UACM

Universidad Autónoma

de la Ciudad de México

Nada humano me es ajeno

COLEGIO DE CIENCIA Y TECNOLOGÍA

LICENCIATURA EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS
Y DE TELECOMUNICACIONES

"Implementación de un modelo IPv4 Multicast"

TRABAJO RECEPCIONAL

PARA OBTENER EL TÍTULO DE LICENCIADO EN

INGENIERÍA EN SISTEMAS ELECTRÓNICOS Y DE TELECOMUNICACIONES

PRESENTA:

Juan Arnulfo López Ruiz

Director del trabajo recepcional

M. en C. José Ignacio Castillo Velázquez

México, D.F. Abril, 2015

UACM
Universidad Autónoma de la Ciudad de México
Nada humano me es ajeno

CA Coordinación Académica

SISTEMA BIBLIOTECARIO DE INFORMACIÓN Y DOCUMENTACIÓN

UNIVERSIDAD AUTÓNOMA DE LA CIUDAD DE MÉXICO
COORDINACIÓN ACADÉMICA

RESTRICCIONES DE USO PARA LAS TESIS DIGITALES

DERECHOS RESERVADOS©

Resumen

IMPLEMENTACIÓN DE UN MODELO IPV4 MULTICAST

Al inicio del presente libro el lector encontrará los fundamentos teóricos base que explican el funcionamiento de los métodos de transmisión *unicast* y *multicast*; tales fundamentos sirvieron para realizar una comparación teórica *unicast* vs *multicast*, y ofrecer con ello, las diferencias encontradas entre los dos métodos que no es fácil hallar en textos que hablen del tema. Posteriormente, se realizó un estudio y descripción del protocolo de enrutamiento *unicast* OSPF y de los protocolos de enrutamiento *multicast* IGMP y PIM-SM.

Con base en el estudio de los protocolos de enrutamiento *unicast* y *multicast*, así como con la comparación entre los métodos de transmisión *unicast* vs *multicast*, se logró comprender con mayor facilidad los fundamentos teóricos de la tecnología *IP multicast*; y posteriormente, proponer un modelo de red de datos *IPv4 multicast* realizando el diseño de la topología de red, la configuración de aquellos equipos que intervinieron en el modelo de red *IPv4 multicast*, y finalmente, la emulación del modelo para lograr transmitir a un grupo de receptores información multimedia, en específico, audio y video *streaming*. Parte importante para lograr el diseño, configuración y emulación del modelo propuesto, fue hacer uso del simulador-emulador avanzado de redes GNS3, el cual permitió gestionar en su totalidad equipos de red (como *routers* y computadoras) de tal manera que los mismos desempeñaran funciones tan precisas como si fueran equipos físicos reales.

Agradecimientos

A la Universidad Autónoma de la Ciudad de México, UACM, por brindar a miles de jóvenes la oportunidad de vivir la irreemplazable experiencia de ser universitarios. Por contribuir a que México tenga más y mejores mujeres y hombres profesionales comprometidos con el país, con la ciudad de México y su sociedad. De igual manera, agradezco a la UACM por el apoyo otorgado para la impresión y empastado del presente trabajo.

A mis padres Arnulfo López y Elizabeth Ruiz por su apoyo incondicional; mis logros son resultado del gran trabajo que han realizado como padres.

Particular agradecimiento al M. en C. José Ignacio Castillo Velázquez por dirigirme en el largo pero fructuoso proceso para la realización de este trabajo. Así mismo, por compartirme su amplio conocimiento y experiencia, brindarme todo su apoyo, y ser parte fundamental en mi formación académica.

Agradezco a las profesoras M. en C. Magali Cortez Vázquez y Mtra. Rita Vázquez Padilla, así como a los profesores M. en C. Joel Yazbek Buendía Gómez e Ing. Ricardo Galindo Reyes por su apoyo como lectores del presente trabajo, e indudablemente como importantes participes de mi formación académica.

Índice de contenido

CAPÍTULO III

CAPÍTULO IV

CAPÍTULO V

Capítulo I

Introducción

I.1 Antecedentes

En las redes de datos la transmisión y recepción de información representada como datos y éstos representados en forma de texto, imágenes, voz, audio y video es posible gracias a la intervención (en el mismo proceso de transmisión/recepción) de diversos protocolos, tecnologías y métodos de comunicación de red que siguen determinados estándares.

En cuanto a los métodos de comunicación de red usados para difundir, emitir o transmitir información a los elementos finales (también llamados usuarios dentro de un sistema de telecomunicaciones o de una red de datos) que soliciten en determinado momento dicha información podemos destacar a tres: el método de transmisión *unicast* (unidifusión), *multicast* (multidifusión), y *broadcast* (difusión).

El uso de cada uno de los tres métodos será de acuerdo al servicio que se quiera ofrecer a los receptores en un sistema de telecomunicaciones, y principalmente, en una red de datos. Cada uno de los tres métodos de transmisión presentan diferencias entre sí, y la principal diferencia existente entre estos tres métodos está en la cantidad de usuarios que recibirán cierta información (datos), principalmente de manera simultánea dentro de una red de datos.

En la actualidad, la transmisión/recepción y reproducción de audio y video es ya una práctica cotidiana y hasta indispensable que se lleva a cabo dentro de las redes de datos internas (*Intranets*) de empresas e instituciones ya sea públicas o privadas. Y en ese sentido, la transmisión/recepción de audio y video en forma simultánea a un gran número de usuarios que se encuentran dentro de una Intranet, y que se lleva a cabo haciendo uso del método *unicast* es muy común. Sin embargo, hacer uso del mencionado método de comunicación puede ocasionar que el BW (*Bandwidth*) disponible en la

Intranet, y la capacidad de procesamiento de los equipos de red (como servidores, *routers* y computadoras personales) se vean sobrepasados, y como consecuencia de ello, que la información solicitada por los usuarios quizá nunca llegue a su destino, o en el mejor de los casos, llegue con muy mala calidad para poder ser visualizada o escuchada por estos últimos.

Una solución eficaz que resuelve el problema planteado con anterioridad, y que se sustenta en el método de transmisión *multicast*, es la tecnología IP *multicast*.

La tecnología IP *multicast* nace en 1992, y actualmente es considerada como una tecnología avanzada usada como solución al hecho de hacer eficiente el consumo del BW, ya que minimiza el tráfico de datos dentro de una red; y así mismo, reduce el procesamiento de los equipos, principalmente servidores y *routers*; ya que IP *multicast* provee un único flujo de datos para muchos usuarios que desean recibir información de manera simultánea. Como ejemplo claro de la implementación y uso de *IP multicast* se encuentran las *Intranets* de distintas redes avanzadas del mundo que integran la denominada *Internet 2* (como por ejemplo Cudi y CLARA). Las cuales hacen uso de los servicios que se desprenden de la propia tecnología para ofrecerlos y beneficiar a todos los usuarios que se conectan a las redes avanzadas, y donde además se posibilitan el estudio y la experimentación con dicha tecnología.

El desarrollo de IP *multicast* a lo largo de las últimas dos décadas ha tenido grandes resultados especialmente para la transmisión de audio y video. Si se suma el crecimiento de IP *multicast* con el uso de la tecnología *streaming* (surgida en el año de 1995) para audio y video; el resultado es que la implementación de IP *multicast* para la transmisión y reproducción de audio y video streaming con aplicación directa en audioconferencias y videoconferencias sobre redes internas (*Intranets*) empresariales o de instituciones tanto públicas como privadas es muy prometedor en un futuro muy cercano. Por ello la importancia del presente trabajo, que está enfocado en el estudio (abarcando el diseño, la configuración y emulación) de un modelo de red de datos funcionando con tecnología *IPv4 multicast*, que sirva como referencia base para la transmisión de audio y video streaming.

I.2 Objetivo

Obtener un modelo *IPv4 multicast* para la transmisión de audio y video *streaming*. El modelo se obtendrá a través del diseño, configuración y emulación de una propuesta topológica de red de datos interna (Intranet). Se empleará el *software* GNS3 para obtener el modelo y realizar la transmisión de audio y video *streaming* dentro del mismo. La configuración y emulación del modelo de red *IPv4 multicast* servirán como referencia base para la implementación de la tecnología *IPv4 multicast* y la transmisión de audio y video *streaming* en cualquier otro diseño topológico de Intranet de una empresa o institución tanto pública como privada.

I.3 Organización del proyecto

El trabajo se compone de tres etapas. La primera de ellas es la etapa introductoria, donde se hace una revisión de la base teórica de acuerdo a las referencias obtenidas.

La segunda etapa consiste en presentar una propuesta de un modelo o topología de red *IPv4 multicast*, que sirva como modelo de referencia para realizar transmisión de audio y video *streaming*.

En la tercera etapa se recauda la información obtenida de lo realizado en las etapas uno y dos para presentar los resultados y conclusiones.

En el capítulo 2 se presenta una revisión teórica del proceso de funcionamiento del método de transmisión *unicast*, así como algunas generalidades del mismo; se hace lo propio con el método de transmisión *multicast.* Con lo anterior, se sustenta la importancia de hacer una comparación entre el método de transmisión *unicast vs multicast*. Posteriormente, se hace una descripción general del funcionamiento y de algunas de las características de la tecnología *IP multicast,* se hace una revisión de los antecedentes históricos de la misma, se describen las aplicaciones y los modelos de transmisión que hacen uso de la tecnología, se hace una revisión del uso de *IPv4 multicast* sobre escenarios de *Intranet* e *Internet*. Finalmente, se exponen algunas generalidades de los servicios multimedia, se hace una revisión de la tecnología *streaming* y se describe brevemente la importancia que tiene el uso del protocolo RTP (*Real-time Transport Protocol* ‘Protocolo de transporte en tiempo real’).

En el capítulo 3, se presentan brevemente los protocolos de tipo IGP (*Internal Gateway Protocol* 'Protocolo de puerta de enlace interno'*)* y EGP (*External Gateway Protocol* 'Protocolo de puerta de enlace externo') usados en redes de tipo *unicast.* Se profundiza en la teoría del protocolo OSPF (*Open Shortest Path Protocol* 'Protocolo primero el camino más corto'); ya que es el protocolo que se eligió para proveer el enrutamiento dinámico *unicast* en el modelo de red *IPv4 multicast* propuesto. Así mismo, se ofrece una breve introducción de los principales tipos de protocolos que se usan en la implementación de redes *multicast*, pero se profundiza en los protocolos IGMPv2 (*Internet Group Management Protocol* 'Protocolo de administración de grupos de Internet') y PIM-SM (*Protocol Independent Multicast – Sparse Mode* 'Protocolo independiente multicast- modo esparcido'), ya que también fueron útiles para ser usados en el modelo de red IPv4 de multidifusión propuesto.

Para el capítulo 4 se presenta el diseño, configuración y emulación del modelo de red *IPv4 multicast* propuesto. Así como a la transmisión de audio y video *streaming* a través de la red *multicast*; con efecto de validar el funcionamiento de la red se ofrece la captura que se realizó del tráfico generado antes y en el momento de la transmisión de audio y video.

En el capítulo 5 se presentan los resultados y conclusiones que se obtuvieron del trabajo en su totalidad.

Capítulo II

Unicast, Multicast e IP Multicast

La mejor manera de explicar la tecnología *IP multicast* es haciendo una comparativa entre *unicast vs multicast*. Por tal motivo, en este capítulo se empieza por dar una definición concreta del significado de transmisión *unicast,* se hace una revisión del cómo funciona *unicast* a nivel de transporte, a nivel IP y a nivel MAC, para posteriormente hacer lo propio con el concepto de transmisión *multicast*. En seguida, se realiza una comparativa entre los modos *unicast vs multicast*, se ofrece un contexto general del funcionamiento de la tecnología *IP multicast* incluyendo una revisión histórica de la misma, una revisión de las aplicaciones que son soportadas en esta tecnología, así como de los modelos de transmisión existentes para la distribución de servicios en una red de datos. Finalmente, se ofrece una revisión a los servicios denominados multimedia, específicamente aquellos llamados *multimedia streaming* y su importancia en redes con servicio *IP multicast*, se concluye con un resumen del protocolo RTP.

Actualmente existen diversos métodos de transmisión que definen la manera en que los equipos o sistemas que componen una red de computadoras (o simplemente red de datos) envían o transmiten información a uno, varios o a todos los sistemas finales participantes dentro de esa red. Entre los métodos de transmisión existentes para transmitir información podemos encontrar a tres que destacan; *Unicast* (Unidifusión), *Multicast* (Multidifusión), y *Broadcast* (Difusión). Existen también otros métodos de transmisión como *Anycast*, que tiene como principio enviar información a cualquier destino que sea el más cercano, considerando a este como el mejor destino en una topología de red. Otro método es *Geocast*, cuyo funcionamiento se basa en enviar información a una zona geográfica específica, como por ejemplo, destinos ubicados en un único país o en un sólo continente [1].

Con base en lo descrito anteriormente, es importante tener claro el concepto de los métodos de transmisión *unicast*, *multicast y broadcast*; métodos utilizados para la transmisión de datos (ya sea en forma de texto, imágenes, audio y video), para posteriormente hacer una revisión más detallada únicamente de los métodos de transmisión *unicast* y *multicast*.

A continuación, se da una definición básica para cada uno de los tres métodos. Con las definiciones ofrecidas, se hará evidente la principal diferencia existente entre cada tipo de transmisión: enviar información a uno, a varios, o a todos los equipos en una red de datos.

Unicast es el concepto que refiere a la comunicación que se realiza entre un sistema emisor y un único destinatario; en la figura II.1.1 se observa una representación simbólica de dicha comunicación. El concepto de *multicast*, hace referencia a la comunicación que se realiza desde un equipo emisor a un conjunto de destinatarios; en la figura II.1.2 se muestra una representación simbólica de *multicast*. Por último, una definición breve de *broadcast* es la de la comunicación o transmisión de información entre un sistema emisor y todos los posibles destinatarios, como se muestra en la representación simbólica de la figura II.1.3 [2].

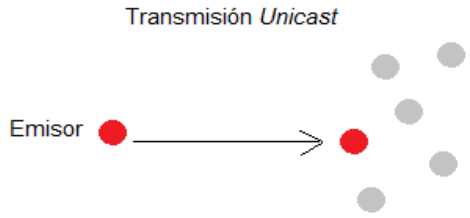


Figura II.1.1. Representación simbólica del método de transmisión *unicast*; un sistema emisor transmite a un único sistema receptor, usualmente se usa en redes de datos como *Internet* [Diagrama propio].

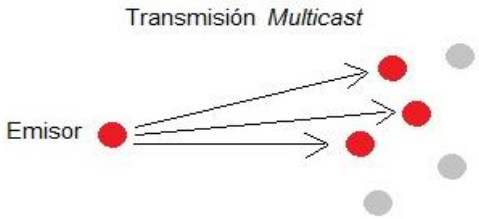


Figura II.1.2. Representación simbólica de *multicast*; un sistema emisor transmite a un grupo compuesto por muchos sistemas receptores, usualmente se usa en redes de datos que ofrecen aplicaciones multimedia [Diagrama propio].

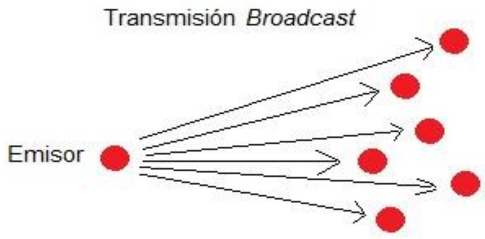


Figura II.1.3. Representación simbólica del método de transmisión *broadcast*; un sistema emisor transmite a todos los sistemas receptores, en particular este tipo de comunicación se usa en servicios como el de radiodifusión [Diagrama propio].

En las figuras II.1.1, II.1.2 y II.1.3 se muestran esquemas simbólicos que representan los tres métodos. Ciertamente, estos modos de transmisión no pueden ser aplicados a todo tipo de sistemas de telecomunicaciones por igual, ni tampoco pueden ser usados sin distinción para la transmisión de información dentro de las redes de datos; por tal motivo, es preciso puntualizar que el método de transmisión *unicast* es el más empleado actualmente en la comunicación y transferencia de información que se realiza en pequeñas redes LAN creadas en los años 60, de igual manera que en redes de computadoras de mayor extensión como las redes de tipo MAN y WAN, así como en su conjunto, representadas por la red de *Internet.*

Ya se ha dado una definición muy general de *unicast, multicast y broadcast*. Sin embargo, en el presente trabajo se realizó un estudio más puntual de los métodos de transmisión *unicast* y *multicast,* para presentar posteriormente una comparativa entre los mismos.

II.1 Unicast

Dentro de los métodos de transmisión (*unicast*, *multicast* y *broadcast*) en las redes de datos se encuentra *unicast*, el cual permite una comunicación del tipo *host-to-host* y suele ser el más empleado en redes LAN, MAN, así como en redes WAN, y por supuesto en la red de *Internet*.

El término *unicast*, se define como la transmisión en la que se envían paquetes de datos desde una fuente emisora a un destino o receptor específico, como se indica en la figura II.1.4 [3].

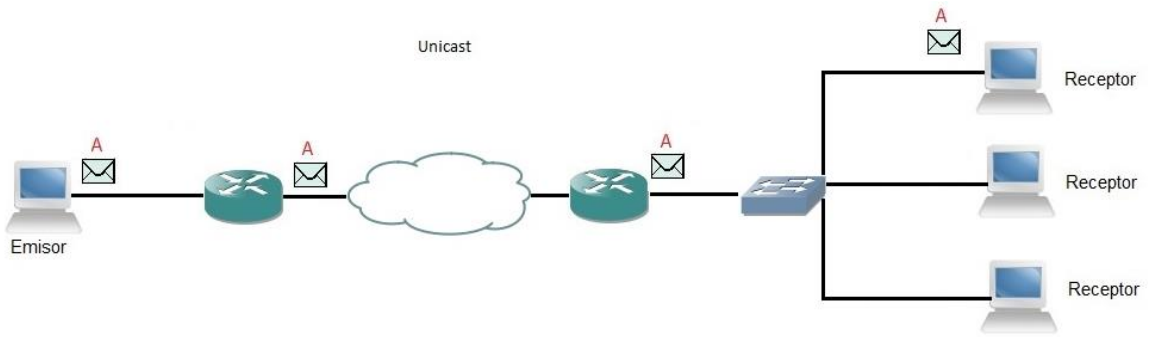


Figura II.1.4. Transmisión *unicast* de un paquete de datos A desde un emisor a un receptor específico en una red. [Diagrama propio con base en la referencia 3]

Los servicios más comunes que hacen uso del método *unicast* son diversos; entre ellos podemos encontrar ejemplos muy comunes como: la conexión a servidores FTP para la descarga de archivos, la visualización de una página web haciendo una conexión a un servidor Web a través de HTTP, el envío y recepción de correos a través de SMTP, e incluso la visualización de videos usando *streaming* o VoD *(Video on Demand 'Video bajo demanda')*.

Debido a que en una red *unicast* la comunicación es *host-to-host*, el emisor o fuente principal envía un flujo de datos a cada receptor o usuario que lo solicite; es decir, el número de paquetes que genere el sistema emisor será igual al número de sistemas receptores que se lo soliciten. La situación antes descrita, más el hecho de que en muchas ocasiones un gran número de equipos solicitan simultáneamente un mismo flujo de información que se encuentra almacenada en un lugar de la red (usualmente un servidor como se muestra en la figura II.1.5), darán como resultado que la fuente emisora tenga un aumento en su carga de procesamiento conforme mayor sea el número de equipos receptores; y que por lo tanto, se vea afectada su función de enviar la información adecuadamente a todo aquel que la solicite, que la congestión en la red donde se realiza el proceso de comunicación entre emisor y receptores sea grande en cuanto al número de paquetes que fluyen en la misma, ocasionando que el consumo del BW disponible en la red aumente pudiendo llegar a saturar el máximo de su capacidad, lo cual ocasionaría un colapso de la red.

Las tres situaciones anteriores, son temas que durante mucho tiempo y hasta la actualidad han sido ampliamente estudiados debido a su importancia para el buen funcionamiento de las redes datos, y por supuesto, para mejorar la calidad en los servicios que son ofrecidos en estas últimas [4].

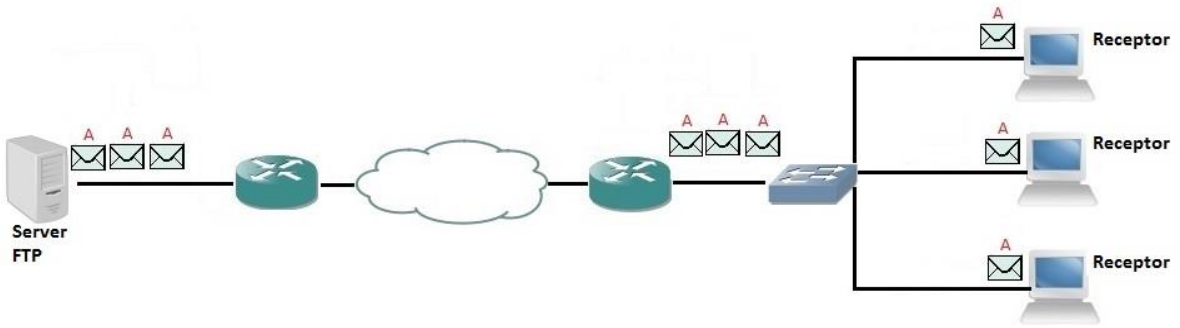


Figura II.1.5. Transmisión *unicast*; un servidor envía una copia de paquete de datos a cada uno de los tres receptores que solicitan el paquete de datos A [Diagrama propio].

II.1.1 Protocolos TCP y UDP

En las redes de computadoras destacan dos protocolos que se encargan de transportar los datos y que funcionan en la capa 4 de transporte, ya sea del modelo de interconexión de sistemas abiertos ISO/OSI o del modelo de protocolos de red TCP/IP. Estos dos protocolos son:

- TCP (*Transport Control Protocol*); se conoce como protocolo de servicio confiable y orientado a conexión.
- UDP (*User Datagram Protocol*); denominado protocolo de servicio no confiable y no orientado a conexión.

II.1.1.1 Protocolo TCP

Una de las funciones que realiza la capa de transporte es segmentar o dividir en bloques un mensaje grande (proveniente de la capa de aplicación). A estos bloques se les denomina segmentos cuando el protocolo de transporte utilizado es TCP, teniendo este último dos funciones; asignar un número consecutivo secuenciando a cada uno de los segmentos que sean parte del mensaje, y además, agregarle un encabezado TCP a cada segmento creado. En la figura II.1.6 se observa un esquema demostrativo de lo descrito anteriormente, y en la figura II.1.7 se muestra encabezado (HDR-*Header*) del protocolo TCP.

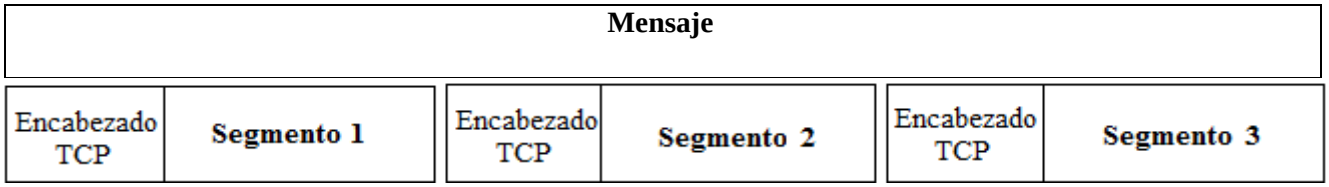


Figura II.1.6. Partición de un mensaje en segmentos numerados consecutivamente y la agregación de un encabezado a cada uno de los segmentos por el protocolo TCP en la capa 4 de transporte.

Bit (0)		Bit (15) Bit (16)	Bit (31)
Source Port (16)		Destination Port (16)	
Sequence number (32)			
Acknowledgement number (32)			
Header (4)	Reserve (6)	Flags (6)	Window (16)
TCP Checksum (16)		Urgent Pointer (16)	
Options (0 or 32 if any)			
APPLICATION LAYER DATA SEGMENT			

Figura II.1.7. Encabezado TCP; todo el contenido de este encabezado tiene un peso de 20 o 24 Bytes y se inserta en cada segmento TCP de un mensaje [Diagrama propio con base en la referencia 5].

Cada uno de los campos del encabezado TCP tiene una función específica como parte del protocolo, y además, recibe un peso en bits (números entre paréntesis). Si sumamos los bits de cada campo, el total resultante es de 192 bits o su equivalente 24 Bytes; peso total de la cabecera TCP.

El método de direccionamiento de la capa de transporte son los puertos; elementos que forman parte del primer campo de la cabecera TCP. El puerto fuente (*Source port*), al igual que el puerto destino (*Destination port*) reciben un peso de 16 bits cada uno, lo que nos da una relación para saber el número de puertos existentes haciendo: $2^{16} = 65536$ puertos. Algunos de los 65536 puertos existentes están asignados de manera fija por el *Service Name and Transport Protocol Port Number Registry* 'Servicio de nombres y Registro de Protocolo de Transporte y Número de puerto', organismo de la IANA (*Internet Assigned Numbers Authority* 'Autoridad de Internet para la asignación de números'), ahora ICCAN (*Internet Corporation for Assigned Names and Numbers* 'Corporación de Internet para la asignación de nombres y números'), que define los números de puertos para aplicaciones que corren en equipos de cómputo. Por ejemplo, la aplicación de correo POP3 hace uso del puerto 110, para una conexión a través de HTTP se usa el puerto 80, una aplicación de chat en *Internet* se hace a través del puerto 531 [5] [6].

Es importante tener claro la importancia de la existencia de los puertos (desde la perspectiva de puerto como canal de comunicación), los cuales permiten que una computadora pueda conectarse y correr múltiples aplicaciones o servicios sin que estos se interrumpan entre sí.

Para mayor información de la función de cada uno de los campos del encabezado TCP consulte el apartado 3.1 de la página 14 del documento RFC-793 de la referencia 5.

Por otra parte, TCP se conoce como un protocolo de servicio confiable y orientado a conexión. Se dice que es de servicio confiable debido a que asegura que los datos enviados por un equipo fuente lleguen al equipo destino y además lleguen de manera ordenada, es decir, en el orden tal cual la fuente los envió al equipo destino en su totalidad; sin que falte ningún dato. Por otro lado, se dice que es un protocolo orientado a conexión debido a su mecanismo de operación, que se ejecuta en la capa 5 de sesión del modelo ISO/OSI, y que asegura un inicio de sesión, un mantenimiento de conexión y una finalización de sesión.

El proceso de inicio de sesión o establecimiento de inicio de sesión TCP se conoce como saludo de tres vías debido a que cuando un equipo, llamémoslo A, trata de establecer una comunicación con otro equipo, llamémosle equipo B; el primero, envía un mensaje de saludo SYN (*Synchronise*), cuando este mensaje llega al equipo B, éste último responderá con un mensaje SYN+ACK (*Synchronise + Acknowledgement*) hacia el equipo A, es decir, la respuesta al saludo más un mensaje de acuse de recibido; por último, el equipo A enviará un mensaje final ACK que indicará al equipo B que el mensaje SYN+ACK fue recibido satisfactoriamente, y por lo tanto, el inicio de sesión estará establecido. El proceso de saludo de tres vías se muestra en el esquema de la figura II.1.8 [7] [8].

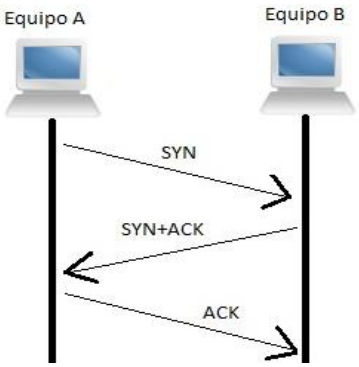


Figura II.1.8. Saludo de tres vías: establecimiento de inicio de sesión TCP [Diagrama propio con base en la referencia 7].

En el proceso que realiza TCP para mantener una sesión activa entre un equipo A y un equipo B simplemente se hace uso de un mensaje denominado *Keep a Live*; este mensaje será enviado por el equipo A cada cierto tiempo en segundos al equipo B, indicándole que aún tiene interés por mantener activa una sesión, tal y como se muestra en el esquema de la figura II.1.9.

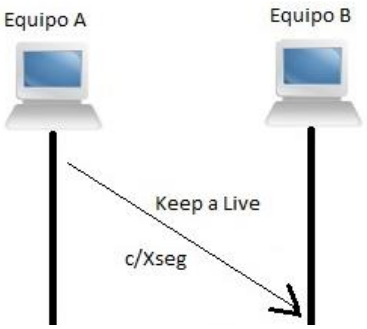


Figura II.1.9. Proceso de mantener una sesión activa en el protocolo TCP [Diagrama propio con base en la referencia 7 y 8].

El proceso de finalización de sesión es muy similar al proceso de inicio de sesión y mantenimiento de sesión realizado por TCP. La única diferencia radica en el número de mensajes que un equipo A y un equipo B se envían entre ellos, y el nombre con el que se identifica cada mensaje tal y como se muestra en el esquema de la figura II.1.10.

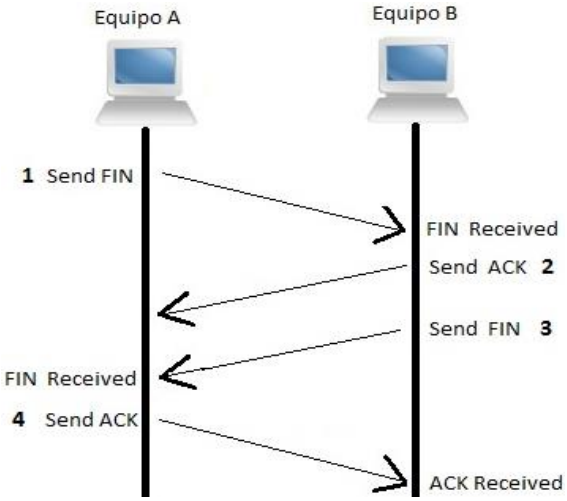


Figura II.1.10. Proceso de finalización de sesión entre dos equipos usando el protocolo TCP [Diagrama propio con base en la referencia 7 y 8].

II.1.1.2 Protocolo UDP

A los bloques que se crean al dividir un mensaje grande en la capa de transporte se les denomina datagramas cuando el protocolo de transporte usado es UDP. A diferencia de TCP, UDP no asigna un número a cada datagrama que sea parte del mensaje, sin embargo, sí agrega un encabezado UDP a cada datagrama creado. En la figura II.1.11 se observa un esquema demostrativo de lo descrito anteriormente, y en la figura II.1.12 se muestra el encabezado del protocolo UDP.

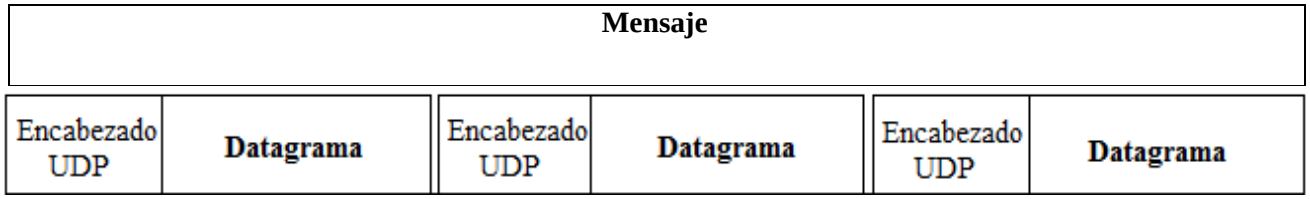


Figura II.1.11. Esquema que representa la partición de un mensaje en datagramas y la agregación de un encabezado por el protocolo UDP en la capa 4 de transporte.

Bit (0)	Bit (15) Bit (16)	Bit (31)
Source Port (16)	Destination Port (16)	
Lenght (16)	Checksum (16)	
APPLICATION LAYER DATA SEGMENT		

Figura II.1.12. Encabezado UDP; todo el contenido de este encabezado tiene un peso de 8 Bytes y se inserta en cada segmento UDP de un mensaje [Diagrama propio con base en la referencia 9].

Cada uno de los campos del encabezado UDP contiene un peso en bits (números entre paréntesis) que nos indican el peso total del encabezado. Sumando los bits de cada campo el total resultante es de 64 bits o su equivalente 8 Bytes; peso total de la cabecera UDP. El hecho de que la cabecera tenga un peso de 8 Bytes, notablemente más pequeño que el de la cabecera TCP con 24 Bytes de peso, reduce considerablemente el uso de BW en el proceso de comunicación entre equipos que usan el protocolo UDP [9].

Al igual que en TCP, en UDP los puertos tanto de fuente como de destino son elementos que forman parte del primer campo de la cabecera. Para mayor información de la función de cada uno de los campos de la cabecera UDP consulte el documento RFC-768 de la referencia 9.

UDP se conoce como un protocolo de servicio no confiable y no orientado a conexión debido a que no tiene un proceso de inicio de sesiones ni de mantener activa una sesión ni tampoco de finalizar una sesión; únicamente envía el flujo de datos como tal, además de que no etiqueta con números a los datagramas, por lo tanto, UDP no contiene un mecanismo para ordenar el flujo de datagramas tal y como fueron enviados por la fuente. A diferencia de TCP, UDP no reenvía datagramas perdidos en el camino, por lo que la información es susceptible a llegar al destino de forma incompleta o con errores.

En definitiva, podemos decir que TCP es un protocolo que se utiliza para servicios que necesitan una transmisión segura, en términos de conectividad, sin importar la rapidez con que sea enviada la información. Mientras que UDP, se utiliza para servicios que requieren una conectividad lo más rápida posible aunque no se asegure la confiabilidad de conexión y transmisión de información.

Por otra parte, la mejor manera de establecer una relación existente entre los métodos de transmisión *unicast* y *multicast* con los protocolos TCP y UDP, es definiendo los requerimientos que las diferentes aplicaciones necesitan en el proceso de su transporte o transmisión. Algunos ejemplos de aplicaciones como la conexión a servidores FTP, la conexión a un servidor Web a través de HTTP, el servicio de correo electrónico a través de SMTP o POP, son aplicaciones que habitualmente requieren una transmisión y conexión segura, por lo que es común que estén diseñadas para usar como protocolo de transporte a TCP, además de que no tendrían ningún problema de ser utilizadas en una red que trabajará con el modo *unicast*, ya que son aplicaciones que no utilizan una gran cantidad de ancho de banda y la conexión que se realiza para su uso es frecuentemente uno a uno.

Por otro lado, aplicaciones como audio y video *streaming*, telefonía IP y videojuegos en línea, son aplicaciones donde se considera más importante una transmisión o conexión rápida y no una conexión segura, por lo que están diseñadas para usar como protocolo de transporte a UDP; sin embargo, estas aplicaciones en su mayoría requieren gran cantidad de BW, lo que ocasiona que el método de transmisión *unicast* muchas veces resulte ineficiente, ya que recordemos que en éste se realiza una conexión entre una fuente y un único destinatario a la vez, mientras que usando el modo *multicast*, es posible lograr que las aplicaciones basadas en UDP sean más eficientes; debido a que la conexión que se realiza es entre una fuente con muchos destinatarios a la vez.

En conclusión, en *unicast* se puede hacer uso tanto de TCP como de UDP, sin embargo, la aplicación definirá que tipo de protocolo será el conveniente. Por el contrario, las aplicaciones y servicios que corren en *multicast*, en su mayoría, estarán diseñadas para funcionar únicamente a través del protocolo UDP [10].

II.1.2 Unicast a nivel IP (Capa 3)

Cuando el método de transmisión usado dentro de una red de datos es *unicast*, tanto el *host* fuente como el *host* destino requieren utilizar dos tipos de direcciones diferentes para reconocerse entre sí, y para ser reconocidos de forma individual en la red. La primera dirección se denomina dirección IP (dirección de interfaz lógica) y existen dos versiones en uso actualmente; las direcciones IP versión 4 (IPv4) y las direcciones IP versión 6 (IPv6). El presente trabajo va enfocado en la versión 4 de direcciones IP.

Las direcciones IPv4 son una secuencia de cuatro conjuntos de números en decimal separados por un punto entre sí, o cuatro grupos de 8 bits cada uno en su forma binaria equivalente. Un ejemplo se muestra en la figura II.1.13.

Dirección IPv4 en su representación decimal	192.	168.	1.	2
Dirección IPv4 en su representación binaria	11000000	10101000	00000001	00000010

Figura II.1.13. Ejemplo de una dirección IPv4 en su representación decimal y su equivalente representación en binario [Diagrama propio con base en la referencia 12].

Este tipo de direcciones se emplean en las redes de computadoras y por el protocolo IP para direccionar paquetes de datos creados en la capa 3 de *Internet* en el modelo TCP/IP, o en la capa de Red del modelo ISO/OSI.

Las direcciones IPv4 *unicast* se dividen en grupos o clases de direcciones, las direcciones de clase A, B y C, y en rangos ya sean privados o públicos. Además, cada una de las clases está asociada a una máscara de subred natural que tiene como función definir la porción de la red y las porciones para las direcciones IP de los *hosts*. Una dirección de clase A, B o C, puede ser identificada fácilmente si la representamos en su forma binaria; ya que para una clase A el primer bit (de izquierda a derecha; bit más significativo) del primer octeto siempre será 0, en una dirección clase B el primer bit siempre será 1 y el segundo bit será 0, y en una clase C el primer y segundo bit serán siempre 1 y un tercer bit será siempre 0. Todo lo descrito se representa en la tabla II.1 [11] [12] [13].

Clase	Rango de direcciones IPv4 públicas	Rango de direcciones IPv4 privadas	Clase por bits en el primer octeto	Mascara de red natural asociadas	Porción de red (CIDR) / número de subredes	Porción para direcciones IP de *host´s*
A	**1**.0.0.1- **126**.255.255.254	**10**.0.0.1- **10**.255.255.254	0	**255**.0.0.0	8 bits / 2^7 = 128 redes	24 bits equivale a 2^{24}-2= 16, 777, 214 IPs
B	**128.0**.0.1- **191.255**.255.254	**172.16**.0.1- **172.31**.255.254	10	**255.255**.0.0	16 bits / 2^{14} = 16, 384 redes	16 bits equivale a 2^{16}-2= 65, 534 IPs
C	**192.0.0**.1- **223.255.255**.254	**192.168.0**.1- **192.168.255**.254	110	**255.255.255**.0	24 bits / 2^{21} = 2, 097, 152 redes	8 bits equivale a 2^8-2= 254 IP's

Tabla II.1. La tabla muestra las clases, rangos públicas y privadas de las direcciones IPV4 *unica*st, la máscara natural asociada a cada clase y la porción de bits para red y direcciones IPs de *hosts* [Tabla con base en la referencia 11, 12 y 13].

II.1.3 Unicast a nivel MAC (Capa 2)

La segunda dirección que requiere un *host* fuente y un *host* destino para reconocerse entre ellos se llama dirección MAC *unicast* (dirección de interfaz física). Una dirección MAC, es un número conformado por seis conjuntos de números hexadecimales o su equivalente de 48 bits, cada uno de ellos separados por dos puntos entre sí como lo muestra la figura II.1.14. Las direcciones MAC son utilizadas como un número identificador único de las tarjetas o equipos de red físicos en las redes de computadoras. A diferencia de una dirección IPv4 que funciona en capa 3, una dirección MAC esta soportada sobre la capa 2 de enlace de datos, principalmente en redes de tipo LAN/MAN que usan tecnologías *ethernet*, 802.3 CSMA/CD, 802.11, entre otras.

Dirección MAC en su representación hexadecimal	80:	F6:	A0:	80:	B1:	C0
Dirección MAC en su representación binaria	10000000	11110110	10100000	10001010	10110001	11000000

Figura II.1.14. Ejemplo de una dirección MAC en su representación hexadecimal y su equivalente representación en binario [Diagrama propio con base en la referencia 15].

Los 48 bits que forman una dirección MAC se dividen en dos bloques; el primero de ellos con una longitud de 24 bits (primeros 24 bits o primeros 3 bloques en hexadecimal), corresponde al número del OUI (*Organitationally Unique Identifier* 'Identificador único de organización'), es decir, la organización o empresa que fabrica la tarjeta o equipo de red. El segundo bloque, también con una longitud de 24 bits, definen la dirección física de la tarjeta o equipo de red, tal y como se muestra en la figura II.1.15. Cada dirección MAC *unicast* es única, y la mayoría son asignadas y gestionadas por el IEEE *Standars Associations* [14] [15].

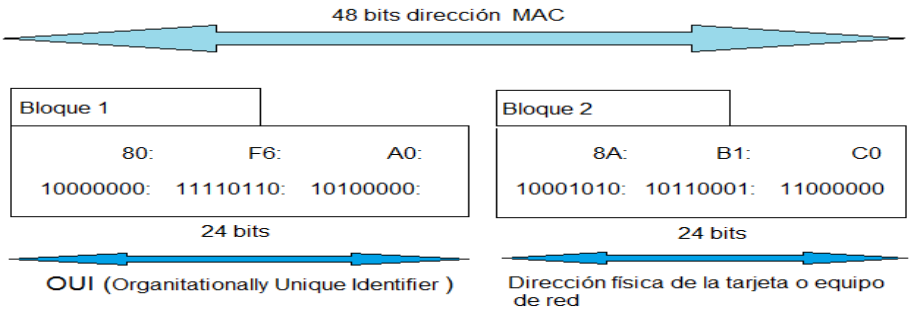


Figura II.1.15. Ejemplo de dirección MAC dividida en dos bloques; bloque 1 define al OUI y el bloque 2 representa la dirección física del equipo [Diagrama propio con base en la referencia 15].

