

VLAN

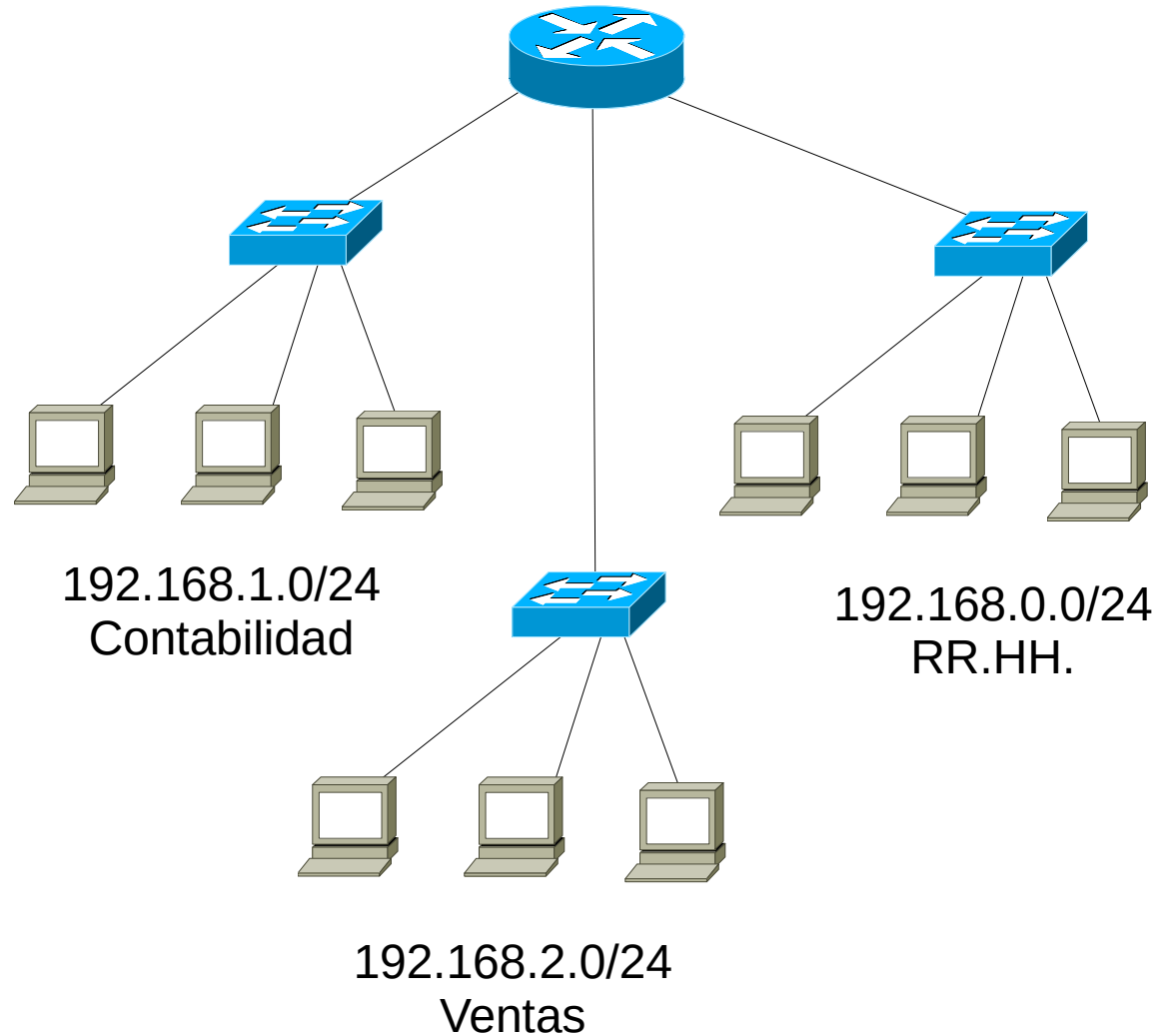


Curso 2024-2025

Escuela Superior de Ingeniería y Tecnología

