GUIA DE LABORATORIO PARA LA IMPLEMENTACION DE QoS BASADO EN DIFFSERV EN GNS3.

Ing. ENAIRO JOSE VERGARA LAMBRAÑO

Ing. LUIS CARLOS CARBALLO BELTRAN

UNIVERSIDAD TECNOLOGICA DE BOLIVAR

PROGRAMA ESPECIALIZACION EN TELECOMUNICACIONES

CARTAGENA D.T.Y C,

2012

GUIA DE LABORATORIO PARA LA IMPLEMENTACION DE QoS BASADO EN DIFFSERV EN GNS3.

Guía de laboratorio para la implementación de QoS basado en Diffserv en GNS3.

Gonzalo López
Ingeniero Electrónico
M.s.c. En Ingeniería

UNIVERSIDAD TECNOLOGICA DE BOLIVAR

PROGRAMA ESPECIALIZACION EN REDES Y TELECOMUNICACIONES

CARTAGENA D.T. Y C.

2012

Nota de Aceptación

_____

_____

_____

_____

_____

_____

Firma de presidente del jurado

_____

Firma del jurado

_____

Firma del jurado

Cartagena, Abril 26 de 2012

Cartagena de Indias, Abril 26 de 2012

SEÑORES
COMITÉ DE REVISIÓN DE MONOGRAFÍA
UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR
La Ciudad

Apreciados señores:

Por medio de la presente nos permitimos informarles que la siguiente guía de laboratorio titulada "**GUIA DE LABORATORIO PARA LA IMPLEMENTACION DE QoS BASADO EN DIFFSERV EN GNS3**", ha sido desarrollada de acuerdo a los objetivos establecidos.

Como autores del proyecto consideramos que el trabajo es satisfactorio y amerita ser presentado para su evaluación.

Atentamente,

_____ _____
ENAIRO J. VERGARA LAMBRAÑO **LUIS C. CARBALLO BELTRAN**

Cartagena de Indias, Abril 26 de 2012

SEÑORES
COMITÉ DE REVISIÓN DE MONOGRAFÍA
UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR
La Ciudad

Apreciados señores:

Por medio de la presente me permito informarles que la siguiente guía de laboratorio titulada "**GUIA DE LABORATORIO PARA LA IMPLEMENTACION DE QoS BASADO EN DIFFSERV EN GNS3**", ha sido desarrollada de acuerdo a los objetivos establecidos.

Como director del proyecto considero que el trabajo es satisfactorio y amerita ser presentado para su evaluación.

Atentamente,

_____
GONZALO LOPEZ
Director Especialización Telecomunicaciones

AUTORIZACION

Cartagena de Indias D.T. y C., Abril 26 de 2012

Yo, **ENAIRO JOSE VERGARA LAMBRAÑO**, identificado con la cédula de ciudadanía Número 73166283 expedida en Cartagena (Bolívar), Autorizo a la Universidad Tecnológica de Bolívar a que haga uso de mi trabajo de grado y lo publique en el catalogo ONLINE de la biblioteca.

_____

ENAIRO JOSE VERGARA LAMBRAÑO

AUTORIZACION

Cartagena de Indias D.T. y C., Abril 26 de 2012

Yo LUIS CARLOS CARBALLO BELTRAN identificado con la cédula de ciudadanía Número 8850969 Expedida en Cartagena, Bolívar, Autorizo a la Universidad Tecnológica de Bolívar a que haga uso de mi trabajo de grado y lo publique en el catalogo ONLINE de la biblioteca.

_____

LUIS CARLOS CARBALLO BELTRAN

ARTICULO 105

La Universidad Tecnológica de Bolívar se reserva el derecho de propiedad intelectual de todos los trabajos de grado aprobados, los cuales no pueden ser explotados comercialmente sin la debida autorización de la misma.

DEDICATORIA

Esta monografía va dedicada a Dios, por brindarme la dicha de la salud y bienestar físico y espiritual.

A mi querida madre,NADINA LAMBRAÑO, por brindarme todo su amor, apoyo incondicional y consejos para ser de mí la persona que soy hoy en día.

A mi querida esposa,BERLYS BALLESTAS, y a mis hijos LINA MARCELA y ENAIRO JOSÉ por ser tan comprensivos y pacientes conmigo.

Gracias a todo ellos por contribuir en el logro de este objetivo.

_____

ENAIRO JOSE VERGARA LAMBRAÑO

DEDICATORIA

A Dios.

Por haberme dado la vida, la salud en la consecución de mis objetivos, además de su infinita bondad y amor.

A mis hijos Salomón Enrique Carballo Arrieta y Luis Carlos Carballo Arrieta

Por ser comprensivos cuando les quite un poco de espacio y el tiempo que muchas veces no pude compartir con ellos.

A mi esposa Neorlis Arrieta Periñán

Por todo el apoyo, la comprensión y los ejemplos de perseverancia y constancia que la caracterizan, por el valor mostrado para salir adelante y por su amor.

A mis Padres Orlando Carballo Arnedo y Anita Beltrán Martínez

Por haberme traído a este mundo lleno de alegría y esperanza, por el amor y los valores inculcados durante toda la vida.

A mis Hermanas

Por la comprensión y el apoyo incondicional que me han brindado cada vez que solicite su colaboración.

_____

LUIS CARLOS CARBALLO BELTRAN

AGRADECIMIENTOS

LOS AUTORES EXPRESAN SUS AGRADECIMIENTOS A:

Nuestro director de monografía, GONZALO LOPEZ por su constante colaboración y apoyo durante el desarrollo de la misma.

Los docentes, por su gran apoyo y motivación para la culminación de este posgrado.

Nuestros compañeros de la especialización, con quienes nos apoyamos mutuamente en nuestra formación.

A la Universidad Tecnológica de Bolívar y en especial a la Facultad de Ingeniería, por permitirnos ser parte de una generación de triunfadores y gente productiva para el país.

A nuestras diferentes empresas por habernos permitido un horario flexible y por brindarnos todo su apoyo y colaboración en la realización de esta monografía.

A todas las personas de una u otra forma apoyaron nuestro proceso de formación.

ENAIRO J. VERGARA LAMBRAÑO **LUIS C. CARBALLO BELTRAN**

INDICE DE CONTENIDOS

LISTA DE FIGURAS Pág.

INTRODUCCIÓN

Se van a llevar a cabo una serie de pruebas cuya finalidad será comprobar el funcionamiento de los routers Cisco de la serie 7200 en un entorno de Servicios Diferenciados emulado con el programa GNS3[1]. Para ello, se verá primero cómo configurar las herramientas que proporcionan estos dispositivos para implementar la arquitectura de Servicios Diferenciados.

En la arquitectura de Servicios Diferenciados hay dos tipos de nodos dentro de un dominio DiffServ: los nodos interiores y los nodos frontera. Los nodos frontera son los encargados de acondicionar el tráfico que entra al dominio y marcarlo con los valores de DSCP[2]apropiados. Mientras, los nodos interiores tratan al tráfico dependiendo de esos valores de DSCP.

Para las pruebas se dispone de dos routers Cisco 7200. Al carecer de un tercer router que actúe de nodo interior. Se decide en las pruebas que sólo uno de los routers realice la diferenciación de servicios, actuando como nodo frontera, es decir, que en él se lleven a cabo todas las funciones necesarias para implementar los Servicios Diferenciados, en concreto el servicio Assured Forwarding: acondicionando el tráfico mediante funciones de policy (con Token

[1] GNS3:Simulador Grafico de Red
[2] DSCP: Es un campo de un paquete IP que permite la asignación de niveles de servicio al tráfico de la red

Buckets), marcándolo, clasificándolo y dándole un tratamiento particular en función de los valores de DSCP de cada paquete. El otro router sólo servirá de interconexión entre el router frontera y los PC's. Esto permitirá crear un cuello de botella en el enlace entre ambos routers que provoque congestión y permita observar cómo actúan las diferentes herramientas de DiffServ de Cisco frente a ella.

Con el fin de poder comparar cómo el uso del Assured Forwarding de DiffServ permite garantizar los anchos de banda contratados y observar cómo se realiza el reparto del ancho de banda en exceso, realizando experimentos en diferentes escenarios con y sin emplear las herramientas que Cisco proporciona para la diferenciación de servicios.

1. OBJETIVOS

Identificar DiffServ como una buena estrategia de Calidad de Servicio, al igual que esta guía sea una base para que se pueda llevar a cabo la implementación de soluciones a los problemas de congestión utilizando el modelo DiffServ en GNS3.

A través de esta guía damos a conocer llas herramientas que existen para implementar los elementos de la arquitectura de Servicios Diferenciados y entre ellas el MQC, que es una utilidad del software Cisco IOS que permite configurar de modo sencillo, mediante el uso de una interfaz de comandos, todas las demás herramientas disponibles en el IOS que implementan los DiffServ (basadas en clases)

Comparar cómo el uso del Assured Forwarding de DiffServ permite garantizar los anchos de banda contratados y observar cómo se realiza el reparto del ancho de banda en exceso.

Utilizar listas de control de acceso de los routers, que filtraran el tráfico por dirección IP para poder obtener la diferenciación de tráficos por red LAN.

2. DESCRIPCIÓN DEL ENTORNO DE REALIZACIÓN DE LAS PRUEBAS

En primer lugar se llevan a cabo diferentes pruebas con dos fuentes generadoras de tráfico TCP (telnet). Cada una de estas fuentes simula una red LAN diferente y las dos están conectadas directamente al router (cada una por una interfaz) que implementa los Servicios Diferenciados.

Para la diferenciación de tráficos por red LAN se emplearán las listas de control de acceso de los routers, que filtraran el tráfico por dirección IP. Luego ese tráfico será clasificado en dos clases, una para cada red LAN. Si la diferenciación fuera por aplicación el modo de actuar sería el mismo pero las listas de acceso filtrarían el tráfico por protocolo y puerto y no por IP.

Como se ha comentado, se creará un cuello de botella en el enlace serial entre los dos routers. Esto producirá congestión y se podrá observar cómo actúan las diferentes herramientas de DiffServ para solventarla.

Para acondicionar los flujos de tráfico entrante que llegan a las interfaces del router se emplearán mecanismos de Token Bucket que comprobarán las características de los flujos y marcarán los paquetes con un DSCP u otro, dependiendo de si esos flujos permanecen dentro de los límites contratados o por

el contrario exceden esos límites. Es decir que el Token Buckets no se descartará paquetes.

Como se ha mostrado, las pruebas van encaminadas a comprobar si, usando los mecanismos que proporciona el software Cisco IOS, se garantizan los contratos a las diferentes redes LAN. Por esta razón, se supondrá que una de las redes LAN tiene contratados 256 Kbps y la otra red variará su contrato para cada prueba desde 256 Kbps hasta 1,768 Mbps, es decir, se irá variando el contrato hasta que la suma de los anchos de banda contratados por las dos redes LAN sea igual al ancho de banda del cuello de botella.

Naturalmente, ambas redes intentarán enviar más tráfico que el contratado y los mecanismos de DiffServ serán los encargados de asegurar que ambas redes reciben el ancho de banda contratado y de repartir el ancho de banda sobrante de manera equitativa.

Para la generación de tráfico no se cuenta con un entorno virtual en el cual se pueda trabajar (GNS3), por tal razón se usara el programa VPCS[3] que simulara el tráfico con mensajes de trama de internet. La duración de cada una de las pruebas será de 120 segundos. El tamaño de los paquetes será de 1024 bytes.

[3]Es una aplicación que permite simular PC y ejecutar comandos como ping o traceroute.

La figura 1 resume el entorno expuesto:

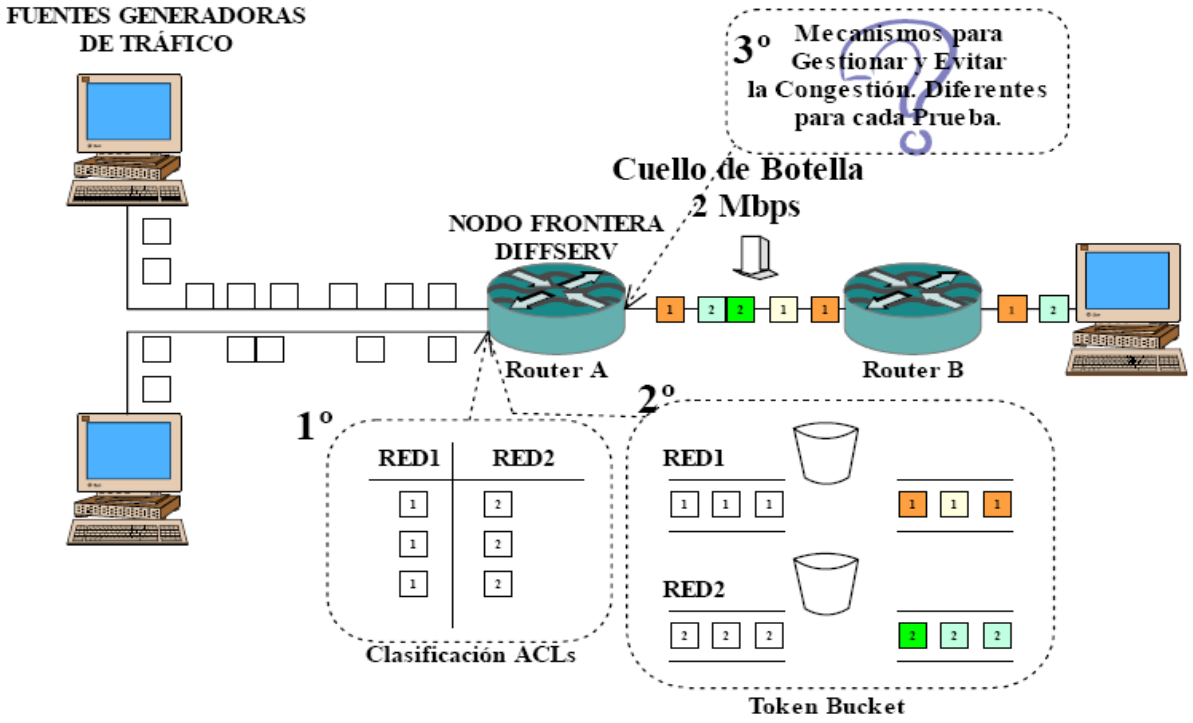

Figura1: Esquema del entorno de realización de las pruebas.

En la figura 1 se aprecian dos fuentes que generan tráfico TCP oUDP. Estas dos fuentes simulan cada una, una red LAN diferente. En el Router A, que actúa como nodo frontera del dominio de Servicios Diferenciados es donde se configuran todas las herramientas de DiffServ de Cisco que se van probar.

