# Trabajo práctico sobre Netkit, STP, Port Bonding y VLAN

Mauro Bachur, Francisco Elías, y Camilo Jimenez

Mayo de 2019

# 1 Introducción teórica

En este trabajo práctico tenemos como objetivo probar cómo funciona el bridge/swtich, el protocolo stp, el vlan y el port bonding, a traves de la utilización de la herramienta de creación de dispositivos virtuales llamado netkit.

## 1.1 Netkit

Netkit es una herramienta que permite la creación de dispositivos de red virtuales donde podemos realizar diferentes experimentos.

## 1.2 Bridge

Un brigde es un dispositivo de interconexión de redes de computadoras que opera en la capa 2 del modelo OSI. Se encarga de interconectar dos segmentos de red haciendo el pasaje de datos de una red a la otra.

## 1.3 Switch

Un switch es un dispositivo de interconexión encargado de unir dos o mas equipos de red entre si para formar una red de area local (LAN) que tiene que seguir el estándar conocido como Ethernet (IEEE 802.3).

## 1.4 STP

El STP es un protocolo que usa un algoritmo denominado "Spanning-tree Algorithm" el cual crea la topologia de la red en donde se aplica. Con esto creado, se procede a la eleccion de un bridge como root, que va a ser elegida a partir del "bridge priority",un numero que posee cada uno que por default va a ser 32,768. Si el valor no es modificado, todos los bridge van a tener el mismo valor, entonces en este caso se procede a elegir con la MAC ID que tiene cada bridge (unica), siendo el de menor MAC ID el que va a ser el bridge root. Una vez establecido el root, se procede a establecer cuales conexiones va a estar bloqueadas y cuales van a estar en funcionamiento. Esto lo hace viendo la mejor conexion para llegar al bridge root y se lo pone en estado de funcionamiento, todas las demas que no se usan para una mejor llegada, son puestas en estado de bloqueo.

Si la configuración de STP cambia, o si un segmento en la red redundante llega a ser inalcanzable, el algoritmo reconfigura los enlaces y restablece la conectividad, activando uno de los enlaces de reserva. Es decir, que el algoritmo vuelve a crear la topologia, vuelve a crear un bridge root, y vuelve a crear conexiones en estado de bloqueo y en estado de funcionamiento.

En resumen: "El STP permanece vigente hasta que ocurre un cambio en la topología, situación que el protocolo es capaz de detectar de forma automática. Cuando ocurre uno de estos cambios, el puente raíz actual redefine la topología del árbol de expansión o se elige un nuevo puente raíz."

### 1.5 Port Bonding

Port bonding es un método para la combinación de multiples conexiones de red en paralelo para mejorar lo que una sola conexion podria hacer. Soluciona los problemas de ethernet tales como limitaciones en el ancho de banda y la carencia de resistencia. Port bonding posee 7 modos distintos, cada uno varía en como el tráfico es enviado a traves de la interfaz creada. Los mismos son:

- Modo 0 (balance-rr)
- Modo 1 (active-backup)
- Modo 2 (balance-xor)
- Modo 3 (broadcast)
- Modo 4 (802.3ad)
- Modo 5 (balance-tlb)
- Modo 6 (balance-alb)

### 1.6 VLAN

Vlan es un método para crear un dominio de difusión dentro de una misma red, por ejemplo tengo 4 PCs y hago una vlan entre 2 PCs y otra vlan entre las otras 2 PCs, entonces habra un dominio de difusión entre las primeras 2 PCs y otro entre las PCs restantes.

### 1.7 Sniffer

Un sniffer, en este contexto, es una PC que se encuentra en modo promiscuo, o sea, que se encuentra conectada a una red compartida y se encarga de capturar todo el tráfico que circula por la misma.

## 2 Experimentos

### Funcionamiento del Netkit

Para armar un laboratorio de Netkit, se debe crear una carpeta con el nombre deseado y crear dentro del la misma los siguientes archivos y carpetas:

- Un archivo lab.conf en donde se introduzca la topologia del laboratorio de forma escrita.
- La misma cantidad de carpetas que de dispositivos queramos. Las mismas deberán tener el mismo nombre que el especificado en el archivo lab.conf.

- De forma opcional, podemos crear archivos .startup para cada dispositivo que se encargan de ejecutar los comandos escritos en el al iniciar el laboratorio.