De acuerdo a todo lo anterior, en una red *unicast* cada *host*, ya sea fuente o destino, dentro de una red tiene asignada de manera individual una dirección IPv4 (dirección lógica) asociada de igual forma con una dirección MAC (dirección física). Esta asociación entre una dirección lógica y una dirección física se realiza mediante el protocolo ARP (*Address Resolution Protocol*) que funciona entre la capa de enlace de datos y la capa de red [16].

II.2 Multicast

A diferencia de *unicast*, el método de transmisión *multicast* no es el más empleado para la transferencia de información en redes de tipo LAN, MAN o WAN, y ciertamente, no el más popular para ser empleado en la red de *Internet* comercial. Sin embargo, actualmente su popularidad ha ido en crecimiento gracias al surgimiento de la tecnología *IP multicast*, a la mejora de los protocolos que

soportan su funcionamiento, y a los beneficios que ofrece su implementación para la distribución de información multimedia en las redes de datos.

La definición de *multicast* es: transmisión en la que se envía únicamente una copia de datos desde una fuente emisora a múltiples (grupo) receptores desconocidos, como se muestra en la figura II.1.16 [17].

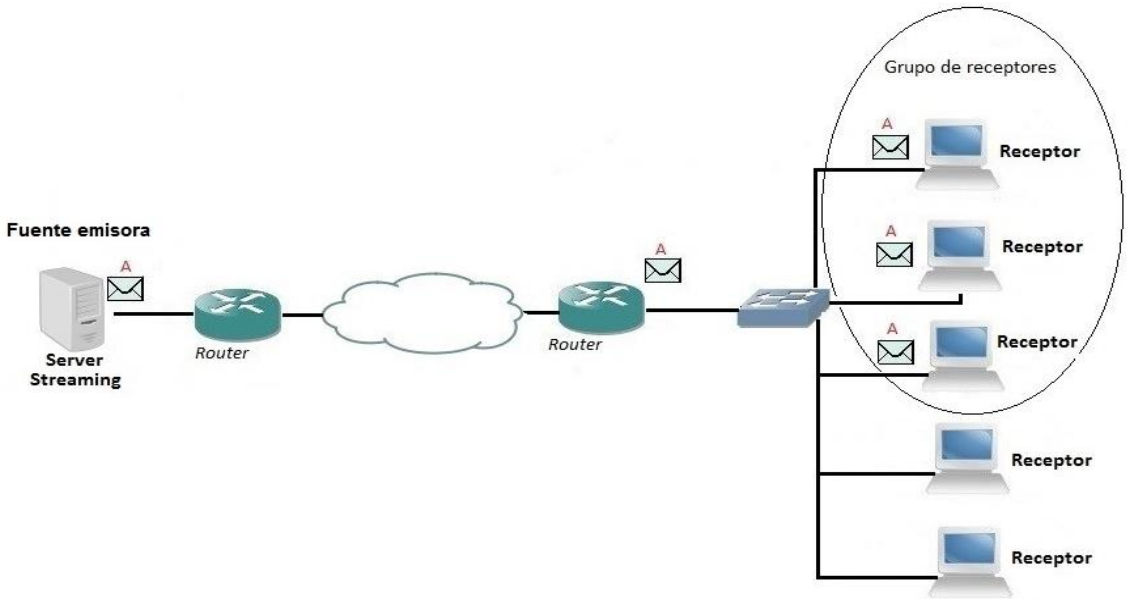


Figura II.1.16. Tipo de comunicación *multicast* de un mismo paquete de datos A desde un emisor a un grupo específico de receptores en una red [Diagrama propio con base en la referencia 17].

Los servicios más comunes que hacen uso de *multicast* son aquellos que contienen información multimedia, principalmente de audio y video que se transmite en tiempo real como audioconferencias y videoconferencias, y a través de la tecnología *streaming*.

Es importante destacar que *multicast*, y en consecuencia la tecnología *IP multicast*, se basan en el concepto de grupo. El concepto de **grupo *multicast*** o **grupo IP *multicast***, refiere al hecho de que un conjunto específico de receptores localizados en una red de datos desean recibir de manera simultánea un mismo tipo de información. A cada grupo *multicast*, para poder ser reconocido en una red, se le asigna una dirección IP de grupo *multicast*, y en una red de datos pueden existir uno o más grupos IP *multicast*.

A diferencia de una red *unicast*, donde el emisor envía un flujo de datos para cada receptor, en una red *multicast*, el emisor envía un único flujo de datos (paquete) para todo el grupo de usuarios que lo soliciten; es decir, el sistema emisor genera un único paquete de datos, por lo que el número de paquetes no será proporcional al número de equipos receptores que lo soliciten.

Este hecho, dará como resultados que la carga de procesamiento de datos que realice la fuente emisora (usualmente un servidor) sea la misma cuando un grupo de equipos soliciten simultáneamente un mismo flujo de información que dicha fuente emisora almacena, y que por lo tanto, su función de enviar la información adecuadamente no se vea afectada independientemente a la cantidad de equipos dentro de un grupo. Que la congestión en la red (número de paquetes que fluyen en la red,) donde se realiza el proceso de comunicación entre emisor y receptores, no sobrepase la capacidad de la red; y en consecuencia, que el consumo del ancho de banda BW disponible en la red sea mejor gestionado y utilizado, evitando así un colapso de la red. Las tres situaciones anteriores, son temas muy estudiados y sirven como elementos para justificar y realizar una comparación *unicast vs multicast* [18].

II.2.1 Multicast a nivel IP (Capa 3)

En una red de datos *multicast*, un *host* que actúa como fuente emisora, y el grupo de *hosts* que actúan como destinatarios, además de tener una dirección IP y una dirección MAC *unicast*, estos, necesitan tener una dirección *IP multicast* (dirección IP de grupo) en común para ser reconocidos como miembros de un grupo de multidifusión, así como una dirección MAC *multicast* asociada a esa dirección IP.

En *unicast*, las direcciones IPv4 de capa 3 se dividen en clases A, B y C. En *multicast*, las direcciones IPv4 se conocen como la clase D y contiene rangos de direcciones reservadas para ciertas funciones en específico, tal y como se muestra en la tabla II.2. La máscara de subred para dirección *IP multicast* reservadas está establecida por la IANA (ahora ICCAN) y será la asignada a un grupo de equipos (grupo *multicast*) y no a un equipo o *host* único [19] [20].

Clase	Direcciones IPv4 *multicast* reservada	Función	Porción de red	# de direcciones IPv4 de grupos *multicast*
D	**224**.0.0.1 /24	Todos los *hosts* en una subred.	**1110**	28 bits equivale a 2^{28}= 268, 435, 456 IP's
	224.0.0.2 /24	Todos los *routers multicast*		
	224.0.0.4 /24	Todos los *routers* configurados con el protocolo DVMR		
	224.0.0.5 /24	Todos los *routers* configurados con el protocolo OSPF		
	224.0.0.6 /24	Los *routers* DR-BDR designados por OSPF		
	224.0.0.9 /24	*Routers* que corran protocolo RIPv2		
	224.0.0.13 /24	Todos los *routers* configurados con el protocolo *multicast* PIMv2		
	224.0.1.39 /24	Equipos configurados con *CISCO-RP-ANNOUNCE* (auto-RP)		
	224.0.1.40 /24	Equipos configurados con *CISCO-RP-DISCOVERY* (auto-RP)		
	224.0.0.0-224.0.0.255 (**224**.0.0 /24)	Local Network Control Block (Bloque de direcciones de enlace de red local).		
	224.0.1.0-224.0.1.255 (**224**.0.1 /24)	Internetwork Control Block (Bloque de direcciones de enlace de local de red interna).		
	235.0.0.0-**238**.2555.255.255	Scoped Multicast Ranges (Reserved IANA) Rango de Multidifusión de ámbito (Reservado por IANA).		
	239.0.0.0-**239**.255.255.255	Scoped Multicast Ranges (Organization Local scope) Rango de Multidifusión de ámbito (Ámbito local para una organización).		

Tabla II.2. La tabla muestra algunas de las direcciones reservadas IPv4 de clase D, su función dentro de una red *multicast,* y la porción de bits para direcciones de grupo *multicast*. Para consulta de todas las direcciones IP multicast establecidas por la IANA (ahora ICANN), vaya a referencia 19 [Tabla con base en la referencia 19, 20 y 21].

En *multicast*, se asignan 28 bits para direcciones IPv4, por lo que existen $2^{\wedge 28}$= 268, 435, 200 IP's de multidifusión. Las direcciones clase D *multicast* representadas en forma binaria tienen como característica común que el primer, segundo y tercer bit del primer octeto (de izquierda a derecha) siempre será 1, y el cuarto bit del mismo octeto siempre será 0. Estos 4 bits del primer octeto (los de mayor peso) en conjunto con las combinaciones que se realizan con los 4 bits restantes del mismo octeto (los de menor peso), representan el direccionamiento *multicast* entre los valores 224 al 239 en decimal como lo muestra la tabla II.3. Los 28 bits restantes de una dirección *IP multicast* representan a los grupos de multidifusión en una red [21].

Primer octeto en decimal que representan el conjunto de direcciones clase D multicast.	Primer octeto en binario que representan el conjunto de direcciones clase D multicast.
224. X. X. X	1110 0000 . X. X. X
225. X. X. X	1110 0001 . X. X. X
226. X. X. X	1110 0010 . X. X. X
227. X. X. X	1110 0011 . X. X. X
228. X. X. X	1110 0100 . X. X. X
229. X. X. X	1110 0101 . X. X. X
230. X. X. X	1110 0110 . X. X. X
231. X. X. X	1110 0111 . X. X. X
232. X. X. X	1110 1000 . X. X. X
233. X. X. X	1110 1001 . X. X. X
234. X. X. X	1110 1010 . X. X. X
235. X. X. X	1110 1011 . X. X. X
236. X. X. X	1110 1100 . X. X. X
237. X. X. X	1110 1101 . X. X. X
238. X. X. X	1110 1110 . X. X. X
239. X. X. X	1110 1111 . X. X. X

Tabla II.3. Representación del primer octeto de una dirección IP *multicast*; los primeros 4 bits son fijos y unidos a las combinaciones de los 4 bits restantes se logra una dirección *multicast* entre el 224 y 239 decimal [Diagrama propio].

II.2.2 Multicast a nivel MAC (Capa 2)

En *unicast* cada *host* fuente o destino tiene una dirección IPv4 asignada, así como una dirección MAC que se asocian con ayuda del protocolo ARP. En *multicast*, se asocia una dirección MAC por cada dirección IPv4 correspondiente a un grupo *multicast*. La asociación se realiza mediante un proceso que se conoce como mapeo de *IP multicast* (IPCM) sobre direcciones de MAC físicas (a nivel de capa 2), se realiza principalmente en redes *ethernet* con ayuda de un protocolo llamado IGMP (RFC 1112) [22].

Supongamos una dirección IPv4 *multicast* en su representación binaria de 32 bits como la que se muestra en la figura II.1.17. Del bit 31 al bit 24 del primer octeto, se representa el direccionamiento de clase D entre el rango decimal 224 a 239.

Figura II.1.17 Representación de una dirección IPv4 *multicast* de 32 bits [Con base en referencias 23 y 24].

Ahora supongamos una dirección MAC *multicast* de 48 bits como se muestra en la figura II.1.18. Del bit 47 al bit 24 se define la dirección OUI 01:00:5e en forma hexadecimal, y representará siempre de forma preestablecida direcciones MAC *ethernet* de tipo *multicast*.

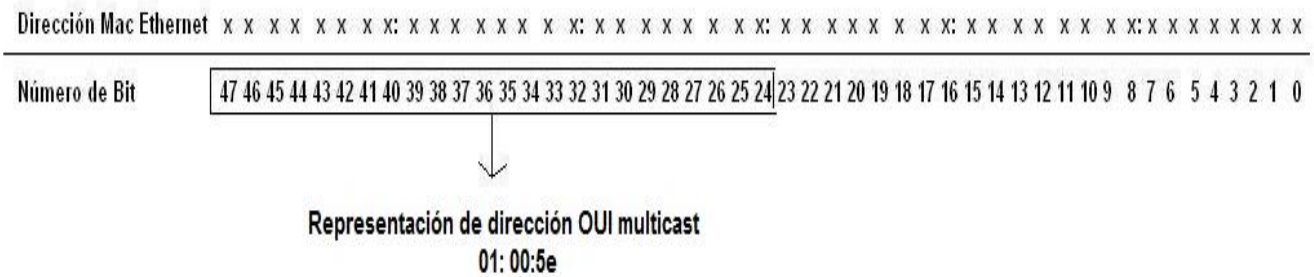


Figura II.1.18. Representación de una dirección MAC *ethernet multicast* de 48 bits [Con base en referencias 23 y 24].

El bit 23 de la dirección *IPv4 multicast* se mapea al bit 23 de la MAC siempre con un valor de 0 y no se toma en cuenta en el mapeo. Los 23 bits restantes (del 22 al 0) de la IP se mapean a los 23 bits restantes (del 22 al 0) de la MAC haciendo la conversión de binario (dirección IP en binario) a hexadecimal. Por ejemplo, si hacemos el mapeo de la dirección clase D 224.0.0.5 a su correspondiente MAC, el resultado sería 01:00:5e:00:00:05, tal y como se muestra en la figura II.1.19 [23] [24].

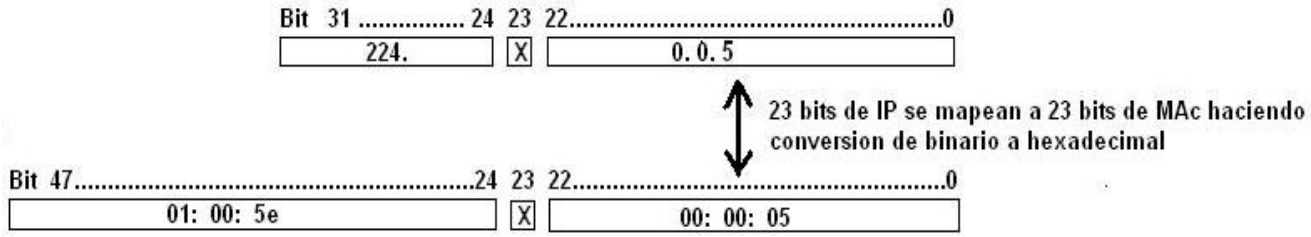


Figura II.1.19. Mapeo de una dirección IP de grupo *multicast* a una dirección MAC de grupo *multicast.*

II.3 Unicast vs Multicast

Un vez que se ha dado el contexto general de comunicación *unicast* y *multicast*, ahora es posible hacer una comparativa entre los dos tipos de comunicación. Esta comparación se hace necesaria cuando se requiere que una misma información se envié simultáneamente a un conjunto específico de receptores en una red. En este sentido, dentro de las propiedades de funcionamiento comunes de ambos tipos de comunicación existen diferencias que son visibles. Identificar estas diferencias es útil para decidir cuál de los métodos utilizar para el envío de información, y en particular, cuando el tipo de información que se desea transmitir dentro de una red corporativa, e incluso en una red que pase por *Internet*, es de tipo multimedia en formato de voz, audio y video.

La tabla II.4 aporta una inspección detallada de las diferencias entre los métodos *unicast* y *multicast,* que posiblemente no es fácil encontrar de manera tan explícita en documentos o libros que hablen de este tema. Por otra parte, si bien resulta importante conocer cada uno de los nueve aspectos detallados en la tabla, es indispensable destacar los puntos del 5 al 9, ya que estos últimos llevan una relación directa entre sí, y además, con base en ellos se han generado diversos documentos de estudio relacionados a la multidifusión.

	Propiedades	**Unicast**	**Multicast**
1	Tipo de Comunicación (Por definición)	Uno a uno (*host to host*) (Un solo emisor – con un solo receptor)	Uno a varios (Un emisor – con un grupo de receptores)
2	Rango de direcciones que emplea (Por definición)	Clase A: 0.0.0.1-127.255.255.254 B: 128.0.0..0-191.255.255.254 C: 192.0.0.0-223.255.255.254	Clase D: 224.0.0.0-239.255.255.255
3	Protocolo de comunicación empleado en capa de transporte (L4) (Por definición)	TCP (Orientado a conexión) o UDP	UDP (No orientado a conexión)
4	Control de flujo de datos y confiabilidad de entrega	Sí se asegura (TCP)	No se asegura (UDP)
5	Consumo de BW (*bandwidth*-ancho de banda)	Mayor consumo del BW en función del # de receptores. (Como ejemplo: la transmisión de un audio a 56Kbps requerirá restarle 56Kbps al BW disponible en una red para cada usuario que solicite dicha transmisión)	Menor consumo del BW, no importa el # de receptores. (Como ejemplo: la transmisión de un audio a 56Kbps requerirá restarle sólo 56Kbps al BW disponible en una red para todos los receptores que soliciten la transmisión)
6	Número de paquetes de datos que envía el emisor	Un paquete por cada receptor (100 receptores $\Rightarrow$ se requieren 100 paquete de datos)	Un paquete para el grupo de receptores (100 receptores $\Rightarrow$ se requiere 1 paquete de datos)
7	Congestión en la red (# de paquetes)	Aumenta en todo el trayecto de red. $P_T = P \times N_R$ donde, P_T: Paquetes Transmitidos P: Paquetes N_R: Número de Receptores	Se reduce desde el emisor hasta el último *router* en la red. $P_T = P \times 1$
8	Carga de procesamiento de los equipos de red (servidores y *routers*)	Aumenta entre más receptores exista $C_E = C \times N_R$ donde, C_E: Carga en el equipo de red C: Carga N_R: Número de Receptores	Se mantiene sin importar el número de receptores $C_E = C \times 1$
9	Número de receptores escalable	En función de la capacidad del equipo emisor (usualmente un servidor).	Sí, en función de la capacidad del emisor y *router de último salto* que se comunica con receptores.

Tabla II.4. Diferencias de las propiedades comunes de una comunicación *unicast* frente a una comunicación *multicas*t

[Tabla propia].

El punto 5 hace referencia al consumo de ancho de banda BW disponible en una red de datos, si bien en la tabla se puntualiza el ahorro considerable que se obtiene haciendo uso de una difusión *multicast* en comparación con una difusión *unicast*, el tema es cómo hacer una medición del consumo de acho de banda. Bueno, pues la respuesta está en el término tasa de bits o tasa de transferencia, ya que este término define el número de bits que se transmiten por unidad de tiempo a través de dos sistemas digitales, es decir, la tasa de transferencia se refiere al ancho de banda real medido en un momento concreto mientras se transmite un flujo específico de datos. Por lo tanto, en *multicast*, el ancho de banda referido como la tasa de transferencia de datos se mantiene estable (sin aumento), debido a que se transmite un único paquete de datos a un grupo de receptores (punto 6 de la tabla) y no es necesario el envío de dicho paquete para cada uno de los receptores que existan en la red como en el caso de *unicast*.

La congestión en la red, punto 7 de la tabla, que se mediría con las variables tanto del consumo de ancho de banda en una momento determinado y el escalamiento en aumento de receptores en la red, es decir, del número de receptores (punto 9 de la tabla) partícipes en la recepción de datos *multicast*, que será mayor mientras más receptores existan para el caso *unicast.* En caso contrario, para una red *multicast* la congestión en dicha red se mantendrá estable debido a que la cantidad de flujo es único, sin aumento, y el número de receptores no influirán para que la congestión en la red se incremente.

Hasta aquí, se realizó una revisión y comparación de los métodos de transmisión *unicast* y *multicast*.

II.4 Funcionamiento y características de la tecnología IP Multicast

Si un grupo o grupos específicos de *hosts* que sean miembros de una red LAN/MAN e incluso WAN solicitan simultáneamente el envío de datos en forma de texto, imágenes, y principalmente de audio y video, la tecnología que mejor respondería a esa solicitud sería la tecnología *IP multicast*. *IP multicast* está basada en el método de transmisión *multicast*, la cual es utilizada para ofrecer servicios y aplicaciones que usualmente requieren un ancho de banda grande.

El funcionamiento de *IP multicast* se realiza mediante el envío de paquetes por parte de un *host* fuente a través de una topología de red a un grupo de *hosts* receptores que pertenecen a un grupo *multicast* con una dirección *IP multicast* común.

IP multicast combina el funcionamiento de los métodos de transmisión *unicast* y *multicast*, y para que en una red de datos se pueda implementar *IP multicast*, es indispensable tener equipos que soporten dicha tecnología. Los equipos indispensables son:

- Equipo emisor. El equipo emisor o fuente emisora, generalmente es un equipo de cómputo o un equipo que actúa como servidor con la capacidad suficiente para almacenar información, y tiene como función principal proveer la información que almacena a los miembros de un grupo *multicast*.
- Equipos intermedios. Los equipos intermedios *routers multicast* (*mrouters*), son muy importantes en la topología de red; ya que se usan como equipos para interconectar dos o más redes separadas geográficamente así como de hacer copias de la información (tráfico *multicast*) hacia los grupos IP *multicast.*
- Equipo destino. Los equipos destinos (receptores) deben pertenecer a un grupo *multicast*, y son los que reciben tráfico *multicast* en forma de texto, audio y video.

El equipo emisor, generalmente, debe tener instalado algún *software* o aplicación capaz de distribuir y procesar información multimedia vía *multicast*. Los equipos receptores deben tener instalado, de igual manera, un *software* o aplicación compatible con la del equipo emisor, capaz de recibir y procesar tráfico de tipo multimedia vía *multicast*. Por último, los equipos emisor, intermedios y receptores, deben tener la capacidad de soportar protocolos *multicast* a nivel *hardware* (equipo físico) y a nivel de *software* (sistema operativo).

Con la finalidad de dar una visión muy general del funcionamiento de IP *multicast,* supongamos que un equipo fuente envía datos a un grupo *multicast* dentro de una red de computadoras. Los *mrouters*, que corren protocolos de enrutamiento *multicast*, se encargarán de realizar copias de los datos en una cantidad igual al número de interfaces físicas donde existan posibles equipos receptores conectados y que pertenezcan a un grupo *multicast*. Los receptores que pertenecen a ese grupo, expresarán el interés a los *mrouters* por recibir tráfico, así como su pertenencia al grupo gracias a que corren protocolos *multicast*.

Para entender mejor el proceso de copiado que realizan los *mrouters*, gracias a los protocolos que corren en sus interfaces, supongamos que un servidor *streaming* transmite un paquete de datos A y está conectado directamente con un *mrouter1*. Este *mrouter1*, conectado directamente al servidor, se encargará de duplicar el paquete original y enviarlo a las dos interfaces de salida donde existen conectados un *mrouter2* y un *mrouter3*, quienes finalmente tendrán la tarea de construir las copias necesarias de paquetes A y enviarlas por las interfaces de salida a las que estén conectados los receptores que forman parte de los grupos *multicast* 1, 2, 3 que hayan solicitado el mismo paquete de datos, tal y como se muestra en la figura II.1.20 [25].

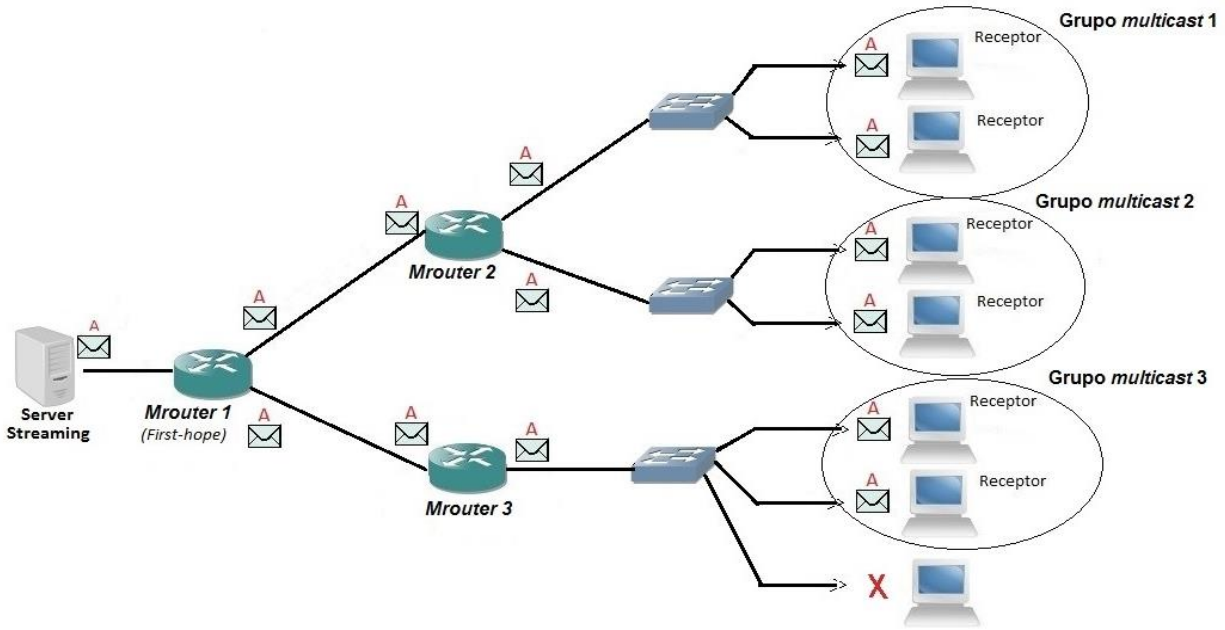


Figura II.1.20. Proceso de replicación de un paquete de datos A por los *routers multicast* que forman parte de una topología de red *multicast* [Diagrama propio con base en referencia 28].

II.4.1 Revisión histórica de la tecnología IP Multicast

En la década de 1980 los métodos de transmisión *unicast* y *multicast* ya eran conocidos, el primero de ellos ciertamente con una ventaja muy superior sobre el segundo en desarrollo y aplicación. En esa misma década, surge una primera aplicación práctica de *IP multicast* llamada *vSystem*; un sistema operativo que tuvo como objetivo distribuir datos a un grupo de computadoras localizadas en una red local *ethernet*, de modo tal, que una sola computadora fuera capaz de enviar información de manera simultánea a un grupo de computadoras localizadas dentro de la misma red. Al tratarse de una red local, la comunicación entre emisor y el grupo de receptores se veía limitada a ser funcional sólo en capa 2 del modelo de referencia ISO/OSI, lo que ocasionó que al tratar de extender el sistema de multiprocesamiento y el envío de información hacia computadoras que se encontraban en una red en otro sitio del campus no fuera posible; a menos que se lograra trasladar el proceso de comunicación a un nivel o capa superior. Para lograr lo anterior, *Steve Deering* se dio a la tarea de estudiar el reciente protocolo de enrutamiento dinámico RIP (*Routing Information Protocol – Protocolo de Información de Enrutamiento*) creado en 1982.

Tras varias investigaciones, *Deering* llegó a la hipótesis de que las bases teóricas y los mecanismos de funcionamiento de RIP se podrían utilizar para desarrollar un protocolo de enrutamiento y así poder extender la comunicación *multicast* al nivel IP en capa 3 del modelo OSI. La hipótesis propuesta por *Deering* tuvo su conclusión con la creación de dos protocolos de enrutamiento *multicast;* el primero de ellos DVMRP (*Distance Vector Multicast Routing Protocol*) descrito en el RFC 1075 en Noviembre de 1988, y el segundo, nombrado IGMPv1 (*Internet Group Management Protocol version 1*) publicación en el RFC 1112 en Agosto de 1989. Para Diciembre de 1991 el problema de transmitir mensajes *multicast* en dos redes locales *ethernet* separadas geográficamente quedo resuelto y ese mismo año se publicó la tesis doctoral de *Deering* titulada *"Multicast Routing in a Datagram Network",* sentando las bases para posteriores investigaciones que dieron origen al desarrollo de nuevos protocolos de enrutamiento *multicast* como; IGMPv2, MOSPF (*Multicast Extension to OSPF*), así como a protocolos con funciones más específicas utilizados en la actualidad sobre la tecnología IP m*ulticast* como son; IGMP *Snooping*, PIM-DM (*Protocol Independent Multicast-Dense Mode*) o PIM-SM (*Protocol Independent Multicast- Sparse Mode*) [26] [27].

A continuación, se incluye una línea del tiempo en la cual se muestran las fechas y los principales acontecimientos que sentaron las bases para el desarrollo de la tecnología *IP Multicast.*

Figura II.1.21. Línea del tiempo que describe y resume los principales acontecimientos que dieron origen a la tecnología IP *Multicast* [Diagrama propio].

II.4.2 Aplicaciones y modelos de transmisión IP Multicast

La cantidad de aplicaciones que han surgido a lo largo de las últimas dos décadas y que pueden hacer uso de esta tecnología es muy amplia. Básicamente, estas aplicaciones se pueden agrupar en dos bloques; el bloque de aplicaciones que transmiten datos en tiempo real y el bloque de aquellas aplicaciones que toman información almacenada en algún lugar y la transmiten cuando se requiere, es decir, bajo demanda.

Servicios multimedia de audio y video *streaming*, la difusión de datos en forma de texto, la visualización de televisión digital sobre IP (IPTV), las videoconferencias y audioconferencias, simulaciones, emulaciones o procesos interactivos (como el uso de pizarras electrónicas) en grupo, los servicios de cómputo distribuido y uso de videojuegos en línea pertenecen al primer bloque de aplicaciones en tiempo real. Mientras que servicios multimedia AoD (*Audio on Demand* 'Audio Bajo demanda') y VoD (*Video on Demand* 'Video Bajo demanda') y la transferencia de archivos y la replicación de datos cada cierto tiempo pertenecen al segundo bloque de aplicaciones bajo demanda.

Actualmente el uso de *IP multicast* para las aplicaciones mencionadas con anterioridad repercute en hacer eficientes los modelos de negocio en compañías así como en mejorar el trabajo en grupo de instituciones que cuentan con redes locales privadas, y que requieren ejecutar audioconferencias, videoconferencias, transmitir datos en tiempo real, o simplemente enviar información a un determinado grupo de usuarios finales. Asimismo, empresas dedicadas a ofrecer servicios de *Internet* ISP's (*Internet Service Provider*) así como servicios *Tripleplay* (voz, "Internet de banda ancha" y televisión), cada vez más, ven a esta tecnología como una solución para mejorar los servicios que ofrecen a sus usuarios y al mismo tiempo mejorar el desempeño de su infraestructura de red [28].

Hasta ahora se ha mencionado el modelo de transmisión donde existe un único equipo emisor, responsable de transmitir algún tipo de información a un grupo o grupos de receptores para explicar *multicast* y la tecnología *IP multicast*. Este modelo de transmisión se denomina de **uno a muchos**, y ciertamente, es el más conocido y el más empleado en las redes de datos *IP multicast*. Sin embargo, existen otros modelos de transmisión *IP multicast* tales como el de **muchos a muchos** y el de **muchos a uno**. Cada uno de los modelos se diferencia entre sí por la función que desempeñan los equipos dentro de la red de multidifusión y por las aplicaciones que se adaptan a cada tipo de modelo. Por ejemplo, en el modelo muchos a muchos dos o más equipos receptores tienen la capacidad de actuar también como equipos emisores, mientras que en un modelo uno a muchos, los equipos receptores sólo reciben el tráfico *multicast* y no son capaces de transmitirlo [29].

En el modelo **uno a muchos**, una fuente emisora envía a múltiples (grupo) receptores. Suele usarse en aplicaciones para la distribución de audio y video *streaming*, la difusión de datos en forma de texto en tiempo real, la visualización de televisión digital sobre IP, etc.

En el modelo **muchos a muchos**, representado en la figura II.1.22, dos o más equipos receptores además de recibir tráfico *multicast* pueden transmitirlo de igual forma, es decir, los receptores también actúan como emisores. Este tipo de método de transmisión requiere un manejo más complejo que el método de uno a muchos tanto de sincronización del tráfico *multicast*.

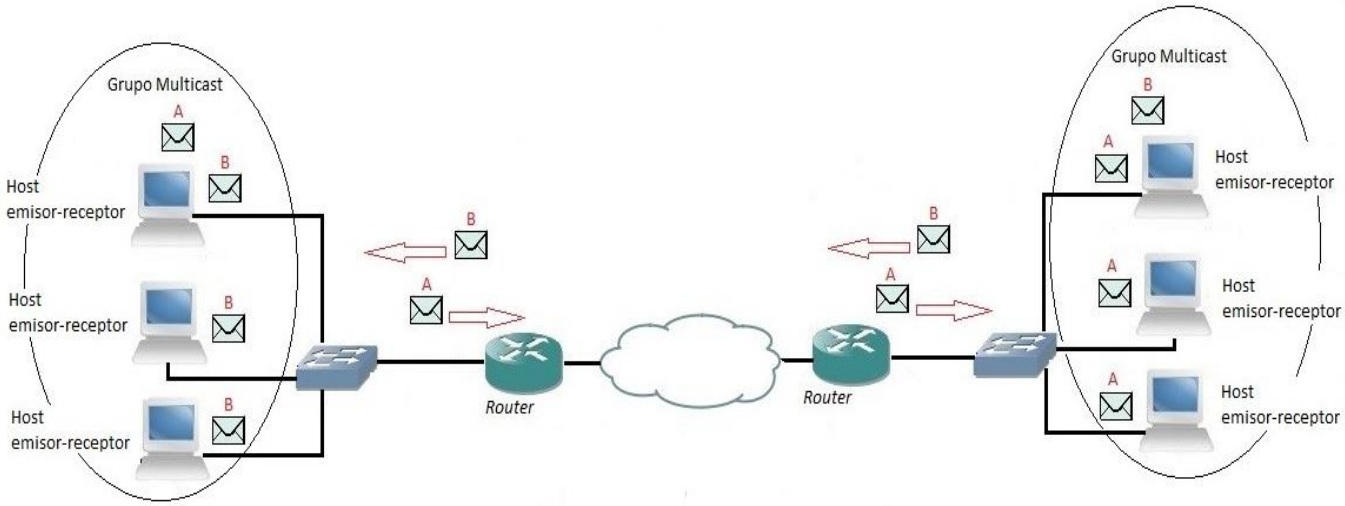


Figura II.1.22. Modelo de aplicación mucho a muchos. Varios sistemas que actúan como fuente pueden actuar también como receptores, de manera inversa varios sistemas receptores en grupos *multicast* tienen la capacidad de actuar como fuente [Diagrama propio con base en la referencia 36].

Por último, **el modelo muchos a uno**, representado en la figura II.1.23 es menos utilizado debido al poco desarrollo que ha tenido y en teoría se podría utilizar para que muchos equipos emisores enviaran recursos y datos a un mismo destino y disponer de ellos en el momento que lo desearán.

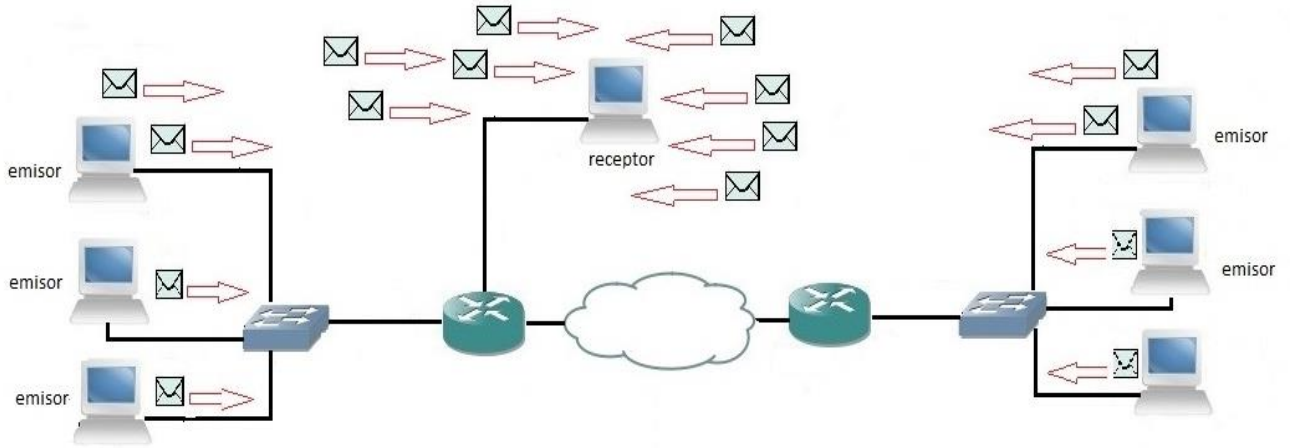


Figura II.1.23. Modelo de aplicación mucho a uno. Varios sistemas que actúan como fuente; envían tráfico a un único destino [Diagrama propio con base en la referencia 36].

II.4.3 Escenarios IP Multicast: Intranet, Internet y MBone

IP multicast es un modelo tecnológico que a lo largo de 20 años ha tenido avances significativos con la creación de diferentes protocolos *multicast* y con el surgimiento de tecnologías que se desprenden de la misma técnica *multicast* tales como, *Multicast* VPN, *Multicast Label Distribution Protocol* (MLDP), *Multicast Session Description Protocol* (MSDP), *Multiprotocol Label Switching* (MPLS), IPv6 *Multicast*, y *Wireless Multicast*.

Actualmente, la implementación y uso de IP *multicast* junto con las diferentes tecnologías que se desprenden de la misma, dentro de las redes de datos internas pertenecientes a corporaciones, organizaciones e instituciones privadas y públicas, es posible gracias a la existencia de diversos protocolos de comunicación y al surgimiento de diversas tecnologías.

Importante es destacar, particularmente, la implementación y uso de *IP multicast* en las *Intranets* de distintas redes avanzadas del mundo. Debido a que estas últimas, sirven como plataforma de prueba e investigación tanto de la propia tecnología *IP multicast*, así como de muchas otras; sólo por mencionar algunas: IPv6, IPv6 *multicast*, VPN y QoS.

CUDI (Corporación Universitaria para el Desarrollo de Internet A.C) y la Red CLARA (Cooperación Latino Americana de Redes Avanzadas), son ejemplos muy claros de redes avanzadas que incorporan dentro de sus servicios la tecnología *IP multicast*. La cual les proporciona, por una parte, la posibilidad de experimentar con dicha tecnología, y por otra parte, la mejora en el uso de la capacidad de su red para aplicaciones donde se requiere la transmisión de audio y video (audioconferencias y videoconferencias), así como para el funcionamiento de aplicaciones como Opera Oberta (Tecnología *Multicast* aplicada a las artes escénicas) [30].

Cuando hablamos de implementar *IP multicast* en la red de *Internet* los requerimientos se hacen mayores, debido a que el *backbone* de *Internet* (o columna vertebral de *Internet*) está en gran parte constituida por la infraestructura perteneciente a compañías alrededor del mundo que ofrecen servicios de conectividad, los ISP. Estos, en su mayoría, hacen uso del modo de transmisión *unicast* con el fin de tener un mejor control y monitoreo de tráfico que genera cada uno de sus usuarios, y por tanto, restringen la transmisión de tráfico *multicast* para los mismos; por otro lado, no todos los ISP del *backbone* están dispuestos a invertir en *hardware* y *software* con capacidad para gestionar tráfico *multicast*. Estos dos aspectos, así como la carencia de uniformidad de un ancho de banda en toda la red, responden al por qué una infraestructura que soporte servicios *IP multicast* en *Internet* sea casi inexistente [31].

Ya que se carece de infraestructura para el flujo de tráfico *multicast* en *Internet*; en 1992 Steve *Deering* y *Steve Casner* crean la red *Mbone* (*Multicast Backbone*). Una red de datos compuesta de islas en distintas zonas geográficas, de los EE.UU principalmente, interconectadas entre sí mediante túneles DVMRP independientes o superpuestas sobre la red de *Internet*.

El objetivo de la creación de *Mbone*, es lograr la conectividad de algún sitio específico a cualquier país del mundo dentro de la red de *Internet*, como una Universidad o centro de investigación mediante la creación de los propios túneles DVMRP y enlaces dedicados, de tal manera, que el tráfico *multicast* generado en la *Mbone* pudiera ser encapsulado en tramas *unicast* TCP/IP y pudiera llegar al lugar específico dentro de la red de *Internet*, tal y como si se tratase de tráfico de datos *unicast* [32].

II.5 Tecnología Streaming

Hoy en día la utilización de aplicaciones y servicios multimedia en sus diferentes presentaciones como son texto, imágenes, animaciones, gráficos, audio y video en las redes de datos internas de alguna empresa u organización, así como por los usuarios de *Internet* en todo el mundo, se ha convertido en una práctica muy común y necesaria. En este sentido, se vuelve indispensable destacar la tecnología *streaming*; tecnología útil para la transmisión y reproducción de información multimedia, principalmente de audio y video a través de las redes de datos.

Streaming surge en 1995 (mismo año en el que surgió *Internet* comercial) como una tecnología que permitiera al usuario final reproducir de forma continua flujo de datos en forma de audio y video sin la necesidad de descargarlos previamente a su equipo de cómputo. Es decir, con *streaming* un usuario final podrá escuchar y visualizar un archivo de audio y video conforme este se vaya descargando del equipo donde se encuentre almacenado, sin la necesidad de esperar que el archivo se descargue por completo a su equipo, tal y como sucedía antes de 1995 [33].

La tecnología *streaming* destaca como una tecnología que permite y facilita la emisión y reproducción de eventos musicales, audio-conferencias y video-conferencias en vivo (llamado *streaming* en directo); de igual forma, permite la transmisión y reproducción de archivos de audio y video almacenados en PC´s o servidores (llamado *streaming* bajo demanda), ya sea a través de *Internet*, o dentro de las *Intranets* empresariales o de alguna organización. Sin embargo, el uso de esta tecnología acarrea principalmente dos desventajas que deben ser consideradas a la hora de su utilización. La primera desventaja tiene que ver con la necesidad de un ancho de banda suficiente como para que la tecnología pueda correr y funcionar de forma aceptable dentro de una red de datos. La segunda desventaja salta a la vista con el hecho de que se requiere forzosamente que el equipo emisor

y los equipos que actúan como receptores, tengan instalada una aplicación o *software* compatible entre los dos mencionados, capaz de soportar la tecnología *streaming* y con la capacidad de transmitir diversos estándares y formatos de codificación tanto de audio y video. En este sentido, es pertinente adelantar que la utilización de la aplicación VLC *media player* será de gran ayuda en el modelo de red *IPv4 multicast* propuesto en este trabajo ya que es un *software* de libre descarga y bastante completo que permite la transmisión y reproducción de audio y video *streaming multicast*, soportando además, diversos formatos de audio y video.

