

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA**

MAESTRÍA EN REDES DE COMUNICACIONES



TESIS

**IMPLEMENTACIÓN DE UN SERVICIO DE STREAMING DE VÍDEO SOBRE
UNA PLATAFORMA IMS -.IP MULTIMEDIA SUBSYSTEM.- UTILIZANDO
SOFTWARE LIBRE.**

CRISTHIAN RENÉ IZA PAREDES

Quito, 2011

Resumen

La penetración del internet banda ancha, el gran incremento de los terminales inteligentes y la convergencia de las infraestructuras en torno a IP están provocando un cambio radical en el sector de las Telecomunicaciones. En este nuevo panorama, la tecnología IMS (IP Multimedia Subsystem) juega un papel clave. Los usuarios han aumentado sus exigencias y los operadores han visto la necesidad de adaptarse ante estas nuevas exigencias. Las Redes de Siguiete Generación (Next Generation Networks: NGN) proponen solucionar estas exigencias al migrar al modelo de Plataforma de Entrega de Servicios.

El proyecto de Tesis presenta la plataforma IP Multimedia Subsystem (IMS) cómo una Plataforma de Entrega de Servicios en la adopción de redes NGN. Este documento detalla la construcción de un prototipo de maqueta IMS. Además describe el diseño e implementación del servicio Streaming de Video usando los bloques de servicio proporcionado por el prototipo. Este trabajo analiza la relación entre los conceptos teóricos y la experiencia práctica de gestionar una plataforma IMS.

Abstract

The penetration of Internet broad band, the great increase of the intelligent terminals and the convergence of the infrastructures concerning IP are provoking a radical change in the sector of the Telecommunications. This new panorama, the technology IMS (IP Multimedia Subsystem) plays a key paper. The users have increased his requirements and the operators have seen the need to adapt before these new requirements. The Next Generation Networks (NGN) proposes to solve these requirements on having migrated to the model of Platform of Delivery of Services

The project of Thesis presents the platform IP Multimedia Subsystem (IMS) how a Platform of Delivery of Services in the adoption of networks NGN. This document details the construction of a model IMS. In addition it describes the design and implementation of the service streaming of Video using the blocks of service provided by the model. This work analyzes the relation between the theoretical concepts and the practical experience of administering a platform IMS.

Agradecimientos

A mis padres

Tabla de Contenidos

Resumen	2
Abstract	3
Agradecimientos	4
Tabla de Contenidos	5
Índice de figuras	7
Índice de tablas	8
OBJETIVOS	16
1 CAPÍTULO I. ESTADO DEL ARTE	18
1.1 Introducción	18
1.2 Redes de Siguiete Generación	18
1.2.1 Definición NGN	18
1.2.2 Características	19
1.2.3 Arquitectura NGN	20
1.3 IP Multimedia Subsystem	21
1.3.1 Definición IMS	22
1.3.2 Evolución	22
1.3.3 Arquitectura IMS	25
1.4 Redes Móviles e IMS	28
1.5 Características IMS	28
1.6 Servicios IMS	30
1.7 Open IMS Core	31
1.8 Cliente IMS	32
1.9 Streaming	34
1.9.1 Transmisión multimedia	34
1.9.2 Video Streaming	35
1.9.3 Tipos de Streaming	36
1.9.4 Trasmisión de video Multicast	39
1.9.5 Beneficios de la tecnología Multicast	40
1.9.6 Operación básica del servicio Multicast	41
1.9.7 Parámetros básicos en la transmisión multimedia	44
1.9.8 Funcionamiento del servicio Streaming	47
1.9.9 Calidad de Video	48
1.10 Streaming de Video Open Source	52
1.10.1 Videolan	52
1.10.2 Darwin Streaming Server	53
1.11 Protocolos de Comunicación	54
1.11.1 Protocolo de Inicio de Sesión SIP	54
1.11.2 Real-Time Streaming Protocol. RTSP	56
1.11.3 Session Description Protocol. SDP	56
1.11.4 Protocolo de transmisión en tiempo real RTP y RTCP	58
1.11.5 Procolo de Datagrama de Usuario UDP	58
1.12 Trabajos relacionados	58
2 CAPÍTULO II. IMPLEMENTACIÓN DEL CORE DEL PROTOTIPO	61
2.1 Introducción	61
2.2 Instalación	61
2.2.1 Requerimientos Hardware y Software	62
2.2.2 Instalación Open IMS Core	63

3	CAPÍTULO III. IMPLEMENTACIÓN DEL SERVICIO DE STREAMING DE VIDEO	70
3.1	Introducción.....	70
3.2	Instalación Servidor Streaming de Video.....	70
3.3	Instalación Servidor Streaming de Video bajo demanda	75
3.4	Configuración Streaming de Video Multicast	77
3.5	Configuración UCT IPTV Streaming Server.....	78
4	CAPÍTULO IV. CONFIGURACIÓN DEL SERVICIO.....	81
4.1	Configuración Open IMS Core & Streaming server.....	81
4.1.1	Agregar el Application Server (AS) al registro HSS	81
4.1.2	Agregar un Trigger Point al registro HSS.....	82
4.1.3	Unir los pasos 1 y 2 con un Initial Filter Criteria (iFC).....	83
4.1.4	Agregar el iFC a un perfil de Servicio	83
4.2	Instalación y configuración de Open IMS Client	84
5	CAPÍTULO V. ESCENARIO DE PRUEBAS	88
5.1	Análisis Servidor Streaming de Video.....	88
5.1.1	Formatos soportados.....	88
5.1.2	Multicast	89
5.1.3	Unicast. Video Bajo demanda.....	90
5.1.4	Simulaciones	92
5.2	Análisis Plataforma Open IMS Core	98
5.3	Análisis del servicio de Streaming de video en la plataforma IMS.....	102
6	CAPÍTULO VI. CONCLUSIONES Y LÍNEAS FUTURAS.....	106
6.1	Conclusiones.....	106
6.2	Recomendaciones y trabajo futuro.....	110
7	BIBLIOGRAFÍA	111
8	ANEXOS	113
8.1	Anexo A. Protocolos Multicast.....	113
8.2	Anexo B. Protocolo SIP	118
8.3	Anexo C. Atributos del Protocolo UDP	122
8.4	Anexo D. Configuración IGMP Snooping en switch Enterasys.....	124
8.5	Anexo E. WireShark	125
8.6	Anexo F. Monitor de recursos servidor. Conky.....	127
8.7	Anexo G. Control de ancho de banda. Wondershaper	127

Índice de figuras

Figura 1. Next Generation Network	19
Figura 2. Arquitectura NGN	20
Figura 3. Evolución IMS.....	22
Figura 4. Evolución 3G hacia SAE-LTE.....	24
Figura 5. Vista simplificada de 3 capas. Arquitectura IMS.....	25
Figura 6. Redes Móviles e IMS.....	28
Figura 7. Componentes Open IMS Core	31
Figura 8. UCT IMS Client.....	34
Figura 9. Unicast	37
Figura 10. Multicast	38
Figura 11. Proceso básico de distribución Multicast.....	42
Figura 12. Play out, buffer en los paquetes recibidos	45
Figura 13. Streaming básico.....	47
Figura 14. Resolución CIF y 4CIF.....	48
Figura 15. Comparación Bit Rate.....	49
Figura 16. H.264/SVC [1Mbps] - MPEG-4 [2Mbps]	50
Figura 17. Comparación MPEG-2, MPEG-4, y H.264/SVC	51
Figura 18. Arquitectura Streaming de Video.....	52
Figura 19. Darwin Streaming Server. DSS	53
Figura 20. Protocolo SIP.....	54
Figura 21. Diagrama de Red del Prototipo	61
Figura 22. Diagrama Lógico prototipo IMS	62
Figura 23. Máquina Virtual para Open IMS Core.....	63
Figura 24. Servicio pcscf.sh.....	67
Figura 25. Servicio icscf.sh.....	68
Figura 26. Servicio scscf.sh.....	68
Figura 27. Servicio fhoss.sh	68
Figura 28. Consola de Administración Web. FHoSS	69
Figura 29. Log In .DSS	72
Figura 30. Server DSS Snapshot.....	72
Figura 31. Exportar videos MOV	73
Figura 32. Playlist DSS.....	74
Figura 33. Funcionamiento Video Bajo Demanda	76
Figura 34. Funcionamiento Streaming Multicast.....	77
Figura 35. Arquitectura prototipo IMS Core - Streaming de video	78
Figura 36. FHoSS. Interfaz web.....	81
Figura 37. Registro Application Server	82
Figura 38. Agregar un Trigger Point	82
Figura 39. Initial Filter Criteria (iFC).....	83
Figura 40. Configuración Shared iFC	83
Figura 41. UCT IMS Client v 1.0.12.....	84
Figura 42. Características de Hardware & Software Cliente 1.....	85
Figura 43. Características de Hardware & Software Cliente 2.....	85
Figura 44. Video UCT IMS Client.....	87
Figura 45. Diseño simplificado Servidor Streaming de Video- Clientes.....	88
Figura 46. Tramas Multicast	90
Figura 47. Tramas Unicast	91
Figura 48. Esquema de red para simulación	92

Figura 49. Analisis de recursos del servidor - tipo de streaming- Número de Clientes	93
Figura 50. Ancho de banda en la red Lan.....	94
Figura 51. Plataforma Open Ims Core - Streaming de Video	98
Figura 52. Tráfico IMS. Consulta DNS.....	99
Figura 53. Tráfico IMS. Register.....	100
Figura 54. Usuario No Registrado	100
Figura 55. Usuario Registrado.....	101
Figura 56. Tráfico IMS. Subscribe	101
Figura 57. UCT IMS Client v1.0.12.....	102
Figura 58. Análisis de Tráfico. Servicio Streaming de Video	103
Figura 59. Solicitud de Canal IPTV.....	103
Figura 60. Tráfico de Vídeo	104
Figura 61. sip:channel2@iptv.open-ims.test.....	105
Figura 62. Transacción SIP	118
Figura 63. Switch SIN IGMP Snooping.....	124
Figura 64. Switch con IGMP Snooping.....	124

Índice de tablas

Tabla 1. Clases direcciones IP	39
Tabla 2. Direcciones Multicast.....	40
Tabla 3. Campos SDP.....	57
Tabla 4. Requerimientos hardware y software para IMS	62
Tabla 5. Puertos utilizados Open IMS Core.....	69
Tabla 6. Hardware y Software Requerido	71
Tabla 7. Archivos de configuración DSS	73
Tabla 8. Formatos examinados Servidores Streaming de Video	89
Tabla 9. Clasificación MOS	95
Tabla 10. Parámetros de los videos de prueba	95
Tabla 11. Evaluación de calidad de video MOS para el servidor VLC en difusión VoD.....	96
Tabla 12. Protocolos de encaminamiento y soporte Multicast.....	117
Tabla 13. Clases de respuesta SIP	120

Marco conceptual

Los conceptos y términos más importantes que se van a utilizar en este proyecto de tesis son:

[3GPP] 3rd Generation Partnership Project. Esta organización realiza la supervisión del proceso de elaboración de estándares relacionados con 3G.

[Bluetooth] Estándar global de comunicación inalámbrica que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia segura.

[Bitrate] El número de bits procesados por unidad de tiempo, cuantificada en bits por segundo (bps) o en grandes incrementos de kilo bits por segundo (kbps), mega bits por segundo (Mbps), giga bits por segundo (Gbps) o Tera (Tbps)

[Codec] Un programa que codifica una corriente digital para la transmisión, el almacenamiento y la encriptación y / o decodifica una secuencia de datos digital para la visualización y edición de compresión / descompresión.

[CIF] (Common Intermediate Format) Define la resolución en términos de anchura y altura del fotograma de vídeo y la velocidad de fotogramas. CIF se indica como 352 x 240 píxeles en formato NTSC (National Television System Committee - televisión en formato estándar de vídeo utilizado en América del Norte). Otras resoluciones comunes son: 2 CIF (704 x 240), 4CIF (704 x 480).

[CONVERGENCIA]. Capacidad de diferentes plataformas de red de transportar tipos de servicios esencialmente similares. La convergencia puede entenderse en al menos dos sentidos complementarios: convergencia de servicios y convergencia de tecnologías. La convergencia de servicios se refiere a la confluencia de servicios, dentro de la infraestructura de telecomunicaciones de un mismo proveedor, que se entendían como independientes y provistos cada uno de ellos por un operador de telecomunicaciones distinto. La convergencia tecnológica se refiere a la integración, dentro de un mismo equipo de telecomunicaciones, de tecnologías identificadas con servicios específicos.

[GPRS] (General Packet Radio Service) Sistema para la transmisión de datos no conmutada. Esta tecnología permite los siguientes servicios: servicios de mensaje multimedia, mensajería instantánea, aplicaciones wap, servicios P2P, servicio de mensajes cortos, posibilidad de usar el dispositivo como módem USB. Los sistemas móviles de segunda generación (2G), combinados con la tecnología GPRS reciben el nombre de 2.5G.

[GSM] Sistema global para las comunicaciones móviles, por sus características se considera un estándar de segunda generación (2G). Esta tecnología permite llamadas, mensajes cortos, servicios de localización, y aplicaciones wap.

[IANA] La Internet Assigned Numbers Authority, es la Agencia de Asignación de Números de Internet. Era el antiguo registro central de los protocolos Internet, como puertos, números de protocolo y empresa, opciones y códigos. Fue sustituido en 1998 por ICANN.

[ICANN].- Internet Corporation for Assigned Names and Numbers, es una organización que opera a nivel internacional, siendo la responsable de asignar las direcciones del protocolo IP, de los identificadores de protocolo, de las funciones de gestión del sistema de dominio y de la administración del sistema de servidores raíz. En un principio, estos servicios los realizaba IANA (Internet Assigned Numbers Authority) y otras entidades del gobierno estadounidense. ICANN se dedica a preservar la estabilidad de Internet por medio de procesos basados en el consenso.

[IMS] (IP Multimedia Subsystem). El despliegue de este elemento en la red ofrece la convergencia de servicios ya que es una arquitectura que permite tener servicios multimedia sobre IP. Además, soporta múltiples tipos de tecnologías de acceso, como GSM, GPRS, UMTS, HSDPA, DSL, HFC, Wi-Fi, Wi-MAX, Bluetooth, etc.

[IPTV] (Internet Protocol Television) Se ha convertido en la denominación más común para los sistemas de distribución por suscripción de señales de televisión y/o vídeo usando conexiones de banda ancha sobre el protocolo IP.

[LATENCIA] Es el tiempo o lapso necesario para que un paquete de información se transfiera de un lugar a otro. La latencia, junto con el ancho de banda, son determinantes para la velocidad de una red.

[NGN] (Next Generation Network - Red de siguiente generación) Es una arquitectura de red orientada a reemplazar las redes telefónicas conmutadas de telefonía, para brindar servicios de voz y multimedia.

[SIP] (Session Initiation Protocol) Es un protocolo desarrollado con la intención de ser el estándar para la iniciación, modificación y finalización de sesiones interactivas de usuario donde intervienen elementos multimedia como el video, voz, mensajería instantánea, juegos en línea y realidad virtual.

[IETF] (Internet Engineering Task Force - Grupo de Trabajo en Ingeniería de Internet) Es una organización internacional abierta de normalización, que tiene como objetivos el contribuir a la ingeniería de Internet, actuando en diversas áreas, como transporte, encaminamiento, seguridad. Fue creada en EE. UU. en 1986. La IETF es mundialmente conocida por ser la entidad que regula las propuestas y los estándares de Internet, conocidos como RFC.

[ITU – T] La misión de ITU-T es asegurar la producción eficiente y oportuna de estándares que cubren todos los campos de telecomunicaciones sobre una base mundial, así como la definición de los principios de la tarifa y de contabilidad para los servicios de telecomunicaciones internacionales.

[JITTER] Cambio o variación en cuanto a la cantidad de latencia entre paquetes de datos que se reciben.

[MULTICAST] Es el envío de información en una red a múltiples receptores de forma simultánea, un emisor envía un mensaje y son varios los receptores que reciben el mismo.

[OPEN IMS CORE] Es una implementación del Instituto Fraunhofer, y es parte del proyecto “Open IMS Playground @ FOKUS”, un marco de desarrollo de aplicaciones para la tecnología IMS en el que participan tanto el Instituto como compañías para probar funcionalidades IMS. FOKUS implementa los componentes del core de IMS.

[PIXEL] Es la porción más pequeña de información en una imagen.

[RESOLUCIÓN] Número de píxeles en cada dimensión que puede ser visualizada en la pantalla. La resolución puede ser especificada como el número de columnas de píxeles (ancho) por el número de filas de píxeles (la altura).

[TISPAN] (Telecoms & Internet converged Services & Protocols for Advanced Networks). Es una rama de estandarización de ETSI (European Telecommunications Standards Institute). TISPAN se encarga de todos los aspectos relativos a la estandarización de la convergencia de redes, en el presente y en el futuro, incluyendo la NGN, considerando aspectos de servicio, de arquitectura, de protocolos, de seguridad y de movilidad dentro de las redes fijas. Todo ello, usando tecnologías existentes, así como nuevas tecnologías que puedan surgir.

[TELEFONÍA IP] La Telefonía IP es una aplicación inmediata de la Voz sobre IP, de forma que permite la realización de llamadas telefónicas sobre redes IP u otras redes de paquetes utilizando un PC, Gateway o teléfonos estándares. En general, servicios de comunicación - voz, fax, aplicaciones de mensajes de voz - son transportados vía redes IP, Internet normalmente, en lugar de ser transportados a través de la red telefónica convencional.

[QoE] (Quality of Experience). Es una medida de extremo a extremo del desempeño del sistema, realizada en la capa de servicio y desde la perspectiva del usuario, además indica el grado en que el sistema satisface las necesidades del usuario.

[QoS End-to-End] (Quality of Service). Es un conjunto de requisitos de servicio que la red debe cumplir para asegurar un nivel de servicio adecuado para la transmisión de los datos.

[UMTS] (Universal Mobile Telecommunications System - Sistema universal de telecomunicaciones móviles). Tecnología que permite la transmisión a grandes velocidades, es considerada por sus características de tercera generación (3G). Servicios UMTS son: internet, correo móvil, mensajería multimedia, video llamadas, etc.

[UNICAST] Hace referencia al envío de paquetes o información desde un único emisor a un único receptor. Ejemplos básicos de aplicaciones unicast son los protocolos http, smtp, ftp o telnet. Actualmente es la forma predominante de transmisión en Internet.

[VLC] Es un reproductor multimedia avanzado del proyecto VideoLAN, y cuenta con versiones para varias plataformas, con ediciones compiladas para Linux, Windows Vista, Mac OS X, BeOS, BSD, Pocket PC, Solaris. Soporta varios códecs de audio y video, así como diferentes tipos de archivos, además soporta los formatos de DVD, VCD y varios protocolos streaming. También puede ser utilizado como servidor en IPv4 o IPv6, en redes de banda ancha.

[VoIP] (Voice over IP - Voz sobre IP) Es una tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos.

[VoD] (video on demand). Es un sistema de televisión que permite al usuario el acceso a contenidos multimedia de forma personalizada ofreciéndole, de este modo, la posibilidad de solicitar y visualizar una película o programa concreto en el momento exacto que el telespectador lo desee.

[Wi-fi] Sistema de envío de datos inalámbrico, utilizando ondas de radio.

[WiMax] (Worldwide Interoperability for Microwave Access - Interoperabilidad Mundial para Acceso por Microondas). Es un sistema parecido a Wi-fi pero con mayor cobertura y ancho de banda.

INTRODUCCIÓN

Debido al crecimiento del Internet y al incremento en la demanda de información multimedia, el servicio de streaming de video ha generado un gran interés tanto en sectores académicos como industriales, comerciales y de negocios. Los requisitos de red que los servicios multimedia requieren son más exigentes que para otro tipo de servicios. Disminuciones en el ancho de banda disponible o variaciones de latencia en recepción, traen consecuencias en la calidad del servicio, con interrupciones, pérdidas de calidad y saltos muy notables de audio y vídeo. [1]

El servicio de streaming de video en una red conmutada de paquetes (packed switching network) está siendo estandarizado basándose en las partes de control y transporte de los protocolos RTSP (Real Time Streaming Protocol), RTP (Real Time Transport Protocol), RTCP (Real Time Control Protocol) y SDP (Session Description Protocol). Estos protocolos sientan las bases para los servicios integrados en tiempo real. [2]

La evolución de las Redes Móviles de Tercera Generación incorpora un nuevo subsistema para el desarrollo de Servicios Multimedia y Peer-to-Peer. Esta arquitectura del 3GPP se denomina IMS (IP Multimedia Subsystem) y se basa en el protocolo de control SIP (Session Initiation Protocol), estandarizado por el IETF. Las Redes de Próxima Generación (NGN) introducen un nuevo modelo donde los servicios son independientes de las tecnologías que la transportan. IP Multimedia Subsystem, en adelante se denotará como IMS surge por la necesidad de integrar los servicios de transmisión de video, audio y datos de alta velocidad, con las redes de telefonía móvil de tercera generación 3G. [3]

El Instituto Fraunhofer for Open Communications Systems está desarrollando implementaciones del núcleo IMS usando Software Libre y Abierto, lo que permite a los interesados probar las funcionalidades de IMS. Open IMS Core es un entorno de pruebas de la plataforma IMS, abierto e independiente para el ambiente académico y productivo. El objetivo fundamental de esta aplicación es desarrollar prototipos, protocolos y aplicaciones relacionados con IMS; así como realizar pruebas y mediciones de los componentes existentes [4].

La implementación de un servicio de streaming de video sobre un prototipo de la plataforma IMS permitirá: una exploración del modelo de redes de próxima generación (NGN), conocer alternativas de software libre para la emisión de información multimedia mediante la técnica de Streaming de audio o video/audio y la documentación relacionada facilitará futuros proyectos de investigación a fin de agregar funcionalidades al prototipo.

De lo expuesto, se considera que el proyecto de investigación es factible porque se cuenta con los conocimientos, recursos técnicos y materiales necesarios.

Los resultados que se obtendrán con el desarrollo del proyecto de tesis son:

- Conocimiento del estándar IP Multimedia Subsystem, así como los protocolos utilizados, y las relaciones de los elementos que la componen.
- Análisis de las razones por las cuales la plataforma IP Multimedia Subsystem es reconocida como la tecnología ideal para la convergencia de servicios de voz, y multimedia.
- Experiencia práctica con la plataforma Open IMS Core.
- Conocimiento del rendimiento de la plataforma Open IMS Core bajo determinados escenarios de tráfico de datos.
- Conocimiento de alternativas de software libre para la emisión de información multimedia mediante la técnica de Streaming de video en redes NGN.

Planteamiento del problema

La tendencia actual de los modelos de las Redes de Próxima Generación (NGN) hacia un mundo todo-IP ha remarcado las falencias que el protocolo IP provee para: el transporte de multimedia en tiempo real, la seguridad, la calidad de servicio (QoS) entre otras. Así, el desarrollo de una infraestructura IMS permitirá a los operadores de telecomunicaciones migrar de forma paulatina a un modelo horizontal de red convergente, capaz de dar soporte a todos los servicios (voz, datos, video, multimedia) tanto actuales como futuros independientemente del tipo de red desde la que accede el cliente.

Por lo expuesto se formula el problema:

¿CÓMO CONVERGE EL SERVICIO DE STREAMING DE VIDEO EN UNA PLATAFORMA DE COMUNICACIONES IP MULTIMEDIA SUBSYSTEM?

OBJETIVOS

General:

Diseñar e implementar el servicio de streaming de vídeo en una plataforma de referencia IP Multimedia Subsystem en la que sea posible desarrollar y experimentar el servicio.

Específicos:

- Proveer el marco conceptual y la descripción de la plataforma IMS.
- Implementar un prototipo de plataforma IMS utilizando software libre.
- Analizar la interoperabilidad de la plataforma Open IMS Core con tecnologías de streaming de video.
- Analizar los protocolos de red que intervienen en la plataforma Open IMS Core.
- Analizar los protocolos de red que intervienen en el streaming de video.
- Analizar el rendimiento del servicio de streaming de vídeo sobre la plataforma Open IMS bajo determinados escenarios de tráfico de datos.

ESTRUCTURA DEL DOCUMENTO

El presente documento se encuentra estructurado en seis capítulos. A continuación se describe el contenido de cada capítulo:

CAPÍTULO I. ESTADO DEL ARTE.

Se describe el entorno tecnológico en el que la tesis se va a desarrollar. En el caso de esta investigación, el núcleo será el IMS, cuya arquitectura se describirá de forma detallada, así como los protocolos que participan en el sistema. Además, se analizarán aplicaciones open source para streaming de video

CAPÍTULO II. IMPLEMENTACIÓN DEL CORE DEL PROTOTIPO

Se definen los requisitos del sistema para la implementación. Una vez establecidos los requisitos, el capítulo se concentra en la implementación del Open IMS Core. Se describirán las herramientas y los procedimientos utilizados.

CAPÍTULO III. IMPLEMENTACIÓN DEL SERVICIO DE STREAMING DE VIDEO

Se detalla los procedimientos para implementar el servicio de streaming de video en el prototipo de la arquitectura IMS.

CAPÍTULO IV. CONFIGURACIÓN DEL SERVICIO

En este capítulo se anexan el servicio implementados en el capítulo 3 en el core de la plataforma IMS. Se describirán los procedimientos utilizados.

CAPÍTULO V. ESCENARIO DE PRUEBAS

En este capítulo se define un escenario de pruebas donde se valida el funcionamiento del servidor de Streaming de video. De igual forma se realiza el análisis del funcionamiento del Open IMS Core como prototipo de una plataforma IP Multimedia Subsystem.

CAPÍTULO VI. CONCLUSIONES Y LÍNEAS FUTURAS

Se hace una presentación de las conclusiones extraídas de la realización del proyecto y se indican las líneas futuras de investigación que pueden ser identificadas con el presente trabajo.

LISTADO DE ANEXOS. Se adjuntan un listado de anexos como detalle a ciertos capítulos del documento.

1 CAPÍTULO I. ESTADO DEL ARTE

1.1 Introducción

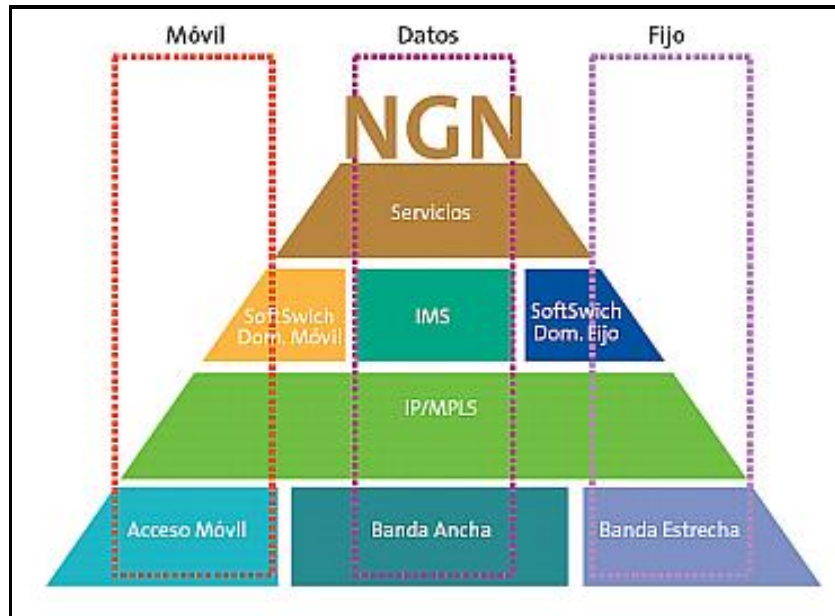
En este capítulo se presenta una descripción detallada del entorno tecnológico en el que se ha desarrollado el proyecto para la implementación de un servicio de streaming de vídeo sobre una plataforma IMS. Se hará un estudio de la situación actual de las NGN, la evolución del sistema IMS, los elementos que forman su arquitectura, los protocolos empleados en el sistema y los conceptos relacionados con el streaming de video.