Una vez cumplidos estos requisitos, basta con escribir el siguiente comando para iniciar el laboratorio:

```
lstart
```

En el caso de que queramos iniciar los dispositivos en forma paralela, se usa el comando:

```
lstart -p
```

Por último, si queremos iniciar solo un dispositivo, basta con escribir:

```
lstart <dispositivo>
```

Cuando terminemos de trabajar con los dispositivos, existen dos comandos para "apagarlos":

```
lcrash
lhalt
```

El primero apaga los dispositivos borrando el historial de comandos en los mismos, por lo que no se mantiene ningún comando usados en los mismo. El segundo apaga los dispositivos de forma que al inciarlos en otro momento, todos los comandos ingresados se mantienen.
Siguiendo este procedimiento, realizamos los experimentos que se encuentran detallados abajo.

Link a GitHub con todos los escenarios: `https://github.com/franelias/netkit.git`

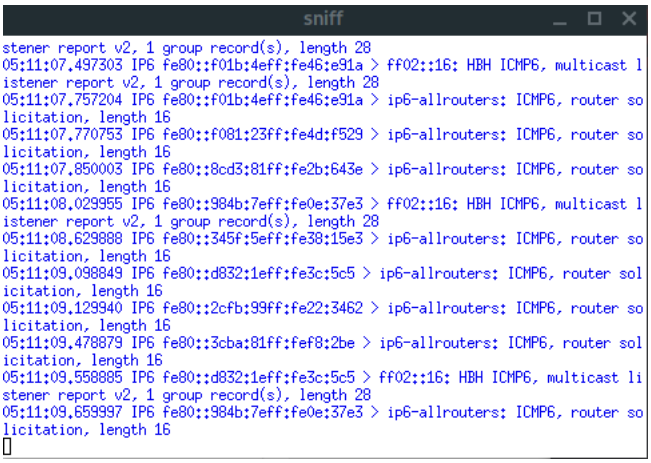
## 2.1 Escenario Bridge/switch

Para este primer experimentos, armamos un laboratorio utilizando 4 PCs (PC1, PC2, PC3, PC4), en el cual cada par está conectado a un cable A y B (PC1 y PC2 al cable A y PC3 y PC4 al cable B). Estos cables están unidos a un switch y a un sniffer. Los .startup del switch contenía los siguientes comandos:

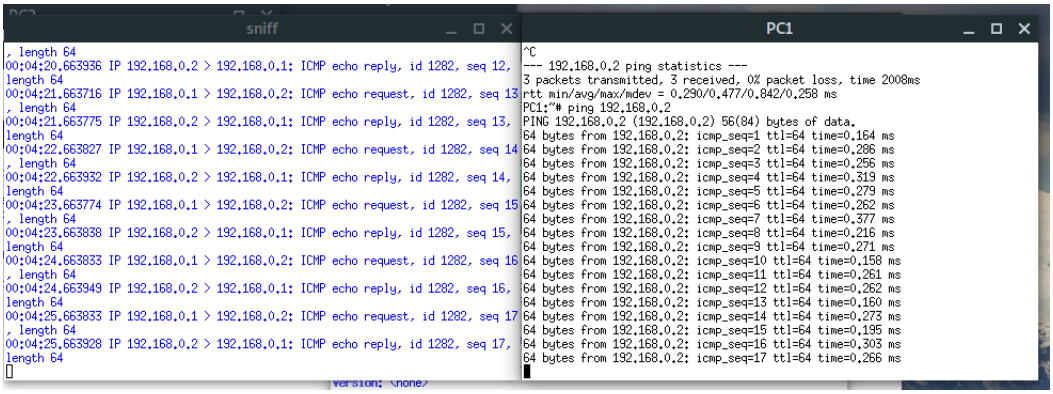
```
brctl addbr br0
brctl addif br0 eth0
brctl addif br0 eth1
ifconfig eth0 up
```

```
ifconfig eth1 up
ifconfig br0 up
```

