UD3. STP

Índice

Tormentas de broadcast

Una tormenta de difusión, en redes informáticas, es una situación que puede darse cuando se transmiten tramas de difusión o broadcast en la red, cada una de las cuales requiere que el nodo receptor responda reenviando su mensaje. La posible consecuencia es un aumento exponencial del tráfico de red, que conduce a una saturación completa de los recursos de red disponibles o, en cualquier caso, a una disminución drástica del rendimiento. Por tanto, es evidente que debe evitarse, por todos los medios posibles, que se puedan producir las tormentas de difusión.

A veces al interconectar conmutadores se producen bucles, es decir hay más de un camino posible entre dos redes. Estos bucles pueden hacerse por error o porque se quiere disponer de varios caminos para tener mayor fiabilidad y tolerancia a fallos. Debido a la forma como funcionan los puentes transparentes cuando se produce un bucle la red se bloquea.

Si en una red de conmutadores se produce un bucle la red queda fuera de servicio en cuanto se envía la primera trama broadcast o a un destino desconocido. Esto ocurre en cualquier red a los pocos segundos de entrar en funcionamiento.

Esto se debe a dos características de los puentes transparentes

Proceden por inundación cuando la dirección de destino no está en su tabla de direcciones

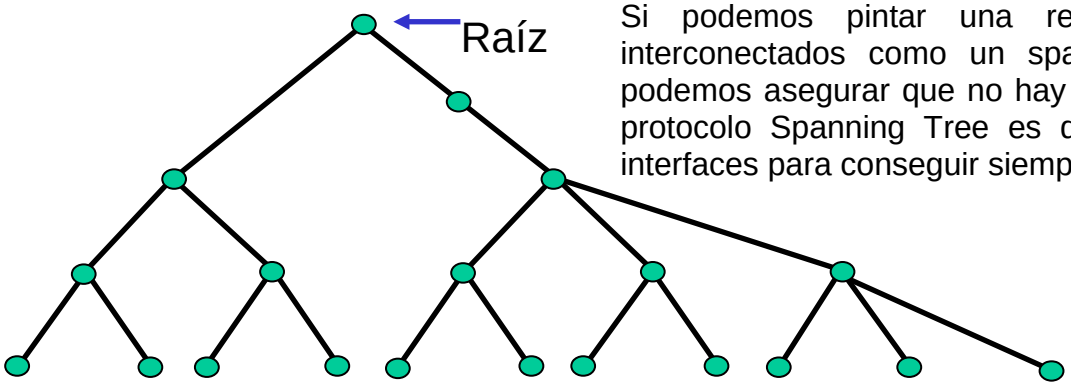
Cuando reenvían una trama la copia es indistinguible del original. No existe ningún campo (p. ej. un contador de saltos) que permita diferenciar las sucesivas copias

Existen dos posibles estrategias:

Se prohíbe taxativamente la creación de redes con bucles

Se habilita algún mecanismo, por software, que permita a los conmutadores detectar la presencia de bucles en la topología para que en ese caso desactiven las interfaces necesarias para que no haya bucles

Spanning Tree



Un Spanning Tree, o árbol de expansión, es un grafo en el que hay uno y solo un camino posible entre cualquier par de nodos (un árbol sin bucles).
Si podemos pintar una red de conmutadores interconectados como un spanning tree, entonces podemos asegurar que no hay bucles. El objetivo del protocolo Spanning Tree es desactivar lógicamente interfaces para conseguir siempre un spanning tree.

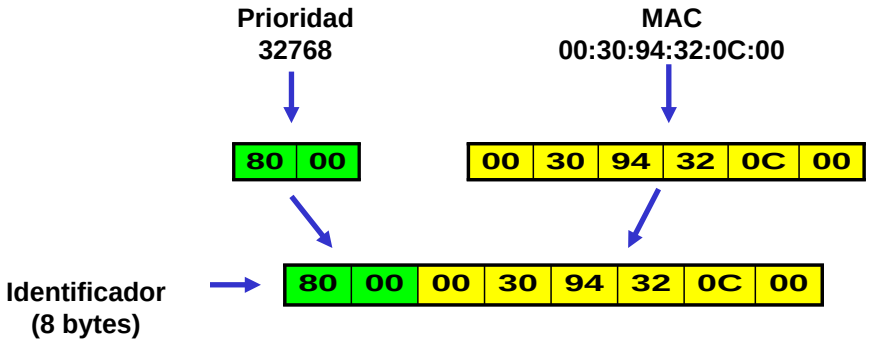
Funcionamiento de STP

Los conmutadores intercambian regularmente información sobre la topología de la red. Los mensajes que utilizan se denominan BPDUs (Bridge Protocol Data Units).

Las BPDUs emplean un Ethertype propio y se envían a una dirección multicast reservada, la 01-80-C2-00-00-00. Así se asegura que se identifican fácilmente y que los conmutadores sin ST los propagarán de forma transparente.

Cada conmutador dispone de un identificador único (ID) que crea a partir de una dirección MAC globalmente única que le ha asignado el fabricante. Además cada puerto del conmutador recibe un identificador y tiene asociado un costo.

El ID se construye a partir de una prioridad (configurable) y de la dirección MAC canónica del conmutador (fija). La prioridad puede valer entre 0 y 65535. Por defecto es 32768



Velocidad	Costo
4 Mb/s	250
10 Mb/s	100
16 Mb/s	62
45 Mb/s	39
100 Mb/s	19
155 Mb/s	14
200 Mb/s	12
622 Mb/s	6
1 Gb/s	4
2 Gb/s	3
10 Gb/s	2

Si se usa siempre la prioridad por defecto el conmutador con la MAC más baja es elegido raíz. Si a un conmutador le ponemos prioridad 32767 y dejamos el valor por defecto en el resto ese será seguro el de ID más bajo, y por tanto será elegido raíz.

Al principio los switches empiezan eligiéndose a sí mismos como raíz y comunicando lo que saben por todos los puertos. Cuando un switch ve a otro que tiene una prioridad mejor deja de proclamarse a sí mismo como raíz y anunciará la nueva raíz en pasos siguientes.

Cada conmutador calcula el grafo de la red y observa si existe algún bucle; en ese caso se van desactivando interfaces siguiendo unas reglas claras hasta cortar todos los bucles y construir un 'spanning tree'. Así, los conmutadores eligen como raíz del árbol a aquel que tiene el ID más bajo. Todos eligen al mismo.

Cada conmutador envía BPDUs por sus interfaces indicando su ID, el ID del conmutador raíz y el costo de llegar a él; los mensajes se van propagando por toda la red; cada conmutador al reenviar los mensajes de otros les suma el costo de la interfaz por la que los emite.

