

LABORATORY NO.06 - Application Layer Protocols, Base  
Platform, and Network Layer



ELABORADO POR:

ESTEBAN AGUILERA CONTRERAS  
JUAN DAVID RODRIGUEZ RODRIGUEZ

PROFESOR(ES):

JOHN PACHON

RECO  
2025-1

## **OBJECTIVES**

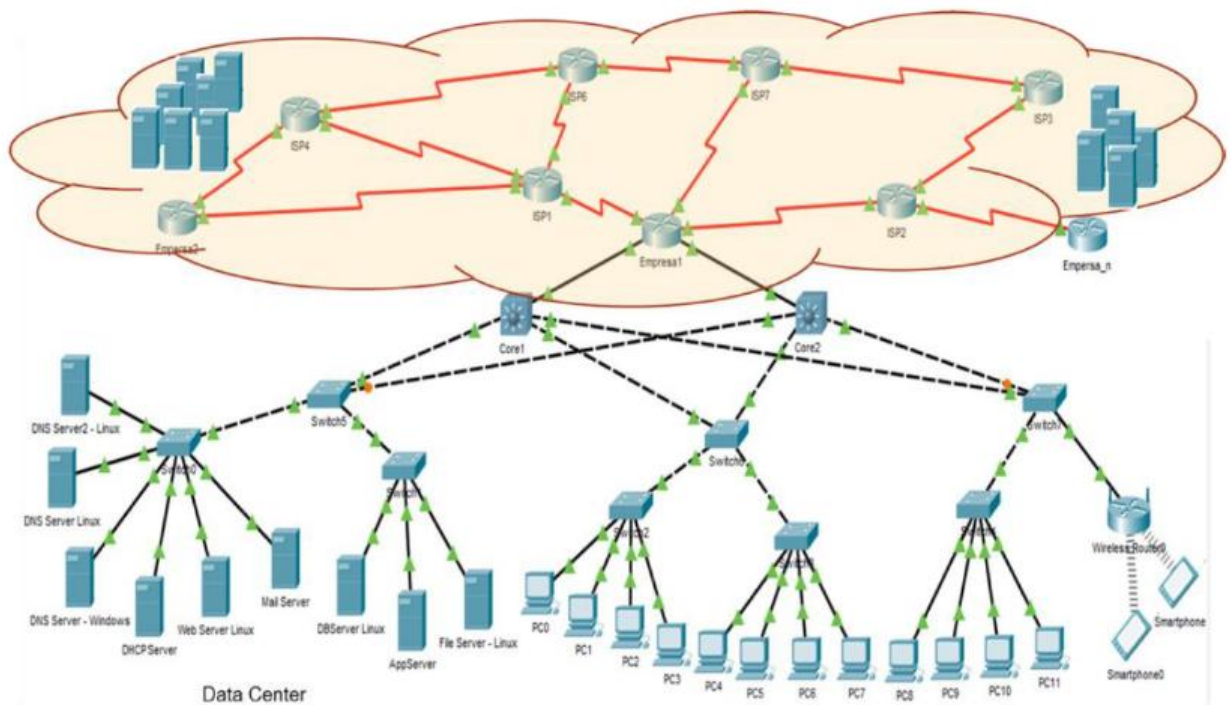
- Install and configure basic software for web servers

## **TOOLS TO BE USED**

- Computers
- Internet Access

## CONTEXT

We are continuing to work on the infrastructure of a company that typically hosts several IT infrastructure services. This infrastructure includes both wired and wireless user workstations, as well as physical and virtualized servers, all interconnected through Layer 2 and Layer 3 switches, wireless devices, and routers that provide Internet access. Additionally, it is common to have cloud infrastructures from which resources are provisioned based on the organization's needs. Among the servers, you can find web services, DNS, email, databases, storage, and applications, among others. Let's recall the base configuration we are using



In this lab, we will focus on server infrastructure.

## INTRODUCTION

En este laboratorio trabajamos con diferentes servicios de red que hacen parte de la infraestructura de una empresa. Comenzamos con la configuración de routers en Packet Tracer, donde asignamos direcciones IP, configuramos rutas y probamos la conexión entre diferentes equipos. Esta parte está relacionada con la capa de red, ya que nos permite establecer la comunicación entre dispositivos dentro de una red.

Luego pasamos a la instalación de servidores web en diferentes sistemas operativos (netBSD, slackware y Windows ). En cada uno configuramos servicios como Apache, Nginx e IIS, y creamos páginas web básicas para probar su funcionamiento. También configuramos el servicio DNS para que los sitios puedan ser accesibles por nombre. Esta parte corresponde a la capa de aplicación, ya que trabajamos con servicios que los usuarios finales usan directamente, como el acceso a páginas web.

## THEORETICAL FRAMEWORK

La infraestructura de red es el conjunto de dispositivos, servicios y protocolos que permiten la comunicación dentro de una organización. Este sistema incluye estaciones de trabajo, servidores, dispositivos de red, y servicios en la nube. Cada uno cumple un papel específico para que la información pueda transmitirse de forma rápida y segura.

- Componentes de una Red
  - Estaciones de trabajo: Son los computadores físicos o virtuales que los usuarios utilizan para acceder a los servicios de la red, ejecutar programas y conectarse a Internet.
  - Servidores: Equipos dedicados a proporcionar recursos a otros dispositivos. Pueden ofrecer servicios como páginas web, bases de datos, correo electrónico o almacenamiento. Pueden estar alojados localmente o en la nube.
- Dispositivos de red:
  - Switches de Capa 2: Interconectan dispositivos dentro de una red local (LAN) usando direcciones MAC.
  - Routers de Capa 3: Encargados de dirigir el tráfico entre distintas redes usando direcciones IP. También pueden implementar funciones como:
    - NAT (Network Address Translation): Técnica que permite que varios dispositivos de una red privada compartan una sola IP pública para acceder a Internet.
    - ACL (Access Control List): Permite filtrar tráfico en función de direcciones IP, puertos o protocolos, para mejorar la seguridad.
- Simulación de Redes

Para la parte práctica del laboratorio se utilizó Cisco Packet Tracer, una herramienta que permite simular redes de forma virtual. Dentro del simulador se usan elementos como:

- Router 1941: Dispositivo virtual que permite hacer configuraciones reales como asignación de IP, enrutamiento estático, NAT, entre otros.
- PC-PT: Computadoras virtuales que permiten ejecutar comandos y realizar pruebas.
- Módulo Multiusuario: Función que permite interconectar simulaciones de diferentes estudiantes para simular una red más grande.

Al configurar la red, se usaron comandos como:

- ping: Verifica si existe conectividad entre dos dispositivos. Si responde, indica que los paquetes están llegando correctamente.
- traceroute (o tracert en Windows): Muestra la ruta que siguen los paquetes desde el origen hasta el destino, identificando todos los "saltos" entre routers.

- Capa de Aplicación y Protocolos

La capa de aplicación (capa 7 del modelo OSI) es la que permite al usuario interactuar directamente con los servicios de red. En este laboratorio se trabajaron principalmente los siguientes protocolos:

- HTTP (Hypertext Transfer Protocol): Protocolo que permite la visualización de páginas web. Funciona bajo un esquema de solicitud y respuesta, y utiliza el puerto 80.
- DNS (Domain Name System): Sistema que traduce nombres de dominio a direcciones IP. Esto hace que sea más fácil para los usuarios acceder a sitios web sin tener que recordar números.

- Servidores Web

Durante el laboratorio se instalaron y configuraron distintos servidores web en varios sistemas operativos:

- Apache (NetBSD): Servidor de código abierto muy utilizado por su flexibilidad. Permite configurar múltiples sitios mediante virtual hosts.
- Nginx (Slackware): Servidor web ligero, ideal para manejar muchas conexiones al mismo tiempo. También se puede usar como proxy.
- IIS (Internet Information Services) (Windows Server): Servidor web propio de Microsoft. Ofrece una interfaz gráfica para la gestión de sitios web y servicios.

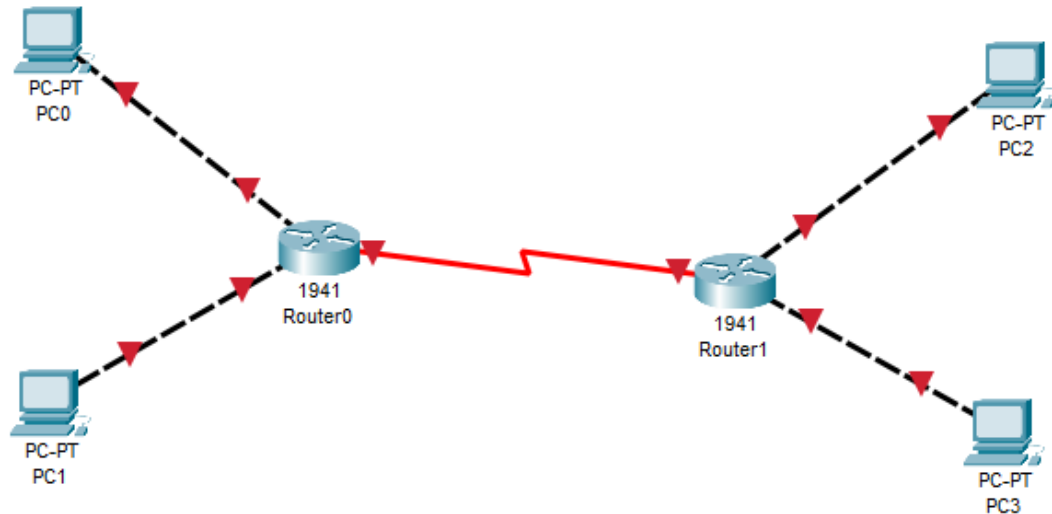
- Hosting y Virtual Hosts

El servicio de hosting consiste en alojar páginas web dentro de un servidor para que puedan ser accesibles desde Internet o una red local. Para poder alojar varios sitios en un mismo servidor, se usan los virtual hosts, que son configuraciones que permiten distinguir entre diferentes sitios según el nombre del dominio.

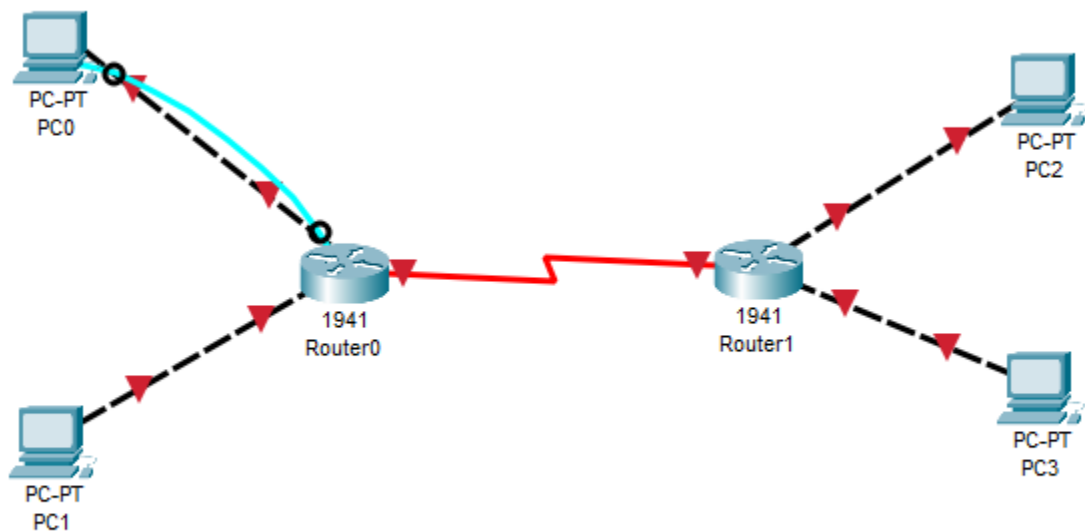
- Virtual Hosts por nombre: Varios sitios comparten una misma dirección IP, y se diferencian por el nombre de dominio.
- Virtual Hosts por IP: Cada sitio tiene una IP diferente. Aunque es menos común, puede usarse cuando se necesitan certificados SSL distintos por sitio.

## ROUTER SIMULATION

- Open Packet Tracer and create a network following the diagram

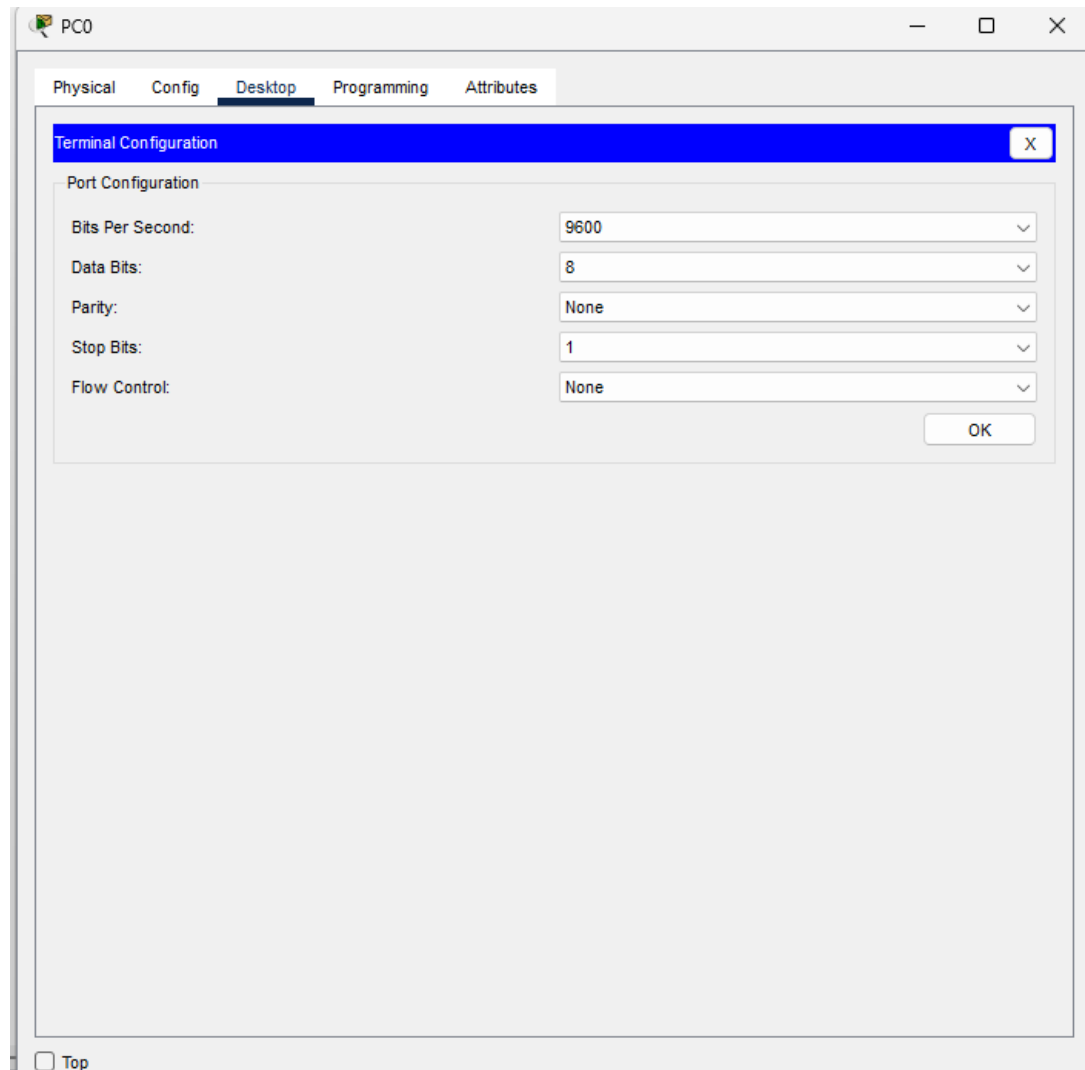


- Now, to configure it, connect from a console similarly to how you do with switches. (The blue cable is a console cable and connects to the serial port – RS232 of each device).



- Configure the networks to allow addressing the specified hosts. Networks:
  - Entramos a la terminal





- Accedemos la modo de configuración global

```
Router>enable
Router#configure terminal
```

- Cambiamos el nombre del router a Esteban por comodidad

```
Router(config)#hostname Esteban
```

- Configuramos la linea de consola

```
Esteban(config)#line console 0
Esteban(config-line)#logging synchronous
Esteban(config-line)#password root123
Esteban(config-line)#login
Esteban(config-line)#exit
```

Donde:

- line console 0: Configura la línea de consola.
- logging synchronous: Evita que los mensajes del sistema interrumpen la escritura.
- password Clave\_C: Establece la contraseña **Clave\_C** para acceder desde la consola.
- login: Requiere una contraseña para acceder.
- Configurar acceso remoto por Telnet o SSH:

```
Esteban(config)#line vty 0 15
Esteban(config-line)#logging synchronous
Esteban(config-line)#password root123
Esteban(config-line)#login
Esteban(config-line)#exit
```

Donde

- line vty 0 15: Configura acceso remoto para 16 sesiones (0 a 15).
- password Clave\_T: Establece la contraseña root123 **para Telnet/SSH.**
- login: Requiere autenticación.
- Deshabilitamos la búsqueda de nombres DNS con el fin de evitar que el router intente resolver nombres mal escritos como si fueran dominios.

```
Esteban(config)#no ip domain-lookup
Esteban(config)#
```

- Ahora, desde el pc 1 configuramos las IP de los pc\_0 y pc\_1 del primer router
- Para juan, el uso student 1
  - 132.18.0.0/16
  - 132.19.0.0/16
- Para Esteban se uso student 2
  - 72.0.0.0/8
  - 73.0.0.0/8
- Para configurar las ip, entramos a cada interfaz y agregamos la ip correspondiente
  - Para el caso de juan:

```

Juanito(config)#interface GigabitEthernet 0/0
Juanito(config-if)#description "Coneccion Router0 with PC0"
Juanito(config-if)#ip address 132.18.0.1 255.255.0.0
Juanito(config-if)#no shutdown

Juanito(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Juanito(config-if)#interface GigabitEthernet0/1
Juanito(config-if)#description "Coneccion Router0 to PC1"
Juanito(config-if)#ip address 132.19.0.1 255.255.0.0
Juanito(config-if)#no shutdown

Juanito(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

```

- Para el caso de esteban:

```

Esteban(config-if)#interface GigabitEthernet0/0
Esteban(config-if)#description "Connection Router0 with PC0"
Esteban(config-if)#ip address 72.0.0.1 255.0.0.0
Esteban(config-if)#no shutdown
Esteban(config-if)#interface GigabitEthernet0/1
Esteban(config-if)#description "Connection Router0 with PC1"
Esteban(config-if)#ip address 73.0.0.1 255.0.0.0
Esteban(config-if)#no shutdown
Esteban(config-if)#exit

```

- Luego de configurar las ip de los pc, configuramos el serial que en el caso de los dos estudiantes es el mismo

```

Juanito(config-if)#exit
Juanito(config)#interface Serial0/1/0
Juanito(config-if)#ip address 20.0.0.1 255.0.0.0
Juanito(config-if)#description "Configuracion interface Router0"
Juanito(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/1/0, changed state to down

```

```

Esteban(config)#interface Serial0/1/0
Esteban(config-if)#ip address 20.0.0.1 255.0.0.0
Esteban(config-if)#description "Configuration interface Router0"
Esteban(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/1/0, changed state to down
Esteban(config-if)#