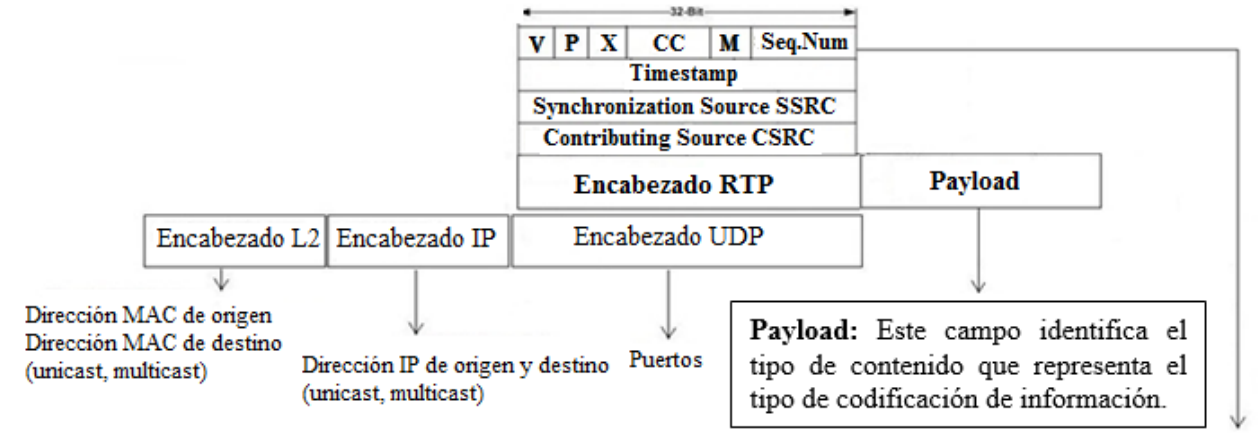
Los principales protocolos involucrados para la transmisión y reproducción de audio y video en una *Intranet* con la tecnología *streaming* son UDP, HTTP, RTP, RTCP y RTSP; protocolos que soporta VLC para la transmisión y reproducción de audio y video en una red interna [34].

II.5.1 Protocolo RTP/RTCP

En el apartado II.1.1.2 se mencionó que el protocolo UDP es preferentemente el protocolo usado para el transporte de audio y video *streaming* o en tiempo real; sin embargo, también se mencionó que UDP carece de un mecanismo de control de flujo de datagramas y que no asegura la entrega de los mismos. En este sentido, el protocolo RTP (*Real Time Protocol*), es un protocolo diseñado para complementar las carencias de transporte que presenta UDP, y de alguna manera, resolver la falta de un mecanismo de control y orden de flujo de este último. Es decir, RTP, es el protocolo que proporciona los mecanismos necesarios para que una transmisión de audio y video en tiempo real llegue a su destino en orden, tal y como es enviada, y para ayudar a identificar la pérdida de información en algún momento de la transmisión. Estos mecanismos, son proporcionados por dos campos; *Sequence Number* y *Time Stamp*, que forman parte del encabezado RTP; el cual se describe más adelante.

RTP también se utiliza, aunque con menor frecuencia, en transmisiones basadas en TCP; ya que este último tiene un mecanismo propio que asegura una conexión y transmisión confiable de flujo de datos. Por otro lado, RTP funciona indistintamente tanto en redes de tipo *unicast* así como *multicast*, y es un protocolo que por defecto utiliza el puerto 5004 para transmitir. En la figura II.1.24 se muestra el contenido del paquete y encabezado RTP, y se da una breve explicación del contenido de los mismo [35].

Junto a RTP se integra un protocolo de control llamado RTCP (*Real Time Control Protocol*), el cual hace uso del puerto 5005 para transmitir y su función es controlar el flujo de paquetes RTP que se realiza entre emisor-receptor en *unicast*, así como entre emisor y un grupo de receptores en *multicast* durante una transmisión de audio y video.



V (Versión): Este campo indica el número de versión RTP utilizada. La versión actual es la versión 2.

CC (CSRC Count): Contiene el número de identificadores CSRC que siguen la cabecera fija.

M (Marker): Se trata de un bit de señalización. Su significado depende de los datos transportados.

Sequence Number: Este campo permite detectar paquetes RTP perdidos. El valor de este campo se incrementa en 1 por cada paquete RTP enviado.

Timestamp: Son marcas temporales que utiliza el protocolo RTP como información base para los cálculos que permiten evaluar los retardos introducidos por un sistema de comunicación.

Synchronization Source (SSRC): Este campo identifica la fuente que ha producido el paquete.

Contributing Source (CSRC): De 0 a 15 instancias de este campo pueden estar presentes en la cabecera del paquete RTP.

Figura II.1.24. Se muestra el encabezado RTP y su contenido por campo [Diagrama propio con base en referencia 35].

Capítulo III

Protocolos de enrutamiento

En el presente capítulo se ofrece un resumen de los protocolos de enrutamiento dinámico *unicast* IGP's (*Internal Gateway Protocols*) y EGP's (*External Gateway Protocols*); pero se profundiza en la teoría y funcionamiento del protocolo de enrutamiento dinámico *unicast* OSPF (*Open Shortest Path First Protocol*). Posteriormente, se hace una breve revisión de los protocolos de enrutamiento *multicast Dense*, y *Sparse Mode*. En seguida, se hace una revisión más extensa del protocolo de red IGMPv2 (*Internet Group Management Protocol Version 2*), así como del protocolo de modo esparcido PIM-SM (*Protocol Independent Multicast - Sparse Mode*). Se profundiza en la teoría y funcionamiento del protocolo *unicast* OSPF, así como en la de los protocolos *multicast* IGMPv2 y PIM-SM, debido a que estos nos serán útiles para la configuración de los equipos virtuales (*routers*) del emulador GNS3 en el modelo *IPv4 multicast* propuesto en el capítulo IV del presente trabajo.

III.1 Protocolos de enrutamiento dinámico unicast

La ruta que toma algún tipo de información para llegar a su destino final, dentro de una red de datos que funciona en modo *unicast*, en general, está definida por el tipo de protocolo de enrutamiento dinámico que corre en los *routers* que forman parte de la topología de red. En este sentido, los protocolos de enrutamiento dinámico *unicast* realizan una función de ruteo para construir tablas de enrutamiento (direccionamiento *unicast*) y tienen definida una métrica con la cual los protocolos determinan cuál es la mejor ruta para que los datos lleguen a un destino. Además, los protocolos *unicast* tienen como funciones específicas:

- ✓ Compartir información de forma dinámica entre *routers*.
- ✓ Actualizar las tablas de enrutamiento de forma automática cuando cambia la topología de red.

Así mismo, tienen como objetivos principales:

- ✓ Determinar cuál es la mejor ruta a un destino.
- ✓ Descubrir redes remotas.
- ✓ Mantener la información de enrutamiento actualizada.
- ✓ Brindar la funcionalidad necesaria para que los datos encuentren una mejor ruta si la actual deja de estar disponible.

Para cumplir dichas funciones y objetivos, los protocolos de enrutamiento dinámico usan una serie de mensajes, por ejemplo, mensajes *update* (mensajes de actualización), así como algoritmos matemáticos que facilitan la información de enrutamiento y definen las características funcionales de los protocolos.

Existen dos tipos de protocolos dinámicos *unicast*, los protocolos IGP y los protocolos EGP. Los IGP se clasifican de acuerdo a si son protocolos de vector distancia o protocolos de estado de enlace. Y en los EGP sólo se considera una clasificación de protocolo denominada de vector ruta.

Los protocolos de vector distancia se caracterizan por el hecho de que los *routers* no construyen una tabla de ruteo que contenga toda la topología de la red, sino que estos últimos se basan en la información proporcionada por sus vecinos conectados directamente para tener una visión de la topología de red. RIP (*Routing Information Protocol* 'Protocolo de información de enrutamiento') y EIGRP *Enhanced Interior Gateway Routing Protocol* 'Protocolo de enrutamiento de puerta de enlace interior mejorado'*)* son protocolos de vector distancia.

Por otra parte, los protocolos de estado de enlace ofrecen una vista completa de la topología de red a cada uno de los *routers*, gracias a que estos crean un paquete de estado de enlace que incluye información sobre todos los *routers* en una red; y no se basan sólo en la información proporcionada por sus vecinos conectados directamente, tal y como ocurre en los protocolos de vector distancia. Los protocolos OSPF e IS-IS (*Intermediate System to Intermediate System Protocol* 'Protocolo de sistema intermedio a sistema intermedio'), son protocolos de estado de enlace.

Finalmente, el único protocolo considerado de vector ruta es el protocolo BGP (*Border Gateway Protocol* 'Protocolo de puerta de enlace de borde'); generalmente se usa en *routers* de frontera, es decir, aquellos *routers* que interconectan a diferentes AS (*Autonomous Systems* 'Sistemas autónomos'). En el esquema de la figura III.1.1 se resume lo descrito anteriormente [36].

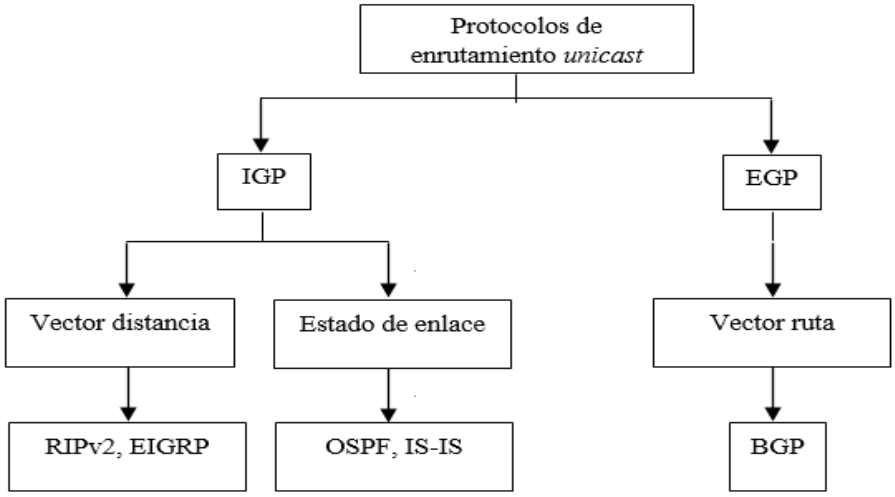


Figura III.1.1. Clasificación de los protocolo de enrutamiento *unicast* IGP y EGP [Diagrama propio con base en referencia 36].

Para que protocolos de enrutamiento *multicast* sean considerados en una red de datos, se debe partir de la premisa de que existe una ruta viable *unicast* para esa misma red, mejor dicho, para que una red *IP* m*ulticast* funcione correctamente, deben coexistir en la misma red tanto protocolos de enrutamiento *unicast* como de tipo *multicast*. Por lo tanto, se hace necesario hacer una revisión breve del protocolo OSPF, debido a que será el protocolo que realizará el enrutamiento *unicast* en nuestra topología de red IPv4 *multicast* propuesta.

III.1.1 Protocolo de enrutamiento dinámico de estado de enlace OSPF

Existen tres versiones del protocolo OSPF; OSPFv1 publicado en el RFC 1131 en 1989, OSPFv2 descrito en el RFC 1247 del año 1991 y la actualización del mismo en el RFC 2328 en 1998, por último, OSPFv3 publicado en el RFC 2740 en 1999. La versión OSPFv2 es la que se usa en redes IPv4 y OSPFv3 para redes IPv6.

OSPF es un protocolo que se considera del tipo estado de enlace ya que ofrece una vista completa de la topología de red, tanto OSPF como los demás protocolos de estado de enlace también se conocen como protocolos basados en el algoritmo matemático *Dijkstra*, o algoritmo SPF (*Shortest Path First* 'Primero la ruta más corta'). El concepto de primero la ruta más corta no hace referencia a la cantidad de saltos (como es el caso de RIP), sino más bien a un valor que asigna el algoritmo SPF a una interfaz que conecta a un *router* con otro. Este valor que asigna se denomina costo, y el costo es la métrica que utiliza el protocolo OSPF para determinar la ruta más corta, es decir, la mejor ruta será aquella con el costo más bajo.

El costo, en OSPF, se relaciona con el BW de una interfaz (enlace) entre un *router* y otro. Debido a lo anterior, el costo en una interfaz que conecta a dos equipos se calcula mediante la ecuación 1:

$$\text{Costo} = 10^8 / \text{BW} \quad \text{[Ecuación 1]}$$

El costo total de la ruta más corta hacia un destino será pues, la suma total de los costos de cada interfaz que construyan dicha ruta más corta.

En la tabla III.1 se muestran las principales velocidades (como BW) establecidas para diferentes enlaces (interfaz) de red, y el valor del costo correspondiente para cada tipo de enlace. La velocidad o ancho de banda máxima que puede alcanzar cada uno de los enlaces, es sólo una medida de referencia, ya que generalmente, la velocidad real de un enlace suele variar por diferentes razones, y por tanto, es diferente al BW por defecto.

Interfaz	C= 10^8/BW (bps)
FastEthernet (100Mbps)	C= 10^8/100 000 000bps= 1
Ethernet (10Mbps)	C= 10^8/10 000 000bps= 10
E1 (2.048Mbps)	C= 10^8/2 048 000bps= 48
T1 (1.544Mbps)	C= 10^8/1 544 000bps= 64
128kbps	C= 10^8/128 000bps= 781
64kbps	C= 10^8/64 000bps= 1562
56kbps	C= 10^8/56 000bps= 1785

Tabla III.1. Principales velocidades o anchos de banda para enlaces de red de datos y el costo calculado de cada uno de los enlaces.

En OSPF existen 5 tipos de mensajes, cada uno de ellos contiene ciertos datos que son encapsulados junto al encabezado OSPF; se utilizan para enviar información de enrutamiento OSPF a todos los *routers* OSPF de una red, y todos los mensajes se envían a la dirección *multicast* 224.0.0.5.

1) Mensaje de saludo (*Hello*). Descubre vecinos y construye adyacencia entre ellos.
2) Mensaje de descripción de la base de datos (DBD). Encargado de controlar la sincronización de la base de datos que se construye entre los *routers* para que todos los *routers* manejan la misma información.
3) Mensaje de solicitud de estado de enlace (LSR). Solicita registros del estado de enlace de las interfaces de *router* a *router*.
4) Mensaje de actualización de estado de enlace (LSU). Encargado de enviar los registros de estado de enlace específicos solicitados por el LSR.
5) Acuse de recibo de estado de enlace (LSAck). Mensaje que reconoce los tipos de paquetes LSA, los cuales contiene la información específica de los enlaces, y van dentro del mensaje LSU.

El encabezado OSPF contiene cierta información con la que se hace posible su funcionamiento. El encapsulamiento del encabezado OSPF y de los diferentes mensajes que utiliza se realiza directamente sobre el protocolo IP; es decir, junto a un respectivo encabezado de capa 3 (paquete IP), tal y como se muestra en la figura III.1.2 [37].

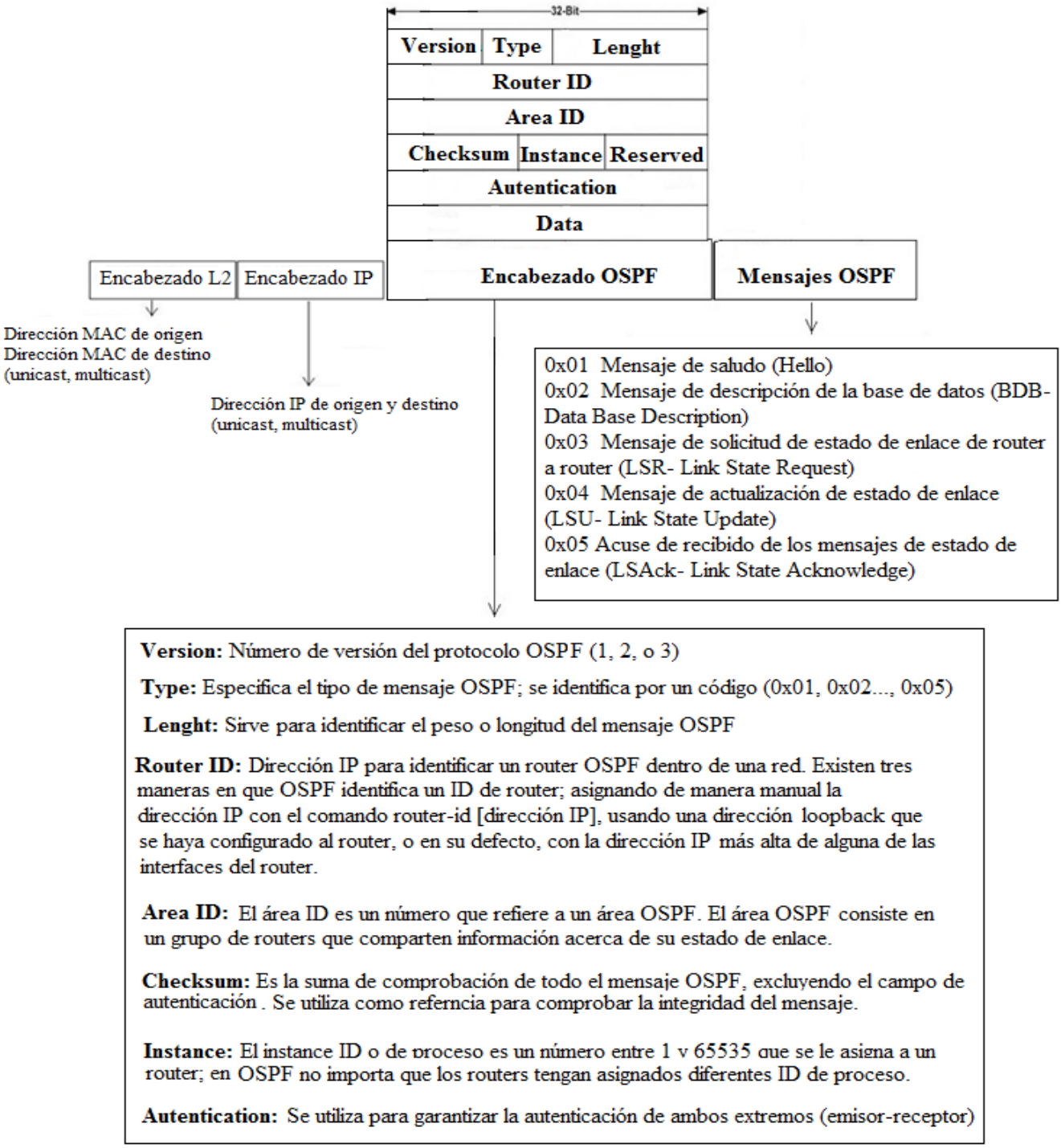


Figura III.1.2. Encapsulación del encabezado OSPF [Diagrama propio con base en referencia 37].

OSPF además de ser un protocolo abierto (corre en equipos, *routers*, de diferentes proveedores), ofrece ciertas bondades que lo hacen ser en la actualidad un protocolo muy recurrido para el diseño de redes de datos de mediana y gran escala. Por ello, se eligió como el protocolo para proveer el enrutamiento *unicast* en el modelo de red *IPv4 multicast* propuesto en este trabajo.

Entre las principales bondades que caracterizan y ofrece el uso del protocolo de ruteo *unicast* OSPF se encuentran:

- La manipulación parcial de la métrica (costo) de los enlaces, mediante la asignación de un ancho de banda manualmente (por línea de comandos en un *router*); la cual permite al administrador cambiar valores de costo de ruta para tener un mejor control de las mismas, y a la vez, cumplir con las necesidades que en algún momento deba requerir una topología de red. Es importante señalar, que generalmente la velocidad real de los enlaces es diferente al ancho de banda por defecto de los mismos, lo que hace al valor de costo únicamente una referencia para elegir la mejor ruta, y no un valor que mida la velocidad real de los enlaces.

- La elección de un DR (*Designated Router* 'Router designado') y un BDR (*Backup Designated Router* 'Router designado de respaldo'). Los cuales tienen la función de ser los *routers* encargados de recibir, concentrar, y enviar a todos los *routers* OSPF mensajes LSA; los cuales contienen información específica de los enlaces, usando para ello los 5 tipos de mensajes descritos con anterioridad (*Hello*, DBD, LSR, LSU, LSAck).

  Con la asignación de un *router* DR y un *router* BDR se reduce de manera considerable el flujo de paquetes en la red, evitando así una inundación (*flooding*) y posibles bucles de enrutamiento en redes principalmente de acceso múltiple ; por ello, no se considera indispensable utilizar DR y BDR en redes de tipo punto apunto [38].

III.2 Protocolos de enrutamiento multicast Dense Mode y Sparse Mode

Para que una red *IP multicast* funcione correctamente deben coexistir en la misma red tanto protocolos *unicast* como protocolos de enrutamiento *multicast.* En este sentido, podemos decir que todos los protocolos de ruteo *multicast* parten de la premisa que existe una ruta viable *unicast*.

Los *mrouters*, junto con los protocolos *multicast*, resuelven el problema de cómo distribuir el tráfico en una red de datos *multicast* y hacerlo llegar desde una fuente emisora hasta el grupo de receptores *multicast* mediante la construcción de árboles de distribución (método de direccionamiento *multicast*).

Los protocolos *multicast* están clasificados en dos principales categorías. La primera categoría corresponde a los denominados protocolos *Dense Mode,* y la segunda categoría es llamada *Sparse Mode*. Dentro de la categoría de protocolos *dense mode* podemos encontrar a DVMRP (*Distance Vector Multicast Routing Protocol* 'Protocolo de enrutamiento multicast de vector distancia'*),* MOSPF (*Multicast OSPF*), y PIM-DM (*Protocol Independent Multicast - Dense Mode* 'Protocolo independiente multicast- modo denso'); mientras que en la categoría de protocolos *sparse mode* encontramos a PIM-SM, CBT (*Core Based Trees* 'Protocolo basado en árboles de núcleo'*)* y MBGP (*Border Gateway Multicast Protocol* 'Protocolo de puerta de enlace de borde multicast'). Por otra parte, tanto los protocolos *dense mode* como *sparse mode* se clasifican de acuerdo a si son los utilizados en una región Intradomain, o si se usan en una región denominada Interdomain.

Además de los protocolos de modo denso y modo disperso, existe un protocolo de red muy importante que debe ser empleado en redes *IP multicast*. Este protocolo es IGMP, considerado, en términos generales, como el protocolo que permite la comunicación entre los equipos receptores que conforman los grupos *multicast*, y los *mrouters* de una red. La función y las características que describen a este último protocolo así como las del protocolo PIM-SM se explican más a detalle en el apartado III.3 y III.4 respectivamente.

El concepto Intradomain en *multicast* refiere a aquella región que abarca un AS, mientras que Interdomain es la región que abarca la interconexión de dos o más AS. Los protocolos DVMRP, MOSPF, PIM-DM, CBT y PIM-SM, son protocolos usados para el enrutamiento *multicast* en una región Intradomain, y el protocolo MBGP es usado para hacer posible el enrutamiento en una región Interdomain tal y como se muestra en el esquema de la figura III.1.3.

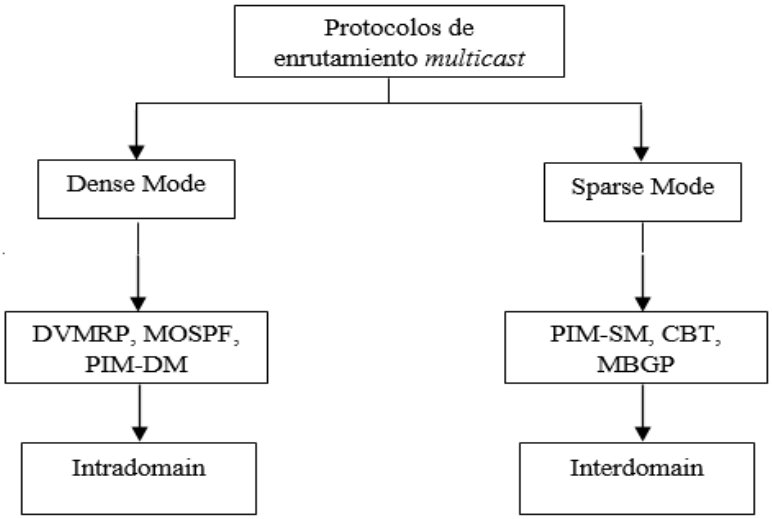


Figura III.1.3. Clasificación de los protocolos de enrutamiento *multicast* [Diagrama propio con base en referencia 39].

Los protocolos dense mode se caracterizan por manejar un proceso de *Flood* and *Prune* (Inundar y Podar/Cortar); este proceso consiste en inundar de tráfico de multidifusión toda la red cada determinado tiempo que establece el protocolo de enrutamiento *multicast* para que la información de multidifusión llegue a todos los puntos posibles donde pueda existir algún receptor, y para posteriormente podar o cortar aquellas rutas en las que no existen receptores. Cuando la red se inunda de tráfico se construyen los denominados árboles de distribución, donde las ramas del árbol representan las rutas hacia cada uno de los receptores de grupo *multicast*; de ahí la utilización del concepto podar o cortar. En el apartado III.2.1 se da una explicación de cada tipo de árbol de distribución y su importancia en *multicast*.

La mayoría de los protocolos sparse mode, en cambio, realizan un proceso denominado modelo de *Pull* (Empuje) o *Join* (Unión). El modelo de *Pull/Join* consiste en que los *routers* conectados directamente con los receptores *multicast* (llamados *routers* de último salto) reciben una notificación explicita de los receptores, quienes expresan su interés de recibir flujo de multidifusión, logrando con esto que el tráfico no se propague o inunde por toda la red sino únicamente llegue a los receptores que soliciten dicho tráfico. Cabe mencionar que en este modelo se considera un punto de encuentro (comúnmente un *router* denominado RP) donde llega el tráfico de las posibles fuentes que existan en la red y de donde se envía el tráfico de dichas fuentes a cada uno de los receptores de grupo *multicast* [39].

III.2.1 Árboles de distribución multicast

En *IP multicast* el direccionamiento del tráfico de multidifusión está basado en la construcción de árboles de distribución; es decir, los árboles de distribución *multicast* definen la trayectoria que sigue un flujo de datos en una red con *multicast* habilitada. La teoría define dos posibles tipos de árboles de distribución *multicast*, los árboles de camino más corto SPT's (*Shortest Path Trees*), y los árboles compartidos ST (*Shared Tree*) o árboles con punto de reunión RPT's (*Rendezvous Point Trees*).

Los SPT's se crean desde la fuente emisora hasta cada receptor perteneciente a un grupo *multicast*, usando para tal acción, el camino más corto que define algún protocolo de enrutamiento *unicast*. Así, si en una red de datos existen dos fuentes emisoras, en la red existirán dos árboles de distribución usando la ruta más corta desde cada una de esas fuentes hacia el grupo de receptores. Los protocolos de tipo *dense mode* como DVMRP, MOSPF y PIM-DM hacen uso de los SPT's para definir las rutas por donde pasará el flujo de multidifusión.

Por el contrario, los arboles ST se construyen usando un *router* como punto de reunión (*router* RP) o raíz, el cual se encargará de distribuir el flujo de multidifusión hasta el grupo de receptores *multicast*; en este caso, cuando dos fuentes emisoras se encuentran en la misma red y transmitan flujo, este último, llegará al punto de encuentro y a partir del mismo el flujo seguirá su camino por un árbol de distribución compartido hasta el grupo de receptores. Los protocolos *sparse mode* como PIM-SM y CBT, hacen uso preferentemente de los RPT's, sin embargo, es conveniente mencionar que PIM-SM, protocolo usado para proveer el enrutamiento *multicast* en el modelo *IPv4 multicast* propuesto, es capaz de pasar de un árbol ST a un árbol SPT.

En síntesis, los SPT's son árboles de distribución que se crean para cada fuente que envía a un grupo *multicast*, lo que implica que un *router* utilice mayor cantidad de memoria debido a que existe una entrada para cada emisor, pero ofrece como ventaja que el envío de información se realice a través del SPT construido como el camino más corto hacia el grupo de receptores. Mientras que los ST o RPT's son árboles compartidos desde un punto de encuentro RP, los cuales se utilizan para que el flujo de información *multicast* llegue a los receptores del grupo reduciendo considerablemente el uso de memoria de los *routers*; sin embargo, los ST no garantizan que el árbol compartido sea el camino más corto hacia los receptores [40].

III.2.2 TTL (*Time To Live*)

En las redes de datos es muy común que se generen bucles a nivel IP (Capa 3) debido a una mala configuración de la red o fallas en la misma. El uso del campo TTL del encabezado IP en un flujo de información sirve para evitar tales bucles.

El principio del campo TTL radica en considerar un valor numérico de entre 0 hasta 255. Este valor insertado en la cabecera IP de un paquete de datos en una red *unicast*, o de un datagrama en una red con *multicast* activado, disminuye en un valor de uno por cada *router* de la ruta hacia el destino. Cuando el valor TTL llega a 0, el paquete o datagrama es descartado; evitando así que siga fluyendo por la red sin rumbo, y en consecuencia, eliminando la posibilidad de que se genere un bucle en la red.

Algunos *routers* de la industria, como por ejemplo los fabricados por *Cisco Systems*, agregan la posibilidad de asignar un valor de umbral de tráfico TTL a sus interfaces, a partir del cual los routers hacen el cambio (conmutan) de árbol compartido ST usando como raíz un RP, al árbol de la ruta más corta SPT.

Por otra parte, el valor de TTL que se debe determinar para transmitir datagramas (flujo) *multicast* debe ser un TTL $\geq$ a la cantidad de saltos (*routers*) que conforman el camino hacia donde se quiere recibir el flujo de multidifusión, con el fin de lograr que la interfaz no bloquee el flujo y este llegue a los destinatarios que lo soliciten.

Algunos valores TTL para definir el alcance de datagramas *multicast* son los siguientes:

0 – Limitado al router local.

1 - Se limite el flujo a la red local

32 – Limitado dentro de una red.

64 – Limitado a redes interconectadas en una región

128 - Limitado a redes interconectadas en un mismo continente

255- Flujo no limitado

El TTL que se debe determinar para transmitir datagramas dentro de una red *multicast*, usualmente, es asignado a través del equipo emisor; específicamente, asignándolo directamente en la aplicación o *software* instalado en el equipo emisor al momento de realizar la emisión del flujo. Considerando lo dicho anteriormente, es preciso señalar y adelantar que el software VLC *media player* que se encargará de realizar la transmisión de audio y video *streaming* del modelo *IPv4 multicast* propuesto y explicado en el capítulo IV, permite modificar y asignar de manera libre el valor de TTL para la transmisión de flujo de multidifusión.

III.2.3 RPF (*Reverse Path Forwarding*)

Partiendo del principio de que los *mrouters* en una red *IP multicast* replican un mismo flujo de información para cada interfaz de salida donde se encuentre un *mrouter* conectado; podemos decir que esta acción, en muchos casos, puede ocasionar que se generen bucles por inundación; ya que un *mrouter* recibirá el mismo flujo por todas sus interfaces, y éste a su vez, enviará el flujo por sus interfaces donde se encuentren otros *mrouters* conectados, y así sucesivamente cada *mrouter* que integre la red.

Para evitar la generación de estos bucles por inundación en una red *IP multicast*, además del valor TTL en la cabecera IP, se utiliza un mecanismo de verificación llamado RPF (*Reverse Path Forwarding* 'Reenvío por camino en reversa'). En términos generales, RPF es un mecanismo que desecha el flujo *multicast* de aquellas interfaces por las cuales se considera innecesario que llegue el flujo y decide hacia dónde replicar el flujo *multicast.* Por ello, RPF es de gran utilidad cuando ocurre la construcción de árboles de distribución *multicast* ya sea de tipo SPT o tipo ST que se realiza en los protocolos PIM-DM como PIM-SM.

RPF realiza tres acciones muy puntuales para desechar flujo *multicast* de aquellas interfaces por las cuales se considera innecesario que arribe flujo:

1. Cuando un *mrouter* recibe un datagrama de multidifusión, el *mrouter* analiza su tabla de ruteo *unicast* y verifica que la interfaz por la cual recibió el datagrama pertenezca o sea parte de la mejor ruta hacia la raíz donde se emitió el flujo, es decir, la dirección de la fuente emisora. Dos resultados pueden surgir como consecuencia de esta primera acción que realiza el mecanismo RPF; el mecanismo falla, o el mecanismo es correcto.

2. Cuando el mecanismo de verificación falla, es debido a que la ruta (interfaz) por la que llegó el flujo de información *multicast* al *mrouter* no corresponde o no es parte de la ruta más corta hacia la fuente emisora que fue determinada por el protocolo de enrutamiento *unicast*. En la figura III.1.4, se muestra una representación explicita de lo descrito con anterioridad [41].

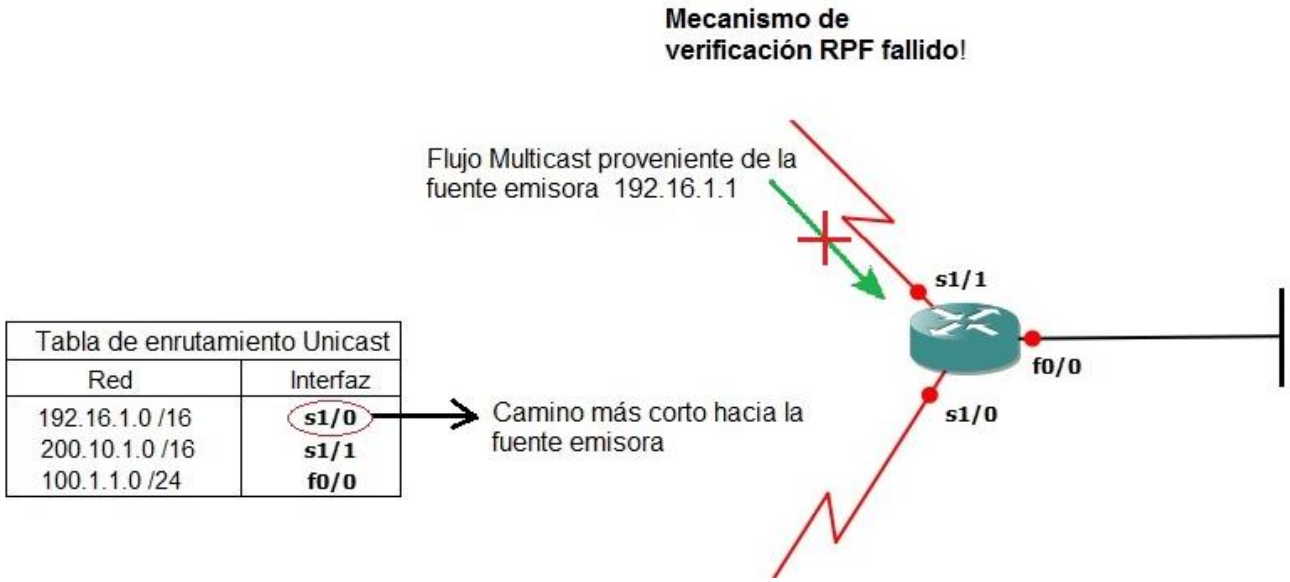


Figura III.1.4. El mecanismo de verificación falla debido a que el tráfico que llega por la interfaz s1/1 del *router* no corresponde a la interfaz que indica el camino más corto la tabla de ruteo [Diagrama propio con base en referencia 41].

3. En caso contrario, cuando el mecanismo de verificación RPF es correcto, es debido a que la ruta por la que llegó el flujo de información *multicast* al *mrouter* coincide y es parte de la ruta más corta hacia a la fuente emisora que fue determinada por el protocolo de enrutamiento *unicast*. En la figura III.1.5 se muestra una representación de lo descrito.

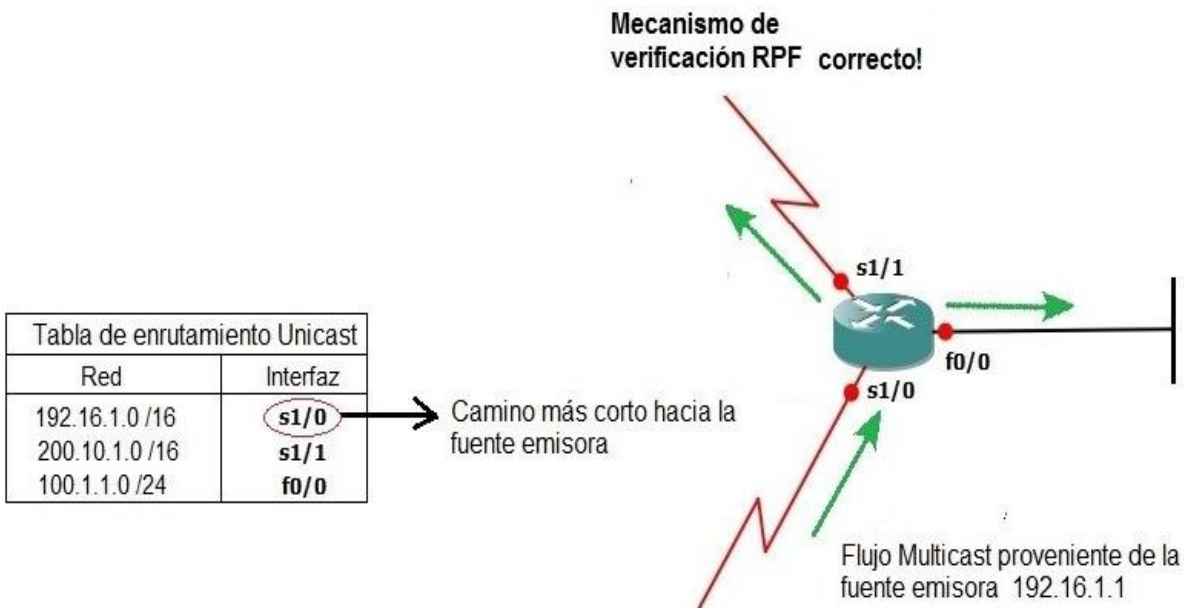


Figura III.1.5. El mecanismo de verificación en este caso es válido debido a que el tráfico que llega por la interfaz s1/0 del *router*, la cual corresponde a la interfaz que indica el camino más corto la tabla de ruteo [Diagrama propio con base en la referencia 41].

III.3 Protocolo de red IGMP

Existen tres versiones del protocolo IGMP (*Internet Group Management Protocol* 'Protocolo de Administración de Grupos para Internet'); IGMPv1 publicado en el RFC 1112 en 1989, IGMPv2 descrito en el RFC 2236 del año 1997 y la versión 3 publicado en el RFC 3376 del 2002. Tal y como se resume en la figura III.1.6.

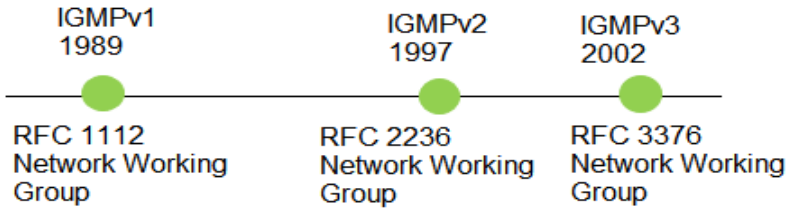


Figura III.1.6. Tres versiones del protocolo de red IGMP, en orden cronológico de acuerdo a la aparición de los documentos RFC donde se describen.

Independientemente de la versión, IGMP es un protocolo que se encapsula en paquetes IP y es el responsable de realizar el intercambio de información acerca del estado de existencia y pertenencia entre receptores que forman un grupo *multicast* y los *routers* que tienen la capacidad de gestionar tráfico *multicast.* Los cambios y mejoras respecto de una versión con otra se dan a partir de las necesidades que han surgido con los años por mejorar la tecnología de multidifusión IP, y principalmente la comunicación que se da entre los miembros de grupo *multicast* con los *routers multicast*.

En la actualidad, IGMP es compatible en la mayoría de los sistemas operativos para computadoras personales, incluyendo los sistemas operativos basados en Unix, Windows (PC´s) y Mac IOS.

Con lo que respecta a los *routers* Cisco, por defecto, a partir del IOS 11.1 se ejecuta IGMP en su versión 2 al activarse enrutamiento *IP multicast* en los *routers*. La versión 3 del mismo protocolo se encuentra limitada para algunos equipos con funciones más avanzadas.

En este sentido, es oportuno mencionar que se utilizó la versión IGMPv2 por defecto en los *routers* empleados para diseñar el modelo de red IPv4 multicast de este trabajo. Por ello, este apartado estará enfocado a explicar el funcionamiento de la versión 2 del Protocolo de Administración de Grupos para *Internet* IGMP.

El encabezado del protocolo IGMPv2 contiene cierta información útil para hacer una mejor descripción del mismo. El encabezado IGMPv2 y los campos que lo constituyen se muestra en la figura III.1.7 [42].