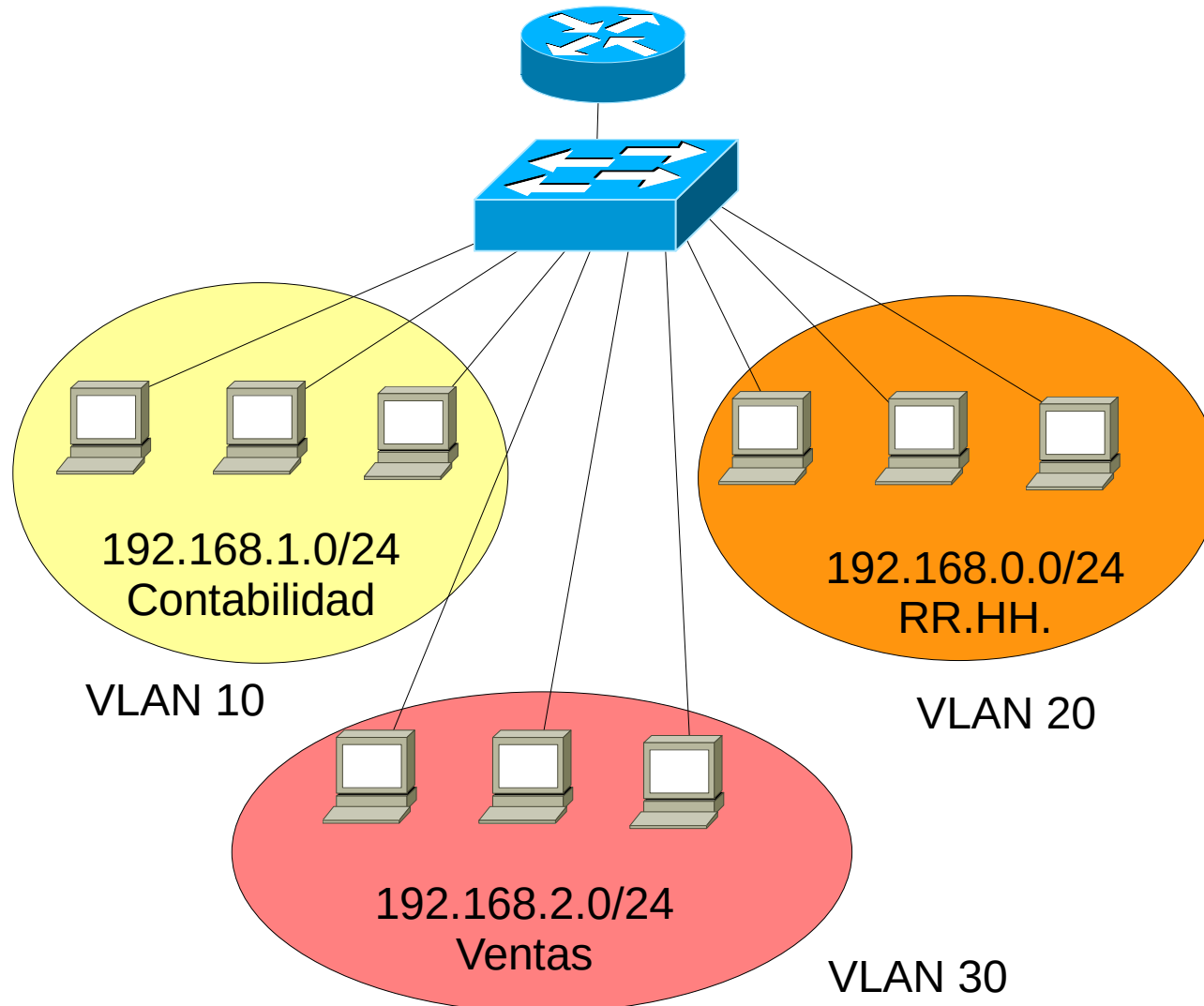
Introducción

Configuración sin utilizar VLANs: La topología física y la topología lógica coinciden

