




**Fase 6 - Evaluación de la Red NGN y QoS**

**Entregado por: Oscar Arturo Gómez Gualteros**

**Presentado a: Ing. Omar Albeiro Trejo Narváz  
Universidad Nacional Abierta y a Distancia – UNAD**

**CEADJAG  
Escuela de Ciencias Básicas, Tecnología e Ingeniería (ECBTI)  
Diplomado de Profundización en Redes de Nueva Generación  
Bogotá, Mayo de 2020**



## Tabla de contenido

Resumen .....	4
Abstract.....	4
Palabras Clave.....	4
Introducción.....	5
Objetivos.....	6
Objetivo General.....	6
Objetivos Específicos .....	6
Funcionamiento de un Servidor VoIP .....	7
Análisis de cada elemento de los Bloques .....	7
Elementos y consideraciones servicio IPTV .....	10
Descripción de un sistema IPTV (Televisión sobre el Protocolo IP) .....	10
Arquitectura.....	10
Calidad de servicio y calidad de la experiencia sobre redes IPTV .....	12
IPTV sobre ADSL.....	13
Protocolos implementados. ....	13
Mecanismos de calidad de servicio para redes IPTV .....	15
Mecanismos en la red de Core. ....	15
Mecanismos en la red de Acceso. ....	16
Protocolo MPLS sobre la red diseñada en la Fase 4.....	18
Ancho de banda entre las Sedes es de 100Mbps .....	18
Servicio IPTV entre las Sedes .....	18
Configuración de Multicast en los Router IPTV .....	19
Configuración de servicio IPTV entre las Sedes .....	21
Configuración de QoS .....	25
Conclusiones.....	29
Bibliografía.....	30

## Tabla de figuras

<i>Figura 1. Diagrama de Bloques funcionamiento Servidor VoIP. Elaboración propia. ....</i>	<i>7</i>
<i>Figura 2. Bandwidth 100000. Elaboración propia. ....</i>	<i>18</i>
<i>Figura 3. Diseño de red general. Elaboración propia. ....</i>	<i>18</i>
<i>Figura 4. Paso 1 Configuración de Multicast Medellín. Elaboración propia. ....</i>	<i>19</i>
<i>Figura 5. Paso 2 Configuración de Multicast Medellín. Elaboración propia. ....</i>	<i>19</i>
<i>Figura 6. Paso 1 Configuración de Multicast Barranquilla. Elaboración propia. ....</i>	<i>19</i>
<i>Figura 7. Paso 2 Configuración de Multicast Barranquilla. Elaboración propia. ....</i>	<i>20</i>
<i>Figura 8. Paso 1 Configuración de Multicast Bogotá. Elaboración propia. ....</i>	<i>20</i>
<i>Figura 9. Paso 2 Configuración de Multicast Bogotá. Elaboración propia. ....</i>	<i>20</i>
<i>Figura 10. Paso 1 Configuración de servicio IPTV Bogotá. Elaboración propia. ....</i>	<i>21</i>
<i>Figura 11. Paso 2 Configuración de servicio IPTV Bogotá. Elaboración propia. ....</i>	<i>21</i>
<i>Figura 12. Paso 3 Configuración de servicio IPTV Bogotá. Elaboración propia. ....</i>	<i>22</i>
<i>Figura 13. Paso 4 Configuración de servicio IPTV Bogotá. Elaboración propia. ....</i>	<i>22</i>
<i>Figura 14. Paso 5 Configuración de servicio IPTV Bogotá. Elaboración propia. ....</i>	<i>22</i>
<i>Figura 15. Paso 1 Configuración de servicio IPTV Barranquilla. Elaboración propia. ....</i>	<i>23</i>
<i>Figura 16. Paso 2 Configuración de servicio IPTV Barranquilla. Elaboración propia. ....</i>	<i>23</i>
<i>Figura 17. Paso 3 Configuración de servicio IPTV Barranquilla. Elaboración propia. ....</i>	<i>23</i>
<i>Figura 18. Paso 4 Configuración de servicio IPTV Barranquilla. Elaboración propia. ....</i>	<i>23</i>
<i>Figura 19. Paso 5 Configuración de servicio IPTV Barranquilla. Elaboración propia. ....</i>	<i>23</i>
<i>Figura 20. Paso 1 Configuración de servicio IPTV Medellín. Elaboración propia. ....</i>	<i>24</i>
<i>Figura 21. Paso 2 Configuración de servicio IPTV Medellín. Elaboración propia. ....</i>	<i>24</i>
<i>Figura 22. Paso 3 Configuración de servicio IPTV Medellín. Elaboración propia. ....</i>	<i>24</i>
<i>Figura 23. Paso 4 Configuración de servicio IPTV Medellín. Elaboración propia. ....</i>	<i>24</i>
<i>Figura 24. Mapa de clases y asignación de protocolos para VoIP. Elaboración propia. ..</i>	<i>25</i>
<i>Figura 25. Paso 1 Configuración de QoS Medellín. Elaboración propia. ....</i>	<i>26</i>
<i>Figura 26. Paso 2 Configuración de QoS Medellín. Elaboración propia. ....</i>	<i>26</i>
<i>Figura 27. Paso 3 Configuración de QoS Medellín. Elaboración propia. ....</i>	<i>26</i>
<i>Figura 28. Paso 1 Configuración de QoS Barranquilla. Elaboración propia. ....</i>	<i>27</i>
<i>Figura 29. Paso 2 Configuración de QoS Barranquilla. Elaboración propia. ....</i>	<i>27</i>
<i>Figura 30. Paso 3 Configuración de QoS Barranquilla. Elaboración propia. ....</i>	<i>27</i>
<i>Figura 31. Paso 1 Configuración de QoS Bogotá. Elaboración propia. ....</i>	<i>28</i>
<i>Figura 32. Paso 2 Configuración de QoS Bogotá. Elaboración propia. ....</i>	<i>28</i>
<i>Figura 33. Paso 3 Configuración de QoS Bogotá. Elaboración propia. ....</i>	<i>28</i>



## Resumen

Se detalla satisfactoriamente el propósito de una red IP dentro de una arquitectura NGN para el soporte de servicios convergentes en donde se realiza la creación y configuración de un entorno MPLS usando el programa de simulación GNS3.


Se analiza claramente temáticas como servicios de multimedia IPTV, VoIP, Calidad de Servicio (QoS) y los conceptos principales de las redes MPLS: Distribución de Etiquetas, Tráfico, Protocolos y VPN.

## Abstract

The purpose of an IP network within an NGN architecture for the support of converged services where the creation and configuration of an MPLS environment is carried out using the GNS3 simulation program is satisfactorily detailed.

Topics such as IPTV multimedia services, VoIP, Quality of Service (QoS) and the main concepts of MPLS networks are clearly analyzed: Label Distribution, Traffic, Protocols and VPN.

**Palabras Clave:** red IP, arquitectura NGN, entorno MPLS, GNS3, IPTV, VoIP, QoS, Etiquetas, Tráfico, Protocolos, VPN.






## Introducción

En la siguiente investigación se identifica de una manera clara, precisa y concisa el propósito de una red IP entro de una arquitectura NGN para el soporte de servicios convergentes.

Se analizara puntual y detalladamente las funciones, entidades y requisitos a nivel funcional de una arquitectura NGN utilizada en la interconexión de redes, respondiendo a los estándares definidos.

Adicionalmente se implementara servicios multimedia para un escenario definido a lo largo de esta temática a nivel de simulación, aplicando los conceptos de arquitectura funcional y garantizando la QoS.






## Objetivos

### Objetivo General

Identificar y analizar el propósito de una red IP dentro de una arquitectura NGN para el soporte de servicios convergentes.