```

- Se establece una contraseña segura para acceder al modo administrador en el dispositivo
  - Para el caso de Juan

```

Juanito2(config-if)#exit
Juanito2(config)#enable secret root1234
Juanito2(config)#exit
Juanito2#
%SYS-5-CONFIG_I: Configured from console by console

Juanito2#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]

```

- Para el caso de Esteban

```

Esteban(config)#enable secret root123
Esteban(config)#exit
Esteban#
%SYS-5-CONFIG_I: Configured from console by console

Esteban#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Esteban#

```

- Repetimos el proceso pero en este caso con el router 1 y la maquina que lo va configurar es el pc03
- Configuramos la línea de consola

- Para el caso de Juan

```

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname Juanito2
Juanito2(config)#banner motd "Uso exclusivo para requisito del host2"
Juanito2(config)#line console 0
Juanito2(config-line)#logging synchronous
Juanito2(config-line)#password Clave_C
Juanito2(config-line)#login
Juanito2(config-line)#exit
Juanito2(config)#no ip domain-lookup

```

- Para el caso de Esteban

```

Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname Esteban2
Esteban2(config)#line console 0
Esteban2(config-line)#logging synchronous
Esteban2(config-line)#password root1234
Esteban2(config-line)#login
Esteban2(config-line)#exit
Esteban2(config)#line vty 0 15
Esteban2(config-line)#logging synchronous
Esteban2(config-line)#password root1234
Esteban2(config-line)#login
Esteban2(config-line)#exit
Esteban2(config)#no ip domain-lookup
Esteban2(config)#

```

- Las ip de Juan en este caso son:
  - 132.20.0.0/16
  - 132.21.0.0/16
- Las ip de Esteban en este caso son:
  - 74.0.0.0/8
  - 75.0.0.0/8
- Para configurar las ip, entramos a cada interfaz y agregamos la ip correspondiente
  - Para el caso de juan:

```

Juanito2(config-if)#interface GigabitEthernet 0/0
Juanito2(config-if)#description "Connection Router1 with PC2"
Juanito2(config-if)#ip address 132.20.0.1 255.255.0.0
Juanito2(config-if)#no shutdown

Juanito2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

```

```

Juanito2(config-if)#exit
Juanito2(config)#interface GigabitEthernet 0/1
Juanito2(config-if)#description "Coneccion a Router1 con PC3"
Juanito2(config-if)#ip address 132.21.0.1 255.255.0.0
Juanito2(config-if)#no shutdown

Juanito2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

```

- Para el caso de esteban:

```

Esteban2(config)#interface GigabitEthernet 0/0
Esteban2(config-if)#description "Connection Router1 with PC2"
Esteban2(config-if)#ip address 74.0.0.1 255.0.0.0
Esteban2(config-if)#no shutdown

Esteban2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Esteban2(config-if)#interface GigabitEthernet 0/1
Esteban2(config-if)#description "Connection Router1 with PC3"
Esteban2(config-if)#ip address 75.0.0.1 255.0.0.0
Esteban2(config-if)#no shutdown

Esteban2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

```

- Luego de configurar las ip de los pc, configuramos el serial que en el caso de los dos estudiantes es el mismo

```

Juanito2(config-if)#exit
Juanito2(config)#interface Serial0/1/0
Juanito2(config-if)#ip address 20.0.0.2 255.0.0.0
Juanito2(config-if)#description "Configuracion interface Router1"
Juanito2(config-if)#no shutdown

Juanito2(config-if)#
%LINK-5-CHANGED: Interface Serial0/1/0, changed state to up

```

```

Esteban2(config-if)#interface Serial0/1/0
Esteban2(config-if)#ip address 20.0.0.2 255.0.0.0
Esteban2(config-if)#description "Configuration interface Router1"
Esteban2(config-if)#no shutdown

Esteban2(config-if)#
%LINK-5-CHANGED: Interface Serial0/1/0, changed state to up

```

- Se establece una contraseña segura para acceder al modo administrador en el dispositivo

- Para el caso de Juan

```

Juanito2(config-if)#exit
Juanito2(config)#enable secret Clave_E
Juanito2(config)#exit
Juanito2#
%SYS-5-CONFIG_I: Configured from console by console

```

- Para el caso de Esteban

```

Esteban2(config-if)#exit
Esteban2(config)#enable secret root123
Esteban2(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1/0, changed state to up

Esteban2(config)#exit
Esteban2#
%SYS-5-CONFIG_I: Configured from console by console

Esteban2#

```

- Ahora, ingresamos las configuraciones IP respectivas en cada PC. Asignamos a cada computadora su parte correspondiente de la red y la configuramos para que termine en .2, ya que .0 es el identificador de red y .1 es la puerta de enlace configurada en el enrutador.

- Para Juan

- PC0

PC0

Physical Config **Desktop** Programming Attributes

IP Configuration [X]

Interface: FastEthernet0

IP Configuration

☐ DHCP ☒ Static

IPv4 Address: 132.18.0.2

Subnet Mask: 255.255.0.0

Default Gateway: 132.18.0.1

DNS Server: 0.0.0.0

IPv6 Configuration

☐ Automatic ☒ Static

IPv6 Address: /

Link Local Address: FE80::290:21FF:FE52:6AC4

Default Gateway:

DNS Server:

802.1X

☐ Use 802.1X Security

Authentication: MD5

Username:

Password:

☐ Top

- PC1

PC1

Physical Config Desktop Programming Attributes

IP Configuration X

Interface FastEthernet0

IP Configuration

☐ DHCP ☒ Static

IPv4 Address 132.19.0.2

Subnet Mask 255.255.0.0

Default Gateway 132.19.0.1

DNS Server 0.0.0.0

IPv6 Configuration

☐ Automatic ☒ Static

IPv6 Address /

Link Local Address FE80::20A:41FF:FE88:76C8

Default Gateway

DNS Server

802.1X

☐ Use 802.1X Security

Authentication MD5

Username

Password

Top

- PC2



PC2

Physical Config **Desktop** Programming Attributes

**IP Configuration** [X]

Interface: FastEthernet0

IP Configuration

☐ DHCP ☒ Static

IPv4 Address: 132.21.0.2

Subnet Mask: 255.255.0.0

Default Gateway: 132.21.0.1

DNS Server: 0.0.0.0

IPv6 Configuration

☐ Automatic ☒ Static

IPv6 Address: /

Link Local Address: FE80::210:11FF:FE3D:3C04

Default Gateway:

DNS Server:

802.1X

☐ Use 802.1X Security

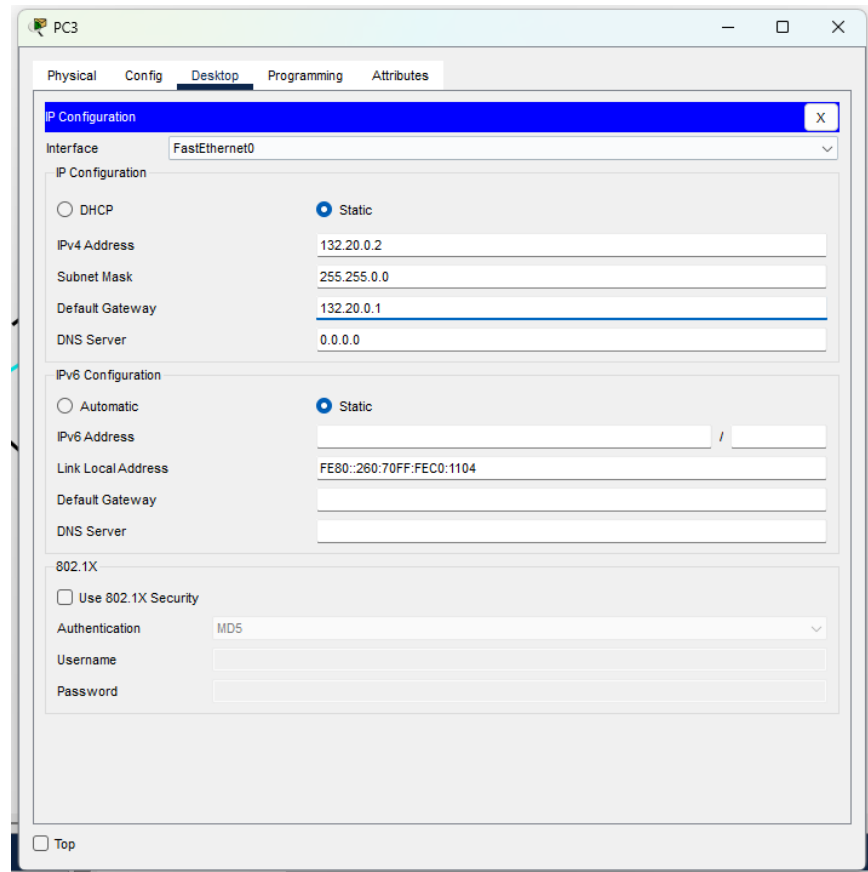
Authentication: MD5

Username:

Password:

☐ Top

- PC3



- Para Esteban
  - PC0

PC0

Physical Config **Desktop** Programming Attributes

IP Configuration X

Interface FastEthernet0

IP Configuration

☐ DHCP ☒ Static

IPv4 Address 72.0.0.2

Subnet Mask 255.0.0.0

Default Gateway 72.0.0.1

DNS Server 0.0.0.0

IPv6 Configuration

☐ Automatic ☒ Static

IPv6 Address /

Link Local Address FE80::260:3EFF:FEA5:BB0C

Default Gateway

DNS Server

802.1X

☐ Use 802.1X Security

Authentication MD5

Username

Password

- PC1

PC1

Physical Config **Desktop** Programming Attributes

IP Configuration X

Interface FastEthernet0

IP Configuration

☐ DHCP ☒ Static

IPv4 Address 73.0.0.2

Subnet Mask 255.0.0.0

Default Gateway 73.0.0.1

DNS Server 0.0.0.0

IPv6 Configuration

☐ Automatic ☒ Static

IPv6 Address /

Link Local Address FE80::205:5EFF:FE88:4D31

Default Gateway

DNS Server

802.1X

☐ Use 802.1X Security

Authentication MD5

Username

Password

- PC2

PC2

Physical Config Desktop Programming Attributes

IP Configuration [X]

Interface: FastEthernet0

IP Configuration

☐ DHCP ☒ Static

IPv4 Address: 74.0.0.2

Subnet Mask: 255.0.0.0

Default Gateway: 74.0.0.1

DNS Server: 0.0.0.0

IPv6 Configuration

☐ Automatic ☒ Static

IPv6 Address: /

Link Local Address: FE80::290:2BFF:FE6B:722E

Default Gateway:

DNS Server:

802.1X

☐ Use 802.1X Security

Authentication: MD5

Username:

Password:

- PC3

PC3

Physical Config Desktop Programming Attributes

IP Configuration [X]

Interface: FastEthernet0

IP Configuration

☐ DHCP ☒ Static

IPv4 Address: 75.0.0.2

Subnet Mask: 255.0.0.0

Default Gateway: 75.0.0.1

DNS Server: 0.0.0.0

IPv6 Configuration

☐ Automatic ☒ Static

IPv6 Address: /

Link Local Address: FE80::203:E4FF:FEC8:E79B

Default Gateway:

DNS Server:

802.1X

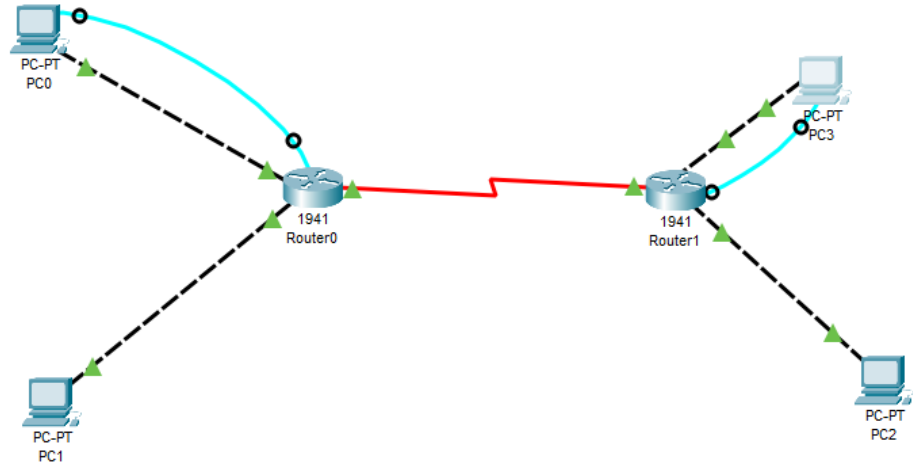
☐ Use 802.1X Security

Authentication: MD5

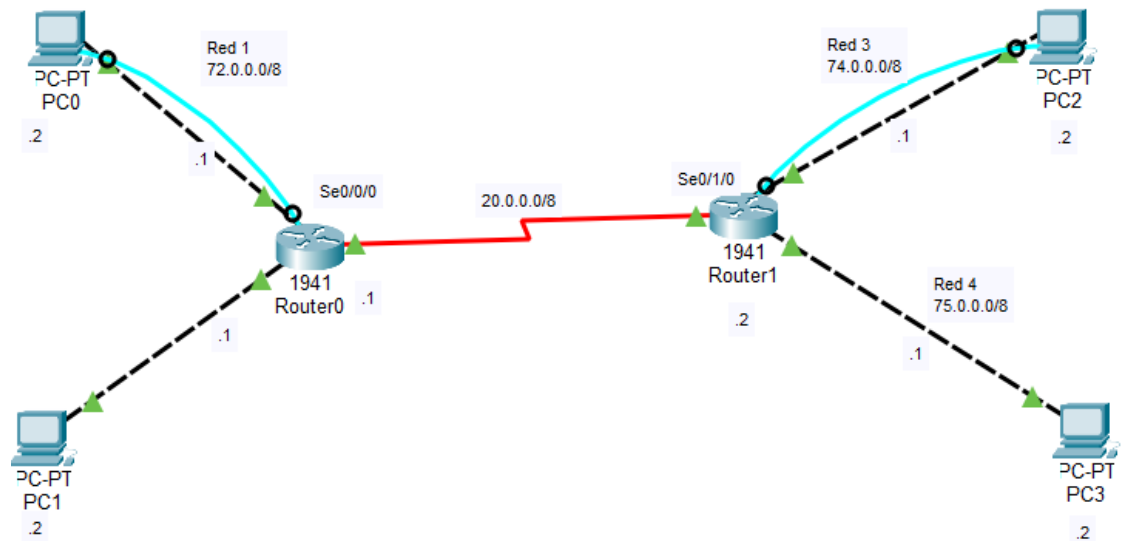
Username:

Password:

- Una vez configuradas los routers, veremos que la conexión entre las maquinas y estos esta establecida
  - Juan



- Esteban



- Test connectivity between the PCs on the same LAN and to different networks. Which ones work and which ones don't?
  - Test juan
    - PC0
      - Test PC1

```
C:\>ping 132.19.0.2

Pinging 132.19.0.2 with 32 bytes of data:

Reply from 132.19.0.2: bytes=32 time<1ms TTL=127
Reply from 132.19.0.2: bytes=32 time<1ms TTL=127
Reply from 132.19.0.2: bytes=32 time<1ms TTL=127
Reply from 132.19.0.2: bytes=32 time<1ms TTL=127

Ping statistics for 132.19.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- Test PC2

```
C:\>ping 132.20.0.2

Pinging 132.20.0.2 with 32 bytes of data:

Reply from 132.18.0.1: Destination host unreachable.
Reply from 132.18.0.1: Destination host unreachable.
Reply from 132.18.0.1: Destination host unreachable.
Reply from 132.18.0.1: Destination host unreachable.

Ping statistics for 132.20.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test PC3

```
C:\>ping 132.21.0.2

Pinging 132.21.0.2 with 32 bytes of data:

Reply from 132.18.0.1: Destination host unreachable.
Reply from 132.18.0.1: Destination host unreachable.
Reply from 132.18.0.1: Destination host unreachable.
Reply from 132.18.0.1: Destination host unreachable.

Ping statistics for 132.21.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test router0

```
C:\>ping 132.18.0.1

Pinging 132.18.0.1 with 32 bytes of data:

Reply from 132.18.0.1: bytes=32 time<1ms TTL=255
Reply from 132.18.0.1: bytes=32 time<1ms TTL=255
Reply from 132.18.0.1: bytes=32 time<1ms TTL=255
Reply from 132.18.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 132.18.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- PC1
  - Test PC0

```
C:\>ping 132.18.0.2

Pinging 132.18.0.2 with 32 bytes of data:

Reply from 132.18.0.2: bytes=32 time<1ms TTL=127
Reply from 132.18.0.2: bytes=32 time<1ms TTL=127
Reply from 132.18.0.2: bytes=32 time<1ms TTL=127
Reply from 132.18.0.2: bytes=32 time<1ms TTL=127

Ping statistics for 132.18.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- Test PC2

```
C:\>ping 132.20.0.2

Pinging 132.20.0.2 with 32 bytes of data:

Reply from 132.19.0.1: Destination host unreachable.
Reply from 132.19.0.1: Destination host unreachable.
Reply from 132.19.0.1: Destination host unreachable.
Reply from 132.19.0.1: Destination host unreachable.

Ping statistics for 132.20.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test PC3

```
C:\>ping 132.21.0.2

Pinging 132.21.0.2 with 32 bytes of data:

Reply from 132.19.0.1: Destination host unreachable.
Request timed out.
Reply from 132.19.0.1: Destination host unreachable.
Reply from 132.19.0.1: Destination host unreachable.

Ping statistics for 132.21.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test router0

```

Cisco Packet Tracer PC Command Line 1.0
C:\>ping 132.19.0.1

Pinging 132.19.0.1 with 32 bytes of data:

Reply from 132.19.0.1: bytes=32 time<1ms TTL=255
Reply from 132.19.0.1: bytes=32 time<1ms TTL=255
Reply from 132.19.0.1: bytes=32 time<1ms TTL=255
Reply from 132.19.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 132.19.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

```

- PC2
  - Test PC0

```

C:\>ping 132.18.0.2

Pinging 132.18.0.2 with 32 bytes of data:

Reply from 132.20.0.1: Destination host unreachable.
Reply from 132.20.0.1: Destination host unreachable.
Reply from 132.20.0.1: Destination host unreachable.
Reply from 132.20.0.1: Destination host unreachable.

Ping statistics for 132.18.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

```

- Test PC1

```

C:\>ping 132.19.0.2

Pinging 132.19.0.2 with 32 bytes of data:

Reply from 132.20.0.1: Destination host unreachable.
Reply from 132.20.0.1: Destination host unreachable.
Request timed out.
Reply from 132.20.0.1: Destination host unreachable.