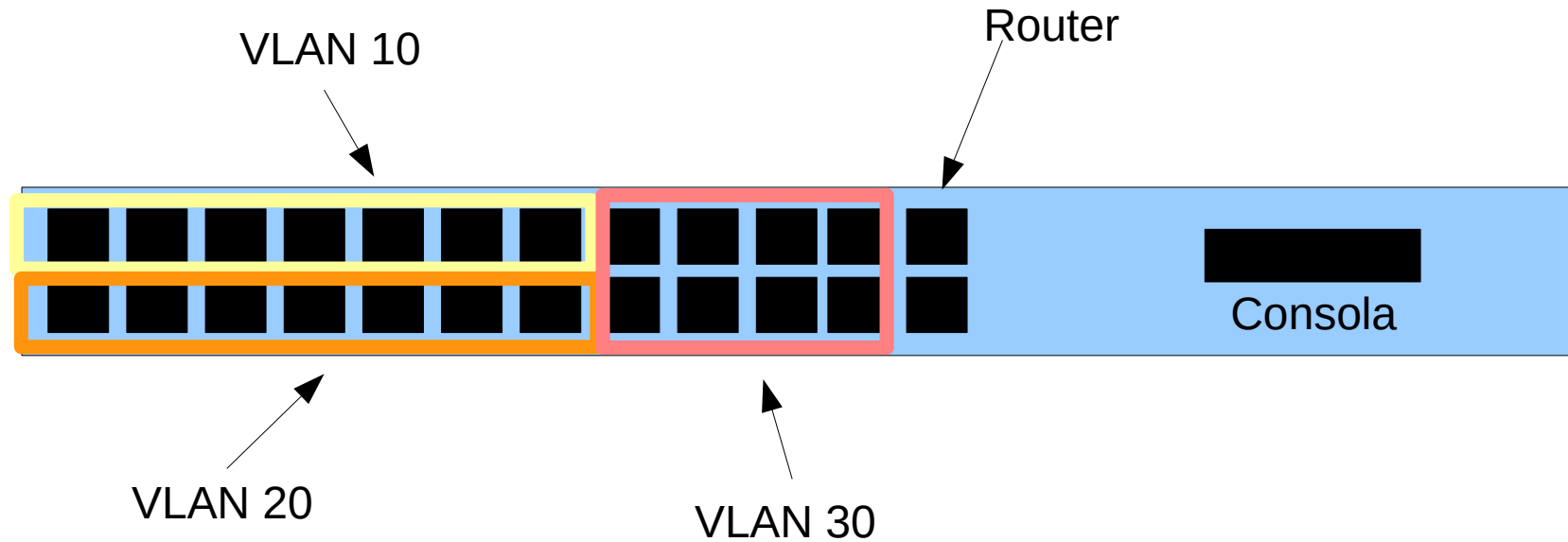


Introducción

Configuración utilizando VLANs: La topología lógica no coincide con la topología física



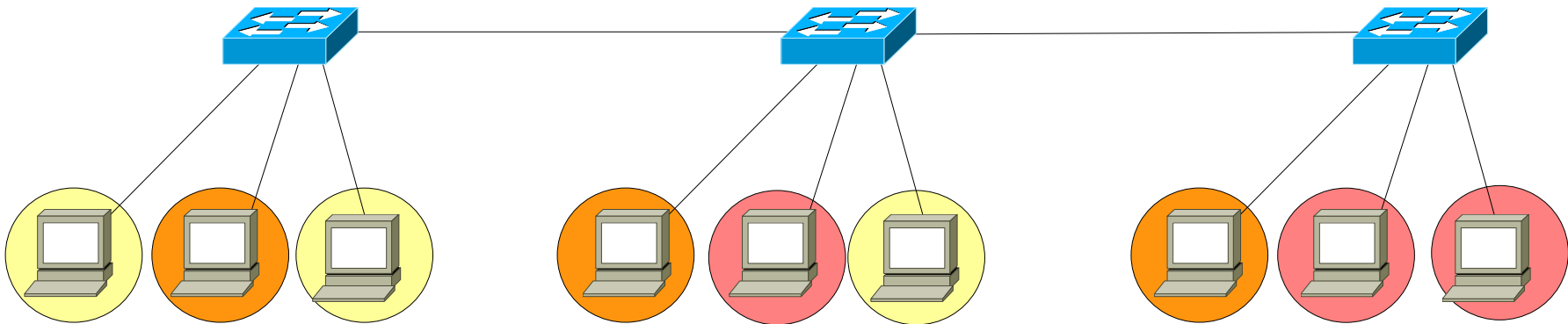
Introducción



Se abstrae un único switch en los tres switches diferentes de la visión lógica de la red.

Introducción

La naturaleza de las VLANs permite que se puedan formar varias VLANs a partir de distintos segmentos LAN ubicados en lugares físicamente separados.



