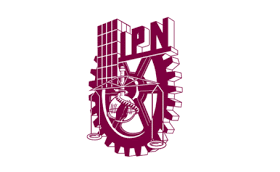
**Instituto Politécnico Nacional**

**Escuela Superior de Cómputo**

Aplicaciones para comunicaciones de red

“Práctica 1”

2do departamental

Profesor: Rangel González Josué

Alumnos:

-Frausto Hernández Omar

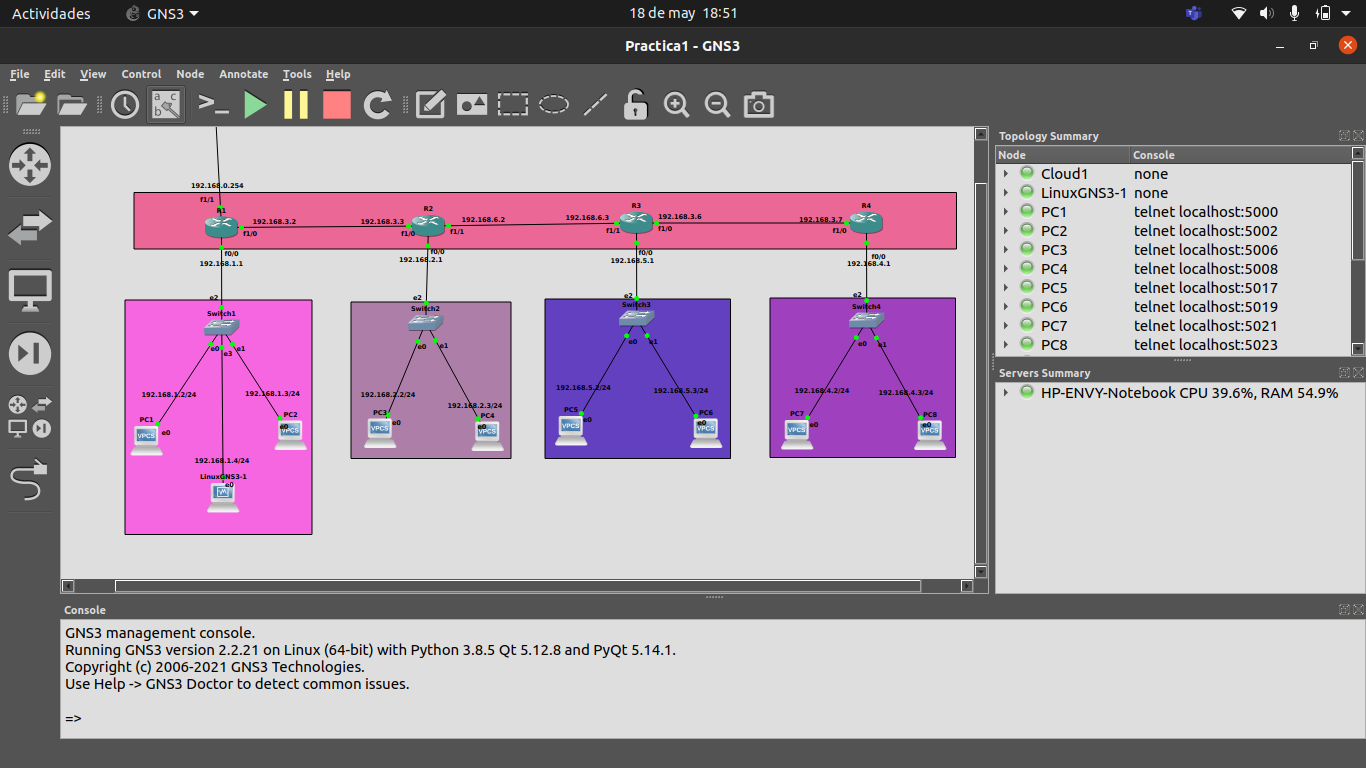
-Guzman Pérez Oscar

-Castro Cruces Jorge Eduardo

Grupo: 3CV18

**Manual de usuario:**

1. **El primer paso fue conectar y configurar la siguiente topología en GNS3:**

****

Los comandos que se utilizaron para configurar las VPCS son los siguientes:

PC1> ip 192.168.1.2/24 192.168.1.1

PC1> save

Para poder ver que la ip fue asignada correctamente, tecleamos:

PC1> show ip

Los comandos que se utilizaron para configurar los Routers Cisco 7200 son los siguientes:

R1# configure terminal

R1(config)# interface f0/0

R1(config-if)# ip address 192.168.1.1 255.255.255.0

R1(config-if)# no shutdown

R1(config-if)# exit

R1(config)# interface f1/0

R1(config-if)# ip address 192.168.3.2 255.255.255.0

R1(config-if)# no shutdown

R1(config-if)# exit

R1(config)# interface f1/1

R1(config-if)# ip address 192.168.0.254 255.255.255.0

R1(config-if)# no shutdown

R1(config-if)# exit

R1(config)# router rip

R1(config-router)# ver 2

R1(config-router)# network 192.168.1.0

R1(config-router)# network 192.168.2.0

R1(config-router)# network 192.168.3.0

R1(config-router)# network 192.168.4.0

R1(config-router)# network 192.168.5.0

R1(config-router)# network 192.168.6.0

R1(config-router)# end

R1# copy running-config startup-config

Para verificar que se configuró de forma correcta el Router, ingresamos los siguientes comandos:

R1# show ip interface brief

R1# show ip route

R1# show run

1. **El siguiente paso fue crear una tap en linux:**

Para este paso, tuvimos que instalar la siguiente biblioteca en nuestra terminal de Linux:

:~$ sudo apt-get install uml-utilities

Luego, tuvimos que crear la tap:

:~$ sudo tunctl -t tap0 -u eduardocastro

Después, asignamos dirección IP y levantamos la tap:

:~$ sudo ifconfig tap0 192.168.0.21 netmask 255.255.255.0 up

Verificamos que se haya creado de forma correcta la tap:

:~$ ifconfig

1. **Para esto, debemos agregar las rutas a la tabla de ruteo para que la tap sepa por donde enviar y recibir los datos:**

Agregamos las rutas indicadas en la tabla de ruteo de nuestro equipo Linux:

:~$ sudo ip route add 192.168.1.0/24 via 192.168.0.254 dev tap0

:~$ sudo ip route add 192.168.2.0/24 via 192.168.0.254 dev tap0

:~$ sudo ip route add 192.168.3.0/24 via 192.168.0.254 dev tap0

:~$ sudo ip route add 192.168.4.0/24 via 192.168.0.254 dev tap0

:~$ sudo ip route add 192.168.5.0/24 via 192.168.0.254 dev tap0

:~$ sudo ip route add 192.168.6.0/24 via 192.168.0.254 dev tap0

Para verificar que se agregó correctamente las rutas indicadas en la tabla de ruteo de nuestro equipo Linux, tecleamos:

:~$ ip route

:~$ route -n

Siguiendo la recomendación del profesor, se nos hizo buena idea codificar un Script que nos facilite la tarea que acabamos de hacer:

#!/bin/bash

#variables

ipTap=192.168.0.21

mask=255.255.255.0

ipTapBroadcast=192.168.0.254

#Crear tap

sudo tunctl -t tap0 -u eduardocastro

echo tap creada...

#Levantar y asignar IP

sudo ifconfig tap0 $ipTap netmask $mask up

echo tap lista y configurada...

echo ip = $ipTap

echo mascara = $mask

#Agregar rutas a la tabla de ruteo

sudo ip route add 192.168.1.0/24 via $ipTapBroadcast dev tap0

sudo ip route add 192.168.2.0/24 via $ipTapBroadcast dev tap0

sudo ip route add 192.168.3.0/24 via $ipTapBroadcast dev tap0

sudo ip route add 192.168.4.0/24 via $ipTapBroadcast dev tap0

sudo ip route add 192.168.5.0/24 via $ipTapBroadcast dev tap0

sudo ip route add 192.168.6.0/24 via $ipTapBroadcast dev tap0

echo Rutas agregadas a la tabla de ruteo

Sin duda, se le tuvo que dar permisos de ejecución al fichero:

:~$ sudo chmod +x ScriptTap0.sh

1. **Después, creamos una máquina virtual de Ubuntu Linux:**

Nosotros ocupamos la aplicación de Oracle VM VirtualBox para crear la Máquina Virtual de Ubuntu Linux, con su respectivo archivo imagen descargado de las correspondientes páginas oficiales.

Una vez instalada la Máquina Virtual, procedemos a realizar algunas configuraciones previas a su vinculación a GNS3:

* Corremos la Máquina Virtual y abrimos una terminal de comandos.
* Instalamos actualizaciones

:~$ sudo apt-get update

:~$ sudo apt-get upgrade

* Instalamos la biblioteca de ifconfig

:~$ sudo apt-get install net-tools

Listo, ya estamos preparados para la vinculación.

