Integrantes: Martina Nahman y Emiliano Germani

Trabajo Práctico N°2: Componentes, equipamiento y tramas en redes Ethernet y IEEE802.11. VLANs

Introducción

El presente trabajo práctico tuvo como objetivo aplicar los conocimientos adquiridos sobre protocolos de la capa de enlace (Ethernet e IEEE 802.11), analizando la estructura de redes mediante el uso de Cisco Packet Tracer. Las actividades propuestas abarcaron desde la configuración de una red Ethernet, pasando por la creación de VLANs, hasta el análisis de colisiones dentro de una red.

Actividad 1: Ethernet, Switches y Hubs

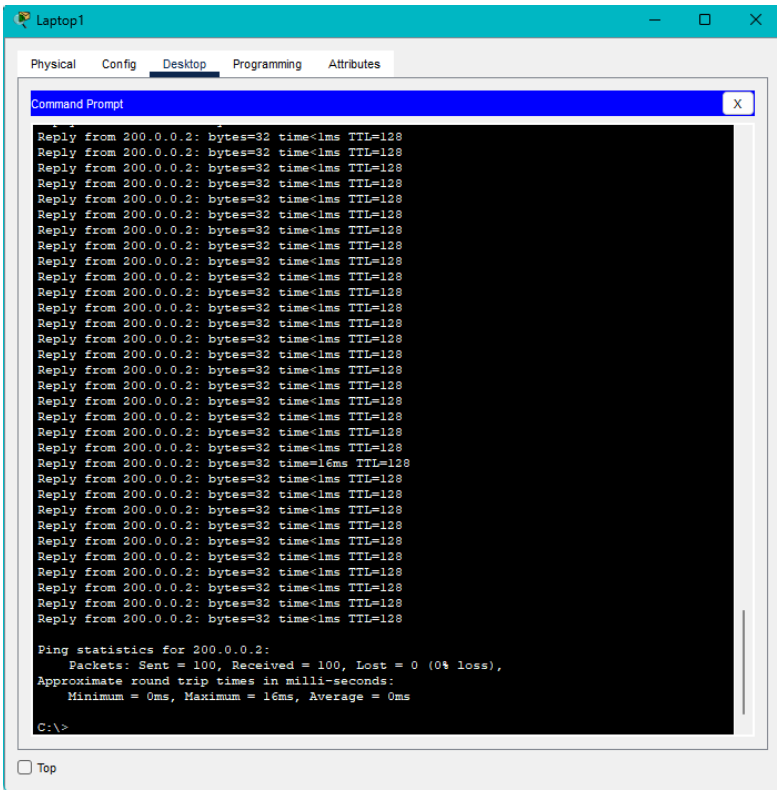
Se implementó en Packet Tracer una red compuesta por laptops, switches y hubs. Se utilizó el modelo Switch 2950-24, conectando los dispositivos con cables directos (Copper Straight-Through) y cables cruzados (Copper Cross-Over) según correspondía.

Cada laptop fue configurada con una IP estática del rango 200.0.0.X, utilizando la máscara de subred 255.255.255.0. Se verificó la conectividad entre dispositivos mediante el comando ping desde la consola de cada laptop.

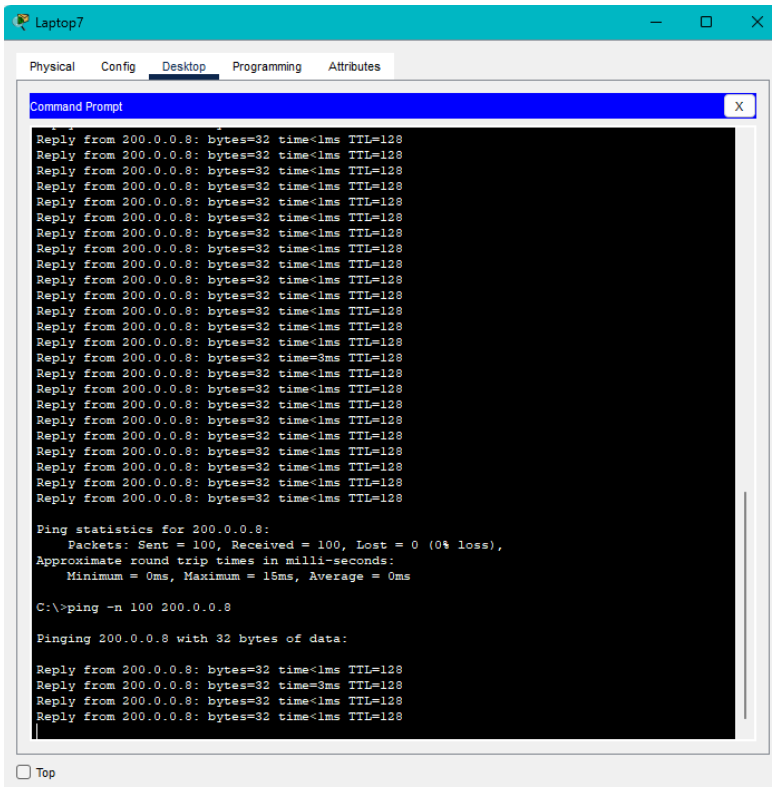
Luego, se analizó si los sniffers eran capaces de capturar tráfico entre laptops. Para ello se envió tráfico ICMP entre laptops específicas (por ejemplo, entre Laptop7 y Laptop8 con ping -n 100) y se comprobó desde la interfaz gráfica del sniffer si las tramas ICMP podían ser detectadas. Se identificaron también tramas STP generadas por los switches.

