**El estándar IEEE 802.1Q**

Aunque varios fabricantes inicialmente crearon otros formatos propietarios VLAN, el estándar vigente en uso hoy en día es 802.1Q. En pocas palabras, 802.1Q proporciona una forma sencilla para que múltiples switches VLAN puedan interoperar agregando datos específicos de la VLAN directamente en los encabezados de cada uno de los paquetes Ethernet.

El estándar IEEE 802.1Q es un estándar abierto, lo que significa, en teoría, que todos los dispositivos compatibles pueden interoperar independientemente del fabricante. Las VLANs de Linux se basan en 802.1Q, y prácticamente cualquier switch que implemente "VLAN" será compatible con este estándar.

Los switches VLAN son una evolución natural de los dispositivos de Ethernet, cuyos ancestros ​​se puede decir que fueron el switch y el hub. La diferencia fundamental entre un switch y un hub es que el switch toma decisiones. No enviará paquetes a puertos donde la dirección MAC de destino no se puede encontrar. Los switches automáticamente se enteran de mapeos puerto/MAC en tiempo real al procesar los paquetes (y almacenar esa información en su "caché ARP").

Un switch VLAN tiene otra característica especial. No va a enviar los paquetes a puertos que no son "miembros" de la VLAN a la que pertenece el paquete. Esto se basa en el identificador de VLAN (VID) del paquete, que es un número entre 1 y 4096.

Si un paquete no tiene ya un VID, se le asigna uno basado en el puerto en el que llegó. Este es el VID primario (PVID) del puerto. Cada puerto del switch puede ser miembro de múltiples VLANs, una de las cuales se debe configurar como su PVID (Parent VID).

El VID se almacena en encabezado extra de 4 bytes que se añade al paquete llamado Tag o etiqueta. Agregar una Tag a un paquete que se llama de etiquetado. Sólo los dispositivos VLAN saber qué hacer con los paquetes etiquetados: los dispositivos Ethernet normales no los verán. A menos que el paquete se envíe a otro switch VLAN, el Tag debe ser eliminado antes de ser enviado. Esta eliminación de la etiqueta se realiza después de que el switch determina el puerto de salida.

Si un puerto está conectado a otro switch VLAN, las etiquetas deben ser preservadas, de modo que el otro switch pueda identificar la VLAN de los paquetes y manejar en consecuencia. Cuando un paquete tiene que atravesar varios switches en una VLAN, todos los switches subsecuentes se basan en la VID que le fue asignado al paquete por el primer switch que lo recibió.

En el caso de una VLAN sólo con un solo switch, no hay paquetes etiquetados que deban ser enviados o recibidos. Sin embargo, aún es útil pensar en el etiquetado y eliminación de etiquetas como algo que ocurre:

* El paquete llega y se etiqueta de acuerdo con el PVID del puerto de entrada.
* El puerto de salida es determinado en base al VID en la etiqueta.
* Al paquete se le quita la etiqueta y se envía.

Todos los paquetes comienzan como no etiquetados al entrar en la red, y también siempre debe terminar como no etiquetados al salir de la red y llegar a su destino. A lo largo de su viaje, si se cruzan una red VLAN, se les etiqueta con un VID, switcheado de acuerdo al VID por uno o más switches VLAN, y, finalmente, desetiquetado por el último switch VLAN.

Según lo visto hasta ahora, sabemos que hay tres cosas que hay que configurar para cada puerto de cada switch:

* Miembros VLAN (lista de VID).
* PVID (debe ser uno de los miembros VLANs).
* Si los paquetes se deben dejar etiquetadas o no cuando se envían (salida).

Con uno o más switches, se puede lograr cualquier topología de VLAN configurando selectivamente las tres opciones mencionadas en cada puerto.

**Configuración de interfaces VLAN**

El soporte de VLAN requiere un kernel compilado con CONFIG\_VLAN\_8021Q y el paquete vlan de línea de comando (se sugiere que también habilitar CONFIG\_BRIDGE\_EBT\_VLAN para que pueda aplicar reglas en base a VID en las reglas de ebtables).

Utilizamos la herramienta vconfig para crear las interfaces virtuales de VLAN basada en la combinación de una interfaz física Ethernet y un ID de VLAN específica. Estas interfaces se pueden utilizar como cualquier otra interfaz de Ethernet en su sistema.

Ejecuta los siguientes comandos para agregar una nueva interfaz asociada con eth0 y VID 5:

vconfig add eth0 5

ip link set eth0.5 up

Esto creará la interfaz virtual eth0.5, que tendrá los siguientes comportamientos especiales específicos de VLAN:

* Los paquetes enviados desde eth0.5 serán etiquetados con VID 5 y enviados desde eth0.
* Los paquetes recibidos en eth0 etiquetados con VID 5 aparecerán en eth0.5 como paquetes normales (es decir, sin etiqueta).

Sólo los paquetes que fueron marcados con VID 5 llegarán a la interfaz virtual VLAN.

**Juntándolo todo**

La mayor diferencia entre Linux y un switch VLAN adhoc, es que Linux puede participar como equipo normal en la red en lugar de sólo reenviar paquetes a otras máquinas. Debido a que el equipo Linux en sí mismo puede ser el punto final de la red de comunicaciones, el enfoque de configuración es diferente a la de un típico switch de VLAN.

En lugar de establecer la pertenencia a la VLAN para cada puerto, cada combinación puerto/VID tiene su propia interfaz virtual *eth*. Mediante la adición de estas interfaces y, opcionalmente, haciendo bridge con interfaces físicas, se puede crear cualquier configuración deseada VLAN.

No hay ninguna opción PVID por puerto en Linux. Está implícitamente basado en qué interfaz VLAN la interfaz de entrada física está haciendo bridge o puenteada. Los paquetes son etiquetados si son enviados a una interfaz virtual VLAN de acuerdo con el VID de la interfaz. Las operaciones de etiquetado y eliminación de etiquetas se producen automáticamente conforme el flujo de paquetes se da entre las interfaces físicas y virtuales de un bridge dado. Hay que recordar que la configuración de PVID sólo es relevante cuando los paquetes reenviados se recibieron sin etiqueta.

Con un típico switch VLAN sólo hay un bridge (el switch per se), de los que cada puerto es un miembro. La segmentación del tráfico se realiza por separado con cada puerto de entrada (PVID) y de reglas de pertenencia VLAN de salida. Debido a que Linux puede tener varios bridges, el establecimiento de PVID es innecesario.

Estos detalles son simplemente convenciones, las configuraciones efectivas siguen siendo las mismas en todas las plataformas VLAN. Suena más complicado de lo que realmente es. La mejor manera de entender todo esto con algunos ejemplos del mundo real.

**Uniendo VLAN existentes**

Digamos que tenemos una máquina Linux con una sóla interfaz física (*eth0*) que requiere unirse a tres VLAN existentes: VID 10, 20 y 30. En primer lugar, es necesario verificar la configuración actual del switch/puerto, al que se podrá conectar la máquina Linux. Tiene que ser un miembro de las tres VLANs, con marcado en las tres VLANs. A continuación, ejecute estos comandos en la máquina Linux:

ip link set eth0

vconfig add eth0 10

ip link set eth0.10 up

vconfig add eth0 20

ip link set eth0.20 up

vconfig add eth0 30

ip link set eth0.30 up

A continuación, puede utilizar eth0.10, eth0.20 y eth0.30 como interfaces normales (agregar direcciones IP, ejecutar dhclient y así sucesivamente). Estas se comportan igual que las interfaces físicas normales conectados a cada una de las VLANs.