### Objetivos Específicos

- Detallar claramente los componentes funcionales de un Servidor VoIP, identificando los elementos, protocolos y las consideraciones que se deben tener en cuenta para implementar el servicio de IPTV.
  - Analizar las consideraciones necesarias para la correcta funcionalidad de una determinada red de nueva generación, desde su diseño, simulación y la implementación del servicio IPTV y el QoS.
- 

## Funcionamiento de un Servidor VoIP

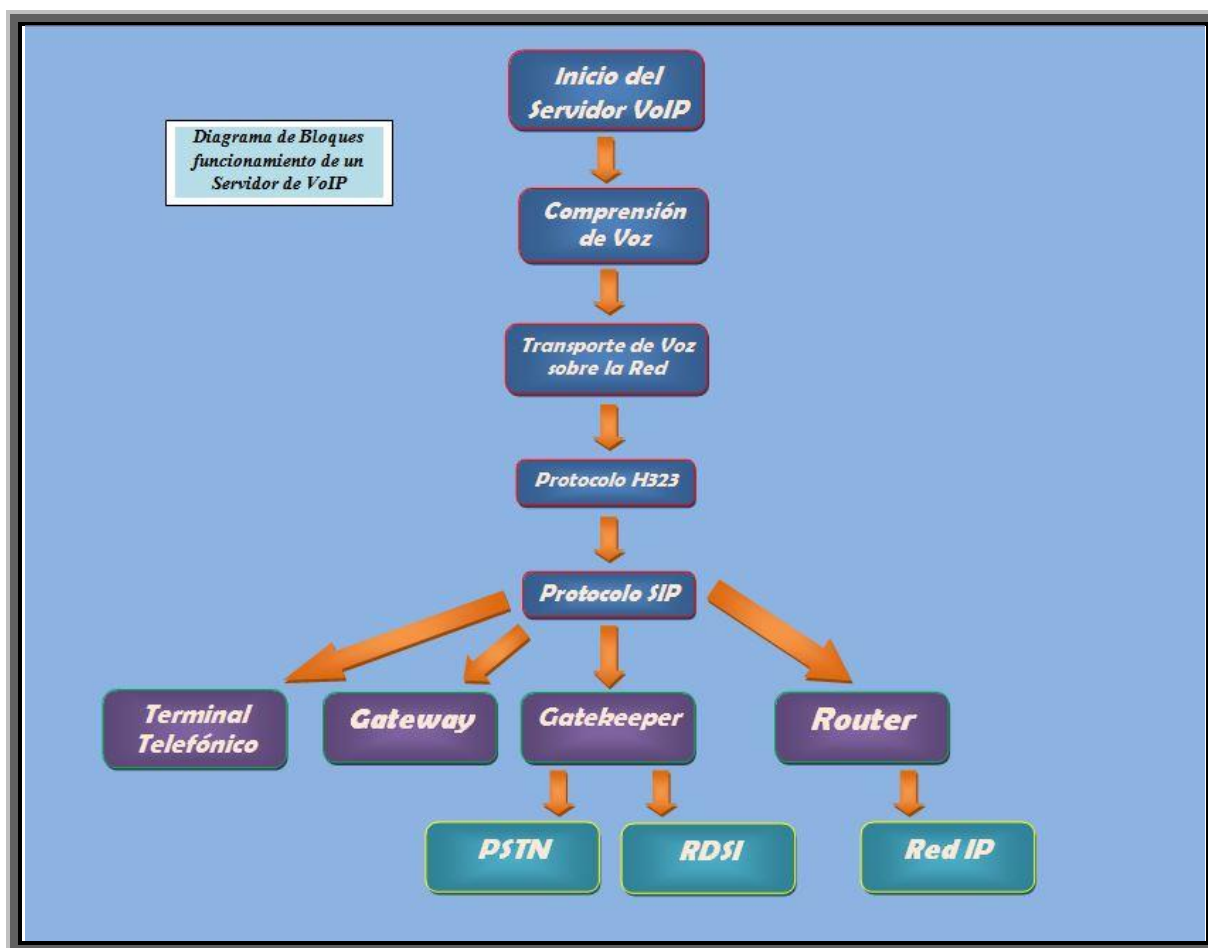



Figura 1. Diagrama de Bloques funcionamiento Servidor VoIP. Elaboración propia.

### Análisis de cada elemento de los Bloques

En primera instancia se debe detallar que el servidor de VoIP es un conjunto de recursos que hacen posible que la señal de voz viaje a través de Internet empleando el Protocolo IP (Protocolo de Internet). Esto significa que se envía la señal de voz en forma digital, en paquetes de datos, en lugar de enviarla en forma analógica a través de circuitos utilizables solo por telefonía convencional, como las redes PSTN (siglas de Public Switched Telephone Network, red telefónica pública conmutada) y también se encarga de manejar



operaciones de bases de datos, la recolección, el enrutamiento, la administración y control de servicio y el registro de los usuarios.

**Inicio del Servidor VoIP:** Inicia la operación del servidor para la comunicación de voz a través del protocolo IP.

**Comprensión de Voz:** Para la comprensión de la voz se utilizan diferentes métodos como los son la comprensión logarítmica y la modulación por impulsos modificados diferencial y adaptable (ADPCM), todo con el fin de comprender el mensaje que se transmite, teniendo en cuenta que la voz es codificada con la utilización de códecs y gracias a esta codificación se determinará que tanto ancho de banda se utilizará.

**Transporte de Voz Sobre la Red:** Teniendo en cuenta el gran éxito de las redes IP para el transporte de datos, se ha implementado el transporte de la voz sobre esta misma infraestructura y protocolo IP, gracias al empaquetamiento la información contenida en la voz y transmitida en forma de paquetes de datos IP; reemplazando así las redes telefónicas tradicionales PSTN (Red Telefónica Pública Conmutada).

**Protocolo H323:** Este protocolo fue diseñado para la administración, configuración y terminación de una sesión de comunicación, algo muy similar a la función del protocolo SIP, el cual es utilizado cada vez con más frecuencia.


**Terminal Telefónico:** Son los puntos de inicio y fin de la comunicación voz, pueden ser utilizados en forma de Hardware (Teléfonos IP físicos), y Software (Teléfonos IP a través de un Softphone, en una aplicación ejecutable desde el PC).

**Gateway:** Es el encargado de convertir, en tiempo real, las llamadas de voz generadas mediante una PSTN y las redes de datos IP. Dentro de sus funciones principales están la compresión y descompresión de la voz, empaquetamiento de la voz, enrutamiento de llamadas y señalización de control.


**Gatekeeper:** Ejecuta las funciones de gestión dentro de una red de voz IP o en las diferentes aplicaciones de intercambio de contenido multimedia como videoconferencia, entre otras. Los Gatekeepers suministran inteligencia de red, como lo evidencia en la resolución de direcciones IP, servicios de autenticación, autorización, entre otras funciones.

Gracias a su inteligencia de red, permite controlar de manera eficiente el ancho de banda, realizar un balance de carga y compatibilidad entre los diferentes sistemas.

**Router:** Este dispositivo permite la conexión de diversas estaciones de trabajo, con el fin de que compartan entre sí una única conexión a internet.








**PSTN:** Public Switched Telephone Network (Red Telefónica Pública Conmutada) red con conmutación de circuitos tradicional.

**RDSI:** (Red Digital de Servicios Integrados) facilita las conexiones digitales de extremo a extremo, entre los dispositivos que se encuentren conectados a esta. Por sus grandes costos de ejecución, no es ampliamente utilizada.

**Red IP:** Son todas las redes de datos e internet basadas en el protocolo IP. Provee conectividad entre todos los terminales.





## Elementos y consideraciones servicio IPTV

### Descripción de un sistema IPTV (Televisión sobre el Protocolo IP)

IPTV es un sistema que utiliza el protocolo IP para transportar los datos; se basa en el video Streaming y a diferencia de la televisión tradicional, su difusión se hace mediante Multicast o Unicast.

El envío de los contenidos se realiza desde los servidores, hacia los ordenadores o Set Top Box (STB) (dependiendo desde donde esté haciendo la recepción de la señal); dicha señal puede provenir de una señal satelital o una transmisión en vivo, si es el caso, primero tendrá que pasar por un codificador para convertir la señal de análoga a digital, después los datos serán encapsulados en paquetes IP, previo a su distribución la señal debe ser encriptada y posteriormente será transmitida hacia el usuario.

