

## Práctica 7: Otras tecnologías en Ethernet

### 1- Objetivos

Con esta práctica completamos los temas alrededor de la configuración de equipos con interfaces Ethernet. Para ello veremos cómo emplear 802.1Q desde routers Cisco y PCs Linux. Veremos cómo agregar varios interfaces Ethernet en uno solo lógico. Haremos que un router actúe simultáneamente como un puente y finalmente alteraremos el comportamiento de un conmutador para que nos permita ver todo el tráfico que reenvía por una VLAN.

### 2- Material necesario

- 1 router Cisco
- 2 conmutadores Cisco
- 3 cables rectos
- 2 cables cruzados
- 3 PCs

### 3- Conocimientos previos

- Configuración IP básica de PCs con Linux y de routers Cisco
- VLANs y 802.1Q
- Configuración básica y de VLANs en conmutadores Cisco
- 802.3ad

### 4- 802.1Q en router Cisco

Sobre el interfaz físico Ethernet de un router Cisco se pueden configurar interfaces “virtuales”, en concreto nos interesan en esta práctica los “subinterfaces”. Indica la documentación de Cisco:

*“A subinterface is a mechanism that allows a single physical interface to support multiple logical interfaces or networks. That is, several logical interfaces or networks can be associated with a single hardware interface.”*

Queremos que un interfaz del router pueda enviar tramas ethernet con encapsulado 802.1Q y etiquetas de diferentes VLAN. Si el puerto del conmutador al que está conectado está en “trunk” y acepta tramas de todas esas VLANs entonces podremos conseguir que efectivamente el router tenga un interfaz lógico en cada VLAN.

Conecte el interfaz FastEthernet del router1 a un puerto del switch1. Active ese interfaz (no shutdown) en el router y configure el puerto del switch para que esté en trunk. Cree un par de VLANs en el switch (por ejemplo la VLAN 2 y la 3).

Vamos a crear un subinterfaz en el router para cada VLAN. El subinterfaz representará al interfaz del router en esa LAN. Los subinterfaces se indentifican por un número. Para recordar mejor qué

subinterfaz corresponde a cada VLAN los identificaremos con el número de la VLAN (aunque no es obligatorio). Para entrar en el modo de configuración del subinterfaz 2 deberá usar:

```
Router(config)# interface FastEthernet0/0.2
```

Ahora, en el modo de configuración de ese subinterfaz indique que desea emplear encapsulado 802.1Q y la VLAN 2 (comando `encapsulation`).

Asigne dirección IP a este subinterfaz en una LAN, por ejemplo en la LAN 192.168.1.0/24 (que llamaremos LAN A).

Repita los pasos con el subinterfaz 3, asignándolo a la VLAN 3 y con dirección IP en la LAN B (que debe ser independiente de la LAN A)

Puede ver ahora ambos interfaces por ejemplo con:

```
Router# show ip interface brief
```

Y vea también las entradas en la tabla de rutas.

Para probar la configuración colocaremos un PC en cada una de esas VLANs.

Conecte el interfaz 0 del PC A al switch1 y configure ese puerto del switch en modo acceso y en la VLAN 2. Configure la IP del interfaz del PC dentro de la LAN A. Configure la ruta por defecto de este PC con siguiente salto la IP del router1 en la LAN A.

Conecte el interfaz 0 del PC B al switch1 y configure ese puerto del switch en modo acceso y en la VLAN 2. Configure la IP del interfaz del PC dentro de la LAN B. Configure la ruta por defecto de este PC con siguiente salto la IP del router1 en la LAN B.

Pruebe a hacer ping y traceroute entre ambos PCs y verifique el correcto funcionamiento de la comunicación y que efectivamente los paquetes están siendo enrutados por router1.

**Punto de control (1%): Muestre esta última configuración a su profesor de prácticas**

## 5- 802.1Q en un PC

A continuación vamos a ver cómo emplear 802.1Q en un PC con Linux. En primer lugar haremos que un PC tenga interfaces lógicas en varias VLANs con un solo interfaz físico.