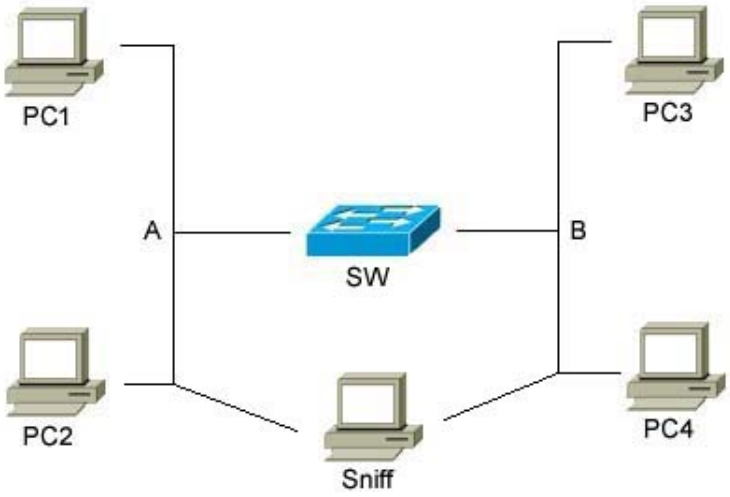
Realizamos este primer experimento para poder aprender como armar un switch y ver como funciona. Para ver esto, pingueamos desde una PC a cualquier otra, y observamos el sniffer, que nos mostraba como el switch enviaba el broadcast a todas las PCs, pero solo le llegaba el paquete a la PC pingueada.




Switch aprendiendo los dispositivos que tiene conectado.



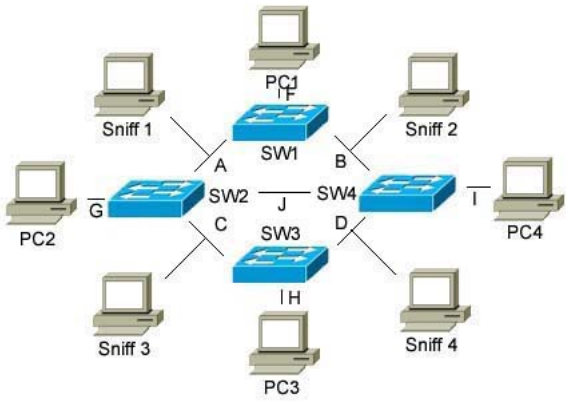

El sniffer pudo captar que las conexiones estaban realizar de forma correcta y el switch creado estaba funcionando de manera adecuada.



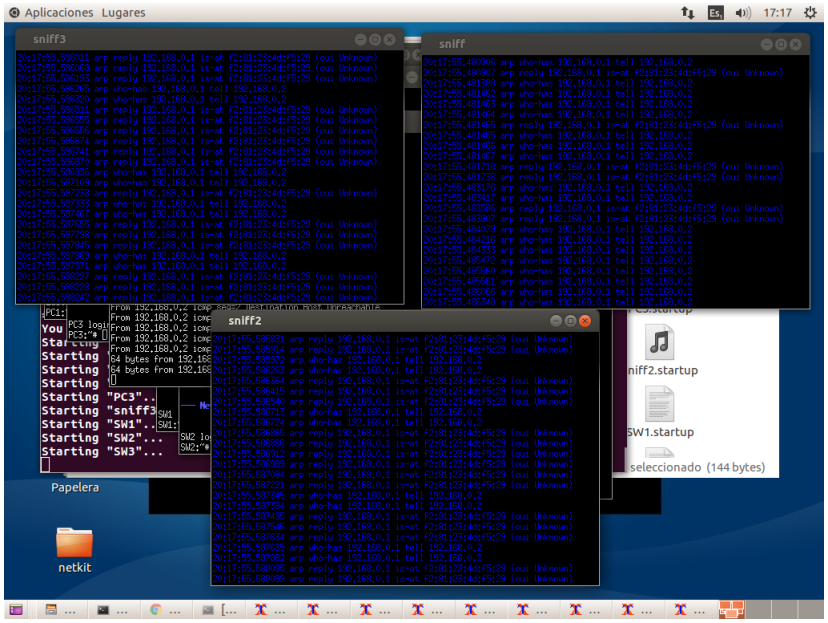

Topologia del laboratorio.

## 2.2 Escenario STP

Utilizamos 4 PCs (PC1, PC2, PC3, PC4) y las conectamos mediante 4 switches (SW1, SW2, SW3, SW4), formando un bucle. Por último, colocamos 4 sniffers, que serán los encargados de registar todo lo que pase por los cables.

