

## Práctica 2: Evaluación de los mecanismos de QoS en un conmutador Ethernet

### 1- Objetivos

En esta práctica pondremos a prueba los mecanismos de QoS que ofrecen los conmutadores Cisco Catalyst 2950 con Standard Image disponibles en los armarios de prácticas.

Se busca evaluar la flexibilidad del conmutador en cuestión de QoS así como los posibles problemas o limitaciones del mismo.

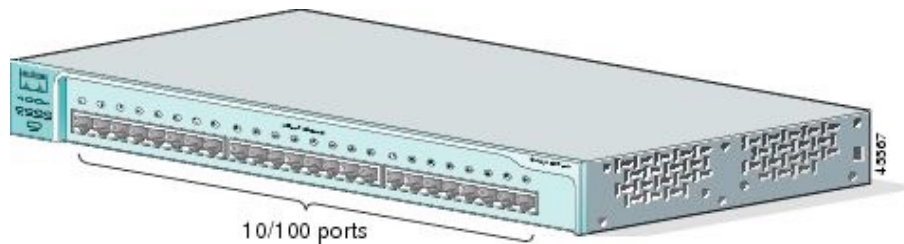


Figura 1- Catalyst 2950-24 (<http://www.cisco.com>)

### 2- Trabajo previo

Repase cómo configurar el nivel IP de los interfaces Ethernet de un PC con Linux.

Revise la documentación de los conmutadores antes de la práctica. Se recomienda revisar lo básico que se vio en prácticas de RO/FTPR sobre acceso al CLI por puerto serie, configuración de velocidad de puertos, etc. A continuación consulte el capítulo sobre QoS (ignore las partes sobre Auto-QoS pues vamos a hacer configuración manual).

Existen dos versiones del software de estos equipos, lo que se llama la *Standard Image* y la *Enhanced Image*. El hardware de los conmutadores es también diferente para soportar una o la otra. Evidentemente, la versión *Enhanced* ofrece más funcionalidades. Los conmutadores de los armarios llevan la versión *Standard*. Eso quiere decir que de la documentación solo nos aplican las secciones que hagan referencia a la misma y no todas las funcionalidades, que solo existen en la versión avanzada. La versión *Standard* no soporta clasificación ni policing ni marcado. Soporta solo queueing y scheduling. Así pues, se reduce en mucho las secciones que aplican y que son las que se recomienda consultar antes de la práctica (es decir, puede ignorar las secciones que dicen que esa funcionalidad solo está disponible en la EI = *Enhanced Image*).

### 3- Escenarios de evaluación

Los objetivos de la práctica van a ser:

- Probar la configuración y funcionamiento del planificador con prioridades (PQ)
- Probar la configuración y funcionamiento del planificador Weighted Round Robin (WRR)
- Combinar ambos planificadores
- Evaluar la implementación de estos planificadores y sus efectos sobre el tráfico según los parámetros configurados y las características del mismo

La topología básica que vamos a emplear en las pruebas es la que se ve en la Figura 2. Habrá varios hosts enviando tráfico (por ejemplo A, B y C) a otro (por ejemplo D) de forma que el planificador que nos interesa es el del Puerto por donde se agrega el tráfico. La clasificación tendremos que hacerla en función del puerto por el que llega el tráfico.

Los planificadores son más útiles en casos de congestión. Para lograr esta congestión se recomienda configurar el puerto del planificador (el del host receptor del flujo) a 10Mbps, de forma que sea sencillo saturarlo enviando tráfico desde puertos a 100Mbps.

Pruebe en primer lugar con flujos UDP con los que puede controlar con sencillez la velocidad a la que envía. Una vez tenga claro el funcionamiento puede probar con flujos TCP.