Manteniendo la configuración del apartado anterior conecte ahora el PC C a un puerto del switch1. Configure ese puerto del switch en trunk. Para crear los interfaces lógicos en el PC primero debemos asegurarnos de que el interfaz físico está activo, por ejemplo:

```
$ sudo ifconfig eth0 up
```

Ahora puede emplear el comando `vconfig` para crear cada interfaz lógico asociado a una VLAN. Por ejemplo, para crear uno asociado a la VLAN 2 (empleando encapsulado 802.1Q) sería:

```
$ sudo vconfig add eth0 2
```

A partir de ese momento debería tener un interfaz llamado `eth0.2`

Si en algún momento quiere borrar un subinterfaz de estos deberá emplear el mismo comando `vconfig` con la opción `'rem'`

Cree de esta forma el subinterfaz en el PC para la VLAN 2 y el de la VLAN 3. Asigne dirección IP a cada uno de ellos en la LAN que les corresponde. No configure ruta por defecto en este PC.

Ahora pruebe que puede hacer ping a PC A, PC B y ambos interfaces del router1. Puede ver el contenido de las tramas con ethereal.

Punto de control (1%): Muestre esta última configuración a su profesor de prácticas

¿Sabría hacer que este PC actuara ahora como router de forma análoga a router1?

## 6- Etherchannel

Etherchannel es la forma que tiene Cisco de llamar a la agregación de interfaces Ethernet. Esta agregación se puede basar en el protocolo estándar LACP (802.3ad) o en el protocolo propietario de Cisco PAgP.

Cuando se crea un EtherChannel se crea un interfaz lógico. A partir de ahí se pueden asignar manualmente interfaces físicas al interfaz lógico del Etherchannel.

En este apartado vamos a crear un canal entre dos conmutadores que agregue dos puertos de cada uno.

En primer lugar lleve a cabo la configuración de la figura 1, donde los dos enlaces entre los routers (por ejemplo los puertos 23 y 24 en ambos) estarán configurados como trunks y spanning-tree habrá desactivado automáticamente uno de ellos. Coloque los tres PCs en la misma LAN y genere tráfico simultáneamente desde los PCs B y C hacia el PC A. Compruebe por qué enlace entre los switches circula el tráfico.

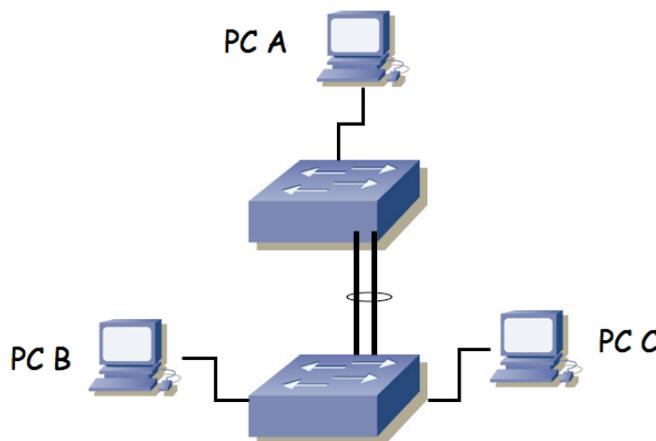


Figura 1.- Escenario con interfaz Etherchannel

Ahora crearemos la agrupación de los enlaces de trunk. En cada switch debemos colocar los puertos de interconexión dentro del mismo grupo, para lo cual, dentro de la configuración de cada uno de esos interfaces deberíamos hacer:

```
Switch(config-if)# channel-group 1 mode active
```

Suponiendo que queremos añadir el interfaz al grupo 1 y queremos que emplee LACP (802.3ad en lugar de la solución propietaria de Cisco, ¿qué ventajas puede tener emplear la solución estándar?)

Dado que vamos a repetir el mismo comando en varios interfaces, existe otra forma más cómoda de hacerlo. Podemos aplicar los mismos comandos a varios interfaces. Para ello debemos entrar en modo configuración de varios al mismo tiempo. Por ejemplo, en este caso se podría hacer del siguiente modo:

```
Switch(config)#interface range fa0/23 -24
Switch(config-if-range)#channel-group 1 mode active
```

El resultado es el mismo, es una simple cuestión de comodidad y rapidez.

