UNIVERSITAT POLITÈCNICA DE VALÈNCIA

_ TELECOM ESCUELA TÉCNICA VLC SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN

CALIDAD DE SERVICIO (QoS) CON ROUTERS CISCO

Luis Miguel Sobreviela Blasco

Tutor: José Oscar Romero Martínez

Trabajo Fin de Grado presentado en la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universitat Politècnica de València, para la obtención del Título de Graduado en Ingeniería de Tecnologías y Servicios de Telecomunicación

Curso 2017-18

Valencia, 28 de junio de 2019

Escuela Técnica Superior de Ingeniería de Telecomunicación
Universitat Politècnica de València
Edificio 4D. Camino de Vera, s/n, 46022 Valencia
Tel. +34 96 387 71 90, ext. 77190
www.etsit.upv.es

VLC/ CAMPUS VALENCIA, INTERNATIONAL CAMPUS OF EXCELLENCE

EMAS Gestión ambiental verificada REG.NO. ES-CV- 000030

Agradecimientos

A mi tutor, José Oscar Romero Martínez, por ayudarme a plantear este trabajo, así como por ayudarme a resolver cualquier duda que me ha surgido durante su realización. A mi familia que siempre me ha apoyado durante la realización de mis estudios. A mis amigos que siempre me han animado cuando los estudios me han sometido a una gran presión, permitiéndome seguir adelante. Y a la Universidad Politécnica de Valencia por haberme permitido realizar mis estudios en la Escuela Técnica Superior de Ingeniería de Telecomunicaciones, así como permitirme utilizar sus recursos para la realización de las pruebas de este trabajo.

Resumen

En el presente trabajo expondré un análisis de las configuraciones de Calidad de Servicio que se pueden obtener al utilizar Routers de las series 1800 y 1900 de Cisco. En primer lugar, expondré lo que significa la Calidad de Servicio tanto a nivel técnico, como a nivel de usuario. A continuación, describiré la red que he utilizado para la obtención de los datos y la configuración de los componentes que conforman dicha red. Siguiendo eso explicare las distintas configuraciones de Calidad de Servicio que he usado y las ventajas y desventajas que tienen. Y finalmente usare las configuraciones previamente explicadas para simular el comportamiento de una red congestionada y observar el efecto que la Calidad de Servicio tiene sobre el tráfico que circula por ella.

Resum

En el present treball exposaré un anàlisi de les configuracions de Qualitat de Servici que es poden obtindre a l'utilitzar Routers de les sèries 1800 y 1900 de Cisco. En primer lloc, exposaré el que significa la Qualitat de Servici tant a nivell tècnic, com a nivell de usuari. A continuació descriuré la xarxa que he utilitzat per a l'obtenció de les dades i la configuració dels components que conformen la dita xarxa. Seguint això explicaré les distintes configuracions de Qualitat de Servici que he usat i els avantatges i desavantatges que tenen. I finalment usaré les configuracions prèviament explicades per a similar el comportament d'una xarxa congestionada y observar l'efecte que la Qualitat de Servici té sobre el tràfic que circula per ella.

Abstract

In the preset work I will expose an analysis of the configurations o Quality of Service that can be obtained when using Cisco's 1800 and 1900 series Routers. In first place I will expose what Quality of service means, both at a technical level and as well at user level. Next, I will describe the network I have used to obtain the data and the configuration of the elements that compose said network. Following that I will explain the different Quality of Service configurations I have used, as well as the advantages and disadvantages of each one. Finally, I will use the previously explained configurations to simulate the behavior of a heavily congested network and observe the effect the Quality of Service has on the traffic flowing through it.

Índice

Contenido

Abreviaturas

QoS	Quality of Service
QoE	Quality of Experience
WFQ	Weighted Fair Queueing
CBWFQ	Class-Based Weighted Fair Queuing
CAR	Committed Access Rate
DiffServ	Differentiated Services
LLQ	Low Latency Queueing
RED	Random Early Detect
WRED	Weighted Random Early Detect
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
IP	Internet Protocol
FIFO	First In First Out
FFQ	Fluid Fair Queueing

Capítulo 1. Introducción y Objetivos

1.1 Introducción

Debido al auge de la tecnología que está ocurriendo en todo el mundo, la cantidad de información que circula por las redes del mundo ha alcanzado niveles inimaginados cuando se diseñaron dichas redes. Hoy en día no solo dispone casi todo el mundo de un terminal con el que acceder a Internet, sino que lo normal es tener varios dispositivos que necesiten conectarse para enviar o recibir información. Además, ya no se transmite solo texto por Internet, como cuando la red fue diseñada, sino que ahora se pueden acceder a imágenes, videos, juegos, etc. lo que genera una cantidad de datos circulando por la red mucho mayor que solo con texto. Estas dos cosas hacen que hoy en día Internet sea una red muy congestionada.

A pesar de esa congestión los usuarios quieren que se pueda acceder a todos los recursos de internet lo más rápido posible con la mejor calidad. Si estos parámetros no se cumplen el usuario estará insatisfecho con la experiencia ofrecida y abandonará su uso. La QoE percibida por el usuario es fundamental para que los usuarios puedan disfrutar de los contenidos digitales sin tener que sufrir esperas o calidades bajísimas. Pero debido al inmenso tráfico que circula por la red hoy en día, es imposible ofrecer a los usuarios la QoE que esperan si solo se usa la red de Internet, debido a que al ser una red Best Effort no prioriza los flujos de información más importantes, como los de un video que se está viendo en streaming, frente a los flujos que no requieren un alto nivel de rendimiento, como el envío de un e-mail.

Para ello se ha encontrado una solución. Mediante el uso del QoS para gestionar el tráfico que circula por la red se puede priorizar la llegada de los tráficos que requieren más recursos de la red para ofrecer una mejor QoE a costa de reducir estos recursos para el tráfico que no requiere tanto para seguir ofreciendo una calidad aceptable.

Para saber cómo aplicar la QoS se deben conocer los siguientes requisitos que tiene un flujo de datos: retardo, jitter, tasa de perdías y ancho de banda. El retardo es el tiempo que tarda un paquete en ir desde el servidor al cliente. Es un valor fijo ya que depende de los componentes que atraviese el paquete hasta llegar al destino. El jitter es la variación del retardo entre paquetes consecutivos o de un mismo flujo de datos. El valor del jitter es variable ya que depende del tiempo que haya pasado cada paquete en las colas de cada componente de la red antes de llegar al usuario. La tasa de perdidas indica el porcentaje de paquetes que se han perdido durante la transmisión, debido a errores en los paquetes que haga que sean descarados o por acceder a una cola que esté llena y al no haber sitio en ella se descarte el paquete. Por último, el ancho de banda es la capacidad máxima de tráfico que puede circular por una red. Todos los flujos de la red comparten el mismo ancho de banda, pero no todos requieren el mismo. En una red Best Effort, como Internet, todos los flujos intentarían usar el ancho de banda que necesitaran para transmitir todos sus paquetes, pero si se supera el ancho de banda máximo del enlace se verán obligados a reducir su ancho de banda para que todos puedan seguir enviando paquetes, aunque algunos se pierdan.

La QoS puede gestionar los recursos de la red para que se cumplan los requisitos que necesitan los distintos flujos de datos. Si se cambia la gestión de las colas de los Routers se puede reducir el jitter y la tasa de perdidas, al gestionar mejor el envío de paquetes lo que reduce los paquetes descartados o que tengan que esperar demasiado en una cola. Y si se asigna el ancho de banda de forma fija a los tipos de flujos que circulan por un Router se puede reducir el jitter y las tasas de perdidas, al permitir que los flujos que necesiten mayor ancho de banda puedan disponer de él lo que evitaría que se acumularan paquetes en la cola y obligara a descartar cuando la cola se llenara.

Teniendo en cuenta lo explicado anteriormente, en este trabajo veré cómo se comporta la tasa de pérdidas y el ancho de banda en función de la QoS aplicada a los Routers usados.

1.2 Objetivos

El objetivo de este documento es estudiar las distintas opciones de Calidad de Servicio que ofrecen los Routers de las series 1800 y 1900 de Cisco y analizar los resultados obtenidos al utilizar dichas configuraciones en un Router congestionado. Para realizar dicho objetivo seguiré los siguientes pasos:

- Estudiar las configuraciones de QoS en los Routers de la serie 1800 y 1900 de Cisco
- Plantear un entorno en el que simular una red altamente congestionada mediante un Router y un Switch
- Analizar un generador de tráfico capaz de generar flujos de paquetes en los que se puedan variar los tipos de paquetes, para poder diferenciar su procedencia, así como variar su tamaño para simular tráfico real.
- Configurar las distintas opciones de QoS de forma individual en un Router Cisco y analizar cómo se comporta la red al existir congestión en uno de sus enlaces, comparando además los resultados con los valores obtenidos al no haber QoS configurada en el Router

1.3 Metodología

Para analizar el funcionamiento de las distintas configuraciones de QoS vamos a utilizar un esquema de red en el que se pueda generar tráfico desde varios ordenadores y que circule por una red que contenga un Router de la serie 1800 o 1900 de Cisco hasta llegar a un ordenador que reciba el tráfico. Además, estudiare un generador de tráfico que sea capaz de generar flujos de tráfico y que además pueda capturar el tráfico recibirlo y permita obtener información sobre él. Una vez estudiado, investigare las distintas configuraciones de QoS que proporcionan los Routers de la serie 1800 y 1900 de Cisco, consultando en la bibliografía como realizan la gestión del tráfico. Además, explicare los comandos utilizados para aplicar dichas configuraciones al Router para que se aplica la configuración de QoS. En último lugar hare las pruebas generando el tráfico mediante el generador en los ordenadores y obtendré información de las pérdidas y del ancho de banda recibido por el otro extremo de la red. Analizare la información para comparar el efecto de la configuración de QoS frente a la ausencia de gestión por parte del Router.

Capítulo 2. Escenario de Trabajo

2.1 Materiales

Para la realización de las pruebas he utilizado el material listado a continuación:

- 4 Ordenadores (PCs)
- 1 Router Cisco serie 1800
- 1 Router Cisco serie 1900
- 1 Switch
- 2 Cables de alimentación
- 4 Cables de red directos
- 1 Cable de red cruzado
- 1 Cable de consola
- Generador de tráfico en todos los PCs

2.2 Escenario

Para comprobar las características de QoS tendré que plantear un escenario en el que haya tres PCs que envíen tráfico en una red y otro PC en una red distinta que reciba el tráfico. El tráfico pasará por un Router que sufrirá congestión en el enlace de salida, lo que permitirá aplicar políticas de QoS en el enlace para comprobar su funcionamiento.

La disposición de los materiales descritos anteriormente será la siguiente

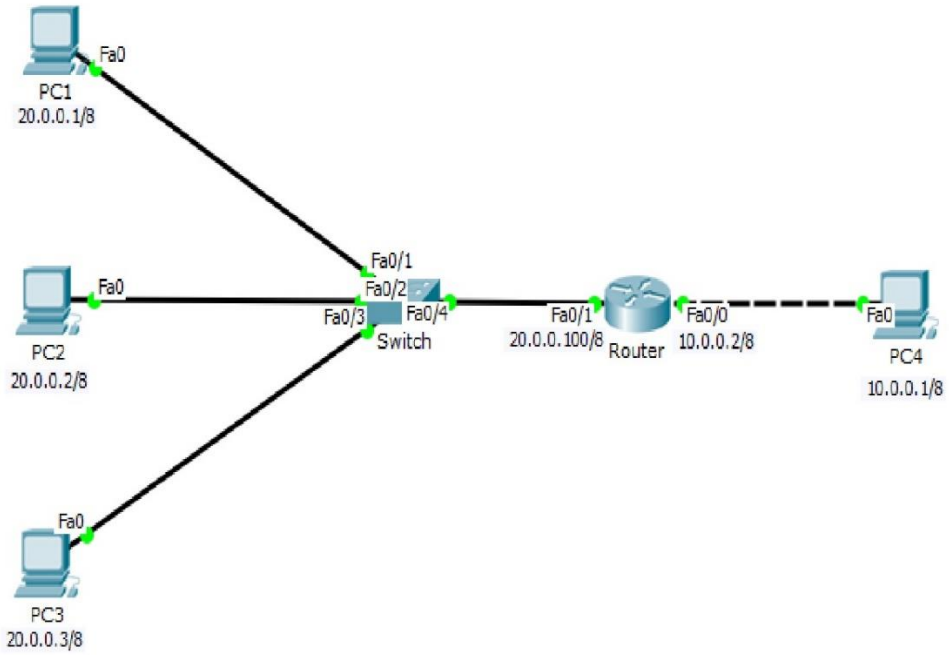


Figura 4.1. Escenario de trabajo

El escenario estará constituido por tres PCs, con las direcciones IP 20.0.0.1, 20.0.0.2 y 20.0.0.3 respectivamente, que generaran cada uno un tráfico de 16 Mbps, obteniendo en total un tráfico de 45 Mbps. El PC1 ira conectado al puerto FastEthernet 0/1 del Switch, el PC2 ira conectado al puerto FastEthernet 0/2 y el PC3 ira conectado al puerto FastEthernet 0/3 mediante cables de red

directos. El Switch se conectará desde su puerto FastEthernet 0/4 a, el puerto FastEthernet 0/1 del Router de la serie 1800 o al puerto GigabitEthernet 0/1 del Router de la serie 1900, que tendrá asignada la dirección IP 20.0.0.100 mediante un cable de red directo. Por último, el Router se conectará con el interfaz FastEthernet o GigabitEthernet 0/0, con la dirección IP 10.0.0.2, al PC4, que tendrá asignada la dirección IP 10.0.0.1, mediante un cable de red cruzado. El PC4 recibirá todo el tráfico generado por los PCs.

	PC1	PC2	PC3	PC4
Dirección IP	20.0.0.1	20.0.0.2	20.0.0.3	10.0.0.1
Mascara	255.0.0.0	255.0.0.0	255.0.0.0	255.0.0.0
Interfaz de Conexión	FE 0/1(S)	FE 0/2(S)	FE 0/3(S)	FE 0/0(R 1800) GE 0/0 (R 1900)

Tabla 1. Configuración de IP, mascara e interfaces de los PCs

	Router 1800	Router 1900
Dirección IP 0/0	10.0.0.2	10.0.0.2
Dirección IP 0/1	20.0.0.100	20.0.0.100
Mascara	255.0.0.0	255.0.0.0
Interfaz de Conexión 0/0	FE 0/0(PC4)	FE 0/0(PC4)
Interfaz de Conexión 0/1	FE 0/4(S)	FE 0/4(S)

Tabla 2. Configuración de IP, mascara e interfaces de los Routers

Sin embargo, con el anterior escenario no se produciría ninguna congestión ya que en todos los enlaces se soporta un ancho de banda de 100 Mbps mientras que los PCs están generando 45 Mbps. Para provocar la congestión se limitará el ancho de banda del interfaz 0/0 de los Routers a 10 Mbps de la siguiente manera

Router# configure terminal

Router(config)# interface [fastethernet | gigabitethernet] 0/0

Router(config-if)# speed 10

Router(config-if)# exit

Router(config)# exit

Router# show interface [fastethernet | gigabitethernet] 0/0

De esta forma se limitará el ancho de banda del interfaz 0/0 a 10 Mbps lo que provocará congestión en el tráfico que se envíe por él.

Teniendo en cuenta que el ancho de banda del Switch es mayor que el tráfico generado por los PCs no hará falta cambiar la configuración del Switch. Así mismo como el Router no aplica ningún tipo de QoS por defecto tampoco habrá que configurar nada.

2.3 Generador de tráfico

El generador de tráfico utilizado para generar los flujos de paquetes se llama **Ostinato**. Este generador permite generar varios flujos de tráfico de varios tipos (TCP, UDP, ICMP). En los flujos se puede configurar parametros como las dirreciones Ip de origen y destino, las dirrecciones MAC de origen y destino, valores de cabecera de los distintos flujosy el contenido de los paquetes.Tambien permite rcibir los paquetes generados desde otro Ostinato y mostrar con el programa Wireshark la informacion de los paquetes recibidos.

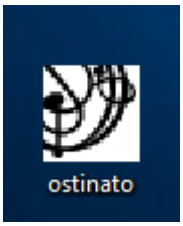


Figura 4.2. Generador Ostinato

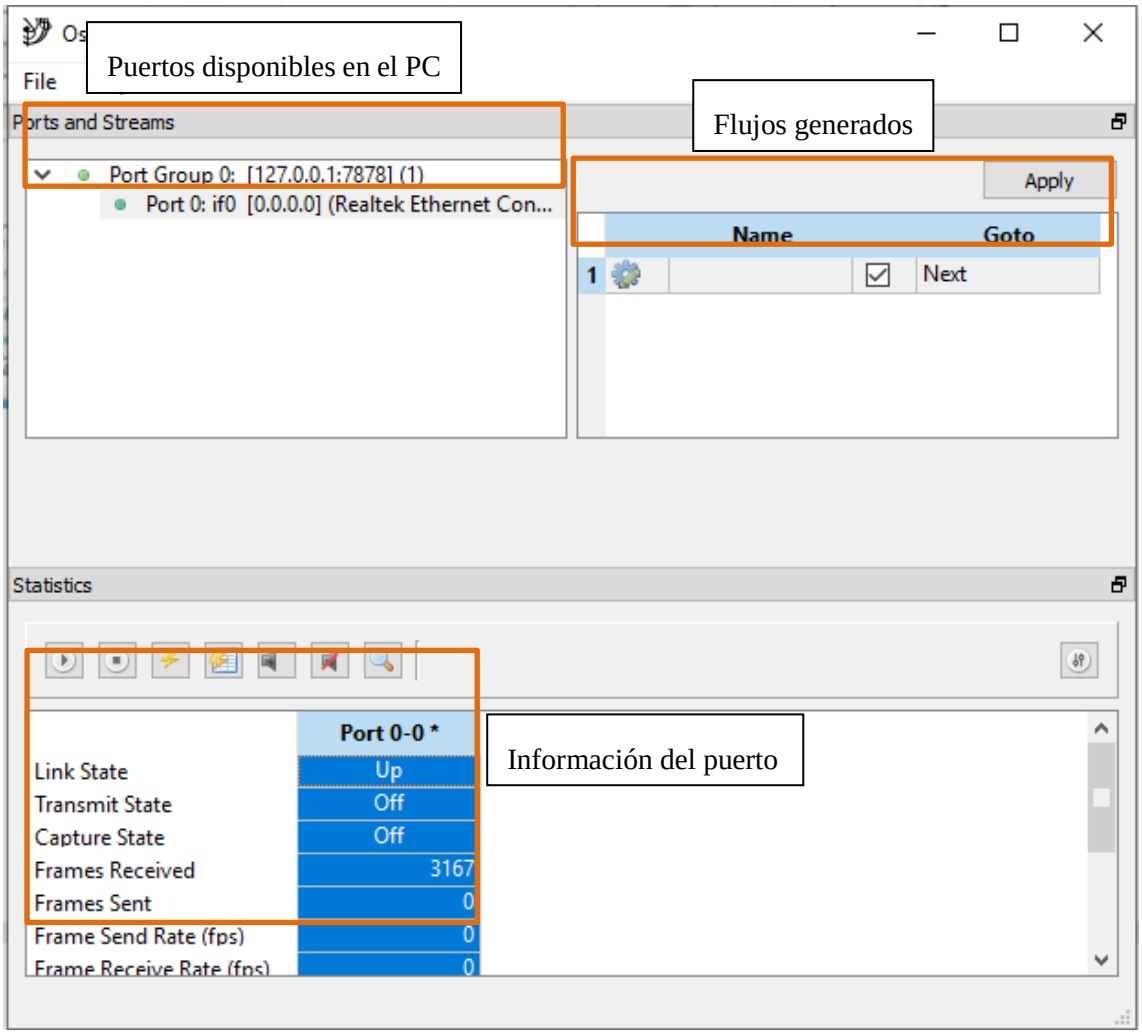


Figura 4.3. Ventana principal

Para configurar los flujos que se van a generar primero se debe seleccionar el puerto por el que se quieren transmitir los flujos. Una vez seleccionado, en la ventana de los flujos se selecciona la opción *New Stream* que aparece al hacer clic con el botón derecho. Una vez hecho aparecerá un flujo en la ventana con un espacio para el nombre, una casilla que indica si el flujo será enviado al iniciar la transmisión y la instrucción que ejecutara el generador cuando finalice la transmisión del flujo, que por defecto es *Next* para iniciar el siguiente flujo o detener el generador si no hay más flujos. Para configurar un flujo hay que seleccionar el icono que aparece antes de la casilla del nombre.

