

Universidad de San Carlos de Guatemala
Centro Universitario de Occidente
División de Ciencias de la Ingeniería
Lab Redes de Computadoras 1
Ing. Juan Francisco Rojas Santizo
Eriksson José Hernández López – 201830459



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala

Proyecto Final - Documentación Técnica

Eriksson José Hernández López

201830459

Quetzaltenango, 11 de mayo de 2022

Marco Teórico

Debian GNU/Linux:

Debian es una distribución de software libre para sistemas de ordenadores con diferentes arquitecturas de hardware. Debido a que el paquete completo hace uso, por una parte, del núcleo de Linux desarrollado por Linus Torvalds y, por otra, de las herramientas básicas del sistema GNU, también es conocido como Debian GNU/Linux. Actualmente, Debian sigue siendo desarrollado como proyecto comunitario por más de 1000 desarrolladores oficiales alrededor del mundo. Como una de las colecciones más utilizadas y antiguas. Debian cuenta con más de 43000 paquetes de software listos para usar y es considerado, por lo tanto, como una solución universal de sistema operativo. Por otra parte, Debian es una de las distribuciones más influyentes y sirve de base para muchas distribuciones nuevas, como por ejemplo el popular Ubuntu. Adicionalmente, desde Debian 6.0 se ha publicado una versión basada en un núcleo FreeBSD.

Lenguaje Bash

GNU Bash o simplemente Bash (Bourne-again shell) es una popular interfaz de usuario de línea de comandos, específicamente un shell de Unix; así como un lenguaje de comandos o scripting.

Bash es un procesador de comandos que generalmente se ejecuta en una ventana de texto donde el usuario escribe comandos que causan acciones. Bash también puede leer y ejecutar comandos desde un archivo, llamado script de shell.

Bridge - Utils

Este paquete contiene utilidades para configurar el puente Ethernet en Linux. El puente Ethernet de Linux se puede utilizar para conectar varios dispositivos Ethernet juntos. La conexión es totalmente transparente: los hosts conectados a un dispositivo Ethernet ven los hosts conectados a los otros dispositivos Ethernet directamente.

Net Tools

El paquete Net-tools contiene una colección de programas que forman la base del trabajo en red para Linux.

Ifconfig

Es un programa disponible en varias versiones del sistema operativo UNIX, que permite configurar o desplegar numerosos parámetros de las interfaces de red residentes en el núcleo, como la dirección IP (dinámica o estática), o la máscara de red. Si se llama sin argumentos suele mostrar la configuración vigente de las interfaces de red activas, con detalles como la dirección MAC o el tráfico que ha circulado por las mismas hasta el momento. Las interfaces de red en Linux se suelen denominar eth.

Network Manager

NetworkManager es una utilidad de software para simplificar el uso de redes de computadoras en Linux y otros sistemas operativos basados en Unix.

Red Hat inició el proyecto NetworkManager en 2004 con el objetivo de permitir que los usuarios de Linux hacer frente con mayor facilidad a las necesidades de redes modernas, en particular, de redes inalámbricas . NetworkManager adopta un enfoque oportunista de selección de red, tratando de usar la mejor conexión disponible cuando se producen cortes, o cuando el usuario se mueve entre redes inalámbricas. Prefiere conexiones Ethernet a través de redes inalámbricas "conocidas". Se pide al usuario claves WEP o WPA, según sea necesario.

NetworkManager tiene dos componentes:

- Un servicio que administra las conexiones y los informes de cambios en la red.
- Una aplicación gráfica de escritorio que permite al usuario manipular las conexiones de red. El subprograma nmcli proporciona una funcionalidad similar en la línea de comandos.

Iptables

Iptables es una herramienta avanzada de filtrado de paquetes en Linux. Es una herramienta muy establecida, ya que hay millones de sitios en todo el mundo que funcionan y utilizan iptables de forma continua.

De lo que se encarga iptables, dicho de una forma sencilla, es de analizar cada uno de los paquetes del tráfico de red entra en una máquina y decidir, en función de un conjunto de reglas, qué hacer con ese paquete, siempre desde un punto de vista amplio, ya que iptables permite hacer muchas cosas diferentes con el tráfico de red.

Interfaz de Red

Una interfaz de red es el software específico de red que se comunica con el controlador de dispositivo específico de red y la capa IP a fin de proporcionar a la capa IP una interfaz coherente con todos los adaptadores de red que puedan estar presentes.

Qemu

Es un emulador de procesadores basado en la traducción dinámica de binarios (conversión del código binario de la arquitectura fuente en código entendible por la arquitectura huésped). QEMU también tiene capacidades de virtualización dentro de un sistema operativo, ya sea GNU/Linux, Windows, o cualquiera de los sistemas operativos admitidos; de hecho es la forma más común de uso. Esta máquina virtual puede ejecutarse en cualquier tipo de Microprocesador o arquitectura

Virtual Machine Manager

El Hypervisor, también conocido como monitor de máquina virtual, es una herramienta para la creación, ejecución y gestión de máquinas virtuales. Estas máquinas virtuales son las encargadas de alojar servicios virtualizados como escritorios remotos o CPD Virtuales.

Esta herramienta también actúa como panel de control y monitorización de las máquinas virtuales y los recursos asociados a los mismos.

Red de Edificios y Clientes

Descripción

Consiste en la creación de una red de Edificios en Serie, la cual permite la comunicación entre todos los edificios por medio de los enrutadores definidos, y en la creación de computadores Clientes, las cuales se comunicarán por medio de una asignación de IP específica según el Edificio al que pertenezcan. Esta asignación se dará por medio de una Tabla de Routing definida por cada estudiante.

Herramientas

- Imagen ISO del Sistema Operativo Debian 11 sin Interfaz gráfica
- Una Computadora Madre, encargada de la Virtualización
- Virtual Machine Manager
- Qemu
- Sistema de Paquetería de Debian
- Lenguaje Bash
- Conexión a Internet mediante un Router
- Servidor SSH
- Bridge's

Características

- Uso del Sistema Operativo Debian 11 sin Interfaz Gráfica mediante Máquinas Virtuales
- Creación de Puentes de Red, mediante el uso de la Tarjeta de Red de la Computadora a utilizar
- Comunicación entre Edificios por medio de los Enrutadores
- Comunicación entre Clientes por medio de Routing
- Conexión eficiente entre dispositivos conectados a la red