Ping statistics for 132.19.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>

```

- Test PC3



```
C:\>ping 132.21.0.2

Pinging 132.21.0.2 with 32 bytes of data:

Reply from 132.21.0.2: bytes=32 time<1ms TTL=127
Reply from 132.21.0.2: bytes=32 time<1ms TTL=127
Reply from 132.21.0.2: bytes=32 time<1ms TTL=127
Reply from 132.21.0.2: bytes=32 time<1ms TTL=127

Ping statistics for 132.21.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- Test router1

```
C:\>ping 132.20.0.1

Pinging 132.20.0.1 with 32 bytes of data:

Reply from 132.20.0.1: bytes=32 time<1ms TTL=255
Reply from 132.20.0.1: bytes=32 time<1ms TTL=255
Reply from 132.20.0.1: bytes=32 time<1ms TTL=255
Reply from 132.20.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 132.20.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 8ms, Average = 2ms
```

- PC3
  - Test PC0

```
C:\>ping 132.18.0.2

Pinging 132.18.0.2 with 32 bytes of data:

Reply from 132.21.0.1: Destination host unreachable.
Reply from 132.21.0.1: Destination host unreachable.
Reply from 132.21.0.1: Destination host unreachable.
Reply from 132.21.0.1: Destination host unreachable.

Ping statistics for 132.18.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>|
```

- Test PC1

```
C:\>ping 132.19.0.2

Pinging 132.19.0.2 with 32 bytes of data:

Reply from 132.21.0.1: Destination host unreachable.
Reply from 132.21.0.1: Destination host unreachable.
Request timed out.
Reply from 132.21.0.1: Destination host unreachable.

Ping statistics for 132.19.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test PC2

```
C:\>ping 132.20.0.2

Pinging 132.20.0.2 with 32 bytes of data:

Reply from 132.20.0.2: bytes=32 time<1ms TTL=127
Reply from 132.20.0.2: bytes=32 time<1ms TTL=127
Reply from 132.20.0.2: bytes=32 time<1ms TTL=127
Reply from 132.20.0.2: bytes=32 time<1ms TTL=127

Ping statistics for 132.20.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- Test router1

```
C:\>ping 132.21.0.1

Pinging 132.21.0.1 with 32 bytes of data:

Reply from 132.21.0.1: bytes=32 time=6ms TTL=255
Reply from 132.21.0.1: bytes=32 time<1ms TTL=255
Reply from 132.21.0.1: bytes=32 time<1ms TTL=255
Reply from 132.21.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 132.21.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 6ms, Average = 1ms
```

- Test Esteban
  - PC0
    - Test PC1

```
C:\>ping 73.0.0.1

Pinging 73.0.0.1 with 32 bytes of data:

Reply from 73.0.0.1: bytes=32 time<1ms TTL=255
Reply from 73.0.0.1: bytes=32 time<1ms TTL=255
Reply from 73.0.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 73.0.0.1:
    Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- Test PC2

```
C:\>ping 74.0.0.2

Pinging 74.0.0.2 with 32 bytes of data:

Reply from 72.0.0.1: Destination host unreachable.
Reply from 72.0.0.1: Destination host unreachable.

Ping statistics for 74.0.0.2:
    Packets: Sent = 2, Received = 0, Lost = 2 (100% loss),
```

- Test PC3

```
C:\>ping 75.0.0.2

Pinging 75.0.0.2 with 32 bytes of data:

Reply from 72.0.0.1: Destination host unreachable.
Reply from 72.0.0.1: Destination host unreachable.
Request timed out.
Reply from 72.0.0.1: Destination host unreachable.

Ping statistics for 75.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test router0

```
C:\>ping 72.0.0.1

Pinging 72.0.0.1 with 32 bytes of data:

Reply from 72.0.0.1: bytes=32 time<1ms TTL=255
Reply from 72.0.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 72.0.0.1:
    Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- PC1

- Test PC0

```
C:\>ping 72.0.0.1

Pinging 72.0.0.1 with 32 bytes of data:

Reply from 72.0.0.1: bytes=32 time<1ms TTL=255
Reply from 72.0.0.1: bytes=32 time<1ms TTL=255
Reply from 72.0.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 72.0.0.1:
    Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- Test PC2

```
C:\>ping 74.0.0.2

Pinging 74.0.0.2 with 32 bytes of data:

Reply from 73.0.0.1: Destination host unreachable.
Reply from 73.0.0.1: Destination host unreachable.
Reply from 73.0.0.1: Destination host unreachable.
Reply from 73.0.0.1: Destination host unreachable.

Ping statistics for 74.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test PC3

```
C:\>ping 75.0.0.2

Pinging 75.0.0.2 with 32 bytes of data:

Reply from 73.0.0.1: Destination host unreachable.
Request timed out.
Reply from 73.0.0.1: Destination host unreachable.
Reply from 73.0.0.1: Destination host unreachable.

Ping statistics for 75.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test router0

```
C:\>ping 73.0.0.1

Pinging 73.0.0.1 with 32 bytes of data:

Reply from 73.0.0.1: bytes=32 time<1ms TTL=255
Reply from 73.0.0.1: bytes=32 time=9ms TTL=255
Reply from 73.0.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 73.0.0.1:
    Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 9ms, Average = 3ms
```

- PC2
  - Test PC0

```
C:\>ping 72.0.0.2

Pinging 72.0.0.2 with 32 bytes of data:

Reply from 74.0.0.1: Destination host unreachable.
Reply from 74.0.0.1: Destination host unreachable.
Reply from 74.0.0.1: Destination host unreachable.
Reply from 74.0.0.1: Destination host unreachable.

Ping statistics for 72.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test PC1

```
C:\>ping 73.0.0.2

Pinging 73.0.0.2 with 32 bytes of data:

Reply from 74.0.0.1: Destination host unreachable.
Reply from 74.0.0.1: Destination host unreachable.
Reply from 74.0.0.1: Destination host unreachable.
Reply from 74.0.0.1: Destination host unreachable.

Ping statistics for 73.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test PC3

```
C:\>ping 75.0.0.1
```

```
Pinging 75.0.0.1 with 32 bytes of data:
```

```
Reply from 75.0.0.1: bytes=32 time<1ms TTL=255  
Reply from 75.0.0.1: bytes=32 time<1ms TTL=255  
Reply from 75.0.0.1: bytes=32 time<1ms TTL=255  
Reply from 75.0.0.1: bytes=32 time<1ms TTL=255
```

```
Ping statistics for 75.0.0.1:
```

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

- Test router1

```
C:\>ping 74.0.0.1
```

```
Pinging 74.0.0.1 with 32 bytes of data:
```

```
Reply from 74.0.0.1: bytes=32 time<1ms TTL=255  
Reply from 74.0.0.1: bytes=32 time<1ms TTL=255  
Reply from 74.0.0.1: bytes=32 time<1ms TTL=255  
Reply from 74.0.0.1: bytes=32 time<1ms TTL=255
```

```
Ping statistics for 74.0.0.1:
```

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

- PC3

- Test PC0

```
C:\>ping 72.0.0.2
```

```
Pinging 72.0.0.2 with 32 bytes of data:
```

```
Reply from 75.0.0.1: Destination host unreachable.  
Reply from 75.0.0.1: Destination host unreachable.  
Reply from 75.0.0.1: Destination host unreachable.  
Reply from 75.0.0.1: Destination host unreachable.
```

```
Ping statistics for 72.0.0.2:
```

```
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Test PC1

```
C:\>ping 73.0.0.2

Pinging 73.0.0.2 with 32 bytes of data:

Reply from 75.0.0.1: Destination host unreachable.
Reply from 75.0.0.1: Destination host unreachable.
Reply from 75.0.0.1: Destination host unreachable.
Reply from 75.0.0.1: Destination host unreachable.

Ping statistics for 73.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>ping 72.0.0.2
```

○ Test PC2

```
C:\>ping 74.0.0.1

Pinging 74.0.0.1 with 32 bytes of data:

Reply from 74.0.0.1: bytes=32 time<1ms TTL=255
Reply from 74.0.0.1: bytes=32 time<1ms TTL=255
Reply from 74.0.0.1: bytes=32 time<1ms TTL=255
Reply from 74.0.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 74.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

○ Test router1

```
C:\>ping 75.0.0.1

Pinging 75.0.0.1 with 32 bytes of data:

Reply from 75.0.0.1: bytes=32 time<1ms TTL=255
Reply from 75.0.0.1: bytes=32 time=18ms TTL=255
Reply from 75.0.0.1: bytes=32 time<1ms TTL=255
Reply from 75.0.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 75.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 18ms, Average = 4ms
```

- No funciona cuando se hace ping desde una maquina del router 0 hacia una maquina del router 1 ya que los router no saben como llegar al destino al cual se el esta pidiendo llegar ( falta la definición de las rutas)
- Errores:
  - Request timed out: No hay respuesta del destino, puede ser por problemas de red, bloqueo por firewall o que no exista una ruta.

- Destination host unreachable: Un router no sabe cómo llegar al destino y lo informa, normalmente porque falta una ruta configurada.
- Configure the routers with static routes so that all devices on the network can ping each other. Note: On each router, include the routes (exit interface) needed to reach networks that are not directly connected
  - Entramos nuevamente a la configuración de los router y configuramos las rutas
    - Para el caso de Juan
      - Router 0

```
Juanito>enable
Password:
Juanito#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Juanito(config)#ip route 132.20.0.0 255.255.0.0 20.0.0.2
Juanito(config)#ip route 132.21.0.0 255.255.0.0 20.0.0.2
Juanito(config)#exit
Juanito#
%SYS-5-CONFIG_I: Configured from console by console

Juanito#write memory
Building configuration...
[OK]
Juanito#
```

- Router 1

```
User Access Verification

Password:

Juanito2>enable
Password:
Juanito2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Juanito2(config)#ip route 132.18.0.0 255.255.0.0 20.0.0.1
Juanito2(config)#ip route 132.19.0.0 255.255.0.0 20.0.0.1
Juanito2(config)#exit
Juanito2#
%SYS-5-CONFIG_I: Configured from console by console

Juanito2#write memory
Building configuration...
[OK]
Juanito2#
```

- Para el caso de Esteban
  - Router 0



```

User Access Verification

Password:

esteban>enable
Password:
esteban#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
esteban(config)#ip route 74.0.0.0 255.0.0.0 20.0.0.2
esteban(config)#ip route 75.0.0.0 255.0.0.0 20.0.0.2
esteban(config)#exit
esteban#
%SYS-5-CONFIG_I: Configured from console by console

esteban#write memory
Building configuration...
[OK]
esteban#

```

- Router 1

```

User Access Verification

Password:

Esteban2>enable
Password:
Esteban2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Esteban2(config)#ip route 72.0.0.0 255.0.0.0 20.0.0.1
Esteban2(config)#ip route 73.0.0.0 255.0.0.0 20.0.0.1
Esteban2(config)#exit
Esteban2#
%SYS-5-CONFIG_I: Configured from console by console

Esteban2#write memory
Building configuration...
[OK]
Esteban2#

```

- Using the ICMP traceroute command, try to verify communication between the two computers
  - Probamos nuevamente con el comando ping la conexión entre las dos maquinas
    - Para el caso de Juan
      - PC0 to PC2

```

C:\>tracert 132.20.0.2

Tracing route to 132.20.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    132.18.0.1
  2  14 ms   2 ms    2 ms    20.0.0.2
  3  0 ms    4 ms    1 ms    132.20.0.2

Trace complete.

C:\>

```

- PC2 to PC0

```
C:\>tracert 132.18.0.2

Tracing route to 132.18.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    132.20.0.1
  2  1 ms    11 ms   1 ms    20.0.0.1
  3  1 ms    1 ms    0 ms    132.18.0.2

Trace complete.

C:\>
```

- Para el caso de Esteban

- PC0 to PC2

```
C:\>tracert 74.0.0.2

Tracing route to 74.0.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    72.0.0.1
  2  3 ms    4 ms    1 ms    20.0.0.2
  3  0 ms    0 ms    0 ms    74.0.0.2

Trace complete.
```

- PC2 to PC0

```
C:\>tracert 72.0.0.2

Tracing route to 72.0.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    74.0.0.1
  2  1 ms    1 ms    3 ms    20.0.0.1
  3  1 ms    0 ms    0 ms    72.0.0.2

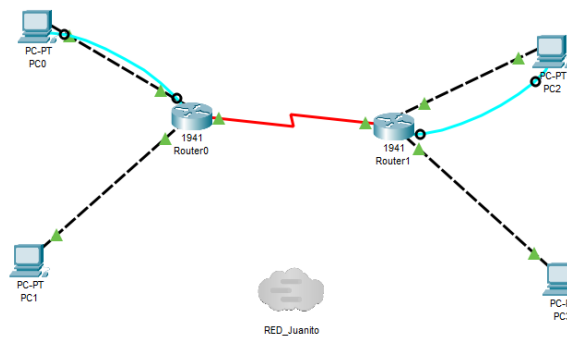
Trace complete.

C:\>
```

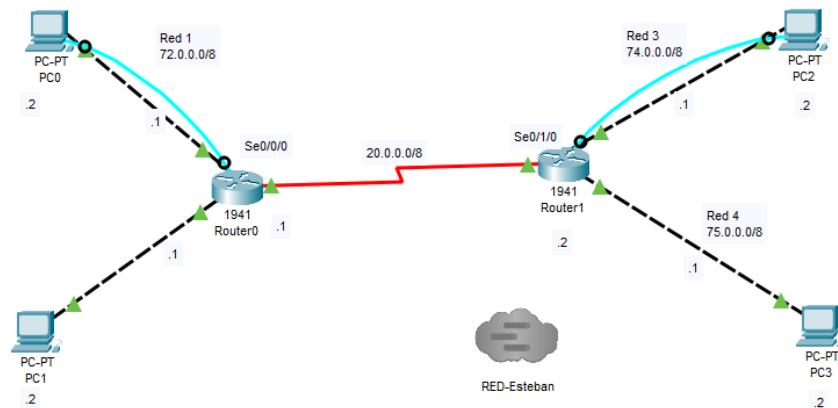
- Connect the setups using the multiuser option so that all computers on the network are visible, and verify the path that the packets follow. Connect the setups through Router1. In the case of three students, connect one setup like this: between student1 and student2 through Router0, and between student2 and student3 through Router1.

- Creamos la red con la opción multiuser

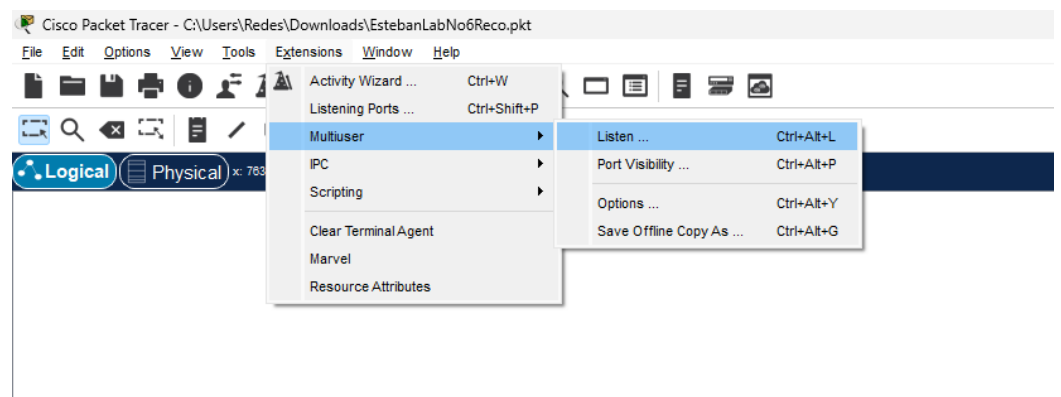
- Para el caso de Juan



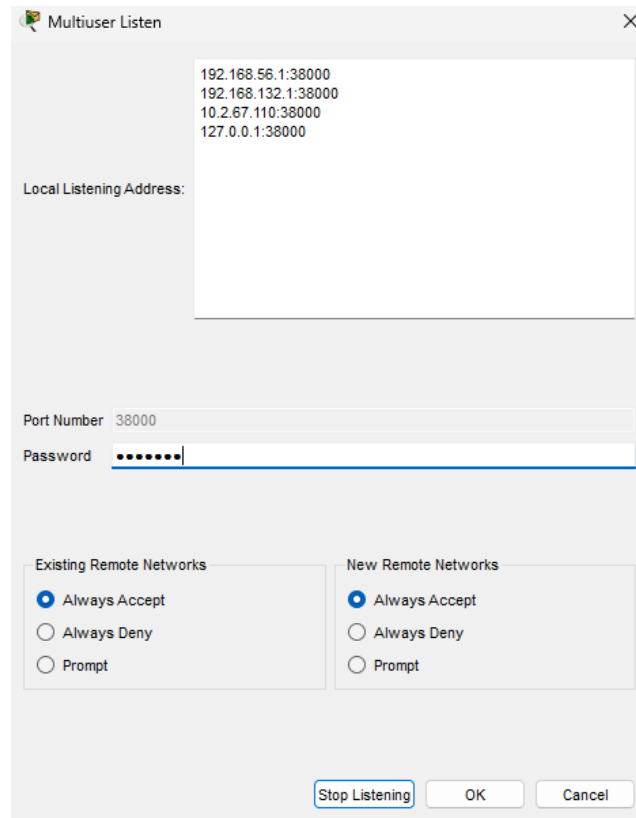
- Para el caso de Esteban



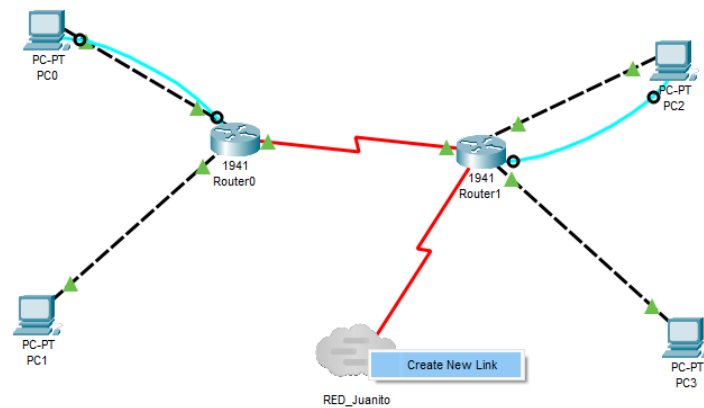
- Para el caso de Juan, configuramos las opciones del multiuser en el siguiente apartado



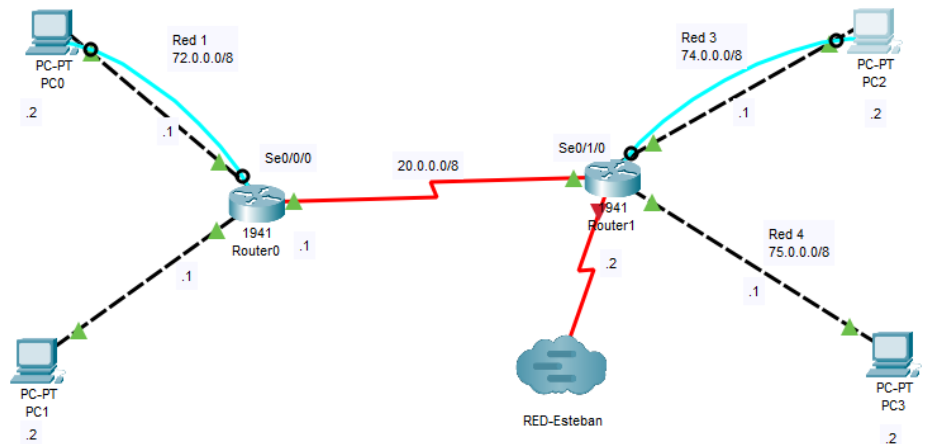
- Seleccionamos Always accept y ponemos la contraseña ( root123)