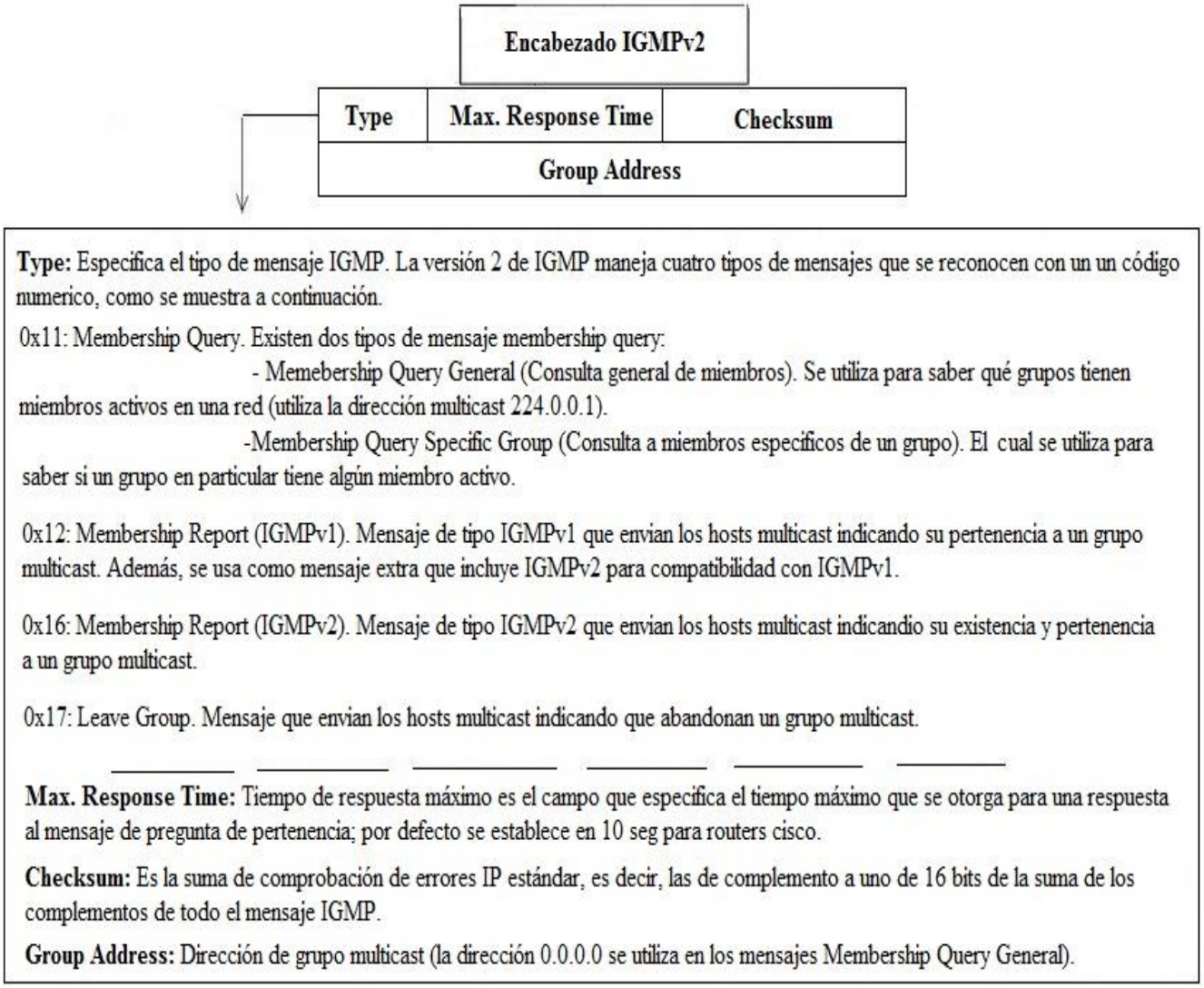


Figura III.1.7. Se muestra el encabezado IGMPv2 y su contenido por campo [Diagrama propio con base en referencia 42].

El funcionamiento de la mayoría de los protocolos de enrutamiento es con base en el uso de una serie de mensajes con funciones específicas. Para el caso del protocolo IGMP, y en particular de IGMPv2 no hay excepción. IGMPv2 utiliza los mensajes IGMP 0x11, 0x12, 0x16 y 0x17 que se muestran en la figura III.1.7 para realizar la comunicación entre varios receptores pertenecientes a un grupo *multicast* y el *router* de último salto que está conectado directamente con tal grupo, así como se muestra en la figura III.1.8.

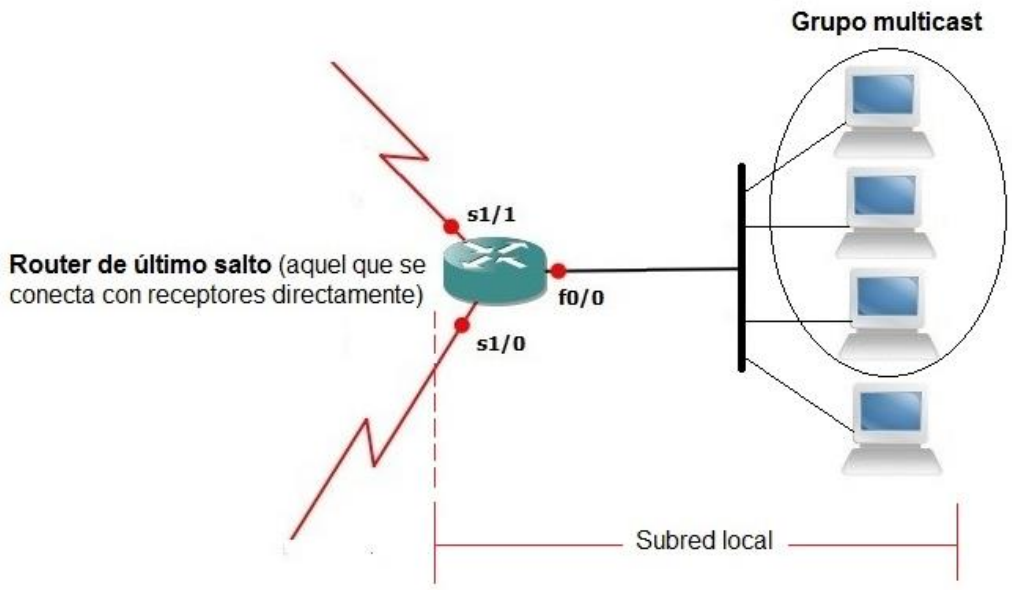


Figura III.1.8. El protocolo IGMP (v1, v2 y v3) funciona entre un grupo *multicast* y el *router* que conectado directamente con ese grupo (*router* de último salto) [Diagrama propio].

El mensaje IGMP 0x11, es el mensaje *Membership Query* (Consulta de pertenencia) y en IGMPv2 existen dos subtipos:

Membership Query general; es un mensaje de consulta general que inicialmente hace un *mrouter* de último salto enviándolo a cualquier grupo *multicast*, usando para ello, la dirección 0.0.0.0 del campo *Group Address* en el encabezado IGMPv2 y la dirección *multicast* 224.0.0.1 (de todos los sistemas en una subred local), con la finalidad de obtener información acerca de qué grupos *multicast* existen en la subred. Este subtipo de mensaje tiene prioridad sobre el subtipo de mensaje *Membership Query specific*, ya que con él se hace el descubrimiento de todos los posibles grupos en una subred local. Además, periódicamente *mrouter* de último salto envía periódicamente una consulta general para mantener un conocimiento de aquellos grupos que persisten, que han dejado de pertenecer o que se han integrado en la subred local.

Membership Query specific es un mensaje de consulta que inicialmente hace un *mrouter* de último salto enviándolo a la dirección *multicast* de un grupo específico dentro del campo *Group Address*. Este mensaje usualmente participa hasta que el *mrouter* sabe de la existencia de un grupo *multicast* específico en la subred local.

El mensaje IGMP 0x16, es el mensaje de tipo *Membership Report* (Reporte de pertenencia) que envían los sistemas de una subred (receptores de un grupo *multicast*) para unirse a un grupo de multidifusión e indicar al mismo tiempo al *router* de último salto que pertenecen a determinado grupo *multicast*; el mensaje es enviado a la dirección de grupo *multicast* al que se unieron o pertenecen. Para que los sistemas de una subred local reporten su pertenencia a un grupo no es necesario que previamente un *router* envíe un *membership query general*. Sin embargo, cuando reciban el mensaje de consulta general los sistemas responderán inmediatamente con un *membership report.*

Para IGMPv2 se incluye el mensaje 0x17 *Leave Group - Dejar grupo* (no existe en IGMPv1), que tiene como función informar al *router* de último salto que abandona un grupo *multicast*. La importancia principal del mensaje *Leave Group* recae en el hecho de que el *router* de la subred local tendrá conocimiento del abandono de cada uno de los miembros de un grupo, y en determinado momento cuando todos los miembros abandonen, entonces el *router* sabrá que el grupo ya no existe y por tanto dejará de enviar mensajes *membership query.*

En la figura III.1.9 se presenta de manera gráfica cómo actúan los mensajes *membership query general*, *membership report,* y *leave group* en el sentido *output* -salida *Inutput* -entrada de los mismos.

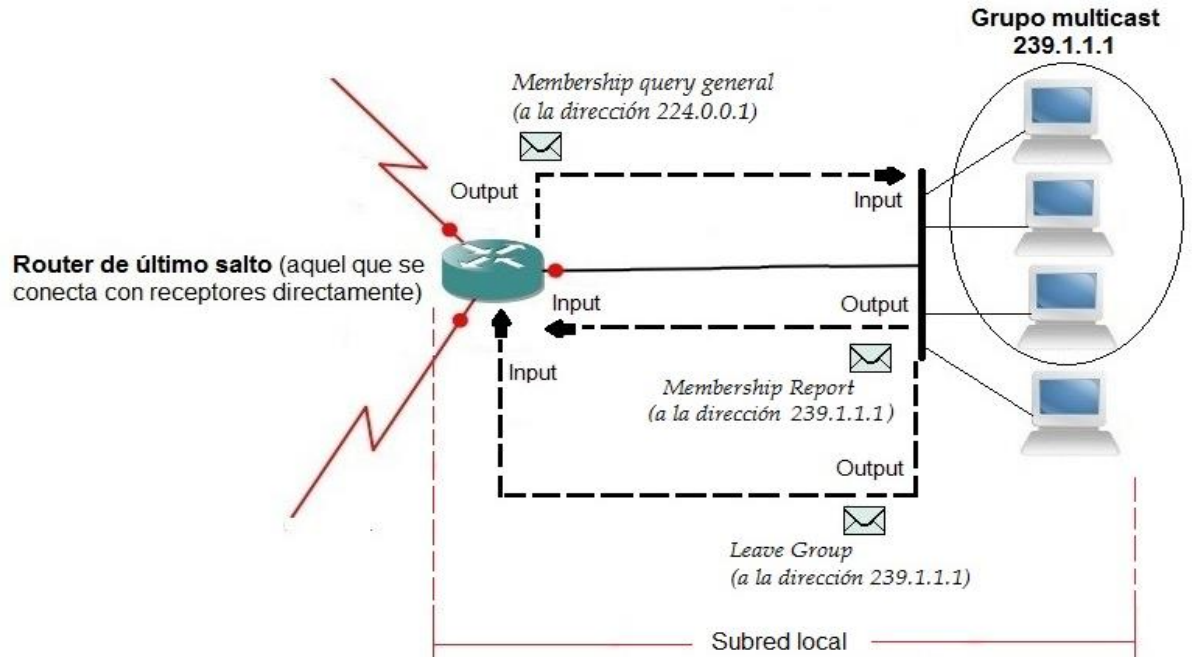


Figura III.1.9 Esquema que muestra la salida y entrada de los mensajes 0x11, 0x16 y 0x17 que utiliza IGMPv2 para comunicar a grupos *multicast* y un *router* de último salto.

El mensaje IGMP 0x12 se presenta cuando existen varios *routers* conectados a una subred local (red de acceso múltiple), y alguno e incluso varios de estos *routers* corren el protocolo IGMPv1 junto a otros *routers* que fueron configurados para trabajar con IGMPv2. Así mismo, para que exista interoperabilidad entre un *router* IGMPv2 con aquellos sistemas receptores que corren el protocolo IGMPv1 [43].

III.4 Protocolo de enrutamiento multicast PIM-SM

PIM *(Protocol Independent Multicast* 'Protocolo Independiente Multicast'*)*, es el protocolo Intradomain de enrutamiento más usado actualmente en redes *IP multicast*. PIM se denomina independiente debido a que utiliza la información de la tabla de enrutamiento de alguno de los protocolos *unicast* existentes para saber las mejores rutas hacia los receptores. PIM existe en dos formatos: PIM-DM de tipo *dense mode*, y PIM-SM, que es la versión de tipo *sparse mode* y de la cual se hablará en esta sección.

El documento RFC (*Request for comments* 'Solicitud de comentarios') 4601 (Agosto del 2006) es el documento más actual que habla de las especificaciones para el protocolo PIM-SM. Y para seguir con la misma línea que se ha trazado en la explicación de los protocolos UDP, RTP, OSPF e IGMPv2; a continuación se presenta en la figura III.1.10 el contenido (campos) del encabezado perteneciente al protocolo PIM-SM, y en particular el contenido de la versión 2 de este último, debido a que es la versión que actualmente se utiliza por defecto en los *routers* Cisco con capacidad de enrutamiento *multicast* [44].

Una parte que se debe realizar para cumplir con el objetivo de este proyecto es la transmisión de audio o video a un grupo *multicast*; para lo cual se requiere además de la participación de los protocolos de comunicación y de enrutamiento UDP, RTP, OSPF e IGMPv2 tratados con anterioridad, la fundamental intervención del protocolo PIM-SM. Por ello, en este apartado se explica de manera muy puntual los tipos de mensajes que utiliza el protocolo para lograr el enrutamiento de tráfico *IP multicast* en una red. Pero se ofrece en el apartado IV.4 una explicación más profunda del funcionamiento de PIM-SM, usando para ello el modelo de red *IPv4 multicast* propuesto en este trabajo.

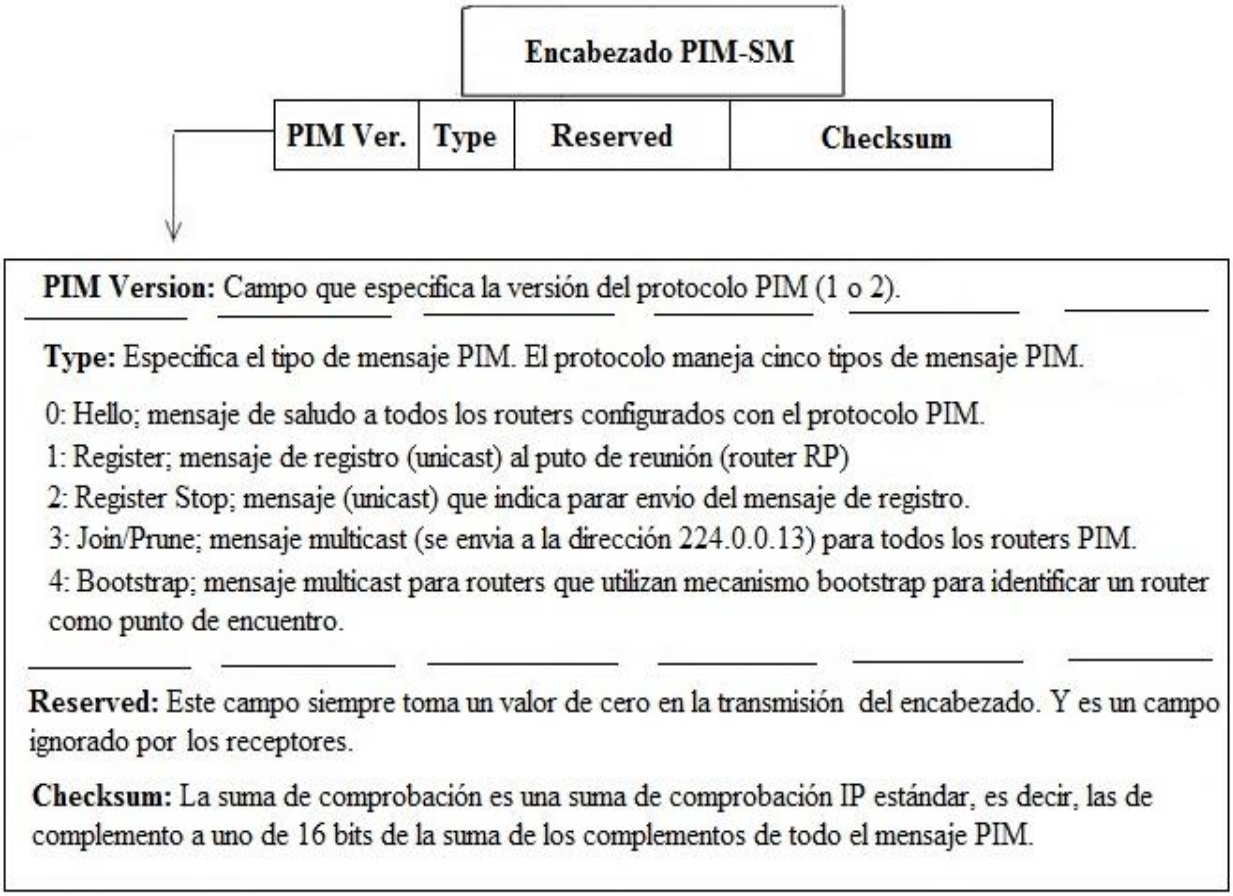


Figura III.1.10: Encabezado PIMv2 S-M [Diagrama propio con base en referencia 44].

El funcionamiento de PIM-SM y el enrutamiento de tráfico *IP multicast* se logra principalmente a través de los mensajes 0, 1, 2 y 3 que se muestran en la figura III.1.10. Además, estos mensajes sirven para crear los árboles de distribución SPT's, así como los ST. Los árboles de distribución en general, son el camino que tomará un flujo *multicast* para llegar desde una fuente a un grupo de receptores.

El mensaje 0: *Hello*. Tiene como función hacer que un *router* en una red descubra que otros *routers* en la misma red hablan el mismo idioma, es decir, un *router* PIM-SM, hará uso del mensaje *Hello* enviándolo a la dirección 224.0.0.13 para saber si existen más *routers* corriendo el mismo protocolo, así mismo para que esos *routers* sepan de la existencia del *router* PIM-SM. Generalmente, en una red *IP Multicast* trabajando con PIM-SM éste es el primer mensaje que es lanzado por los *routers.*

El mensaje 1: *Register*; permite que un *router* (DR) denominado de primer salto (aquel conectado directamente con la fuente emisora) registre a una fuente emisora de tráfico de multidifusión a un *router* (RP) (*router* de punto de encuentro), para que posteriormente la fuente emisora pueda transmitir el flujo *multicast.* El mensaje *Register*, en consecuencia, sólo se presenta y actúa entre la fuente emisora de multidifusión, un *router* (DR) y el *router* RP. La participación del mensaje *Register* entre estos tres equipos permite la construcción de un árbol SPT desde la fuente hasta el *router* RP. Por último, este tipo de mensaje es enviado a través de unidifusión hacia el *router* RP usando la mejor ruta definida por un protocolo *unicast*, por tanto no usa una dirección *multicast*.

El mensaje 2: *Register-stop*. El mensaje *Register-stop*, como su nombre lo indica, tiene estrecha relación con el mensaje 1 *Register*. Ya que el mensaje *Register-stop* se usa en PIM-SM para detener el mensaje *Register* cuando una fuente emisora de tráfico *multicast* ha sido registrada a un *router* (RP). Este mensaje es enviado por el *router* (RP) hacia el *router* (DR) y a la fuente emisora *multicast* a través de unidifusión por la misma ruta por la que fue enviado el mensaje *Register*. Una vez que se envía el mensaje de registro a un *router* (RP), y este devuelve el mensaje de alto al registro. El árbol SPT entre fuente, *router* (DR) y *router* (RP) habrá sido construido.

El mensaje 3: *Join/Prune.* El mensaje *Join/Prune* actúa de dos maneras; la primera como mensaje que permite crear entradas (*, Group) en las tablas de enrutamiento *multicast* de los *routers* para que estos se unan (Join) a árboles de tipo ST. Así como también crear entradas (S, Group) en las tablas de enrutamiento *multicast* de los *routers* para que estos se unan a árboles SPT.

Las entradas (*, Group) se establecen para cualquier fuente emisora *multicast*, representada por *, y un grupo *multicast* en particular. Mientras que las entradas (S, Group) son para una fuente emisora S y grupo *multicast* específicos.

La segunda manera en la actúa Join/Prune es como mensaje que permite podar (Prune) aquellas rutas que utilizaban los routers unidos a árboles ST, y que posteriormente pasaron a ser parte de un árbol SPT.

Por último los mensajes 4: (Bootstrap) y 5: (Assert), se usan cuando se define un RP de forma dinámica. En PIM-SM el definir un RP es necesario para que el protocolo pueda cumplir su función dentro de la red. Y los mecanismos que existen para asignar un RP en la configuración del protocolo PIM-SM son:

RP estático

- Bootstrap Router (BSR)
- Auto-RP
- Anycast-RP
- RP Phantom
- RP Embedded

Se adelanta que la asignación de RP durante la configuración del protocolo PIM- SM en el modelo IPv4 *multicast* se realiza mediante el uso de RP estático, ya que es el más sencillo y se obtiene un mejor control de la red [45].

Capítulo IV

Diseño, Configuración y Emulación de un Modelo IPv4 Multicast

En este cuarto capítulo, con base en la teoría vista en los capítulos II y III, se propone el diseño, configuración y emulación de un modelo *IPv4 multicast* con el fin de transmitir audio y video s*treaming* a un grupo determinado de receptores usando como herramienta el emulador de redes avanzado GNS3. Con el diseño, configuración y emulación del modelo que a continuación se propone, se obtiene un modelo de red como referencia para ser usado en una *Intranet* de alguna empresa o institución pública y privada donde se desee transmitir audio y video *streaming* usando la tecnología IP *multicast.*

IV.1 Un modelo IPv4 Multicast para la red interna MAN/WAN de la UACM

Llevar a cabo el diseño, configuración y emulación de un modelo *IPv4 multicast*, requiere que tengamos como referente una topología de alguna red de datos real. Por tal motivo, se propone diseñar dicho modelo *IPv4 ulticast* sobre la *Intranet* de datos de la Universidad Autónoma de la Ciudad de México, únicamente como una referencia donde se toma en cuenta la ubicación geográfica de cinco planteles que se encuentran dentro de la Ciudad de México; San Lorenzo Tezonco, Casa Libertad, Cuautepec, Centro Histórico, y Del Valle. Los cinco planteles se interconectan para formar la *Intranet* UACM de extensión geográfica MAN/WAN. Cada plantel estará representado por un *router* con capacidad de soportar enrutamiento *unicast* así como enrutamiento *multicast*, es decir, capaz de soportar la tecnología *IPv4 multicast*; y por tanto, con la posibilidad de que cada plantel contenga una fuente transmisora de flujo *multicast* o sea parte de un grupo *multicast* donde varios receptores reciban

el flujo. En el esquema de la figura IV.1.1 se representa la ubicación geográfica de cada plantel y su representación como una red MAN/WAN.

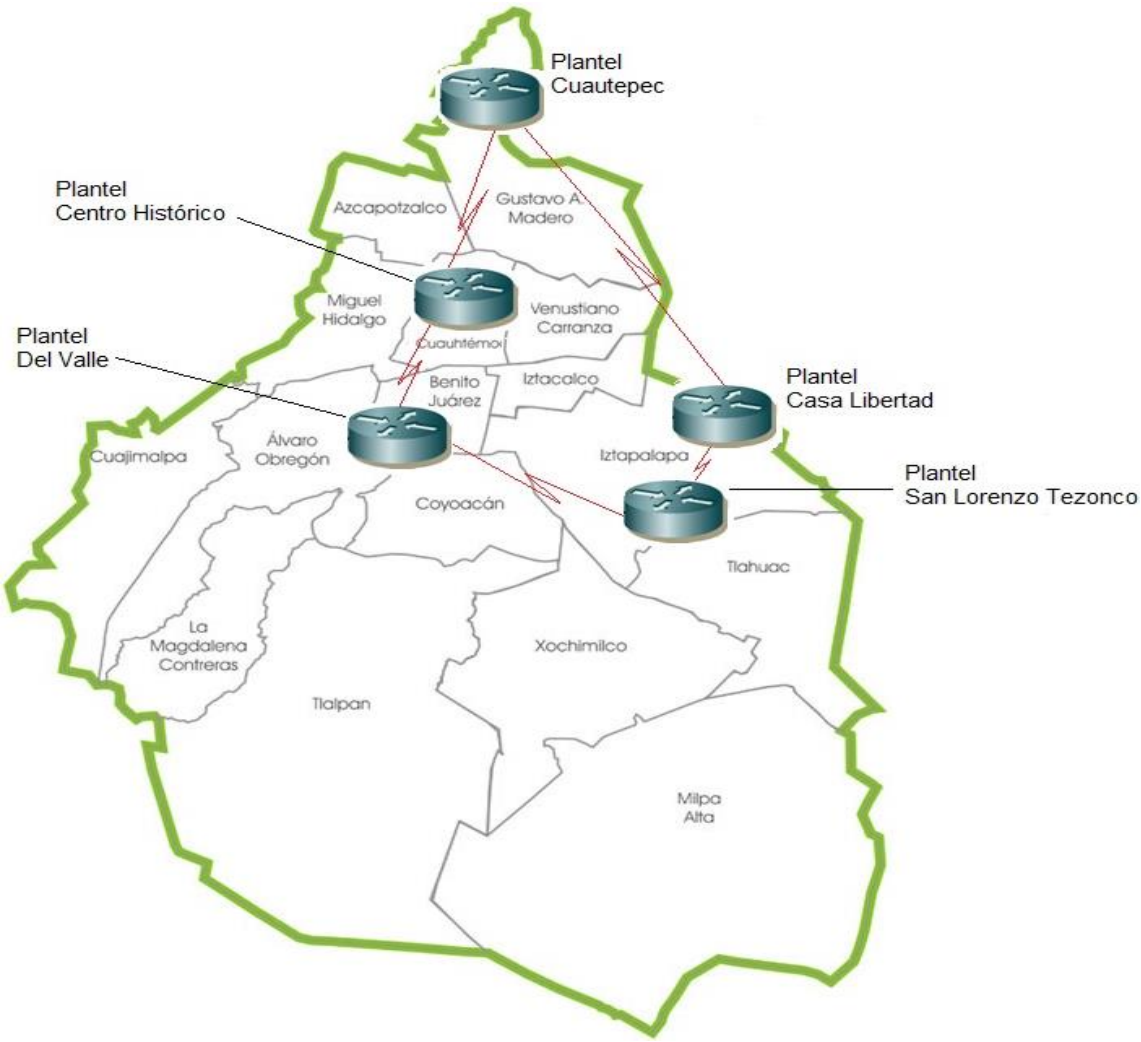


Figura IV.1.1. Representación geográfica de la red de datos interna (*Intarnet*) de la UACM, conformada por cinco *routers* capaces de soportar la tecnología *IPv4 Multicast* [Diagrama propio].

Cabe hacer énfasis que para el modelo de red *IPv4 multicast* propuesto, se consideró la red de datos de la UACM únicamente como una referencia geográfica, en el sentido de la ubicación de cinco de sus planteles, y no se tomó en cuenta información real y específica acerca de la red como por ejemplo, tipo físico de enlaces, velocidad de enlaces, cantidad de *hosts* en cada plantel, ancho de banda disponible, etc. Por la razón de que el modelo propuesto, considerando su diseño, configuración y emulación, es útil como modelo de referencia base para ser usado en cualquier otra red de datos interna ya sea de una empresa o institución pública y privada.

El usar la red de datos de la UACM como referencia para la implementación de la tecnología *IPv4 multicast*, tiene como objeto dar claridad a un posible escenario donde la implementación de la tecnología beneficie a la propia red; reduciendo el uso del ancho disponible, la congestión de tráfico de información dentro de la red, la carga de procesamiento de los equipos, y dando posibilidad de aumentar el número de usuarios para ofrecerles servicios multimedia de audio y video *streaming*.

IV.2 GNS3 como herramienta para diseñar, configurar y emular una red de datos IPv4 Multicast

GNS3 (*Graphical Network Simulator* 'Simulador grafico de redes') es un *software* emulador de alta capacidad que permite diseñar tanto sencillas como complejas topologías de red, así como configurar y emular equipos virtuales de red como *routers*, *switches*, *firewalls*, *servers* y *hosts*. Se añade al presente trabajo el apéndice A, el cual contiene la explicación de algunos aspectos importantes de la instalación y de la configuración que se debe realizar a GNS3 para que funcione de manera correcta [46].

A través de GNS3, fue posible diseñar así como configurar y emular el modelo de red propuesto en el apartado IV.1. Para ello, se requirió hacer uso de cinco *routers*; los *routers* SLT, Casa Libertad, Cuautepec, Centro Histórico, y Del Valle. Los diferentes modelos o plataformas de *routers* que incorpora GNS3 tienen como principio de funcionamiento la ejecución de Cisco IOS (*Cisco Internetwork Operating System*); el *software* que soporta diversos protocolos de red, incluyendo por supuesto los protocolos para implementar el enrutamiento tanto *unicast* como *IP multicast*, así como diversas tecnologías de red.

GNS3 también permite la incorporación y ejecución de *hosts* como máquinas virtuales creadas en *Oracle VM VirtualBox*; con ello, se logró incorporar a nuestro modelo de red cuatro máquinas virtuales corriendo el SO Windows XP. Una máquina virtual como responsable de ser la fuente transmisora de tráfico *multicast* desde el plantel (*router*) Del Valle, y las tres restantes; una ubicada en el plantel SLT, otra en Casa Libertad y la tercera en el plantel Cuautepec, actuando como receptores del tráfico de multidifusión y miembros pertenecientes al grupo *multicast* **239.1.1.1**.

La posibilidad que ofrece GNS3 de emular máquinas virtuales (PC's) con *VirtualBox*, así como equipos de enrutamiento a través del uso de Cisco IOS; da como resultado, tener un acercamiento cuasi real del comportamiento que tendría el modelo de red *IPv4 multicast* funcionando en un escenario de la vida real, así como la ventaja de tener un control de gestión completo de la topología de red de datos diseñada.

En la tabla IV.1, se indica la plataforma de los cinco *routers* utilizados para diseñar y emular el modelo de red *IPv4 multicast* de la UACM, la versión de Cisco IOS y la imagen de la misma que se utilizó para cada plataforma de *router*, así como los tipos de sistemas operativos integrados al equipo transmisor y a los tres receptores (todas máquinas virtuales) pertenecientes al grupo *multicast*.

Router	Serie/ Plataforma	Version Cisco IOS Imagen IOS	VM *Virtual Box* Máquinas virtuales	Sist. Operativo
R1 SLT	c7200/ c7200	12.4 (4) T1 c7200-advipservicesk9-mz.124-4.T1.image	Receptor	Windows XP
R2 Casa Libertad	3700/ 3725	12.4 (4) T1 C3725-AD.image	Receptor	Windows XP
R3 Cuautepec	c7200/ c7200	12.4 (4) T1 c7200-advipservicesk9-mz.124-4.T1.image	Receptor	Windows XP
R4 Centro Histórico	3700/ 3725	12.4 (3) C3725-AD.image		
R5 Del Valle	3700/ 3725	12.4 (3) C3725-AD.image	Emisor	

Tabla IV.1. Las plataformas de los *routers* aquí propuestos funcionan a través de la ejecución de la imagen de Cisco IOS que se integran en la configuración del emulador GNS3. Sin estas imágenes simplemente los equipos no funcionarían.

El *Cisco* IOS (*software*) de los *routers Cisco*, de acuerdo a la versión, permite configurar y activar por línea de comandos diferentes funcionalidades de comunicación, ruteo, seguridad, QoS, tecnologías y servicios de redes así como de telecomunicaciones en general a los propios equipos. Sin

embargo, no sólo se debe considerar la versión IOS de los *routers* para que los servicios y tecnologías de red funcionen; sino además, se debe tener en cuenta la serie o plataforma del equipo físico (*hardware*), ya que de ella depende el soporte de la tecnología o servicio que se desea configurar y activar. Por tanto, la capacidad de un equipo de red para soportar algún servicio o tecnología depende tanto del *software* (versión de IOS) como del *hardware* (versión o serie de plataforma).

Para la tecnología avanzada de red *IPv4 multicast*, la versión de IOS 12.4 (4) T1 y 12.4 (3) utilizadas en las plataformas de *routers* c7200 y 3700 respectivamente, no presentaron ningún problema para la realización de la configuración y emulación del modelo de red propuesto en GNS3, ya que tanto la versión de *software* como de *hardware* soportan el servicio [47].

Considerando lo descrito con anterioridad, en la figura IV.1.2 se muestra el diseño de la topología de red del modelo *IPv4 multicast* creada en GNS3, evidentemente haciendo uso de los cinco *routers* y las cuatro máquinas virtuales descritas.

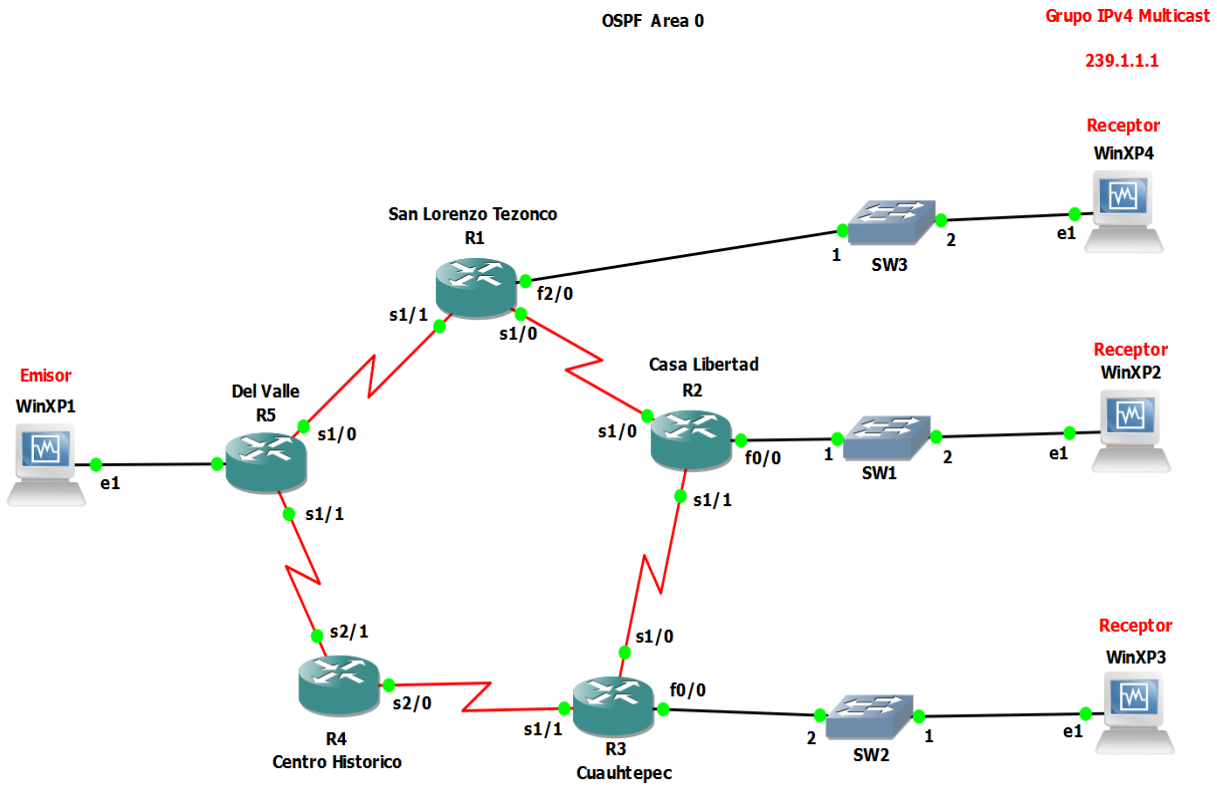


Figura IV.1.2. Topología de red *IPv4 multicast* diseñado en el emulador GNS3. Este diseño sirve como referencia base para otras topologías de red similares [Diagrama propio].

Se rescata que los protocolos de capa de enlace de datos para enlaces seriales en redes de tipo MAN/WAN que ofrece de manera específica el IOS de Cisco para configurar los *routers* dentro de GNS3 son tres; HDCL (*High Level Data Link Control* 'Alto nivel de control de enlaces de datos'), PPP (*Point to Point Protocol* 'Protocolo punto a punto'), y FR (*Frame Relay* 'Retransmisión de tramas'). Mientras que para las redes locales, el protocolo de capa 2 para la interconexión de equipos que ofrece GNS3 es *ethernet* (*Fast y Giga ethernet*).

Los cinco enlaces MAN/WAN en nuestro modelo de red *IPv4 Multicast* se establecieron como de tipo HDCL, y se definió una única velocidad o BW T1= 1.544Mbps para cada uno de los cinco enlaces. Esta velocidad o ancho de banda máxima que puede alcanzar cada uno de los enlaces, es sólo una medida de referencia, ya que generalmente la velocidad real de un enlace es diferente al BW por defecto. En este sentido, es preciso recordar que el protocolo OSPF; protocolo encargado de hacer posible la comunicación *unicast* en nuestro modelo de red *IPv4 multicast*, hace uso del BW por defecto para realizar el cálculo del costo (métrica) de cada enlace, y en general, para calcular la mejor ruta o rutas entre una punto a otro de la red.

IV.3 Configuración del modelo IPv4 Multicast para la transmisión de audio y video Streaming

Una vez que se diseñó la red *IPv4 Multicast* sobre GNS3, lo siguiente fue realizar la configuración de esta última, primero, asignando direcciones IPv4 a cada interfaz activa, y posteriormente, añadiendo los protocolos adecuados para proveer la comunicación entre los equipos de la red y se logrará que el tráfico de audio y video transmitido desde la fuente (en este caso ubicada en el plantel Del Valle), llegará hasta el grupo de receptores *multicast* ubicados en los planteles SLT, Casa Libertad y Cuautepec.

Antes de realizar la configuración de los equipos con los protocolos de enrutamiento *unicast* y *multicast*, es conveniente destacar cuáles direcciones de red IPv4 fueron asignadas a las interfaces de los *routers* y a los equipos en general participes en la red para lograr el direccionamiento lógico. En la tabla IV.2, se muestra el conjunto de direcciones IPv4 asignadas a las interfaces activas de cada uno de

los *routers* así como de los *hosts* partícipes en la topología de red. Se agrega además a la tabla el protocolo que se configuró a cada *router* para proveer el direccionamiento *unicast*, y los protocolos utilizados para proveer el enrutamiento *multicast.*

***Router* (Plantel)**	**Protocolo unicast**	**Interfaces**	**Dirección IPv4 /mascara de red para c/interface**	**Protocolo multicast**
R1 SLT	OSPFv2	FastEthernet2/0 Serial1/0 Serial1/1	192.170.3.254 /24 10.0.0.1 /30 100.0.0.1 /30	PIM-SM, IGMPv2 PIM-SM (RP estático) PIM-SM
R2 Casa Libertad	OSPFv2	FastEthernet0/0 Serial1/0 Serial1/1	192.168.1.254 /24 10.0.0.2 /30 11.0.0.1 /30	PIM-SM, IGMPv2 PIM-SM PIM-SM
R3 Cuautepec	OSPFv2	FastEthernet0/0 Serial1/0 Serial1/1	172.16.1.254 /24 11.0.0.2 /30 101.0.0.1 /30	PIM-SM, IGMPv2 PIM-SM PIM-SM
R4 Centro Histórico	OSPFv2	Serial2/0 Serial2/1	101.0.0.2 /30 1.1.1.1 /30	PIM-SM PIM-SM
R5 Del Valle	OSPFv2	FastEthernet0/0 Serial1/0 Serial1/1	172.30.1.254 /24 100.0.0.2 /30 1.1.1.2 /30	PIM-SM PIM-SM
Host Emisor (WinXP1)		Ethernet1	172.30.1.1 /24 Gateway:192.170.2.254	IGMP
Host Receptor (WinXP2)		FastEthernet1	192.168.1.1 /24 Gateway:192.168.1.254	IGMP
Host Receptor (WinXP3)		FastEthernet1	172.16.1.1 /24 Gateway:172.16.1.254	IGMP
Host Receptor (WinXP4)		FastEthernet1	192.170.3.3 /24 Gateway:192.170.3.254	IGMP

Tabla IV.2. Conjunto de direcciones IPv4 de clase A, B y C asignadas a cada una de las interfaces activas de todos los equipos participes en la topología de red de datos IPv4 *multicast* de la UACM [Diagrama propio].

IV.3.1 Configuración de enrutamiento unicast (OSPF)

Para activar el enrutamiento *unicast* en el modelo *IPv4 multicast*, y con ello lograr la mejor y mayor eficiencia de comunicación *host to host* entre los equipos receptores y el equipo emisor, se propone el protocolo de enrutamiento OSPFv2 para ser configurado en los cinco *routers* Cisco que componen la red de datos.

La configuración básica que a continuación se ofrece fue realizada en cada *router* Cisco de la red de datos y abarca en general; la activación del enrutamiento *unicast* OSPF uniárea sin asignación de DR ni BDR, ya que no se recomienda en topologías de red punto a punto como es el caso del modelo de red aquí propuesto.

Para activar el enrutamiento OSPF en cada uno de los cinco *routers* se ejecutó el comando:

```
Router(config)# router ospf [Id del proceso]
```

Donde:

ID del proceso; es un número que usa internamente el *router* para identificar múltiples procesos OSPF en ejecución. En la configuración realizada en los cincos *routers* del modelo de red *IPv4 Mcast* se asignó un Id del proceso= 1.