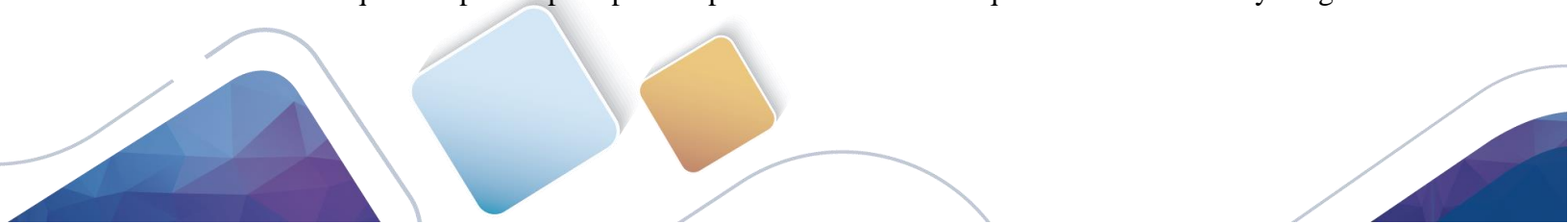
#### Arquitectura.


Las redes IPTV son sistemas conformados por numerosos elementos. Esta es una de las razones por las que existen diferentes arquitecturas utilizadas para transporte de servicios de video. Sin embargo, en general, es importante que dicha arquitectura contemple los siguientes componentes:

- Cabecera de la red
- Gestión de contenidos
- Red de distribución
- Red de acceso
- Red del cliente

1) *Cabecera.* En la cabecera de la red se encuentran los contenidos del proveedor. Es aquí donde se alojan los dispositivos encargados de recibir, transformar y distribuir los diferentes contenidos a los suscriptores.

Como es en la cabecera donde se reciben los contenidos, se debe tener en cuenta que se pueden dar dos situaciones, la primera, que los datos ya sean digitales y que por lo tanto no sea necesaria su digitalización, y la segunda, que el contenido sea una señal analógica. Si este es el caso, se tendría que pasar esta señal por los codificadores para digitalizar el video, se tendría que comprimir para poder optimizar los recursos que se van a utilizar y luego





pasaría a la etapa de encriptación, con el fin de que la señal no sea fácilmente interceptada en el transcurso de su recorrido desde el proveedor hasta el cliente.

2) *Middleware*. Es la plataforma que gestiona, controla y coordina la interacción entre los usuarios (mediante las peticiones enviadas por los STB) y los contenidos y servicios que se encuentran en la cabecera de la red.

La plataforma Middleware está basada en un conjunto de aplicaciones de software, las cuales se ejecutan sobre una cierta arquitectura de servidores, con el fin de soportar toda la interacción requerida para entregar de manera óptima una serie de servicios de IPTV.

3) *Sistema de gestión de contenidos*. Es el encargado de auto gestionar todo el ciclo de vida de los contenidos del proveedor. Desde la codificación online, la fijación de precios y el empaquetamiento, hasta la introducción de marketing en dichos contenidos.

Su meta es facilitar la gestión de los recursos comunes a todas las aplicaciones y servicios. Para que esta gestión sea mucho más fácil, se deben tener diferentes categorías de gestores, con el fin de que cada tipo se centre y especialice en una única función, logrando así el mejor desempeño y los mejores resultados.


4) *Red de distribución*. Será la que conectará la cabecera de red con el inicio de la red de acceso (donde se encontrarán los Router de agregación para Multicast). La red de distribución es la encargada de la conmutación y la transmisión de datos.


Esta red debe ser de alta capacidad ya que el flujo de datos que pasa por ella es bidireccional; es importante que esta red cuente con esta alta capacidad de transferencia puesto que es necesario ofrecer buena calidad a los clientes.

En esta parte de la arquitectura se encuentran los Router de agregación, estos dispositivos son una parte fundamental del sistema y por lo tanto deben tener ciertas características para poder cumplir a cabalidad sus funciones. Dichas características son:

- Calidad de servicio por abonado y servicio
- Soporte de Multicast IP
- Control de acceso para conmutación Ethernet
- Soporte de protocolos de señalización
- Alta disponibilidad
- Transición de IPv4 a IPv6
- Escalabilidad y rendimiento

5) *Red de acceso*. Es el enlace entre el proveedor y el usuario, es decir, dicha red termina donde comienza el equipo del usuario. Esta red está conformada por diferentes elementos,





los cuales están encargados de entender y suministrar los contenidos de las peticiones hechas por los usuarios.

Es necesario que esta red provea el suficiente ancho de banda al cliente, pues deberá soportar múltiples canales de televisión (HD y SD) y reservar una porción de ancho de banda suficiente para los demás servicios. En la actualidad existen diferentes tecnologías de acceso, como son: xDSL, HFC, FFTx, entre otras.

6) *Red del cliente.* Permite el paso de información entre los diferentes dispositivos que tienen la posibilidad de conectarse a ella y acceder a los recursos que esta le ofrece. Para poder disponer de dichos servicios, cada dispositivo debe estar conectado (sea por red cableada o inalámbrica) al Gateway residencial, el cual actuará de puente entre la red de acceso y la residencial.

Los dispositivos más comunes en esta red son: los Set Top Box (STB), los computadores (portátiles o de mesa), Teléfonos IP y SmartPhones.


### **Calidad de servicio y calidad de la experiencia sobre redes IPTV**

En las redes que prestan servicios de IPTV la QoS suele ser una medida del rendimiento de la red. Calidad de servicio incluye cualquier mecanismo que contribuya a la mejora del rendimiento general de la red y, por lo tanto, a la mejora de la calidad de la experiencia del usuario final.

Dichos mecanismos se pueden implementar en diferentes niveles. Por ejemplo, a nivel de red, con políticas para la gestión de tráfico, tales como el almacenamiento en buffers o la diferenciación de servicios y el manejo de prioridades en los dispositivos de red; a nivel de transporte, con la ocultación de pérdida de paquetes y la aplicación de Forward Error Correction (FEC). En general, existe una relación entre la calidad de la experiencia, que es subjetiva, y varios parámetros objetivos del rendimiento de los servicios, como por ejemplo, la tasa de bits de codificación, la pérdida de paquetes, los retardos, el Jitter y la disponibilidad. En general, los niveles de rendimiento de la red tendrán impacto sobre la calidad de la experiencia; por lo tanto, la relación entre estos dos conceptos se da en dos vías:

a) midiendo la calidad de servicio se podría predecir la calidad de la experiencia que percibirán los usuarios.

b) dado un objetivo de la calidad de la experiencia que se le quiera presentar a los usuarios, se podría deducir el rendimiento requerido en la red.





## **IPTV sobre ADSL.**

Diversas degradaciones pueden suceder desde cuando el contenido sale de los servidores de IPTV, pasa por la red de núcleo y la red de acceso del proveedor hasta el cliente.


Algunos de los problemas más frecuentes suceden por: ancho de banda insuficiente, errores en la transmisión en el acceso ADSL, calidad en la compresión del video, la no priorización del tráfico de video y el descarte de paquetes. Al ser ADSL una de las tecnologías de acceso más utilizadas actualmente, es importante tener claro sus bondades y limitaciones, para poder saber que problemas se pueden encontrar en el despliegue del servicio.

Cuando se diseña una red para IPTV es necesario calcular el ancho de banda que se necesitará para soportar todos los servicios. Es necesario tener en cuenta que habrá dos capacidades a calcular, una para el *Core* de la red y otro para la red de Acceso. Esta última debe ser bastante robusta, ya que sin importar cuál sea el número de suscriptores, necesitará soportar la totalidad de los canales, tanto SDTV como HD. Sin embargo, en la red de acceso es posible lograr una eficiente optimización de su rendimiento; una forma de hacerlo es que el DSLAM no reciba el tráfico de todos los canales, sino que se suscriba a los canales que pedidos por los clientes que estén conectados a él.


El DSLAM es el último dispositivo del lado del proveedor que manipula los flujos de datos que viajan desde la red IP hacia los usuarios y por lo tanto es el que hará que la calidad de servicio sea realmente de extremo a extremo; por otro lado, será el que gestione las peticiones de los usuarios y, en consecuencia, el que tendrá una gran influencia en los tiempos de respuesta generados a los clientes. Por estas razones, es muy importante que este dispositivo sea dimensionado acorde con los clientes que deberá atender y claro está, que se configuren adecuados mecanismos de calidad de servicio

Es necesario anotar que la mayoría de los proveedores han migrado a los IP-DSLAM para poder sacar el mayor provecho a sus redes de acceso. Estos DSLAM son capaces de manejar VLANs para la separación de flujos de tráfico; es de esta manera como garantizan QoS.