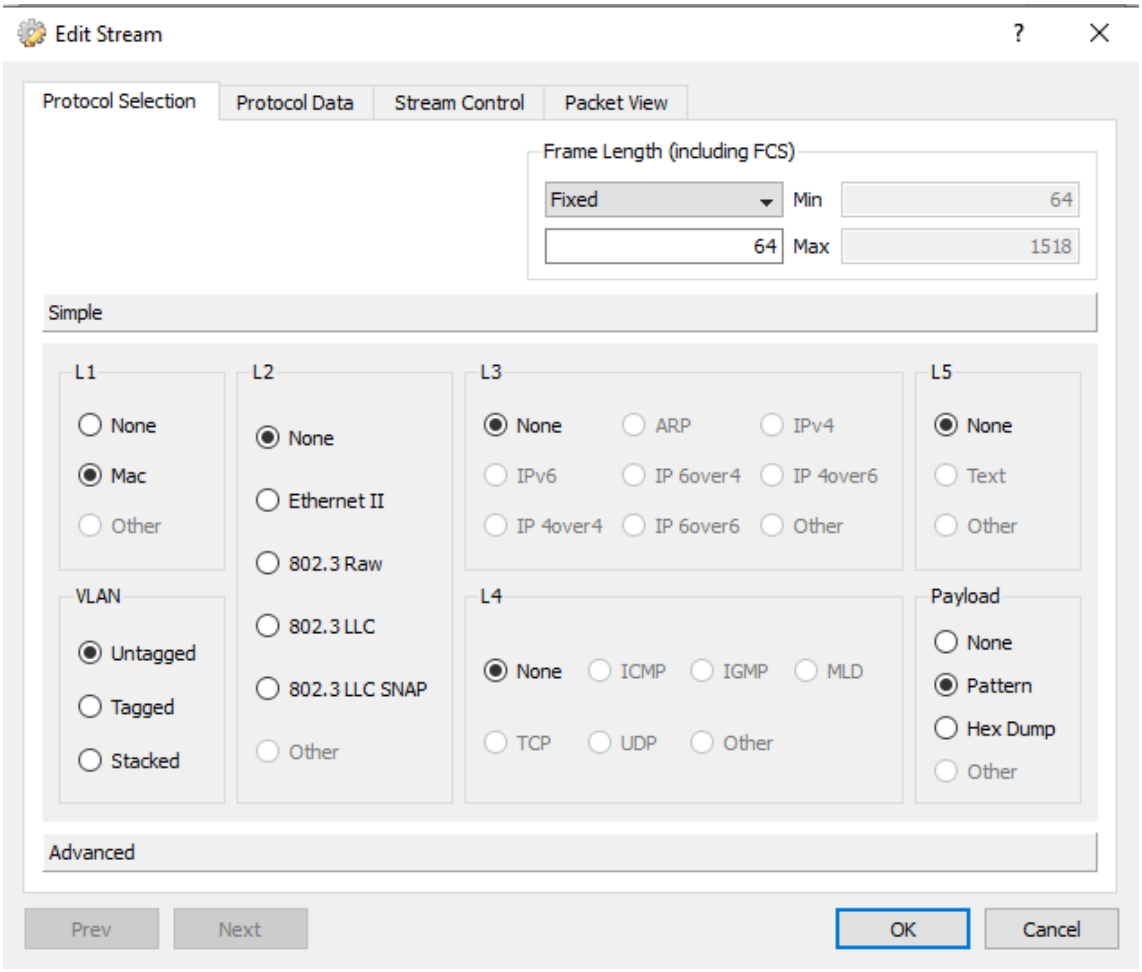


Figura 4.4. Selección del protocolo

En esta pestaña se configura la elección de los protocolos que van a componer los paquetes del flujo elegido. Se puede configurar el protocolo de nivel 2 (Ethernet II. 802.3 Raw, 802.3 LLC, 802.3 SNAP) y una vez elegido permitirá elegir el protocolo de nivel 3 (IPv4, IPv6, ARP). Una vez elegido permitirá seleccionar el protocolo de nivel 4(ICMP, IGMP, TCP, UDP, MLD). Además, se puede configurar el tamaño de los paquetes para que tengan un valor fijo, para que vayan incrementando de un tamaño mínimo a un tamaño máximo, para que vayan decrementando de un tamaño máximo a un tamaño mínimo o que se genere un tamaño aleatorio entre los tamaños indicados como máximo y mínimo. En este caso configurare cada PC de forma que el PC1 envíe tráfico TCP, el PC2 envíe tráfico UDP y el PC3 envíe tráfico ICMP, todos ellos sobre IPv4. Además, configurare el tamaño de los paquetes para que sea aleatorio entre el tamaño mínimo y máximo que soporta cada protocolo para simular tráfico más real.

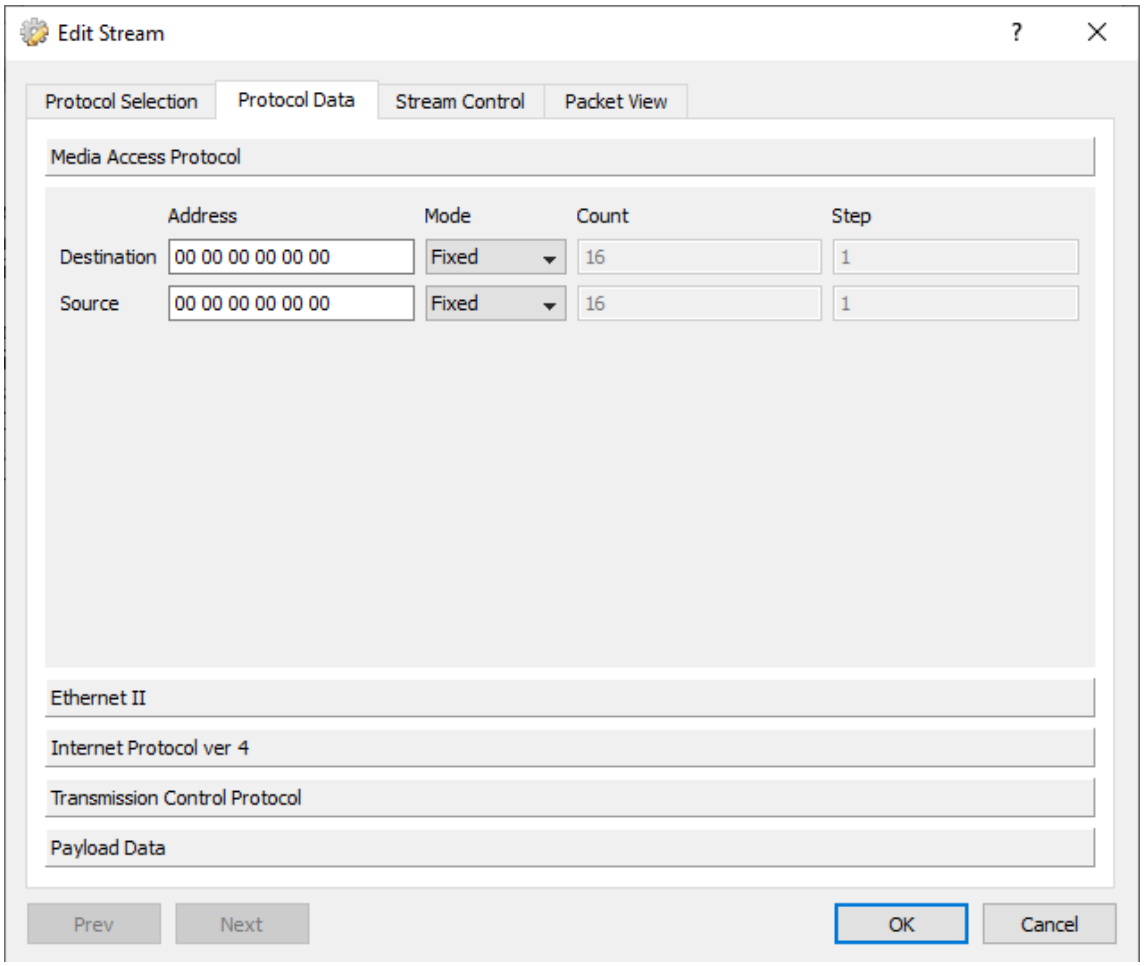


Figura 4.5. Configuración de los protocolos

En esta ventana se pueden configurar los parámetros de las cabeceras de cada uno de los protocolos, así como el contenido del paquete. Hay una pestaña para cada uno de los protocolos que se hayan seleccionado, así como una pestaña para el contenido del paquete. La primera pestaña permite configurar la dirección MAC origen y destino de los paquetes, pudiendo ser direcciones fijas, incrementales o decreméntales. La segunda pestaña permite configurar el Ethernet Type del paquete, que por defecto es el 0x0800. La tercera pestaña permite modificar los campos de la cabecera IP (dirección IP origen, dirección IP destino, valor DSCP, TTL, etc.). La siguiente pestaña permite modificar las cabeceras del protocolo de nivel 4 seleccionado. La última pestaña permite añadir un patrón al contenido del paquete; bien un patrón fijo, un patrón que va incrementando, un patrón que va decrementando o un patrón al azar. El patrón será repetido tantas veces como sea necesario para alcanzar el tamaño del paquete.

En este caso configurare las direcciones MAC de cada PC que genere tráfico para que sean la misma que la del PC y la dirección MAC destino será la dirección MAC del puerto 0/1 del Router. Además, configurare la dirección IP de origen para que sea la de cada PC que genere tráfico y la dirección destino para que sea la dirección del PC4 que recibirá el tráfico generado.

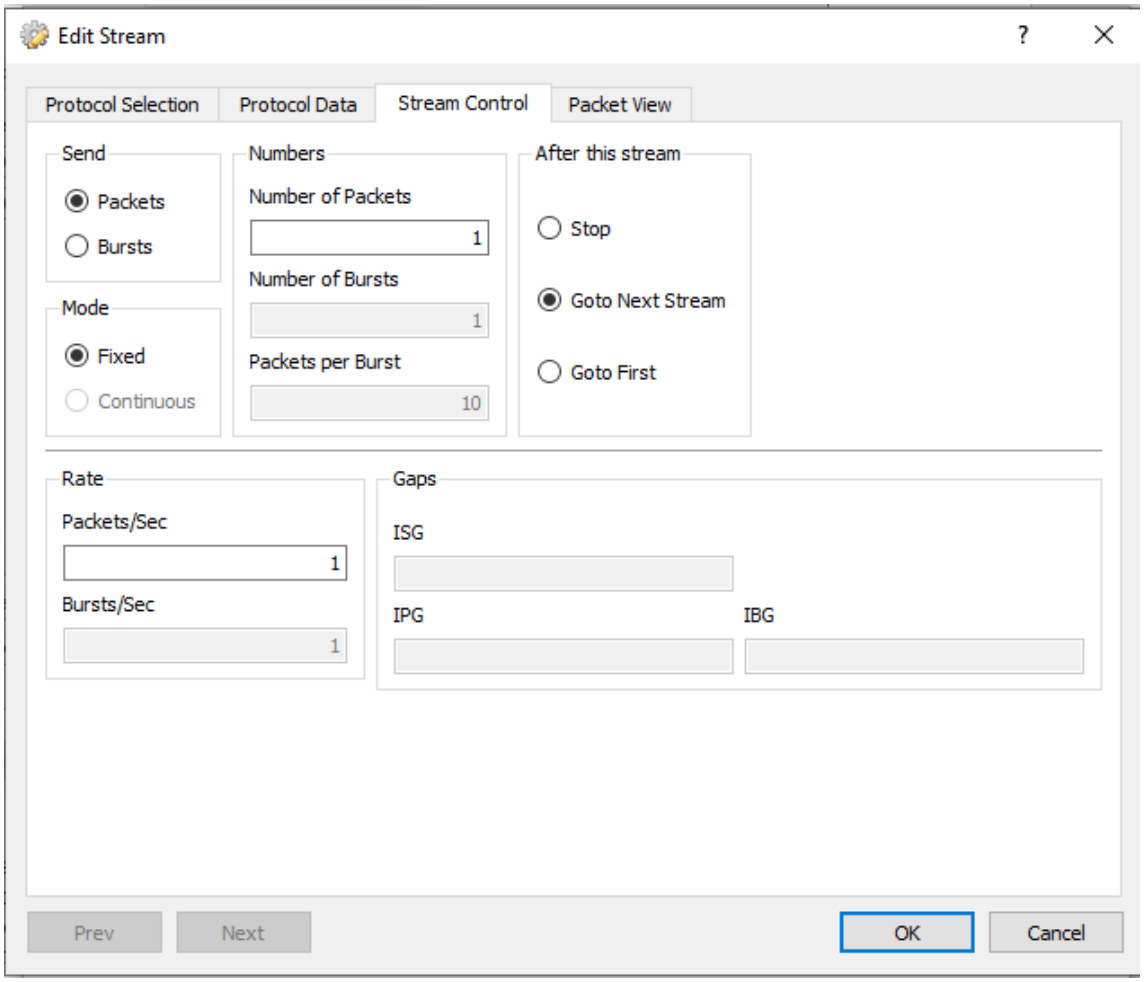


Figura 4.6. Configuración del flujo

En esta ventana se configuran los atributos que tendrá el flujo. En primer lugar, se selecciona si se van a enviar paquetes individuales o ráfagas de paquetes. Después se debe configurar el número de paquetes o ráfagas que se generaran y en caso de haber seleccionado enviar ráfagas, el número de paquetes que contendrá cada ráfaga. Después e configura cuantos paquetes o ráfagas se enviarán por segundo. Por último, se elegirá la instrucción que ejecutará el programa al acabar de enviar el flujo.

Como el programa se bloquea si se le piden valores elevados de tráfico, configurare el generador para que genere 5000 paquetes y los envíe a una tasa de 3000 paquetes por segundo lo que impedirá que el programa se bloquee. Esto debería generar una tasa máxima aproximada de 36 Mbps sin embargo el programa solo puede enviar 1300 paquetes por segundo lo que genera una tasa de aproximadamente 16 Mbps. Debido a que el tráfico generado es muy poco indicare que una vez acabada la transmisión del flujo la instrucción que ejecuto sea *Goto First* lo que hará que vuelva a enviar el mismo flujo indefinidamente hasta que finalice manualmente la transmisión.

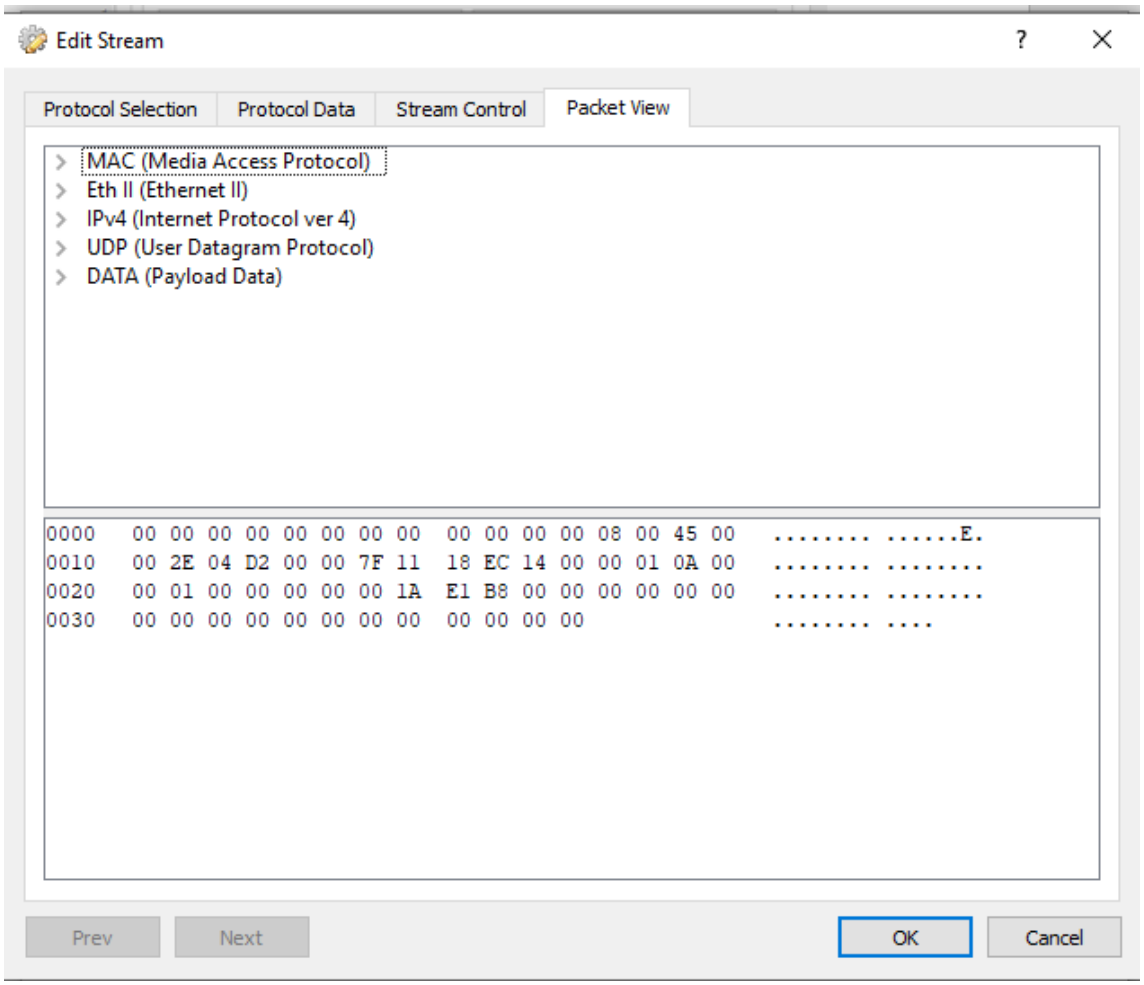


Figura 4.7. Ventana del paquete

Por último, en esta ventana podemos ver los bytes que conforman el paquete que será generado en este flujo. Además, se pueden desplegar cada una de las cabeceras para comprobar la información que transportaran.

2.4 Transmisión y recepción de los paquetes

Para comprender mejor el uso del generador, generare hare una prueba con el Router 1800 sin limitar la velocidad, para comprobar que el tráfico generado por lo PCs sea correcto. Una vez configurados los flujos de los tres PCs vuelvo a la ventana inicial del generador. Ahí estarán los botones para la generación y la recepción de los flujos.

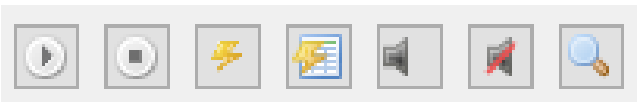
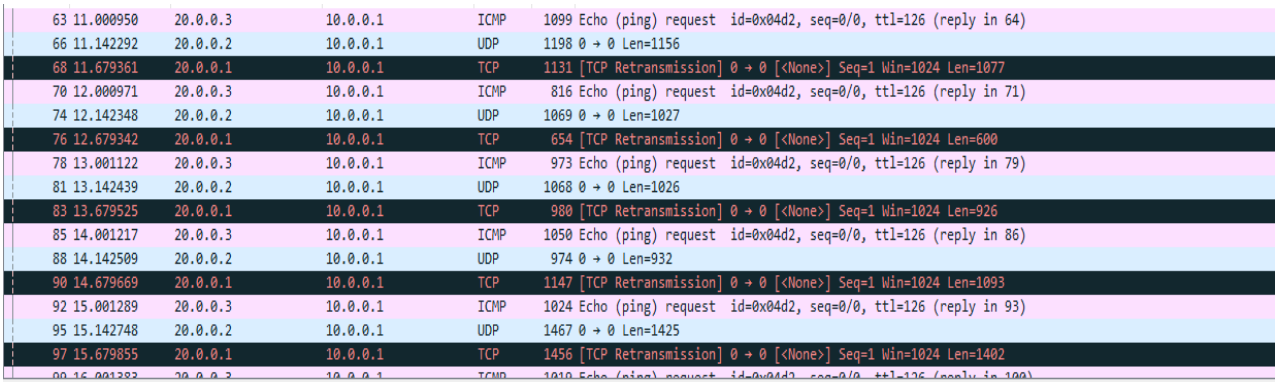


Figura 4.8. Controles de generación y recepción de flujos

Usando estos botones generare los flujos en PC1, PC2 y PC3 y recibiré los flujos en el PC4. El primer botón empieza la transmisión de los flujos configurados, el segundo detiene la transmisión en curso, el tercero borra la información del puerto seleccionado, el cuarto borra la información de todos los puertos, el quinto activa la recepción de paquetes, el sexto detiene la recepción de paquetes y el séptimo abre los paquetes recibidos con el programa Wireshark. Para que funciones se debe elegir el puerto con el que se quiere trabajar en la sección de información del puerto.

Una vez haya habilitado la recepción en el PC4 empezare a generar paquetes en PC1, PC2 y PC3. Después de unos segundos generando paquetes detengo la generación de los paquetes y abro

desde el PC4 los paquetes recibidos. Esta es la imagen obtenida en el momento que los tres PCs están generando tráfico a la vez.



63	11.000950	20.0.0.3	10.0.0.1	ICMP	1099 Echo (ping) request id=0x04d2, seq=0/0, ttl=126 (reply in 64)
66	11.142292	20.0.0.2	10.0.0.1	UDP	1198 0 → 0 Len=1156
68	11.679361	20.0.0.1	10.0.0.1	TCP	1131 [TCP Retransmission] 0 → 0 [<None>] Seq=1 Win=1024 Len=1077
70	12.000971	20.0.0.3	10.0.0.1	ICMP	816 Echo (ping) request id=0x04d2, seq=0/0, ttl=126 (reply in 71)
74	12.142348	20.0.0.2	10.0.0.1	UDP	1069 0 → 0 Len=1027
76	12.679342	20.0.0.1	10.0.0.1	TCP	654 [TCP Retransmission] 0 → 0 [<None>] Seq=1 Win=1024 Len=600
78	13.001122	20.0.0.3	10.0.0.1	ICMP	973 Echo (ping) request id=0x04d2, seq=0/0, ttl=126 (reply in 79)
81	13.142439	20.0.0.2	10.0.0.1	UDP	1068 0 → 0 Len=1026
83	13.679525	20.0.0.1	10.0.0.1	TCP	980 [TCP Retransmission] 0 → 0 [<None>] Seq=1 Win=1024 Len=926
85	14.001217	20.0.0.3	10.0.0.1	ICMP	1050 Echo (ping) request id=0x04d2, seq=0/0, ttl=126 (reply in 86)
88	14.142509	20.0.0.2	10.0.0.1	UDP	974 0 → 0 Len=932
90	14.679669	20.0.0.1	10.0.0.1	TCP	1147 [TCP Retransmission] 0 → 0 [<None>] Seq=1 Win=1024 Len=1093
92	15.001289	20.0.0.3	10.0.0.1	ICMP	1024 Echo (ping) request id=0x04d2, seq=0/0, ttl=126 (reply in 93)
95	15.142748	20.0.0.2	10.0.0.1	UDP	1467 0 → 0 Len=1425
97	15.679855	20.0.0.1	10.0.0.1	TCP	1456 [TCP Retransmission] 0 → 0 [<None>] Seq=1 Win=1024 Len=1402

Figura 4.9. Paquetes generados y recibidos

Se puede ver en la imagen que han llegado paquetes TCP del PC1, paquetes UDP del PC2 y paquetes ICMP del PC3. Además, se puede apreciar que el tamaño de los paquetes no es constante lo cual simula mejor el tráfico real que circula por la red.