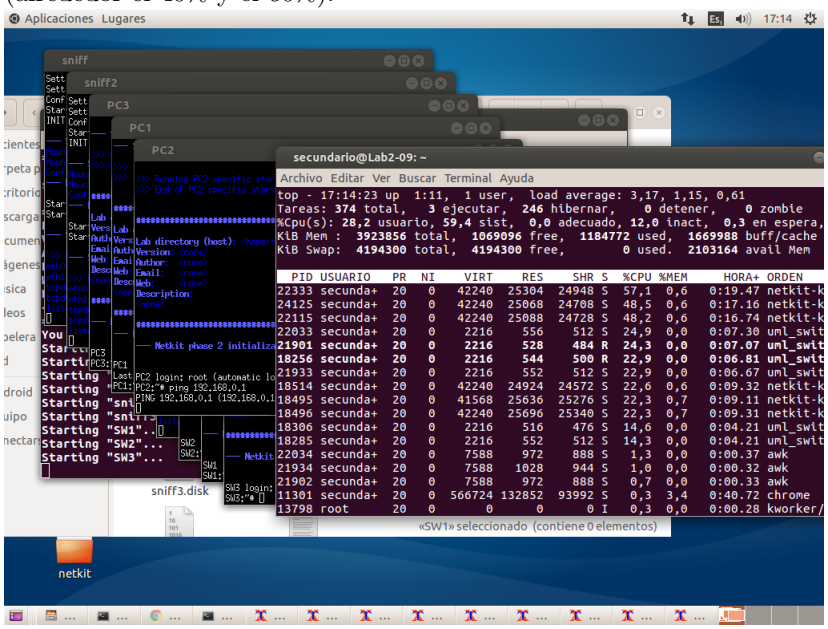



Cuando encendimos el laboratorio y pingueamos la PC2 a través de la PC1, los sniffers mostraron la tormenta de broadcast que se produjo por conectar los switches de esa manera.




Tormenta de broadcast

Paralelamente, otra consola que ejecutaba el comando top mostró que al realizar este comando en la PC1, el uso de la cpu aumentó exponencialmente (alrededor el 40% y el 60%).




Comando top

Esto se puede arreglar activando el protocolo STP con el siguiente procedimiento:

- Apagar el switch:

  `ifconfig br0 down`

- Activar el protocolo STP en el mismo:

  `brctl stp br0 on`

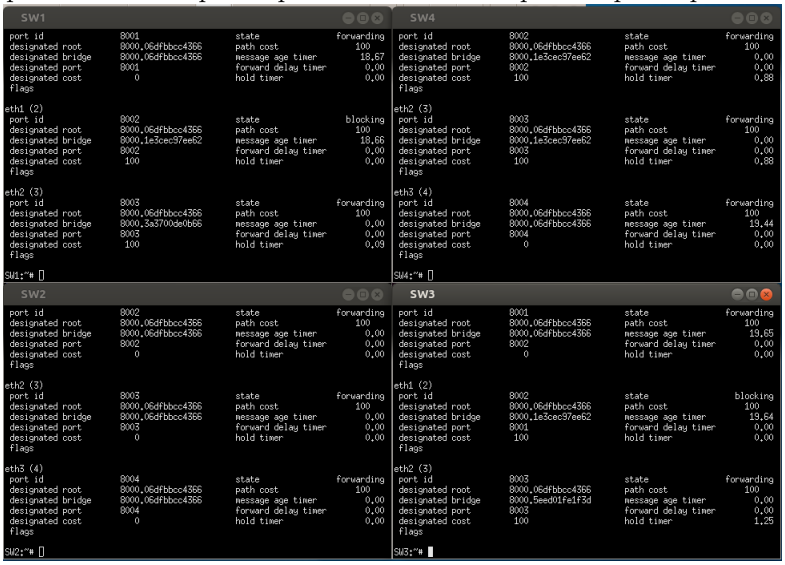
- Activarlo de vuelta:

  `ifconfig br0 up`

Realizado el procedimiento, a traves del comando:

`brctl showstp br0`

vimos como el SW2 fue el seleccionado como root, con la caracteristica adicional de que era el unico que no poseia interfaces bloqueadas por el protocolo.