Para que cada *router* identificará las redes conectadas directamente a ellos, se tuvo que ejecutar el siguiente comando tantas veces como redes conectadas a cada *router:*

```
Router(config-router)# network[ IP de la red][máscara wildcard] area[ID
area]
```

Donde:

Máscara *wildcard*; es el inverso de la máscara de subred de la dirección IP de red.

ID de área se refiere al área OSPF. El área OSPF es un grupo de *routers* que comparten información sobre el estado de enlace. El ID de área OSPF en este caso se asignó como 0.

La configuración de enrutamiento *unicast* con OSPF realizada a cada *router* del modelo *IPv4 multicast* se muestra en la tabla IV.3.

Router	Configuración de enrutamiento unicast OSPF
R1  SLT	SLT#conf t SLT(config)#router ospf 1 SLT(config-router)# network 10.0.0.0 0.0.0.3 area 0 SLT(config-router)# network 100.0.0.0 0.0.0.3 area 0 SLT(config-router)# network 192.170.3.0 0.0.0.255 area 0
R2  Casa Libertad	CasaLibe#conf t CasaLibe(config)#router ospf 1 CasaLibe(config-router)# network 10.0.0.0 0.0.0.3 area 0 CasaLibe(config-router)# network 11.0.0.0 0.0.0.3 area 0 CasaLibe(config-router)# network 192.168.0.0 0.0.0.255 area 0
R3  Cuautepec	Cuautepec#conf t Cuautepec(config)#router ospf 1 Cuautepec(config-router)# network 11.0.0.0 0.0.0.3 area 0 Cuautepec(config-router)# network 101.0.0.0 0.0.0.3 area 0 Cuautepec(config-router)# network 172.16.1.0 0.0.0.255 area 0
R4  Centro Histórico	CentroHist#conf t CentroHist(config)#router ospf 1 CentroHist(config-router)# network 101.0.0.0 0.0.0.3 area 0 CentroHist(config-router)# network 1.1.1.0 0.0.0.3 area 0
R5  Del Valle	Valle#conf t Valle(config)#router ospf 1 Valle(config-router)# network 100.0.0.0 0.0.0.3 area 0 Valle(config-router)# network 1.1.1.0 0.0.0.3 area 0 Valle(config-router)# network 172.30.1.00.0.0.255 area 0

Tabla IV.3. Configuración de los 5 equipos Cisco con la cual se logra el enrutamiento *unicast* a través del protocolo OSPF.

Finalmente, para comprobar que la comunicación *unicast* existiera gracias al enrutamiento de OSPF en la red; se realizó el envió de *pings* entre todos los equipos (*routers* y *hosts* emisor y receptores) de la red.

IV.3.2 Configuración de enrutamiento multicast (PIM-SM e IGMP)

Una vez que se configuró el modelo de red para activar la comunicación de tipo *unicast* a través del protocolo de enrutamiento OSPF, lo siguiente fue realizar la configuración de de enrutamiento *multicast* a través de los protocolos PIM-SM e IGMP. Para lo cual, se realizaron los siguientes pasos:

1) Cada *router* de la red se configuró para soportar enrutamiento y tráfico *multicast*, activando *IP multicast* y a la vez el protocolo IGMP en cada uno de los *routers* a través del comando:

```
Router(config)#ip multicast-routing
```

2) Con el propósito de que los cinco *routers* soportarán y entendieran el tráfico que se genera con el protocolo PIM-SM. Este último, se tuvo que activar en todas las interfaces en funcionamiento de cada uno de los *routers* mediante el comando:

```
Router(config-if)#ip pim sparse-mode
```

3) Al configurar PIM-SM en una red, por lo menos un *router* debe ser designado como un punto de encuentro (RP - *Rendezvous Point*). El RP podría configurarse estáticamente o de forma dinámica a través de los métodos *Bootstrap Router* (BSR), o Auto-RP. Para elegir al RP en nuestro modelo, se optó por el mecanismo RP estático. Para ello, cada *router* de la red debe saber de forma explícita la dirección IP que corresponde al RP, y que para ser más precisos es una dirección IP de alguna interfaz del *router* que se quiere usar como punto de reunión (en este caso se eligió al *router* SLT y su interfaz Serial1/0 con IP: 10.0.0.1 como RP). Así, para que cada *router* estuviera informado de cuál es la dirección del RP, se ejecutó el siguiente comando en cada *router* de la red:

```
Router(config)#ip pim rp-address 10.0.0.1
```

4) Para definir al grupo *multicast* dentro de una red IP *multicast*, la teoría recomienda elegir una dirección IPv4 clase D no reservada de entre el rango 239.0.0.0 a 239.255.255.255, denominado como un rango con ámbito de Organización Local. Para el grupo de receptores *multicast* del modelo IPv4 *multicast*, se determinó y asignó la dirección IPv4 de grupo *multicast* 239.1.1.1 a aquellas interfaces de los *mrouters* donde existan receptores pertenecientes al grupo *multicast* (en este caso la interfaz FastEthernet2/0 del *router* SLT y las interfaces FastEthernet0/0 de los *routers* Casa Libertad y Cuautepec), ejecutando el comando en los planteles mencionados:

```
Router(config-if)#ip igmp join-group 239.1.1.1
```

La configuración realizada en cada uno de los cinco *routers* del modelo *IPv4 Mcast*, para lograr enrutamiento *multicast* a través de los protocolos PIM-SM e IGMP, se muestra en la tabla IV.4 y IV.5.

Configuración de enrutamiento multicast (PIM-SM e IGMP)	
 SLT	 Cuautepec
SLT(config)#ip multicast-routing SLT(config)#int loop0 SLT(config-if)#ip pim sparse-mode SLT(config-if)#exit SLT(config)#int fa2/0 SLT(config-if)#ip pim sparse-mode SLT(config-if)#ip igmp join-group 239.1.1.1 SLT(config-if)#exit SLT(config)#int s1/0 SLT(config-if)#ip pim sparse-mode SLT(config-if)#exit SLT(config)#int s1/1 SLT(config-if)#ip pim sparse-mode SLT(config-if)#exit SLT(config)#ip pim rp-address 10.0.0.1	Cuautepec(config)#ip multicast-routing Cuautepec(config)#int fa0/0 Cuautepec(config-if)#ip pim sparse-mode Cuautepec(config-if)#ip igmp join-group 239.1.1.1 Cuautepec(config-if)#exit Cuautepec(config)#int s1/0 Cuautepec(config-if)#ip pim sparse-mode Cuautepec(config-if)#exit Cuautepec(config)#int s1/1 Cuautepec(config-if)#ip pim sparse-mode Cuautepec(config-if)#exit Cuautepec(config)#ip pim rp-address 10.0.0.1

Tabla IV.4. Configuración de 5 equipos Cisco con la cual se logra el enrutamiento multicast a través de los protocolos PIM-SM e IGMP.

 Casa Libertad	 Del Valle	 Centro Histórico
CasaLibe(config)#ip multicast-routing CasaLibe(config)#int fa0/0 CasaLibe(config-if)#ip pim sparse-mode CasaLibe(config-if)#ip igmp join-group 239.1.1.1 CasaLibe(config-if)#exit CasaLibe(config)#int s1/0 CasaLibe(config-if)#ip pim sparse-mode CasaLibe(config-if)#exit CasaLibe(config)#int s1/1 CasaLibe(config-if)#ip pim sparse-mode CasaLibe(config-if)#exit CasaLibe(config)#ip pim rp-address 10.0.0.1	CentroHist(config)#ip multicast-routing CentroHist(config)#int s2/0 CentroHist(config-if)#ip pim sparse-mode CentroHist(config-if)#exit CentroHist(config)#int s2/1 CentroHist(config-if)#ip pim sparse-mode CentroHist(config-if)#exit CentroHist(config)#ip pim rp-address 10.0.0.1	Valle(config)#ip multicast-routing Valle(config)#int fa0/0 Valle(config-if)#ip pim sparse-mode Valle(config-if)#exit Valle(config)#int s1/0 Valle(config-if)#ip pim sparse-mode Valle(config-if)#exit Valle(config)#int s1/1 Valle(config-if)#ip pim sparse-mode Valle(config-if)#exit Valle(config)#ip pim rp-address 10.0.0.1

Tabla IV.5. Configuración de los 5 equipos Cisco con la cual se logra el enrutamiento *multicast*.

IV.3.3 Transmisión de audio y video streaming con VLC media player

Para comprobar que el modelo IPv4 *multicast* funcionará de manera adecuada, se realizó la transmisión de un archivo de audio y un archivo de video usando como fuente emisora una máquina virtual Windows XP ubicada en el plantel Del Valle, y tres máquinas virtuales también Windows XP actuando como receptores (ubicados en SLT, Casa Libertad y Cuautepec) del grupo *multicast* **239.1.1.1.**

Para realizar la transmisión y reproducción de audio y video *streaming multicast* se requirió forzosamente que el equipo emisor y los equipos receptores tuvieran instalada una aplicación o *software* compatible entre los equipos con la capacidad de soportar la tecnología *streaming*, la capacidad de transmisión *multicast*, y la capacidad de compatibilidad para diversos estándares y formatos de codificación tanto de audio como de video.

En este sentido, se propuso el *software* VLC *media player* para que el equipo fuente *multicast* (en este caso una fuente en el plantel Del Valle) transmitiera audio y video, así como para que los equipos receptores (ubicados en los planteles SLT, Casa Libertad y Cuautepec) del grupo *multicast* 239.1.1.1 fueran capaces de recibir y reproducir el audio y video *streaming*.

VLC *media player* es una solución de libre descarga con versiones disponibles para diferentes sistemas operativos como Windows (la cual se usó para el emisor y receptores en este proyecto), GNU/Linux, MacOS, BeOS y diferentes MobileOS. Además, VLC permite la transmisión y reproducción de audio y video *streaming multicast* usando protocolos como TCP *unicast*, UDP/RTP *unicast*, UDP/RTP *multicast*, RTCP, y soporte para diversos formatos de audio (como por ejemplo, MP3, ACC, AC3, WMA-1/2/3, FLAC, Wabpack, Real Audio 2) y video (como MPEG-1/2, DIV-1/2/3, MPEG-4, H.263, H.264 AVC, MJPEG-A/B por mencionar algunos). La manera de cómo se realizó la transmisión/recepción de información a través de *multicast*, y en particular los pasos que se siguieron para lograr tanto la transmisión como la reproducción de audio y video *streaming* con lo UDP/RTP *multicast* en el modelo IPv4 *multicast* de la UACM usando VLC, se explica en el apéndice B de este trabajo.

IV.4 Análisis del enrutamiento multicast en el modelo IPv4 Multicast

Con base en la teoría descrita en el capítulo III sobre los protocolos OSPF, IGMPv2 y PIM-SM, se realiza un análisis sobre el enrutamiento *multicast* que se obtuvo en el modelo *IPv4 multicast* de la UACM gracias a los protocolos mencionados, y con ello la ruta que toma el flujo de tráfico en el mismo.

1) Supongamos que el receptor WinXP2 es el primer *host* que desea unirse al grupo 239.1.1.1. Para ello, envía un IGMP *Membership report* a la dirección de grupo.

2) Cuando *R*2 (Casa Libertad) recibe el *Membership report*, inmediatamente crea una entrada (*, 239.1.1.1) en su tabla de enrutamiento *mcast*, y en la que (*) representa cualquier fuente. Así mismo, el *R*2 añade la interfaz por la que recibió el reporte como interfaz de salida para flujo *multicast*. Y por último, el mismo *R*2 envía un PIM Join/Prune al *R*1 (RP San Lorenzo Tezonco, interfaz 10.0.0.1 definida estáticamente).

3) Cuando *R*1 (RP) recibe el mensaje PIM/Join (*, 239.1.1.1) crea una entrada (*, 239.1.1.1), igual y como lo hizo *R*2 en su tiempo. También el *R*1 (RP) añade la interfaz por la que recibió el mensaje PIM Join/Prune como interfaz de salida para flujo *mcast.* Hasta aquí, se ha construido un árbol compartido ST entre el receptor WinXP2- *R*2- *R*1 (RP).

La figura IV.1.3 es una representación gráfica de las tres primeras acciones descritas con anterioridad para demostrar como el receptor WInXP2 se une al grupo multicast 239.1.1.1 a través de un ST.

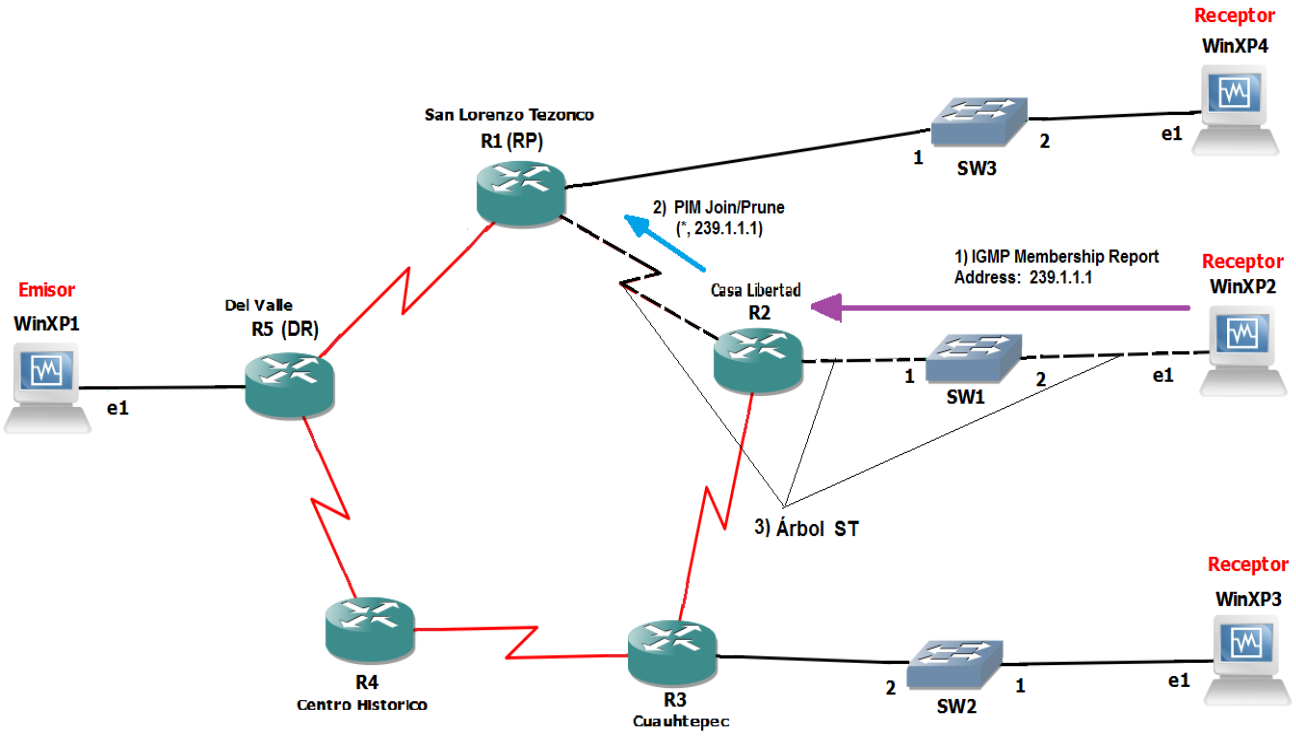


Figura IV.1.3. Creación de árbol ST entre un receptor y un RP [Diagrama propio].

Ahora supongamos que el siguiente en unirse al grupo de multidifusión 239.1.1.1 es el receptor WinXP4, el proceso sería básicamente el mismo que los pasos descritos con anterioridad. Sin embargo, en éste caso en particular el *R*1 (RP) ya tiene una entrada en su tabla de enrutamiento *mcast* (*, 239.1.1.1); por lo que al recibir el reporte el *R*1 (RP) por parte del *host* WinXP4 en seguida se crea el árbol compartido ST entre WinXP4- *R*1 (RP).

Finalmente, cuando el host WinXP3 se una al grupo 239.1.1.1, enviará IGMP *Report* al *R*3 (Cuautepec). Éste último recibirá el reporte; creará una entrada en su tabla *mcast* (*, 239.1.1.1), integrará la interfaz por la que recibió el reporte como interfaz de salida para flujo *mcast,* y enviará un mensaje PIM Join al *R*1 (RP). El mensaje PIM Join llegará al *R*2 y *R*4 (Centro Histórico). El *R2* que ya cuenta con una entrada (*, 239.1.1.1) y pertenece a un ST, simplemente añadirá la interfaz que conecta a *R*3 a la lista de interfaces de salida para (*, 239.1.1.1), creándose inmediatamente el árbol compartido entre WinXP3- *R*3- *R*2- *R*1 (RP). Mientras que *R*4 únicamente creará una entrada (*, 239.1.1.1) en su tabla de enrutamiento *mcast*.

4) Asumamos que ahora la fuente emisora de multidifusión WinXP1 comienza a transmitir flujo al grupo 239.1.1.1. El primero en recibir el flujo será el *R5 (DR* Del Valle), por lo que encapsulará el flujo en un mensaje *Register* y lo enviará por unidifusión al *R*1 (RP) usando la ruta más corta (con ayuda del protocolo OSPF que define la ruta más corta hacia la dirección 10.0.0.1).

5) Cuando *R1 (*RP) reciba el mensaje PIM *Register*, des-encapsulará el mensaje y se dará cuenta que el flujo está dirigido hacia el grupo 239.1.1.1 desde la fuente 172.30.1.1, es decir, una entrada (172.30.1.1, 239.1.1.1). El *R1 (RP)* al contar con el mismo grupo activo 239.1.1.1 que construye un árbol ST, se dará a la tarea de enviar un mensaje PIM Join con la misma entrada (172.30.1.1, 239.1.1.1) y por la misma ruta a *R*5 (DR). Creándose de esta manera un árbol de camino más corto SPT entre WinXP1-*R*5 (DR)-*R*1 (RP), de tal forma como se muestra en la figura IV.1.4.

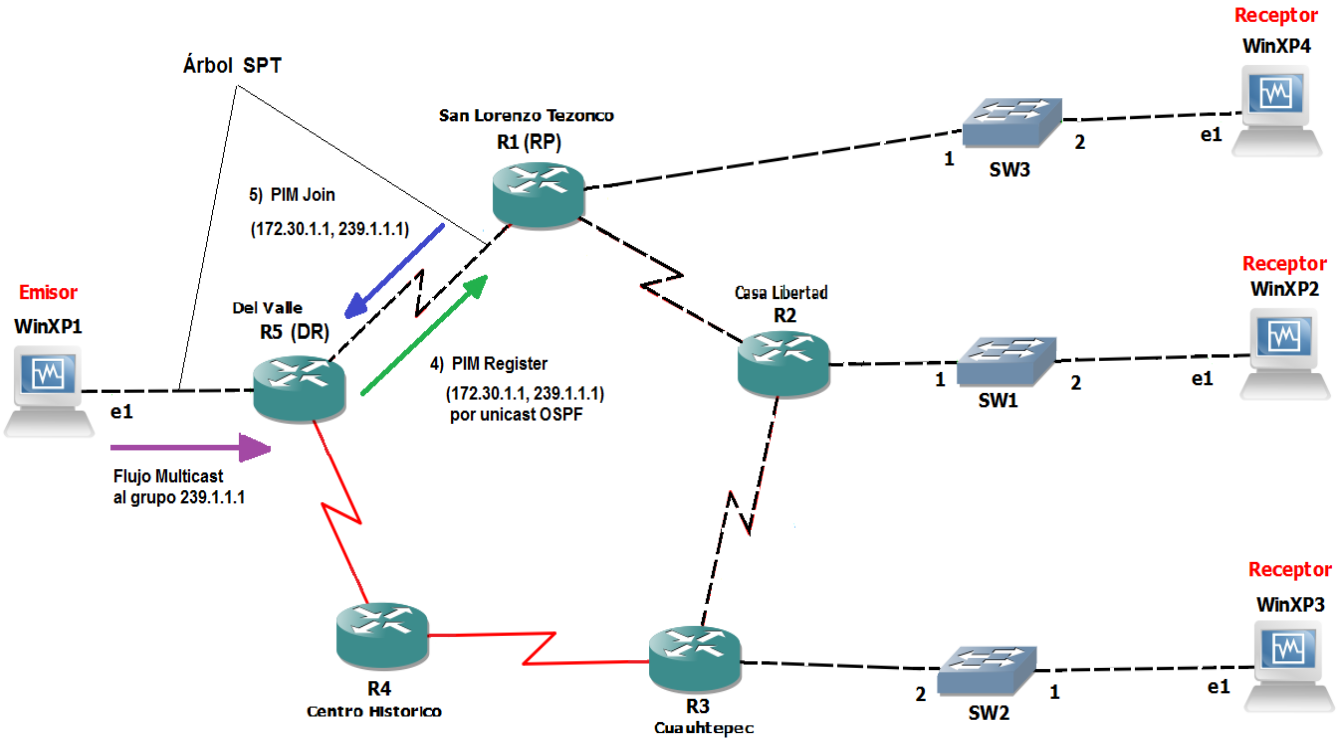


Figura IV.1.4. Creación de árbol STP entre una y un RP [Diagrama propio].

6) Una vez que se ha creado el SPT entre WinXP1-*R*5 (DR)-*R*1 (RP) el flujo *multicast* comenzará a fluir desde la fuente emisora hasta el *R*1 (RP), por lo que éste último ya no necesitará más recibir mensaje *Register*. Así que *R1* (RP) mediante *unicast* enviará un mensaje PIM *Register-Stop* al *R*5 (DR) para que pare de enviar.

Ahora el flujo *multicast* emitido por WinXP1 llegará a R5 (DR) hasta R1 (RP) a través de un árbol STP. Y desde R1 (RP) a los *routers* R2 y R3, hasta llegar al grupo *multicast* 239.1.1.1 integrado por los receptores WinXP2, WinXP3 y WinXP4 a través de un ST. Dando como resultado que todos los *routers* conozcan la dirección de la fuente emisora. Como se ilustra en la figura IV.1.5.

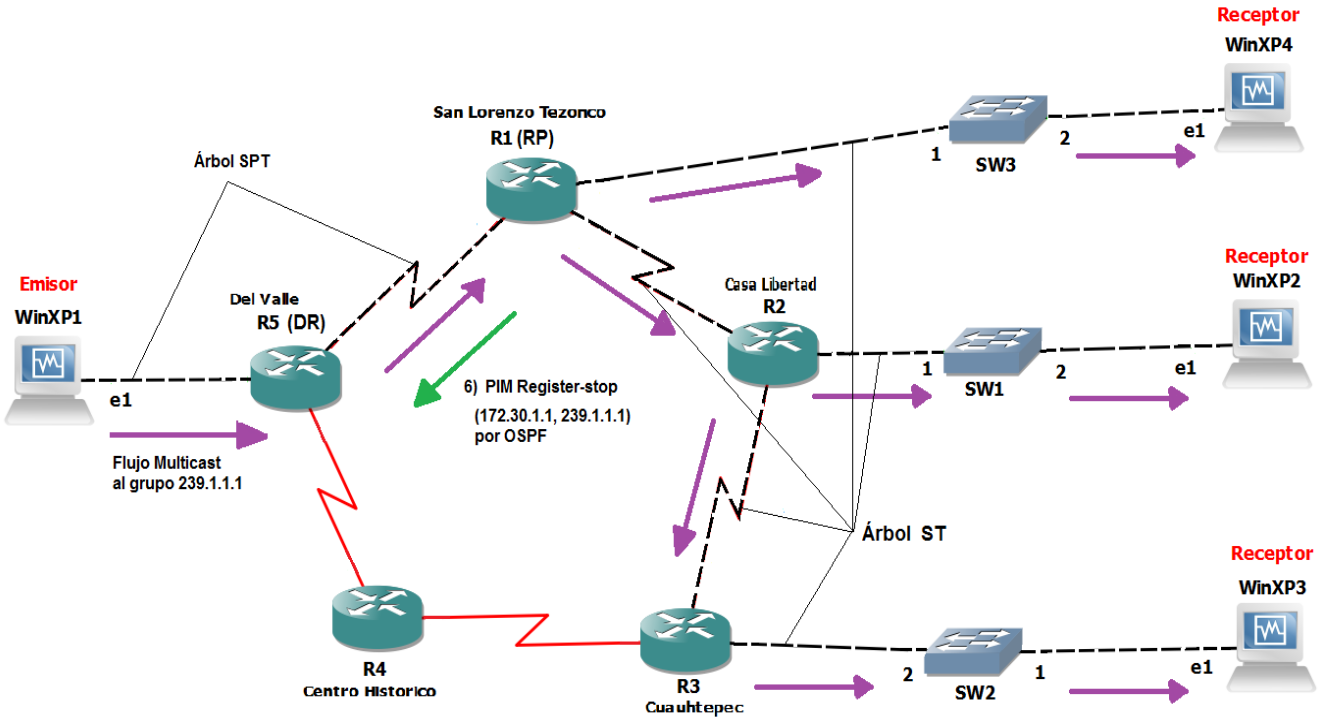


Figura IV.1.5. Creación de árbol STP entre una y un RP [Diagrama propio].

En este sentido, recordemos que los equipos Cisco configurados con el protocolo PIM-SM permiten pasar de un árbol ST a un árbol SPT para que la ruta de flujo *multicast* sea la óptima (ruta más corta) entre emisor y el grupo mcast. Por tanto, R2 y R3 cambiarán a un árbol de ruta más corta SPT de la siguiente manera:

7) R2 enviará un mensaje PIM Join con la entrada (172.30.1.1, 239.1.1.1) a R5 (DR) a través de la mejor ruta definida por OSPF. Una vez que el mensaje PIM Join llegue a R5 (DR), éste último agregará la interfaz por la que recibió la el mensaje PIM Join como interfaz de salida para para flujo *multicast*. Esto da como resultado que la interfaz desde R2, que conecta a R1 (RP) ahora sea parte del árbol SPT creado por WinXP1, R1 (RP) y R5 (DR). Lo mismo ocurrirá con R3; sin embargo, éste se unirá al árbol SPT formado por WinXP1, R1 (RP) y R5 (DR) a través de su interfaz que lo conecta con R4 hasta llegar a R5 (DR).

8) Una vez que R2, y R3 se hayan unido al árbol SPT, estos mismos *routers* enviarán un mensaje PIM Prune para podar aquellas rutas que se unen a los árboles ST. Quedando únicamente el árbol SPT (líneas discontinuas) desde cada receptor hacia la fuente a través de la mejor ruta, tal y como se muestra en la figura IV.1.6.

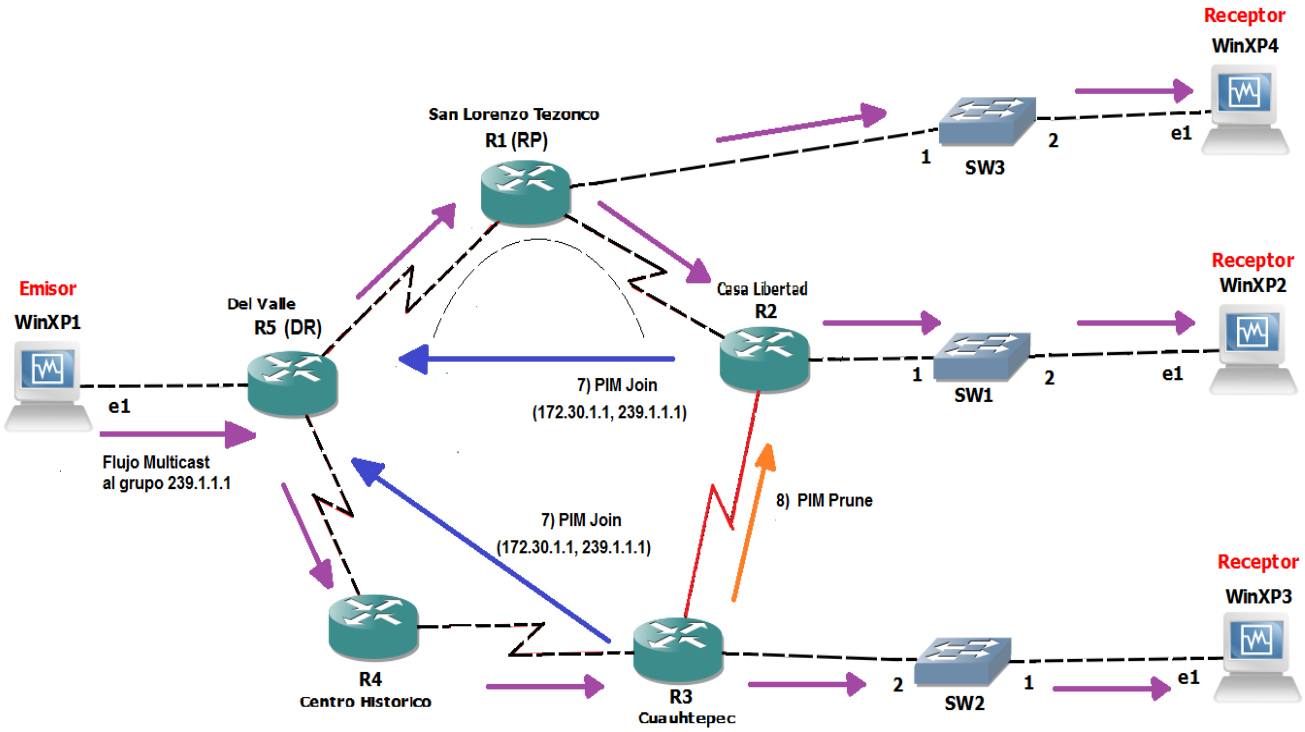


Figura IV.1.6. Cambio de un ST a STP realizado a través del protocolo PIM-SM [Diagrama propio].

Capítulo V

Resultados y Conclusiones

A continuación se presentan los resultados más significativos que se obtuvieron al configurar y emular el modelo *IPv4 multicast*, y del uso de este último para realizar la transmisión de audio y video *streaming* a un grupo de receptores.

En general, se logró con éxito configurar y emular el modelo *IPv4 multicast*, así como transmitir audio y video *streaming* haciendo uso de GNS3. Sin embargo, se pudo observar que hubo una variación del consumo tanto del procesador como de la memoria RAM de la máquina física real en el momento en que se corrió la emulación y se emitió primero un archivo de audio únicamente, y posteriormente, cuando se emitió un archivo de video.

V.1 Resultados de la configuración de enrutamiento en el modelo IPv4 Multicast

Con el objeto de mostrar el resultado de la configuración que se realizó en el capítulo IV del enrutamiento *unicast* OSPF, del enrutamiento *multicast* PIM-SM, y de las direcciones IPv4 asignadas a cada interfaz activa en cada uno de los cinco *routers*, se ejecutó el comando **show run** en cada uno de los mismos**.** Comando útil que nos muestra información sobre la configuración realizada en los equipos.

La información resaltada de las tablas V.1 y V.2 muestra la configuración del enrutamiento *unicast* OSPF que se realizó con éxito. Mientras que en las tablas V.3 y V.4 se resalta la información que muestra el resultado de haber logrado la configuración de enrutamiento *multicast.*

R1 SLT	R2 Casa Libertad
SLT#sh run Building configuration... Current configuration : 1553 bytes ! version 12.4 hostname SLT ! interface Loopback0 ip address 200.10.1.1 255.255.255.0 ! interface FastEthernet2/0 ip address 192.170.3.254 255.255.255.0 duplex full ! interface Serial1/0 ip address 10.0.0.1 255.255.255.252 serial restart-delay 0 ! interface Serial1/1 ip address 100.0.0.1 255.255.255.252 serial restart-delay 0 ! router ospf 1 network 10.0.0.0 0.0.0.3 area 0 network 100.0.0.0 0.0.0.3 area 0 network 192.170.3.0 0.0.0.255 area 0 ! end	CasaLibertad#sh run Building configuration... Current configuration : 1678 bytes ! version 12.4 ! hostname CasaLibertad ! interface FastEthernet0/0 ip address 192.168.1.254 255.255.255.0 duplex full speed 100 ! interface Serial1/0 ip address 10.0.0.2 255.255.255.252 serial restart-delay 0 ! interface Serial1/1 ip address 11.0.0.1 255.255.255.252 serial restart-delay 0 ! router ospf 1 network 10.0.0.0 0.0.0.3 area 0 network 11.0.0.0 0.0.0.3 area 0 network 192.168.1.0 0.0.0.255 area 0 ! end

Tabla V.1. Muestra el resultado de realizar la configuración del enrutamiento unicast OSPF, además de las direcciones IPv4 asignadas a cada interfaz en los routers San Lorenzo Tezonco y Casa Libertad.

R3 Cuautepec	R4 Centro Histórico	R5 Del Valle
Cuauhtepec#sh run Building configuration... Current configuration : 1558 bytes ! version 12.4 ! hostname Cuauhtepec ! interface FastEthernet0/0 ip address 172.16.1.254 255.255.255.0 duplex full speed 100 ! interface Serial1/0 ip address 11.0.0.2 255.255.255.252 serial restart-delay 0 ! interface Serial1/1 ip address 101.0.0.1 255.255.255.252 serial restart-delay 0 ! router ospf 1 network 11.0.0.0 0.0.0.3 area 0 network 101.0.0.0 0.0.0.3 area 0 ! end	CentroHistorico#sh run Building configuration... Current configuration : 1364 bytes ! version 12.4 ! hostname CentroHistorico ! interface Serial2/0 ip address 101.0.0.2 255.255.255.252 serial restart-delay 0 ! interface Serial2/1 ip address 1.1.1.1 255.255.255.252 serial restart-delay 0 ! router ospf 1 network 1.1.1.0 0.0.0.3 area 0 network 101.0.0.0 0.0.0.3 area 0 ! end	Del_Valle#sh run Building configuration... Current configuration : 1364 bytes ! version 12.4 ! hostname Del_Valle ! interface FastEthernet0/0 ip address 172.30.1.254 255.255.255.0 duplex full speed 100 ! interface Serial1/0 ip address 100.0.0.2 255.255.255.252 serial restart-delay 0 ! interface Serial1/1 ip address 1.1.1.2 255.255.255.252 serial restart-delay 0 ! router ospf 1 network 100.0.0.0 0.0.0.3 area 0 network 1.1.1.0 0.0.0.3 area 0 network 172.30.1.0 0.0.0.255 area 0 ! end

Tabla V.2. Muestra el resultado de realizar la configuración del enrutamiento unicast OSPF, además de las direcciones IPv4 asignadas a cada interfaz para los routers Cuautepec, Centro Histórico y del Valle.

R1 SLT	R2 Casa Libertad
SLT#sh run hostname SLT ip multicast-routing interface Loopback0 ip address 200.10.1.1 255.255.255.0 ip pim sparse-mode ! interface FastEthernet2/0 ip address 192.170.3.254 255.255.255.0 ip pim sparse-mode ip igmp join-group 239.1.1.1 duplex full ! interface Serial1/0 ip address 10.0.0.1 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! interface Serial1/1 ip address 100.0.0.1 255.255.255.252 ip pim sparse-mode serial restart-delay 0 router ospf 1 network 10.0.0.0 0.0.0.3 area 0 network 100.0.0.0 0.0.0.3 area 0 network 192.170.3.0 0.0.0.255 area 0 ! ip pim rp-address 10.0.0.1 ! end	CasaLibertad#sh run hostname CasaLibertad ! ip multicast-routing ! interface FastEthernet0/0 ip address 192.168.1.254 255.255.255.0 ip pim sparse-mode ip igmp join-group 239.1.1.1 duplex full speed 100 ! interface Serial1/0 ip address 10.0.0.2 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! interface Serial1/1 ip address 11.0.0.1 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! router ospf 1 network 10.0.0.0 0.0.0.3 area 0 network 11.0.0.0 0.0.0.3 area 0 network 192.168.1.0 0.0.0.255 area 0 ! ip pim rp-address 10.0.0.1 ! end

Tabla V.3. Muestra el resultado de realizar la configuración del enrutamiento multicast para los routers San Lorenzo Tezonco y Casa Libertad.

R3 Cuautepec	R4 Centro Histórico	R5 Del Valle
Cuauhtepec#sh run hostname Cuauhtepec ! ip multicast-routing ! interface FastEthernet0/0 ip address 172.16.1.254 255.255.255.0 ip pim sparse-mode ip igmp join-group 239.1.1.1 duplex full speed 100 ! interface Serial1/0 ip address 11.0.0.2 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! interface Serial1/1 ip address 101.0.0.1 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! router ospf 1 network 11.0.0.0 0.0.0.3 area 0 network 101.0.0.0 0.0.0.3 area 0 network 172.16.1.0 0.0.0.255 area 0 ! ip pim rp-address 10.0.0.1	CentroHistorico#sh run hostname CentroHistorico ! ip multicast-routing ! interface Serial2/0 ip address 101.0.0.2 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! interface Serial2/1 ip address 1.1.1.1 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! router ospf 1 network 1.1.1.0 0.0.0.3 area 0 network 101.0.0.0 0.0.0.3 area 0 ! ip pim rp-address 10.0.0.1 ! end	Del_Valle#sh run hostname Del_Valle ! ip multicast-routing ! interface FastEthernet0/0 ip address 172.30.1.254 255.255.255.0 ! interface Serial1/0 ip address 100.0.0.2 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! interface Serial1/1 ip address 1.1.1.2 255.255.255.252 ip pim sparse-mode serial restart-delay 0 ! router ospf 1 network 100.0.0.0 0.0.0.3 area 0 network 1.1.1.0 0.0.0.3 area 0 network 172.30.1.0 0.0.0.255 area 0 ! ip pim rp-address 10.0.0.1 ! end

Tabla V.4. Tabla V.3. Muestra el resultado de realizar la configuración del enrutamiento multicast para los routers Cuautepec, Centro Histórico y del Valle.

V.2 Resultados de la emulación al transmitir audio Streaming

Como resultado de emitir flujo de audio streaming en el modelo IPv4 multicast, se observó que el consumo de procesamiento y memoria RAM en la computadora física real y requerido por el emulador GNS3 alcanzó niveles máximos de 65% de procesamiento y 6.38GB de un total de 8GB, tal y como se muestra en la figura V.1. A pesar de estos niveles requeridos por el emulador, se logró reproducir el audio de manera simultánea por tres máquinas virtuales receptoras del modelo de red sin retrasos ni perdidas perceptibles de flujo a nivel de usuario. Las pérdidas y retrasos de tráfico son medibles cuantitativamente y cualitativamente, sin embargo, se requeriría incorporar al emulador algún *software* capaz de realizar dichas mediciones.

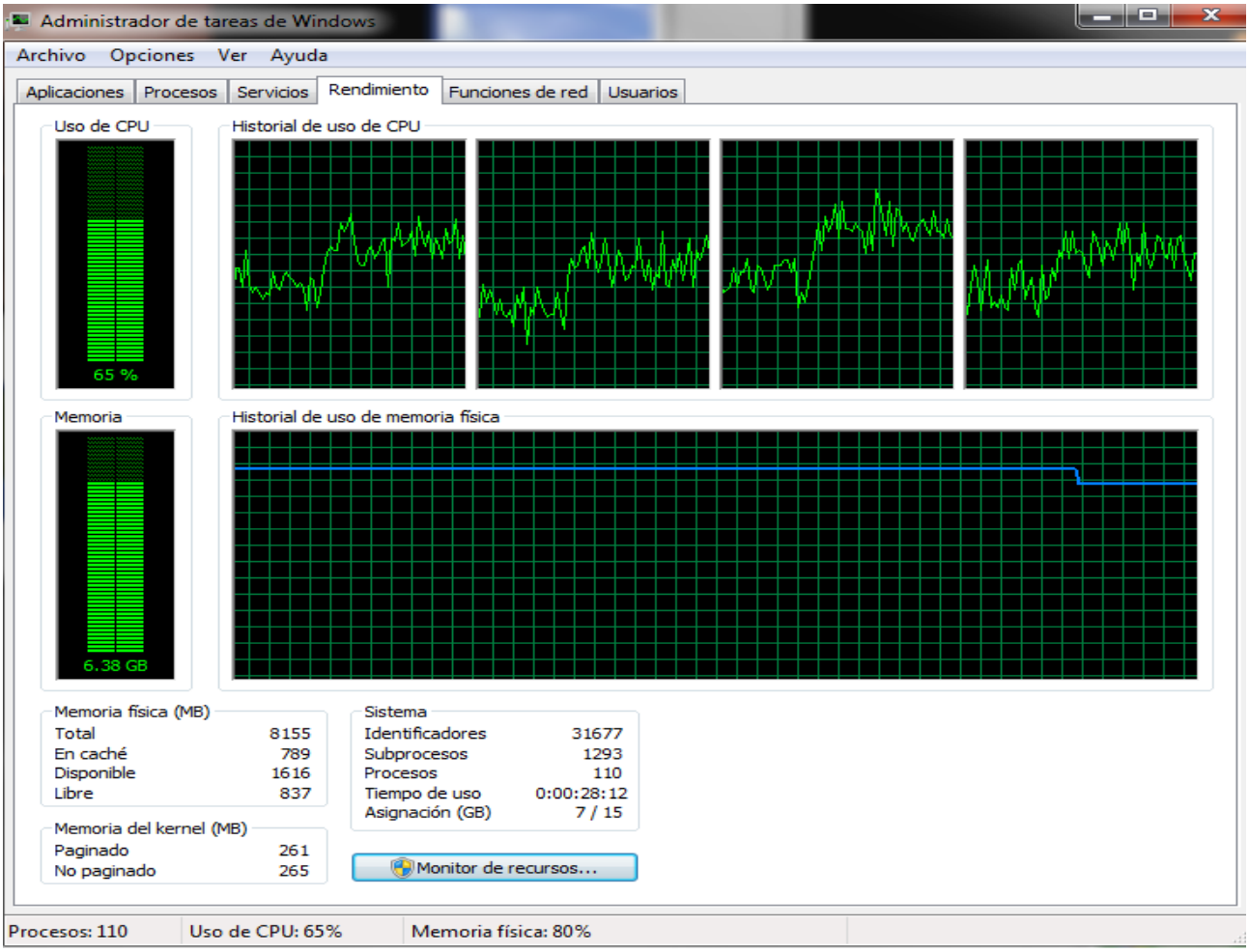


Figura V.1. Porcentaje y cantidad de memoria RAM de una máquina física y requerida por la emulación y transmisión de audio streaming en GNS3.