Finalmente, se analizó mediante Wireshark (o el sniffer de Packet Tracer) la estructura de un paquete ICMP y uno STP, verificando su encapsulamiento en la trama Ethernet.

Integrantes: Martina Nahman y Emiliano Germani

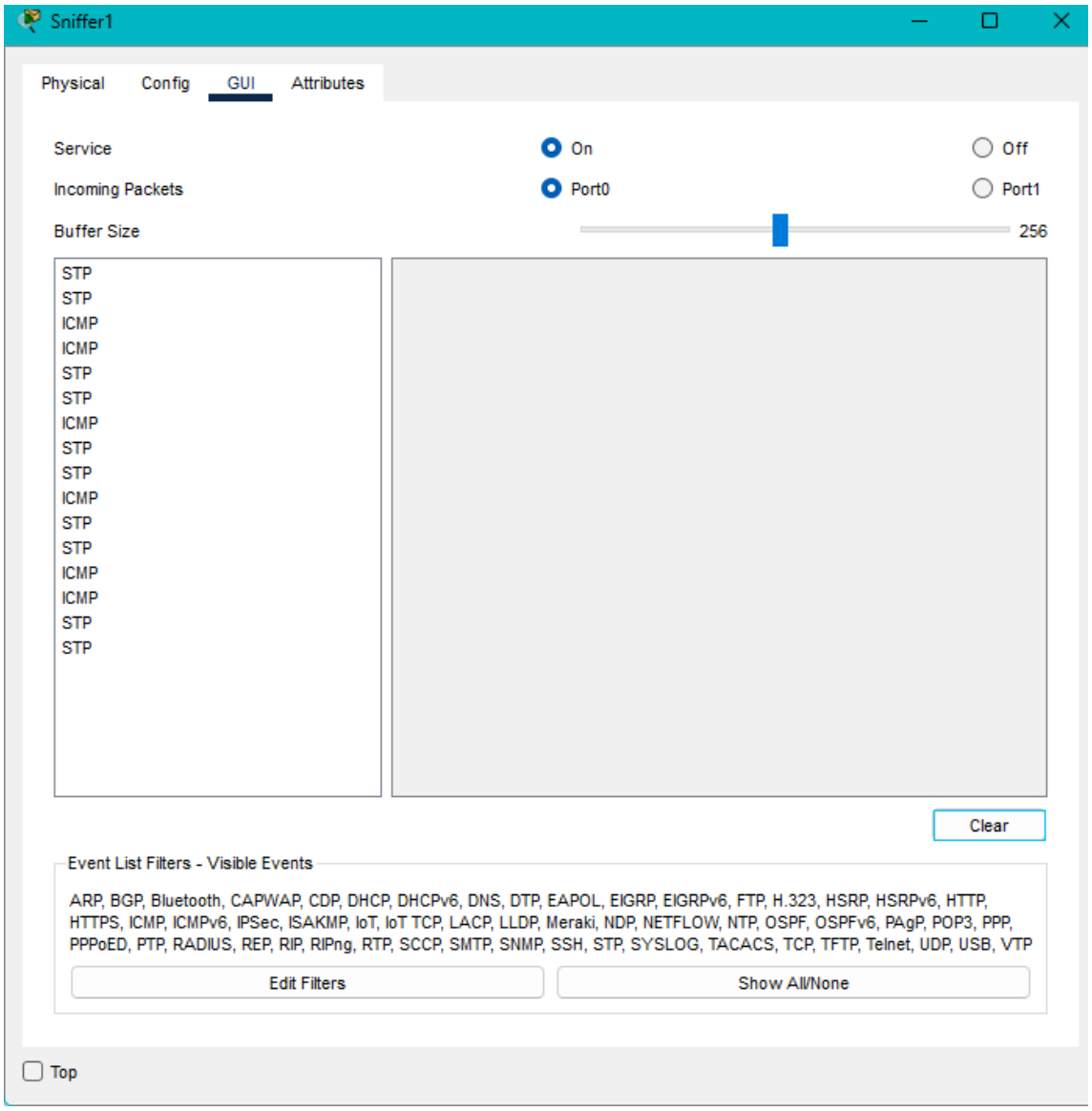


```
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time=16ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128
Reply from 200.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 200.0.0.2:
    Packets: Sent = 100, Received = 100, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 16ms, Average = 0ms

C:\>
```



```
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time=3ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128

Ping statistics for 200.0.0.8:
    Packets: Sent = 100, Received = 100, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 15ms, Average = 0ms

C:\>ping -n 100 200.0.0.8

Pinging 200.0.0.8 with 32 bytes of data:

Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time=3ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
Reply from 200.0.0.8: bytes=32 time<1ms TTL=128
```