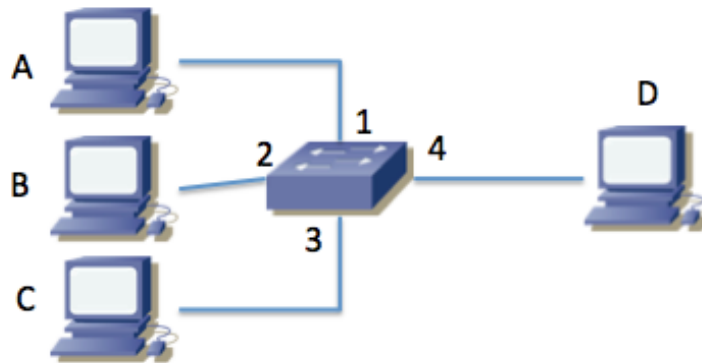


Figura 2 – Topología principal para pruebas

### 3.1- Priority Queueing

Para probar este planificador puede hacerlo con solo dos fuentes, clasificando el tráfico que viene de una como de alta prioridad.

Envíe entre los dos flujos un agregado por debajo de la capacidad del puerto receptor y vea el efecto.

Envíe entre los dos flujos un agregado por encima de la capacidad del puerto pero que cada uno no exceda esa capacidad y vea el efecto. Es decir, por ejemplo si la capacidad del puerto de salida es de 10Mbps podríamos enviar 5Mbps de alta prioridad y 8Mbps de baja prioridad.

En este último escenario, una vez que se llena la cola del puerto del conmutador detenga el flujo del alta prioridad y vea cómo se vacía la cola de baja prioridad. Es decir, como el buffer está lleno de paquetes de baja prioridad, cuando se detenga el flujo de alta prioridad debería enviar todos los paquetes acumulados, a 10Mbps, hasta que vacíe el buffer y pase a reenviar simplemente a los 8Mbps que le llegan. ¿Puede estimar con eso el tamaño del buffer?

**Punto de control:** Muestre al profesor de prácticas el planificador en funcionamiento y describa lo que ha aprendido sobre él

### 3.2- Weighted Round Robin

Para probar este planificador puede hacerlo de nuevo con solo dos fuentes.

Pruebe a enviar un tráfico agregado superior a la capacidad del puerto de salida y varíe los valores de los pesos para ver el efecto de los mismos.

¿Con un par de flujos de 5Mbps y 8Mbps se podría hacer una asignación de pesos que llevara a que el flujo de 5Mbps consiguiera enviar todos los paquetes?

Envíe dos flujos por ejemplo de 10Mbps con una asignación de pesos 1:3. Vea cuál es la asignación resultante. Vea el efecto del funcionamiento del planificador mirando el tráfico en intervalos de tiempo pequeños, de forma que pueda ver cómo envía paquetes de un flujo primero y luego del otro en turnos. A continuación cambie a pesos 10:30. La proporción se mantiene y por lo tanto se debería mantener la capacidad asignada a cada uno en escalas

altas, pero ¿qué cambia en el comportamiento a baja escala?

Analice si el tamaño de los paquetes tiene algún efecto. Por ejemplo envíe dos flujos de 10Mbps y asigne el mismo peso a los dos flujos. A continuación, manteniendo esa configuración, cambie el tamaño de paquete de uno de los dos flujos, por ejemplo a la mitad, manteniendo la tasa en bps (con lo que deberá enviar más paquetes por segundo). Ahora vea si hay cambios en el reparto de capacidad del enlace entre los flujos y explique a qué se debe.

**Punto de control:** Muestre al profesor de prácticas el planificador en funcionamiento y describa lo que ha aprendido sobre él

### 3.3- Weighted Round Robin + Priority Queueing

Combine el tipo de experimentos que ha hecho en los dos apartados anteriores.

Probablemente necesite una tercera fuente. Puede emplear dos interfaces de un mismo PC como fuentes si tiene cuidado para conseguir que el tráfico salga por los dos interfaces (consulte las opciones de `iperf`). Puede hacerlo empleando VLANs.

**Punto de control:** Explique al profesor de prácticas la configuración de red que ha hecho para este apartado (comandos) así como de los resultados obtenidos