En el inciso (2.10) se detallan los procedimientos que se implementan para configurar las herramientas DiffServ en el Router A.

El análisis que se puede observar al aplicar todas esas configuraciones es que primero se puede ver también el camino que seguirán los paquetes desde que salen de las fuentes generadoras hasta que llegan a su destino. Al llegar al Router A, los flujos de tráfico se filtran y se clasifican en dos clases, una para cada red LAN. Luego, se le aplica al tráfico de cada clase un Token Bucket individual. Este Token Bucket comprobará si los flujos de tráfico cumplen sus contratos. Los paquetes que cumplan el contrato serán marcados con un DSCP y aquellos cuya tasa supere lo contratado se marcarán con un DSCP distinto. Por último, los paquetes llegarán a la interfaz serial del Router A, la cual experimenta congestión debido a que las fuentes de tráfico envían a una tasa mayor que la contratada. En este punto se probarán diferentes mecanismos de gestión y servicio de colas, algunos pertenecientes a diferenciación de servicios y otros no, para gestionar el acceso al enlace serie. El Router B no llevará cabo ninguna acción sobre los paquetes que no sea la de cambiarlos de interfaz.

Como se ha visto, para cada prueba se modificarán los contratos de las fuentes. Para esto simplemente se modificarán los parámetros de configuración de los Token Buckets. Las configuraciones que se llevarán a cabo en los routers serán:

- Configuraciones sin ningún tipo de Diferenciación de Servicios:
  - ✓ Disciplina de Servicio FIFO con Tail Drop
  - ✓ Disciplina de Servicio FIFO con WRED
  - ✓ Disciplina de Servicio WFQ

- Configuración con Diferenciación de Servicios:

  - ✓ Disciplina de Servicio FIFO con WRED
  - ✓ Disciplina de Servicio CBWFQ
  - ✓ Disciplina de Servicio CBWFQ con WRED

2.1 Descripción de Equipos a simular en GNS3

Para realizar las pruebas en el entorno virtual de GNS3 se dispone del IOS[4] del router Cisco de la serie 7200 sobre los que se aplicaran las configuraciones necesarias para implementar los Servicios Diferenciados.

También se dispone de tres ordenadores que serán las fuentes generadoras de tráfico y medidoras de la QoS percibida.

Las series de Cisco 7200 son una familia de routers modulares de acceso multiservicio que proporcionan configuraciones de LAN y WAN flexibles, múltiples opciones de seguridad, etc.

Este rango de características hace que la familia Cisco 7200 sea ideal para cubrir las necesidades actuales y futuras de empresas.

Los routers modularesmultiservicio Cisco 7200 ofrecen versatilidad, integración y potencia. Con alrededor de 70 módulos de red e interfaces, la arquitectura

[4] IOS: Sistema operativo que usan los routers CISCO.

modular de la serie Cisco 7200 permite actualizar fácilmente las interfaces para acomodarlas a la expansión de la red.

Las series de routers 7200 de Cisco ofrecen:

- Integración multiservicio de voz y datos
- Acceso a Internet/intranet con firewall
- Acceso a redes privadas virtuales (VPN) con opciones de firewall
- Servicios de acceso telefónico analógico y digital
- Enrutamiento con gestión de ancho de banda
- Enrutamiento entre VLAN

Con un potente procesador de alto rendimiento y procesadores auxiliares en varias interfaces, la serie Cisco 7200 admite calidad de servicio (QoS) avanzada, seguridad y las características de integración en la red.

Se utilizará la topología para realizar las pruebas según la Figura2:

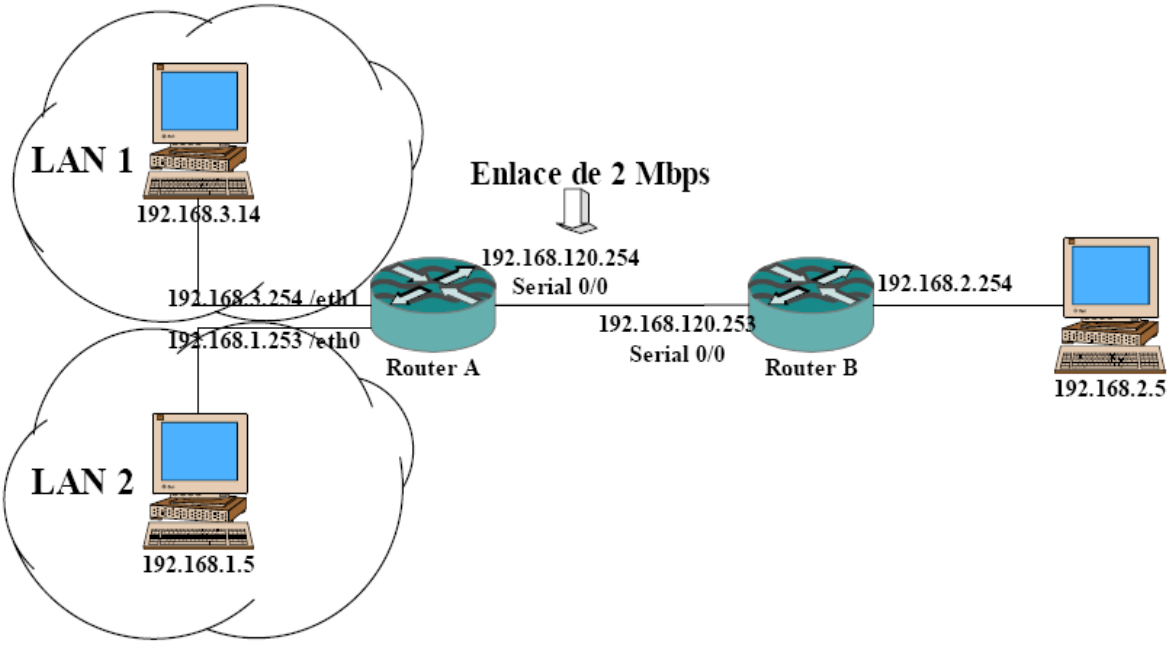

Figura2: Descripción de las redes configuradas.

2.2 Interconexión de los Equipos

La unión física entre ambos routers se realizaría mediante un cable serie V.35, para nuestro caso se hará virtual ya que el programa GNS3 lo hace automáticamente.

La señal de reloj estará en el extremo DCE que será el RouterB, el otro lado es un DTE es decir Router A. Por defecto, los routers Cisco son dispositivos DTE; sin embargo, en algunos casos se pueden utilizar como dispositivos DCE.

Para la realización de las pruebas se simulará un cuello de botella en el enlace serie configurando la velocidad del enlace a 2 Mbps. Las interfaces Ethernet tienen una velocidad máxima de transmisión de 10 Mbps y la interfaz Fast-Ethernet de 100 Mbps.

Los PCs emulados con el programa VPCS dispondrán de tarjetas de red 10/100. Así, los enlaces entre los PCs y el Router A serán de 10 Mbps como máximo debido a que las Ethernet del router sólo alcanzan esa velocidad y el enlace entre el Router B y el PC al que está conectado será de 100 Mbps por que en ambos extremos las tarjetas de red alcanzan esas velocidades.

Como se muestra en la figura 2, el PC con la dirección IP 192.168.1.5 se conecta directamente a través de un cable RJ-45 cruzado a la interfaz FastEthernet 2/0 del Router A cuya dirección IP será 192.168.1.253. El PC, con la dirección IP 192.168.3.14, se conecta directamente mediante un cable RJ-45 cruzado a la interfaz FastEthernet 2/1 del Router A cuya dirección IP es 192.168.3.254. El tercer PC de dirección IP 192.168.2.5 se conectará a la interfaz Fast-Ethernet 2/0 del Router B de dirección IP 192.168.2.254 del mismo modo, mediante el uso de un cable RJ-45 cruzado.

2.3 Configuración Inicial del Router A en GNS3.

Los routers virtuales cargados en GNS3 al principio carecen de configuración inicial, por lo tanto, no estarán configurados ninguno de los módulos de interfaz y será necesario acceder a los routers, a continuación se verá cómo dar esa configuración inicial a los routers A y B cuando se arrancan por vez primera, cómo configurar las interfaces de estos y los protocolos y tablas de enrutamiento necesarios para crear la topología anteriormente expuesta se muestra en la figura 2.

PASO1: Cargar el Router en GNS3.

Cuando se habilita la consola de manera virtual en el router Cisco, se realizan tres operaciones:

1. El router localiza el hardware y lleva a cabo una serie de rutinas de detección del mismo. Figura 3.

2. Una vez que el hardware se muestra en una disposición correcta de funcionamiento, el dispositivo lleva a cabo rutinas de inicio del sistema.

3. Tras cargar el sistema operativo, el dispositivo trata de localizar y aplicar la configuración disponible en la memoria NVRAM.

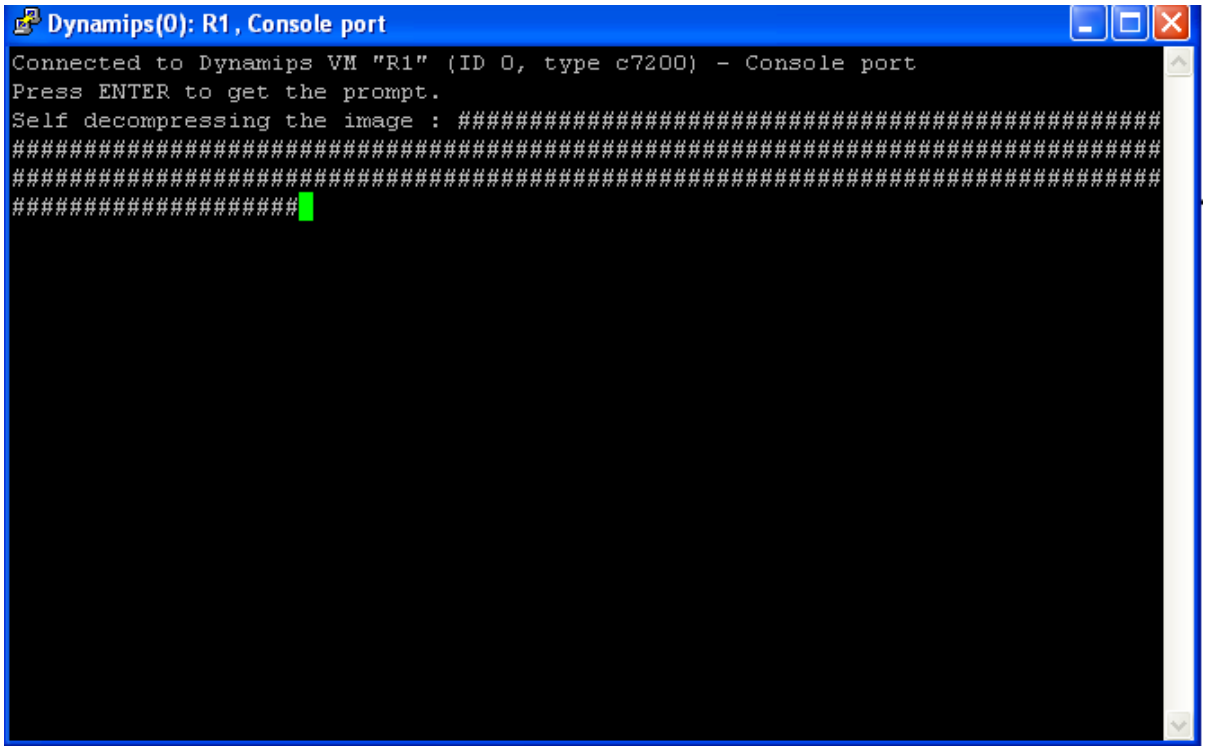

Figura 3: Test de inicio de Router

PASO2: Introducir un nombre de host para el router (se escribira Router A) como lo muestra la Figura 4:

```
Configuring global parameters:
Enter host name [Router]: RouterA
The enable secret is a password used to protect access to
privileged EXEC and configuration modes. This password, after
entered, becomes encrypted in the configuration.
```

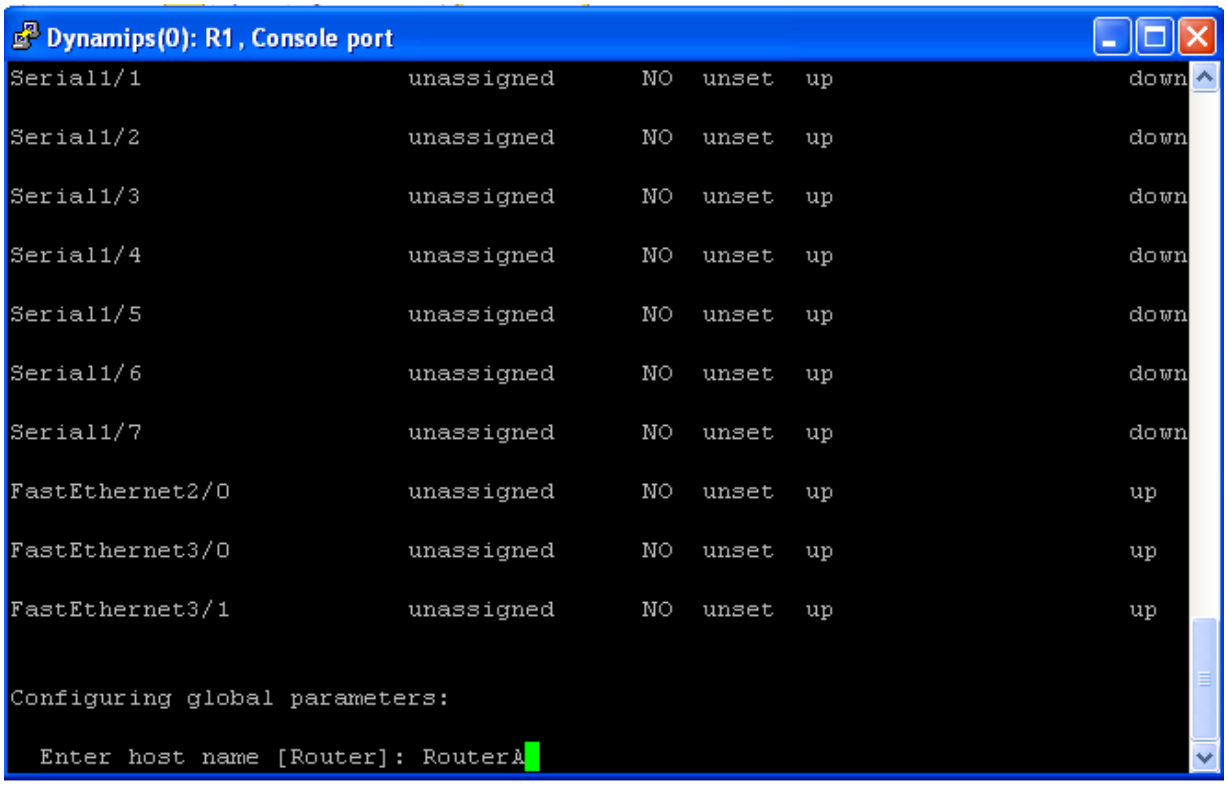

Figura 4: Definición de Hostname

PASO3: Introducir una contraseña enable secret. Esta contraseña está encriptada (más segura) y no se puede ver cuando se mira la configuración. Será la contraseña que el sistema pida para poder entrar a configurar el router:

```
Enter enable secret: ********
The enable password is used when you do not specify an enable
secret password, with some older software versions, and some boot
images.
```