## **Protocolos implementados.**



Para presentar el escenario completo de protocolos utilizados en la red IPTV se requiere detallar las técnicas para la transmisión de video sobre IP. Por lo general se utilizan protocolos de *Streaming* de video, como RTP y *Multicast*, para la difusión de los contenidos. En los siguientes apartados se presenta una explicación de los protocolos más



utilizados en estas redes y posteriormente se muestra cómo estos ayudan al correcto funcionamiento del servicio.


1). *Real-time Transport Protocol [RTP]*. Fue diseñado por el *Audio-Video Transport Working Group* del IETF, planteado para soportar aplicaciones de tiempo real como el flujo multimedia en Internet. RTP es un protocolo ligero el cual por sí solo no proporciona reserva de recursos, ni mecanismos de control de flujo y calidad de servicio. Generalmente viaja sobre los paquetes UDP puesto que ellos poseen un menor retardo que TCP; pero a cambio, sacrifica la confiabilidad que ofrece TCP. Por esta razón, RTP no garantiza la entrega confiable y ordenada de paquetes, por lo que, para esta tarea, es necesario que sea acompañado por otros protocolos de control.


2). *Real-time Control Protocol [RTCP]*. Protocolo complementario a RTP que brinda mecanismos de control sobre las sesiones establecidas con éste, como por ejemplo, información básica sobre los participantes de la sesión y la QoS. RTCP establece comunicación utilizando el puerto adyacente al siguiente puerto que está siendo utilizado por RTP. Este está basado en la transmisión periódica de paquetes de control hacia los participantes de la sesión, ofreciendo información adicional de la calidad de los datos distribuidos desde la fuente hacia el destino.

3). *Real Time Streaming Protocol [RTSP]*. Es un protocolo no orientado a conexión para sistemas de Streaming, que se utiliza para controlar uno o muchos flujos de datos, ya sean de audio o de video, sobre redes IP. Este protocolo le permite al usuario establecer control sobre el contenido que está descargando (reproducir, pausar, retroceder, avanzar), este control se logra mediante el establecimiento de una conexión TCP para enviar los mensajes, permitiéndole al usuario establecer funciones de mando a distancia a través de la red. Cabe resaltar que RTSP no es un protocolo de envío de información sino de control; para esta labor se utiliza el protocolo RTP.

4). *Multicast*. Es el método generalmente utilizado para la transmisión de flujos de datos en redes IP; a través de él, es posible la entrega de datos a sistemas interconectados de la misma o de diferentes redes. Sin embargo, con el rápido crecimiento y alta exigencia de los servicios multimedia, como gráficos, voz y video, se ha hecho necesaria la transmisión de una gran cantidad de datos, los cuales afectan la carga tanto de las redes como la de los servidores que suministran el contenido. Con Multicast solo una copia del paquete de datos se envía desde el servidor hacia la red, utilizando solo una IP para todos los grupos de receptores, lo cual permite liberar a los enlaces y a los servidores de la carga extra que representan estos múltiples flujos.

5). *Multicast en un ambiente IPTV*. En redes que prestan el servicio de IPTV se utiliza tanto *Unicast* como *Multicast*. Habitualmente los canales en vivo se transmiten por Multicast ya que se espera que sean observados por una gran parte de los usuarios y su envío mediante Unicast llenaría los enlaces de información redundante. Por el contrario en





un servicio como VoD, que por naturaleza es punto a punto, ya que un usuario requerirá un contenido en específico, se suele utilizar Unicast.

A continuación se explicarán algunos de los protocolos que soportan Multicast, pero desde el enfoque de una red IPTV, y como estos hacen posible el correcto despliegue de los servicios ofrecidos por IPTV.

1) *Internet Group Multicast Protocol [IGMP]*: Es el protocolo usado por las redes de IPTV para unirse a los grupos Multicast, cuando hay un flujo Multicast disponible en la red; los STB tendrán que unirse a dicho grupo para poder recibir el tráfico Multicast; de igual manera, cuando el usuario no desea recibir más datos de ese flujo, el STB mandará un mensaje de *leave* para notificar que no quiere recibir más contenidos de ese grupo.

2) *Multicast e IGMP para el cambio de canal*. Cuando un usuario desea cambiar de canal, el tiempo que demora desde que es presionado el botón del control, hasta que el contenido del canal escogido se despliega en la pantalla, es conocido como *tiempo de zapping* de canal o cambio de canal.

Cuando el usuario cambia de canal, manda dos mensajes, un *leave* asociado al canal que estaba visualizando, y un *join* para asociarse al nuevo canal que desea ver.


En IGMPv3 es posible optimizar esta señalización, puesto que se puede combinar estos dos mensajes en una sola señal que será mandada a la red; una vez es recibida esta señal, el tráfico asociado con el canal actual es frenado en el DSLAM y posteriormente es asociado con el nuevo canal; en este momento el DSLAM empieza a transmitir el contenido.


## Mecanismos de calidad de servicio para redes IPTV

### Mecanismos en la red de Core.

Como se pudo ver en los anteriores apartados, IPTV es sensible a ciertos problemas puntuales de las redes, los cuales son retardo, Jitter y pérdida de paquetes. Para mejorar la calidad del servicio y mitigar estos problemas, es necesario que en los dispositivos de conexión (especialmente en los enrutadores y en el DSLAM) se configuren ciertos mecanismos de calidad de servicio que garanticen una adecuada priorización del tráfico IPTV. A continuación se presentan dos enfoques tanto para la red de Core como para la red de Acceso, los cuales son ideales según las características del tráfico IPTV.

1) MPLS (*Multiprotocolo de Conmutación de Etiquetas*) es un estándar de conmutación de paquetes propuesto por la IETF, que ofrece algunas características de las redes





orientadas a conexión, a las que no lo son. MPLS ofrece varios beneficios entre ellos: la ingeniería de tráfico de IP sobre ATM, la operación y el diseño de red más sencillo, puesto que está diseñado para operar sobre cualquier tecnología a nivel de enlace y mayor escalabilidad.

En MPLS a cada trama se le asigna un identificador (una etiqueta corta de longitud fija), el cual le indicará a los enrutadores el camino que deben seguir los datos. MPLS permite poner estas etiquetas según algunos parámetros por ejemplo, dirección IP fuente, camino que deben seguir los datos indiferente de su fuente o destino y de acuerdo con el tipo de servicio. Puesto que en MPLS es posible etiquetar los paquetes según el tipo de servicio, se convierte en uno de los mecanismos de calidad de servicio más efectivos. MPLS define cinco clases de servicio conocidos como CoS: video, voz, datos de alta prioridad, datos de prioridad y datos no prioritarios.


2) *RSVP (Protocolo de Reserva de Recursos)*. Protocolo de control de la red que trabaja en conjunto con el protocolo de transporte RTP, puesto que su diseño fue influenciado por los requerimientos de aplicaciones de video, y permite la reserva de ancho de banda para asegurar la QoS a través de toda la red de Core. RSVP diferencia cada paquete, lo marca y lo clasifica para darle un trato diferenciado, dependiendo del servicio que requiera.

La funcionalidad de este protocolo de reserva se basa en tres cosas: sesión, especiación del flujo y especificación del filtro. En la parte de sesión se identifica el flujo de datos según su destino y se hace la reserva de los recursos por un camino en particular, cada dispositivo entiende esto como el establecimiento de una sesión y le asigna los recursos; en la especificación del flujo se indica la calidad de servicio que se desea recibir para posteriormente ser tratado de forma preferente en cada nodo; por último, en la especificación del filtro define el conjunto de paquetes a los cuales se les hará la reserva y por lo tanto a cualquier otro paquete diferente (así vaya hacia el mismo destino) no se le dará calidad de servicio.


RSVP es un protocolo ideal para IPTV puesto que fue hecho para funcionar con tráfico Multicast (en este caso primero es necesario que se establezca el enlace mediante IGMP y posteriormente por RSVP) y de carácter simplex; es decir, está orientado a realizar calidad de servicio desde la fuente hacia el receptor puesto que es el receptor el que recibe los contenidos y las solicitudes que hace este no necesitan que se les reserve recursos.

### **Mecanismos en la red de Acceso.**

En las redes de Acceso donde la tecnología de transmisión es ADSL la manera más óptima de ofrecer calidad de servicio es mediante la configuración de diferentes VLANs.







Esto se logra mediante el etiquetado y encapsulamiento por parte del DSLAM de los diferentes flujos provenientes de la red de Core y la transmisión prioritaria de los flujos que lo requieren.