Capítulo 3. Desarrollo

En este apartado expondré lo que es la Calidad de Servicio y la importancia que tiene en la gestión del tráfico de una red, explicando además lo que es la Calidad de Experiencia y sus diferencias con la Calidad de Servicio.

Además, expondré las distintas configuraciones de Calidad de Servicio implementadas en los Routers. En el Router de la serie 1800, hablare de configuraciones que gestionan el ancho de banda como WFQ, CBWFQ, LLQ y CAR, además de WRED que gestiona el tratamiento de los descartes de la cola y DiffServ que clasifica y gestiona el tráfico mediante un valor de la cabecera IP. Para cada una de ellas explicare los comandos utilizados en su configuración.

En el Router de la serie 1900, explicare como configurar CBWFQ y WRED, además de hablar de HQoS, que gestiona el ancho de banda a varios niveles, y PBR, que gestiona el tratamiento del tráfico en función del mapa de ruta, explicando además los comandos usados para configurarlos.

3.1 Calidad de Servicio

3.1.1 ¿Qué es?

La Calidad de Servicio o QoS es el conjunto de mecanismos utilizados para diferenciar los distintos flujos de tráfico que circulan por una red y garantizar que una conexión pueda mantener la calidad necesaria para que los usuarios disfruten de un servicio satisfactorio.

QoS permite ofrecer los requisitos necesarios para cada flujo de tráfico manejando los valores que más afectan a las conexiones, ya que no todos los tipos de tráfico tienen las mismas necesidades. Por ejemplo, una video llamada puede seguir funcionando correctamente si hay un ancho de banda no muy elevado, pero si se pierden paquetes la video llamada empezara a mostrar fallos. Por otro lado, la transmisión de un fichero puede soportar perdidas mediante el reenvío de información, pero si se dispone de un ancho de banda muy pequeño la transmisión puede durar demasiado y no ser rentable.

Los parámetros que QoS gestiona son los siguientes:

- Retardo: Es un parámetro que depende del tiempo que tarda un paquete en ser transmitido desde un extremo de la red al otro. Es afectado por la distancia entre el origen y el destino de la transmisión, los elementos que debe atravesar el paquete hasta llegar al destino y otros factores. Debido a que depende de factores físicos es un parámetro que siempre estará presente en todas las transmisiones y cuyo valor no suele variar salvo que haya cambios en la red que atraviesen los paquetes (enlaces rotos, caída de dispositivos, etc.)
- Jitter: Es un parámetro que mide la diferencia entre los tiempos de llegada entre paquetes, ya sean consecutivos o entre todos los paquetes de la misma conexión. Este efecto se debe al tiempo que permanecen los paquetes en la cola de los dispositivos hasta que son procesados y enviados. El valor de este parámetro no es constante y varia de un paquete a otro. Este parámetro afecta mayormente a las conexiones de tiempo real como una video llamada o la transmisión de video en directo. También afecta a los videos por demanda al ser necesario que sean reproducidos a la misma tasa. Este parámetro se puede reducir con una adecuada gestión de las colas en los dispositivos de la red, así como la de las colas de los terminales.
- Ancho de banda: Es un parámetro que indica la capacidad máxima de información que un canal puede transportar. Se mide en Hercios, si el canal es analógico, o en bits por segundo(bps) si el canal es digital. El ancho de banda depende del medio por el que se esté transmitiendo.

- Porcentaje de perdidas: Es un parámetro que indica el porcentaje de paquetes que no han podido ser entregados al destino. Esto se puede deber a descartes de paquetes por colas llenas, paquetes que no pueden ser procesados por los dispositivos de una red o paquetes con errores. Para evitar los descartes por parte de las colas se debe realizar una buena gestión de las colas.

Conociendo estos valores para los distintos flujos de tráfico se pueden utilizar los mecanismos de QoS para optimizar los recursos que la red dedica a cada tipo de tráfico y de esa forma garantizar la mejor recepción en los terminales.

3.1.2 QoE y diferencias con QoS

La Calidad de Experiencia o QoE es el otro parámetro fundamental para saber si una conexión es óptima. A diferencia de la QoS no se basa en medidas de la red sino en la respuesta del usuario al contenido recibido. Esto lo diferencia de QoS al ser un valor subjetivo a cada persona en lugar de un valor fijo que depende de la red por la que circulan los paquetes. Una red puede tener una buena QoS, pero seguir ofreciendo una QoE mala. Por ejemplo, si un usuario intenta acceder a un archivo de video, aunque la red ofrezca la QoS óptima para recibir el video sin esperas, la QoE puede ser mala si el usuario está acostumbrado a ver videos de alta definición y le llega un video de baja definición.

La forma de medir la QoE de una conexión es mediante la escala MOS. Esta escala subjetiva para cada persona se compone de 5 valores que indican la satisfacción del usuario al ver el contenido solicitado a la red.

Puntación	Calidad
5	Excelente
4	Buena
3	Aceptable
2	Pobre
1	Mala

Figura 5.1. Escala MOS

Mediante esta escala un usuario puede definir el nivel de calidad del contenido recibido. Sin embargo, como es una medida subjetiva dos usuarios pueden recibir el mismo contenido y darle valores distintos.

Por lo tanto, la principal diferencia entre QoS y QoE es que QoS utiliza parámetros objetivos para medir la calidad con la que llegan los contenidos al usuario y QoE utiliza valores subjetivos para definir la calidad con la que llegan esos contenidos. Sin embargo, ambas son fundamentales para proporcionar contenidos a una calidad óptima.

3.2 Configuraciones de QoS

En este apartado mostrare las diferentes configuraciones de QoS que he probado en los Routers, así como los comandos necesarios para que los Routers ejecuten las configuraciones. Aunque

algunas configuraciones son aplicables a ambos tipos de Routers hay otras que solo se pueden aplicar en uno de los modelos.

3.2.1 Router Serie 1800

En este apartado se explicarán las configuraciones de QoS usadas en el Router de la serie 1800 de Cisco. Por lo general el Router no tiene aplicada ninguna configuración de QoS por lo que podre configurar directamente las distintas configuraciones. Por defecto el Router tendrá una única cola de salida servida según el modelo FIFO.

3.2.1.1 Weighted Fair Queueing (WFQ)

WFQ es un protocolo diseñada para garantizar un reparto justo entre los diferentes flujos que circulan por un enlace. Fue diseñado para los enlaces de baja velocidad para que pudieran transmitir de forma justa los flujos que les llegaban. Para gestionar los flujos de tráfico que le llegan el Router configura sus propias listas de acceso en las que clasifica el tráfico entrante en función de parámetros de capa 3(dirección IP origen, dirección IP destino) o capa 4(puerto origen, puerto destino, tráfico TCP/UDP). Esto implica que el tráfico que necesita menor ancho de banda saldrá más beneficiado que el que necesite un ancho de banda mayor[1].

Para decidir cómo servir el tráfico se basa en el protocolo FFQ que, a pesar de ser un protocolo imposible de aplicar sobre un enlace físico, ya que depende de poder enviar paquetes parcialmente, se utiliza para calcular el orden teórico en el que se enviaran los paquetes. El protocolo calcula en función del peso de cada cola y del número de colas ocupadas, la cantidad de ancho de banda del enlace que se le debería asignar en un instante de tiempo. En función de los valores obtenidos se calcula cuanto se tardaría en enviar cada paquete y se envían los paquetes en función del tiempo en el que serían enviados completamente.

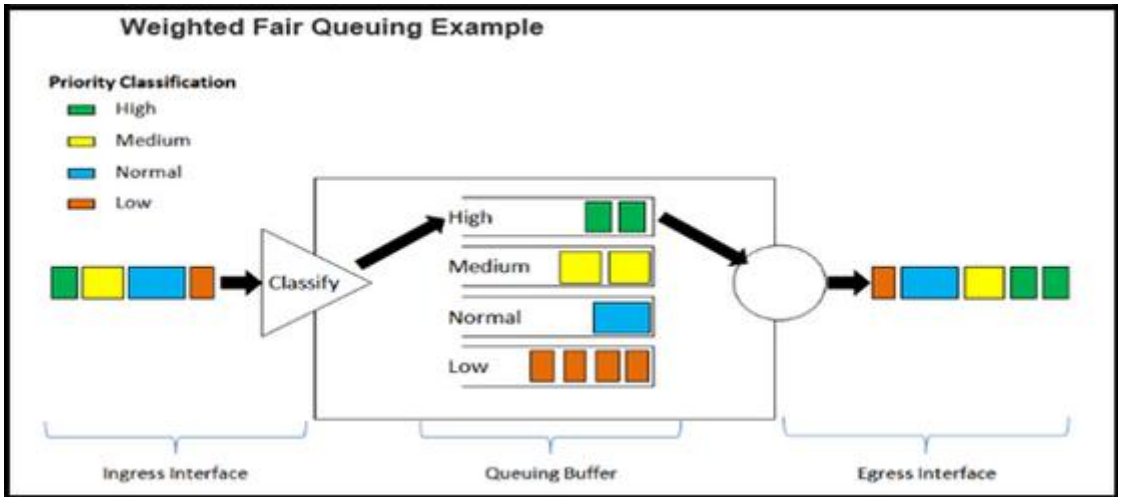


Figura 5.2. Diagrama de funcionamiento de WFQ

Para aplicar esta configuración en el Router hay que introducir los siguientes comandos:

Router# configure terminal

Router(config)# interface fastethernet 0/0

Router(config-if)# fair-queue

Router(config-if)# exit

Router(config)# exit

Una vez configurado WFQ en el interfaz podemos obtener los valores mediante la siguiente instrucción:

Router# show queue fastethernet 0/0

Esta instrucción nos mostrara la siguiente información:

```
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
```

Figura 5.3. Configuración WFQ

El protocolo se ha configurado para tener 1000 colas dinámicas, cada una con una capacidad de 64 paquetes.

3.2.1.2 Class-Based Weighted Fair Queueing (CBWFQ)

CBWFQ extiende la funcionalidad de WFQ para transmitir los paquetes por un enlace, con la salvedad de que las colas son definidas por el usuario. De esta forma se le puede indicar al Router el ancho de banda que se debe adjudicar a cada una de las clases definidas.[2]

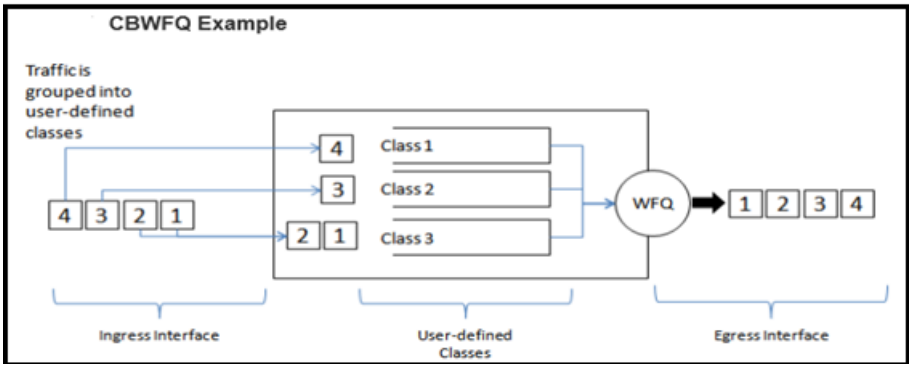


Figura 5.4. Diagrama de funcionamiento de CBWFQ

Para aplicar esta configuración al Router hay que realizar varios pasos:

- Clasificar el tráfico: el tráfico que tenga que ser enviado por el enlace será clasificado mediante listas de acceso en las que se comparara la dirección de origen del paquete para separar los tres flujos que llegaran al Router. Para ello ejecutare los siguientes comandos:
  Router# configure terminal
  Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0
  Router(config)# access-list 10 deny any
  Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0
  Router(config)# access-list 20 deny any
  Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
  Router(config)# access-list 30 deny any
  Router(config)# exit

  De esta forma se clasificará el flujo que provenga de cada PC en una cola distinta.

- Gestionar el ancho de banda: para que cada tráfico sea servido con cierta prioridad se configurara el ancho de banda que se le pueda adjudicar a cada cola, de esa forma la cola con mayor ancho de banda será la cola con mayor prioridad y servirá más paquetes. Para hacerlo definiré una clase para cada tipo de tráfico y una política que indicara el ancho de banda para cada clase con los siguientes comandos:
  Router# configure terminal
  Router(config)# class-map pc1
  Router(config-cmap)# match access-group 10
  Router(config-cmap)# class-map pc2
  Router(config-cmap)# match access-group 20
  Router(config-cmap)# class-map pc3
  Router(config-cmap)# match access-group 30
  Router(config-cmap)# policy-map cbwfq

Router(config-pmap)# class pc1
Router(config-pmap-c)# bandwidth percent 50
Router(config-pmap-c)# class pc2
Router(config-pmap-c)# bandwidth percent 20
Router(config-pmap-c)# class pc3
Router(config-pmap-c)# bandwidth percent 5
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# exit

De esta forma se asigna el 50 por ciento del ancho de banda al flujo que venga del PC1, el 20 al flujo que venga del PC2 y el 5 al flujo que venga del PC3. El Router reserve automáticamente el 25 por ciento del ancho de banda para mensajes de control y enrutamiento por lo que el máximo de ancho de banda disponible para asignar es del 75 por ciento.

- Asignación de la política: para que el tráfico sea gestionado con la configuración indicada se debe asignar la política al interfaz por donde van a circular los flujos. En este case asignare la política al interfaz FastEthernet 0/0 en el sentido de salida, para que se aplique CBWFQ a los paquetes que salgan por ese interfaz. Para ello ejecutare los siguientes comandos:

  Router# configure terminal
  Router(config)# interface fastethernet 0/0
  Router(config-if)# service-policy output cbwfq
  Router(config-if)# exit
  Router(config)# exit

  De esta forma todo el tráfico que salga por el interfaz será clasificado en las colas y servido según el ancho de banda indicado.

Una vez configurado el mecanismo comprobare que la configuración es correcta mediante el siguiente comando:

Router# show policy-map interface fastethernet 0/0

El resultado del comando será el siguiente

```
FastEthernet0/0

 Service-policy output: cbwfq

   Class-map: pc1 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 10
     Queueing
       Output Queue: Conversation 265
       Bandwidth 50 (%)
       Bandwidth 5000 (kbps)Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0

   Class-map: pc2 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 20
     Queueing
       Output Queue: Conversation 266
       Bandwidth 20 (%)
       Bandwidth 2000 (kbps)Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0

   Class-map: pc3 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 30
     Queueing
       Output Queue: Conversation 267
       Bandwidth 5 (%)
       Bandwidth 500 (kbps)Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0
```

Figura 5.5. Configuración en el Router de CBWFQ

Según la configuración aplicada, cada cola tiene asignado el ancho de banda porcentual que ha sido indicado con los comandos. El tráfico del PC1 tiene asignado un ancho de banda del 50 por ciento (5 Mbps), el tráfico de PC2 tiene asignado un ancho de banda del 20 por ciento (2 Mbps) y el tráfico del PC3 tiene asignado un ancho de banda del 5 por ciento(0.5Mbps). Cada cola tiene una capacidad por defecto de 64 paquetes.

3.2.1.3 Low Latency Queueing (LLQ)

LLQ es una evolución de CBWFQ. Fue diseñado para garantizar una prioridad sin tener que depender del peso asignado a cada cola como con CBWFQ [3]. De esta forma se puede asignar a una cola una prioridad más elevada para que no tenga que competir con las demás colas configuradas con CBWFQ.

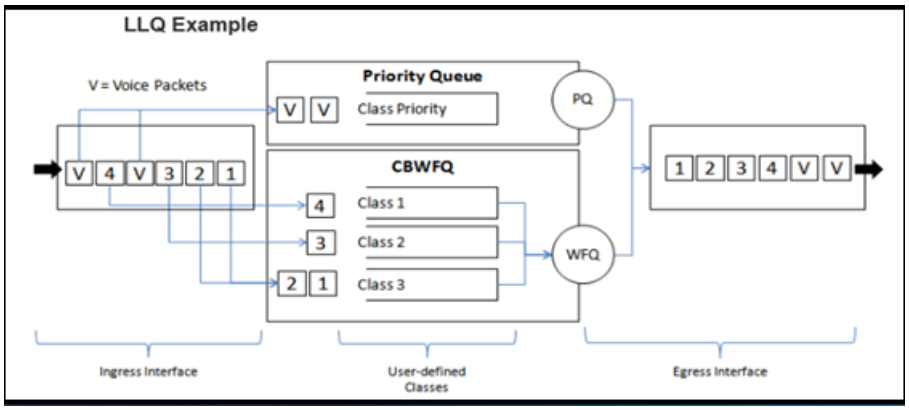


Figura 5.6. Diagrama de funcionamiento de LLQ

Para aplicar esta configuración al Router hay que seguir los mismos pasos que con CBWFQ, pero cambiando la configuración de la cola que tenga el tráfico más prioritario:

- Clasificar el tráfico entrante: el tráfico que tenga que ser enviado por el enlace será clasificado mediante listas de acceso en las que se comparara la dirección de origen del paquete para separar los tres flujos que llegaran al Router. Para ello ejecutare los siguientes comandos:
  Router# configure terminal
  Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0
  Router(config)# access-list 10 deny any
  Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0
  Router(config)# access-list 20 deny any
  Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
  Router(config)# access-list 30 deny any
  Router(config)# exit

  De esta forma se clasificará el flujo que provenga de cada PC en una cola distinta.

- Gestionar el ancho de banda: para el tráfico que proceda del PC1 configurare la cola con prioridad. Después configurare las colas del tráfico de PC2 y PC3 para que se repartan el tráfico mediante CBWFQ. Los comandos utilizados son los siguientes:
  Router# configure terminal
  Router(config)# class-map pc1
  Router(config-cmap)# match access-group 10
  Router(config-cmap)# class-map pc2
  Router(config-cmap)# match access-group 20
  Router(config-cmap)# class-map pc3
  Router(config-cmap)# match access-group 30
  Router(config-cmap)# policy-map llq
  Router(config-pmap)# class pc1
  Router(config-pmap-c)# priority 5000
  Router(config-pmap-c)# class pc2
  Router(config-pmap-c)# bandwidth 1500
  Router(config-pmap-c)# class pc3
  Router(config-pmap-c)# bandwidth 1000
  Router(config-pmap-c)# exit
  Router(config-pmap)# exit
  Router(config)# exit
  De esta forma se asigna una cola con prioridad alta que tendrá disponible un ancho de banda de 5 Mbps, mientras que a las otras colas se les asignará un ancho de banda de 1,5 y 1 Mbps respectivamente.

- Asignación de la política: para que el tráfico sea gestionado con la configuración indicada se debe asignar la política al interfaz por donde van a circular los flujos. En este case asignare la política al interfaz FastEthernet 0/0 en el sentido de salida, para que se aplique LLQ a los paquetes que salgan por ese interfaz. Para ello ejecutare los siguientes comandos:
  Router# configure terminal
  Router(config)# interface fastethernet 0/0
  Router(config-if)# service-policy output llq
  Router(config-if)# exit
  Router(config)# exit
  De esta forma todo el tráfico que salga por el interfaz será clasificado en las colas. Se servirá primero la cola con prioridad alta y después las demás colas en función del ancho de banda asignado pro CBWFQ.

Una vez configurado el mecanismo comprobare que la configuración es correcta mediante el siguiente comando:

Router# show policy-map interface fastethernet 0/0

El resultado obtenido será el siguiente:

```
FastEthernet0/0

 Service-policy output: llq

   Class-map: pc1 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 10
     Queueing
       Strict Priority
       Output Queue: Conversation 264
       Bandwidth 5000 (kbps) Burst 125000 (Bytes)
       (pkts matched/bytes matched) 0/0
       (total drops/bytes drops) 0/0

   Class-map: pc2 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 20
     Queueing
       Output Queue: Conversation 265
       Bandwidth 1500 (kbps)Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0

   Class-map: pc3 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 30
     Queueing
       Output Queue: Conversation 266
       Bandwidth 1000 (kbps)Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0
```

Figura 5.7. Configuración en el Router de LLQ

Según la configuración aplicada la cola del PC1 tiene la más alta prioridad por lo que sus paquetes serán servidos siempre antes con un ancho de banda de 5Mbps. De no haber paquetes en la cola del PC1 se servirán paquetes de las otras colas: de la cola del PC2 a 1.5 Mbps y de la cola del PC3 a 1 Mbps.

3.2.1.4 Weighted Random Early Detection (WRED)

WRED es un mecanismo pensado para aprovechar las ventajas del protocolo TCP para evitar el congestionamiento de las redes. Es una extensión del protocolo RED que permite aplicar distintas configuraciones de descarte a cada cola para descartar el tráfico menso prioritario cuando se detecte congestión en el Router[4]. El protocolo define dos umbrales en la cola de la siguiente manera

- Umbral mínimo: cuando llega un paquete a la cola, si este umbral ha sido superado el paquete se almacenará en la cola.
- Umbral máximo: cuando llega un paquete a la cola y se ha superado el umbral mínimo, si no se ha superado este umbral el paquete será descartado con una determinada probabilidad. La probabilidad de descarte aumentara en función de lo llena que este la cola hasta alcanzar el valor del umbral máximo. Al alcanzarlo cualquier paquete que llegue a la cola será descartado.

