

## LABORATORY NO.07 - BASIC INFRASTRUCTURE AND NETWORK LAYER



ELABORADO POR:

ESTEBAN AGUILERA CONTRERAS  
JUAN DAVID RODRIGUEZ RODRIGUEZ

PROFESOR(ES):

JOHN PACHON

RECO  
2025-1

## **OBJECTIVE**

- Continue learning about the operation of operating systems and network services.
- Install network management tools.
- Configure routers and static routing.

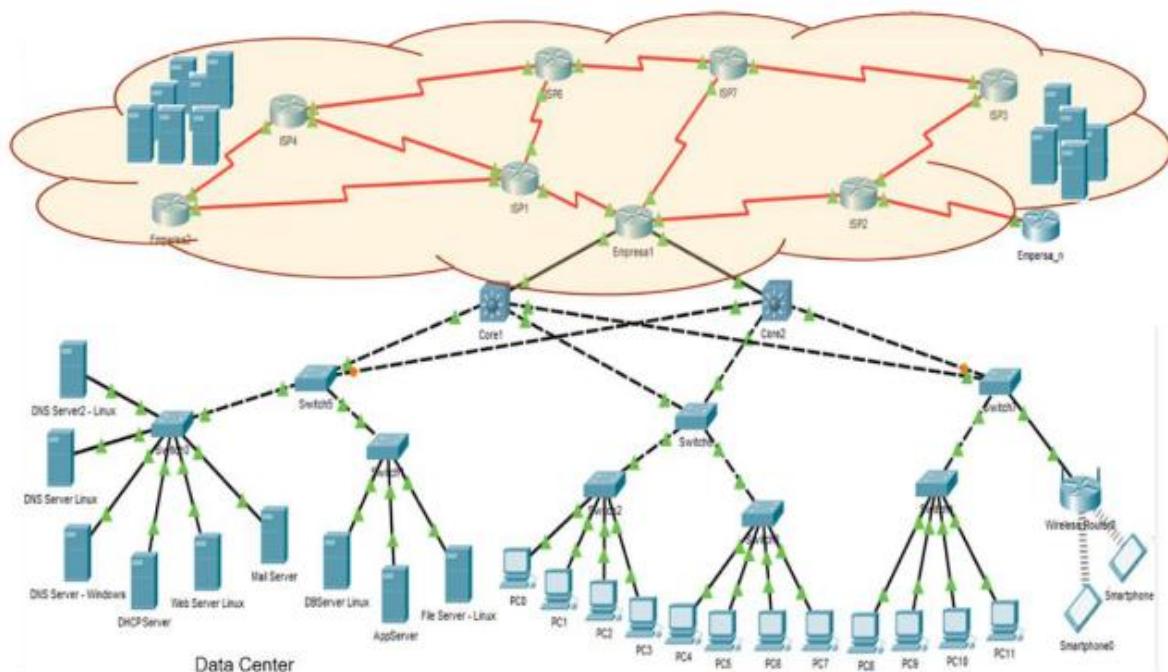
## **TOOLS TO BE USED**

- Computers
- Internet access
- Virtualization software
- Packet Tracer and Wireshark
- Routers and switches

## INTRODUCTION

A company typically has several IT infrastructure services. It includes wired and wireless user stations and servers (both physical and virtualized), all connected through switches (layer 2 and 3), wireless devices, and routers that connect it to the Internet. It is also common to have cloud infrastructures where resources are provisioned according to the organization's needs. Among the servers, there can be web services, DNS, email, database, storage, applications, and others.

The following is a possible configuration:



## THEORETICAL FRAMEWORK

- Capa de Red e Infraestructura de Red
  - La infraestructura de red es el conjunto de dispositivos, protocolos y servicios que permiten la conexión entre estaciones de trabajo, servidores, y dispositivos móviles, tanto de forma cableada como inalámbrica. Esta infraestructura está compuesta por routers, switches de capa 2 y 3, puntos de acceso, y servicios como DNS, servidores web, correo electrónico, y bases de datos, tanto en entornos locales como en la nube.
  - La capa de red del modelo OSI se encarga de enrutar los paquetes desde el origen hasta el destino a través de una o varias redes. Su principal protocolo es IP, que puede ser en versión IPv4 o IPv6. Esta capa define las direcciones IP, máscaras de subred, y determina cómo los paquetes se reparten y se reencaminan entre dispositivos de diferentes redes.
- Direccionamiento IP y Subnetting
  - Cada dispositivo conectado a una red necesita una dirección IP única. Para optimizar el uso de direcciones y facilitar la administración, se usan máscaras de subred para dividir redes grandes en redes más pequeñas. El subnetting permite separar segmentos lógicos dentro de una misma red física, mejorando la eficiencia y la seguridad.
- Enrutamiento estático
  - El enrutamiento estático es el proceso mediante el cual un administrador configura manualmente las rutas que debe seguir un paquete para alcanzar una red destino. A diferencia del enrutamiento dinámico (como RIP), las rutas estáticas son fijas y no cambian automáticamente si ocurre un fallo en la red. Son útiles en redes pequeñas o en enlaces donde se desea un mayor control.
- Dispositivos de Red: Routers y Modems Nulos
  - Los routers son dispositivos fundamentales en una red, ya que permiten la conexión entre distintas redes y gestionan el tráfico mediante tablas de enrutamiento. En el laboratorio, los routers pueden ser configurados por consola mediante un cable console o mediante una conexión serial.
  - Un null modem es un tipo de conexión serial cruzada que permite la comunicación directa entre dos dispositivos DTE sin un módem o dispositivo intermedio. En el laboratorio, se usa para simular enlaces seriales punto a punto.
- DTE y DCE en Conexiones Seriales
  - En una conexión serial, los dispositivos se dividen en DTE (Data Terminal Equipment) y DCE (Data Communications Equipment). El DTE es normalmente el dispositivo final mientras que el DCE proporciona la señal de reloj. En el laboratorio, uno de los routers debe actuar como DCE y, por lo tanto, es necesario usar el comando clock rate para definir la velocidad de transmisión de datos. Sin

esta configuración, la conexión serial no puede sincronizar los datos correctamente.

- Protocolos ICMP, Traceroute y Herramientas de Diagnóstico
  - El protocolo ICMP se utiliza para enviar mensajes de error y de diagnóstico en redes IP. Herramientas como ping y traceroute utilizan ICMP para verificar conectividad y trazar la ruta que siguen los paquetes a través de la red. Estas herramientas son esenciales para detectar fallos de conectividad o rutas mal configuradas en entornos de laboratorio y producción.
- SNMP y Monitorización de Red
  - El protocolo SNMP permite la supervisión y gestión de dispositivos de red, como routers y switches. A través de SNMP, es posible recolectar información sobre el uso de CPU, memoria, interfaces de red, y más. Herramientas como Zabbix o Nagios utilizan SNMP para visualizar el estado de la red en tiempo real, facilitando la detección y solución de fallos antes de que afecten a los usuarios.
- Azure y Aplicaciones en la Nube
  - Con plataformas como Microsoft Azure, es posible desplegar aplicaciones web desde repositorios como GitHub, habilitar servicios de análisis como Application Insights, y monitorizar el rendimiento de los servicios en la nube. Estas funcionalidades permiten observar métricas como tráfico, errores y tiempos de respuesta, y son fundamentales para el mantenimiento de aplicaciones distribuidas geográficamente.

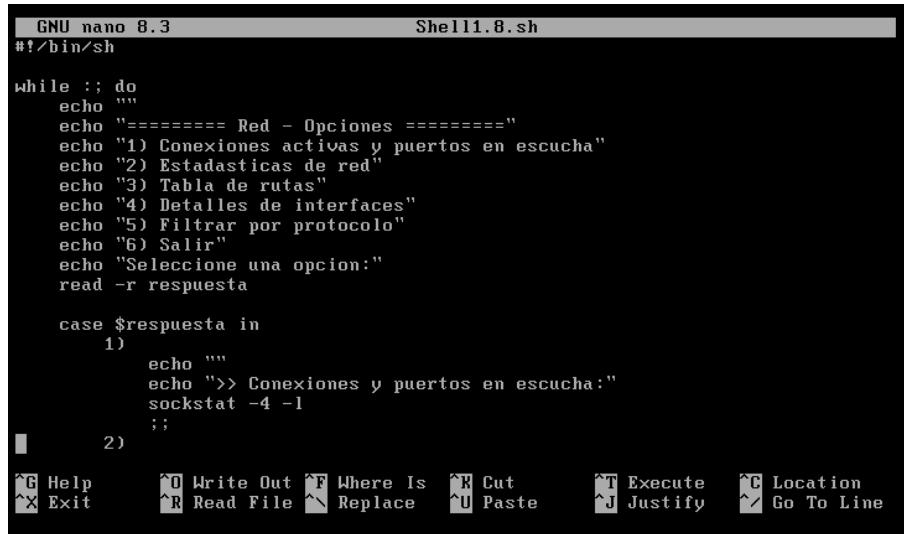
## BASE SOFTWARE INSTALLATION

### 1. Other Useful Commands

- a. Study the functionality of commands that provide network information, e.g., netstat, vnstat, route, and ethtool (or equivalents in Slackware, NetBSD, Centos, and PowerShell). Review different parameters that can be used and create a Shell program that uses them (create a menu with at least 5 different options that show different executions of the commands). Students should be able to understand the output of the commands and present it in an easy-to-understand format

### NETBSD

- Primero, creamos nuestro script de shell y diseñamos un menú con las opciones mostradas en la imagen. Usamos netstat para mostrar información de red como puertos, servicios, estadísticas y conexiones UDP y TCP. También usamos route para ver la tabla de enrutamiento y ethtool para verificar la información de las interfaces de red.



```
GNU nano 8.3                               Shell1.8.sh
#!/bin/sh

while :; do
    echo ""
    echo "===== Red - Opciones ====="
    echo "1) Conexiones activas y puertos en escucha"
    echo "2) Estadísticas de red"
    echo "3) Tabla de rutas"
    echo "4) Detalles de interfaces"
    echo "5) Filtrar por protocolo"
    echo "6) Salir"
    echo "Seleccione una opción:"
    read -r respuesta

    case $respuesta in
        1)
            echo ""
            echo ">> Conexiones y puertos en escucha:"
            sockstat -4 -l
            ;;
        2)
            echo ""
            echo ">> Estadísticas de red"
            route
            ;;
        3)
            echo ""
            echo ">> Tabla de rutas"
            netstat -rn
            ;;
        4)
            echo ""
            echo ">> Detalles de interfaces"
            ifconfig
            ;;
        5)
            echo ""
            echo ">> Filtrar por protocolo"
            netstat -an | grep -i "tcp|udp"
            ;;
        6)
            echo ""
            echo "Saliendo..."
            exit
            ;;
        *)
            echo "Opción no válida"
            ;;
    esac
done
```

The status bar at the bottom of the terminal window includes the following keys:

- Help (F1)
- Write Out (F2)
- Where Is (F3)
- Cut (F4)
- Execute (F5)
- Location (F6)
- Exit (Alt+F4)
- Read File (Shift+F2)
- Replace (Shift+F3)
- Paste (Shift+F4)
- Justify (Shift+F5)
- Go To Line (Shift+F6)

```

GNU nano 8.3                               Shell1.8.sh
2)
    echo ""
    echo ">> Estadísticas de uso de red:"
    netstat -i
    ;;
3)
    echo ""
    echo ">> Mostrando tabla de rutas:"
    netstat -r
    ;;
4)
    echo ""
    echo ">> Información de interfaces de red:"
    ifconfig -a
    ;;
5)
    echo ""
    echo "Seleccione protocolo a visualizar:"
    echo "1 - TCP"
    echo "2 - UDP"
    read -r tipo

```

**Toolbar:**

- ^G Help
- ^O Write Out
- ^F Where Is
- ^R Cut
- ^T Execute
- ^C Location
- ^X Exit
- ^R Read File
- ^N Replace
- ^U Paste
- ^J Justify
- ^/ Go To Line

```

GNU nano 8.3                               Shell1.8.sh
        sockstat -4 -P tcp
        ;;
2)
    echo ""
    echo ">> Conexiones UDP:"
    sockstat -4 -P udp
    ;;
*)
    echo ">> Opción inválida de protocolo."
    ;;
esac
;;
6)
    echo ">> Cerrando el programa."
    exit 0
;;
*)
    echo ">> Opción no reconocida. Intente de nuevo."
    ;;
esac
done

```

**Toolbar:**

- ^G Help
- ^O Write Out
- ^F Where Is
- ^R Cut
- ^T Execute
- ^C Location
- ^X Exit
- ^R Read File
- ^N Replace
- ^U Paste
- ^J Justify
- ^/ Go To Line

- Probamos la opción 1 del menú (Conexiones de red y puertos en escucha). Como podemos ver, se muestran los protocolos de la capa de transporte con los puertos en los que están escuchando, junto con las direcciones IP permitidas.

```

1
Mostrando puertos en escucha y servicios:
tcp      0      0  *.80          *.*          LISTEN
tcp      0      0  *.5666         *.*          LISTEN
tcp      0      0  *.139          *.*          LISTEN
tcp      0      0  *.445          *.*          LISTEN
tcp      0      0  127.0.0.1.953  *.*          LISTEN
tcp      0      0  127.0.0.1.53   *.*          LISTEN
tcp      0      0  10.2.78.35.53  *.*          LISTEN
tcp6     0      0  *.5666          *.*          LISTEN
tcp6     0      0  *.139          *.*          LISTEN
tcp6     0      0  *.445          *.*          LISTEN
tcp6     0      0  ::1.953         *.*          LISTEN
tcp6     0      0  fe80::1%lo0.53  *.*          LISTEN
tcp6     0      0  ::1.53          *.*          LISTEN
tcp6     0      0  fe80::20c:29ff:f.53  *.*          LISTEN

```

- Probamos la opción 2 en el menú (Uso de red). Podemos ver información sobre eth1 (Interfaz de red) y lo (Bucle de retorno).

| Estadísticas de red: |       |               |                   |       |       |       |       |       |  |
|----------------------|-------|---------------|-------------------|-------|-------|-------|-------|-------|--|
| Name                 | Mtu   | Network       | Address           | Ipkts | Ierrs | Opkts | Oerrs | Colls |  |
| wm0                  | 1500  | <Link>        | 00:0c:29:cf:c8:84 | 751   | 0     | 132   | 0     | 0     |  |
| wm0                  | 1500  | fe80::/64     | fe80::20c:29ff:fe | 751   | 0     | 132   | 0     | 0     |  |
| wm0                  | 1500  | 10.2/16       | Juanito.is.escuel | 751   | 0     | 132   | 0     | 0     |  |
| lo0                  | 33624 | <Link>        |                   | 52    | 0     | 52    | 0     | 0     |  |
| lo0                  | 33624 | 127/8         | localhost         | 52    | 0     | 52    | 0     | 0     |  |
| lo0                  | 33624 | localhost/128 | ::1               | 52    | 0     | 52    | 0     | 0     |  |
| lo0                  | 33624 | fe80::/64     | fe80::1           | 52    | 0     | 52    | 0     | 0     |  |

- Probamos la opción 3 en el menú (Tabla de enrutamiento). Podemos ver las rutas en la tabla de enrutamiento.

| 33624 100                |           |         |     |
|--------------------------|-----------|---------|-----|
| fe80::%wm0/64            |           | link#1  | UC  |
| - - - - -                | WM0       |         |     |
| fe80::20c:29ff:fecf:c884 |           | link#1  | UHI |
| - - - - -                | lo0       |         |     |
| fe80::%lo0/64            |           | fe80::1 | U   |
| - - - - -                | lo0       |         |     |
| fe80::1                  |           | lo0     | UHI |
| - - - - -                | lo0       |         |     |
| ff01:1::/32              |           | link#1  | UC  |
| - - - - -                | WM0       |         |     |
| ff01:2::/32              |           | ::1     | UC  |
| - - - - -                | 33624 lo0 |         |     |
| ff02::%wm0/32            |           | link#1  | UC  |
| - - - - -                | WM0       |         |     |
| ff02::%lo0/32            |           | ::1     | UC  |
| - - - - -                | 33624 lo0 |         |     |

- Probamos la opción 4 en el menú (Detalles de la interfaz de red). Podemos ver los detalles de eth1.

```
4
Detalles de interfaces de red:
Ingrese el nombre de la interfaz (ej: wm0, vio0):
wm0
wm0: flags=0x8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        capabilities=0x2bf80<TSO4,IP4CSUM_Rx,IP4CSUM_Tx,TCP4CSUM_Rx>
        capabilities=0x2bf80<TCP4CSUM_Tx,UDP4CSUM_Rx,UDP4CSUM_Tx,TCP6CSUM_Tx>
        capabilities=0x2bf80<UDP6CSUM_Tx>
        enabled=0
        ec_capabilities=0x7<VLAN_MTU,VLAN_HWTAGGING,JUMBO_MTU>
        ec_enabled=0x2<VLAN_HWTAGGING>
        address: 00:0c:29:c8:84
        media: Ethernet autoselect (1000baseT full-duplex, master)
        status: active
        inet6 fe80::20c:29ff:fecf:c884%wm0/64 flags 0 scopeid 0x1
                inet 10.2.78.35/16 broadcast 10.2.255.255 flags 0
```

- Probamos la opción 5 en el menú (Conexiones UDP y conexiones TCP).

```
Conexiones UDP:
udp:
    95 datagrams received
    0 with incomplete header
    0 with bad data length field
    0 with bad checksum
    28 dropped due to no socket
    18 broadcast/multicast datagrams dropped due to no socket
    0 dropped due to full socket buffers
    49 delivered
    74 PCB hash misses
    475 datagrams output
```

## SLACKWARE

- Utilizamos netstat para mostrar información de red como puertos, servicios, estadísticas y conexiones UDP y TCP. También usamos route para ver la tabla

de enrutamiento y ethtool para verificar la información de las interfaces de red.

```
GNU nano 6.0                               Shell11.8.sh
#!/bin/bash
while true;do
    echo "Menu de informacion de red"
    echo "1) Mostrar conexiones de red y puertos en escucha"
    echo "2) Mostrar uso de red"
    echo "3) Mostrar tabla de enrutamiento"
    echo "4) Mostrar detalles de la interfaz de red"
    echo "5) Mostrar conexiones con un protocolo especificos"
    echo "6) Salir"

    read option
    case $option in
        1)
            echo "Mostrando puertos en escucha y servicios:"
            netstat -tunl
            ;;
        2)
            echo "Estadisticas de red"
            netstat -i
            ;;
        3)
            echo "Tabla de enrutamiento"
            route -n
            ;;
        4)
            echo "Detalles de la interfaz de red"
            ethtool eth1
            ;;
        5)
            echo "Seleccione el protocolo:"
            echo "1. UDP"
            ;;
        *)
            echo "Seleccione el protocolo:"
            echo "1. UDP"
            echo "2. TCP"
            read protocol
            case $protocol in
                1)
                    echo "Conexiones TCP"
                    netstat -tan
                    ;;
                2)
                    echo "Conexiones UDP"
                    netstat -uan
                    ;;
                *)
                    echo "Opcion invalida de protocolo"
                    ;;
            esac
            ;;
        6)
            echo "cerrando programa"
            break
            ;;
        *)
            echo "Opcion Invalida"
            ;;
    esac
done
```

- Probamos la opción 1 del menú (Conexiones de red y puertos en escucha). Como podemos ver, se muestran los protocolos de la capa de transporte junto con los puertos en los que están escuchando, además de las direcciones IP permitidas.

|      |   |                           |           |        |
|------|---|---------------------------|-----------|--------|
| tcp  | 0 | 0 0.0.0.0:22              | 0.0.0.0:* | LISTEN |
| tcp  | 0 | 0 127.0.0.1:953           | 0.0.0.0:* | LISTEN |
| tcp  | 0 | 0 127.0.0.1:53            | 0.0.0.0:* | LISTEN |
| tcp  | 0 | 0 127.0.0.1:53            | 0.0.0.0:* | LISTEN |
| tcp  | 0 | 0 127.0.0.1:631           | 0.0.0.0:* | LISTEN |
| tcp  | 0 | 0 0.0.0.0:5432            | 0.0.0.0:* | LISTEN |
| tcp6 | 0 | 0 ::1:953                 | :::*      | LISTEN |
| tcp6 | 0 | 0 ::1:22                  | :::*      | LISTEN |
| tcp6 | 0 | 0 fe80::20c:29ff:feb5::53 | :::*      | LISTEN |
| tcp6 | 0 | 0 fe80::20c:29ff:feb5::53 | :::*      | LISTEN |
| tcp6 | 0 | 0 :::5432                 | :::*      | LISTEN |
| tcp6 | 0 | 0 ::1:53                  | :::*      | LISTEN |
| tcp6 | 0 | 0 ::1:53                  | :::*      | LISTEN |
| tcp6 | 0 | 0 ::1:631                 | :::*      | LISTEN |
| tcp6 | 0 | 0 :::8080                 | :::*      | LISTEN |
| udp  | 0 | 0 10.2.78.36:53           | 0.0.0.0:* |        |
| udp  | 0 | 0 10.2.78.36:53           | 0.0.0.0:* |        |
| udp  | 0 | 0 127.0.0.1:53            | 0.0.0.0:* |        |
| udp  | 0 | 0 127.0.0.1:53            | 0.0.0.0:* |        |
| udp  | 0 | 0 10.2.78.36:123          | 0.0.0.0:* |        |
| udp  | 0 | 0 127.0.0.1:123           | 0.0.0.0:* |        |
| udp  | 0 | 0 0.0.0.0:123             | 0.0.0.0:* |        |
| udp6 | 0 | 0 ::1:53                  | :::*      |        |
| udp6 | 0 | 0 ::1:53                  | :::*      |        |
| udp6 | 0 | 0 fe80::20c:29ff:feb5::53 | :::*      |        |
| udp6 | 0 | 0 fe80::20c:29ff:feb5::53 | :::*      |        |
| udp6 | 0 | 0 fe80::20c:29ff:feb5:123 | :::*      |        |
| udp6 | 0 | 0 ::1:123                 | :::*      |        |
| udp6 | 0 | 0 :::123                  | :::*      |        |

- Probamos la opción 2 del menú (Uso de la red). Podemos ver información sobre eth1 (interfaz de red) y lo (loopback o interfaz de bucle local).

```
Estadísticas de red
Kernel Interface table
Iface      MTU   RX-OK RX-ERR RX-DRP RX-OUR      TX-OK TX-ERR TX-DRP TX-OUR Flg
eth0      1500    1463     0     0       512      0     0       0 BMRU
lo        65536   1214     0     0       1214      0     0       0 LRU
Menu de información de red
```

- Probamos la opción 3 del menú (Tabla de enrutamiento). Podemos ver las rutas presentes en la tabla de enrutamiento.

```
Tabla de enrutamiento
Kernel IP routing table
Destination  Gateway      Genmask      Flags Metric Ref  Use Iface
0.0.0.0      10.2.65.1   0.0.0.0      UG     0      0      0 eth0
10.2.0.0     0.0.0.0      255.255.0.0  U      0      0      0 eth0
127.0.0.0    0.0.0.0      255.0.0.0    U      0      0      0 lo
Menu de información de red
```

- Probamos la opción 4 del menú (Detalles de la interfaz de red). Podemos ver los detalles de eth1.

```
4
Detalles de la interfaz de red
netlink error: no device matches name (offset 24)
netlink error: No such device
netlink error: no device matches name (offset 24)
netlink error: No such device
netlink error: no device matches name (offset 24)
netlink error: No such device
netlink error: no device matches name (offset 24)
netlink error: No such device
netlink error: no device matches name (offset 24)
netlink error: No such device
netlink error: no device matches name (offset 24)
netlink error: No such device
No data available
Menu de información de red
```

- Probamos la opción 5 del menú (Conexiones UDP y conexiones TCP).

```

tcp      0      0 0.0.0.0:80          0.0.0.0:*
tcp      0      0 0.0.0.0:22          0.0.0.0:*
tcp      0      0 127.0.0.1:953       0.0.0.0:*
tcp      0      0 127.0.0.1:53          0.0.0.0:*
tcp      0      0 127.0.0.1:53          0.0.0.0:*
tcp      0      0 127.0.0.1:631       0.0.0.0:*
tcp      0      0 0.0.0.0:5432       0.0.0.0:*
tcp6     0      0 ::1:953           :::*
tcp6     0      0 ::22              :::*
tcp6     0      0 fe80::20c:29ff:feb5::53  :::*
tcp6     0      0 fe80::20c:29ff:feb5::53  :::*
tcp6     0      0 :::5432          :::*
tcp6     0      0 ::1:53           :::*
tcp6     0      0 ::1:53           :::*
tcp6     0      0 ::1:631          :::*
tcp6     0      0 :::8080          :::*
Menu de informacion de red
1) Mostrar conexiones de red y puertos en escucha
2) Mostrar uso de red
3) Mostrar tabla de enrutamiento
4) Mostrar detalles de la interfaz de red
5) Mostrar conexiones con un protocolo especificos
6) Salir
2
Estadisticas de red
Kernel Interface table
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0      1500    1468    0      0 0      530     0      0      0 BMRU
lo       65536   1247    0      0 0      1247    0      0      0 LRU

```

- b. Additionally, create a Shell program that indicates (yes or no) whether a port is open and which service is running on that port.