Comando mostrando las interfaces bloqueadas en cada switch, y el switch root designado

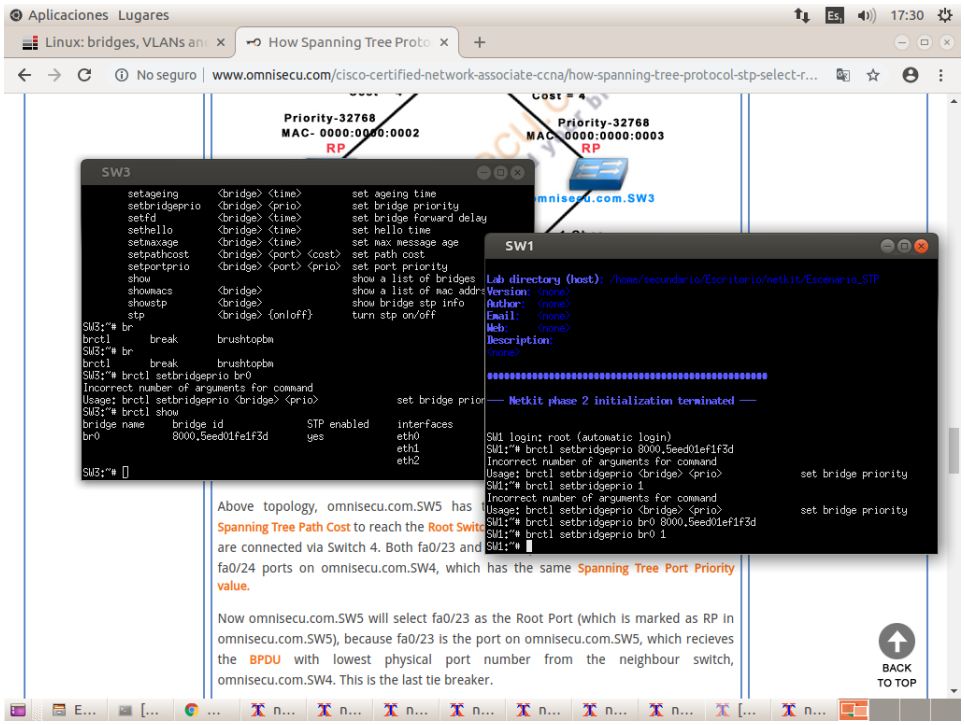
## Qué hace

Evita la tormenta de broadcast asignandole estados a los cables que conectan en las computadoras. Un estado es el de bloqueado, el cual los datos no fluyen por ese cable y va a ser activado (o no) en caso de que se modifique el STP y el otro estado es el de funcionamiento el cual permite los flujos de datos por el mismo. La eleccion de los estados de los cables fue explicado en el punto 1.4

## Cómo establecer un root manualmente

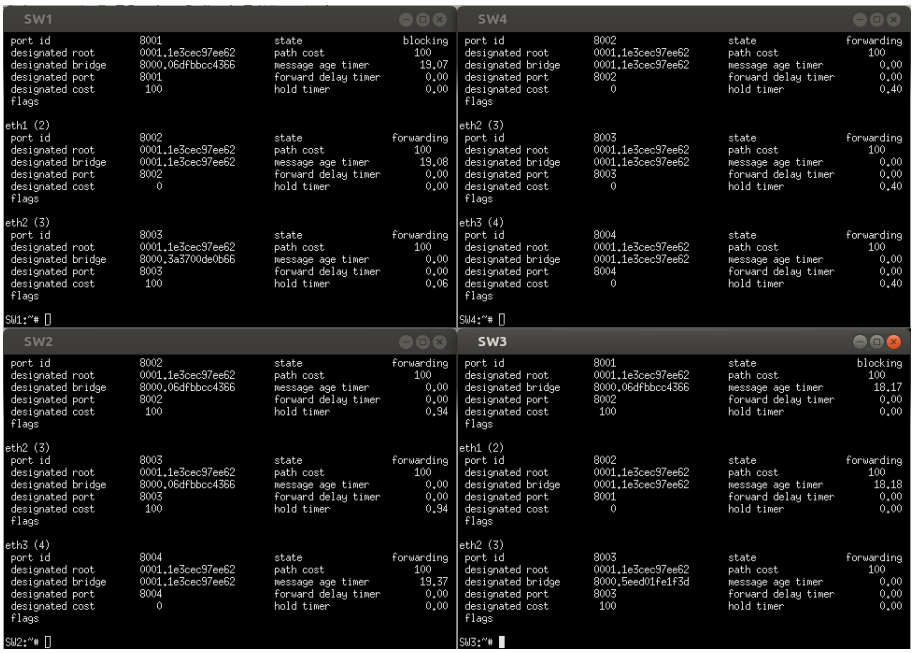
Puede llegar un momento en que nosotros queramos elegir de forma manual un switch que actúe como root, para ello utilizamos el siguiente comando en el switch elegido:

```
brctl setbridgeprio br0 <número>
```