En las especificaciones formuladas por el DSL Forum, se propone dos formas de agregar a los usuarios a la red en cuanto a VLANs se refiere, una de ellas es tener una VLAN por cada suscriptor y la otra es tener una VLAN por cada servicio ofrecido. La ventaja de utilizar VLANs es que permite la segregación de tráfico, sea por tipo de servicio o cliente, y en algunos casos una combinación de ambos.


1) *VLANs dedicada para cada servicio.* Al tener este mecanismo, es posible definir una VLAN por servicio a través de la red. Por ejemplo, en este momento algunos proveedores tienen una VLAN dedicada para la parte de Internet y otra para la voz sobre IP. Este diseño permite que el uso del ancho de banda sea más eficiente, sobre todo para servicios de multidifusión como son algunos contenidos de IPTV.

Tener una VLAN por cada servicio permite a la red ofrecer calidad de servicio en función de cada una de ellas y reservar ancho de banda en la red de núcleo.

2) *VLANs dedicadas para cada suscriptor.* Este enfoque dedica a cada suscriptor una VLAN diferente y todos los servicios que requiera dicho cliente serán suministrados mediante ella. Sin embargo, los 12 bits que están reservados para el ID de las VLANs solo proporcionarán la capacidad de tener 4096 VLANs. Este total de clientes, a través de una red, sería una gran limitación y, por lo tanto, en este enfoque se utilizan las VLANs apiladas.

Una de las principales ventajas de este mecanismo es que le permite al Router de Borde gestionar de manera más fácil el ancho de banda para cada suscriptor, puesto que, primero limita el tráfico en el DSLAM y posteriormente asegura el ancho de banda para cada suscriptor.

Por otro lado, una de las desventajas es que este mecanismo funciona bien si la mayoría del tráfico es Unicast (por ejemplo video bajo demanda) a causa de que las rutas serían punto a punto y no soportarían la replicación mediante multidifusión entre VLANs; esto implica que es necesario adicionar una VLAN más para distribuir el tráfico de Multicast al igual que los mensajes IGMP.



## Protocolo MPLS sobre la red diseñada en la Fase 4

Ancho de banda entre las Sedes es de 100Mbps

```

Router(config)#int fa1/0
Router(config-if)#bandwidth 100000
Router(config-if)#show int fa1/0

% Invalid input detected at '^' marker.

Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#do show int fa1/0

% Invalid input detected at '^' marker.

Router#show int fa1/0
FastEthernet1/0 is up, line protocol is up (connected)
Hardware is Lance, address is 0002.17ca.e668 (bia 0002.17ca.e668)
Internet address is 192.168.10.1/24
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Full-duplex, 100Mb/s, media type is RJ45
    
```

Figura 2. Bandwidth 100000. Elaboración propia.

Servicio IPTV entre las Sedes

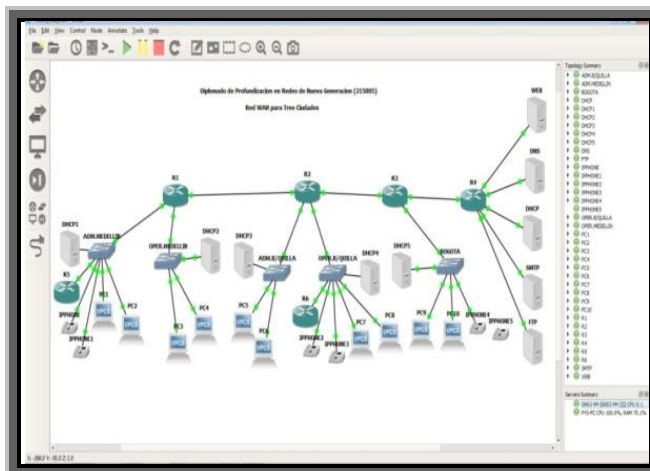


Figura 3. Diseño de red general. Elaboración propia.

## Configuración de Multicast en los Router IPTV

```

#enable
#show ip int brief
Interface          IP-Address      OK? Method Status    Protocol
FastEthernet0/0    192.168.23.1    YES NVRAM  up        up
FastEthernet0/1    192.168.24.1    YES NVRAM  up        up
Loopback0          1.1.1.1         YES NVRAM  up        up
#configure
#router configuration commands, one per line. End with CNTL/Z.
R1(config)#ip multicast-routing
R1(config)#ip pim rp-address 1.1.1.1
R1(config)#int f0/0
R1(config-if)#ip pim sparse-mode
R1(config-if)#
Mar 1 00:04:58.027: SPTN-S-DRCNG: DR change from neighbor 0.0.0.0 to 192.168.23.1 on interface FastEthernet0/0
R1(config-if)#ip pim sparse-mode
R1(config-if)#
% Invalid input detected at '^' marker.
R1(config-if)#ip pim sparse mode
% Invalid input detected at '^' marker.
R1(config-if)#ip pim sparse-mode
R1(config-if)#
Mar 1 00:07:00.979: SPTN-S-DRCNG: DR change from neighbor 0.0.0.0 to 192.168.24.1 on interface FastEthernet0/1
R1(config-if)#int lo0
R1(config-if)#ip pim sparse-mode
R1(config-if)#
Mar 1 00:08:58.979: SPTN-S-DRCNG: DR change from neighbor 0.0.0.0 to 1.1.1.1 on interface Loopback0
R1(config-if)#end
Mar 1 00:09:18.311: SSVS-S-COMP10_1: Configured from console by console
#
Building configuration...
[OK]

```

Figura 4. Paso 1 Configuración de Multicast Medellín. Elaboración propia.

```

R1#
Mar 1 00:11:21.083: SPTN-S-NBRCHG: neighbor 192.168.23.2 UP on interface FastEthernet0/0
Mar 1 00:11:21.099: SPTN-S-DRCNG: DR change from neighbor 192.168.23.1 to 192.168.23.2 on interface FastEthernet0/0
R1#
Mar 1 00:14:02.683: SPTN-S-NBRCHG: neighbor 192.168.24.2 UP on interface FastEthernet0/1
Mar 1 00:14:02.707: SPTN-S-DRCNG: DR change from neighbor 192.168.24.1 to 192.168.24.2 on interface FastEthernet0/1
R1#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, H - MSDP created entry,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - NPT-data group sender,
Y - Joined VDT-data group, y - Sending to VDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.0.1.40), 00:12:00/00:02:09, RP 1.1.1.1, flags: SPL
Incoming interface: Null, RPF nbr 0.0.0.0

```

Figura 5. Paso 2 Configuración de Multicast Medellín. Elaboración propia.

```

#enable
#show ip int brief
Interface          IP-Address      OK? Method Status    Protocol
FastEthernet0/0    192.168.23.2    YES NVRAM  up        up
FastEthernet0/1    192.168.25.1    YES NVRAM  up        up
Loopback0          2.2.2.2         YES NVRAM  up        up
#configure
#router configuration commands, one per line. End with CNTL/Z.
R2(config)#ip multicast-routing
R2(config)#ip pim rp-address 1.1.1.1
R2(config)#int f0/0
R2(config-if)#ip pim sparse-mode
R2(config-if)#
Mar 1 00:11:12.379: SPTN-S-NBRCHG: neighbor 192.168.23.1 UP on interface FastEthernet0/0
Mar 1 00:11:13.311: SPTN-S-DRCNG: DR change from neighbor 0.0.0.0 to 192.168.23.1 on interface FastEthernet0/0
R2(config-if)#int f0/1
R2(config-if)#ip pim sparse-mode
R2(config-if)#
Mar 1 00:11:31.263: SPTN-S-DRCNG: DR change from neighbor 0.0.0.0 to 192.168.25.1 on interface FastEthernet0/1
R2(config-if)#int lo0
R2(config-if)#ip pim sparse-mode
R2(config-if)#
Mar 1 00:12:48.263: SPTN-S-DRCNG: DR change from neighbor 0.0.0.0 to 2.2.2.2 on interface Loopback0
R2(config-if)#end
Mar 1 00:13:01.131: SSVS-S-COMP10_1: Configured from console by console
#
Building configuration...
[OK]