Estructura de Comunicación entre Dispositivos

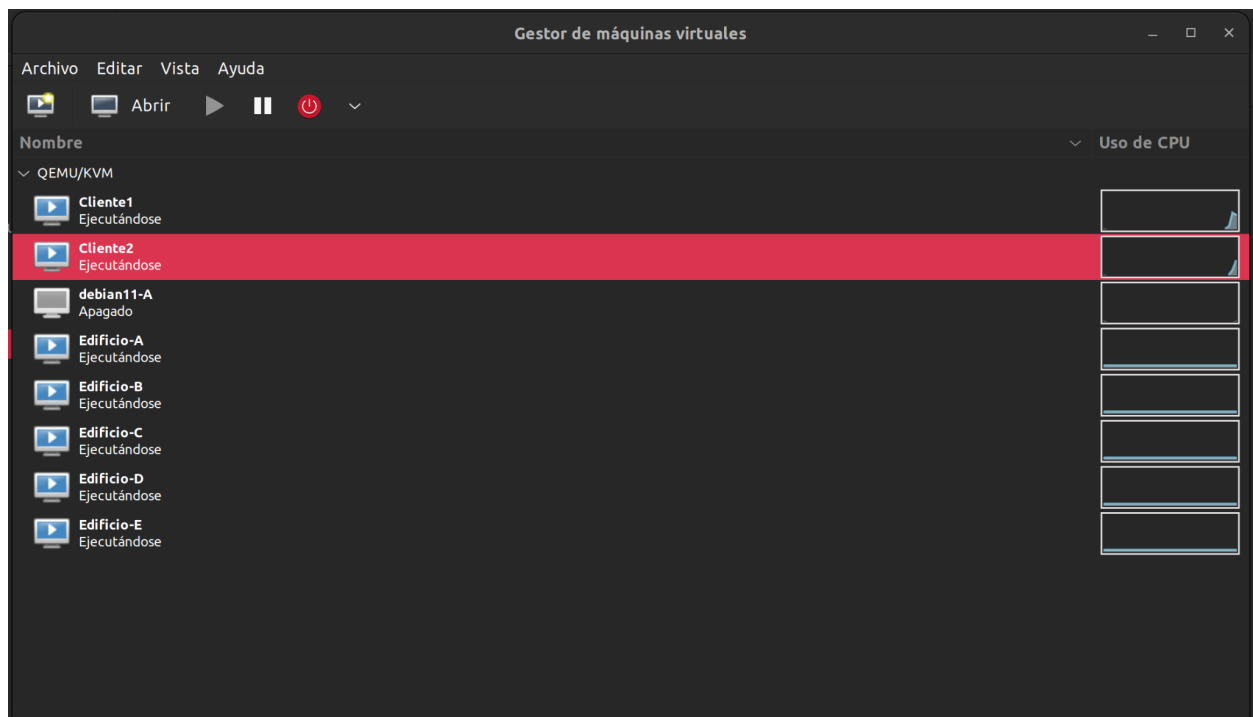
La Comunicación entre dispositivos se da de manera restringida por la siguiente tabla:

Edificio	Dispositivo	Dispositivos con los que se puede comunicar
A		
	A_a	B_a E_a E_b
	A_b	B_a E_a E_b
	A_c	B_a E_a E_b
B		
	B_a	A_a A_b A_c
	B_b	A_a A_b A_c E_a E_b
C		
	C_a	D_a
	C_b	D_a
	C_c	D_a
	C_d	D_a
	C_e	D_a
E		
	E_e	A_a B_c
	E_a	B_c

	E_b	B_b
D		
	D_g	C_a
	D_f	C_a
	D_a	E_g

Creación de Máquinas Virtuales

Se deben crear 7 Máquinas Virtuales, 5 que simularán los Edificios, dentro de los cuales estarán creados los Enrutadores de Conexión. Y 2 que simularán Computadoras Cliente las cuales se comunicarán entre sí, teniendo definida una dirección IP en la red.



Especificaciones Técnicas de Cada Máquina Virtual

Para cada máquina virtual se eligió la siguiente configuración:

- Memoria RAM: 1GB
- Almacenamiento Interno: 5GB
- CPU's: 2

Configuración de Puentes en la Máquina Madre

Para realizar la conexión entre los Edificios es necesaria la creación de Puentes de Red en la Máquina Madre, estos serán los encargados de Comunicar los Edificios entre sí, así como los Puentes de Conexión para la Red Interna de los Clientes. La Configuración de los mismos se hará dentro de cada Máquina Virtual.

Script de Creación:

```
#!/bin/bash -ex

#Instalacion de Dependencias
sudo apt-get install net-tools
sudo apt-get install iw wireless-tools
#sudo apt-get install hostapd
sudo apt-get install bridge-utils
#sudo apt-get install rfkill
#sudo apt-get install resolveconf
#sudo apt-get install iptables

#Creacion de Puentes de Conexion entre Switches
sudo brctl addbr puenteA_B
sudo brctl addbr puenteB_C
sudo brctl addbr puenteC_E
sudo brctl addbr puenteE_D

#Creacion de Puentes de Conexion para la Red Interna
sudo brctl addbr redInternaA
sudo brctl addbr redInternaB
sudo brctl addbr redInternaC
sudo brctl addbr redInternaE
sudo brctl addbr redInternaD

#Levantado de Puentes de Conexion entre Switches
sudo ip link set dev puenteA_B up
sudo ip link set dev puenteB_C up
sudo ip link set dev puenteC_E up
sudo ip link set dev puenteE_D up

#Levantado de Puentes de Conexion para la Red Interna
sudo ip link set dev redInternaA up
sudo ip link set dev redInternaB up
sudo ip link set dev redInternaC up
sudo ip link set dev redInternaE up
```



```
sudo ip link set dev redInternalD up
```

```

5: puenteA_B: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether be:be:1b:41:82:9b brd ff:ff:ff:ff:ff:ff
6: puenteB_C: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 3a:17:e1:5b:f5:75 brd ff:ff:ff:ff:ff:ff state UNKNOWN group default
7: puenteC_E: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether f6:37:d9:34:b3:ee brd ff:ff:ff:ff:ff:ff
8: puenteE_D: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 6e:0a:e9:f0:0c:3b brd ff:ff:ff:ff:ff:ff
26: redInternaA: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 96:ea:3a:96:a0:e3 brd ff:ff:ff:ff:ff:ff
27: redInternaB: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether de:41:2f:4d:89:cd brd ff:ff:ff:ff:ff:ff
28: redInternaC: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether f3:62:ce:c8:53:6d brd ff:ff:ff:ff:ff:ff
29: redInternaE: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 76:a1:3a:85:f3:39 brd ff:ff:ff:ff:ff:ff
30: redInternaD: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 9e:24:5f:65:55:ad brd ff:ff:ff:ff:ff:ff

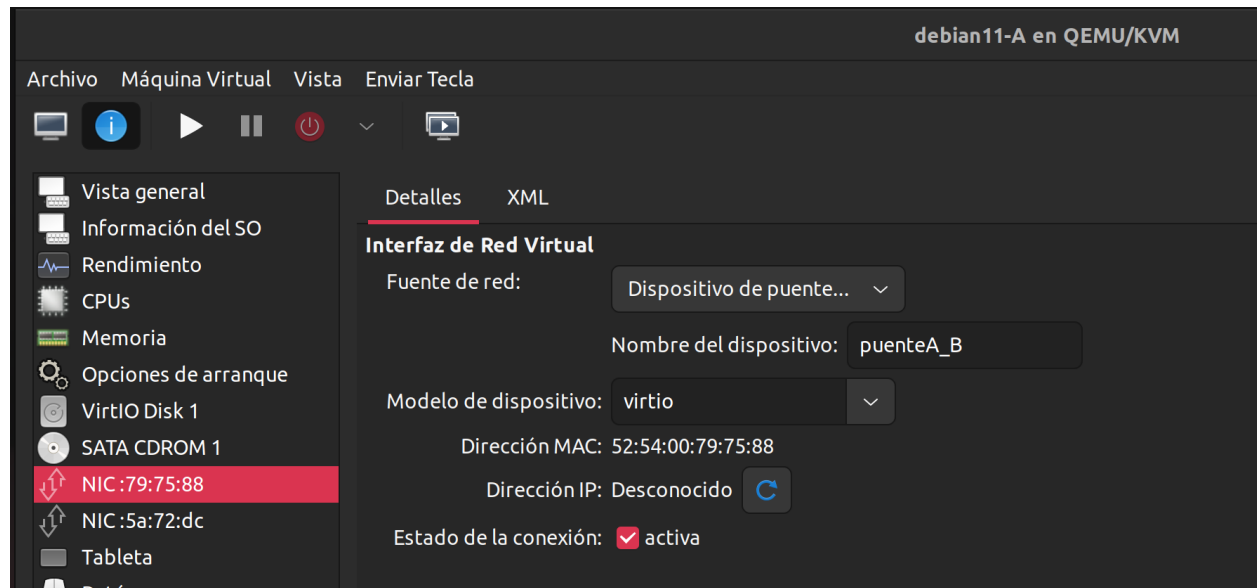
```