1.2 Redes de Siguiete Generación

En la actualidad, cada proveedor de servicios tiende su red por separado de acuerdo a sus necesidades: telefonía, internet, Televisión por Cable (CATV); por lo que en el domicilio de los usuarios convergen diversas redes, cada una para el servicio que les dio origen. Los proveedores han adaptado dichas redes para agregar nuevos servicios y aprovechar la red instalada en el domicilio del usuario. Así nacen tecnologías como Cable Módem para usar las redes de CATV y ADSL para usar las redes telefónicas, y brindar el servicio de Internet a los usuarios; sin embargo, estas tecnologías no satisfacen los requerimientos necesarios para optimizar el servicio agregado; esto dio origen al concepto de tecnologías de Redes de Nueva Generación (NGN). La evolución hacia este tipo de redes implica para los operadores la innovación continua de su oferta de servicios y redes con el fin de satisfacer las necesidades de los usuarios. La convergencia de servicios, aplicaciones y dispositivos impulsa esta tendencia, para beneficio del cliente, pues obtiene cada vez más y mejores servicios, a un costo competitivo.

1.2.1 Definición NGN

El concepto NGN se presenta como una solución para la convergencia de redes con interfaces de alta velocidad, con seguridad y calidad garantizadas facilitando el despliegue de los servicios tanto actuales como futuros. La Unión Internacional de Telecomunicaciones define a las NGN como: ***“Una red por paquetes que proporciona múltiples servicios de banda ancha, que utiliza tecnologías de transporte con una calidad de servicio mínima y en la cual las funciones relacionadas con el servicio son independientes de las tecnologías de transportes subyacentes.”***



Fuente: <http://sociedadinformacion.fundacion.telefonica.com/>

Figura 1. Next Generation Network

1.2.2 Características

Las características más importantes de una Red de Nueva Generación son:

- La NGN provee infraestructura para la creación, desarrollo y gestión de toda clase de servicios actuales y futuros, distinguiendo y separando los servicios y las redes de transporte; es decir posee una arquitectura de red horizontal basada en una división transparente de los planos de transporte, control y aplicación.
- El plano de transporte se basa en tecnología de conmutación de paquetes IP/MPLS
- Migración de las redes actuales (PSTN, ISDN y otras) a NGN, a través de interfaces abiertos y protocolos estándares (SoftSwitch).
- Escalabilidad de la infraestructura de red; esto implica permitir la ampliación de la red de acuerdo a las necesidades, teniendo en cuenta la cantidad de usuarios y la variedad de servicios a ofrecer en cada etapa de su desarrollo.
- Soporte de servicios de diferente naturaleza: tiempo real y no real, streaming, servicios multimedia (voz, video, texto).
- Soporte para múltiples tecnologías de última milla.
- Su arquitectura funcional soporta la conexión a red basada en tres modos de conmutación: de circuitos, de paquetes y de paquetes sin conexión.
- Posibilitar la distribución simultanea de diferentes servicios, como telefonía, televisión, acceso a Internet, datos y otros servicios de valor agregado.

1.2.3 Arquitectura NGN

La figura 2 muestra el diagrama de una arquitectura lógica de una red NGN. Se puede identificar las **Funciones de Mantenimiento** y el Equipo **Terminal del Usuario (UE)**. Además se identifica un bloque de **Otras Redes**, ya que las NGNs deben interactuar con otras redes. La característica del modelo es la separación del plano de servicio y transporte con el objetivo de hacer los servicios independientes de los protocolos de transporte.

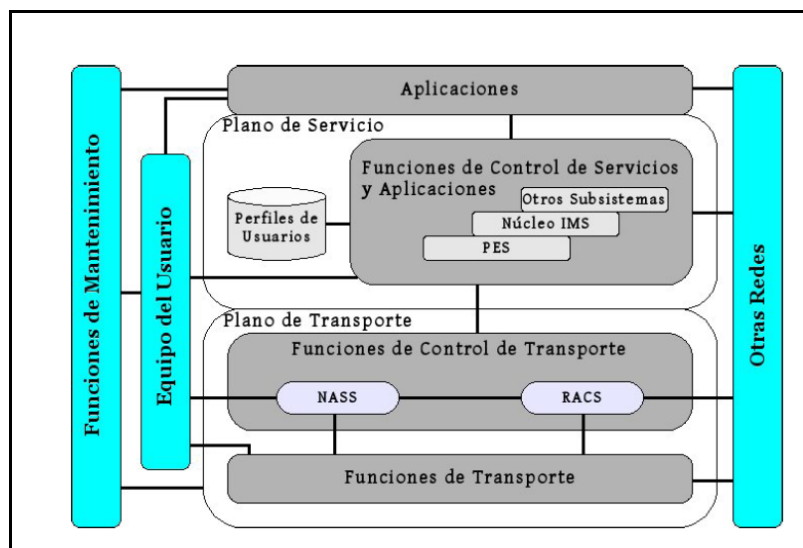


Figura 2. Arquitectura NGN

Fuente: [6]

Plano de Transporte

Este plano es responsable del transporte de información dentro de la red. Además debe permitir la interconectividad de la red NGN con otras redes vía interfaces abiertas. Este plano está formado por dos subsistemas:

Network Attachment Subsystem (NASS) es el responsable de establecer las reglas con las cuales se comunica el UE con la red. Esto se logra mediante la autenticación y autorización de las configuraciones del UE junto con la asignación dinámica y reservación de recursos vía IP. NASS es el encargado de configurar la red en función de las necesidades del UE.

Resource and Administration Control Subsystem (RACS) se encarga de reservar los recursos que requiere el UE, así como la administración y control de dichos recursos. RACS esencialmente es el encargado de establecer las políticas entre los planos de señalización, multimedia y distintos gateways para lograr reservar los recursos que requiere el usuario. Además es el responsable de administrar la creación y destrucción de dichos recursos de manera óptima.

Plano de Servicio

El perfil de los usuarios y los subsistemas que ofrecen los servicios son las principales secciones de este plano. La convergencia requiere una red inteligente que identifique a usuarios y no a dispositivos. Este plano es el encargado de almacenar toda la información acerca del usuario, los servicios a los que está suscrito y sus preferencias en cuanto a la interacción deseada con la red.

Desde el punto de vista de la red, los usuarios son un conjunto de UEs con un número de servicios disponibles y preferencias que describen cómo desean la entrega de los servicios según el dispositivo con el cual se conectan a la red. Por esta razón se incluye el perfil de los usuarios en el plano de servicio. Los subsistemas con los que cuenta una NGN para cambiar radicalmente la entrega de servicios son el subsistema PSTN/ISDN Emulation and Simulation (PES), IMS y otros subsistemas. Éstos no están definidos dentro de la arquitectura NGN, pero pueden ser cualquier subsistema capaz de ofrecer servicios. A continuación se describe en detalle el subsistema IP Multimedia (IMS)

1.3 IP Multimedia Subsystem

IMS es un estándar internacional aún en evolución que define una arquitectura genérica para ofrecer servicios de voz sobre IP y multimedia. Forma parte del núcleo de la arquitectura de redes de nueva generación NGN. Con IMS los operadores pueden combinar: la calidad y la interoperabilidad del mundo de las telecomunicaciones con el rápido e innovador desarrollo de Internet. IMS proporciona servicios multimedia sobre redes: PSTN, RDSI, Internet y redes móviles (GSM, CMDA, etc) usando estándares abiertos.

IMS canaliza temas como convergencia, creación y entrega de servicios, interconexión de éstos y estándares abiertos. A diferencia de los sistemas celulares de Tercera Generación (3G) que ya cuentan con conectividad hacia Internet, IMS busca una mejor integración que permita el control de QoS en el tipo de contenido que se le entrega al usuario.

1.3.1 Definición IMS

IMS se define como una especificación del 3GPP que permite el provisionamiento de servicios con mecanismos de calidad de servicio (QoS) y facilita el establecimiento de sesiones multimedia en redes de IP tales como: voz sobre IP, Push to Talk, Video Calling, Video Streaming, Video Sharing, Audio Streaming, etc.

1.3.2 Evolución

A pesar de ser un principal catalizador para el despliegue de redes NGN, IMS fue creado bajo la visión de los operadores de telefonía móvil. Surge como una solución para integrar servicios del Internet con redes de telefonía móvil. 3GPP realiza las especificaciones técnicas públicas de los sistemas 3G.

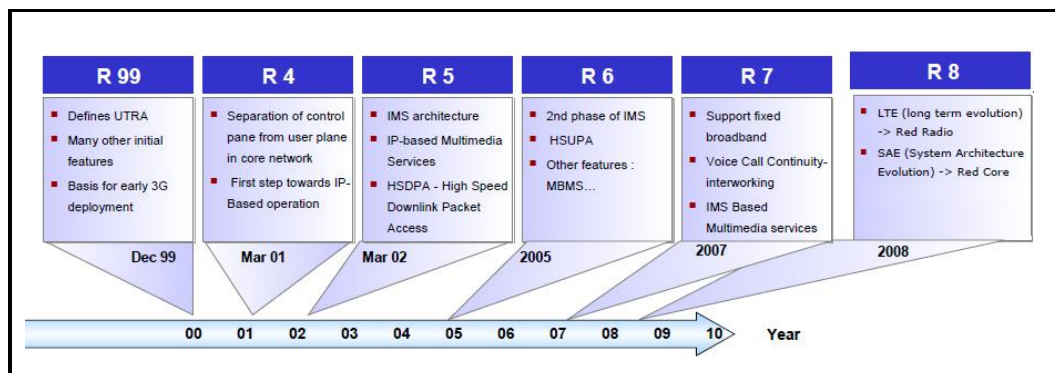


Figura 3. Evolución IMS

Fuente: [15]

IMS fue especificado en la Release 5 y 6 del 3GPP. Además de gestionar el establecimiento de sesión, IMS también se hace cargo de otras funcionalidades tales como el registro, seguridad, facturación, roaming, etc. Además en el Release 5 se definen los interfaces que conectan los distintos elementos de la arquitectura y los protocolos participantes en ella.

La Release 6 añade aspectos y funcionalidades como el Push To Talk (PTT). Además especifica la definición de servicios y la interconexión a otro tipo de redes de acceso como las WLAN.

En el Release 7 aparece HSPA (High Speed Packet Access, es la tecnología empleada en las conexiones de internet móvil) que incorpora sobre todo mejoras en el interfaz radio aumentando el throughput (hasta 14 Mbps) y mejorando la capacidad del medio haciendo posible la transmisión de la voz sobre ip en tiempo real y con unas prestaciones aceptables de retardo y jitter.

La mejora en el interfaz radio se debe a la aparición de antenas tipo MIMO (Multiple Input, Multiple Output) que posibilitan el envío de datos a unas tasas superiores a las obtenidas en Releases anteriores. En la parte del core se sigue manteniendo la estructura del IMS pero aparece nuevas funcionalidades tales como la continuidad y mantenimiento de las conversaciones de voz cuando un usuario migra de una red a otra (por ejemplo de una WLAN a una 3G).

Las últimas contribuciones del 3GPP son la Release 8, 9 y 10 que supone un cambio importante en la arquitectura de las redes 3G. En el core aparecen elementos que se integran y se conectan a los nodos de las anteriores releases mediante la definición de nuevos interfaces.

En la radio se introduce el evolved NodeB (eNodeB) que emplea una modulación en el interfaz radio (OFDM Orthogonal Frequency Division Multiplex) totalmente distinta a las de las anteriores releases.

Aparece un nuevo interfaz que permite conectar los elementos del core con el IMS. La arquitectura resultante es el EPS (Evolved Packet System) que comprende la definición de la arquitectura en la parte radio LTE (Long Term Evolution) y la arquitectura en el core de red EPC (Evolved Packet Core).

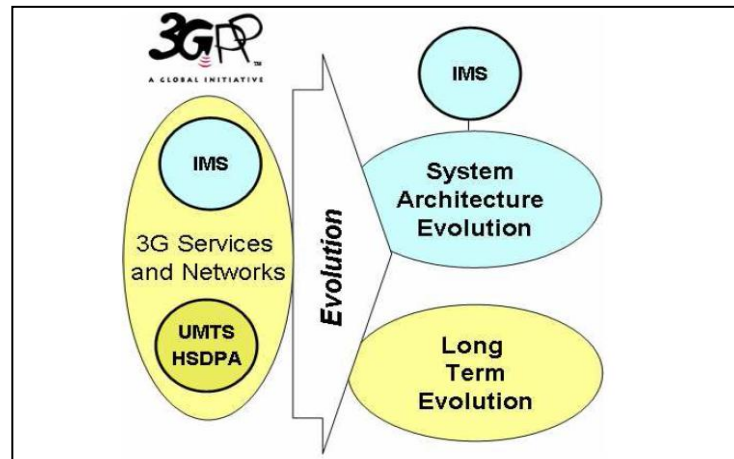


Figura 4. Evolución 3G hacia SAE-LTE

Fuente: [14]

LTE es la primera tecnología íntegramente de red de protocolo de Internet (Internet Protocol, IP) apoyada por la industria de las comunicaciones móviles y se usará, en principio, para brindar servicios de datos móviles muy rápidos y altamente eficaces.

Para que LTE sea compatible con los servicios de voz y mensajería, se necesita una solución basada en IP que ofrezca la misma experiencia interoperable y sin problemas que ofrecen hoy las tecnologías inalámbricas 2G y 3G. El subsistema multimedia IP (IMS) puede lograr esto y es compatible con todas las funciones del servicio de llamadas de voz, como llamada en espera, retención de llamada y restricción de llamadas. Además, es altamente escalable para prestar servicio a bases muy grandes de abonados.

1.3.3 Arquitectura IMS

El core de IMS es definido como una red de capas consistente en: Capa de transporte, Capa de Control y la Capa de aplicación.

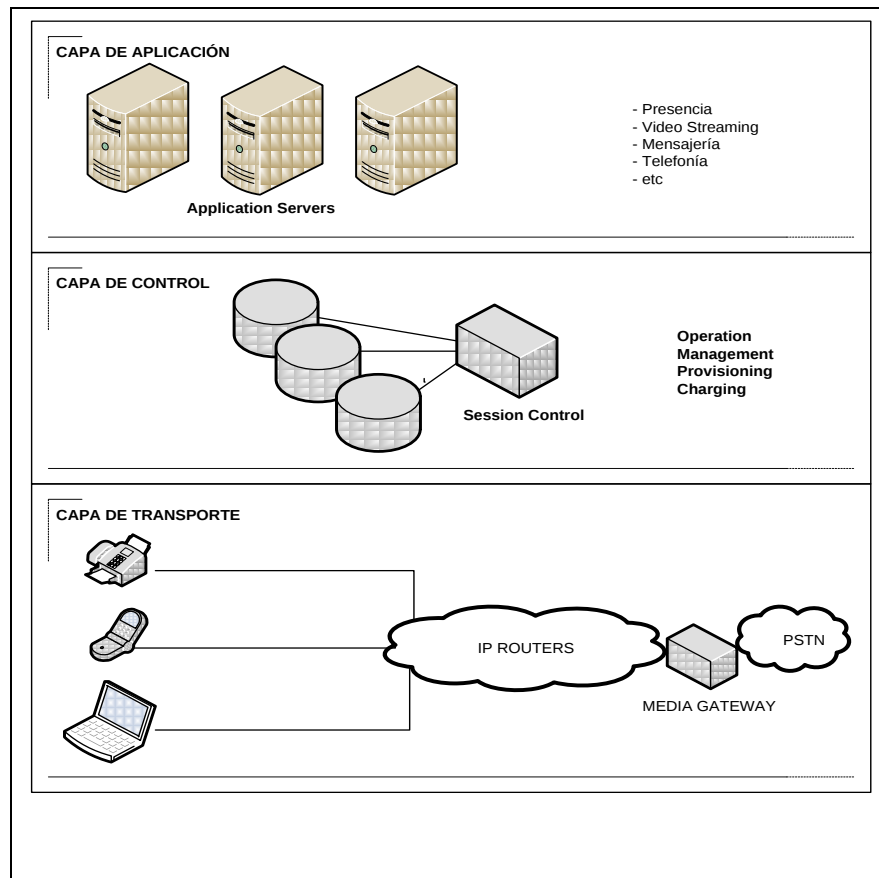


Figura 5. Vista simplificada de 3 capas. Arquitectura IMS

Capa de Transporte

Responsable del acceso a la red. Permite conectar a la red IMS diferentes dispositivos y usuarios. Los dispositivos IMS pueden conectarse a la red en la capa de transporte utilizando diferentes técnicas como: acceso fijo (DSL, módems de cable Ethernet, etc.), acceso móvil (WCDMA wideband code division multiple access, GPRS) y acceso inalámbrico (WiMax o WLAN). Esta capa también permite a los dispositivos IMS recibir y hacer llamadas a y desde las redes telefónicas públicas conmutadas (PSTN) o cualquier otro circuito de red conmutada mediante pasarelas. Además esta capa establece conectividad IP al usuario del equipo. Después de obtener una dirección IP y la capacidad de intercambiar mensajes SIP, el equipo será responsable de la interacción IMS, de manera independiente de la red subyacente tecnología de acceso.

Capa de Control

Esta capa es el home de diversas funciones de control de sesión de llamada CSCFs. Tres servidores SIP manejan estas funciones: Proxy CSCF (P-CSCF), Interrogating CSCF (I-CSCF) y Serving CSCF (S-CSCF):

Proxy CSCF o P-CSCF

Es el primer punto de contacto del usuario dentro de IMS. Todo el tráfico de señalización de usuario atraviesa el P-CSCF. Además se encarga de procesar los mensajes que envía el usuario validando la petición y reenviándola a los destinatarios.

Interrogating CSCF o I-CSCF

Es el primer punto de contacto en la red para todas las conexiones destinadas a un usuario de esa red. Pueden existir varias I-CSCF dentro de una red. Una de sus funciones principales es asignarle un Serving-CSCF a un usuario durante el proceso de registro, preguntándole al HSS por la localización de uno adecuado basándose en parámetros como capacidad y carga.

Otra función importante es la redirección de las peticiones SIP durante una sesión al S-CSCF del usuario que envía dicha petición. Este es un nodo opcional en la arquitectura de IMS. El sistema podría ser configurado de modo que el P-CSCF pudiera entrar en contacto con el S-CSCF directamente.

El I-CSCF realiza un balanceo de carga entre S-CSCFs con la ayuda del HSS. El I-CSCF, además, puede ocultar la configuración y topología específica de la red origen de otros operadores de red proporcionando un solo punto de entrada en la red.

Las funciones realizadas por la entidad I-CSCF incluyen:

- La asignación de un S-CSCF a un usuario en el proceso de registro.
- El encaminamiento de los métodos SIP recibidas desde otra red, al S-CSCF.
- La obtención de la dirección del S-CSCF por parte de HSS.

Serving CSCF o S-CSCF

Se puede considerar como el cerebro del IMS y se localiza en la red del operador. Realiza funciones de control de sesión y registro del usuario. Mientras que un usuario se encuentra activo en la sesión el S-CSCF mantiene el estado de la sesión e interactúa con otras plataformas para llevar por ejemplo el control de la tarificación.

Home Subscriber Server (HSS)

Es la entidad encargada de almacenar los datos de usuario y de servicios del IMS. Incluye información sobre identidades de usuarios, información de registro, parámetros de acceso y servicios.

Las identidades de los usuarios pueden ser públicas o privadas. La identidad privada se asigna en la red del operador donde el usuario reside y se utiliza para los procesos de registro y autorización mientras que la identidad pública es aquella que otros usuarios pueden utilizar para contactar con el usuario en cuestión. Los parámetros de acceso del IMS se usan para establecer sesiones y enviar información de autenticación, roaming y los posibles S-CSCF a los que se puede conectar el usuario de manera que el I-CSCF utilizará esta información para seleccionar el S-CSCF apropiado para el usuario.

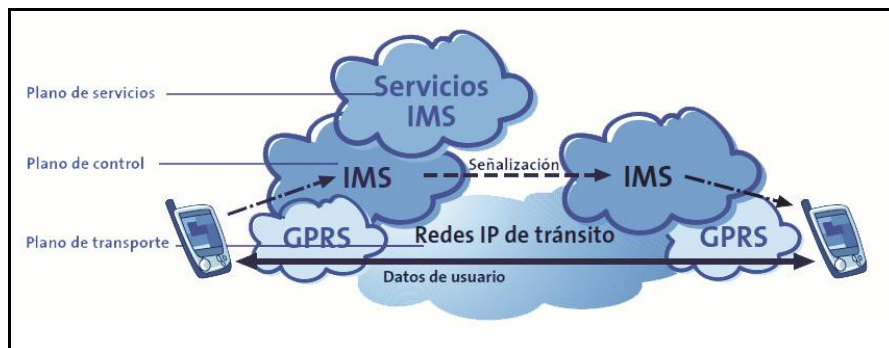
Puede ocurrir que haya más de un HSS en la red de manera que tendremos que saber a que HSS está asociado un determinado usuario. Para ello se utiliza el Subscription Locator Function (SLF) que determina a que HSS está vinculado el usuario.

Capa de aplicación

La capa de transporte y la capa de control proveen una plataforma de red estandarizada e integrada que permite a los proveedores de servicios ofrecer diversos servicios multimedia en la capa de aplicación. Los servidores de Aplicaciones proveen la interfaz con la capa de control usando el protocolo SIP

1.4 Redes Móviles e IMS

El modelo teórico de las redes móviles divide el sistema en diferentes dominios y planos. IMS implementa el plano de control de los servicios IP multimedia, mientras que GPRS 3G proporciona las funciones de plano de transporte, tal como se puede ver en la Figura 6.



Fuente: Las Telecomunicaciones y la Movilidad en la Sociedad de la Información

Figura 6. Redes Móviles e IMS.

De esta forma, la función de IMS en los servicios IP multimedia incorpora el procesamiento y actuación sobre la señalización y el control de los recursos del plano de transporte. GPRS 3G se encarga de proporcionar conectividad IP dentro del dominio de la red móvil, tanto para la señalización como para los datos multimedia de usuario. Siguiendo la filosofía establecida en la separación funcional típica de UMTS y de las redes IP, el flujo de datos correspondiente a la señalización viaja de forma separada a los flujos de datos multimedia de usuario, proporcionando en ambos casos GPRS 3G el servicio de transporte. Sin embargo, el flujo de señalización, que tiene su origen y fin en los terminales de usuario participantes en el servicio, es transferido vía IMS, al contrario de lo que ocurre con los datos de usuario que se transfieren directamente entre los terminales sin atravesar el subsistema IMS.

Este concepto establecido en la arquitectura permite, fundamentalmente, emplear tecnologías de Internet, y origina el atractivo de disponer de los servicios IP multimedia. Por otro lado, como IMS fue diseñado para una red evolucionada de GSM hereda ciertas cualidades intrínsecas del mundo móvil.

1.5 Características IMS

- La comunicación orientada a sesión de un usuario a otro(s) usuario(s), o de un usuario a un servicio.
- La comunicación en tiempo real o diferido.

- Las sesiones IP multimedia compuestas por flujos y contenidos multimedia diversos, con un nivel adecuado de Calidad de Servicio para vídeo, audio y sonido, texto, imagen, datos de aplicación, etc.
 - La identificación de usuarios, servicios y nodos mediante URIs (Universal Resource Identifier), que aumenta la usabilidad de los servicios de cara a los abonados. Éstos deben manejar nombres al estilo de servicios Internet, como el correo electrónico.
 - Los servicios IMS pueden implementarse en una sola aplicación de usuario final que hace un uso coordinado y simultáneo de la mensajería IP multimedia (instantánea y diferida), del servicio de presencia, de los servicios web, de la videoconferencia y llamadas de voz sobre IP, del streaming, de la difusión multimedia, de la descarga de contenidos, de los juegos en red y de cualquier otro servicio de Internet basado en TCP/IP, de forma muy similar a cómo operan las últimas versiones de los populares clientes de mensajería instantánea para PC e Internet, pero ofreciendo una Calidad del Servicio (QoS) garantizada y adaptada a cada flujo de datos, a la vez que permiten al usuario disfrutar de la movilidad y características de su dispositivo personal
- 3G IMS
- Los abonados de un operador IMS tienen la posibilidad de cursar sesiones IP multimedia con abonados localizados en la red 3G IMS de otro operador. La arquitectura de IMS, las entidades funcionales y sus protocolos, están diseñados para la interconexión con otros sistemas IMS de otros operadores. En este caso, los flujos de datos de una sesión se transportan por el sistema GPRS en la red origen, por una red, o redes, IP de tránsito interoperador y por la red GPRS destino. En el caso de las sesiones interoperador, los flujos de señalización siempre recorrerán ambos sistemas IMS.
 - IMS soporta la itinerancia (roaming) de tipo nativo, lo que se define como la capacidad del sistema de admitir y dar servicio a abonados de otros operadores que emplean la misma tecnología, y con los que se tiene el acuerdo de negocio pertinente. Cuando un abonado está en itinerancia, el subsistema IMS visitado encamina la señalización del abonado itinerante hasta el IMS nativo del abonado, desde donde se reencamina la sesión hacia su destino.
 - IMS contempla la interconexión con las redes de circuitos SS7 para servicios de llamadas de voz. Por tanto, existen elementos IMS para el interfuncionamiento entre las sesiones multimedia con los componentes de audio y las redes PSTN, GSM o el dominio de conmutación de circuitos de la propia red 3G. De esta forma, los abonados con la innovadora tecnología

IMS siempre podrán seguir comunicándose con otros abonados no IMS. Así mismo, el sistema está diseñado para proporcionar interoperación con redes de circuitos que admiten teleconferencias multimedia (por ejemplo, videollamadas).

1.6 Servicios IMS

IMS no define las aplicaciones o servicios que pueden ofertarse al usuario final, sino la infraestructura y capacidades del servicio que los operadores o proveedores de servicio pueden emplear para construir sus propias aplicaciones y producir su oferta de servicios. El operador IMS puede elegir orquestar los servicios de forma independiente, combinada o en multitud de variantes.

Como ejemplo se pueden poner los dos grupos siguientes de servicios finales:

1. Los denominados servicios heredados (las llamadas básicas de voz, la mensajería textual, la mensajería multimedia, el correo electrónico, etc.), en los cuales el usuario percibirá la misma calidad que cuando se prestan a través de los sistemas tradicionales.
2. Los servicios multimedia avanzados. En este tipo de servicios se incluyen la videoconferencia; la audioconferencia; la videoconferencia para personas sordas, video más texto de tiempo real; la multiconferencias en video, audio o texto; la difusión de medios de TV o radio; el vídeo bajo demanda; la mensajería instantánea y el chat multimedia; los videojuegos interactivos multiusuario; el servicio push-to-talk (walkie-talkie); etc.

En este sentido, IMS no impone límites, son la capacidad de la red de acceso y las características de los terminales las que fijan las restricciones.

1.7 Open IMS Core

FOKUS (Fraunhofer institute Für Offene Kommunikations Systeme) es la Comunidad del Instituto Tecnológico Fraunhofer de Berlin para la investigación y desarrollo de sistemas de comunicaciones móviles libres (open source) en redes fijas y redes wireless. FOKUS trabaja bajo distintos puntos de vista en todas las partes implicadas en un sistema IMS completo. En la actualidad existe una comunidad de desarrollo Open Source de la especificación 3GPP de IMS, bajo el nombre "Open IMS Core", para la implementación de los servidores CSCF y el HSS.

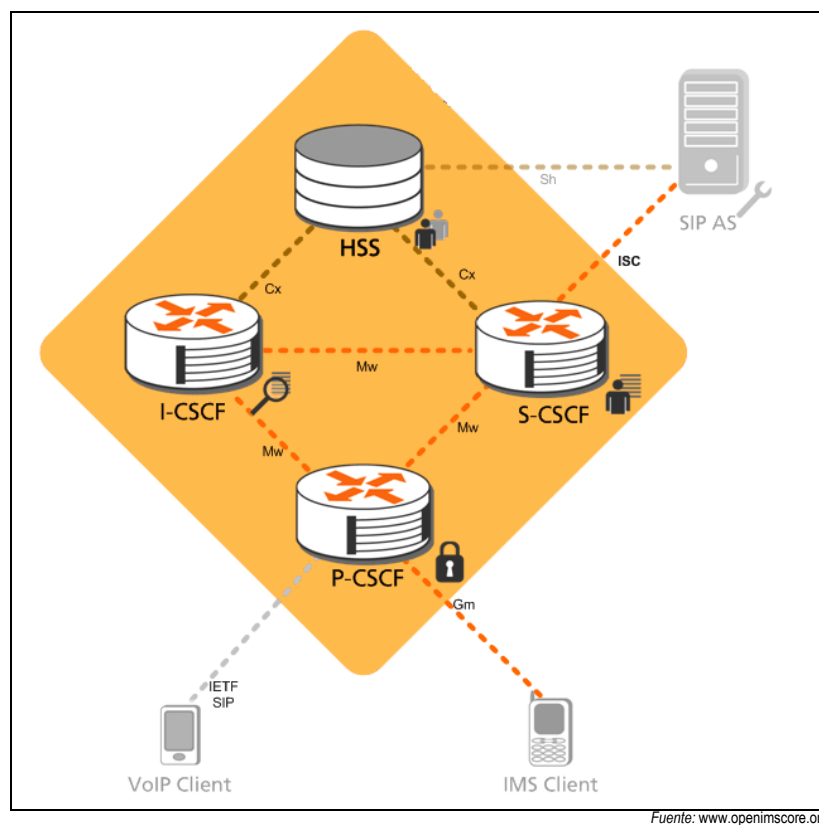


Figura 7. Componentes Open IMS Core

En la figura 7, se muestra los componentes de Open IMS Core. HSS, P-CSCF, I-CSCF y S-CSCF. Ahora se realizará una breve descripción de los componentes del Core desarrollado por Fokus.