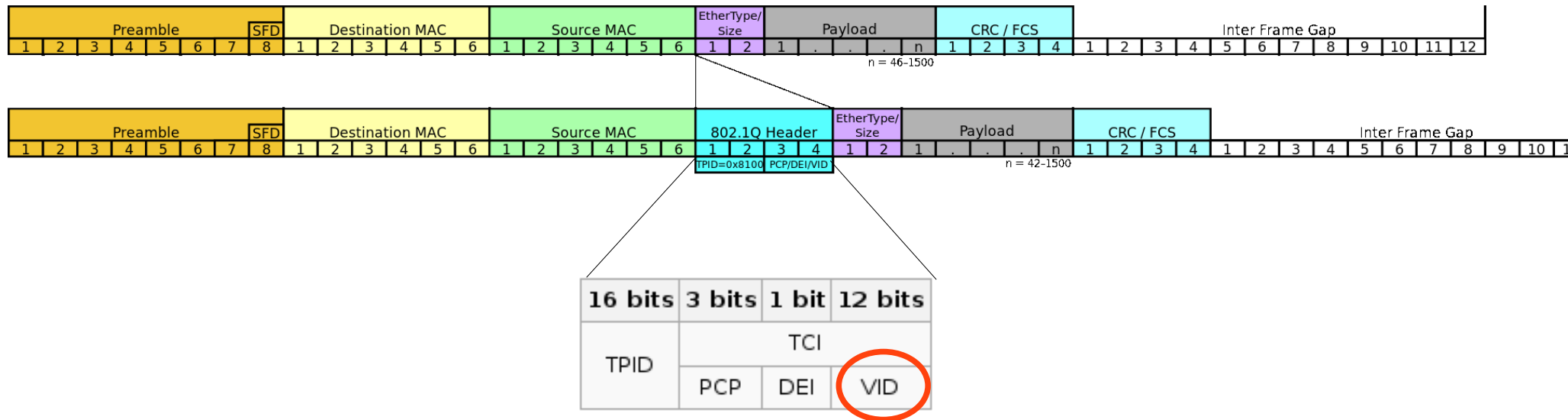
Enlaces troncales

Con un único switch que el tráfico de las distintas VLANs permanezca separado es sencillo.

Al conectar varios switches que soportan distintas VLANs es necesario etiquetar las tramas que se transmiten por los enlaces troncales con su identificador de VLAN, de modo que en el switch de destino sea posible conmutar a la VLAN correcta.

Enlaces troncales

Ethernet y IEEE 802.1Q



Al definir un puerto como puerto troncal se pueden distinguir dos tipos de tramas:

- **Tramas etiquetadas:** Aquellas cuyas tramas se envían con el VID utilizando IEEE 802.1Q u otro protocolo de encapsulamiento para trunks.
- **Tramas de la VLAN nativa:** Las tramas de esta VLAN se envían de forma nativa, es decir, sin VID. Sólo puede existir una única VLAN nativa para el mismo enlace troncal.

PVID (Port VLAN Identifier):

- Es el identificador de VLAN que se asigna por defecto a un puerto.

Tipos de puerto (I):

- **Puertos de acceso:** Este tipo de puertos son los que conectan hosts finales. Trabajan con las tramas clásicas de Ethernet, sin el agregado de las etiquetas de VLAN.
 - Cuando se recibe una trama no etiquetada, se etiqueta con la el PVID y se retransmite.
 - Cuando se recibe una trama etiquetada cuyo VID coincide con el PVID se retransmite.
 - Cuando se recibe una trama no etiquetada cuyo VID no coincide con el PVID se descarta.
 - Cuando emite una trama la emite sin etiquetar.
- **Puertos troncales(trunk):** Los puertos de troncales tienen una función especial que es la de conectar switches entre sí o un switch con un router.
 - Cuando se recibe una trama no etiquetada a través de este puerto se etiqueta con el PVID y se filtra o no en función de la lista de VLAN permitidas para este puerto.
 - Cuando se recibe una trama etiquetada a través de este puerto se filtra o no en función de si el VID está en la lista de VLAN permitidas del puerto.
 - Cuando se va a emitir una trama etiquetada a través de este puerto sólo se emite si la lista de VLAN permitidas para este puerto. Las tramas emitidas son tramas etiquetadas salvo que el VID coincida con el PVID.

Tipos de puerto (II):

- **Puertos generales:** Los puertos de generales son una mezcla entre puerto troncal y de acceso.
 - Cuando se recibe una trama no etiquetada a través de este puerto se etiqueta con el PVID y se filtra o no en función de la lista de VLAN permitidas para este puerto.
 - Cuando se recibe una trama etiquetada a través de este puerto se filtra o no en función de si el VID está en la lista de VLAN permitidas del puerto.
 - Cuando se va a emitir una trama se decide si esta es etiquetada o no en función de la regla de egreso especificada en la configuración del dispositivo (se indica si es tagged o untagged).