## NETBSD

- Shell

```

GNU nano 8.3                               Shell1.9.sh
#!/bin/sh
echo "Ingrese el puerto"
read port
# Verificar si el puerto estAa abierto
result=$(ss -tuln | grep ":$port ")
if [ -z "$result" ]; then
    echo "No hay ningAa servicio ejecutAandose en el puerto $port"
else
    service=$(echo "$result" | awk '{print $1, $5}')
    echo "Servicio ejecutAandose en el puerto $port : $service"
fi

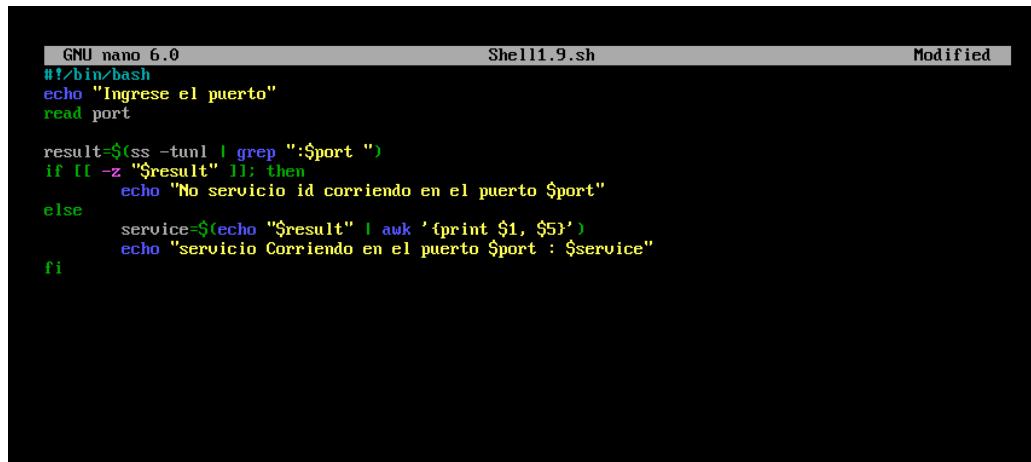
```

- Probamos el script ingresando el puerto 80 y puertos desconocidos.

```
Servicio ejecutándose en el puerto 80 :
Juanito# sh Shell1.9.sh
Ingrrese el puerto
8080
Shell1.9.sh: ss: not found
Shell1.9.sh: [[: not found
Servicio ejecutándose en el puerto 8080 :
Juanito# █
```

## SLACKWARE

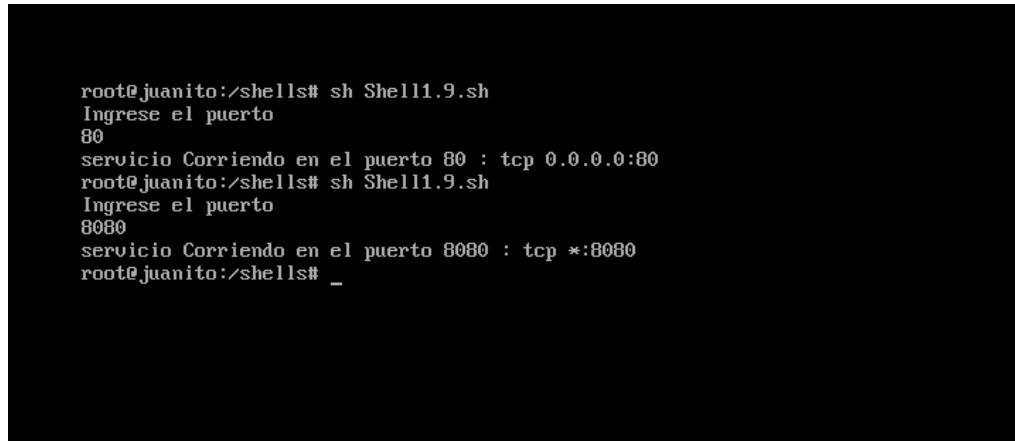
- Además, creamos un programa en Shell que indica (sí o no) si un puerto está abierto y qué servicio se está ejecutando en ese puerto. Utilizamos el comando `telnet`.



```
GNU nano 6.0                               Shell1.9.sh                                Modified
#!/bin/bash
echo "Ingrrese el puerto"
read port

result=$(ss -tunl | grep ":$port ")
if [ ! -z "$result" ]; then
    echo "No servicio id corriendo en el puerto $port"
else
    service=$(echo "$result" | awk '{print $1, $5}')
    echo "servicio Corriendo en el puerto $port : $service"
fi
```

- Probamos el script ingresando el puerto 80 y puertos desconocidos.



```
root@juanito:/shells# sh Shell1.9.sh
Ingrrese el puerto
80
servicio Corriendo en el puerto 80 : tcp 0.0.0.0:80
root@juanito:/shells# sh Shell1.9.sh
Ingrrese el puerto
8080
servicio Corriendo en el puerto 8080 : tcp *:8080
root@juanito:/shells# _
```

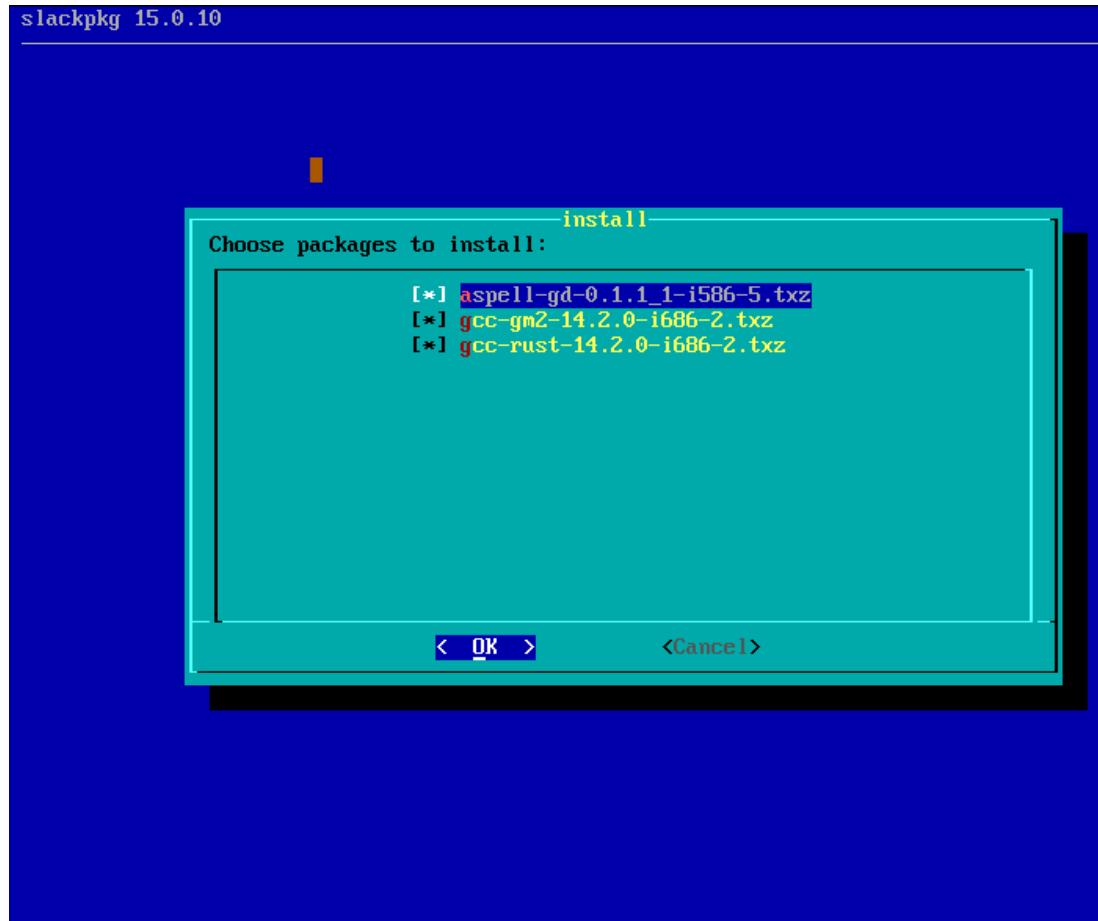
## 2. Network Management

On the other hand, part of an organization's technological platform is the monitoring server. Through it, administrators can check the status of the network equipment they manage. These platform management tools enable remote monitoring of disk space, CPU usage, network performance, memory usage, installed software, among other aspects of the network devices

Each student must install a server to monitor the network (Groups of 2 students: one student should use Slackware and the other NetBSD. Groups of 3 students: additionally, the third student will use Windows) and document their work. They must enable monitoring of all their virtual machines and display it in real-time.

- a. NetBSD
- b. Slackware
  - i. Instalamos las dependencias necesarias para el uso del sistema de monitoreo de Nagios con el comando slackpkg install
  - ii. Seleccionamos la opción “ok”

```
root@juanito:~# slackpkg install gcc make glibc perl apache php php-apache gd libp
```



- iii. Instalamos el sistema de monitoreo de nagios mediante el comando wget  
<https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.1.tar.gz>

```
root@juanito:~# cd /usr/src
root@juanito:/usr/src# ls
linux@ linux-5.15.19/
root@juanito:/usr/src# wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.1.tar.gz
--2025-04-24 13:40:59-- https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.1.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 64:ff9b::2d4f:3178
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11540195 (11M) [application/x-gzip]
Saving to: "nagios-4.5.1.tar.gz"

nagios-4.5.1.tar.gz      100%[=====] 11.00M  2.99MB/s  in 3.9s

2025-04-24 13:41:04 (2.79 MB/s) - "nagios-4.5.1.tar.gz" saved [11540195/11540195]

root@juanito:/usr/src#
```

- iv. Descomprimimos el archive descargado mediante el comando tar -zxvf “nombre\_archivo” en este caso el nombre es nagios-4.5.1.tar.gz

```
root@juanito:/usr/src# tar -zxvf nagios-4.5.1.tar.gz_
```

```
nagios-4.5.1/tap/tests/plan/too-many-tests/test.pl
nagios-4.5.1/tap/tests/plan/too-many-tests/test.t
nagios-4.5.1/tap/tests/skip/
nagios-4.5.1/tap/tests/skip/Makefile.am
nagios-4.5.1/tap/tests/skip/Makefile.in
nagios-4.5.1/tap/tests/skip/test.c
nagios-4.5.1/tap/tests/skip/test.pl
nagios-4.5.1/tap/tests/skip/test.t
nagios-4.5.1/tap/tests/todo/
nagios-4.5.1/tap/tests/todo/Makefile.am
nagios-4.5.1/tap/tests/todo/Makefile.in
nagios-4.5.1/tap/tests/todo/test.c
nagios-4.5.1/tap/tests/todo/test.pl
nagios-4.5.1/tap/tests/todo/test.t
nagios-4.5.1/test/
nagios-4.5.1/test/test-downtime.pl
nagios-4.5.1/update-version
nagios-4.5.1/worker/
nagios-4.5.1/worker/Makefile.in
nagios-4.5.1/worker/ping/
nagios-4.5.1/worker/ping/.gitignore
nagios-4.5.1/worker/ping/Makefile.in
nagios-4.5.1/worker/ping/worker-ping.c
nagios-4.5.1/xdata/
nagios-4.5.1/xdata/.gitignore
nagios-4.5.1/xdata/Makefile.in
nagios-4.5.1/xdata/xcddefault.c
nagios-4.5.1/xdata/xcddefault.h
nagios-4.5.1/xdata/xodtemplate.c
nagios-4.5.1/xdata/xodtemplate.h
nagios-4.5.1/xdata/xpddefault.c
nagios-4.5.1/xdata/xpddefault.h
nagios-4.5.1/xdata/xrddefault.c
nagios-4.5.1/xdata/xrddefault.h
nagios-4.5.1/xdata/xsddefault.c
nagios-4.5.1/xdata/xsddefault.h
```

- v. Usamos el comando ./configure --with-command-group=nagcmd para especificar opciones o parámetros personalizados al script configure durante la preparación para compilar un software.

```

root@juanito:/usr/src/nagios-4.5.1# ./configure --with-command-group=nagcmd

config.status: creating t-tap/Makefile
config.status: creating include/config.h
config.status: creating lib/snprintf.h
config.status: creating lib/iobroker.h

Creating sample config files in sample-config/ ...

*** Configuration summary for nagios 4.5.1 2024-02-28 ***:

General Options:
-----
    Nagios executable: nagios
    Nagios user/group: nagios,nagios
    Command user/group: nagios,nagcmd
        Event Broker: yes
    Install ${prefix}: /usr/local/nagios
    Install ${includedir}: /usr/local/nagios/include/nagios
        Lock file: /run/nagios.lock
    Check result directory: /usr/local/nagios/var/spool/checkresults
        Init directory: /etc/init.d
    Apache conf.d directory: /etc/httpd/conf.d
        Mail program: /usr/bin/mail
        Host OS: linux-gnu
    IOBroker Method: epoll

Web Interface Options:
-----
        HTML URL: http://localhost/nagios/
        CGI URL: http://localhost/nagios/cgi-bin/
    Traceroute (used by WAP): /usr/bin/traceroute

Review the options above for accuracy. If they look okay,
type 'make all' to compile the main program and CGIs.

```

- vi. Usamos el comando make all para compilar todo el código fuente del programa según las configuraciones definidas previamente.

```

root@juanito:/usr/src/nagios-4.5.1# make all

```

```
make install-exfoliation
- This installs the Exfoliation theme for the Nagios
  web interface
```

```
make install-classicui
- This installs the classic theme for the Nagios
  web interface
```

\*\*\* Support Notes \*\*\*\*\*

If you have questions about configuring or running Nagios,  
please make sure that you:

- Look at the sample config files
- Read the documentation on the Nagios Library at:  
<https://library.nagios.com>

before you post a question to one of the mailing lists.  
Also make sure to include pertinent information that could  
help others help you. This might include:

- What version of Nagios you are using
- What version of the plugins you are using
- Relevant snippets from your config files
- Relevant error messages from the Nagios log file

For more information on obtaining support for Nagios, visit:

<https://support.nagios.com>

\*\*\*\*\*  
Enjoy.

- vii. Usamos useradd nagios para crear un usuario dedicado que ejecutará el servicio Nagios de forma segura.

```
root@juanito:/usr/src/nagios-4.5.1# useradd nagios
root@juanito:/usr/src/nagios-4.5.1# groupadd nagcmd
root@juanito:/usr/src/nagios-4.5.1# usermod -a -G nagcmd nagios
root@juanito:/usr/src/nagios-4.5.1#
```

```
root@juanito:/usr/src/nagios-4.5.1# usermod -a -G nagcmd apache
```

- viii. Usamos make install para instalar Nagios en el sistema después de compilarlo.

```
root@juanito:/usr/src/nagios-4.5.1# make install
```

```

/usr/bin/ginstall -c -m 664 -o nagios -g nagios bootstrap-3.3.7/css/bootstrap-theme.min.css /usr/local/nagios/share/bootstrap-3.3.7/css
/usr/bin/ginstall -c -m 664 -o nagios -g nagios d3/d3.min.js /usr/local/nagios/share/d3
/usr/bin/ginstall -c -m 664 -o nagios -g nagios ./spin/spin.min.js /usr/local/nagios/share/spin
make[1]: Leaving directory '/usr/src/nagios-4.5.1/html'
make install-exfoliation
make[1]: Entering directory '/usr/src/nagios-4.5.1'

*** Exfoliation theme installed ***
NOTE: Use 'make install-classicui' to revert to classic Nagios theme

make[1]: Leaving directory '/usr/src/nagios-4.5.1'
make install-basic
make[1]: Entering directory '/usr/src/nagios-4.5.1'
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/libexec
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/archives
/usr/bin/ginstall -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/spool/checkresults
chmod g+s /usr/local/nagios/var/spool/checkresults

*** Main program, CGIs and HTML files installed ***

You can continue with installing Nagios as follows (type 'make' without any arguments for a list of all possible options):

make install-init
  - This installs the init script in /etc/init.d

make install-commandmode
  - This installs and configures permissions on the directory for holding the external command file

make install-config
  - This installs sample config files in /usr/local/nagios/etc

make[1]: Leaving directory '/usr/src/nagios-4.5.1'

```

- ix. Usamos make install-config para instalar los archivos de configuración predeterminados de Nagios.

```

root@juanito:/usr/src/nagios-4.5.1# make install-config
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg /usr/local/nagios/etc/cgi.cfg
/usr/bin/ginstall -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg

*** Config files installed ***

Remember, these are *SAMPLE* config files. You'll need to read the documentation for more information on how to actually define services, hosts, etc. to fit your particular needs.

```

- x. Usamos make install-init para instalar el script que permite iniciar, detener o reiniciar Nagios como un servicio del sistema.

```
root@juanito:/usr/src/nagios-4.5.1# make install-init  
/usr/bin/ginstall -c -m 755 -o root -g root /etc/init.d  
/usr/bin/ginstall -c -m 755 -o root -g root startup/default-init /etc/init.d/nagios  
root@juanito:/usr/src/nagios-4.5.1# make install-commandmode  
/usr/bin/ginstall -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw  
chmod g+s /usr/local/nagios/var/rw  
  
*** External command directory configured ***
```

```
make: *** [makefile:551: install-webconf] Error 1  
root@juanito:/usr/src/nagios-4.5.1# mkdir -p /etc/httpd/conf.d  
  
root@juanito:/usr/src/nagios-4.5.1# make install-webconf  
/usr/bin/ginstall -c -m 644 sample-config/httpd.conf /etc/httpd/conf.d/nagios.conf  
if [ 0 -eq 1 ]; then \  
    ln -s /etc/httpd/conf.d/nagios.conf /etc/apache2/sites-enabled/nagios.conf  
fi  
  
*** Nagios/Apache conf file installed ***  
root@juanito:/usr/src/nagios-4.5.1# _
```

- xi. Usamos mkdir -p /etc/httpd/conf.d para crear el directorio donde se almacenarán archivos de configuración adicionales para el servidor web Apache, si no existe.

```
make: *** [makefile:551: install-webconf] Error 1  
root@juanito:/usr/src/nagios-4.5.1# mkdir -p /etc/httpd/conf.d
```

- xii. Usamos make install-webconf para instalar el archivo de configuración web de Nagios en Apache, permitiendo el acceso a la interfaz web de Nagios.

```
root@juanito:/usr/src/nagios-4.5.1# make install-webconf  
/usr/bin/ginstall -c -m 644 sample-config/httpd.conf /etc/httpd/conf.d/nagios.conf  
if [ 0 -eq 1 ]; then \  
    ln -s /etc/httpd/conf.d/nagios.conf /etc/apache2/sites-enabled/nagios.conf  
fi  
  
*** Nagios/Apache conf file installed ***  
root@juanito:/usr/src/nagios-4.5.1# _
```

- xiii. Usamos wget https://nagios-plugins.org/download/nagios-plugins-2.4.6.tar.gz para descargar desde Internet el archivo comprimido que contiene los plugins de Nagios, necesarios para que Nagios pueda realizar tareas de monitoreo.
- xiv. Luego usamos tar xzf nagios-plugins-2.4.6.tar.gz para descomprimir y extraer ese archivo, preparándolo para su instalación.

```

root@juanito:/# wget https://nagios-plugins.org/download/nagios-plugins-2.4.6.tar.gz
2025-04-28 19:21:16-- https://nagios-plugins.org/download/nagios-plugins-2.4.6.tar.gz
Connecting to nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
HTTP request sent, awaiting response... 200 OK
Length: 2751770 (2.6M) [application/x-gzip]
Saving to: "nagios-plugins-2.4.6.tar.gz"

nagios-plugins-2.4.6.tar 100%[=====] 2.62M 3.39MB/s in 0s

2025-04-28 19:21:17 (3.39 MB/s) - "nagios-plugins-2.4.6.tar.gz" saved [2751770/2751770]

root@juanito:/# tar xzf nagios-plugins-2.4.6.tar.gz

```

- xv. Usamos ./configure --with-nagios-user=nagios --with-nagios-group=nagios para preparar la compilación de los plugins de Nagios, indicándole que se instalarán para ser usados por el usuario y grupo nagios.

```

root@juanito:/nagios-plugins-2.4.6# ./configure --with-nagios-user=nagios --with-nagios-group=nagios
-
```

- xvi. Usamos make para compilar el código fuente de los plugins de Nagios que fue preparado por el comando ./configure.

```

root@juanito:/nagios-plugins-2.4.6# make

```

```

chmod +x check_file_age
sed -e 's|[@]NP_VERSION[@]|2.4.6|g' -e 's|[@]TRUSTED_PATH[@]|/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin|g' -e 's|[@]PERL[@]|/usr/bin/perl|g' -e 's|[@]libexecdir[@]|/usr/libexec|g' check_ssl_validity.pl > check_ssl_validity
chmod +x check_ssl_validity
make[2]: Leaving directory '/nagios-plugins-2.4.6/plugins-scripts'
Making all in plugins-root
make[2]: Entering directory '/nagios-plugins-2.4.6/plugins-root'
gcc -DLOCALEDIR=\"/usr/local/nagios/share/locale\" -DHAVE_CONFIG_H -I. -I.. -I... -I... -I...
I../intl -I../plugins -I/usr/include -DNP_VERSION='2.4.6' -g -O2 -MT check_dhcp.o -MD
ps/check_dhcp.Tpo -c -o check_dhcp.o check_dhcp.c
mv -f .deps/check_dhcp.Tpo .deps/check_dhcp.Po
/bin/sh ./libtool --tag=CC --mode=link gcc -DNP_VERSION='2.4.6' -g -O2 -L. -o check_dhcp
k_dhcp.o ../plugins/netutils.o ../plugins/utils.o ../../lib/libnagiosplug.a ../../gl/libgnu.a
olv -lssl -lcrypto -lpthread -ldl
libtool: link: gcc -DNP_VERSION='2.4.6' -g -O2 -o check_dhcp check_dhcp.o ../plugins/
/plugins/utils.o -L. ../../lib/libnagiosplug.a ../../gl/libgnu.a -lssl -lresolv -lssl -lcrypt
-ldl
gcc -DLOCALEDIR=\"/usr/local/nagios/share/locale\" -DHAVE_CONFIG_H -I. -I.. -I... -I...
I../intl -I../plugins -I/usr/include -DNP_VERSION='2.4.6' -g -O2 -MT check_icmp.o -MD
ps/check_icmp.Tpo -c -o check_icmp.o check_icmp.c
mv -f .deps/check_icmp.Tpo .deps/check_icmp.Po
/bin/sh ./libtool --tag=CC --mode=link gcc -DNP_VERSION='2.4.6' -g -O2 -L. -o check_icmp
k_icmp.o ../plugins/netutils.o ../plugins/utils.o ../../lib/libnagiosplug.a ../../gl/libgnu.a
olv -lssl -lresolv -lssl -lcrypto -lpthread -ldl
libtool: link: gcc -DNP_VERSION='2.4.6' -g -O2 -o check_icmp check_icmp.o ../plugins/
/plugins/utils.o -L. ../../lib/libnagiosplug.a ../../gl/libgnu.a -lssl -lresolv -lssl -lcrypt
-ldl
make[2]: Leaving directory '/nagios-plugins-2.4.6/plugins-root'
Making all in po
make[2]: Entering directory '/nagios-plugins-2.4.6/po'
make[2]: Nothing to be done for 'all'.
make[2]: Leaving directory '/nagios-plugins-2.4.6/po'
make[2]: Entering directory '/nagios-plugins-2.4.6'
make[2]: Leaving directory '/nagios-plugins-2.4.6'
make[1]: Leaving directory '/nagios-plugins-2.4.6'

```

- xvii. Usamos make install para instalar los plugins ya compilados en el sistema, normalmente en /usr/local/nagios/libexec.

```
root@juanito:/nagios-plugins-2.4.6# make install_
```

```
Making install in plugins-root
make[1]: Entering directory '/nagios-plugins-2.4.6/plugins-root'
make[2]: Entering directory '/nagios-plugins-2.4.6/plugins-root'
/usr/bin/ginstall -c -o nagios -g nagios check_dhcp /usr/local/nagios/libexec/check_dhcp
chown root /usr/local/nagios/libexec/check_dhcp
chmod ug=rx,u+s /usr/local/nagios/libexec/check_dhcp
/usr/bin/ginstall -c -o nagios -g nagios check_icmp /usr/local/nagios/libexec/check_icmp
chown root /usr/local/nagios/libexec/check_icmp
chmod ug=rx,u+s /usr/local/nagios/libexec/check_icmp
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/nagios-plugins-2.4.6/plugins-root'
make[1]: Leaving directory '/nagios-plugins-2.4.6/plugins-root'
Making install in po
make[1]: Entering directory '/nagios-plugins-2.4.6/po'
/usr/bin/mkdir -p /usr/local/nagios/share
installing fr.gmo as /usr/local/nagios/share/locale/fr/LC_MESSAGES/nagios-plugins.mo
installing de.gmo as /usr/local/nagios/share/locale/de/LC_MESSAGES/nagios-plugins.mo
if test "nagios-plugins" = "gettext-tools"; then \
  /usr/bin/mkdir -p /usr/local/nagios/share/gettext/po; \
  for file in Makefile.in.in remove-potcdate.sin  Makevars.template; do \
    /usr/bin/ginstall -c -o nagios -g nagios -m 644 ./${file} \
      /usr/local/nagios/share/gettext/po/${file}; \
  done; \
  for file in Makevars; do \
    rm -f /usr/local/nagios/share/gettext/po/${file}; \
  done; \
else \
  :; \
fi
make[1]: Leaving directory '/nagios-plugins-2.4.6/po'
make[1]: Entering directory '/nagios-plugins-2.4.6'
make[2]: Entering directory '/nagios-plugins-2.4.6'
make[2]: Nothing to be done for 'install-exec-am'.
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/nagios-plugins-2.4.6'
make[1]: Leaving directory '/nagios-plugins-2.4.6'
root@juanito:/nagios-plugins-2.4.6#
```

- xviii. Ahora accedemos a la ruta “/etc/httpd/httpd.conf” y agregamos las siguientes configuraciones:

1. Incluir configuración de Nagios
  - a. Línea
    - i. include /etc/httpd/conf.d/nagios.conf
      1. Usamos esta línea para que Apache cargue la configuración necesaria para acceder a la interfaz web de Nagios.
  2. Habilitar soporte PHP en Apache
    - a. Líneas
      - i. LoadModule php7\_module lib/httpd/modules/libphp7.so
      - ii. AddType application/x-httpd-php .php

