

Configuración del servicio de TVIP sobre el emulador GNS3

Jose Luis Gonzalez Medina

Ing. Omar Albeiro Trejo

Director de Curso

Universidad Nacional Abierta y a Distancia – UNAD

Escuela de Ciencias Básicas Tecnología e Ingeniería ECBTI

Tecnología En Automatización Electrónica

Diciembre 2021

Resumen

El objetivo de este informe es registrar los pasos esenciales para establecer los mecanismos de la configuración de la calidad de Servicio (QoS), utilizando diagramas de bloques y flujos para describirlos de manera concisa. Además, se detallan los procedimientos necesarios para asignar e instalar un plan QoS, que incluye porcentajes relativos al ancho de banda y los flujos de datos en la red. En el contexto de la configuración del servicio de Televisión por Protocolo de Internet (IPTV), se consideran todos los componentes y aspectos relevantes para su implementación exitosa. Esto abarca la asignación de ancho de banda específica a cada interfaz del enrutador.

Luego, se procede a la ejecución de una red de Nueva Generación (NGN) simulada, utilizando máquinas virtuales. Para la transmisión de una película desde un dispositivo fuente a un dispositivo receptor, se emplea el reproductor multimedia VLC.

Palabras Claves: NGN, IPTV, GNS3, QoS

Abstract

The objective of this report is to record the essential steps to establish Quality of Service (QoS) mechanisms, using block diagrams and flows to describe them concisely. In addition, the procedures necessary to assign and install a QoS plan are detailed, which includes percentages related to bandwidth and data flows in the network. In the context of configuring the Internet Protocol Television (IPTV) service, all relevant components and aspects for its successful implementation are considered. This encompasses the specific bandwidth allocation to each router interface. Then, we proceed to the execution of a simulated New Generation Network (NGN), using virtual machines. For streaming a movie from a source device to a receiving device, the VLC media player is used.

Keywords: NGN, IPTV, GNS3, QoS

Tabla de Contenido

Introducción	9
Desarrollo.....	10
Configuración de la calidad de servicio (QoS).....	10
Configuración del servicio de IPTV	15
Clasificación – Markin Packet.....	19
Configuración de habilitación de servicio CEF	21
Configuración de habilitación de servicio MPLS	22
Configuración de los puertos de interfaz	22
Configuración del servicio Multicast.....	25
Configuración de entradas e identificación topológica.....	26
Configuraciones estáticas para la habilitación del servicio Multicast	27
Configuraciones de las redes globales	28
Configuraciones de DHCP para el establecimiento dinámico de conexiones TCP-IP....	29
Configuración de los tiempos para el enrutamiento.....	29
Configuraciones de grupos Multicast	30
Configuración de los grupos de enrutamiento de cada ciudad.....	31
Configuración de cada uno de los grupos de acceso para cada ciudad.....	32
Creación de enrutamientos tabulares	33
Establecimiento de conexiones entre los router vecinos.....	34

Simulación	37
VirtualBox – GNS3.....	37
Configuración para llevar a cabo las transmisiones VLC.....	40
Conclusiones	43
Bibliografía	45

Tabla de Figuras

Figura 1 <i>Diagrama de bloques weighted fair</i>	10
Figura 2 <i>Diagrama de bloques priority</i>	11
Figura 3 <i>Configuraciones QoS implementadas en router Medellín</i>	13
Figura 4 <i>Configuraciones QoS implementadas en router Bogotá</i>	13
Figura 5 <i>Configuraciones QoS implementadas en router barranquilla</i>	14
Figura 6 <i>Red implementada</i>	15
Figura 7 <i>Simulación topológica del modelo de conexiones con routers</i>	17
Figura 8 <i>Modelo de conexiones con routers</i>	18
Figura 9 <i>Modelo de conexiones con routers sedes</i>	19
Figura 10 <i>Pasos para configurar</i>	20
Figura 11 <i>Habilitación del servicio CEF</i>	21
Figura 12 <i>Habilitación servicio MPLS</i>	22
Figura 13 <i>Puertos de interfaces</i>	23
Figura 14 <i>Redes BGP</i>	24
Figura 15 <i>Servicios multicast</i>	25
Figura 16 <i>Configuración de entrada</i>	26
Figura 17 <i>Habilitación servicio multicast</i>	27
Figura 18 <i>Redes globales</i>	28
Figura 19 <i>Configuración DHCP</i>	29
Figura 20 <i>Tiempos de enrutamiento</i>	29
Figura 21 <i>Grupos multicast</i>	30
Figura 22 <i>Enrutamiento en cada ciudad</i>	31

Figura 23 <i>Acceso a cada ciudad</i>	32
Figura 24 <i>Segunda topología</i>	32
Figura 25 <i>Tabuladores</i>	33
Figura 26 <i>Router vecinos Bogotá</i>	35
Figura 27 <i>Router vecinos</i>	35
Figura 28 <i>Router vecinos R1</i>	36
Figura 29 <i>Router vecinos R2</i>	36
Figura 30 <i>Virtual box</i>	37
Figura 31 <i>GNS3-virtual box</i>	37
Figura 32 <i>Topología y servicio IPTV</i>	38
Figura 33 <i>Máquinas de virtual box</i>	38
Figura 34 <i>Descarga y búsqueda del video</i>	39
Figura 35 <i>Video en la sede principal</i>	39
Figura 36 <i>Configuración para empezar a transmitir</i>	40
Figura 37 <i>Configuración VCL emisor</i>	40
Figura 38 <i>Configuración VCL receptor</i>	41
Figura 39 <i>Configuración sede barranquilla</i>	41
Figura 40 <i>URL de la sede de Bogotá en la sede barranquilla</i>	42
Figura 41 <i>Transmisión desde la sede Bogotá hasta la sede barranquilla</i>	42

Lista de Tablas

Tabla 1 <i>Código de comando para configuración QoS de terminales</i>	12
Tabla 2 <i>Direccionamiento IP</i>	16

Introducción

El propósito de este proyecto es examinar el proceso de transferencia de información entre tres sedes situadas en distintas ciudades dentro del país. El proyecto implica la implementación de un emulador y un simulador. En el emulador, se incorporarán los códigos necesarios para cada componente, asegurando así la distribución adecuada de los datos. Por otro lado, en el simulador, se instalarán dos máquinas virtuales que ejecutarán dos sistemas operativos. En este caso, optó por utilizar Windows XP como ejemplo. El meta principal es enviar un vídeo desde una sede hasta otra, empleando la aplicación VLC y configurándola para funcionar en modo de transmisión continua.

Somos conscientes de la relevancia de profundizar en este estudio y de poner en práctica la creación de nuevas redes, ya que estas desempeñan un papel fundamental en nuestra rutina diaria.

Desarrollo

Configuración de la Calidad de Servicio (QoS)

QoS se refiere a la capacidad de una red para ofrecer un mejor servicio a un tráfico de red seleccionado a través de varias tecnologías subyacentes, entre ellas, Frame Relay, modo de transferencia asíncrona (ATM, asynchronous transfer mode), Ethernet y redes 802.1X, SONET y redes con routing IP.

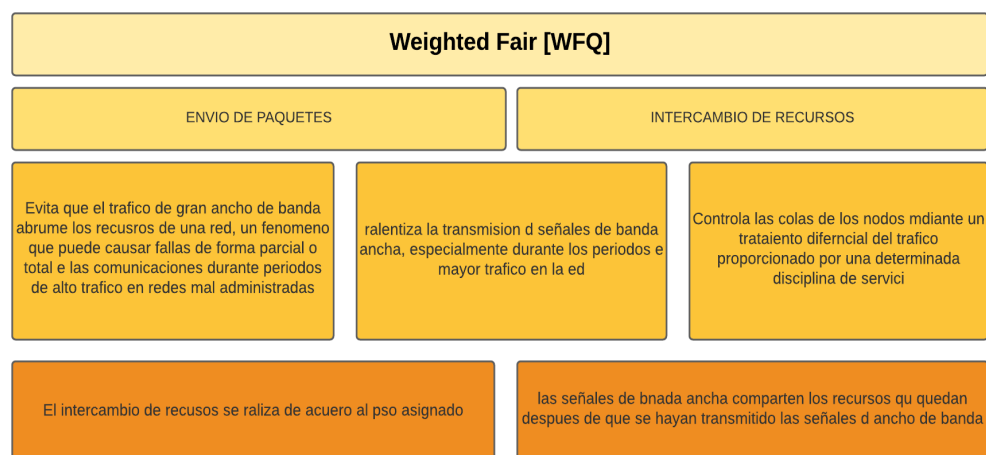
QoS es una recolección de tecnologías que permite que las aplicaciones pidan y que reciban los niveles del servicio predecible en términos de capacidad de producción de datos (ancho de banda), variaciones de latencia (inquietud), y retraso. (Cisco, 2009)

Seleccionar dos mecanismos de QoS y describir el proceso que realiza cada uno mediante un diagrama de bloques.

Mecanismos de QoS: Weighted Fair [WFQ]

Figura 1

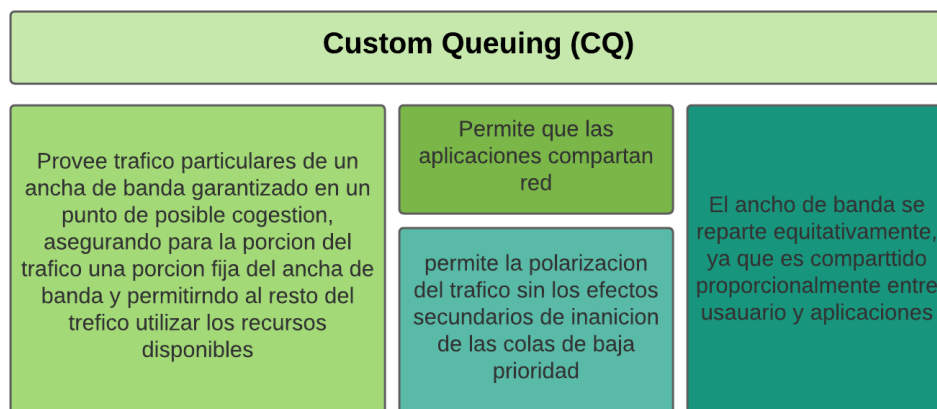
Diagrama de bloques weighted fair



Fuente: Autoría propia

Figura 2

Diagrama de bloques Priority



Fuente: Autoría propia

Documente los pasos requeridos para definir un plan de QoS que incluya los siguientes porcentajes sobre el ancho de banda total (separar tráficos mediante definición de clases):

10% del ancho de banda total para tráfico web

15% para tráfico de voz

20% para tráfico de streaming de video.

Los pasos requeridos para la implementación de un plan QoS se debe realizar los siguientes pasos:

Definición de las Listas con los Protocolos y Porcentaje de Ancho de Banda para cada dispositivo.

Configuración de Clases (asignación de nombres y listas) y verificación de las Clases en los Router.

Configuración de Políticas QoS en cada Router y verificación de las Políticas implementadas en cada dispositivo.