Puerto raíz: es un puerto que indica que es el mejor camino para llegar a la raíz.

Puerto designado: es un puerto no raíz que es el mejor del segmento para llegar a la raíz.

Puerto bloqueado: en un segmento es un puerto que no se usa.

Con las BPDUs recibidas cada conmutador calcula por que puerto puede llegar él al raíz al mínimo costo. Ese es su puerto raíz. En caso de empate elige el puerto de ID más bajo.

Cada LAN tiene un puerto designado, que es aquel por el que esa LAN accede al conmutador raíz al mínimo costo.

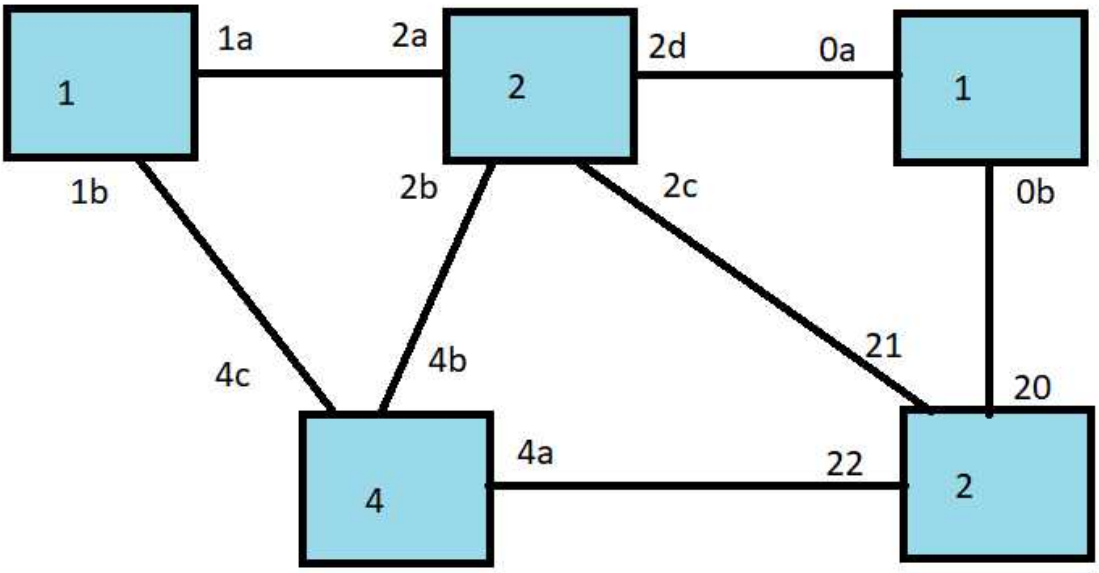
Los puertos que no son ni raíz ni designados son puertos bloqueados. Esos puertos son innecesarios para la comunicación y si se les deja funcionar provocan bucles

Así, el proceso es más o menos este:

- El switch raíz pone todos sus puertos a **designado**. Es lógico, el switch raíz es el mejor y sus puertos son siempre los mejores de cualquier segmento.
- El resto tienen que ir puerto por puerto examinando lo que han enviado y lo que han recibido. Este proceso es a su vez más complicado y lo desglosamos aparte.

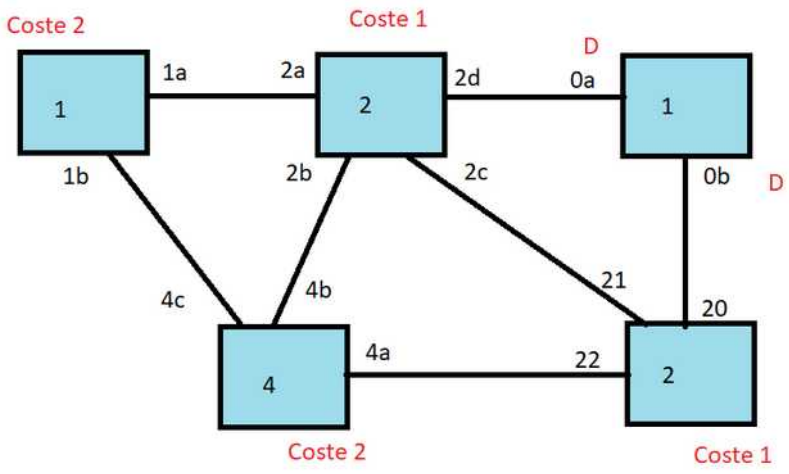
1. Si un puerto tiene un coste mejor que el otro puerto del segmento se pone a «designado».
2. Si un puerto tiene igual coste que el otro puerto del segmento se examinan las prioridades y se elige el puerto que lleve al switch con la mejor prioridad.
3. Los puertos que queden son puertos que se ponen a «bloqueado». Han perdido frente a sus competidores, ya sea por coste o por MAC.

En la figura siguiente se observa una red de switches. Podemos ver en su interior las prioridades que se les han dado y en los cables podemos ver la MAC de cada interfaz. Observando esto, ¿en qué estado quedarán los distintos puertos de los distintos switches?



Switch 1 (derecha), raíz

En primer lugar debemos saber quien actuará como raíz. Despues del proceso de elecciones ocurrirá que el switch 1 de la derecha ganará el proceso, ya que aunque tiene la misma prioridad que otro, el switch 1 de la derecha tiene la menor MAC (tiene la 0a). Esto significa que el proceso empieza declarando al switch 1 de la derecha el switch raíz y poniendo todos sus puertos a «designado». Todos los switches toman nota del coste que les supone llegar a la raíz.



A continuación iremos viendo los distintos switches y las decisiones que toman

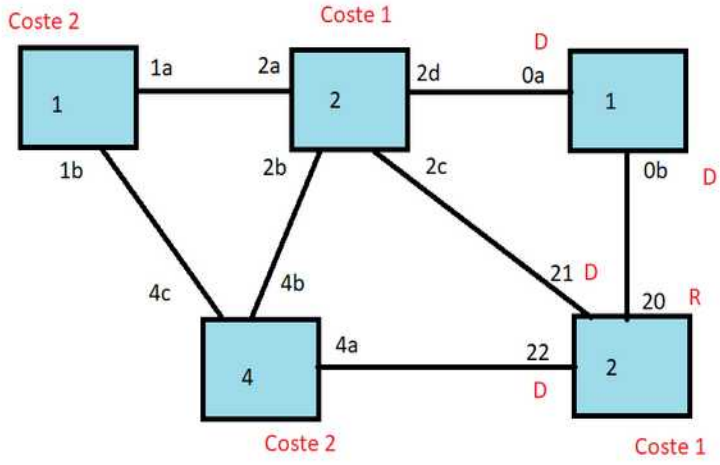
Switch 2 (abajo)

Ocurre esto:

Todos los switches deben empezar indicando su puerto raíz. En este caso, su mejor puerto para llegar a la raíz es el 20

Examinamos el puerto 21. Este switch tiene un coste 1 y el switch de «enfrente» también, pero nuestra mac 21 es mejor que la del vecino (que es 2c), así que ponemos este puerto a «designado».