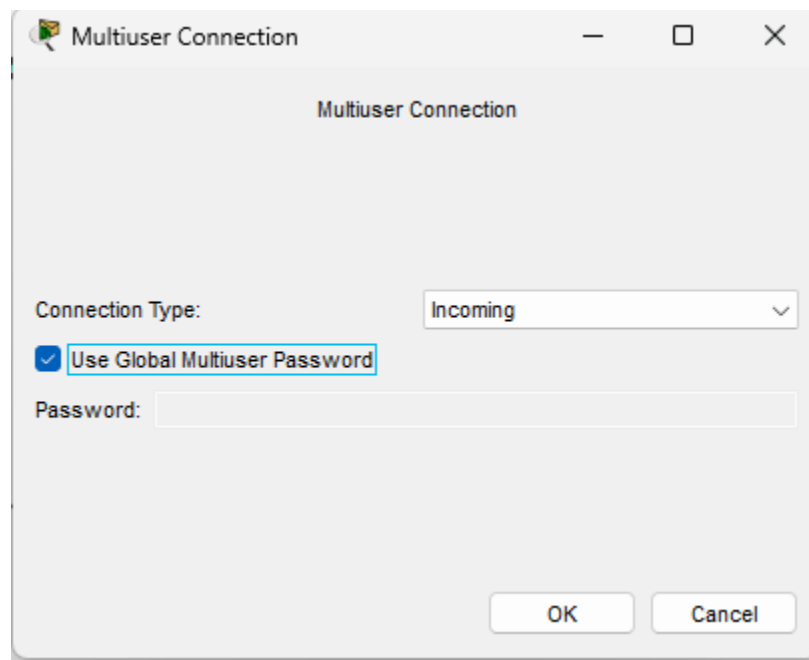
- Conectamos el router 1 con la Red (multiuser)
  - Para el caso de Juan



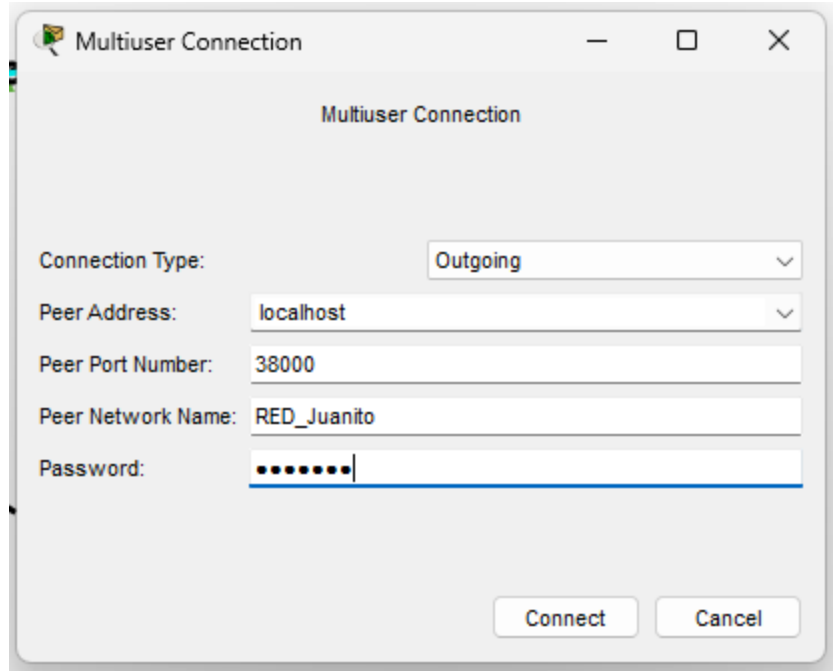
- Para el caso de Esteban



- Para el caso de Juan , seleccionamos la configuración de la RED ( multiuser ) y dejamos la opción inconming y aceptar
  - Para el caso de juan



- Para el caso de Esteban es Outgoing , colocamos el nombre de la red y la contraseña colocada actualmente en Juanito ( root123)



- Ahora configuramos la conexión de la red (multiuser) con el router 1
  - Para el caso de Juan

```
Juanito2(config)#interface serial0/1/1
Juanito2(config-if)#description "Configure peer module"
Juanito2(config-if)#ip address 30.0.0.4 255.0.0.0
Juanito2(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/1/1, changed state to down
```

- Para el caso de Esteban

```
Esteban2(config)#interface serial0/1/1
Esteban2(config-if)#description "Configure peer module"
Esteban2(config-if)#ip address 30.0.0.3 255.0.0.0
Esteban2(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/1/1, changed state to down
Esteban2(config-if)#
```

- Configuramos las rutas para permitir las conexiones entre las redes Esteban y Juanito y sus respectivas maquinas tanto para entrada y salida
  - Para el caso de Juan
    - Router 0

```

Juanito(config)#
Juanito(config)#ip route 132.18.0.0 255.255.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito(config)#ip route 132.19.0.0 255.255.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito(config)#ip route 132.20.0.0 255.255.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito(config)#ip route 132.21.0.0 255.255.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito(config)#ip route 20.0.0.0 255.0.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito(config)#ip route 132.18.0.0 255.0.0.0 20.0.0.2
%Inconsistent address and mask
Juanito(config)#ip route 132.18.0.0 255.255.0.0 20.0.0.2
Juanito(config)#ip route 132.19.0.0 255.255.0.0 20.0.0.2
Juanito(config)#ip route 132.20.0.0 255.255.0.0 20.0.0.2
Juanito(config)#ip route 132.21.0.0 255.255.0.0 20.0.0.2
Juanito(config)#ip route 72.0.0.0 255.0.0.0 20.0.0.2
Juanito(config)#ip route 73.0.0.0 255.0.0.0 20.0.0.2
Juanito(config)#ip route 74.0.0.0 255.0.0.0 20.0.0.2
Juanito(config)#ip route 75.0.0.0 255.0.0.0 20.0.0.2
Juanito(config)#

```

- Router 1

```

Juanito2(config)#
Juanito2(config)#
Juanito2(config)#no ip route 132.18.0.0 255.255.0.0 20.0.0.1
Juanito2(config)#no ip route 132.19.0.0 255.255.0.0 20.0.0.1
Juanito2(config)#ip route 132.18.0.0 255.255.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito2(config)#ip route 132.19% Bad secrets
.0.0 255.255.0.0 Serial0/1/0Juanito2(config)#
Juanito2(config)#
Juanito2(config)#ip route 132.19.0.0 255.255.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito2(config)#ip route 132.20.0.0 255.255.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito2(config)#ip route 132.21.0.0 255.255.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito2(config)#ip route 20.0.0.0 255.0.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may impact performance
Juanito2(config)#ip route 72.0.0.0 255.0.0.0 30.0.0.3
Juanito2(config)#ip route 73.0.0.0 255.0.0.0 30.0.0.3
Juanito2(config)#ip route 74.0.0.0 255.0.0.0 30.0.0.3
Juanito2(config)#ip route 75.0.0.0 255.0.0.0 30.0.0.3
Juanito2(config)#

```

- Para el caso de Esteban
  - Router 0

```

esteban>esteban>enable
Password:
esteban#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
esteban(config)#ip route 75.0.0.0 255.0.0.0 20.0.0.2
esteban(config)#ip route 74.0.0.0 255.0.0.0 20.0.0.2
esteban(config)#ip route 73.0.0.0 255.0.0.0 20.0.0.2
esteban(config)#ip route 72.0.0.0 255.0.0.0 20.0.0.2
esteban(config)#exit
esteban#
%SYS-5-CONFIG_I: Configured from console by console

esteban#tracert 132.18.0.2
^
% Invalid input detected at '^' marker.

esteban#enable
esteban#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
esteban(config)#ip route 132.18.0.0 255.255.0.0 20.0.0.2
esteban(config)#ip route 132.19.0.0 255.255.0.0 20.0.0.2
esteban(config)#ip route 132.20.0.0 255.255.0.0 20.0.0.2
esteban(config)#ip route 132.21.0.0 255.255.0.0 20.0.0.2
esteban(config)#exit
esteban#
%SYS-5-CONFIG_I: Configured from console by console

```

- Router 1

```

Esteban2(config)#
Esteban2(config)#
Esteban2(config)#ip route 72.0.0.0 255.0.0.0 20.0.0.1
Esteban2(config)#ip route 73.0.0.0 255.0.0.0 20.0.0.1
Esteban2(config)#ip route 74.0.0.0 255.0.0.0 20.0.0.1
Esteban2(config)#ip route 75.0.0.0 255.0.0.0 20.0.0.1
Esteban2(config)#ip route 132.18.0.0 255.255.0.0 30.0.0.4
Esteban2(config)#ip route 132.19.0.0 255.255.0.0 30.0.0.4
Esteban2(config)#ip route 132.20.0.0 255.255.0.0 30.0.0.4
Esteban2(config)#ip route 132.21.0.0 255.255.0.0 30.0.0.4
Esteban2(config)#exit
Esteban2#
%SYS-5-CONFIG_I: Configured from console by console

Esteban2#

```

- Pruebas
  - PC0 Esteban to PC0 Juan



```
C:\>tracert 132.18.0.2

Tracing route to 132.18.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    72.0.0.1
  2  7 ms    0 ms    8 ms    20.0.0.2
  3  43 ms   28 ms   49 ms   30.0.0.4
  4  *        *        *        Request timed out.
  5  80 ms   78 ms   67 ms   132.18.0.2

Trace complete.
```

- PC0 Juan to PC0 Esteban

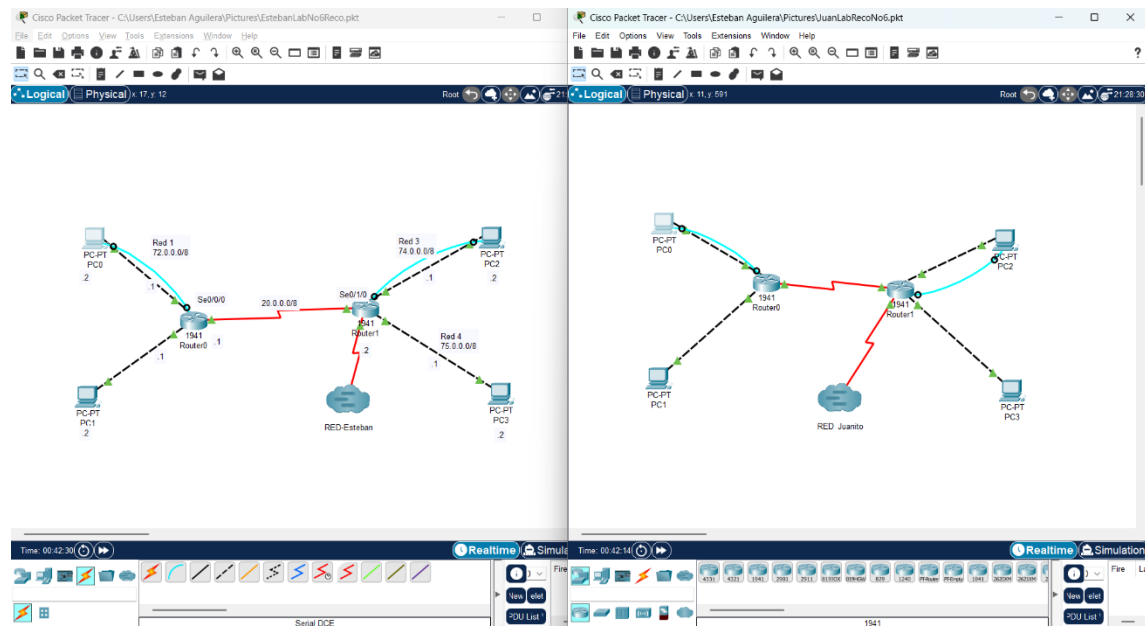
```
C:\>tracert 72.0.0.2

Tracing route to 72.0.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    132.18.0.1
  2  13 ms   28 ms   3 ms    20.0.0.2
  3  50 ms   46 ms   57 ms   30.0.0.3
  4  *        *        *        Request timed out.
  5  43 ms   79 ms   68 ms   72.0.0.2

Trace complete.
```

- Redes finales:



## BASE SOFTWARE INSTALLATION

### 1. Web Service Installation

- a. Install the Apache web server on the virtual machine with the NetBSD operating system.
- I. Como no necesariamente necesitamos trabajar desde la universidad, sino que también podemos trabajar desde la casa, lo único que hacemos es activar el dhcp para poder tener internet e instalar apache, accedemos al siguiente achivo “/etc/ifconfig.wm0”y hay activamos el dhcp, cabe aclarar que toca comentariar o eliminar la ip que ya teníamos configurada



```
GNU nano 8.3 /etc/ifconfig.wm0
up
media autoselect
#10.2.78.35 netmask 255.255.0.0 media autoselect
dhcp
```

The screenshot shows a terminal window with the nano text editor open to the file /etc/ifconfig.wm0. The file contains four lines: 'up', 'media autoselect', '#10.2.78.35 netmask 255.255.0.0 media autoselect', and 'dhcp'. The nano editor's status bar at the bottom shows 'Read 4 lines' and various keyboard shortcuts like ^G Help, ^X Exit, ^O Write Out, etc.

- II. Primero verificamos si tenemos instalado Apache ya en nuestras maquinas, con “httpd -v” o “which httpd” en este caso nos debería de arrojar una ruta si, si lo tenemos instalado, pero en este caso no está instalado



```
Juanito# sh
Juanito# which httpd
Juanito# httpd -v
sh: httpd: not found
Juanito#
```

The screenshot shows a terminal window with the prompt 'Juanito#'. The user enters 'sh' and then 'which httpd', which returns no output. Then the user enters 'httpd -v', which returns 'sh: httpd: not found'.

- III. Instalamos Apache con el comando “pkgin install apache” y se empieza a instalar

```

Juanito# pkgin install apache
calculating dependencies...done.

19 packages to refresh:
cmocka-1.1.7 dbus-1.14.10 gettext-lib-0.22.5 gmp-6.3.0 jansson-2.14
libdaemon-0.14nb2 libgcrypt-1.11.0nb2 libiconv-1.17 libunistring-1.2
lmdb-0.9.33 mozilla-rootcerts-1.1.20241118 p11-kit-0.25.5 p5-JSON-4.10nb2
p5-Parse-Yapp-1.21nb7 pkgin-24.12.0 popt-1.19 py312-dns-2.7.0
py312-iso8601-2.1.0nb1 readline-8.2nb2

18 packages to upgrade:
avahi-0.8nb7 gdbm-1.24 glib2-2.82.5 gnutls-3.8.9 libffi-3.4.7 libtasn1-4.20.0
libuuid-2.40.2 nettle-3.10.1 pcre2-10.45 perl-5.40.1 py312-idna-3.10nb1
py312-markdown-3.7nb1 python312-3.12.9 samba-4.21.3 sqlite3-3.48.0
talloc-2.4.3 tdb-1.4.13 tevent-0.16.2

7 packages to install:
apache-2.4.63 apr-1.7.5 apr-util-1.6.3nb7 brotli-1.1.0 libxml2-2.12.10
nghttp2-1.65.0 xmlcatmgr-2.2nb1

0 to remove, 19 to refresh, 18 to upgrade, 7 to install
72M to download, 40M of additional disk space will be used

proceed ? [Y/n]

```

```

/etc/rc.d/winbindd (m=0755)
/etc/rc.d/smbd (m=0755)
/etc/rc.d/samba (m=0755)
/etc/rc.d/nmbd (m=0755)

=====
$NetBSD: MESSAGE.rcd,v 1.1 2015/05/26 15:17:56 jperkin Exp $
=====

For ActiveDirectory/Domain Controller, use:

    /etc/rc.d/samba start

For traditional simple CIFS file sharing, use:

    /etc/rc.d/nmbd start
    /etc/rc.d/winbindd start
    /etc/rc.d/smbd start
=====

```

- IV. Ahora copiamos el archivo apache que esta en esta ruta “/usr/pkg/share/examples/rc.d/” y lo copiamos a “/etc/rc.d/” y el comando para copiar es el siguiente “cp /usr/pkg/share/examples/rc.d/apache /etc/rc.d/” y ya una vez copiado podemos iniciar apache con este comando “/etc/rc.d/apache start”

```

Juanito# cp /usr/pkg/share/examples/rc.d/apache /etc/rc.d/
Juanito# ls /usr/pkg/share/examples/rc.d/
apache      dbus        samba       winbindd
avahi-daemon nmbd        smbd
Juanito# /etc/rc.d/apache start
Starting apache.
Juanito#

```

- V. Si queremos ver si esta corriendo digitamos el comando “/etc/rc.d/apache status”

```

Juanito# /etc/rc.d/apache status
apache is running as pid 1467.
Juanito#

```

- VI. Accedemos al archivo que guarda la pagina inicial ( index) que esta ubicada en “/usr/pkg/share/httpd/htdocs” y aparece como index.html

```
Juanito# ls /usr/pkg/share/httpd/htdocs
index.html
Juanito# nano /usr/pkg/share/httpd/htdocs/index.html
```

Si queremos saber de donde obtuvimos esa ruta, es por el documento httpd.conf

```
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/usr/pkg/share/httpd/htdocs"
<Directory "/usr/pkg/share/httpd/htdocs">
    #
    # Possible values for the Options directive are "None", "All",
    # or any combination of:
    #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
    #
    #G Help      ^O Write Out  ^F Where Is   ^K Cut        ^T Execute    ^C Location
    ^X Exit      ^R Read File  ^\ Replace   ^U Paste      ^J Justify    ^_ Go To Line
```

- VII. Modificamos la pagina inicial a nuestro gusto (ya dentro del index.html).

```
GNU nano 8.3 /usr/pkg/share/httpd/htdocs/index.html
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to Juanito Page</title>
  <style>
    body {
      background-color: #f0fff0;
      text-align: center;
      padding: 50px;
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      color: #006400;
    }
    h1 {
      color: #228B22;
      font-size: 3em;
    }
    p {
      font-size: 1.5em;
    }
  </style>
</head>

Juanito#
```

- VIII. Para ver la pagina que este funcionando accedemos a el mediante un navegador colocando la ip configurada en NetBSD ( en este caso es 192.168.1.118)

- b. Install the Nginx web server on the virtual machine with the Slackware Linux operating system

- i. Descargamos el archivo .tar.gz de Nginx desde Slackbuilds mediante el comando : wget

<https://slackbuilds.org/slackbuilds/15.0/network/nginx.tar.gz>

```
root@juanito:~# wget https://slackbuilds.org/slackbuilds/15.0/network/nginx.tar.gz
--2025-04-08 19:05:36-- https://slackbuilds.org/slackbuilds/15.0/network/nginx.tar.gz
Resolving slackbuilds.org (slackbuilds.org)... 66.85.79.67, 2604:5800:0:90::67
Connecting to slackbuilds.org (slackbuilds.org)|66.85.79.67|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5714 (5.6K) [application/x-gzip]
Saving to: ■nginx.tar.gz■

nginx.tar.gz          100%[=====>]    5.58K  --.-KB/s  in
2025-04-08 19:05:37 (536 MB/s) - ■nginx.tar.gz■ saved [5714/5714]
```

- ii. Descomprimos el archivo y navegamos en el directorio de nginx mediante el comando : tar -xzf nginx.tar.gz

```
root@juanito:~# tar -xzf nginx.tar.gz
nginx/
nginx/doinst.sh
nginx/nginx.info
nginx/nginx.logrotate
nginx/slack-desc
nginx/rc.nginx
nginx/README
nginx/nginx.SlackBuild
```

- iii. Descargamos la version de nginx requerida por slackbuilds (1.26.2) mediante el comando : wget <http://nginx.org/download/nginx-1.26.2.tar.gz>

```
root@juanito:~# wget https://nginx.org/download/nginx-1.26.2.tar.gz
--2025-04-08 19:08:22-- https://nginx.org/download/nginx-1.26.2.tar.gz
Resolving nginx.org (nginx.org)... 3.125.197.172, 52.58.199.22, 2a05:d014:5c0:2601::6, .
Connecting to nginx.org (nginx.org)|3.125.197.172|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1244789 (1.2M) [application/octet-stream]
Saving to: ■nginx-1.26.2.tar.gz■

nginx-1.26.2.tar.gz    100%[=====>]    1.19M  1.28MB/s  in
2025-04-08 19:08:24 (1.28 MB/s) - ■nginx-1.26.2.tar.gz■ saved [1244789/1244789]

root@juanito:~#
```