Integrantes: Martina Nahman y Emiliano Germani



Actividad 2: LAN Virtuales (VLANs)

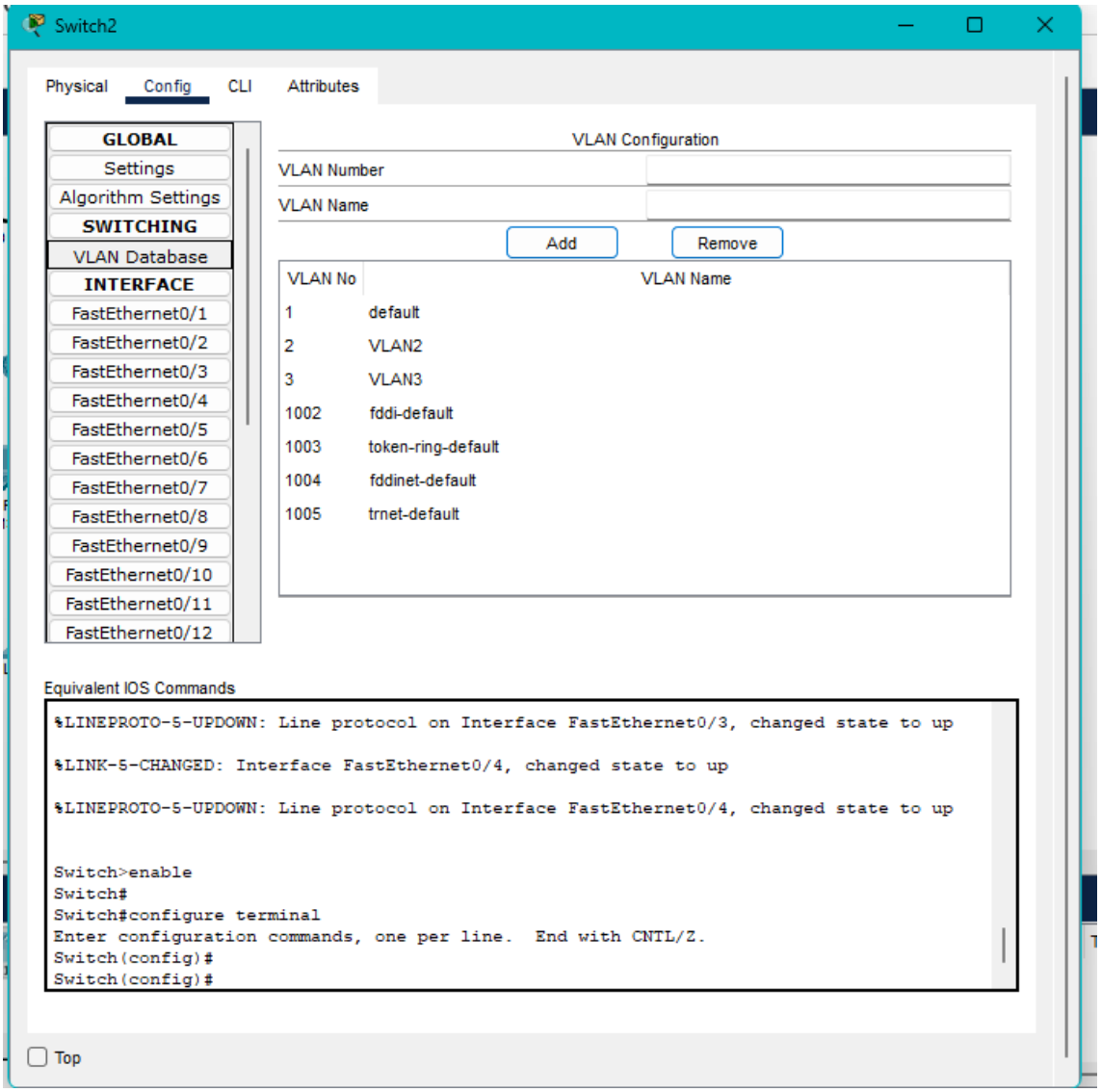
Se modificó la red anterior para dividirla en **dos VLANs**, con los siguientes criterios:

- **VLAN1**: laptops 1 al 6 y 13
- **VLAN2**: laptops 7 al 12 y 14

Para configurar las VLANs, se accedió al menú **"VLAN Database"** de cada switch y se asignaron las laptops a los puertos correspondientes utilizando el modo **Access**. En caso de necesitar conexión entre VLANs en un mismo puerto, se utilizó el modo **Trunk**.