Creación de Enrutadores en cada Máquina Virtual

Se realiza por medio de un puente de red entre la conexión Ethernet y la conexión inalámbrica.

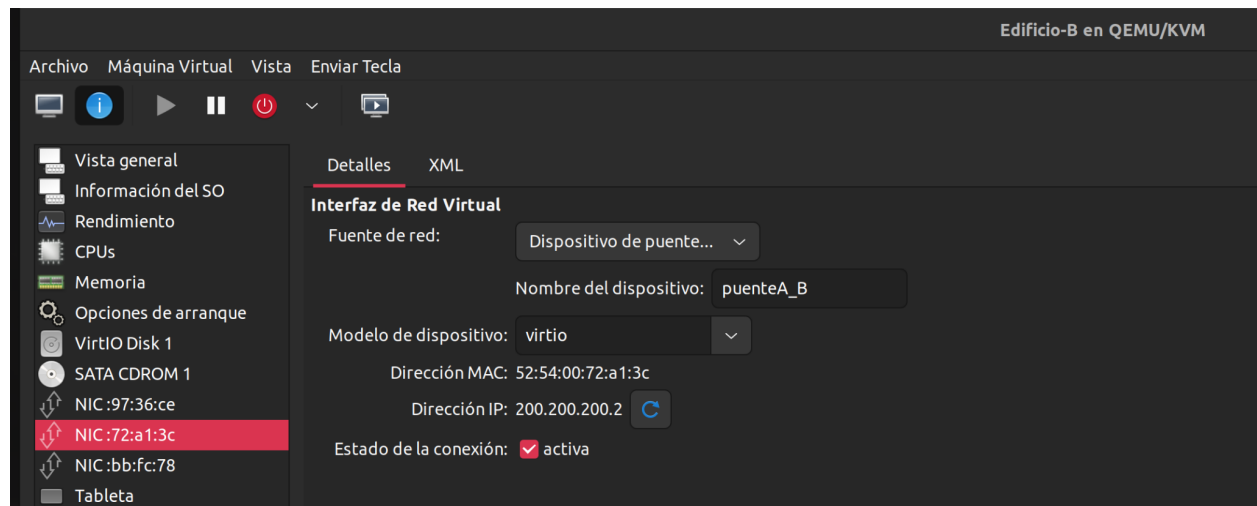
Desde la Aplicación de VMM creamos un nuevo puente, el cual debe coincidir con el nombre del puente creado en la Máquina Madre. En caso de las Máquinas A y D, sólo se debe crear el Puente Emisor y Receptor respectivamente. Este puente es el utilizado para la comunicación entre edificios, pero también se debe crear uno para la comunicación entre Clientes.

Emisor de A a B

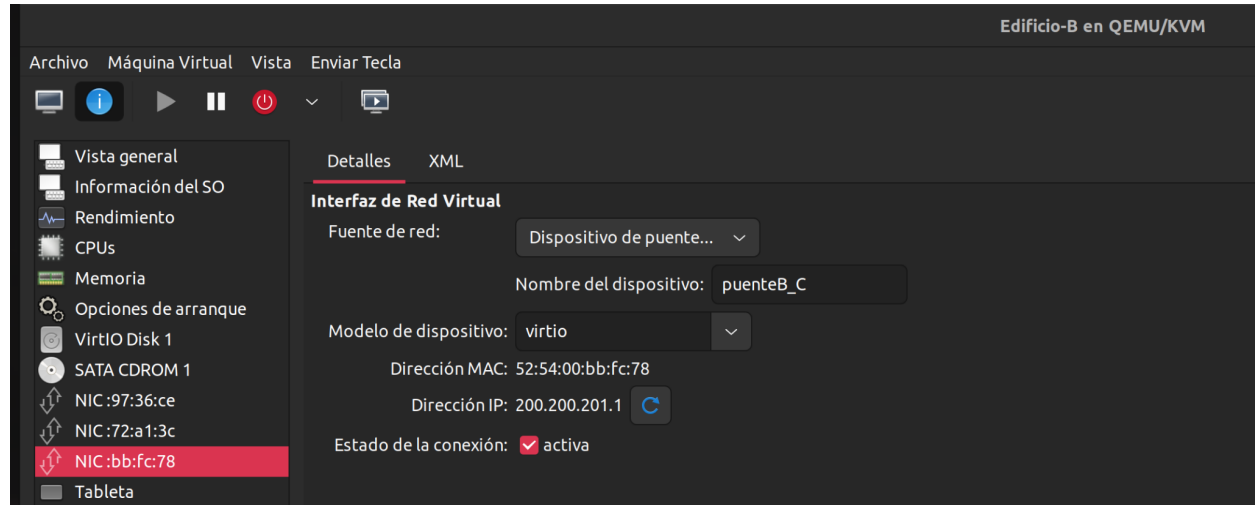


Pero en el Caso de los Edificios Internos, se necesitan dos puentes, el Emisor y el Receptor:

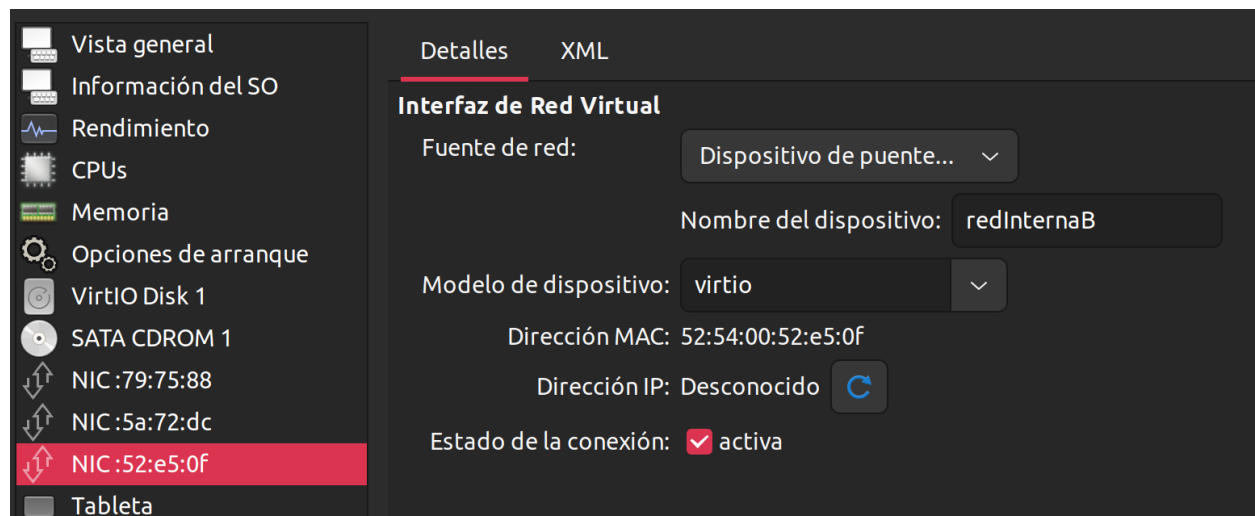
Receptor de A a B



Emisor de B a C

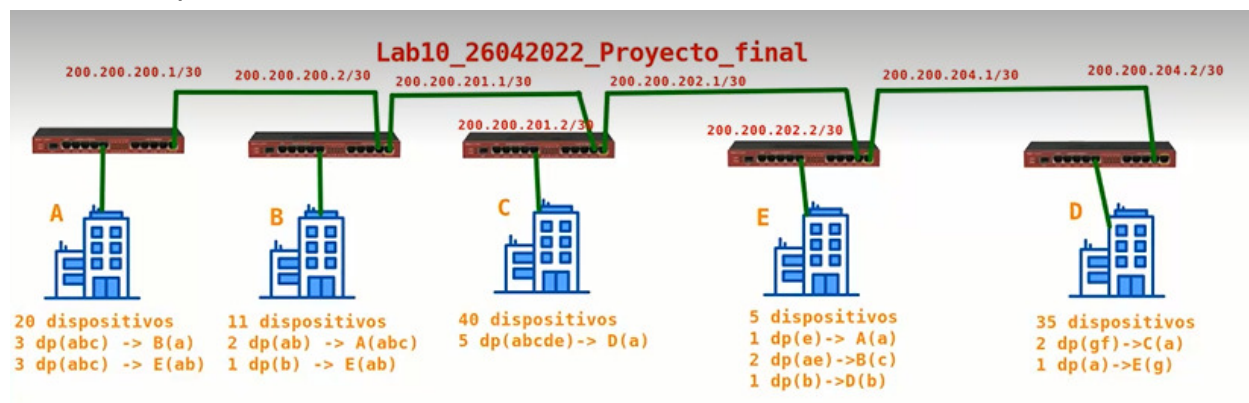


Puente de Red Interna B



Configuración del Enrutador en cada Edificio

Luego de crear los Enrutadores, es necesario configurarlos en base a la configuración establecida por el docente.



Estructura de los Enrutadores

Los enrutadores tendrán dos modos, definido por la dirección IP y su ubicación.

Enrutador	Modo de Conexión	Dirección IP
Puente del Edificio A al Edificio B	Emisor	200.200.200.1/30
Puente del Edificio B al Edificio A	Receptor	200.200.200.2/30
Puente del Edificio B al Edificio C	Emisor	200.200.201.1/30
Puente del Edificio C al Edificio B	Receptor	200.200.201.2/30
Puente del Edificio C al Edificio E	Emisor	200.200.202.1/30
Puente del Edificio E al Edificio C	Receptor	200.200.202.2/30
Puente del Edificio E al Edificio D	Emisor	200.200.204.1/30
Puente del Edificio D al Edificio E	Receptor	200.200.204.2/30

Sin embargo es importante recalcar que los Edificios podrán comunicarse con cualquier otro Edificio por medio de los Enrutadores. Para esto se modificó la información de

cada interfaz de Red en cada Máquina virtual, la cual se ubica en la ruta: /etc/network/interfaces; utilizando la siguiente configuración:

Enrutador del Edificio A

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.25
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion de Bridge A_B desde el Edificio A (Emisor)
auto enp7s0
iface enp7s0 inet static
    address 200.200.200.1/30

#Configuracion desde el Edificio A (Emisor)
#Conexion del Edificio A al Edificio C
post-up ip r replace 200.200.201.0/30 via 200.200.200.2
pre-down ip r delete 200.200.201.0/30 via 200.200.200.2 || true
#Conexion del Edificio A al Edificio E
post-up ip r replace 200.200.202.0/30 via 200.200.200.2
pre-down ip r delete 200.200.202.0/30 via 200.200.200.2 || true
#Conexion del Edificio A al Edificio D
post-up ip r replace 200.200.204.0/30 via 200.200.200.2
pre-down ip r delete 200.200.204.0/30 via 200.200.200.2 || true

#Configuracion de la Red Interna A para el Cliente
auto enp8s0
iface enp8s0 inet static
    address 10.10.40.1/27
    #Red Interna B
post-up ip r replace 10.10.41.0/28 via 200.200.200.2
pre-down ip r delete 10.10.41.0/28 via 200.200.200.2 || true
#Red Interna E
```