Proxy CSCF (P-CSCF): se trata del punto de acceso externo al subsistema CSCF. Actúa como proxy entre el cliente y la red interna.

Interrogating CSCF (I-CSCF): es el dispositivo que interroga al HSS (Home Subscriber Server) mediante el protocolo DIAMETER, y localiza al servidor S-CSCF (local o remoto) correspondiente.

Serving CSCF (S-CSCF): es el servidor central de señalización SIP, se encarga de la gestión de cada sesión SIP, de acuerdo con la información residente en el servidor HSS

1.8 Cliente IMS

Se trata de una aplicación sobre un equipo de usuario que emite y recibe solicitudes SIP. Es necesario contar con clientes que puedan aprovechar los servicios ofrecidos por el prototipo Open IMS Core.

UCT IMS Client fue elaborado por la Universidad de Cape Town (UCT) y destaca respecto del resto de usuarios IMS por dos razones:

- Fue diseñado específicamente para trabajar con el núcleo IMS desarrollado por FOKUS.
- Su interfaz gráfica cuenta con varias opciones de configuración y acceso a distintos servicios.

El objetivo del proyecto UCT IMS Client es producir software libre en términos de costo y accesibilidad al código, es decir que pueda ser instalado, operado y modificado por investigadores que desean investigar acerca de IMS. Además el específico uso de Linux como plataforma está basado en la disponibilidad de librerías que tiene este sistema operativo.

Una lamentable consecuencia de la plataforma Linux es que el cliente no puede correr de forma nativa en los ambientes Windows, sin embargo esto no limita el propósito primario de esta aplicación para investigadores como herramienta de pruebas.

Un cliente IMS requiere varios protocolos IETF (Internet Engineering Task Force) para varias funciones, tales como Session Initiation Protocol (SIP), Real-time Transport Protocol (RTP), XML Configuration Access Protocol, Session Relay Protocol (MSRP) y Real Time Streaming Protocol (RTSP). El usuario UCT IMS Client usa varias librerías que implementan estos protocolos.

Características UCT IMS Client

El primer release del cliente UCT IMS Client soporto únicamente llamadas de voz utilizando los códecs μ -law y A-law. El último release, 1.0.13 del UCT IMS Client, además de soportar las funcionalidades básicas, ahora soporta sesiones de voz y video, presencia, mensajería instantánea, streaming de video y video bajo demanda VoD.

- **Sesiones de Voz y Video**

El cliente puede hacer llamadas de voz y video con otro cliente una vez que éstos se han registrado con su red IMS. El cliente extrae las rutas de servicio desde el proceso de registro y rutea las llamadas a la red que pertenece S-CSCF. El cliente soporta los códecs de audio μ -law y A-law -law, GSM y MP2, y el códec de video H263.

- **Presencia, Mensajería instantánea**

El cliente soporta el protocolo SIP para mensajería instantánea y Presence Leveraging Extensions (SIMPLE) donde los mecanismos de suscripción SIP son utilizados para presentar el estatus de presencia.

- **Streaming Video y Video on Demand**

El cliente soporta dos modos de visualización de video. El cliente puede simplemente realizar una solicitud de sesión con un servidor de video de una manera similar como lo sería con otro cliente IMS a través del envío de un mensaje SIP INVITE y luego de una negociación de codecs vía SIP/SDP. Alternativamente el cliente puede configurar la sesión utilizando el protocolo RTSP. La otra alternativa es la de la reproducción bajo demanda del video, en donde el usuario manipula la reproducción del video.

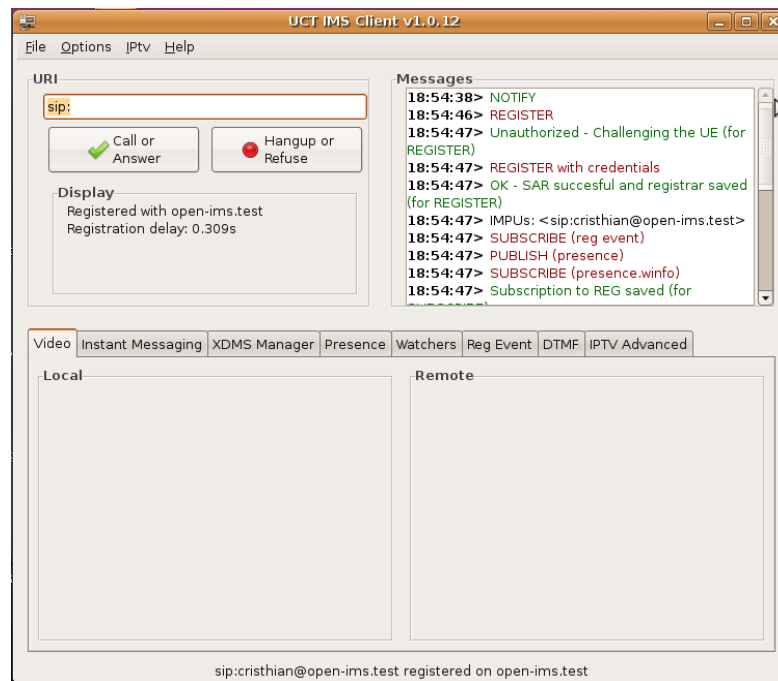


Figura 8. UCT IMS Client

Fuente: Autor

1.9 Streaming

1.9.1 Transmisión multimedia

Para realizar una transmisión multimedia a través de la red, es importante conocer los requerimientos mínimos. La calidad de la señal de video digital se basa principalmente en tres factores:

Frame Rate. El número de imágenes mostradas por segundo para dar al espectador percepción de movimiento. El estándar del NTSC para televisión marca un ratio de 30 frames por segundo (fps), compuesto de campos pares e impares. Para el cine, se utiliza un ratio de 24 fps.

Profundidad de color. El número de bits por píxel, que representan la información del color. Por ejemplo. 24-bits (color verdadero) pueden representar 16,7 millones de colores, 16-bits unos 65.535 y 8-bits unos 256 colores.

Resolución del Frame. Habitualmente, se expresa en ancho x alto de píxeles. Un ejemplo de resolución, de pantalla completa, de un monitor de un PC es de 640x480, un cuarto de la pantalla sería 320x240 y un octavo 160x102. Aunque, la resolución es bastante mayor para las televisiones de alta definición (1920x1080).

Haciendo cálculos podemos obtener el número de bits por segundos necesarios para transmitir una señal de video. Con un frame rate con movimiento completo (30 fps), con una resolución de pantalla de (640x480) y con color verdadero (24-bits):

Rate Date = (640x480 pixels) * (3 bytes/pixel) * (30 fps) / (1024000 bytes/ Megabyte) * (8 bits / byte) = 216 Megabits por Segundo.

El ancho de banda requerido para transmitir señales de video es muy elevado (216 Mbits/s) y pocas son las redes y dispositivos actuales que nos permiten la transmisión de esa tasa de datos.

Para reducir la cantidad de bits, que supone transmitir una señal de video a través de la red se utilizan tres métodos básicos: scaling, compresing y streaming

Dado que este proyecto se enfoca en streaming de video a continuación revisaremos en detalle este concepto. El streaming es una tecnología que se está implantado para la visualización de contenidos multimedia, entregando los datos de video digitalizados escalados y comprimidos.

1.9.2 Video Streaming

El Vídeo Streaming es un sistema de distribución, que se caracteriza por la visualización de contenidos mientras todavía se están descargando los mismos. La tecnología streaming permite al usuario visualizar y escuchar contenidos digitalizados como video, audio y animaciones eliminando la necesidad de tener totalmente descargado el archivo.

En este apartado se van a explicar los diferentes métodos de entrega de Video Streaming que existen, Unicast, Multicast y Broadcast.

1.9.3 Tipos de Streaming

El proceso de streaming se puede dividir en tres categorías, en función de cómo se obtiene la información a difundir: streaming en directo, diferido o bajo demanda.

- **Streaming en directo (live).** Es aquel que transmite eventos que están sucediendo justo en el momento de la difusión. Las imágenes y el sonido son digitalizados y retransmitidos en tiempo real. En este caso, los usuarios pueden seguir el desarrollo de un evento en el mismo momento que éste se está produciendo. La transmisión de radio y televisión por Internet tiene estas características. En este tipo de transmisión se emplea el término difusión (broadcast) porque realmente se está transmitiendo “en vivo” a todos los clientes la misma información, que no es más que el evento que se está produciendo en ese momento. Así, independientemente de cuando se conecta un cliente al servidor, todos ven el mismo punto del stream en un instante determinado (excepto las lógicas variaciones de los retardos en la red que hacen que unos clientes reciban antes los datos que otros). Para poder efectuar este tipo de transmisión no es suficiente con disponer de un servidor de streaming, sino que también es necesario un equipo que realice el proceso de captura y compresión en tiempo real.
- **Streaming diferido.** Las imágenes y el sonido proceden de un archivo digitalizado previamente y los usuarios pueden seguir el evento en diferido.
- **Streaming bajo demanda.** Conocido mediante las siglas VoD, en inglés Video on Demand, consiste en una aplicación que espera y sirve peticiones de uno a varios clientes. La petición contiene un comando donde pide qué video desea recibir el cliente. Una vez que el servidor ha recibido el comando de reproducción empieza a transmitir el video. Estos datos, al llegar a la aplicación cliente, se almacenan en un buffer de recepción para absorber posibles cortes de reproducción debido a la latencia y el jitter. En los casos que se vacía el buffer se producen cortes. Otro efecto son las distorsiones que se producen al perderse algunos fragmentos de datos que envía el servidor.

El proceso de streaming se puede dividir en tres categorías, en función de las características técnicas del servicio: Unicast, Multicast y Broadcast.

- **Unicast.** El tipo de Video Streaming Unicast entrega streams uno a uno para cada cliente y se utiliza para el video on Demand (VoD), ya que cualquier usuario puede hacer una petición de una secuencia de stream en cualquier momento.

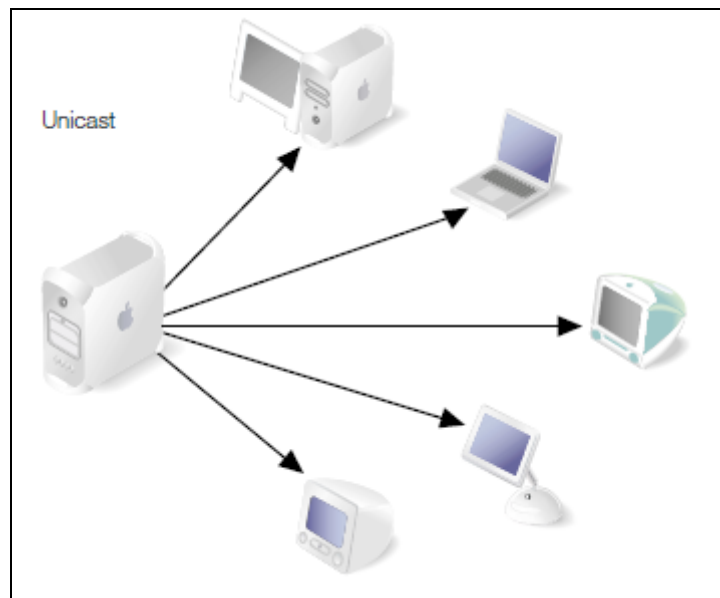


Figura 9. Unicast

Fuente: Administrator's Guide DSS

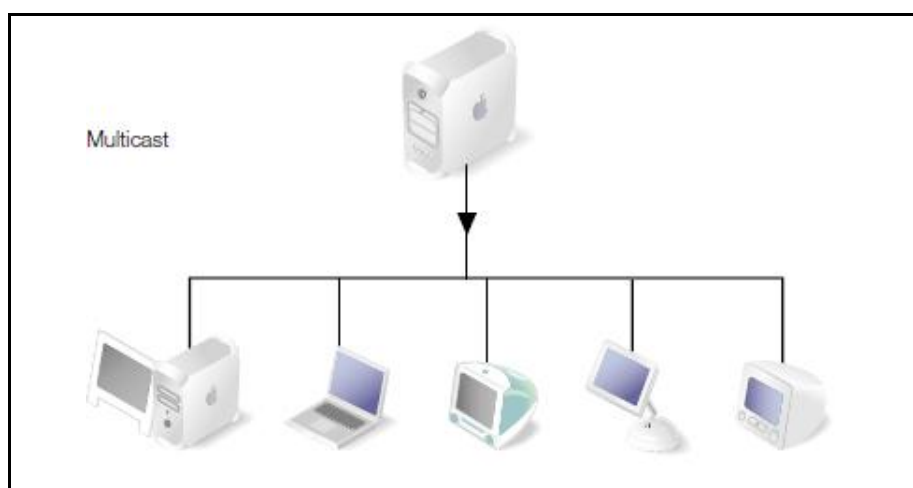
El servicio Unicast consiste en un servidor, que envía paquetes de datos a cada computador que solicita un stream. El método de entrega Unicast es una buena opción para recibir transmisiones en tiempo real, pero tiene sus desventajas. El servidor, debe enviar el flujo de datos individualmente a todo aquel, que quiera recibir la transmisión. Si se tiene a unas cuantas personas recibiendo el stream el funcionamiento es correcto, pero si se trata de difundir el material a miles de usuarios se deberá considerar dos inconvenientes con el proceso de unicast: demasiadas peticiones y demasiados paquetes

Con unicast, el servidor tiene que procesar cada solicitud de stream y atenderla. Cada stream, utiliza una pequeña porción de procesamiento del servidor. Si el servidor posee muchas solicitudes no podrá sostener la sobrecarga y muchas personas no podrán recibir la transmisión.

El segundo problema existente con unicast, y un gran número de solicitantes simultáneos de stream, es que una serie separada de paquetes de datos debe ser enviada a cada ordenador. Incluso, si el servidor pudiera hacer esta tarea, el número de paquetes de datos en tránsito por la red haría "flooding", es decir, inundaría el sistema entero haciendo que la transmisión se volviera muy lenta, o incluso se detuviese. Con una mayor cantidad de transmisión, los paquetes pueden desbordar la red.

- **Multicast**

El tipo de Video Streaming Multicast entrega los streams, simultáneamente, del servidor a muchos clientes.



Fuente: Administrator's Guide DSS

Figura 10. Multicast

Multicast ofrece la difusión a varios clientes de un mismo contenido, ya sea creado en ese momento en vivo (live broadcast), o almacenado previamente en el servidor. El usuario no posee el control, como si fuera un video doméstico, sin poder realizar las acciones propias del mismo (posicionamiento, paro, retroceso o avance rápido...), pero si puede parar y volverse a conectar a la emisión.

Multicast hace su trabajo de transmisión de manera similar a como funcionan los canales de televisión o las estaciones de radio: El programa (archivo de audio/video) se emite desde la estación hacia los transmisores (servidores streaming conectados a la red) quienes se encargan de distribuir la señal (el

stream) a los televidentes. Cuando el espectro de televidentes (usuarios, visitantes) se extiende, se agregan repetidores (servidores, lo que es conocido como splitting). Multicast es muy eficiente. Una simple serie de paquetes de datos puede ser usada para transmitir a múltiples clientes.

1.9.4 Transmisión de video Multicast

La información que habitualmente fluye a través de Internet lo hace en paquetes unicast. Este tipo de paquetes se adapta bien para las necesidades de las aplicaciones tradicionales que los usuarios de Internet utilizan: ftp, www,... pero plantea problemas cuando se desea audio y video.

Si para este tipo de servicios utilizásemos el mismo tipo de paquetes, tendríamos que emitir un paquete para cada uno de los participantes en la videoconferencia, multiplicando así el ancho de banda consumido por el número de participantes. Por ello, se definió otra forma de transferir información entre grupos de hosts sin necesidad de emitir una copia para cada uno: el Multicast IP.

Normalmente los hosts sólo aceptan de la red los paquetes cuya dirección de destino corresponde a una de sus direcciones IP. Si el host soporta Multicast y pertenece a algún grupo, aceptará también los paquetes dirigidos a ese grupo. Las aplicaciones distribuidas utilizan en gran medida aplicaciones Multicast, ya que ofrece mayores optimizaciones que el uso de transmisiones unicast.

▪ Direcciones Multicast

El conjunto de las direcciones IP está dividido en clases basado en los bits de mayor orden en una dirección IP de 32 bits.

Clase	Rango de direcciones	Bits de más peso
Clase A	1.0.0.0 – 127.255.255.255	0
Clase B	128.0.0.0 – 191.255.255.255	10
Clase C	192.0.0.0 – 223.255.255.255	110
Clase D	224.0.0.0 – 239.255.255.255	1110

Tabla 1. Clases direcciones IP

La única diferencia entre un paquete IP Unicast y uno Multicast, está en la dirección de destino. Las Clases A, B y C son utilizadas para Unicast. La Clase D para Multicast. Cada datagrama de IP cuya dirección destino empieza por «1110» es un datagrama IP de Multicast. Los 28 bits restantes identifican el grupo Multicast al que se envía el datagrama. Las direcciones IP Multicast se asignan a un grupo de equipos determinado y de manera temporal.

Hay algunos grupos especiales de Multicast asignados por el IANA. (Internet Assigned Numbers Authority).

IP Multicast	Detalle
224.0.0/24	Direcciones reservadas para grupos locales a una subred. Por ejemplo: 224.0.0.1: Todas las máquinas de una subred. 224.0.0.2: Todos los routers de una subred.
224.0.1/24 - 224.0.2/24	Reservadas para distintas organizaciones y protocolos. Por ejemplo: 224.0.1.1: NTP (Network Time Protocol).
224.0.3/24 - 238.255/16	Para cualquier grupo de ámbito mundial (los paquetes destinados a estos grupos pueden viajar por todo Internet).
239.255/16	Para grupos locales a una organización (los paquetes destinados a estos grupos no pueden salir de la organización).

Tabla 2. Direcciones Multicast

1.9.5 Beneficios de la tecnología Multicast

La tecnología Multicast ofrece ventajas significativas para el suceso de algunas aplicaciones avanzadas. Algunas de estas ventajas son:

- **Desempeño optimizado de la red.**

El uso inteligente de los recursos de la red evita replicación redundante de flujos de datos. Así, se puede optimizar el ancho de banda, a través de una mejor arquitectura para distribución de datos.

- **Soporte para aplicaciones distribuidas**

La tecnología Multicast está orientada hacia las aplicaciones distribuidas. Las aplicaciones multimedia como la videoconferencia se pueden utilizar en la red de forma dimensionable y eficiente.

- **Economía de Recursos.**

El costo de los recursos de la red se reduce a través de la economía en el ancho de banda en los enlaces y de la economía de procesamiento en servidores y equipos de la red. Nuevas aplicaciones y servicios se pueden implementar, sin requerir la renovación de recursos de la red.

- **Facilidad de crecimiento en Escala**

El uso eficiente de la red y la reducción de la carga en fuentes de tránsito permiten que los servicios y aplicaciones sean accesibles para un gran número de participantes. Por lo tanto, servicios que operan sobre Multicast se pueden dimensionar con facilidad, distribuyendo paquetes para pocos y para muchos receptores.

- **Mayor disponibilidad de la red.**

La economía de recursos de la red asociada a la reducción de carga en las aplicaciones y servidores torna la red menos susceptible a embotellamientos y por lo tanto, más disponible para uso. La transmisión de Multicast envía un solo paquete de Multicast dirigido a todos los recipientes. Esto proporciona la comunicación eficiente y la transmisión, optimiza el funcionamiento, y permite usos realmente distribuidos.

1.9.6 Operación básica del servicio Multicast

Para la realización de Multicast en una red es necesario un soporte por parte de los sistemas finales y los equipos intermedios de la red. En concreto los routers de la red han de ser capaces de ejecutar una serie de protocolos, uno principal para realizar el encaminamiento con replicación de flujos y varios protocolos auxiliares para mantener bases de datos de rutas, mecanismos de seguridad, interacción con los miembros de los grupos, control de admisión, descubrimiento de fuentes de datos y mantenimiento de redundancia en previsión de fallos. Para implementar el estado de encaminamiento Multicast es necesario que se ejecuten entidades del protocolo de encaminamiento Multicast que permitan crear dicho estado y así generar los llamados árboles de distribución Multicast.

La distribución de los flujos de datos se realiza mediante la asignación de unas direcciones IP detalladas en la tabla 2, a las cuales una fuente envía los datos y éstos, mediante las estructuras de distribución que se crean (árboles de distribución Multicast) son entregados a todos los receptores interesados (miembros suscritos al grupo Multicast).

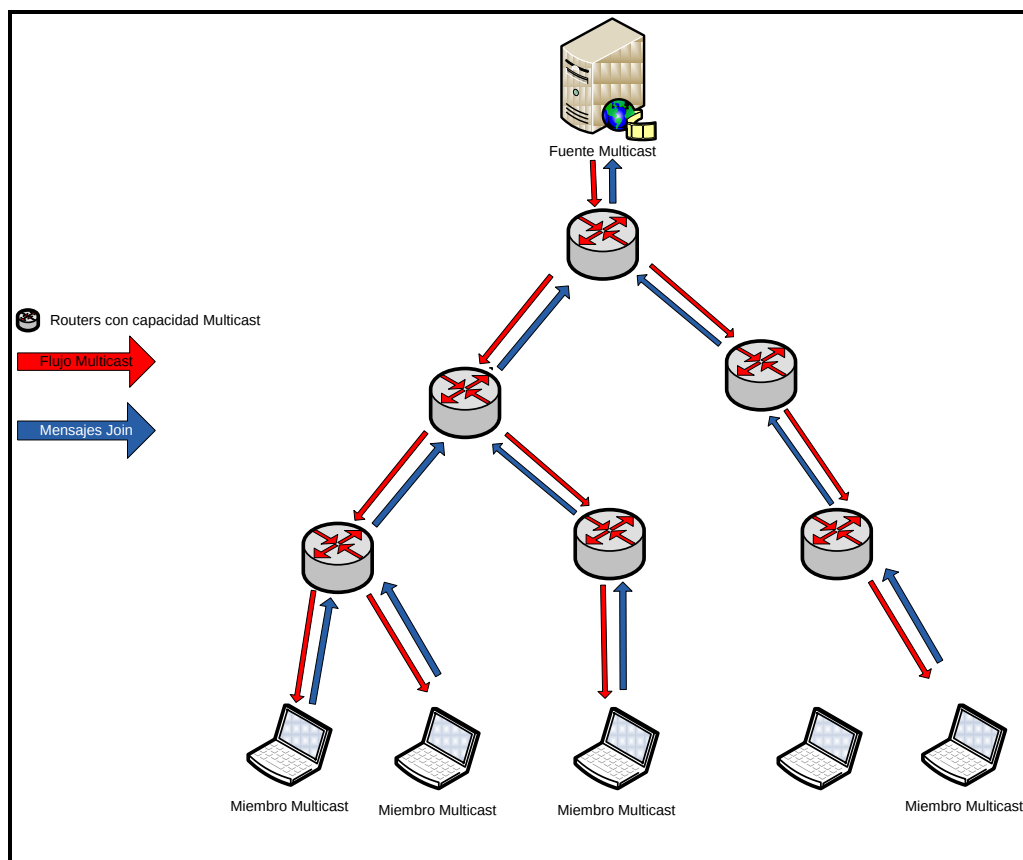


Figura 11. Proceso básico de distribución Multicast

Fuente: Administrator's Guide DSS

En la figura 11 se observa una red con cuatro niveles de jerarquía. En el nivel superior hay una fuente Multicast, que es la encargada de proveer los flujos de datos para su distribución a los hosts finales. Los hosts finales que se unen al árbol de distribución se denominan miembros Multicast. Además se observan 5 hosts finales de los cuales 4 son miembros Multicast. El proceso es el siguiente:

1. Cuando un potencial miembro Multicast desea unirse a un árbol de distribución envía un mensaje de unión, o mensaje Join, con dirección de destino la dirección IP del router designado de su subred. El miembro envía información adicional con la dirección IP Multicast del grupo al cual desea unirse (flechas azules inferiores). Los mensajes Join forman parte de los flujos de control.

2. Al llegar al router designado de la subred (routers inferiores) éste busca una correspondencia entre el grupo Multicast al que desea unirse el potencial miembro y la dirección IP unicast de alguna fuente que esté sirviendo al grupo. Si la encuentra, traduce el mensaje Join del miembro en un mensaje Join del protocolo de encaminamiento Multicast.
3. El mensaje Join del protocolo de encaminamiento Multicast es reenviado por los routers en el camino existente hasta llegar a la fuente. Durante el proceso de reenvío los mensajes de Join van creando un estado de encaminamiento Multicast en los routers que atraviesan. En cada router se crea una entrada con la dirección del grupo Multicast que va contenida en el mensaje Join y dicho router pasa a formar parte de una rama del árbol de distribución Multicast para el grupo en cuestión.
4. Cuando el mensaje Join llega a la fuente Multicast ésta comienza a enviar los datos al grupo. El envío se realiza con dirección destino la dirección IP Multicast del grupo.
5. Los paquetes IP del flujo Multicast que alcanzan cada router son inspeccionados por éste, que comprueba la dirección IP de destino. Si es una dirección IP Multicast, comprueba si alguna de sus interfaces de salida pertenece a una rama para el árbol. Si es así reenvía una copia del paquete por cada interfaz de salida que pertenezca a una rama (flechas rojas), creándose flujos de datos Multicast.
6. En caso de no haber coincidencia en las tablas de grupos Multicast del router, se descarta el paquete Multicast. En todo momento y por cada enlace de la red circula como máximo una copia del flujo de datos.

En el anexo A, se detalla los protocolos Protocolo de gestión de grupos internet (IGMP) y un resumen general de los protocolos que intervienen en el streaming Multicast.

1.9.7 Parámetros básicos en la transmisión multimedia

Las transmisiones multimedia, tanto de audio como de video, que se realizan, a través, de entornos complicados y heterogéneos (Internet), pueden producir, en ocasiones, calidades de servicio, que no son óptimas para la transmisión que se está realizando. Los factores fundamentales para poder obtener una buena calidad de servicio (QoS) en una transmisión se resumen en estos tres conceptos:

- Temporales: Latencia, jitter
- Productividad: ancho de banda
- Fiabilidad: tasa de pérdidas.

Parámetros temporales: latencia, jitter. La Latencia es el retraso entre la información enviada desde un terminal o servidor hasta la recepción de esa información en el terminal de destino. Mientras ese retardo no afecte a la calidad de la recepción y al procesamiento de las señales de audio y video no afectará a la percepción de una correcta calidad. En las redes IP, además, del retardo estándar (o físico) de la transmisión hay que añadir el retardo asociados a los algoritmos de compresión/descompresión (codecs), el del empaquetado IP de las muestras y el de los propios equipos de encaminamiento (routers, switch, etc.)

Se puede definir Jitter como la varianza de la latencia durante una transmisión. Las comunicaciones basadas en redes de conmutación de paquetes no garantizan el orden de la transmisión y cada uno de los paquetes puede ir por rutas diferentes entre el transmisor y el receptor por lo que estos pueden llegar en tiempos diferentes. Así pues, puede ser transmitido un paquete con el mínimo retardo (latencia) y posteriormente, llegar otro paquete con un retardo significativo (latencia). Cuando la información transmitida tiene que ser reconstruida la varianza de los retardos de algunos paquetes puede llegar a interrumpir la transmisión.