- iv. Realizamos la descompresión del archivo descargado mediante el comando : tar -xzf nginx-1.26.2.tar.gz

```

nginx-1.26.2/auto/lib/libxslt/
nginx-1.26.2/auto/lib/perl/
nginx-1.26.2/auto/lib/geoip/
nginx-1.26.2/auto/lib/zlib/
nginx-1.26.2/auto/lib/google-perftools/
nginx-1.26.2/auto/lib/make
nginx-1.26.2/auto/lib/pcre/
nginx-1.26.2/auto/lib/openssl/
nginx-1.26.2/auto/lib/conf
nginx-1.26.2/auto/lib/openssl/makefile.bcc
nginx-1.26.2/auto/lib/openssl/makefile.msuc
nginx-1.26.2/auto/lib/openssl/make
nginx-1.26.2/auto/lib/openssl/conf
nginx-1.26.2/auto/lib/pcre/makefile.bcc
nginx-1.26.2/auto/lib/pcre/makefile.msuc
nginx-1.26.2/auto/lib/pcre/make
nginx-1.26.2/auto/lib/pcre/conf
nginx-1.26.2/auto/lib/pcre/makefile.owc
nginx-1.26.2/auto/lib/google-perftools/conf
nginx-1.26.2/auto/lib/zlib/makefile.bcc
nginx-1.26.2/auto/lib/zlib/makefile.msuc
nginx-1.26.2/auto/lib/zlib/make
nginx-1.26.2/auto/lib/zlib/conf
nginx-1.26.2/auto/lib/zlib/makefile.owc
nginx-1.26.2/auto/lib/geoip/conf
nginx-1.26.2/auto/lib/perl/make
nginx-1.26.2/auto/lib/perl/conf
nginx-1.26.2/auto/lib/libxslt/conf
nginx-1.26.2/auto/lib/libatomic/make
nginx-1.26.2/auto/lib/libatomic/conf
nginx-1.26.2/auto/lib/libgd/conf
nginx-1.26.2/auto/types/typedef
nginx-1.26.2/auto/types/value
nginx-1.26.2/auto/types/uinptr_t
nginx-1.26.2/auto/types/sizeof
nginx-1.26.2/man/nginx.8

```

- v. Ingresamos al directorio donde quedo descargado nginx mediante el comando : `cd nginx-1.26.2`

```

root@juanito:~# cd nginx-1.26.2

```

- vi. Configuramos el paquete de nginx con el comando : `./configure`

```

root@juanito:~/nginx-1.26.2# ./configure_

```

```
Configuration summary
+ using system PCRE2 library
+ OpenSSL library is not used
+ using system zlib library

nginx path prefix: "/usr/local/nginx"
nginx binary file: "/usr/local/nginx/sbin/nginx"
nginx modules path: "/usr/local/nginx/modules"
nginx configuration prefix: "/usr/local/nginx/conf"
nginx configuration file: "/usr/local/nginx/conf/nginx.conf"
nginx pid file: "/usr/local/nginx/logs/nginx.pid"
nginx error log file: "/usr/local/nginx/logs/error.log"
nginx http access log file: "/usr/local/nginx/logs/access.log"
nginx http client request body temporary files: "client_body_temp"
nginx http proxy temporary files: "proxy_temp"
nginx http fastcgi temporary files: "fastcgi_temp"
nginx http uwsgi temporary files: "uwsgi_temp"
nginx http scgi temporary files: "scgi_temp"

root@juanito:~/nginx-1.26.2# _
```

- vii. Usamos el comando “make” para compilar el paquete configurado de nginx

```
root@juanito:~/nginx-1.26.2# make
```

```

objs/src/http/modules/nginx_http_static_module.o \
objs/src/http/modules/nginx_http_autoindex_module.o \
objs/src/http/modules/nginx_http_index_module.o \
objs/src/http/modules/nginx_http_mirror_module.o \
objs/src/http/modules/nginx_http_try_files_module.o \
objs/src/http/modules/nginx_http_auth_basic_module.o \
objs/src/http/modules/nginx_http_access_module.o \
objs/src/http/modules/nginx_http_limit_conn_module.o \
objs/src/http/modules/nginx_http_limit_req_module.o \
objs/src/http/modules/nginx_http_geo_module.o \
objs/src/http/modules/nginx_http_map_module.o \
objs/src/http/modules/nginx_http_split_clients_module.o \
objs/src/http/modules/nginx_http_referer_module.o \
objs/src/http/modules/nginx_http_rewrite_module.o \
objs/src/http/modules/nginx_http_proxy_module.o \
objs/src/http/modules/nginx_http_fastcgi_module.o \
objs/src/http/modules/nginx_http_uwsgi_module.o \
objs/src/http/modules/nginx_http_scgi_module.o \
objs/src/http/modules/nginx_http_memcached_module.o \
objs/src/http/modules/nginx_http_empty_gif_module.o \
objs/src/http/modules/nginx_http_browser_module.o \
objs/src/http/modules/nginx_http_upstream_hash_module.o \
objs/src/http/modules/nginx_http_upstream_ip_hash_module.o \
objs/src/http/modules/nginx_http_upstream_least_conn_module.o \
objs/src/http/modules/nginx_http_upstream_random_module.o \
objs/src/http/modules/nginx_http_upstream_keepalive_module.o \
objs/src/http/modules/nginx_http_upstream_zone_module.o \
objs/nginx_modules.o \
-lld -lpthread -lcrypt -lpcres2-8 -lz \
-Wl,-E
sed -e "s|%%PREFIX%%|/usr/local/nginx|" \
    -e "s|%%PID_PATH%%|/usr/local/nginx/logs/nginx.pid|" \
    -e "s|%%CONF_PATH%%|/usr/local/nginx/conf/nginx.conf|" \
    -e "s|%%ERROR_LOG_PATH%%|/usr/local/nginx/logs/error.log|" \
    < man/nginx.8 > objs/nginx.8
make[1]: Leaving directory '/root/nginx-1.26.2'
root@juanito:~/nginx-1.26.2#

```

viii. Usamos “make install” para completar la instalación



```

cp objs/nginx '/usr/local/nginx/sbin/nginx'
test -d '/usr/local/nginx/conf' \
    || mkdir -p '/usr/local/nginx/conf'
cp conf/koi-win '/usr/local/nginx/conf'
cp conf/koi-utf '/usr/local/nginx/conf'
cp conf/win-utf '/usr/local/nginx/conf'
test -f '/usr/local/nginx/conf/mime.types' \
    || cp conf/mime.types '/usr/local/nginx/conf'
cp conf/mime.types '/usr/local/nginx/conf/mime.types.default'
test -f '/usr/local/nginx/conf/fastcgi_params' \
    || cp conf/fastcgi_params '/usr/local/nginx/conf'
cp conf/fastcgi_params \
    '/usr/local/nginx/conf/fastcgi_params.default'
test -f '/usr/local/nginx/conf/fastcgi.conf' \
    || cp conf/fastcgi.conf '/usr/local/nginx/conf'
cp conf/fastcgi.conf '/usr/local/nginx/conf/fastcgi.conf.default'
test -f '/usr/local/nginx/conf/uwsgi_params' \
    || cp conf/uwsgi_params '/usr/local/nginx/conf'
cp conf/uwsgi_params \
    '/usr/local/nginx/conf/uwsgi_params.default'
test -f '/usr/local/nginx/conf/scgi_params' \
    || cp conf/scgi_params '/usr/local/nginx/conf'
cp conf/scgi_params \
    '/usr/local/nginx/conf/scgi_params.default'
test -f '/usr/local/nginx/conf/nginx.conf' \
    || cp conf/nginx.conf '/usr/local/nginx/conf/nginx.conf'
cp conf/nginx.conf '/usr/local/nginx/conf/nginx.conf.default'
test -d '/usr/local/nginx/logs' \
    || mkdir -p '/usr/local/nginx/logs'
test -d '/usr/local/nginx/logs' \
    || mkdir -p '/usr/local/nginx/logs'
test -d '/usr/local/nginx/html' \
    || cp -R html '/usr/local/nginx'
test -d '/usr/local/nginx/logs' \
    || mkdir -p '/usr/local/nginx/logs'
make[1]: Leaving directory '/root/nginx-1.26.2'
root@juanito:~/nginx-1.26.2# _

```

- ix. Ejecutamos la ruta “/usr/local/nginx/sbin/nginx” para verificar que nginx este corriendo

```
root@juanito:~/nginx-1.26.2# /usr/local/nginx/sbin/nginx
```

- x. Accedemos al archivo que guarda la pagina inicial ( index) mediante el comando “ nano /usr/local/nginx/html/index.html ”

```
root@juanito:/# nano /usr/local/nginx/html/index.html
```

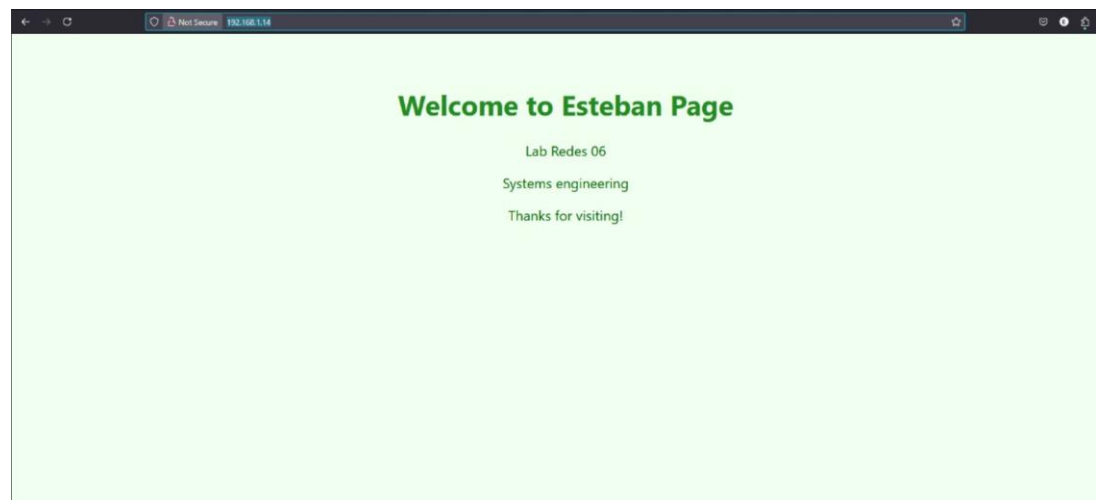
- xi. Modificamos la pagina inicial a nuestro gusto. En este caso colocamos que muestre:

1. Welcome to Esteban Page
  2. Lab redes 06
  3. Systems engineering
  4. Thanks for visiting!
  5. ( Todo de color verde)
- xii. Para mirar que el servidor quedo configurado correctamente accedemos a el mediante un navegador colocando la ip configurada en slackware ( en este caso es 192.168.1.14 ya que se trabajo con DHCP desde un dispositivo fuera de la universidad)

```
root@juanito:/# ipconfig
-bash: ipconfig: command not found
root@juanito:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.14 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::20c:29ff:fef7:561c prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:f7:56:1c txqueuelen 1000 (Ethernet)
    RX packets 7095 bytes 1693495 (1.6 MiB)
    RX errors 0 dropped 2 overruns 0 frame 0
    TX packets 894 bytes 60731 (59.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

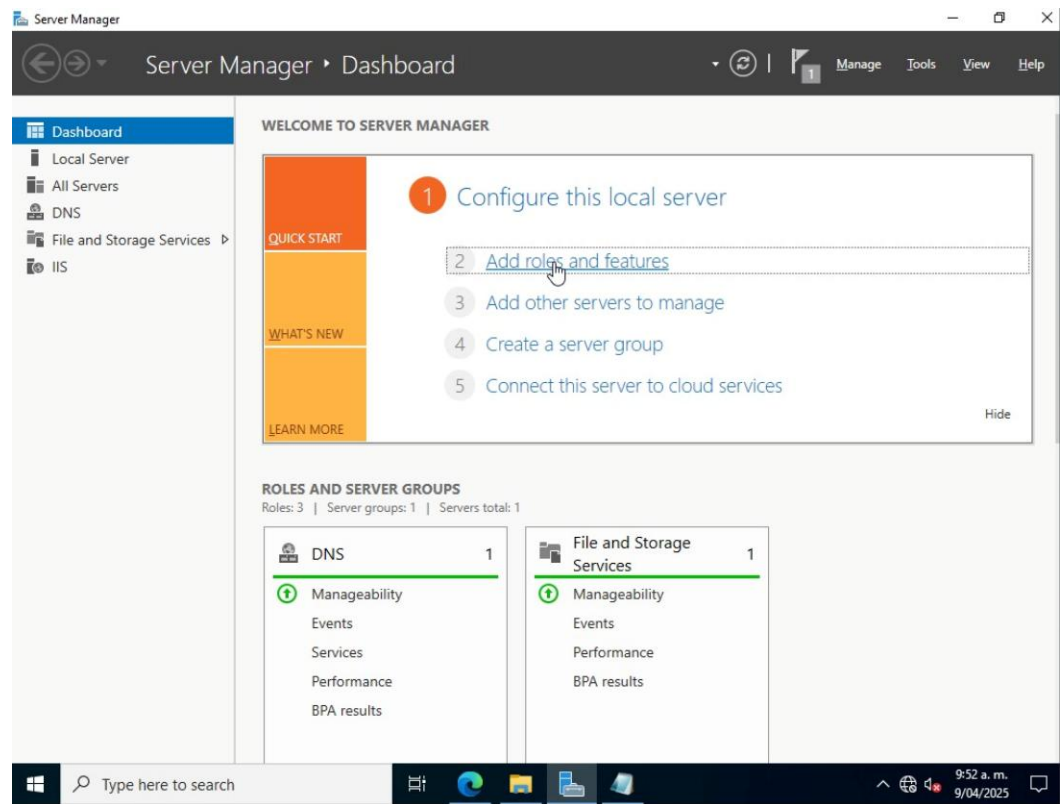
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 91 bytes 8306 (8.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 91 bytes 8306 (8.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@juanito:/#
```

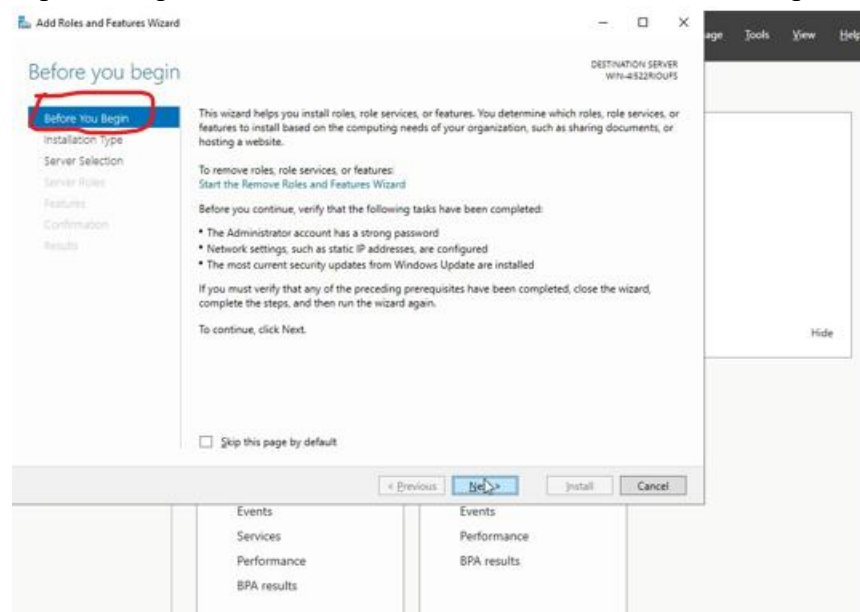


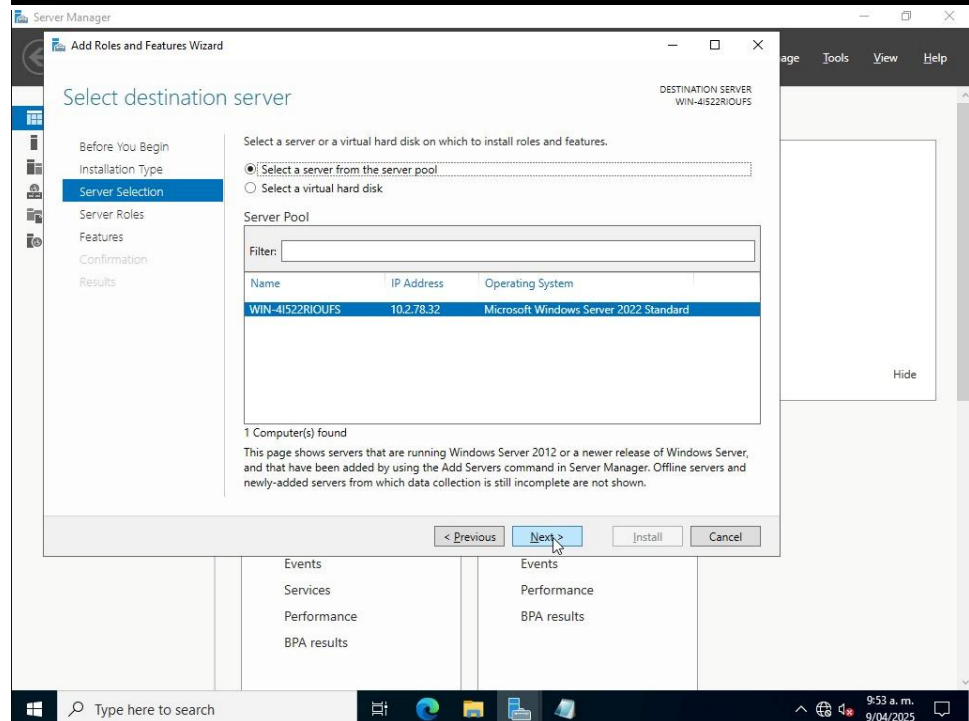
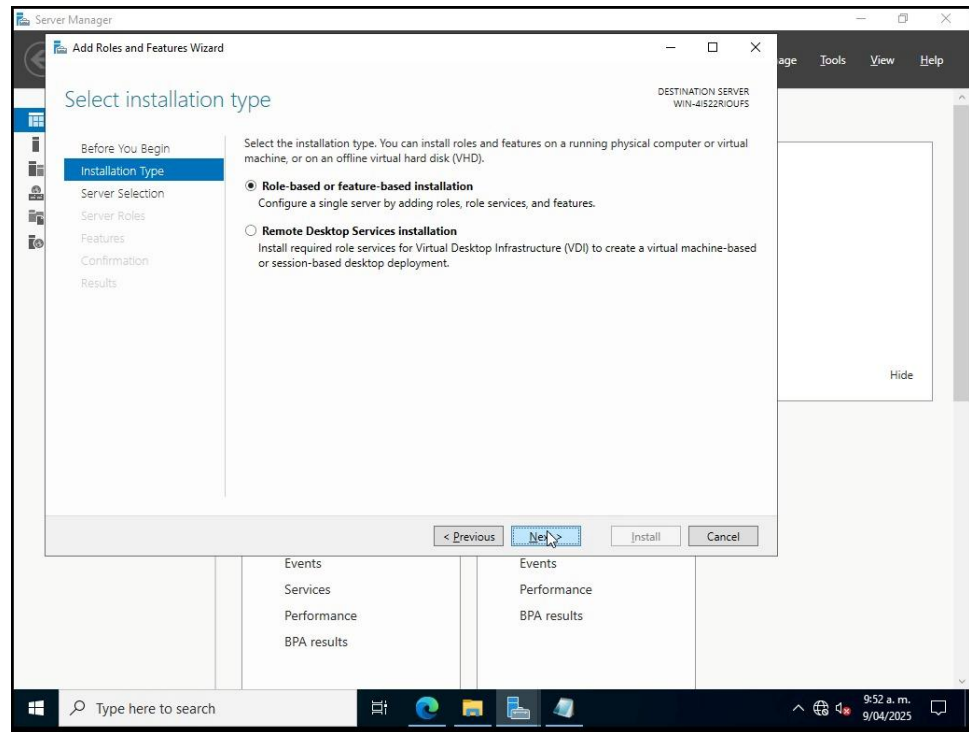
- c. Configure the web server that comes with Windows Server

