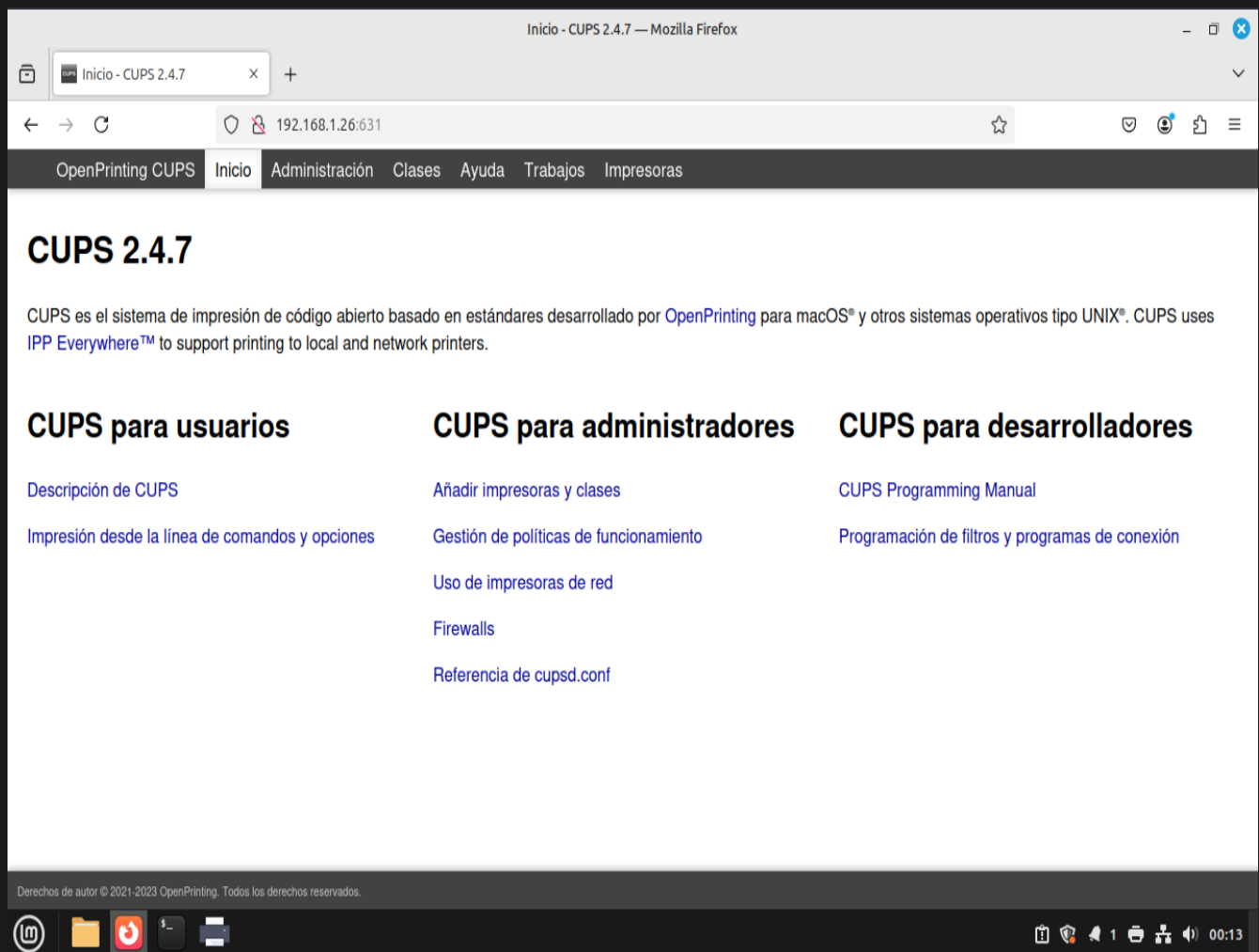
```
Allow @LOCAL
```

Reiniciar CUPS:

```
sudo systemctl restart cups
```

Acceder desde el navegador del cliente:

<http://192.168.1.26:631>



✓ 6. Verificación de conectividad

En el cliente probar:

```
ping 192.168.1.26
```

Impresoras - CUPS 2.4.7 — Mozilla Firefox

192.168.1.26/631/printers/

OpenPrinting CUPS Inicio Administración Clases Ayuda Trabajos Impresoras

Impresoras

Buscar en impresoras

Nombre de la cola	Descripción	Estado
CUPsSOHO	192.168.1.26	Inactiva
PDF	PDF	Inactiva

```
hakatuciente@hakatuciente-VirtualBox: ~  
hakatuciente@hakatuciente-VirtualBox:~$ ping 192.168.1.26  
PING 192.168.1.26 (192.168.1.26) 56(84) bytes of data:  
64 bytes from 192.168.1.26: icmp_seq=1 ttl=64 time=0.360 ms  
64 bytes from 192.168.1.26: icmp_seq=2 ttl=64 time=0.356 ms  
64 bytes from 192.168.1.26: icmp_seq=3 ttl=64 time=0.433 ms  
64 bytes from 192.168.1.26: icmp_seq=4 ttl=64 time=0.454 ms  
64 bytes from 192.168.1.26: icmp_seq=5 ttl=64 time=0.563 ms  
64 bytes from 192.168.1.26: icmp_seq=6 ttl=64 time=0.524 ms  
64 bytes from 192.168.1.26: icmp_seq=7 ttl=64 time=1.03 ms  
64 bytes from 192.168.1.26: icmp_seq=8 ttl=64 time=3.19 ms  
64 bytes from 192.168.1.26: icmp_seq=9 ttl=64 time=12.5 ms  
64 bytes from 192.168.1.26: icmp_seq=10 ttl=64 time=8.66 ms  
64 bytes from 192.168.1.26: icmp_seq=11 ttl=64 time=0.412 ms  
64 bytes from 192.168.1.26: icmp_seq=12 ttl=64 time=7.05 ms  
64 bytes from 192.168.1.26: icmp_seq=13 ttl=64 time=0.364 ms  
64 bytes from 192.168.1.26: icmp_seq=14 ttl=64 time=4.40 ms  
64 bytes from 192.168.1.26: icmp_seq=15 ttl=64 time=0.396 ms  
64 bytes from 192.168.1.26: icmp_seq=16 ttl=64 time=0.828 ms  
64 bytes from 192.168.1.26: icmp_seq=17 ttl=64 time=0.313 ms  
64 bytes from 192.168.1.26: icmp_seq=18 ttl=64 time=0.460 ms  
64 bytes from 192.168.1.26: icmp_seq=19 ttl=64 time=0.405 ms  
^Z  
[2]+ Detenido                  ping 192.168.1.26  
hakatuciente@hakatuciente-VirtualBox:~$
```

Verificar puertos abiertos:

`netstat -tulnp | grep mysql`

Si todo funciona:

- ✓ Cliente conectado a servidor MariaDB.
- ✓ Workbench funcionando en ambos lados.
- ✓ CUPS accesible desde cliente.

📌 “Si completaste todos los videos 🎉. Ahora intenta implementar esto en un entorno real de tu oficina o PyME.”



Da el siguiente paso hoy mismo

Empieza con nuestro **Curso Gratis de 5 días**(<https://hakatu.com/cursoxemail.php>) o accede directo al **Paquete A** o **Paquete B** y resuelve tus problemas de impresoras desde hoy.



Paquete A — \$99 MXN

- Instalación de drivers por ID de hardware.
- Uso de drivers genéricos.
- Extracción de controladores de ejecutables.
- Guías paso a paso para Windows y Ubuntu.
👉 Disponible en:

<https://hakatu.com/PaqueteA.php>



Paquete B — \$149 MXN

- Todo lo del Paquete A.
 - Extras de soporte remoto.
 - Curso por email con casos prácticos.
👉 Disponible en:

<https://hakatu.com/PaqueteA.php>



Soporte directo por correo o chat después de tu compra.

Sin llamadas, sin complicaciones.

Resuelve fácil con **APA System Tech Support**.

Agenda consulta gratis aqui: <https://hakatu.com/AgendarCita.php>