Examinamos el puerto 22. Nuestro coste es mejor que el del vecino de enfrente, así que nuestro puerto «gana» y se pone a «designado».

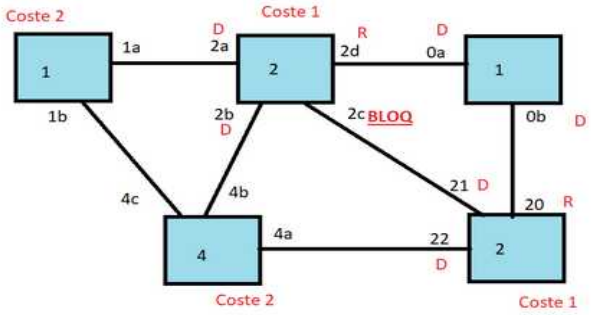


Switch 2 (centro-arriba)

El puerto 2d es el mejor. Se declara puerto raíz.

El puerto 2c se compara con el vecino de enfrente. El vecino y nosotros tenemos el mismo coste (que es 1) pero nuestra MAC es peor. Perdemos y bloqueamos este puerto con la MAC 2c.

El puerto 2b se compara con el vecino de enfrente. El vecino (switch 4) tiene un coste peor, así que él pierde y declaramos este puerto 2b como «designado».



El puerto 2a se compara con el vecino de enfrente. El vecino (switch 1, izquierda) tiene un coste peor. Él pierde y declaramos nuestro puerto 2a como «designado».

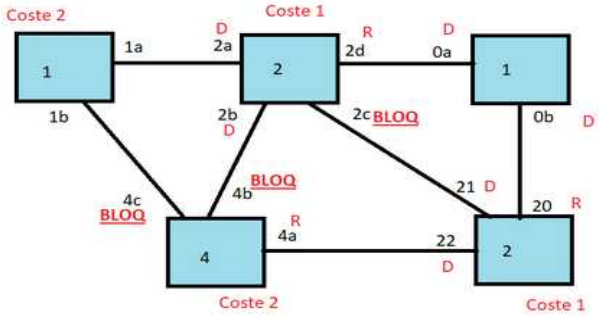
Switch 4

Examina sus propios puertos:

Su puerto 4a es el mejor, se declara raíz.

Examina su puerto 4b. El vecino de enfrente tiene un coste 1 y nosotros 2. Perdemos y declaramos este puerto 4b como «bloqueado».

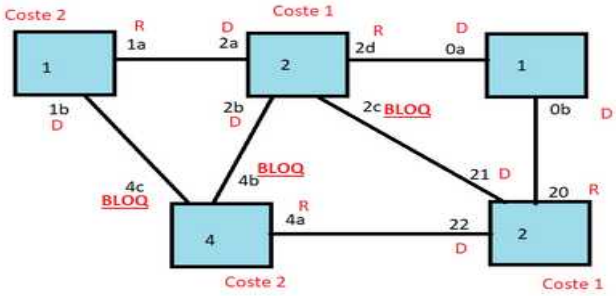
Examina su puerto 4c. El vecino tiene un coste 2 y nosotros también. Nuestra MAC 4c es peor que la suya (que es 1b), así que perdemos y declaramos este puerto como «bloqueado».



Switch 1 (izquierda)

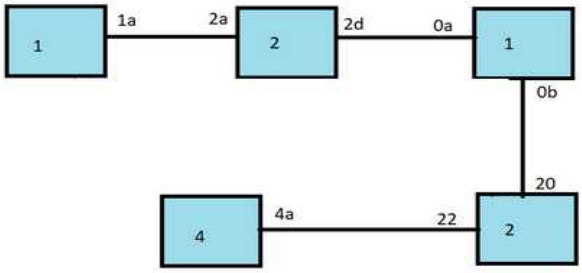
Su puerto 1a es el mejor para llegar a la raíz, así que se declara puerto raíz.

Su puerto 1b se compara con el vecino de enfrente. El vecino coste 2 y nosotros también, pero nuestra MAC es menor. Nuestro puerto 1b gana a su MAC 4c así que él pierde y nosotros ponemos nuestro puerto 1b a «designado».