1. Usamos estas líneas para que Apache pueda interpretar archivos con extensión .php, necesarios para ejecutar interfaces como la de Nagios.
3. Ejecutar archivos PHP y definir archivo por defecto
4. Líneas
  - a. <IfModule php7\_module>
  - b. AddHandler php7-script .php
  - c. DirectoryIndex index.php index.html
  - d. </IfModule>
    - i. Usamos este bloque para que Apache ejecute archivos .php y, al ingresar a un directorio, cargue primero index.php si existe.

```
root@juanito:/nagios-plugins-2.4.6# nano /etc/httpd/httpd.conf_

GNU nano 6.0                               /etc/httpd/httpd.conf
#Include /etc/httpd/extra/httpd-ssl.conf
#
## Note: The following must must be present to support
##       starting without SSL on platforms with no /dev/random equivalent
##       but a statically compiled-in mod_ssl.
##
<IfModule ssl_module>
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
</IfModule>

## Uncomment the following line to enable PHP:
##
#Include /etc/httpd/mod_php.conf

## Uncomment the following lines (and mod_dav above) to enable svn support:
##
##LoadModule dav_svn_module lib/httpd/modules/mod_dav_svn.so
##LoadModule authz_svn_module lib/httpd/modules/mod_authz_svn.so


include /etc/httpd/conf.d/nagios.conf
LoadModule php7_module lib/httpd/modules/libphp7.so
AddType application/x-httpd-php7 .php

<IfModule php7_module>
    AddHandler php7-script .php
    DirectoryIndex index.php index.html
</IfModule>

^G Help      ^O Write Out   ^W Where Is   ^K Cut        ^T Execute   ^C Location   M-U
^X Exit      ^R Read File   ^N Replace   ^U Paste      ^J Justify   ^- Go To Line M-E
```

- xix. Encendemos el servicio mediante el comando /usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg

```
root@juanito:/nagios-plugins-2.4.6# ./usr/local/nagios/bin/nagios ./usr/local/nagios/etc/nagios.cfg

Nagios Core 4.5.1
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-02-28
License: GPL

Website: https://www.nagios.org
Nagios 4.5.1 starting... (PID=13984)
Local time is Mon Apr 28 19:41:04 -05 2025
wproc: Successfully registered manager as @wproc with query handler
wproc: Registry request: name=Core Worker 13985;pid=13985
wproc: Registry request: name=Core Worker 13988;pid=13988
wproc: Registry request: name=Core Worker 13987;pid=13987
wproc: Registry request: name=Core Worker 13986;pid=13986
Successfully launched command file worker with pid 13989
-
```

- xx. Usamos el comando sudo htpasswd /usr/local/nagios/etc/htpasswd.users nagiosadmin para crear o actualizar la contraseña del usuario nagiosadmin en el archivo de autenticación HTTP de Nagios. ( Cambiamos htpasswd por la contraseña que consideremos, en este caso será root123)

```
root@juanito:/nagios-plugins-2.4.6# sudo htpasswd ./usr/local/nagios/etc/htpasswd.users nagiosadmin
```

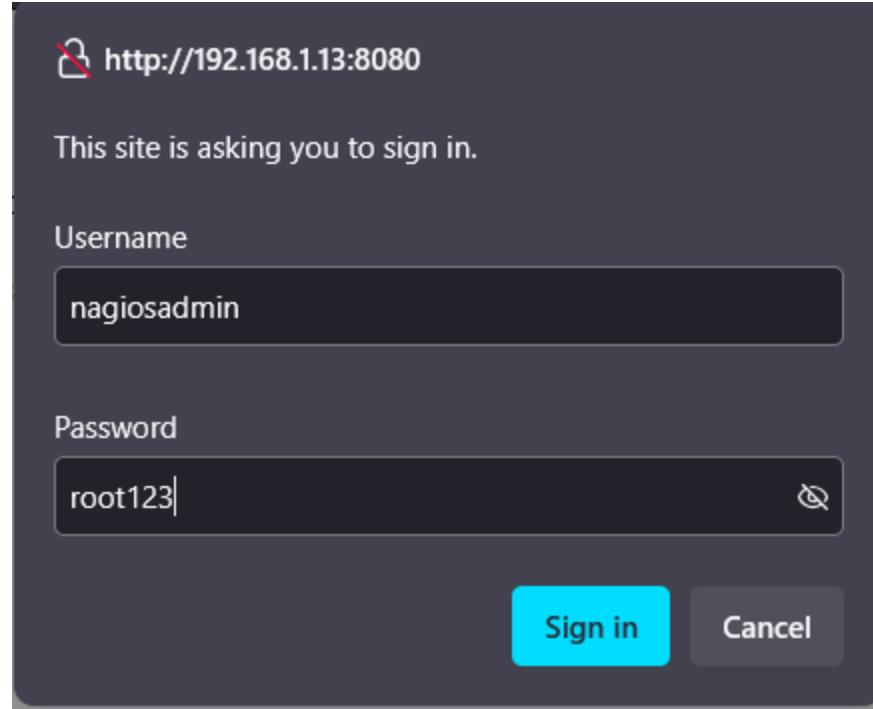
- xxi. Accedemos a la ruta mediante el comando nano /usr/local/nagios/etc/cgi.cfg y agregamos nagios\_check\_command=/usr/local/nagios/libexec/check\_nagios para definir el comando que verifica si el servicio de Nagios está activo desde la interfaz web.

```
root@juanito:/etc/httpd# nano ./usr/local/nagios/etc/cgi.cgi_
```

```
nagios_check_command=/usr/local/nagios/libexec/check_nagios
```

- xxii. Ingresamos en un navegador web con la ip de nuestra maquina y el puerto de apache. En este caso: http://192.168.1.13:8080/nagios

- xxiii. Al entrar nos pedirá la contraseña y el usuario en los cuales en este caso son: nagiosadmin , root123 los cuales fueron configurados anteriormente



- xxiv. Luego de ingresar la contraseña nos mostrará el servidor de monitoreo de nagios

- xxv. Accedemos a la ruta /usr/local/nagios/etc/hosts.cfg mediante el comando nano /usr/local/nagios/etc/hosts.cfg para registrar un nuevo host que será monitoreado por Nagios (en este caso netBSD), agregando las siguientes líneas que definen sus características, comportamiento de monitoreo y notificaciones:

1. define host {

- a. Inicia el bloque de configuración para un nuevo host.
- 2. use linux-server
  - a. Hereda parámetros comunes desde la plantilla linux-server, evitando repetir configuraciones estándar.
- 3. host\_name netbsd-client
  - a. Asigna un nombre interno al host, utilizado por Nagios en su interfaz y registros.
- 4. address 192.168.1.17
  - a. Dirección IP del host (netBSD), que Nagios utilizará para ejecutar los chequeos.
- 5. max\_check\_attempts 5
  - a. Especifica que Nagios intentará verificar el estado del host hasta 5 veces antes de marcarlo como caído.
- 6. check\_interval 5
  - a. Define que, si el host está en estado normal, se revisará cada 5 minutos.
- 7. retry\_interval 1
  - a. Si hay una falla, se reintenta la verificación cada 1 minuto hasta llegar a los 5 intentos.
- 8. check\_period 24x7
  - a. Indica que el host debe ser monitoreado en todo momento, sin restricción de horario.
- 9. notification\_interval 30
  - a. Si el host sigue en estado problemático, se enviará una notificación cada 30 minutos.
- 10. notification\_period 24x7
  - a. Las notificaciones pueden enviarse en cualquier momento del día o la semana.

```
root@juanito:/# nano /usr/local/nagios/etc/hosts.cfg
```

```

GNU nano 6.0                               /usr/local/nagios/etc/hosts.cfg
define host {
    use linux-server
    host_name      netbsd-client
    address 192.168.1.17
    max_check_attempts 5
    check_interval 5
    retry_interval 1
    check_period   24x7
    notification_interval 30
    notification_period 24x7
}

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo
^X Exit ^R Read File ^H Replace ^U Paste ^J Justify ^L Go To Line M-E Redo

```

- xxvi. Accedemos a la ruta /usr/local/nagios/etc/objects/commands.cfg mediante el comando nano /usr/local/nagios/etc/objects/commands.cfg para definir nuevos comandos personalizados de verificación en Nagios, agregando los siguientes bloques define command que permiten supervisar el uso de CPU y disco de un host remoto mediante el plugin check\_nrpe:

1. Primer bloque:

```

define command {

    command_name  check_nrpe_cpu
    command_line  /usr/local/nagios/libexec/check_nrpe -H
$HOSTADDRESS$ -c check_load
}

```

- a. Usamos este bloque para crear el comando check\_nrpe\_cpu, que revisa la carga del CPU del host remoto usando NRPE.

2. Segundo bloque:

```
define command {  
    command_name  check_nrpe_disk  
    command_line  /usr/local/nagios/libexec/check_nrpe -H  
$HOSTADDRESS$ -c check_disk  
}
```

- a. Usamos este bloque para crear el comando check\_nrpe\_disk, que verifica el uso del disco del host remoto mediante NRPE.

```
root@juanito:/# nano /usr/local/nagios/etc/objects/commands.cfg
```

```
GNU nano 6.0          /usr/local/nagios/etc/objects/commands.cfg      Modified  
#  
# SAMPLE PERFORMANCE DATA COMMANDS  
#  
# These are sample performance data commands that can be used to send performance  
# data output to two text files (one for hosts, another for services). If you  
# plan on simply writing performance data out to a file, consider using the  
# host_perfdata_file and service_perfdata_file options in the main config file.  
#  
#  
define command {  
    command_name  process-host-perfdata  
    command_line  /usr/bin/printf "%b" "$LASTHOSTCHECK$\t$HOSTNAME$\t$HOSTSTATE$\t$HOSTATTEMPT"  
}  
  
define command {  
    command_name  process-service-perfdata  
    command_line  /usr/bin/printf "%b" "$LASTSERVICECHECK$\t$HOSTNAME$\t$SERVICEDESC$\t$SERVICESTATE$"  
}  
  
define command {  
    command_name  check_nrpe_cpu  
    command_line  /usr/local/nagios/libexec/check_nrpe -H $HOSTADDRESS$ -c check_load  
}  
define command {  
    command_name  check_nrpe_disk  
    command_line  /usr/local/nagios/libexec/check_nrpe -H $HOSTADDRESS$ -c check_disk  
}  
  
^G Help      ^O Write Out  ^W Where Is   ^K Cut      ^T Execute   ^C Location  M-U Undo  
^X Exit      ^R Read File  ^H Replace   ^U Paste    ^J Justify   ^- Go To Line M-E Redo
```

- xxvii. Accedemos a la ruta /usr/local/nagios/etc/services.cfg mediante el comando nano /usr/local/nagios/etc/services.cfg para definir los servicios que Nagios debe monitorear en el host remoto netbsd-client.

1. Primer servicio

- a. Define un servicio que verifica la conectividad con netbsd-client mediante PING.
  - b. Usa el comando check\_ping con límites de advertencia (100ms, 20% pérdida) y críticos (500ms, 60% pérdida).
  - c. Se intentan 3 chequeos como máximo, con intervalos de 1 minuto entre ellos.
2. Segundo servicio
    - a. Define un servicio que monitorea el uso de CPU en netbsd-client usando NRPE.
    - b. Ejecuta el comando remoto check\_load.
    - c. Se repite cada 1 minuto y se intenta hasta 3 veces en caso de error.
  3. Tercer servicio
    - a. Define un servicio que verifica el uso del disco del host netbsd-client mediante NRPE.
    - b. Ejecuta el comando check\_disk en el host remoto.
    - c. Mismos intervalos y número de intentos que el servicio anterior.

```
root@juanito:/# nano /usr/local/nagios/etc/services.cfg
```

```

GNU nano 6.0                               /usr/local/nagios/etc/services.cfg
define service {
    use generic-service
    host_name      netbsd-client
    service_description PING
    check_command  check_ping!100.0,20%;!500.0,60%
    check_interval 1
    retry_interval 1
    max_check_attempts     3
}
define service {
    use generic-service
    host_name      netbsd-client
    service_description CPU
    check_command  check_nrpe_cpu!$HOSTADDRESS$
    check_interval 1
    retry_interval 1
    max_check_attempts     3
}
define service {
    use generic-service
    host_name      netbsd-client
    service_description Disk Usage
    check_command  check_nrpe_disk!$HOSTADDRESS$
    check_interval 1
    retry_interval 1
    max_check_attempts     3
}

^G Help      ^O Write Out   ^W Where Is   ^K Cut          ^T Execute   ^C Location   ^U Undo
^X Exit      ^R Read File   ^P Replace    ^U Paste        ^J Justify   ^L Go To Line ^E Redo

```

- xxviii. Accedemos a la ruta /usr/local/nagios/etc/nagios.cfg para agregar la configuración de los archivos hosts.cfg y services.cfg con las líneas:
1. cfg\_file=/usr/local/nagios/etc/objects/hosts.cfg
  2. cfg\_file=/usr/local/nagios/etc/services.cfg
    - a. Estas líneas hacen que Nagios cargue las definiciones de los hosts que se van a monitorear y los servicios que se deben supervisar en cada host. Esto permite que, al iniciar Nagios, ya sepa qué máquinas y qué aspectos de esas máquinas debe vigilar.

```
root@juanito:/# nano /usr/local/nagios/etc/nagios.cfg_
```

```

GNU nano 6.0                               /usr/local/nagios/etc/nagios.cfg

# LOG FILE
# This is the main log file where service and host events are logged
# for historical purposes. This should be the first option specified
# in the config file!!!

log_file=/usr/local/nagios/var/nagios.log


# OBJECT CONFIGURATION FILE(S)
# These are the object configuration files in which you define hosts,
# host groups, contacts, contact groups, services, etc.
# You can split your object definitions across several config files
# if you wish (as shown below), or keep them all in a single config file.

# You can specify individual object config files as shown below:
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg
cfg_file=/usr/local/nagios/etc/hosts.cfg
cfg_file=/usr/local/nagios/etc/services.cfg

# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
#cfg_file=/usr/local/nagios/etc/objects/windows.cfg

# Definitions for monitoring a router/switch
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg

^G Help      ^D Write Out   ^W Where Is   ^K Cut      ^T Execute   ^C Location   M-U Undo
^X Exit      ^R Read File    ^E Replace    ^U Paste    ^J Justify   ^L Go To Line M-E Redo

```

- xxix. Usamos el comando wget para descargar el archivo comprimido nrpe-4.0.3.tar.gz desde el repositorio oficial de GitHub, que contiene el código fuente del complemento NRPE para Nagios. Esto debido a que los servicios de monitoreo que implementamos anteriormente usan NRPE.
- xxx. Luego, usamos el comando tar -xzf nrpe-4.0.3.tar.gz para descomprimir el archivo y extraer su contenido en una carpeta llamada nrpe-4.0.3.
- xxxi. Finalmente, usamos el comando cd nrpe-4.0.3 para ingresar a ese directorio y continuar con el proceso de compilación e instalación del complemento.

```

root@juanito:~# wget https://github.com/NagiosEnterprises/nrpe/releases/download/nrpe-4.0.3/nrpe-4.0.3.tar.gz
--2025-04-28 22:57:36--  https://github.com/NagiosEnterprises/nrpe/releases/download/nrpe-4.0.3/nrpe-4.0.3.tar.gz
Resolving github.com (github.com)... 140.82.113.3
Connecting to github.com (github.com)|140.82.113.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/16119653/32805a80-896b-11ea-905e-de706cd33ff1?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction/2F20250429%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250429T035556Z&X-Amz-Expires=300&X-Amz-Signature=7f8079170ba25b29badc73dc52b56319c936d332cb24282b0daef5b1ff22160b&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dnrpe-4.0.3.tar.gz&response-content-type=application%2Foctet-stream [following]
--2025-04-28 22:57:36--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/16119653/32805a80-896b-11ea-905e-de706cd33ff1?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction/2F20250429%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250429T035556Z&X-Amz-Expires=300&X-Amz-Signature=7f8079170ba25b29badc73dc52b56319c936d332cb24282b0daef5b1ff22160b&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dnrpe-4.0.3.tar.gz&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.109.133, 185.199.111.133, 185.199.110.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 524160 (512K) [application/octet-stream]
Saving to: "nrpe-4.0.3.tar.gz"

nrpe-4.0.3.tar.gz      100%[=====] 511.88K --.-KB/s   in 0.04s

2025-04-28 22:57:37 (11.6 MB/s) - "nrpe-4.0.3.tar.gz" saved [524160/524160]

root@juanito:~# tar -xzf nrpe-4.0.3.tar.gz
root@juanito:~# cd nrpe-4.0.3
root@juanito:~/nrpe-4.0.3#

```

- xxxii. Usamos el comando ./configure --enable-command-args para configurar la instalación de NRPE y permitir que el servidor Nagios envíe argumentos personalizados a los comandos ejecutados remotamente.

```

root@juanito:~/nrpe-4.0.3# ./configure --enable-command-args

```

- xxxiii. Usamos el comando make check\_nrpe para compilar el plugin check\_nrpe, el cual es necesario para que el servidor Nagios pueda verificar el estado de los servicios en los equipos remotos a través del protocolo NRPE.

```

root@juanito:~/nrpe-4.0.3# make check_nrpe
cd ./src;/ make
make[1]: Entering directory '/root/nrpe-4.0.3/src'
gcc -g -O2 -I/usr/include/krb5 -DHAVE_CONFIG_H -I ../../include -I ../../include -o nrpe ./nrpe.c .
ls.c ./acl.c -lssl -lcrypto -lssl -lwrap
gcc -g -O2 -I/usr/include/krb5 -DHAVE_CONFIG_H -I ../../include -I ../../include -o check_nrpe ./check_nrpe.c ./utils.c -lssl -lcrypto -lssl
make[1]: Leaving directory '/root/nrpe-4.0.3/src'
root@juanito:~/nrpe-4.0.3#

```

- xxxiv. Luego, usamos el comando make install-plugin para instalar el plugin compilado en el sistema, dejándolo disponible en la ruta correspondiente para que Nagios pueda utilizarlo al ejecutar las comprobaciones remotas.

```

root@juanito:~/nrpe-4.0.3# make install-plugin
cd ./src;/ make install-plugin
make[1]: Entering directory '/root/nrpe-4.0.3/src'
/usr/bin/ginstall -c -m 755 -d /usr/local/nagios/bin
/usr/bin/ginstall -c -m 755 ../../uninstall /usr/local/nagios/bin/nrpe-uninstall
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/libexec
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/libexec
/usr/bin/ginstall -c -m 775 -o nagios -g nagios check_nrpe /usr/local/nagios/libexec
make[1]: Leaving directory '/root/nrpe-4.0.3/src'
root@juanito:~/nrpe-4.0.3#

```

c. NETBSD

- i. Creamos el grupo nagios con el comando groupadd nagios para definir una entidad de control de permisos que será utilizada por los procesos relacionados con Nagios.
- ii. Luego, creamos el usuario nagios con el comando useradd -g nagios nagios para que pertenezca a ese grupo, lo cual permite que los servicios de monitoreo (NRPE) se ejecuten bajo este usuario de forma segura y controlada, tanto en el servidor de monitoreo como en la máquina que será monitoreada( en este caso netbsd)

```
Juanito# groupadd nagios
Juanito# useradd -g nagios nagios
useradd: Warning: home directory '/home/nagios' doesn't exist, and -m was not specified
Juanito#
```

- iii. Usamos el comando pkgin install nagios-plugins para instalar los plugins de Nagios, que son esenciales para ejecutar los chequeos de monitoreo de servicios, recursos y conectividad en los sistemas supervisados por Nagios.

```
Juanito# pkgin install nagios-plugins
calculating dependencies...done.

3 packages to refresh:
p5-JSON-4.10nb2 p5-Parse-Yapp-1.21nb7 samba-4.21.3

1 package to upgrade:
perl-5.40.2

1 package to install:
nagios-plugins-2.4.3nb2

0 to remove, 3 to refresh, 1 to upgrade, 1 to install
28M to download, 2299K of additional disk space will be used

proceed ? [Y/n] y
[1/5] nagios-plugins-2.4.3nb2.tgz          100%  366KB 366.5KB/s  00:00
[2/5] p5-JSON-4.10nb2.tgz                 100%   44KB  44.3KB/s  00:00
[3/5] p5-Parse-Yapp-1.21nb7.tgz          100%   29KB  29.4KB/s  00:00
[4/5] perl-5.40.2.tgz                   73%  9216KB  2.1MB/s  00:01 100%
```

- iv. A

```
Juanito# pkgin install wget autoconf automake gettext openssl libmcrypt
calculating dependencies...done.

9 packages to install:
autoconf-2.72 automake-1.16.5nb3 gettext-0.22.5 gettext-asprintf-0.22.5
gettext-m4-0.22.5 gettext-tools-0.22.5 libmcrypt-2.5.8 m4-1.4.19nb1
openssl-3.4.1

0 to remove, 0 to refresh, 0 to upgrade, 9 to install
9011K to download, 48M of additional disk space will be used

proceed ? [Y/n] ■
```

- v. Usamos el comando wget

<https://github.com/NagiosEnterprises/nrpe/archive/nrpe-4.1.0.tar.gz> para

descargar el código fuente de NRPE versión 4.1.0, luego tar xzf nrpe-4.1.0.tar.gz para descomprimir y extraer su contenido en el directorio actual, dejándolo listo para su compilación.

```
Juanito# wget https://github.com/NagiosEnterprises/nrpe/archive/nrpe-4.1.0.tar.gz
--2025-04-28 21:53:40--  https://github.com/NagiosEnterprises/nrpe/archive/nrpe-4.1.0.tar.gz
Resolving github.com (github.com)... 140.82.112.4
Connecting to github.com (github.com):140.82.112.4:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/NagiosEnterprises/nrpe/tar.gz/refs/tags/nrpe-4.1.0 [following]
--2025-04-28 21:53:41--  https://codeload.github.com/NagiosEnterprises/nrpe/tar.gz/refs/tags/nrpe-4.1.0
Resolving codeload.github.com (codeload.github.com)... 140.82.112.10
Connecting to codeload.github.com (codeload.github.com):140.82.112.10:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/x-gzip]
Saving to: 'nrpe-4.1.0.tar.gz'

nrpe-4.1.0.tar.gz      [ =>          1 515.88K  523KB/s   in 1.0s

2025-04-28 21:53:42 (523 KB/s) - 'nrpe-4.1.0.tar.gz' saved [528263]

Juanito#
```

```
Juanito# tar xzf nrpe-4.1.0.tar.gz
```

- vi. Usamos el comando cd nrpe-nrpe-4.1.0 para entrar al directorio del código fuente extraído de NRPE.
- vii. Luego ejecutamos./configure --enable-command-args --with-ssl=/usr/local/bin --with-ssl-inc=/usr/local/include/openssl --with-ssl-lib=/usr/local/lib para preparar la compilación del software, especificando soporte para argumentos de comandos y rutas personalizadas de OpenSSL necesarias para compilar con soporte de cifrado seguro.

```
Juanito# cd nrpe-nrpe-4.1.0
```

```
Juanito# ./configure --enable-command-args --with-ssl=/usr/local/bin --with-ssl-inc=/usr/local/include/openssl --with-ssl-lib=/usr/local/lib
```

```

config.status: creating startup/solaris-inetd.xml
config.status: creating startup/tmpfile.conf
config.status: creating startup/upstart-init
config.status: creating startup/rh-upstart-init
config.status: creating include/common.h
config.status: creating include/config.h

*** Configuration summary for nrpe 4.1.0 2022-07-18 ***:

General Options:
-----
NRPE port:      5666
NRPE user:      nagios
NRPE group:     nagios
Nagios user:    nagios
Nagios group:   nagios

Review the options above for accuracy. If they look okay,
type 'make all' to compile the NRPE daemon and client
or type 'make' to get a list of make options.

```

- viii. Usamos el comando make all para compilar el código fuente de NRPE, es decir, transformar el código en archivos ejecutables listos para ser utilizados. Este paso toma el resultado de la configuración previa (./configure) y construye los binarios necesarios.

```

Juanito# make all
cd ./src/; make
gcc -g -O2 -I/usr/include/krb5 -I/usr/include/openssl -DHAVE_CONFIG_H -I .
ude -I ./../include -o nrpe ./nrpe.c ./utils.c ./acl.c -L/usr/lib -lssl -
o
gcc -g -O2 -I/usr/include/krb5 -I/usr/include/openssl -DHAVE_CONFIG_H -I .
ude -I ./../include -o check_nrpe ./check_nrpe.c ./utils.c -L/usr/lib -l
rypto

*** Compile finished ***

You can now continue with the installation or upgrade process.

Read the PDF documentation (docs/NRPE.pdf) for information on the next
steps you should take to complete the installation or upgrade.