```

Figura 6. Paso 1 Configuración de Multicast Barranquilla. Elaboración propia.

```
R2#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, M - MSDP created entry,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - MDT-data group sender,
Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 00:06:23/00:02:21, RP 1.1.1.1, flags: SJCL
Incoming interface: Null, RPF nbr 0.0.0.0
Outgoing interface list:
FastEthernet0/0, Forward/Sparse, 00:06:23/00:02:21

R2#wr
Building configuration...
[OK]
```

Figura 7. Paso 2 Configuración de Multicast Barranquilla. Elaboración propia.

```
Senable
S#show ip int brief
Interface IP-Address OK? Method Status Protocol
FastEthernet0/0 192.168.24.2 YES NVRAM up up
FastEthernet0/1 192.168.25.2 YES NVRAM up up
Loopback0 3.3.3.3 YES NVRAM up up
S#config t
Enter configuration commands, one per line. End with CNTL/Z.
S(config)#ip multicast-routing
S(config)#int f0/0
S(config-if)#ip pim sparse-mode
S(config-if)#
S#
S# 1 00:14:02.011: SPDH-5-NBRCHG: neighbor 192.168.24.1 UP on interface FastEthernet0/0
S# 1 00:14:03.007: SPDH-5-DRCHE: DR change from neighbor 0.0.0.0 to 192.168.24.2 on interface FastEthernet0/0
S(config-if)#int f0/1
S(config-if)#ip pim sparse-mode
S(config-if)#
S# 1 00:14:21.027: SPDH-5-NBRCHG: neighbor 192.168.25.1 UP on interface FastEthernet0/1
S(config-if)#
S# 1 00:14:22.959: SPDH-5-DRCHE: DR change from neighbor 0.0.0.0 to 192.168.25.2 on interface FastEthernet0/1
S(config-if)#int lo0
S(config-if)#ip pim sparse-mode
S#
S# Invalid input detected at '^' marker.
S#
S# 1 00:15:23.907: SPDH-5-DRCHE: DR change from neighbor 0.0.0.0 to 3.3.3.3 on interface Loopback0
S#
S# 1 00:15:30.075: XSYS-5-COMP10_1: Configured from console by console
S#wr
Building configuration...
[OK]
```

Figura 8. Paso 1 Configuración de Multicast Bogotá. Elaboración propia.

```
R3#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, M - MSDP created entry,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - MDT-data group sender,
Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 00:03:56/00:02:11, RP 0.0.0.0, flags: DCL
Incoming interface: Null, RPF nbr 0.0.0.0
Outgoing interface list:
FastEthernet0/0, Forward/Sparse, 00:03:56/00:02:11
```

Figura 9. Paso 2 Configuración de Multicast Bogotá. Elaboración propia.

## Configuración de servicio IPTV entre las Sedes

De acuerdo a lo solicitado en la actividad para la implementación del Servicio IPTV entre las Sedes, el cual permitirá transferir contenidos multimedia se realiza la configuración de cada dispositivo y la validación de su funcionalidad con los comandos *show ip mroute* y *show mpls forwarding-table* y *ping* de la siguiente manera:

- Configuración básica de cada Router junto con sus Interfaces.
- Configuración en cada Router para que soporte el Multicast.
- Configuración en cada Router de LDP.
- Configuración entre el R1 y el R3 de MPLS BGP.

### Bogotá (Router R3)

```

Sin título: Bloc de notas
Archivo Edición Formato Ver Ayuda
ROUTER BOGOTA
Enable
Config t
hostname Bogota
int lo0
ip add 1.1.1.1 255.255.255.255
ip ospf 1 area 0
int fa0/0
ip add 10.0.0.1 255.255.255.0
no shut
ip ospf 1 area 0
ip multicast-routing
ip pim rp-address 10.0.0.1
interface fa 0/0
ip pim sparse-mode
exit
int lo0
ip pim sparse-mode
exit

router ospf 1
mpls ldp autoconfig

router bgp 1
neighbor 3.3.3.3 remote-as 1
neighbor 3.3.3.3 update-source Loopback0
no auto-summary
address-family vpnv4
neighbor 3.3.3.3 activate
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
ROUTER BOGOTA
  
```

Figura 10. Paso 1 Configuración de servicio IPTV Bogotá. Elaboración propia.

```

Bogota#
Bogota#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, W - MSDP created entry,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - MDT-data group sender,
Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.0.1.40), 00:09:03/00:02:26, RP 10.0.0.1, Flags: SPL
Incoming interface: Null, RPF nbr 0.0.0.0
Outgoing interface list: Null

Bogota#
Bogota#show mpls forwarding-table
Local Outgoing Prefix Bytes tag Outgoing Next Hop
tag tag or VC or Tunnel Id switched interface
16 Untagged 2.2.2.0/32 0 Fa0/0 10.0.0.2
17 Untagged 2.2.2.0/32 0 Fa0/0 10.0.0.2
18 Untagged 10.0.1.0/24 0 Fa0/0 10.0.0.2
  
```

Figura 11. Paso 2 Configuración de servicio IPTV Bogotá. Elaboración propia.

```
Bogota#show mpls forwarding-table
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id    switched   interface
16     Untagged  2.2.2.2/32     0          Fa0/0     10.0.0.2
17     Untagged  3.3.3.3/32     0          Fa0/0     10.0.0.2
18     Untagged  10.0.1.0/24    0          Fa0/0     10.0.0.2
Bogota#show mpls forwarding-table
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id    switched   interface
16     Untagged  2.2.2.2/32     0          Fa0/0     10.0.0.2
17     Untagged  3.3.3.3/32     0          Fa0/0     10.0.0.2
18     Untagged  10.0.1.0/24    0          Fa0/0     10.0.0.2
```

Figura 12. Paso 3 Configuración de servicio IPTV Bogotá. Elaboración propia.

```
Bogota#
Bogota#ping 2.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/11/16 ms
```

Figura 13. Paso 4 Configuración de servicio IPTV Bogotá. Elaboración propia.

```
Bogota#ping 3.3.3.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/40/44 ms
```

Figura 14. Paso 5 Configuración de servicio IPTV Bogotá. Elaboración propia.

## Barranquilla (Router R2)

```
Sin título: Bloc de notas
Archivo Edición Formato Ver Ayuda
exit
router ospf 1
mpls ldp autoconfig
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
ROUTER BARRANQUILLA
enable
config t
hostname Barranquilla
int 100
ip add 3.3.3.3 255.255.255.255
ip ospf 1 are 0
int 101
ip add 10.0.1.3 255.255.255.0
no shut
ip ospf 1 area 0
ip multicast-routing
ip pim rp-address 10.0.1.3
interface fa 0/0
ip pim sparse-mode
exit
int 100
ip pim sparse-mode
exit
router ospf 1
mpls ldp autoconfig
router bgp 1
neighbor 1.1.1.1 remote-as 1
neighbor 1.1.1.1 update-source Loopback0
no auto-summary
address-family vpnv4
neighbor 1.1.1.1 activate
```

Figura 15. Paso 1 Configuración de servicio IPTV Barranquilla. Elaboración propia.

```
Barranquilla#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, M - MSDP created entry,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - MDT-data group sender,
Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.0.1.40), 00:14:51/00:01:59, RP 10.0.1.3, flags: SACL
Incoming interface: Null, RPF nbr 0.0.0.0
Outgoing interface list:
FastEthernet0/0, Forward/Sparse, 00:14:51/00:01:59
```

Figura 16. Paso 2 Configuración de servicio IPTV Barranquilla. Elaboración propia.

```
Barranquilla#show mpls forwarding-table
Local Outgoing Prefix Bytes tag Outgoing Next Hop
tag tag or VC or Tunnel Id switched interface
16 Untagged 1.1.1.1/32 0 Fa0/0 10.0.1.2
17 Untagged 2.2.2.2/32 0 Fa0/0 10.0.1.2
18 Untagged 10.0.0.0/24 0 Fa0/0 10.0.1.2
```

Figura 17. Paso 3 Configuración de servicio IPTV Barranquilla. Elaboración propia.

```
Barranquilla#ping 1.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/40/48 ms
```

Figura 18. Paso 4 Configuración de servicio IPTV Barranquilla. Elaboración propia.

```
Barranquilla#ping 2.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/22/24 ms
```

Figura 19. Paso 5 Configuración de servicio IPTV Barranquilla. Elaboración propia.