Estado final

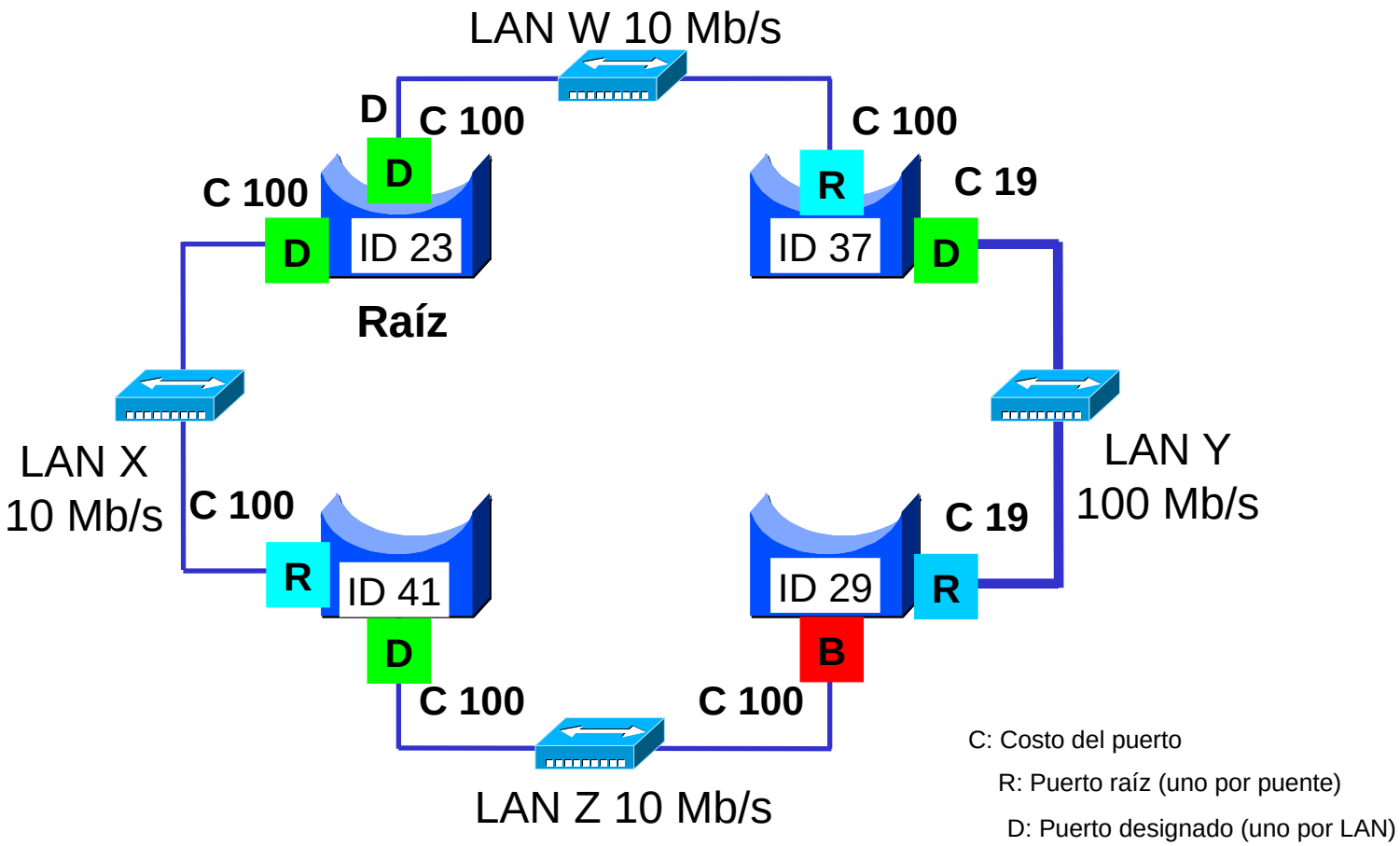
Si asumimos que los puertos bloqueados anulan el cable al que pertenecen observamos que la topología ha cambiado y queda algo como esto:



Pasos para averiguar la topología Spanning Tree

1. Asignar costos a todas las interfaces
2. Elegir el conmutador raíz (el de ID más bajo)
3. Elegir el puerto raíz de los demás conmutadores (el que les lleva al menor costo al puente raíz). En caso de empate elegir el puerto con ID más bajo
4. Elegir el puerto designado para cada LAN (el que le lleva al menor costo al puente raíz). En caso de empate elegir el conmutador con ID más bajo
5. Los puertos que no han sido elegidos como raíz ni como designados deben bloquearse
6. En Spanning Tree todo sigue reglas deterministas, ninguna elección se hace al azar. En caso de empate siempre hay una regla que dice que opción tomar. Dada una misma topología siempre se tomarán las mismas decisiones y siempre se llegará al mismo resultado

ejemplo:



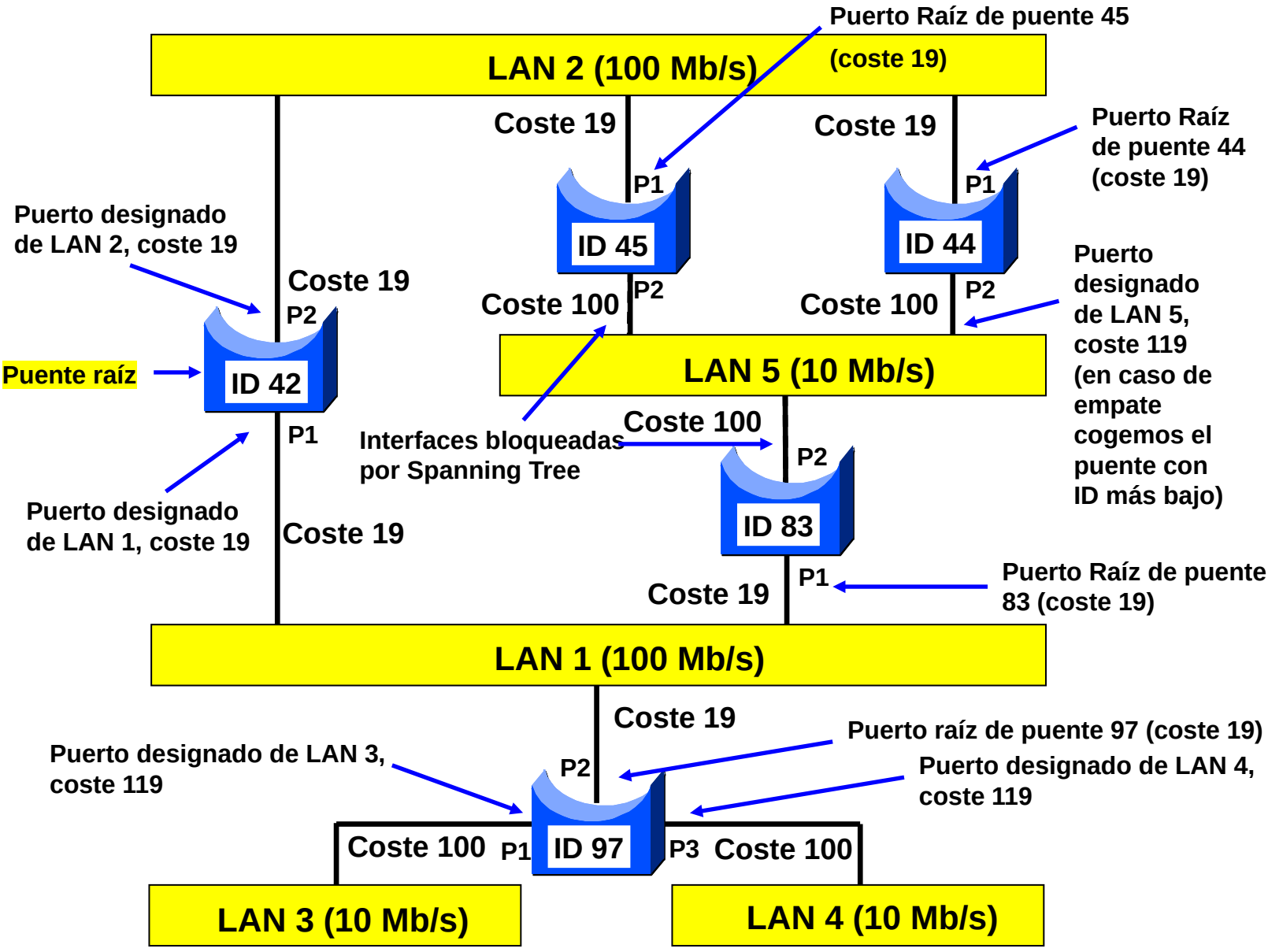
C: Costo del puerto

R: Puerto raíz (uno por puente)