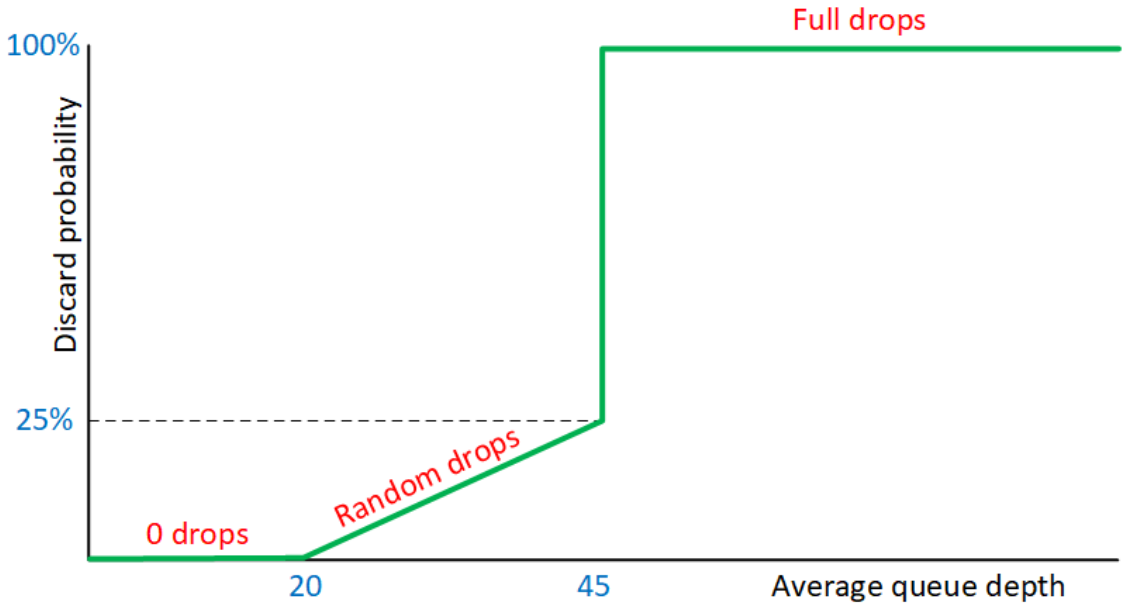


Figura 5.8. Funcionamiento del protocolo WRED con una cola

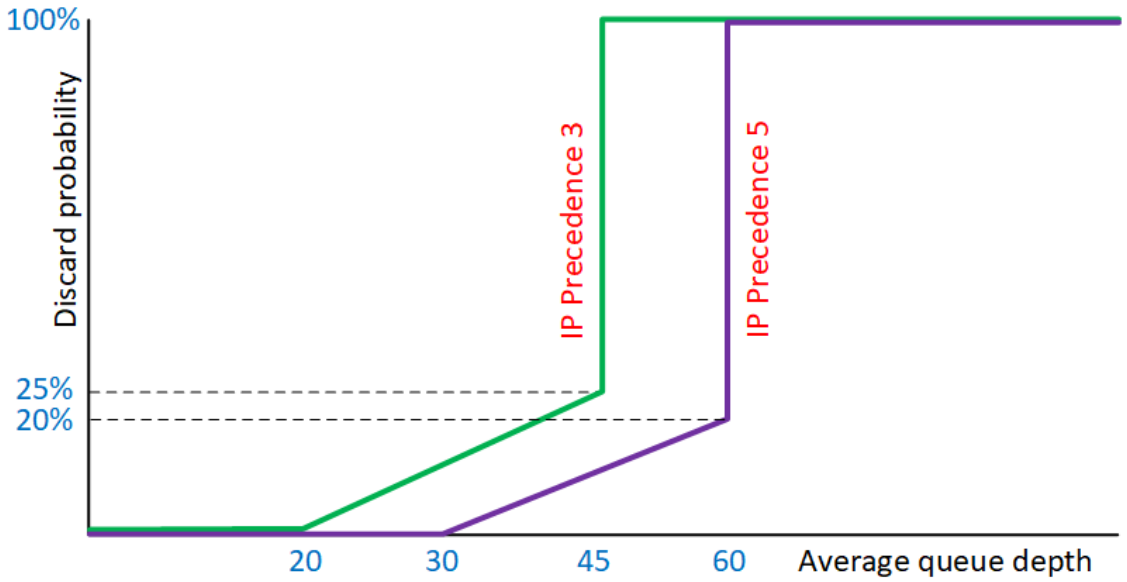


Figura 5.9. Funcionamiento del protocolo WRED con varias colas

En los Routers de la serie 1800 se puede configurar WRED de dos formas:

- Aplicándolo al enlace de salida: de esta forma la cola de salida ejecutara WRED con todo el tráfico que intente salir por el interfaz. Para configurarlo hay que ejecutar las siguientes instrucciones:
  Router# configure terminal
  Router(config)# interface fastethernet 0/0
  Router(config-if)# random-detect
  Router(config-if)# exit
  Router(config)# exit

  Para comprobar que se ha aplicado de forma correcta ejecutare el siguiente comando:
  Router# show queueing random-detect
  El resultado obtenido es el siguiente

```
Current random-detect configuration:
  FastEthernet0/0
    Queueing strategy: random early detection (WRED)
    Random-detect not active on the dialer
    Exp-weight-constant: 9 (1/512)
    Mean queue depth: 0

  class          Random drop      Tail drop    Minimum Maximum  Mark
                  pkts/bytes       pkts/bytes    thresh  thresh  prob
      0      0/0              0/0              20      40  1/10
      1      0/0              0/0              22      40  1/10
      2      0/0              0/0              24      40  1/10
      3      0/0              0/0              26      40  1/10
      4      0/0              0/0              28      40  1/10
      5      0/0              0/0              31      40  1/10
      6      0/0              0/0              33      40  1/10
      7      0/0              0/0              35      40  1/10
   rsvp      0/0              0/0              37      40  1/10
```

Figura 5.10. Configuración en el Router de WRED

Según la configuración, el Router ha generado nueve colas distintas para una prioridad de tráfico distinta cada una con un umbral mínimo distinto, pero todas con un umbral máximo de 40. Una vez se supere el umbral mínimo la probabilidad de descarte ira

aumentando a medida que se almacenen paquetes hasta alcanzar la probabilidad de descarte del 10 por ciento. Al llegar a esta probabilidad, el siguiente paquete que sea almacenado en la cola alcanzara el umbral máximo y todos los demás paquetes que lleguen serán descartados hasta que la cola se vacíe.

- Aplicándolo a cada clase: de esta forma se clasifica el tráfico en distintas colas y se puede aplicar WRED de forma independiente a cada tipo de tráfico. Para configurarlo hay que seguir los siguientes pasos:
  - Clasificar el tráfico: el tráfico que tenga que ser enviado por el enlace será clasificado mediante listas de acceso en las que se comparara la dirección de origen del paquete para separar los tres flujos que llegaran al Router. Para ello ejecutare los siguientes comandos:

    Router# configure terminal
    Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0
    Router(config)# access-list 10 deny any
    Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0
    Router(config)# access-list 20 deny any
    Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
    Router(config)# access-list 30 deny any
    Router(config)# exit

    De esta forma se clasificará el flujo que provenga de cada PC en una cola distinta.

  - Gestionar el ancho de banda: para que cada tráfico sea servido con cierta prioridad se configurara el ancho de banda que se le pueda adjudicar a cada cola, de esa forma la cola con mayor ancho de banda será la cola con mayor prioridad y servirá más paquetes. Para hacerlo definiré una clase para cada tipo de tráfico y una política que indicara el ancho de banda para cada clase con los siguientes comandos:

    Router# configure terminal
    Router(config)# class-map pc1
    Router(config-cmap)# match access-group 10
    Router(config-cmap)# class-map pc2
    Router(config-cmap)# match access-group 20
    Router(config-cmap)# class-map pc3
    Router(config-cmap)# match access-group 30
    Router(config-cmap)# policy-map wred
    Router(config-pmap)# class pc1
    Router(config-pmap-c)# bandwidth percent 50
    Router(config-pmap-c)# random-detect
    Router(config-pmap-c)# class pc2
    Router(config-pmap-c)# bandwidth percent 20
    Router(config-pmap-c)# random-detect
    Router(config-pmap-c)# class pc3
    Router(config-pmap-c)# bandwidth percent 5
    Router(config-pmap-c)# random-detect
    Router(config-pmap-c)# exit
    Router(config-pmap)# exit
    Router(config)# exit

    De esta forma se aplica WRED a cada cola a la que se le ha asignado un flujo de tráfico proveniente de cada uno de los PCs. Para poder ejecutar el comando **random-detect** hay que indicar un ancho de banda que asignar a la clase, de lo contrario no se podrá configurar.
  - Asignación de la política: para que se aplique WRED al tráfico que tenga que salir por el interfaz FastEthernet 0/0 hay que ejecutar las siguientes instrucciones:

Router# configure terminal
Router(config)# interface fastethernet 0/0
Router(config-if)# service-policy output wred
Router(config-if)# exit
Router(config)# exit
De esta forma los paquetes de cada una de las colas serán gestionados mediante el protocolo WRED.

Para comprobar que se ha aplicado correctamente WRED a cada cola ejecutare el siguiente comando:

Router# show policy-map interface fastethernet 0/0

El resultado obtenido es el siguiente:

```
  Class-map: pc1 (match-all)
    0 packets, 0 bytes
    5 minute offered rate 0 bps, drop rate 0 bps
    Match: access-group 10
    Queueing
      Output Queue: Conversation 265
      Bandwidth 50 (%)
      Bandwidth 5000 (kbps)
      (pkts matched/bytes matched) 0/0
      (depth/total drops/no-buffer drops) 0/0/0
       exponential weight: 9
       mean queue depth: 0

class    Transmitted      Random drop      Tail drop    Minimum Maximum  Mark
         pkts/bytes       pkts/bytes       pkts/bytes    thresh  thresh  prob
    0       0/0              0/0              0/0           20      40  1/10
    1       0/0              0/0              0/0           22      40  1/10
    2       0/0              0/0              0/0           24      40  1/10
    3       0/0              0/0              0/0           26      40  1/10
    4       0/0              0/0              0/0           28      40  1/10
    5       0/0              0/0              0/0           30      40  1/10
    6       0/0              0/0              0/0           32      40  1/10
    7       0/0              0/0              0/0           34      40  1/10
 rsvp       0/0              0/0              0/0           36      40  1/10
```

Figura 5.11. Configuración WRED para la clase PC1

```
  Class-map: pc2 (match-all)
    0 packets, 0 bytes                                  |
    5 minute offered rate 0 bps, drop rate 0 bps
    Match: access-group 20
    Queueing
      Output Queue: Conversation 266
      Bandwidth 20 (%)
      Bandwidth 2000 (kbps)
      (pkts matched/bytes matched) 0/0
      (depth/total drops/no-buffer drops) 0/0/0
       exponential weight: 9
       mean queue depth: 0

class    Transmitted      Random drop      Tail drop    Minimum Maximum  Mark
         pkts/bytes       pkts/bytes       pkts/bytes    thresh  thresh  prob
    0       0/0              0/0              0/0           20      40  1/10
    1       0/0              0/0              0/0           22      40  1/10
    2       0/0              0/0              0/0           24      40  1/10
    3       0/0              0/0              0/0           26      40  1/10
    4       0/0              0/0              0/0           28      40  1/10
    5       0/0              0/0              0/0           30      40  1/10
    6       0/0              0/0              0/0           32      40  1/10
    7       0/0              0/0              0/0           34      40  1/10
 rsvp       0/0              0/0              0/0           36      40  1/10
```

Figura 5.12. Configuración WRED para la clase PC2

```
  Class-map: pc3 (match-all)
    0 packets, 0 bytes
    5 minute offered rate 0 bps, drop rate 0 bps
    Match: access-group 30
    Queueing
      Output Queue: Conversation 267
      Bandwidth 5 (%)
      Bandwidth 500 (kbps)
      (pkts matched/bytes matched) 0/0
      (depth/total drops/no-buffer drops) 0/0/0
       exponential weight: 9
       mean queue depth: 0

class    Transmitted      Random drop      Tail drop    Minimum Maximum  Mark
         pkts/bytes       pkts/bytes       pkts/bytes    thresh  thresh  prob
    0       0/0              0/0              0/0           20      40  1/10
    1       0/0              0/0              0/0           22      40  1/10
    2       0/0              0/0              0/0           24      40  1/10
    3       0/0              0/0              0/0           26      40  1/10
    4       0/0              0/0              0/0           28      40  1/10
    5       0/0              0/0              0/0           30      40  1/10
    6       0/0              0/0              0/0           32      40  1/10
    7       0/0              0/0              0/0           34      40  1/10
 rsvp       0/0              0/0              0/0           36      40  1/10
```

Figura 5.13. Configuración WRED para la clase PC3

Según la configuración cada clase tiene ahora nueve colas para distintas prioridades, cada una con un umbral mínimo distinto y un umbral máximo de 40 paquetes. La probabilidad de descarte tras ser superado el umbral mínimo ira aumentando hasta alcanzar una probabilidad de descarte del 10 por ciento. una vez alcanzada cuando se almacene en la cola el siguiente paquete se alcanzará el umbral máximo y los demás paquetes que lleguen a la cola serán descartados hasta que la cola se vacíe. Además, cada clase servirá el tráfico de las colas en función del ancho de banda adjudicado a la clase.: la clase pc1 servirá el tráfico a 5 Mbps, la clase pc2 servirá el tráfico a 2 Mbps y la clase pc3 servirá el tráfico a 0.5 Mbps.

3.2.1.5 Commited Access Rate (CAR)

CAR es una funcionalidad que se utiliza para la optimización y seguridad de las redes. Limita la entrada o salida del tráfico que circula por un interfaz o subinterfaz basándose en varios criterios. Cuando el tráfico alcanza el límite establecido, se especifican las acciones a llevar a cabo[5].

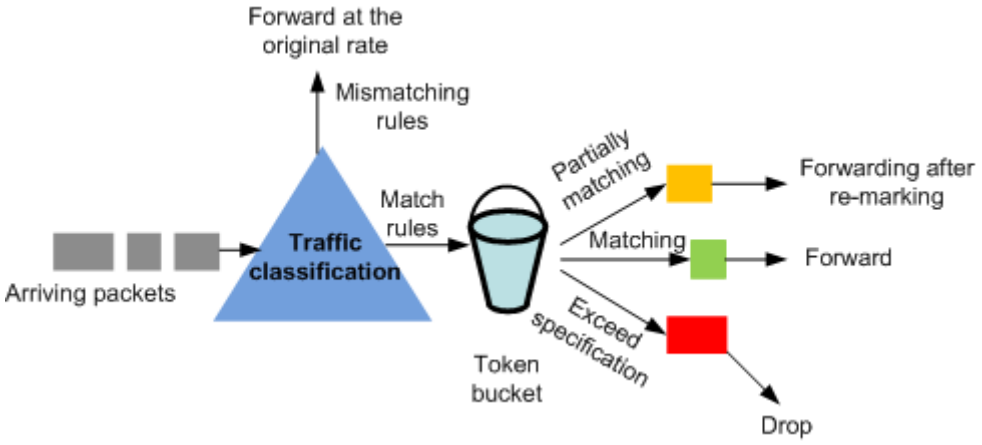


Figura 5.14. Diagrama de funcionalidad de CAR

CAR permite aplicar distintos tipos de tratamiento al tráfico que sea clasificado según sus parámetros. Estas son las distintas funciones que CAR puede aplicar al tráfico y la palabra clave que debe usarse para que el Router las aplique al configurar CAR:

Palabra Clave	Descripción
continue	Pasa a la siguiente configuración CAR sin hacer nada
drop	Descarta los paquetes
set-prec-continue ***newprec***	Cambia el valor del campo de precedencia de la cabecera IP a *newprec* y pasa a la siguiente configuración CAR
set-prec-transmit ***newprec***	Cambia el valor del campo de precedencia de la cabecera IP a *newprec* y transmite los paquetes
transmit	Transmite los paquetes

Tabla 3. Acciones realizables por CAR

Esta funcionalidad puede aplicarse en los Routers de la serie 1800 de dos maneras:

- Aplicándolo al enlace: de esta forma todo el tráfico que circule por el enlace en la dirección en la que se ha configurado, será clasificado según los criterios aplicados y en función de ellos se tratara al paquete de una forma determinada. Esta configuración se aplica introduciendo las siguientes instrucciones:
  Router# configure terminal
  Router(config)# interface fastethernet 0/0
  Router(config-if)# rate-limit output 6000000 3000000 4500000 conform-action transmit exceed-action drop
  Router(config-if)# exit
  Router(config)# exit

  Con esta configuración se limita el ancho de banda del enlace a 6 Mbps. Además, se transmiten todas las ráfagas que no superen los 3 MB de tamaño, se ejecuta la función de **conform-action**(transmitir) si están entre los 3 y 4.5 MB y se ejecuta la función de **exceed-action(**descartar) si superan los 4.5 MB.

  Para comprobar que se ha aplicado correctamente la funcionalidad CAR ejecutare el siguiente comando:
  Router# show interfaces fastethernet 0/0 rate-limit

  El resultado obtenido es el siguiente:

```
FastEthernet0/0
  Output
    matches: all traffic
      params:  6000000 bps, 3000000 limit, 4500000 extended limit
      conformed 0 packets, 0 bytes; action: transmit
      exceeded 0 packets, 0 bytes; action: drop
      last packet: 3430120ms ago, current burst: 0 bytes
      last cleared 00:00:30 ago, conformed 0 bps, exceeded 0 bps
```

Figura 5.15. Configuración en el Router de CAR en el interfaz 0/0

Según la configuración, la capacidad máxima del enlace se limita a 6 Mbps, el tamaño de ráfaga antes de aplicar la primera acción es de 3 MB y el tamaño extendido antes de aplicar la segunda acción es de 4.5 MB.

- Aplicándolo a cada flujo: de esta forma cada flujo puede ser tratado de distinta forma en función del tráfico que transporte. Para aplicar esta configuración hay que seguir los siguientes pasos:
    - Clasificar el tráfico entrante: el tráfico que tenga que ser enviado por el enlace será clasificado mediante listas de acceso en las que se comparara la dirección de origen del paquete para separar los tres flujos que llegaran al Router. Para ello ejecutare los siguientes comandos:

      Router# configure terminal
      Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0
      Router(config)# access-list 10 deny any
      Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0
      Router(config)# access-list 20 deny any
      Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
      Router(config)# access-list 30 deny any
      Router(config)# exit

      De esta forma se clasificará el flujo que provenga de cada PC en una cola distinta.

    - Asignación de CAR a cada flujo: una vez clasificado el tráfico se le aplica a cada lista de acceso una funcionalidad CAR. Para ello se ejecutan los siguientes comandos

      Router# configure terminal
      Router(config)# interface fastethernet 0/0
      Router(config-if)# rate-limit output access-group 10 4000000 2000000 3000000 conform-action transmit exceed-action drop
      Router(config-if)# rate-limit output access-group 20 3000000 1000000 2000000 conform-action transmit exceed-action drop
      Router(config-if)# rate-limit output access-group 30 1500000 500000 1000000 conform-action transmit exceed-action drop
      Router(config-if)# exit
      Router(config)# exit

      De esta forma cada flujo tendrá una funcionalidad CAR distinta. Los paquetes del PC1 tendrán un ancho de banda máximo de 4 Mbps, se transmitirán ráfagas hasta los 2 MB de tamaño, se aplicará el **conform-action** de los 2 a los 3 MB y se aplicará el **exceed-action** a partir de los 3 MB. Los paquetes del PC2 tendrán un ancho de banda máximo de 3 Mbps, se transmitirán ráfagas hasta el 1 MB de tamaño, se aplicará el **conform-action** de los 1 a los 2 MB y se aplicará el **exceed-action** a partir de los 3 MB. Los paquetes del PC3 tendrán un ancho de banda máximo de 1.5 Mbps, se transmitirán ráfagas hasta los 500 kB de tamaño, se aplicará el **conform-action** de los 0.5 al 1.5 MB y se aplicará el **exceed-action** a partir de los 3 MB.

Para comprobar que se ha aplicado correctamente la funcionalidad CAR ejecutare el siguiente comando:
Router# show interfaces fastethernet 0/0 rate-limit

El resultado obtenido es el siguiente:

```
FastEthernet0/0
  Output
    matches: access-group 10
      params:  4000000 bps, 2000000 limit, 3000000 extended limit
      conformed 0 packets, 0 bytes; action: transmit
      exceeded 0 packets, 0 bytes; action: drop
      last packet: 3698936ms ago, current burst: 0 bytes
      last cleared 00:01:00 ago, conformed 0 bps, exceeded 0 bps
    matches: access-group 20
      params:  3000000 bps, 1000000 limit, 2000000 extended limit
      conformed 0 packets, 0 bytes; action: transmit
      exceeded 0 packets, 0 bytes; action: drop
      last packet: 3698936ms ago, current burst: 0 bytes
      last cleared 00:00:46 ago, conformed 0 bps, exceeded 0 bps
    matches: access-group 30
      params:  1496000 bps, 500000 limit, 1000000 extended limit
      conformed 0 packets, 0 bytes; action: transmit
      exceeded 0 packets, 0 bytes; action: drop
      last packet: 3698936ms ago, current burst: 0 bytes
      last cleared 00:00:31 ago, conformed 0 bps, exceeded 0 bps
```

Figura 5.16. Configuración de CAR para cada flujo en el interfaz 0/0

Según la configuración, se ha aplicada a cada flujo que coincida con las listas de acceso un límite distinto de ancho de banda, un límite distinto antes de aplicar la primera acción y otro limite antes de aplicar la segunda acción.

3.2.1.6 Differentiated Services (DiffServ)

DiffServ es un protocolo QoS que permite distinguir diferentes tipos de tráficos mediante el marcado de paquetes. Los paquetes son marcados o etiquetados, permitiendo a los Routers modificar la forma en la que envían los paquetes[6]. Se implementa mediante los 8 bits usados originalmente para definir Type of Service (TOS). Este campo ha sido modificado para convertirse en el campo DSCP usado para clasificar los paquetes con mayor precisión.