PASO4: Introducir una contraseña enable que es diferente de la contraseña enable secret. Esta contraseña no está encriptada (menos segura) y se muestra cuando se mira la configuración. Se usa cuando no está la otra contraseña y cuando se usan versiones antiguas del software.

```
Enter enable password: guia12345
The virtual terminal password is used to protect access to the
router over a network interface.
```

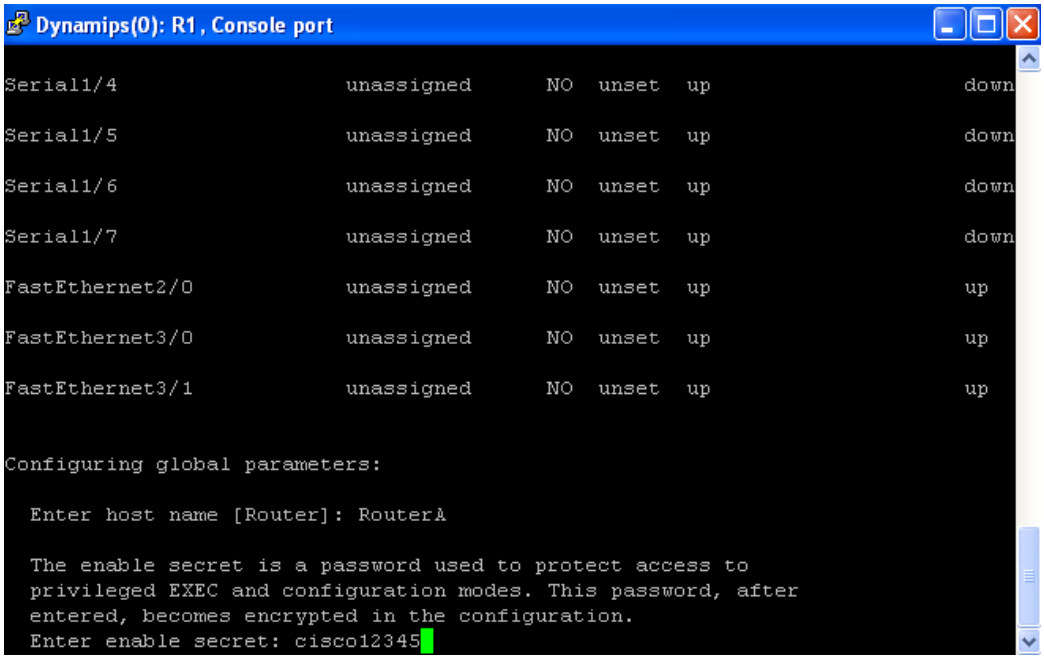

Figura 5: Definición de contraseña

PASO5: Finalmente se mostrará en pantalla tres opciones, donde se teclea 2 para guardar la configuración en la memoria de acceso aleatorio no volátil NVRAM y salir:

```
[0] Go to the IOS command prompt without saving this config
[1] Return back to the setup without saving this config
[2] Save this configuration to nvram and exit.
```

Posterior a esto ya se puede ingresar al router

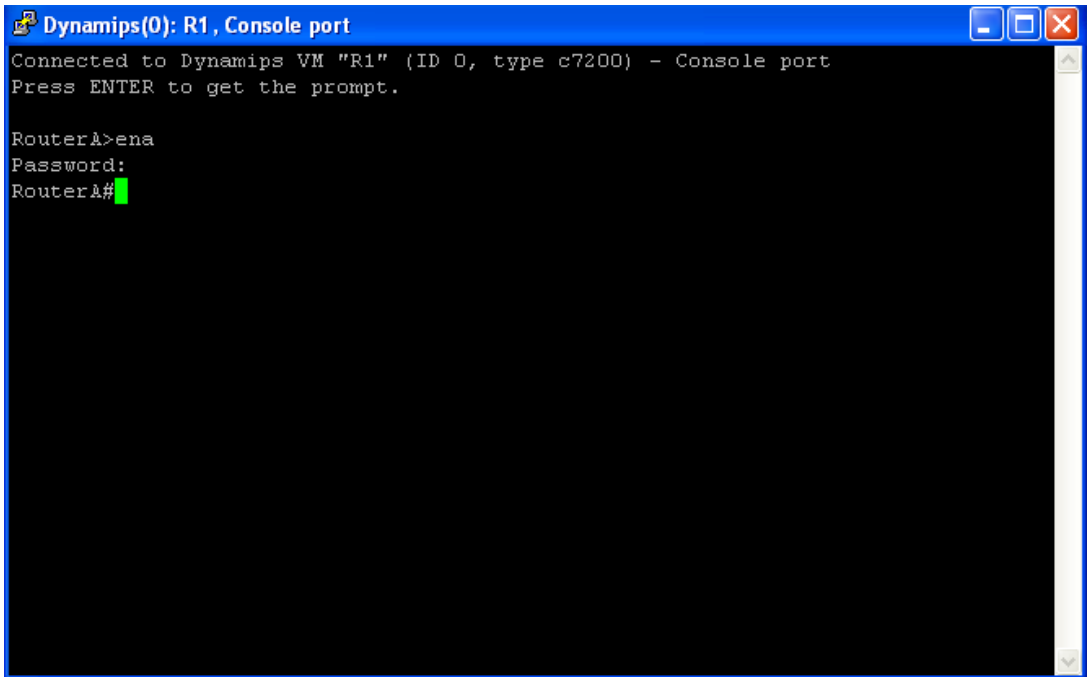

Figura 6: Ingreso al router

2.3.1 Configuración de Interfaces FastEthernet con GNS3:

Una vez salvada la configuración inicial se entra al router. La interfaz FastEthernet 2/0 se podía haber configurado mediante el diálogo de inicio del router. En caso contrario se configura del siguiente modo:

Configuración de la Interfaz FastEthernet 2/0 del Router A

Comando Descripción

```
Router A>enable
Router A>password: *****
RouterA#configure terminal
Router A(config)#interface FastEthernet 2/0
Router A(config-if)# ip address 192.168.1.253 255.255.255.0
Router A(config-if)# half-duplex
Router A(config-if)#no shutdown
Router A(config-if)#exit
```

Para configurar la interfaz fastEthernet 2/1 se seguirán los mismos pasos que para configurar la interfaz fastEthernet 2/0 sólo que en este caso la dirección IP será 192.168.3.253.

2.3.2 Configuración de la Interfaz Serial 1/0 del Router A

```
Router A>enable
Router A>password:*****
RouterA#configure terminal
Router A(config)#interface serial 1/0
Router A(config-if)# ip address 192.168.120.254 255.255.255.0
Router A(config-if)# encapsulation hdlc
Router A(config-if)#no shutdown
Router A(config-if)#exit
```

No se emplea el comando clockrate para indicar la velocidad del enlace porque la interfaz serial del Router A hará de DTE y por tanto, la interfaz serial del Router B será la que determine la velocidad del enlace.

2.3.3 Configuración del Encaminamiento del Router A

```
Router A>enable
Router A>password: *****
RouterA#configure terminal
Router A(config)# ip routing
Router  A(config)#  ip  classless
```
Comando utilizado para que el router no descarte paquetes destinados a redes con diferente clase de la perteneciente al router.

`Router A(config)# ip cef` Activa CEF; necesario para poder configurar algunas de las características de QoS

Comando router rip:

Para activar el protocolo RIP en el router se hará mediante el comando router rip, después se indicarán las redes a las que el router se conecta directamente mediante el comando network seguido de la dirección ip de la red.

```
Router A(config)# router rip
                    network 192.168.1.0
                    network 192.168.3.0
                    network 192.168.120.0
```

Tablas estáticas, gateway por defecto:

Para que el router sepa hacia donde dirigir los paquetes para los que no tengan ruta, es decir, aquellos paquetes que vayan a una red diferente de las especificadas en el paso anterior con el comando router rip. En este caso el router encaminará los paquetes por la Fastethernet 2/0 hacia la dirección ip 192.168.1.254 que es el router del laboratorio que tiene acceso a Internet.

```
ip route 0.0.0.0 0.0.0.0 FastEthernet2/0 192.168.1.254.
```

2.4 Configuración del Router B

La inicialización es muy similar a la del router A. se escribe el nombre de host del router que será Router B.

2.4.1 Configuración de la Interfaz Fast-Ethernet con GNS3:

Una vez salvada la configuración inicial se entra al router. La interfaz Fast-Ethernet 2/0 se podía haber configurado mediante el diálogo de inicio del router.

En caso contrario se configura del siguiente modo:

```
>enable
>password: *****
RouterB#configure terminal
RouterB(config)#interface fastEthernet 2/0
RouterB(config-if)# ip address 192.168.2.254 255.255.255.0
RouterB(config-if)# half-duplex
RouterB(config-if)#no shutdown
(config-if)#exit
```

2.4.2 Configuración de la Interfaz Serial 1/0 del Router B

```
RouterB>enable
RouterB>password: *****
RouterB#configure terminal
RouterB(config)#interface serial 1/0
RouterB(config-if)# ip address 192.168.120.253 255.255.255.0
RouterB(config-if)# encapsulation hdlc
RouterB(config-if)# clockrate 2000000
RouterB(config-if)#no shutdown
(config-if)#exit
```

Aquí si se utiliza el comando clockrate ya que la interfaz serial 1/0 del Router B actúa de DCE.

2.4.3 Configuración del Encaminamiento del Router B

```
RouterB>enable
RouterB>password: ******
RouterB#configure terminal
(config)# ip routing        Habilita el enrutamiento ip
(config)#  ip  classless Comando utilizado para que el router no descarte
```

paquetes destinados a redes con diferente clase de la perteneciente al router.

`(config)# ip cef` Activa CEF; necesario para poder configurar algunas de las características de QoS

Comando router rip: Para activar el protocolo RIP en el router se hará mediante el comando router rip, después se indicarán las redes a las que el router se conecta directamente mediante el comando network seguido de la dirección ip de la red.

```
RouterB(config)# router rip
                 network 192.168.2.0
                 network 192.168.120.0
```

Después de configurados los routers se procede a hacer una vista preliminar con el comando sh run config:

```
RouterA>ena
Password:
RouterA#sh running-config
Building configuration...

Current configuration : 1915 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RouterA
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$.6kb$dOhHZDf5WAA2AJIrCMer70
enable password guia12345
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
!
!
```

```
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex half
!
interface Serial1/0
 ip address 192.168.120.254 255.255.255.0
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/4
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/5
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/6
 no ip address
 shutdown
```

```
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/7
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface FastEthernet2/0
 ip address 192.168.1.253 255.255.255.0
 duplex half
!
interface FastEthernet2/1
 ip address 192.168.3.253 255.255.255.0
 duplex half
 speed auto
!
interface FastEthernet3/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router rip
 network 192.168.1.0
 network 192.168.3.0
 network 192.168.120.0
!
ip classless
ip route 0.0.0.0 0.0.0.0 FastEthernet2/0 192.168.1.254
no ip http server
no ip http secure-server
!
!
logging alarm informational
dialer-list 1 protocol ip permit
!
!
control-plane
!
!
gatekeeper
 shutdown
```

```
!
!
line con 0
 stopbits 1
line aux 0
line vty 0 4
 password guia12345
 login
!
!
end

RouterA#
```

La configuración del RouterB en GNS3 quedaría de la siguiente manera:

```
RouterB#sh running-config
Building configuration...

Current configuration : 1621 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RouterB
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
!
!
!
!
```

```
interface FastEthernet0/0
 no ip address
 shutdown
 duplex half
!
interface Serial1/0
 ip address 192.168.120.253 255.255.255.0
 serial restart-delay 0
 clock rate 2016000
 no dce-terminal-timing-enable
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/4
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/5
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/6
 no ip address
 shutdown
```

```
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/7
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface FastEthernet2/0
 ip address 192.168.2.254 255.255.255.0
 duplex half
!
interface FastEthernet2/1
 no ip address
 shutdown
 duplex half
!
router rip
 network 192.168.2.0
 network 192.168.120.0
!
ip classless
no ip http server
no ip http secure-server
!
!
!
logging alarm informational
!
!
control-plane
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 stopbits 1
line aux 0
line vty 0 4
!
```

```
!
end

RouterB#
```

2.5 Configuraciones Específicas

Se ha visto cómo crear configuraciones generales para preparar el entorno de realización de las pruebas. En ellas se han inicializado los routers, se han configurado las interfaces y se han activado los protocolos de encaminamiento.

En los pasos siguientes se explicará cómo crear las configuraciones específicas de los routers para llevar a cabocada una de las pruebas. Primero se verá la creación de las listas de control de acceso y las clases de tráfico que se emplearán en todas las pruebas así como un ejemplo de creación de los Traffic Policing que medirán y marcarán el tráfico de las redes.

Después, las configuraciones específicas de cada prueba se expondrán en sus resultados correspondientes.

2.5.1 Creación de las listas de control de acceso

Estas listas permitirán filtrar el tráfico que llega a las interfaces del router para poder clasificarlo posteriormente.

Desde el modo privilegiado, se accede al modo de configuración global para crear las listas de acceso.

```
RouterA# configure terminal
RouterA(config)# access-list 101 permit tcp host 192.168.3.14 host
192.168.2.5
RouterA(config)# access-list 101 permit udp host 192.168.3.14 host
192.168.2.5
RouterA(config)# access-list 102 permit tcp host 192.168.1.5 host
192.168.2.5
RouterA(config)# access-list 102 permit udp host 192.168.1.5 host
192.168.2.5
```

Con esto se han creado dos listas de acceso: una para filtrar el tráfico TCP o UDP de origen el host 192.168.3.14 y con destino 192.168.2.5 y otra para filtrar el tráfico TCP o UDP de origen el host 192.168.1.5 y con destino 192.168.2.5.