Configuración de VLAN

Crear VLANs:

```
S1(config)#vlan 10
```

```
S1(config-vlan)#name contabilidad
```

```
S1(config-vlan)#exit
```

```
S1(config)#exit
```

Configuración de VLAN

Añadir un puerto de acceso a una VLAN:

```
S1(config)# interface Gi1/0/7
S1(config-if)# switchport mode access
S1(config-if)# switchport access vlan 10
S1(config-if)# no shutdown
S1(config-if)# end
```

Este procedimiento configura el puerto Gi1/0/7 como puerto de acceso para la VLAN 10

Configuración de VLAN

Añadir un puerto troncal a una VLAN:

```
S1(config)# interface Gi1/0/1
S1(config-if)# switchport mode trunk
S1(config-if)# switchport trunk allowed vlan 10,20,99
S1(config-if)# no shutdown
S1(config-if)# end
```

Este procedimiento configura el puerto Gi1/0/1 como puerto troncal que permite las VLAN 10, 20 y 99

Configuración de VLAN

Configurar un puerto general:

```
S1(config)# interface Gi1/0/1
S1(config-if)# switchport general allowed vlan 20, 99 tagged
S1(config-if)# end
S1# end
```

```
#show interface switchport gi 1/0/1
Port Gi1/0/1:
PVID: 1
Acceptable frame type: All
Ingress Checking: Enable
Member in LAG: N/A
Link Type: General
Member in VLAN:
```

Vlan	Name	Egress-rule
----	-----	-----
1	N/A	Untagged
20	N/A	Tagged
99	N/A	Tagged

Configuración de VLAN

Configurar un puerto general (ahora se comporta como trunk):

```
S1(config)# interface Gi1/0/1
S1(config-if)# switchport general allowed vlan 20, 99 tagged
S1(config-if)# no switchport general allowed vlan 1
S1(config-if)# end
S1# end
```

```
#show interface switchport gi 1/0/1
```

```
Port Gi1/0/1:
```

```
PVID: 1
```

```
Acceptable frame type: All
```

```
Ingress Checking: Enable
```

```
Member in LAG: N/A
```

```
Link Type: General
```

```
Member in VLAN:
```

Vlan	Name	Egress-rule
----	-----	-----
20	N/A	Tagged
99	N/A	Tagged

Configuración de VLAN

Configurar un puerto general (ahora se comporta como access):

```
S1(config)# interface Gi1/0/1
S1(config-if)# switchport pvid 10
S1(config-if)# switchport general allowed vlan 10 untagged
S1(config-if)# no switchport general allowed vlan 1
S1(config-if)# end
```

```
#show interface switchport gi 1/0/1
```

```
Port Gi1/0/1:
```

```
PVID: 10
```

```
Acceptable frame type: All
```

```
Ingress Checking: Enable
```

```
Member in LAG: N/A
```

```
Link Type: General
```

```
Member in VLAN:
```

Vlan	Name	Egress-rule
----	-----	-----
10	N/A	Untagged

Configuración de VLAN

Configurar un puerto general (trunk con VLAN nativa):

```
S1(config)# interface Gi1/0/1
S1(config-if)# switchport pvid 10
S1(config-if)# switchport general allowed vlan 20, 99 tagged
S1(config-if)# switchport general allowed vlan 10 untagged
S1(config-if)# end
```

```
#show interface switchport gi 1/0/1
```

```
Port Gi1/0/1:
```

```
PVID: 10
```

```
Acceptable frame type: All
```

```
Ingress Checking: Enable
```

```
Member in LAG: N/A
```

```
Link Type: General
```

```
Member in VLAN:
```

Vlan	Name	Egress-rule
----	-----	-----
10	N/A	Untagged
20	N/A	Tagged
99	N/A	Tagged

Configuración de VLAN

Configurar un puerto general (trunk con VLAN nativa):

```
S1(config)# interface Gi1/0/1
S1(config-if)# switchport pvid 10
S1(config-if)# switchport general allowed vlan 20, 99 tagged
S1(config-if)# switchport general allowed vlan 10 untagged
S1(config-if)# end
```

```
#show interface switchport gi 1/0/1
```

```
Port Gi1/0/1:
```

```
PVID: 10
```

```
Acceptable frame type: All
```

```
Ingress Checking: Enable
```

```
Member in LAG: N/A
```

```
Link Type: General
```

```
Member in VLAN:
```