Cambio del número del switch, convitiendolo en el root

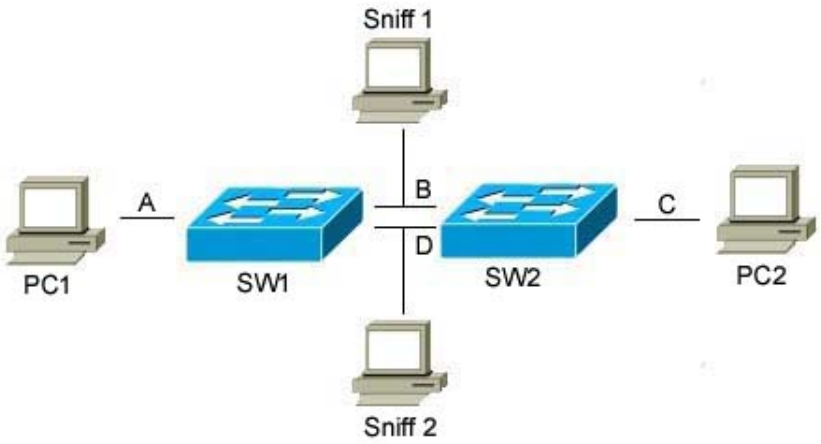
En nuestro laboratorio, cambiamos el numero del SW4 a 1, convirtiendolo en el root, y a traves del comando brctl showstp br0, analizamos como quedo el arbol topologico.




brctl showstp br0 mostrando que interfaces estan bloqueadas.

## 2.3 Port bonding

Para realizar el laboratorio, utilizamos dos PCs, nombradas PC1 y PC2 respectivamente y dos switches unidos entre si por dos cables.

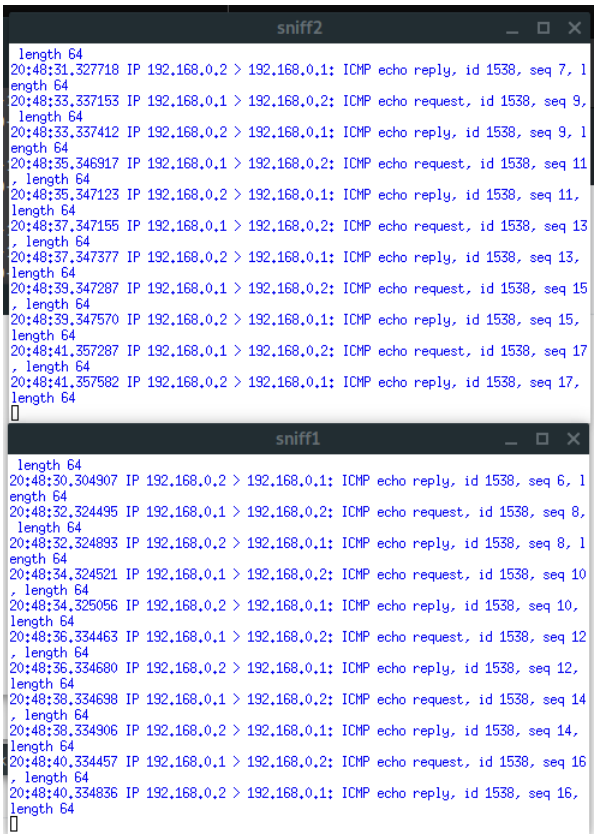



Topologia del laboratorio.

En el archivo .startup de los dos switches utilizamos los siguientes comandos:

```
brctl addbr br0
modprobe bonding
echo balance-rr > /sys/class/net/bond0/bonding/mode
echo +eth0 > /sys/class/net/bond0/bonding/slaves
echo +eth1 > /sys/class/net/bond0/bonding/slaves
brctl addif br0 eth2
brctl addif br0 bond0
ifconfig eth0 up
ifconfig eth1 up
ifconfig eth2 up
ifconfig bond0 up
ifconfig br0 up
```

Estos activan el port bonding en modo “Balance Round Robin” utilizando los dos cables que los conectan entre si. Al pinguear a la PC2 desde la PC1, los sniffers conectados en cada uno de los cables que unían los switches mostraron que los paquetes se alternaban para enviarse, uno iba por el cable B y otro por el D.