2.5.2 Creación de las clases de tráfico

Desde el modo privilegiado, se accede al modo de configuración global para crear las clases.

```
RouterA# configure terminal
RouterA(config)# class-map match-all trafico3.14-2.5
RouterA(config-cmap)# match access-group 101
```

```
RouterA(config-cmap)# exit
RouterA(config)# class-map match-all trafico1.5-2.5
RouterA(config-cmap)# match access-group 102
RouterA(config-cmap)# exit
```

Se han creado dos clases, una para el tráfico perteneciente al host 192.168.3.14 y de destino 192.168.2.5 y otra para el tráfico perteneciente al host 192.168.1.5 y de destino 192.168.2.5.

2.5.3 Configuración de los Traffic Policy.

Se tendrán que crear dos policy-maps diferentes ya que hay dos interfaces distintas por las que entra el tráfico (FastEthernet 2/0 y 2/1).

```
RouterA# configure terminal
RouterA(config)# policy-map TrafficPolicing3.14_2.5
RouterA(config-pmap)# class trafico3.14-2.5
```

La configuración llevada a cabo será para aquellos casos en los que los paquetes de ambas redes se marquen de igual forma, distinguiendo únicamente si son IN u OUT. Estas políticas marcarán todos los paquetes conformes con el contrato de ambas redes con el DSCP 18 y todos los paquetes de ambas redes que excedan su respectivo contrato con el DSCP 20.

2.6 Configuraciones sin ningún tipo de diferenciación de servicios

Para esta prueba se dispone de la configuración vista anteriormente. Ninguno de los dos routers utilizará un mecanismo de diferenciación, es decir, independientemente de cómo se realice el marcado de paquetes todos ellos irán a

parar a colas que tratarán a todos los paquetes por igual. El tráfico será generado por dos fuentes TCP que generarán tráfico de manera virtual usando la configuración de 2 pcs en el programa VPCS.

También se llevarán a cabo pruebas en las que una fuente ofrece tráfico TCP a la red y la otra ofrece tráfico UDP para ver cómo afecta el tráfico no 'colaborador' en estas configuraciones.

Las configuraciones que se probarán son:

- Disciplina de Servicio FIFO con Tail Drop
- Disciplina de Servicio FIFO con WRED (en este caso actúa como RED)
- Disciplina de Servicio WFQ

2.6.1 Disciplina de Servicio de Colas FIFO con Tail Drop

Si se recuerda brevemente sobre los mecanismos de Gestión y Control activo de la congestión, se determina que:

a) El encolamiento de primero en entrar, primero en salir (First-in, first-out FIFO), es la disciplina para servir paquetes más básica, todos los paquetes se tratan igual, colocándolos en una única cola y sirviéndolos en el mismo orden en el

que fueron colocados en la misma. FIFO se denomina también primero en llegar, primero en servirse (First-come,first-served FCFS).

b) Tail Drop significa la ausencia completa de un gestor de la memoria de la cola. Cuando un paquete llega al final de una cola completamente llena. El paquete se descarta al igual que todos los que lleguen tras él hasta que hay espacio disponible en la cola.

De lo que se deduce que los paquetes irán llenando la cola FIFO en periodos de congestión y cuando se llegue al límite de la memoria de la cola, los nuevos paquetes que lleguen serán descartados. Esto puede provocar:

a) La sincronización global de las fuentes de tráfico TCP (en ambientes reales no aplica para ambientes virtuales), es decir, que todas comiencen o dejen de transmitir al mismo tiempo. Dando lugar a periodos en los que el enlace se encuentra desaprovechado seguidos de otros en los que en enlace está congestionado. Produciéndose muchas pérdidas.

b) Flujos de tráfico UDP que no saben cuando hay congestión acaparan todo el ancho de banda del enlace.

2.6.2 Configuraciones

Las listas de acceso, las clases y las políticas de policing son las mismas que en caso general. Lo único que se debe configurar en este caso es la cola FIFO que habrá en la interfaz serial 1/0. La interfaz serial tiene como disciplina de servicio de colas por defecto a WFQ. Para cambiarla se entrará en el modo de configuración de la interfaz y empleará el comando no fair-queue, con este comando se deshabilitará la cola WFQ establecida por defecto y se activará FIFO. En el router se hará lo siguiente:

```
RouterA# configure terminal
RouterA(config)# interface serial 1/0
RouterA(config-if)# no fair-queue
RouterA(config-if)# exit
```

En la figura 7 se muestra un esquema que muestra el proceso que se llevará a cabo con los paquetes que ingresen en el Router A:

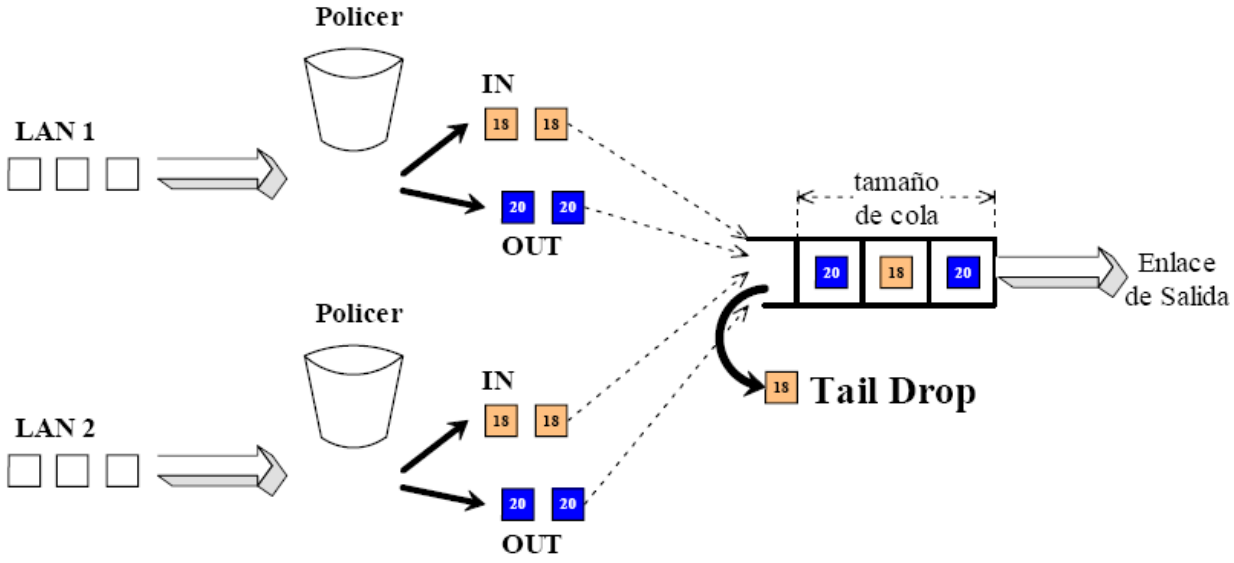

Figura 7:Funcionamiento de la Configuración de la cola FIFO con Tail Drop

Los paquetes llegan a las interfaces del Router A y son filtrados mediante las listas de acceso 101 y 102 configuradas. Los paquetes se clasifican en dos clases diferentes una para cada red LAN. Los paquetes de cada clase son medidos y marcados según cumplan o no el contrato mediante un Token Bucket. Todos los paquetes de ambas clases conformes con el contrato se marcan con el DSCP 18 y todos los que excedan el contrato se marcan con el DSCP 20. Una vez marcados, los paquetes se envían a una cola FIFO si se produce congestión en el enlace de salida y allí serán descartados por Tail Drop si se produce desbordamiento de la cola.

2.7 Análisis de Resultados.

A continuación se muestran los resultados de las pruebas llevadas a cabo. La figura 8 indica la fuente donde se generara el tráfico. Se puede observar el “tráfico IN” tráfico que el Token Bucket ha marcado como conforme por respetar el contrato. Luego se observa el “tráfico OUT” que muestra el tráfico que el router marca como fuera del contrato. Finalmente se muestra el tráfico total de cada una de las fuentes que llega al Router A durante los 120 segundos de duración de la prueba. Los resultados vienen dados Kbits/seg.

Como se ha visto en la configuración: los paquetes se filtrarán mediante las ACLs y se clasificarán en dos clases, una para cada red. Posteriormente, pasarán a través de un mecanismo de Token Bucket que se encargará de marcarlos.

Finalmente, llegarán a la interfaz serial 1/0 del Router A donde serán encolados en una cola FIFO en caso de congestión y posiblemente descartados por Tail Drop si se produce desbordamiento de la cola. Estos resultados los proporciona el Router usando los comandos show policy-map interface FastEthernet 2/0 y show policy-map interface FastEthernet 2/1.

2.7.1 Resultados del Token Bucket (Cola FIFO con Tail Drop)

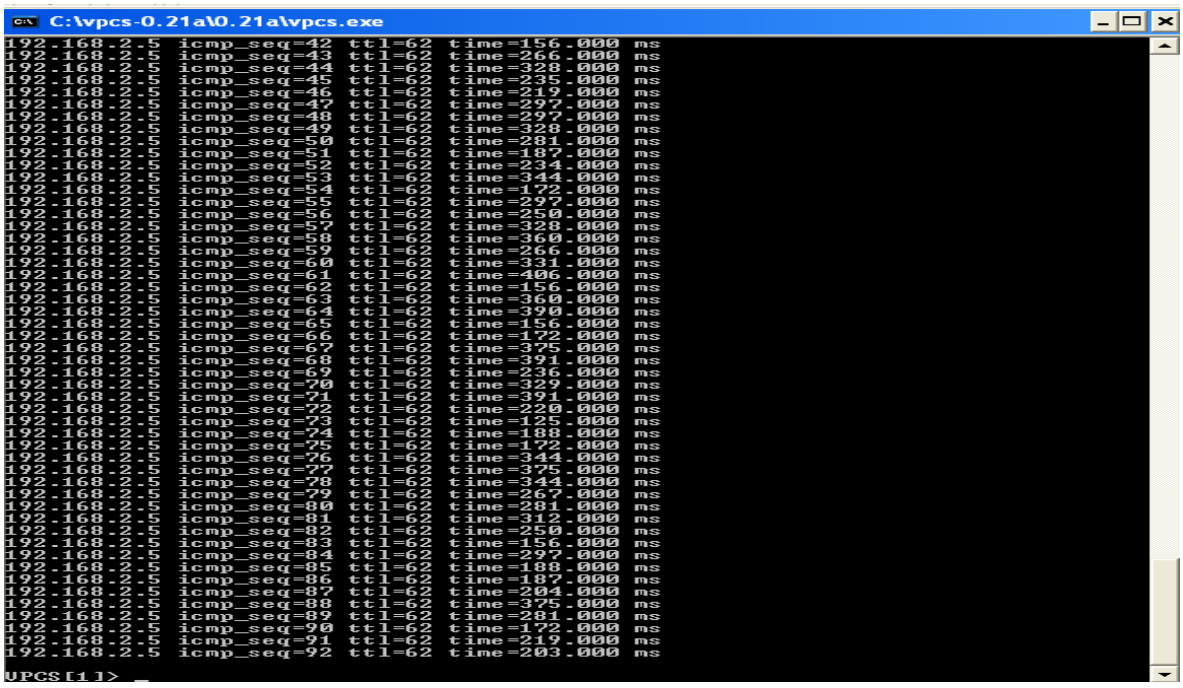

Figura 8: Generación de trafico

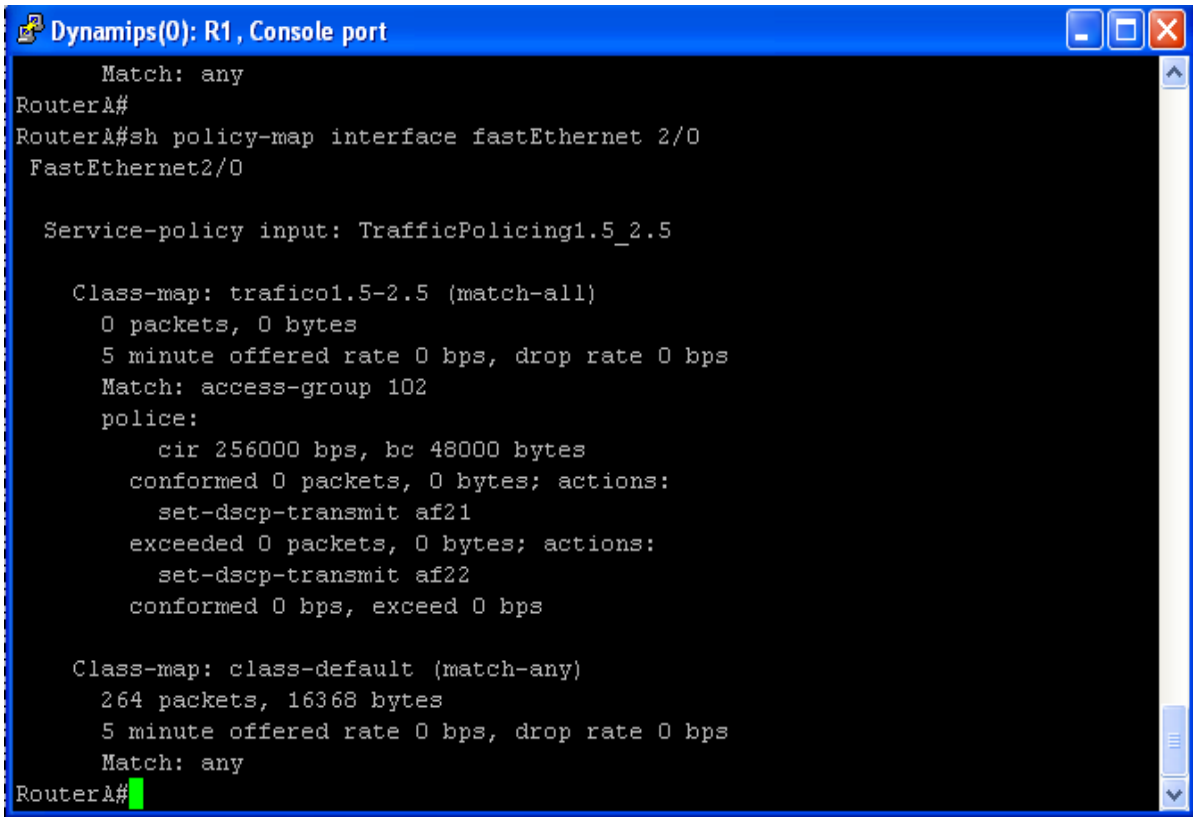

Figura 9:Resultado sh run de la interface del trafico

En la figura 10 los paquetes descartados a causa de la congestión en la cola FIFO de la interfaz serial 1/0 del RouterA. Estos resultados los proporciona el Router usando los comandos show interface serial 1/0.