Vlan	Name	Egress-rule
----	-----	-----
10	N/A	Untagged
20	N/A	Tagged
99	N/A	Tagged

Configuración de VLAN

Revisar la configuración

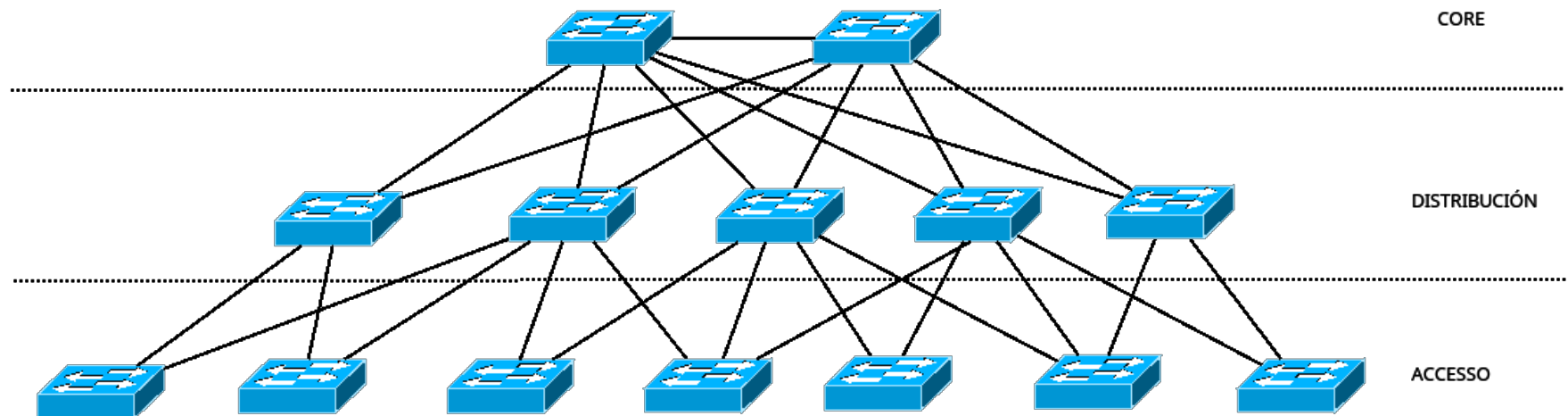
S1# show vlan

VLAN	Name	Status	Ports
1	System-VLAN	active	Gi1/0/5, Gi1/0/6, Gi1/0/7, Gi1/0/8, Gi1/0/9, Gi1/0/10
10	N/A	active	Gi1/0/1, Gi1/0/2
20	N/A	active	Gi1/0/1, Gi1/0/3
99	N/A	active	Gi1/0/1, Gi1/0/4

Es importante que vayamos comprobando que los puertos no se quedan en la VLAN por defecto.

Protocolos spanning tree

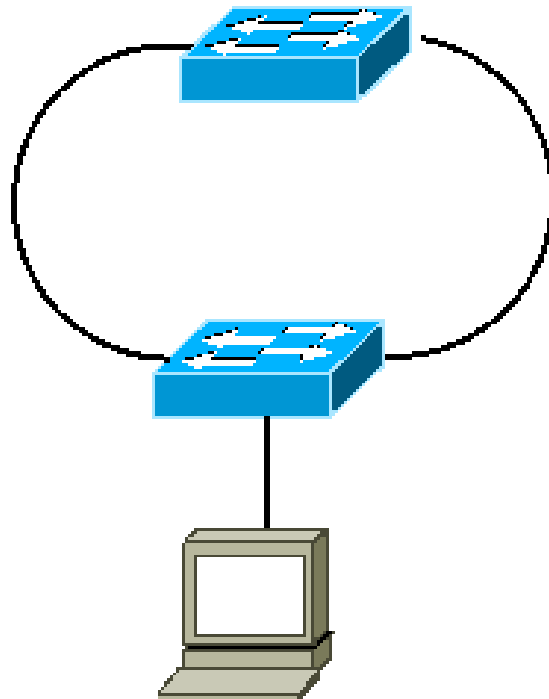
Enlaces redundantes para obtener alta disponibilidad:



Protocolos spanning tree

El problema de los enlaces redundantes es que se produce un fenómeno que se conoce como **tormenta de broadcast**.