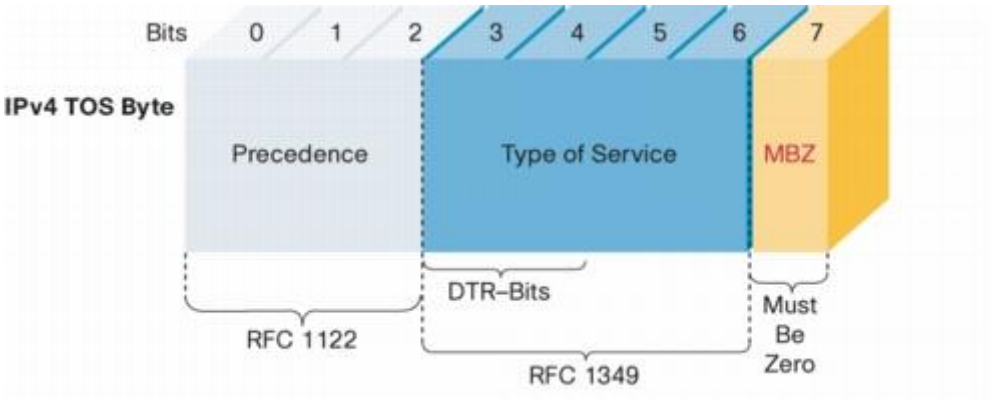


Figura 5.17. Campo ToS

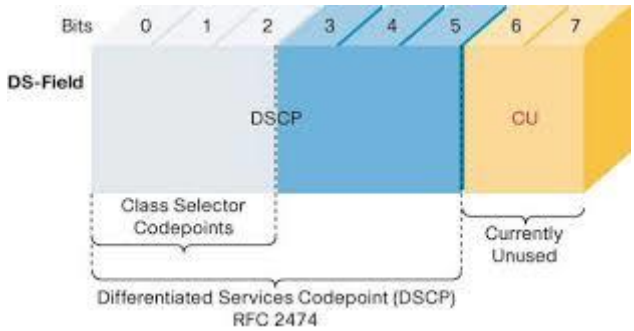


Figura 5.18. Campo DSCP

En el campo ToS se utilizaban los primeros tres bits para indicar la prioridad del paquete, los siguientes cuatro indicaban el tipo de servicio que debía darse al paquete y el ultimo bit siempre estaba a cero. Esto permitía definir solo ocho niveles de prioridad para el tráfico.

En el campo DSCP se utilizan seis bits para indicar la prioridad del paquete permitiendo configurar 64 prioridades distintas. Los últimos dos bits del campo aún no se utilizan. Debido a que el campo DSCP ocupa la misma posición que el campo ToS se definió que los valores cuyos últimos tres bits fueran cero coincidirían con los valores de *precedence* de ToS, permitiendo así la compatibilidad entre los dos campos.

Los valores del campo DSP se pueden clasificar en cuatro grupos:

- Expedited Forwarding (EF): la clase más prioritaria con valor 46 (101110). Esta clase es usada para clasificar el tráfico que necesite unos parámetros de QoS asegurados.
- Assured Forwarding (AF): es la segunda clase más prioritaria. Se puede dividir en cuatro subclases, a las que se les adjudica un ancho de banda distinto en función del tráfico que vaya a circular por ellas. Además, se puede definir cada clase con tres probabilidades de descarte distintas lo que permite tener 12 clases distintas.

Drop	Class 1	Class 2	Class 3	Class 4
Low	001010	010010	011010	100010
	AF11	AF21	AF31	AF41
Medium	001100	010100	011100	100100
	AF12	AF22	AF32	AF42
High	001110	010110	011110	100110
	AF13	AF23	AF33	AF43

Figura 5.19. Valores de AF

- Class Selector (CS): es la clase usada para la compatibilidad con el campo ToS. Tiene ocho valores que coinciden con los de precedence y a los que se le han añadido ceros al final.

DSCP Class Selector Names	Binary DSCP Values	IPP Binary Values
Default/CS0*	000000	000
CS1	001000	001
CS2	010000	010
CS3	011000	011
CS4	100000	100
CS5	101000	101
CS6	110000	110
CS7	111000	111

Figura 5.20. Valores de CS y su correspondencia con ToS

- Default Forwarding (DF): es la clase con la prioridad más baja y la que se aplica por defecto en la generación de paquetes IP. Su valor es de 0(000000).

El mecanismo DiffServ se compone de dos partes:

- Marcado: para poder marcar los paquetes deben clasificar los flujos de tráfico. Para hacerlo ejecutare los siguientes comandos:

```
Router# configure terminal
Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0
Router(config)# access-list 10 deny any
Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0
Router(config)# access-list 20 deny any
Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
Router(config)# access-list 30 deny any
Router(config)# exit
```

Una vez clasificado el tráfico de cada PC se marcará con un valor DSCP distinto con las siguientes instrucciones:

```
Router(config)# class-map pc1
Router(config-cmap)# match access-group 10
Router(config-cmap)# class-map pc2
Router(config-cmap)# match access-group 20
Router(config-cmap)# class-map pc3
Router(config-cmap)# match access-group 30
Router(config-cmap)# policy-map dscp-input
Router(config-pmap)# class pc1
Router(config-pmap-c)# set ip dscp 46
Router(config-pmap-c)# class pc2
Router(config-pmap-c)# set ip dscp 20
Router(config-pmap-c)# class pc3
Router(config-pmap-c)# set ip dscp 26
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# exit
```

De esta forma se aplicará el valor 46 al tráfico del PC1 marcándolo como tráfico EF, el valor 20 al tráfico del PC2 marcándolo como AF22 y el valor 26 al tráfico del PC3 marcándolo como tráfico AF31.

Para comprobar que se ha aplicado correctamente la configuración introduciré la siguiente instrucción:

Router# show policy-map

Este es el resultado obtenido

```
Policy Map dscp-input
  Class pc1
    set ip dscp ef
  Class pc2
    set ip dscp af22
  Class pc3
    set ip dscp af31
```

Figura 5.21. Configuración de la política dscp-input

Se puede comprobar que una vez clasificado el tráfico se marcara el tráfico de PC1 como EF, el tráfico de PC2 como AF22 y el tráfico de PC3 como AF31.

- Gestión: al recibir tráfico marcado se debe indicar como se debe tratar el tráfico en función del valor DSCP que lleve. Para ello primero clasificare los paquetes en función del valor DSCP que lleven mediante los siguientes comandos:

```
Router(config)# class-map premium
```

Router(config-cmap)# match ip dscp 46
Router(config-cmap)# class-map silver
Router(config-cmap)# match ip dscp 20
Router(config-cmap)# class-map bronze
Router(config-cmap)# match ip dscp 26

De esta forma el tráfico se clasifica en función del valor DSCP que tenga. Una vez clasificado se indica cómo tratar el tráfico de la siguiente manera:

Router(config-cmap)# policy-map dscp-output
Router(config-pmap)# class premium
Router(config-pmap-c)# bandwidth percent 50
Router(config-pmap-c)# class silver
Router(config-pmap-c)# bandwidth percent 20
Router(config-pmap-c)# class bronze
Router(config-pmap-c)# bandwidth percent 5
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# exit

De esta forma al tráfico con prioridad EF se le asignara un ancho de banda del 50 por ciento, al tráfico con prioridad AF22 se le asignara un ancho de banda del 20 por ciento y al tráfico que venga con prioridad AF31 se le asignara un ancho de banda del 5 por ciento.

Para comprobar que la configuración se ha aplicado correctamente ejecutare la siguiente instrucción:

Router# show policy-map

Este es el resultado obtenido:

```
Policy Map dscp-output
  Class premium
    Bandwidth 50 (%) Max Threshold 64 (packets)
  Class silver
    Bandwidth 20 (%) Max Threshold 64 (packets)
  Class bronze
    Bandwidth 5 (%) Max Threshold 64 (packets)
```

Figura 5.22. Configuración de la política dscp-output

Se puede comprobar que al tráfico clasificado como Premium se le asignará el 50 por ciento del ancho de banda, al tráfico marcado como silver se le asignará el 20 por ciento el ancho de banda y que al tráfico marcado como bronze se le asignará el 5 por ciento del ancho de banda.

Una vez configuradas las dos partes del mecanismo DSCP ejecutare las siguientes instrucciones para aplicar las políticas a los interfaces del Router:

Router# configure terminal

Router(config)# interface fastethernet 0/0

Router(config-if)# service-policy output dscp-output

Router(config)# interface fastethernet 0/1

Router(config-if)# service-policy input dscp-input

Router(config-if)# exit

Router(config)# exit

Una vez aplicadas las políticas comprobare que se han aplicado correctamente mediante las siguientes instrucciones:

Router# show policy-map interface fastethernet 0/0

Router# show policy-map interface fastethernet 0/1

Los resultados obtenidos son los siguientes:

```
FastEthernet0/0

 Service-policy output: dscp-output

   Class-map: premium (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: ip dscp ef (46)
     Queueing
       Output Queue: Conversation 265
       Bandwidth 50 (%)
       Bandwidth 5000 (kbps)Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0

   Class-map: silver (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: ip dscp af22 (20)
     Queueing
       Output Queue: Conversation 266
       Bandwidth 20 (%)
       Bandwidth 2000 (kbps)Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0

   Class-map: bronze (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: ip dscp af31 (26)
     Queueing
       Output Queue: Conversation 267
       Bandwidth 5 (%)
       Bandwidth 500 (kbps)Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0
```

Figura 5.23. Configuración de la política dscp en el interfaz 0/0

```
FastEthernet0/1

 Service-policy input: dscp-input

   Class-map: pc1 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 10
     QoS Set
       dscp ef
         Packets marked 0

   Class-map: pc2 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 20
     QoS Set
       dscp af22
         Packets marked 0

   Class-map: pc3 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 30
     QoS Set
       dscp af31
         Packets marked 0
```

Figura 5.24. Configuración de la política dscp en el interfaz 0/1

Se puede comprobar que todo el tráfico que entre por el interfaz 0/1 será clasificado en función de su dirección Ip origen y marcado con un valor DSCP, mientras a todo el tráfico que salga por el interfaz 0/0 se le asignara un ancho de banda distinto en función del valor DSCP que tengan los paquetes.

3.2.2 Router serie 1900

En este apartado se explicarán las configuraciones de QoS usadas en el Router de la serie 1800 de Cisco. Por lo general el Router no tiene aplicada ninguna configuración de QoS por lo que podre configurar directamente las distintas configuraciones. Por defecto el Router tendrá una única cola de salida servida según el modelo FIFO.

3.2.2.1 Class-Based Weighted Fair Queueing (CBWFQ)

Al igual que en el modelo de la serie 1800, CBWFQ sirve para clasificar los distintos flujos que circulan por el Router y servirlos de acorde con las especificaciones que el usuario ha implementado. Para configurarlo hay que seguir los mismos pasos que en el modelo de la serie 1800:

- Clasificar el tráfico: el tráfico que tenga que ser enviado por el enlace será clasificado mediante listas de acceso en las que se comparara la dirección de origen del paquete para separar los tres flujos que llegaran al Router. Para ello ejecutare los siguientes comandos:
  Router# configure terminal
  Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0
  Router(config)# access-list 10 deny any
  Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0

Router(config)# access-list 20 deny any
Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
Router(config)# access-list 30 deny any
Router(config)# exit

De esta forma se clasificará el flujo que provenga de cada PC en una cola distinta.

- Gestionar el ancho de banda: para que cada tráfico sea servido con cierta prioridad se configurara el ancho de banda que se le pueda adjudicar a cada cola, de esa forma la cola con mayor ancho de banda será la cola con mayor prioridad y servirá más paquetes. Para hacerlo definiré una clase para cada tipo de tráfico y una política que indicara el ancho de banda para cada clase con los siguientes comandos:
  Router# configure terminal
  Router(config)# class-map pc1
  Router(config-cmap)# match access-group 10
  Router(config-cmap)# class-map pc2
  Router(config-cmap)# match access-group 20
  Router(config-cmap)# class-map pc3
  Router(config-cmap)# match access-group 30
  Router(config-cmap)# policy-map cbwfq
  Router(config-pmap)# class pc1
  Router(config-pmap-c)# bandwidth percent 50
  Router(config-pmap-c)# class pc2
  Router(config-pmap-c)# bandwidth percent 20
  Router(config-pmap-c)# class pc3
  Router(config-pmap-c)# bandwidth percent 5
  Router(config-pmap-c)# exit
  Router(config-pmap)# exit
  Router(config)# exit
  De esta forma se asigna el 50 por ciento del ancho de banda al flujo que venga del PC1, el 20 al flujo que venga del PC2 y el 5 al flujo que venga del PC3. El Router reserve automáticamente el 25 por ciento del ancho de banda para mensajes de control y enrutamiento por lo que el máximo de ancho de banda disponible para asignar es del 75 por ciento.
- Asignación de la política: para que el tráfico sea gestionado con la configuración indicada se debe asignar la política al interfaz por donde van a circular los flujos. En este case asignare la política al interfaz GigabitEthernet 0/0 en el sentido de salida, para que se aplique CBWFQ a los paquetes que salgan por ese interfaz. Para ello ejecutare los siguientes comandos:
  Router# configure terminal
  Router(config)# interface gigabitethernet 0/0
  Router(config-if)# service-policy output cbwfq
  Router(config-if)# exit
  Router(config)# exit
  De esta forma todo el tráfico que salga por el interfaz será clasificado en las colas y servido según el ancho de banda indicado.

Una vez configurado el mecanismo comprobare que la configuración es correcta mediante el siguiente comando:

Router# show policy-map interface gigabitethernet 0/0

El resultado del comando será el siguiente

```
GigabitEthernet0/0

 Service-policy output: cbwfq

   Class-map: pc1 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 10
     Queueing
     queue limit 64 packets
     (queue depth/total drops/no-buffer drops) 0/0/0
     (pkts output/bytes output) 0/0
     bandwidth 50% (5000 kbps)

   Class-map: pc2 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 20
     Queueing
     queue limit 64 packets
     (queue depth/total drops/no-buffer drops) 0/0/0
     (pkts output/bytes output) 0/0
     bandwidth 20% (2000 kbps)

   Class-map: pc3 (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 30
     Queueing
     queue limit 64 packets
     (queue depth/total drops/no-buffer drops) 0/0/0
     (pkts output/bytes output) 0/0
     bandwidth 5% (500 kbps)
```

Figura 5.25. Configuración en el Router de CBWFQ

Según la configuración aplicada, cada cola tiene asignado el ancho de banda porcentual que ha sido indicado con los comandos. El tráfico del PC1 tiene asignado un ancho de banda del 50 por ciento (5 Mbps), el tráfico de PC2 tiene asignado un ancho de banda del 20 por ciento (2 Mbps) y el tráfico del PC3 tiene asignado un ancho de banda del 5 por ciento(0.5Mbps). Cada cola tiene una capacidad por defecto de 64 paquetes.

3.2.2.2 Weighted Random Early Detect (WRED)

WRED en los Routers de la serie 1900 sigue el mismo principio que en los Routers de la serie 1800. Almacena los paquetes en la cola hasta llegar al umbral mínimo. Una vez alcanzado va descartando los paquetes que llegan a la cola con una probabilidad que va aumentando hasta llegar al umbral máximo. Una vez alcanzado el umbral máximo se descartan todos los paquetes que llegan a la cola.

A diferencia de la configuración en el modelo de la serie 1800, WRED no puede configurarse directamente en el enlace de salida. Para configurarlo hay que definirlo en la política que luego se aplicara al enlace de salida. Configurare WRED siguiendo los mismos pasaos que tome con el modelo 1800:

- Clasificar el tráfico: el tráfico que tenga que ser enviado por el enlace será clasificado mediante listas de acceso en las que se comparara la dirección de origen del paquete para separar los tres flujos que llegaran al Router. Para ello ejecutare los siguientes comandos:

  Router# configure terminal
  Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0
  Router(config)# access-list 10 deny any
  Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0
  Router(config)# access-list 20 deny any
  Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
  Router(config)# access-list 30 deny any
  Router(config)# exit

  De esta forma se clasificará el flujo que provenga de cada PC en una cola distinta.

- Gestionar el ancho de banda: para que cada tráfico sea servido con cierta prioridad se configurara el ancho de banda que se le pueda adjudicar a cada cola, de esa forma la cola con mayor ancho de banda será la cola con mayor prioridad y servirá más paquetes. Para hacerlo definiré una clase para cada tipo de tráfico y una política que indicara el ancho de banda para cada clase con los siguientes comandos:

  Router# configure terminal
  Router(config)# class-map pc1
  Router(config-cmap)# match access-group 10
  Router(config-cmap)# class-map pc2
  Router(config-cmap)# match access-group 20
  Router(config-cmap)# class-map pc3
  Router(config-cmap)# match access-group 30
  Router(config-cmap)# policy-map wred
  Router(config-pmap)# class pc1
  Router(config-pmap-c)# bandwidth percent 50
  Router(config-pmap-c)# random-detect
  Router(config-pmap-c)# class pc2
  Router(config-pmap-c)# bandwidth percent 20
  Router(config-pmap-c)# random-detect
  Router(config-pmap-c)# class pc3
  Router(config-pmap-c)# bandwidth percent 5
  Router(config-pmap-c)# random-detect
  Router(config-pmap-c)# exit
  Router(config-pmap)# exit
  Router(config)# exit

  De esta forma se aplica WRED a cada cola a la que se le ha asignado un flujo de tráfico proveniente de cada uno de los PCs. Para poder ejecutar el comando **random-detect** hay que indicar un ancho de banda que asignar a la clase, de lo contrario no se podrá configurar.

- Asignación de la política: para que se aplique WRED al tráfico que tenga que salir por el interfaz GigabitEthernet 0/0 hay que ejecutar las siguientes instrucciones:

  Router# configure terminal
  Router(config)# interface gigabitethernet 0/0
  Router(config-if)# service-policy output wred
  Router(config-if)# exit
  Router(config)# exit

  De esta forma los paquetes de cada una de las colas serán gestionados mediante el protocolo WRED.

Para comprobar que se ha aplicado correctamente WRED a cada cola ejecutare el siguiente comando:

Router# show policy-map interface fastethernet 0/0

El resultado obtenido es el siguiente:

```
    Class-map: pc1 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: access-group 10
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 50% (5000 kbps)
        Exp-weight-constant: 1 (1/2)
        Mean queue depth: 0 packets
        class     Transmitted       Random drop      Tail drop          Minimum
       Maximum     Mark
                  pkts/bytes     pkts/bytes       pkts/bytes          thresh
    thresh     prob

        0                 0/0               0/0               0/0                2
0           40  1/10
        1                 0/0               0/0               0/0                2
2           40  1/10
        2                 0/0               0/0               0/0                2
4           40  1/10
        3                 0/0               0/0               0/0                2
6           40  1/10
        4                 0/0               0/0               0/0                2
8           40  1/10
        5                 0/0               0/0               0/0                3
0           40  1/10
        6                 0/0               0/0               0/0                3
2           40  1/10
        7                 0/0               0/0               0/0                3
4           40  1/10
```

Figura 5.26. Configuración WRED para la clase PC1

```
    Class-map: pc2 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: access-group 20
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 20% (2000 kbps)
        Exp-weight-constant: 9 (1/512)
        Mean queue depth: 0 packets
        class     Transmitted       Random drop      Tail drop          Minimum
       Maximum     Mark
                  pkts/bytes     pkts/bytes       pkts/bytes          thresh
    thresh     prob

        0                 0/0               0/0               0/0                2
0           40  1/10
        1                 0/0               0/0               0/0                2
2           40  1/10
        2                 0/0               0/0               0/0                2
4           40  1/10
        3                 0/0               0/0               0/0                2
6           40  1/10
        4                 0/0               0/0               0/0                2
8           40  1/10
        5                 0/0               0/0               0/0                3
0           40  1/10
        6                 0/0               0/0               0/0                3
2           40  1/10
        7                 0/0               0/0               0/0                3
4           40  1/10
```

Figura 5.27. Configuración WRED para la clase PC2

```
    Class-map: pc3 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: access-group 30
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 5% (500 kbps)
        Exp-weight-constant: 9 (1/512)
        Mean queue depth: 0 packets
        class     Transmitted       Random drop      Tail drop          Minimum
       Maximum     Mark
                   pkts/bytes     pkts/bytes       pkts/bytes          thresh
     thresh     prob

        0                0/0              0/0              0/0                2
0            40  1/10
        1                0/0              0/0              0/0                2
2            40  1/10
        2                0/0              0/0              0/0                2
4            40  1/10
        3                0/0              0/0              0/0                2
6            40  1/10
        4                0/0              0/0              0/0                2
8            40  1/10
        5                0/0              0/0              0/0                3
0            40  1/10
        6                0/0              0/0              0/0                3
2            40  1/10
        7                0/0              0/0              0/0                3
4            40  1/10
```

Figura 5.28. Configuración WRED para la clase PC3

Según la configuración cada clase tiene ahora nueve colas para distintas prioridades, cada una con un umbral mínimo distinto y un umbral máximo de 40 paquetes. La probabilidad de descarte tras ser superado el umbral mínimo ira aumentando hasta alcanzar una probabilidad de descarte del 10 por ciento. una vez alcanzada cuando se almacene en la cola el siguiente paquete se alcanzará el umbral máximo y los demás paquetes que lleguen a la cola serán descartados hasta que la cola se vacíe. Además, cada clase servirá el tráfico de las colas en función del ancho de banda adjudicado a la clase.: la clase pc1 servirá el tráfico a 5 Mbps, la clase pc2 servirá el tráfico a 2 Mbps y la clase pc3 servirá el tráfico a 0.5 Mbps.

3.2.2.3 Hierarchical QoS (HQos)

HQoS es una función que permite controlar los flujos que circulan por la red de forma más granulada que solo con una política de QoS. Esto se realiza mediante la capacidad para definir una política por servicio o una política dentro de un servicio, de forma que se pueda dar la prioridad y el ancho de banda adecuado para diferentes tipos de paquetes o para diferentes flujos[7].