Las técnicas usadas para minimizar los efectos del jitter, pasan por enumerar el paquete con un número de secuencia, que permita saber el orden de los paquetes. Se marca el instante de generación, para poder saber el ritmo al que deben ser reproducidos los paquetes y atrasar, así, toda la reproducción un tiempo (que se sumará a la latencia existente), que permita absorber todas las

variaciones producidas por los retrasos. De esta manera, es menos probable que un paquete llegue fuera de tiempo. Este retardo en la reproducción (también llamado playout buffer) puede ser un tiempo fijo o variable, que debe ser escogido conforme al servicio que se ofrece o al retardo máximo que se soporta.

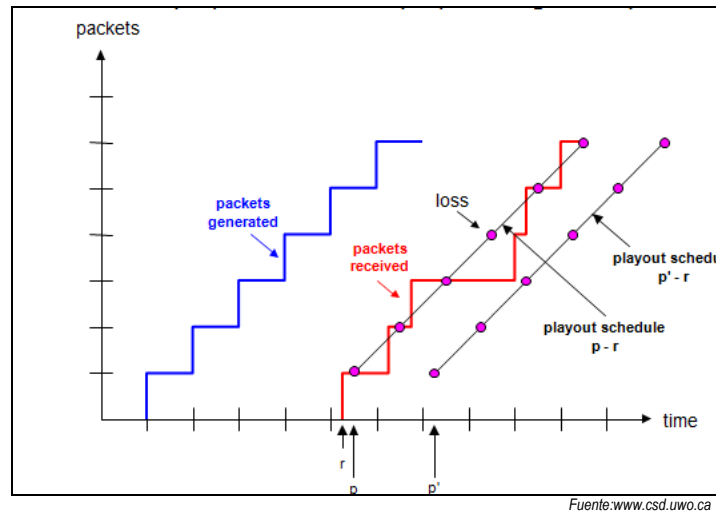


Figura 12. Play out, buffer en los paquetes recibidos

La solución técnica consiste en añadir un buffer entre la llegada de paquetes de la red y el dispositivo reproductor, que introduce un retraso para compensar los efectos del jitter como se puede observar en la figura 12. Se reconstruye la señal original a la tasa requerida por el reproductor. Se elige el retardo máximo, que debe ser un compromiso entre el retardo de la señal y las pérdidas de paquetes. Cuanto mayor es el playout buffer de retardo menos pérdidas de paquetes hay, pero produce un mayor retardo y cuanto menor es el buffer mayores son las pérdidas de paquetes por lo que hay que llegar a un compromiso. Una transmisión de streaming puede tolerar retardos de hasta 10 segundos, mientras que las aplicaciones interactivas en tiempo real no pueden tolerar retardos mayores de 400 mili segundos. El jitter es un parámetro dinámico, de hecho es una varianza, por lo que, lo más conveniente es calcular el playout buffer de una manera adaptativa, estimando el retardo para adaptar el buffer de salida.

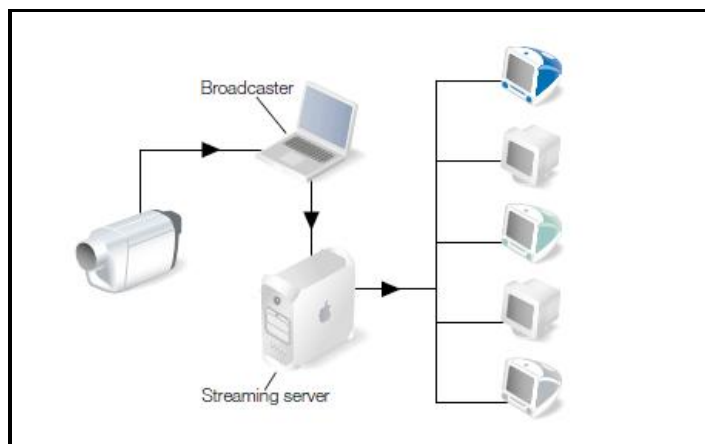
La productividad: ancho de banda El ancho de banda puede ser clasificado como una relación costo - calidad, donde si se invierte más dinero se consigue añadir más ancho de banda. El tamaño de los paquetes influye en el ancho de banda a utilizar. La capacidad de ancho de banda necesaria para una red debe ser mayor a la de la información que se quiere enviar, ya que a la información por paquete a enviar se le añade información adicional necesaria para el empaquetado de las muestras (cabecera Ip, cabecera UDP, cabecera RTP,

etc.). Estos datos adicionales permiten a los switch y a los router encaminar los paquetes hasta su destino. La utilización de paquetes de tamaño grande reduce la carga de información adicional y consecuentemente el ancho de banda adicional necesario y el retardo, pero por el contrario puede degradar la calidad de la información si se pierden paquetes (cuando un paquete se pierde implica la pérdida de mayor cantidad de datos). La elección de paquetes de menor tamaño aumenta el tamaño del ancho de banda adicional y el retardo, pero por el contrario la pérdida de un paquete produce una menor degradación de la señal.

Si se transmiten paquetes a mayor velocidad que el ancho de banda disponible pueden ocurrir congestiones en la red y por lo tanto, pérdidas de paquetes y disminución en la calidad del video. Si se transmite a menor velocidad que el ancho de banda disponible, la calidad del video será subóptima, ya que se podría transmitir a mayor calidad si se aumentase la velocidad. El ancho de banda disponible puede ser estimado por métodos basados en pruebas o por métodos basados en modelos (o basado en ecuaciones). El objetivo es tener una tasa de transferencia de bits (bit rate) lo más cercana al ancho de banda disponible. Se pueden utilizar otro tipo de herramientas, además, de las monetarias, para mitigar el impacto del “best-effort” en Internet. Se puede usar el protocolo UDP, en lugar, del TCP para evitar el sistema de detección de tasa de envío slow-start y optimizar el ancho de banda con una tasa de envío más eficiente. También, se puede adaptar los niveles de compresión de datos al ancho de banda disponible y mejorar los métodos de compresión y codificación mediante nuevos algoritmos.

La fiabilidad: tasa de pérdidas. El objetivo es encontrar una fiabilidad alta para las transmisiones multimedia con una tasa de pérdidas adecuada. La pérdida de paquetes es uno de los problemas que resulta de no utilizar un protocolo fiable. Las pérdidas de paquetes se producen básicamente porque los paquetes no llegan o porque llegan más tarde que el tiempo de retardo del buffer del reproductor. Para evitar tasas elevadas de pérdidas hay que introducir mecanismos de prevención y recuperación de esas pérdidas. Como prevención, podemos realizar el control de la tasa de transmisión y como recuperación de paquetes encontramos mecanismos como el FEC (Forward Error Correction) o el Interleaving.

1.9.8 Funcionamiento del servicio Streaming



Fuente: Administrator's Guide DSS

Figura 13. Streaming básico

En un proceso de streaming estándar de audio y vídeo sincronizado, las peticiones de servicio por parte de los clientes se pueden manejar utilizando el protocolo RTSP (Real- Time Streaming Protocol). Este protocolo se encarga de controlar el stream de contenido multimedia en dos direcciones, de forma que los clientes pueden pedir al servidor hacer cosas como rebobinar la película, saltar al siguiente capítulo, etc. Esto se puede conseguir con streaming ya que el medio no se descarga linealmente sino que se reproduce conforme se obtiene, y se permiten saltos en la reproducción, consiguiendo un acceso aleatorio al medio, incluso en saltos hacia delante. De otro lado, los datos del medio (el stream que contiene típicamente audio y vídeo sincronizados) se pueden transportar usando el protocolo estándar RTP (Real-Time Transport Protocol), que es un protocolo de transporte que permite la transmisión de información multimedia en tiempo real sobre cualquier tipo de red (aunque su uso más habitual es sobre redes usando el protocolo UDP).

Para poder transmitir por streaming un evento, necesitaremos los siguientes componentes:

Codificador. Este elemento consiste en una fuente de video que se convierte en el formato pertinente mediante una capturadora de video y el software codificador correspondiente.

Servidor de Streaming. Software encargado de la gestión de las peticiones de los usuarios para acceder el video.

Reproductor. Software encargado de la reproducción de los archivos suministrados por el servidor de streaming a través de Internet. El usuario deberá tener instalado en su ordenador el reproductor correspondiente para recibir los videos del servidor. Información de los usados por nosotros, más adelante. El inicio de la sesión de streaming y el control de la misma (comandos PLAY/STOP/FFWD/REW/SEEK) se realiza mediante el protocolo RTSP. El contenido se referencia a través de una URL tal como `rtsp://iptv.open-ims.test:8000/video.mpg`. Esta URL se obtendrá mediante la navegación por una aplicación web convencional (Application Server) que permitirá al usuario seleccionar el contenido que quiere visualizar, navegando por un catálogo de contenidos.

1.9.9 Calidad de Video

Un tema confuso encontrado en este proyecto es el tema de las comparaciones de calidad de vídeo. Se habla frecuentemente de alta "resolución" o imágenes de alta "resolución", y se tiende a asumir que la palabra "resolución" es sinónimo de calidad. Pero la resolución en términos de vídeo no equivale a calidad.

Resolución

La resolución es el número de píxeles que puede aparecer, no que aparecen. Por lo tanto, se debe pensar de la resolución como un trozo de papel milimetrado. Haciendo una analogía, los pixeles equivalen a un montón de cajas vacías. Esta es la verdadera definición de la resolución. La resolución CIF sería análoga a un pedazo de papel cuadriculado con 352 cajas de ancho por 240 cajas de alta (aunque un píxel no es perfectamente cuadrado). Una imagen 4CIF sería un trozo de papel milimetrado con 704 cajas de ancho por 480 cajas de alto.

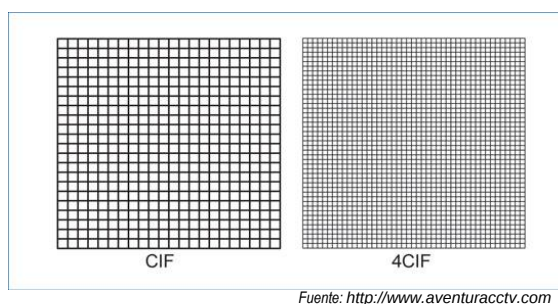


Figura 14. Resolución CIF y 4CIF

Dado que no existen "bits" de información en estos cuadros sin embargo, la resolución es el número de píxeles que se puede visualizar en la pantalla.

Tasa de bits

Un factor clave determinante, pero no la única consideración de la calidad, es la tasa de bits o bitrate. Los bits son la información que se insertará en los píxeles y la tasa de bits es la velocidad con la que los bits de información se colocan en los píxeles. Cuanta más información o bits aplicada al pixel, cuanto más densa es la imagen, mayor será la calidad.

La resolución cuantifica sólo los píxeles. Una imagen 4CIF puede potencialmente producir una imagen de mejor calidad que una imagen CIF, pero no necesariamente. Como 4CIF no menciona la cantidad o la calidad de los datos que se muestran en los píxeles, una imagen CIF con una alta tasa de bits puede producir una mejor calidad de imagen que una imagen 4CIF con una baja tasa de bits. Se necesita otras consideraciones para determinar la calidad de video.

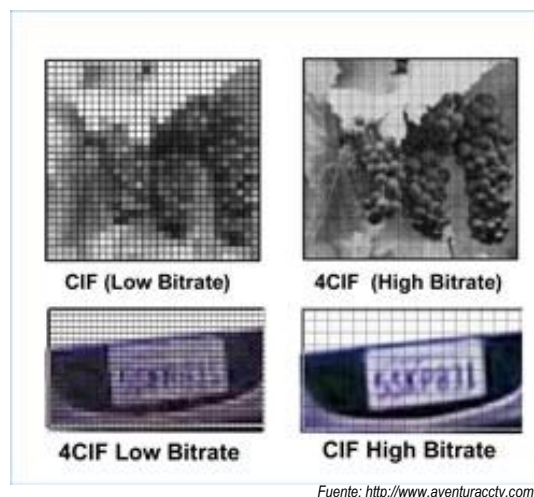


Figura 15. Comparación Bit Rate

Una imagen de mayor tasa de bits con una resolución más alta puede producir una imagen de mayor calidad que una imagen de menor tasa de bits con una resolución más baja.

Una vez que se ha determinado la tasa de bits, el siguiente factor principal a considerar es el códec que se aplica y la calidad de los datos resultantes.

Códecs

Una imagen de video no comprimida ocupa 1 MB aproximadamente. Para obtener un video fluido, se necesita una frecuencia de al menos 25 ó 30 imágenes por segundo, lo que genera un flujo de datos de 30 MB/s aproximadamente, es decir, más de 1,5 GB por minuto. Es obvio que este tipo de flujo es muy poco compatible con el espacio de almacenamiento de los ordenadores personales o incluso con las conexiones de red domésticas o de compañías pequeñas o medianas. Por lo tanto, para superar esta dificultad, se puede recurrir a algoritmos que permitan reducir de forma significativa el flujo de datos por medio de la compresión/descompresión de datos de video. A estos algoritmos se los denomina CóDec (por COmpresión/DEsCompresión).

En el vídeo digital para reducir la tasa de bits, se utiliza compresión con pérdida. La compresión con pérdida es un método que descarta algunas de las informaciones o datos que no es crítica en la escena, lo que resulta en una menor tasa de bits. Algunos códecs son más eficientes que otros en la compresión de datos. Los principios de codificación se remontan a más de medio siglo (algoritmo de Huffman), que sorprendentemente, se siguen utilizando hoy en día. El códec H.264, utiliza los algoritmos de cálculo más modernos. El resultado es imágenes de mayor calidad a velocidades de bits más baja en comparación con los códecs heredados (MPEG1, MPEG2, MPEG4 Part 2, H.261, H.262 y H.263) H.264AVC se conoce como códec complejo, puede rendir hasta 50% de compresión MPEG2 y hasta un 30% mejor que el MPEG4. Con una buena compresión se reducen las necesidades de ancho de banda y menor tiempo de descarga. Además, la calidad de vídeo H.264 resultante es hasta un 40% superior a MPEG2 y MPEG4.

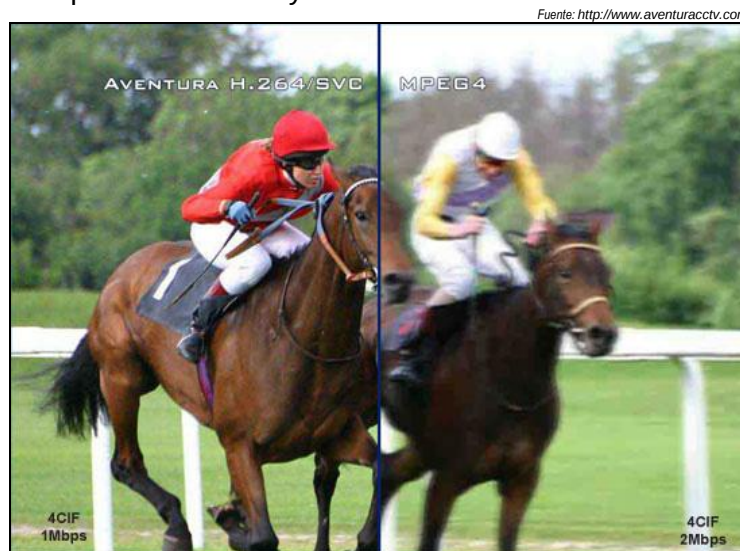


Figura 16. H.264/SVC [1Mbps] - MPEG-4 [2Mbps]

Por consiguiente, si dos archivos de vídeo utilizan la misma tasa de bits, uno podría ser mejor que el otro, como resultado de un códec más eficiente.

Un clip de vídeo H.264 grabado a 2 Mbps podría ser sustancialmente mejor que un clip de vídeo MPEG2 grabado a 2 Mbps, debido a la eficiencia del códec. De hecho, una imagen de 1 Mbps H.264, puede parecer mejor que un MPEG2 o MPEG4 grabada en 2Mbps.

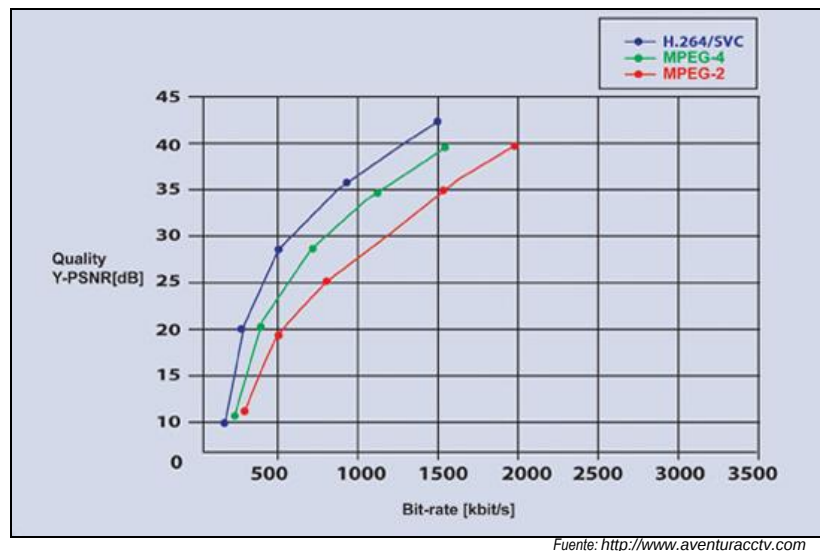


Figura 17. Comparación MPEG-2, MPEG-4, y H.264/SVC

En conclusión, la Resolución no es más que el número de píxeles que se llena con los datos. El Bitrate es el factor clave determinante en la calidad de vídeo. Una imagen de alta resolución (4CIF) no garantiza una mejor imagen que una con una resolución más baja (CIF). Algunos códecs pueden proporcionar imágenes de mayor calidad y menor tamaño de los archivos que otros. Para determinar la calidad del video es necesario la combinación de:

Alta resolución + Alto BitRate + Mejor Códec= Mejor calidad de video

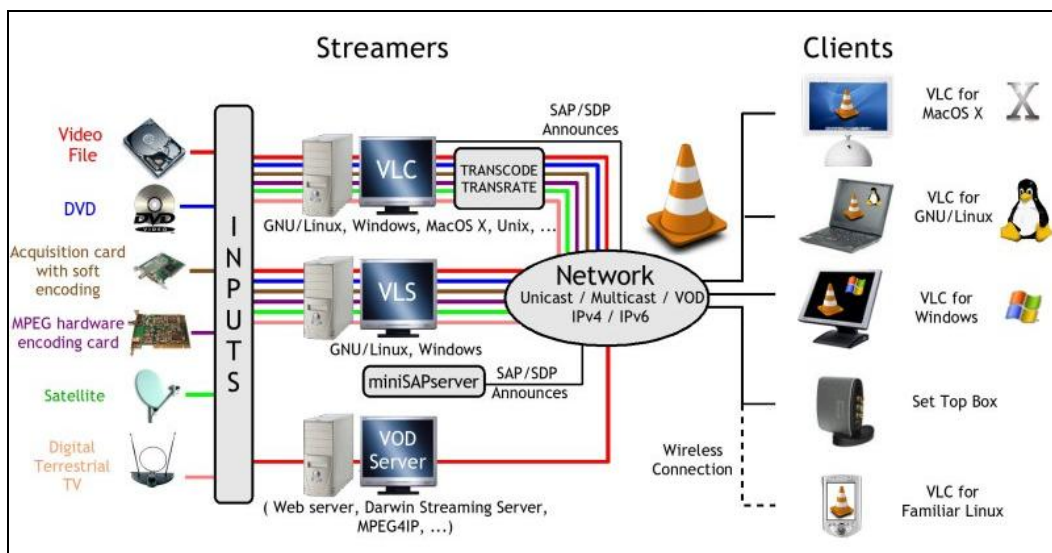
Un factor que no se ha tomado en cuenta en esta fórmula es la calidad del hardware utilizado, sin embargo para efectos de este proyecto de tesis únicamente se ha tomado en cuenta los parámetros: resolución, bitrate y códec.

1.10 Streaming de Video Open Source

Este apartado busca conocer alternativas libres para la emisión de información multimedia mediante la técnica de Streaming de video. La transmisión de este tipo señales a través de las redes se está popularizando con el ofrecimiento por parte de los proveedores de servicio de Internet de un mayor ancho de banda y la disminución de los costos. Sin embargo, aún es muy poco lo que se aprovecha de este tipo de servicios por el desconocimiento de tecnologías accesibles y de bajo costo que posibiliten su implementación, de ahí la importancia en explorar y probar aplicaciones libres.

1.10.1 Videolan

Este proyecto es una solución completa de software para el streaming de video, desarrollada bajo la licencia pública general GNU (GPL). Videolan está diseñado para generar flujos de videos bajo el estándar MPEG sobre redes de alto ancho de banda. Soporta el transporte de datos a través de los protocolos IPv4 e IPv6 además de poder generar tráfico Unicast o Multicast.



Fuente: <http://www.videolan.org/>

Figura 18. Arquitectura Streaming de Video

VLC permite que se opere a través de línea de comandos, o, por su interfaz gráfica. La utilización en línea de comandos permite mayor control sobre los parámetros con los cuales se invoca al servidor. Las herramientas de Videolan sobresalen sobre los otros sistemas de Streaming ya que existe una interfaz amigable, lo que permite una rápida adaptación de los usuarios. La característica diferenciadora radica en que la misma aplicación puede funcionar tanto como reproductor y como servidor.

Características principales:

- Soporta múltiples plataformas, entre estas: Windows, Mac OS X, Linux, BeOS, FreeBSD, OpenBSD,
- Soporta protocolos: RTSP, UDP, RTP, HTTP, MMS, MMSH, SAP, IGMPv3.
- Soporta Multicast y Unicast sin ningún tipo de limitación ni restricción.
- Formatos soportados: MPEG (ES, PS, TS, PVA, MP3), ID3, AVI, ASF, WMV, WMA; MP4, MOV, 3GP, OGG, OGM, WAV, FLV, FLAC, RAW DV, RAW audio (DTS, AAC, AC3/A52).
- Soporta video bajo demanda.
- Es posible recibir un flujo de video de un servidor, y enviarlo a otro cliente mientras se visualiza.
- Soporta Subtítulos

1.10.2 Darwin Streaming Server

Darwin Streaming Server es el servidor de código abierto de Apple. Comparte el mismo código base que la versión comercial Apple QuickTime Streaming Server. Este servidor comparte las mismas características que la versión comercial, pero en este caso sólo se distribuye el servidor en sí y no tiene las herramientas y extensiones adicionales que son exclusivas de la versión comercial.

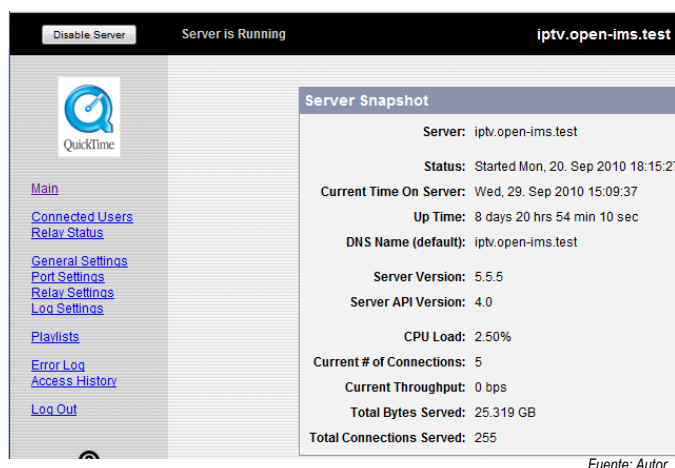


Figura 19. Darwin Streaming Server. DSS

Este servidor está recomendado para su uso por desarrolladores que necesitan enviar flujos de video codificados en formatos Quick Time y MPEG-4 en plataformas alternativas a Mac, como Windows, Linux, Solaris, etc. Este servidor está soportado por la comunidad de código abierto y no tiene soporte técnico de Apple.

Características principales:

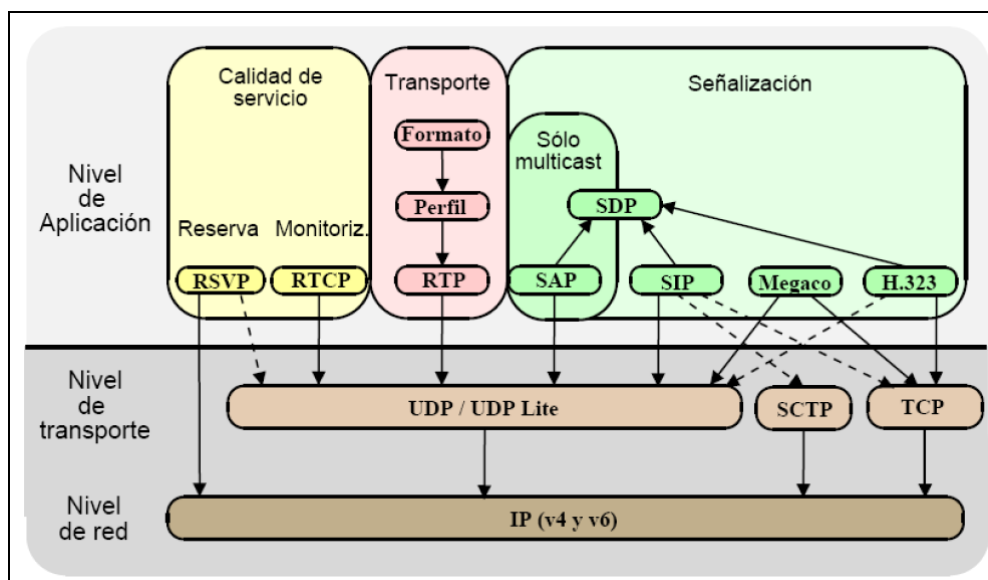
- Es compatible con la mayoría de los clientes disponibles sobre cualquier plataforma.
- Soporta los protocolos RTP/RTSP, tanto Multicast como unicast.
- Soporta video bajo demanda.
- Se puede administrar a través de su interfaz web.

1.11 Protocolos de Comunicación

El objetivo de este apartado es dar una visión general de los diferentes protocolos de comunicación implicados en la implementación de un servicio de streaming de vídeo en una plataforma IMS. Se presentan los protocolos de inicio de sesión (SIP), de establecimiento y gestión de sesión por parte del cliente (RTSP), los relativos al transporte y control (RTP/RTCP) y el de presentación (SDP) con las principales funciones disponibles en cada uno de ellos relativas a este proyecto.

1.11.1 Protocolo de Inicio de Sesión SIP.

El protocolo de inicio de sesión (SIP) fue desarrollado por la IETF (RFC 3261), por tanto su desarrollo está orientado a la integración con aplicaciones y servicios de internet. Tiene mayor flexibilidad para incorporar nuevas funciones y su implementación es simple.



Fuente: www.aventuracctv.com

Figura 20. Protocolo SIP

SIP es un protocolo de la capa de Aplicación del Stack de Protocolos TCP/IP. Como se observa en la figura 20, está relacionado estrechamente con el Protocolo SDP y coexiste junto con los protocolos del mismo nivel y funciones: Megaco y H323.

El protocolo SIP es un protocolo de señalización, presencia y mensajería instantánea desarrollado para establecer, modificar y finalizar sesiones multimedia. SIP permite el establecimiento de sesiones multimedia entre dos usuarios finales. SIP ofrece los siguientes mecanismos:

- Establecer, modificar y finalizar sesiones entre dos o más participantes
- Registro y localización de participantes.
- Descripción de características de las sesiones y negociación de capacidades de los participantes.

Algunas de sus características son:

- Basado en texto
- Sintaxis similar a HTTP o SMTP
- Uso de URI's
- Métodos básicos: INVITE, ACK, BYE, CANCEL, REGISTER, OPTIONS
- Los mensajes se agrupan en transacciones y llamadas.
- Generalmente, el cuerpo de los mensajes contiene descripciones de sesiones multimedia (SDP)
- Localización basada en DNS

SIP se ayuda de otros protocolos (SDP, RTP-RTCP) para lograr una llamada telefónica, una sesión de videoconferencia o de mensajería instantánea. RTP es usado para transportar los datos multimedia en tiempo real mientras que SDP se emplea para describir y codificar las características y capacidades en la sesión.