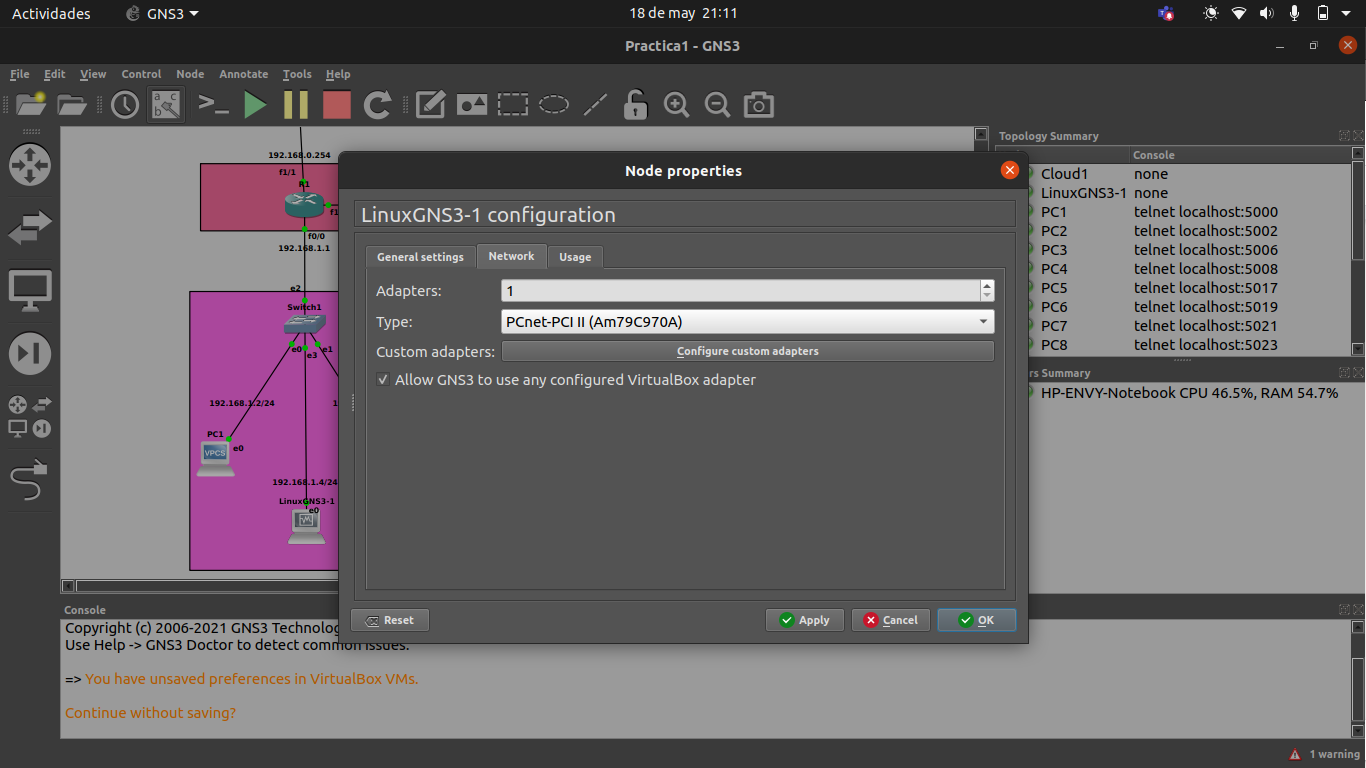
1. **Luego, vinculamos esta máquina virtual a GNS3:**

Para esto, nos vamos al menú superior **Edit > Preferences > VirtualBox VMs**, una vez dentro de esa sección, seleccionamos el botón de **New** y seleccionamos nuestra máquina virtual que creamos, y marcamos el recuadro que dice: **Use as linked base VM (Experimental)**, por último finalizamos la ventana.

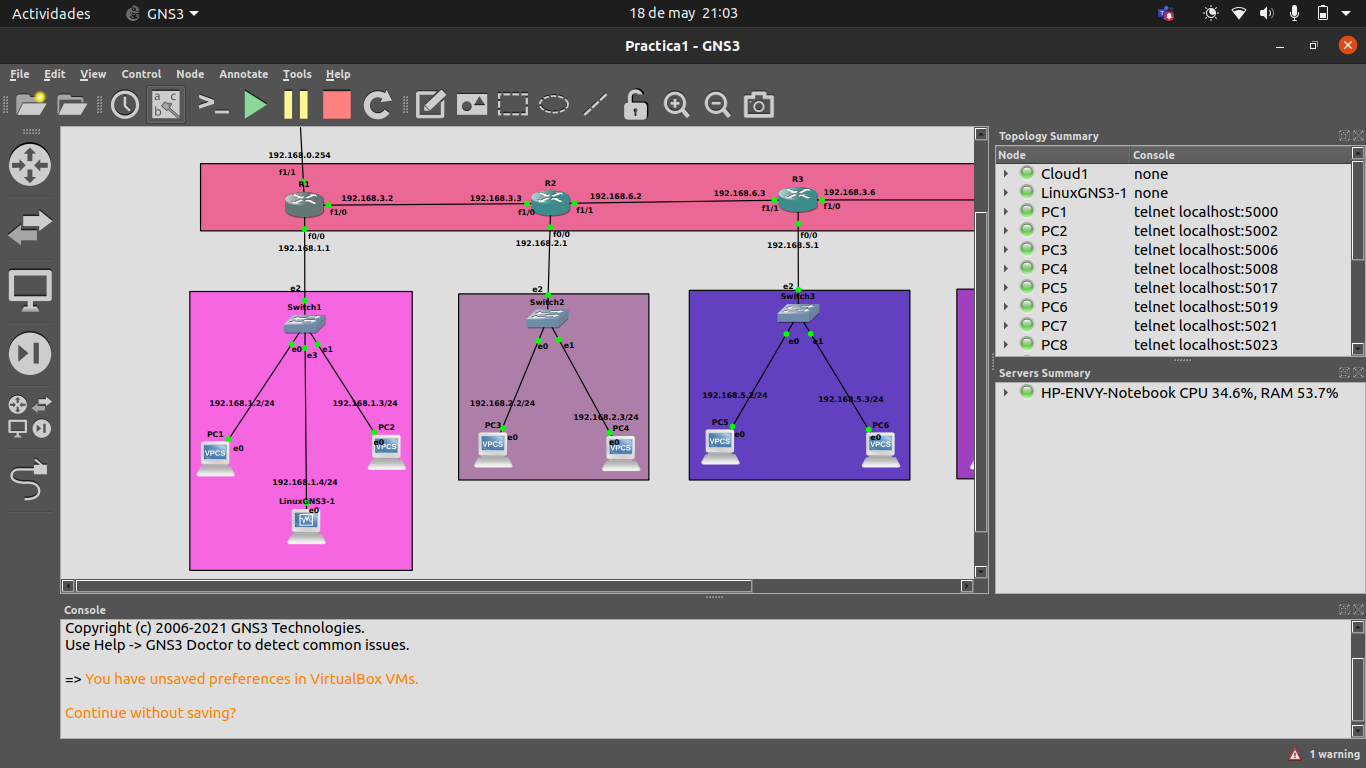
Ahora, nos vamos al ícono lateral izquierdo **Browse End Devices** y arrastramos la Máquina dentro de nuestra topología.