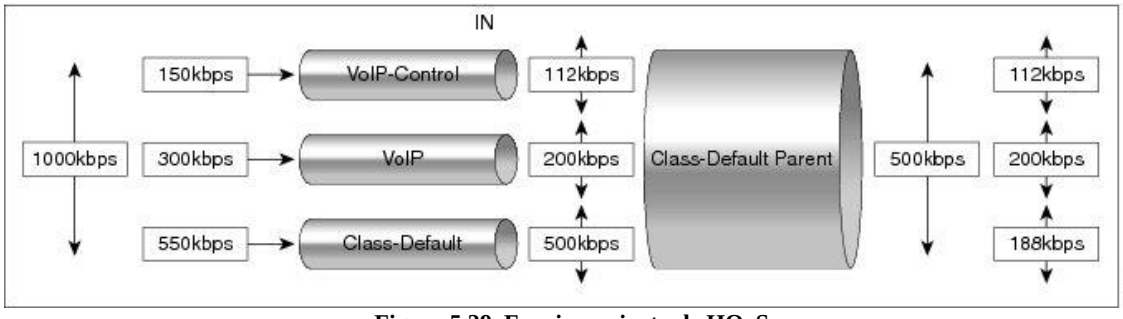


Figura 5.29. Funcionamiento de HQoS

Para configurar HQoS hay que seguir los siguientes pasos:

- Clasificar el tráfico: el tráfico que tenga que ser enviado por el enlace será clasificado mediante listas de acceso en las que se comparara la dirección de origen del paquete para separar los tres flujos que llegaran al Router. Para ello ejecutare los siguientes comandos:

  Router# configure terminal
  Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0
  Router(config)# access-list 10 deny any
  Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0
  Router(config)# access-list 20 deny any
  Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
  Router(config)# access-list 30 deny any
  Router(config)# exit

- Definir las "clases hijo": una vez filtrado el tráfico hay que aplicar una política a cada flujo de tráfico que se ejecutaran primero. Para hacerlo hay que ejecutar las siguientes instrucciones:

  Router# configure terminal
  Router(config)# class-map pc1
  Router(config-cmap)# match access-group 10
  Router(config-cmap)# class-map pc2
  Router(config-cmap)# match access-group 20
  Router(config-cmap)# class-map pc3
  Router(config-cmap)# match access-group 30
  Router(config-cmap)# policy-map child-police
  Router(config-pmap)# class pc1
  Router(config-pmap-c)# police rate 2000000
  Router(config-pmap-c)# class pc2
  Router(config-pmap-c)# police rate 3000000
  Router(config-pmap-c)# class pc3
  Router(config-pmap-c)# police rate 1000000
  Router(config-pmap-c)# exit
  Router(config-pmap)# exit
  Router(config)# exit

  De esta forma se limitará el tráfico que proceda de PC1 a 2 Mbps, el tráfico que proceda de PC2 a 3 Mbps y el tráfico que proceda de PC3 a 1 Mbps.

- Definir la "clase padre": la "clase padre" englobará a la clase hijo y aplicará un límite distinto al enlace al que se aplica, respetando la clase hijo. Para configurarla hay que ejecutar las siguientes instrucciones:

  Router# configure terminal
  Router(config)# policy-map parent-police
  Router(config-pmap)# class class-default
  Router(config-pmap-c)# service-policy child-police
  Router(config-pmap-c)# pólice rate 6000000
  Router(config-pmap-c)# exit
  Router(config-pmap)# exit
  Router(config)# exit

  De esta forma se limitará el ancho de banda del interfaz de salida a 6 Mbps para todo el tráfico evitando que cualquier tráfico pudiera sobrepasar el valor asignado, pero asegurando los valores definidos por la política hijo.

- Asignación de la "clase padre": una vez definida la "clase padre" se asigna al interfaz GigabitEthernet 0/0 de salida mediante las siguientes instrucciones

  Router# configure terminal
  Router(config)# interface gigabitethernet 0/0

Router(config-if)# service-policy output parent-police
Router(config-if)# exit
Router(config)# exit

De esta forma todo el tráfico que salga por el interfaz GigabitEthernet 0/0 será limitado por la "clase hijo" y después por la "clase padre".

Para comprobar que la configuración se ha realizado correctamente ejecutare el siguiente comando:

Router# show policy-map

El resultado obtenido es el siguiente:

```
Policy Map child-police
  Class pc1
   police rate 2000000 bps burst 62500 bytes
     conform-action transmit
     exceed-action drop
  Class pc2
   police rate 3000000 bps burst 93750 bytes
     conform-action transmit
     exceed-action drop
  Class pc3
   police rate 1000000 bps burst 31250 bytes
     conform-action transmit
     exceed-action drop
```

Figura 5.30. Configuración de la "clase hijo"

```
Policy Map parent-police
  Class class-default
   police rate 6000000 bps burst 187500 bytes
     conform-action transmit
     exceed-action drop
    service-policy child-police
```

Figura 5.31. Configuración de la "clase padre"

Se puede comprobar que la política *child-police* limita primero los anchos de banda a los valores indicados, pero permite un burst de tráfico que puede superar el ancho de banda asignado a cada tráfico. Además, la clase parent-police también limita el ancho de banda de todo el tráfico al valor indicado permitiendo también un burst de tráfico que puede superar ese límite.

3.2.2.4 Policy Based Routing (PBR)

PBR es un mecanismo que utiliza los mapas de ruta del Router para clasificar el tráfico que circula por él. PBR puede usarse para seleccionar un flujo de tráfico y aplicarle QoS[8]. De esta forma se puede seleccionar el tráfico entrante del Router y definir funciones de QoS mediante el campo ToS o eligiendo el siguiente salto que el paquete tomara para llegar a su destino, permitiendo una gestión más eficiente de los enlaces para evitar la congestión.

Para configura este mecanismo hay que seguir una serie de pasos:

- Definir las direcciones de origen: para saber los distintos flujos que el mecanismo tiene que tratar hay que definir las direcciones IP de origen de los flujos mediante listas de acceso de la siguiente manera:

  Router# configure terminal
  Router(config)# access-list 10 permit 20.0.0.1 0.0.0.0

```
Router(config)# access-list 20 permit 20.0.0.2 0.0.0.0
Router(config)# access-list 30 permit 20.0.0.3 0.0.0.0
Router(config)# exit
```

De esta manera el mecanismo podrá identificar las direcciones IP de los paquetes que tenga que tratar.

- Definir el mapa de ruta: una vez definidas las direcciones IP de los flujos habrá que crear un mapa de ruta e indicarle como debe tratar a cada flujo mediante las siguientes instrucciones:

```
Router# configure terminal
Router(config)# route map pbr permit 10
Router(config-route-map)# match ip address 10
Router(config-route-map)# set ip precedence 5
Router(config)# route map pbr permit 20
Router(config-route-map)# match ip address 20
Router(config-route-map)# set ip precedence 3
Router(config)# route map pbr permit 30
Router(config-route-map)# match ip address 30
Router(config-route-map)# set ip precedence 1
Router(config-route-map)# exit
Router(config)# exit
```

De esta forma al tráfico cuya dirección IP origen coincida con la de la lista de acceso 10 se le aplicara un valor de prioridad de ToS de 5, a la que coincida con la dirección IP de la lista de acceso 20 de le aplicara una prioridad de 3 y a la que coincida con la dirección IP de la lista de acceso 30 se le aplicara una prioridad de 1.

- Aplicar la ruta al enlace: para que el mecanismo se aplique a los flujos hay que aplicarlo al enlace por el que van a entrar los flujos, en este caso el enlace GigabitEthernet 0/1. Para hacerlo hay que ejecutar las siguientes instrucciones:

```
Router# configure terminal
Router(config)# interface gigabitethernet 0/1
Router(config-if)# ip policy route-map pbr
Router(config-if)# exit
Router(config)# exit
```

De esta forma todo el tráfico entrante por el interfaz 0/1 será gestionado por el mecanismo PBR.

Para comprobar que la configuración se ha realizado correctamente introduciré el siguiente comando:

```
Router# show route-map
```

El resultado obtenido es el siguiente:

```
route-map pbr, permit, sequence 10
  Match clauses:
    ip address (access-lists): 10
  Set clauses:
    ip precedence critical
  Policy routing matches: 0 packets, 0 bytes
route-map pbr, permit, sequence 20
  Match clauses:
    ip address (access-lists): 20
  Set clauses:
    ip precedence flash
  Policy routing matches: 0 packets, 0 bytes
route-map pbr, permit, sequence 30
  Match clauses:
    ip address (access-lists): 30
  Set clauses:
    ip precedence priority
  Policy routing matches: 0 packets, 0 bytes
```

Figura 5.32. Configuración de PBR

Se puede comprobar que cuando la dirección IP coincida con la de la lista de acceso 10 se pondrá el valor de prioridad de ToS a *critical* (5), cuando coincida con la de la lista de acceso 20 se pondrá a *flash* (3) y cuando coincida con la lista de acceso 30 se pondrá a *priority* (1).

Para comprobar cómo podría usarse esta configuración para la gestión del ancho de banda utilizare la siguiente configuración para definir un CBWFQ que dependa de la prioridad y poder simular el funcionamiento del mecanismo:

Router# configure terminal
Router(config)# class-map cinco
Router(config-cmap)# match ip precedence 5
Router(config-cmap)# class-map tres
Router(config-cmap)# match ip precedence 3
Router(config-cmap)# class-map uno
Router(config-cmap)# match ip precedence 1
Router(config-cmap)# policy-map pbr
Router(config-pmap)# class cinco
Router(config-pmap-c)# bandwidth percent 50
Router(config-pmap-c)# class tres
Router(config-pmap-c)# bandwidth percent 20
Router(config-pmap-c)# class uno
Router(config-pmap-c)# bandwidth percent 5
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# exit

Una vez configurada la política de servicio se aplica al interfaz GigabitEthernet 0/0 para que gestione el tráfico de salida de la siguiente manera:

Router# configure terminal
Router(config)# interface gigabitethernet 0/0
Router(config-if)# service-policy output pbr
Router(config-if)# exit
Router(config)# exit

De esta forma se le asignara el 50 por ciento del ancho de banda a todo el tráfico con una prioridad de 5, el 20 por ciento al tráfico con una prioridad de 3 y el 5 por ciento al tráfico con una prioridad de 1.

Capítulo 4. Resultados

En este apartado expondré los resultados obtenidos de configurar el Router del entorno de trabajo con las configuraciones anteriormente descritas para observar cómo se reparte el ancho de banda entre los flujos generados por los PCs, así como viendo el porcentaje de paquetes perdidos de cada flujo. Para la simulación empezará primero a generar tráfico ICMP el PC3, seguido del tráfico UDP generado por el PC2 y por último el tráfico TCP generado por el PC1. Todos los PCs generan el tráfico a una velocidad de 16 Mbps dirigido al PC4 que recibirá los paquetes que puedan pasar por el Router. Una vez acabada la generación de tráfico extraeré la información de los paquetes con la herramienta Wireshark, que usare para analizar el ancho de banda, así como ver cuantos paquetes han sido recibidos de cada flujo.

4.1 Router Serie 1800

Primero expondré los resultados obtenidos con el Router de la serie 1800. Para poder analizar los resultados de forma correcta primero mostrare los resultados obtenidos sin haber aplicado ninguna configuración de QoS al Router. Una vez finalizada la generación de tráfico el ancho de banda que ha llegado al PC4 es el siguiente:

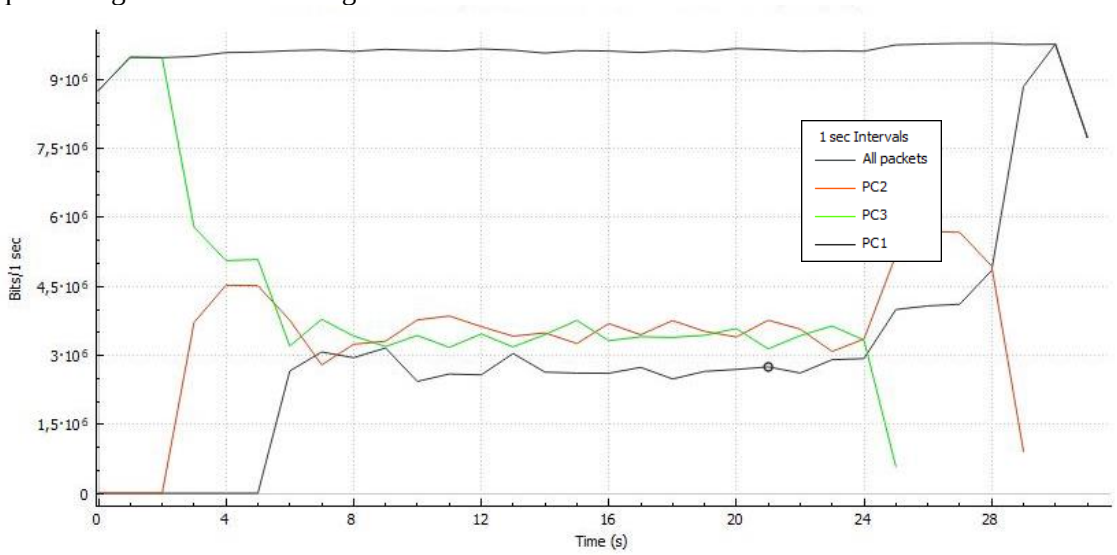


Figura 6.1. Grafica del ancho de banda recibido sin QoS

En la figura 6.1 se puede ver que el valor del ancho de banda recibido no supera los 10 Mbps máximos que tiene el enlace. Además, se puede ver que hasta que mientras el PC3 era el único que generaba tráfico utilizaba todo el ancho de banda disponible. Una vez empezaban a transmitir los otros PCs, el ancho de banda se repartía entre los tres flujos de tráfico, favoreciendo a los tráficos de ICMP y UDP sobre el tráfico TCP.

La siguiente tabla muestra los paquetes enviados y recibidos sin el uso de QoS:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	34130	34551	32902
Recibidos	11676	10809	11626
Porcentaje de perdidas	78,22%	79,64%	78,67%

Tabla 4. Porcentaje de perdidas sin QoS.

Se puede apreciar que, debido a la alta congestión, los tres flujos han sufrido grandes pérdidas, pero no ha habido grandes diferencias entre las pérdidas de cada flujo.

4.1.1 Weighted Fair Queueing (WFQ)

Una vez aplicada la configuración de WFQ al Router empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

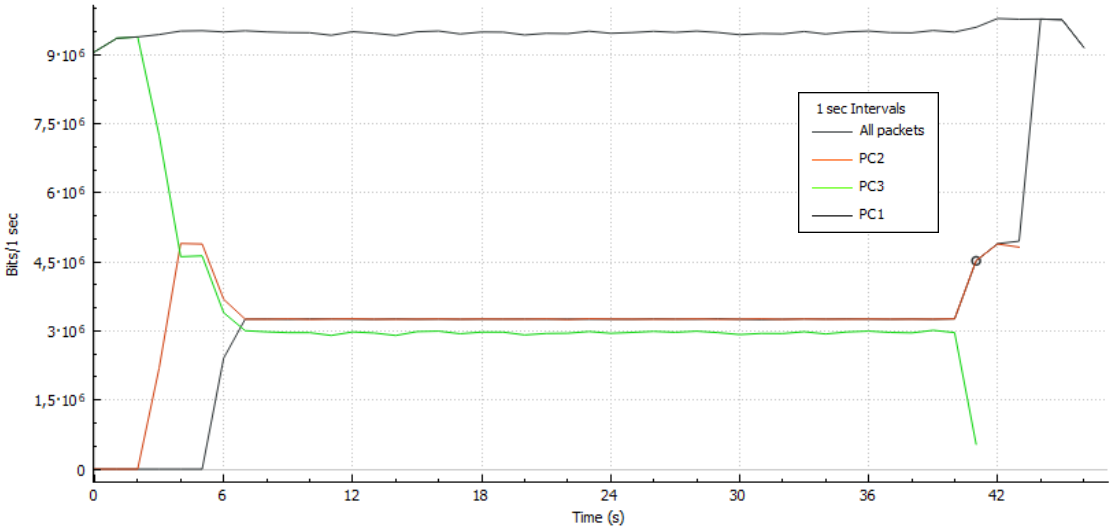


Figura 6.2. Grafica del ancho de banda recibido con WFQ

En la figura 6.2 se puede ver que el ancho de banda de los flujos del PC1 y PC2 son un poco superiores a los 3Mbps, mientras que el ancho de banda del flujo del PC3 es de unos 3Mbps. Se puede ver que el reparto del ancho de banda es más uniforme que sin QoS como cabía esperar ya que debido a la configuración WFQ aplicada el Router está sirviendo los paquetes de forma justa en función del peso que el protocolo adjudique a cada flujo cuando hay más de un flujo transmitiendo.

La siguiente tabla muestra los paquetes enviados y recibidos aplicando WFQ:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	53638	53373	54130
Recibidos	20433	15539	16315
Porcentaje de perdidas	61,9%	70,8%	69,8%

Tabla 5. Porcentaje de perdidas con WFQ.

Se puede apreciar que las pérdidas son menores que sin QoS a pesar de la congestión del enlace.

Esta configuración podría ser útil para una red en la que se desconocen los tipos de flujos que circulan por ella y se requiere que todos los flujos reciban un trato más justo.

4.1.2 Class-based Weighted Fair Queueing (CBWFQ)

Una vez aplicada la configuración de CBWFQ al Router empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

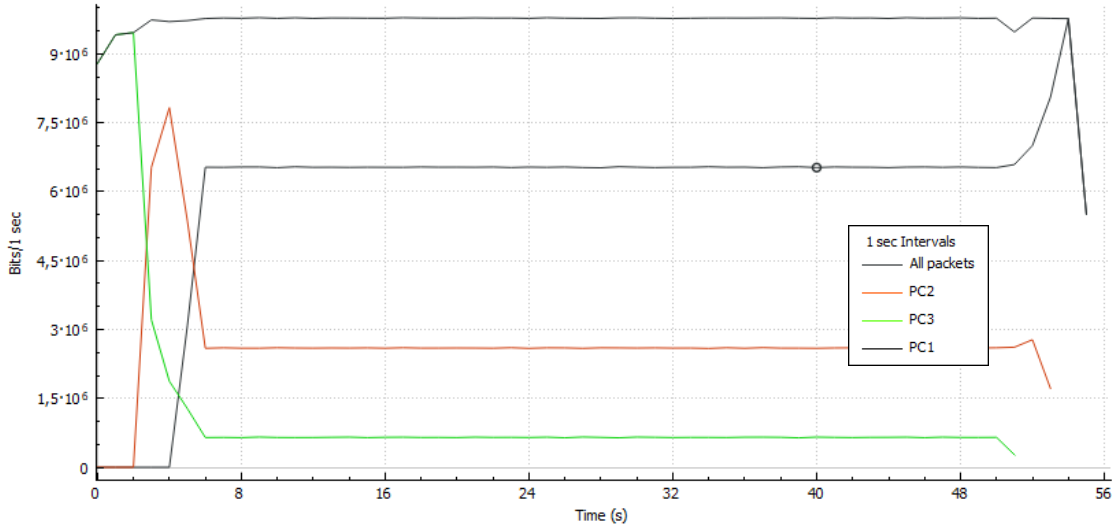


Figura 6.3. Grafica del ancho de banda recibido con CBWFQ

Como cabía esperar CBWFQ ha reservado un ancho de banda distinto para el tráfico proveniente de cada PC, lo que se puede comprobar en la figura 6.3. Debido a que solo se puede reservar un 75 por ciento del ancho de banda, los valores que aparecen en la gráfica no corresponden con los que debería haber reservado CBWFQ, ya que el comando bandwidth permite que se asigne más ancho de banda a los flujos si hay ancho de banda sobrante, en proporción al ancho de banda especificado. De esa forma el ancho de banda para el tráfico del PC1, que debería ser de 5 Mbps, es de 6.5 Mbps. El ancho de banda del tráfico del PC2, que debería ser de 2 Mbps, es de 2.5 Mbps. Y el ancho de banda del tráfico del PC3, que debería ser de 0.5 Mbps, es de 0.6 Mbps.

La siguiente tabla muestra los paquetes enviados y recibidos aplicando CBWFQ:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	66704	66071	66437
Recibidos	40522	15352	6820
Porcentaje de perdidas	39,2%	76,7%	89,7%

Tabla 6. Porcentaje de perdidas con CBWFQ.

Se puede apreciar que las pérdidas para el tráfico del PC1 una disminuido drásticamente en comparación con las que había cuando no había QoS. Las pérdidas del tráfico del PC2 han disminuido en comparación con las que había cuando no había QoS, pero no de forma significativa. Por ultimo las pérdidas del tráfico de PC3 han aumentado drásticamente debido a la falta de ancho de banda para enviar paquetes de ese flujo. Esto concuerda con los resultados esperados ya que se perderán más paquetes en las conexiones con menor ancho de banda mientras que en las conexiones con mayor ancho de banda llegarán más paquetes con éxito.

Este tipo de configuración podría ser útil para redes en la que un tráfico necesite tener menos perdidas que los otros y que los demás flujos puedan permitirse un mayor número de pérdidas. De esta forma si no se está transmitiendo el tráfico más importante, el ancho de banda que tiene reservado se reparte entre los demás flujos de tráfico.