Por supuesto, debemos repetir los comandos de configuración en el segundo conmutador.

Podemos ver ahora información sobre el estado de la agrupación con:

```
Switch> show interfaces etherchannel
```

Analice la información que puede obtener.

También podemos ver el nuevo interfaz lógico (port-channel1) en el resultado de comandos como por ejemplo `show interfaces summary`.

Estudie el estado actual de los puertos en el spanning-tree.

Repita el envío de tráfico simultáneamente de PC B y C a PC A. ¿Qué enlace emplean las tramas? Y si genera tráfico de PC A a PC B y C simultáneamente ¿qué camino siguen?

Las respuestas a las cuestiones anteriores dependen de cómo estén haciendo los conmutadores el reparto de la carga entre los enlaces del etherchannel. Cisco permite dos modos de reparto de carga que se controlan con el comando:

```
Router(config)# port-channel load-balance {dst-mac|src-mac}
```

Estudie las diferentes opciones y pruébelas.

Pista:

[http://www.cisco.com/en/US/docs/switches/lan/catalyst2950/software/release/12.1\\_19\\_ea1/configuration/guide/swethchl.html#wp1020011](http://www.cisco.com/en/US/docs/switches/lan/catalyst2950/software/release/12.1_19_ea1/configuration/guide/swethchl.html#wp1020011)

Punto de control (1%): Muestre esta última configuración a su profesor de prácticas

En la topología de la figura 1 ¿cuál sería la configuración de reparto de carga más rentable para aprovechar ambos enlaces?

## 7- Router Cisco como puente

Cisco IOS permite que un router actúe también como un puente entre varios interfaces. Es decir, se puede hacer que varios interfaces del router actúen como interfaces enrutados con normalidad mientras que otros se unan para formar un puente. Sería equivalente a lo que se representa en la figura 2. En esta figura tenemos un router con 8 interfaces. Entre 3 de ellos está enrutando IP. Otros 3 están siendo puenteados y los dos restantes forman un puente diferente. Ni las tramas ethernet ni los paquetes IP pueden pasar de un dominio a otro. Es decir, un paquete IP en uno de los dominios puenteados no será reenviado por el otro dominio ni salir por uno de los interfaces enrutados. Veremos de nuevo configuraciones de este estilo en las prácticas sobre la tecnología ADSL.

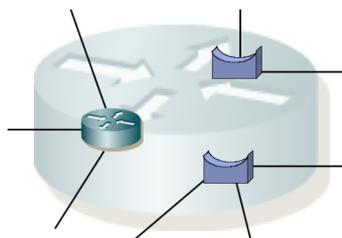


Figura 2.- Router Cisco actuando simultáneamente como puente

Lo que vamos a hacer en este apartado es simplemente configurar los 2 interfaces ethernet de un router (por ejemplo router2) como un puente. Cisco IOS permite que haya unos interfaces enrutados y otros puenteados mediante lo que llama CRB o *Concurrent Routing and Bridging*. Lo primero que hemos de hacer es activarlo en el router:

```
Router(config)# bridge crb
```

A continuación le debemos indicar que queremos crear un puente virtual que tendrá un identificador numérico y qué algoritmo de spanning-tree queremos que emplee. Optaremos por la solución estándar del IEEE y por ejemplo crearemos el bridge 1:

```
Router(config)# bridge 1 protocol ieee
```

Lo que nos falta es indicar qué interfaces del router van a formar parte del puente. Esto lo configuraremos para cada uno con:

```
Router(config-if)# bridge-group 1
```

Con estas indicaciones empiece por conectar un PC a cada interfaz del router. Asígneles direcciones IP en la misma red y compruebe que no hay comunicación de uno a otro. A continuación haga la configuración de bridge en el router y compruebe que ahora sí la hay.

IOS permite enrutar paquetes IP entre interfaces puenteados e interfaces enrutados. Para ello, emplea lo que llama *Integrated Routing and Bridging* (IRB). IRB se basa en crear un interfaz lógico asignado al puente. Ese interfaz actúa como un interfaz del router conectado a dicho puente como se representa en la figura 3.