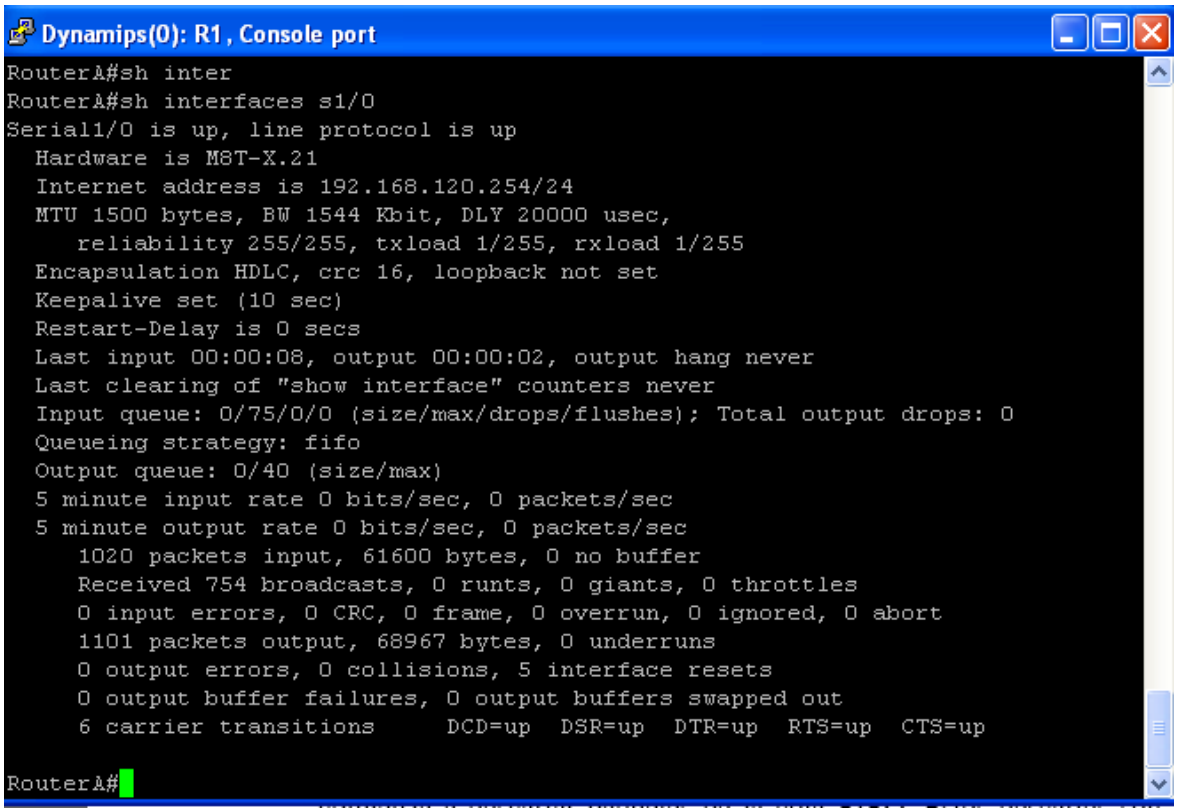

Figura 10: Resultados del trafico por la Serial

Se puede observar en la figura 10 que, una cola FIFO sirve los paquetes de tal manera que se reparte el ancho de banda de manera más o menos equitativa entre ambas redes. Por esta razón, el caudal total, es aproximadamente el mismo (1M para cada red, que es igual al 50% del enlace serie).

En un ambiente simulado con GNS3 es difícil ver las diferencias en comparación con un ambiente real donde se observarían más significativamente las diferencias en cuanto al reparto del ancho de banda en exceso.

Con los resultados obtenidos se podía pensar que usando disciplinas de colas FIFO se pueden garantizar los contratos, por lo menos, para los primeros casos. Pero hay que tener en cuenta que las pruebas se han realizado en un ambiente virtual usando tráfico ICMP, en un ambiente real donde el tráfico es TCP. Cuando se produce congestión, Tail Drop comienza a descartar paquetes de la cola FIFO. Estos descartes son una manera implícita de avisar a las fuentes de tráfico TCP que disminuyan su ventana de transmisión y por tanto la tasa de transmisión. Por todo esto, ambas conexiones consumen más o menos la misma cantidad de ancho de banda del enlace y no se producen excesivos descartes.

Pero donde se aprecia claramente que este tipo de disciplina no permite garantizar contratos, es cuando alguna de las dos fuentes emplea un tipo de tráfico no 'colaborador' como UDP.

Cuando se producen descartes de paquetes en la cola FIFO debido a la congestión, UDP no tiene ningún método para saber de esas pérdidas y por tanto continuará transmitiendo a la tasa máxima. En el caso de que a una misma cola lleguen paquetes UDP y TCP, si se produce congestión, TCP reducirá su tasa de transmisión para no seguir congestionando al router pero

UDP no, aprovechándose de que TCP transmite menos para introducir más paquetes en el enlace. Esto provocará que TCP apenas consiga ancho de banda del enlace sea cual sea su contrato.

2.8 Disciplina de Servicio FIFO con WRED

La siguiente configuración es una cola FIFO con WRED como método para el descarte de paquetes de la cola. Se recuerda que en la disciplina de gestión de la memoria de cola WRED, hay dos tipos de WRED: un WRED con un único perfil de descarte para todos los paquetes de una cola y un WRED con varios perfiles de descarte para los paquetes de una misma cola según fueran estos paquetes IN o OUT. En la siguiente configuración se empleará WRED con un solo perfil de descarte para todos los paquetes de la cola, pertenezcan éstos a una red LAN o a otra y respeten o no los contratos pre establecidos. De esta forma, aunque al entrar al Router A los paquetes se clasifiquen según pertenezcan a la red LAN 1 o a la 2 y se marquen como IN o como OUT dependiendo de si cumplen o no sus contratos, todos irán a parar a una misma cola FIFO con un único perfil de descarte en caso de congestión.

Independientemente de que los Token Bucket para los paquetes de ambas redes se realicen por separado, los paquetes se marcarán con los mismos DSCP, es decir, todos los paquetes IN de ambas redes se marcarán con el valor de DSCP 18 y todos los paquetes OUT de ambas redes se marcarán con el valor de DSCP 20.

Cuando se sobrepase el umbral mínimo, WRED comenzará a descartar algunos paquetes y cuando se sobrepase el umbral máximo WRED descartará todos los paquetes nuevos que lleguen a la cola.

En la figura 11 aparece un esquema del recorrido de los paquetes a través del Router A.

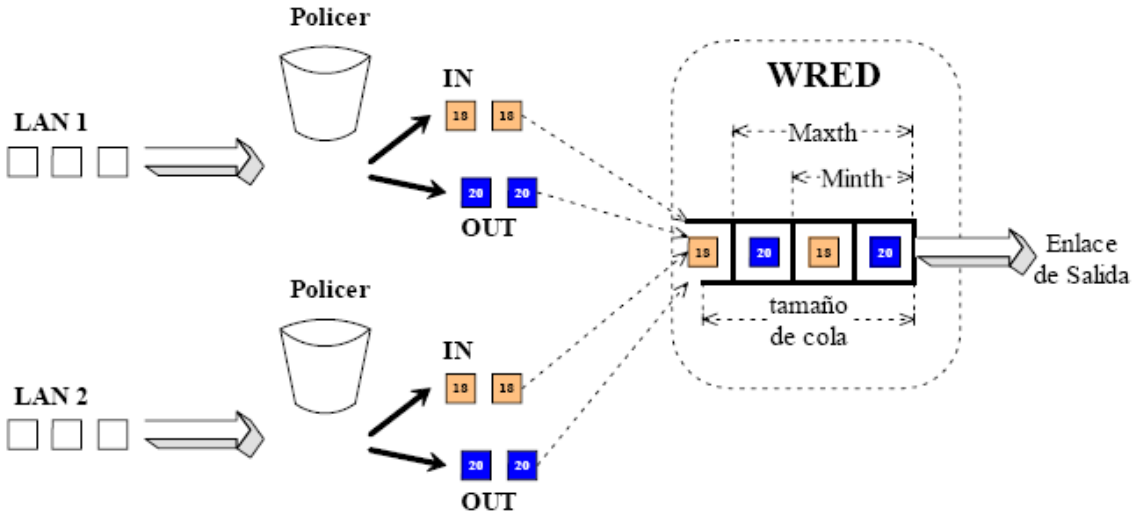

Figura 11: Funcionamiento de la Configuración de la cola FIFO con WRED

2.8.1 Configuraciones

Las listas de acceso, las clases y las políticas de policing son las mismas que en caso general. Lo único que se debe configurar en este caso es WRED en vez de Tail Drop como método de descarte de paquetes sobre la interfaz de salida serial 1/0. En el router se hará lo siguiente:

```
RouterA# configure terminal
RouterA(config)# interface serial 1/0
RouterA(config-if)# random-detect dscp-based
RouterA(config-if)# random-detect dscp 18 24 40 10
RouterA(config-if)# random-detect dscp 20 24 40 10
RouterA(config-if)# exit
```

Con esto se activa WRED a nivel de interfaz para que use los parámetros asociados a los valores de DSCP a la hora de descartar paquetes. Como en este caso sólo le llegarán al router paquetes marcados con los valores de DSCP 18 y 20 sólo se configurarán los parámetros de WRED para esos valores. El resto de parámetros que WRED tiene para cada uno de los valores de DSCP se dejarán a sus valores por defecto. Notar que se configuran los mismos perfiles de descarte para el tráfico IN y para el tráfico OUT de ambas redes.

El tráfico IN es el tráfico de valor DSCP 18 y el tráfico OUT es el tráfico con valor de DSCP 20. Se recuerda que los paquetes son marcados al entrar al Router A; los paquetes que cumplen el contrato se marcan con el valor de DSCP 18 y los que sobrepasan el contrato se marcan con el valor de DSCP de 20, se pueden observar prácticamente los mismos resultados que para la cola FIFO con Tail Drop, es decir, que los contratos se cumplen para todos las pruebas. El host cuyo contrato es menor en las pruebas, se lleva un mayor porcentaje del ancho de banda en exceso.

Al igual que en el caso de la cola FIFO, la prueba de que empleando solamente una cola con WRED con un solo perfil de descarte no se pueden garantizar de manera satisfactoria los contratos, la dan los resultados usando tráfico UDP.

```
RouterA#sh running-config
Building configuration...

Current configuration : 3362 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RouterA
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$.6kb$dOhHZDf5WAA2AJIrCMer70
enable password guia12345
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
!
!
!
class-map match-all trafico1.5-2.5
 match access-group 102
class-map match-all trafico3.14-2.5
 match access-group 101
!
!
policy-map TrafficPolicing3.14_2.5
  class trafico3.14-2.5
   police 256000 48000 96000 conform-action set-dscp-transmit 18
exceed-action set-dscp-transmit 20
policy-map TrafficPolicing1.5_2.5
  class trafico1.5-2.5
   police 256000 48000 96000 conform-action set-dscp-transmit 18
exceed-action set-dscp-transmit 20
!
!
```

```
interface FastEthernet0/0
 no ip address
 shutdown
 duplex half
!
interface Serial1/0
 ip address 192.168.120.254 255.255.255.0
 random-detect dscp-based
 random-detect dscp 18 24 40 10
 random-detect dscp 20 24 40 10
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/4
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/5
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/6
```

```
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface Serial1/7
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
!
interface FastEthernet2/0
 ip address 192.168.1.253 255.255.255.0
 duplex half
 speed auto
 service-policy input TrafficPolicing1.5_2.5
!
interface FastEthernet2/1
 ip address 192.168.3.253 255.255.255.0
 duplex half
 speed auto
 service-policy input TrafficPolicing3.14_2.5
!
interface Ethernet3/0
 no ip address
 shutdown
 duplex half
!
interface Ethernet3/1
 no ip address
 shutdown
 duplex half
!
interface Ethernet3/2
 no ip address
 shutdown
 duplex half
!
interface Ethernet3/3
 no ip address
 shutdown
 duplex half
!
interface Ethernet3/4
```

```
 no ip address
 shutdown
 duplex half
!
interface Ethernet3/5
 no ip address
 shutdown
 duplex half
!
interface Ethernet3/6
 no ip address
 shutdown
 duplex half
!
interface Ethernet3/7
 no ip address
 shutdown
 duplex half
!
router rip
 network 192.168.1.0
 network 192.168.3.0
 network 192.168.120.0
!
ip classless
ip route 0.0.0.0 0.0.0.0 FastEthernet2/0 192.168.1.254
no ip http server
no ip http secure-server
!
!
logging alarm informational
access-list 101 permit tcp host 192.168.3.14 host 192.168.2.5
access-list 101 permit udp host 192.168.3.14 host 192.168.2.5
access-list 102 permit udp host 192.168.1.5 host 192.168.2.5
access-list 102 permit tcp host 192.168.1.5 host 192.168.2.5
dialer-list 1 protocol ip permit
!
!
control-plane
!
!
!
!
gatekeeper
```

```
 shutdown
!
!
line con 0
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 password guia12345
 login
!
!
end

RouterA#
```

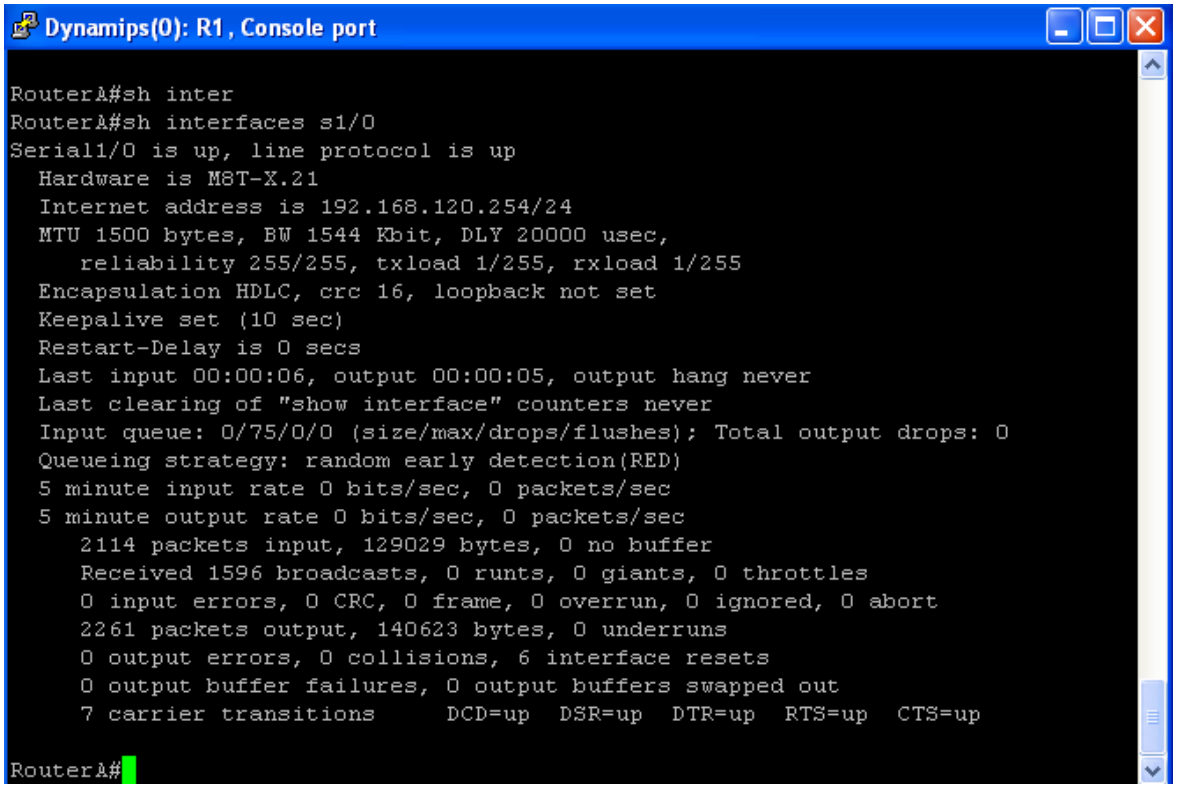

Figura 12: Resultados cola FIFO con WRED

En la figura 12 se puede apreciar claramente la imposibilidad de garantizar el cumplimiento de los contratos usando esta configuración. De manera física de podría observar que la fuente de tráfico UDP se queda con todo el ancho de banda del enlace ya que colapsa la cola con sus paquetes, provocando que las fuentes TCP reduzcan sus ventanas de transmisión enormemente.