V.3 Resultados de la emulación al transmitir video Streaming

De la misma manera que se realizó en audio, en la emisión de video se pudo observar, como resultado, que el consumo en el equipo de cómputo físico de procesamiento y memoria RAM requerido por GNS3 alcanzó un máximo de entre 99% y 100% para el procesamiento, y llegó a un máximo de consumo de 7.02GB de un total de 8GB de RAM; tal y como se muestra en la figura V.2. Dicho procesamiento y consumo de memoria influyeron de manera significativa para que en la reproducción de video realizada por las tres máquinas virtuales receptoras del modelo de red se percibieran a nivel de usuario retrasos y pérdidas de la secuencia del flujo de video. Sin embargo, no se descarta que aspectos relacionados con la cantidad de consumo de BW, el tipo de codificación de video, o la cantidad de *jitter* que se presentan al transmitir un flujo de video, pudieron influir en los retardos y pérdidas que se presentaron.

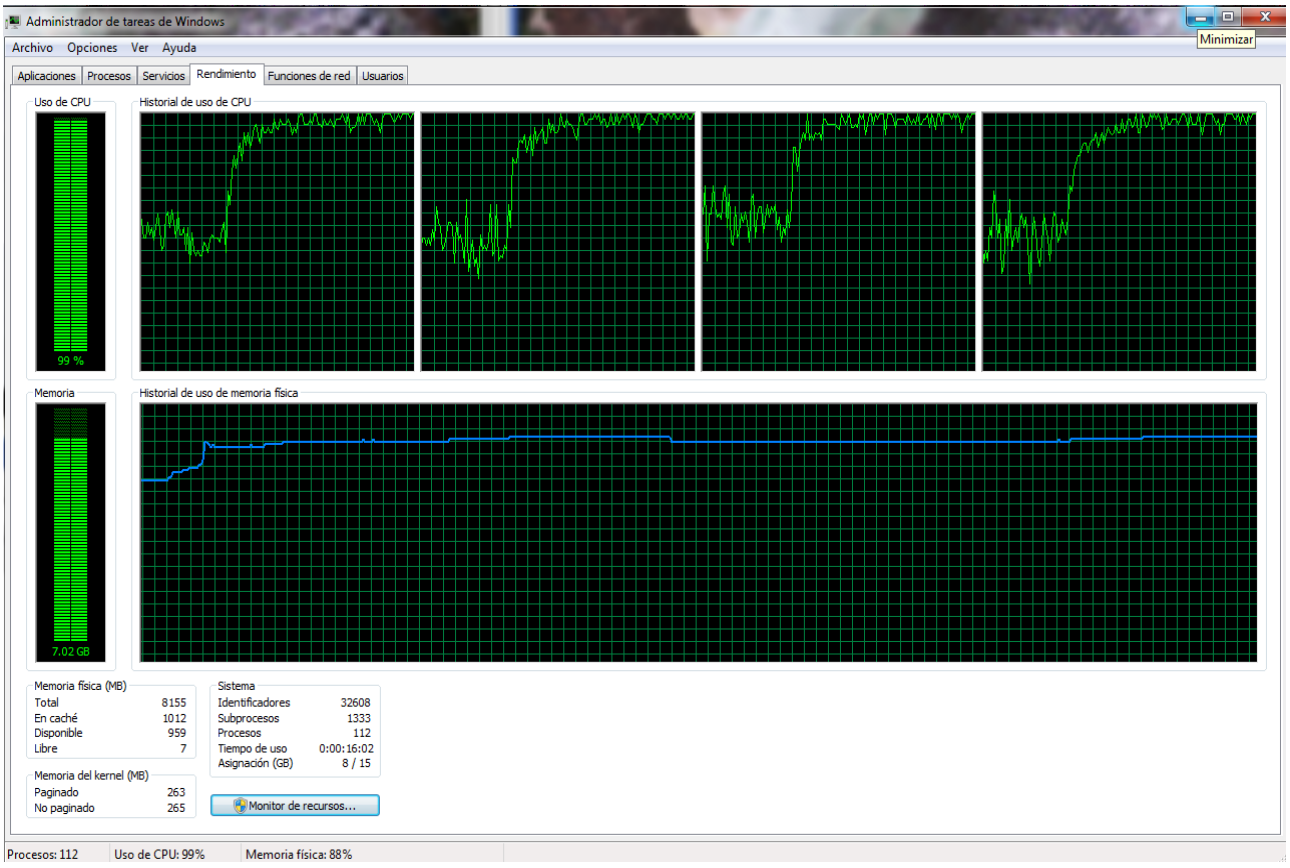


Figura V.2. Porcentaje y cantidad de memoria RAM de una máquina física requerida por la emulación y transmisión de video Streaming en GNS3.

Realizar una medición cuantitativa del consumo del BW y de la cantidad de *jitter* requeriría un *software* capaz de realizar dichas mediciones. Por otra parte, si se realizará una medición cualitativa en términos de QoE (Quality of Experience), donde uno representara el nivel más bajo de calidad y 5 el nivel más alto; a la reproducción de video en el modelo de red se le daría un nivel 3.

En general como resultados se logró transmitir audio y video streaming a través del modelo de red IPv4 multicast, con lo que se cumple con una parte del objetivo planteado al comienzo del documento. Sin embargo, se hace énfasis en que los aspectos relacionados con mejorar la calidad en que se visualizó el archivo de video transmitido no se trataron en el presente trabajo. No obstante, sería un excelente tema para trabajar en un futuro.

Por otra parte, recordemos que todo proceso que se ejecute en un equipo (en este caso una PC) independientemente del SO que tenga instalado, requiere hacer uso de cierto nivel de procesamiento (del procesador y sus núcleos), así como de un espacio de memoria RAM. En este sentido, los procesos que se llevaron a cabo durante la emulación con GNS3 del modelo en su totalidad y durante la transmisión de audio y video *streaming* fueron:

- El de los 5 *routers* ejecutando diferentes versiones de IOS reales, protocolos de enrutamiento *unicast* y *multicast* configurados, así como gestionando tráfico de audio y video.
- El de 1 máquina virtual como responsable de gestionar) y transmitir audio y video *streaming* (a través de VLC, más tres máquinas virtuales gestionando y reproduciendo (a través de VLC) el flujo de audio y video *streaming*.

Con base en el desempeño que se obtuvo al momento de realizar las pruebas, emular el modelo *IPv4 multicast*, y de transmitir audio y video *streaming* con GNS3; se hace la recomendación de considerar las características principalmente de procesador y memoria RAM presentadas a continuación, como mínimas necesarias para obtener un funcionamiento aceptable de la emulación del modelo aquí propuesto, y evitar la saturación de los recursos propios del equipo físico; de lo contrario, es muy probable que se presenten problemas de desempeño y funcionalidad del *software* emulador, y en consecuencia del funcionamiento del modelo de red propuesto.

La instalación de GNS3 se realizó en una computadora con las siguientes características de sistema:

Windows 7 Home Premium

Fabricante: Dell

Procesador: Intel (R) Core (TM) i5-3450 CPU @ 3.10GHz

Memoria Instalada: (RAM): 8.00GB

Capacidad de Disco duro: 1TB

Tipo de Sistema: Sistema operativo de 64 bits

V. 4 Captura del tráfico antes y durante la transmisión de audio y video Streaming

Haciendo uso del analizador de protocolos *Wireshark* (*sniffer*) que integra GNS3 como herramienta, fue posible visualizar y capturar los protocolos de comunicación UDP y RTP, así como los diferentes mensajes que usan los protocolos de enrutamiento OSPF, IGMPv2, y PIM-SM. Con ello, se logró como resultado validar el funcionamiento del modelo de red *IPv4 multicast*, y observar el comportamiento (en cuanto al flujo de mensajes) que se presentó dentro del modelo antes y durante la transmisión de audio-video *streaming*.

Antes de la transmisión de flujo *streaming* (audio, video), se pudieron capturar y analizar los mensajes que utiliza el protocolo de enrutamiento OSPF. Con lo que se comprobó el funcionamiento correcto del enrutamiento *unicast* en el modelo. En la figura V.3 se muestran los mensajes OSPF capturados.

Destacan el mensaje de Saludo (*Hello Packet*), el mensaje de descripción de base de datos (DB *Description*), el mensaje de actualización de estado de enlace (LS *Update*) y el mensaje de acuse de respuesta de estado de enlace (LS *Acknowledge*). Todos ellos descritos con anterioridad en el apartado III.1.1 del capítulo III. Por lo que únicamente se rescata la concordancia de la dirección *multicast* 224.0.0.5 (dirección mcast para *routers* OSPF) a la que se envían los mensajes; así como la correspondencia de la información que se muestra en el encabezado de OSPF de la figura con la descrita en la teoría.

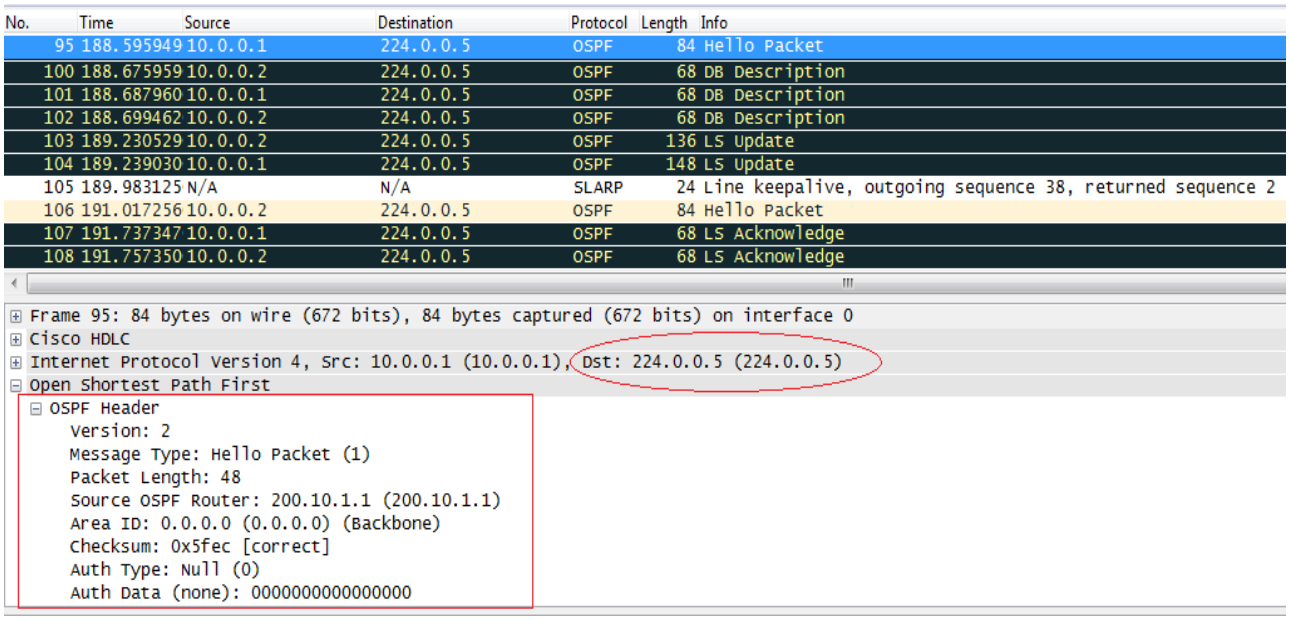


Figura V.3. Mensajes OSPF capturados a través de Wireshark.

Antes de la transmisión de audio y video *streaming*, también se analizaron los mensajes *Membership Query general* y *Membership Report* que utiliza el protocolo IGMPv2. Recordemos que se explicaron con anterioridad en el apartado III. 3; donde se hizo énfasis en su importancia debido a que a través de estos mensajes se logra el anuncio y deseo de pertenencia de los receptores a un grupo de multidifusión.

La figura V.4 nos revela que el mensaje *Membership Query general* (0x11) está activo en la red, y ello da como resultado que los *routers* conectados directamente con los receptores (*routers* de último salto) tengan información acerca del estado de existencia de cualquier grupo *multicast* presente en la red, ya que utiliza la dirección de grupo 0.0.0.0 (como se indica con un círculo en la figura V. 4) y además los mensajes se envían a la dirección *multicast* 224.0.0.1 (para todos los sistemas en una subred local).

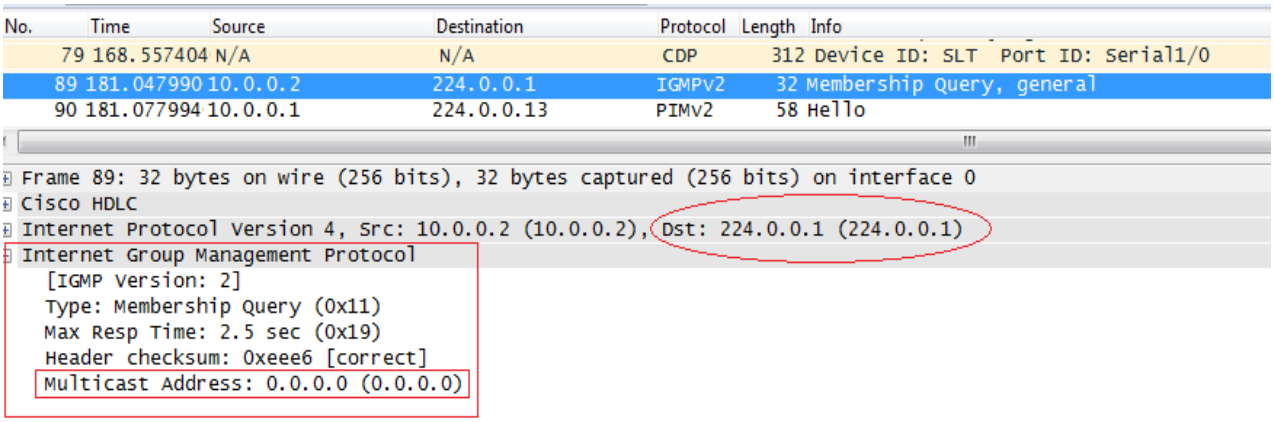


Figura V.4. Mensaje de tipo Membership Query general que incorpora el protocolo IGMPv2.

En la figura V.5 se observa el mensaje *Membership report group* (0x16) que envían los sistemas receptores de los planteles SLT, Casa Libertad y Cuautepec para unirse al grupo *multicast* 239.1.1.1 e indicar al mismo tiempo a los *routers* de último salto que pertenecen a ese grupo *multicast.* Con ello, se verificó la correcta configuración que se hizo del modelo *IPv4 multicast* para incluir un grupo de multidifusión capaz de recibir tráfico.

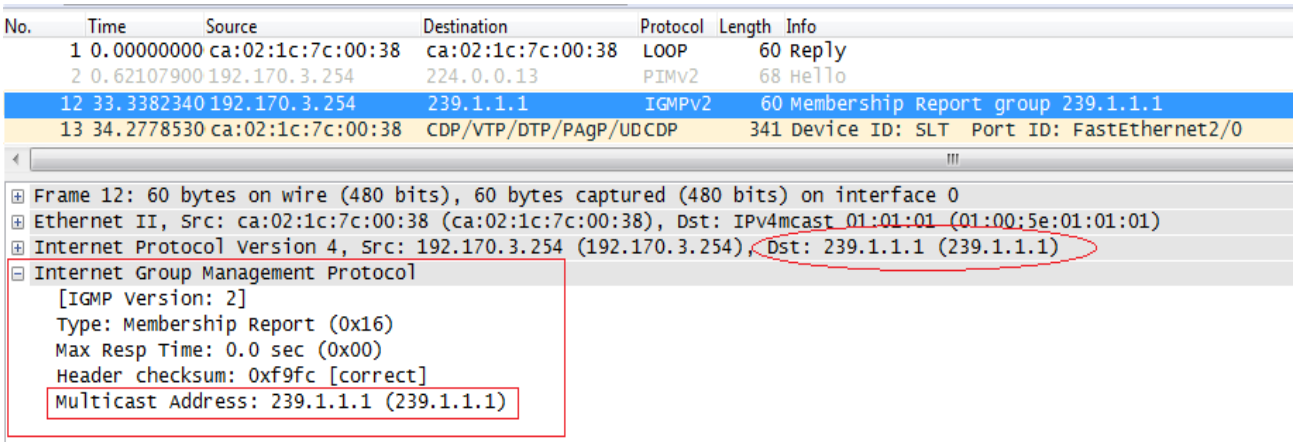


Figura V.5. Mensaje de tipo *Membership Report group* que incorpora el protocolo IGMPv2.

Se descubrió que los receptores también enviaban un *Membership report group* reportando la pertenencia al grupo 239.255.255.250. Está dirección *multicast* es la que utiliza el protocolo SSDP (*Simple Service Discovery Protocol*); un protocolo de red que utilizan los equipos (*hosts*) para el descubrimiento de servicios de red, en este caso el de *streaming*.

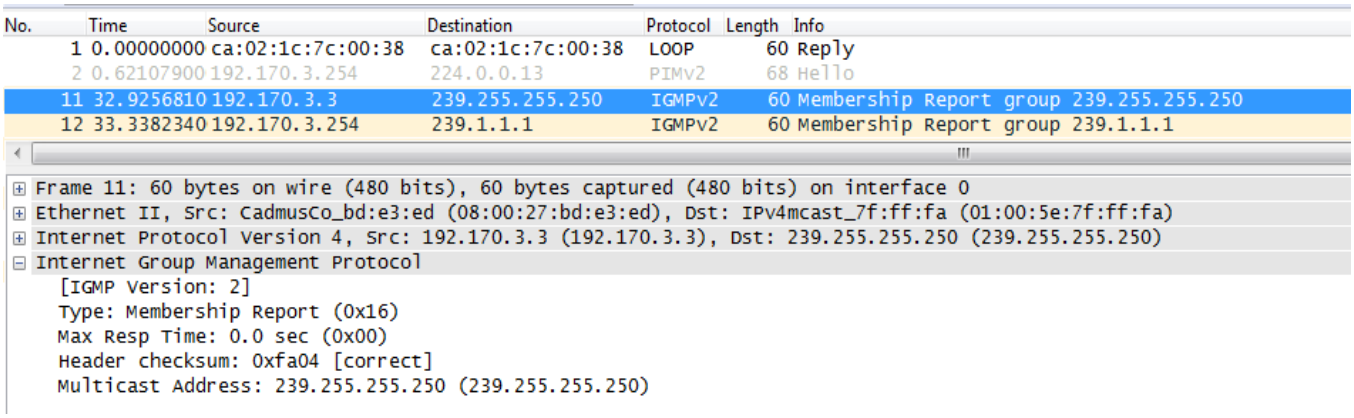


Figura V.6. Mensaje de tipo *Membership Report group* que incorpora el protocolo IGMPv2.

Se comprobó la existencia de 3 mensajes usados por el protocolo PIM-SM. Estos mensajes que se logró capturar fueron:

El mensaje PIM-SM *Hello* mostrado en la figura V.7, y con el cual fue posible que los *routers* se reconocieran entre sí.

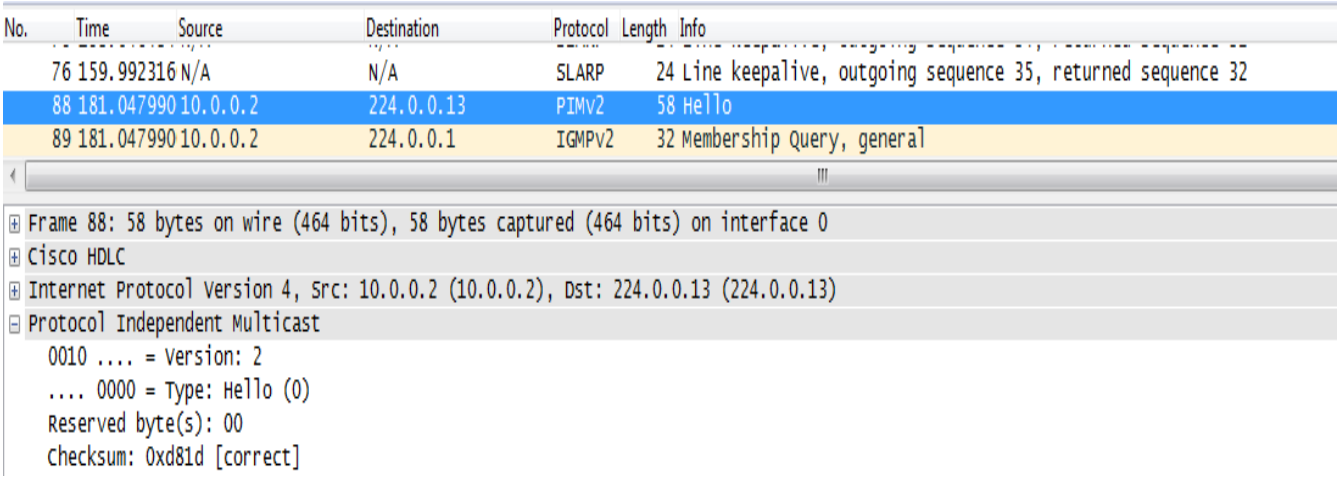


Figura V.7. Mensaje de tipo *Hello* que incorpora el protocolo PIM-SM para su funcionamiento.

Los mensajes de tipo PIM *Join/Prune* de la figura V.8. Con los cuales se crearon entrada en las tablas *multicast* de los *routers* para posteriormente unirse grupos de multidifusión a través de ST's o SPT's. Se observó la concordancia con la dirección 224.0.0.13 a la que se envían este tipo de mensajes así como el de PIM *Hello*.

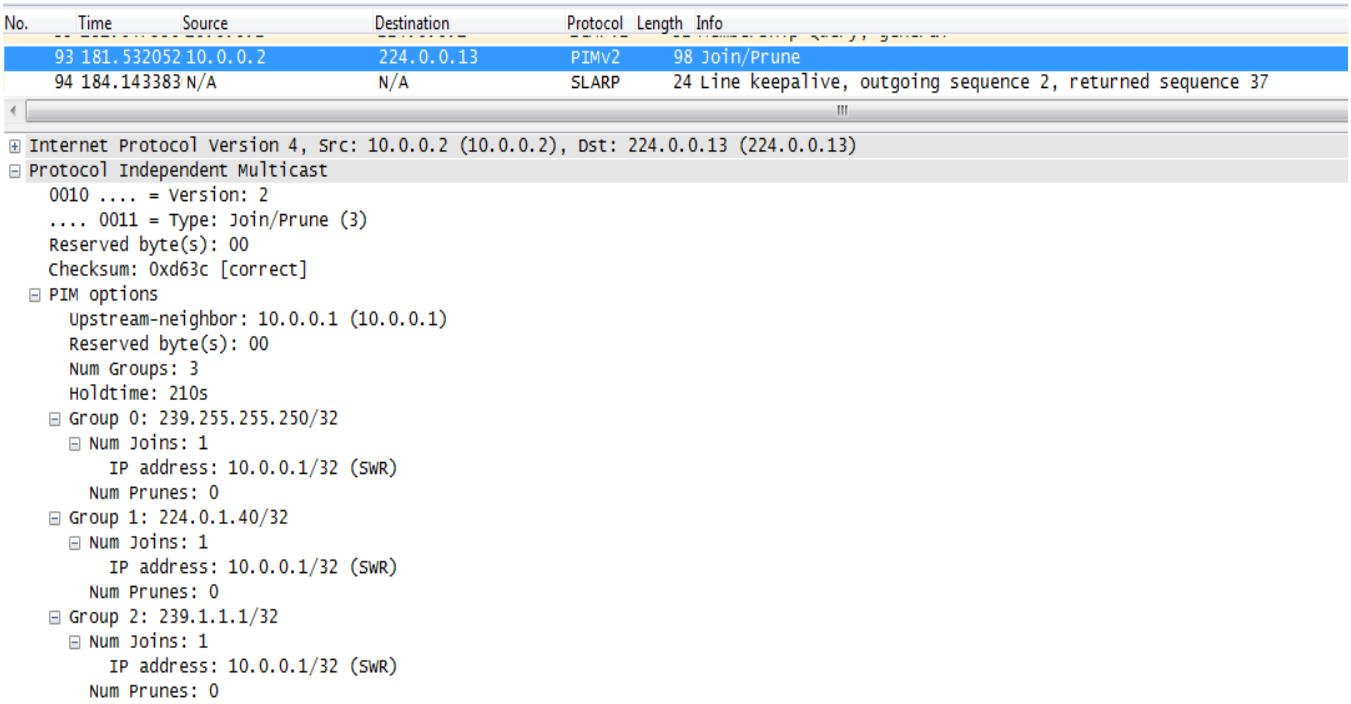


Figura V.8. Mensaje de tipo *Join/Prune* que incorpora el protocolo PIM-SM.

Los mensajes PIM *Register* y PIM *Register-stop* de las figuras V.9 y V.10 respectivamente. Resultado de ello, que la fuente 239.1.1.1 se pudiera unir al *router* RP definido estáticamente durante la configuración del enrutamiento multicast.

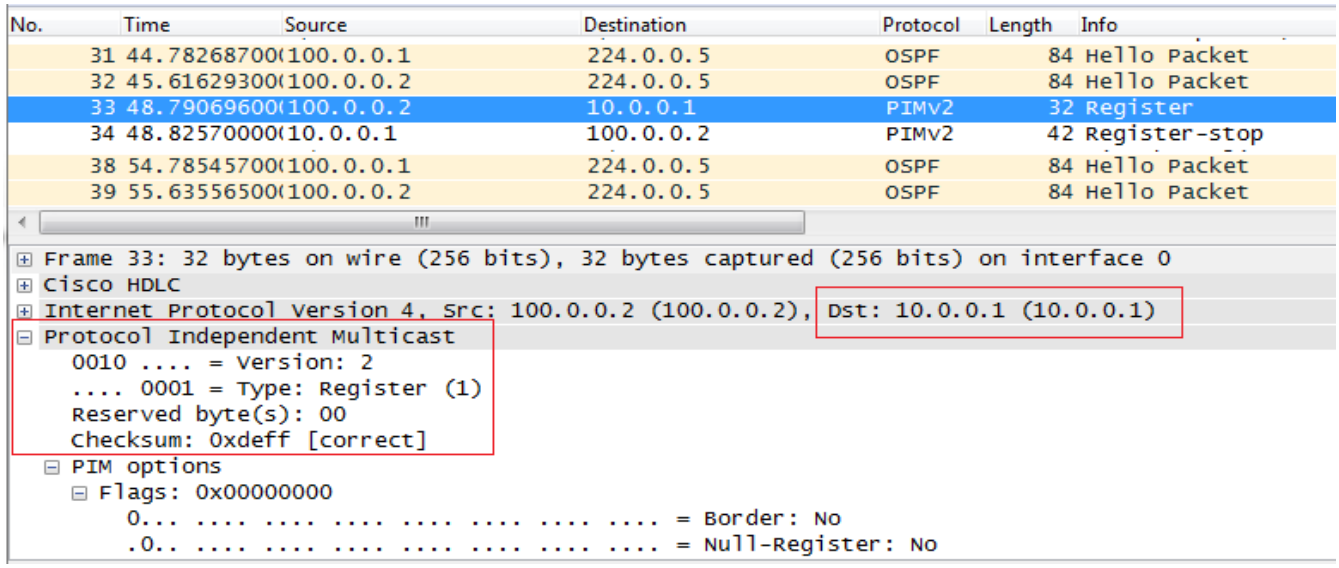


Figura V.9. Mensaje de tipo *Register* que incorpora el protocolo PIM-SM. Se destaca la dirección de destino 10.0.0.1, la cual pertenece al router RP.

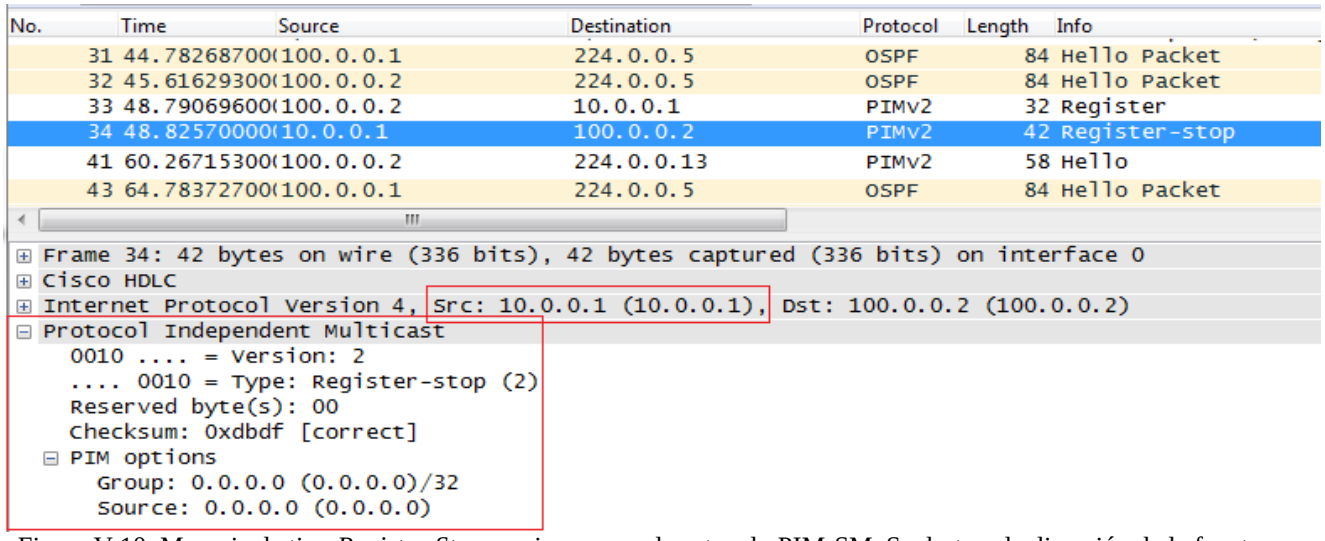


Figura V.10. Mensaje de tipo *Register-Stop* que incorpora el protocolo PIM-SM. Se destaca la dirección de la fuente que envía este mensaje, la cual es 10.0.0.1, que representa al router RP.

Durante la transmisión de audio y video *streaming* se pudo observar que el tráfico generado por el mismo se realizaba a través del protocolo UDP y RTP. Lo cual nos da como resultados que la configuración y transmisión mediante VLC se haya realizado de manera exitosa. Además, destacan en la figura V.11 y V.12 la dirección de *multicast* a la cual se transmitió el flujo de datos (audio y video), el protocolo UDP y RTP, así como el puerto 5004 que utiliza el protocolo RTP.

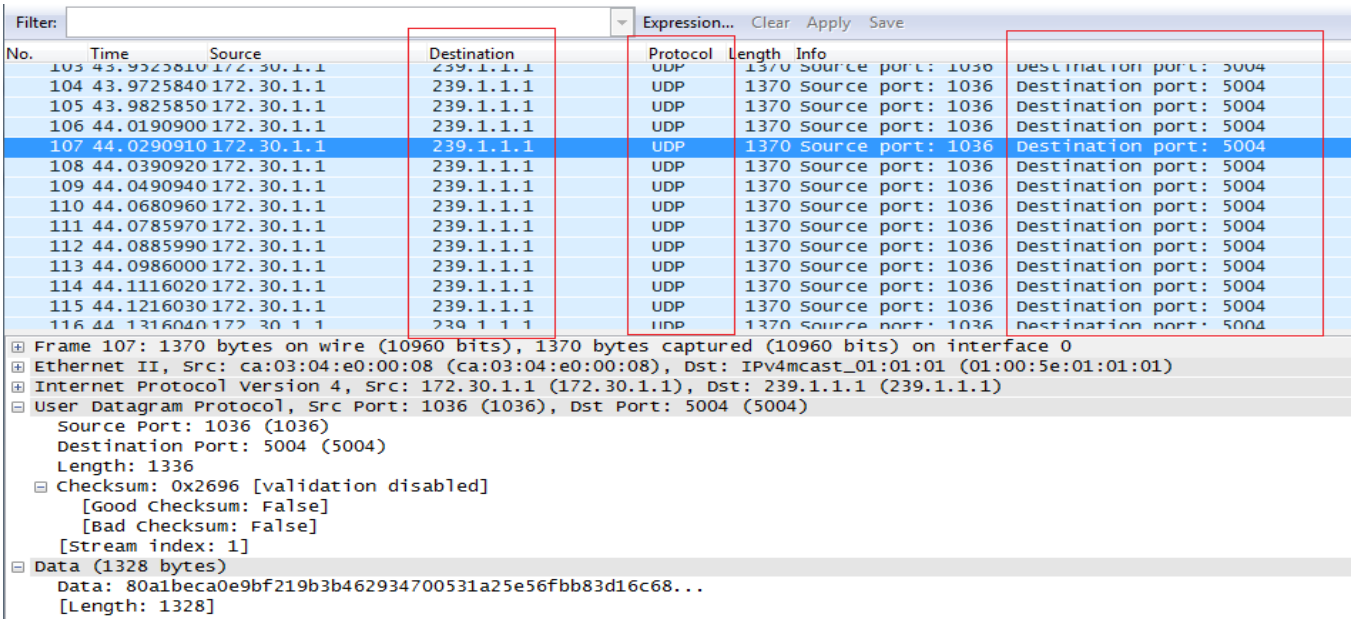


Figura V.11. Protocolo UDP capturado a través de Wireshark durante la transmisión de audio y video Streaming

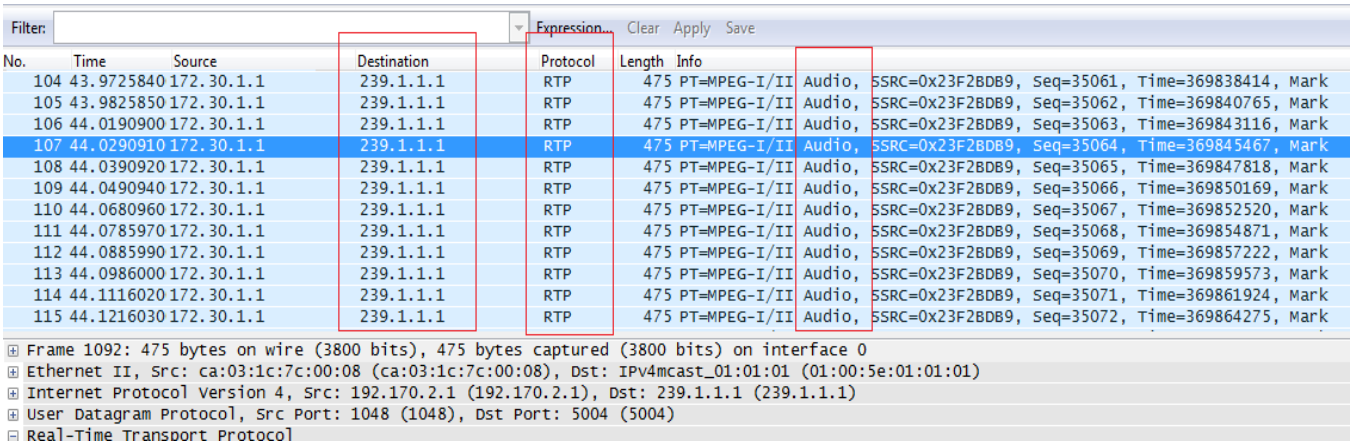


Figura V.12. Protocolo RTP capturado durante la transmisión de audio y video streaming.

Finalmente en la figura V.13, se muestra la captura del mensaje IGMPv2 0x17 *Leave Group*. Resultado de que cada uno de los miembros del grupo 239.1.1.1 abandonará dicho grupo (esto ocurre cuando un equipo físico receptor es apagado o la aplicación VLC se detiene para dejar de recibir flujo). Se destaca la dirección de envió del mensaje *Leave Group* 224.0.0.2 (a todos los *routers multicast*).

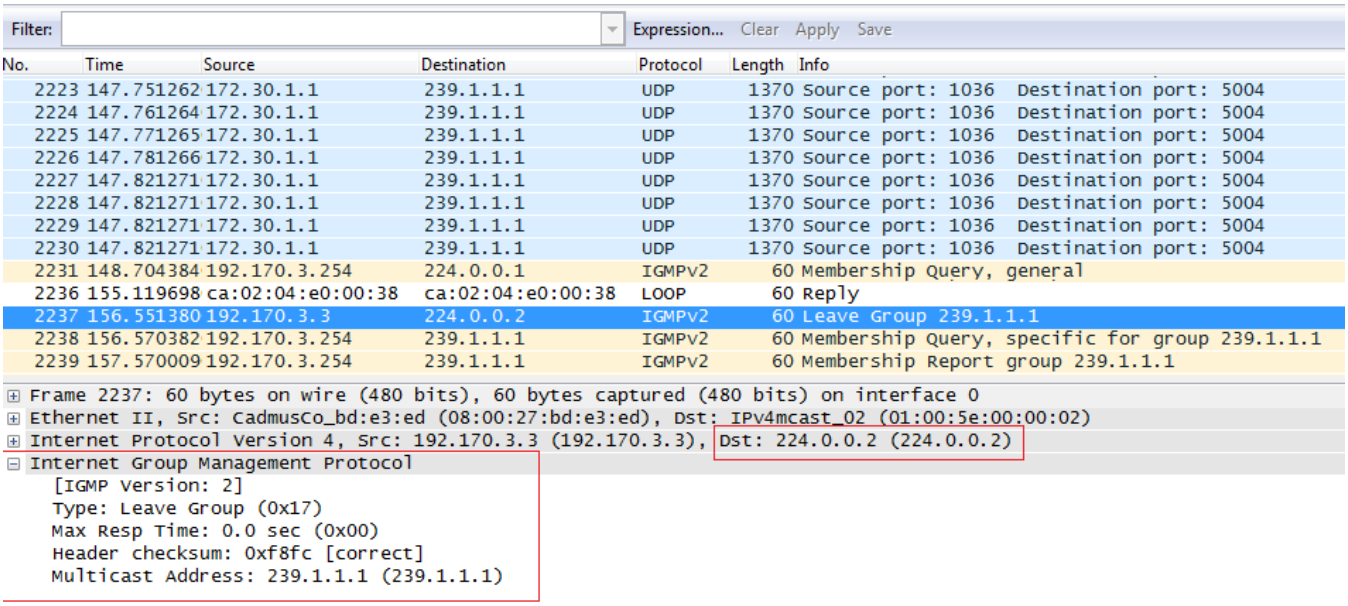


Figura V.13. Mensaje de tipo *Leave Group* que incorpora el protocolo IGMPv2.

V.5 Conclusiones

Encontrar un simulador capaz de soportar tecnologías avanzadas de redes como lo es *IP multicast* en ocasiones resulta una tarea difícil. Con el uso de GNS3, la tarea se volvió asequible principalmente a las características funcionales que lo hacen actuar no sólo como un simulador, sino además, como un emulador de topologías y equipos de red capaz de soportar diferentes tecnologías avanzadas de redes.

Es decir, la implementación física de un modelo *IPv4 multicast* requiere contar con los recursos necesarios como son los *routers* y *switches* principalmente, capaces de soportar la tecnología IP *multicast*. Sin embargo, no es fácil obtener lo anterior, por lo que el emplear el emulador GNS3 resulto en este trabajo en particular, y resulta para cualquier otra propuesta, una opción óptima que permite obtener un funcionamiento similar al de los equipos de red (*routers* y *switches*) reales.

Mediante la virtualización de una topología de red de datos en GNS3, se logró configurar y emular *routers* y computadoras con el fin de obtener un modelo que funcionará con la tecnología *IPv4 multicast*. Con el modelo obtenido, se logró además, realizar la transmisión de audio y video *streaming* dentro de una intranet. Sin embargo, se encontró que GNS3 al tratarse de un emulador que hace uso de los recursos de procesador y memoria RAM del equipo de cómputo donde se encuentre instalado, en ocasiones, estos recursos pueden no ser suficientes para lograr un buen funcionamiento del propio emulador.

Finalmente, el haber realizado el diseño, configuración y emulación de un modelo *IPv4 muiticast*, permitió adquirir conocimientos sobre las redes de datos (como los diferentes protocolos de enrutamiento existentes de tipo unicast y multicast,) a nivel de acceso y distribución. Así mismo, se obtuvo un acercamiento en forma casi real de los equipos de ruteo para *backbone* que usualmente sólo son vistos dentro de las empresas que se dedican a proveer servicios de comunicaciones (CSP's) y de *Internet* (ISP's). Además de que se logró aplicar de manera directa la tecnología avanzada de redes IP Multicast a una intranet.

Apéndice A

Instalación del emulador GNS3 para Windows

Existen versiones del emulador GNS3 para poder ser instaladas y ejecutadas en diferentes distribuciones Linux, en MAC OS y en el SO Windows. Sin embargo, considerando que el uso del SO Windows es muy común en los equipos de cómputo, la instalación que a continuación se realiza aplica para una computadora con sistema operativo Windows 7, 8 y 8.1. Por otro lado, ya que GNS3 permite emular diferentes equipos de red al mismo tiempo, la cantidad de memoria RAM y los niveles de procesamiento del equipo de cómputo físico real donde esté corriendo GNS3 se verán afectados, y en el peor de los casos, superados si no son suficientes. La página web oficial de GNS3 [https://community.gns3.com/welcome] no especifica la cantidad de memoria o el procesador correcto que debe tener como mínimo un equipo de cómputo para que pueda funcionar el emulador; sin embargo, en la página sí se indica que el consumo de memoria RAM y del procesador dependerán de cuán grande sea la red que se emule y de la función que realice cada equipo en la red.