- i. Accedemos a la aplicación Server Manager en el apartado “Add Roles and Features”

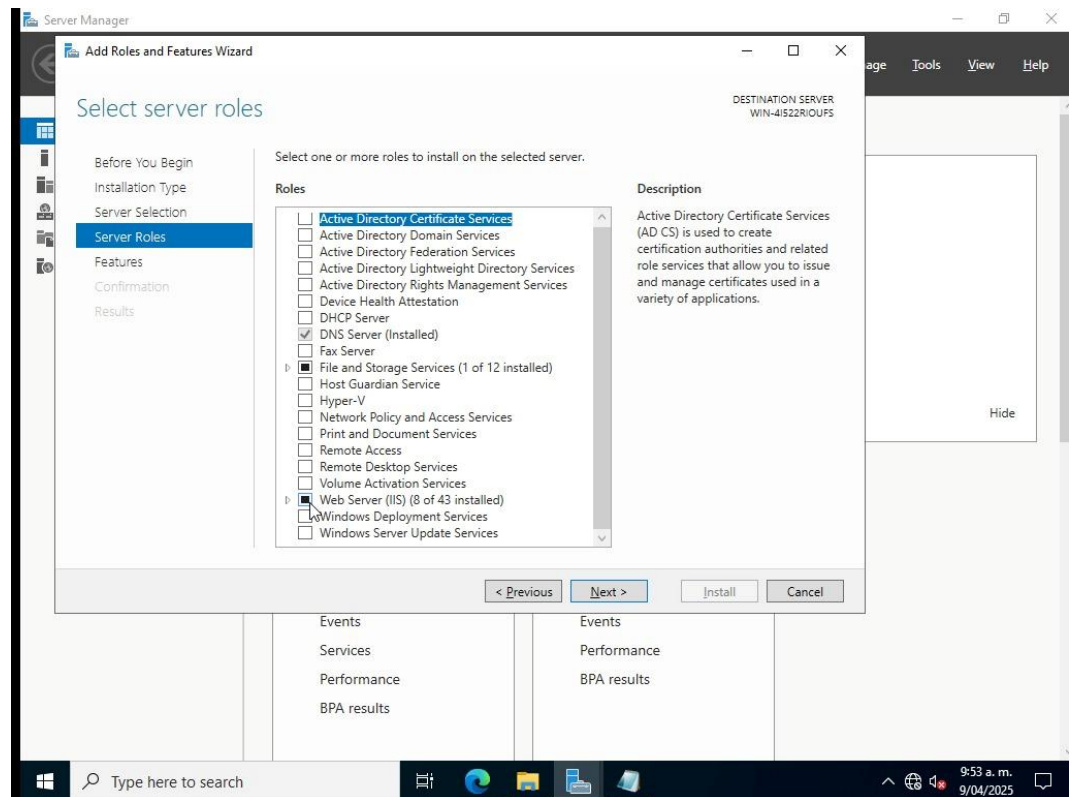


- ii. Damos click en el apartado “Next” 3 veces hasta llegar al apartado de “Server Roles”. Podemos ver el apartado en el cual estamos en la parte superior izquierda con un resaltado azul como se muestra en pantalla

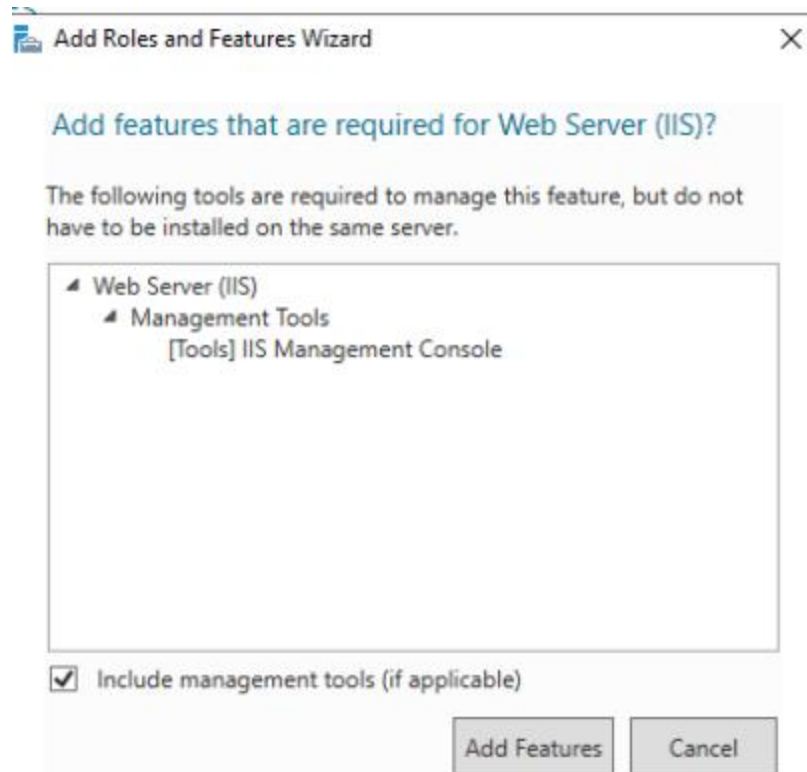




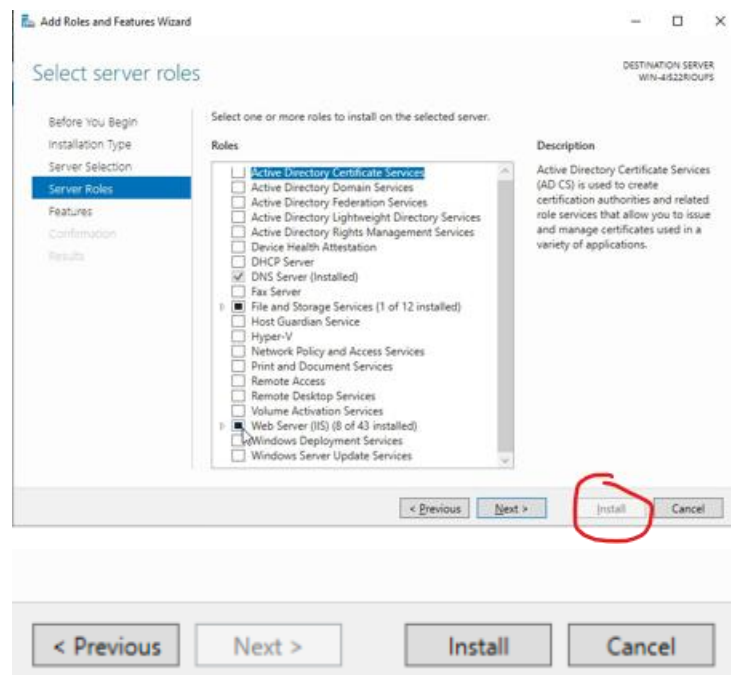
- iii. Al llegar al apartado Server Roles . Seleccionamos el apartado que tiene como nombre “Web Server (IIS)”



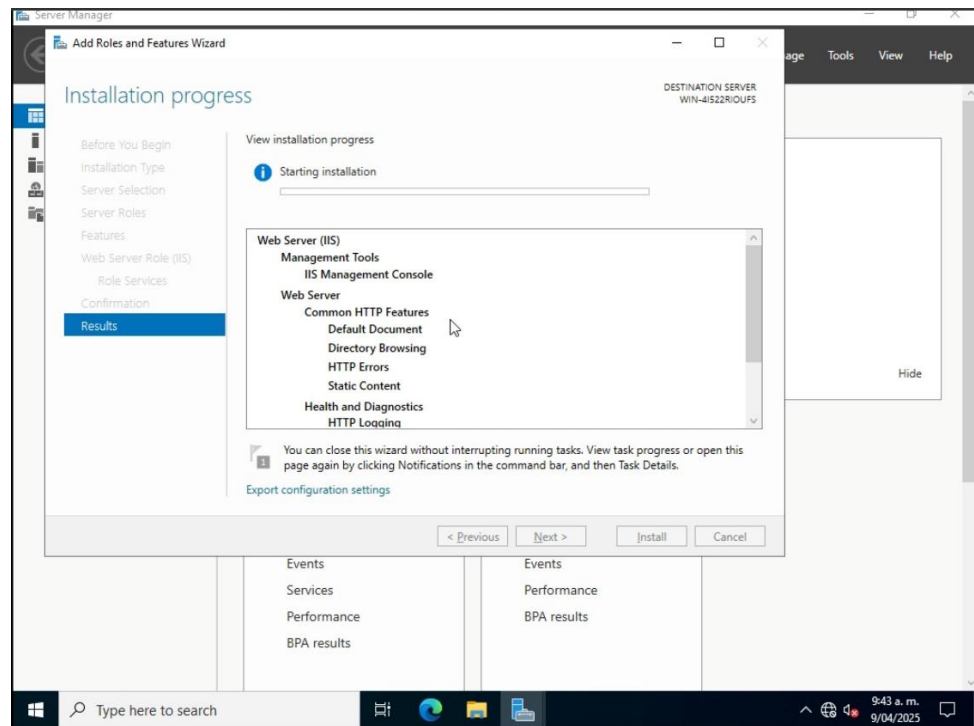
- iv. En el momento que seleccionamos el apartado de “Web Server” se nos desplegara una ventana en la cual seleccionamos la opción “ Add Features”



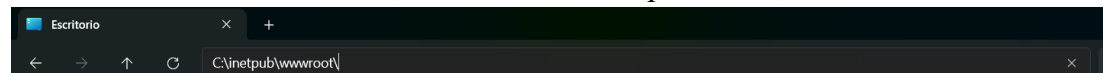
- v. Posteriormente se nos habilitara la opción install la cual anteriormente estaba deshabilitada. Damos click en ella



- vi. Ahora esperamos que se instale el Web Server y cerramos todas las ventanas



- vii. Una vez instalado el web server, abrimos el explorador de archivos y buscamos la ruta de instalación la cual es “C:\inetpub\wwwroot\”



- viii. Nos aparecerán 2 archivos los cuales tendrán los siguientes nombres.



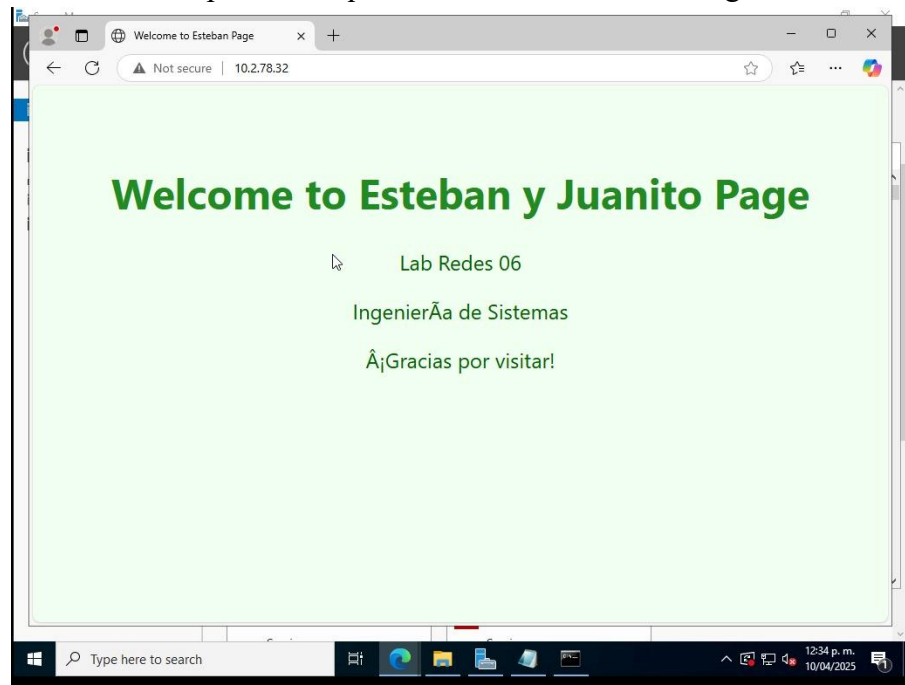
- ix. Abrimos el primer archivo en un bloc de notas y modificamos el código a nuestro gusto para la pagina web. En nuestro caso colocamos la misma aplicación de netBSD y Slackware pero con un título diferente

```

<!DOCTYPE html>
<html>
<head>
  <title>Welcome to Esteban Page</title>
  <style>
    body {
      background-color: #ffff00;
      text-align: center;
      padding: 50px;
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      color: #006400;
    }
    h1 {
      color: #228B22;
      font-size: 3em;
    }
    p {
      font-size: 1.5em;
    }
  </style>
</head>
<body>
  <h1>Welcome to Esteban and Juanito Page</h1>
  <p>Lab Redes 06</p>
  <p>Ingeniería de Sistemas</p>
  <p>¡Gracias por visitar!</p>
</body>
</html>

```

- x. Guardamos el archivo y probamos que quedo bien configurado escribiendo la ip de la maquina de Windows en un navegador web

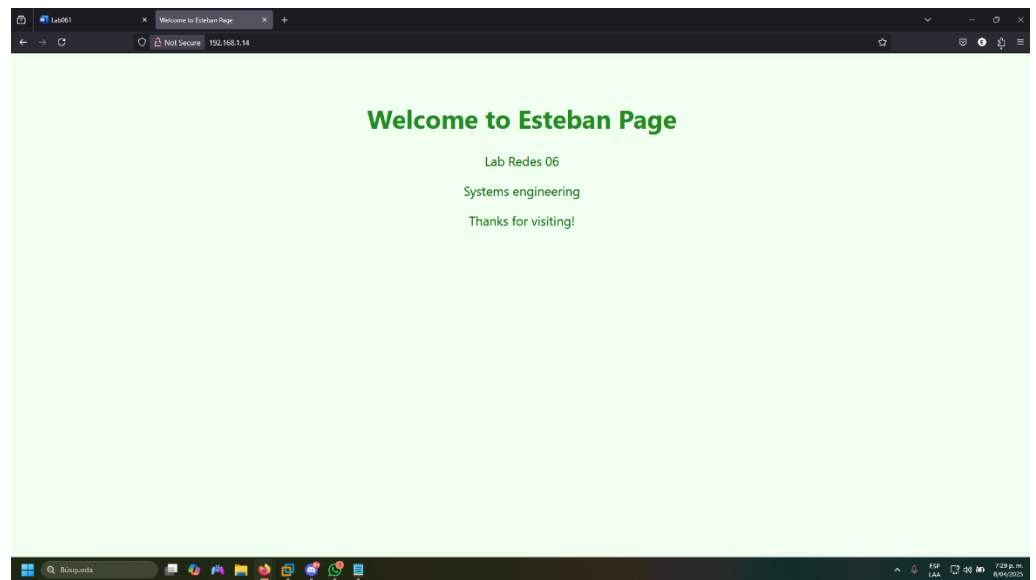




- d. Create a simple webpage on each server to test that it works
  - i. NetBSD



- ii. Slackware



- e. Configure the web servers to start automatically when the operating system boots.
  - i. NetBSD
    - 1. Una vez instalado, configuramos el servicio para iniciarse automáticamente en la ruta “/etc/rc.conf” con nano accedemos, y ya adentro ponemos “apache=YES”

```
GNU nano 8.3 /etc/rc/.conf Modified
# If this is not set to YES, the system will drop into single-user mode.
#
rc_configured=YES

# Add local overrides below.
#
wscons=YES

# ifconfig_wm0="dhcp"

smbd=YES
nmbd=YES
winbindd=YES
samba=YES
named=YES
ntpd=YES

apache = YES

hostname=Juanito.is.escuelaing.edu.co

^G Help      ^O Write Out ^F Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify   ^_ Go To Line
or press Ctrl+G.
```

## ii. Slackware

1. Accedemos a la ruta “/etc/rc.d/rc.local”

```
root@juanito:~# nano /etc/rc.d/rc.local_
```

2. Agregamos “/usr/local/nginx/sbin/nginx” para iniciar automaticamente el nginx

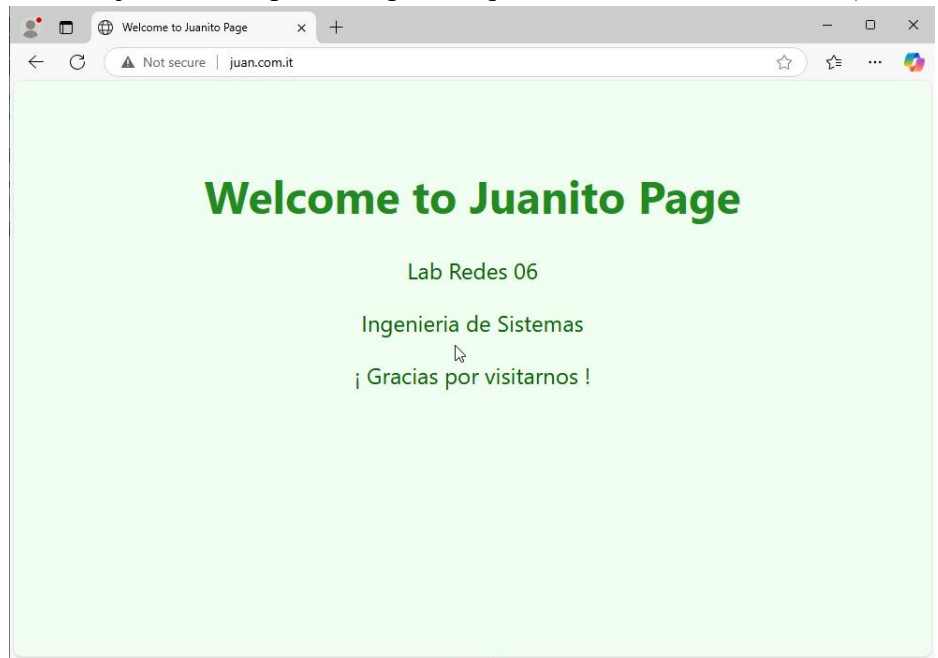
```
GNU nano 6.0 /etc/rc.d/rc.local Modified
#!/bin/bash
#
# /etc/rc.d/rc.local: Local system initialization script.
#
# Put any local startup commands in here. Also, if you have
# anything that needs to be run at shutdown time you can
# make an /etc/rc.d/rc.local_shutdown script and put those
# commands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

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

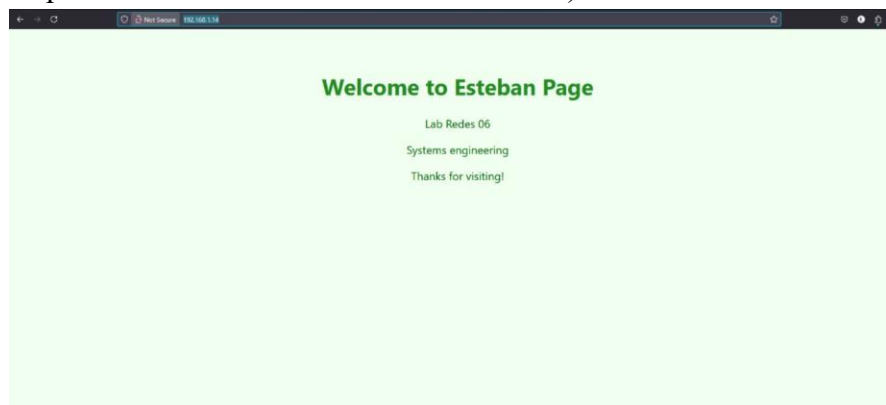
^G Help      ^O Write Out ^F Where Is  ^K Cut       ^T Execute   ^C Location  ^M-U Undo
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify   ^_ Go To Line ^M-E Redo
```