D: Puerto designado (uno por LAN)

B: Puerto bloqueado

Ejemplo de red con bucles

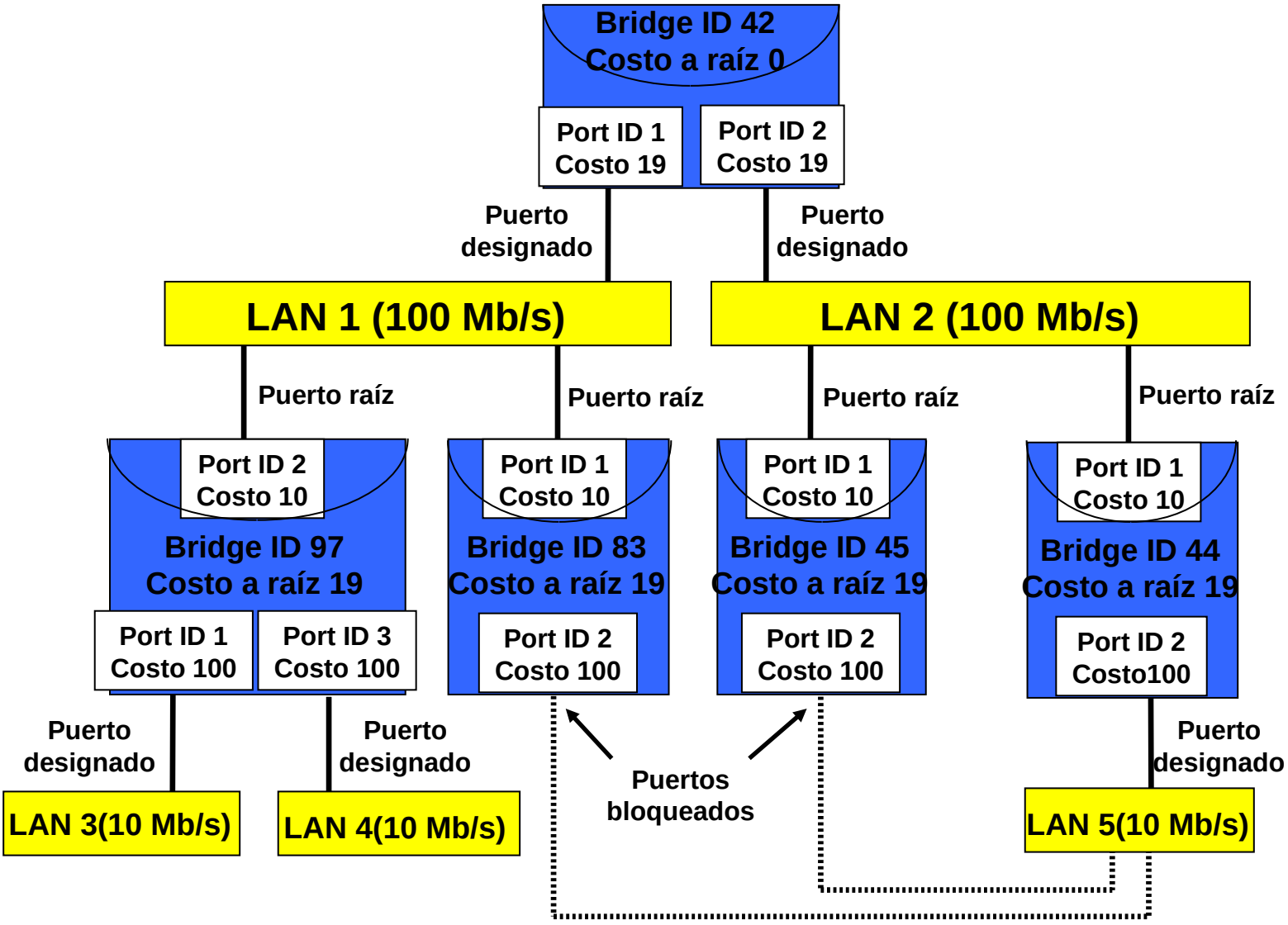


Dada una topología de red el conmutador raíz es siempre el mismo, independientemente del orden como se enciendan los equipos o como se conecten los cables

Si se utiliza la prioridad por defecto el conmutador raíz es el de la MAC más baja, que puede ser cualquier conmutador, probablemente uno periférico o poco importante. Si el conmutador raíz se apaga los demás han de elegir de entre ellos un nuevo raíz y recalcular el árbol, esto consume CPU y puede provocar inestabilidad si se tarda en llegar a la convergencia.

La prioridad permite controlar la selección del conmutador raíz asegurando que esa función recaiga por ejemplo en uno que esté siempre encendido, evitando así problemas de convergencia.

Spanning Tree de la red anterior



Rapid Spanning Tree

En 1998 se estandarizó el Rapid Spanning Tree (RST, IEEE 802.1w) una variante del protocolo original que reduce el tiempo de convergencia a unos 6 seg. Actualmente el ST tradicional esta declarado obsoleto.

Entre otras mejoras en RST los conmutadores mantienen información sobre la segunda ruta de menor costo al raíz, con lo que la conmutación a la nueva topología en caso de fallo de la actual es mucho más rápida.