```

- ix. Luego, ejecutamos make install para instalar los binarios compilados en el sistema, copiándolos a los directorios correspondientes del sistema para que puedan ser ejecutados desde cualquier parte.

```
Juanito# make install
cd ./src/; make install
Make install-plugin
/usr/bin/install -c -M 755 -d /usr/local/nagios/bin
/usr/bin/install -c -M 755 ../uninstall /usr/local/nagios/bin/nrpe-uninstal
/usr/bin/install -c -M 775 -o nagios -g nagios -d /usr/local/nagios/libexec
/usr/bin/install -c -M 775 -o nagios -g nagios -d /usr/local/nagios/libexec
/usr/bin/install -c -M 775 -o nagios -g nagios check_nrpe /usr/local/nagios
xec
Make install-daemon
/usr/bin/install -c -M 755 -d /usr/local/nagios/bin
/usr/bin/install -c -M 755 ../uninstall /usr/local/nagios/bin/nrpe-uninstal
/usr/bin/install -c -M 755 nrpe /usr/local/nagios/bin
/usr/bin/install -c -M 755 -o nagios -g nagios -d /usr/local/nagios/var
Juanito#
```

- x. Despues, usamos make install-config para instalar los archivos de configuración por defecto de NRPE, como el archivo nrpe.cfg, que luego se podrá editar para definir qué servicios se van a monitorear remotamente.

```
Juanito# make install-config
/usr/bin/install -c -M 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -M 644 -o nagios -g nagios sample-config/nrpe.cfg /usr/l
/nagios/etc
Juanito#
```

- xi. Finalmente, el comando make install-init se utiliza para instalar el script de inicio del servicio NRPE, lo que permite que el servicio pueda iniciarse automáticamente al arrancar el sistema o manualmente mediante comandos como service o rc.d.

```
Juanito# make install-init
/usr/bin/install -c -M 755 startup/newbsd-init /etc/rc.d/nrpe
Juanito#
```

- xii. Accedemos a la ruta /usr/local/nagios/etc/nrpe.cfg para editar el archivo de configuración principal de NRPE, y cambiamos la línea
  1. dont\_blame\_nrpe=0 por dont\_blame\_nrpe=1.
- xiii. Con esto permitimos que el servidor Nagios pueda enviar argumentos a los comandos definidos en el archivo, lo cual es necesario para que los chequeos sean dinámicos y personalizados según las necesidades del monitoreo.

```
Juanito# nano /usr/local/nagios/etc/nrpe.cfg
```

```

GNU nano 8.3          /usr/local/nagios/etc/nrpe.cfg      Modif
# to specify arguments to commands that are executed. This option only works
# if the daemon was configured with the --enable-command-args configure script
# option.
#
# *** ENABLING THIS OPTION IS A SECURITY RISK! ***
# Read the SECURITY file for information on some of the security implications
# of enabling this variable.
#
# Values: 0=do not allow arguments, 1=allow command arguments
dont_blame_nrpe=1

# BASH COMMAND SUBSTITUTION
# This option determines whether or not the NRPE daemon will allow clients
# to specify arguments that contain bash command substitutions of the form
# ${(...)}. This option only works if the daemon was configured with both
# the --enable-command-args and --enable-bash-command-substitution configuration
# script options.
#
^G Help      ^O Write Out  ^F Where Is   ^K Cut        ^T Execute   ^C Location
^X Exit      ^R Read File  ^H Replace    ^U Paste      ^J Justify   ^/ Go To L

```

- xiv. Además, editamos la línea allowed\_hosts para agregar la dirección IP 192.168.1.13, que corresponde al servidor Nagios. Con esto autorizamos que ese servidor pueda conectarse al demonio NRPE de esta máquina, permitiendo así que realice chequeos remotos.

```

GNU nano 8.3          /usr/local/nagios/etc/nrpe.cfg      Modif
# ALLOWED HOST ADDRESSES
# This is an optional comma-delimited list of IP address or hostnames
# that are allowed to talk to the NRPE daemon. Network addresses with a bit
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not currently
# supported.
#
# Note: The daemon only does rudimentary checking of the client's IP
# address. I would highly recommend adding entries in your /etc/hosts.allow
# file to allow only the specified host to connect to the port
# you are running this daemon on.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd.
allowed_hosts=127.0.0.1, 192.168.1.13, ::1

# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE daemon will allow clients
# to specify arguments to commands that are executed. This option only works
# if the daemon was configured with the --enable-command-args configure script
#
^G Help      ^O Write Out  ^F Where Is   ^K Cut        ^T Execute   ^C Location
^X Exit      ^R Read File  ^H Replace    ^U Paste      ^J Justify   ^/ Go To L

```

- xv. También descomentamos las líneas command[check\_load] y command[check\_disk] en el archivo /usr/local/nagios/etc/nrpe.cfg. Con esto habilitamos comandos de monitoreo predefinidos que permiten al

servidor Nagios verificar la carga del sistema (check\_load) y el uso del disco (check\_disk) en la máquina remota.

```
GNU nano 8.3                               nrpe.cfg                                Modified

# The following examples allow user-supplied arguments and can
# only be used if the NRPE daemon was compiled with support for
# command arguments *AND* the dont_blame_nrpe directive in this
# config file is set to '1'. This poses a potential security risk, so
# make sure you read the SECURITY file before doing this.

### MISC SYSTEM METRICS ####
#command[check_users]=/usr/local/nagios/libexec/check_users $ARG1$
#command[check_load]=/usr/local/nagios/libexec/check_load -w 5.0,4.0,3.0 -c 1
#command[check_disk]=/usr/local/nagios/libexec/check_disk -w 20% -c 10% -p /
#command[check_swap]=/usr/local/nagios/libexec/check_swap $ARG1$
#command[check_cpu_stats]=/usr/local/nagios/libexec/check_cpu_stats.sh $ARG1$
#command[check_mem]=/usr/local/nagios/libexec/custom_check_mem -n $ARG1$

### GENERIC SERVICES ####
#command[check_init_service]=sudo /usr/local/nagios/libexec/check_init_service
#command[check_services]=-p $ARG1$

### SYSTEM UPDATES ####
#command[check_yum]=/usr/local/nagios/libexec/check_yum

[G] Help      [O] Write Out [F] Where Is   [E] Cut      [T] Execute    [C] Location
[X] Exit      [R] Read File [R] Replace    [U] Paste      [J] Justify    [/] Go To Line
```

- xvi. Usamos el comando cp /usr/pkg/libexec/nagios/\* /usr/local/nagios/libexec/ para copiar todos los archivos del directorio de plugins de Nagios desde su ubicación original a la ruta estándar de ejecución, con el fin de que Nagios pueda acceder y utilizar correctamente dichos plugins durante su operación.

```
Juanito# cp /usr/pkg/libexec/nagios/* /usr/local/nagios/libexec/
```

- xvii. Finalmente iniciamos el servicio de nrpe

```
Juanito# /etc/rc.d/nrpe start
Starting nrpe.
```

d. Pruebas

- En el apartado de servicios de nagios, verificamos que todo este correcto en la parte de netbsd Client (la que configuramos anteriormente con los 3 servicios) y

The screenshot shows the Nagios web interface at <http://Nagios:192.168.1.13>. The left sidebar includes links for General, Home, Documentation, Current Status, Host Groups, Services, Problems, Reports, and System. The main content displays the "Current Network Status" and "Service Status Totals". Below this is a detailed table of "Service Status Details For All Hosts", listing services like "localhost" and "netbsd-client" with their status, last check time, duration, and attempt counts.

## 1. Apartado CPU

This screenshot shows the service information for the "CPU" service on the "localhost" host. It includes sections for Service Information, Service State Information, and Service Commands. The service is currently OK with 0.00 load average over 1 minute. The service state is also OK with 0.00m downtime. In the service commands section, there are several options available for managing the service.

## 2. Apartado Disco

This screenshot shows the service information for the "netbsd-client" service, which is a disk service. It includes sections for Service Information, Service State Information, and Service Commands. The service is currently OK with 0.00 disk usage. The service state is also OK with 0.00m downtime. In the service commands section, there are several options available for managing the service.

## 3. Apartado Ping

**Service Information**

Last Updated: Mon Apr 28 23:02:56 -05 2025  
Nagios Core™ 4.5.1 - www.nagios.org  
Logged in as nagiosadmin

[View Information For This Host](#) | [View Host History](#) | [View Alert History For This Service](#) | [View Trends For This Service](#) | [View Log For This Service](#) | [View Availability Report For This Service](#) | [View Notifications For This Service](#)

**Service**  
**PING**  
On Host  
**netbsd-client**  
(netbsd-client)

Member of  
**No servicegroups.**

192.168.1.17

**Service State Information**

|                           |  |
|---------------------------|--|
| Current Status:           | OK (for 0d 0h 36m 28s)                                 |
| Status Information:       | PING OK - Packet loss = 0%, RTA = 0.83 ms              |
| Performance Data:         | rash=82600ms;100.00000;500.00000;0.00000 pl=0%;20;60;0 |
| Current Attempt:          | 1/1 (HARD state)                                       |
| Current Time:             | 04-28-2025 23:02:28                                    |
| Check Type:               | ACTIVE   |
| Check Latency / Duration: | 0.000 / 4.000 seconds                                  |
| Next Scheduled Check:     | 04-28-2025 23:03:28                                    |
| Last Check Outcome:       | OK (notification 0)                                    |
| Last Notification:        | N/A (notification 0)                                   |
| Is This Service Flapping? | NO (0.00% state change)                                |
| In Scheduled Downtime?    | NO   |
| Last Update:              | 04-28-2025 23:02:50 ( 0d 0h 0m 6s ago)                 |

Active Checks: **ENABLED**  
Passive Checks: **ENABLED**  
Observation: **ENABLED**  
Notifications: **ENABLED**  
Event Handler: **ENABLED**  
Flap Detection: **ENABLED**

**Service Commands**

- Disable active checks for this service
- Reschedule the next check of this service
- Submit passive check result for this service
- Stop accepting passive checks for this service
- Stop observing over this service
- Disable notifications for this service
- Send custom service notification
- Schedule downtime for this service
- Disable event handler for this service
- Disable flap detection for this service
- Clear flapping state for this service

Add a new comment

Entry Time Author Comment Comment ID Persistent Type Expires Actions

### e. Inicio automático

- Agregamos en slackware en la ruta /etc/rc.d/rc.local la línea /etc/rc.d/rc.nagios start para que el servidor de monitoreo nagios se inicie automáticamente

```
root@juanito:~# nano /etc/rc.d/rc.local_
GNU nano 6.0                                     /etc/rc.d/rc.local
#!/bin/bash
#
# /etc/rc.d/rc.local: Local system initialization script.
#
# Put any local startup commands in here. Also, if you have
# anything that needs to be run at shutdown time you can
# make an /etc/rc.d/rc.local_shutdown script and put those
# commands in there.

/etc/rc.d/rc.ntpd start
su - postgres -c "pg_ctl -D /var/lib/pgsql/data start"
/usr/local/nginx/sbin/nginx
/etc/rc.d/rc.nagios start_
```

### 3. Network Administration – Azure

- Lo primero es ingresar a nuestra cuenta de Azure

The screenshot shows the Microsoft Azure portal homepage. At the top, there's a navigation bar with links for 'Create a resource', 'Azure DevOps organizations', 'App Services', 'Subscriptions', 'All resources', 'Quickstart Center', 'Azure AI services', 'Kubernetes services', 'Virtual machines', and 'More services'. Below this is a 'Resources' section with a table of recent resources, including 'AppVids' (Static Web App), 'ReservasLaboratorios' (App Service), 'CVDS' (Resource group), 'ovsparcial' (App Service), and 'Reco' (Resource group). Further down are sections for 'Navigate' (Subscriptions, Resource groups, All resources, Dashboard) and 'Tools' (Microsoft Learn, Azure Monitor, Microsoft Defender for Cloud, Cost Management).

- En el navegador de Azure ponemos Education y seleccionamos la opcion de “Education”

This screenshot shows the Microsoft Azure portal with a search bar at the top containing the word 'Education'. The results page displays several categories under 'Services': 'Education' (selected), 'Intune for Education', and 'Connected Caches for Enterprise & Education'. Under 'Marketplace', there are links to 'SwiftrAccess (Azure App)', 'SAS® Premium Learning Subscription on Azure Marketplace', 'Just in Time Solution with Implementation Support for Healthcare', and 'UDS Education: workspace virtualization solution'. The 'Documentation' section includes links to 'A: Microsoft AI Bootcamp for Educators - Azure AI Fundamentals - Training', 'azure.mgmt.education package', 'azure.mgmt.education package', 'FieOps documentation - Cloud Computing', and 'Microsoft Entra ID'. On the right side, there are sections for 'Last Viewed' (links to '3 weeks ago') and 'Microsoft Entra ID' (listing users like 'LEGO EDUCATION', 'KELVIN EDUCATIONAL', and 'LATIN AMERICAN SCIENCE EDUCATION RESEARCH ASSOCIATION').

- Una vez estemos en “Education” seleccionamos “template” o en español “platilla”

This screenshot shows the 'Education | Templates' page. On the left, there's a sidebar with 'Overview', 'Learning resources', 'Roles', 'Software', 'Templates' (which is selected and highlighted in grey), 'GitHub', and 'Need help?'. The main content area is titled 'Azure Quickstart Templates' and describes how to deploy Azure resources using community-contributed templates. It features a search bar and filters for 'Scenario', 'Difficulty', 'Cost', and 'Roles'. A table lists 29 items, each with a brief description, scenario, difficulty level, roles required, and cost. Examples include 'Deploy a Cognitive Service Universal key' (Cognitive Services, Beginner, AI Engineer, \$), 'Deploy a Cognitive Services Translate API' (Cognitive Services, Beginner, AI Engineer, FREE), 'Provision a function app with source deployed from GitHub' (Azure Functions GitHub, Intermediate, Developer, \$\$), and 'Create an IoT Hub and a Device to Cloud Consumer' (Azure IoT, Beginner, Other, FREE).

- Buscamos “Web app deployment from GitHub” y seleccionamos la unica opcion que existe

The screenshot shows the Microsoft Azure portal's 'Education | Templates' section. A search bar at the top has 'Web app deployment from GitHub' typed into it. Below the search bar, there are filters for Scenario: All, Difficulty: All, Cost: All, and Roles: All. A table lists one item: 'Web App Deployment from GitHub' under the 'Azure Quickstart Templates' category. The table columns include Name, Scenario, Difficulty, Roles, and Cost. The item details are: Name: Web App Deployment from GitHub, Scenario: Azure App Service, Difficulty: Intermediate, Roles: Developer, and Cost: FREE.

- Desplegamos la plantilla

This screenshot is similar to the previous one, showing the 'Web App Deployment from GitHub' template selected. However, the right side of the screen displays more detailed information about the template. It includes a brief description: 'This template allows you to create an WebApp linked with a GitHub Repository linked.', a 'Browse learning' section with a diagram, and a 'Useful links' section with links to 'Getting Started', 'GitHub', 'Read Me', and 'Browse on GitHub'. At the bottom right, there are 'Deploy' and 'Cancel' buttons.

- Si no tenemos recurso creamos uno nuevo en el apartado de “Create new”, pero si ya tenemos un recurso creado simplemente lo seleccionamos, y finalmente seleccionamos “Review + Create”

## Custom deployment

Deploy from a custom template

New! Deployment Stacks let you manage the lifecycle of your deployments. Try it now →

### Template

 Customized template [↗](#)  
3 resources

 Edit template

 Edit parameters

 Visualize

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Azure for Students

Resource group \* ⓘ

(New) web-deploy

[Create new](#)

### Instance details

Region \* ⓘ

East US

Site Name ⓘ

[format('WebApp-{0}', uniqueString(resourceGroup().id))]

Location ⓘ

[resourceGroup().location]

Sku ⓘ

F1

Worker Size ⓘ

0

Repo URL ⓘ

<https://github.com/Azure-Samples/app-service-web-html-get-started....>

Branch ⓘ

master

[Previous](#)

[Next](#)

**Review + create**

- Finalmente le damos create

Basics    [Review + create](#)

**Summary**

Customized template  
3 resources

**Terms**

[Azure Marketplace Terms](#) | [Azure Marketplace](#)

By clicking "Create," I (a) agree to the applicable legal terms associated with the offering; (b) authorize Microsoft to charge or bill my current payment method for the fees associated with the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s); and (c) agree that, if the deployment involves 3rd party offerings, Microsoft may share my contact information and other details of such deployment with the publisher of that offering.

Microsoft assumes no responsibility for any actions performed by third-party templates and does not provide rights for third-party products or services. See the [Azure Marketplace Terms](#) for additional terms.

Deploying this template will create one or more Azure resources or Marketplace offerings. You acknowledge that you are responsible for reviewing the applicable pricing and legal terms associated with all resources and offerings deployed as part of this template. Prices and associated legal terms for any Marketplace offerings can be found in the [Azure Marketplace](#); both are subject to change at any time prior to deployment.

Neither subscription credits nor monetary commitment funds may be used to purchase non-Microsoft offerings. These purchases are billed separately.

If any Microsoft products are included in a Marketplace offering (e.g. Windows Server or SQL Server), such products are licensed by Microsoft and not by any third party.

**Basics**

|                |                    |
|----------------|--------------------|
| Subscription   | Azure for Students |
| Resource group | web-deploy         |

[Previous](#) [Next](#) [Create](#)

- Una vez este creado podemos seguir para saber que se desplego de manera correcta nos debe de aparecer de la siguiente manera y seleccionamos “Go to resource group”

Home > Microsoft.Template-20250424124549 | Overview ...

[Search](#) [Delete](#) [Cancel](#) [Redeploy](#) [Download](#) [Refresh](#)

**Overview**

Your deployment is complete

Deployment name: Microsoft.Template-20250424124549

Subscription: Azure for Students

Resource group: web-deploy

Start time: 4/24/2023, 1:24:54 PM

Correlation ID: 09afcc9b-41c1-4f7b-bc7d-0bda7bd37706

[Deployment details](#) [Next steps](#) [Go to resource group](#)

Give feedback [Tell us about your experience with deployment](#)

**Cost management**  
Get notified to stay within your budget and prevent unexpected charges on your bill.  
[Set up cost alerts >](#)

**Microsoft Defender for Cloud**  
Secure your apps and infrastructure.  
[Go to Microsoft Defender for Cloud >](#)

**Free Microsoft tutorials**  
[Start learning today >](#)

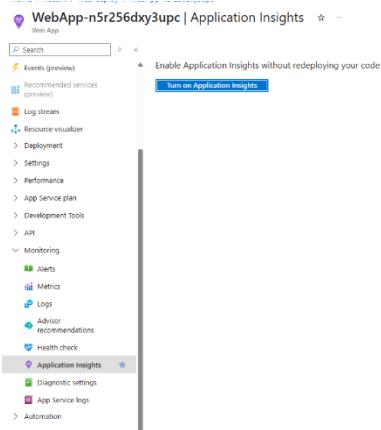
**Work with an expert**  
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.  
[Find an Azure expert >](#)

- Una vez estemos en el recurso seleccionamos la “app service”

- Una vez adentro seleccionamos “default domain” para poder acceder a la pagina



- En el “app service” en el apartado izquierdo seleccionamos supervisión o monitoring y ya adentro seleccionamos application insights y seleccionamos “turn on application insights ”



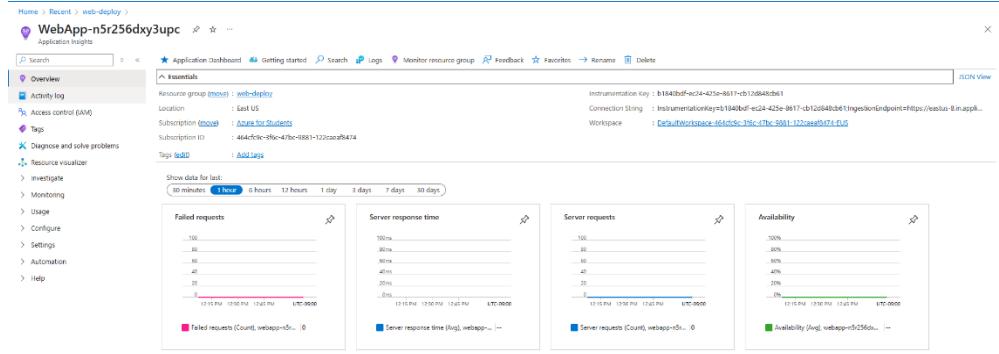
- Una vez habilitado lo aplicamos

The screenshot shows the 'Application Insights' blade for a web application. At the top, there's a message about connecting to an auto-created resource and collecting monitoring data. Below that, there's a section for creating a new resource, where a new resource name 'WebApp-n5r256dy3upc' is entered and the location is set to 'East US'. A modal dialog box is overlaid on the page, prompting the user to apply monitoring settings to the web app. The dialog contains a message about applying changes and restarting the site, with 'Yes' and 'No' buttons. The 'Yes' button is highlighted with a blue border.

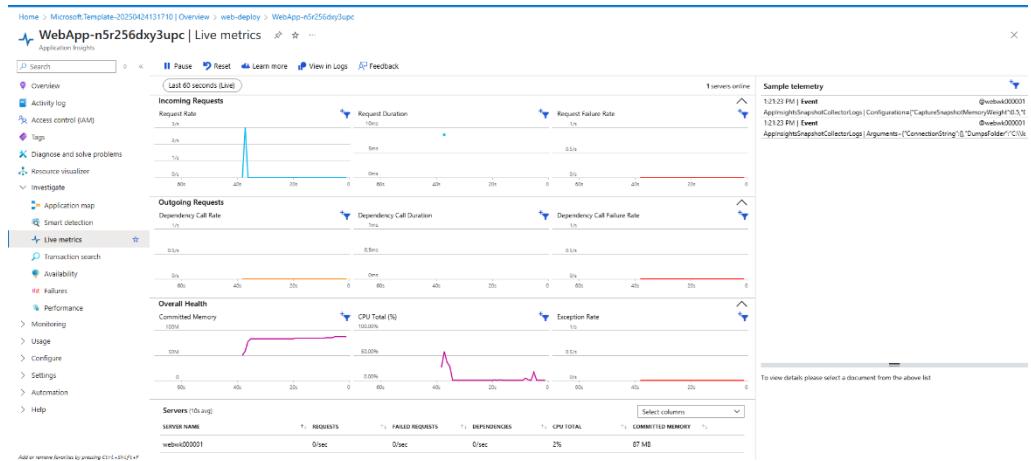
- Así se debería de ver cuando seleccionamos yes

The screenshot shows the same 'Application Insights' blade as before, but now it displays a green banner at the top stating 'Validation passed'. The modal dialog box from the previous screenshot has been closed. The rest of the page remains the same, showing the configuration options for the application insights resource.

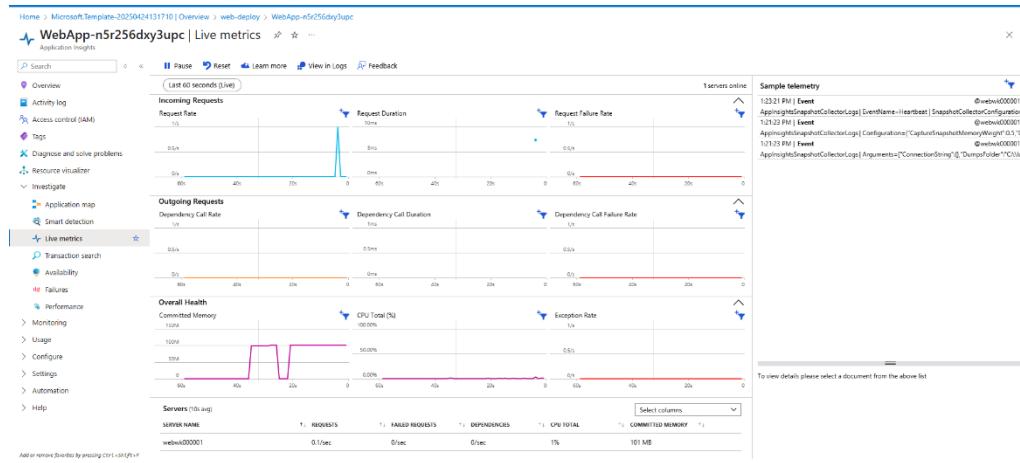
- Podemos ver la descripción general del servicio.



- En el menú de la izquierda, vamos a "Investigate"" y seleccionamos "Live Metric". Podemos observar en tiempo real el rendimiento y el comportamiento de nuestra aplicación.



1. Explore the "Overview" and "Live Metric" tabs in the "Investigate" section.
  - What do you observe when you refresh the website repeatedly?
    - Cada actualización genera nuevos datos y actualiza métricas en tiempo real, como las tasas de solicitudes, la duración de las solicitudes, la memoria comprometida y el total de CPU.



- What does each of the items displayed there represent?

### Incoming Request

- Request Rate: Número de solicitudes realizadas a la aplicación
- Request Duration: Representa el tiempo promedio que la aplicación tarda en procesar cada solicitud
- Request Failure Rate: Mide el porcentaje de solicitudes que no se completaron con éxito

### Outgoing Request

- Dependency Call Rate: Indica el número de solicitudes que la aplicación realiza a otros servicios externos, como bases de datos o APIs de terceros.
- Dependency Call Duration: Es el tiempo promedio que tardan en completarse las solicitudes enviadas a servicios externos.
- Dependency Call Failure Rate: Muestra el porcentaje de solicitudes a servicios externos que fallaron.

### Overall Health

- Committed Memory: Representa la cantidad de memoria que está asignada o comprometida para la aplicación en un momento dado.
- CPU Total (%): Indica el porcentaje de uso de CPU en el servidor o recurso que hospeda la aplicación.
- Exception Rate: Es el porcentaje de excepciones o errores inesperados que ocurren en la aplicación.

- What other functionalities does the "Application Insights" system offer for web services, databases, and other cloud-deployed systems?
  - o Application Insights ofrece múltiples herramientas para mejorar el rendimiento, la confiabilidad y la calidad de nuestras aplicaciones.
  - o Investigate
    - **Application dashboard:** Una evaluación rápida del estado y rendimiento de tu aplicación.
    - **Application map:** Una vista visual de la arquitectura de la aplicación y las interacciones entre sus componentes.
    - **Live metrics:** Un panel de análisis en tiempo real para obtener información sobre la actividad y el rendimiento de la aplicación.
    - **Transaction search:** Rastrea y diagnostica transacciones para identificar problemas y optimizar el rendimiento.
    - **Availability view:** Supervisa y prueba proactivamente la disponibilidad y capacidad de respuesta de los puntos finales de la aplicación.
    - **Failures view:** Identifica y analiza fallos en tu aplicación para minimizar el tiempo de inactividad.
    - **Performance view:** Revisa métricas de rendimiento de la aplicación e identifica posibles cuellos de botella.
  - o Monitoring
    - **Alerts:** Supervisa una amplia variedad de aspectos de tu aplicación y activa diversas acciones.
    - **Metrics:** Profundiza en los datos de métricas para entender patrones de uso y tendencias.
    - **Diagnostic settings:** Configura la exportación continua de registros de plataforma y métricas al destino de tu elección.
    - **Logs:** Recupera, consolida y analiza todos los datos recopilados en los Azure Monitoring Logs.