Tabla 1*Códigos de comando para configuración QoS de terminales*

Comando	Descripción
Enable	Entrar
configure terminal	configuración de la terminal
access-list 100 permit tcp any any eq 443	Listas de acceso para navegación, tráfico de voz y de streaming de video
access-list 101 permit udp any any eq 5060	
access-list 102 permit udp any any eq 5004	
no cdp log mismatch duplex	Nombre de la clase
class-map match-all Name	Asignación de clases a lista de accesos
match access-group Numero	
policy-map Ngn-Grupo1	Definición de políticas
class web	
bandwidth percent 10	Asignación de porcentaje de banda ancha
int f0/0	Asignación de política QoS en las interfaces de salida
service-policy output Ngn-Grupo1	
show policy-map	Muestra las políticas implementadas

Figura 3

Configuraciones QoS implementadas en Router Medellín

```

Medellin#enable
Medellin#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Medellin(config)#access-
Medellin(config)#access-
Medellin(config)#access-list 100 per
Medellin(config)#access-list 100 permit tcp any any eq 443
Medellin(config)#access-list 101 permit udp any any eq 5060
Medellin(config)#access-list 102 permit udp any any eq 5004
Medellin(config)#no cdp log mismatch duplex
Medellin(config-cmap)#match access-group 100
Medellin(config-cmap)#class-map match-all VOIP
Medellin(config-cmap)#match access-group 101
Medellin(config-cmap)#class-map match-all IPTV
Medellin(config-cmap)#match access-group 102
Medellin(config-cmap)#end
Medellin#
*Mar 1 00:04:07.579: %SYS-5-CONFIG_I: Configured from console by console
Medellin#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Medellin(config)#policy-map NGN-GRUPO1
Medellin(config-pmap)#class WEB
Medellin(config-pmap-c)#bandwidth percent 10
Medellin(config-pmap-c)#class VOIP
Medellin(config-pmap-c)#bandwidth percent 15
Medellin(config-pmap-c)#class IPTV
Medellin(config-pmap-c)#bandwidth percent 20
Medellin(config-pmap-c)#end
Medellin#
*Mar 1 00:05:30.647: %SYS-5-CONFIG_I: Configured from console by console
Medellin#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Medellin(config)#int f0/0
Medellin(config-if)#service-policy output NGN-GRUPO1
Medellin(config-if)#
*Mar 1 00:06:09.691: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
Medellin(config-if)#end
Medellin#co
*Mar 1 00:06:16.935: %SYS-5-CONFIG_I: Configured from console by console
Medellin#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Medellin#show pol
Medellin#show policy-map
Policy Map NGN-GRUPO1
  Class WEB

```

Fuente: Autoría propia

Figura 4

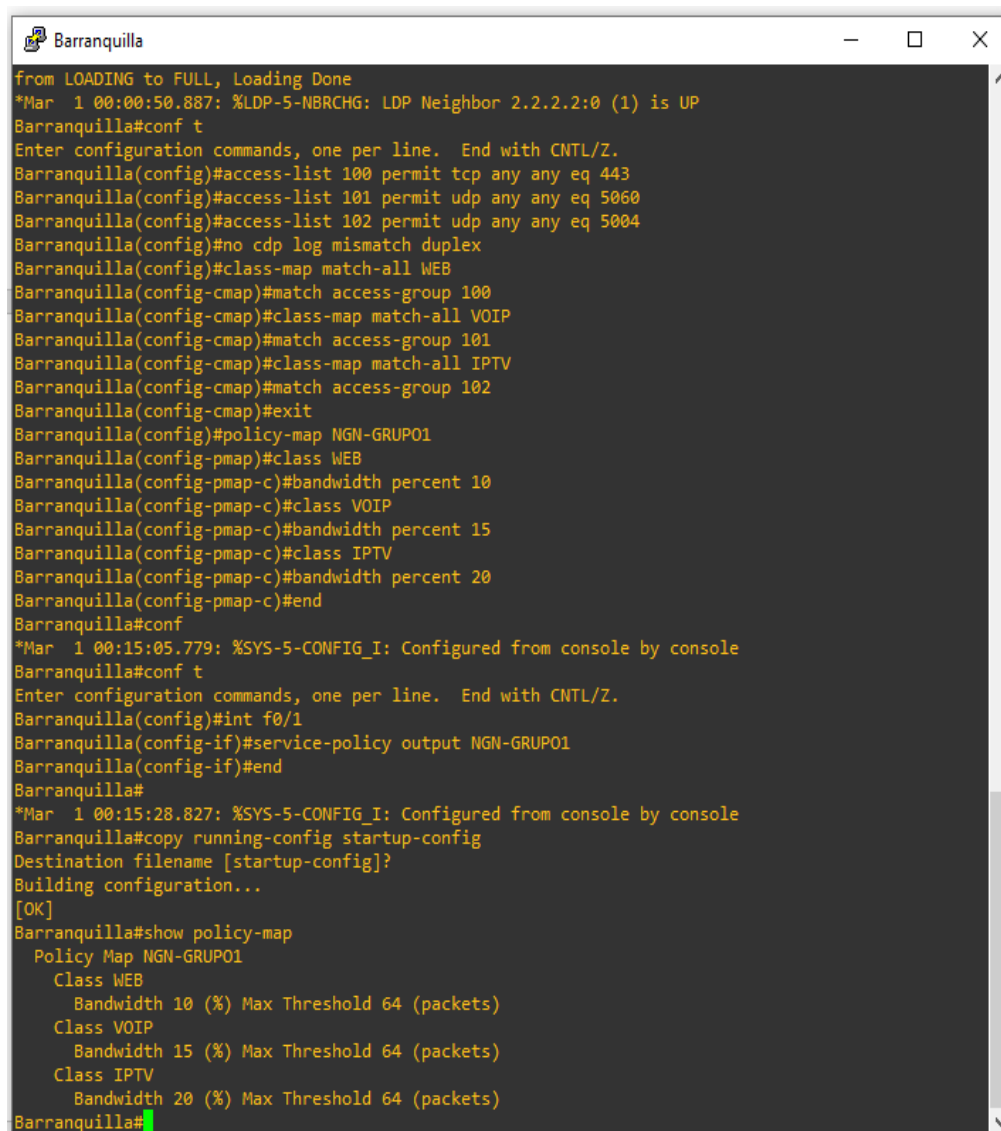
Configuraciones QoS implementadas en Router Bogotá

```

Bogota#
*Mar 1 00:08:50.903: %LDP-S-NBRCHG: LDP Neighbor 3.3.3.3:0 (2) is up
Bogota#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Bogota(config)#access-
Bogota(config)#access-
Bogota(config)#access-list 100 permit tcp any any eq 443
Bogota(config)#access-list 101 permit udp any any eq 5060
Bogota(config)#access-list 102 permit udp any any eq 5004
Bogota(config)#no cdp log mismatch duplex
Bogota(config-cmap)#match access-group 100
Bogota(config-cmap)#class-map match-all VOIP
Bogota(config-cmap)#match access-group 101
Bogota(config-cmap)#class-map match-all IPTV
Bogota(config-cmap)#match access-group 102
Bogota(config-cmap)#exit
Bogota(config)#policy-map NGN-GRUPO1
Bogota(config-pmap)#class WEB
Bogota(config-pmap-c)#bandwidth percent 10
Bogota(config-pmap-c)#class VOIP
Bogota(config-pmap-c)#bandwidth percent 15
Bogota(config-pmap-c)#class IPTV
Bogota(config-pmap-c)#bandwidth percent 20
Bogota(config-pmap-c)#exit
Bogota(config-pmap)#exit
Bogota(config)#int f0/0-1
Bogota(config)#
% Invalid input detected at '^' marker.
Bogota(config)#int f0/0
Bogota(config-if)#service-policy output NGN-GRUPO1
Bogota(config-if)#service-policy output NGN-GRUPO1
*Mar 1 00:09:44.715: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
Bogota(config-if)#int f0/1
Bogota(config-if)#service-policy output NGN-GRUPO1
Bogota(config-if)#
*Mar 1 00:09:56.543: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
Bogota(config-if)#exit
Bogota(config)#exit
Bogota#
*Mar 1 00:10:05.615: %SYS-5-CONFIG_I: Configured from console by console
Bogota#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Bogota#show policy-map
Policy Map NGN-GRUPO1
  Class WEB
    Bandwidth 10 (%) Max Threshold 64 (packets)
  Class VOIP
    Bandwidth 15 (%) Max Threshold 64 (packets)
  Class IPTV
    Bandwidth 20 (%) Max Threshold 64 (packets)
Bogota#

```

Fuente: Autoría propia

Figura 5*Configuraciones QoS implementadas en Router Barranquilla*

```
Barranquilla
from LOADING to FULL, Loading Done
*Mar 1 00:00:50.887: %LDP-5-NBRCHG: LDP Neighbor 2.2.2.2:0 (1) is UP
Barranquilla#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Barranquilla(config)#access-list 100 permit tcp any any eq 443
Barranquilla(config)#access-list 101 permit udp any any eq 5060
Barranquilla(config)#access-list 102 permit udp any any eq 5004
Barranquilla(config)#no cdp log mismatch duplex
Barranquilla(config)#class-map match-all WEB
Barranquilla(config-cmap)#match access-group 100
Barranquilla(config-cmap)#class-map match-all VOIP
Barranquilla(config-cmap)#match access-group 101
Barranquilla(config-cmap)#class-map match-all IPTV
Barranquilla(config-cmap)#match access-group 102
Barranquilla(config-cmap)#exit
Barranquilla(config)#policy-map NGN-GRUPO1
Barranquilla(config-pmap)#class WEB
Barranquilla(config-pmap-c)#bandwidth percent 10
Barranquilla(config-pmap-c)#class VOIP
Barranquilla(config-pmap-c)#bandwidth percent 15
Barranquilla(config-pmap-c)#class IPTV
Barranquilla(config-pmap-c)#bandwidth percent 20
Barranquilla(config-pmap-c)#end
Barranquilla#conf
*Mar 1 00:15:05.779: %SYS-5-CONFIG_I: Configured from console by console
Barranquilla#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Barranquilla(config)#int f0/1
Barranquilla(config-if)#service-policy output NGN-GRUPO1
Barranquilla(config-if)#end
Barranquilla#
*Mar 1 00:15:28.827: %SYS-5-CONFIG_I: Configured from console by console
Barranquilla#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Barranquilla#show policy-map
  Policy Map NGN-GRUPO1
    Class WEB
      Bandwidth 10 (%) Max Threshold 64 (packets)
    Class VOIP
      Bandwidth 15 (%) Max Threshold 64 (packets)
    Class IPTV
      Bandwidth 20 (%) Max Threshold 64 (packets)
Barranquilla#
```

Fuente: Autoría propia

Configuración del Servicio de IPTV

Mediante el emulador GNS3 y el uso de máquinas virtuales, a partir del análisis del servicio de IPTV desarrollado en la Fase 10, implemente IPTV Multicast entre las sedes del escenario de red descrito en la Fase 1, el cual permitirá transferir contenidos multimedia entre dos sedes. Documente los pasos en un informe:

Configuración de dispositivos y servidor TVIP

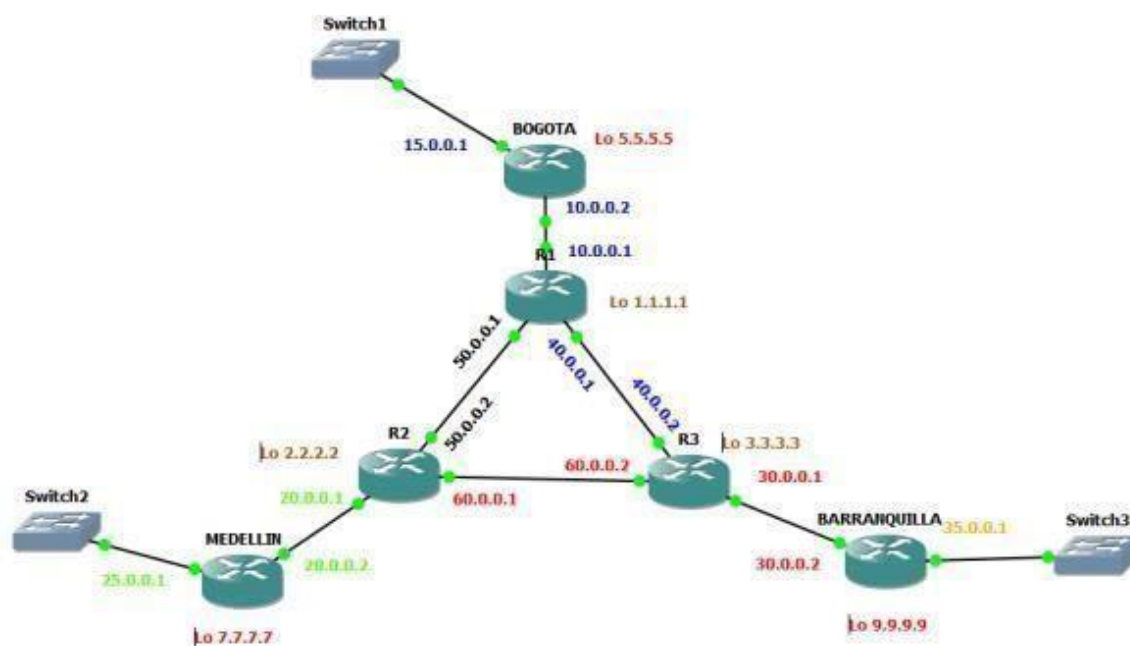
Habilitación de protocolos de enrutamiento, Multicast, RTP, etc.

Configuración de cliente de video VLC

Pruebas funcionales

Figura 6

Red Implementada



Fuente: Autoría propia

Para llevar a cabo la configuración del servicio de IPTV bajo los escenarios propuestos, es importante el reconocimiento de protocolos y direccionamientos IP asociados a cada una de las sedes en cuestión que, para este caso corresponde a Bogotá, Medellín y Barranquilla. Dentro de este orden de ideas, se hace necesaria la parametrización y estructuración de elementos como WAN, Router, PC1 (escritorio primario), PC2 (escritorio secundario) y para cada uno de estos se hace uso de una máscara de redireccionamiento como se muestra en la *tabla 2*.

Tabla 2

Direccionamiento IP

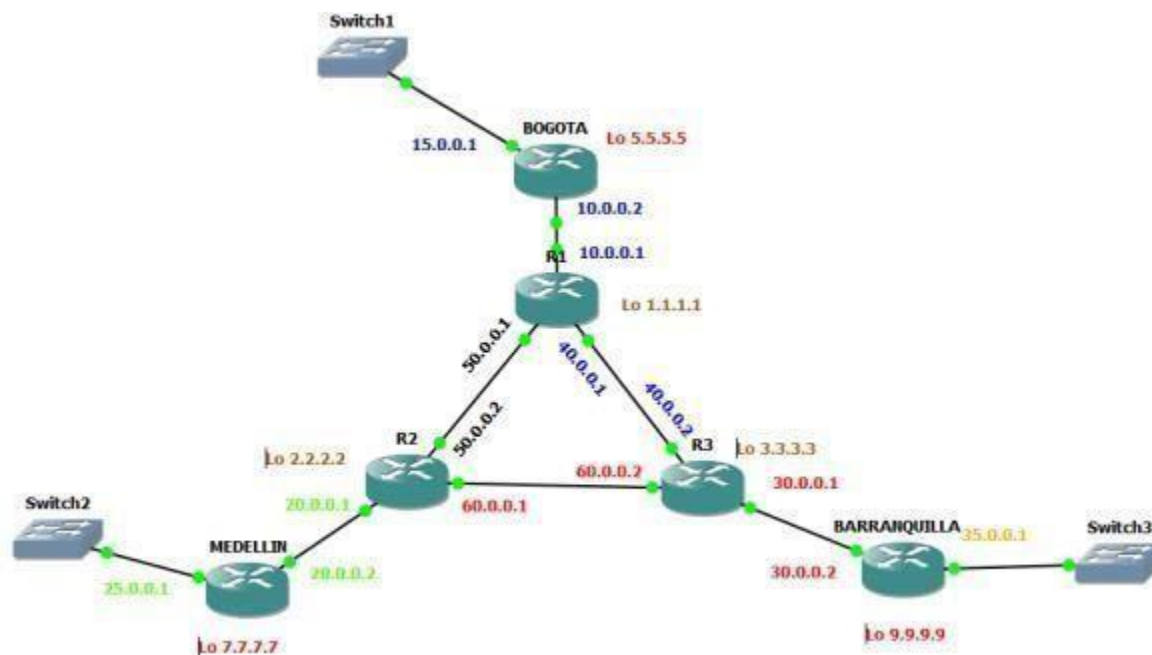
	Bogotá	Medellín	Barranquilla
WAN	10.0.1.1	10.0.2.2	10.0.3.3
Mascara	255.0.0.0	255.0.0.0	255.0.0.0
Router	201.168.1.1	201.168.2.1	201.168.3.1
Mascara	255.255.255.0	255.255.255.0	255.255.255.0
PC1	201.168.1.2	201.168.2.2	201.168.3.2
Mascara	255.255.255.0	255.255.255.0	255.255.255.0
PC2	201.168.1.3	201.168.2.3	201.168.3.3
Mascara	255.255.255.0	255.255.255.0	255.255.255.0

Posteriormente, es necesario la construcción o simulación topológica del modelo de conexión de redes, en donde sea posible identificar cada uno de los elementos esenciales de este. En consecuencia, cada uno de los routers presentes en el diagrama deben ser configuraciones inicialmente bajo parametrizaciones de *Interface loopback 0*, *Interface f0[0,1]/[0,1]* con el objetivo de obtener resultados coherentes en el proceso de conexión.

A continuación, se muestran las configuraciones iniciales para cada uno de estos routers, haciendo conexiones con redes WAN.

Figura 7

Simulación topológica del modelo de conexiones con Routers.



Fuente: Autoría propia

A partir de la figura mostrada anteriormente es posible evidenciarse la forma en que cada uno de los Router R1, R2 y R3 están interconectados con las diferentes bases de transmisión ubicadas en Barranquilla, Bogotá y Medellín. Teniendo este panorama claro, a continuación, se muestran las configuraciones iniciales para los Router en cuestión.

Figura 8

modelo de conexiones con Routers.

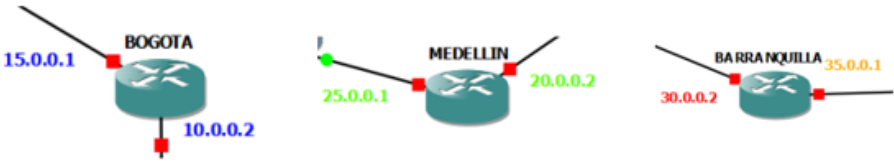
Interface loopback 0	1.1.1.1 255.255.255.255	2.2.2.2 255.255.255.255	3.3.3.3 255.255.255.255
Interface f0/0	50.0.0.1 255.255.255.252	50.0.0.2 255.255.255.250	40.0.0.2 255.255.255.252
Interface f0/1	40.0.0.1 255.255.255.252	60.0.0.1 255.255.255.252	60.0.0.2 255.255.255.252
Interface f1/0	10.0.0.1 255.255.255.252	20.0.0.1 255.255.255.252	30.0.0.1 255.255.255.252

Fuente: Autoría propia

Por otra parte, es importante realizar la configuración inicial de las bases de transmisión que intervienen en el modelo, por ello, en la tabla mostrada a continuación se evidencia cada una de estas configuraciones de manera respectiva.

Figura 9

modelo de conexiones con routers sedes.



Interface loopback 0	201.168.1.1	201.168.1.1	201.168.3.1
Interface f0/0	5.5.5.5 255.255.255.255	7.7.7.7 255.255.255.255	9.9.9.9 255.255.255.255
Interface f0/1	10.0.0.2 255.255.255.252	20.0.0.2 255.255.255.252	30.0.0.2 255.255.255.252
Interface f1/0	15.0.0.1 255.255.255.252	25.0.0.1 255.255.255.252	35.0.0.1 255.255.255.252

Fuente: Autoría propia

Clasificación – Markin Packet

Esto nos permite una manera sencilla de realizar conexiones de marcado mediante la implementación de conexiones CLI para optimizar las parametrizaciones de marcado, es decir, esta clasificación permite asignar una determinada conexión de transmisión o recepción que el usuario define previamente mediante la implementación de comando MQC. Estas conexiones realizadas pueden ser activadas mediante las configuraciones de políticas asignadas para una clase determinada. Dentro de este orden de ideas, esta metodología permite a los diferentes usuarios llevar a cabo la ejecución de los siguientes comandos:

Marcado de diferentes paquetes mediante la consideración de bit asociados a la IP de procedencia o en su defecto con conexiones ToS asociadas a la cabecera del DSCP.

Marcado de diferentes paquetes mediante la asignación de multicapas CoS.

Asociación de valores por agrupaciones locales para proporcionar calidad a los servicios.

Asignación de bits para la propagación de prioridades en las conexiones CLC

relacionadas a las cabeceras ATM clasificadas de manera binaria.

Proporcional un FRDE en los diferentes campos de direcciones relay binarios.

Por otra parte, este proceso permite llevar a cabo diferentes configuraciones de interfaces y de circuitos categorizados virtualmente por conexiones ATM con PVC. No obstante, este servicio no proporciona configuraciones para interfaces como Tunner, ATMS-SVC, DLCI, CEF, entre otros. En consideración con esto, es necesario definir cada uno de los pasos que permite la configuración de este servicio, los cuales se presentan a continuación:

Figura 10

Pasos para configurar.