SIP es un protocolo de señalización orientado a conexiones end-to-end. Esto significa que toda la lógica se encuentra almacenada en los dispositivos finales (excepto el ruteo de mensajes SIP). La ventaja está en la escalabilidad que se obtiene pues los servidores no son saturados con mensajes SIP.

La forma de identificar a una entidad SIP es similar a la empleada para definir una cuenta de correo electrónico. A esta forma se le denomina URI (Uniform Resource Identifier). EL URI de SIP tiene una estructura de la forma: sip:usuario@dominio, por ejemplo sip:cristhian@open.ims.test. En el anexo B, se detalla los mensajes sip.

1.11.2 Real-Time Streaming Protocol. RTSP

El protocolo de flujo de datos en tiempo real, establece y controla uno o varios flujos sincronizados de datos ya sea de audio o de vídeo. RTSP es un protocolo no orientado a la conexión, en lugar de esto, el servidor mantiene una sesión asociada a un identificador, en la mayoría de los casos RTSP utiliza TCP para datos de control del reproductor y UDP para los datos de audio y vídeo.

Gracias a este protocolo el usuario es capaz de iniciar, detener o pausar la reproducción de un vídeo. Además, si el reproductor se lo permite, podría dar saltos temporales en el streaming. RTSP puede utilizarse en combinación con diferentes protocolos aunque lo más habitual, como ocurre en este proyecto, es que gestione conexiones RTP/RTCP. RTSP guarda ciertas similitudes con protocolos como HTTP al ser ambos protocolos destinados a solicitar información de un elemento de la comunicación a otro; a diferencia de HTTP, RTSP es un protocolo que necesita mantener el estado de las conexiones, además, la información se transporta en un protocolo diferente. RTSP se emplea en conjunto con SDP (Session Description Protocol), que es el encargado de proporcionar información sobre la sesión: número de flujos, tipo de contenido, duración, ancho de banda, etc.

1.11.3 Session Description Protocol. SDP

El protocolo SDP es utilizado para la negociación y descripción de sesiones multimedia. Mas que un protocolo es un lenguaje con una sintaxis claramente definida que permite la negociación de los parámetros de calidad de servicio para el establecimiento de una sesión multimedia así como el tipo de flujo que se

va a transmitir (audio, video, video-conferencia,...). La carga SDP se encapsula dentro del cuerpo del mensaje SIP. Esta carga está compuesta por una serie de líneas llamadas campos cuyos nombres dentro del mensaje están abreviados por una letra minúscula indicando el tipo de campo que representan y su valor. El contenido de la información dentro del cuerpo SDP se puede estructurar de la siguiente manera:

- **Descripción de la sesión:** contiene información referente a la sesión y al creador de la misma (información de la misma, dirección IP, ...)
- **Descripción de tiempos:** contiene los tiempos de inicio y finalización y las repeticiones.
- **Descripción multimedia:** contiene información referente al protocolo de transporte o los formatos multimedia soportados.

La estructura de cualquier campo SDP es la siguiente:

x=parámetro1 parámetro2... parámetroN

A continuación se listan todos los campos existentes en SDP:

Campo	Nombre
V	Version: versión
o	Origin: creador/identificador de la sesión
s	Session Name: nombre de la sesión
i	Information: información de la sesión
u	URI: identificador uniforme del recurso
e	Email: dirección de correo electrónico
p	Phone: número de teléfono
c	Connection: información de conexión
b	Bandwidth: información de ancho de banda
t	Time: información de comienzo y fin de sesión
r	Repeat times: número de repeticiones
z	Time zones: ajustes de las zonas horarias
k	Encryption key: clave encriptada
a	Attributes: línea con atributos
m	Media: información multimedia

Tabla 3. Campos SDP

En el anexo C, se detalla los atributos utilizados.

1.11.4 Protocolo de transmisión en tiempo real RTP y RTCP

El protocolo de transmisión en tiempo real RTP (Real-time Transport Protocol IETF RFC 3550) es el encargado de transmitir la información del vídeo propiamente dicho. Como su nombre indica RTP es utilizado en la transmisión en tiempo real y ofrece funcionalidades como marcas temporales que permiten sincronizar la información en recepción o identificación del tipo de carga transportada. Utiliza como protocolo de transporte UDP y pese a ser un protocolo pensado para transmitir información en tiempo real, no provee ningún tipo de calidad de servicio ni garantía.

RTP transporta, además de los datos en sí, cierta información útil para realizar un cierto control de la comunicación, no obstante, carece de un mecanismo para informar del estado de la misma. Es por ello que trabaja junto a Real Time Control Protocol RTCP, el cual genera una serie de informes para controlar el estado de las transferencias gracias a que incluye información sobre los paquetes perdidos, el retardo, jitter, etc. Gracias a estos informes será posible conocer también el estado del buffer el cliente.

1.11.5 Procolo de Datagrama de Usuario UDP

User Datagram Protocol. Protocolo abierto, no orientado a la conexión (como el TCP) y por lo que no establece un diálogo previo entre las dos partes, ni tampoco mecanismos de detección de errores. Generalmente se emplea en lugar del TCP, cuando no se necesitan fuertes controles en la comunicación de los datos. Brinda mayor velocidad y menos complejidad.

1.12 Trabajos relacionados

Alton Kenneth MacDonald del Instituto Tecnológico Autónomo de México (ITAM) presenta un trabajo de investigación para la configuración de un Test bed IMS. Este documento sirve como guía para la instalación y configuración de una plataforma IP Multimedia Subsystem. También se analiza la operación de servicios Servicios Basados en Localización (Location Based Services: LBS) dentro de este contexto para mejorar la Calidad de Experiencia (Quality of Experience: QoE) presenciada por los usuarios.

Uno de los primeros problemas con el streaming de multimedia es la adaptabilidad a las capacidades del dispositivo, el codec de red y las condiciones de red. Camarillo en la RFC 4117 titulada “Transcoding Services Invocation in the Session Initia” describe varios escenarios para el transcoding. Se discute el incluir un servicio de transcoding para un usuario final. Para comunicaciones en tiempo real tales como una llamada de voz y video incluir un servicio adicional incrementaría significativamente los valores de retraso (delay) y reduciría la experiencia de usuario.

Afaq Hasan Khan, Mohammed Abdul Qadeer presentan un paper titulado User Centric Quality of Experience Testing for Video on Demand over IMS, en donde se exponen varios tests para entender el comportamiento del servicio streaming de video en diferentes dispositivos y condiciones de red.

Niklas Blum, Fabricio Carvalho, Thomas Magedanz del instituto Fraunhofer Institute for Open Communication Systems (FOKUS), presentan un paper que valida la aplicabilidad de la infraestructura IMS. Además este paper presenta las actividades de investigación del instituto FOKUS hacia la exploración del impacto de LTE/SAE en la estandarización de IMS. Subsistema Multimedia IP provisiona sobre las arquitecturas LTE/SAE servicios equivalentes que no son soportados por LTE, por ejemplo conmutación de circuitos.

Mario Ramos de la Universidad Carlos III de Madrid presenta en marzo del 2009 su proyecto titulado “Desarrollo de una aplicación de audiconferencia basada en Multicast para escenarios de red con soporte IMS”. El objetivo fue la implementación de una aplicación que sea capaz de generar la señalización basada en protocolos SIP y SDP para permitir el establecimiento de una sesión de voz multiusuario. Además se realizan pruebas de emisión y recepción de paquetes de voz enviados por multifusión a los participantes de la conversación.

David Waiting, Richard Good, Richard Spiers y Neco Ventura, de la Universidad de Cape Town, Rondebosch, South África, presentan un paper con las características funcionales del cliente UTC IMS Client. El paper describe la historia del desarrollo del proyecto, la arquitectura del software, los servicios soportados y los resultados de varias pruebas de interoperabilidad entre el cliente UCT IMS Client y otros clientes IMS.

Luís López, Micael Gallego, José Recio y Francisco López de la Universidad Rey Juan Carlos presentan un paper relacionado con un servidor de aplicaciones IMS para la entrega del servicio streaming de video (Palpatine) para dispositivos 3G. Palpatine se integra dentro de una infraestructura IMS sin necesidad de modificar la red y completamente compatible con todos los estándares requeridos. Por esta razón, Palpatine puede ser directamente utilizado sobre cualquier red IMS sin la necesidad de requerir la instalación de una aplicación externa para proveer el servicio de streaming sobre dispositivos móviles.

Juan Miguel Colores del centro de Investigación y Desarrollo de Tecnología Digital. Presenta un estudio comparativo de sistemas de difusión de video afluente. Estudia cada elemento que integra la arquitectura de difusión de video afluente y se evalúa el desempeño de la arquitectura.

2 CAPÍTULO II. IMPLEMENTACIÓN DEL CORE DEL PROTOTIPO

2.1 Introducción

Este capítulo presenta la implementación del prototipo de IMS con el fin de proporcionar los bloques de servicio de telecomunicaciones básicos sobre los cuales se apoya el servicio de streaming de video. La construcción de una maqueta IMS brinda un ambiente de desarrollo para futuros proyectos.

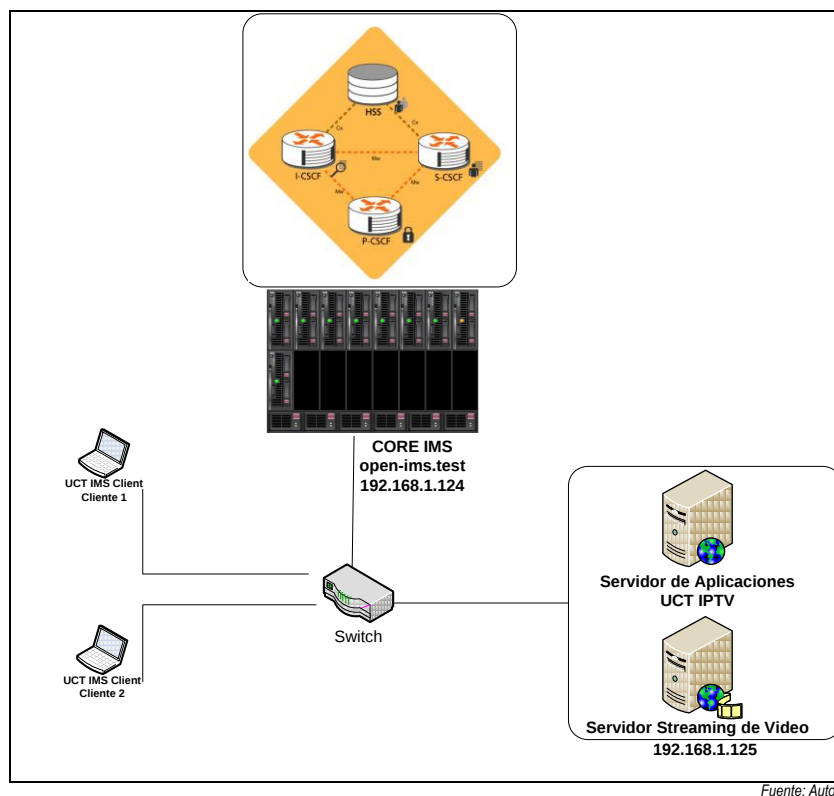


Figura 21. Diagrama de Red del Prototipo

2.2 Instalación

Como entorno para la implementación del Core se ocupó una red donde todos los hosts están conectados a través de un Switch. El diagrama de esta red se presenta en la figura 21. Se utilizó un servidor para el core de la plataforma, otro equipo como servidor de aplicaciones y como servidor para el streaming de video, y dos PC's clientes.

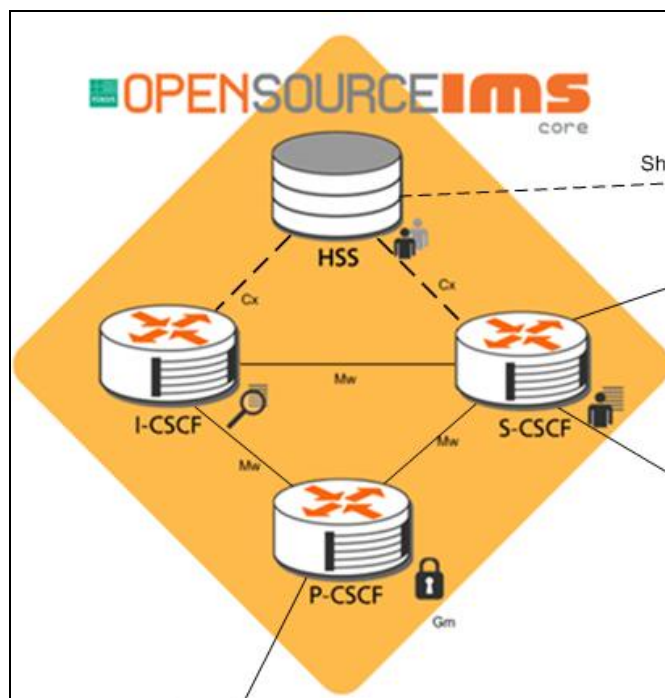


Figura 22. Diagrama Lógico prototipo IMS

Open IMS Core elaborado por FOKUS esta formado por las entidades P-CSCF, I-CSCF y S-CSCF. Estas entidades forman el corazón del prototipo, ya que son quienes controlan las sesiones y flujos entre los usuarios. La instalación de las entidades requiere bajar el código fuente y compilarlo. La instalación y configuración fueron probadas exitosamente sobre el sistema operativo Linux, distribución Ubuntu, la versión 10.04 LTS.

2.2.1 Requerimientos Hardware y Software

Hardware		Software
	HP Blade	Darwin Streaming Server
Servidor Virtual VmWare	Intel® Xeon a 2.8 GHz. Memoria Ram:4 GB Disco Duro: 80 Gb	GCC3/4, make, JDK1.5, ant MySQL instalado y corriendo Bison, Flex, libxml2 (> 2.6), libmysql - both with development
Acceso de Red	192.168.1.124	Bind instalado y corriendo.
Sistema Operativo	Linux	Ubuntu Server 10.04
Hostname	iptv.open-ims.test	Browser

Tabla 4. Requerimientos hardware y software para IMS

2.2.2 Instalación Open IMS Core

Para la implementación del Core de IMS, se utilizó una máquina virtual en un ambiente Vmware, con las características que se detallan:

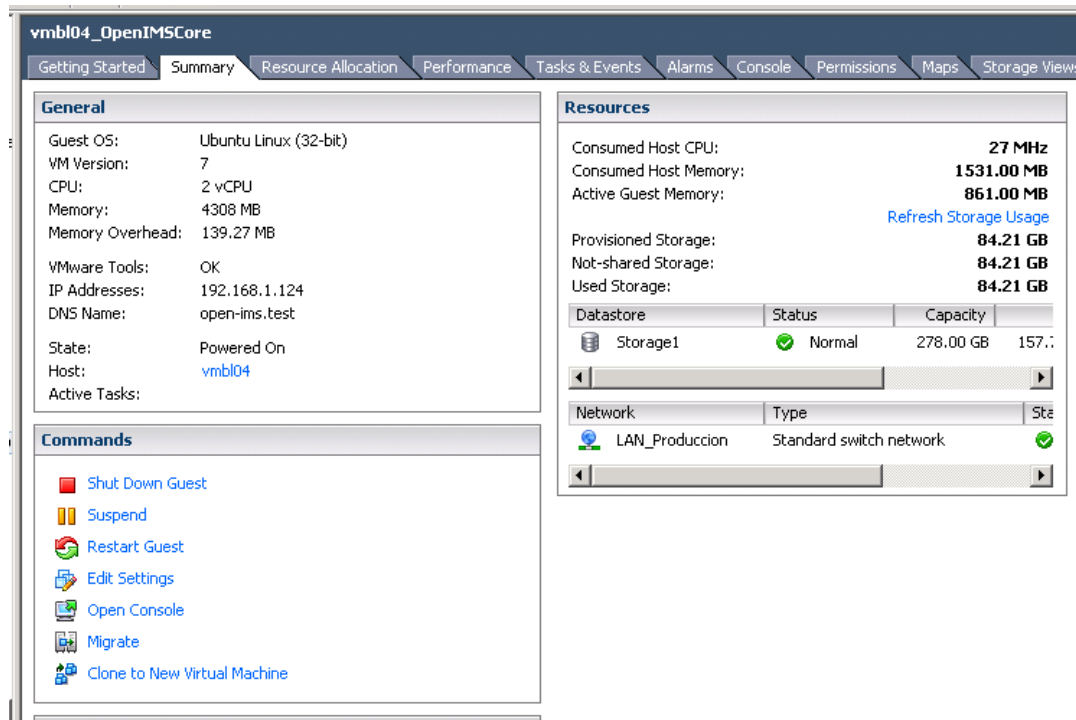


Figura 23. Máquina Virtual para Open IMS Core

Fuente: Autor

Prerequisitos

Habilitar el repositorio partner en /etc/apt/source.list

```
root@open-ims:/# cd /etc/apt/  
root@open-ims:/etc/apt# vi source.list
```

Con éste comando actualizaremos los repositorios de nuestro sistema.

```
root@open-ims:/etc/apt# apt-get update
```

Instalar el software utilizando sudo apt-get install:

```
root@open-ims:/etc/apt# apt-get install subversion ant sun-java6-jdk bison flex mysql-  
server libmysqlclient15-dev libxml2 libxml2-dev libxml2-dev bind9
```

Ahora es necesario configurar JAVA HOME. Después de la instalación Java (JRE & JDK), modifica el archivo profile, ubicado en /etc/profile. Es necesario verificar que exista el path usr/lib/jvm/java-6-sun-1.6.0.20

```
export JAVA_HOME=/usr/lib/jvm/java-6-sun-1.6.0.20  
export CLASSPATH=.:$JAVA_HOME/lib:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar  
export PATH=$JAVA_HOME/bin:$PATH
```

Una vez editado el archivo profile, se debe reiniciar el servidor.

Obtener el Código Open Ims Core

Iniciar un terminal linux. Crear el directorio /opt/OpenIMSCore

```
root@open-ims:/opt# mkdir /opt/OpenIMSCore  
root@open-ims:/opt# cd /opt/OpenIMSCore
```

Crear el directorio ser_ims

```
root@open-ims:/opt/OpenIMSCore# mkdir ser_ims  
root@open-ims:/opt/OpenIMSCore # svn checkout  
http://svn.berlios.de/svnroot/repos/openimscore/ser_ims/trunk ser_ims
```

Crear el directorio FHoSS

```
root@open-ims:/opt/OpenIMSCore # mkdir FHoSS  
root@open-ims:/opt/OpenIMSCore # svn checkout  
http://svn.berlios.de/svnroot/repos/openimscore/FHoSS/trunk FHoSS
```

Compilación de código.

```
root@open-ims:/opt/OpenIMSCore # cd ser_ims  
root@open-ims:/opt/OpenIMSCore/ser_ims# make install-libs all  
root@open-ims:/opt/OpenIMSCore/ser_ims# cd ..
```

Si se produce algún error, probablemente no se instalaron los prerequisites indicados.

```
root@open-ims:/opt/OpenIMSCore # java -version
java version "1.6.0_20"
Java(TM) SE Runtime Environment (build 1.6.0_20-b02)
Java HotSpot(TM) Server VM (build 16.3-b01, mixed mode)
```

Se debe verificar que la version de JDK sea superior a 1.5 antes de continuar.

```
root@open-ims:/opt/OpenIMSCore # cd FHoSS
root@open-ims:/opt/OpenIMSCore/FHoSS# ant compile
root@open-ims:/opt/OpenIMSCore/FHoSS# ant deploy
root@open-ims:/opt/OpenIMSCore/FHoSS# cd ..
```

Configurar el servicio DNS.

Dentro de la carpeta `ser_ims/cfg/` esta el archivo `open-ims.dnszone`. Se debe copiar este archivo en el directorio de configuración bind `/etc/bind/`

```
root@open-ims:/opt/OpenIMSCore # cd ser_ims/cfg
root@open-ims:/opt/OpenIMSCore/ser_ims/cfg# cp open-ims.dnszone /etc/bind/.
```

Ahora es necesario editar el archivo `named.conf` ubicado en el directorio `/etc/bind`. Se debe tener cuidado cuando se modifique el archivo `named.conf` para evitar errores sintácticos puesto que hasta el error más pequeño puede impedir que el servicio `named` arranque.

```
root@open-ims:/# cd /etc
root@open-ims:/etc# cd bind
root@open-ims:/etc/bind# vim named.conf
```

Insertamos las siguientes líneas

```
zone "open-ims.test" {
    type master;
    file "/etc/bind/open-ims.dnszone";
};
```

Una declaración zone define las características de una zona tal como la ubicación de su archivo de configuración y opciones específicas de la zona. En la declaración, la zona es identificada como *open-ims.test*, el tipo es configurado como master y el servicio named se instruye para leer el archivo */etc/bind/open-ims.dnszone*. Terminamos la configuración del servicio dns reiniciando el servicio.

```
root@open-ims:/etc/bind# /etc/init.d/bind9 restart
```

Se verifica que el archivo */etc/resolv.conf* apunte como servidor DNS a la ip 192.168.1.124, de esta manera aseguramos que el mismo servidor resuelva el dominio open-ims.test. Verificamos que la resolución se realice en el servidor de la siguiente manera:

```
root@open-ims:/etc/bind# nslookup hss.open-ims.test
Server:      192.168.1.124
Address:     192.168.1.124#53

Name:  hss.open-ims.test
Address: 192.168.1.124
```

Configurar la base de datos.

Como servidor base de datos, se utiliza Mysql, para crear las tablas utilizamos los scripts ubicados en las carpetas */opt/OpenIMSCore/ser_ims/cfg/* y */opt/OpenIMSCore/FhoSS/scripts/*

```
root@open-ims:/opt/OpenIMSCore # mysql -u root -p -h localhost < ser_ims/cfg/icscf.sql
root@open-ims:/opt/OpenIMSCore # mysql -u root -p -h localhost <
FHoSS/scripts/hss_db.sql
root@open-ims:/opt/OpenIMSCore # mysql -u root -p -h localhost <
FHoSS/scripts/userdata.sql
```

Configurar el Core de Open IMS Core.

Hasta aquí los servicios MySQL y DNS deben estar corriendo. Ahora se debe copiar los archivos *pcscf.cfg*, *pcscf.sh*, *icscf.cfg*, *icscf.xml*, *icscf.sh*, *scscf.cfg*, *scscf.xml*, *scscf.sh* ubicados en la carpeta */opt/OpenIMSCore/ser_ims/cfg* en la carpeta */etc/OpenIMSCore/*

```
root@open-ims:/opt/OpenIMSCore # cd ser_ims/cfg
root@open-ims:/opt/OpenIMSCore/ser_ims/cfg# cp *.* /etc/OpenIMSCore/
```

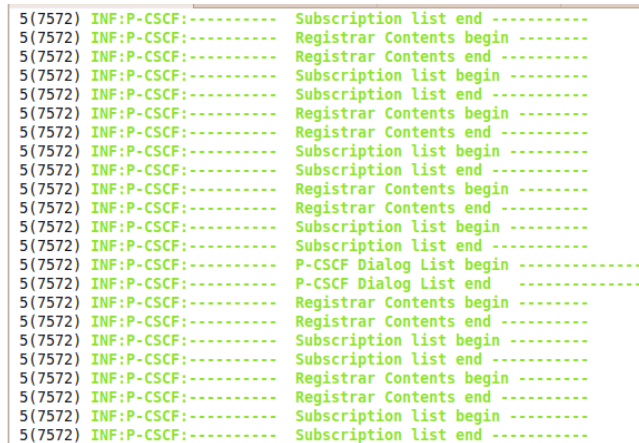
El script *configurator.sh* ubicado en el directorio */opt/OpenIMSCore*, permite la personalización de la dirección ip y del dominio. La ip utilizada en los archivos de configuración es la del loopbak 127.0.0.1 y el dominio open-ims.test. Para la implementación de este prototipo se personalizó únicamente la ip a 192.168.1.124. No hubo cambios con el dominio.

```
root@open-ims:/opt/OpenIMSCore# ./configurator.sh
Domain Name:open-ims.test
IP Adress:192.168.1.124
File to change ["all" for everything, "exit" to quit]:all
```

Inicializar los Componentes.

Ahora ejecutamos los scripts en terminales diferentes que levantan los servicios *pcscf.sh*, *icscf.sh*, *scscf.sh* y *fhoss.sh*.

```
root@open-ims:/opt/OpenIMSCore # ./pcscf.sh
```



```
5(7572) INF:P-CSCF:----- Subscription list end -----
5(7572) INF:P-CSCF:----- Registrar Contents begin -----
5(7572) INF:P-CSCF:----- Registrar Contents end -----
5(7572) INF:P-CSCF:----- Subscription list begin -----
5(7572) INF:P-CSCF:----- Subscription list end -----
5(7572) INF:P-CSCF:----- Registrar Contents begin -----
5(7572) INF:P-CSCF:----- Registrar Contents end -----
5(7572) INF:P-CSCF:----- Subscription list begin -----
5(7572) INF:P-CSCF:----- Subscription list end -----
5(7572) INF:P-CSCF:----- Registrar Contents begin -----
5(7572) INF:P-CSCF:----- Registrar Contents end -----
5(7572) INF:P-CSCF:----- Subscription list begin -----
5(7572) INF:P-CSCF:----- Subscription list end -----
5(7572) INF:P-CSCF:----- P-CSCF Dialog List begin -----
5(7572) INF:P-CSCF:----- P-CSCF Dialog List end -----
5(7572) INF:P-CSCF:----- Registrar Contents begin -----
5(7572) INF:P-CSCF:----- Registrar Contents end -----
5(7572) INF:P-CSCF:----- Subscription list begin -----
5(7572) INF:P-CSCF:----- Subscription list end -----
5(7572) INF:P-CSCF:----- Registrar Contents begin -----
5(7572) INF:P-CSCF:----- Registrar Contents end -----
5(7572) INF:P-CSCF:----- Subscription list begin -----
5(7572) INF:P-CSCF:----- Subscription list end -----
```

Fuente: Autor

Figura 24. Servicio *pcscf.sh*

```
root@open-ims:/opt/OpenIMSCore # ./icscf.sh
```

```

14(7599) [16777216,10415]
14(7599) [16777216,4491]
14(7599) [16777216,13019]
14(7599) [16777217,10415]
14(7599) [16777221,10415]
14(7599) -----
14(7599) DBG:peer_timer(): Peer hss.open-ims.test State 5
14(7599) --- Peer List: ---
14(7599) S[R_Open] hss.open-ims.test:3868 D[ ]
14(7599) [16777216,10415]
14(7599) [16777216,4491]
14(7599) [16777216,13019]
14(7599) [16777217,10415]
14(7599) [16777221,10415]
14(7599) -----
14(7599) --- Peer List: ---
14(7599) S[R_Open] hss.open-ims.test:3868 D[ ]
14(7599) [16777216,10415]
14(7599) [16777216,4491]
14(7599) [16777216,13019]
14(7599) [16777217,10415]
14(7599) [16777221,10415]
14(7599) -----
14(7599)

```

Figura 25. Servicio icscf.sh

root@open-ims:/opt/OpenIMScore # ./scscf.sh

Fuente: Autor

```

14(7625) [16777217,10415]
14(7625) [16777221,10415]
14(7625) -----
5(7616) INF:S-CSCF:----- Registrar Contents begin -----
5(7616) INF:S-CSCF:----- Registrar Contents end -----
14(7625) --- Peer List: ---
14(7625) S[R_Open] hss.open-ims.test:3868 D[ ]
14(7625) [16777216,10415]
14(7625) [16777216,4491]
14(7625) [16777216,13019]
14(7625) [16777217,10415]
14(7625) [16777221,10415]
14(7625) -----
14(7625) --- Peer List: ---
14(7625) S[R_Open] hss.open-ims.test:3868 D[ ]
14(7625) [16777216,10415]
14(7625) [16777216,4491]
14(7625) [16777216,13019]
14(7625) [16777217,10415]
14(7625) [16777221,10415]
14(7625) -----
5(7616) INF:S-CSCF:----- Registrar Contents begin -----
5(7616) INF:S-CSCF:----- Registrar Contents end -----

```

Figura 26. Servicio scscf.sh

root@open-ims:/opt/OpenIMScore # ./fhoss.sh

Fuente: Autor

```

dt.jar:/usr/lib/jvm/java-6-sun-1.6.0.15/lib/tools.jar:log4j.properties:..
2010-09-04 15:39:16,760 INFO de.fhg.fokus.hss.main.TomcatServer - startTomcat T
omcat-Server is started.
2010-09-04 15:39:17,598 WARN org.apache.catalina.connector.MapperListener - reg
isterEngine Unknown default host: 127.0.0.1
2010-09-04 15:39:19,266 INFO de.fhg.fokus.hss.web.servlet.ResponseFilter - init
Response Filter Initialisation!
2010-09-04 15:39:19,715 INFO de.fhg.fokus.hss.main.TomcatServer - startTomcat W
ebConsole of FHOSS was started !
2010-09-04 15:39:22,848 WARN org.hibernate.impl.SessionFactoryObjectFactory - a
ddInstance InitialContext did not implement EventContext
2010-09-04 15:39:22,884 INFO de.fhg.fokus.diameter.DiameterPeer.DiameterPeer -
<init> Bean style constructor called, don't forget to configure!
2010-09-04 15:39:22,887 INFO de.fhg.fokus.diameter.DiameterPeer.DiameterPeer -
configure FQDN: hss.open-ims.test
2010-09-04 15:39:22,887 INFO de.fhg.fokus.diameter.DiameterPeer.DiameterPeer -
configure Realm: open-ims.test
2010-09-04 15:39:22,888 INFO de.fhg.fokus.diameter.DiameterPeer.DiameterPeer -
configure Vendor ID : 10415
2010-09-04 15:39:22,888 INFO de.fhg.fokus.diameter.DiameterPeer.DiameterPeer -
configure Product Name: JavaDiameterPeer
2010-09-04 15:39:22,888 INFO de.fhg.fokus.diameter.DiameterPeer.DiameterPeer -
configure AcceptUnknownPeers: true
2010-09-04 15:39:22,888 INFO de.fhg.fokus.diameter.DiameterPeer.DiameterPeer -
configure DropUnknownOnDisconnect: true
2010-09-04 15:39:22,911 INFO de.fhg.fokus.hss.main.HSSContainer - waitForExit
Type "exit" to stop FHOSS!