## Medellín (Router R1)

```
Sin título: Bloc de notas
Archivo Edición Formato Ver Ayuda
no auto-summary
address-family ipv4
neighbor 3.3.3.3 activate
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
ROUTER MEDELLIN
enable
config t
hostname Medellin
int lo0
ip add 2.2.2.2 255.255.255.255
ip ospf 1 area 0
no shut
ip add 10.0.0.2 255.255.255.0
int fa0/0
ip ospf 1 area 0
ip add 10.0.1.2 255.255.255.0
no shut
ip ospf 1 area 0
ip multicast-routing
ip pim rp-address 10.0.0.2
interface fa 0/0
ip pim sparse-mode
exit
int lo0
ip pim sparse-mode
exit
router ospf 1
mpls ldp autoconfig
```

Figura 20. Paso 1 Configuración de servicio IPTV Medellín. Elaboración propia.

```
Medellin#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, W - MSDP created entry,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - MDT-data group sender,
Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.0.1.40), 00:11:02/00:02:47, RP 10.0.0.2, flags: S3CL
Incoming interface: Null, RPF nbr 0.0.0.0
Outgoing interface list:
FastEthernet0/0, Forward/Sparse, 00:11:02/00:02:47
```

Figura 21. Paso 2 Configuración de servicio IPTV Medellín. Elaboración propia.

```
Medellin#ping 3.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/22/24 ms
```

Figura 22. Paso 3 Configuración de servicio IPTV Medellín. Elaboración propia.

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/22/24 ms
```

Figura 23. Paso 4 Configuración de servicio IPTV Medellín. Elaboración propia.





## Medellín (Router R1)

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list 100 permit udp any any range 16384 32000
% Invalid input detected at '^' marker.

R1(config)#access-list 100 permit udp any any range 16384 32000
% Invalid input detected at '^' marker.

R1(config)#access-list 100 permit udp any any range 16384 32000
R1(config)#access-list 101 permit tcp any any eq 80
R1(config)#access-list 102 permit udp any any range 12000 12255
R1(config)#access-list 103 permit tcp any any range 80 1024
R1(config)#access-list 104 permit udp any any range 5060 8000
R1(config)#end
R1#
Mar 1 00:52:06.839: %SYS-5-CONFIG_I: Configured from console by console
R1#show access list
% Ambiguous command: "show access list"
R1#show access-list
Extended IP access list 100
 10 permit udp any any range 16384 32000
Extended IP access list 101
 10 permit tcp any any eq www
Extended IP access list 102
 10 permit udp any any range 12000 12255
Extended IP access list 103
 10 permit tcp any any range www 1024
Extended IP access list 104
 10 permit udp any any range 5060 8000
```

Figura 25. Paso 1 Configuración de QoS Medellín. Elaboración propia.

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#class-map match-all HTTP
R1(config-cmap)#match access-group 100
R1(config-cmap)#class-map match-all RTP
% Invalid input detected at '^' marker.

R1(config)#class-map match-all RTP
R1(config-cmap)#match access-group 101
R1(config-cmap)#class-map match-all VIDEOIN
R1(config-cmap)#match access-group 102
R1(config-cmap)#class-map match-all VIDEOOUT
R1(config-cmap)#match access-group 103
R1(config-cmap)#end
R1#
Mar 1 01:22:09.063: %SYS-5-CONFIG_I: Configured from console by console
R1#
Building configuration...
[OK]
R1#show class-map
Class Map match-all HTTP (id 1)
  Match access-group 100
Class Map match-any class-default (id 0)
  Match any
Class Map match-all VIDEOOUT (id 4)
  Match access-group 103
Class Map match-all RTP (id 2)
  Match access-group 101
Class Map match-all VIDEOIN (id 3)
  Match access-group 102
```

Figura 26. Paso 2 Configuración de QoS Medellín. Elaboración propia.

```
Match access-group 102
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#policy-map QoS
R1(config-pmap)#class HTTP
R1(config-pmap-c)#priority 200
R1(config-pmap-c)#class RTP
R1(config-pmap-c)#priority 200
R1(config-pmap-c)#class VIDEOIN
R1(config-pmap-c)#priority 200
R1(config-pmap-c)#class VIDEOOUT
R1(config-pmap-c)#priority 200
R1(config-pmap-c)#int fe/1
R1(config-if)#service-policy output QoS
R1(config-if)#end
R1#
Mar 1 01:32:48.331: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
R1#
Mar 1 01:32:49.219: %SYS-5-CONFIG_I: Configured from console by console
R1#
Building configuration...
[OK]
R1#show policy-map QoS
Policy Map QoS
Class HTTP
  Strict Priority
  Bandwidth 200 (kpbs) Burst 5000 (Bytes)
Class RTP
  Strict Priority
  Bandwidth 200 (kpbs) Burst 7500 (Bytes)
Class VIDEOIN
  Strict Priority
  Bandwidth 200 (kpbs) Burst 5000 (Bytes)
Class VIDEOOUT
  Strict Priority
  Bandwidth 200 (kpbs) Burst 5000 (Bytes)
```

Figura 27. Paso 3 Configuración de QoS Medellín. Elaboración propia.

## Barranquilla (Router R2)

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#access-list 100 permit udp any any range 16384 32000
R2(config)#access-list 101 permit tcp any any eq 80
R2(config)#access-list 102 permit udp any any range 12000 12255
R2(config)#access-list 103 permit tcp any any range 80 1024
R2(config)#access-list 104 permit udp any any range 5060 8000
R2(config)#end
R2#
*Mar 1 01:10:30.135: %SYS-5-CONFIG_I: Configured from console by console
R2#wr
Building configuration...
[OK]
R2#show access-list
Extended IP access list 100
 10 permit udp any any range 16384 32000
Extended IP access list 101
 10 permit tcp any any eq www
Extended IP access list 102
 10 permit udp any any range 12000 12255
Extended IP access list 103
 10 permit tcp any any range www 1024
Extended IP access list 104
 10 permit udp any any range 5060 8000
```

Figura 28. Paso 1 Configuración de QoS Barranquilla. Elaboración propia.

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#class-map match-all HTTP
R2(config-cmap)#match access-group 100
R2(config-cmap)#class-map match-all RTP
R2(config-cmap)#match access-group 101
R2(config-cmap)#class-map match-all VIDEOIN
R2(config-cmap)#match access-group 102
R2(config-cmap)#class-map match-all VIDEOOUT
R2(config-cmap)#match access-group 103
R2(config-cmap)#end
R2#
*Mar 1 01:24:07.187: %SYS-5-CONF16_I: Configured from console by console
R2#wr
Building configuration...
[OK]
R2#show class-map
Class Map match-all HTTP (id 1)
 Match access-group 100
Class Map match-any class-default (id 0)
 Match any
Class Map match-all VIDEOOUT (id 4)
 Match access-group 103
Class Map match-all RTP (id 2)
 Match access-group 101
Class Map match-all VIDEOIN (id 3)
 Match access-group 102
```

Figura 29. Paso 2 Configuración de QoS Barranquilla. Elaboración propia.

```
R2(config-pmap)#class HTTP
R2(config-pmap-c)#priority 200
R2(config-pmap-c)#class RTP
R2(config-pmap-c)#priority 300
R2(config-pmap-c)#class VIDEOIN
R2(config-pmap-c)#priority 200
R2(config-pmap-c)#class VIDEOOUT
R2(config-pmap-c)#priority 200
R2(config-pmap-c)#int fa/1
R2(config-if)#service-policy output QoS
R2(config-if)#end
R2#
*Mar 1 01:34:17.051: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
R2#
*Mar 1 01:34:18.135: %SYS-5-CONFIG_I: Configured from console by console
R2#wr
Building configuration...
[OK]
R2#show policy-map qos
% Invalid input detected at '^' marker.
R2#show policy-map qos
R2#show policy-map qos
Policy Map qos
Class HTTP
 Strict Priority
 Bandwidth 200 (kbps) Burst 5000 (Bytes)
Class RTP
 Strict Priority
 Bandwidth 300 (kbps) Burst 7500 (Bytes)
Class VIDEOIN
 Strict Priority
 Bandwidth 200 (kbps) Burst 5000 (Bytes)
Class VIDEOOUT
 Strict Priority
 Bandwidth 200 (kbps) Burst 5000 (Bytes)
```

Figura 30. Paso 3 Configuración de QoS Barranquilla. Elaboración propia.

## Bogotá (Router R3)

```
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#access-list 100 permit udp any any range 16384 32000
R3(config)#access-list 101 permit tcp any any eq 80
R3(config)#access-list 102 permit udp any any range 12000 12255
R3(config)#access-list 103 permit tcp any any range 80 1024
R3(config)#access-list 104 permit udp any any range 5060 8000
R3(config)#end
R3#
*Mar 1 01:12:08.447: %SYS-5-CONFIG_I: Configured from console by console
R3#wr
Building configuration...
[OK]
R3#show access-list
Extended IP access list 100
 10 permit udp any any range 16384 32000
Extended IP access list 101
 10 permit tcp any any eq www
Extended IP access list 102
 10 permit udp any any range 12000 12255
Extended IP access list 103
 10 permit tcp any any range www 1024
Extended IP access list 104
 10 permit udp any any range 5060 8000
```

Figura 31. Paso 1 Configuración de QoS Bogotá. Elaboración propia.