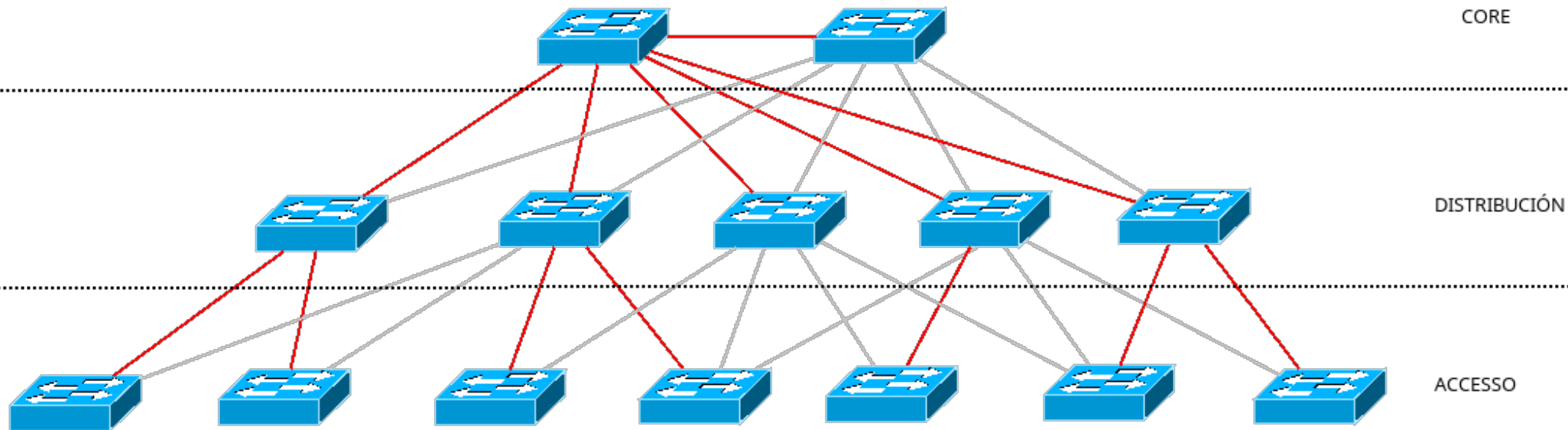
Si el ordenador emite una trama broadcast, ésta se quedará circulando por el bucle indefinidamente. Recuerda el funcionamiento de un switch !!!



¡Hay que evitar los bucles! Pero necesitamos tener bucles para tener alta disponibilidad y mejorar el rendimiento

Protocolos spanning tree

La solución es que la topología no contenga bucles. Debemos tener una topología en árbol, que por definición no contiene bucles.



Un árbol de expansión o spanning-tree es un subconjunto de las conexiones o enlaces de la red que **conecta todos los dispositivos** de la misma, sin redundancia (sin ciclos o lazos).

Protocolos spanning tree

Los protocolos Spanning Tree son un conjunto de protocolos de red utilizados para evitar la formación de bucles o ciclos en las redes conmutadas.

El protocolo más conocido y utilizado para este fin es el Spanning Tree Protocol (STP), definido en el estándar IEEE 802.1D. El STP crea un árbol de expansión (spanning tree) que selecciona el camino más corto entre todos los dispositivos de la red, deshabilitando los enlaces redundantes que podrían formar bucles. Solo se mantiene un camino activo, y los demás se bloquean hasta que se produzca una falla en el enlace principal.

Existen varias versiones y mejoras del protocolo Spanning Tree, como:

- **STP (Spanning Tree Protocol):** La versión original y estándar de STP, que utiliza el algoritmo de árbol de expansión para evitar bucles. Su tiempo de convergencia es de aproximadamente 30 segundos.
- **RSTP (Rapid Spanning Tree Protocol):** Una versión mejorada de STP que permite una convergencia más rápida en caso de que ocurran cambios en la topología de la red, minimizando el tiempo de inactividad.
- **MSTP (Multiple Spanning Tree Protocol):** Una extensión de RSTP que permite crear múltiples instancias de árboles de expansión para diferentes VLANs, lo que mejora la eficiencia y la utilización del ancho de banda en redes complejas.