```

Figura 27. Servicio fhoss.sh

Fuente: Autor

El correcto funcionamiento del Core Open-IMS se ve reflejado en la información que despliegan estos procesos al momento de comunicarse entre sí. Los puertos que utilizan las funciones del dominio IMS son:

Función	Número de Puerto Ocupado
P-CSCF	4060
I-CSCF	5060
S-CSCF	6060
HSS (Diameter)	3868, 3869, 3870, 8080

Tabla 5. Puertos utilizados Open IMS Core

Ahora se puede visualizar la interface web para administrar el Core del prototipo en <http://localhost:8080/>

- Username: hssAdmin
- password: hss

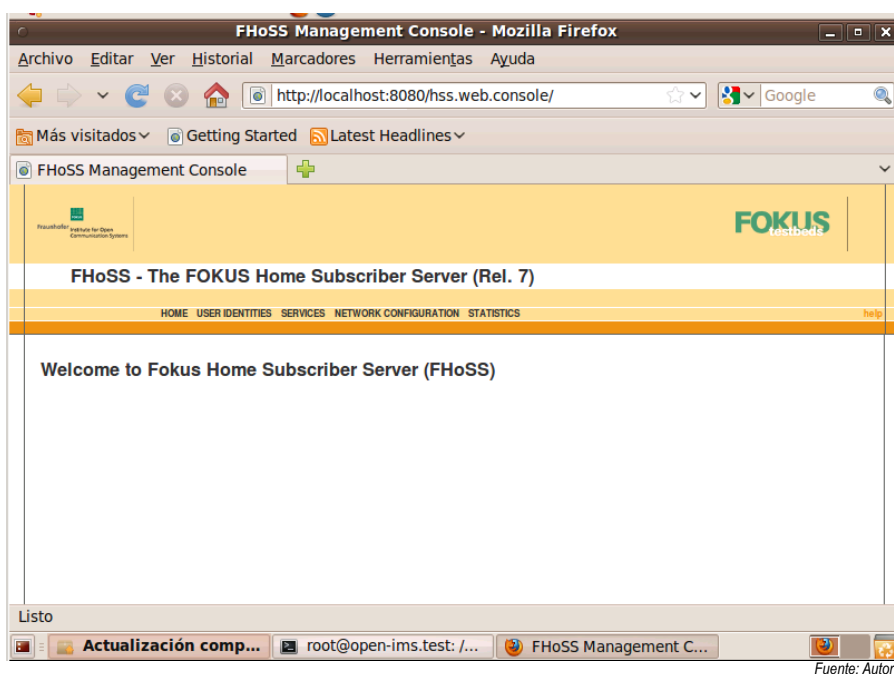


Figura 28. Consola de Administración Web. FHoSS

Es importante mencionar que las entidades IMS ya cuentan con usuarios de prueba llamados sip:alice@open-ims.test y sip:bob@open-ims.test con las contraseñas alice y bob respectivamente.

3 CAPÍTULO III. IMPLEMENTACIÓN DEL SERVICIO DE STREAMING DE VIDEO

3.1 Introducción.

Para implementar una solución de streaming se utilizaron dos herramientas basadas en proyectos de código abierto: Darwin Streaming Server y VideoLan VLC.

Para realizar el proceso de streaming, utilizamos la version open source del software proporcionado por Apple dentro de su suite QuickTime, debido al compromiso de esta compañía por desarrollar software bajo estándares y RFCs. Este servidor nos permite transmitir videos a diferentes tipos de clientes soportados por Internet, una red local, redes inalámbricas, etc., mediante el uso de los protocolos RTP y RTSP. Al ser de código abierto, es un servidor altamente configurable y adaptable a nuestras necesidades, ya que es posible modificar, manipular y adaptar a los requerimientos de este proyecto.

VideoLAN es una solución de software completa para transmisión de vídeo, desarrollada por estudiantes de École centrale Paris y desarrolladores de todo el mundo, dentro de GNU General Public License (GPL). VideoLAN está diseñado para transmitir vídeo en redes con gran capacidad de ancho de banda. No necesita la descarga e instalación de codecs puesto que los lleva incorporados siendo de mucha utilidad.

Soporta muchos códecs de audio y video, así como diferentes tipos de archivos, además soporta los formatos de DVD, VCD y varios protocolos streaming. Permite streaming múltiple y Vídeo bajo demanda (VoD).

Además, puede ser usado como servidor unicast o multicast en redes IPv4 o IPv6.

3.2 Instalación Servidor Streaming de Video.

Para la implementación del servidor streaming de video se ha utilizado un servidor con las siguientes características:

Hardware		Software
	HP Proliant ML 150	Darwin Streaming Server
Servidor virtualizado VmWare	Intel® Xeon a 2.8 GHz. Memoria Ram:4 GB Disco Duro: 72 Gb	Repositorio : http://dss.macosforge.org/post/previous-releases/
Acceso de Red	192.168.1.125	
Sistema Operativo	Linux	Ubuntu Server 10.04
Hostname	iptv.open-ims.test	

Tabla 6. Hardware y Software Requerido

En la dirección <http://dss.macosforge.org/post/previous-releases/> esta disponible el código de Darwing Streaming Server compatible con Linux.

Bajamos el archivo y lo guardamos en el disco del equipo. Extraemos el archivo y nos ubicamos en el directorio DarwinStreamingSrvr5.5.5-Linux

```
root@iptv:/home/user/Escritorio# tar xvf DarwinStreamingSrvr5.5.5-Linux.tar.gz
root@iptv:/home/user/Escritorio# cd DarwinStreamingSrvrlinux-Linux/
root@iptv:/home/user/Escritorio/DarwinStreamingSrvrlinux-Linux#
```

Ejecutamos el script de instalación:

```
root@iptv:/home/user/Escritorio/DarwinStreamingSrvrlinux-Linux#./Install
```

Crear el usuario administrador:

```
In order to administer the Darwin Streaming Server you must create an administrator user
[Note: The administrator user name cannot contain spaces, or single or double quote
characters, and cannot be more than 255 characters long]. Please enter a new
administrator user name:
```

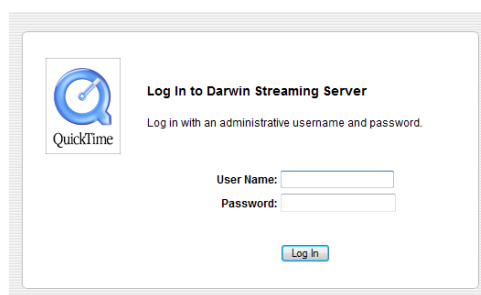
Ingresa el password del administrador

```
Re-enter the new administrator password:
```

Si los passwords son los mismos, la instalación finalizará, se visulizará un mensaje de confirmación:

```
Setup Complete!
```

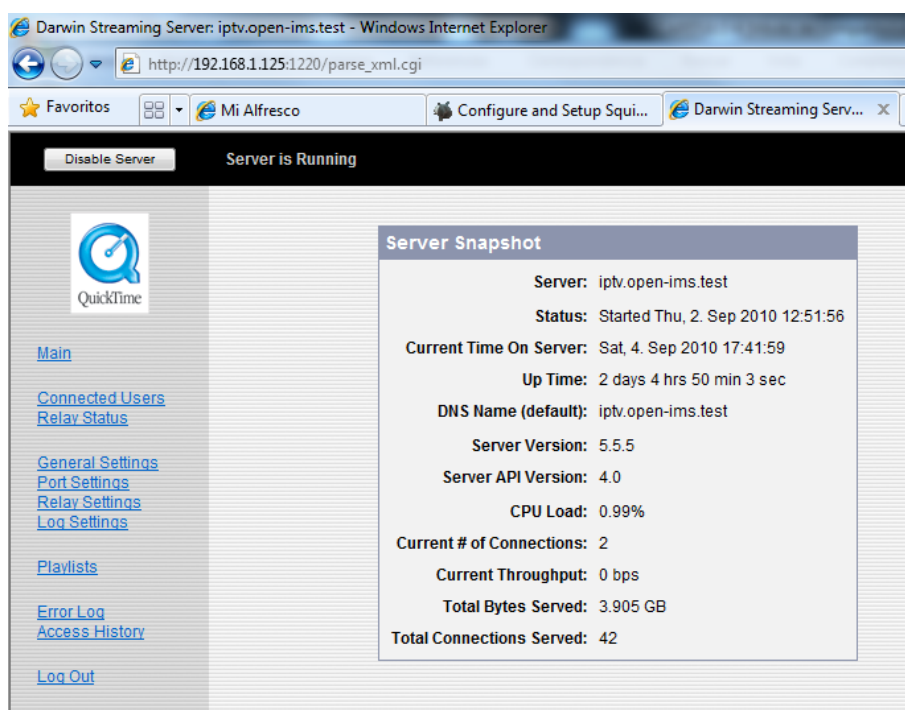
Utilizando el browser, habre el URL: <http://192.168.1.125:1220>



Fuente: Autor

Figura 29. Log In .DSS

Ahora es necesario crear Listas de reproducción Playlist. La interfaz web que ofrece es lo bastante simple como para no detenernos en ella, sin embargo no ocurre lo mismo con la codificación de los archivos multimedia.



Fuente: Autor

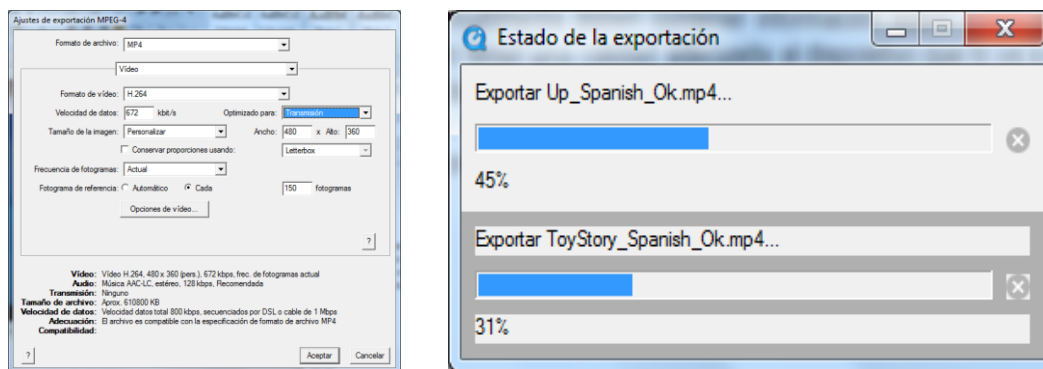
Figura 30. Server DSS Snapshot

Tras finalizar la ejecución del instalador se crean los siguientes archivos:

Darwin Streamign Server	
/usr/local/sbin/DarwinStreamingServer	Servidor
/usr/local/bin/PlaylistBroadcaster	Permite crear nuevas listas de reproducción
/usr/local/bin/MP3Broadcaster)	Permite crear nuevas listas de reproducción
/usr/local/sbin/streamingadminserver.pl	Interfaz Web
/etc/streaming	Directorio de configuración
/etc/streaming/streamingserver.xml	Configuración por defecto
/var/streaming/logs	Directorio de logs
/usr/local/movies	Directorio de medios

Tabla 7. Archivos de configuración DSS

Los archivos multimedia deben contener información sobre cómo se debe realizar el streaming, y además deben tener una calidad adecuada al dispositivo que lo va a reproducir, por lo que debemos codificar cada archivo que queremos servir. El DSS solo admite formatos 3gp, mov, mp4 y mp3, por lo que es posible que haya que recodificarlos. Para ello en la implementación de este prototipo, se utilizó Quick Time, la versión profesional.



Fuente: Autor

Figura 31. Exportar videos MOV

Una vez recodificados los archivos podemos crear las listas de reproducción sin problemas. Esto puede ser verificado si no se producen errores cuando ejecutamos play.

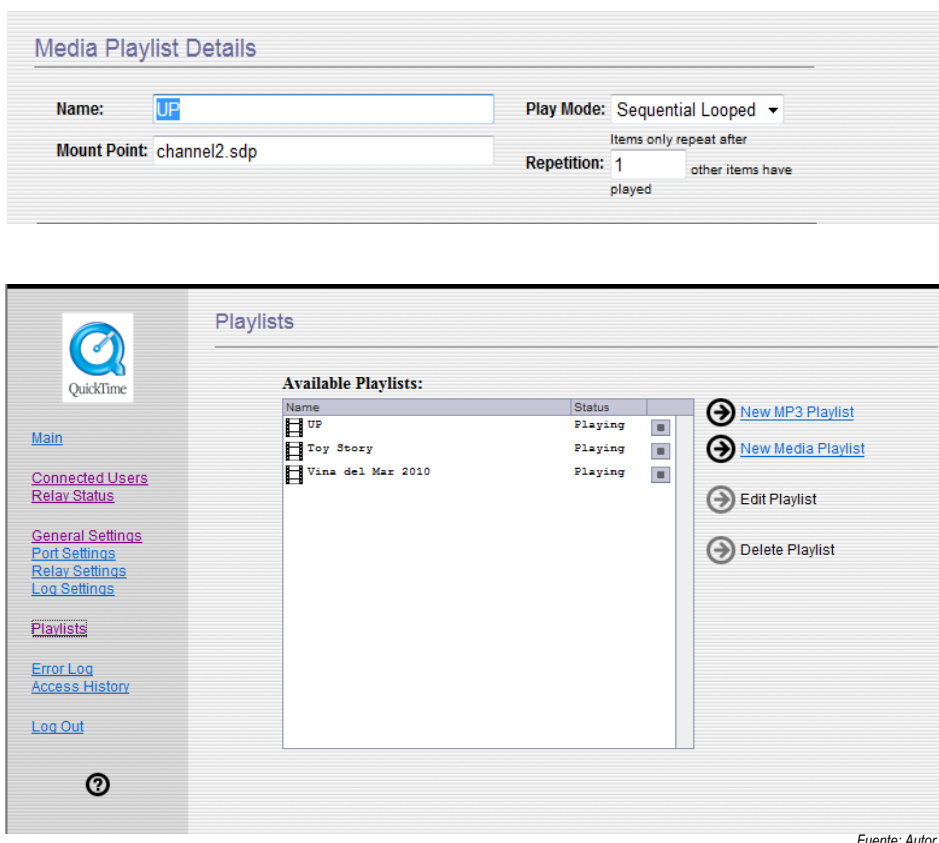


Figura 32. Playlist DSS

Los protocolos soportados son Real time Transport Protocol (RTP), Real Time Streaming Protocol (RTSP). La compresión de la transmisión es excelente y permite transmisiones de buena calidad, esto se debe a la utilización de los CODEC (COder-DECoder) de video H.263, H.264 y MPEG-4. Detecta el ancho de banda del cliente y transmite información acorde al mismo. El formato soportado por este servidor es MOV por lo tanto se requiere descargar e instalar un cliente que soporte este formato, por ejemplo VLC.

Para visualizar la correcta funcionalidad del servidor Darwin Streamign Server, es posible utilizar el media player VLC. Registramos en la opción Medio / Abrir volcado de red, la dirección ip del servidor streaming con el puerto por el que se entrega el servicio y el canal requerido.

rtsp://192.168.1.125:8000/channel1.sdp

3.3 Instalación Servidor Streaming de Video bajo demanda

Para el servicio de video bajo demanda se utilizó el software VideoLan el cual es una solución integral para el streaming de video y que posee licencia GNU (GPL). Instalamos el servidor de streaming en el equipo.

```
root@iptv:/home/user/Escritorio# apt-get install vlc
```

Para configurar el inicio manual del servidor de streaming de video bajo demanda ejecutamos el siguiente comando:

```
root@iptv:/home/user/Escritorio/ vlc --ttl 12 -vvv --color -I telnet --telnet-password videolan --rtsp-host 0.0.0.0:5554
```

A través de este comando, comenzamos a escuchar al servidor de VLC la IP que va a escuchar y el puerto 5554 a utilizar. Este puerto se utiliza para intercambiar tramas RTP / RTSP. Además se configura el acceso al servidor a través de telnet con el password videolan.

Añadir los medios de comunicación para escuchar, para ello se debe abrir una sesión telnet en el servidor utilizando el puerto 4212. Cuando la consola telnet de VLC pide una contraseña, se debe introducir la contraseña telnet que había fijado, en este caso **videolan**.

```
root@iptv:/home/user/Escritorio/ telnet localhost 4212
Trying 127.0.0.1...
Connectd to localhost.
Escape character is '^]'.
Password:
Welcome, Master
>
```

Ahora se debe añadir la película. Para añadir una película se debe

Welcome, Master

➤ New avatar vod enabled input I\VIDEOS\Up_Spanish.avi

Puede personalizar el nombre de la sesión de video bajo demanda por cualquier nombre, ya que es sólo una etiqueta para identificar la película. La última parte es la ruta de acceso a la película en el disco duro. Se puede agregar más películas usando el mismo comando y el formato - sólo hay que asegurarse de tener etiquetas únicas y nombres para cada película que agregue a la lista para escuchar.

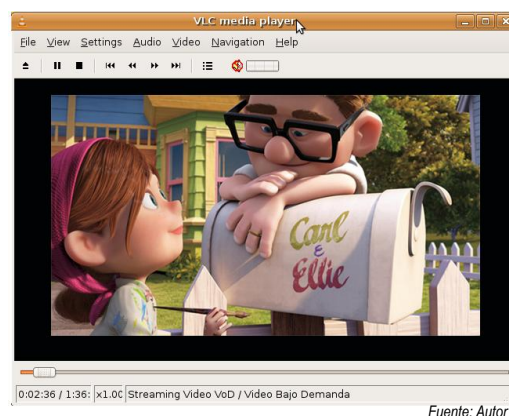
Para probar la correcta funcionalidad del servidor de video bajo demanda VLC, es recomendable usar el cliente de VLC, ya que muchos reproductores tienen diferentes implementaciones del protocolo RTSP.

Por ejemplo el Reproductor de Windows Media utiliza RTSP sobre RTP, mientras que la aplicación VLC utiliza el protocolo RTSP a través de TCP. Así que el Reproductor de Windows Media no será capaz de reproducir un streaming de video VLC RTSP a pesar de que admite el protocolo RTSP.

Registamos en la opción Medio / Abrir volcado de red, la dirección ip del servidor streaming con el puerto por el que se entrega el servicio y el canal requerido.

rtsp://192.168.1.125:5554/up

El tipo de streaming creado es el de video bajo demanda. El streaming y los comandos de programación del video bajo demanda vod comandos pueden ser analizados utilizando el comando **help** en la consola telnet.



Fuente: Autor

Figura 33. Funcionamiento Video Bajo Demanda

3.4 Configuración Streaming de Video Multicast

En la transmisión Multicast, el stream de datos es enviado a una dirección IP Multicast (las direcciones IP reservados son de 224.0.0.0 a 239.255.255.255). Entonces, cualquier máquina de la red puede unirse al grupo Multicast mediante el envío de una solicitud en la red, y recibirá automáticamente el stream de datos. Cuando se envía una invitación a abandonar el grupo, automáticamente dejará de recibir el stream de datos. La ventaja de la transmisión de Multicast es que sólo las máquinas que desea recibir el streaming de video la reciben efectivamente, y el servidor de streaming sólo envía una corriente, incluso si hay varios clientes que lo recibe.

Para iniciar el servidor VLC se debe:

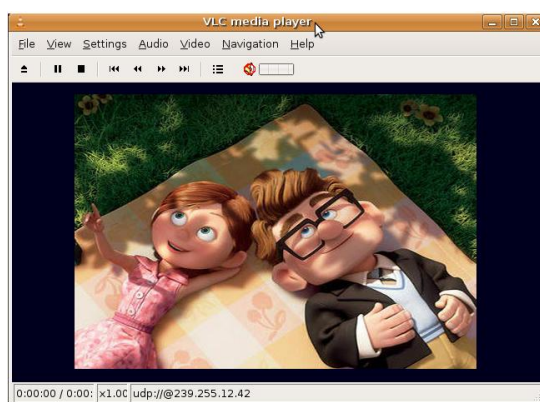
```
root@iptv:/home/user/Escritorio/ vlc -vvv /VIDEOS/Up_Spanish.mp4 --sout udp: 239.255.12.42 --ttl 12
```

donde :

- Up_Spanish.mp4 es el archivo que quiere transmitir
- 239.255.12.42 es la dirección IP Multicast
- 12 es el valor de la TTL (Time To Live) de los paquetes IP (lo que -- significa que la secuencia de multidifusión podrá cruzar 12 routers).

Para iniciar el cliente VLC

```
root@iptv:/home/user/Escritorio/ vlc -vvv udp:@239.255.12.42
```



Fuente: Autor

Figura 34. Funcionamiento Streaming Multicast

Para que el servicio streaming de video Multicast funcione es un requerimiento que el switch soporte esta característica. En el anexo D se detalla la configuración realizadas en el switch enterasys

3.5 Configuración UCT IPTV Streaming Server.

Basado en la arquitectura del prototipo hasta ahora se han configurado el Core de la plataforma IMS, se ha implementado el servidor streaming de video. Ahora se va a configurar en el plano de aplicación el servidor denominado UCT IPTV AS.

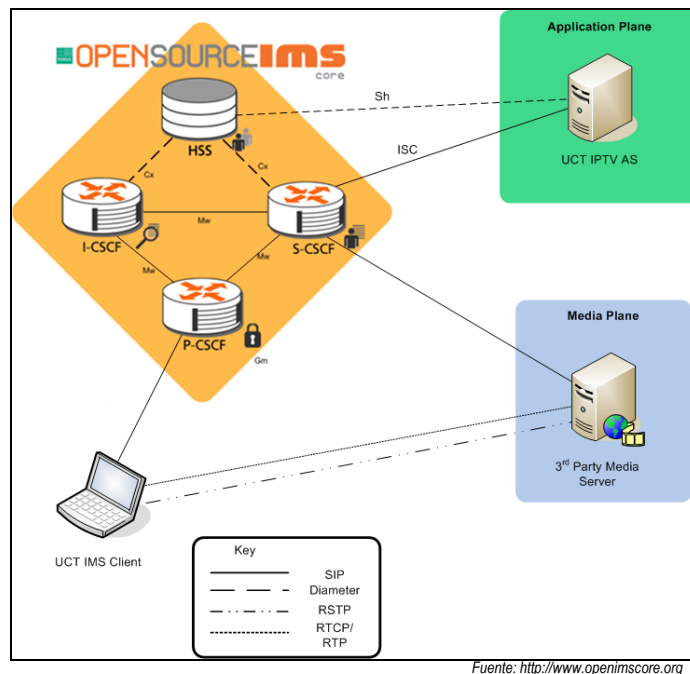


Figura 35. Arquitectura prototipo IMS Core - Streaming de video

El UCT IPTV AS, es una implementación de código abierto de un sistema de IPTV basada en estándares ETSI TISPAN. El sistema propuesto incorpora métodos de adaptación de contenidos multimedia para adaptar el vídeo para satisfacer diferentes dispositivos de cliente de Terminal Server y las características de acceso a la red. A continuación se detallan los prerequisites para instalar el servidor de aplicaciones.

```
root@iptv:~/Escritorio/uctiptv_advanced# apt-get update
root@iptv:~/Escritorio/uctiptv_advanced# apt-cache search libosip
>> libosip2-2 - Session Initiation Protocol (SIP) library
>> libosip2-dev - development files for the SIP library

root@iptv:~/Escritorio/uctiptv_advanced# apt-get install libosip2-2 libosip2-dev
root@iptv:~/Escritorio/uctiptv_advanced# apt-cache search libexosip
>> libexosip2-4 - eXtended OSIP library
>> libexosip2-dev - eXtended OSIP library development files

root@iptv:~/Escritorio/uctiptv_advanced# apt-get install libexosip2-4 libexosip2-dev
```

La configuración realizada fue probada con éxito en el servidor `iptv.open-ims.test`. La instalación del componente consiste en bajar el archivo `uctiptv_advaneecd.tar.gz`, localizado en https://developer.berlios.de/project/showfiles.php?group_id=7844 e instalarlo. El archivo se encuentra con un tipo de compresión `tar.zip`.

```
root@iptv:~/Escritorio/uctiptv_advanced# ls -l
```

```
root@iptv:~/Escritorio/uctiptv_advanced# ls -l
total 40
-rwxrwxrwx 1 user user 714 2010-09-06 17:57 key_value_file
-rw-r--r-- 1 root root 1067 2010-07-12 15:39 key_value_file.bak
-rw-r--r-- 1 user user 722 2008-08-01 08:17 Makefile
drwxr-xr-x 2 user user 4096 2010-08-14 12:28 obj
drwxr-xr-x 2 user user 4096 2010-02-04 15:37 src
-rwxr-xr-x 1 root root 18790 2010-02-06 10:01 uctiptv_as
```

Las listas de reproducción creadas en el apartado anterior deben ser mapeadas a SIP-URI. Éste es un esquema de direccionamiento SIP que identifica un recurso de comunicaciones, en este caso el servidor de streaming de aplicaciones (Application Server) utiliza para establecer sesiones SIP. El siguiente código muestra un ejemplo de cinco canales configurados en el archivo `key_value_file`

```
root@iptv:~/Escritorio/uctiptv_advanced# vim key_value_file
```

```
<key-value_pairs>
  <key-value_pair>
    <key>channel1</key>
    <value>rtsp://iptv.open-ims.test:8000/channel1.sdp</value>
  </key-value_pair>

  <key-value_pair>
    <key>channel2</key>
    <value>rtsp://iptv.open-ims.test:8000/channel2.sdp</value>
  </key-value_pair>

  <key-value_pair>
    <key>channel3</key>
    <value>rtsp://iptv.open-ims.test:8000/channel3.sdp</value>
  </key-value_pair>

  <key-value_pair>
    <key>channel4</key>
    <value>rtsp://iptv.open-ims.test:5554/avatar</value>
  </key-value_pair>

  <key-value_pair>
    <key>channel5</key>
    <value>rtsp://iptv.open-ims.test:5554/avatar2</value>
  </key-value_pair>

  <key-value_pair>
    <key>channel6</key>
    <value>rtsp://iptv.open-ims.test:5554/up</value>
  </key-value_pair>
</key-value_pairs>
```

El UCTP IPTV Streaming server esta listo para aceptar solicitudes SIP INVITE en el puerto 8010. El servidor de aplicaciones puede ser iniciada de la siguiente manera.

```
root@iptv:~/Escritorio/uctiptv_advanced# ./uctiptv_as key_value_file
```

```
Dave Waiting and Robert Marston (2008)
```

```
Exosip started and listening on port = 8010
```

```
Creating Hashtable
```

```
Population table with key-value pairs---
```

```
Numer of key-value pairs found in fle key_file is 6
```

```
Done
```

```
Server is ready to accept client requests.....
```

4 CAPÍTULO IV. CONFIGURACIÓN DEL SERVICIO

En este capítulo se agrega el Application Server (AS) con el core de la plataforma IMS. Se describen los procedimientos realizados.