- f. From another computer, test access to the web servers
- Testeamos el servicio desde la maquina de Windows hacia los otros servidores web
  - Para esto, abrimos el navegador web y escribimos la dirección ip de las maquinas
  - Pruebas:
- NOTA: Se realizo pruebas con el DNS en netBSD, slack con otra ip asignada por DHCP y Windows con la ip de la maquina para comprobar que tanto el DNS como las configuraciones de red estén funcionando adecuadamente

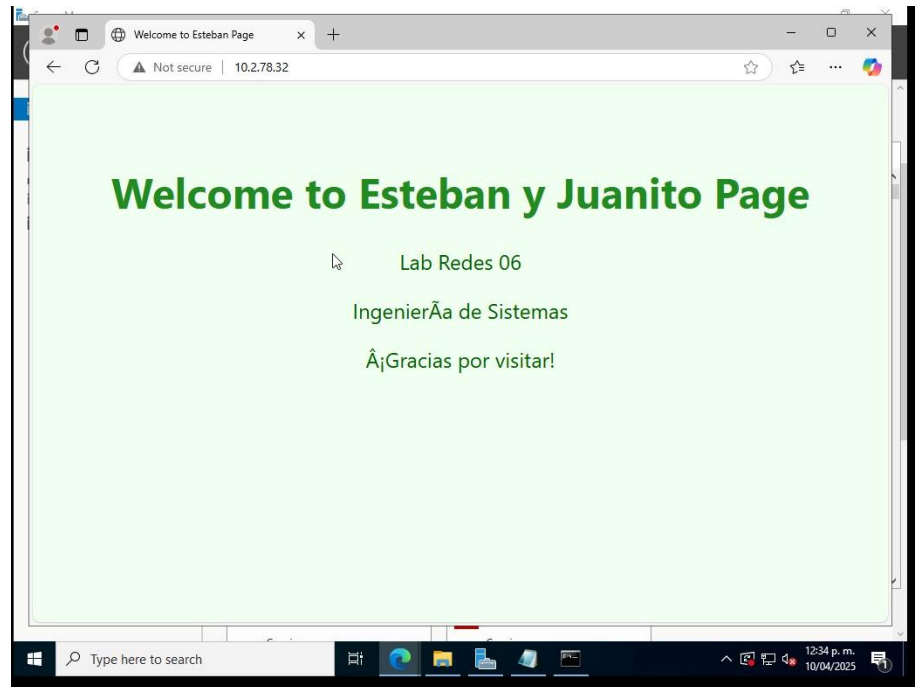
1. Netbsd (juan.com.it para comprobar que el DNS esté funcionando)



2. Slackware (192.168.1.14 ya que se trabajo con DHCP desde un dispositivo fuera de la red de la universidad)



3. Windows (10.2.78.32)



- g. Configure the DNS service so that each web server can be accessed by name
  - i. NetBSD
    - 1. Accedemos al archivo de configuración del DNS que es la ruta “/etc/namedb/juan.com.it” en nuestro caso ( se puede ver en la ruta /etc/named.conf )

```
GNU nano 8.3 /etc/named.conf
#
# This forces all queries to come from port 53; might be
# needed for firewall traversals but should be avoided if
# at all possible because of the risk of spoofing attacks.
#
#query-source address * port 53;
};

zone "juan.com.it" {
    type master;
    file "/etc/namedb/juan.com.it";
    allow-transfer { 10.2.78.36; 10.2.78.32; };
};

zone "esteban.org.uk" {
    type slave;
    file "/etc/namedb/esteban.org.uk";
    masters { 10.2.78.36; };
};

Juanito# nano /etc/namedb/juan.com.it
```

- 2. Como podemos ver ya está configurado la página web, con la ip “10.2.78.35” y el nombre web o con el alias “pagina” .
    - 3. La cambiamos a 10.2.78.36 ya que es la ip de la maquina de netBSD

```

GNU nano 8.3 /etc/namedb/juan.com.it Modified
3600000 ; Expire
3600 ) ; Minimum
                                IN      NS      juan.com.it.
juan.com.it. IN      A      10.2.78.35
juan.com.it. IN      AAAA   ::ffff:a02:4e23
esteban.org.uk. IN     NS     esteban.org.uk
esteban.org.uk. IN     A      10.2.78.36
mail IN      A      10.2.78.35
web IN      A      10.2.78.37
services IN     A      10.2.78.38
rodriguez IN     AAAA   ::ffff:0a02:4e24
aguilera IN     AAAA   ::ffff:0a02:4e25
correo IN      CNAME   mail
pagina IN      CNAME   web
juanito IN      CNAME   rodriguez

```

## ii. Slackware

1. Accedemos al archivo de configuración de DNS (en este caso es esteban.org.uk) mediante la ruta:

```
root@juanito:/var/named/caching-example# nano esteban.org.uk_
```

2. En el laboratorio pasado habíamos configurado varios server, por lo que agregamos en el servidor web la ip de la máquina de slackware (10.2.78.36) , por lo que cuando accedamos a un navegador mediante web.esteban.org.uk nos va a mostrar la web que creamos anteriormente (asimismo funciona para esteban.org.uk)

```

GNU nano 6.0                                esteban.org.uk
; $NetBSD: localhost,v 1.2 2000/05/19 13:07:37 sommerfeld Exp $

$TTL      3600
@         IN      SOA      esteban.org.uk. hostmaster.esteban.org.uk. (
                                1999012100      ; Serial
                                3600             ; Refresh
                                300              ; Retry
                                3600000          ; Expire
                                3600 )           ; Minimum

                                IN      NS       esteban.org.uk.

esteban.org.uk. IN      A      10.2.78.36

mail      IN      A      10.2.78.36
web        IN      A      10.2.78.36
services  IN      A      10.2.78.38

@         IN      MX       mail.juan.com.it
rodriguez IN      AAAA     ::ffff:0a02:4e24
aguilera  IN      AAAA     ::ffff:0a02:4e25

correo    IN      CNAME    mail
pagina    IN      CNAME    web
juanito    IN      CNAME    rodriguez

```

### iii. Windows

1. Para el caso de Windows , en la configuración de dns de netbsd ( a la cual accedimos anteriormente) agregamos la ip y el nombre del servidor . En este caso es webwindows

```

GNU nano 8.3                                juan.com.it
juan.com.it. IN      A      10.2.78.35
juan.com.it. IN      AAAA     ::ffff:a02:4e23

esteban.org.uk. IN      NS       esteban.org.uk
esteban.org.uk. IN      A      10.2.78.36

web        IN      A      10.2.78.35
webwindows IN      A      10.2.78.32
services  IN      A      10.2.78.38

rodriguez  IN      AAAA     ::ffff:0a02:4e24
aguilera   IN      AAAA     ::ffff:0a02:4e25

pagina     IN      CNAME    web
juanito    IN      CNAME    rodriguez

```

### iv. Pruebas

Probamos que los servidores puedan ser accedidos por nombre desde las 3 maquinas

NOTA: Para el caso de netbsd y slackware usamos curl “nombreServidor” para acceder a los servidores creados anteriormente. En el caso de

Windows simplemente buscamos el nombre del servidor en cualquier navegador.

1. Desde netbsd

a. Web.Esteban.org.uk

```

                                background-color: #f0fff0;
                                text-align: center;
                                padding: 50px;
                                font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-s
erif;
                                color: #006400;
                                }
                                h1 {
                                color: #228B22;
                                font-size: 3em;
                                }
                                p {
                                font-size: 1.5em;
                                }
                                }
                                </style>
</style>
</head>
<body>
    <h1>Welcome to Esteban Page</h1>
    <p>Lab Redes 06</p>
    <p>Systems engineering</p>
    <p>Thanks for visiting!</p>
</body>
</html>
Juanito# curl http://web.esteban.org.uk
```

b. Web.juan.com.it

```

                                <style>
                                body {
                                background-color: #f0fff0;
                                text-align: center;
                                padding: 50px;
                                font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
                                color: #006400;
                                }
                                h1 {
                                color: #228B22;
                                font-size: 3em;
                                }
                                p {
                                font-size: 1.5em;
                                }
                                }
                                </style>
</head>
<body>
    <h1>Welcome to Juanito Page</h1>
    <p>Lab Redes 06</p>
    <p>Ingenieria de Sistemas</p>
    <p>¡ Gracias por visitarnos !</p>
</body>
</html>
Juanito# curl http://web.juan.com.it
```

c. Webwindows.juan.com.it

```

<title>Welcome to Esteban Page</title>
<style>
  body {
    background-color: #f0fff0;
    text-align: center;
    padding: 50px;
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    color: #006400;
  }
  h1 {
    color: #228B22;
    font-size: 3em;
  }
  p {
    font-size: 1.5em;
  }
</style>
</head>
<body>
  <h1>Welcome to Esteban y Juanito Page</h1>
  <p>Lab Redes 06</p>
  <p>Ingeniería de Sistemas</p>
  <p>¡Gracias por visitar!</p>
</body>
Juanito# curl http://webwindows.juan.com.it

```

## 2. Desde slacware

### a. Web.Esteban.org.uk

```

root@juanito:/var/named/caching-example# curl http://web.esteban.org.uk
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to Esteban Page</title>
  <style>
    body {
      background-color: #f0fff0;
      text-align: center;
      padding: 50px;
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      color: #006400;
    }
    h1 {
      color: #228B22;
      font-size: 3em;
    }
    p {
      font-size: 1.5em;
    }
  </style>
</head>
<body>
  <h1>Welcome to Esteban Page</h1>
  <p>Lab Redes 06</p>
  <p>Systems engineering</p>
  <p>Thanks for visiting!</p>
</body>
</html>
root@juanito:/var/named/caching-example# _

```

### b. Web.juan.com.it



```

root@juanito:/var/named/caching-example# curl http://web.juan.com.it
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to Juanito Page</title>
  <style>
    body {
      background-color: #f0fff0;
      text-align: center;
      padding: 50px;
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      color: #006400;
    }
    h1 {
      color: #228B22;
      font-size: 3em;
    }
    p {
      font-size: 1.5em;
    }
  </style>
</head>
<body>
  <h1>Welcome to Juanito Page</h1>
  <p>Lab Redes 06</p>

```

c. Webwindows.juan.com.it

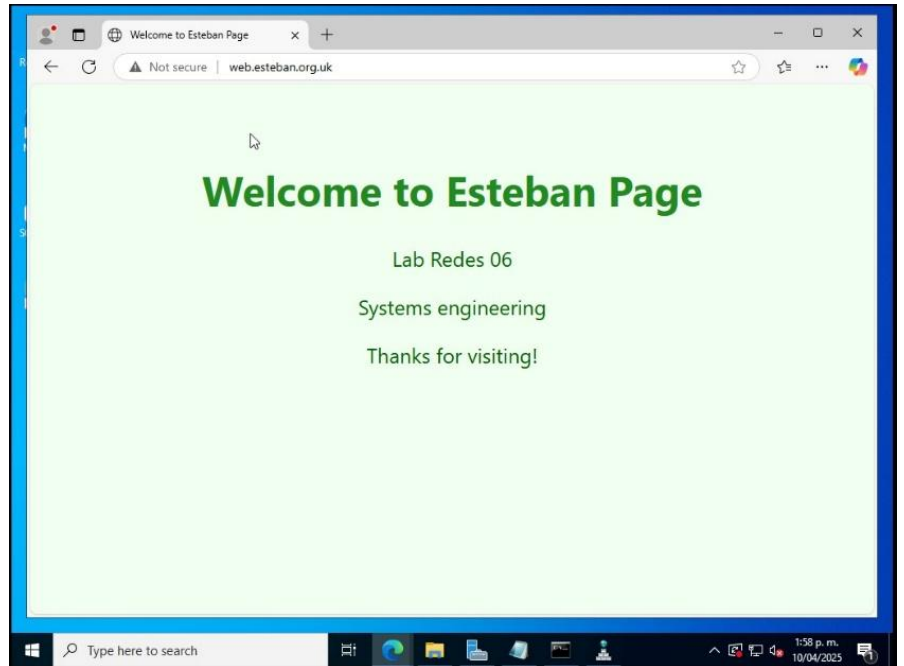
```

root@juanito:/var/named/caching-example# curl http://webwindows.juan.com.it
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to Esteban Page</title>
  <style>
    body {
      background-color: #f0fff0;
      text-align: center;
      padding: 50px;
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      color: #006400;
    }
    h1 {
      color: #228B22;
      font-size: 3em;
    }
    p {
      font-size: 1.5em;
    }
  </style>
</head>
<body>
  <h1>Welcome to Esteban y Juanito Page</h1>
  <p>Lab Redes 06</p>
  <p>Ingeniería de Sistemas</p>
  <p>¡Gracias por visitar!</p>
</body>
</html>root@juanito:/var/named/caching-example#

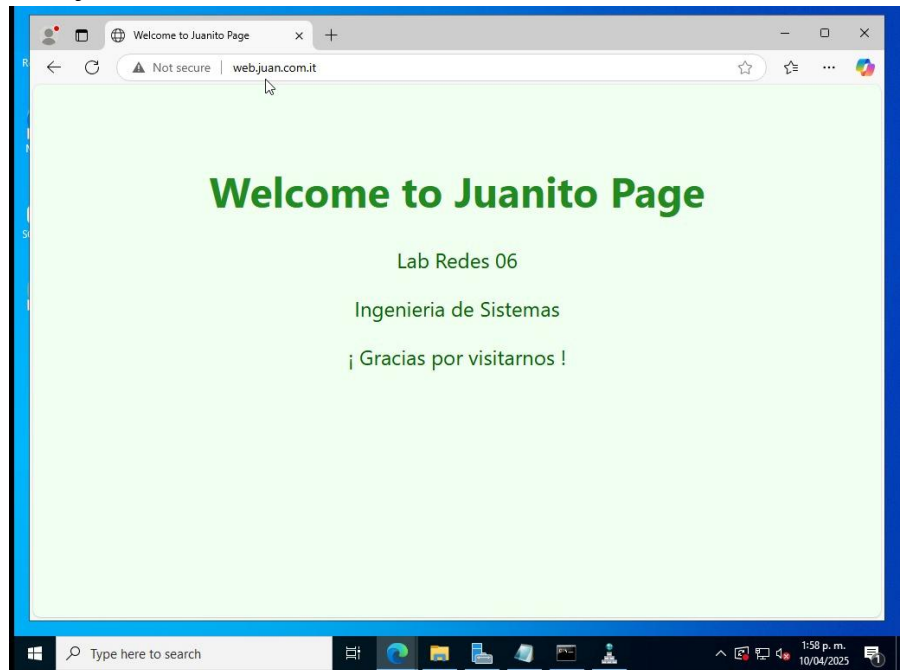
```

3. Desde windows

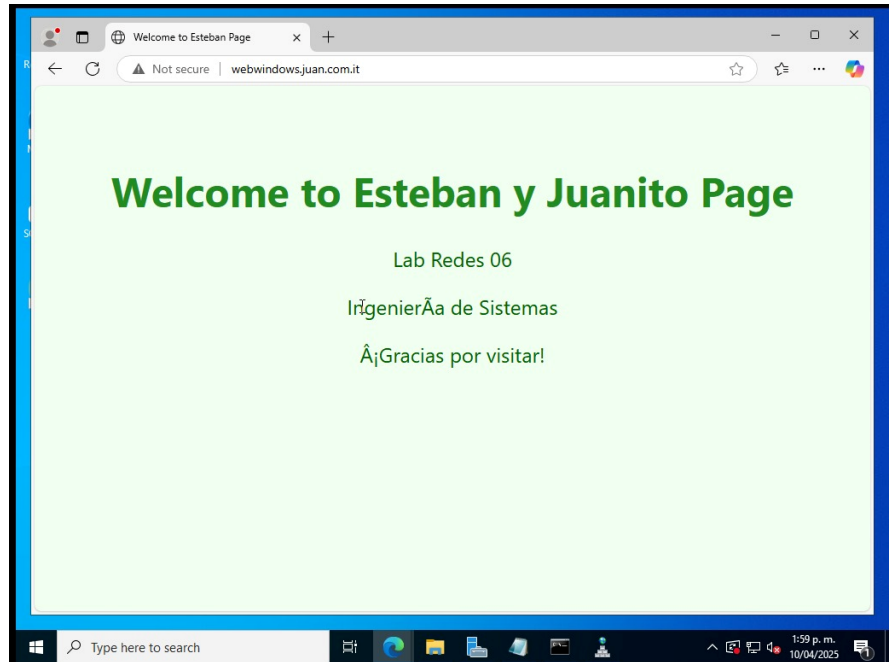
a. Web.Esteban.org.uk



b. [Web.juan.com.it](http://Web.juan.com.it)



c. [Webwindows.juan.com.it](http://Webwindows.juan.com.it)



## 2. Hosting Service Configuration

On the machine with the NetBSD operating system, configure the virtual host service so that hosting services can be provided to multiple companies. The service should be configured in such a way that three webpages hosted on the same web server can be accessed by name, using the URLs of two different domains.

Create virtual hosts that can be accessed as shown below, all of which will reach the same web server installed on NetBSD, but each will direct to different webpages.