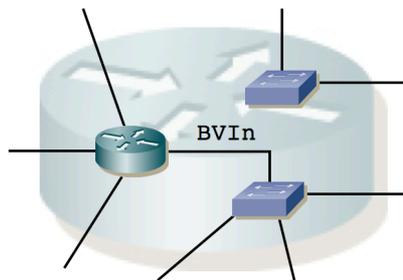


Figura 3.- Router Cisco empleando IRB

Activamos este modo de funcionamiento mediante:

```
Router(config)# bridge irb
```

El bridge lógico se crea igual que en el caso CRB, así como la asignación de interfaces al bridge. Lo que nos falta es indicar qué protocolo se va a enrutar, que en nuestro caso es IP:

```
Router(config)# bridge 1 route ip
```

Y crear el interfaz lógico o BVI (*Bridge Virtual Interface*):

```
Router(config)# interface bvi 1
```

Dentro del modo configuración de este interfaz virtual podremos asignarle dirección IP teniendo en cuenta que como se representa en la figura 3 ese es el interfaz del router en la otra LAN.

A continuación cree la configuración de la figura 4.

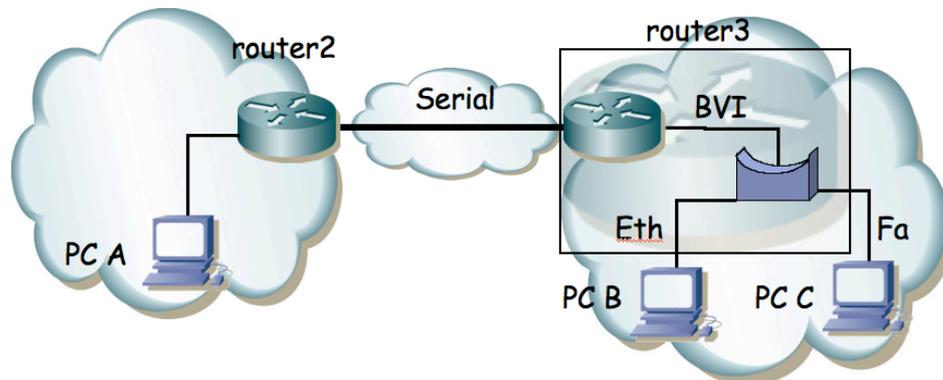


Figura 4.- Escenario empleando IRB

En ella PC A está en una LAN. PC B y PC C están en otra LAN puenteados entre sí por router3. La comunicación entre una LAN y la otra se lleva a cabo enrutada a través de los dos routers y sus interfaces serie.

Punto de control (1%): Muestre esta última configuración a su profesor de prácticas

## 8- SPAN

Como recordará, si tenemos varios PCs conectados en un hub o en varios que formen un solo dominio de colisión, cualquiera de ellos puede ver el tráfico que generan todos los demás. Esta característica es muy útil en muchas tareas de mantenimiento. Sin embargo, en el caso de tener conmutadores el dominio de colisión se limita a un puerto por lo que desde un PC conectado a un puerto no podremos ver los paquetes que se intercambian otras máquinas (salvo que vayan dirigidos a la MAC de broadcast o alguna de multicast).

Algunos conmutadores soportan la funcionalidad SPAN (*Switched Port Analyzer*) que permite que las tramas que se envían/reciben por uno o varios puertos o VLANs se copien en otro puerto al que se conectaría el analizador de tráfico.

Con los conmutadores Cisco del laboratorio podemos configurar un puerto de SPAN que vea el tráfico de otros puertos del conmutador. Esto se hace con el comando `monitor` en modo configuración.

Conecte 3 PCs a uno de sus conmutadores Cisco. Establezca una comunicación entre dos de ellos y compruebe que desde el tercero no puede capturar esos paquetes. Ahora active en el conmutador que clone en el puerto del tercer PC los paquetes enviados y recibidos por uno de los puertos de los otros PCs.

## 9- Evaluación

Mediante puntos de control