4.1 Configuración Open IMS Core & Streaming server.

Las configuraciones especificadas en este apartado deben ser repetidas por cada servicio a ser incluido en la plataforma IMS. Para esto se debe:

1. Agregar el Application Server (AS) al registro HSS. (El servidor utiliza el puerto 8010)
2. Agregar un Trigger Point al registro HSS. El Trigger Point define el comportamiento del AS.
3. Unir los pasos 1 y 2 con un Initial Filter Criteria (iFC).
4. Agregar el iFC a un perfil de Servicio. Este perfil es utilizado para administrar los servicios dentro de grupos. Lo que permitirá habilitar o deshabilitar servicios a usuarios con determinados perfiles.

4.1.1 Agregar el Application Server (AS) al registro HSS

Primero se crea un AS con la dirección IP, puerto y FQDN del servidor. Para registrar el Application Server, se debe seleccionar en el menú **Services** → **Application Servers** → **Create**

The screenshot shows the FHoSS web interface for creating a new Application Server. The interface has a top navigation bar with 'HOME', 'USER IDENTITIES', 'SERVICES', 'NETWORK CONFIGURATION', and 'STATISTICS'. The 'SERVICES' menu is selected, and the 'Create' option under 'Application Servers' is active. The form contains the following fields:

- ID: [Text input]
- Name*: [Text input]
- Server Name*: [Text input]
- Diameter FQDN*: [Text input]
- Default Handling*: [Dropdown menu, currently showing 'Session - Continued']
- Service Info: [Text input]
- Rep-Data Limit: [Text input, value '1024']

On the right side, there is a 'Permissions' table with columns for 'Permission for', 'UDR', 'PUR', and 'SNR'. The table lists various permissions and their status:

Permission for	UDR	PUR	SNR
Allowed Request	☒	☒	☒
Repository-Data	☐	☐	☐
IMPU	☐	☐	☐
IMS User State	☐	☐	☐
S-CSCF Name	☐	☐	☐
iFC	☐	☐	☐
Location	☐	☐	☐
User-State	☐	☐	☐
Charging-Info	☐	☐	☐
MS-ISON	☐	☐	☐
PSI	☐	☐	☐
Activation	☐	☐	☐
DSAI	☐	☐	☐
Aliases	☐	☐	☐
Rep-Data	☐	☐	☐

At the bottom of the form, there are 'Save', 'Refresh', and 'Reset' buttons. A note states: 'Mandatory fields were marked with "*"'. The FOKUS logo is visible in the top right corner.

Figura 36. FHoSS. Interfaz web

Fuente: Autor

La figura 37 muestra la configuración realizada desde la interfaz web. Se configura la dirección IP y el puerto utilizado por el servidor de aplicaciones.

Sh Interface - Permissions			
Permission for	UDR	PUR	SNR
Allowed Request	☒	☒	☒
Repository-Data	☐	☐	☐
IMPU	☐	☐	☐
IMS User State	☐	☐	☐
S-CSCF Name	☐	☐	☐
IFC	☐	☐	☐
Location	☐	☐	☐
User-State	☐	☐	☐
Charging-Info	☐	☐	☐
MS-ISDN	☐	☐	☐
PSI Activation	☐	☐	☐
DSAI	☐	☐	☐
Aliases Rep Data	☐	☐	☐

Fuente: Autor

Figura 37. Registro Application Server

4.1.2 Agregar un Trigger Point al registro HSS.

Aquí se define un patrón (expresiones regulares) que permite el S-CSCF detectar la solicitud del servicio dentro de los encabezados de mensajes SIP predefinidos. El tipo de mensaje y el patrón que identifica la solicitud dependen del funcionamiento lógico del AS. Esta combinación de condiciones se denomina Trigger Point. Para crearlo: En el menú **Services** → **Trigger Points** → **Create**. La figura 38 muestra la configuración del Trigger Point.

Trigger Point -TP-

ID: 2
 Name*: IPTV_trigger
 Condition Type CNF*: Conjunctive Normal Form

Mandatory fields were marked with "*"

Attach IFC: Select IFC... [Attach]

List of attached IFCs:

ID	IFC Name	Detach
2	iptv_filter	☐

Save Refresh Delete

Add SPTs to Trigger Point

Not ☐ SIP Method INVITE [Delete]

OR

Request-URI [v]

AND

Not ☐ SIP Header SIP Header To [Delete]

SIP Header Content .*iptv.open-ims.test.*

OR

Request-URI [v]

AND

Request-URI [v]

Fuente: Autor

Figura 38. Agregar un Trigger Point

4.1.3 Unir los pasos 1 y 2 con un Initial Filter Criteria (iFC).

La información que el HSS almacena en los iFC (Initial Filter Criteria) es muy importante, su propósito es proveer un mecanismo de filtrado que permite mandar los mensajes SIP al servidor de aplicaciones correspondiente a la hora de provisionar un servicio. El iFC detecta la solicitud de los servicios a partir de expresiones regulares que se encuentran como parámetros dentro de los encabezados específicos de mensajes SIP. Cuando se cumplen las condiciones dispara la entrega del servicio correspondiente, en este caso permite definir la ruta y el trato que recibe el mensaje SIP hacia el AS.

The screenshot shows a web form titled "Initial Filter Criteria -iFC-". It contains several input fields: "ID" with the value "2", "Name*" with "iptv_filter", "Trigger Point" with a dropdown menu showing "IPTV_trigger", "Application Server*" with a dropdown menu showing "iptv", and "Profile Part Indicator" with a dropdown menu showing "Any". Below the fields, a note states "Mandatory fields were marked with '*'". At the bottom, there are three buttons: "Save", "Refresh", and "Delete".

Fuente: Autor

Figura 39. Initial Filter Criteria (iFC)

4.1.4 Agregar el iFC a un perfil de Servicio

Éste perfil permite administrar los servicios que tienen registrados los usuarios, de tal forma que los servicios pueden ser habilitados para usuarios que cuenten con dicho perfil y deshabilitados para aquello que no.

The screenshot shows a web form titled "Shared iFC Sets -Sh-iFC-". It contains two input fields: "ID-Set" with the value "1" and "Name*" with "default_shared_set". Below the fields, a note states "Mandatory fields were marked with '*'". At the bottom, there are three buttons: "Save", "Refresh", and "Delete".

Below the buttons, there is a section titled "Attach iFC" with a dropdown menu labeled "Select iFC...", a "Priority" input field with the value "0", and an "Attach" button.

Below the "Attach iFC" section, there is a warning message: "Warning: Priority values defined here can overwrite priority values defined in SP-iFC setup!".

Below the warning message, there is a section titled "List of attached iFCs" with a table:

ID	Name	Priority	Detach
1	default_ifc	0	☐
2	iptv_filter	2	☐

Fuente: Autor

Figura 40. Configuración Shared iFC

4.2 Instalación y configuración de Open IMS Client

Open IMS client es un cliente desarrollado por el Instituto Fraunhofer para que sea plenamente funcional con Core Open IMS. Se propone un entorno configurable donde desarrollar diversas aplicaciones IMS, alineado con las especificaciones 3GPP, IETF y TISPAN.

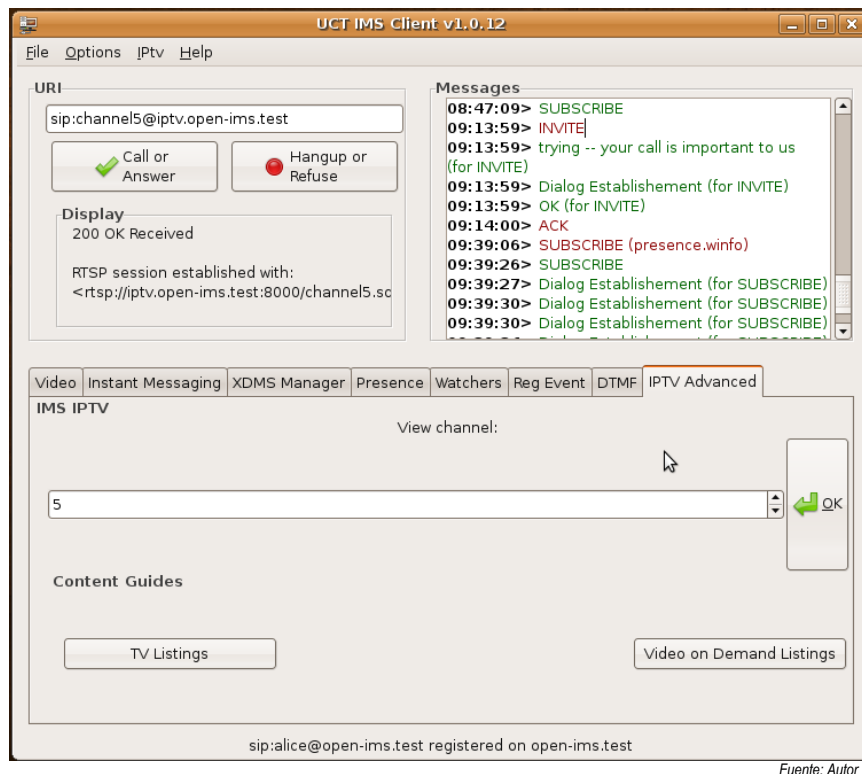


Figura 41. UCT IMS Client v 1.0.12

Fuente: Autor

UCT IMS Client se divide en: El interfaz gráfico donde el usuario tiene acceso a los servicios disponibles, La capa de servicios donde se encuentra la complejidad del cliente para diversos servicios IMS (VoIP, Presencia, Registro, Notificación de Eventos, Mensajería, IPTV Advanced...) que puede ser extensible y finalmente la capa del motor IMS, donde se encuentran las pilas de protocolos a utilizar.

UCT IMS Client debe ser instalado en un equipo con ubuntu 8.04. Se realizaron varias pruebas con las versiones actuales de Ubuntu, sin embargo se genera un error de segmentación cuando se requiere el funcionamiento de IPTV. UCT IMS Client ofrece un amplio rango de funcionalidades.

Para la instalación del cliente se utilizó un equipo con las siguientes características:

Cliente 1		
Hardware & Software	HP Pavilion ze5600 Notebook	
	Intel® Pentium 4. Memoria Ram:512 MB Disco Duro: 50 Gb	Repositorio : deb http://mirrors.163.com/ubuntu/ intrepid main universe multiverse UCT IMS Client: https://developer.berlios.de/project/showfiles.php?group_id=7844
Acceso de Red	192.168.1.128	
Sistema Operativo	Linux	Ubuntu Server 8.04
Hostname	user1.open-ims.test	

Figura 42. Características de Hardware & Software Cliente 1

Cliente 2		
Hardware & Software	HP Pavilion ze5600 Notebook	
	Intel® Core™ i5 . Memoria Ram:4GB Disco Duro: 250 Gb	Repositorio : deb http://mirrors.163.com/ubuntu/ intrepid main universe multiverse UCT IMS Client: https://developer.berlios.de/project/showfiles.php?group_id=7844
Acceso de Red	192.168.1.129	
Sistema Operativo	Linux	Ubuntu Server 8.04
Hostname	user2.open-ims.test	

Figura 43. Características de Hardware & Software Cliente 2

Una vez instalado el sistema operativo Ubuntu Server 8.04, es necesario adicionar el path del repositorio para instalar los requerimientos de UCT IMS Client.

```
root@user1:~# cd /etc/apt/
root@user1:~# vim source.list

deb http://mirrors.163.com/ubuntu/ intrepid main universe multiverse
```

Ahora actualizamos los repositorios

```
root@user1:~# apt-get update
```

Ahora instalamos las dependencias antes de la instalación del cliente IMS.

```
root@user1:~# apt-get install libosip2-3deb
root@user1:~# apt-get install libexosip2-dev
root@user1:~# apt-get install libgtk2.0-dev
root@user1:~# apt-get install libxml2-dev
root@user1:~# apt-get install libcurl4-openssl-dev
root@user1:~# apt-get install libgstreamer0.10-0
root@user1:~# apt-get install libgstreamer-plugins-base0.10-dev
root@user1:~# apt-get install gstreamer0.10-plugins-base
root@user1:~# apt-get install gstreamer0.10-plugins-good
root@user1:~# apt-get install gstreamer0.10-plugins-bad
root@user1:~# apt-get install gstreamer0.10-plugins-ugly
root@user1:~# apt-get install gstreamer0.10-ffmpeg
root@user1:~# apt-get install libavcodec-unstripped-51
root@user1:~# apt-get install libvlc-dev
root@user1:~# apt-get install vlc
```

Para instalar el cliente es necesario bajar el instalador de uctimsclient1.0.12.deb.
Para la instalación se debe:

```
root@user1:~# dpkg -i uctimsclient1.0.12.deb
(Reading database ... 100370 files and directories currently installed.)
Preparing to replace uctimsclient 1.0.12 (using uctimsclient1.0.12.deb) ...
Unpacking replacement uctimsclient ...
Setting up uctimsclient (1.0.12) ...
```

Antes de ejecutar el cliente IMS, es necesario verificar que el equipo tenga apuntado al servidor DNS 192.168.1.124, para ello se debe editar el archivo resolv.conf. Para ejecutar el cliente

```
root@user1:~# uctimsclient
(uctimsclient:27416): Gtk-WARNING **: GtkSpinButton: setting an adjustment with non-zero page
size is deprecated
Segmentation fault
```

Después del registro del usuario, y al utilizar el servicio de streaming de video, se genera un error de segmentación. Esto es provocado por la version de los prerequisites libvlc-dev y vlc. Para solventar este error se sugiere comentar el repositorio añadido en el archivo source.list y realizar la reinstalación de los prerequisites y el cliente IMS. Para ello:

```
root@user1:~# sudo apt-get update

root@user1:~# apt-get remove --purge vlc vlc-plugin-esd mozilla-plugin-vlc vlc* && sudo
apt-get install vlc vlc-plugin-esd mozilla-plugin-vlc vlc*

root@user1:~# dpkg -i uctimsclient1.0.12.deb

root@user1:~# uctimsclient
```



Figura 44. Video UCT IMS Client.

Fuente: Autor

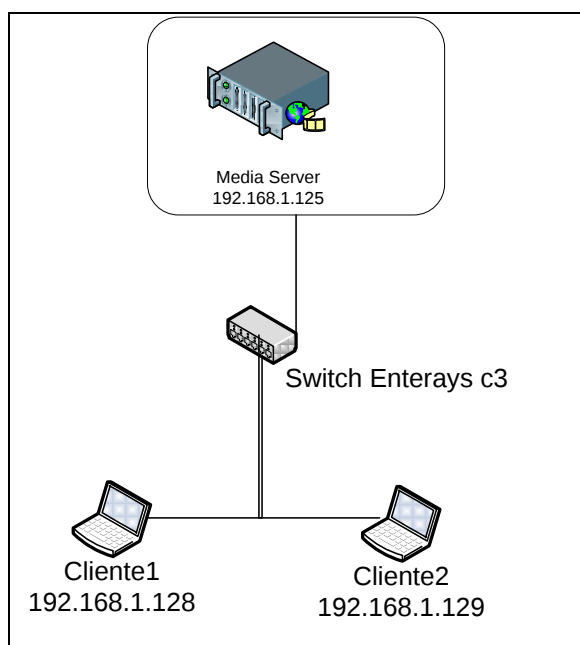
5 CAPÍTULO V. ESCENARIO DE PRUEBAS

En este capítulo se define un escenario de pruebas donde se valida el funcionamiento del servidor streaming de video, y del Open IMS Core como prototipo de una plataforma IP Multimedia Subsystem.

5.1 Análisis Servidor Streaming de Video

Para determinar el funcionamiento del servidor de Video Streaming, se llevaron a cabo una serie de pruebas basadas en el software wireshark (Analizador de protocolos), el servidor Emisor de video Streaming VLC y el cliente VLC. Con esto se obtuvo una serie de capturas del tráfico cursado, lo que permitió analizar las tramas que se intercambian entre el servidor y el cliente. Para las pruebas de funcionamiento del servidor streaming de video se excluye el Core de la plataforma IMS.

En el diseño simplificado de la figura 45, el servidor emite el streaming de video, y un cliente accesa al streaming de video proveniente del servidor.



Fuente: Autor

Figura 45. Diseño simplificado Servidor Streaming de Video- Clientes

5.1.1 Formatos soportados

Los servidores streaming de video pueden leer algunos formatos de video. A continuación se detalla los formatos examinados en los servidores DSS y VLC implementados en este proyecto de tesis, tanto en transmisiones unicast como en Multicast

Formato	Darwin Streaming Server DSS	VideoLan Server VLC
AVI		
MP4		
VOB		
MOV		

Tabla 8. Formatos examinados Servidores Streaming de Video

De la tabla 8 se puede concluir que el servidor streaming de video VLC presenta mayor soporte para una variedad de formatos; así, para los posteriores escenarios de pruebas se ha seleccionado videos con formato mp4. VLC es rápido codificando el vídeo de alta calidad, para contenidos en tiempo real y bajo demanda. Como se detallo en el capítulo 1.9.12 la calidad del video esta dada por el Bitrate + Resolución + Codec.

5.1.2 Multicast

En el capítulo 3.4, se configuró el servidor streamign de video con la dirección ip 192.168.1.125, para que emita en Multicast el streaming a la dirección 239.255.12.43. En Multicast, un único stream se comparte entre diferentes clientes, de forma que el servidor envía la información una única vez, y ésta llega a todos los clientes que han demandado el servicio.

Debido a que Multicast, utiliza el protocolo UDP, para la transmisión de una serie de paquetes de datos, no hay una manera sencilla de que el reproductor solicite un paquete de datos para que sea enviado de nuevo. Esto quiere decir que algunos paquetes se pierden, incluso antes de que el cliente pueda notarlo, debido en parte, a la manera en que el reproductor codifica los archivos y los enumera. Si se pierden una gran cantidad continua de paquetes será perceptible por el usuario en forma de cortes de sonido y de pausado de imágenes, siempre dependiendo de la amplitud del buffer que se haya configurado en el cliente.

En la figura 46 se muestra la captura en la que se filtra únicamente el protocolo IGMP. El cliente de la dirección 192.168.1.128 decide formar parte del grupo 239.255.12.43 y envía un Reporte No Solicitado de adherencia IGMP con dirección destino MAC 0x01:00:5E:00:00:02 (No. 3975). Esto para participar en el streaming de video que el servidor 192.168.1.125 esta emitiendo.

No. .	Time	Source	Destination	Protocol	Info
3975	32.024839	192.168.1.128	224.0.0.22	IGMP	V3 Membership Report / Join group 239.255.12.43 for
3979	32.044461	192.168.1.254	224.0.0.1	IGMP	V2 Membership Report / Join group 224.0.0.1
3980	32.046591	192.168.1.200	224.0.0.1	IGMP	V2 Membership Query, general
3984	32.064843	192.168.1.128	224.0.0.251	IGMP	V2 Membership Report / Join group 224.0.0.251
3993	32.124834	192.168.1.128	239.255.12.43	IGMP	V2 Membership Report / Join group 239.255.12.43
3995	32.131117	192.168.1.25	224.0.0.251	IGMP	V2 Membership Report / Join group 224.0.0.251
7688	53.611229	192.168.1.128	224.0.0.2	IGMP	V2 Leave Group 239.255.12.43
7691	53.623434	192.168.1.200	239.255.12.43	IGMP	V2 Membership Query / Join group 43.12.255.239
7921	54.622735	192.168.1.200	239.255.12.43	IGMP	V2 Membership Query / Join group 43.12.255.239

▶ Frame 7688 (46 bytes on wire, 46 bytes captured) ▶ Ethernet II, Src: HewlettP_28:32:5e (00:0f:20:28:32:5e), Dst: IPv4mcast_00:00:02 (01:00:5e:00:00:02) ▶ Internet Protocol, Src: 192.168.1.128 (192.168.1.128), Dst: 224.0.0.2 (224.0.0.2) ▶ Internet Group Management Protocol

Fuente: Autor

Figura 46. Tramas Multicast

Cuando el cliente 192.168.1.128 ya no desea formar parte del grupo 239.255.12.43, envía un mensaje de desincorporación no solicitado de grupo. (No. 7688).

5.1.3 Unicast. Video Bajo demanda

En el capítulo 3.3 se detallo el proceso para la instalación del servidor streaming de Video bajo demanda. El tipo de Video Streaming Unicast entrega streams uno a uno para cada cliente y se utiliza para el video on Demand (VoD), ya que cualquier usuario puede hacer una petición de una secuencia de stream en cualquier momento.

En la figura se presenta la captura de tramas donde se evidencia como el servidor 192.168.1.125 envia una corriente de datos de video al cliente 192.168.1.128, quien solicitó un video bajo demanda.

5.1.4 Simulaciones

En este apartado se van a analizar los resultados de las diferentes simulaciones con el número de usuarios, anchos de banda y bitrates con el objetivo de analizar el rendimiento del servidor streaming de video en los diferentes escenarios.

Se han seleccionado dos videos con diferente tasa de bits de compresión (bitrate), con el fin de realizar pruebas con distinto tipo de carga.

Para poder evaluar el rendimiento del servidor streaming de video en función del ancho de banda se ha optado por realizar observaciones subjetivas, desde el punto de vista del usuario final, en cuanto a la calidad del video recibido.

El laboratorio se ha llevado a cabo en el edificio del Ministerio de Electricidad con 47 clientes que utilizan equipos marca HP dotados de procesador Intel(R) Core(TM)2 Duo a 2,4Ghz, 4 GB y tarjeta Ethernet de 100Mbps. El método estadístico utilizado para el análisis de datos es el promedio de todos los datos por cada variable.

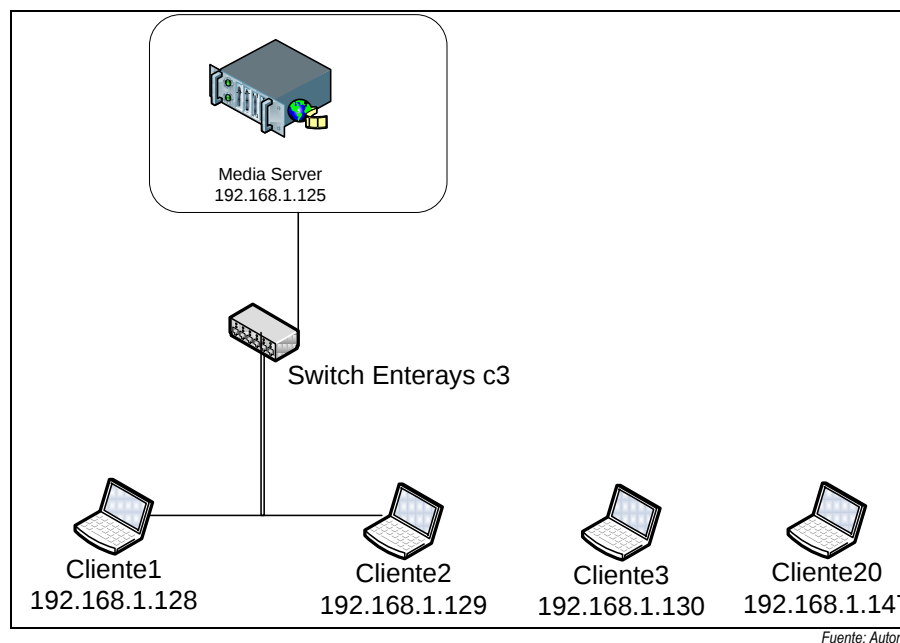


Figura 48. Esquema de red para simulación

5.1.4.1 Análisis Recursos del Servidor – Tipo de Streaming - Número de Clientes.

Para realizar la recolección de datos de los recursos de hardware en el servidor se ha empleado herramienta proporcionada para la distribución Ubuntu llamada “Conky”. Esta aplicación es capaz de registrar todo tipo de medidas de la mayoría de los componentes del sistema. En el anexo F se describe la instalación de esta aplicación.

Se han analizado los escenarios de estreaming de video: Unicast y Multicast, en donde se ha tomado dos tipos de medida: Porcentaje de uso de CPU y porcentaje de uso de memoria RAM. El uso de CPU, donde se espera que el rendimiento de la máquina no se vea deteriorado hasta alcanzar el 60 ó 70%. El uso de la memoria RAM, que en principio no debe saturarse, aunque si se llega a un número de peticiones suficientemente elevado, llegará a sobrecargarse, de manera que se tendrá que hacer paginación y el servicio se degradará considerablemente. Estas medidas se registrarán al mismo tiempo a intervalos regulares para poder comprobar la evolución en el tiempo del sistema. Con los datos obtenidos en cada una de las pruebas se creo un listado de datos con el fin de facilitar el análisis de los datos mediante una manera gráfica, se recurre al uso de MS Excel para crear gráficos de línea que nos relacionen el número de peticiones con los diversos indicadores.

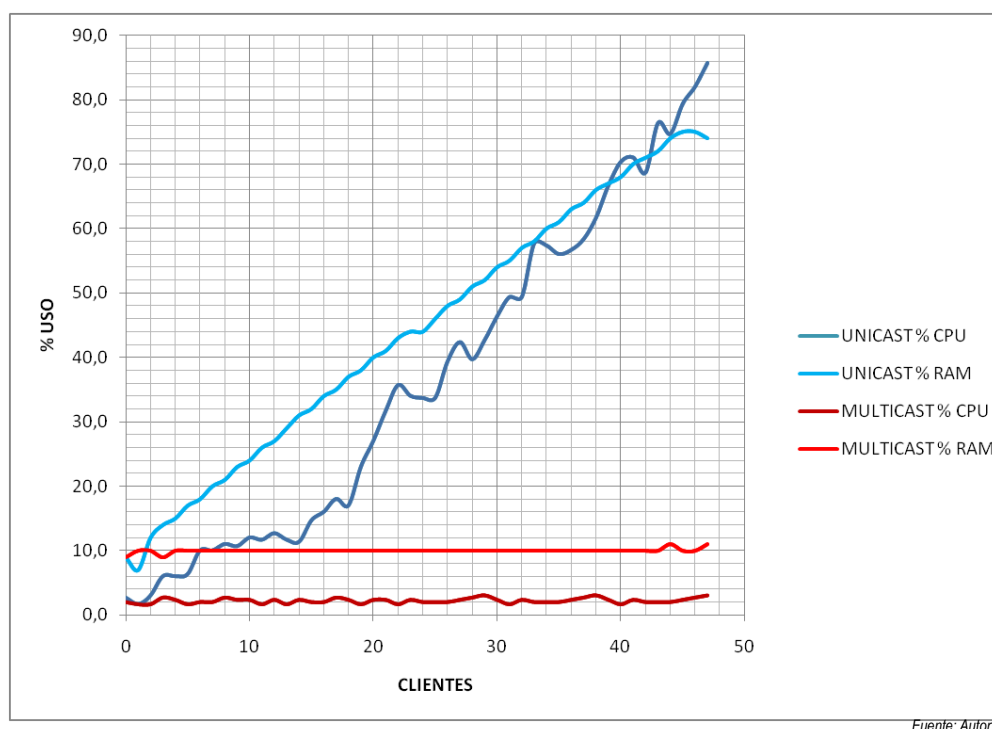


Figura 49. Analisis de recursos del servidor - tipo de streaming- Número de Clientes

Caso Unicast (VoD): En la figura 49 se observa que el porcentaje de uso del CPU y de memoria Ram es directamente proporcional al número de peticiones Unicast (VoD) realizada al servidor. Al comenzar una demanda mayor se empieza a sobrecargar la memoria, lo que lleva a la realización de más cantidad de paginado. Cuando el número de peticiones alcanza a las 47, el porcentaje de uso de CPU y memoria Ram llega a nivel de sobrecarga, 86% y 74% respectivamente. Es en este momento en que el índice de paginado aumentan desmesuradamente, como consecuencia el servicio VoD deja de funcionar.