- **Workbooks:** Crea informes interactivos y paneles que visualizan los datos de monitoreo de la aplicación.

o Usage

- **Users, sessions, and events:** Determina cuándo, dónde y cómo interactúan los usuarios con tu aplicación web.
- **Funnels:** Analiza las tasas de conversión para identificar en qué etapas los usuarios avanzan o abandonan el embudo.
- **Flows:** Visualiza los recorridos de los usuarios en tu sitio para detectar áreas de alta interacción y puntos de salida.
- **Cohorts:** Agrupa a los usuarios según características comunes para facilitar la identificación de tendencias, la segmentación y la resolución de problemas de rendimiento.

o Code analysis

- **Profiler:** Captura, identifica y visualiza trazas de rendimiento de la aplicación.
- **Code optimizations:** Aprovecha la inteligencia artificial para crear aplicaciones más eficientes y de mayor calidad.
- **Snapshot debugger:** Recoge automáticamente capturas de depuración cuando ocurren excepciones en aplicaciones .NET.

– How might this functionality be beneficial in a corporate environment?

Ayuda a detectar problemas de forma temprana, optimizar el uso de recursos, mejorar la experiencia del usuario y facilitar una supervisión proactiva, lo que minimiza el tiempo de inactividad y mejora el rendimiento general.

– How does the network layer contribute to the transmission of data between the web app and clients in different geographic locations?

La capa de red se encarga del direccionamiento IP y el enrutamiento, asegurando que los paquetes de datos se transmitan a las ubicaciones correctas a través de diversas redes, optimizando la velocidad y la confiabilidad según la distancia geográfica y las condiciones de la red.

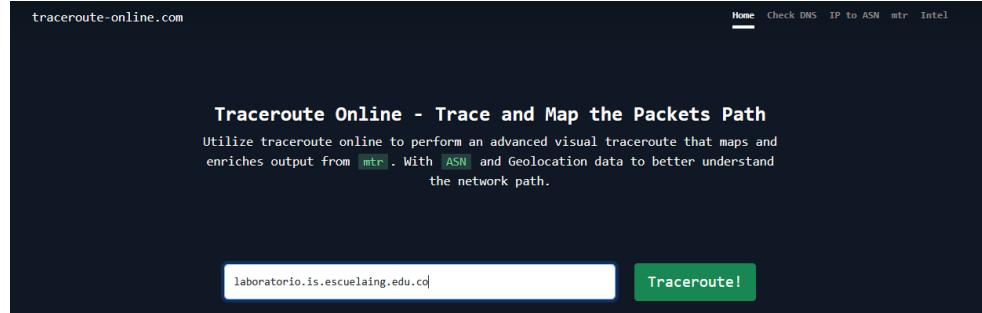
– In what ways do the application and transport layers ensure reliable data transfer and correct interpretation of web app responses?

- **Transport Layer:** Utiliza protocolos como TCP para garantizar que los paquetes se entreguen de forma confiable y en el orden correcto. Incluye verificación de errores y solicitudes de retransmisión.
- **Application Layer:** Convierte los datos hacia y desde formatos comprensibles por la aplicación y el usuario, asegurando la consistencia de los datos y su correcta interpretación entre dispositivos.

## EXPERIMENTS

### 1. Use of ICMP Messages

- Go to <https://traceroute-online.com/> and search for the Computer Science Laboratory page and the Stanford University page. Show the results
  - Entramos en la pagina y buscamos la pagina del laboratorio de la Escuela Colombiana de Ingenieria



- **Paso 1 al 3: Red local (IP privada)**

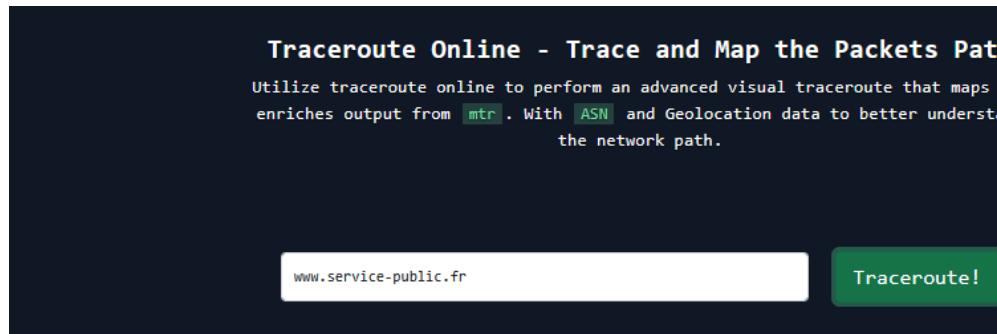
- IPs: 10.204.3.49, 10.204.35.11, 10.204.32.1
- Primeros saltos dentro de la red interna. Corresponde a routers y gateways locales.

- **Paso 4: Salida a Internet (Linode)**

- IP: 74.207.239.106
- Hostname: lo0-gw4.atl1.us.linode.com

- País: Estados Unidos
- El paquete entra a un proveedor de nube en Atlanta, saliendo oficialmente de la red local.
- **Paso 5: Nodo intermedio de Akamai**
  - IP: 23.203.144.36
  - Red: AKAMAI-ASN1
  - País: Estados Unidos
  - Transita por la red de Akamai.
- **Paso 6 al 8 – Backbone Arelion**
  - IPs: 62.115.141.98, 62.115.143.236, 62.115.118.197
  - Países: Francia y Estados Unidos
  - El paquete cruza la red internacional **Twelve99/Arelion**, conectando EE. UU. con Francia. Aún no entra a Colombia.
- **Paso 9: Nodo de Telia**
  - IP: 213.248.64.81
  - Hostname: sunset-ic-31313.ip.twelve99-cust.net
  - País: Suecia
  - Último salto internacional antes de entrar al proveedor colombiano.
- **Paso 10: Entrada a red colombiana (Liberty)**
  - IP: 69.63.64.88
  - País: Jamaica (pero enrutado hacia Colombia)
  - Aquí el tráfico entra por la red de Liberty, proveedor con presencia en América Latina.
- **Paso 11 y 12: Red Liberty y C&W**
  - IPs: 190.27.226.205, 190.144.193.81
  - País: Colombia
  - Los paquetes ya circulan por redes troncales nacionales.
- **Paso 13: Nodo intermedio (Bogotá)**
  - IP: 190.131.237.183
  - Red local colombiana (probablemente un punto de interconexión)
- **Paso 14: Salto dentro del destino**
  - IP: 45.239.88.78
  - El paquete ha llegado al rango de IP final, que pertenece a la Escuela Colombiana de Ingeniería.
- **Paso 15: Destino final alcanzado**
  - IP: 45.239.88.88
  - Hostname: escuela.colombiana.de.ingenieria

- País: Colombia
- El servidor final responde. La ruta se ha completado con éxito
- Using the tracert or traceroute command, search for a page in France and check the route
  - Para este caso vamos a buscar una pagina de información administrativa de Francia



- Nos arroja los siguientes pasos :

**Traceroute Online - Trace and Map the Packets Path**

Utilize traceroute online to perform an advanced visual traceroute that maps and enriches output from `mtr`. With `ASN` and Geolocation data to better understand the network path.

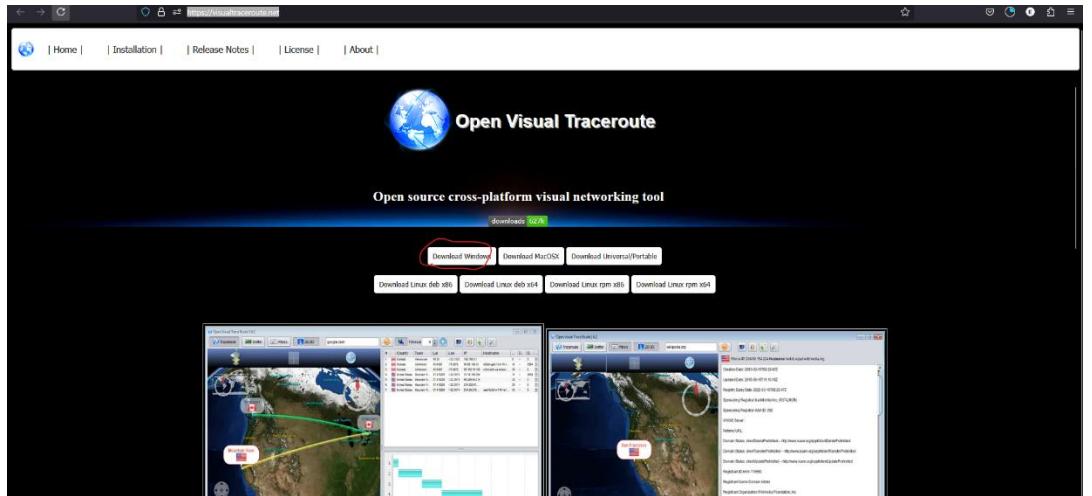
www.service-public.fr Traceroute!

| Hop | Hostname                                 | IP              | AS  | Network          | Country | Avg (ms) |
|-----|--|-----------------|---|------------------|---------|----------|
| 1   | 10.204.3.49                              | 10.204.3.49     |   |                  |         | ....     |
| 2   | 10.204.35.11                             | 10.204.35.11    |   |                  |         | ....     |
| 3   | 10.204.32.1                              | 10.204.32.1     |   |                  |         | ....     |
| 4   | lo0-0.gw4.atl1.us.linode.com             | 74.207.239.106  | AKAMAI-LINODE-AP Akamai Connected Cloud, SG | 74.207.224.0/20  | USA     | ....     |
| 5   | ae45.r22.atl01.ien.netarch.akamai.com    | 23.203.144.36   | AKAMAI-ASN1, NL                             | 23.203.144.0/20  | USA     | ....     |
| 6   | atl-b24-link.ip.twelve99.net             | 62.115.171.94   | TWELVE99 Arelion, fka Telia Carrier, SE     | 62.115.0.0/16    | FR      | ....     |
| 7   | atl-bb2-link.ip.twelve99.net             | 62.115.143.236  | TWELVE99 Arelion, fka Telia Carrier, SE     | 62.115.0.0/16    |         | ....     |
| 8   | ash-bb2-link.ip.twelve99.net             | 62.115.137.132  | TWELVE99 Arelion, fka Telia Carrier, SE     | 62.115.0.0/16    |         | ....     |
| 9   | ???                                      | ???             |   |                  |         | ....     |
| 10  | adm-bb2-link.ip.twelve99.net             | 62.115.137.4    | TWELVE99 Arelion, fka Telia Carrier, SE     | 62.115.0.0/16    | ....    | ....     |
| 11  | adm-b10-link.ip.twelve99.net             | 62.115.138.251  | TWELVE99 Arelion, fka Telia Carrier, SE     | 62.115.0.0/16    | FR      | ....     |
| 12  | retailint-ic-371383.ip.twelve99-cust.net | 213.155.129.251 | TWELVE99 Arelion, fka Telia Carrier, SE     | 213.155.128.0/19 | DE      | ....     |
| 13  | ???                                      | ???             |   |                  |         | ....     |
| 14  | ???                                      | ???             |   |                  |         | ....     |
| 15  | secure-www-sp-dila-prod.as8677.net       | 160.92.168.33   | ING-AS, FR                                  | 160.92.168.0/24  | FR      | ....     |

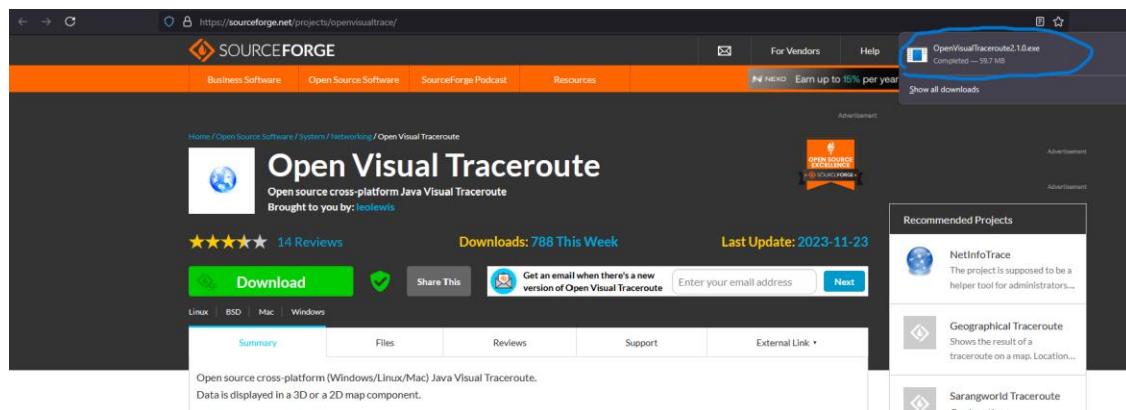
**ASN Information**

- **Paso 1 al 3: Red local**
  - IPs: 10.204.3.49, 10.204.35.11, 10.204.32.1
  - Saltos dentro de la red interna (IP privada). Es el recorrido que hace el paquete hasta salir del entorno local.
- **Paso 4: Salida a Internet (Proveedor: Linode)**
  - IP: 74.207.239.106
  - Hostname: lo0-gw4.atl1.us.linode.com
  - País: Estados Unidos
  - El paquete entra a la red del proveedor de servicios en la nube Linode, ubicado en Atlanta, EE. UU.
- **Paso 5: Nodo intermedio (Akamai)**
  - IP: 23.203.144.36
  - Hostname: ae4-5.r22.atl01.ien.netarch.akamai.com
  - País: Estados Unidos

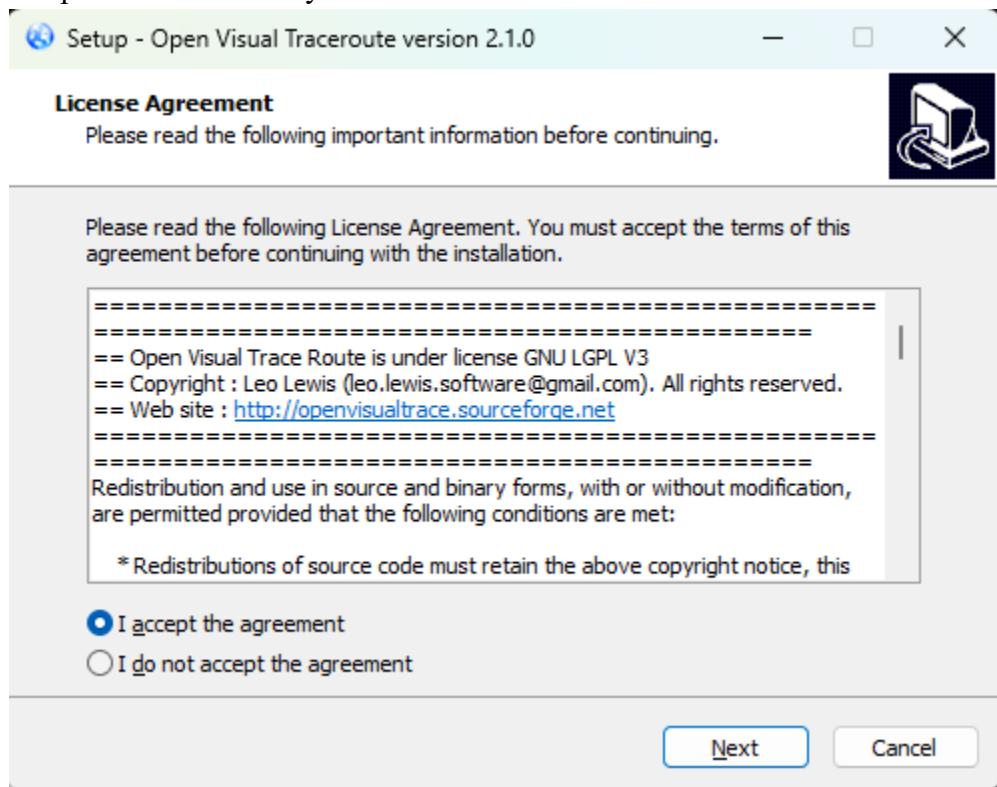
- Pasa por un nodo de Akamai, proveedor CDN que optimiza el tráfico antes de llegar a Europa.
- **Paso 6 al 12: Red de Arelion**
  - IPs: Rango 62.115.x.x y 213.155.129.251
  - Países: Francia, Suecia
  - El tráfico pasa por varios nodos de **Twelve99/Arelion**, una red troncal (backbone) internacional. Aquí se ve el tránsito desde EE. UU. hacia Europa, pasando por Suecia y Francia.
- **Paso 13 al 14: Saltos sin respuesta**
  - Resultado: \_\_\_\_\_
  - Algunos routers intermedios no responden a ICMP. Esto es normal y no indica alguna perdida de conexión
- **Paso 15 – Destino final: Francia**
  - IP: 160.92.168.33
  - Hostname: secure-www-sp-dila-prod.as8677.net
  - País: Francia
  - El paquete llega al servidor del gobierno francés, que aloja el sitio web.
- Download and install software such as VisualRoute, Open Visual Traceroute, or similar. They can be free tools or demos.
  - Nos dirigimos a la página de <https://visualtraceroute.net/> e damos click en Download Windows



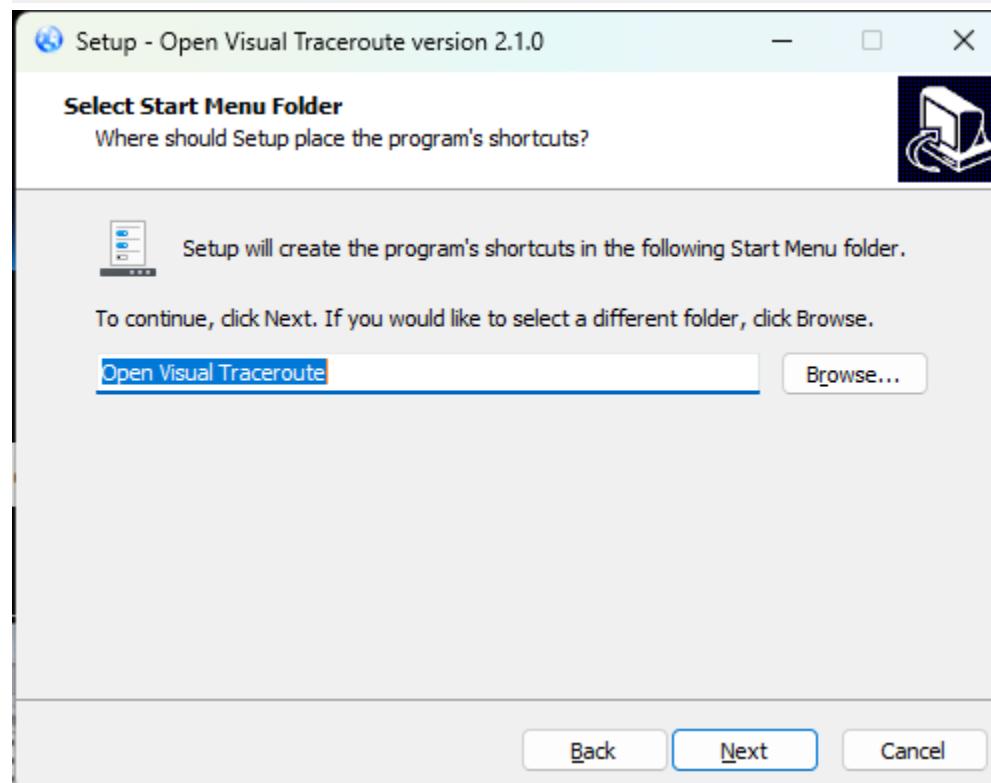
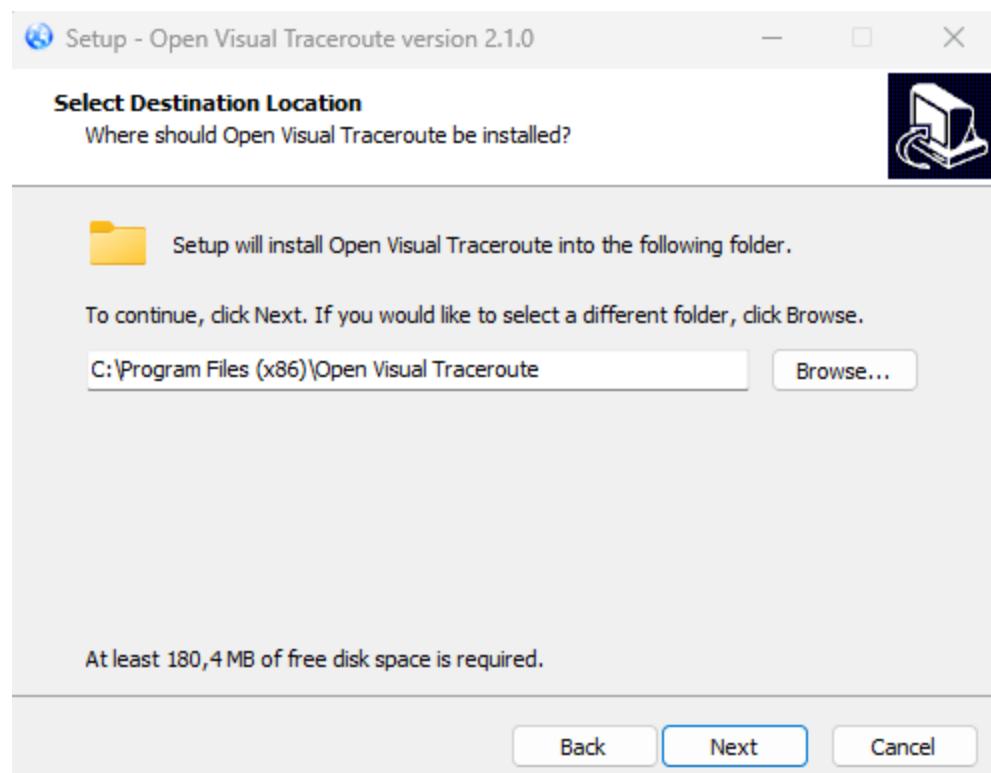
- Nos redirigirá a otra página y se iniciara una descarga, esperamos que esta finalice y damos click en el archivo de instalación

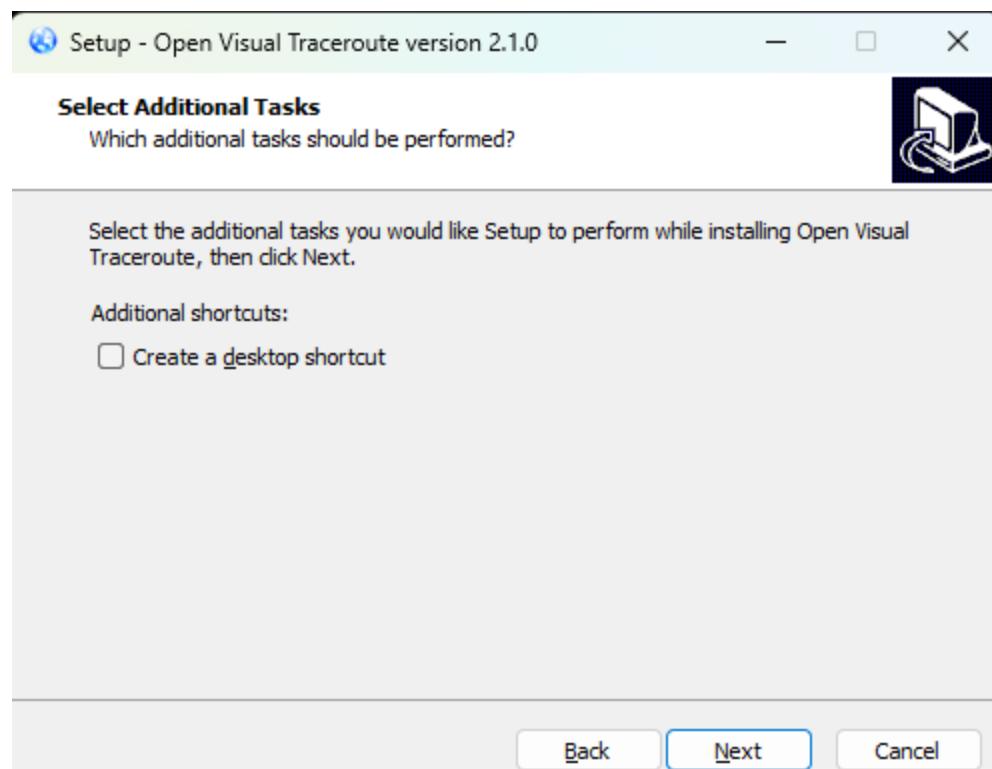


- Aceptamos el contrato y damos click en next

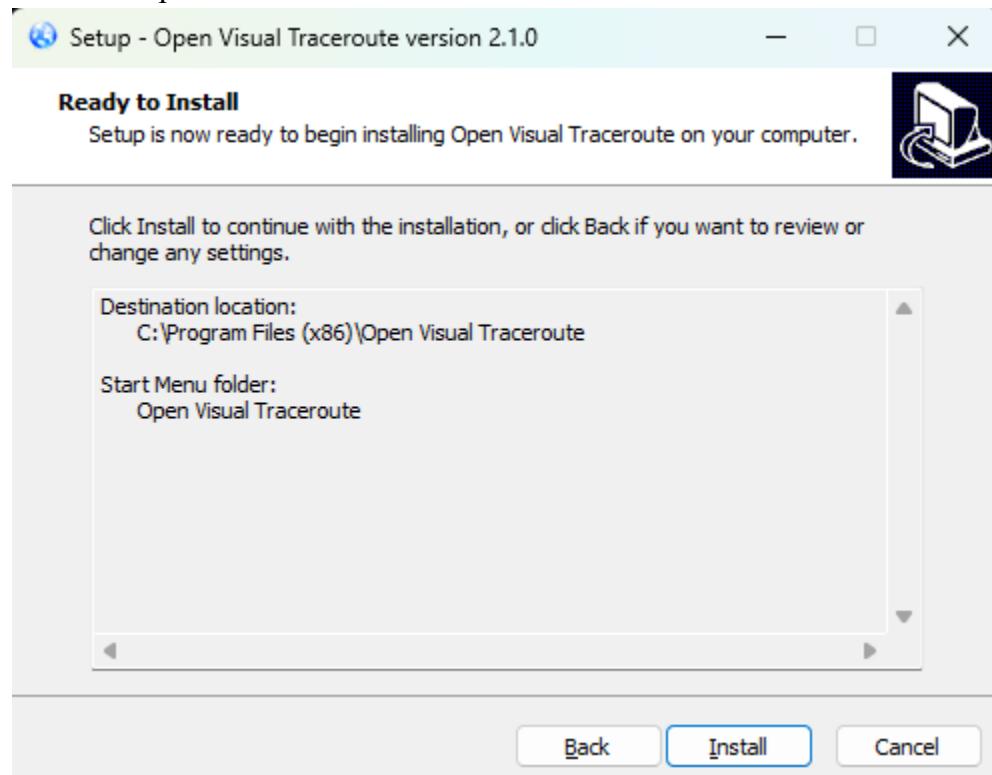


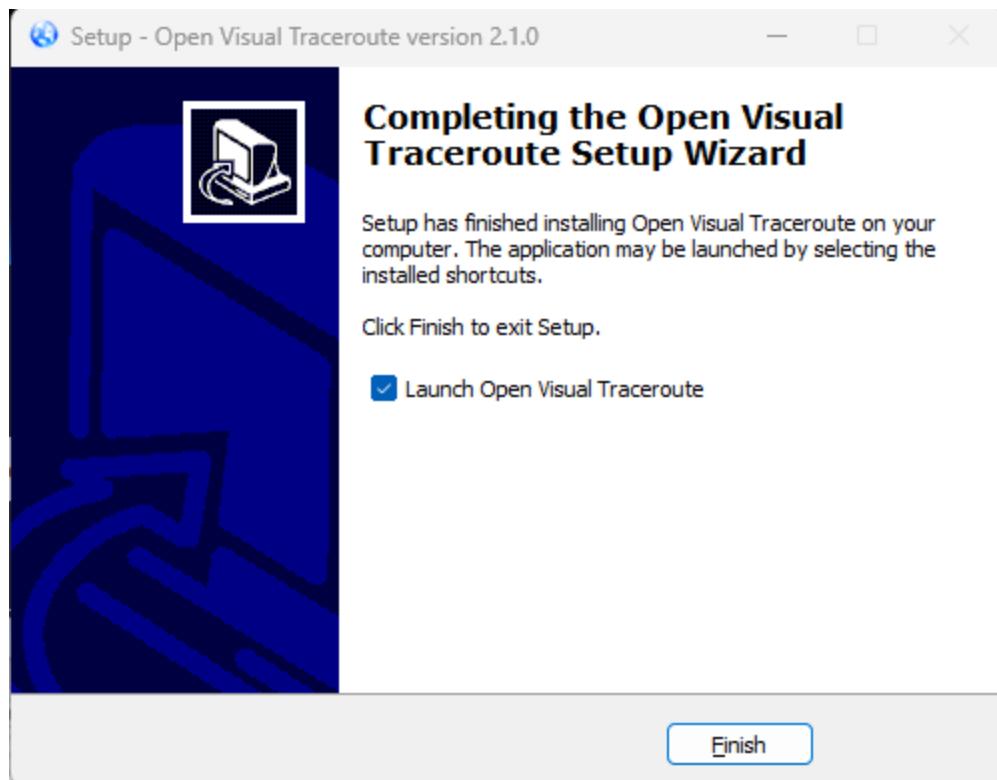
- Seleccionamos donde se va guardar la instalación . En este caso hay una ruta por defecto, damos click en next en los siguientes 3 apartados