```
post-up ip r replace 10.10.43.0/28 via 200.200.200.2
pre-down ip r delete 10.10.43.0/28 via 200.200.200.2 || true
```

Enrutador del Edificio B

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.26
    netmask 255.255.255.0
    gateway 192.168.122.1

#Modo Receptor
#Configuracion de Bridge A_B desde el Edificio B (Receptor)
auto enp7s0
iface enp7s0 inet static
    address 200.200.200.2/30

#Modo Emisor
#Configuracion de Bridge B_C desde el Edificio B (Emisor)
auto enp8s0
iface enp8s0 inet static
    address 200.200.201.1/30

#Conexion Edificio B al Edificio E
post-up ip r replace 200.200.202.0/30 via 200.200.201.2
pre-down ip r delete 200.200.202.0/30 via 200.200.201.2 || true
#Conexion del Edificio B al Edificio D
post-up ip r replace 200.200.204.0/30 via 200.200.201.2
pre-down ip r delete 200.200.204.0/30 via 200.200.201.2 || true

#Configuracion de la Red Interna B para el Cliente
auto enp9s0
iface enp9s0 inet static
    address 10.10.41.1/28
```

```

#Red Interna A
post-up ip r replace 10.10.40.0/27 via 200.200.200.1
pre-down ip r delete 10.10.40.0/27 via 200.200.200.1 || true

#Red Interna C
post-up ip r replace 10.10.42.0/26 via 200.200.201.1
pre-down ip r delete 10.10.42.0/26 via 200.200.201.1 || true

#Red Interna E
post-up ip r replace 10.10.43.0/28 via 200.200.201.2
pre-down ip r delete 10.10.43.0/28 via 200.200.201.2 || true

```

Enrutador del Edificio C

```

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.27
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion de Bridge B_C desde el Edificio C (Receptor)
auto enp7s0
iface enp7s0 inet static
    address 200.200.201.2/30

#Conexion Edificio C al Edificio A
post-up ip r replace 200.200.200.0/30 via 200.200.201.1
pre-dwon ip r delete 200.200.200.0/30 via 200.200.201.1 || true

#Configuracion de Bridge C_E desde el Edificio C (Emisor)
auto enp8s0
iface enp8s0 inet static
    address 200.200.202.1/30

#Conexion Edificio C al Edificio D
post-up ip r replace 200.200.204.0/30 via 200.200.202.2
pre-dwon ip r delete 200.200.204.0/30 via 200.200.202.2 || true

```

```

#Configuracion de la Red Interna C para el Cliente
auto enp9s0
iface enp9s0 inet static
    address 10.10.42.1/26
    #Red Interna A
    post-up ip r replace 10.10.40.0/27 via 200.200.201.1
    pre-down ip r delete 10.10.40.0/27 via 200.200.201.1 || true
    #Red Interna B
    post-up ip r replace 10.10.41.0/28 via 200.200.201.1
    pre-down ip r delete 10.10.41.0/28 via 200.200.201.1 || true
    #Red Interna E
    post-up ip r replace 10.10.43.0/28 via 200.200.202.2
    pre-down ip r delete 10.10.43.0/28 via 200.200.202.2 || true
    #Red Interna D
    post-up ip r replace 10.10.44.0/26 via 200.200.202.2
    pre-down ip r delete 10.10.44.0/26 via 200.200.202.2 || true

```

Enrutador del Edificio E

```

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.28
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion de Bridge C_E desde el Edificio E (Receptor)
auto enp7s0
iface enp7s0 inet static
    address 200.200.202.2/30

#Conexion de Edificio E al Edificio B
post-up ip r replace 200.200.201.0/30 via 200.200.202.1
post-up ip r replace 200.200.201.0/30 via 200.200.202.1 || true
#Conexion de Edificio E al Edificio A

```



```

post-up ip r replace 200.200.200.0/30 via 200.200.202.1
post-up ip r replace 200.200.200.0/30 via 200.200.202.1 || true

#Configuracion de Bridge E_D desde el Edificio E (Emisor)
auto enp8s0
iface enp8s0 inet static
    address 200.200.204.1/30

#Configuracion de la Red Interna E para el Cliente
auto enp9s0
iface enp9s0 inet static
    address 10.10.43.1/28
    #Red Interna A
    post-up ip r replace 10.10.40.0/27 via 200.200.202.1
    pre-down ip r delete 10.10.40.0/27 via 200.200.202.1 || true
    #Red Interna B
    post-up ip r replace 10.10.41.0/28 via 200.200.202.1
    pre-down ip r delete 10.10.41.0/28 via 200.200.202.1 || true
    #Red Interna C
    post-up ip r replace 10.10.42.0/26 via 200.200.202.1
    pre-down ip r delete 10.10.42.0/26 via 200.200.202.1 || true
    #Red Interna D
    post-up ip r replace 10.10.44.0/26 via 200.200.204.2
    pre-down ip r delete 10.10.44.0/26 via 200.200.204.2 || true

```

Enrutador del Edificio D

```

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.29
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion de Bridge E_D desde el Edificio D (Receptor)

```

```

auto enp7s0
iface enp7s0 inet static
    address 200.200.204.2/30

#Conexion Edificio D al Edificio C
post-up ip r replace 200.200.202.0/30 via 200.200.204.1
pre-down ip r delete 200.200.202.0/30 via 200.200.204.1 || true
#conexion Edificio D al Edificio B
post-up ip r replace 200.200.201.0/30 via 200.200.204.1
pre-down ip r delete 200.200.201.0/30 via 200.200.204.1 || true
#Conexion Edificio D al Edificio A
post-up ip r replace 200.200.200.0/30 via 200.200.204.1
pre-down ip r delete 200.200.200.0/30 via 200.200.204.1 || true

#Configuracion de la Red Interna D para el Cliente
auto enp8s0
iface enp8s0 inet static
    address 10.10.43.1/28
#Red Interna E
post-up ip r replace 10.10.43.0/28 via 200.200.204.1
pre-down ip r delete 10.10.43.0/28 via 200.200.204.1 || true
#Red Interna C
post-up ip r replace 10.10.42.0/26 via 200.200.204.1
pre-down ip r delete 10.10.42.0/26 via 200.200.204.1 || true

```

Uso de SSH

Es importante recalcar que la interfaz “enp1s0” que se repite en cada configuración es utilizada para el servicio de SSH, para facilitar el acceso a cada máquina virtual desde la terminal de la Máquina Madre, de la siguiente manera:

```

erikssonherlo@erikssonherlo ~$ ssh debian@192.168.122.25
debian@192.168.122.25's password:
Linux debian 5.10.0-14-amd64 #1 SMP Debian 5.10.113-1 (2022-04-29) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue May 10 18:44:58 2022 from 192.168.122.1
debian@debian:~$ uname
Linux
debian@debian:~$ whoami
debian
debian@debian:~$

```

Por la dirección IP “192.168.122.25”, podemos conocer que estamos conectados al Edificio A, pero podemos conectarnos a todos los edificios para facilitar el acceso.

Enrutamiento

Para la conexión entre el Cliente 1 y el Cliente 2, es necesario definir una tabla de enrutamiento, la cual consiste en la definición del espacio requerido para cada Edificio, los usuarios que tendrán acceso y la dirección IP asignada.

Teniendo las ip's disponibles 10.10.40.00 hacia 10.10.49.00, realizamos la siguiente asignación de ip.