- i. network.<name student 01>.com.co
- ii. security.<name student 02>.org.jp

### a. Configure the virtual host service in Apache.

- i. Primero necesitamos configurar Apache para que maneje los tres sitios web diferentes. En nuestro caso NetBSD, el archivo de configuración de Apache se encuentra en “/usr/pkg/etc/httpd/httpd.conf”, y una vez adentro de ese archivo tenemos que descomentariar la siguiente línea “LoadModule vhost\_alias\_module libexec/apache24/mod\_vhost\_alias.so” con el fin de proporcionar funcionalidades adicionales para hosts virtuales

```
GNU nano 8.3 /usr/pkg/etc/httpd/httpd.conf Modified
#LoadModule heartbeat_module lib/httpd/mod_heartbeat.so
#LoadModule heartMonitor_module lib/httpd/mod_heartMonitor.so
#LoadModule dav_module lib/httpd/mod_dav.so
LoadModule status_module lib/httpd/mod_status.so
LoadModule autoindex_module lib/httpd/mod_autoindex.so
#LoadModule asis_module lib/httpd/mod_asis.so
#LoadModule info_module lib/httpd/mod_info.so
#LoadModule cgid_module lib/httpd/mod_cgid.so
#LoadModule dav_fs_module lib/httpd/mod_dav_fs.so
#LoadModule dav_lock_module lib/httpd/mod_dav_lock.so
LoadModule vhost_alias_module lib/httpd/mod_vhost_alias.so
#LoadModule negotiation_module lib/httpd/mod_negotiation.so
LoadModule dir_module lib/httpd/mod_dir.so
#LoadModule actions_module lib/httpd/mod_actions.so
#LoadModule speling_module lib/httpd/mod_speling.so
#LoadModule userdir_module lib/httpd/mod_userdir.so
LoadModule alias_module lib/httpd/mod_alias.so
LoadModule rewrite_module lib/httpd/mod_rewrite.so

<IfModule unixd_module>
#

^G Help      ^O Write Out ^F Where Is  ^R Cut       ^T Execute   ^G Location
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify   ^_ Go To Line
e or press Ctrl+G.
```

- ii. También debemos de descomentariar esta línea “Include etc/httpd/vhosts/\*.conf” en el mismo archivo con el fin de indicarle a Apache que incluya todos los archivos con extensión .conf del directorio vhosts

```
# Virtual hosts
Include etc/httpd/httpd-vhosts.conf
```

- iii. Ahora creamos los dos directorios para el sitio web con el comando mkdir -p “la ruta en donde lo vamos a guardar y la final el nombre de como queremos que quede guardado”

```
Juanito# mkdir -p /usr/pkg/www/network_Juanito
Juanito# mkdir -p /usr/pkg/www/security_Tebitan
Juanito#
```

- iv. Agrega un index.html de prueba en cada uno, con el siguiente comando
- echo "<h1>Network Page - Juanito</h1>" >  
/usr/pkg/www/network\_Juanito/index.html
  - echo "<h1>Security Page - Tebitan</h1>" >  
/usr/pkg/www/security\_Tebitan/index.html

```
Juanito# mkdir -p /usr/pkg/www/network_Juanito
Juanito# mkdir -p /usr/pkg/www/security_Tebitan
Juanito</h1>" > /usr/pkg/www/network_Juanito/index.html
Tebitan</h1>" > /usr/pkg/www/security_Tebitan/index.html
Juanito#
```

- v. Primero, necesitamos configurar Apache para que maneje los tres sitios web diferentes, el archivo de configuración de Apache se encuentra en `/usr/pkg/etc/httpd/httpd-vhosts.conf`

```
Juanito# ls /usr/pkg/etc/httpd
httpd-autoindex.conf          httpd-multilang-errordoc.conf
httpd-dav.conf                httpd-ssl.conf
httpd-default.conf            httpd-userdir.conf
httpd-info.conf               httpd-vhosts.conf
httpd-languages.conf          httpd.conf
httpd-manual.conf              magic
httpd-mpm.conf                mime.types
```

- vi. Una vez dentro añadimos los host Virtuales en apache

```
GNU nano 8.3 /usr/pkg/etc/httpd/httpd-vhosts.conf Modified
<VirtualHost *:80>
    ServerAdmin webmaster@dummy-host2.example.com
    DocumentRoot "/usr/pkg/docs/dummy-host2.example.com"
    ServerName dummy-host2.example.com
    ErrorLog "/var/log/httpd/dummy-host2.example.com-error_log"
    CustomLog "/var/log/httpd/dummy-host2.example.com-access_log" common
</VirtualHost>

<VirtualHost *:80>
    ServerName network.Juanito.com.co
    DocumentRoot "/usr/pkg/www/network_Juanito"
</VirtualHost>

<VirtualHost *:80>
    ServerName security.Tebitan.org.jp
    DocumentRoot "/usr/pkg/www/security_Tebitan"
</VirtualHost>
```

- vii. Una vez finalizamos la configuración virtuales ya estarían listos, pero falta modificar unas líneas para que puedan acceder desde cualquier lugar, no solo de manera local y esto se hace en el archivo `"/usr/pkg/etc/httpd/httpd.conf"` y cambiamos lo de la imagen

```
GNU nano 8.3 /usr/pkg/etc/httpd/httpd.conf Modified
#
# If your host doesn't have a registered DNS name, enter its IP address here.
#
#ServerName www.example.com:80
#
# Deny access to the entirety of your server's filesystem. You must
# explicitly permit access to web content directories in other
# <Directory> blocks below.
#
<Directory />
    AllowOverride ALL
    Require all granted
</Directory>
#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#
^G Help      ^O Write Out ^F Where Is  ^R Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

- viii. Ahora reiniciamos apache para aplicar los cambios “/etc/rc.d/apache restart”

```
Juanito# /etc/rc.d/apache restart
Stopping apache.
Starting apache.
Juanito#
```

b) Configure the DNS service so that the server names point to the same IP address. Keep in mind that two of the entries belong to one domain and the other to a different domain, so you will need to configure this from the primary DNS servers of each domain.

- i. Lo primero es añadir las nuevas zonas de los host virtuales en la ruta “/etc/named.conf”

```
GNU nano 8.3 /etc/named.conf Modified
zone "." {
    type hint;
    file "/etc/namedb/named.ca";
};

#LAB06
zone "juanito.com.co" {
    type master;
    file "juanito.com.co.hosts";
};

zone "tebitan.org.jp" {
    type master;
    file "tebitan.org.jp.hosts";
};

#zone "localhost" {
#    type master;

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

- ii. Una vez creada las zonas modificamos el archivo de juanito, accedemos a el en donde nosotros lo hallamos guardo, en nuestro caso es “/etc/named/juanito.com.co.hosts” y adentro ponemos nuestro host Virtual que es “network.Juanito.com.co” con la ip de nuestro computador, el NS, no es necesario

```
GNU nano 8.3 juanito.com.co.hosts
; $NetBSD: localhost,v 1.2 2000/05/19 13:07:37 sommerfeld Exp $

$TTL 3600
@ IN SOA juanito.com.co. hostmaster.juanito.com.co. (
    1999012100 ; Serial
    3600       ; Refresh
    300        ; Retry
    3600000    ; Expire
    3600 )     ; Minimum

juanito.com.co. IN NS reco.juan.com.co.
juanito.com.co. IN A 10.2.78.35

reco.juan.com.co. IN A 10.2.78.35
network.Juanito.com.co. IN A 10.2.78.35

```

[ Read 17 lines ]

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

- iii. Una vez creada las zonas modificamos el archivo de tebitan, accedemos a el en donde nosotros lo hallamos guardo, en nuestro caso es “/etc/named/tebitan.org.jp.hosts” y adentro ponemos nuestro host Virtual que es “security.Tebitan.org.jp” con la ip de nuestro computador, el NS, no es necesario

```
GNU nano 8.3 /etc/namedb/tebitan.org.jp.hosts
: $NetBSD: localhost,v 1.2 2000/05/19 13:07:37 sommerfeld Exp $

$TTL      3600
@          IN      SOA      tebitan.org.jp. hostmaster.tebitan.org.jp. (
                                1999012100      ; Serial
                                3600             ; Refresh
                                300              ; Retry
                                3600000         ; Expire
                                3600 )          ; Minimum

tebitan.org.jp.      IN      NS      reco.juan.com.co.
tebitan.org.jp.      IN      A       10.2.78.35

reco.juan.com.co.    IN      A       10.2.78.35
security.Tebitan.org.jp.  IN      A       10.2.78.35

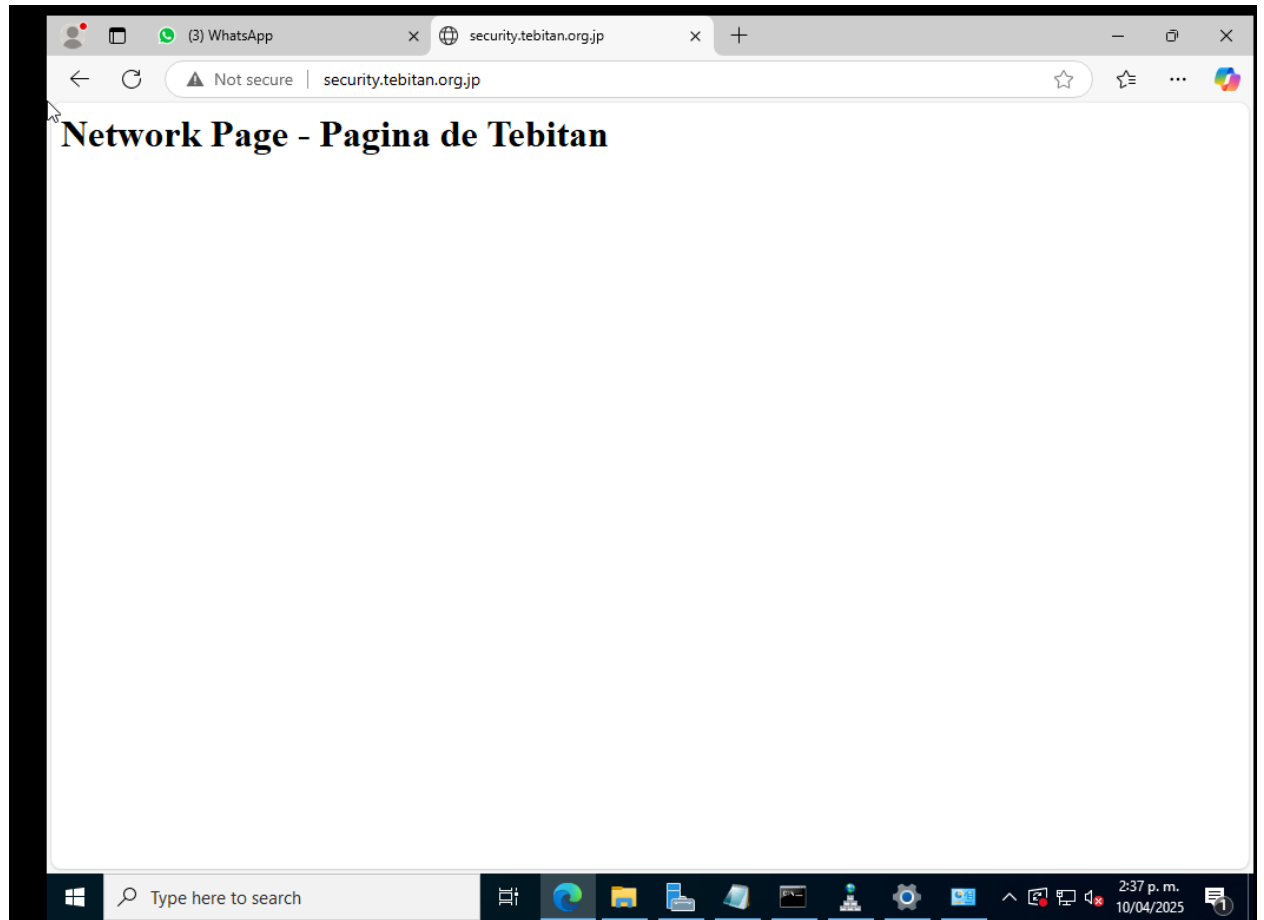
[ Read 17 lines ]
^G Help      ^O Write Out ^F Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

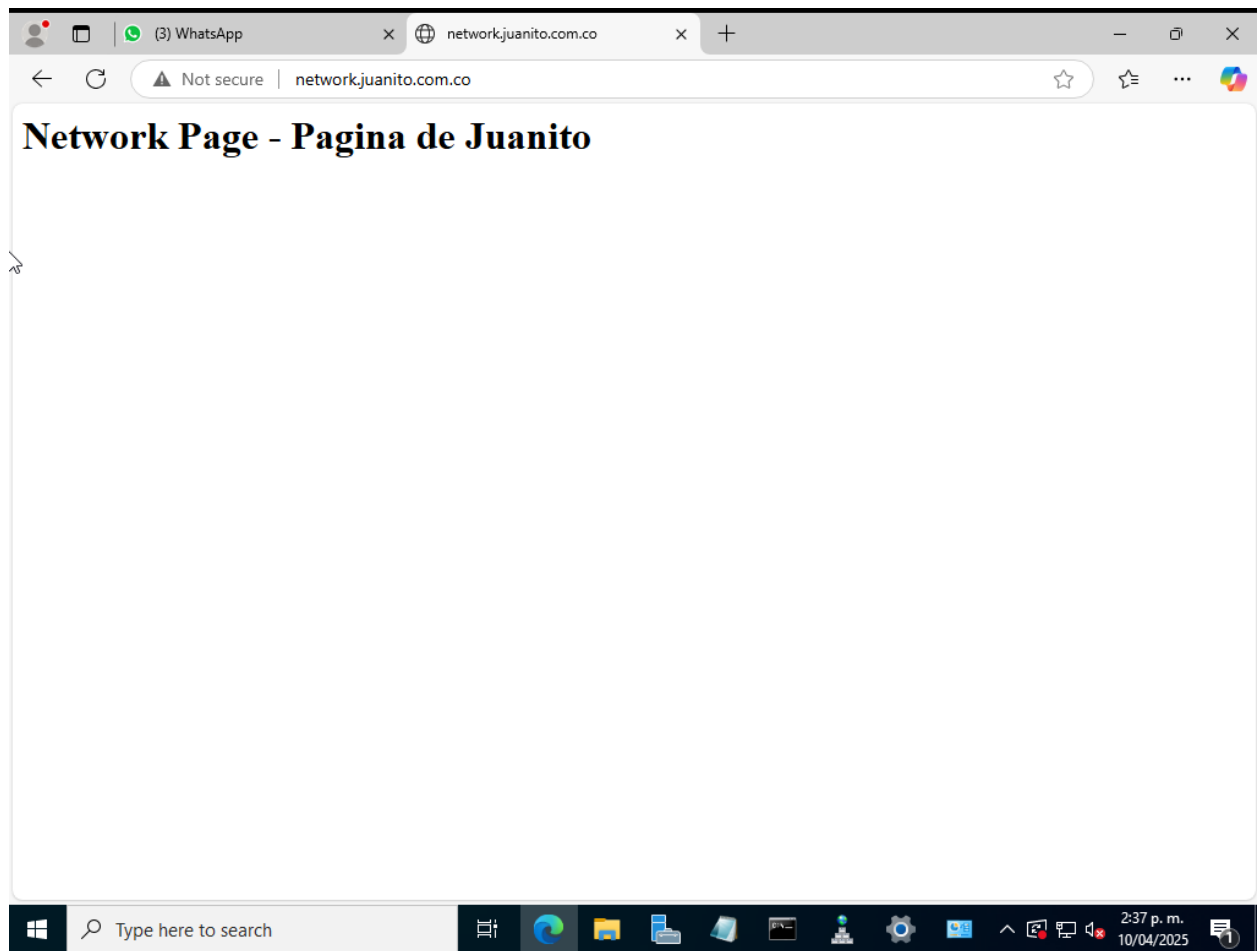
- iv. Ya configuradas las zonas, podemos probarla en nuestro sistema operativo de NetBSD con el comando “curl http:// nuestro host virtual”

```
Juanito# curl http://security.Tebitan.org.jp
<h1> Network Page - Pagina de Tebitan</h1>
Juanito# curl http://network.Juanito.com.co
<h1> Network Page - Pagina de Juanito</h1>
Juanito#
```

- v. Lo probamos en windows con interfaz, y el dns de Windows debe dirigir a nuestra ip de NetBSD







vi. Probamos en slack, de la misma manera

```
root@juanito:/var/named# curl http://security.Tebitan.org.jp
<h1> Network Page - Pagina de Tebitan</h1>
root@juanito:/var/named# curl http://network.Juanito.com.co
<h1> Network Page - Pagina de Juanito</h1>
root@juanito:/var/named# _
```

## CONCLUSIONS

- Durante el desarrollo del laboratorio se logró configurar correctamente los routers Cisco 1941, asignándoles nombres, contraseñas para el acceso privilegiado y configuraciones básicas en las interfaces. Esta parte fue fundamental para entender cómo se gestiona un router desde la consola, cómo se asegura el acceso y cómo se prepara para integrarse en una red más amplia. Además, se practicó el uso de comandos como enable, configure terminal, hostname, y otros esenciales para la administración de dispositivos de red.
- La implementación de rutas estáticas permitió que todos los dispositivos dentro de distintas subredes pudieran comunicarse entre sí. Este ejercicio reforzó el conocimiento sobre el enrutamiento IP y mostró la importancia de definir rutas manuales en redes sin protocolos dinámicos. También fue útil para aprender a identificar errores de conectividad mediante mensajes como “host unreachable” o “request timed out”, y a resolverlos de forma lógica.
- La simulación de la red en Cisco Packet Tracer fue clave para visualizar el comportamiento de la red y comprobar que la configuración de dispositivos era correcta. Se configuraron PCs, routers, y se usaron comandos como ping y traceroute para verificar conectividad. El módulo multiusuario permitió conectar diferentes redes trabajadas por varios estudiantes, simulando un entorno colaborativo donde fue necesario coordinar configuraciones entre equipos.
- En la parte de servicios, se realizó la instalación de servidores web en distintos sistemas operativos: Apache en NetBSD, Nginx en Slackware y IIS en Windows Server. Cada instalación implicó distintos comandos y procedimientos, lo que permitió comparar entornos y entender las diferencias entre sistemas Unix-like y Windows en cuanto a administración de servicios. Además, se configuraron los servidores para que inicien automáticamente con el sistema operativo, una práctica común en ambientes de producción.
- Se crearon páginas web básicas en cada servidor para comprobar su funcionamiento. Estas páginas fueron editadas directamente en los archivos de inicio (como el index.html) y accedidas desde navegadores web y comandos como curl. Esta parte ayudó a reforzar conocimientos básicos sobre rutas del sistema, servicios web y verificación de funcionamiento mediante IPs.
- Una de las partes más importantes fue la configuración de Virtual Hosts en Apache. Esta técnica permitió alojar varios sitios web en un mismo servidor utilizando diferentes nombres de dominio. Gracias a esto, fue posible acceder a sitios como network.nombre.com.co y security.nombre.org.jp, simulando un entorno real de hosting web donde múltiples páginas comparten una sola máquina física o virtual.
- Finalmente, se configuró el servicio DNS para que cada nombre de dominio apuntara a su respectiva IP. Este paso fue esencial para lograr que los sitios fueran accesibles por nombre, y no solo por dirección IP, mejorando la experiencia del usuario. Se aprendió a editar archivos de zona y a integrar el DNS con los servidores web, lo cual es una habilidad clave en administración de redes y servicios.

## BIBLIOGRAPHY

- Wikipedia contributors. (2025, 22 marzo). *Ping (networking utility)*.  
[https://en.wikipedia.org/wiki/Ping\\_\(networking\\_utility\)](https://en.wikipedia.org/wiki/Ping_(networking_utility))
- Wikipedia contributors. (2025b, abril 7). *Domain Name System*. Wikipedia.  
[https://en.wikipedia.org/wiki/Domain\\_Name\\_System](https://en.wikipedia.org/wiki/Domain_Name_System)
- Documentation Group. (s. f.). *Documentation: Apache HTTP Server - The Apache HTTP Server Project*. <https://httpd.apache.org/docs/>
- John-Hart. (s. f.). *IIS documentation*. Microsoft Learn. <https://learn.microsoft.com/en-us/iis/>
- *nginx documentation*. (s. f.). <https://nginx.org/en/docs/>
- *Cisco Packet Tracer*. (s. f.). <https://www.netacad.com/cisco-packet-tracer>
- OpenAI. (2025). *ChatGPT* (versión GPT-4) [Modelo de lenguaje]. <https://chat.openai.com/>