Se verificó la conectividad **intra-VLAN** mediante ping entre laptops del mismo grupo y se comprobó la imposibilidad de comunicación entre dispositivos de diferentes VLANs, confirmando el aislamiento.

Integrantes: Martina Nahman y Emiliano Germani



Actividad 3: Colisiones

Para analizar el comportamiento de las tramas y detectar posibles colisiones, se utilizó el **modo simulación** de Packet Tracer.

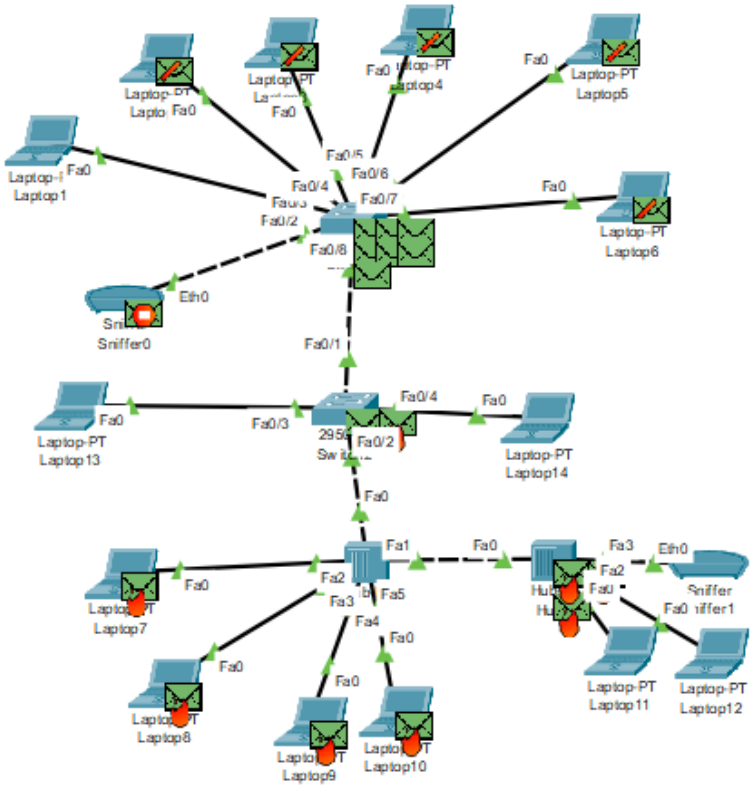
Se enviaron paquetes de difusión (ping -n 1 200.0.0.255) desde:

- Laptop13 (a todas las máquinas de **VLAN1**)
- Laptop14 (a todas las máquinas de **VLAN2**)

Se analizaron los paquetes ICMP enviados y las respuestas recibidas. Durante la simulación, se utilizaron filtros para mostrar solo paquetes ICMP, facilitando el análisis. En los resultados se observaron cómo los paquetes se movían dentro de la red, si eran recibidos por los destinos correctos, y si se descartaban o colisionaban en algún tramo de la red.

Se ajustaron las preferencias del simulador para mejorar la visualización, como la limpieza automática de eventos y el filtrado de paquetes mostrados.

Integrantes: Martina Nahman y Emiliano Germani



Las colisiones se producen cuando dos o más dispositivos intentan transmitir datos al mismo tiempo en un mismo medio compartido, generando interferencia y pérdida de datos. Esto es común en redes donde existe un dominio de colisión, como aquellas que usan hubs (dispositivos que replican las señales recibidas en todos sus puertos) o enlaces compartidos sin mecanismos de control de acceso eficientes.

En el simulador, al enviar paquetes de difusión desde una laptop a muchas otras al mismo tiempo, se incrementa la probabilidad de colisión en la red, sobre todo si están conectadas a través de hubs. Al no haber un mecanismo de control como el que ofrecen los switches modernos, los paquetes pueden interferirse entre sí.

Conclusión

Este trabajo práctico permitió consolidar los conocimientos sobre el funcionamiento de la capa de enlace, observando cómo se comportan distintos elementos de red en situaciones de transmisión, encapsulamiento y segmentación lógica mediante VLANs. Se logró verificar el aislamiento de tráfico entre VLANs, el funcionamiento del protocolo STP, la correcta configuración IP, y el uso de herramientas como Packet Tracer y sniffers para visualizar la comunicación en la red.