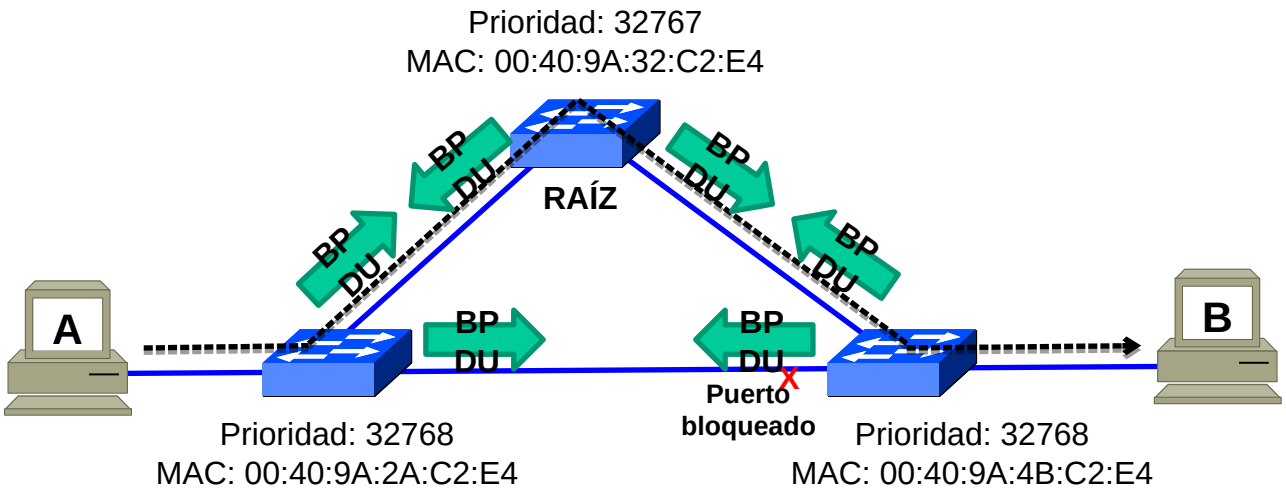
Para habilitarlo usamos

SW(config)# **spanning-tree mode rapid-pvst**

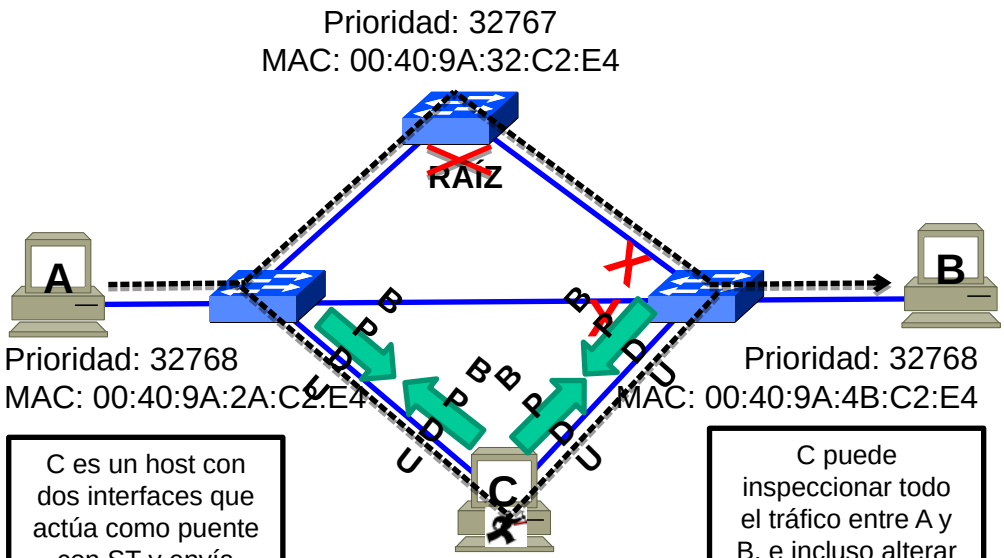
Ataques a STP

El protocolo Spanning Tree (ST) no incorpora ningún mecanismo de protección frente a ataques. Los mensajes se envían de forma no segura, sin autentificar ni encriptar. Cualquier equipo (un host por ejemplo) puede enviar BPDUs.
ST se basa en elegir un puente raíz y fijar un único camino para llegar a él desde cualquier punto
Como ya hemos visto el puente de menor prioridad es siempre elegido como raíz.

Situación inicial del ataque



Ataque producido por un host (host C)



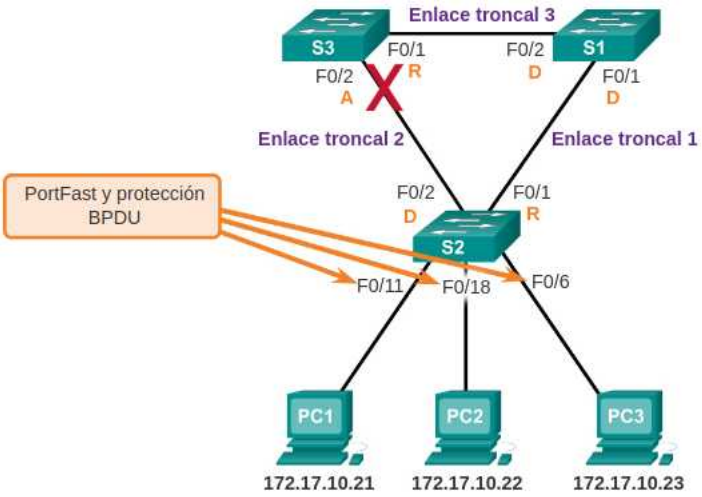
C es un host con dos interfaces que actúa como puente con ST y envía BPDUs, pero se ha puesto prioridad 1

C puede inspeccionar todo el tráfico entre A y B, e incluso alterar su contenido (atauqe de 'man in the middle')

No hay ningún motivo razonable que justifique el envío de BPDUs por parte de un host
En los conmutadores podemos activar la función 'BPDU Guard' en los puertos donde se conectan hosts. Así si se recibe por ellos una BPDU el puerto se desactiva (estado shutdown)
Alternativamente se puede activar el 'Root Guard'. En este caso no se bloquean todas las BPDUs, solo las que pretendan cambiar el raíz
Lo normal sería activar estas protecciones en todos los puertos, excepto aquellos en que se vayan a conectar conmutadores

PortFast y BPDU Guard

PortFast es una característica de Cisco para los entornos PVST+. Cuando un puerto de switch se configura con PortFast, ese puerto pasa del estado de bloqueo al de reenvío de inmediato, omitiendo los estados de transición de STP 802.1D usuales (los estados de escucha y aprendizaje). Puede utilizar PortFast en los puertos de acceso para permitir que estos dispositivos se conecten a la red inmediatamente, en lugar de esperar a que STP IEEE 802.1D converja en cada VLAN. Los puertos de acceso son puertos conectados a una única estación de trabajo o a un servidor.



En una configuración de PortFast válida, nunca se deben recibir BPDU, ya que esto indicaría que hay otro puente o switch conectado al puerto, lo que podría causar un bucle de árbol de expansión. Los switches Cisco admiten una característica denominada “protección BPDU”. Cuando se habilita, la protección BPDU coloca al puerto en estado *deshabilitado por error* al recibir una BPDU. Esto desactiva el puerto completamente. La característica de protección BPDU proporciona una respuesta segura a la configuración no válida, ya que se debe volver a activar la interfaz de forma manual.