Dispositivos Requeridos por cada Edificio

Las Redes de cada Edificio requieren de los siguientes espacios, ordenados de forma descendente:

- C: 40 dispositivos
- D: 35 dispositivos
- A: 20 dispositivos
- B: 11 dispositivos
- E: 5 dispositivos

Rangos de IP resultantes

Los rangos de red asignados son:

Red	IPv4 Inicio	IPv4 Fin	Dispositivos Disponibles	Dispositivos Requeridos	Red WAN
10.10.40.0/27	10.10.40.0	10.10.40.29	30	20	A
10.10.41.0/28	10.10.41.0	10.10.41.13	14	11	B
10.10.42.0/26	10.10.42.0	10.10.42.61	62	40	C
10.10.43.0/28	10.10.43.0	10.10.43.13	14	5	E
10.10.44.0/26	10.10.44.0	10.10.44.61	62	35	D

Asignación de IP estática a los Dispositivos

Realizamos la asignación de IP para cada Dispositivo de cada Edificio

Edificio	Dispositivo	IP Asignada
A		10.10.40.1
	A_a	10.10.40.2
	A_b	10.10.40.3
	A_c	10.10.40.4
B		10.10.41.1
	B_a	10.10.41.2
	B_b	10.10.41.3
C		10.10.42.1
	C_a	10.10.42.2
	C_b	10.10.42.3
	C_c	10.10.42.4
	C_d	10.10.42.5
	C_e	10.10.42.6
E		10.10.43.1
	E_e	10.10.43.2
	E_a	10.10.43.3
	E_b	10.10.43.4
D		10.10.44.1
	D_g	10.10.44.2
	D_f	10.10.44.3
	D_a	10.10.44.4

Asignación de IP estática para el Servicio de SSH

Realizamos la asignación de IP para cada Máquina virtual, para poder utilizar el servicio de SSH en la terminal de nuestra Maquina Madre.

Máquina Virtual	IP Asignada
Edificio A	192.168.122.25
Edificio B	192.168.122.26
Edificio C	192.168.122.27
Edificio E	192.168.122.28
Edificio D	192.168.122.29
Cliente 1	192.168.122.30
Cliente 2	192.168.122.31

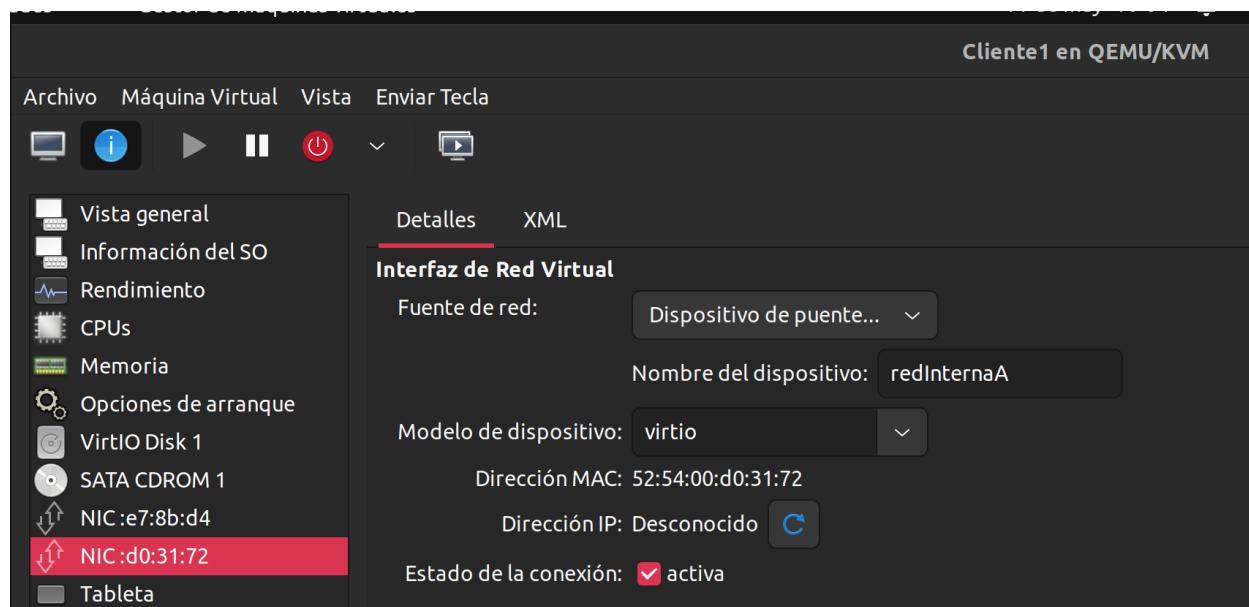
Clientes

Para la comunicación entre Clientes, es necesario crear 2 máquinas virtuales (Cliente1 y Cliente2), cada uno podrá tener un rol diferente a lo largo de la presentación, pero siempre desde Clientes que se puedan comunicar entre sí. La configuración por cliente es la siguiente:

Cliente 1

Para cada Cliente, en nuestra máquina virtual debemos crear un Enrutador, el cual estará conectado a el Puente de Red Interna Respectivamente.

Enrutador Red Interna A



De igual forma, la configuración de la Interfaz es la siguiente:

Interfaz (Cliente A_a)

```
GNU nano 5.4 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

#source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

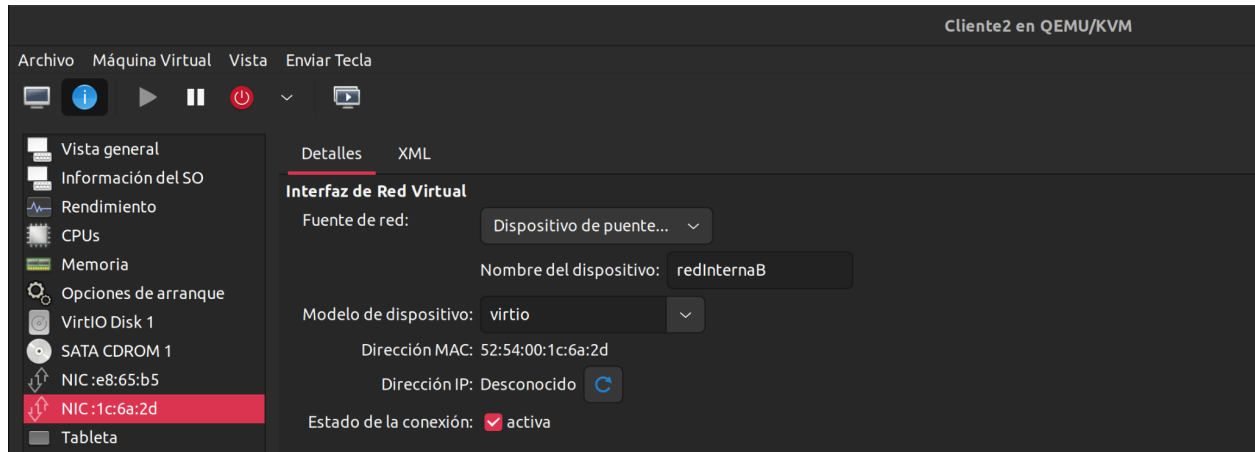
#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.30
    netmask 255.255.255.0
    gateway 192.168.122.1

auto enp7s0
iface enp7s0 inet static
    address 10.10.40.2/27
#Conexion hacia la Red Interna B
post-up ip r replace 10.10.41.0/28 via 10.10.40.1
pre-down ip r delete 10.10.41.0/28 via 10.10.40.1
#Conexion hacia la Red Interna E
post-up ip r replace 10.10.43.0/28 via 10.10.40.1
pre-down ip r delete 10.10.43.0/28 via 10.10.40.1
```

Cliente 2

Para cada Cliente, en nuestra máquina virtual debemos crear un Enrutador, el cual estará conectado a el Puente de Red Interna Respectivamente.