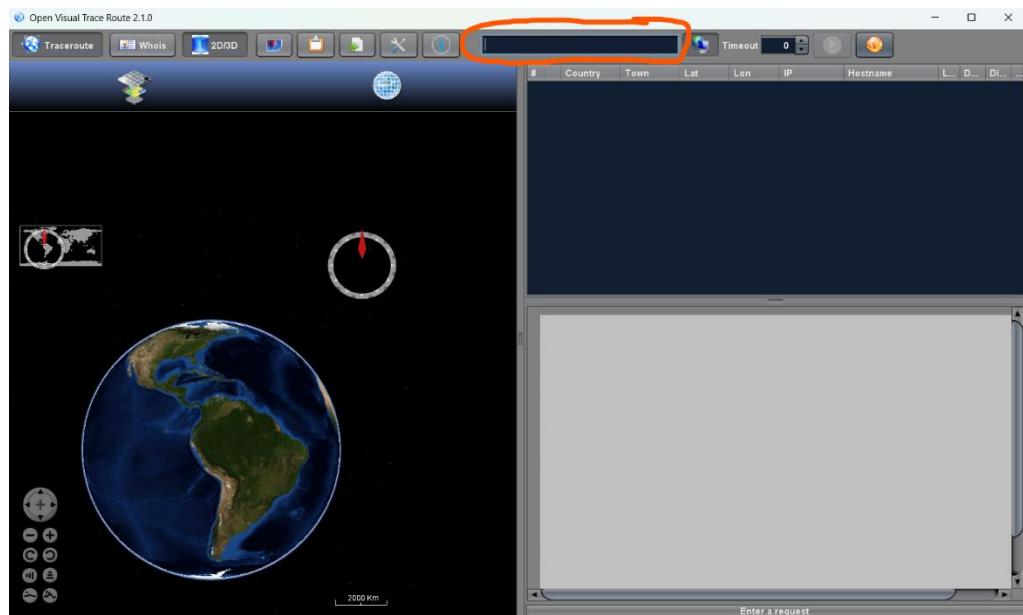


- Finalmente damos install y luego de unos segundos ya tendremos instalado Open Visual Traceroute





- Test the tool, get to know some of its features.
  - Al probar la herramienta Open Visual Traceroute notamos las siguientes principales funcionalidades:
    - Permite visualizar gráficamente la ruta que siguen los paquetes desde mi equipo hasta un servidor destino, mostrando los saltos sobre un mapa mundial en tiempo real.
    - Muestra detalles técnicos como IP, ASN, país, y tiempos de respuesta (latencia) por cada salto.
    - Identifica dónde se pierde la conexión o si algún nodo no responde.
- Document the tool's functionality by searching for 5 car manufacturers' websites around the world with the tool.
  - Para hacer consultas a direcciones específicas, agregamos el nombre del sitio web en la parte superior



- En nuestro caso, serán 5 consultas diferentes de marcas de carros
  - Marca 1 (Colombia) [www.toyota.jp](http://www.toyota.jp)
    - Podemos notar dos principales cosas . En la parte superior derecha nos muestra las rutas por las cuales paso antes de llegar a [www.toyota.jp](http://www.toyota.jp)

| # | Country  | Town      | Lat    | Lon      | IP             | Hostname              | L... | D... | Di... | ... |
|---|----------|-----------|--------|----------|----------------|-----------------------|------|------|-------|-----|
| 1 | Colombia | Bogotá    | 4.6115 | -74.0833 | 192.168.1.1    | (None)                | 3    | 54   | 0     | ?   |
| 2 | Colombia | Bogotá    | 4.6115 | -74.0833 | 152.203.32.1   | (None)                | 6    | 10   | 0     | ?   |
| 3 | Colombia | (Unknown) | 4.5981 | -74.0799 | 206.223.124... | telefonica2-nap.cc... | 7    | 9    | 1     | ?   |
| 4 | Colombia | (Unknown) | 4.5981 | -74.0799 | 206.223.124... | internexa1-nap.cci... | 74   | 10   | 0     | ?   |
| 5 | Colombia | Bogotá    | 4.6115 | -74.0833 | 179.1.81.214   | (None)                | 8    | 8    | 1     | ?   |
| 6 | ?        | *         | 4.6115 | -74.0833 | *              | *                     | 0    | <1   | 0     | ?   |
| 7 | ?        | *         | 4.6115 | -74.0833 | *              | *                     | 0    | <1   | 0     | ?   |
| 8 | ?        | *         | 4.6115 | -74.0833 | *              | *                     | 0    | <1   | 0     | ?   |
| 9 | Colombia | Bogotá    | 4.6115 | -74.0833 | 184.31.181...  | a184-31-181-57.de...  | 8    | 10   | 0     | ?   |

- En segunda parte podemos ver visualmente la ruta la cual siguió hasta llegar a [www.toyota.jp](http://www.toyota.jp)
- En este caso no salió de Colombia entonces la ruta no fue a otros países

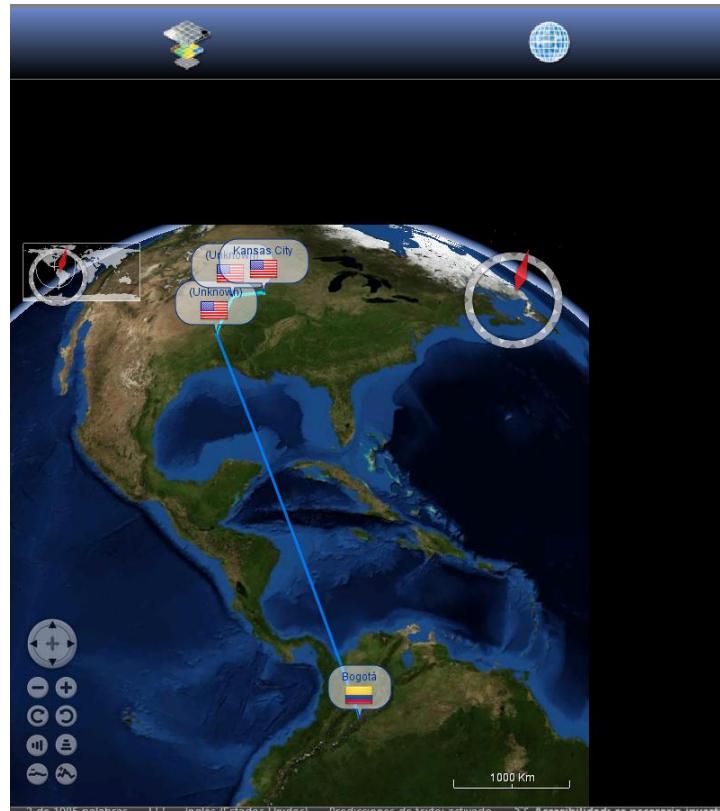


- Repetimos el mismo proceso para los otros sitios web de carros escogidos
- Marca 2 (Estados Unidos) <https://www.hyundaiusa.com/us/en>
  - Rutas

<https://www.hyundaiusa.com/us/en>

| # | Country      | Town        | Lat     | Lon      | IP             | Hostname               | L... | D... | Di... | ... |
|---|--------------|-------------|---------|----------|----------------|------------------------|------|------|-------|-----|
| 1 | Colombia     | Bogotá      | 4.6115  | -74.0833 | 192.168.1.1    | (None)                 | 3    | 9    | 0     | ?   |
| 2 | Colombia     | Bogotá      | 4.6115  | -74.0833 | 152.203.32.1   | (None)                 | 6    | 11   | 0     | ?   |
| 3 | United St... | (Unknown)   | 4.6115  | -74.0833 | *              | *                      | 0    | <1   | 0     | ?   |
| 4 | United St... | (Unknown)   | 32.7797 | -96.8022 | 72.14.222.1... | (None)                 | 7    | 26   | 39... | ?   |
| 5 | United St... | (Unknown)   | 37.751  | -97.822  | 192.178.111... | (None)                 | 10   | 11   | 561   | ?   |
| 6 | United St... | (Unknown)   | 37.751  | -97.822  | 142.250.215... | (None)                 | 7    | 82   | 0     | ?   |
| 7 | United St... | Kansas City | 39.1027 | -94.5778 | 34.107.166...  | 244.166.107.34.bc.g... | 16   | 82   | 320   | ?   |

- Mapa visual
- En este caso podemos notar que la ruta ya salió del país Colombia y podemos ver visualmente porque zonas pasa hasta llegar a su destino

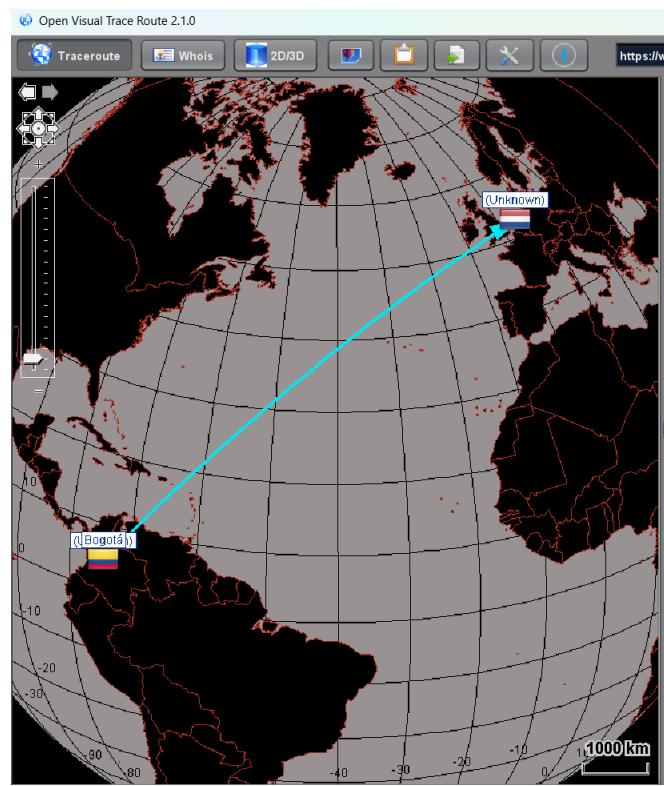


- Marca 3 (Holanda) <https://www.tesla.com/>
  - Rutas

https://www.tesla.com/

| # | Country      | Town      | Lat     | Lon      | IP             | Hostname |
|---|--------------|-----------|---------|----------|----------------|----------|
| 1 | Colombia     | Bogotá    | 4.6115  | -74.0833 | 192.168.1.1    |          |
| 2 | Colombia     | Bogotá    | 4.6115  | -74.0833 | 152.203.32.1   |          |
| 3 | Colombia     | (Unknown) | 4.5981  | -74.0799 | 206.223.124... |          |
| 4 | Colombia     | (Unknown) | 4.5981  | -74.0799 | 206.223.124... |          |
| 5 | Colombia     | Bogotá    | 4.6115  | -74.0833 | 179.1.81.214   |          |
| 6 | *            | *         | 4.6115  | -74.0833 | *              | *        |
| 7 | *            | *         | 4.6115  | -74.0833 | *              | *        |
| 8 | *            | *         | 4.6115  | -74.0833 | *              | *        |
| 9 | Netherlan... | (Unknown) | 52.3824 | 4.8995   | 2.19.32.68     |          |

- Mapa
  - Si lo deseamos, también se puede ver el recorrido de manera 2d dando click en la tercera opción 2d/3d



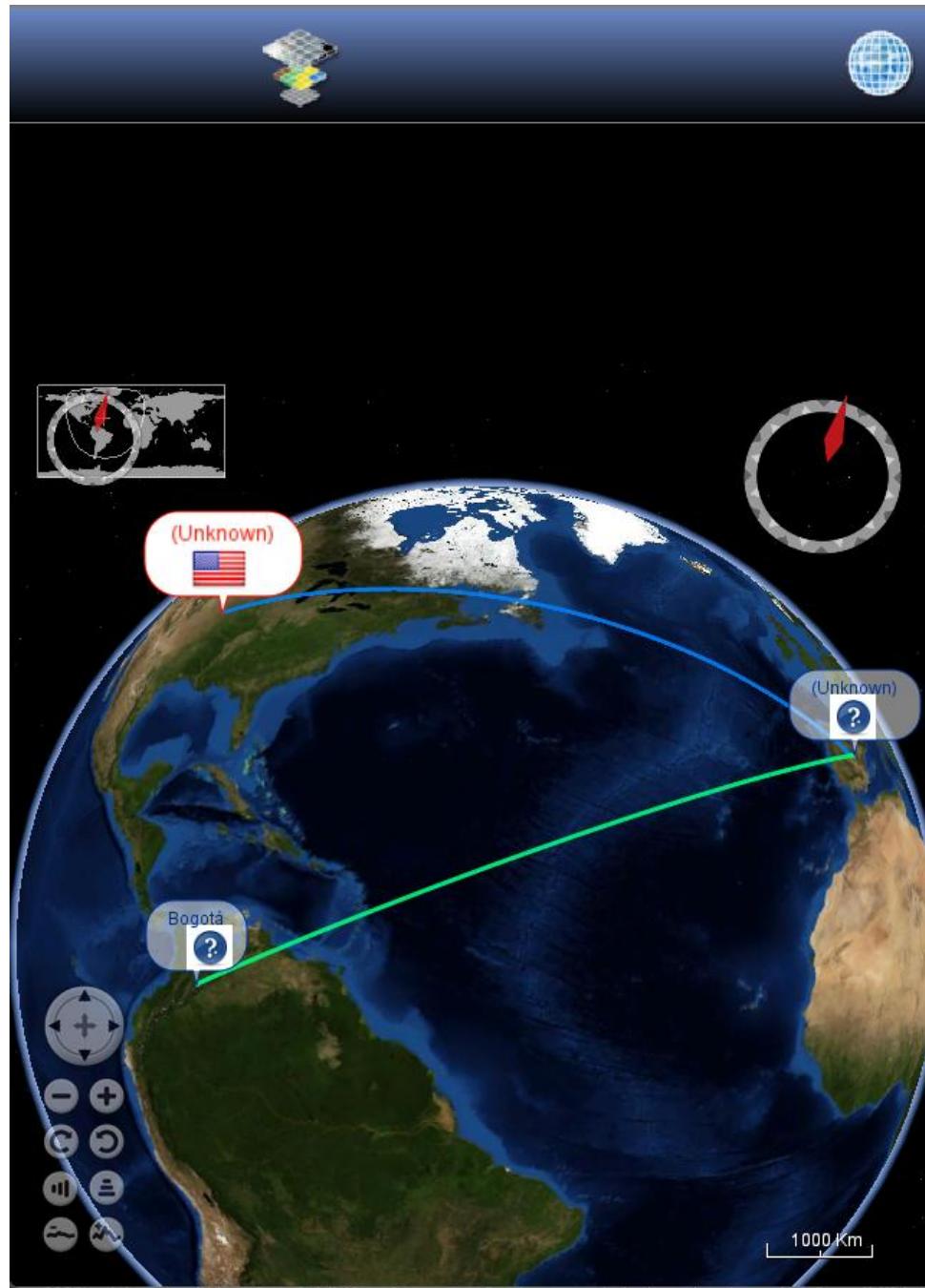
- Marca 4 (Alojada en USA, pero pasa por Europa)

<https://www.ferrari.com/es-ES>

- Rutas

| #  | Country      | Town      | Lat     | Lon      | IP             | Hostname              | L... | D... |
|----|--------------|-----------|---------|----------|----------------|-----------------------|------|------|
| 1  | Colombia     | Bogotá    | 4.6115  | -74.0833 | 192.168.1.1    | (None)                | 9    | 1    |
| 2  | Colombia     | Bogotá    | 4.6115  | -74.0833 | 152.203.32.1   | (None)                | 8    | 1    |
| 3  | ?            | *         | 4.6115  | -74.0833 | *              | *                     | 0    | <    |
| 4  | Spain        | (Unknown) | 40.4172 | -3.684   | 81.173.106.... | be12-grtbotgm1.n...   | 20   | 1    |
| 5  | ?            | *         | 40.4172 | -3.684   | *              | *                     | 0    | <    |
| 6  | ?            | *         | 40.4172 | -3.684   | *              | *                     | 0    | <    |
| 7  | ?            | *         | 40.4172 | -3.684   | *              | *                     | 0    | <    |
| 8  | ?            | *         | 40.4172 | -3.684   | *              | *                     | 0    | <    |
| 9  | ?            | *         | 40.4172 | -3.684   | *              | *                     | 0    | <    |
| 10 | ?            | *         | 40.4172 | -3.684   | *              | *                     | 0    | <    |
| 11 | United St... | (Unknown) | 37.751  | -87.822  | 3.163.60.101   | server-3-163-60-10... | 7    | 1    |

- Mapa



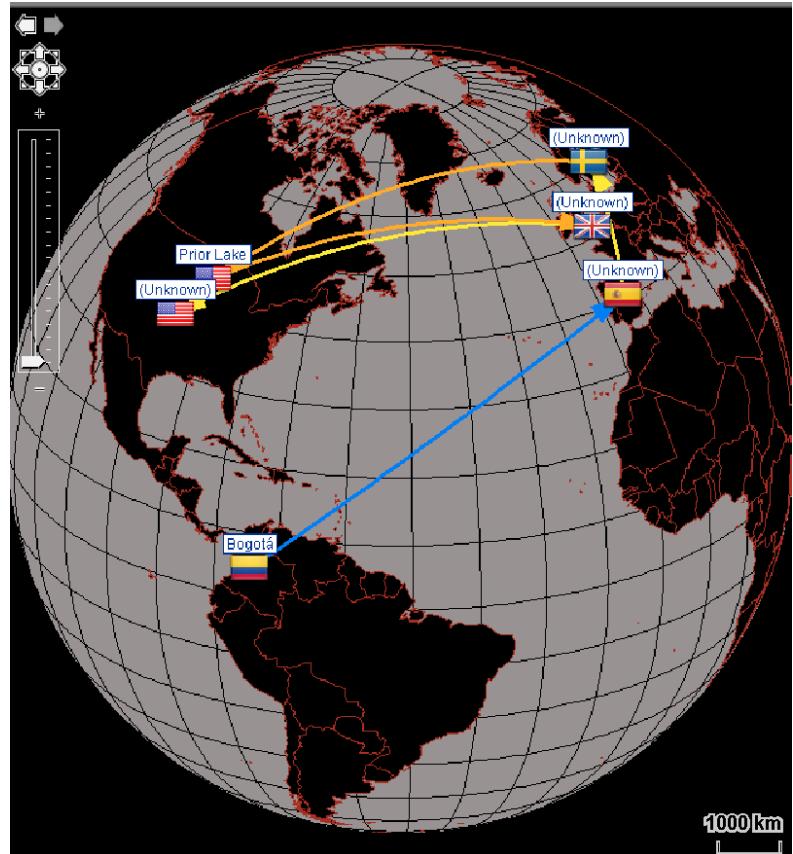
- Marca 5 (varias rutas pasando por Europa y América)  
[www.volvocars.com/se/](http://www.volvocars.com/se/)
  - Rutas

volvocars.com/se/

Timeout 0

| #  | Country      | Town       | Lat     | Lon      | IP             | Hostname         |
|----|--------------|------------|---------|----------|----------------|------------------|
| 1  | Colombia     | Bogotá     | 4.6115  | -74.0833 | 192.168.1.1    | (None)           |
| 2  | Colombia     | Bogotá     | 4.6115  | -74.0833 | 152.203.32.1   | (None)           |
| 3  | *            | *          | 4.6115  | -74.0833 | *              | *                |
| 4  | Spain        | (Unknown)  | 40.4172 | -3.684   | 81.173.106.... | be12-grtbo...    |
| 5  | Spain        | (Unknown)  | 40.4172 | -3.684   | 84.16.15.55    | (None)           |
| 6  | Spain        | (Unknown)  | 40.4172 | -3.684   | 5.53.7.117     | (None)           |
| 7  | *            | *          | 40.4172 | -3.684   | *              | *                |
| 8  | *            | *          | 40.4172 | -3.684   | *              | *                |
| 9  | Sweden       | (Unknown)  | 59.3247 | 18.056   | 62.115.119.... | rest-bb1-link... |
| 10 | Sweden       | (Unknown)  | 59.3247 | 18.056   | 62.115.140.... | prs-bb1-link...  |
| 11 | Sweden       | (Unknown)  | 59.3247 | 18.056   | 62.115.134.... | adm-bb1-link...  |
| 12 | Sweden       | (Unknown)  | 59.3247 | 18.056   | 62.115.138.... | adm-b3-link...   |
| 13 | United St... | Prior Lake | 44.7098 | -93.4129 | 62.115.186.... | microsoft-ic...  |
| 14 | United Ki... | (Unknown)  | 51.4964 | -0.1224  | 51.10.13.106   | ae1230-0.ice...  |
| 15 | *            | *          | 51.4964 | -0.1224  | *              | *                |
| 16 | *            | *          | 51.4964 | -0.1224  | *              | *                |
| 17 | *            | *          | 51.4964 | -0.1224  | *              | *                |
| 18 | *            | *          | 51.4964 | -0.1224  | *              | *                |
| 19 | *            | *          | 51.4964 | -0.1224  | *              | *                |
| 20 | United St... | (Unknown)  | 37.751  | -97.822  | 20.33.25.29    | (None)           |

- Mapa



## 2. Some Questions About Router Commands

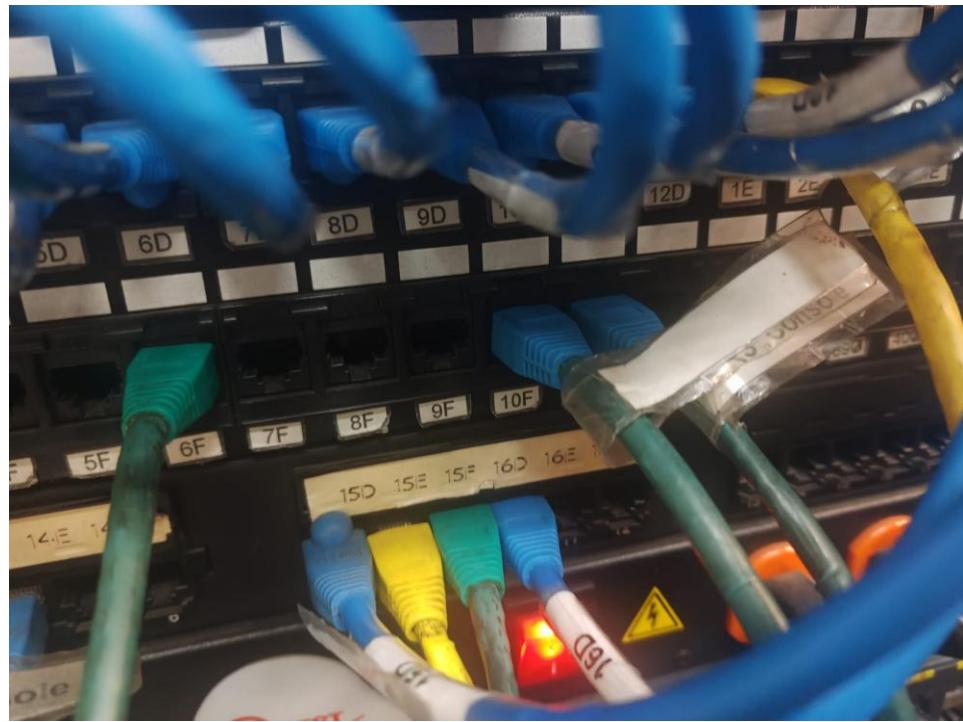
- What is the difference between enable password and enable secret? If both are configured, which one takes precedence
  - **Enable password:** Es una contraseña en texto plano que se usa para acceder al modo privilegiado EXEC en los dispositivos Cisco. No está cifrada por defecto, lo que la hace menos segura.
  - **Enable secret:** Es una contraseña cifrada (encriptada), lo que la hace más segura que enable password. Si ambas se configuran, la contraseña enable secret toma prioridad sobre la enable password.
- What is the difference between console and VTY?
  - **Console:** Es un puerto físico en el router o switch utilizado para gestionar el dispositivo localmente. Requiere un cable de consola y se usa principalmente para la configuración inicial y la solución de problemas.
  - **VTY (Virtual Teletype):** Son líneas virtuales que permiten el acceso remoto al router o switch a través de Telnet o SSH. Las líneas VTY se utilizan para administrar dispositivos de forma remota y se pueden configurar varias sesiones VTY simultáneamente.
- What is the boot process of the routers in the Network Laboratory?
  - El proceso de arranque de un router del laboratorio sigue estos pasos:
    - **POST (Power-On Self-Test):** El router realiza una prueba de autodiagnóstico de hardware.
    - **Bootstrap:** El programa bootstrap (almacenado en la ROM) localiza la imagen del IOS en la memoria flash.
    - **Cargar el IOS:** El router carga el sistema operativo Cisco IOS desde la memoria flash a la memoria RAM.
    - **Cargar la configuración:** El router busca el archivo de configuración de inicio (startup-config) almacenado en la NVRAM y lo carga en la memoria RAM
    - **Ejecutar la configuración:** El router aplica la configuración cargada para iniciar las interfaces, protocolos de enrutamiento y demás parámetros.
- What types of memory do the routers in the Network Laboratory have
  - Los routers del laboratorio tienen varios tipos de memoria:
    - **ROM (Read-Only Memory):** Almacena el programa bootstrap y el software básico de diagnóstico.
    - **RAM (Random Access Memory):** Es la memoria temporal donde el router almacena la configuración en ejecución, tablas de enrutamiento y el IOS durante la operación.