Se puede apreciar también que se tiran muchos más paquetes del tráfico OUT, lo cual es normal ya que el Token Bucket no realiza descarte alguno y la mayoría de los paquetes UDP sobrepasaran el límite de lo contratado. Pero como no hay distinción entre los paquetes UDP o TCP, WRED descartará indistintamente paquetes de uno u otro protocolo, saliendo muy perjudicado el flujo TCP que reduce su ventana de transmisión al darse cuenta de los descartes.
Cabe aclarar que en un ambiente virtual con GNS3 es muy complicado y complejo poder capturar estos flujos.

2.9 Disciplina de Servicio WFQ

De nuevo, recordando sobre las disciplinas de servicio de colas, se sabe que FQ está diseñado para asegurar que cada flujo tenga un acceso justo a los recursos de la red y evita que un flujo de ráfagas consuma más ancho de banda que la

parte que le corresponde. En FQ, primero el sistema clasifica los paquetes en flujos y los asigna a una cola dedicada especialmente para ese flujo. Las colas se sirven siguiendo un tiempo en orden roundrobin.

Los beneficios de FQ son:

- El primer beneficio de FQ es que un flujo con demasiadas ráfagas o un flujo que no colabore no degradarán la calidad de servicio que reciban otros flujos debido a que se aísla a cada flujo en su propia cola.

- Si un flujo intenta consumir más de su ancho de banda, esto sólo afectará a su cola y por lo tanto no influirá en la ejecución de las otras colas.
- Por otro lado, WFQ es la base de un tipo de disciplinas para servir colas diseñadas para solucionar las limitaciones del modelo FQ:

- WFQ soporta flujos con diferentes requerimientos de ancho de banda. Esto lo logra dándole a cada cola un peso que le asigna un porcentaje diferente del ancho de banda de salida.

- WFQ también soporta paquetes de longitud variable de forma que los flujos con paquetes mayores no dispongan de un ancho de banda mayor que los flujos cuyos paquetes sean de menor tamaño.

Tal como lo muestra la figura 13.

Después de esta breve explicación, se va a configurar el router para emplear WFQ como disciplina de servicio de colas en la interfaz serial 1/0, que como se sabe está conectada directamente con el enlace de 2 M donde se produce la congestión. En este caso solo se mencionara su configuración en el router con GNS3, ya que no se cuenta con un generador de tráfico adecuado para tomar las mediciones.

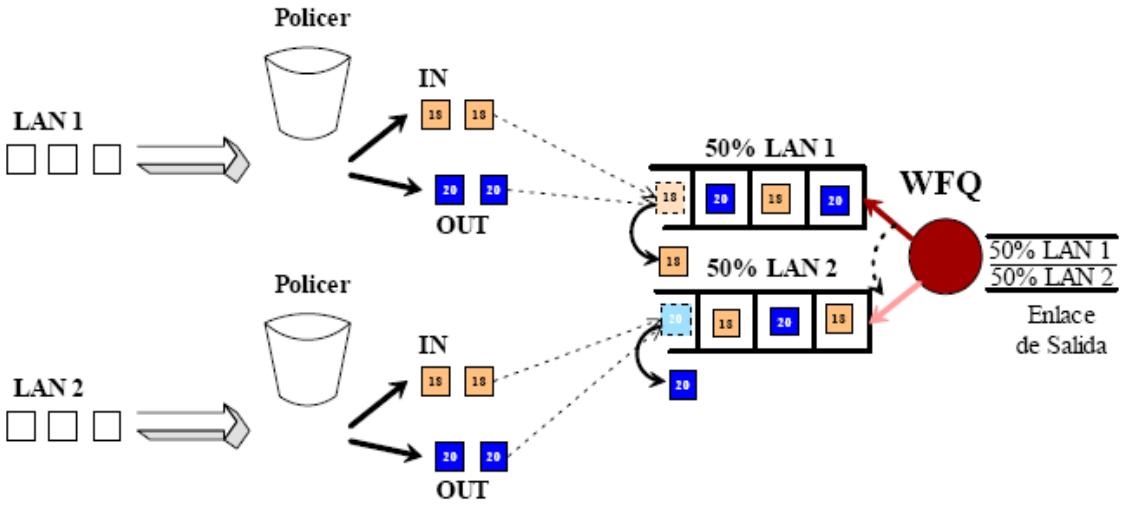

Figura 13: Funcionamiento de la Configuración de WFQ

2.9.1 Configuraciones

Las listas de acceso, las clases y las políticas de policing son las mismas que en caso general.

Por lo tanto, lo único que tendrá que hacer en este caso es activar WFQ en la interfaz serial 1/0. WFQ viene activado por defecto en esta interfaz pero si no fuera así el comando para activarlo es fair-queue. En el router se hará lo siguiente:

```
RouterA# configure terminal
RouterA(config)# interface serial 0/0
RouterA(config-if)# fair-queue
RouterA(config-if)# exit
```

En conclusión, aunque evidentemente WFQ funciona mucho mejor que las configuraciones anteriores usando Tail Drop o WRED para todo tipo de tráfico. Con ninguna de estas configuraciones se ha conseguido garantizar los contratos cuando alguno de ellos supera el 50% del ancho de banda del enlace de salida.
En las configuraciones siguientes se emplearán las herramientas de Cisco para ver si así se logran garantizar los contratos independientemente de su tamaño y del tipo de tráfico que envíen las redes.

2.10 Configuraciones con diferenciación de servicios

Como se ha comprobado con las pruebas anteriores, es imposible con las configuraciones realizadas garantizar contratos que superen el 50% de la capacidad disponible o repartir de manera equitativa el ancho de banda en exceso. Además solamente con WFQ se podían garantizar los contratos cuando las fuentes generaban tráfico UDP, siempre y cuando fuera inferior al 50% del ancho de banda del enlace.

En las configuraciones con diferenciación de servicios, se emplearán las herramientas que ofrece Cisco de manera virtual con GNS3 para implementar el

servicio Assured Forwarding de DiffServ, se llevarán a cabo pruebas y se comentarán los resultados obtenidos.

La topología será la misma que en las configuraciones, es decir, primero se clasifican los paquetes, luego se les aplica un Token Bucket y se marcan y por último llegan a la interfaz congestionada donde se les aplicará un tratamiento u otro dependiendo de la herramienta de DiffServ configurada.

Para este caso, también se llevarán a cabo tres configuraciones diferentes que serán:

- WRED con varios perfiles de descarte para cada tipo de tráfico.
- CBWFQ
- CBWFQ con WRED

2.10.1 WRED

La topología será prácticamente la misma que para la configuración anterior de WRED, pero en este caso se configurarán dos perfiles de descarte diferentes para los paquetes de la cola. Así, para los paquetes que los Token Bucket marquen como IN los parámetros de WRED serán; un umbral mínimo de 40, un umbral máximo de 70 y un mark probability denominador de 50.Y para los paquetes marcados como OUT por el Token Bucket los parámetros de WRED serán; un

umbral mínimo de 10, un umbral máximo de 40 y un mark probability denominador de 5. Todo esto quiere decir que se comenzarán a tirar antes los paquetes OUT que los IN dentro de la cola FIFO. Hay que decir que el tamaño máximo de la cola es de tamaño 200 aunque el umbral máximo de WRED sea de 70 paquetes.

Un esquema de la topología sería como se muestra en la figura 14:

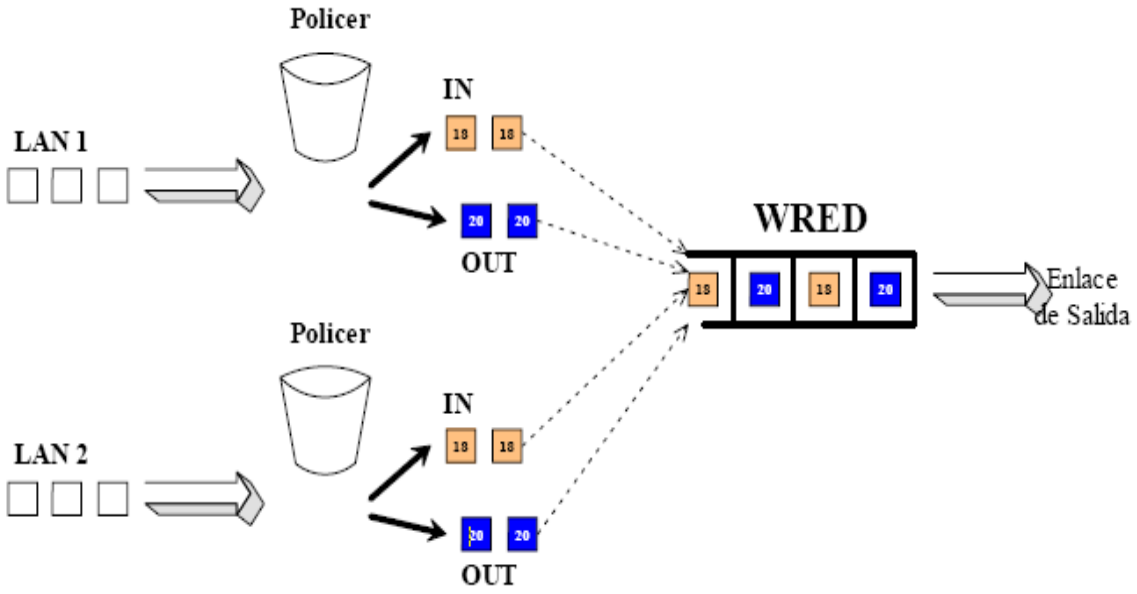

Figura 14: Funcionamiento de la Configuración de WRED de DiffServ

En la figura 14 se observa que el funcionamiento es muy parecido a las configuraciones anteriores, esto es, los paquetes se clasifican, se marcan a través de un Token Bucket y se envían a una cola FIFO con WRED. Al igual que en los casos anteriores, todos los paquetes que cumplen el contrato se marcan como IN y todos los paquetes que exceden el contrato se marcan como OUT, independientemente de cuál sea su fuente origen.

La diferencia, como se ha comentado, está en que en esta ocasión la cola FIFO tiene dos perfiles de descarte diferentes para cada tipo de tráfico.

2.10.1.1 Configuraciones

Las listas de acceso, las clases y las políticas de policing son las mismas que en caso general.

Por tanto, lo único que se deberá activar y configurar en este caso es WRED a nivel de interfaz con los parámetros adecuados. En el router se hará lo siguiente:

```
RouterA# configure terminal
RouterA(config)# interface serial 1/0
RouterA(config-if)# hold-queue 200 out
RouterA(config-if)# random-detect dscp-based
RouterA(config-if)# random-detect dscp 18 40 70 50
RouterA(config-if)# random-detect dscp 20 10 40 5
RouterA(config-if)# exit
```

Mediante el comando hold-queue 200 out se establece el tamaño de la cola en 200. Luego se activa WRED a nivel de interfaz y se asocian los parámetros adecuados a cada valor de DSCP.

2.10.1.2 Análisis de Resultados.

En la figura 15 se muestran los resultados obtenidos, por ser una herramienta Virtual GNS3, no se obtienen resultados, a diferencia de la implementación en

forma física. En ella se puede observar que ahora sí se garantizan los contratos aunque sobrepasen estos el 50% del ancho de banda del enlace. Se ve por otro lado que el ancho de banda en exceso sigue sin repartirse al 50% entre las dos fuentes al igual que ocurría en anteriores ocasiones.