La tecnología Cisco PortFast es útil para DHCP. Sin PortFast, un equipo puede enviar una solicitud de DHCP antes de que el puerto se encuentre en estado de enviar e impedirle al host la posibilidad de obtener una dirección IP utilizable y cualquier otra información. Debido a que PortFast cambia el estado a enviar de manera inmediata, el equipo siempre obtiene una dirección IP utilizable.

Nota: debido a que el propósito de PortFast es minimizar el tiempo que los puertos de acceso deben esperar a que converja el árbol de expansión, solo se debe utilizar en puertos de acceso conectados a hosts. Si habilitas PortFast en un puerto que se conecta a otro switch, corres el riesgo de crear un bucle de árbol de expansión. Sin embargo hay una excepción, cuando el puerto está conectado mediante un enlace trunk a un equipo y no a un switch entonces podremos forzar el portfast con **`spanning-tree portfast trunk`**

Para configurar PortFast es un puerto de switch, introducimos el comando

`spanning-tree portfast` del modo de configuración de interfaz en cada interfaz en la que se deba habilitar PortFast

El comando **`spanning-tree portfast default`** del modo de configuración global habilita PortFast en todas las interfaces no troncales.

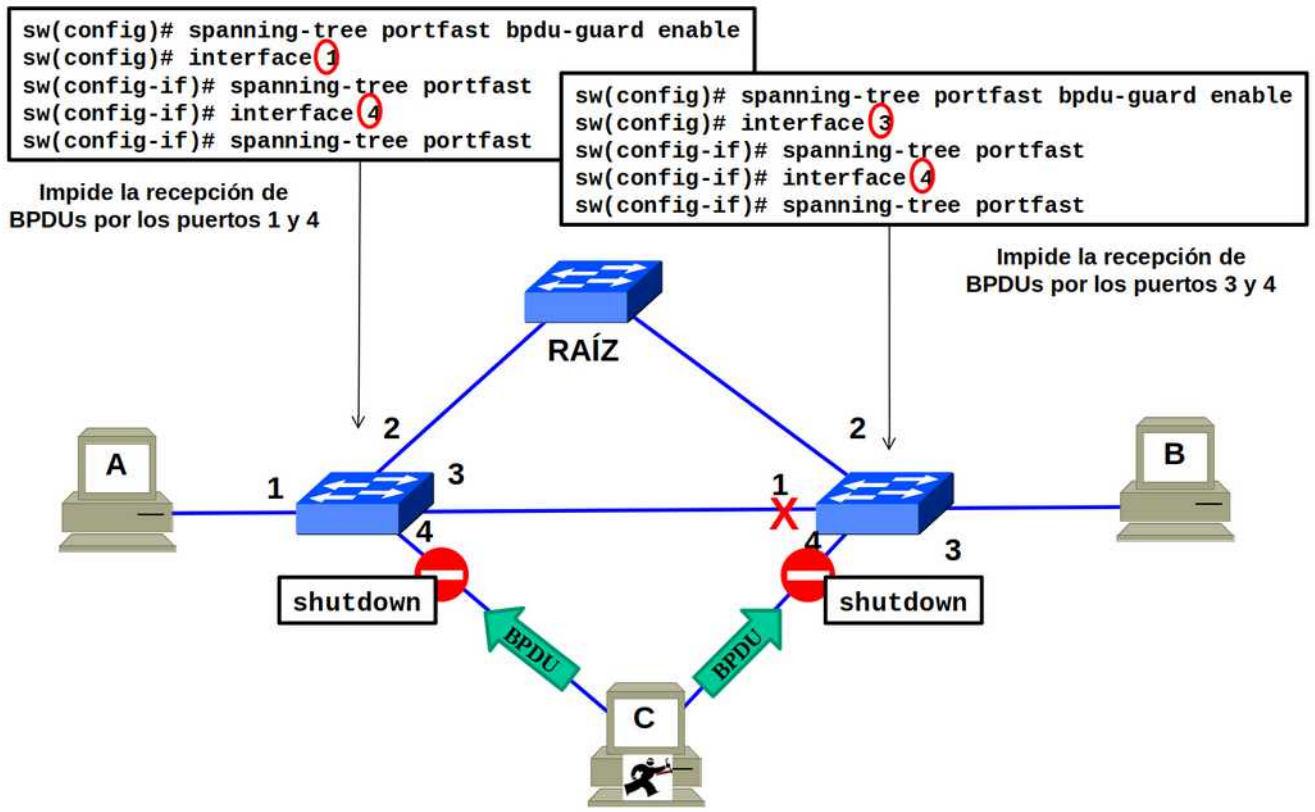
Para configurar la protección BPDU en un puerto de acceso de capa 2, utilizamos el comando

`spanning-tree bpduguard enable` del modo de configuración de interfaz.

El comando **`spanning-tree portfast bpduguard default`** del modo de configuración global habilita la protección BPDU en todos los puertos con PortFast habilitado.

Para verificar que se hayan habilitado PortFast y la protección BPDU para un puerto de switch, utilizamos el comando **`show running-config`**

La característica PortFast y la protección BPDU están deshabilitadas en todas las interfaces de manera predeterminada.