Enrutador Red Interna B



Interfaz (Cliente B_a)

```
GNU nano 5.4 /etc/network/interfaces *
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

#source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuración de Cliente B_a
auto enp7s0
iface enp7s0 inet static
    address 10.10.41.2/28
    #Conexion hacia la Red Interna A
    post-up ip r replace 10.10.40.0/27 via 10.10.41.1
    pre-down ip r delete 10.10.40.0/27 via 10.10.41.1
```

Podemos observar que en ambas interfaces está definida la IP estática en base al tipo de Cliente con el que estamos realizando las pruebas. Dichas IP's fueron definidas con anterioridad en la sección de Enrutamiento.

Comunicación entre Cliente 1 y Cliente 2

Para esta demostración, se está trabajando con el Cliente 1 tomando el rol de Cliente a del Edificio A (Cliente A_a), mientras que el Cliente 2 esta tomando el rol de Cliente a del Edificio B (Cliente B_a).

Ping desde A_a hacia B_a

```
debian@debian:~$ ip -c a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 52:54:00:e7:8b:d4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.30/24 brd 192.168.122.255 scope global enp1s0
        valid_lft forever preferred_lft forever
    inet6 fe80::5054:ff:fee7:8bd4/64 scope link
        valid_lft forever preferred_lft forever
3: enp7s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 52:54:00:d0:31:72 brd ff:ff:ff:ff:ff:ff
    inet 10.10.40.2/27 brd 10.10.40.31 scope global enp7s0
        valid_lft forever preferred_lft forever
    inet6 fe80::5054:ff:fed0:3172/64 scope link
        valid_lft forever preferred_lft forever
debian@debian:~$ ping -c 5 10.10.41.2
PING 10.10.41.2 (10.10.41.2) 56(84) bytes of data.
64 bytes from 10.10.41.2: icmp_seq=1 ttl=62 time=2.49 ms
64 bytes from 10.10.41.2: icmp_seq=2 ttl=62 time=2.35 ms
64 bytes from 10.10.41.2: icmp_seq=3 ttl=62 time=1.88 ms
64 bytes from 10.10.41.2: icmp_seq=4 ttl=62 time=2.21 ms
64 bytes from 10.10.41.2: icmp_seq=5 ttl=62 time=2.24 ms

--- 10.10.41.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 1.878/2.232/2.489/0.202 ms
debian@debian:~$
```

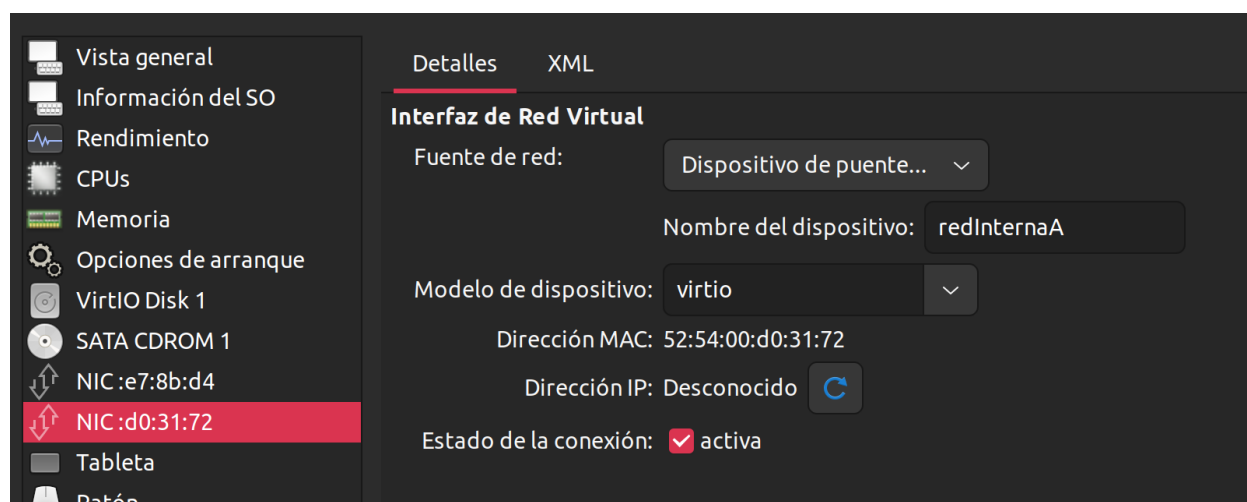
Ping desde B_a hacia A_a

```
debian@debian:~$ ip -c a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 52:54:00:e8:65:b5 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.31/24 brd 192.168.122.255 scope global enp1s0
        valid_lft forever preferred_lft forever
    inet6 fe80::5054:ff:fe8:65b5/64 scope link
        valid_lft forever preferred_lft forever
3: enp7s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 52:54:00:1c:6a:2d brd ff:ff:ff:ff:ff:ff
    inet 10.10.41.2/28 brd 10.10.41.15 scope global enp7s0
        valid_lft forever preferred_lft forever
    inet6 fe80::5054:ff:fe1c:6a2d/64 scope link
        valid_lft forever preferred_lft forever
debian@debian:~$ ping -c 5 10.10.40.2
PING 10.10.40.2 (10.10.40.2) 56(84) bytes of data.
64 bytes from 10.10.40.2: icmp_seq=1 ttl=62 time=2.00 ms
64 bytes from 10.10.40.2: icmp_seq=2 ttl=62 time=1.98 ms
64 bytes from 10.10.40.2: icmp_seq=3 ttl=62 time=2.00 ms
64 bytes from 10.10.40.2: icmp_seq=4 ttl=62 time=1.99 ms
64 bytes from 10.10.40.2: icmp_seq=5 ttl=62 time=2.41 ms

--- 10.10.40.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 1.978/2.073/2.405/0.165 ms
debian@debian:~$
```