<i>Pasos para configuración</i>		
	Comando	Propósito
Paso 1	Router(config)# policy-map policy-map	Especifica el nombre de la política que va a ser creada.
Paso 2	Router(config-pmap)# class class-map-name	Especifica el nombre de la clase sobre la que se aplica a la política.
Paso 3	Router(config-pmap-c)# set [depende de cómo lo queramos marcar]	Activa la herramienta Class-Based Packet Marking para que se marquen los paquetes de la clase. El argumento depende de lo que queramos marcar.
Paso 4	Router(config)# interface nombre-de-la-interfaz	Especifica el nombre de la interfaz a la que se asociará la política.
Paso 5	Router(config-if)# service-policy input output policy-map	Asignamos la política creada a la interfaz

Fuente: Autoría propia

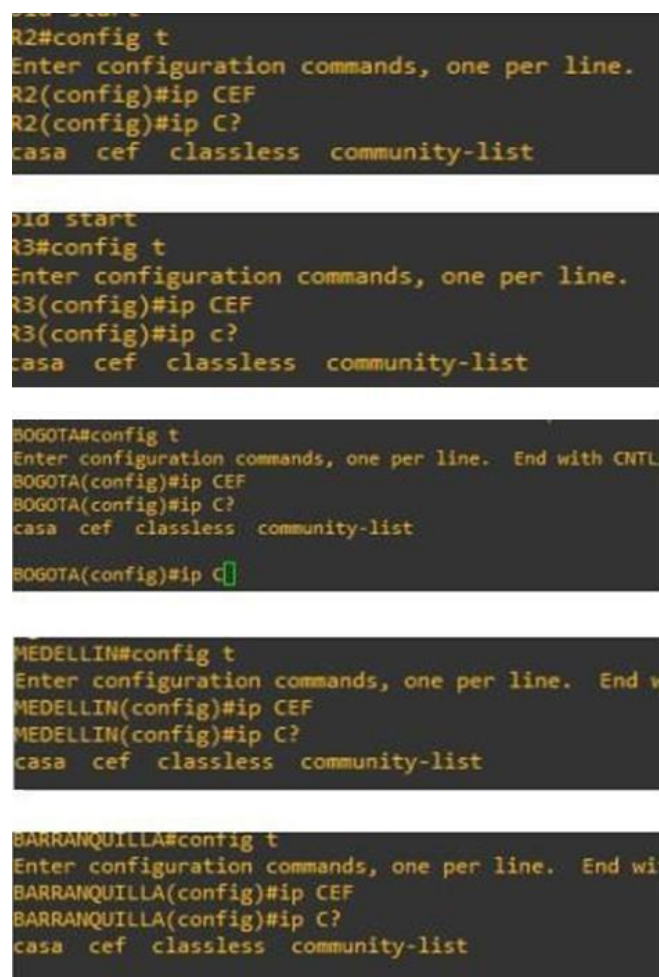
Cuando ya se tienen definidos cada uno de los pasos necesarios para esta configuración, es necesario iniciar con el marcado de los diferentes paquetes IP de precedencia.

Configuración de Habilitación de Servicio CEF

A continuación, se muestra el proceso para llevar a cabo la configuración del servicio CEF. Para ello es necesario ingresar a la consola que corresponde a R1, R2 y R3 para asignar las configuraciones iniciales de este. Así mismo, se deben realizar las configuraciones de los transmisores ubicados en Medellín, Bogotá y Barranquilla.

Figura 11

Habilitación del servicio CEF.



```
R2#config t
Enter configuration commands, one per line.
R2(config)#ip CEF
R2(config)#ip C?
casa cef classless community-list

did start
R3#config t
Enter configuration commands, one per line.
R3(config)#ip CEF
R3(config)#ip c?
casa cef classless community-list

BOGOTA#config t
Enter configuration commands, one per line. End with CNTL
BOGOTA(config)#ip CEF
BOGOTA(config)#ip C?
casa cef classless community-list
BOGOTA(config)#ip C

MEDELLIN#config t
Enter configuration commands, one per line. End w
MEDELLIN(config)#ip CEF
MEDELLIN(config)#ip C?
casa cef classless community-list

BARRANQUILLA#config t
Enter configuration commands, one per line. End wi
BARRANQUILLA(config)#ip CEF
BARRANQUILLA(config)#ip C?
casa cef classless community-list
```

Fuente: Autoría propia

Configuración de Habilitación de Servicio MPLS

Se debe ingresar al espacio de consola de cada uno de los routers para ingresar las configuraciones generales para la habilitación del servicio MPLS.

Figura 12

Habilitación servicio MPLS.

```
BOGOTA(config)#MPLS ip
BOGOTA(config)#MPLS label protocol LDP
BOGOTA(config)#MPLS label range 21 99
BOGOTA(config)#exit
BOGOTA#
```

```
BARRANQUILLA(config)#MPLS ip
BARRANQUILLA(config)#MPLS label protocol LDP
BARRANQUILLA(config)#MPLS label range 21 99
BARRANQUILLA(config)#^Z
BARRANQUILLA#wr
```

```
R1(config)#MPLS ip
R1(config)#MPLS label protocol LDP
R1(config)#MPLS label range 21 99
R1(config)#^Z
R1#
```

```
R2(config)#MPLS ip
R2(config)#MPLS label protocol LDP
R2(config)#MPLS label range 21 99
R2(config)#^Z
R2#
```

Fuente: Autoría propia

Configuración de los Puertos de Interfaz

Dentro de este proceso es importante que se lleve a cabo la configuración de cada uno de los puertos de interfaz correspondientes al modelo planteado inicialmente, para ello se requiere un acceso a la consola para el establecimiento y ejecución de los parámetros iniciales, comenzando por las revisiones del loopback.

Figura 13

Puertos de interfaces.

```

BOGOTA#config t
Enter configuration commands, one per line. End with
BOGOTA(config)#interface f0/0
BOGOTA(config-if)#ip address 10.0.0.2 255.255.255.252
BOGOTA(config-if)#no shutdown
BOGOTA(config-if)#
*Mar 1 00:39:35.007: %LINK-3-UPDOWN: Interface FastE
*Mar 1 00:39:36.007: %LINEPROTO-5-UPDOWN: Line proto
BOGOTA(config-if)#interface f0/1
BOGOTA(config-if)#ip address 15.0.0.1 255.255.255.252
BOGOTA(config-if)#no shutdown
BOGOTA(config-if)#
*Mar 1 00:40:12.223: %LINK-3-UPDOWN: Interface FastE
*Mar 1 00:40:13.223: %LINEPROTO-5-UPDOWN: Line proto
BOGOTA(config-if)#^

```

```

BOGOTA#show ip int brief
Interface IP-Address OK? Method Status Protocol
FastEthernet0/0 10.0.0.2 YES manual up
FastEthernet0/1 15.0.0.1 YES manual up
Loopback0 5.5.5.5 YES manual up
BOGOTA#
BOGOTA#
BOGOTA#

```

```

MEDELLIN#config t
Enter configuration commands, one per line. End with
MEDELLIN(config)#interface f0/0
MEDELLIN(config-if)#ip address 20.0.0.2 255.255.255.252
MEDELLIN(config-if)#no shutdown
MEDELLIN(config-if)#
*Mar 1 00:40:09.183: %LINK-3-UPDOWN: Interface FastEthernet0/0,
*Mar 1 00:40:10.183: %LINEPROTO-5-UPDOWN: Line protocol on Inter
MEDELLIN(config-if)#interface f0/1
MEDELLIN(config-if)#ip address 25.0.0.1 255.255.255.252
MEDELLIN(config-if)#no shutdown
MEDELLIN(config-if)#
*Mar 1 00:40:32.051: %LINK-3-UPDOWN: Interface FastEthernet0/1,
*Mar 1 00:40:33.051: %LINEPROTO-5-UPDOWN: Line protocol on Inter
MEDELLIN(config-if)#
MEDELLIN#^

```

```

MEDELLIN#show ip int brief
Interface IP-Address OK? Method Status Protocol
FastEthernet0/0 20.0.0.2 YES manual up
FastEthernet0/1 25.0.0.1 YES manual up
Loopback0 7.7.7.7 YES manual up
MEDELLIN#

```

```

R1#config t
Enter configuration commands, one per line. End with
R1(config)#interface f0/0
R1(config-if)#ip address 30.0.0.1 255.255.255.252
R1(config-if)#no shutdown
R1(config-if)#interface f0/1
*Mar 1 00:25:45.711: %LINK-3-UPDOWN: Interface FastEthernet0/0, change
*Mar 1 00:25:46.711: %LINEPROTO-5-UPDOWN: Line protocol on Interface
R1(config-if)#interface f0/1
R1(config-if)#ip address 40.0.0.1 255.255.255.252
R1(config-if)#no shutdown
R1(config-if)#
*Mar 1 00:26:11.201: %LINK-3-UPDOWN: Interface FastEthernet0/1, change
*Mar 1 00:26:12.091: %LINEPROTO-5-UPDOWN: Line protocol on Interface
R1(config-if)#interface f1/0
R1(config-if)#ip address 10.0.0.1 255.255.255.252
R1(config-if)#no shutdown
R1(config-if)#

```

```

R1#show ip int brief
Interface IP-Address OK? Method Status Protocol
FastEthernet0/0 30.0.0.1 YES manual up
FastEthernet0/1 40.0.0.1 YES manual up
FastEthernet1/0 10.0.0.1 YES manual up
FastEthernet1/1 unassigned YES not administratively down due
Loopback0 1.1.1.1 YES manual up
R1#

```

Fuente: Autoría propia

Por otra parte, se requiere llevar a cabo las configuraciones de redes BGP, siendo necesario nuevamente el ingreso a la consola general del sistema asociado a cada uno de los routers en cuestión y asignando las configuraciones generales para tal efecto.

Figura 14

Redes BGP.

```
BOGOTA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BOGOTA(config)#router bgp 50
BOGOTA(config-router)#^Z
BOGOTA#
```

```
MEDELLIN#config t
Enter configuration commands, one per line. End with CNTL/Z.
MEDELLIN(config)#router bgp 50
MEDELLIN(config-router)#^Z
MEDELLIN#
```

```
BARRANQUILLA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BARRANQUILLA(config)#router bgp 50
BARRANQUILLA(config-router)#^Z
BARRANQUILLA#
```

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#router bgp 50
R1(config-router)#^Z
R1#
```

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router bgp 50
R2(config-router)#^Z
R2#
```

```
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#router bgp 50
R3(config-router)#^Z
R3#
```

Fuente: Autoría propia

Configuración del Servicio Multicast

Figura 15

Servicio Multicast.

```
BOGOTA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BOGOTA(config)#ip multicast-routing
BOGOTA(config)#^Z
BOGOTA#wr
```

```
MEDELLIN#config t
Enter configuration commands, one per line. End with CNTL/Z
MEDELLIN(config)#ip multicast-routing
MEDELLIN(config)#^Z
MEDELLIN#w
```

```
BARRANQUILLA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BARRANQUILLA(config)#ip multicast-routing
BARRANQUILLA(config)#^Z
BARRANQUILLA#
```

```
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#ip multicast-routing
R3(config)#^Z
R3#
```

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#ip multicast-routing
R1(config)#^Z
R1#
```

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#ip multicast-routing
R2(config)#^Z
R2#
```

Fuente: Autoría propia

Configuración de Entradas e Identificación Topológica

Figura 16

Configuración de entrada.

```
BOGOTA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BOGOTA(config)#router ospf 3
BOGOTA(config-router)#mpls ldp autoconfig
BOGOTA(config-router)#router-id 5.5.5.5
OSPF: router-id 5.5.5.5 in use by ospf process 1
BOGOTA(config-router)#^Z
BOGOTA#
```

```
MEDELLIN#config t
Enter configuration commands, one per line. End with CNTL/Z.
MEDELLIN(config)#router ospf 3
MEDELLIN(config-router)#mpls ldp autoconfig
MEDELLIN(config-router)#router-id 7.7.7.7
OSPF: router-id 7.7.7.7 in use by ospf process 1
MEDELLIN(config-router)#^Z
MEDELLIN#
```

```
BARRANQUILLA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BARRANQUILLA(config)#router ospf 3
BARRANQUILLA(config-router)#mpls ldp autoconfig
BARRANQUILLA(config-router)#router-id 9.9.9.9
OSPF: router-id 9.9.9.9 in use by ospf process 1
BARRANQUILLA(config-router)#^Z
BARRANQUILLA#
```

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router ospf 3
R2(config-router)#mpls ldp autoconfig
R2(config-router)#router-id 2.2.2.2
OSPF: router-id 2.2.2.2 in use by ospf process 1
R2(config-router)#^Z
R2#
```