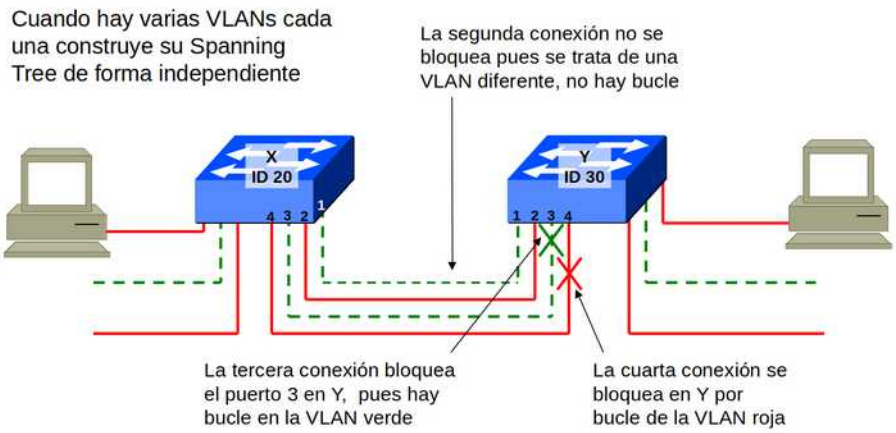


Vlans y STP

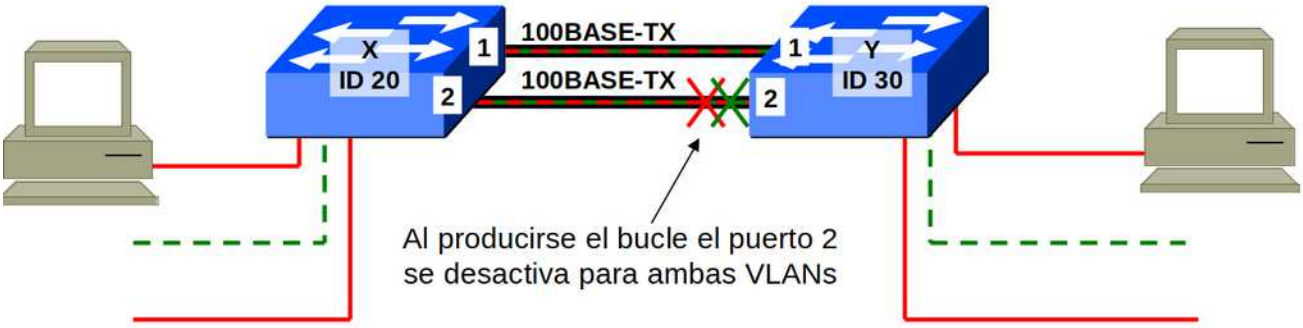
En principio cuando hay VLANs configuradas en un conmutador este ejecuta una instancia independiente de Spanning Tree para cada VLAN

Todos los parámetros característicos de Spanning Tree (prioridad, costo, etc.) se configuran independientemente para cada VLAN

Si se hace un bucle entre puertos asignados a diferentes VLANs no se bloqueará ningún puerto ya que en la topología de Spanning Tree no hay ningún bucle



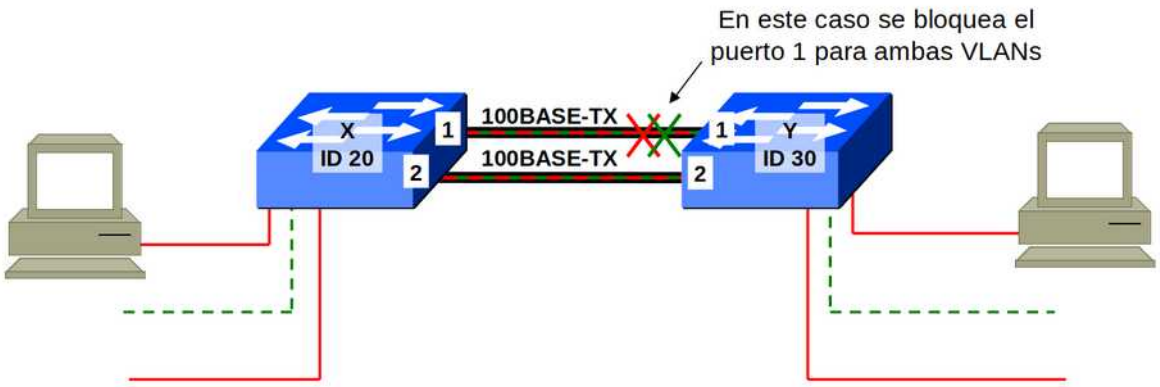
STP con enlaces Trunk



Dado un mismo costo y prioridad se elige como raíz el puerto de número menor, y por tanto se bloquea el de número mayor. La prioridad por defecto es 128.

VLAN	Puerto	Costo	Prioridad
Roja	1	19	128
	2	19	128
Verde	1	19	128
	2	19	128

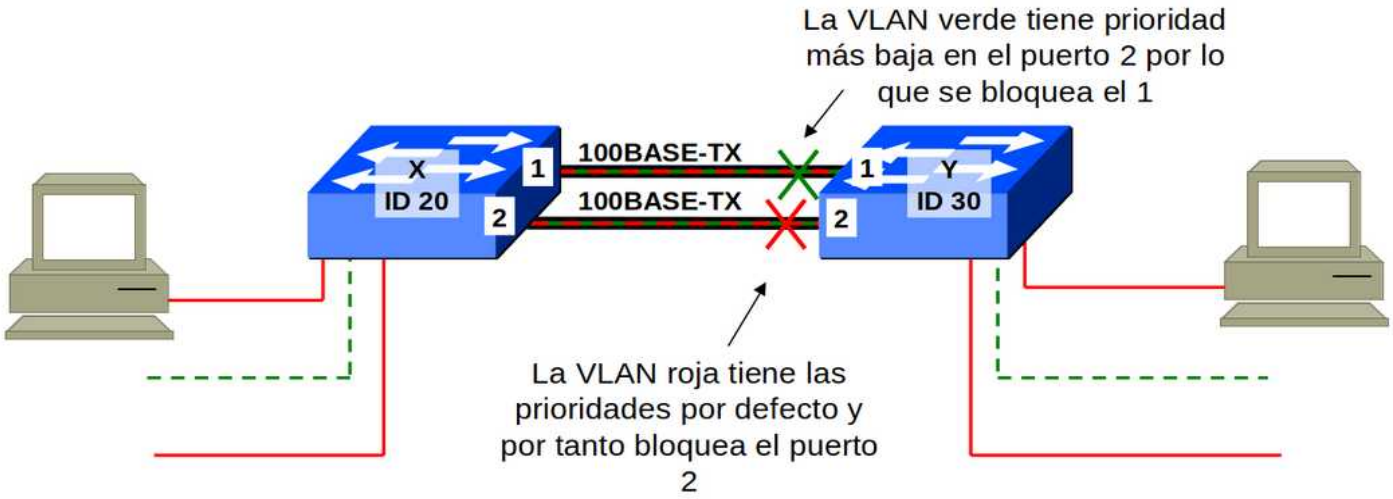
Si modificamos la prioridad, modificamos el puerto que se bloqueará



Modificando la prioridad se puede alterar la elección del spanning tree. Si se le da una prioridad menor al puerto 2 se le sitúa por delante del 1 y se le elige como puerto raíz, bloqueando entonces el 1.

VLAN	Puerto	Costo	Prioridad
Roja	1	19	128
	2	19	**127**
Verde	1	19	128
	2	19	**127**

Balanceo de carga en enlaces Trunk



Si modificamos la prioridad en una VLAN y a la otra le dejamos los valores por defecto el puerto bloqueado será diferente en cada VLAN

El resultado es que la VLAN roja usa el enlace 1-1 y la verde el 2-2. Se consigue balancear tráfico entre ambos enlaces.

VLAN	Puerto	Costo	Prioridad
Roja	1	10	128
	2	10	128
Verde	1	10	128
	2	10	**127**