4.1.3 Low Latency Queueing (LLQ)

Una vez aplicada la configuración de LLQ al Router empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

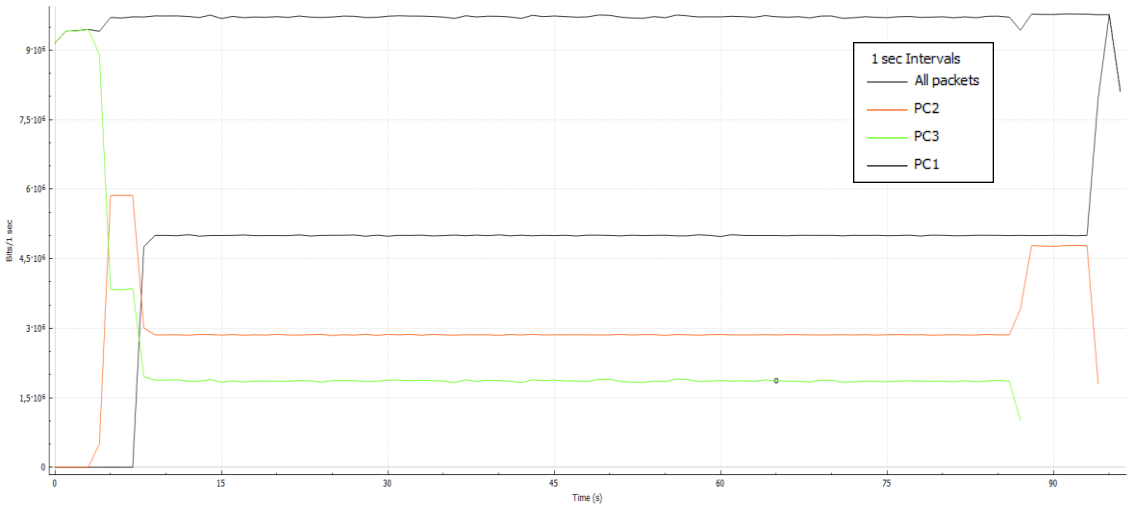


Figura 6.4. Grafica del ancho de banda recibido con LLQ

Como se esperaba LLQ ha reservado un ancho de banda distinto para los paquetes que provienen de cada uno de los PCs, como se ve en la figura 6.4. El valor del ancho de banda para el tráfico de PC1 se mantiene siempre estable en 5 Mbps cuando haya más de un flujo transmitiéndose, ya que el comando priority fija un ancho de banda al que no se le puede añadir más, aunque haya disponible si hay más ocupación. Los valores de los anchos de banda para el tráfico del PC2 y del PC3 no corresponden con los indicados ya que el comando bandwidth permite el reparto del ancho de banda sobrante en función del valor indicado, por lo que el tráfico de PC2 recibirá más ancho de banda que el tráfico de PC3.

La siguiente tabla muestra los paquetes enviados y recibidos aplicando LLQ:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	116826	117727	116353
Recibidos	59024	29669	21998
Porcentaje de perdidas	49,4%	74,7%	81,1%

Tabla 7. Porcentaje de perdidas con LLQ.

Se puede apreciar que las pérdidas para el tráfico del PC1 una disminuido drásticamente en comparación con las que había cuando no había QoS, Pero no son tan bajas como con CBWFQ ya que no se le ha asignado el exceso de ancho de banda. Las pérdidas del tráfico del PC2 han disminuido en comparación con las que había cuando no había QoS, pero no de forma significativa. Por ultimo las pérdidas del tráfico de PC3 han aumentado debido a la falta de ancho de banda para enviar paquetes de ese flujo. Estos valores cuadran con lo esperado ya que LLQ no solo reserva los valores del ancho de banda, sino que le ha dado prioridad a los paquetes que procedan de PC1 por lo que deben tener menos perdidas que los que provengan de PC2.

Este tipo de configuración podría ser útil para redes en la que un tráfico necesite un ancho de banda estable y menos perdidas que los demás tráficos que se pueden repartir el sobrante del ancho de banda entre ellos. Por eso esta configuración es usada en redes por las que circula tráfico de voz.

4.1.4 Weighted Random Early Detection (WRED)

Primero aplicare al Router la configuración de WRED al enlace de salida. Una vez aplicada la configuración empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

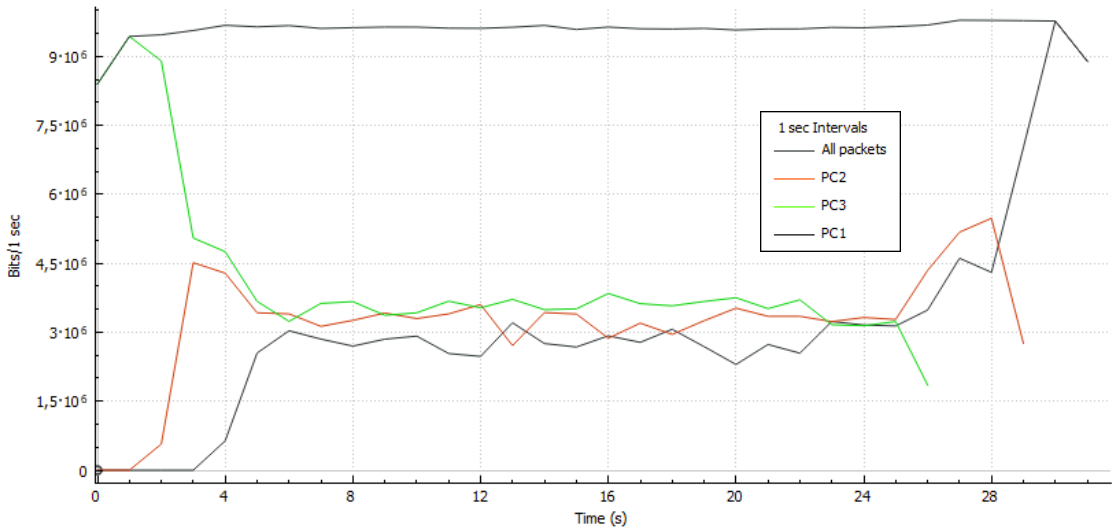


Figura 6.5. Grafica del ancho de banda recibido con WRED en el enlace

En la figura 6.5 se puede ver que el ancho de banda es muy parecido al obtenido sin aplicar QoS lo que era de esperar ya que WRED solo gestiona el mecanismo de descarte de los paquetes de la cola y no el ancho de banda que sale del Router.

La siguiente tabla muestra los paquetes enviados y recibidos mediante la aplicación de WRED al enlace:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	36257	35225	35172
Recibidos	11846	10221	12022
Porcentaje de perdidas	67,3%	70,9%	65,8%

Tabla 8. Porcentaje de perdidas con WRED en el enlace.

Se puede apreciar que las perdidas han disminuido para los tres flujos comparándolo con las perdidas sin QoS. Esto concuerda con los resultados esperado ya que al no esperar a que la cola este llena para descartar los paquetes se permite que más paquetes puedan ser almacenados en la cola y transmitidos.

Este tipo de configuración podría ser útil para redes por las que circule tráfico TCP ya que este responderá mejor a los descartes del protocolo y así evitara la congestión.

A continuación, aplicare WRED a cada clase de tráfico filtrado. Una vez aplicada la configuración empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

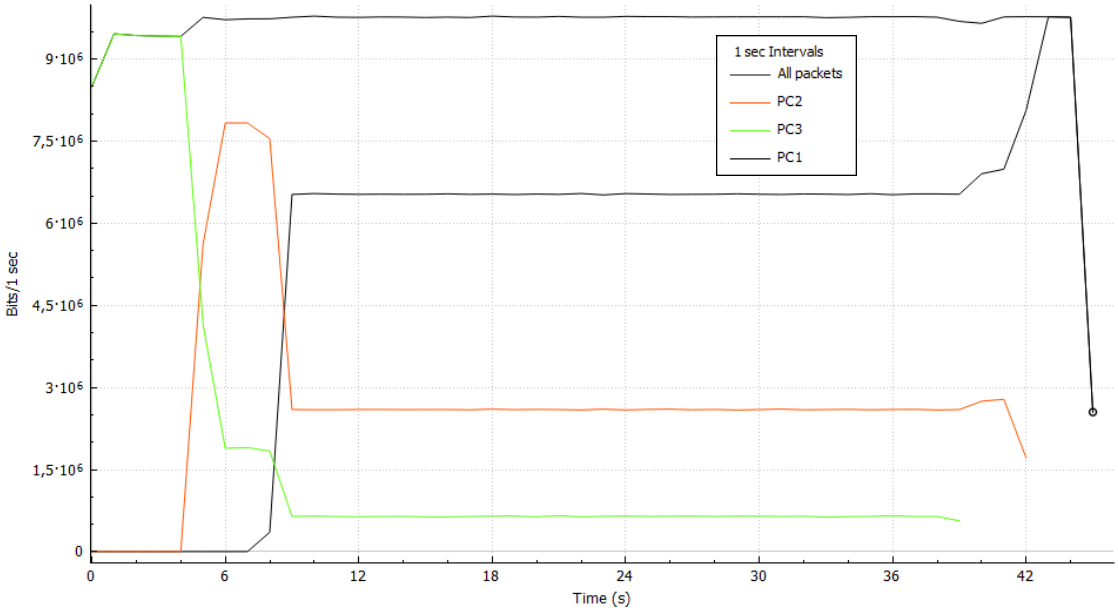


Figura 6.6. Grafica del ancho de banda recibido con WRED por clases

En la figura 6.6 se puede ver que el ancho de banda es el mismo que con CBWFQ, lo que era de esperar ya que se ha aplicado la misma gestión del ancho de banda. De esta forma se podrá comparar WRED con CBWFQ ya que se ha configurad WRED para que gestione el ancho de banda como CBWFQ.

La siguiente tabla muestra los paquetes enviados y recibidos mediante la aplicación de WRED a las clases:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	48190	49682	52185
Recibidos	30494	12504	8149
Porcentaje de perdidas	36,7%	74,8%	84,3%

Tabla 9. Porcentaje de perdidas con WRED en el enlace.

Se puede apreciar que las perdidas han disminuido para los tres flujos comparándolo con las pérdidas de CBWFQ, aunque no significativamente. Esto concuerda con los resultados esperado ya que al no esperar a que la cola este llena para descartar los paquetes se permite que más paquetes puedan ser almacenados en la cola y transmitidos.

Este tipo de configuración podría ser útil para redes por las que circulen varios flujos TCP que necesitan anchos de banda distintos ya que TCP es el protocolo que mejor responde a los descartes para evitar la congestión.

4.1.5 Commited Access Rate (CAR)

Primero aplicare al Router la configuración de CAR al enlace de salida. Una vez aplicada la configuración empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

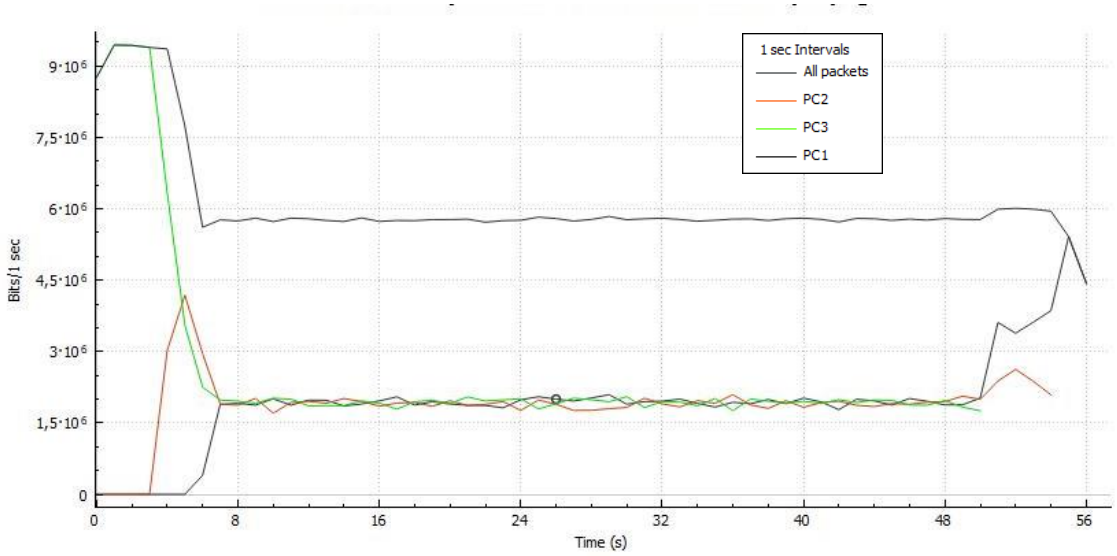


Figura 6.7. Grafica del ancho de banda recibido con CAR en el enlace

En la figura 6.7 se puede ver que el ancho de banda total que puede circular por el enlace ha sido limitado a el valor de 6 Mbps lo que concuerda con la configuración de CAR que se ha aplicado al Router. Dentro de ese límite se ha permitido la transmisión de rafas que no superen los 4.5 MB de tamaño, eliminándose las ráfagas que los superaran.

La siguiente tabla muestra los paquetes enviados y recibidos mediante la aplicación de CAR al enlace:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	66973	67278	67414
Recibidos	14708	11376	14553
Porcentaje de perdidas	78,0%	83,1%	78,4%

Tabla 10. Porcentaje de perdidas con CAR en el enlace.

Se puede apreciar que las pérdidas son muy similares a las obtenidas sin aplicar QoS debido a pesar de que cuando no aplicábamos QoS los flujos disponían de un mayor ancho de banda. Esto es debido a que se han podido transmitir ráfagas del mismo flujo lo que permite reducir el porcentaje de pérdidas.

Este tipo de configuración puede ser útil para redes en las que se quiera limitar el ancho de banda que circule por el enlace y a las que se le quiera dar un tratamiento especial a las ráfagas de tráfico que circulan por ella.

A continuación, aplicare CAR a cada clase de tráfico en función de las direcciones de las listas de acceso. Una vez aplicada la configuración empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

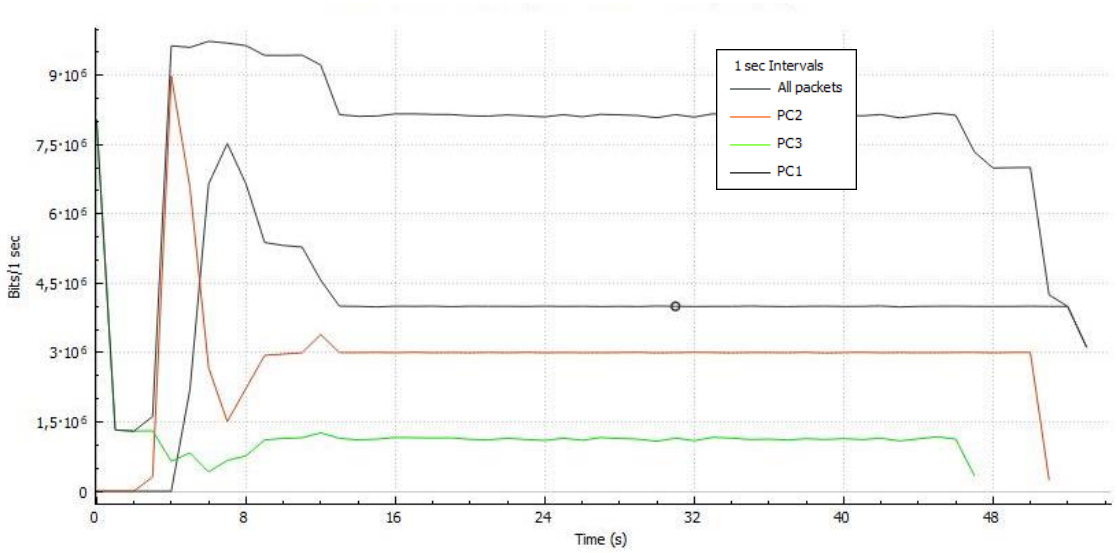


Figura 6.8. Grafica del ancho de banda recibido con CAR por flujo

En la figura 6.8 se puede ver que tras un periodo inicial donde se ha permitido la transmisión de múltiples ráfagas de tráfico, el protocolo ha limitado los anchos de banda hasta los valores indicados en la configuración del protocolo CAR para cada uno de los flujos que circulan por el Router.

La siguiente tabla muestra los paquetes enviados y recibidos mediante la aplicación de WRED a las clases:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	64015	62286	62456
Recibidos	26767	16457	7161
Porcentaje de perdidas	58,1%	73,5%	88,5%

Tabla 11. Porcentaje de perdidas con WRED en el enlace.

Se puede apreciar que las perdidas han disminuido para el flujo del PC1 en comparación con las perdidas sin QoS mientras que las pérdidas para el flujo de PC2 han disminuido poco y las del flujo de PC3 han aumentado, debido a las limitaciones del ancho de banda.

Este tipo de configuración podría ser útil para redes en las que se quiera limitar el ancho de banda de cada flujo de forma distinta, además de querer dar un tratamiento diferente a las ráfagas que circulen por ella.

4.1.6 Differentiated Services (DiffServ)

Una vez aplicada la configuración de *DiffServ* tanto al interfaz de entrada, para que marque los paquetes, como al interfaz de salida del Router, para que gestione el ancho de banda, empiezo la

generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

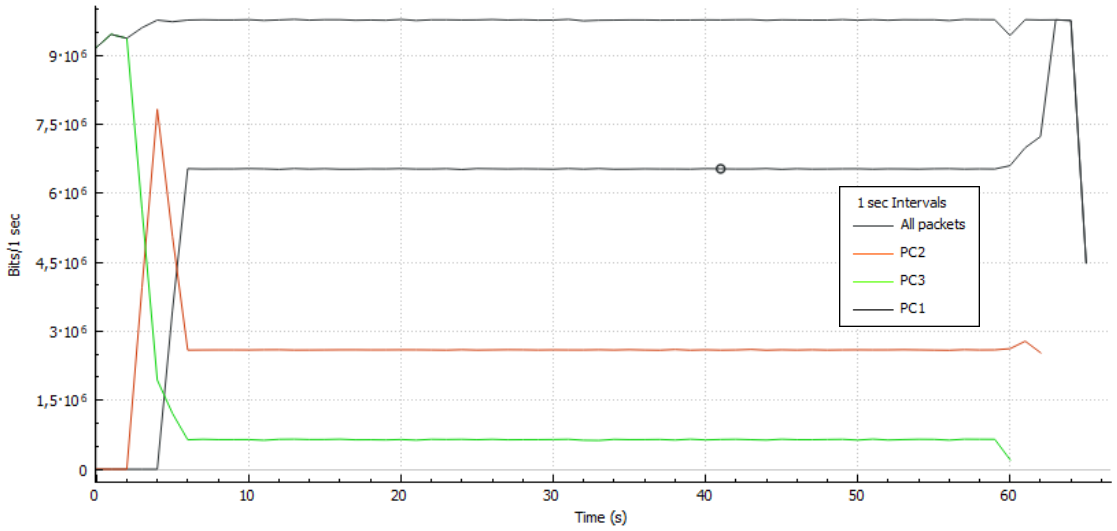


Figura 6.9. Grafica del ancho de banda recibido con *DiffServ*

En la figura 6.9 se puede ver que el ancho de banda es el mismo que con la configuración de CBWFQ. Esto concuerda con los resultados obtenidos ya que se había configurado el Router para que cambiara el valor del campo DSCP de los paquetes de entrada en función de su dirección de origen y luego aplicara CBWFQ a la salida del Router en función del valor DSCP que tenían los paquetes.

Además, en las siguientes imágenes se puede ver como se ha modificado el campo DSCP de los paquetes recibidos:

```
Internet Protocol Version 4, Src: 20.0.0.1, Dst: 10.0.0.1
   0100 .... = Version: 4
   .... 0101 = Header Length: 20 bytes (5)
 > Differentiated Services Field: 0xb8 (DSCP: EF PHB, ECN: Not-ECT)
```

Figura 6.10. Valor de *DiffServ* para paquetes del PC1

```
Internet Protocol Version 4, Src: 20.0.0.2, Dst: 10.0.0.1
   0100 .... = Version: 4
   .... 0101 = Header Length: 20 bytes (5)
 > Differentiated Services Field: 0x50 (DSCP: AF22, ECN: Not-ECT)
```

Figura 6.11. Valor de *DiffServ* para paquetes del PC2

```
Internet Protocol Version 4, Src: 20.0.0.3, Dst: 10.0.0.1
   0100 .... = Version: 4
   .... 0101 = Header Length: 20 bytes (5)
 > Differentiated Services Field: 0x68 (DSCP: AF31, ECN: Not-ECT)
```

Figura 6.12. Valor de *DiffServ* para paquetes del PC3

Se puede comprobar que el valor del campo DSCP de los tres paquetes, que originalmente era cero, ha sido modificado según la configuración aplicada lo que permitiría una gestión en cualquier Router que tuviera configurado *DiffServ*, aunque estuviera fuera de la red actual.

La siguiente tabla muestra los paquetes enviados y recibidos aplicando *DiffServ*:

Paquetes	PC1	PC2	PC3
Enviados	79712	78401	79482
Recibidos	49256	17594	7719
Porcentaje de perdidas	38,2%	77,5%	90,2%

Tabla 12. Porcentaje de perdidas con *DiffServ*.

Se puede apreciar que las pérdidas son prácticamente iguales a las obtenidas con CBWFQ ya que la gestión del ancho de banda se ha realizado de la misma manera, siendo *DiffServ* encargado solo de marcar los paquetes al entrar a la red.

Este tipo de configuración podría ser útil para paquetes que tengan que circular por varias redes y que necesiten un tratamiento específico en cada una. De esta forma si el paquete fuera marcado se podrían asignar los recursos que necesita, independientemente del origen, destino o protocolo transportado.