```
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#router ospf 3
R3(config-router)#mpls ldp autoconfig
R3(config-router)#router-id 3.3.3.3
OSPF: router-id 3.3.3.3 in use by ospf process 1
R3(config-router)#^Z
R3#
```

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#router ospf 3
R1(config-router)#mpls ldp autoconfig
R1(config-router)#router-id 1.1.1.1
OSPF: router-id 1.1.1.1 in use by ospf process 1
R1(config-router)#^Z
R1#
```

Fuente: Autoría propia

Configuraciones Estáticas para la Habilitación del Servicio Multicast

Figura 17

Habilitación servicio Multicast.

```
BOGOTA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BOGOTA(config)#ip pim rp-address 1.1.1.1
BOGOTA(config)#^Z
BOGOTA#w
```

```
MEDELLIN#config t
Enter configuration commands, one per line. End with CNTL/Z.
MEDELLIN(config)#ip pim rp-address 1.1.1.1
MEDELLIN(config)#^Z
MEDELLIN#
```

```
BARRANQUILLA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BARRANQUILLA(config)#ip pim rp-address 1.1.1.1
BARRANQUILLA(config)#^Z
BARRANQUILLA#
```

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#ip pim rp-address 1.1.1.1
R1(config)#^Z
R1#
```

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#ip pim rp-address 1.1.1.1
R2(config)#^Z
R2#
```

```
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#ip pim rp-address 1.1.1.1
R3(config)#^Z
R3#
```

Fuente: Autoría propia

Configuraciones de las Redes Globales

Figura 18

Redes globales.

```
router ospf 3
mpls ldp autoconfig
router-id 5.5.5.5
log-adjacency-changes
network 5.5.5.5 0.0.0.0 area 0
network 10.0.0.0 0.0.0.3 area 0
network 15.0.0.0 0.0.0.3 area 0
!
```

```
router ospf 3
mpls ldp autoconfig
router-id 7.7.7.7
log-adjacency-changes
network 7.7.7.7 0.0.0.0 area 0
network 20.0.0.0 0.0.0.3 area 0
network 25.0.0.0 0.0.0.3 area 0
!
```

```
router ospf 3
mpls ldp autoconfig
router-id 9.9.9.9
log-adjacency-changes
network 9.9.9.9 0.0.0.0 area 0
network 30.0.0.0 0.0.0.3 area 0
network 35.0.0.0 0.0.0.3 area 0
!
```

```
R1(config)#router ospf 3
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.1 0.0.0.0 area 0
R1(config-router)#network 10.0.0.0 0.0.0.3 area 0
R1(config-router)#network 10.0.0.0 0.0.0.3 area 0
*Mar 1 00:58:35.455: %OSPF-5-ADJCHG: Process 3, Nbr
com LOADING to FULL, Loading Done
R1(config-router)#network 40.0.0.0 0.0.0.3 area 0
*Mar 1 00:58:42.667: %LDP-5-NBRCHG: LDP Neighbor 5.5
R1(config-router)#network 40.0.0.0 0.0.0.3 area 0
R1(config-router)#network 50.0.0.0 0.0.0.3 area 0
R1(config-router)#^Z
```

```
R2(config)#router ospf 3
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 2.2.2.2 0.0.0.0 area 0
R2(config-router)#network 50.0.0.0 0.0.0.3 area 0
R2(config-router)#network 50.0.0.0 0.0.0.3 area 0
*Mar 1 01:07:15.499: %OSPF-5-ADJCHG: Process 3, Nbr
com LOADING to FULL, Loading Done
R2(config-router)#network 60.0.0.0 0.0.0.3 area 0
R2(config-router)#network 60.0.0.0 0.0.0.3 area 0
*Mar 1 01:07:28.899: %LDP-5-NBRCHG: LDP Neighbor 1.1
R2(config-router)#network 20.0.0.0 0.0.0.3 area 0
R2(config-router)#
*Mar 1 01:07:42.675: %OSPF-5-ADJCHG: Process 3, Nbr
```

```
router ospf 3
mpls ldp autoconfig
log-adjacency-changes
network 3.3.3.3 0.0.0.0 area 0
network 30.0.0.0 0.0.0.3 area 0
network 40.0.0.0 0.0.0.3 area 0
network 60.0.0.0 0.0.0.3 area 0
!
```

Fuente: Autoría propia

Configuraciones de DHCP para el Establecimiento Dinámico de Conexiones TCP-IP

Figura 19

Configuración DHCP.

```
!
ip dhcp pool BOGOTA
  network 201.168.1.0 255.255.255.0
  default-router 201.168.1.1
!

MEDELLIN(config)#
MEDELLIN(config)#ip dhcp pool MEDELLIN
MEDELLIN(dhcp-config)#network 201.168.2.0 255.255.255.0
MEDELLIN(dhcp-config)#default-router 201.168.2.1
MEDELLIN(dhcp-config)#^Z
MEDELLIN#

ip dhcp pool BARRANQUILLA
  network 201.168.3.0 255.255.255.0
  default-router 201.168.3.1
!
```

Fuente: Autoría propia

Configuración de los Tiempos para el Enrutamiento

Figura 20

Tiempos de enrutamiento.

```
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#ip tcp synwait-time 5
R3(config)#^Z

BOGOTA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BOGOTA(config)#ip tcp synwait-time 5
BOGOTA(config)#

MEDELLIN(config)#
MEDELLIN(config)#ip tcp synwait-time 5
MEDELLIN(config)#

BARRANQUILLA(config)#
BARRANQUILLA(config)#ip tcp synwait-time 5
BARRANQUILLA(config)#
```

Fuente: Autoría propia

Configuraciones de Grupos Multicast

Figura 21

Grupos Multicast

```
BOGOTA(config)#int Lo0
BOGOTA(config-if)#ip pim sparse-mode
BOGOTA(config-if)#
*Mar 1 02:57:20.939: %PIM-5-DRCHG: DR change from neighbor
BOGOTA(config-if)#int f0/0
BOGOTA(config-if)#ip pim sparse-mode
BOGOTA(config-if)#
*Mar 1 02:58:01.811: %PIM-5-DRCHG: DR change from neighbor
BOGOTA(config-if)#int f0/1
BOGOTA(config-if)#ip pim sparse-mode
BOGOTA(config-if)#^Z
BOGOTA#

MEDELLIN#config t
Enter configuration commands, one per line. End with CNTL/Z.
MEDELLIN(config)#int Lo0
MEDELLIN(config-if)#ip pim sparse-mode
MEDELLIN(config-if)#int
*Mar 1 02:57:11.135: %PIM-5-DRCHG: DR change from neighbor 6
MEDELLIN(config-if)#int f0/0
MEDELLIN(config-if)#ip pim sparse-mode
MEDELLIN(config-if)#
*Mar 1 02:57:32.779: %PIM-5-DRCHG: DR change from neighbor 6
MEDELLIN(config-if)#int f0/1
MEDELLIN(config-if)#ip pim sparse-mode
MEDELLIN(config-if)#
*Mar 1 02:57:46.063: %PIM-5-DRCHG: DR change from neighbor 6
MEDELLIN(config-if)#^Z
MEDELLIN#
*Mar 1 02:57:46.063: %PIM-5-DRCHG: DR change from neighbor 6

BARRANQUILLA#config t
Enter configuration commands, one per line. End with CNTL/Z.
BARRANQUILLA(config)#int Lo0
BARRANQUILLA(config-if)#ip pim sparse-mode
BARRANQUILLA(config-if)#
*Mar 1 02:53:11.543: %PIM-5-DRCHG: DR change from neighbor 0
BARRANQUILLA(config-if)#int f0/0
BARRANQUILLA(config-if)#ip pim sparse-mode
BARRANQUILLA(config-if)#
*Mar 1 02:53:24.847: %PIM-5-DRCHG: DR change from neighbor 0
BARRANQUILLA(config-if)#int f0/1
BARRANQUILLA(config-if)#ip pim sparse-mode
BARRANQUILLA(config-if)#
*Mar 1 02:53:38.751: %PIM-5-DRCHG: DR change from neighbor 0
BARRANQUILLA(config-if)#^Z
BARRANQUILLA#
```

Fuente: Autoría propia

Configuración de los Grupos de Enrutamiento de Cada Ciudad

Figura 22

Enrutamiento en cada ciudad

<pre>BOGOTA(config)# BOGOTA(config)#class-map match-all VOIP BOGOTA(config-cmap)#match access-group 100 BOGOTA(config-cmap)#class-map match-all PING BOGOTA(config-cmap)#match access-group 103 BOGOTA(config-cmap)#class-map match-all IPTV BOGOTA(config-cmap)#match access-group 102 BOGOTA(config-cmap)#class-map match-all WEB BOGOTA(config-cmap)#match access-group 101 BOGOTA(config-cmap)#EXIT</pre>	<pre>BOGOTA#config t Enter configuration commands, one per line. End with CNTL/Z. BOGOTA(config)#policy-map QoS1 BOGOTA(config-pmap)#class VOIP BOGOTA(config-pmap-c)#priority percent 15 BOGOTA(config-pmap-c)#class WEB BOGOTA(config-pmap-c)#bandwidth percent 15 BOGOTA(config-pmap-c)#class IPTV BOGOTA(config-pmap-c)#bandwidth percent 10 BOGOTA(config-pmap-c)#class PING BOGOTA(config-pmap-c)#bandwidth percent 2 BOGOTA(config-pmap-c)#class class-default BOGOTA(config-pmap-c)#fair-queue BOGOTA(config-pmap-c)^Z BOGOTA#</pre>
<pre>MEDELLIN(config)# MEDELLIN(config)#class-map match-all VOIP MEDELLIN(config-cmap)#match access-group 100 MEDELLIN(config-cmap)#class-map match-all PING MEDELLIN(config-cmap)#match access-group 103 MEDELLIN(config-cmap)#class-map match-all IPTV MEDELLIN(config-cmap)#match access-group 102 MEDELLIN(config-cmap)#class-map match-all WEB MEDELLIN(config-cmap)#match access-group 101 MEDELLIN(config-cmap)#EXIT MEDELLIN(config)#</pre>	<pre>MEDELLIN#config t Enter configuration commands, one per line. End with CNTL/Z. MEDELLIN(config)#policy-map QoS1 MEDELLIN(config-pmap)#class VOIP MEDELLIN(config-pmap-c)#priority percent 15 MEDELLIN(config-pmap-c)#class WEB MEDELLIN(config-pmap-c)#bandwidth percent 15 MEDELLIN(config-pmap-c)#class IPTV MEDELLIN(config-pmap-c)#bandwidth percent 10 MEDELLIN(config-pmap-c)#class PING MEDELLIN(config-pmap-c)#bandwidth percent 2 MEDELLIN(config-pmap-c)#class class-default MEDELLIN(config-pmap-c)#fair-queue MEDELLIN(config-pmap-c)^Z MEDELLIN#w</pre>
<pre>BARRANQUILLA(config)# BARRANQUILLA(config)#class-map match-all VOIP BARRANQUILLA(config-cmap)#match access-group 100 BARRANQUILLA(config-cmap)#class-map match-all PING BARRANQUILLA(config-cmap)#match access-group 103 BARRANQUILLA(config-cmap)#class-map match-all IPTV BARRANQUILLA(config-cmap)#match access-group 102 BARRANQUILLA(config-cmap)#class-map match-all WEB BARRANQUILLA(config-cmap)#match access-group 101 BARRANQUILLA(config-cmap)#</pre>	<pre>BARRANQUILLA#config t Enter configuration commands, one per line. End with CNTL/Z. BARRANQUILLA(config)#policy-map QoS1 BARRANQUILLA(config-pmap)#class VOIP BARRANQUILLA(config-pmap-c)#priority percent 15 BARRANQUILLA(config-pmap-c)#class WEB BARRANQUILLA(config-pmap-c)#bandwidth percent 15 BARRANQUILLA(config-pmap-c)#class IPTV BARRANQUILLA(config-pmap-c)#bandwidth percent 10 BARRANQUILLA(config-pmap-c)#class PING BARRANQUILLA(config-pmap-c)#bandwidth percent 2 BARRANQUILLA(config-pmap-c)#class class-default BARRANQUILLA(config-pmap-c)#fair-queue BARRANQUILLA(config-pmap-c)^Z BARRANQUILLA#</pre>