Protocolos spanning tree

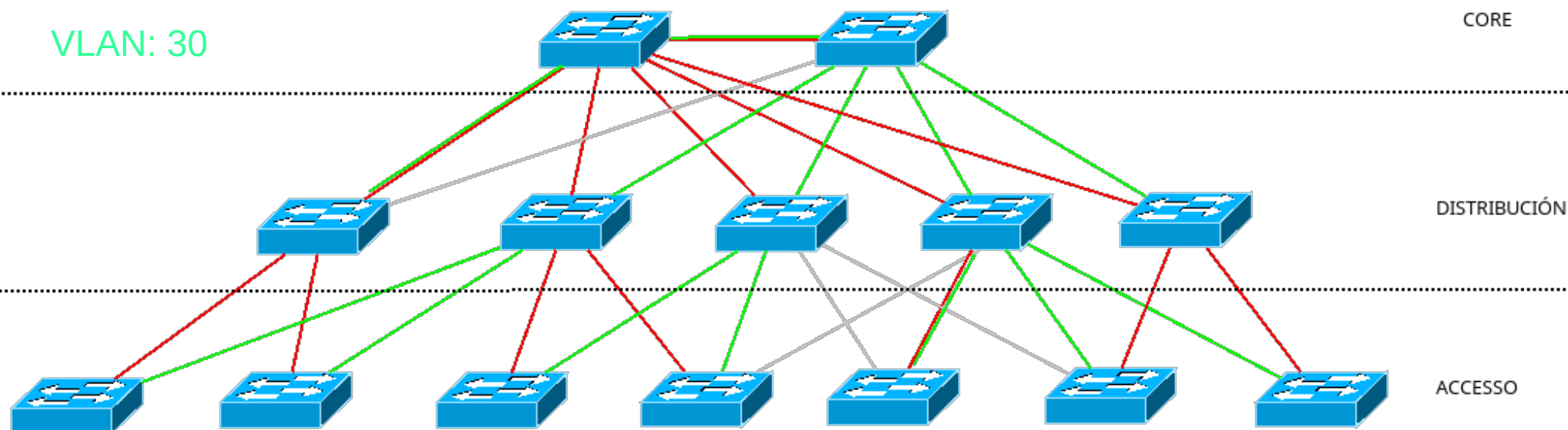
MSTP (Multiple Spanning Tree Protocol): Una extensión de RSTP que permite crear múltiples instancias de árboles de expansión para diferentes VLANs, lo que mejora la eficiencia y la utilización del ancho de banda en redes complejas.

Se puede utilizar cuando hay más de una VLAN. Imaginemos que una red tenemos las siguientes VLAN: 10, 20, 30

El protocolo permite la creación de lo que se conoce como asociación de VLAN. Esto permite agrupar varias VLAN con un mismo spanning-tree. Por ejemplo, si agrupamos 10,20 en una asociación y 30 en otra. Podríamos tener dos instancias del spanning-tree.

VLAN: 10,20

VLAN: 30



Protocolos spanning tree

¿Como se determina la raíz del spanning tree?

Cada uno de los switches del spanning tree quedan identificados mediante un identificador de puente (BID - Bridge Identifier). Este identificador está conformado por dos partes: una prioridad asignada al switch y la dirección MAC del switch:



La prioridad del switch está compuesta por dos partes una prioridad (4 bits) configurable por el administrador de red y un identificador de sistema extendido (12 bits). El identificador de sistema extendido se utiliza, por ejemplo, para incluir el identificador de instancia en MSTP. Esto permite distinguir las distintas instancias de spanning tree.

Protocolos spanning tree

¿Como se determina la raíz del spanning tree?

El BID, a parte de identificar a cada uno de los switches que forman parte del spanning tree, se utiliza para establecer el puente raíz del spanning tree. Una vez se inician los switches que forman parte de la red, intercambian una serie de BPDUs (Bridge Protocol Data Units) que, entre otros datos, llevan el BID de cada switch.

De esta forma, cada switch compara el BID de la BPDU entrante con su propio BID y **establece como switch raíz aquel con BID menor**. Después del proceso de intercambio de BPDUs el switch con menor BID se convierte en raíz.

Para controlar qué switch ejerce de puente raíz el administrador de red deberá modificar **las prioridades** que por defecto están establecidas a 32768 (Un valor menor es más prioritario).

Los valores posibles de prioridad son múltiplos de 4096: 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, **32768**, 36864, 40960, 45056, 49152, 53248, 57344, 61440.

Protocolos spanning tree

¿Qué switches deben ser más prioritarios?