La descarga de GNS3 para Windows es libre y se realiza en el sitio web oficial de la referencia 48.

Una vez ejecutado el archivo GNS3-1.1-all-in-one.exe (GNS3-1.1 última versión para Noviembre de 2014) como se muestra en la figura A.1, el proceso de instalación en Windows es relativamente sencillo, pero se recomienda conocer la función de los programas que se incluyen en la instalación de GNS3, por ello, a continuación se hace una pequeña revisión de la función que realiza cada uno de los programas que contienen GNS3.

Figura A1. Pantalla que muestra el inicio de instalación del emulador GNS3 1.1

La parte más importante de instalación de GNS3 es la que se muestra en la figura A.2; en ella elegiremos instalar o no instalar los programas que se integran a GNS3.

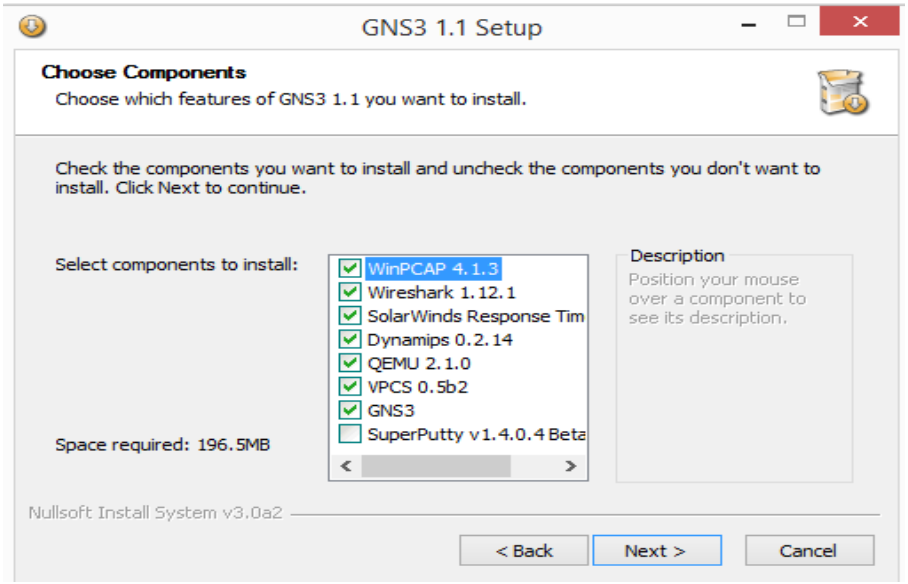


Figura A.2. Se muestra los programas que se instalarán junto con GNS3 para desempeñar diversas funciones.

WinPcap es una herramienta que funciona sobre plataformas Windows y que permite a las aplicaciones, en este caso a GNS3, tener acceso, capturar e incluso filtrar la información (paquetes) que se transmite en las capas bajas de los modelos de red. GNS3 lo integra como herramienta adicional pero no necesaria para su funcionamiento.

Wireshark es un programa que actúa como *sniffe*r, captura el tráfico y muestra los protocolos de red que corren en una red de computadoras, facilitando la visualización de estos mediante una interfaz gráfica. Su utilidad en GNS3 es importante, ya que nos permitirá visualizar y analizar el tráfico que se genere en nuestra red de datos. En particular, en este trabajo nos permitió visualizar y analizar protocolos de enrutamiento *unicast* y *multicast*, protocolos de red como IGMP, así como de transporte UDP y RTP/RTCP.

SolarWinds Response Time es una funcionalidad que trabaja en conjunto con *Wireshark*; le ayuda a este último en el procesamiento profundo de los paquetes que se capturan en la red, y por tanto, eleva la capacidad de funcionamiento de *Wireshark.*

Dynamips es quizá el *software* más importante que se integra en GNS3, ya que es el programa informático que permite emular los *routers* Cisco plataformas 1700, 2600, 3600, y 7200. *Dynamips* realiza la emulación de estos *routers* mediante el llamado y ejecución de archivos .bin; es decir, imágenes de distintas versiones de las IOS Cisco para *routers*. *Dynamips*, además, brinda la posibilidad de emular *switches* mediante la ejecución de la IOS nm-16esw.bin. Por todo lo anterior, la instalación de *Dynamips* junto con GNS3 no es opcional, sino necesaria, con el objetivo de lograr la emulación de los equipos de red y el funcionamiento de la misma.

QEMU permite la emulación y virtualización de distintos equipos; actuando como emulador y como máquina virtual de una computadora o de un servidor (*Qemu virtual/host*), permite correr sistemas operativos y programas informáticos dentro de los mismos. La instalación de *QEMU* en GNS3 se vuelve necesaria cuando se pretende hacer uso de equipos para que funcionen como PC´s o servidores reales dentro de una red de datos.

VPCS (*Virtual PC Simulator*) es un simulador de *hosts* con la posibilidad de ser configurados con direcciones IP, *Subnet Mask* y *Gateway* para hacer ping a los equipos de red, y con ello comprobar que la conectividad dentro de la misma este activa y funcionando adecuadamente. *VPCS* permite el uso de hasta nueve computadoras virtuales a la vez dentro de una topología de red virtual de GNS3.

Se recomienda instalar todos los programas en la instalación de GNS3 ya que permitirá optimizar la capacidad de este último, y además de que los programas utilizan un espacio de almacenamiento bastante pequeño como lo muestra la figura A.3. La instalación de GNS3 habrá concluido cuando aparezca una ventana que muestre el mensaje *Installation Complete-Setup was completed successfully*, como se muestra en la figura A.4.

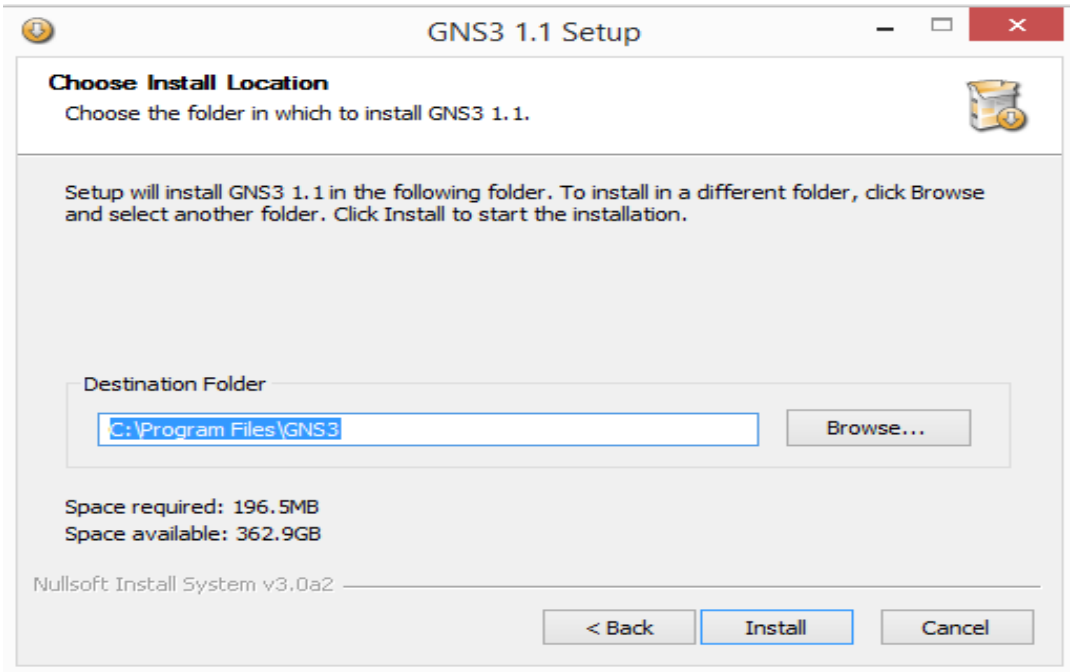


Figura A.3 Muestra la ubicación donde se instalara el programa GNS3 y el tamaño de la carpeta.

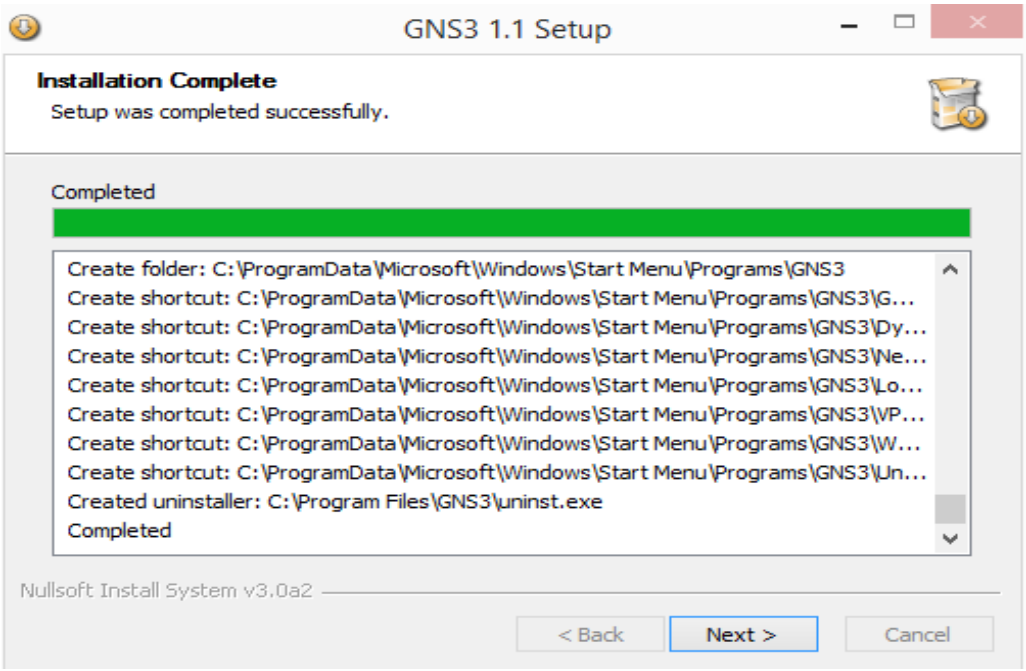


Figura A.4. La instalación ha sido completada y satisfactoria.

Una vez instalado GNS3, lo siguiente es comenzar con su configuración. En los siguientes pasos y figuras se describe la configuración para GNS3:

1. Crear una carpeta GNS3 en un algún lugar de rápido acceso de la computadora real, por ejemplo en documentos, y dentro de GNS3 crear una carpeta con nombre *Images*, otra con nombre *Proyects* y una tercera llamada *Captures*. Por ejemplo en la ruta: Equipo>Documentos>GNS3

2. Descargar las imágenes de IOS Cisco para las plataformas de *routers* 1700, 2600, 3600, 3700 y 7200, las cuales serán guardadas en la carpeta *Images* creada en el paso 1. Las imágenes están disponibles en varios sitios web de *Internet*.

3. Arrancar GNS3, elegir la opción *Edit* > *Preferences*, como se muestra en la figura A.5.

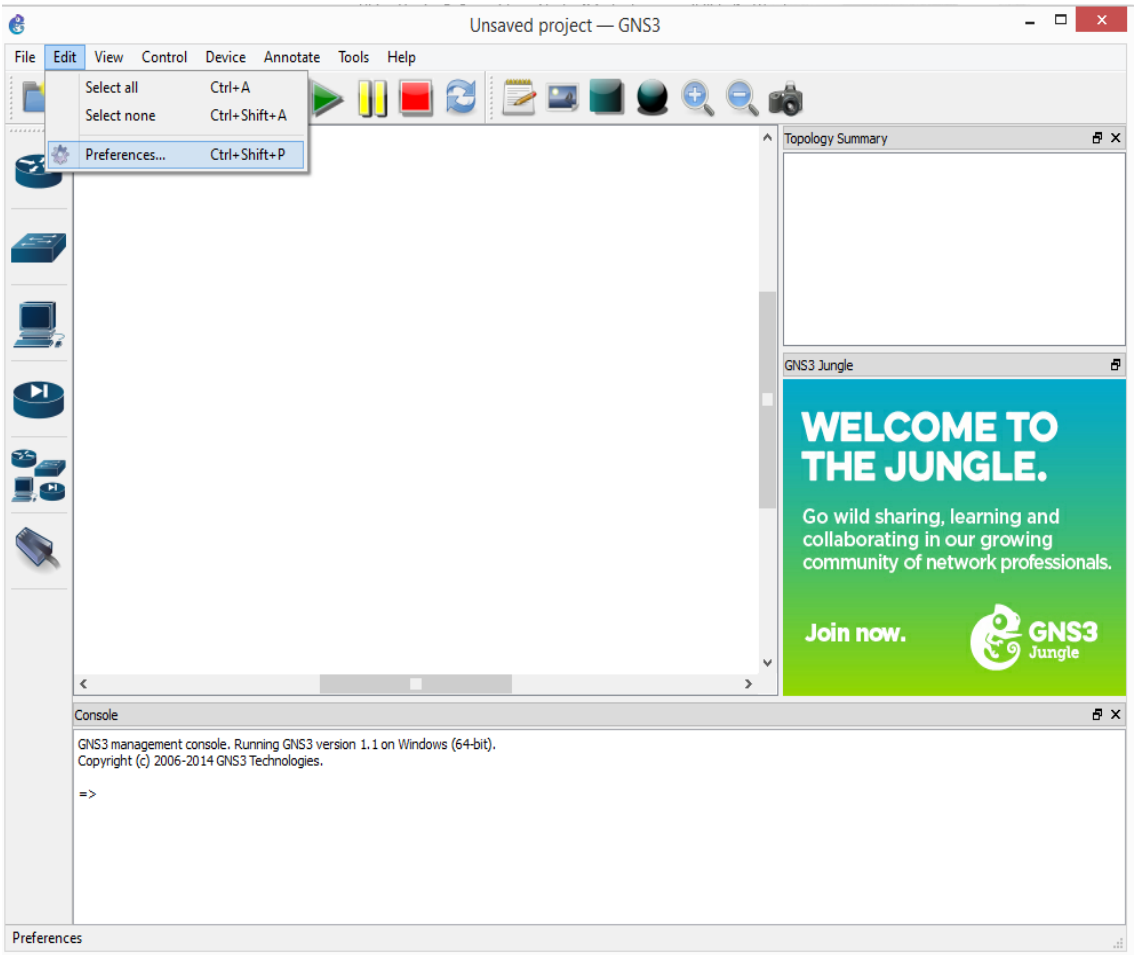


Figura A.5. Programa GNS3 al ser arrancado tras su instalación. La opción para su configuración es *Edit* > *Preferences*.

4. Una vez que se desplegó la ventana *General preferences*, se debe insertar la ruta donde se encuentran las carpetas *Proyects* e *Images* creadas con anterioridad, tal y como se muestra en la figura A.6.

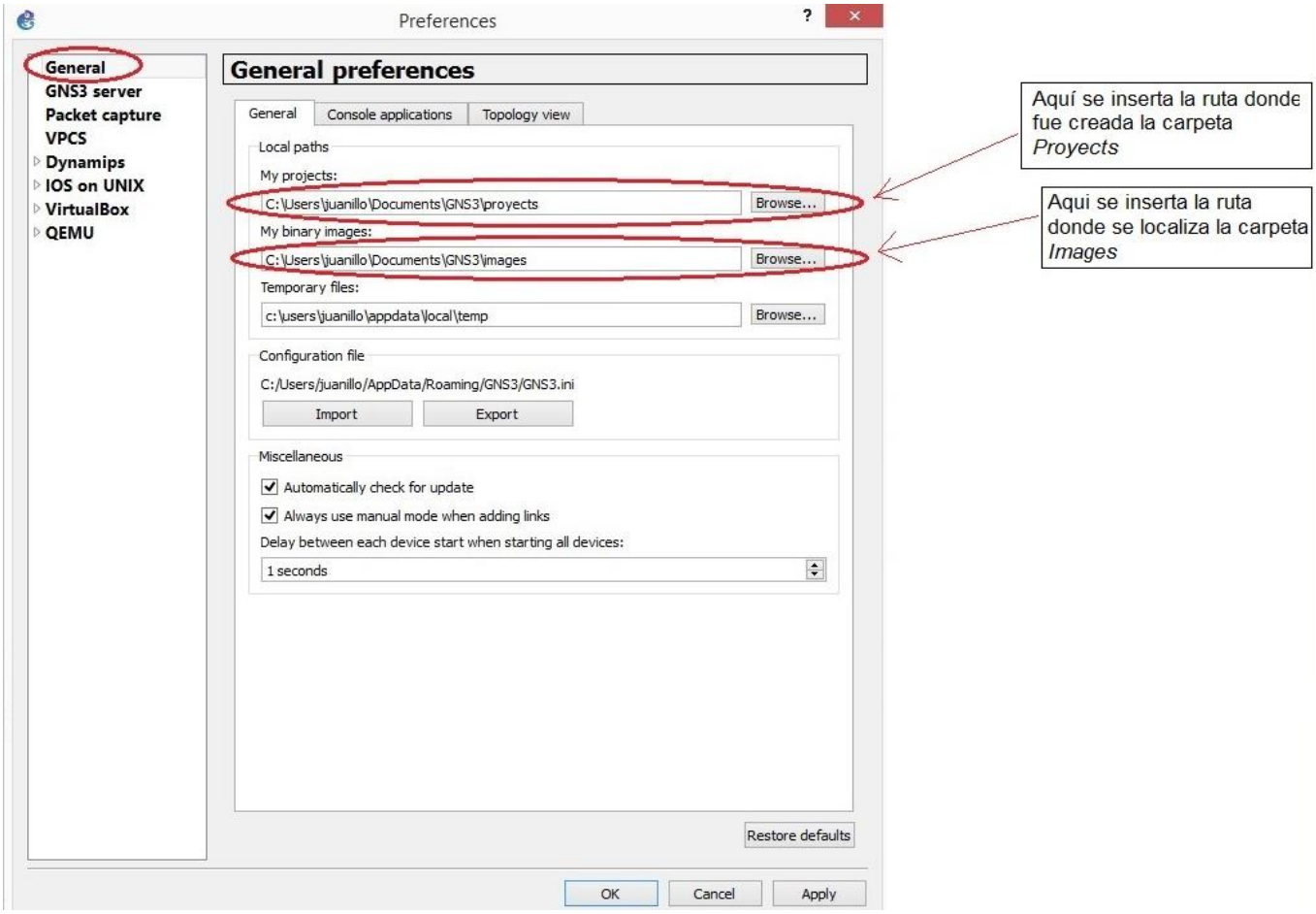


Figura A.6. Muestra la opción de preferencias generales, en donde se lleva a cabo la inserción de las rutas de las carpetas *Projects*.

5. En la misma ventana de *Preferences*, elegimos la opción *Dynamips > IOS routers > New*, con el objeto de insertar cada una de las imágenes de IOS Cisco de los *routers* c1700, c2600, c2691, c3600, c3725 y c7200 (previamente descargadas), como lo ilustra la figura A.7. Esta acción permitirá que *Dynamips* contenga los archivos de imagen de todas las plataformas, y que a través de GNS3, los *routers* puedan llamar y ejecutar dichas imágenes de IOS.

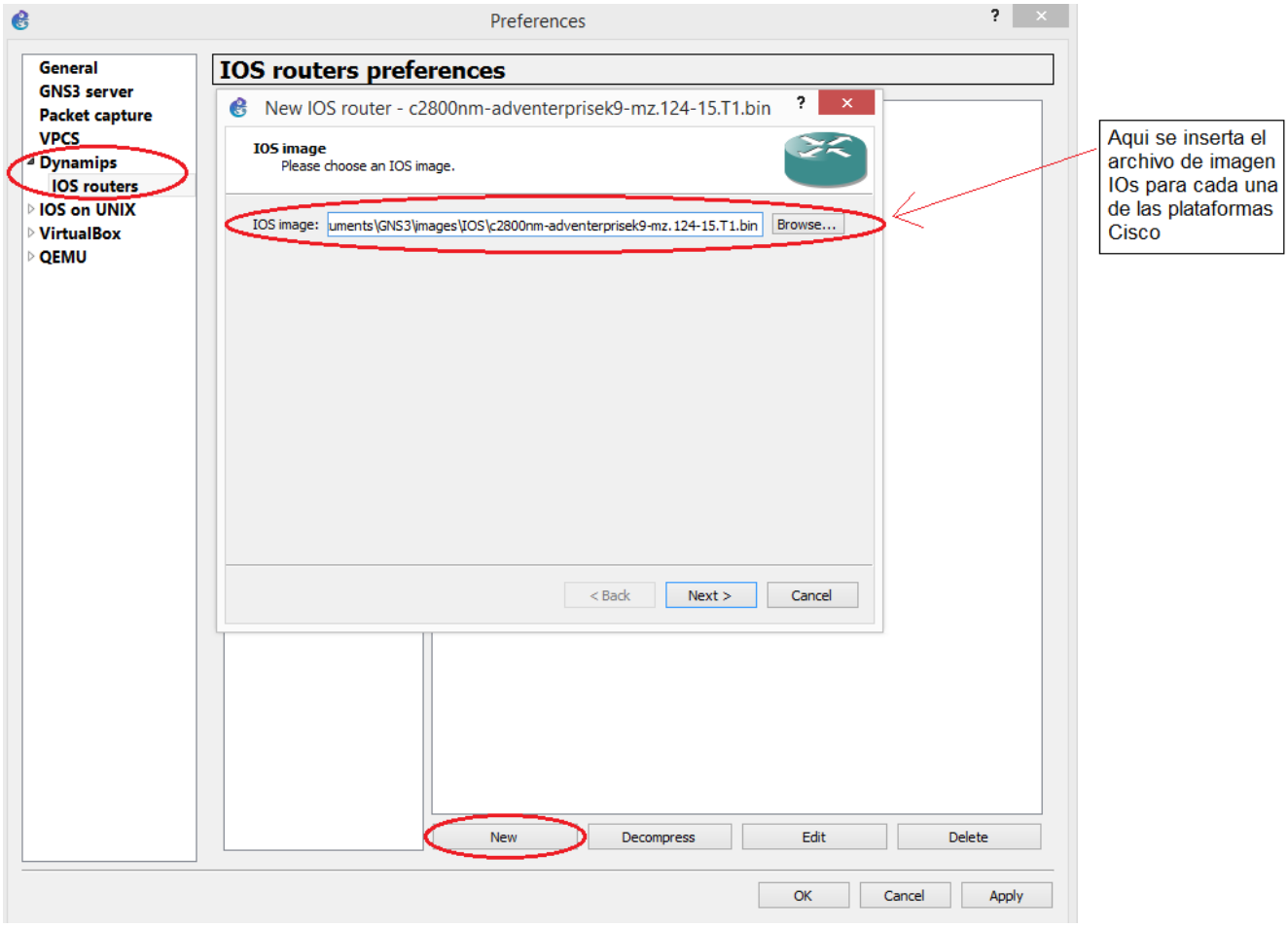


Figura A.7. Ventana que se despliega para añadir una por una las IOS de todas las plataformas de *routers.*

6. Cuando se añade una nueva IOS *routers* en *Dynamips*, se muestran varias opciones de configuración para la plataforma de equipo a la que pertenece la imagen añadida. Las opciones de configuración que se despliegan son *Name and plataform* (nombre que se le quiere dar al equipo y la plataforma a la que corresponde), *Memory* (cantidad de memoria RAM que se desea asignar al equipo), *Network adapters* (la cantidad y tipo de adaptadores de red "interfaces"), *WIC Modules* (*WAN Interface Cards*), y un *Idle-PC* (valor Idle de procesador) mostradas en las figuras A.8, A.9, A.10, A.11 y A.12 respectivamente. Todos estos valores a excepción de *Name and plataforms* y *Memory*, no es necesario asignarlos, ya que se puede hacer cada vez que se ejecute algún equipo al diseñar una topología de red, por tanto, sólo es necesario darle *next* a las opciones de ventana *Nertwork adapters, WIC Modules e Idle-PC*. En la figura A.13 se muestra el resultado del proceso de integración de todas las imágenes IOS Cisco dentro de *Dynamips*.

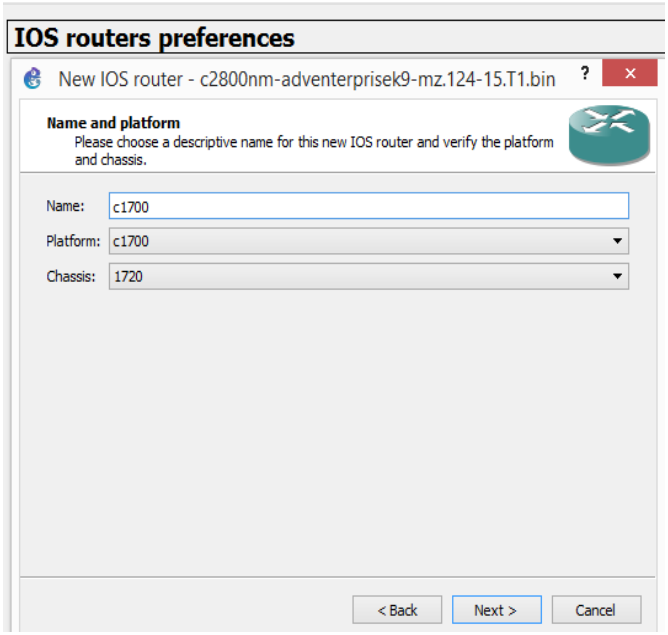


Figura A.8. Configuración del nombre y la plataforma deseada para cada uno de los *routers* que se añaden a *Dynamips*. En la mayoría de los casos se elige la configuración que aparece por defecto.

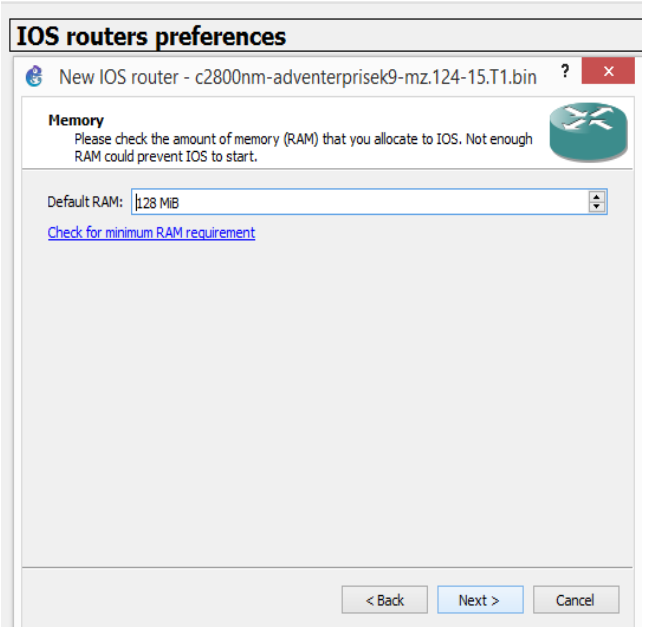


Figura A.9. Configuración de la cantidad de memoria RAM que se desea asignar a cada *router*. La cantidad de memoria dependerá de la versión de la imagen que se haya elegido para el equipo en cuestión.

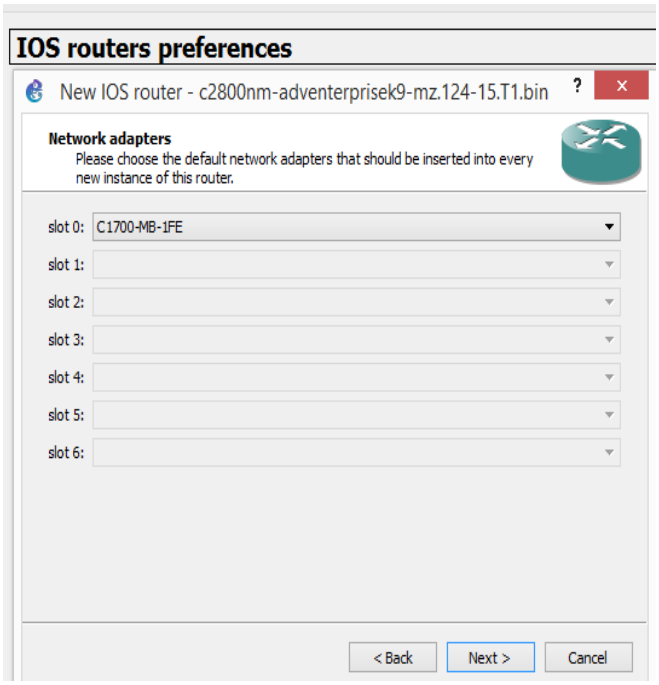


Figura A10. Configuración de cantidad y tipo de adaptadores de red que se desee añadir al *router*. Esta configuración no es necesaria al momento de la configuración, ya que se puede asignar posteriormente de acuerdo a las necesidades que se requieran cuando se diseñe una topología de red.

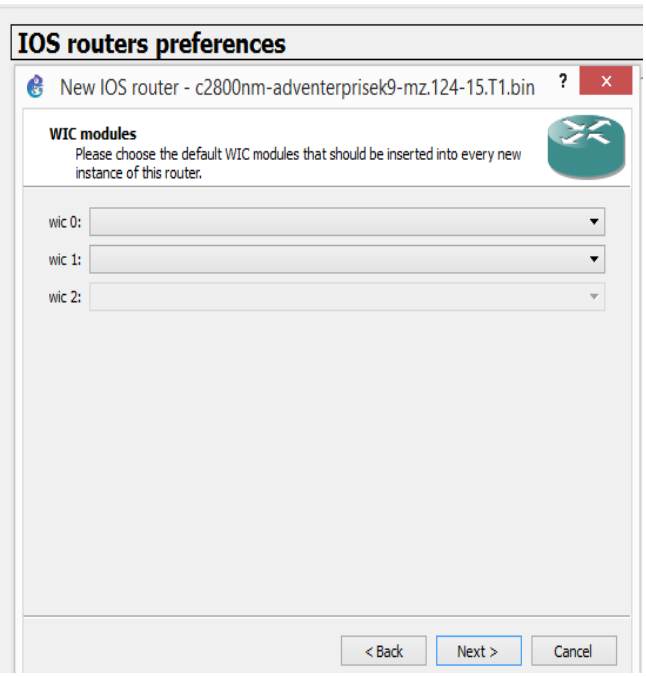


Figura A.11 Configuración de la cantidad y tipos de módulos WIC. Esta configuración no es necesaria al momento de la configuración, se puede realizar en el momento de diseñar una topología de red.

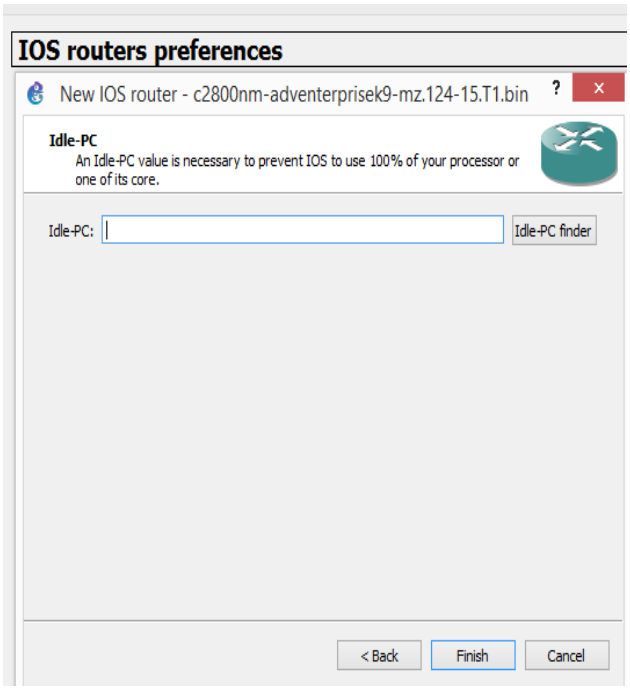


Figura A.12 Configuración de Ide-PC adecuado para cada router. Con el objeto de disminuir a lo máximo los niveles procesamiento de la computadora donde se esté ejecutando GNS3. Este valor no es necesario que sea asignado al momento de la configuración, pero si se recomienda que se asigne un valor cuando se ejecute cada *router.*

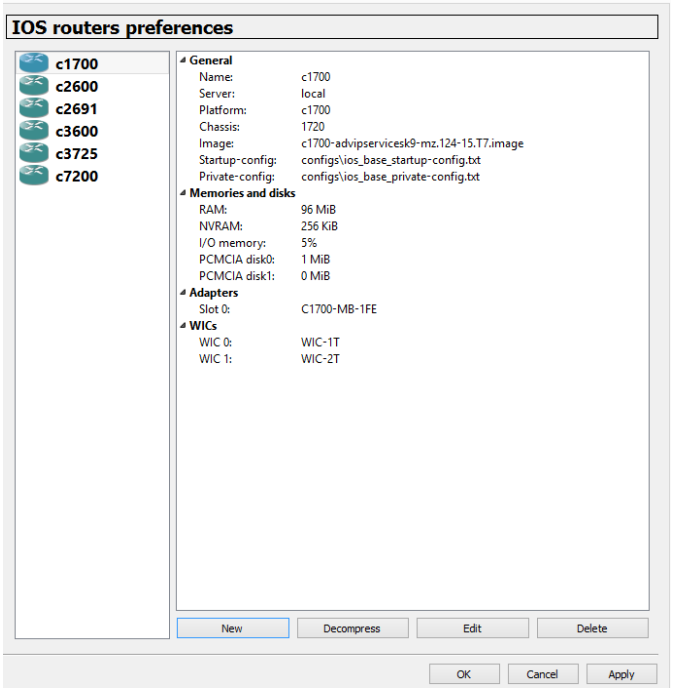


Figura A.13. Una vez que se han añadido todas las IOS se verá una ventana con la información de cada uno de las plataformas de *routers* configuradas.

7. Comprobar la configuración realizada. Esto lo haremos arrastrando cualquier *router* virtual al escritorio de trabajo de GNS3 como se muestra en la figura A.14. Posteriormente, escribimos el comando *start* {nombre del *router* que se ejecute}, por ejemplo *start* R1, en la consola que se encuentra en la parte inferior del escritorio de trabajo, el cual, indica a R1 que se encienda. R1 realizará un llamado de la imagen IOS en *Dynamips*; *Dynamips* localizará la imagen solicitada y se la enviará a R1 para que este último comience a ejecutarla, lo que se conoce como arranque de IOS en un *router*. Para poder visualizar la consola del *router* R1 y el arranque del IOS, será necesario ejecutar el comando *console* R1, como se muestra en la figura A.15.

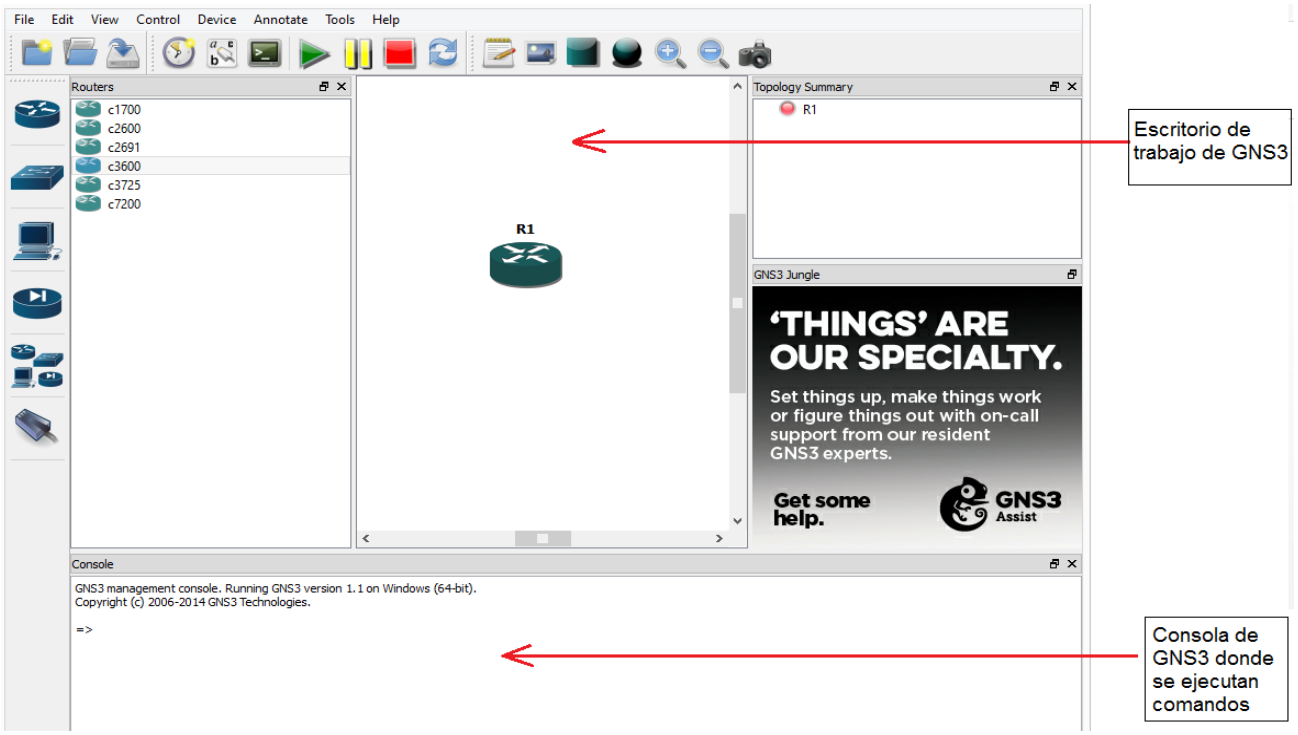


Figura A.14. Seleccionar un *router* de cualquier plataforma y arrastrarlo al escritorio de trabajo de GNS3.

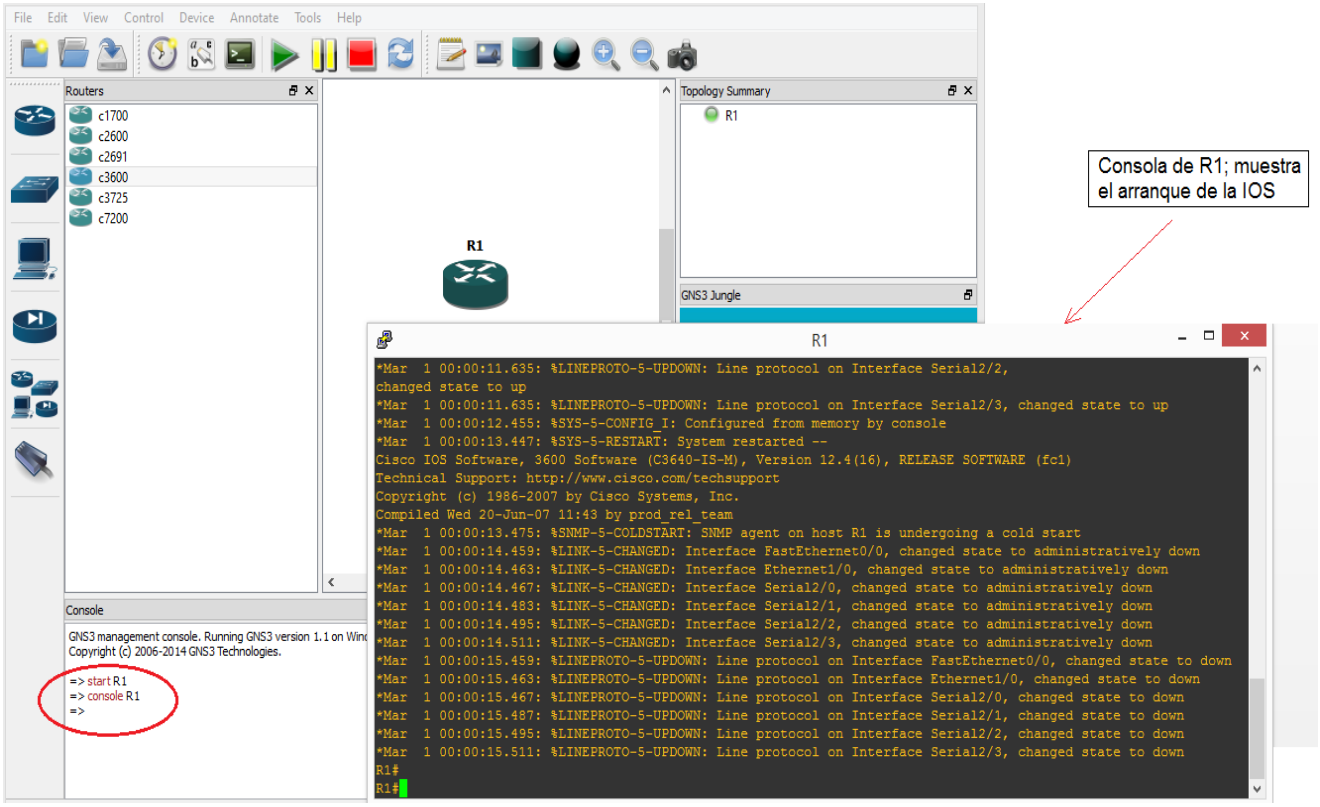


Figura A.15. Se comprueba que la configuración de GNS3 funciona mediante la ejecución de comandos en la consola de GNS3 para que el *router* elegido arranque correctamente.

Apéndice B

Instalación de VLC media player para la transmisión de audio y video streaming sobre el modelo IPv4 multicast

---

En este apéndice se explican los pasos necesarios para lograr la transmisión de información (audio y video) usando VLC *media player*, instalado en una máquina virtual Windows XP que actúa como el equipo fuente *multicast* (en este caso una fuente en el plantel Del Valle). Así mismo, para la reproducción de audio y video *Streaming multicast* de igual manera usando VLC *media player* instalado en tres máquinas virtuales Windows XP que actúan como los equipos receptores *multicast* identificados por la dirección 239.1.1.1 (ubicadas en los planteles SLT, Casa Libertad y Cuautepec).