Fuente: Autoría propia

Configuración de Cada uno de los Grupos de Acceso para Cada Ciudad

Figura 23

Acceso a cada ciudad.

```
BOGOTA(config)#
BOGOTA(config)#mpls ldp router-id loopback 0
BOGOTA(config)#access-list 100 permit udp any any range 16384 32000
BOGOTA(config)#access-list 101 permit tcp any any eq www
BOGOTA(config)#access-list 101 permit tcp any any eq 443
BOGOTA(config)#access-list 102 permit udp any any eq 5004
BOGOTA(config)#access-list 103 permit icmp any any
BOGOTA(config)#^Z
BOGOTA#wr
```

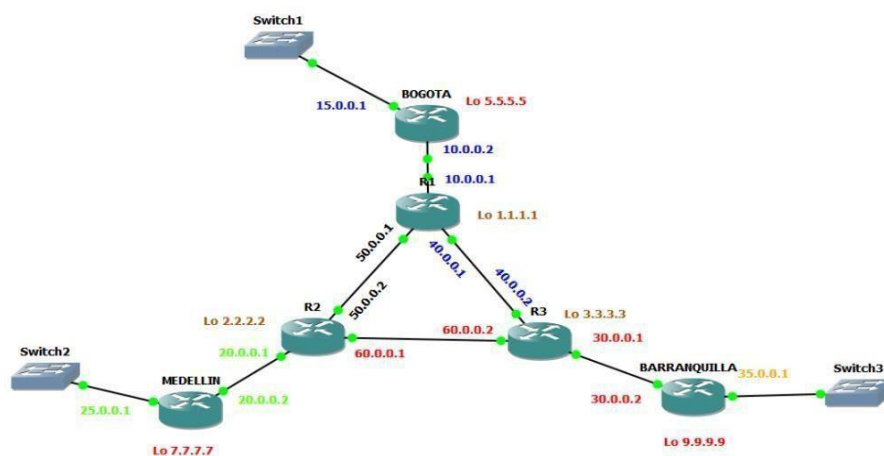
```
MEDELLIN#config t
Enter configuration commands, one per line. End with CNTL/Z.
MEDELLIN(config)#mpls ldp router-id loopback0
MEDELLIN(config)#access-list 100 permit udp any any range 16384 32000
MEDELLIN(config)#access-list 101 permit tcp any any eq www
MEDELLIN(config)#access-list 101 permit tcp any any eq 443
MEDELLIN(config)#access-list 102 permit udp any any eq 5004
MEDELLIN(config)#access-list 103 permit icmp any any
MEDELLIN(config)#^Z
MEDELLIN#
```

Fuente: Autoría propia

Segunda simulación topológica:

Figura 24

Segunda topología.



Fuente: Autoría propia

Creación de Enrutamientos Tabulares

Figura 25

Tabuladores.

```
BOGOTA#
BOGOTA#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

 5.0.0.0/32 is subnetted, 1 subnets
C       5.5.5.5 is directly connected, Loopback0
 10.0.0.0/30 is subnetted, 1 subnets
C       10.0.0.0 is directly connected, FastEthernet0/0
 15.0.0.0/30 is subnetted, 1 subnets
C       15.0.0.0 is directly connected, FastEthernet0/1
BOGOTA#
```

```
MEDELLIN#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

 20.0.0.0/30 is subnetted, 1 subnets
C       20.0.0.0 is directly connected, FastEthernet0/0
  7.0.0.0/32 is subnetted, 1 subnets
C       7.7.7.7 is directly connected, Loopback0
 25.0.0.0/30 is subnetted, 1 subnets
C       25.0.0.0 is directly connected, FastEthernet0/1
MEDELLIN#
```

```

BARRANQUILLA#
BARRANQUILLA#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
I - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - OOR, P - periodic downloaded static route

Gateway of last resort is not set

 35.0.0.0/30 is subnetted, 1 subnets
C    35.0.0.0 is directly connected, FastEthernet0/1
 9.0.0.0/32 is subnetted, 1 subnets
C    9.9.9.9 is directly connected, Loopback0
30.0.0.0/30 is subnetted, 1 subnets
C    30.0.0.0 is directly connected, FastEthernet0/0
BARRANQUILLA#

```

```

R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
I - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - OOR, P - periodic downloaded static route

Gateway of last resort is not set

 1.0.0.0/32 is subnetted, 1 subnets
C    1.1.1.1 is directly connected, Loopback0
50.0.0.0/30 is subnetted, 1 subnets
C    50.0.0.0 is directly connected, FastEthernet0/0
40.0.0.0/30 is subnetted, 1 subnets
C    40.0.0.0 is directly connected, FastEthernet0/1
10.0.0.0/30 is subnetted, 1 subnets
C    10.0.0.0 is directly connected, FastEthernet1/0
R1#

```

```

R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
I - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - OOR, P - periodic downloaded static route

Gateway of last resort is not set

 50.0.0.0/30 is subnetted, 1 subnets
C    50.0.0.0 is directly connected, FastEthernet0/0
 2.0.0.0/32 is subnetted, 1 subnets
C    2.2.2.2 is directly connected, Loopback0
20.0.0.0/30 is subnetted, 1 subnets
C    20.0.0.0 is directly connected, FastEthernet1/0
60.0.0.0/30 is subnetted, 1 subnets
C    60.0.0.0 is directly connected, FastEthernet0/1
R2#

```

Fuente: Autoría propia

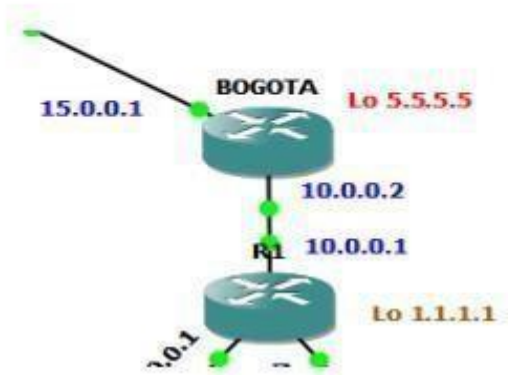
Establecimiento de Conexiones entre los Router Vecinos

Cuando se ha llevado a cabo la configuración de cada uno de los router, es necesario realizar las conexiones entre estos. Estas conexiones no pueden ser realizadas de manera aleatoria, en cambio, deberán realizarse haciendo uso de las rutas vecinas establecidas en el modelo.

Figura 26

Router vecinos Bogotá.

```
Neighbor ID      Pri   State           Dead Time   Address      Interface
1.1.1.1         1    FULL/BDR        00:00:34   10.0.0.1    FastEthernet0/0
BOGOTA#
```



Fuente: Autoría propia

Figura 27

Router vecinos Medellín y Barranquilla.

```
Neighbor ID      Pri   State           Dead Time   Address      Interface
2.2.2.2         1    FULL/BDR        00:00:36   20.0.0.1    FastEthernet0/0
MEDELLIN#
```



```
Neighbor ID      Pri   State           Dead Time   Address      Interface
9.9.9.9         1    FULL/BDR        00:00:39   30.0.0.1    FastEthernet0/0
BARRANQUILLA#
```



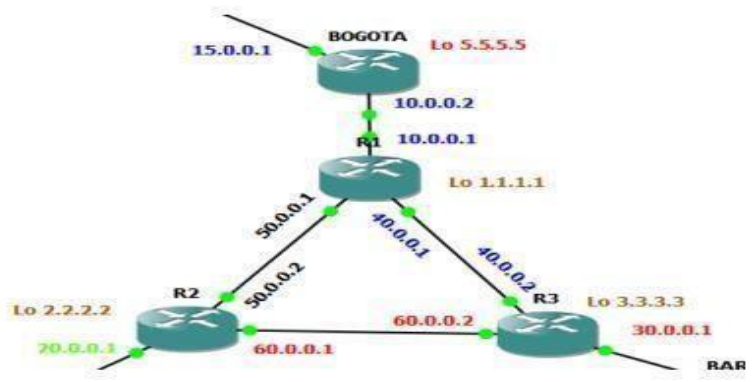
Fuente: Autoría propia

Figura 28

Router vecinos R1

```

R1#show ip ospf neighbor
Neighbor ID      Pri  State           Dead Time   Address      Interface
9.9.9.9          1    FULL/DR         00:00:39   40.0.0.2    FastEthernet0/1
15.0.0.1         1    FULL/DR         00:00:30   10.0.0.2    FastEthernet1/0
R1#
    
```



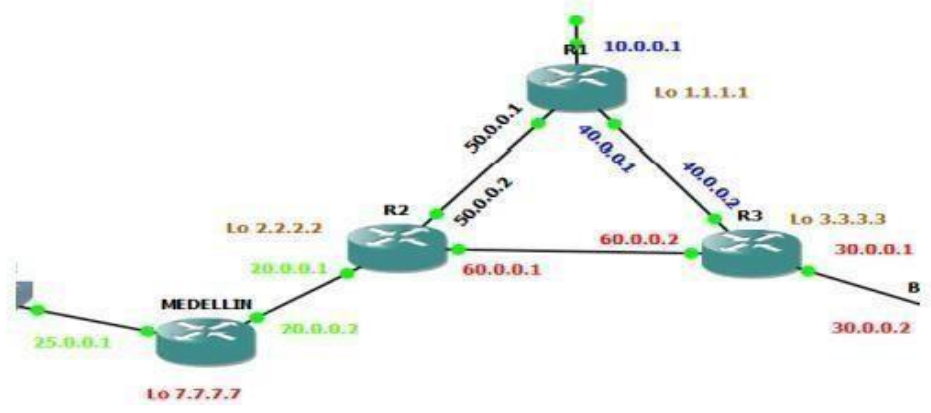
Fuente: Autoría propia

Figura 29

Router vecinos R2

```

Neighbor ID      Pri  State           Dead Time   Address      Interface
25.0.0.1         1    FULL/DR         00:00:31   20.0.0.2    FastEthernet1/0
9.9.9.9          1    FULL/BDR        00:00:33   60.0.0.2    FastEthernet0/1
1.1.1.1          1    FULL/DR         00:00:34   50.0.0.1    FastEthernet0/0
R2#
    
```