De igual forma, procedemos a realizar una pequeña configuración de nuestra Máquina Virtual:

* Primeramente, antes de encender la Máquina, debemos hacer clic derecho sobre la misma, e ir a la opción de **Configure.**
* Una vez dentro de la ventana emergente, nos vamos a la pestaña de **Network** y hacemos la siguiente modificación.



Listo, nuestra Máquina Virtual está lista para ser conectada al Switch deseado



Lo siguiente es encender la Máquina Virtual y abrir un terminal para poder configurar la nueva dirección IP de la anterior:

:~$ sudo ifconfig enp0s3 192.168.1.4 netmask 255.255.255.0

Ahora, configuramos su nueva puerta de acceso (gateway):

:~$ sudo route add default gw 192.168.1.1 enp0s3

Para verificar que se asignó de forma correcta la dirección IP y el gateway, tecleamos:

:~$ ifconfig

:~$ ip route

:~$ route -n

Podemos realizar ping’s de prueba con los demás equipos para cerciorarse de la correcta conexión:

:~$ ping 192.168.1.2

:~$ ping 192.168.2.2

:~$ ping 192.168.3.2

:~$ ping 192.168.4.2, etc

1. **Acto seguido, realizamos una modificación en el Router 1:**

La única modificación que se le hizo al Router, fue para que permitiese transmitir los mensajes de Broadcast a las demás subredes:

R1# configure terminal

R1(config)# interface f0/0

R1(config-if)# ip directed-broadcast

R1(config-if)# exit

R1(config)# interface f1/0

R1(config-if)# ip directed-broadcast

R1(config-if)# exit

R1(config)# interface f1/1

R1(config-if)# ip directed-broadcast

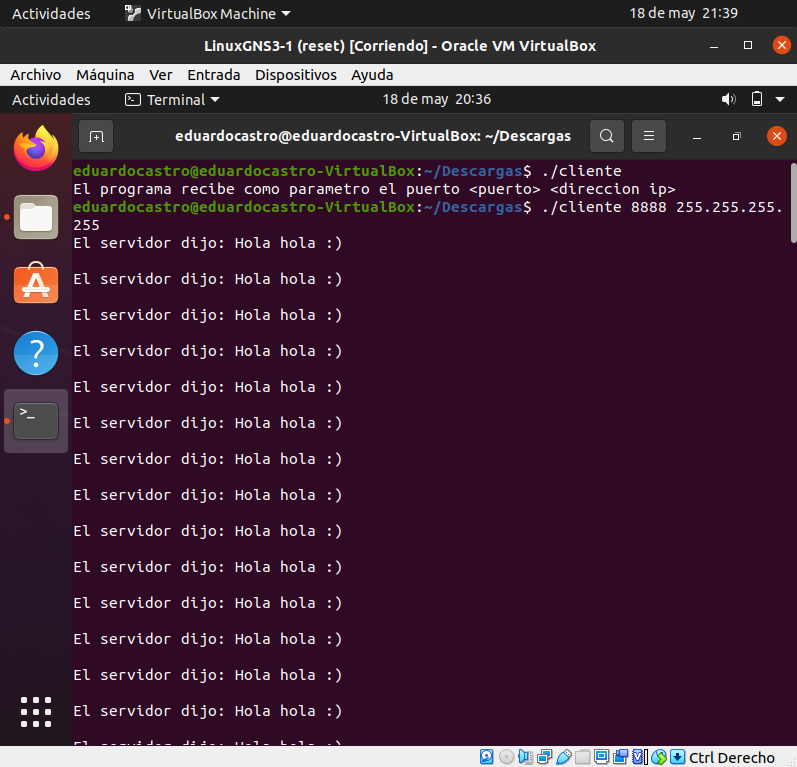
R1(config-if)# end

R1# copy running-config startup-config

1. **Después, ocupamos el programa cliente.c y servidor.c broadcast:**

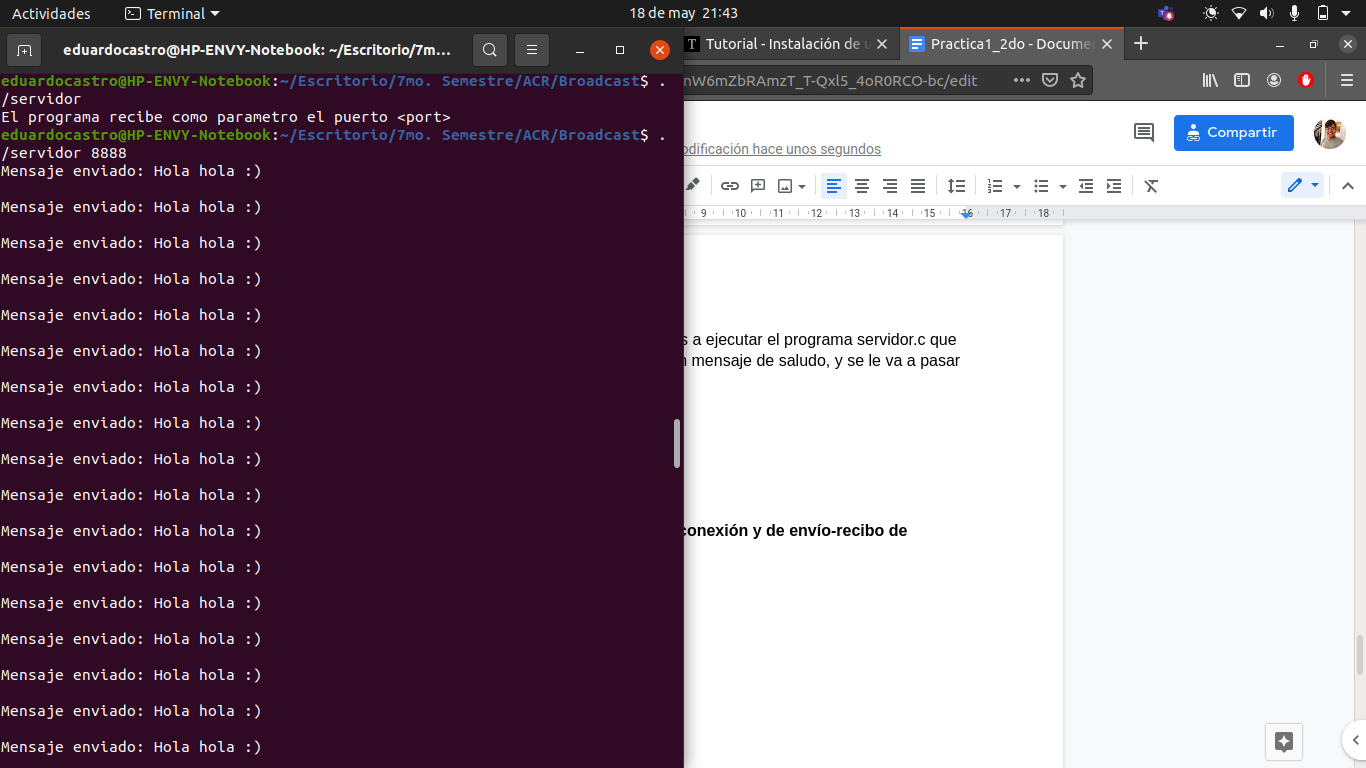
El programa cliente.c se va a ejecutar en todo momento desde la Máquina Virtual de GNS3, y se le van a pasar los parámetros siguiente:

* Puerto: 8888
* Dirección IP: 255.255.255.255



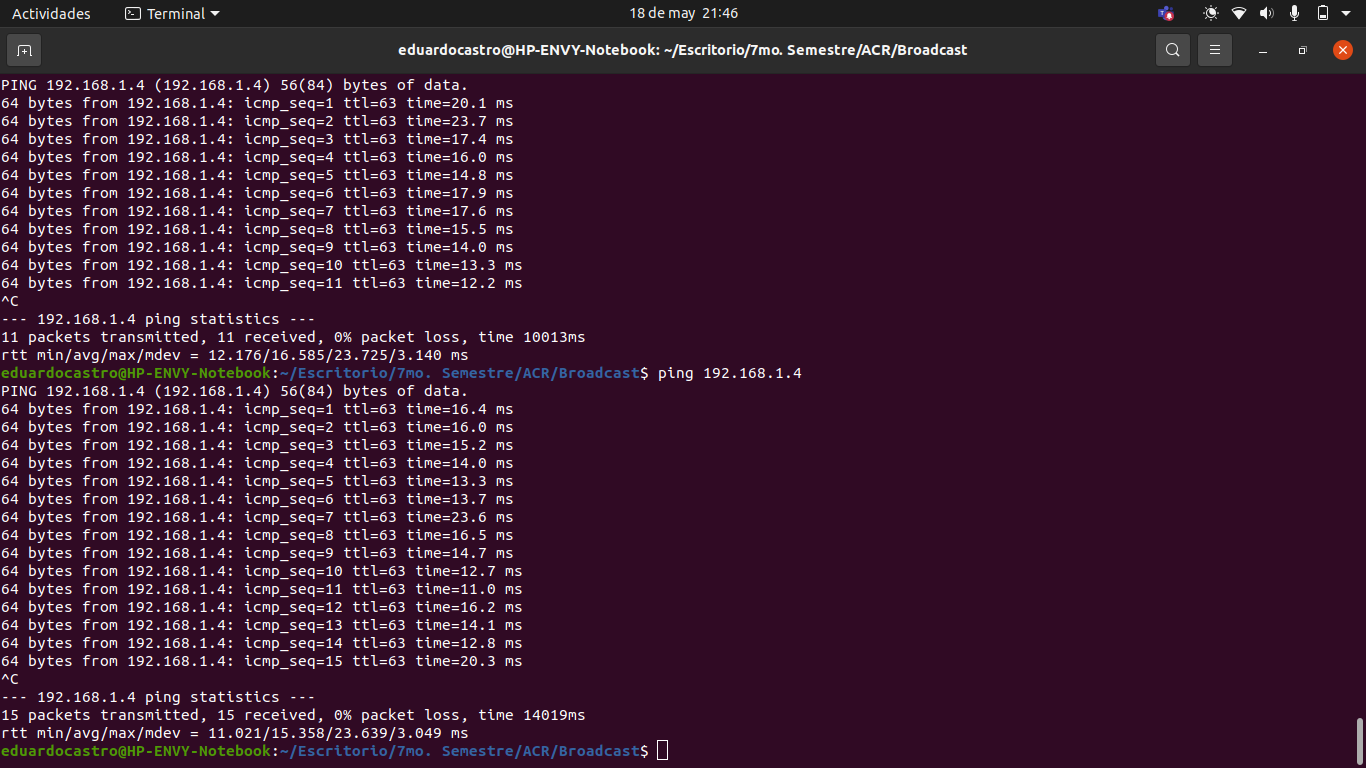
Por otro lado, en nuestro equipo Linux vamos a ejecutar el programa servidor.c que en todo momento va a estar transmitiendo un mensaje de saludo, y se le va a pasar como parámetro lo siguiente:

* Puerto: 8888

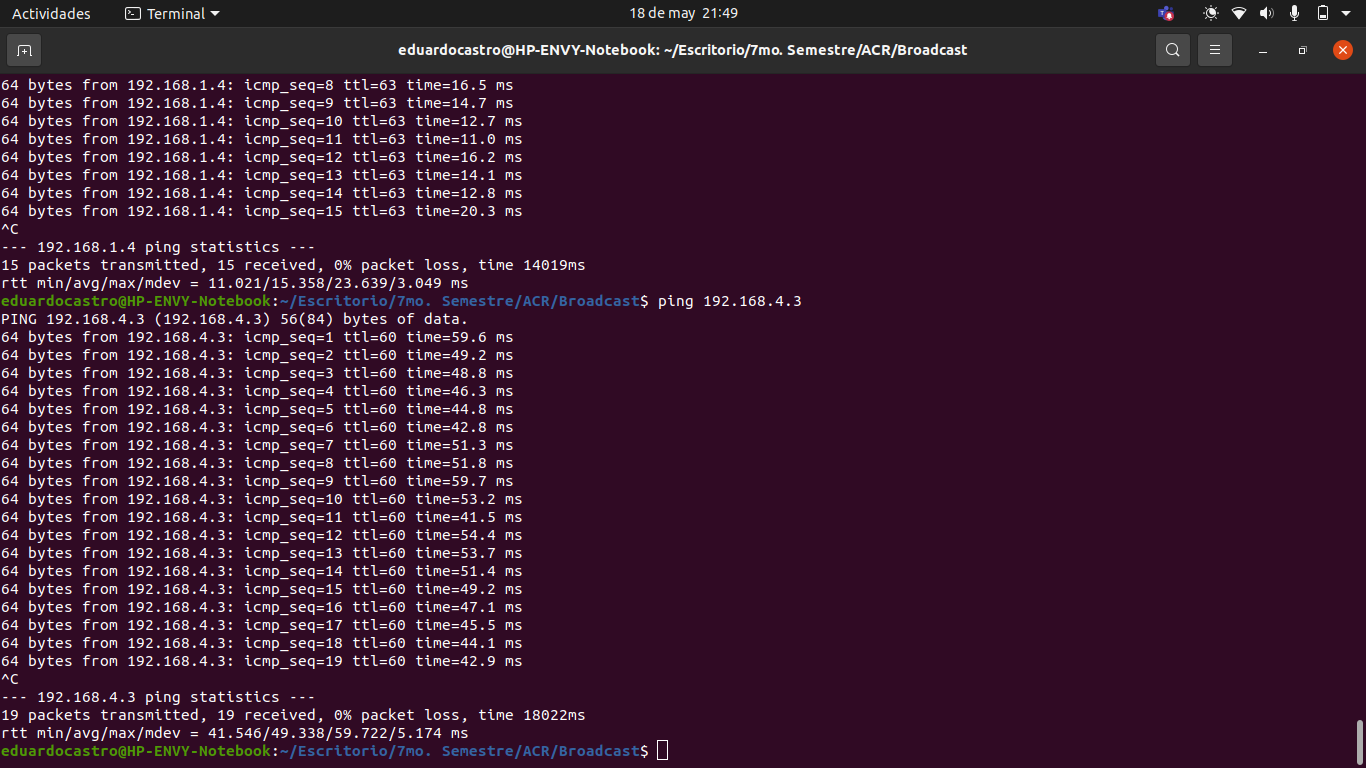


1. **Por último, realizamos pruebas de conexión y de envío-recibo de mensajes:**

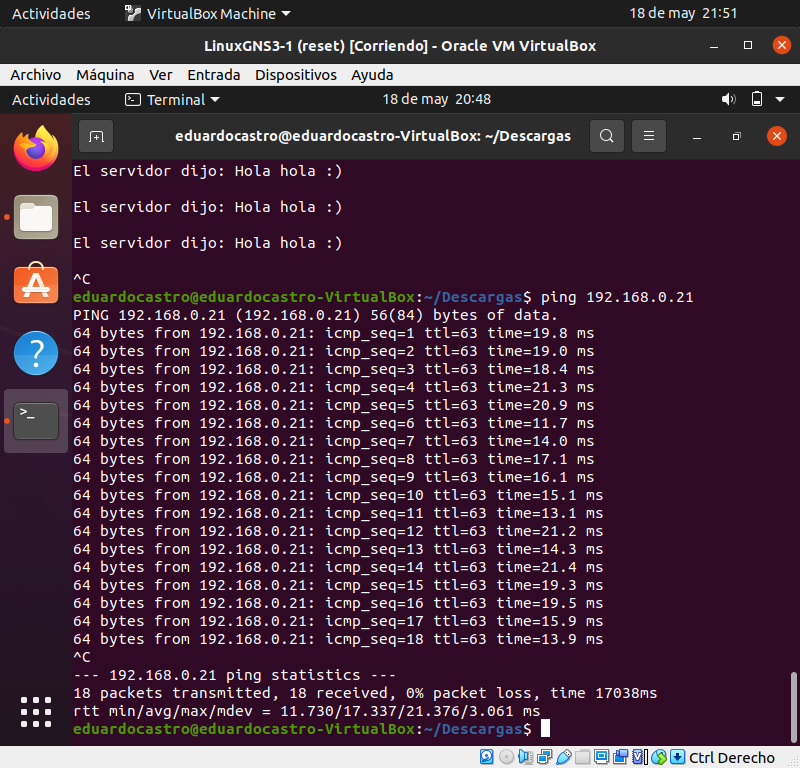
Prueba de conexión entre el Equipo Linux y la Máquina Virtual De GNS3 Linux:

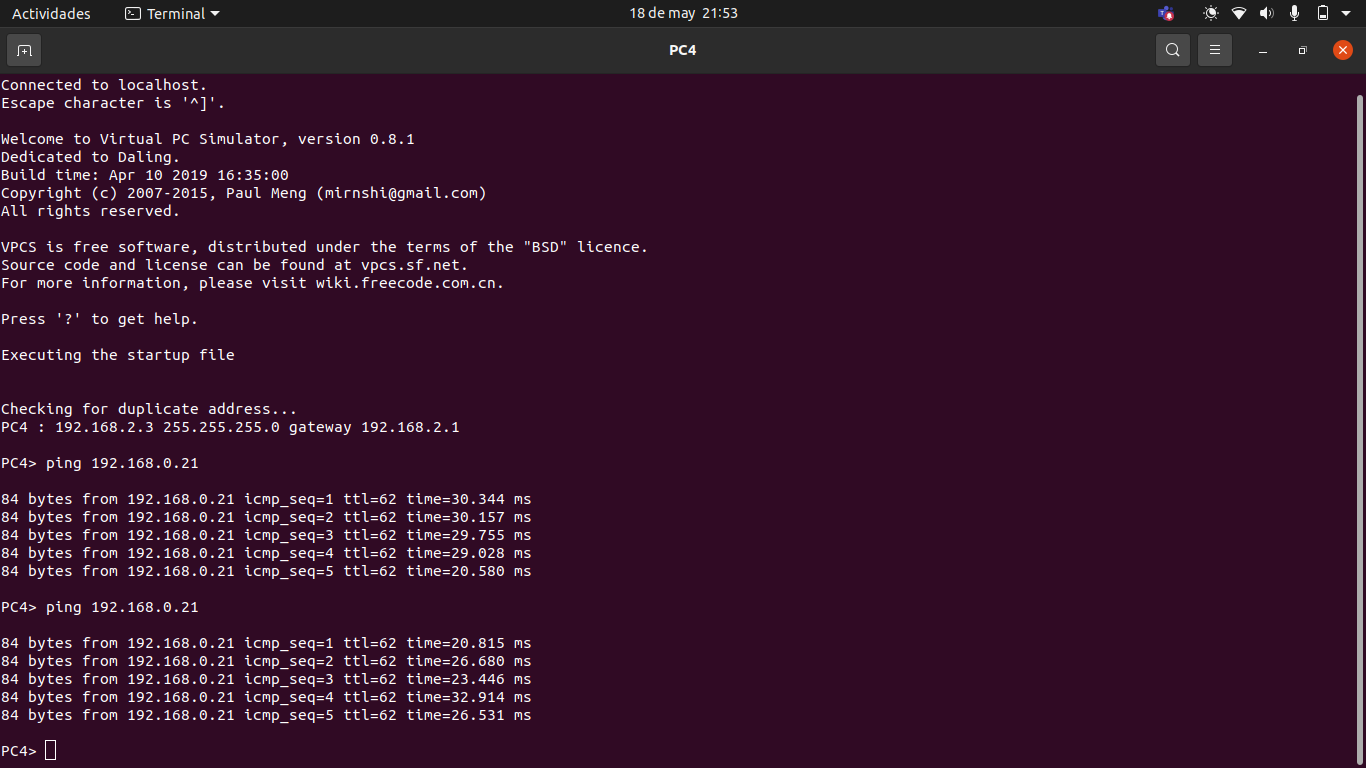


Prueba de conexión entre el Equipo Linux y cualquier VPCs de la topología:



Prueba de conexión entre la Máquina Virtual De GNS3 Linux y el Equipo Linux :



Prueba de conexión entre cualquier VPCs de la topología y el Equipo Linux :

**Conclusiones**

* **Castro Cruces Jorge Eduardo**

En conclusión, puedo decir que el objetivo principal se cumplió, el cual era desarrollar un manual de usuario que muestre paso a paso el procedimiento que realizamos para concluir la práctica. Sin duda, la aplicación GNS3, es bastante parecida a Packet Tracer de CISCO, con la gran diferencia de que en GNS3 se pueden incluir Máquinas Virtuales. Tengo que reconocer que es bastante complicado aprender el correcto funcionamiento y manejo de las Máquinas Virtuales vinculadas a GNS3, porque un pequeño error que cometas, y el archivo de tu topología se arruina permanentemente sin manera de recuperarla, ni siquiera los Snapshots guardados. Tampoco deja guardar Snapshots si hay una Máquina Virtual conectada dentro de la topología. Quedé bastante decepcionado de esta aplicación, espero no tener que usarla de nuevo en otras materias.

* **Guzman Pérez Oscar**

En esta práctica pudimos observar la comunicación a diferente nivel entre una aplicativo cliente y servidor, lo cual considero que fue interesante porque este tipo de comunicación es predominante en la red, a fin de cuentas día a día nos conectamos a una red con una topología inmensa, si bien en este caso la comunicación fue broadcast y es menos común que se retransmita a subredes, podría ser de utilidad cuando realmente este mensaje si tiene que pasar de esta manera, como podría ser un sistema de seguridad que notifique a diferentes áreas si surge un problema por ejemplo, o por ejemplo para recursos compartidos como podría ser una impresora notifique a distintas áreas cuando se está utilizando, ya sea para que en otras terminales sepan que tienen que esperar antes de poder acceder al dispositivo o de igual forma por seguridad.

* **Frausto Hernández Omar**

Fue una práctica bastante interesante ya que fue mi primera vez utilizando Gns3, anteriormente ya había utilizado packet tracer y desde mi punto de vista es mucho mejor y más completo Gns3, lo cual me ayudó mucho para aclarar pequeñas cosas que no sabía o ya se me había olvidado.