Configuración de cada Cliente

En cada máquina virtual de Cliente se encontrarán los scripts para el rol de Dispositivo que deseamos utilizar. Para ello, deberá antes establecer la Red Interna (LAN) a la que se desea conectar:



Luego desde la máquina virtual debe correr el script de configuración por dispositivo, se encuentran divididos en Carpetas por Edificios, y tienen un formato:

```
sudo bash NombreEdificio_NombreDispositivo.sh
```

Ejemplo:

```
sudo bash A_a.sh
```

Edificio A

Dispositivo A_a

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.30
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.40.2/27
```

```
#Conexion hacia la Red Interna B
post-up ip r replace 10.10.41.0/28 via 10.10.40.1
pre-down ip r delete 10.10.41.0/28 via 10.10.40.1
#Conexion hacia la Red Interna E
post-up ip r replace 10.10.43.0/28 via 10.10.40.1
pre-down ip r delete 10.10.43.0/28 via 10.10.40.1
EOF

systemctl restart networking
ip -c a
```

Dispositivo A_b

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.30
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.40.3/27
    #Conexion hacia la Red Interna B
    post-up ip r replace 10.10.41.0/28 via 10.10.40.1
    pre-down ip r delete 10.10.41.0/28 via 10.10.40.1
    #Conexion hacia la Red Interna E
    post-up ip r replace 10.10.43.0/28 via 10.10.40.1
    pre-down ip r delete 10.10.43.0/28 via 10.10.40.1
EOF

systemctl restart networking
ip -c a
```

Dispositivo A_c

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enpls0
iface enpls0 inet static
    address 192.168.122.30
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.40.4/27
    #Conexion hacia la Red Interna B
    post-up ip r replace 10.10.41.0/28 via 10.10.40.1
    pre-down ip r delete 10.10.41.0/28 via 10.10.40.1
    #Conexion hacia la Red Interna E
    post-up ip r replace 10.10.43.0/28 via 10.10.40.1
    pre-down ip r delete 10.10.43.0/28 via 10.10.40.1
EOF

systemctl restart networking
ip -c a
```

Edificio B

Dispositivo B_a

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
```

```
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.41.2/28
    #Conexion hacia la Red Interna A
    post-up ip r replace 10.10.40.0/27 via 10.10.41.1
    pre-down ip r delete 10.10.40.0/27 via 10.10.41.1
EOF

systemctl restart networking
ip -c a
```

Dispositivo B_b

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.41.3/28
```

```
#Conexion hacia la Red Interna A
post-up ip r replace 10.10.40.0/27 via 10.10.41.1
pre-down ip r delete 10.10.40.0/27 via 10.10.41.1
#Conexion hacia la Red Interna E
post-up ip r replace 10.10.43.0/28 via 10.10.41.1
pre-down ip r delete 10.10.43.0/28 via 10.10.41.1
EOF

systemctl restart networking
ip -c a
```

Edificio C

Dispositivo C_a

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.42.2/26
    #Conexion hacia la Red Interna D
    post-up ip r replace 10.10.44.0/26 via 10.10.42.1
    pre-down ip r delete 10.10.44.0/26 via 10.10.42.1
EOF

systemctl restart networking
ip -c a
```


Dispositivo C_b

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.42.3/26
    #Conexion hacia la Red Interna D
    post-up ip r replace 10.10.44.0/26 via 10.10.42.1
    pre-down ip r delete 10.10.44.0/26 via 10.10.42.1
EOF

systemctl restart networking
ip -c a
```

Dispositivo C_c

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
```

```
netmask 255.255.255.0
gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.42.4/26
    #Conexion hacia la Red Interna D
    post-up ip r replace 10.10.44.0/26 via 10.10.42.1
    pre-down ip r delete 10.10.44.0/26 via 10.10.42.1
EOF

systemctl restart networking
ip -c a
```

Dispositivo C_d

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enpls0
iface enpls0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.42.5/26
    #Conexion hacia la Red Interna D
    post-up ip r replace 10.10.44.0/26 via 10.10.42.1
    pre-down ip r delete 10.10.44.0/26 via 10.10.42.1
EOF

systemctl restart networking
```

```
ip -c a
```

Dispositivo C_e

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.42.6/26
    #Conexion hacia la Red Interna D
    post-up ip r replace 10.10.44.0/26 via 10.10.42.1
    pre-down ip r delete 10.10.44.0/26 via 10.10.42.1
EOF

systemctl restart networking
ip -c a
```

Edificio E

Dispositivo E_e

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
```

```
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.43.2/28
    #Conexion hacia la Red Interna A
    post-up ip r replace 10.10.40.0/27 via 10.10.43.1
    pre-down ip r delete 10.10.40.0/27 via 10.10.43.1
    #Conexion hacia la Red Interna B
    post-up ip r replace 10.10.41.0/28 via 10.10.43.1
    pre-down ip r delete 10.10.41.0/28 via 10.10.43.1
EOF

systemctl restart networking
ip -c a
```

Dispositivo E_a

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1
```

```
#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.43.3/28
    #Conexion hacia la Red Interna B
    post-up ip r replace 10.10.41.0/28 via 10.10.43.1
    pre-down ip r delete 10.10.41.0/28 via 10.10.43.1
EOF

systemctl restart networking
ip -c a
```

Dispositivo E_b

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.43.4/28
    #Conexion hacia la Red Interna D
    post-up ip r replace 10.10.44.0/26 via 10.10.43.1
    pre-down ip r delete 10.10.44.0/26 via 10.10.43.1
EOF

systemctl restart networking
ip -c a
```

Edificio D

Dispositivo D_g

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.44.2/26
    #Conexion hacia la Red Interna C
    post-up ip r replace 10.10.42.0/26 via 10.10.44.1
    pre-down ip r delete 10.10.42.0/26 via 10.10.44.1
EOF

systemctl restart networking
ip -c a
```

Dispositivo D_f

```
#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
```

```

auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.44.3/26
    #Conexion hacia la Red Interna C
    post-up ip r replace 10.10.42.0/26 via 10.10.44.1
    pre-down ip r delete 10.10.42.0/26 via 10.10.44.1
EOF

systemctl restart networking
ip -c a

```

Dispositivo D_a

```

#!/bin/bash -ex

#Configuracion del Cliente
cat <<EOF > /etc/network/interfaces
# The loopback network interface
auto lo
iface lo inet loopback

#Configuracion de SSH
auto enp1s0
iface enp1s0 inet static
    address 192.168.122.31
    netmask 255.255.255.0
    gateway 192.168.122.1

#Configuracion del Cliente
auto enp7s0
iface enp7s0 inet static
    address 10.10.44.4/26
    #Conexion hacia la Red Interna E
    post-up ip r replace 10.10.43.0/28 via 10.10.44.1
    pre-down ip r delete 10.10.43.0/28 via 10.10.44.1

```

EOF

```
systemctl restart networking
```

```
ip -c a
```