Fuente: Autoría propia

Simulación

VirtualBox – GNS3

Figura 30

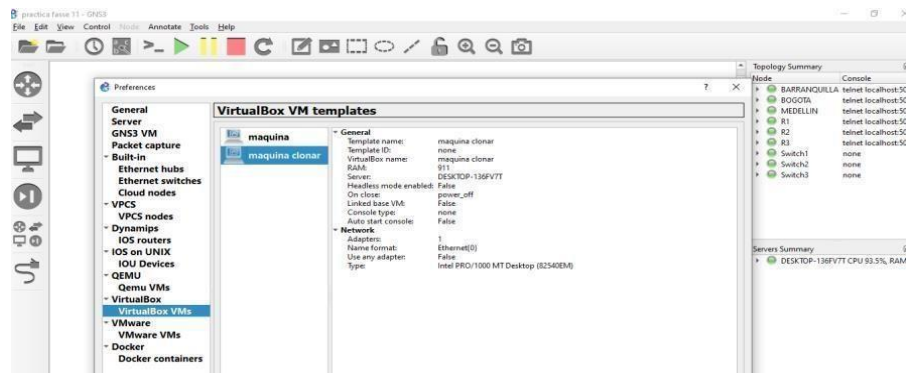
Virtual box



Fuente: Autoría propia

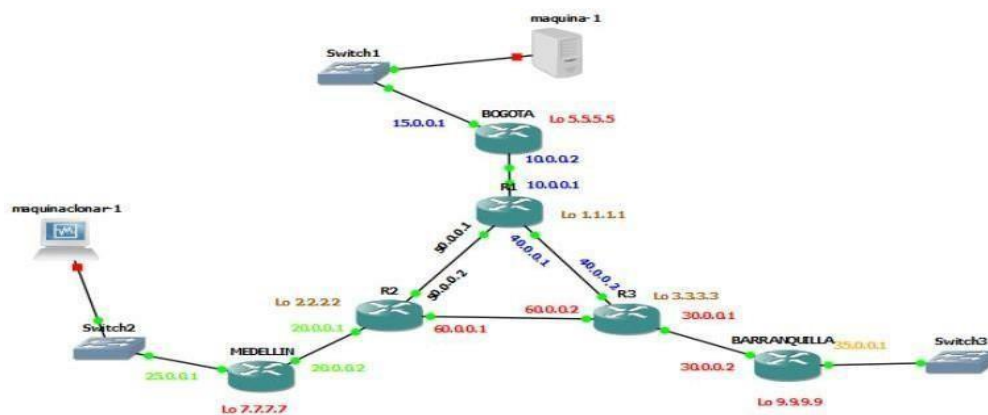
Figura 31

GNS3-Virtual box.



Fuente: Autoría propia

Una vez configurado, se debe conectar al servidor Bogotá y a los transmisores Medellín y Barranquilla.

Figura 32*Topología y servicio IPTV.**Fuente: Autoría propia*

Simulación VLC:

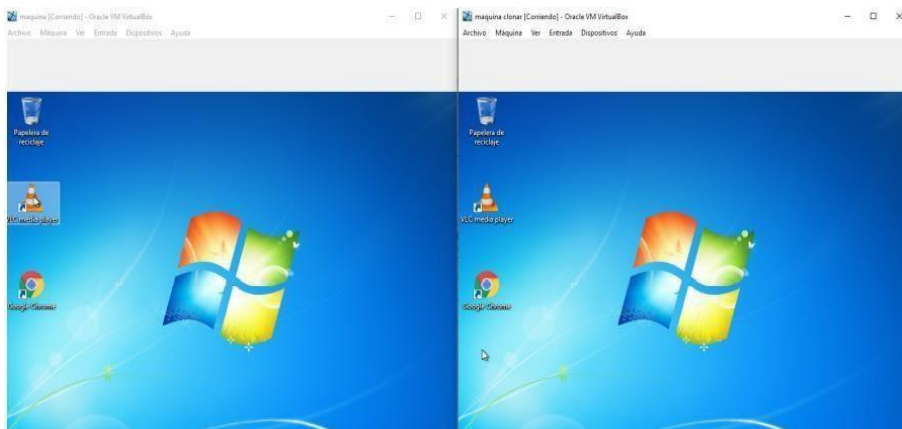
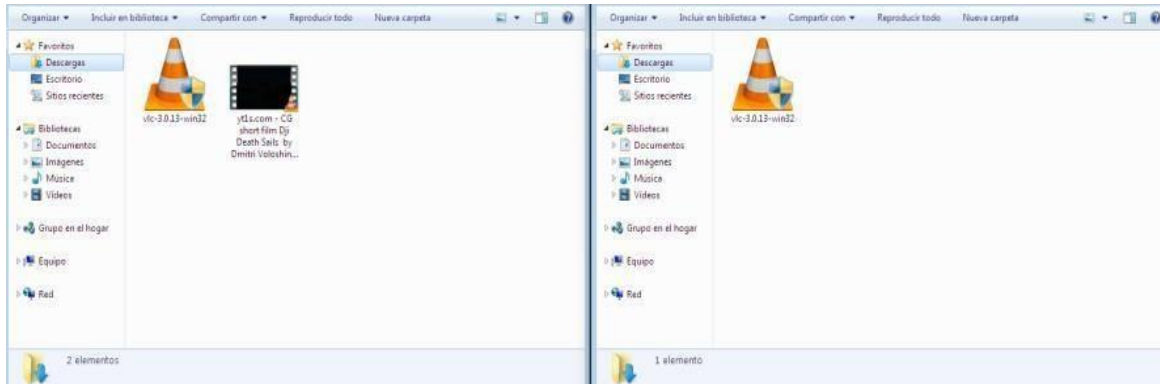
Figura 33*Máquinas de virtual box.**Fuente: Autoría propia*

Figura 34

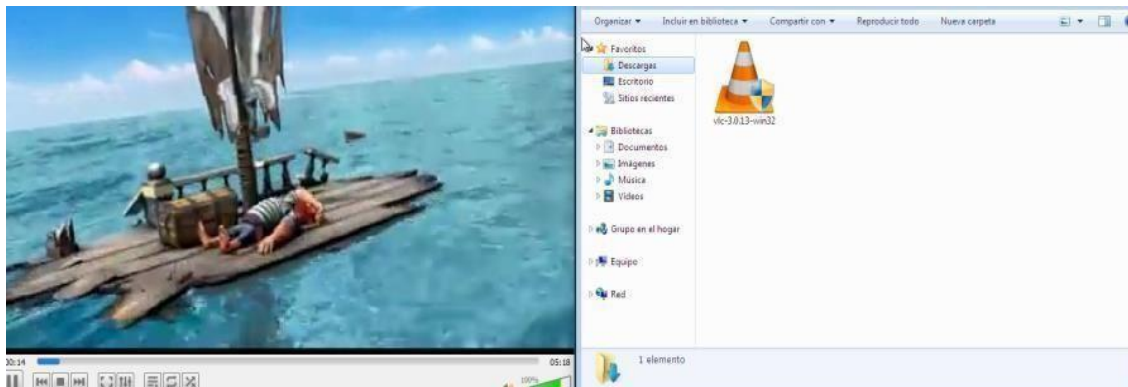
Descarga y búsqueda del video.



Fuente: Autoría propia

Figura 35

Video en la sede principal.

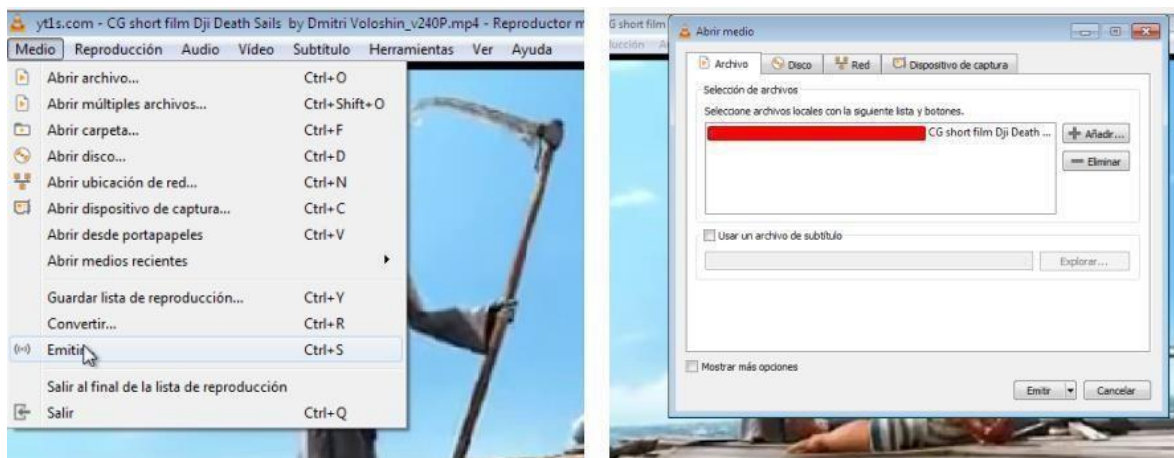


Fuente: Autoría propia

Configuración para Llevar a Cabo las Transmisiones VLC

Figura 36

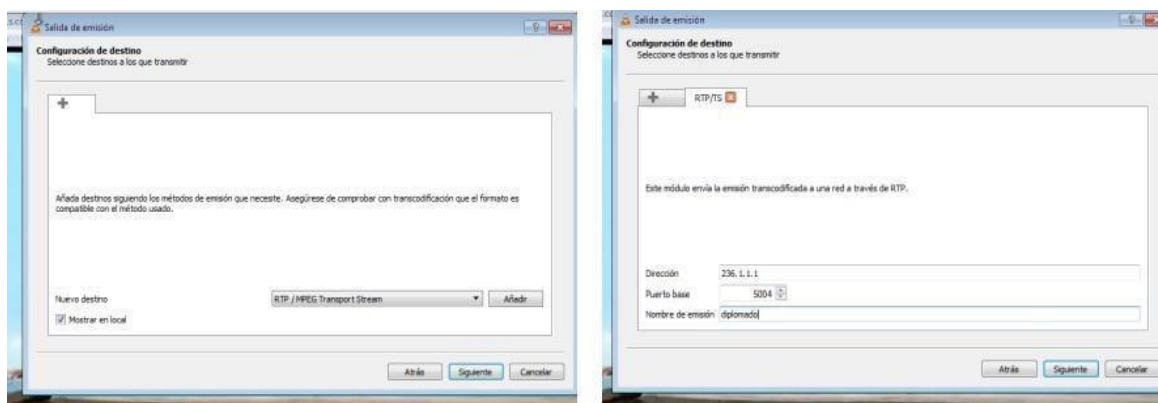
Configuración para empezar a transmitir.



Fuente: Autoría propia

Figura 37

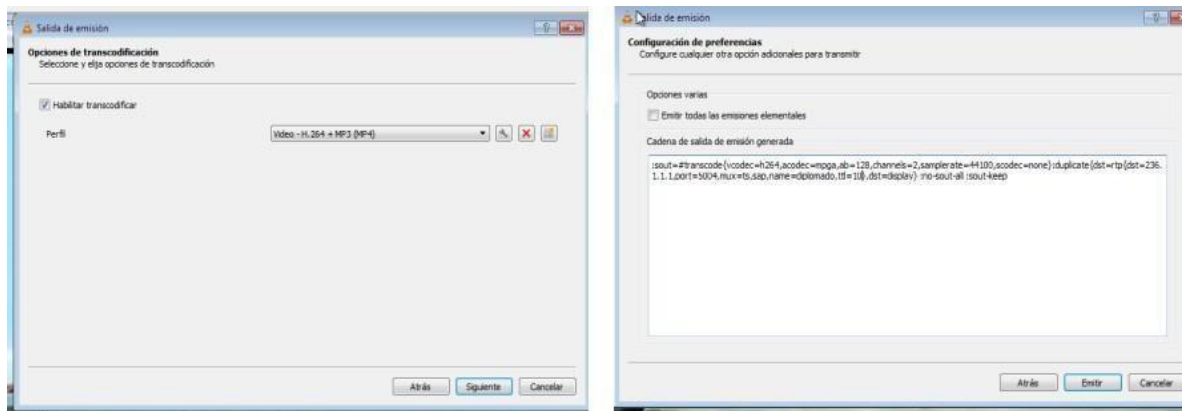
Configuración VCL emisor.