Sniffers mostrando los paquetes enviados. Se puede ver que se alternan en por la secuencia en que se envían.

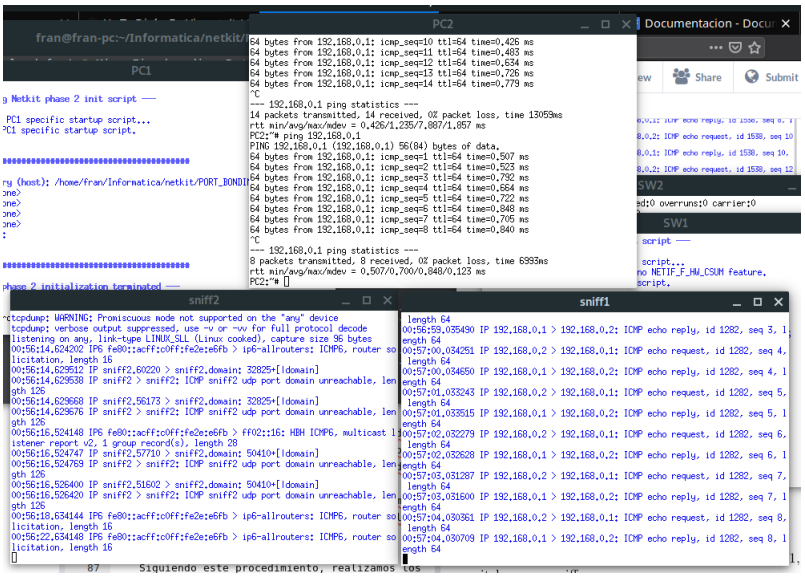
Posteriormente, procedimos a apagar la interfaz eth0 de uno de los switches, que hizo que al pinguear a la otra computadora, solo el 50% de los paquetes lleguen a destino.

Luego, cambiamos el port bonding a modo "active-backup" con el comando:

```
echo active-backup > /sys/class/net/bond0/bonding/mode
```

Observando los sniffers, pudimos ver que los paquetes viajan a través de un solo cable.

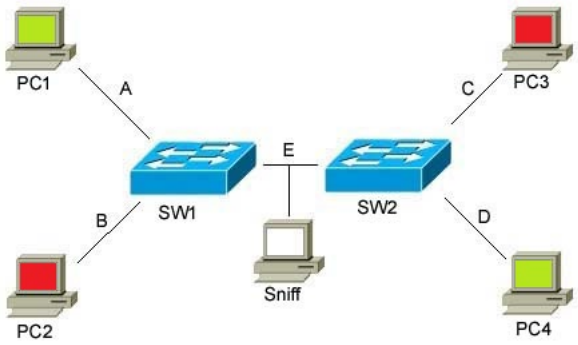
Desgraciadamente, Netkit no nos permite probar el experimento deseado con este modo, tirar abajo uno de los cables, por lo que no pudimos probarlo, pero cuando pasa esto, los paquetes viajan por el otro.




Ping desde la PC2 a la 1. Se observa que los paquetes van a través de un solo cable.

## 2.4 VLAN

Para este caso, utilizamos 4 PCs, denominadas PC1, PC2, PC3 y PC4, dos switches y un sniffer.




Topologia del lab.

En primer lugar, a través de los siguientes comandos en las 4 PCs, establecimos dos VLANs, una con id = 10 (en las PCs 2 y 3) y otra con id = 20 (para las PCs 1 y 4):