- **NVRAM (Non-Volatile RAM)**: Almacena el archivo de configuración de inicio (startup-config), el cual se carga al arrancar el router.
    - **Memoria Flash**: Almacena la imagen del IOS y otros archivos necesarios para el funcionamiento del router.
  - What is the difference between the startup-configuration and running-configuration files
    - **Startup-Configuration (startup-config)**: Este archivo está almacenado en la NVRAM y contiene la configuración que se carga cuando el router arranca. Es persistente a través de los reinicios.
    - **Running-Configuration (running-config)**: Este archivo está almacenado en la RAM y contiene la configuración activa que está siendo utilizada en ese momento por el router. Cualquier cambio realizado durante la sesión se guarda aquí. Se pierde cuando el router se reinicia, a menos que se guarde en el archivo startup-config
3. Setup: Access and Basic Configuration of Routers
- Review and document the different routers available in the Laboratory and the network interfaces they have. To connect to the routers through the console, use a console cable and the hyperterminal or putty application, similar to how you connected to the switches.
    - Usamos la extensión del cable serial RS-232 y lo conectamos a una computadora.

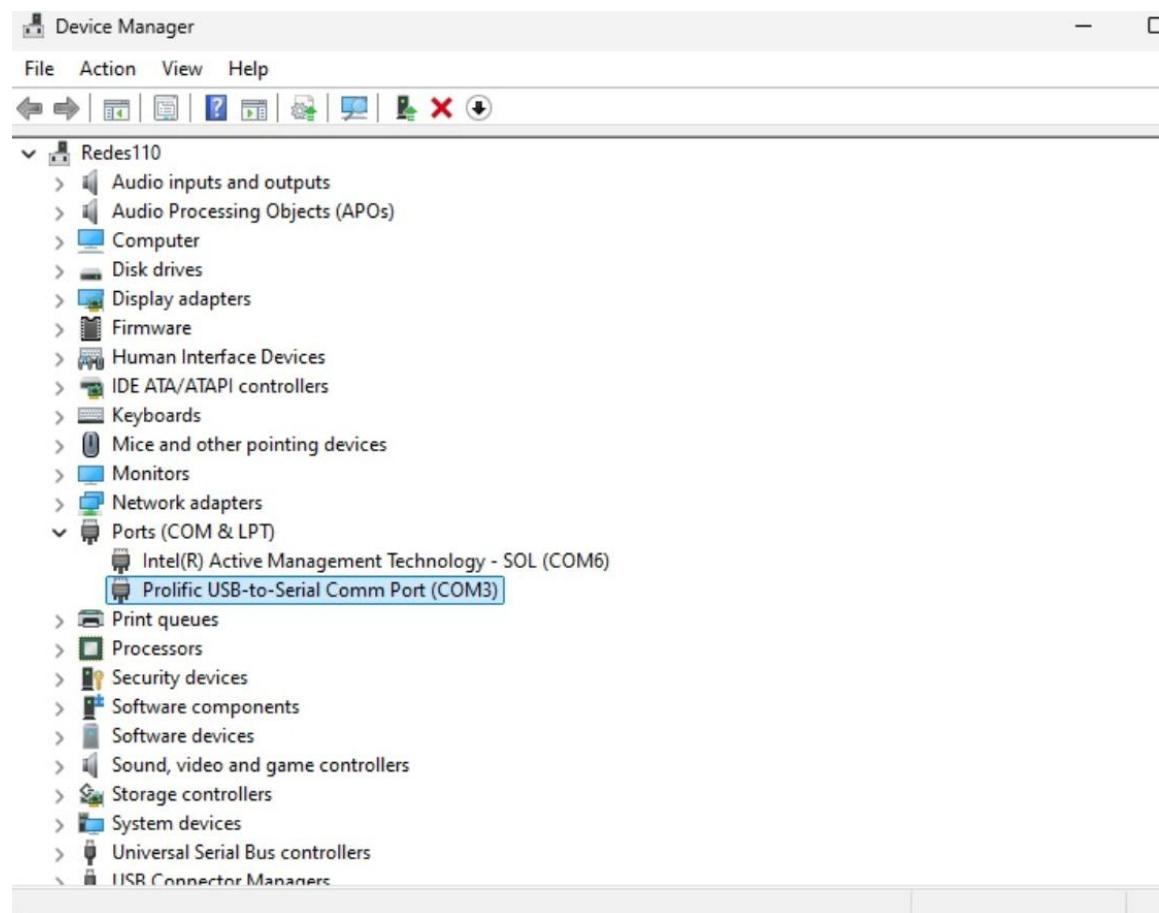




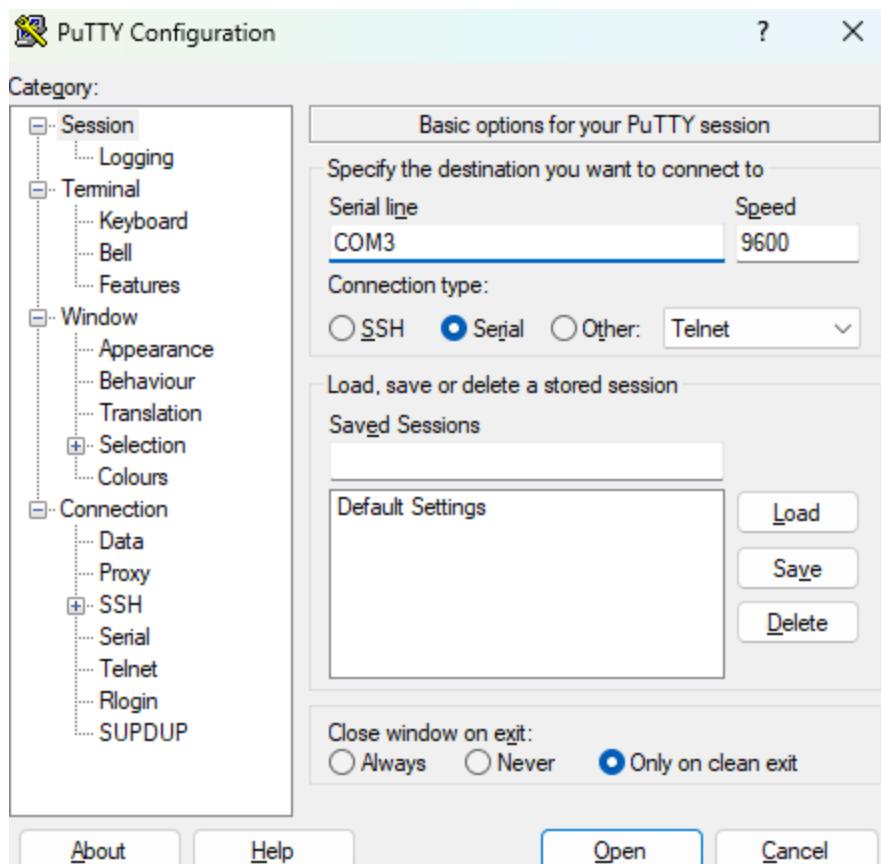
- Teniendo en cuenta el puerto por el cual conectamos el cable (10f) ahora lo conectamos en el enrutador (10f)



- Ya teniendo la capa física lista. Ahora encendemos el computador y abrimos la aplicación Device Manager en el apartado de puertos y notamos cual es nuestro COM.
- En este caso es COM3



- Ahora para conectarnos al router, abrimos la aplicación PuTTY, seleccionamos la opción Serial ingresamos nuestro COM (en este caso es COM3)



- Ahora damos click en Open y nos ingresa a la consola para configurar el router
- For Cisco routers, explain the process the router goes through on boot when configured in modes 0x2102 and 0x2102.
- **Modo 0x2102 – Modo de arranque normal**
  - Este es el valor por defecto en la mayoría de los routers Cisco. Cuando el router tiene este valor configurado:
    - Se hace el POST (Power-On Self Test), que es como un chequeo básico de hardware.
    - Carga el IOS (el sistema operativo del router) desde la memoria flash.
    - Luego, lee la configuración guardada en la NVRAM, es decir, el archivo de configuración llamado startup-config.
    - Carga esa configuración al modo activo (running-config), y el router queda listo para funcionar con todas las interfaces y parámetros que teníamos configurados.
  - Esto es lo que sucede cuando todo está bien y el router debe arrancar de manera normal
- **Modo 0x2142 – Modo para ignorar la configuración guardada**

- Este se usa principalmente cuando queremos recuperar el acceso al router. (casos como que se nos olvidó la contraseña). Lo que hace es:
    - Arranca igual con el POST y carga el IOS, pero ignora la configuración que está en la NVRAM (el startup-config).
    - Como no hay configuración cargada, entra como si estuviera de fábrica, y nos deja entrar sin pedir contraseñas configuradas anteriormente.
    - Desde ahí, podemos entrar al modo privilegiado, copiar la configuración guardada con el comando copy startup-config running-config, y cambiar lo que necesitemos (por ej, la contraseña)
    - Despues de hacer los cambios, volvemos a configurar el registro a 0x2102 (config-register 0x2102) y guardamos (write memory o copy run start) para que arranque normal la próxima vez.
  - Perform the following configuration using physical devices and document the process. The basic configuration of a router should include:
    - *Access keys for privileged mode, console, and remote access. The privileged mode key should be "cisco", the console key "claveC", and the remote access (telnet) key "claveT".*
    - *Router name. Assign the router the last name of one of the students in the group*
    - *Console and remote access screen synchronization*
    - *Description of the interfaces used.*
    - *Disable remote command server lookup.*
    - *Message of the day.*
  - Ya adentro de la consola de configuración del router, realizamos la configuración inicial

```

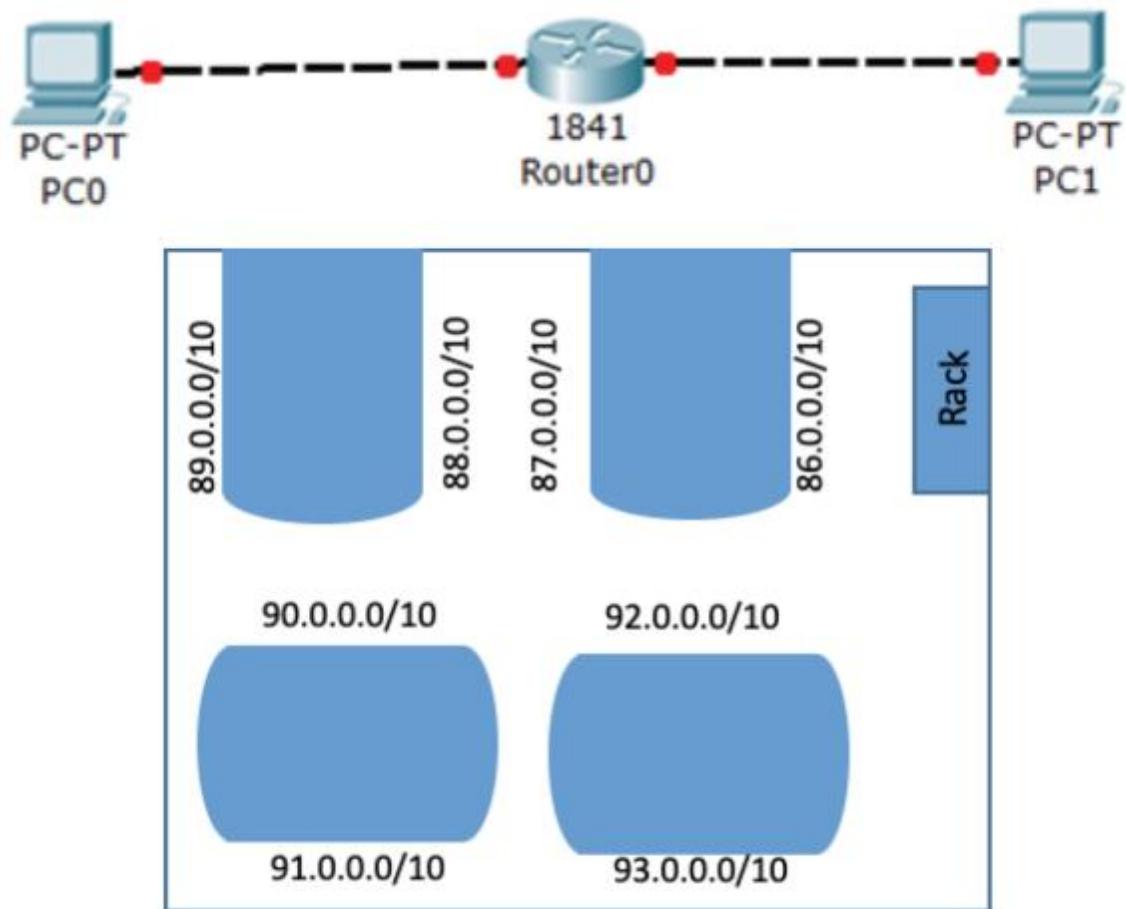
Router(config)#hostname tebinjuanin
tebinjuanin(config)#line console 0
tebinjuanin(config-line)#logging synchronous
tebinjuanin(config-line)#password root123
tebinjuanin(config-line)#login
tebinjuanin(config-line)#exit
tebinjuanin(config)#line vty 0 15
tebinjuanin(config-line)#logging synchronous
tebinjuanin(config-line)#password root123
tebinjuanin(config-line)#login
tebinjuanin(config-line)#exit
tebinjuanin(config)#no ip domain-lookup
tebinjuanin(config)#inteface FastEthernet0/0
                                ^
% Invalid input detected at '^' marker.

```

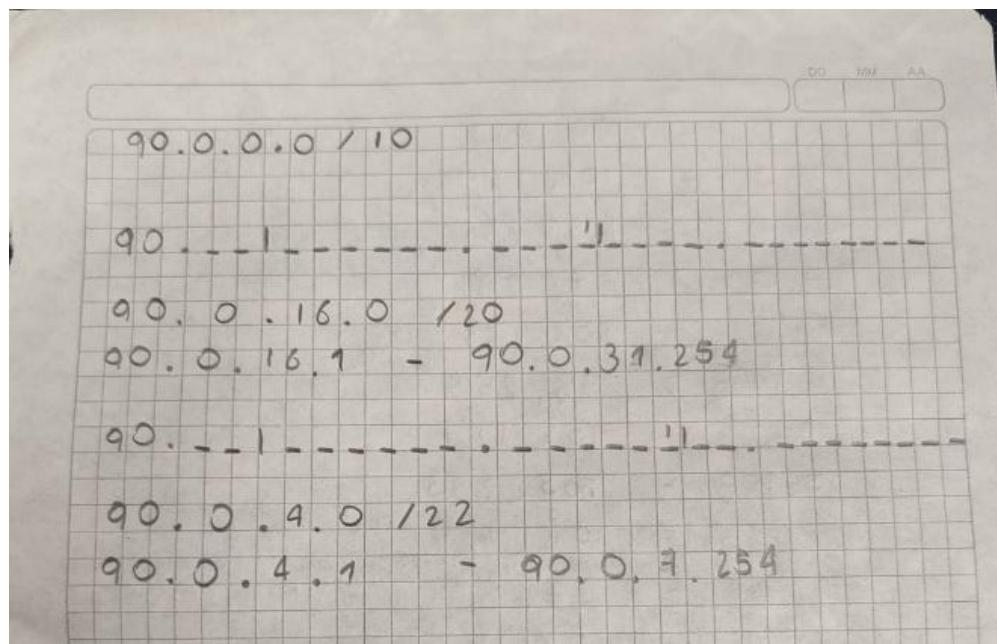
Donde

- hostname tebinjuanin
  - Cambia el nombre del router a tebinjuanin.
- line console 0
  - Entra a la configuración de la consola (acceso físico al dispositivo).
- logging synchronous
  - Evita que los mensajes del sistema interrumpan el ingreso de comandos.
- password root123
  - Establece la contraseña root123 para el acceso por consola.
- Login
  - Requiere que se ingrese la contraseña para poder acceder a través de la consola.
- Exit
  - Sale del modo de configuración de línea y regresa al modo de configuración global.
- line vty 0 15
  - Entra a la configuración de las líneas virtuales (acceso remoto por Telnet o SSH).
- logging synchronous
  - Aplica la misma configuración para evitar interrupciones al usar acceso remoto.
- password root123
  - Establece la misma contraseña para las líneas VTY.
- Login
  - Requiere contraseña para conexiones remotas.

- Exit
  - Regresa al modo de configuración global.
- no ip domain-lookup
  - Desactiva la búsqueda DNS cuando se escribe un comando mal.
- For the network on the left, use the range that allows addressing 4000 hosts, and for the network on the right, a range that allows addressing 600 hosts. Use the IP address range according to the location of the Network Laboratory desk diagram.



- Antes de configurar las interfaces realizamos el subnetting para separar las redes en 4000 y 600 host respectivamente



- Como resultado nos el id de las redes, Gateway, rango utilizable y sus mascaras son:
  - Red 1
    - ID: 90.0.16.0
    - Gateway: 90.0.16.1
    - Rango: 90.0.16.2 – 90.0.31.254
    - Mascara: 255.255.240.0 /20
  - Red 2
    - ID: 90.0.4.0
    - Gateway: 90.0.4.1
    - Rango: 90.0.4.2 – 90.0.7.254
    - Mascara: 255.255.252.0 /22
- Ya teniendo las particiones realizadas, procedemos a realizar la capa física
- Verificamos que numeros tienen los puertos de los computadores y los conectamos en los mismos puertos, pero en el enrutador
  - Pc 1 (10f)

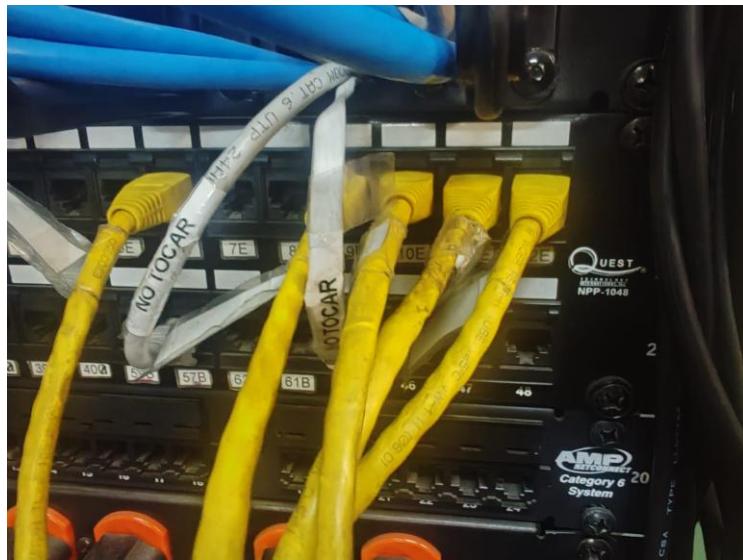


- Pc 2 (9f)



- Enrutador (10f , 9f)
- Para el caso del Pc 1 usamos la interfaz Fa 0/0 y del Pc 2 la interfaz Fa 0/1





- Con la capa física, configuramos las interfaces de red para cada computador

- Configuración PC 1 (4000 host)

```
tebinjuanin(config-if)#interface FastEthernet 0/1
tebinjuanin(config-if)#description "Connection Router 10 with PC2"
tebinjuanin(config-if)#ip address 90.0.4.1 255.255.252.0
tebinjuanin(config-if)#no shutdown
```

- Configuración PC 2 (600 host)

```
tebinjuanin(config-if)#interface FastEthernet 0/1
tebinjuanin(config-if)#description "Connection Router 10 with PC2"
tebinjuanin(config-if)#ip address 90.0.4.1 255.255.252.0
tebinjuanin(config-if)#no shutdown
```

- Guardamos la configuración de las interfaces de red con el comando copy running

```
tebinjuanin(config)#exit
tebinjuanin#
*Jan 1 00:27:10.779: %SYS-5-CONFIG_I: Configured from console by console
tebinjuanin#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

- Verificamos que las dos interfaces hayan quedado configuradas correctamente con el comando “show ip interface brief”

| Interface         | IP-Address | OK? | Method | Status                | Protocol |
|-------------------|------------|-----|--------|-----------------------|----------|
| FastEthernet0/0   | 90.0.16.1  | YES | manual | up                    | up       |
| FastEthernet0/1   | 90.0.4.1   | YES | manual | up                    | up       |
| FastEthernet0/0/0 | unassigned | YES | unset  | up                    | down     |
| FastEthernet0/0/1 | unassigned | YES | unset  | up                    | down     |
| FastEthernet0/0/2 | unassigned | YES | unset  | up                    | down     |
| FastEthernet0/0/3 | unassigned | YES | unset  | up                    | down     |
| Serial0/1/0       | unassigned | YES | unset  | administratively down | down     |
| Serial0/1/1       | unassigned | YES | unset  | administratively down | down     |
| Vlan1             | unassigned | YES | unset  | up                    | down     |

- Ahora asignamos las ip en la configuración del pc. Nos dirigimos a configuración de red e ingresamos la primera ip utilizable del rango y el Gateway configurado anteriormente para cada computador

- PC1

**Editar configuración de IP**

Manual

**IPv4**

Activado

Dirección IP

90.0.16.2

Máscara de subred

255.255.240.0

Puerta de enlace

90.0.16.1

DNS preferido

8.8.8.8

DNS a través de HTTPS

Desactivado

DNS alternativo

Guardar

Cancelar

- PC2

**Edit IP settings**

Manual

**IPv4**

On

IP address: 90.0.4.2

Subnet mask: 255.255.252.0

Gateway: 90.0.4.1

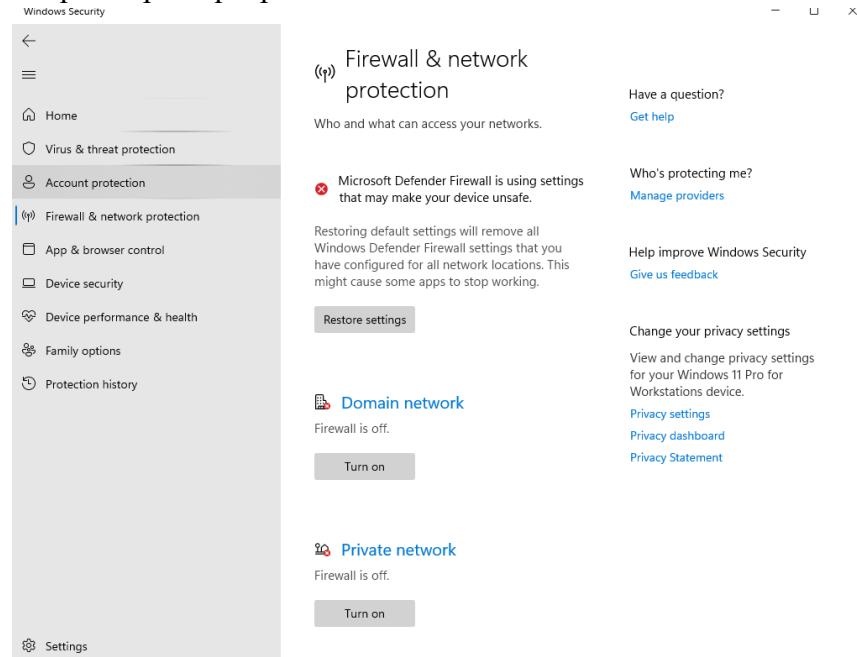
Preferred DNS: 8.8.8.8

DNS over HTTPS: Off

Alternate DNS:

Save Cancel

- Test connectivity between computers
  - Antes de realizar las pruebas debemos desactivar el firewall de las máquinas para que permita hacer conexiones



- PC1 to PC2