Fuente: Autoría propia

Figura 38

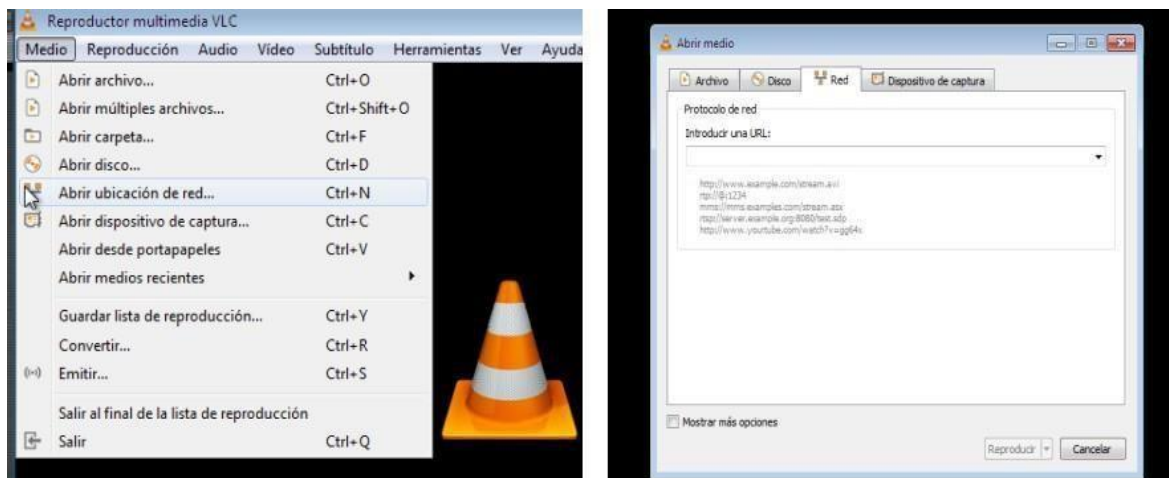
Configuración VCL receptor.



Fuente: Autoría propia

Figura 39

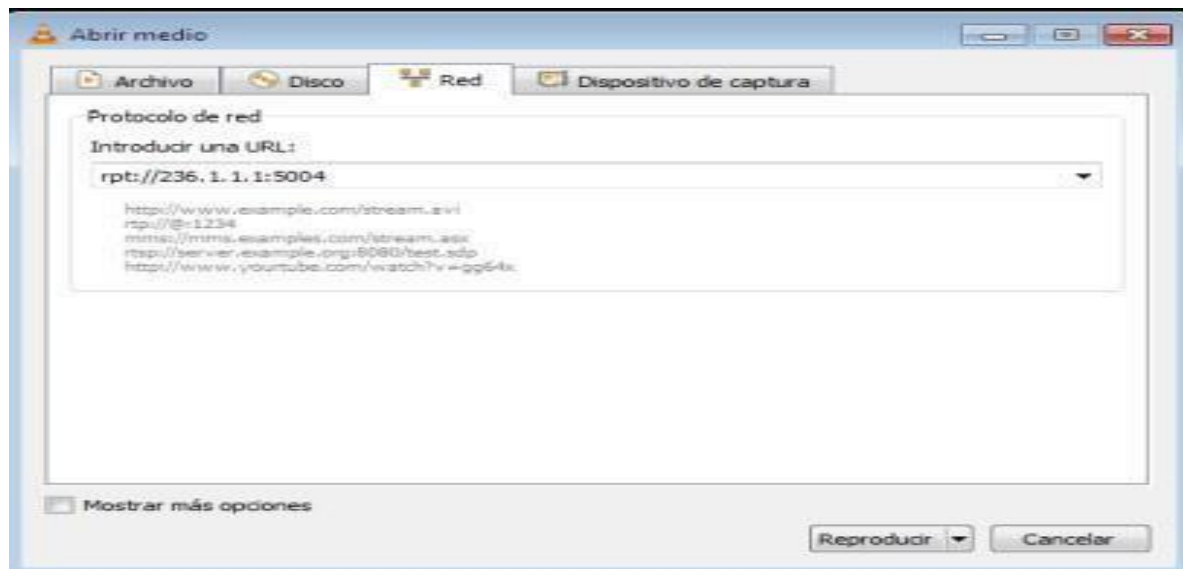
Configuración sede Barranquilla.



Fuente: Autoría propia

Figura 40

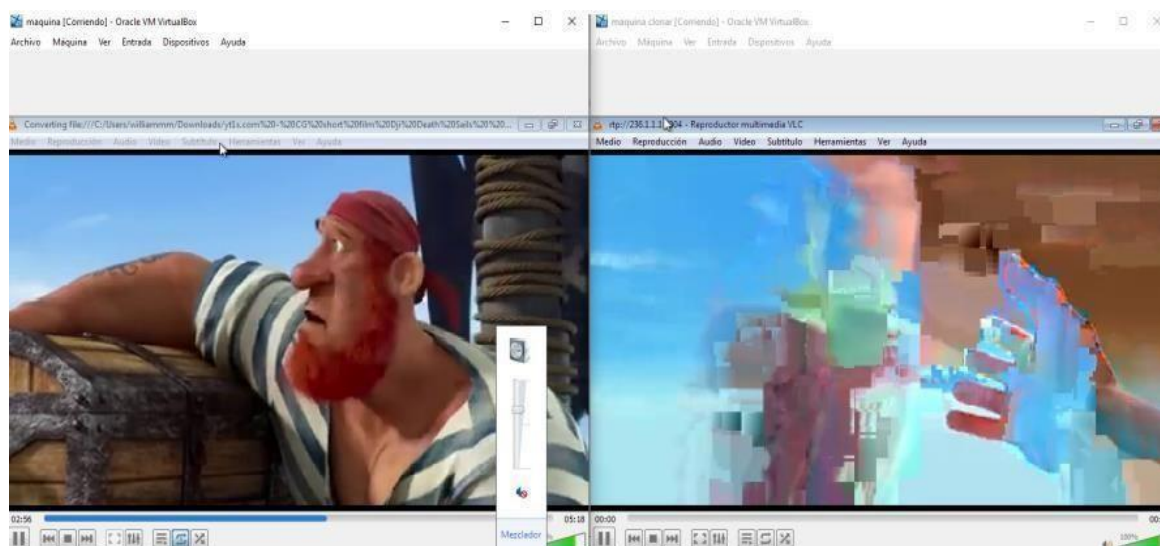
URL de la sede de Bogotá en la sede Barranquilla.



Fuente: Autoría propia

Figura 41

Transmisión desde la sede Bogotá hasta la sede Barranquilla.



Fuente: Autoría propia

Conclusiones

En resumen, en el contexto de la investigación sobre redes NGN en este trabajo, se comenzó teniendo una idea clara de los resultados que buscaba lograr. He concebido una idea de la configuración óptima para facilitar la comunicación y transmisión entre nuestras tres ciudades.

Se presentaron dificultades en la simulación cuando se intenta transmitir vídeo desde una máquina virtual en la caja de virtualización. Esta acción consumía parte de nuestra memoria RAM, lo que resultaba en la pérdida ocasional de la conexión durante la emulación, además de jugar con muchas configuraciones del VLC player para poder transmitir de manera óptima.

Además, se identificó que para implementar una estructura de IPTV, es esencial contar con una arquitectura básica que incluya un ancho de banda mínimo, calidad de servicio (QoS) y elementos fundamentales de IPTV. Estos componentes son cruciales para garantizar un uso adecuado de esta tecnología.

A través de la simulación se verificó meticulosamente si las configuraciones de red y servicios estaban correctamente establecidas y si, en efecto, se podía acceder a los servicios configurados de manera eficiente. Este proceso de verificación fue crucial para asegurar la funcionalidad y la fiabilidad del sistema en cuestión. Durante la simulación, se realizaron pruebas exhaustivas que permitieron evaluar el rendimiento y la respuesta del sistema bajo diferentes condiciones y cargas de trabajo.

El éxito de la simulación se evidenció al lograr transmitir un video de manera fluida y sin interrupciones desde las diferentes sedes, lo cual era un indicador clave de la operatividad del sistema. Este resultado positivo confirmó que la infraestructura de red estaba adecuadamente configurada y que los parámetros de transmisión de datos cumplían con los requisitos necesarios para una comunicación efectiva.

Además, se pudo determinar que la calidad del video se mantenía dentro de los estándares esperados, lo que implicaba que no solo la conectividad era adecuada, sino que también la calidad del servicio (QoS) estaba optimizada para soportar aplicaciones de alto ancho de banda como la transmisión de video en tiempo real. Este hallazgo es fundamental en entornos donde la colaboración a distancia y la comunicación visual son esenciales para las operaciones diarias.

En conclusión, la simulación fue un paso crítico que permitió validar la correcta configuración de los servicios y la infraestructura de red, asegurando así que las diferentes sedes pudieran colaborar y comunicarse efectivamente mediante la transmisión de video, lo cual es un componente vital para el funcionamiento óptimo de la organización.

En última instancia, logre llevar a cabo el desarrollo de la actividad de manera satisfactoria. complementado los avances realizados en etapas previas y los pude aplicar de manera efectiva y dinámica en este proyecto.

Bibliografía

- Arango, P., Portilla, A., y Cuéllar, C. (2013). Procedimiento para implementar QoS en la capa de acceso en redes de próxima generación enfocado en el servicio de voz. (Spanish). *Sistemas & Telemática*, 11(25), (pp. 85-104). Recuperado de [?vid=1&sid=ae eaec00-ccf5-413f-a6fb-43ad91981f48%40sdc-v-sessmgr01](https://repositorio.upct.es/bitstream/handle/10317/184/pfc908.pdf?sequence=1&isAllowed=y) Alarcon Llamas, R. (enero de 2003). Estudio e implementación de mecanismos de calidad de servicio sobre una arquitectura de servicios diferenciados. Recuperado el 5 de Mayo de 2021, de Repositorio UPCT: <https://repositorio.upct.es/bitstream/handle/10317/184/pfc908.pdf?sequence=1&isAllowed=y>
- Barba, M., y Muñoz, P. (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>
- Calidad de servicio (QoS) en un router Cisco, a partir del minuto 18:03 se visualiza esta parte, Recuperado https://www.youtube.com/watch?v=prvaYd2MUm0&t=6s&list=PL0bkKrGaLCQVvYQptv_HHRGKzQIqypV59&index=50
- Colomés, P. [Paulo Colomés]. (2020, 12 12). Fundamentos de Multicast + Ejemplo de IPTV en GNS3 [Archivo de video]. Recuperado de <https://youtu.be/3hco1ebiqo8>
- Evans, J., & Filsfils, C. (2007).