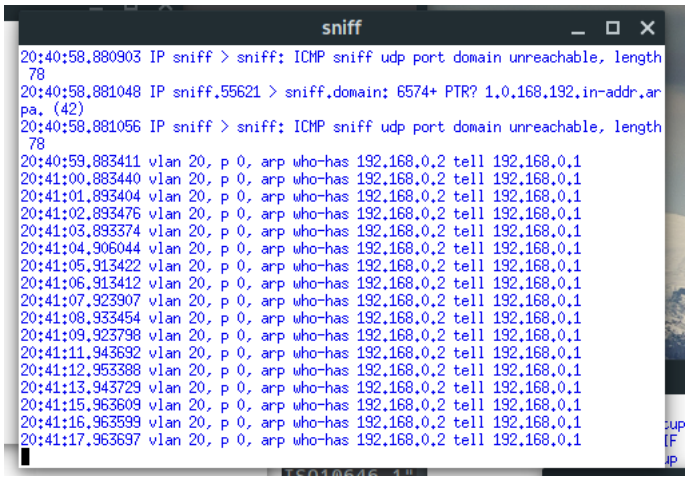
```
Para id = 20
    ifconfig eth0 up
    vconfig add eth0 20
```

```
    ifconfig eth0.20 192.168.0.<Nro de PC>/24 up

Para id = 10
    ifconfig eth0 up
    vconfig add eth0 10
    ifconfig eth0.10 192.168.0.<Nro de PC>/24 up
```

Luego, colocamos los siguientes comandos en el switch, para crear la VLAN en el mismo:

```
brctl addbr br0.10
brctl addbr br0.20

ifconfig eth0 up
ifconfig eth1 up
ifconfig eth2 up
ifconfig eth3 up

vconfig add eth0 20
vconfig add eth1 10
vconfig add eth2 10
vconfig add eth3 20

brctl addif br0.10 eth1.10
brctl addif br0.10 eth2.10
brctl addif br0.20 eth0.20
brctl addif br0.20 eth3.20

ifconfig eth0.20 up
ifconfig eth1.10 up
ifconfig eth2.10 up
ifconfig eth3.20 up

ifconfig br0.10 up
ifconfig br0.20 up
```

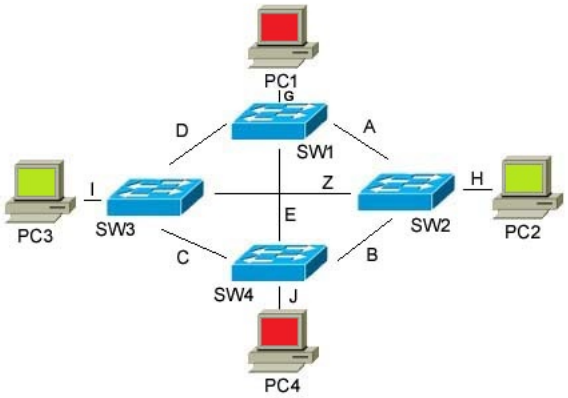
Hecho este paso, probamos pingueando de la PC1 a todas las PCs, con el resultado de que solo la Nro 4 recibe los paquetes. Los sniffers en este procedimiento registraron que los paquetes solo iban por los cables cuya PCs tenian el id de la VLAN buscada.



Ventana de uno de los sniffers.

## 2.5 STP-VLAN

Para este experimento utilizamos 4 PCs y dos switches, donde las PCs 1 y 4 se encontraban en una VLAN, mientras que las 2 y 3 en otra.




Colocamos 4 sniffers en los cables que unen las computadoras con los switches, y realizamos distintas pruebas pingueando desde una PC a la otra. Al igual que en el experimento anterior, los switches derivaban los paquetes solo a las PCs que se encontraban en la misma VLAN.

Analizando còmo estaba funcionando el protocolo STP en el laboratorio, nos dimos cuenta que habian designados dos switches root, que por lo general eran el SW2 y el SW4. Esto se producia debido a como estaba implementada la VLAN en los mismos.

Interfaces bloqueadas para las que poseìan ID de VLAN nro 20


Interfaces bloqueadas para las que poseìan ID de VLAN nro 10

Al igual que en el primer experimento, al cambiar el switch root, cambiaba las interfaces bloqueadas, aunque la VLAN seguia funcionando correctamente en todo momento.

# Fuentes:

- https://camber1redes.wordpress.com/puente-de-red-o-bridge/
- https://www.cloudibee.com/network-bonding-modes/
- http://redestelematicas.com/el-switch-como-funciona-y-sus-principales-caracteristicas/
- https://backdrift.org/manage-linux-bonding-without-ifenslave-using-sysfs
- https://ahelpme.com/linux/how-to-enable-linux-bonding-without-ifenslave/
- https://rm-rf.es/configurar-una-vlan-en-linux-con-vconfig/