1. El formato VLC instalado en las máquinas virtuales se descargó de la página oficial [49].

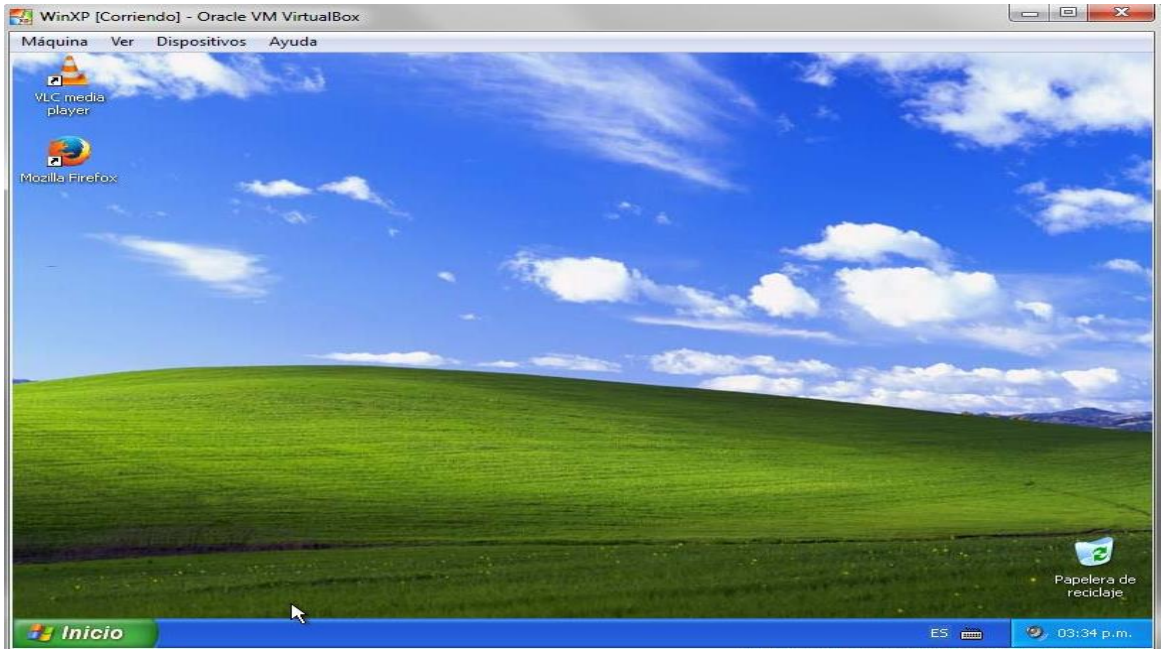


Figura B.1. Ventana de la máquina virtual fuente emisora, y donde se visualiza el software VLC para ser la herramienta que realice la transmisión y recepción de audio y video Streaming.

2. Iniciado VLC en la máquina virtual fuente (del *router* Del Valle) nos dirigimos a Medio (File) y elegimos “Emitir”.

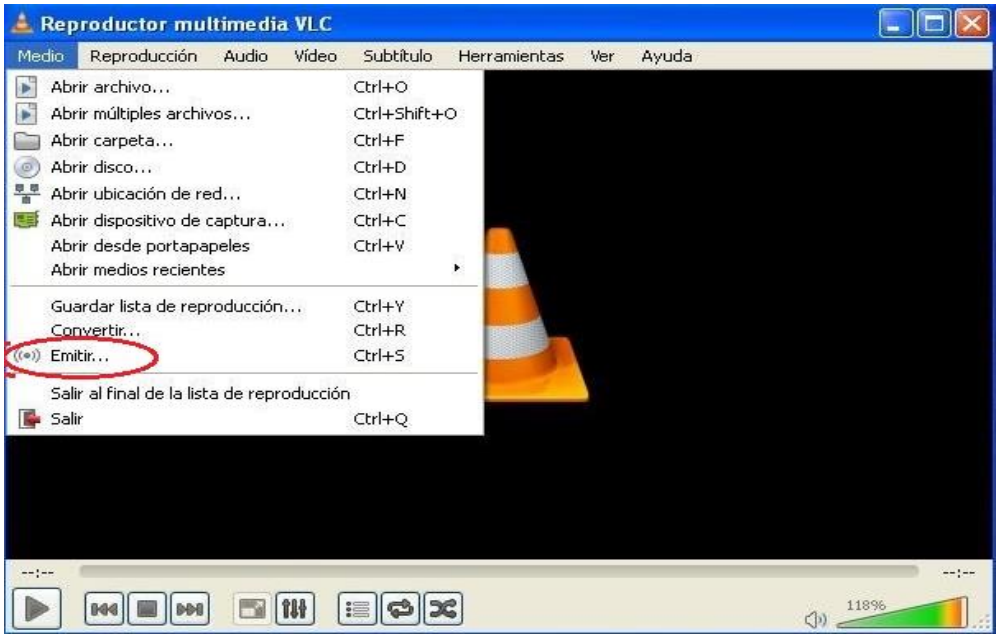


Figura B.2. Ventana de la máquina virtual fuente emisora, donde se visualiza la interfaz gráfica del software VLC media player.

3. Añadimos el archivo de audio o video que se desee transmitir y que se encuentra almacenado en el equipo o dispositivo externo (CD, DVD, memoria flash).

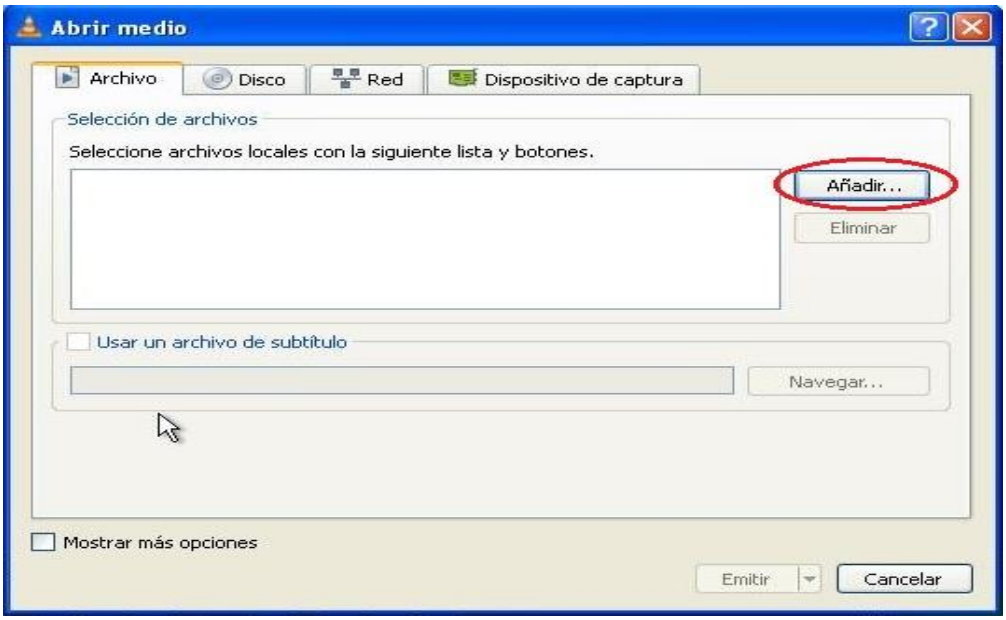


Figura B.3. En este proyecto se propone transmitir un archivo de audio o video almacenado en el equipo (PC o servidor) o en un dispositivo externo como CD o memoria flash. Sin embargo VLC permite diferentes entradas; como por ejemplo de videocámaras que graban una videoconferencia (tiempo real).

4. Se añade el audio o video deseado y se realiza la emisión.

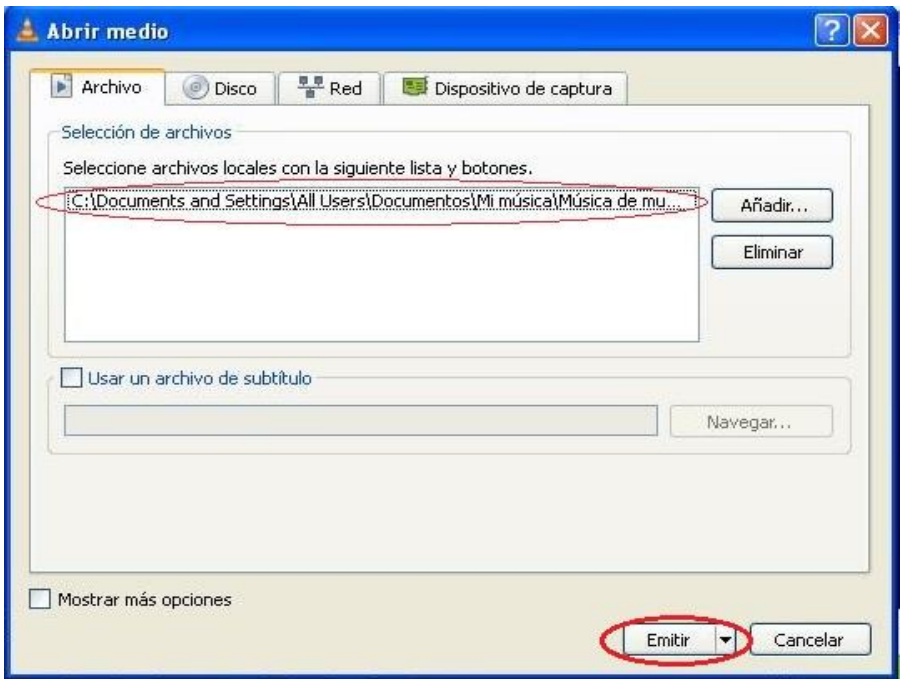


Figura B.4. Añadido el archivo de audio o video; se elige emitir.

5. Se abre una ventana para verificar que el archivo elegido para emitirse coincida con el mostrado en la parte inferior de la ventana, incluyendo la extensión (tipo) del mismo.

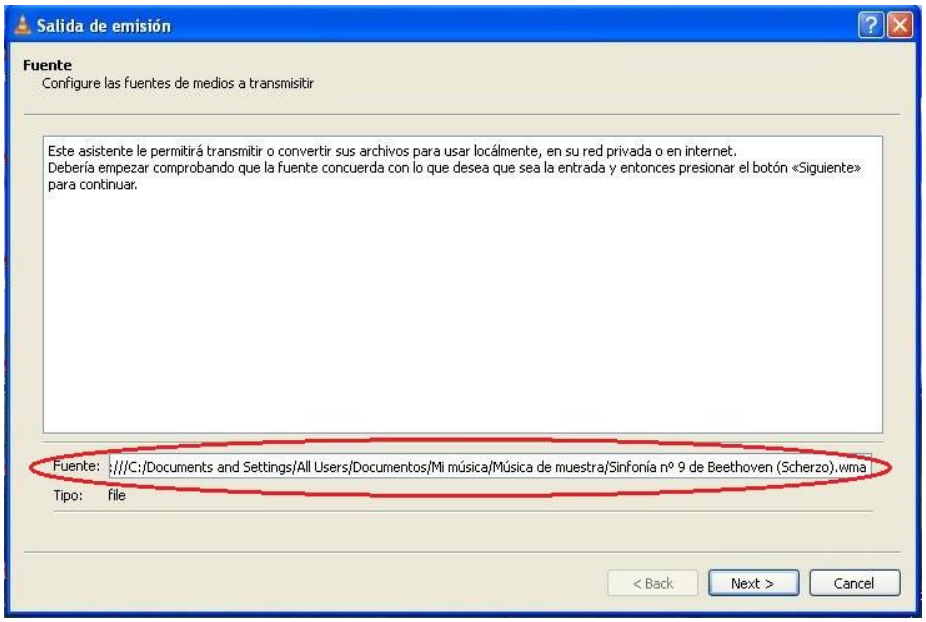


Figura B.5. Se verifica que el archivo fuente coincida con el que aparece en esta zona.

6. VLC permite utilizar distintos protocolos de comunicación, entre los que destacan RTP para audio y video. Además en este paso VLC permite optar por si se desea reproducir el archivo de audio o video en el equipo fuente activando la casilla “Mostrar en local”.

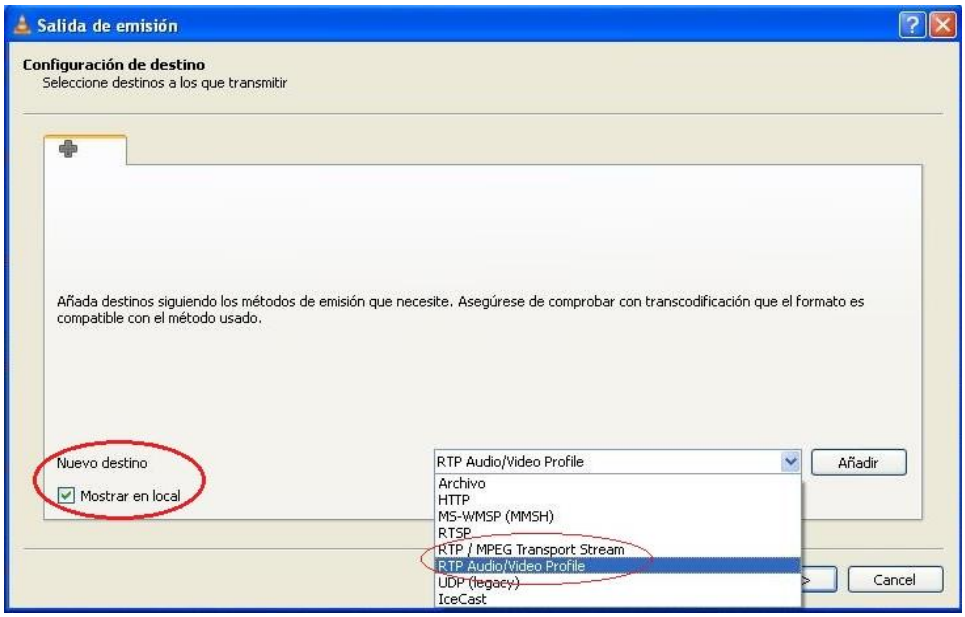


Figura B.6. Para la emisión de audio se utilizó el protocolo RTP Audio/Video Profile, mientras que para la emisión de un archivo de video se empleó RTP/MPEG Transport. Con dicha elección se logró transmitir flujos de audio y video Streaming usando el protocolo RTP.

7. En este paso se elige la dirección *IPv4 multicast* destino para transmitir el flujo.

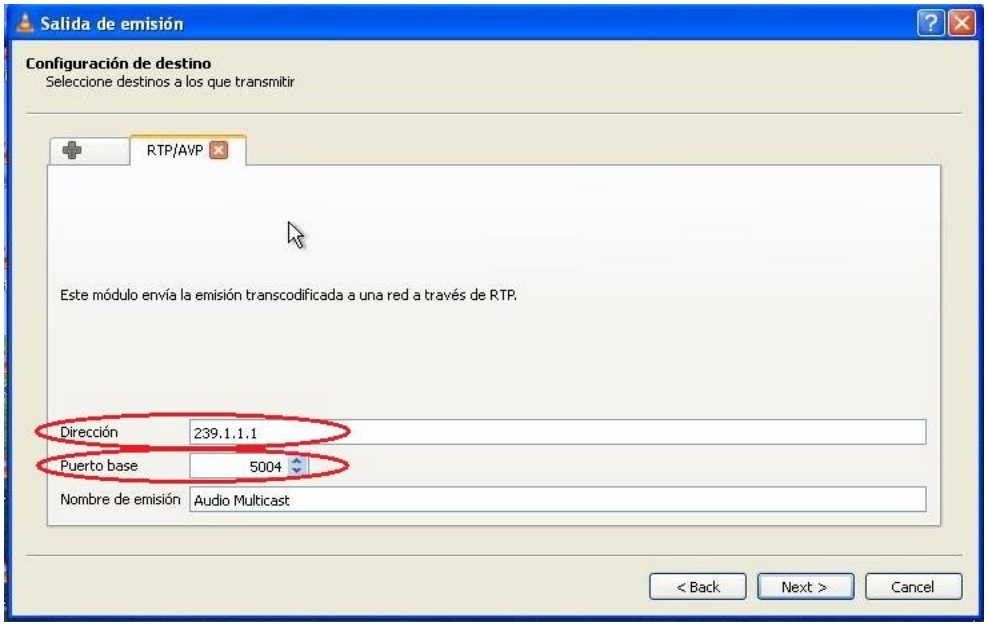


Figura B.7. Se observa que la dirección 239.1.1.1 es la dirección IP de grupo multicast que previamente se eligió y configuró en el modelo IPv4 Multicast de la UACM. Se observa también en la figura el protocolo 5004, que se usa por defecto para el protocolo RTP.

8. En este paso se elige si se desea o no "Habilitar transcodificar" el audio o video original a transmitir, y permite elegir entre un gran variedad de formatos para realizar la transcodificación.

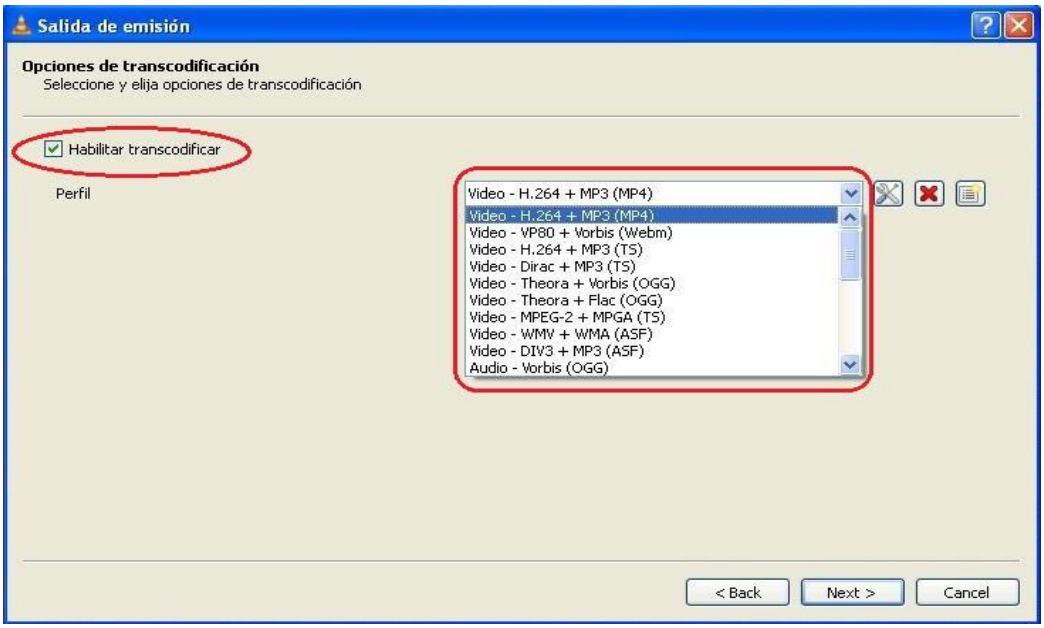


Figura B.8. VLC integra la opción de activar o no transcodificación. Para la transmisión de audio y video que se realizó en este proyecto se eligió activar transcodificar en un formato Video –H.264+MP3 (MP4).

9. En este paso VLC permite elegir y manipular las características de los diferentes códec de audio y video que soporta.

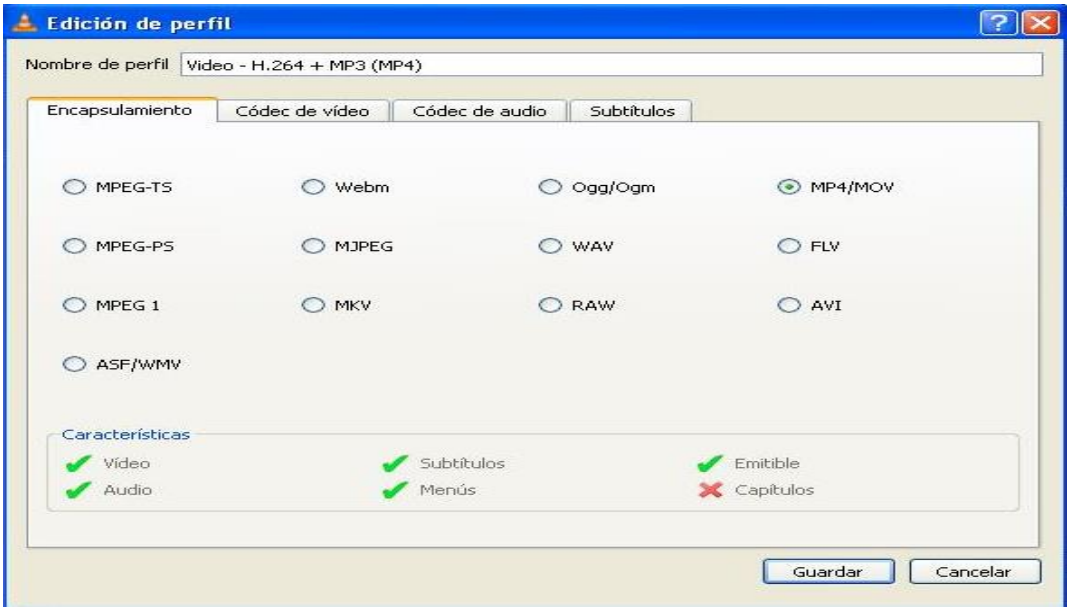


Figura B.9. Para la transmisión de audio y video Streaming que se realizó en este trabajo se eligió el códec que tuviera características de soporte para audio/video y que fuera emitible.

10. En este último paso para realizar la transmisión de audio y video *Streaming* se muestra en código todo lo configurado con anterioridad, de tal manera que VLC ofrece realizar cualquier modificación a través de código. En este sentido, es conveniente hacer énfasis en el TTL para flujo *multicast*; el cual VLC por defecto lo integra con un valor TTL=1, de tal manera que ese TTL debe de modificarse asignandose un valor mayor al número de saltos (routers) existentes en la red *IP Multicast*, con el objetivo de que el flujo llegue sin problemas a los receptores del grupo.

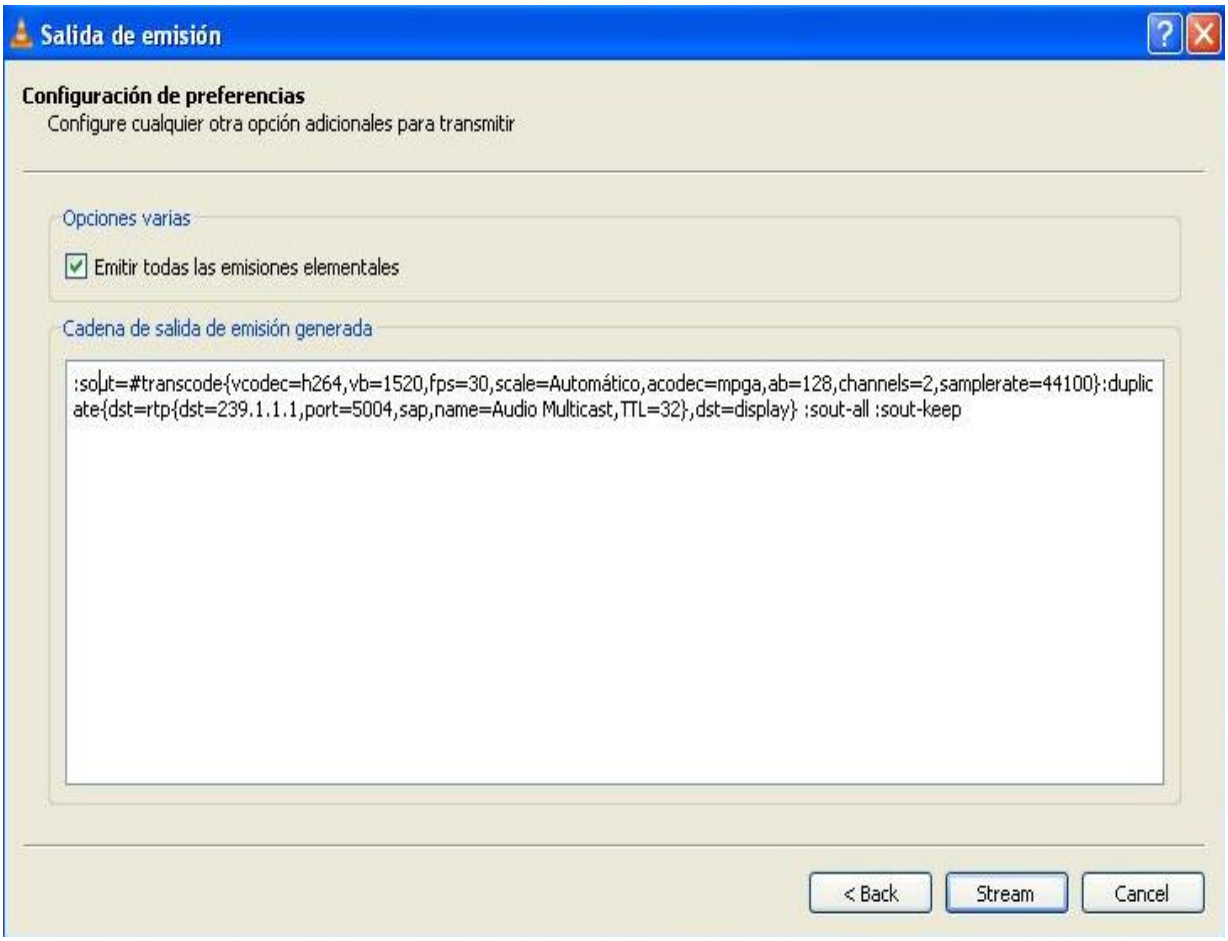


Figura B.10. El valor de TTL, que en este caso se asignó TTL= 32 (Limitado dentro de una red), se añade al código; al final del código entre llaves rtp={dst=239.1.1.1,port=5004,sap,name=audio Multicast, TTL=32}

11. Finalmente, se inicia VLC en las máquina virtuales receptoras (*routers* SLT, Casa Libertad y Cuautepec en el caso del modelo de la UACM) nos dirigimos a Medio (File) y elegimos "Abrir ubicación de red". Como se muestra en la figura B.11.

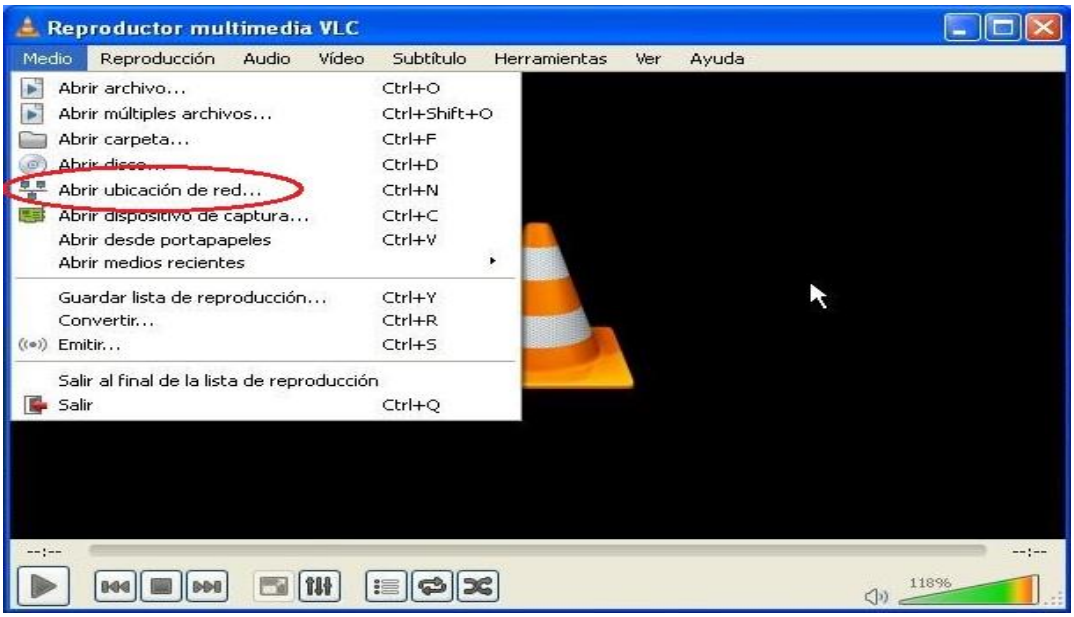


Figura B.11. Ventana de una de las máquinas virtuales donde se visualiza el software VLC para ser la herramienta que realice la reproducción de audio y video *Streaming*.

11. Y para reproducir el tráfico transmitido por la fuente, VLC ofrece la opción de insertar el protocolo para recibir el flujo. Se debe insertar aquel protocolo que fue usado para emitir el flujo y añadir la dirección *multicast* así como el puerto del grupo al que fue transmitido, tal y como se muestra en la figura B.12.

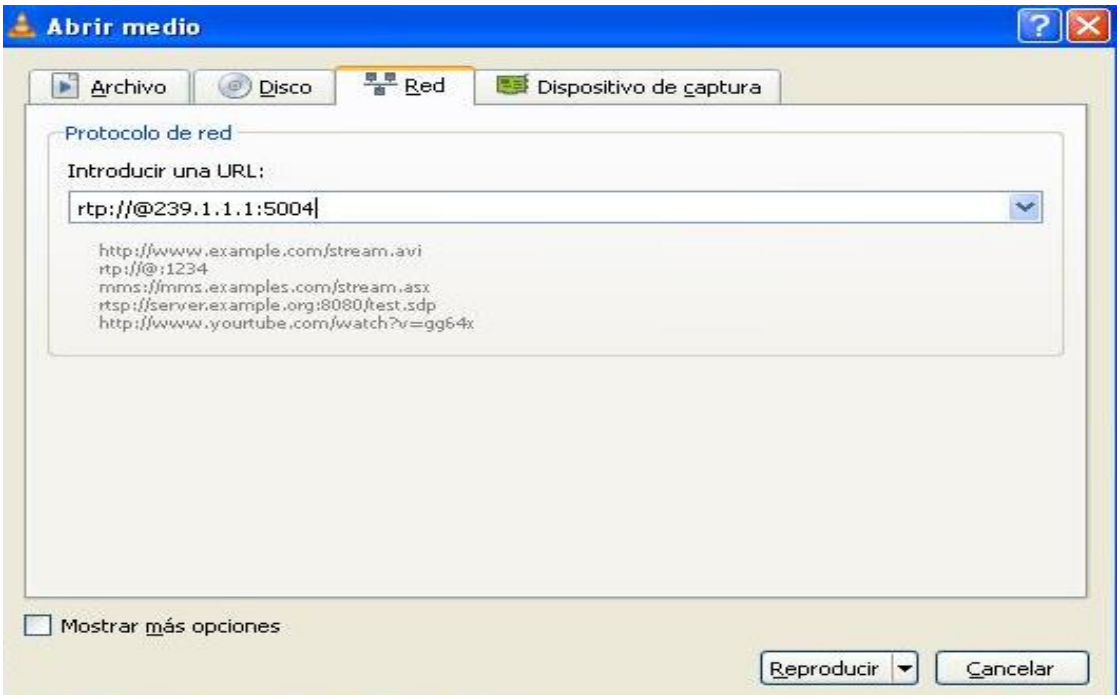


Figura B.12. Se debe insertar manualmente el protocolo, dirección multicast y puerto tal y como se muestra.

Referencias

[1] Ward Louckx, *Streaming media over multicast*, [en línea]; Mayo 2013 [consulta: Sep. 2014] Disponible en: https://www.theseus.fi/bitstream/handle/10024/59829/WL_Thesis_Final.pdf?sequence=1

[2] UNICAST MULTICAST BROADCAST; [video en línea], 2014, Disponible en: http://www.youtube.com/watch?v=ghRtPxQTTG8

[3] Cisco Support Community ITA Terms, *Unicast*, June 2009, Disponible: https://supportforums.cisco.com/document/6196/unicast

[4] IEEE, IEEE Xplore Digital Library, *Bandwidth-allocation policies for unicast and multicast flows* [en línea], Networking, IEEE/ACM Transactions on, Vol. 9, Aug. 2001 [consulta: Agosto 2014], ISSN: 1063-6692 Disponible: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=944344&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D944344

[5] DARPA INTERNET PROGRAM, RFC 793, *Transmission Control Protocol*, September 1981, [en línea]; Disponible en: https://tools.ietf.org/html/rfc793

[6] IANA (Internet Assigned Numbers Authority), *Service Name and Transport Protocol Port Number Registry*, October 17, 2014; Disponible: http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml

[7] Redes Cisco.Net, *TCP Connection Management- TCP General Concepts*, Septiembre 2014; Disponible: http://www.redescisco.net/v2/art/tcp-connection-management-tcp-general-concepts/

[8] Redes Cisco.Net, *TCP Connection Management- TCP General Establishment (Three-Way Handshake*), Septiembre 2014; [Disponible en: http://www.redescisco.net/v2/art/tcp-connection-management-tcp-connection-establishment-three-way-handshake/]

[9] DARPA, J. Postel, RFC 768, *User Datagram Protocol*, August 1980, [en línea] Disponible en: http://tools.ietf.org/html/rfc768

[10] Cisco Networking, *OSI Transport Layer (Network Fundamentals – Chapter 4);* [video en línea], 2014, Disponible:

http://www.youtube.com/watch?v=iECMdbcP6U0&index=3&list=PL2A7l6PiV52d5pQgaunGqTwY63_3-kIv0

[11] IANA (Internet Assigned Numbers Authority), *Number Resources*, October, 2014; Disponible: https://www.iana.org/numbers

[12] IANA, Network Working Group, *Special-Use IPv4 Addresses*, September 2002 Disponible en: https://www.rfc-editor.org/rfc/rfc3330.txt

[13] IETF (Internet Engineering Task Force), ICANN, RFC 5735, *Special Use IPv4 Addresses*, January 2010, [en línea]; Disponible en: http://tools.ietf.org/pdf/rfc5735.pdf

[14] IEEE, IEEE STANDARDS ASSOCIATION, 2014; Disponible en: http://standards.ieee.org/findstds/index.html

[15] IEEE, IEEE STANDARDS ASSOCIATION, *Guidelines for Use Organizationally Unique Identifier (OUI) and Company (CID)*, 2004, [en línea]; Disponible en: https://standards.ieee.org/develop/regauth/tut/eui.pdf

[16] Network Working Group, RFC 5735, *An Ethernet Address Resolution Protocolo or Converting Network Protocol Addresses*, November 1982, [en línea]; Disponible en: http://tools.ietf.org/pdf/rfc826.pdf

[17] Margaret Rose, TechTarget, multicast, [en línea]; Disponible en: http://searchnetworking.techtarget.com/definition/multicast

[18] CISCO, *Benefits of IP Multicast*, [video en línea], 2014, Disponible en: http://www.cisco.com/c/en/us/products/ios-nx-os-software/ip-multicast/index.html

[19] IANA, *IPv4 Multicast Address Space Registry*, September 19, 2014; Disponible: http://www.iana.org/assignments/multicast-addresses/multicast-addresses.xhtml

[20] IANA, *The Multicast Addresses registry*, Disponible: http://www.iana.org/assignments/multicast-addresses

[21] IETF (Internet Engineering Task Force), ICANN, RFC 5771, *IANA Guidelines for IPv4 Multicast Address Assignments*, March 2010, [en línea]; Disponible: http://tools.ietf.org/html/rfc5771

[22] CISCO SYSTEMS, *Implementing Cisco Multicast,* Student Guide, Volumen. 1, V.1.1, 2002.

[23] CISCO SYSTEMS, *Guidelines for Enterprise IP Multicast Address Allocation*, 2004, [en línea]; Disponible en: http://www.cisco.com/c/dam/en/us/support/docs/ip/ip-multicast/ipmlt_wp.pdf

[24] Linux Focus, *Multicast*, 2014; Disponible: http://es.tldp.org/LinuxFocus/pub/mirror/LinuxFocus/Castellano/January2001/article144.shtml

[25] CISCO SYSTEMS, *Introduction to IP Multicast* (RST-1261), 2006, [en línea]; Disponible en: http://www.cisco.com/c/dam/en/us/products/collateral/ios-nx-os-software/ip-multicast/prod_presentation0900aecd80310883.pdf

[26]Stephen E. Deering and David R. Cheriton, Stanford University, *Multicast Routing in Datagram Internetworks and Extended LANs*, May 1990, Disponible en: http://www.utdallas.edu/~kxs028100/Papers/deering-multicast.pdf

[27]Beau Williamson, CISCO SYSTEMS, *Developing IP Multicast Networks*, Vol.1, Indianapolis USA, 2000, ISBN: 1-57870-077-9

[28] Pedro José Piñero Escuer y Pilar Manzanares López, Universidad Pólitecnica de Cartagena, *Evolución de las comunicaciones Multicast: del nivel de red al nivel de aplicación*, [en línea], 2010, [consulta Agosto 2014], ISSN: 2127-2042, Disponible en: http://repositorio.bib.upct.es/dspace/bitstream/10317/2476/1/1.14.pdf

[29] Network Working Group, RFC 3170, *IP Multicast Applications: Challenges and Solutions*, September 2001, [en línea]; Disponible en: https://www.ietf.org/rfc/rfc3170.txt

[30] Harold de Dios Tovar, Cudi Reunión de primavera, Universidad de Guadalajara, *Multicast Red CUDI*, [en línea], 2013, [consulta Noviembre 2014], Disponible en: http://www.cudi.edu.mx/primavera_2013/presentaciones/Multicast_HAROLD_UdeG.pdf

[31] Juan Carlos S., Pilar Manzanares, José María Malgosa, Fernando Cerdán, Universidad Politécnica de Cartagena, *Tecnología Multicast para Entornos Empresariales*, [en línea], 2004 [consulta Agosto 2014], ISSN: 1698-2429, Disponible en: http://repositorio.bib.upct.es/dspace/bitstream/10317/340/1/2004_AI_9.pdf

[32] Kevin C. Almeroth, University of California, *The Evolution of Multicast: From the MBone to Interdomain multicast to Internet2 Deployment*, [en línea], January/February 2000 [consulta Octubre 2014], Disponible en: http://www.cs.ucsb.edu/~almeroth/classes/F05.276/papers/evolution.pdf

[33] Stanford University of California, *The Brief History of Streaming media*, [en línea] [consulta Noviembre 2014], Disponible en: http://web.stanford.edu/class/ee398b/handouts/lectures/08-VideoOverNetworks.pdf

[34] Jon M. Peha, Senior Member IEEE, Yiwei Thomas Hou, Member, IEEE; *Streaming Video over the Internet: Approaches and Directions*, Vol.1, No.1, [en línea], February 2001, [consulta Noviembre 2014], Disponible en: http://inst.eecs.berkeley.edu/~ee290t/sp04/lectures/wu01streaming.pdf

[35] Network Working Group, H. Schulzrinne, S. Casner, Columbia University, RFC 3550, *RTP: Transport Protocol for Real-Time applicaations*, July 2003, [en línea]; Disponible en: https://www.ietf.org/rfc/rfc3550.txt

[36] CISCO CCNA 2, Clase 1; *Protocolos de enrutamiento dinámico*, [video en línea], 2014, Disponible en: https://www.youtube.com/watch?v=1qF8lbt_Vvc&list=PL2A7l6PiV52fTE_tyx-JztVduuF3pWcOg&index=1

[37] Network Working Group, J. Moy, RFC 1247, *OSPF Version 2*, July 1991, [en línea]; Disponible en: http://tools.ietf.org/html/rfc1247

[38] Fire Wolf, *Yo sé Networking*, April 2013, [en línea]; Disponible en: http://www.teleccna.cl/proyecto-yo-seacute-networking.html

[39] Cisco Systems; *Implementing Cisco Multicast*, (Student Guide) Vol.1, Version 1.1, 2002.

[40] Cisco, *IP Multicast Technology Overview*, September 2000, [en línea]; Disponible en: http://www.cisco.com/c/en/us/td/docs/ios/solutions_docs/ip_multicast/White_papers/mcst_ovr.html

[41] Cisco Support Community, *Understnding, Basics of Multicast RPF (Reverse Path Forwarding)*, March 2013, Disponible: https://supportforums.cisco.com/document/128461/understanding-basics-multicast-rpf-reverse-path-forwarding

[42] Network Working Group, S. Deering, RFC 4601, *Host Extensions for IP Multicasting*, August 1989, [en línea]; Disponible en: https://tools.ietf.org/rfc/rfc1112.txt

[43] Network Working Group, W. Fenner, Xerox PARC, RFC 2236, *Internet Group Management Protocol Version 2*, November 1997 [en línea]; Disponible en: https://tools.ietf.org/html/rfc2236

[44] Network Working Group, B. Fenner, M. Handley, I. Kouvelas, RFC 4601, *Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*, August 2006, [en línea]; Disponible en: https://tools.ietf.org/html/rfc4601

[45] Internet Engineering Task Force (IETF), B. Joshi, A. Kessler, RFC 6226, *PIM Group-to-Rendezvous-Point-Mapping*, May 2011, [en línea]; Disponible en: https://tools.ietf.org/html/rfc6226

[46] GNS3, 2015; [consulta Enero 2015]: http://www.gns3.com/

[47] Cisco Systems; *Cisco IOS Software Release 12.4(4)T*, 2014 [en línea]; Disponible en: http://www.cisco.com/c/en/us/support/ios-nx-os-software/ios-software-release-12-4-4-t/model.html

[48] Página oficial de GNS3 [en línea]; Disponible en:

https://community.gns3.com/community/software/download

[49] Página oficial VLC media player [en línea]; Disponible en: http://www.videolan.org/vlc/