```
Ping statistics for 90.0.4.2:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

- PC1 to Gateway

```
C:\Users\Redes>ping 90.0.16.1
```

```
Pinging 90.0.16.1 with 32 bytes of data:  
Reply from 90.0.16.1: bytes=32 time<1ms TTL=255  
Reply from 90.0.16.1: bytes=32 time<1ms TTL=255  
Reply from 90.0.16.1: bytes=32 time<1ms TTL=255  
Reply from 90.0.16.1: bytes=32 time<1ms TTL=255
```

```
Ping statistics for 90.0.16.1:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- PC2 to PC1

```
C:\Users\Redes>ping 90.0.16.2
```

```
Pinging 90.0.16.2 with 32 bytes of data:  
Reply from 90.0.16.2: bytes=32 time=1ms TTL=127  
Reply from 90.0.16.2: bytes=32 time=1ms TTL=127  
Reply from 90.0.16.2: bytes=32 time=1ms TTL=127  
Reply from 90.0.16.2: bytes=32 time=1ms TTL=127
```

```
Ping statistics for 90.0.16.2:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

```
C:\Users\Redes>
```

- PC2 to Gateway

```
C:\Users\Redes>ping 90.0.4.1

Pinging 90.0.4.1 with 32 bytes of data:
Reply from 90.0.4.1: bytes=32 time<1ms TTL=255
Reply from 90.0.4.1: bytes=32 time=1ms TTL=255
Reply from 90.0.4.1: bytes=32 time=1ms TTL=255
Reply from 90.0.4.1: bytes=32 time<1ms TTL=255

Ping statistics for 90.0.4.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

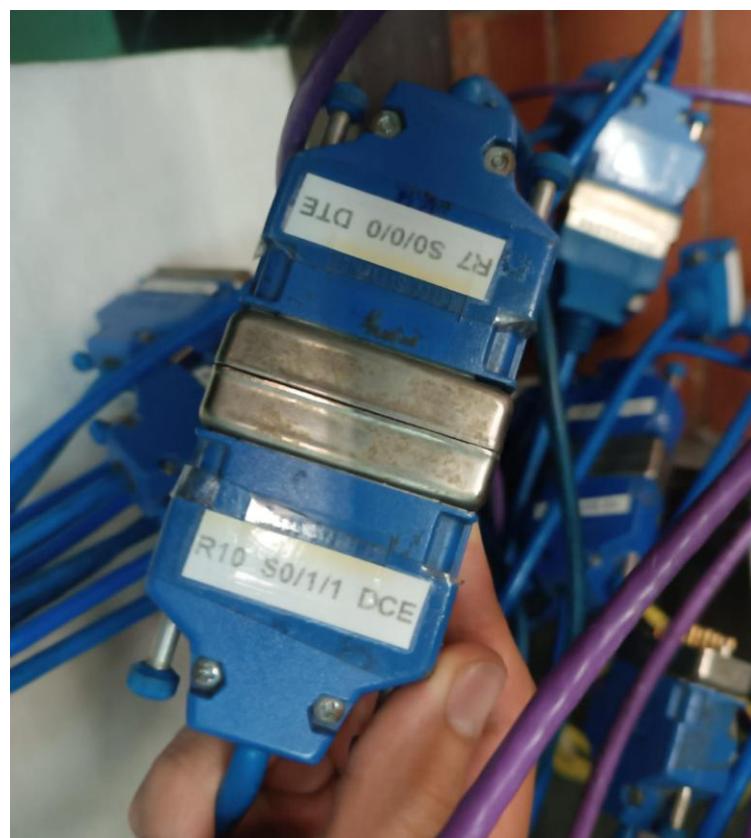
#### 4. Setup - Serial Interconnection

- Using the ping command, verify communication from router 0 and router 1.
- What is a null modem?
  - Es un cable de consola o serial cruzado que se usa para conectar directamente dos dispositivos sin que haya un módem o equipo intermedio.
- What is the clock rate command used for in routers, and why is it needed?
  - El comando clock rate se utiliza para establecer la velocidad de transmisión de datos en una interfaz serial de un router Cisco. Específicamente, se configura en el extremo del enlace que actúa como DCE (Data Communications Equipment), ya que este es el responsable de generar la señal de reloj que sincroniza la comunicación entre los dispositivos conectados.
  - Es necesario porque, en una conexión serial, uno de los dispositivos debe proporcionar una señal de reloj para que la transmisión de datos se realice de forma sincronizada
- What does DTE and DCE mean? What is the relationship with the routers in the Network Labo-ratory?
  - DTE (Data Terminal Equipment)
    - Son los dispositivos que generan, reciben o procesan los datos, como una computadora o un router. En una conexión serial, el DTE depende del DCE para recibir la señal de sincronización.
  - DCE (Data Communications Equipment)
    - Es el equipo de comunicación de datos. Su función principal es proveer la señal de reloj (clock) al enlace serial y establecer la comunicación entre los DTE
  - Cuando conectamos dos routers por una interfaz serial en el laboratorio, uno de ellos debe actuar como DCE para que genere la señal de

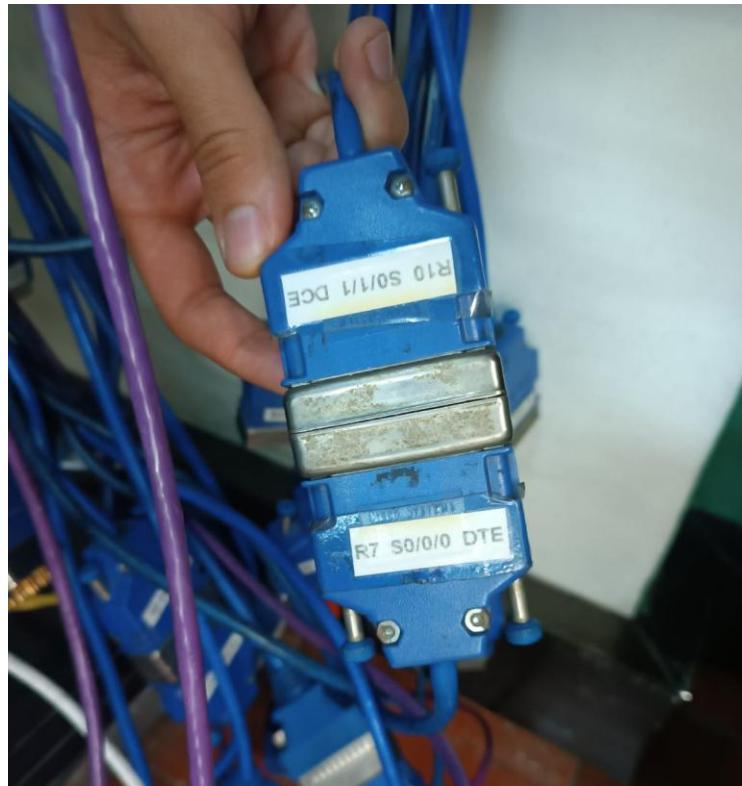
sincronización que permite que ambos se comuniquen correctamente. Esa sincronización se configura usando el comando clock rate en el router que actúa como DCE. Sin esta configuración, la conexión serial no funcionará porque los datos no estarían sincronizados.

## 5. Static Routing

- Configure static routes to see the networks of your groupmates in the other group
  - Primero realizamos la capa física donde conectamos el cable serial de nuestro router (10) con el de nuestro equipo de trabajo (3) donde las interfaces son:
    - Router 10



- Router 7



- Ya con la capa física completa, configuramos de la interfaz de red del serial de nuestro router (10) y nuestro equipo de trabajo realiza la configuración del router (3)
- Para guardar los cambios usamos el comando “write memory” y para verificar que la interfaz este configurada correctamente el comando “show ip interface brief

- Router 10

```
tebinjuanin(config-if)#interface Serial0/1/0
tebinjuanin(config-if)#description "Connection interface Router10"
tebinjuanin(config-if)#ip address 100.0.0.1 255.255.255.0
tebinjuanin(config-if)#no shutdown
tebinjuanin(config-if)#exit
tebinjuanin(config)#write
*Jan 1 01:02:40.943: %LINK-3-UPDOWN: Interface Serial0/1/0, changed state to down
tebinjuanin(config)#write memory
^
% Invalid input detected at '^' marker.

tebinjuanin(config)#exit
tebinjuanin#write
*Jan 1 01:02:46.463: %SYS-5-CONFIG_I: Configured from console by console
tebinjuanin#write memory
Building configuration...
[OK]
```

```

tebinjuanin#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
FastEthernet0/0    90.0.16.1     YES manual up       up
FastEthernet0/1    90.0.4.1      YES manual up       up
FastEthernet0/0/0   unassigned     YES unset  up       down
FastEthernet0/0/1   unassigned     YES unset  up       down
FastEthernet0/0/2   unassigned     YES unset  up       down
FastEthernet0/0/3   unassigned     YES unset  up       down
Serial0/1/0        100.0.0.1     YES manual up       up
Serial0/1/1        unassigned     YES unset administratively down down
Vlan1              unassigned     YES unset  up       down
tebinjuanin#ip route 92.0.4.0 255.255.252.0 100.0.0.2
^
% Invalid input detected at '^' marker.

```

- Router 3

```

Beltran#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Beltran(config)#interface Serial0/0/0
Beltran(config-if)#description "Conexion interfaz router7"
Beltran(config-if)#ip address 100.0.0.2 255.255.255.0
Beltran(config-if)#no shutdown
Beltran(config-if)#exit
Beltran(config)#exit
Beltran#write
*Apr 21 18:41:40.083: *LINK-3-UPDOWN: Interface Serial0/0/0, changed state to down
Beltran#ri
*Apr 21 18:41:40.355: *SYS-5-CONFIG_I: Configured from console by console
Beltran#write memory
Building configuration...
[OK]
Beltran#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
Embedded-Service-Engine0/0 unassigned     YES unset administratively down down
GigabitEthernet0/0   92.0.16.1     YES manual up       up
GigabitEthernet0/1   92.0.4.1      YES manual up       up
Serial0/0/0          100.0.0.2     YES manual down     down
Serial0/0/1          100.0.0.2     YES manual down     down
FastEthernet0/1/0    unassigned     YES unset down     down
FastEthernet0/1/1    unassigned     YES unset down     down
FastEthernet0/1/2    unassigned     YES unset down     down
FastEthernet0/1/3    unassigned     YES unset down     down
Vlan1              unassigned     YES unset down     down
Beltran#

```

- Probamos que la conexión este bien configurada haciendo ping a las direcciones

- Router 10

```

C:\Users\Redes>ping 100.0.0.1

Pinging 100.0.0.1 with 32 bytes of data:
Reply from 100.0.0.1: bytes=32 time<1ms TTL=255
Reply from 100.0.0.1: bytes=32 time<1ms TTL=255
Reply from 100.0.0.1: bytes=32 time=1ms TTL=255
Reply from 100.0.0.1: bytes=32 time=1ms TTL=255

Ping statistics for 100.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

```

- Router 3

```

Command Prompt
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Redes>ping 100.0.0.2

Pinging 100.0.0.2 with 32 bytes of data:
Reply from 100.0.0.2: bytes=32 time<1ms TTL=255

Ping statistics for 100.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

```

- Una vez verificado que todo esté funcionando correctamente. Establecemos las rutas por las cuales los routers van a enviar los paquetes. De esta manera el router sabe a qué red debe mandar el paquete.
- En este caso las rutas del Router 10 va a tener los id de red de nuestro equipo de trabajo (92.0.4.0, 92.0.16.0 ya que trabajamos con otro grupo de lab distinto en horas fuera de la clase para realizar la bitácora) y la ip del Serial tomada por ellos (100.0.0.2) y el router 3 va a tener los id de red de nuestro PC1 y PC2 (90.0.16.0, 90.0.4.0) y nuestro id del Serial (100.0.0.1)
- Nota: Para guardar los cambios ejecutamos el comando “write memory” y para revisar que los cambios estén guardados con el comando “show ip route”

- Router 10

```

Enter configuration commands, one per line. End with CNTL/Z.
tebinjuanin(config)#ip route 92.0.4.0 255.255.252.0 100.0.0.2
tebinjuanin(config)#ip route 92.0.16.0 255.255.240.0 100.0.0.2
tebinjuanin(config)#exit
tebinjuanin#
*Jan  1 01:14:12.903: %SYS-5-CONFIG_I: Configured from console by console
tebinjuanin#write memory
Building configuration...
[OK]
tebinjuanin#

```

```

tebinjuanin#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

      100.0.0.0/24 is subnetted, 1 subnets
C        100.0.0.0 is directly connected, Serial0/1/0
      92.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
S          92.0.16.0/20 [1/0] via 100.0.0.2
S          92.0.4.0/22 [1/0] via 100.0.0.2
      90.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        90.0.16.0/20 is directly connected, FastEthernet0/0
C        90.0.4.0/22 is directly connected, FastEthernet0/1
tebinjuanin#

```

- Router 3

```

Beltran(config)#ip route 90.0.16.0 255.255.240.0 100.0.0.1
Beltran(config)#ip route 90.0.4.0 255.255.252.0 100.0.0.1
Beltran(config)#exit
Beltran#exi
*Apr 21 18:51:45.563: %SYS-5-CONFIG_I: Configured from console by console
Beltran#write memory
Building configuration...
[OK]
Beltran#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      a - application route
      + - replicated route, % - next hop override

Gateway of last resort is not set

      90.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
S          90.0.4.0/22 [1/0] via 100.0.0.1
S          90.0.16.0/20 [1/0] via 100.0.0.1
      92.0.0.0/8 is variably subnetted, 4 subnets, 3 masks
C          92.0.4.0/22 is directly connected, GigabitEthernet0/1
L          92.0.4.1/32 is directly connected, GigabitEthernet0/1
C          92.0.16.0/20 is directly connected, GigabitEthernet0/0
L          92.0.16.1/32 is directly connected, GigabitEthernet0/0
      100.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C          100.0.0.0/24 is directly connected, Serial0/0/0
Beltran#

```

- Verify operation with ping and traceroute commands
  - PC1 (90.0.16.2) to PC1 equipo trabajo (92.0.4.2) and PC 2 equipo trabajo (92.0.16.2)

```
C:\Users\Redes>ping 92.0.16.2

Pinging 92.0.16.2 with 32 bytes of data:
Reply from 92.0.16.2: bytes=32 time=10ms TTL=126

Ping statistics for 92.0.16.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 10ms, Maximum = 10ms, Average = 10ms

C:\Users\Redes>ping 92.0.4.2

Pinging 92.0.4.2 with 32 bytes of data:
Reply from 92.0.4.2: bytes=32 time=10ms TTL=126

Ping statistics for 92.0.4.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 10ms, Maximum = 10ms, Average = 10ms

C:\Users\Redes>
```

- PC1 equipo trabajo (92.0.16.2) to PC1 (90.0.4.2)

```
C:\Users\Redes>ping 90.0.4.2

Pinging 90.0.4.2 with 32 bytes of data:
Reply from 90.0.4.2: bytes=32 time=10ms TTL=126

Ping statistics for 90.0.4.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 10ms, Maximum = 10ms, Average = 10ms

C:\Users\Redes>A|
```

- Interconnect all course group networks and configure the routers so they can see all networks. Define IP ranges for the new serial connections between all
  - Para conectarse con los diferentes grupos, primero se definió las direcciones por las cuales iban a salir los routers (100.0.0.x donde x no se podía repetir entre routers) y cuáles de estos se iban a conectar entre si

- En nuestro caso, conectamos otro cable de serial con los compañeros justamente alado de nosotros (los cuales tienen id de la red 91.0.0.0) y hacemos de puente con la red 92.0.0.0 la cual ya estaba conectada a nosotros en los pasos anteriores
- Aun así, volvemos a probar la conexión con la red 92.0.0.0 para verificar que este configurado correctamente.

```
C:\Users\Redes>tracert 92.0.4.2

Tracing route to 92.0.4.2 over a maximum of 30 hops

 1    <1 ms    <1 ms    <1 ms  90.0.16.1
 2      1 ms      1 ms      1 ms  100.0.0.10
 3      2 ms      1 ms      1 ms  92.0.4.2

Trace complete.
```

- Ahora, para conectarnos a la otra red, configuramos la otra interfaz del cable serial del router

```
tebinjuanin(config)#interface Serial0/1/0
tebinjuanin(config-if)#ip address 100.0.0.12 255.255.255.0
tebinjuanin(config-if)#no shutdown
```

- Una vez configurado el serial, configuramos las rutas que nos permiten llegar a las máquinas de nuestros compañeros
- Verificamos que estén configuradas las dos rutas, tanto como la de la red 91 y la red 92

```
tebinjuanin(config)#ip route 91.0.16.0 255.255.240.0 100.0.0.13
tebinjuanin(config)#ip route 91.0.4.0 255.255.252.0 100.0.0.13
```

```

tebinjuanin#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

      100.0.0.0/24 is subnetted, 1 subnets
C        100.0.0.0 is directly connected, Serial0/1/1
      92.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
S          92.0.16.0/20 [1/0] via 100.0.0.10
S          92.0.4.0/22 [1/0] via 100.0.0.10
      91.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
S          91.0.16.0/20 [1/0] via 100.0.0.13
S          91.0.4.0/22 [1/0] via 100.0.0.13
      90.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        90.0.16.0/20 is directly connected, FastEthernet0/0
C        90.0.4.0/22 is directly connected, FastEthernet0/1
tebinjuanin#ping 91.0.16.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 91.0.16.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
tebinjuanin#

```

- Realizamos pruebas de ping/tracert desde nuestra máquina para verificar la conexión

```

C:\Users\Redes>tracert 91.0.0.2

Tracing route to 91.0.0.2 over a maximum of 30 hops

  1    <1 ms      <1 ms      <1 ms  90.0.16.1
  2    11 ms       11 ms      11 ms  100.0.0.13
  3    13 ms       13 ms      13 ms  91.0.0.2

Trace complete.

C:\Users\Redes>

```

```
C:\Users\Redes>ping 91.0.0.2

Pinging 91.0.0.2 with 32 bytes of data:
Reply from 91.0.0.2: bytes=32 time=9ms TTL=126

Ping statistics for 91.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 9ms, Maximum = 9ms, Average = 9ms
```

- En el laboratorio existieron los siguientes problemas técnicos
  - El grupo anterior no borro la configuración del router y con ayuda del profesor logramos resetear esta después de un largo tiempo
  - Cuando ya teníamos configurado la mayoría de la configuración, hubo un apagón en las maquinas y al momento de prender el router nuevamente se borró la configuración establecida a pesar de realizar el comando copy running-config startup-config (Averiguando más sobre esto, al momento de resetear el router de manera “forzada” faltó ejecutar un comando, sin este se iba a borrar la configuración)
- Por estas razones, no alcanzamos a conectar todos los grupos de trabajo. Solo se interconectaron entre nuestra mesa de trabajo y con la de la parte izquierda a nosotros (red 92 y 91 respectivamente)

## 6. Closure

- Erase and organize the routers and equipment used so that they are ready for a new practice session.

- Ejecutamos el comando erase startup-config para borrar toda la configuración de la red

```
tebinjuanin#erase startup-config
Erasing the nvram filesystem will remove all configuration files! Continue? [c]
[OK]
Erase of nvram: complete
```

- Luego reiniciamos el router con el comando “reload” para borrar la configuración definitivamente

```
tebinjuanin#reload
-
-
System configuration has been modified. Save? [yes/no]: no
Proceed with reload? [confirm]
```

- Ingresamos nuevamente al router y verificamos que no haya ninguna configuración de red guardada

```
Router>show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
FastEthernet0/0    unassigned     YES unset administratively down down
FastEthernet0/1    unassigned     YES unset administratively down down
FastEthernet0/0/0   unassigned     YES unset up      down
FastEthernet0/0/1   unassigned     YES unset up      down
FastEthernet0/0/2   unassigned     YES unset up      down
FastEthernet0/0/3   unassigned     YES unset up      down
Serial0/1/0        unassigned     YES unset administratively down down
Serial0/1/1        unassigned     YES unset administratively down down
Vlan1             unassigned     YES unset up      down
Router>
```

## CONCLUSIONS

- Arranque de Routers y Gestión de la Configuración
  - Durante el laboratorio aprendimos cómo se lleva a cabo el proceso de inicio de los routers Cisco, reconociendo la función de las distintas memorias como RAM, ROM, Flash y NVRAM. También comprendimos las diferencias entre las configuraciones en ejecución (running-config) y las almacenadas (startup-config), así como la utilidad de los modos de arranque 0x2102 y 0x2142 para el mantenimiento y recuperación de accesos administrativos.
- Subredes y Enrutamiento Manual
  - La configuración de rutas estáticas en los routers nos permitió enlazar redes no directamente conectadas, fortaleciendo nuestros conocimientos en direccionamiento IP y uso de subredes. Aprendimos a manipular las tablas de ruteo de forma manual y a validar la conectividad mediante comandos como ping y traceroute, esenciales para asegurar la correcta transmisión de datos entre dispositivos.
- Infraestructura Serial y Comunicación Punto a Punto
  - Exploramos las conexiones seriales entre routers, entendiendo la diferencia entre los dispositivos DTE y DCE. Identificamos cuándo es necesario aplicar el comando clock rate y verificamos la comunicación entre routers a través de enlaces punto a punto. Esta práctica reforzó la importancia de una correcta configuración física y lógica en interconexiones de red.
- Aplicaciones en la Nube y Métricas en Tiempo Real:
  - Al trabajar con Azure y Application Insights, observamos cómo se pueden monitorear aplicaciones desplegadas en la nube, obteniendo métricas detalladas sobre rendimiento, disponibilidad y errores. Esta capacidad es fundamental en entornos empresariales para garantizar una buena experiencia de usuario y detectar fallos antes de que impacten en los servicios.
- Diagnóstico de Red con ICMP y Traceroute:
  - Mediante herramientas como ping, traceroute y servicios en línea, visualizamos las rutas que siguen los paquetes y analizamos el estado de la conectividad en tiempo real. Estas herramientas nos brindan una forma efectiva de identificar fallos, cuellos de botella y pérdidas de paquetes en la red.
- Monitoreo de Infraestructura con SNMP
  - Implementamos herramientas de monitoreo de red utilizando el protocolo SNMP, que nos permitieron supervisar recursos como CPU, memoria, espacio en disco y tráfico. Aprendimos que estas soluciones son esenciales para prevenir fallas, mejorar el rendimiento de los sistemas y gestionar redes de forma centralizada y eficiente.
  -

## BIBLIOGRAPHY

- ServiceNow. (s.f.). *¿Qué es la infraestructura de red?*  
<https://www.servicenow.com/latam/products/it-asset-management/what-is-network-infrastructure.htmlServiceNow>
- Azion. (s.f.). *¿Qué es la capa de red? | Capa de red en el modelo OSI.*  
<https://www.azion.com/es/learning/network-layer/que-es-la-capa-de-red/Azion Technologies>
- AWS. (s.f.). *IPv4 vs IPv6: Diferencia entre las versiones del protocolo de Internet.*  
[https://aws.amazon.com/es/compare/the-difference-between-ipv4-and-ipv6/&#8203;:contentReference\[oaicite:6\]{index=6}](https://aws.amazon.com/es/compare/the-difference-between-ipv4-and-ipv6/&#8203;:contentReference[oaicite:6]{index=6})
- KeepCoding. (2025). *Subnetting: ¿Qué es y para qué sirve? - Guía 2025.*  
[https://keepcoding.io/blog/que-es-subnetting/&#8203;:contentReference\[oaicite:7\]{index=7}](https://keepcoding.io/blog/que-es-subnetting/&#8203;:contentReference[oaicite:7]{index=7})
- OpenWebinars. (s.f.). *Enrutamiento estático vs dinámico.*  
[https://openwebinars.net/blog/enrutamiento-estatico-vs-dinamico/&#8203;:contentReference\[oaicite:8\]{index=8}](https://openwebinars.net/blog/enrutamiento-estatico-vs-dinamico/&#8203;:contentReference[oaicite:8]{index=8})
- Wikipedia. (s.f.). *Módem nulo.*  
[https://es.wikipedia.org/wiki/M%C3%B3dem\\_nulo&#8203;:contentReference\[oaicite:9\]{index=9}](https://es.wikipedia.org/wiki/M%C3%B3dem_nulo&#8203;:contentReference[oaicite:9]{index=9})
- Institut Sa Palomera. (s.f.). *3.1.1.8 DTE-DCE.*  
[https://www.sapalomera.cat/moodlecf/RS/4/course/module3/3.1.1.8/3.1.1.8.html&#8203;:contentReference\[oaicite:10\]{index=10}](https://www.sapalomera.cat/moodlecf/RS/4/course/module3/3.1.1.8/3.1.1.8.html&#8203;:contentReference[oaicite:10]{index=10})
- Cloudflare. (s.f.). *¿Qué es el Protocolo de control de mensajes de Internet (ICMP)?*  
[https://www.cloudflare.com/es-es/learning/ddos/glossary/internet-control-message-protocol-icmp/&#8203;:contentReference\[oaicite:11\]{index=11}](https://www.cloudflare.com/es-es/learning/ddos/glossary/internet-control-message-protocol-icmp/&#8203;:contentReference[oaicite:11]{index=11})
- ManageEngine. (s.f.). *Tutorial: Fundamentos de SNMP | ¿Qué es SNMP?*  
[https://www.manageengine.com/latam/network-monitoring/tutorial-fundamentos-protocolo-snmp.html&#8203;:contentReference\[oaicite:12\]{index=12}](https://www.manageengine.com/latam/network-monitoring/tutorial-fundamentos-protocolo-snmp.html&#8203;:contentReference[oaicite:12]{index=12})
- Microsoft Azure. (s.f.). *¿Qué es la nube: definición?* [https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-the-cloud/&#8203;:contentReference\[oaicite:13\]{index=13}](https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-the-cloud/&#8203;:contentReference[oaicite:13]{index=13})