4.2 Router Serie 1900

A continuación, expondré los resultados obtenidos con el Router de la serie 1900. Para poder analizar los resultados de forma correcta primero mostrare los resultados obtenidos sin haber aplicado ninguna configuración de QoS al Router. Una vez finalizada la generación de tráfico el ancho de banda que ha llegado al PC4 es el siguiente:

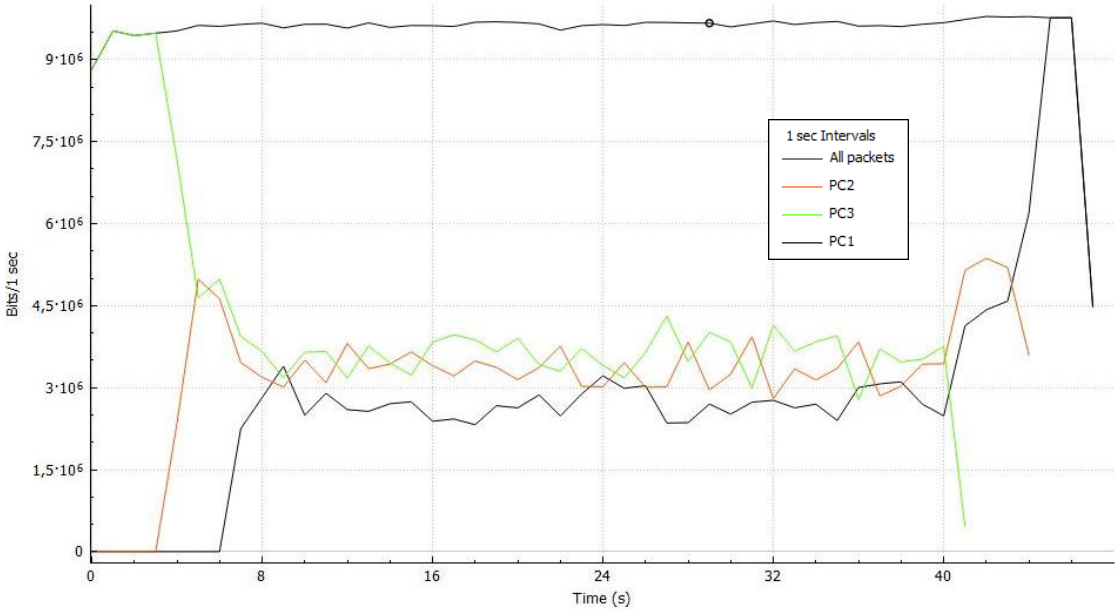


Figura 6.13. Grafica del ancho de banda recibido sin QoS

En la figura 6.13 se puede ver que el valor del ancho de banda recibido no supera los 10 Mbps máximos que tiene el enlace. Además, se puede ver que hasta que mientras el PC3 era el único que generaba tráfico utilizaba todo el ancho de banda disponible. Una vez empezaban a transmitir los otros PCs, el ancho de banda se repartía entre los tres flujos de tráfico, favoreciendo a los tráficos de ICMP y UDP sobre el tráfico TCP.

La siguiente tabla muestra los paquetes enviados y recibidos sin el uso de QoS:

Paquetes	PC1	PC2	PC3
Enviados	53628	53115	54509
Recibidos	16396	15313	18886
Porcentaje de perdidas	69,4%	71,1%	65,3%

Tabla 6.14. Porcentaje de perdidas sin QoS.

Se puede apreciar que, debido a la alta congestión, los tres flujos han sufrido pérdidas, habiendo sido el tráfico proveniente del PC2 el que más perdidas ha tenido.

4.2.1 Class-based Weighted Fair Queueing (CBWFQ)

Una vez aplicada la configuración de CBWFQ al Router empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

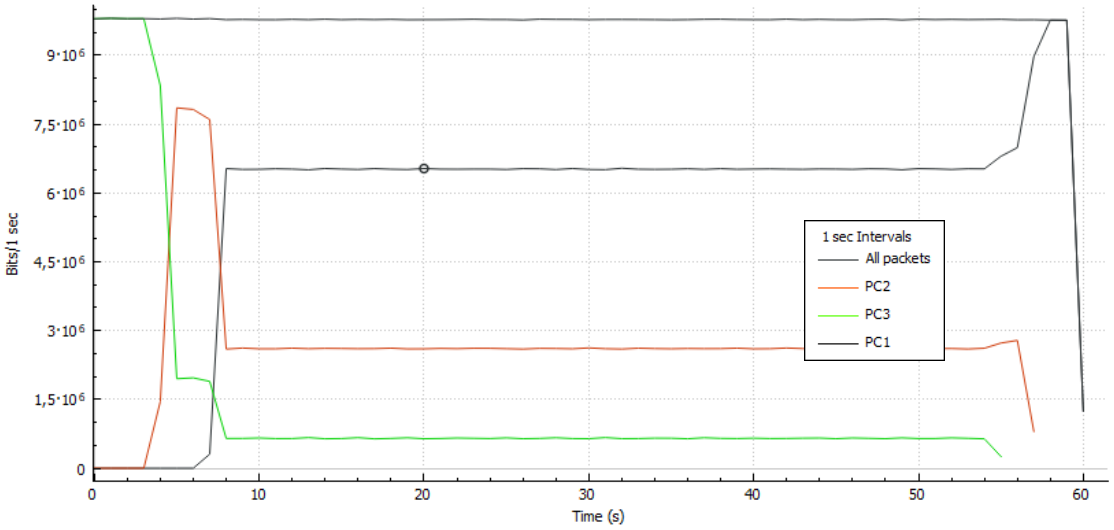


Figura 6.14. Grafica del ancho de banda recibido con CBWFQ

En la figura 6.14 se confirma que el ancho de banda se ha repartido en función del reservado por la configuración CBWFQ aplicada al Router. Debido a que solo se puede reservar un 75 por ciento del ancho de banda, los valores que aparecen en la gráfica no corresponden con los que se habían reservado, ya que el comando bandwidth permite que se asigne más ancho de banda a los flujos si hay ancho de banda sobrante, en proporción al ancho de banda especificado. De esa forma el ancho de banda para el tráfico del PC1, que debería ser de 5 Mbps, es de 6.5 Mbps. El ancho de banda del tráfico del PC2, que debería ser de 2 Mbps, es de 2.5 Mbps. Y el ancho de banda del tráfico del PC3, que debería ser de 0.5 Mbps, es de 0.6 Mbps.

La siguiente tabla muestra los paquetes enviados y recibidos aplicando CBWFQ:

Paquetes	PC1	PC2	PC3
Enviados	68729	69274	71942
Recibidos	42709	16462	9015
Porcentaje de perdidas	37,8%	76,2%	87,4%

Tabla 14. Porcentaje de perdidas con CBWFQ.

Se puede apreciar que las pérdidas para el tráfico del PC1 una disminuido drásticamente en comparación con las que había cuando no había QoS. Las pérdidas del tráfico del PC2 han

disminuido en comparación con las que había cuando no había QoS, pero no de forma significativa. Por ultimo las pérdidas del tráfico de PC3 han aumentado drásticamente debido a la falta de ancho de banda para enviar paquetes de ese flujo. Esto concuerda con lo esperado ya que los flujos con mayor ancho de banda deben tener menos perdidas ya que pueden enviar más paquetes que los flujos con menor ancho de banda.

Este tipo de configuración podría ser útil para redes en la que un tráfico necesite tener menos perdidas que los otros y que los demás flujos puedan permitirse un mayor número de pérdidas. De esta forma si no se está transmitiendo el tráfico más importante, el ancho de banda que tiene reservado se reparte entre los demás flujos de tráfico.

4.2.2 Weighted Random Early Detection (WRED)

Una vez aplicada la configuración WRED al Router empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

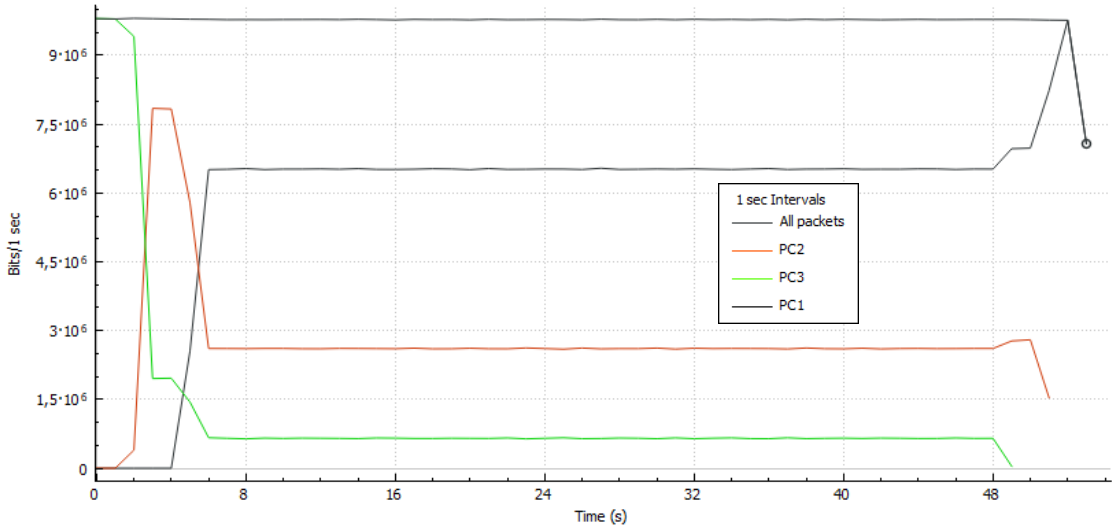


Figura 6.15. Grafica del ancho de banda recibido con WRED

Como cabía esperar en la figura 6.15 se puede comprobar que el ancho de banda es el mismo que con CBWFQ ya que se ha aplicado la misma gestión del ancho de banda para poder compararlo con CBWFQ ya que solo se ha cambiado el método con el que se descartan los paquetes de la cola no la forma en la que se gestiona el ancho de banda.

La siguiente tabla muestra los paquetes enviados y recibidos mediante la aplicación de WRED a las clases:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	63919	69274	64357
Recibidos	39457	15144	6681
Porcentaje de perdidas	38,2%	78,1%	89,6%

Tabla 15. Porcentaje de perdidas con WRED en el enlace.

Se puede apreciar que las perdidas han disminuido para los tres flujos comparándolo con las pérdidas de CBWFQ, aunque no significativamente. Esto es debido a la gestión de la cola.

Este tipo de configuración podría ser útil para redes por las que circulen varios flujos TCP que necesitan anchos de banda distintos ya que TCP es el protocolo que mejor responde a los descartes para evitar la congestión.

4.2.3 Hierarchical QoS (HQoS)

Una vez aplicada la configuración de HQoS al Router empiezo la generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

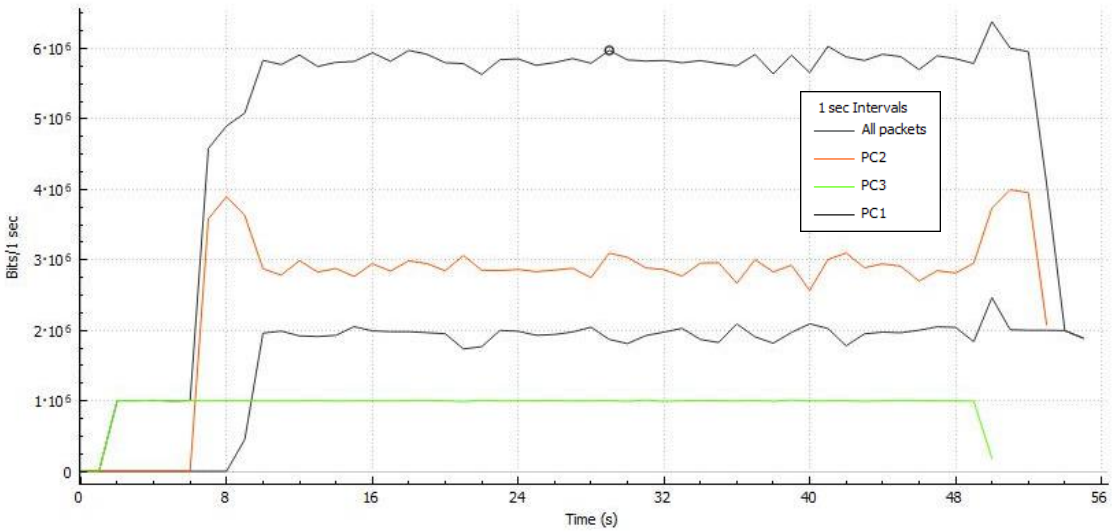


Figura 6.16. Grafica del ancho de banda recibido con HQoS

Como cabía esperar en la figura 6.16 se puede ver que el tráfico que proviene de PC1 tiene el ancho de banda indicado de 2 Mbps, el tráfico de PC2 tiene el ancho de banda de 3 Mbps indicado y el tráfico de PC3 tiene el ancho de banda indicado de 1 Mbps, como se ha configurado en la "clase hijo". Además, se puede ver que el ancho de banda de todo el tráfico tiene el ancho de banda indicado de 6 Mbps, como se ha configurado en la "clase padre". Además, el hecho de que la "clase padre" tuviera un valor de 6 Mbps ha permitido que el tráfico de PC2 pudiera aumentar su ancho de banda al haber ancho de banda disponible para poder cumplir con la condición de la "clase padre".

La siguiente tabla muestra los paquetes enviados y recibidos aplicando HQoS:

Paquetes	**PC1**	**PC2**	**PC3**
Enviados	71977	72291	71853
Recibidos	15589	11619	5742
Porcentaje de perdidas	78,3%	83,9%	92,0%

Tabla 16. Porcentaje de perdidas con HQoS.

Se puede apreciar que las pérdidas del tráfico de PC1 son similares a cuando no había QoS, las de PC2 han aumentado, aunque no significativamente, y las de PC3 han aumentado drásticamente. Esto es debido a la limitación del ancho de banda ya que sin QoS el tráfico de PC2 y Pc3 tenía más ancho de banda.

Este tipo de configuración podría ser útil para redes en las que se quiera dedicar distintos anchos de banda a distintas partes de la red y a su vez limitar el ancho de banda de los distintos tipos de tráfico que procedan de cada parte de la red.

4.2.4 Policy Based Routing (PBR)

Una vez aplicada la configuración de PBR tanto al interfaz de entrada, para que marque los paquetes, como al interfaz de salida del Router, para que gestione el ancho de banda, empiezo la

generación del tráfico. Una vez finalizada este es el ancho de banda que llega al PC4:

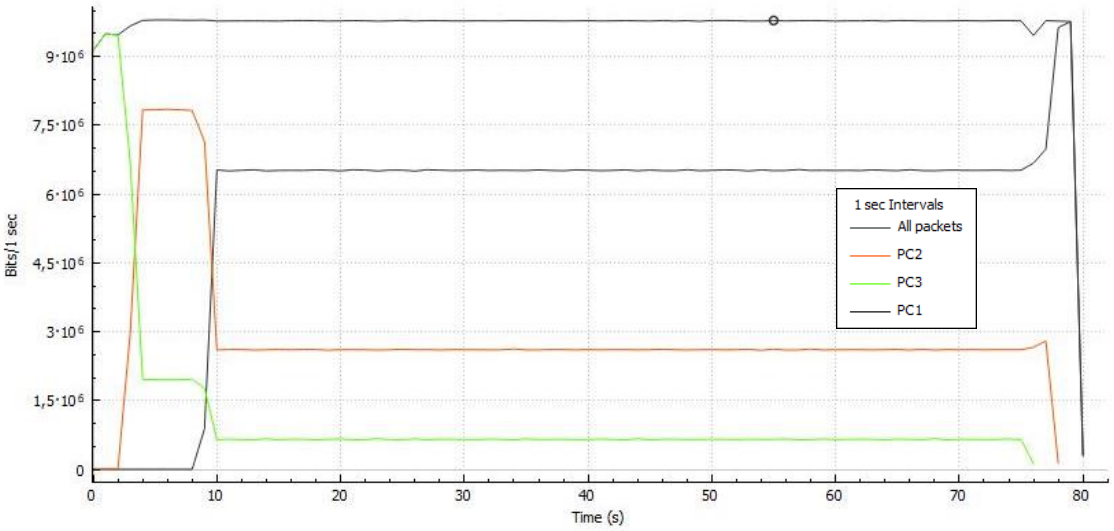


Figura 6.17. Grafica del ancho de banda recibido con PBR

Como cabía esperar en la figura 6.17 se puede ver que el ancho de banda es el mismo que con la configuración de CBWFQ ya que PBR ha modificado el valor del campo ToS de los paquetes a la entrada al Router y luego ha aplicado CBWFQ a los paquetes que salían del Router en función del valor del campo ToS en vez de por su dirección de origen.

Además, en las siguientes imágenes se puede ver como se ha modificado el campo ToS de los paquetes recibidos:

```
Internet Protocol Version 4, Src: 20.0.0.1, Dst: 10.0.0.1
   0100 .... = Version: 4
   .... 0101 = Header Length: 20 bytes (5)
 > Differentiated Services Field: 0xa0 (DSCP: CS5, ECN: Not-ECT)
```

Figura 6.18. Valor de PBR para paquetes del PC1

```
Internet Protocol Version 4, Src: 20.0.0.2, Dst: 10.0.0.1
   0100 .... = Version: 4
   .... 0101 = Header Length: 20 bytes (5)
 > Differentiated Services Field: 0x60 (DSCP: CS3, ECN: Not-ECT)
```

Figura 6.19. Valor de PBR para paquetes del PC2

```
Internet Protocol Version 4, Src: 20.0.0.3, Dst: 10.0.0.1
   0100 .... = Version: 4
   .... 0101 = Header Length: 20 bytes (5)
 > Differentiated Services Field: 0x20 (DSCP: CS1, ECN: Not-ECT)
```

Figura 6.20. Valor de PBR para paquetes del PC3

Hay que tener en cuenta que el valor de ToS coincide con el valor DSCP de tipo Class Selector. Teniendo esto en cuenta se puede ver que se han marcado los paquetes correctamente ya que los paquetes del PC1 tienen el valor CS5, los paquetes del PC2 tienen el valor CS3 y los paquetes de PC3 tienen el valor CS1, lo que coincide con la configuración aplicada al Router.

La siguiente tabla muestra los paquetes enviados y recibidos aplicando PBR:

Paquetes	PC1	PC2	PC3
Enviados	92475	97706	100791
Recibidos	57114	24098	9519
Porcentaje de perdidas	38,2%	75,3%	90,5%

Tabla 17. Porcentaje de perdidas con PBR.

Se puede apreciar que las pérdidas son prácticamente iguales a las obtenidas con CBWFQ ya que la gestión del ancho de banda se ha realizado de la misma manera, siendo PBR encargado solo de marcar los paquetes al entrar a la red.

Este tipo de configuración podría ser útil para paquetes que tengan que circular por varias redes y que necesiten un tratamiento específico en cada una. De esta forma si el paquete fuera marcado se podrían asignar los recursos que necesita, independientemente del origen, destino o protocolo transportado. También puede utilizarse en redes en las que haya varios caminos posibles para llegar al mismo destino. Usando PBR se podría indicar un camino alternativo a los paquetes cuando hubiera congestión.

Capítulo 5. Conclusiones

Tras analizar los resultados obtenidos se ha podido ver la importancia de aplicar una configuración QoS a una red. Si no se utiliza QoS todos los flujos de tráfico tienen que competir entre ellos para enviar los paquetes, provocando un aumento de pérdidas en todos los flujos. Pero con QoS es posible dar prioridad a los flujos que transporten tráfico que necesite unos requerimientos específicos para proporcionar un servicio determinado, a costa de proporcionar un servicio no prioritario al resto del tráfico lo que causara que se generen perdidas. Eso se ha ido comprobando a lo largo del documento ya que se ha visto tanto en la exposición de los diferentes tipos de QoS que se han estudiado como al realizar las pruebas en el entorno de trabajo.

Viendo los resultados obtenidos de generar tráfico en el entorno de trabajo, se puede concluir que el Router dispone de muchas medidas para la aplicación de QoS. Como se ha visto dependiendo de los requisitos de la red y del tráfico se puede aplicar al Router una política distinta para obtener una calidad óptima en la recepción del tráfico.

La prueba en la que mejor se ha observado los efectos de la QoS ha sido en la de CBWFQ ya que ahí se ha podido apreciar el efecto que tiene ceder demasiado ancho de banda a un solo tipo de tráfico, ya que así se produce un aumento en las perdidas de los demás flujos que circulen por el mismo enlace.

Sin embargo, cada configuración de las estudiadas tiene sus ventajas en función de cómo se configure, por lo que todas son útiles para aplicar políticas de QoS a las redes ya que dependiendo de los flujos de tráfico que vayan a circular por ellas necesitaran un tratamiento distinto.

Capítulo 6. Trabajo Futuro

En el presente trabajo se han aplicado políticas de QoS que limitaban el tráfico proveniente de tres PCs a través de un Router. Viendo esto se plantean tres diferentes posibilidades de análisis en el futuro.

En primer lugar, observar cómo afectaría a las políticas el cambiar el tamaño de las colas en lugar del ancho de banda. Además, sería interesante como afectaría a las perdidas el cambiar tanto el ancho de banda como el tamaño de las colas.

En segundo lugar, modificar el entorno de trabajo para que incluya un segundo Router por el que tenga que circular el tráfico. Esto permitiría ver cómo se pueden aplicar distintas configuraciones en cada Router y ver cuál sería el tráfico recibido. Además, esto permitiría estudiar mecanismos de QoS que necesitan más de un Router como RSVP.

Por último, se podría variar el formato de los paquetes para que pertenezcan a protocolos reales que circulan por las redes, como Real Time Protocol (RTP), lo que permitiría utilizar otros mecanismos para filtrar el tráfico por el tipo de protocolo y no solo por la dirección de origen del tráfico.

Capítulo 7. Bibliografía

[1] Cisco Systems, Inc, “Introducción a Weighted Fair Queuing en ATM” https://www.cisco.com/c/es_mx/support/docs/asynchronous-transfer-mode-atm/ip-to-atm-class-of-service/10049-wfq-illustrated.html [Online]

[2] Cisco Systems, Inc “Cisco IOS Quality of Service Solutions Configuration Guide,” Release 12.2SR, pp. 7, July 2018.

[3] Network Lessons “QoS LLQ (Low Latency Queueing) on Cisco IOS” https://networklessons.com/quality-of-service/qos-llq-low-latency-queueing-cisco-ios [Online]

[4] Cisco Systems, Inc, “QoS: Congestion Avoidance Configuration Guide” https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_conavd/configuration/xe-16/qos-conavd-xe-16-book/qos-conavd-wred-ecn.html [Online]

[5] Techopedia “Committed AccessRate (CAR)” https://www.techopedia.com/definition/31001/committed-access-rate-car [Online]

[6] Redes Convergentes “Descripción de DiffServ” https://sites.google.com/site/redescovergentesingluis/unidad-ii/3---descripcion-de-diffserv [Online]

[7] Telco “What is HQoS and how is it different from QoS?” https://www.telco.com/blog/hqos-solution-from-telco-systems/ [Online]

[8] Networkers Online “PBR as a QOS tool” http://www.networkers-online.com/blog/2008/06/pbr-as-a-qos-tool/ [Online]