```
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#class-map match-all HTTP
R3(config-cmap)#match access-group 100
R3(config-cmap)#class-map match-all RTP
R3(config-cmap)#match access-group 101
R3(config-cmap)#class-map match-all VIDEOIN
R3(config-cmap)#match access-group 102
R3(config-cmap)#class-map match-all VIDEOOUT
R3(config-cmap)#match access-group 103
R3(config-cmap)#end
R3#
*Mar 1 01:25:35.899: %SYS-5-CONFIG_I: Configured from console by console
R3#wr
Building configuration...
[OK]
R3#show class-map
Class Map match-all HTTP (id 1)
 Match access-group 100

Class Map match-any class-default (id 0)
 Match any

Class Map match-all VIDEOOUT (id 4)
 Match access-group 103

Class Map match-all RTP (id 2)
 Match access-group 101

Class Map match-all VIDEOIN (id 3)
 Match access-group 102
```

Figura 32. Paso 2 Configuración de QoS Bogotá. Elaboración propia.

```
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#policy-map QoS
R3(config-pmap-c)#class HTTP
R3(config-pmap-c)#priority 200
R3(config-pmap-c)#class RTP
R3(config-pmap-c)#priority 200
R3(config-pmap-c)#class VIDEOIN
R3(config-pmap-c)#priority 200
R3(config-pmap-c)#class VIDEOOUT
R3(config-pmap-c)#priority 200
R3(config-pmap-c)#int fa/1
R3(config-if)#service-policy output QoS
R3(config-if)#end
R3#
*Mar 1 01:35:54.339: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
R3#
*Mar 1 01:35:54.971: %SYS-5-CONFIG_I: Configured from console by console
R3#wr
Building configuration...
[OK]
R3#show policy-map QoS
Policy Map QoS
 Class HTTP
  Strict Priority
  Bandwidth 200 (kbps) Burst 5000 (Bytes)
 Class RTP
  Strict Priority
  Bandwidth 300 (kbps) Burst 7500 (Bytes)
 Class VIDEOIN
  Strict Priority
  Bandwidth 200 (kbps) Burst 5000 (Bytes)
 Class VIDEOOUT
  Strict Priority
  Bandwidth 200 (kbps) Burst 5000 (Bytes)
```

Figura 33. Paso 3 Configuración de QoS Bogotá. Elaboración propia.



## Conclusiones


Se interpreta satisfactoriamente que MPLS tiene como una de sus principales ventajas el soporte de QoS, lo que permite diseñar y configurar un esquema óptimo que entregue mejores prestaciones y alcance altos grados de confiabilidad con los clientes que solicitan servicios a los ISPs.

Se destaca como esta tecnología combina la simplicidad y rapidez de la conmutación de Capa 2 y las funciones de control del enrutamiento de Capa 3 en una sola entidad, ofreciendo capacidades significativas en las áreas de Redes Privadas Virtuales (VPNs), Ingeniería de Tráfico y Calidad de Servicio (QoS), entre otras.

El ofrecimiento de QoS al tráfico para la administración de la red disminuye considerablemente el tiempo de respuesta ante fallas en los equipos, ya que las soluciones se transmiten de forma rápida y eficiente con mejores niveles de servicio que permitan obtener los resultados previstos.

El establecimiento de un esquema de QoS en la red con tecnología IP/MPLS de una red específica contribuye a que todos los servicios y aplicaciones que circulan, en especial las de tiempo real, obtengan una correcta asignación de recursos de acuerdo a sus requerimientos y sean servidas incluso durante periodos de congestión.

Se implementan configuraciones específicas sobre equipos de comunicación como Router, Switch, Teléfonos y PC's logrando entender un funcionamiento óptimo y erróneo no solo de cada dispositivo y de cada conexión sino de toda la red tanto local para cada ciudad como a nivel general.



## Bibliografía

Avellaneda, J. V., Rodríguez, J. R., y López, D. A. (2014). “Servicios de Televisión sobre la Plataforma de Internet (IPTV-IMS) usando Protocolo de Flujo en Tiempo Real (RTSP) y Protocolo de Transferencia de Hipertexto (HTTP)”. *Información Tecnológica*. Recuperado de <https://doi-org.bibliotecavirtual.unad.edu.co/10.4067/S0718-07642014000100008>

Barba Martí, y Pallejà Muñoz. (2013). Calidad de servicio (QoS) basándonos en redes de nueva generación. Recuperado de <http://bibliotecavirtual.unad.edu.co/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=edsbas&AN=edsbas.1B22222E&lang=es&site=eds-live&scope=site>

García, A. y Cuellar, J. (2012). Calidad de Servicio en Proveedores de Servicios IPTV. Revista: *Ingenium Ciencia y Tecnología*. Recuperado desde: [https://www.researchgate.net/publication/270589568\\_Calidad\\_de\\_Servicio\\_en\\_Proveedores\\_de\\_Servicios\\_IPTV](https://www.researchgate.net/publication/270589568_Calidad_de_Servicio_en_Proveedores_de_Servicios_IPTV)


Evans, J., y Filsfils, C. (2007). Deploying IP and MPLS QoS for Multiservice Networks : *Theory and Practice*. Recuperado de <http://bibliotecavirtual.unad.edu.co/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=196159&lang=es&site=eds-live&scope=site>

Jon Goñi Amatriain. (2019). IPTV. *Protocolos empleados y QoS*. [https://www.tlm.unavarra.es/~daniel/docencia/rba/rba06\\_07/trabajos/resumenes/gr16-QoSEnIPTV.pdf](https://www.tlm.unavarra.es/~daniel/docencia/rba/rba06_07/trabajos/resumenes/gr16-QoSEnIPTV.pdf)

Kurt rainer rottmann chávez. (2010). Diseño e implementación de un laboratorio de IPTV, medición y gestión. [http://repositorio.uchile.cl/bitstream/handle/2250/103933/cfrottman\\_k.pdf?sequence=3&isAllowed=y](http://repositorio.uchile.cl/bitstream/handle/2250/103933/cfrottman_k.pdf?sequence=3&isAllowed=y)

Molina, R. F. J., y Polo, O. E. (2014). Servicios en red. Recuperado de <http://bibliotecavirtual.unad.edu.co:2077/lib/unadsp/reader.action?ppg=573&docID=11046839&tm=1488704523542>

Oliveira Guerra, S. de. (2004). Una propuesta de arquitectura MPLS/DiffServ para proveer mecanismos de calidad de servicio (QOS) en el transporte de la telefonía IP. Recuperado de <http://bibliotecavirtual.unad.edu.co/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=edsdnp&AN=edsdnp.2701TES&lang=es&site=eds-live&scope=site>



Pablo Arango, B. j., Alberto Portilla, A. l., y Carlos Cuéllar, Q. j. (2013). Procedimiento para implementar QoS en la capa de acceso en redes de próxima generación enfocado en el servicio de voz. Recuperado de <http://bibliotecavirtual.unad.edu.co:2048/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=aci&AN=99884800&lang=es&site=eds-live>

Pablo Arango., Alberto Portilla., y Carlos Cuéllar. (2013). Procedimiento para implementar QoS en la capa de acceso en redes de próxima generación enfocado en el servicio de voz. Recuperado de <http://bibliotecavirtual.unad.edu.co/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=aci&AN=99884800&lang=es&site=eds-live&scope=site>