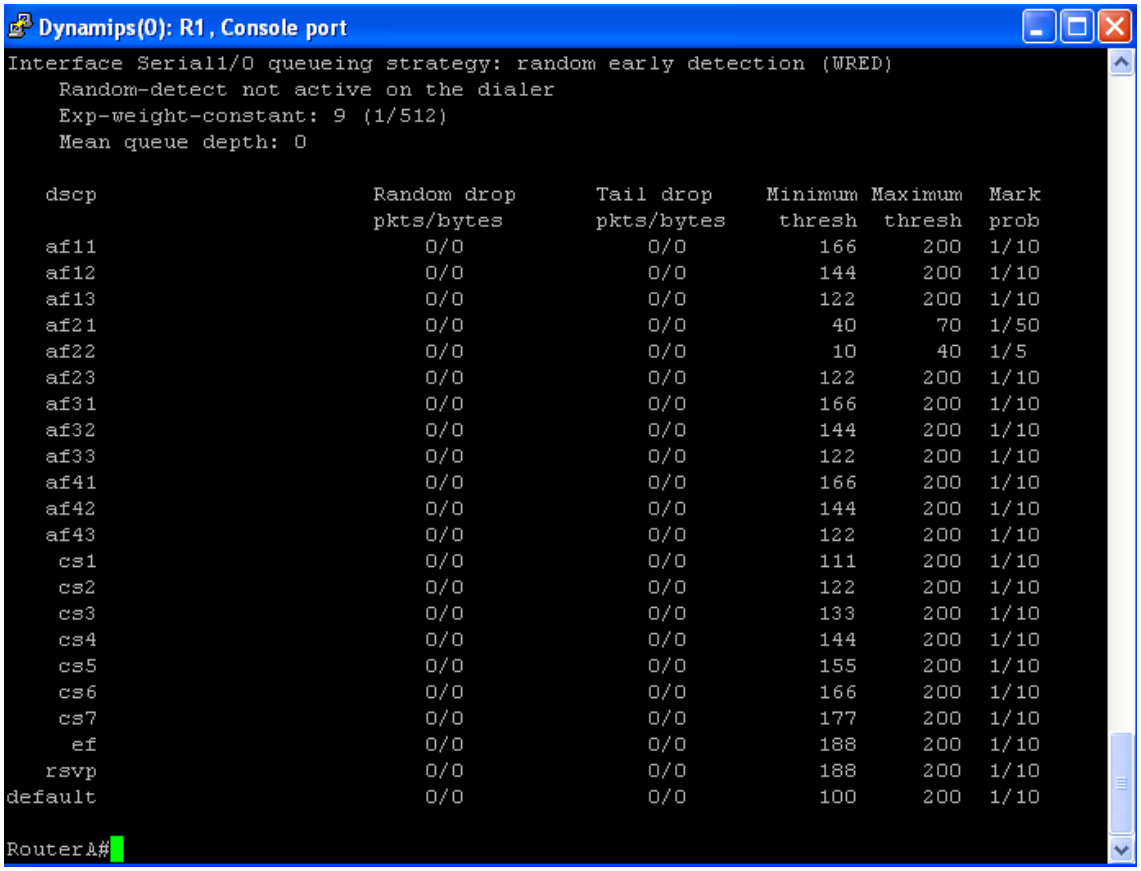

Figura 15:visualización de trafico WRED

En forma física se observarían los descartes realizados por WRED en la interfaz de salida del Router A. Se puede comprobar que sólo se realizan descartes mediante WRED y no por desbordamiento del buffer de la cola. También se puede observar que sólo se descartan paquetes marcados como OUT,

independientemente de cual sea su procedencia. Estos resultados los muestra el router empleando el comando show queueing interface serial 1/0

Como conclusión, con la configuración de WRED con varios perfiles de descarte que se ha implementado aquí se pueden garantizar contratos de cualquier ancho de banda siempre que las fuentes generadoras del tráfico sean fuentes colaboradoras, es decir, TCP.

2.10.2 CBWFQ

Se puede decir que WFQ tiene dos beneficios principalmente:

- Proporciona la protección de cada clase de servicio asegurando un nivel mínimo del ancho de banda del puerto de salida independientemente del comportamiento de otras clases de servicio.

- Cuando se combina con acondicionadores de tráfico en las entradas de una red, WFQ garantiza un reparto equitativo del ancho de banda del puerto de salida de cada clase de servicio con un retardo limitado.

Una mejora a WFQ es CBWFQ. CBWFQ asigna paquetes a colas basándose en criterios de clasificación de paquetes definidos por el usuario. Por ejemplo, los paquetes se pueden asignar a una cola en particular basándose en los bits del IP

DSCP. Después de que los paquetes sean asignados a sus colas, podrán recibir un trato prioritario en función de los pesos configurados por el usuario.

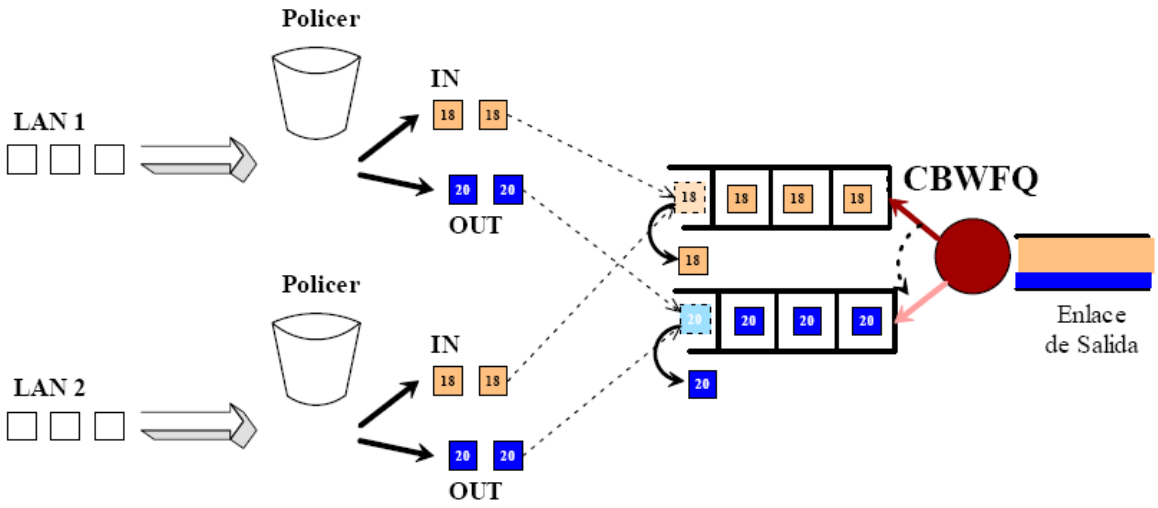

Figura 16:Funcionamiento de la Configuración de CBWFQ

En la figura 16 se muestra el proceso que lleva a cabo el Router A con los paquetes de las redes LAN 1 y 2 que le llegan. Primero, al igual que en casos anteriores, el tráfico se clasifica y se le aplica un Token Bucket que marca los paquetes como IN o como OUT. Después todos los paquetes IN de ambas redes se envían a una cola y todos los paquetes OUT se envían a otra.

Por último, CBWFQ se encargará de servir los paquetes de las colas. Los descartes de paquetes se realizarán cuando las colas se desborden.

El peso de cada una de las colas dependerá del contrato. Por ejemplo, si ambos contratos son de 256 Kbps, el total del tráfico IN será la suma de ambos, es decir,

512 Kbps (26% del enlace). El tráfico en exceso total será el resto del ancho de banda del enlace, esto es, 1,512 M (74% del enlace).

2.10.2.1 Configuraciones

Las listas de acceso, las clases y las políticas de policing son las mismas que en caso general. En este caso se crearán dos clases más; una para clasificar el tráfico IN y otra para el tráfico OUT.

Luego se creará una política que cree dos colas que se sirvan mediante CBWFQ. Como se ha expuesto a una de las colas irán a parar todos los paquetes IN y a la otra los OUT. En el router se haría lo siguiente:

Creación de las dos nuevas clases (además de las que había):

```
RouterA# configure terminal
RouterA(config)# class-map match-all traficoIN
RouterA(config-cmap)# match ip dscp 18
RouterA(config-cmap)# exit
RouterA(config)# class-map match-all traficoOUT
RouterA(config-cmap)# match ip dscp 20
RouterA(config-cmap)# exit
```

Creación de la nueva política (además de las que había):

```
RouterA# configure terminal
RouterA(config)# policy-map DiffServ
RouterA(config-pmap)# class traficoIN
RouterA(config-pmap-c)# bandwidth percent 26
RouterA(config-pmap-c)# exit
RouterA(config-pmap)# class traficoOUT
RouterA(config-pmap-c)# bandwidth percent 74
```

```
RouterA(config-pmap-c)# exit
RouterA(config-pmap)# exit
```

Se asocia la nueva política con la interfaz serial 1/0:

```
RouterA(config)# interface serial 1/0
RouterA(config-if)# service-policy output DiffServ
RouterA(config-if)# exit
```

Esta configuración es para el primero de los casos en el que los contratos de ambas redes eran iguales a 256 Kbps.

2.10.2.2 Análisis de Resultados.

En los resultados obtenidos de manera virtual no son apreciables, sin embargo de manera física se muestra que se cumplen perfectamente los contratos sea cual sea su tamaño. El problema sigue siendo el reparto del ancho de banda en exceso ya que continúa llevándose un porcentaje mayor la fuente cuyo contrato es menor. Se podría decir que de todas las configuraciones estudiadas, es la única que logra garantizar los contratos sea cual sea el tamaño y el tipo de tráfico. Esto se debe a que CBWFQ separa el tráfico en exceso en una cola y el tráfico IN en otra. Así, como la mayor parte del tráfico UDP es marcado como OUT, se aislará casi todo en una de las colas y no influirá en el tráfico IN de la otra. Esto permite garantizar los contratos, ya que el tráfico IN, independientemente de que sea UDP o TCP, no sobrepasará el ancho de banda del enlace.

El problema sigue siendo el reparto equitativo del ancho de banda en exceso. Sobre todo cuando hay tráfico UDP, se observaria que las fuentes de tráfico UDP se quedan con todo el ancho de banda en exceso utilizando esta configuración.

2.10.3 CBWFQ con WRED

Ya se han llevado a cabo configuraciones en ambiente simulado con GNS3 de CBWFQ y de WRED por separado y se han obtenido y comentado los resultados de sus pruebas correspondientes. En las siguientes pruebas se activarán ambas herramientas a la vez. Con esta configuración además de lograr garantizar los contratos, se solucionará el problema del reparto equitativo del ancho de banda en exceso que no se conseguía con ninguna de las otras configuraciones.

Primero, se clasificarán los paquetes separándolos por red de origen. Luego, se procesarán mediante un Token Bucket para ver si cumplen o no los contratos y serán marcados. Esta vez se utilizarán cuatro valores de DSCP diferentes, dos para cada red LAN. Así, los paquetes de la red LAN 1 que cumplan el contrato se marcarán con un valor de DSCP de 18 y los que lo sobrepasen con un DSCP de 20. Los paquetes IN de la red LAN 2 se marcarán con un DSCP de 10 y los OUT con un DSCP de 12.

Una vez marcados los paquetes, se enviarán a dos colas diferentes: una para los paquetes IN y OUT de la red LAN 1 y otra para los paquetes IN y OUT de la red LAN 2. Cada una de esas colas usará un WRED con dos perfiles de descarte

diferentes: uno para el tráfico IN y otro para el tráfico OUT. Ambas colas se servirán usando CBWFQ.

La siguiente figura 17 muestra un esquema de lo expuesto:

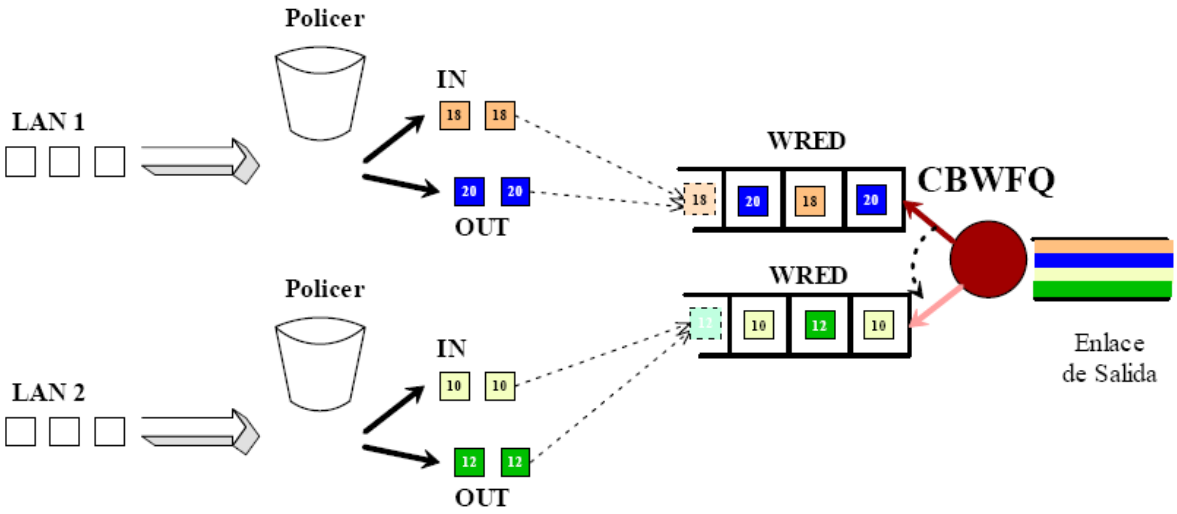

Figura 17:Funcionamiento de la Configuración de CBWFQ con WRED.

Los parámetros de WRED para ambas colas serán los mismos que cuando se usó WRED con doble perfil de descarte, es decir, para los paquetes que los Token Bucket marquen como IN los parámetros de WRED serán: un umbral mínimo de 40, un umbral máximo de 70 y un mark probability denominador de 50.Y para los paquetes marcados como OUT por el Token Bucket los parámetros de WRED serán: un umbral mínimo de 10, un umbral máximo de 40 y un mark probability denominador de 5.

Los parámetros de configuración de CBWFQ, esto es, el peso de cada una de las colas, dependerán del contrato y se establecerán de modo que se intente repartir de forma equitativa el ancho de banda en exceso, para solucionar los problemas de las anteriores configuraciones.

2.10.3.1 Configuraciones

En este caso, de la configuración realizada para el resto de pruebas sólo sirven la creación de las listas de control de acceso y sus clases correspondientes. Por lo tanto, en esta ocasión se crearán dos nuevas clases; una para el tráfico marcado con los DSCPs 18 y 20 y otra para el tráfico marcado con los DSCPs 10 y 12. También se creará una nueva política para configurar CBWFQ con WRED. Y uno de los Token Bucket creados para el resto de configuraciones será configurado para marcar los paquetes de una de las redes con los valores de DSCP 10 y 12 (recuérdese que antes ambos Token Bucket marcaban todos los paquetes con dos DSCPs, el 18 y el 20). La configuración en el router será:

Creación de las dos nuevas clases (además de las clases trafico3.14-2.5 y trafico1.5-2.5):

```
RouterA# configure terminal
RouterA(config)# class-map match-all traficoDSCP18_20
RouterA(config-cmap)# match ip dscp 18 20
RouterA(config-cmap)# exit
RouterA(config)# class-map match-all traficoDSCP10_12
RouterA(config-cmap)# match ip dscp 10 12
RouterA(config-cmap)# exit
```

Con estas clases se clasifican los paquetes a la entrada de la interfaz serial después de que hayan sido marcados por los Token Buckets.

Como se ha dicho, los traffic policing a la entrada del router serán iguales a los de las anteriores configuraciones salvo que ahora los valores de los DSCPs de cada una de las redes serán diferentes. Como ejemplo:

```
RouterA# configure terminal
RouterA(config)# policy-map TrafficPolicing3.14_2.5
RouterA(config-pmap)# class trafico3.14-2.5
RouterA(config-pmap-c)# police 256000 48000 96000 conform-action
set-dscp-transmit 18 exceed-action set-dscp-transmit 20
RouterA(config-pmap-c)# exit
RouterA(config-pmap)# exit
RouterA(config)# policy-map TrafficPolicing1.5_2.5
RouterA(config-pmap)# class trafico1.5-2.5
RouterA(config-pmap-c)# police 256000 48000 96000 conform-action
set-dscp-transmit 10 exceed-action set-dscp-transmit 12
RouterA(config-pmap-c)# exit
RouterA(config-pmap)# exit
```

Creación de la política de Servicios Diferenciados que emplea CBWFQ con WRED:

```
RouterA# configure terminal
RouterA(config)# policy-map DiffServ
RouterA(config-pmap)# class traficoDSCP18_20
RouterA(config-pmap-c)# bandwidth percent 50
RouterA(config-pmap-c)# random-detect dscp-based
RouterA(config-pmap-c)# random-detect dscp 18 40 70 50
RouterA(config-pmap-c)# random-detect dscp 20 10 40 5
RouterA(config-pmap-c)# exit
RouterA(config-pmap)# class traficoDSCP10_12
RouterA(config-pmap-c)# bandwidth percent 50
RouterA(config-pmap-c)# random-detect dscp-based
RouterA(config-pmap-c)# random-detect dscp 10 40 70 50
RouterA(config-pmap-c)# random-detect dscp 12 10 40 5
RouterA(config-pmap-c)# exit
```

```
RouterA(config-pmap)# exit
```

Mediante esta configuración se han creado dos colas; una para los paquetes marcados con los DSCPs 18 y 20 y otra para los paquetes marcados con los DSCPs 10 y 12. A cada una de esas colas se le asigna la mitad del ancho de banda del enlace de salida. Además, se configura WRED dentro de cada una de las colas para que se encargue del descarte de paquetes, descartando antes los paquetes marcados como OUT que los paquetes IN.

Se asocia la política con la interfaz de salida serial 1/0:

```
RouterA(config)# interface serial 1/0
RouterA(config-if)# service-policy output DiffServ
RouterA(config-if)# exit
```

En anteriores configuraciones, lo que parecía funcionar correctamente cuando el tráfico a tratar era tráfico TCP, no servía cuando alguna de las fuentes generaba tráfico UDP. Todo esto se logra mediante la separación de los tráficos de cada red en una cola diferente y la asignación de los pesos adecuados a cada cola. En el caso en el que se usaba CBWFQ con dos colas una para el tráfico IN y otra para el tráfico OUT se podían garantizar los contratos incluso con fuentes UDP ya que la mayor parte del tráfico UDP era marcado como OUT y enviado a una cola aislada, por lo tanto, no influía negativamente en el tráfico IN. Pero el problema estaba en la imposibilidad de repartir por igual el ancho de banda en exceso entre ambas redes, debido a que la red generadora de tráfico UDP consumía todos los recursos de la cola del tráfico OUT.

Separando el tráfico de las redes en colas separadas, el mal comportamiento del tráfico de una red no afectará al contrato de la otra. Usando WRED con dos perfiles de descarte distintos, se les dará preferencia a los paquetes IN frente a los OUT dentro de una misma cola, es decir, los paquetes que no cumplan el contrato tendrán más probabilidad de ser descartados que los paquetes que lo cumplan para el tráfico de una red.

3. CONCLUSIONES

A través de este trabajo se presenta una topología a la cual se le aplicó todas las políticas que encierra la arquitectura de servicios diferenciados e implementación de todos sus comandos como funciones de marcado, políticas de tráfico, entre otras en GNS3, logrando así demostrar que con su uso se puede brindar una buena QoS, además se hace referencia de cuales serian los servicios que aprovecharan los beneficios de esta tecnología.

También se inspeccionó que DiffServ brinda una solución a los problemas de escalabilidad que presenta IntServ. Esto lo logra a través de dar garantías a agregados de flujo y no a flujos individuales. Además en el modelo DiffServ el mayor beneficiario es el destino. Esto puede ser muy productivo en servicios basados en suscripciones, donde el usuario marca los paquetes con un determinado nivel de prioridad; los routers van agregando las demandas de los usuarios y propagándolas por el trayecto. Esto le da al usuario una confianza

razonable de conseguir la QoS solicitada. En servicios como Pay Per View (Video On Demand), canales de radio, canales de televisión, etc. Podrían aparecer en el modelo de negocio de redes DiffServ.

La utilización de DiffServ como QoS de extremo a extremo, como se ha visto reduce la carga de los dispositivos de red y es escalable fácilmente cuando la red crece también se puede decir que permite a los clientes mezclar dispositivos "compliant" de DiffServ con equipos existentes que soportan ToS aliviando los cuellos de botella a través de una gestión eficiente de los recursos actuales de la red.

Es normal que los diferentes tipos de tráfico emergentes actualmente en las redes IP tienen unas necesidades de retardo, varianza del retardo, caudal y pérdidas muy diversas y requieren un tratamiento individualizado. Por tal razón se ha presentado la arquitectura de los Servicios Diferenciados como una muy buena solución para solventar esas necesidades. En esta guía se ha definido los distintos elementos que la componen, como son:

- El dominio de Servicios Diferenciados en el que todos los nodos son capaces de proporcionar un tratamiento específico a los flujos de tráfico en función del valor del campo DSCP de la cabecera IP de cada paquete. Este dominio está compuesto de dos tipos de nodos: los nodos frontera que llevan a cabo funciones de clasificación y acondicionamiento del tráfico entrante o saliente

del dominio y los nodos interiores que le dan un tratamiento particular a cada paquete dependiendo del valor de DSCP de su cabecera IP. Existen dos tipos de clasificadores: los que clasifican basándose solamente en el valor del DSCP (BA classifiers) y los que clasifican basándose en el valor de la combinación de uno o más campos de la cabecera (Clasificadores Multi-Campo). Un acondicionador del tráfico puede contener los siguientes elementos: un medidor, un marcador, un espaciador o un descartador de tráfico.

- También se trata lo que es un PHB que es el tratamiento particular que un nodo realiza sobre un flujo de tráfico con unas características determinadas, es decir, las funciones de programación, encolado, espaciado, etc que un nodo le aplica a los paquetes de ese flujo. además que hay varios PHBs definidos por el IETF que tienen asignados ciertos valores de DSCP, como son: el PHB por defecto que significa tratamiento de Best Effort y tiene asociado el valor de DSCP '000000', el PHB selector de clase diseñado para preservar la compatibilidad con el anterior IP Precedence y tiene asociados un DSCP de la forma 'xxx000', el Expedited Forwarding PHB que significa proporcionar un servicio de pocas pérdidas, bajo retardo, pequeña varianza del retardo y asegurar un ancho de banda constante y el valor de DSCP asociado con este PHB es '101110' y finalmente el Assured Forwarding PHB por medio del cuál a los paquetes de un determinado flujo de tráfico se les reserva cierta cantidad del ancho de banda de la interfaz de salida y pueden ser descartados con diferente probabilidad.

Se observa que uno de los problemas que presenta el emplear los elementos de la arquitectura de servicios diferenciados a la hora de garantizar contratos son:

- El número de contratos diferentes no debe de ser excesivamente grande debido a que el número de DSCPs con los que se diferencian los paquetes de cada contrato son limitados.

- La configuración de estos contratos es relativamente compleja y además se debe implementar en todos los nodos del dominio de Servicios Diferenciados si se sigue la arquitectura definida por el IETF suponiendo un gran esfuerzo. Por ejemplo, en el caso de CBWFQ y CBWFQ con WRED se tiene que el peso de cada una de las colas depende del contrato y si los contratos varían o se añade alguno, esos pesos se tienen que volver a calcular.

- La dificultad de crecimiento debido a que cuando aparece un nuevo contrato se deben modificar todas las políticas de todos los nodos creadas, para adaptarse al nuevo reparto de ancho de banda del canal.

-

Para finalizar se puede que solamente con la última configuración llevada a cabo es posible asegurar los contratos sea cual sea su tamaño y para cualquier tipo de tráfico. Además de repartir de modo equitativo el ancho de banda en exceso.

Pero también hay que notar que conforme se han ido haciendo más precisas las configuraciones, se ha ido aumentando en complejidad y se ha ido disminuyendo en escalabilidad. Por ejemplo, las configuraciones sin Servicios Diferenciados no dependen del contrato por lo tanto su implementación es mucho más sencilla y su escalabilidad muy alta. Por otro lado, dentro de las configuraciones con Servicios Diferenciados, CBWFQ con WRED será más complicado de configurar y menos escalable que WRED con doble perfil de descarte, ya que en el primero se necesitan hacer cálculos específicos de los pesos de las colas según los contratos.

Anexo 1

TABLA COMPARATIVA GENERAL

		TCP		TCP+UDP	
	Configuración	**Garantizar Contratos**	**Reparto equitativo del ancho de banda en exceso**	**Garantizar Contratos**	**Reparto equitativo del ancho de banda en exceso**
Sin DiffServ	**FIFO con Tail Drop**	Sí, pero cuando ninguno de los contratos sobrepase el 50% del ancho de banda del alcance.	**NO**	**NO**	**NO**
	FIFO con WRED	Sí, pero cuando ninguno de los contratos sobrepase el 50% del ancho de banda del alcance.	**NO**	**NO**	**NO**
	WFQ	Sí, pero cuando ninguno de los contratos sobrepase el 50% del ancho de banda del alcance.	**NO**	**Sí, pero cuando ninguno de los contratos sobrepase el 50% del ancho de banda del alcance.**	**NO**
Con DiffServ	**WRED con doble perfil de Descarte**	Si, se cual sea el contrato	**NO**	**NO**	**NO**
	CBWFQ	Si, se cual sea el contrato	**NO**	**Si**	**NO**
	CBWFQ con WRED	Si, se cual sea el contrato	**Si**	**Si**	**Si**

GLOSARIO

AF: Assured Forwarding**.**Reenvío asegurado.

BBE: Better than Best Effort**.**Servicio de flujo de datos IP.

BW: BandWidth. Ancho de banda.

CBQ: Class Based Queing, Es un término general que se refiere a cualquier mecanismo basado en clases

CBWFQ: class-bassed Weighted Fair Queuing- permite un mayor control sobre las colas de tráfico y asignación del ancho de banda

CEF: Cisco Express Forwarding**.** Conjunto de funcionalidades de los routers Cisco para poder ejecutar MPLS

CSC: Class Selector Codepoint. Código selector de clase.

DE: Default Behaviour. Comportamiento por defecto. Tipo de PHB.

DCE: Data Comunication Equipment - equipo de terminación de circuito o de comunicación de datos

DTE: Data Terminal Equipment - equipo terminal de datos

DiffServ Differentiated Services. Servicios diferenciados.

DS Byte Differentiated Services de un paquete IPV4

DSCPDifferentiated Services Codepoint. Código de servicios diferenciados, campo que forma parte del byte DS de un paquete IPV4.

EF Expedited Forwarding. Reenvió acelerado. Tipo de PHB.

FIFO: FIRST IMPUT FIRST OUTPUT; el tipo mas sencillo de encolamiento

GNS3: Es un simulador gráfico de Red.

IOS: Internetwork Operating System) es el software utilizado en la gran mayoría de routers

PHB Per-Hop-Behaviour.Comportamiento por salto.

QoSQuality Of Service.Calidad de servicio.

RFCRequest For Comments. Documento de especificaciones del IETF

TAIL DROP: es un algoritmo para la congestión de colas

TCP Transmision Control Protocol. Protocolo de control de la transmisión.

TETraffic Engineering. Ingeniería de Tráfico.

TOKEN BUCKET: Es un algoritmo usado para controlar la cantidad de datos inyectados a la red, permitiendo el envío de ráfagas de éstos.

TOSType Of Service. Tipo de servicio.

UDPUser Datagram Protocol. Protocolo de datagramas de usuario.

VPCS Es una aplicación que permite simular PC y ejecutar comandos como ping o traceroute.

WRED: WRED (Weighted Random Early Detection) evita el efecto conocido como Sincronización Global.

WFQ: Weighted Fair Queuing (WFQ) se trata de una técnica de encolamiento que proporciona QoS en redes convergentes.Trata de evitar la congestión.

BIBLIOGRÁFIA

- Tanenbaum, A. (2003). Redes de Computadoras. (4ta Edición). México: Prentice Hall.
- Davidson, J. (2001). Fundamentos de Voz sobre IP. Madrid: Cisco Press.
- Sheppard, S. (2002). Convergencia de las Telecomunicaciones. McGraw-Hill.
- Keagy, S. (2001). Integración de Redes de Voz y Datos. Cisco Press.
- Stallings, W. (2004). Comunicaciones y Redes de Computadores. (7ma Edición). Madrid: Pearson-España.
- Boronat Seguí, Fernando, García Pineda, Miguel, Lloret Mauri, Jaime (2009) IPTV, la televisión por Internet. Editorial Vértice.

Web Grafía

- http://www.eurescom.de/~projects-workspace/P1000-series/P1012/presentations/pdf/2nd%
- http://www.google.com
- http://www.wikipedia.com
- http://www.monografias.com/trabajos14/softswitch/softswitch.shtml. Recuperado en septiembre de 2009.
- Blacio, G., Jiménez E. y López P. (2004). Diseño de una red WAN para transmitir voz sobre IP y su utilización futura como red alternativa para la telefonía fija en el Ecuador. Tesis para optar por el título de Ingeniero en Electrónica y Telecomunicaciones. Escuela Superior Politécnica del Litoral.
- http://www.eurescom.de/~projects-workspace/P1000-series/P1012/presentations/pdf/2nd%
- http://books.google.com.co/books?id=PvmZFX00mMYC&pg=PA83&lpg=PA83&dq=QoS+aplicado+a+iptv&source=bl&ots=iGiIBWpDyq&sig=5m_y943QZtdxJrrxB1ajDXF5u48&hl=es&sa=X&ei=YD2DT4HWOIKs9ATvleXxBw&ved=0CFEQ6AEwBA#v=onepage&q=QoS%20aplicado%20a%20iptv&f=false-

- https://www.tlm.unavarra.es/~daniel/docencia/rba/rba06_07/trabajos/resumenes/gr16-QoSEnIPTV.pdf.
- http://www.ie.itcr.ac.cr/faustino/Redes/Clase10/QoS.pdf
- ftp://www.avmeiecuador.com/pub/manuales/qos/practica-qos.pdf.
- http://rua.ua.es/dspace/bitstream/10045/15056/1/Pr3-2009-10-rev.pdf.

http://www.slideshare.net/prestonj_jag/calidad-de-servicio-en-redes.