Caso Multicast: Cuando el servidor streaming de video atiende peticiones de clientes del servicio streaming de video Multicast, se aprecia que no existe incremento del uso de recursos del Servidor y por tanto, los valores de las medidas se mantienen estables cuando el número de peticiones se incrementa. Por tanto se concluye que el streaming de video con IP Multicast es más eficiente en contraste con Unicast. . El Streaming de video en Multicast ahorra recursos en el servidor de video.

5.1.4.2 Análisis MOS - Ancho de banda

Una de las condiciones necesarias para evaluar el desempeño del servidor streaming de video, es la disponibilidad de ancho de banda en la línea de transmisión para el servicio de dicha aplicación. En la Figura 50 se observa el ancho de banda disponible (95 Mbps) entre el servidor streaming de video y un cliente a través de la red de cable.

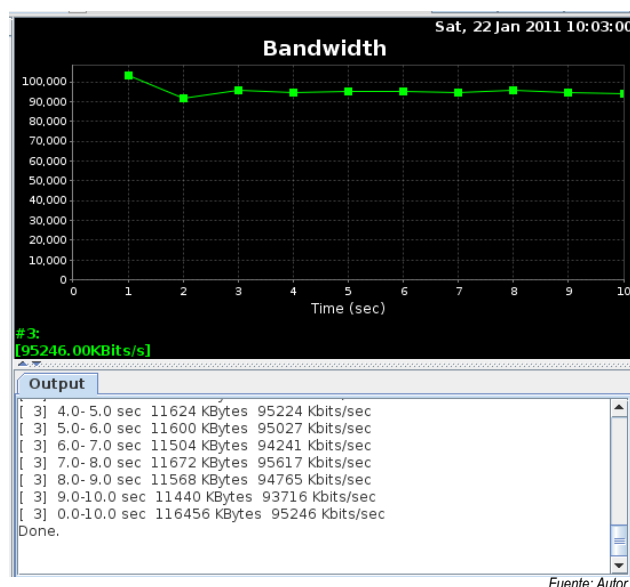


Figura 50. Ancho de banda en la red Lan

Para realizar un análisis del comportamiento de la calidad del video en relación con el ancho de banda la red en el cliente en un ambiente de video bajo demanda, se ha realizado pruebas limitando el ancho de banda disponible. Para efectos de administrar el ancho de banda en la tarjeta de red se ha utilizado la aplicación *wondershaper* para linux ubuntu que permite simular distintos escenarios. Ver anexo G para la instalación de esta herramienta.

Para la evaluación se ha optado por realizar observaciones subjetivas, desde el punto de vista del usuario final, en cuanto a la calidad del video recibido. Se utilizó el método subjetivo conocido como MOS (Mean Opinion Score) para determinar la calidad del video, metodología propuesta y autorizada por ITU-T. Este organismo ha definido parámetros para la medición de la calidad del contenido multimedia basado en la percepción del usuario. De acuerdo al estándar ITU-I, el MOS está clasificado dentro de 5 grupos definidos de la siguiente manera:

MOS	Valor	Observación
Excelente	5	El deterioro de la calidad de video es imperceptible
Bueno	4	El deterioro es perceptible, pero no molesta
Razonable	3	El deterioro de la multimedia es un poco molesto
Pobre	2	La multimedia recibida es molesta
Malo	1	La multimedia recibida está muy deteriorada

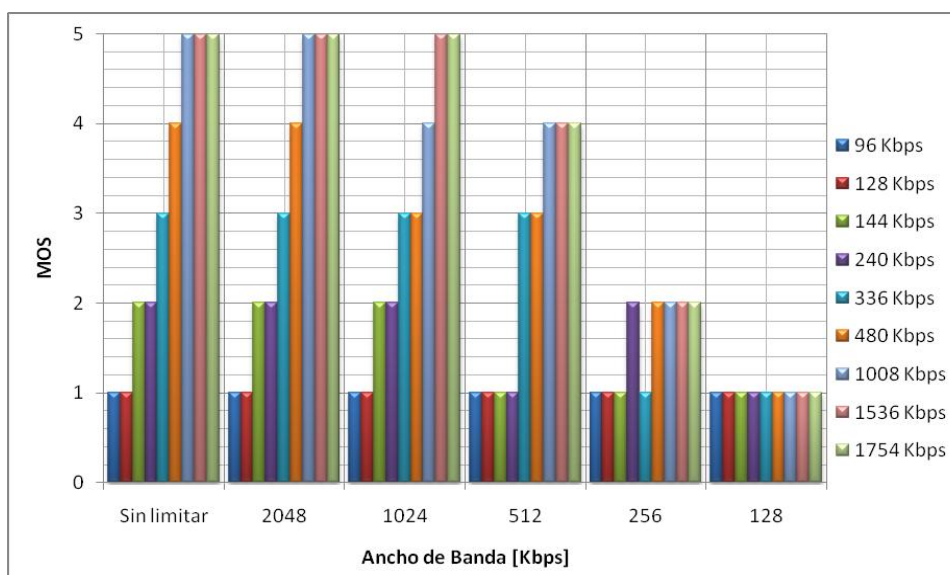
Tabla 9. Clasificación MOS

Las opiniones de la percepción de la calidad de video fueron analizadas en relación a los videos detallados en la tabla 10.

N°	Video	Codec	Bit rate	Frame rate	Tamaño	Duración
1	Up96.mp4	MPEG-4	96 Kbps	30 fps	134 Mb	96 mn
2	Up128.mp4	MPEG-4	128 Kbps	30 fps	142 Mb	96 mn
3	Up144.mp4	MPEG-4	144 Kbps	30 fps	151 Mb	96 mn
4	Up240.mp4	MPEG-4	240 Kbps	30 fps	217 Mb	96 mn
5	Up336.mp4	MPEG-4	336 Kbps	30 fps	281 Mb	96 mn
6	Up480.mp4	MPEG-4	480 Kbps	30 fps	371 Mb	96 mn
7	Up1008.mp4	MPEG-4	1008 Kbps	30 fps	647 Mb	96 mn
8	Up1536.mp4	MPEG-4	1536 Kbps	30 fps	831 Mb	96 mn
9	Up1754.mp	MPEG-4	1754 Kbps	30 fps	1,27 Gb	96 mn

Tabla 10. Parámetros de los videos de prueba

De los resultados obtenidos se detalla la tabla 11.



Fuente: Autor

Kbps	Máximo ancho de banda permitido					
Bit Rate del video	Sin limitar	2048	1024	512	256	128
96	1	1	1	1	1	1
128	1	1	1	1	1	1
144	2	2	2	1	1	1
240	2	2	2	1	2	1
336	3	3	3	3	1	1
480	4	4	3	3	2	1
1008	5	4	4	4	2	1
1536	5	5	5	4	2	1
1754	5	5	5	4	2	1

Tabla 11. Evaluación de calidad de video MOS para el servidor VLC en difusión VoD

Sin limitar el ancho de banda, se observa que las peticiones de contenido codificado a un bit rate bajo (96 kbps, 128 kbps, 144 kbps) mostraron una calidad perceptiva deficiente, es decir MOS=1, 2. La recepción de video es pobre y tiene mucho deterioro. Las pruebas con un bitrate de 240 kbps, 336 kbps, presentan efectos de bloques sobre la imagen y una falta completa de nitidez, estos defectos tiene un impacto en la percepción del cliente MOS=2,3.

Finalmente las peticiones con video codificado a bit rates altos: 480 kbps, 1008 kbps, 1536 kbps y 1754 kbps, las secuencias recibidas casi no presentaron defectos visuales MOS=4, 5.

Con estos resultados se ha verificado que cuanto más elevado sea el bitrate, mejor calidad de imagen obtenemos.

Respecto al manejo del ancho de banda, se observó que a partir de 512kbps, la calidad de video es razonable (MOS=3) con los videos cuya tasa de bits es superior a 336 kbps.

Con 1024 kbps de ancho de banda, la calidad de video llega a niveles entre Razonable y Buena. (MOS=3, 4). A partir de 2048 Kbps de ancho de banda el calidad del video llega a niveles de excelente (MOS=5)

Con esto podemos confirmamos que para que un canal emita video con una definición estándar es necesario tener una conexión con un ancho de banda mínimo 1,5 Mbps.

La versión VLC media player 0.8.6e janus no soporto videos con codificación H.264 y alto bit rate, sin embargo existe software comercial con el que soporta este tipo de servicio. Un canal que emite el servicio de streaming de video de alta definición requiere un ancho de banda mínimo de 8 Mbps. Si tenemos varios canales distintos en forma simultánea se requiere más ancho de banda. A este ancho de banda hay que sumar el necesario para la conexión a internet.

En promedio se puede hablar de 4.5 Mbps para tres canales de definición estándar u 11 Mbps para un canal con alta definición y dos con definición estándar. Estos cálculos son usando MPEG-4 para la compresión/codificación del vídeo.

5.2 Análisis Plataforma Open IMS Core

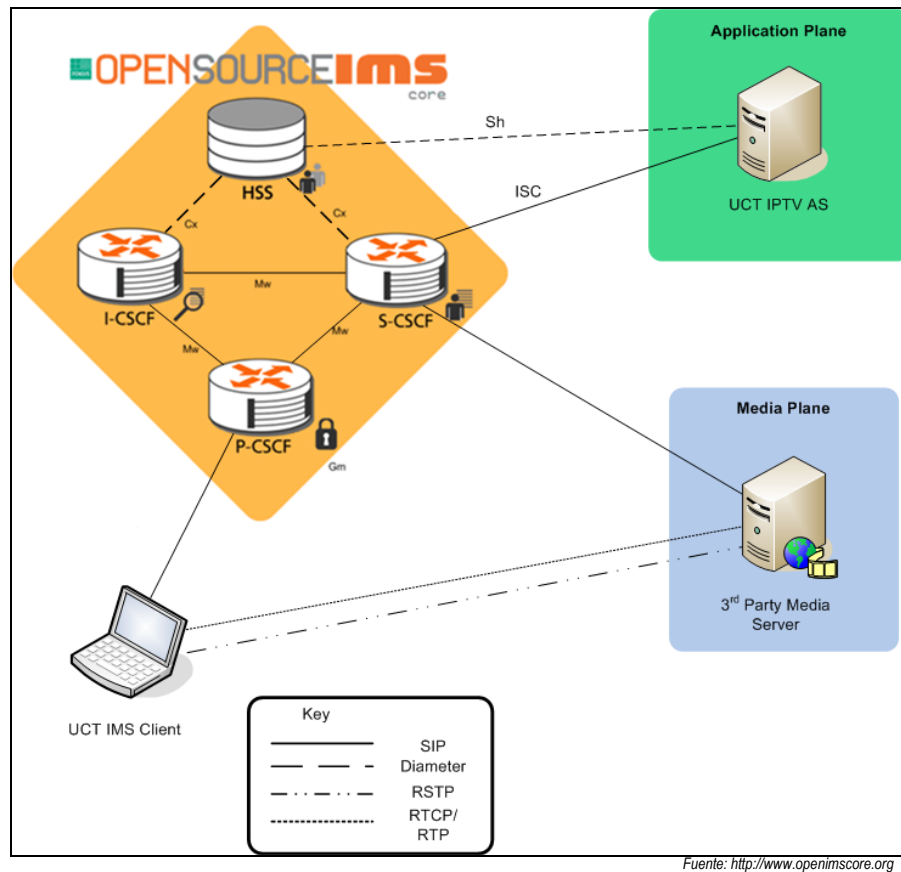


Figura 51. Plataforma Open Ims Core - Streaming de Video

La arquitectura del prototipo de streaming de video en la plataforma IMS consiste en tres etapas:

- 1ª Etapa:** El UCT IMS client crea una solicitud INVITE a uno de los canales y envía esta solicitud al Application Server (AS) a través del Core IMS
- 2ª Etapa:** El AS consulta a una tabla hash que une la solicitud del canal a las direcciones RTSP, y devuelve la correspondiente dirección de RTSP para el cliente UCT IMS responda con un 200 OK.
- 3ª Etapa:** El UCT IMS client inicia una sesión RTSP con el servidor streaming de video.

Está claro que es necesario que haya coordinación entre el canal solicitado por el cliente, la tabla hash que los canales mapean a las direcciones RTSP y las direcciones del servidor de streaming de video.

Todo este proceso es detallado a continuación:

Para entender el comportamiento de la plataforma del prototipo, la operación del Core de IMS, es explicado tomando en consideración como trabajan sus entidades durante el registro del usuario.

En la plataforma, se incluyó la configuración del servidor DNS. Su función es la de localizar cada entidad de la plataforma. El P-CSCF recibe una consulta por pcscf.open-ims.test. La respuesta es la dirección IP y el puerto en el cual el proceso pcscf.sh esta escuchando

Entonces el usuario Cristhian, envía un mensaje REGISTER al P-CSCF. Éste lo mantiene y genera una nueva transacción dirigida al I-CSCF. Ésta nueva transacción es basada en el mensaje recibido REGISTER

El I-CSCF verifica que el mensaje ha sido previamente procesado por el P-CSCF. Después de la confirmación, la entidad reescribe el U-URI y como la entidad previa genera una nueva transacción a la entidad S-CSCF disponible en la plataforma. El S-CSCF confirma el nombre de dominio, realiza un check del mensaje con el HSS y contesta con un mensaje 401 Unauthorized.

No. .	Time	Source	Destination	Protocol	Info
9	4.955411	192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
10	4.955897	192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.1
11	4.957157	192.168.1.128	192.168.1.124	SIP	Request: REGISTER sip:open-ims.test
12	4.998884	192.168.1.124	192.168.1.128	SIP	Status: 401 Unauthorized - Challengin

Fuente: Autor

Figura 52. Tráfico IMS. Consulta DNS

Este mensaje regresa al cliente IMS cristhian. El usuario envía otro mensaje REGISTER, en este caso, se incluye la información para autenticación, mostrada en la figura siguiente

Source	Destination	Protocol	Info
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP	Request: REGISTER sip:open-ims.test
192.168.1.124	192.168.1.128	SIP	Status: 200 OK - SAR succesful and registrar saved (1
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP	Request: SUBSCRIBE sip:cristhian@open-ims.test
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP/XML	Request: PUBLISH sip:cristhian@open-ims.test
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	SIP	Status: 200 Subscription to REG saved
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP	Request: SUBSCRIBE sip:cristhian@open-ims.test
192.168.1.128	192.168.1.124	SIP/XML	Request: PUBLISH sip:cristhian@open-ims.test
192.168.1.128	192.168.1.124	SIP	Request: SUBSCRIBE sip:cristhian@open-ims.test

Fuente: Autor

Figura 53. Tráfico IMS. Register

Este mensaje es trasladado en una nueva transacción entre el P-CSCF y el I-CSCF y otro entre este último y el S-CSCF. La instancia S-CSCF identifica ahora al usuario IMS como autorizado y se asigna así mismo como servidor de ese usuario. Antes de la comunicación sea establecida, el usuario IMS envía la información de registro al HSS para ser almacenada, lo que permite que la consola web sea actualizada y muestre al usuario como REGISTERED.

ID	4
Identity*	sip:cristhian@open-ims.test
Barring	☐
Service Profile*	default_sp
Implicit Set	4
Charging-Info Set	default_charging_set
Can Register	☒
IMPU Type*	Public_User_Identity
Wildcard PSI	
PSI Activation	☐
Display Name	
User-Status	UN-REGISTERED

Fuente: Autor

Figura 54. Usuario No Registrado

ID	4
Identity*	sip:crsthian@open-ims.test
Barring	☐
Service Profile*	default_sp
Implicit Set	4
Charging-Info Set	default_charging_set
Can Register	☒
IMPU Type*	Public_User_Identity
Wildcard PSI	
PSI Activation	☐
Display Name	
User-Status	REGISTERED

Fuente: Autor

Figura 55. Usuario Registrado

Sin embargo, el estado de registro del usuario IMS crsthian es también almacenado por el P-CSCF cuando este recibe el mensaje 200 OK para ser reenviado al usuario, como se muestra en la figura 56. Esta información es vital para el Proxy-CSCF para no permitir a usuarios no registrados introducir algún mensaje dentro de la plataforma Open IMS Core.

Finalmente, el usuario IMS envía un mensaje SUBSCRIBE para solicitar el estado actual y actualizaciones desde el servidor.

Source	Destination	Protocol	Info
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP	Request: REGISTER sip:open-ims.test
192.168.1.124	192.168.1.128	SIP	Status: 200 OK - SAR succesful and registrar saved (1
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP	Request: SUBSCRIBE sip:crsthian@open-ims.test
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP/XML	Request: PUBLISH sip:crsthian@open-ims.test
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	SIP	Status: 200 Subscription to REG saved
192.168.1.128	192.168.1.124	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP	Request: SUBSCRIBE sip:crsthian@open-ims.test
192.168.1.128	192.168.1.124	SIP/XML	Request: PUBLISH sip:crsthian@open-ims.test
192.168.1.128	192.168.1.124	SIP	Request: SUBSCRIBE sip:crsthian@open-ims.test
192.168.1.128	192.168.1.124	STP/XMI	Request: PUBLISH sip:crsthian@open-ims.test

Fuente: Autor

Figura 56. Tráfico IMS. Subscribe

El cliente IMS puede visualizar los eventos en la ventana Messages, figura 57.

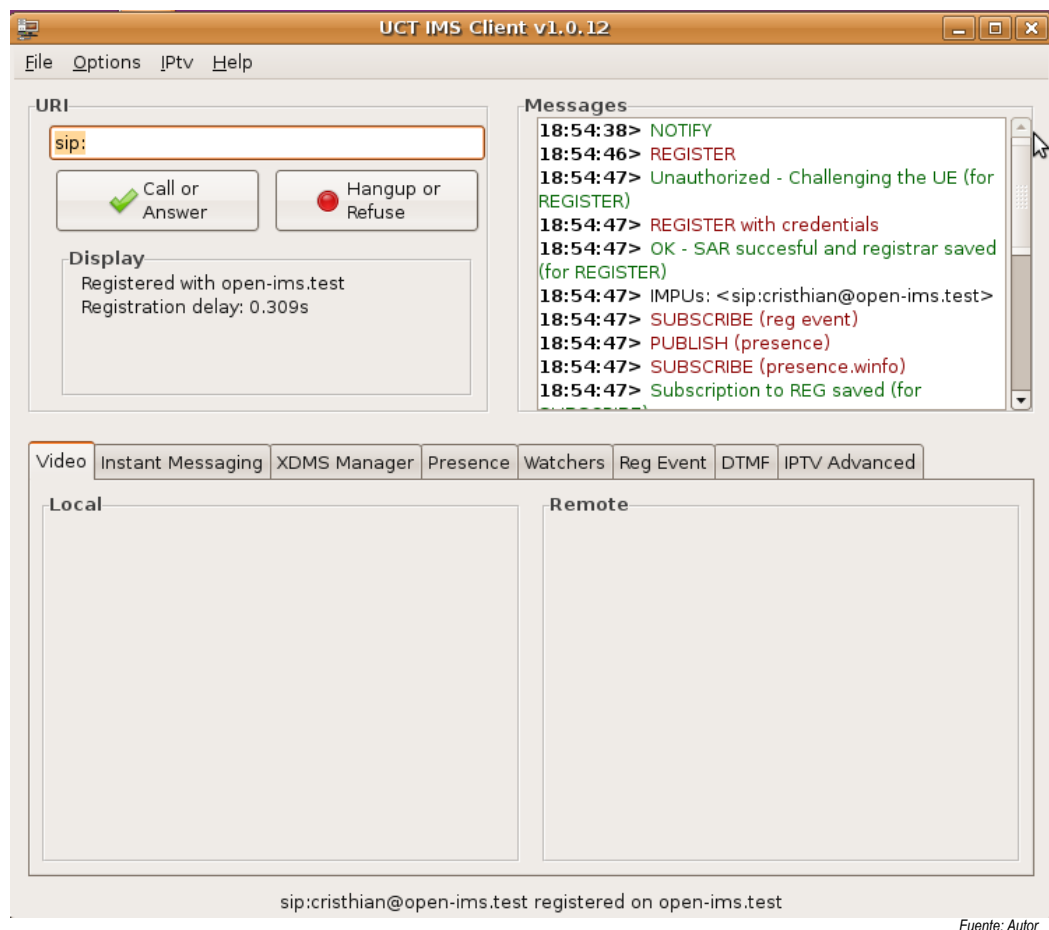


Figura 57. UCT IMS Client v1.0.12

Fuente: Autor

5.3 Análisis del servicio de Streaming de video en la plataforma IMS

UCT IMS Client permite a los usuarios solicitar canales IPTV y visualizarlos. El procedimiento de streaming de video se muestra en la figura 58:

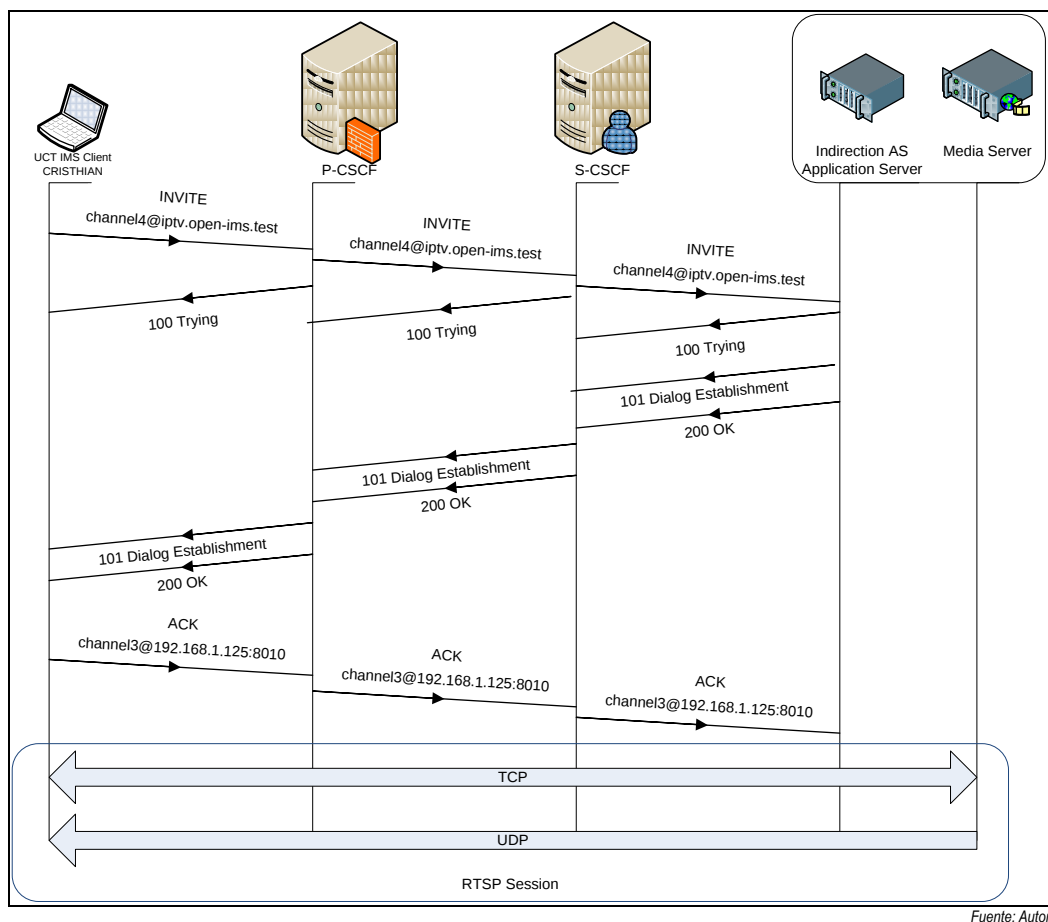


Figura 58. Análisis de Tráfico. Servicio Streaming de Video

El usuario Cristhian envía una mensaje INVITE al P-CSCF solicitando un canal IPTV (en el ejemplo es el canal 4) el cual reenvía al S-CSCF, esta entidad reconoce el dominio iptv.open-ims.test. El S-CSCF conoce esta información es gracias al registro del usuario: Cuando el usuario incluye información para la autenticación, el S-CSCF solicita datos del usuario del HSS, en donde la información del Application Server Indirection esta incluida.

Source	Destination	Protocol	Info
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP/SDP	Request: INVITE sip:channel4@iptv.open-ims.test, with
192.168.1.124	192.168.1.128	SIP	Status: 100 trying -- your call is important to us
192.168.1.124	192.168.1.128	SIP	Status: 101 Dialog Establishment
192.168.1.124	192.168.1.128	SIP	Status: 200 OK
192.168.1.128	192.168.1.124	DNS	Standard query A pcscf.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.124
192.168.1.128	192.168.1.124	SIP	Request: ACK sip:channel4@192.168.1.125:8010
192.168.1.128	192.168.1.124	DNS	Standard query A iptv.open-ims.test
192.168.1.124	192.168.1.128	DNS	Standard query response A 192.168.1.125

Figura 59. Solicitud de Canal IPTV

Fuente: Autor

Finalmente, el Application Server contesta el mensaje INVITE con un mensaje 200 OK, en el cual se indica el URL donde el servicio IPTV puede ser encontrado.

Figura 60. Tráfico de Vídeo Fuente: Autor



Figura 61. sip:channel2@iptv.open-ims.test

6 CAPÍTULO VI. CONCLUSIONES Y LÍNEAS FUTURAS

Una vez finalizada la fase de pruebas y validación de resultados se hace una presentación de las conclusiones extraídas de la realización del proyecto y se indican las líneas futuras de investigación que pueden ser identificadas en el presente trabajo.

6.1 Conclusiones.

Este proyecto concluyo exitosamente con los objetivos planteados al inicio.

1. En el proceso de la convergencia de las telecomunicaciones, han surgido todo tipo de terminales móviles, con capacidades muy heterogéneas que exigen la disponibilidad de medios que permitan el acceso a servicios asociados al usuario que los demanda, y no a los requisitos propios de cada plataforma de red, en este sentido IMS brindará solución a este problema, ya que su operación es independiente de la red de acceso.
2. Se logró diseñar e implementar con éxito el prototipo IMS de acuerdo a los objetivos planteados. El sistema obtenido constituye una plataforma base para la integración de futuros servicios de usuario final.
3. La implementación de una Plataforma de Entrega de Servicios construida a partir de soluciones abiertas, se presenta como el nodo catalizador en la adopción de redes NGN.
4. Se ha implementado una plataforma genérica para la provisión del servicio streaming de video siguiendo los mecanismos del plano de control de IMS. El sistema cliente propuesto para la plataforma es capaz de realizar las labores de control (establecimiento, mantenimiento y liberación de sesión).
5. Al momento de diseñar una solución tecnológica es de suma importancia no realizar ninguna suposición basada en la teoría. Toda decisión tomada en el diseño debe ser apoyada por bases y pruebas concretas probadas tanto en la teoría como en la práctica.

6. SIP es un protocolo de establecimiento de sesión peer-to-peer, es decir, entre agentes iguales, en cambio un servicio de streaming de video debe utilizar un paradigma cliente-servidor.
7. El uso de los protocolos estándar en los despliegues actuales de IPTV o de vídeo sobre Internet permitirá utilizar gran parte de la infraestructura existente.
8. Una desventaja de la solución IMS es la inmadurez de la tecnología especialmente en el área de terminales y aplicaciones, que puede tener el efecto de aumentar los costos de entrada. Sin embargo, un operador podría lanzar los servicios más sencillos o populares en una primera fase y así diferenciarse en el mercado y a raíz de mejoras en las terminales y las aplicaciones, integrar más adelante nuevos servicios en su infraestructura existente.

Prototipo IMS

1. Las experiencias aportadas por los trabajos previos de [6] ayudo de gran manera el trabajo de investigación al momento de definir los componentes del prototipo. Además este trabajo de tesis logra construir sobre las bases que proporcionan [12] y [13].
2. Se logró diseñar e implementar con éxito el prototipo IMS de acuerdo a los objetivos planteados. El sistema obtenido constituye una plataforma base para la integración de futuros servicios de usuario final y representa una herramienta de apoyo para el estudio práctico de la arquitectura IMS.
3. EL prototipo IMS logra de una manera sencilla cumplir con las funcionalidades de control de sesión (establecimiento, mantenimiento y liberación) mediante la utilización del protocolo SIP (Session Initiation Protocol),
4. Los Application Server (AS) poseen la habilidad de incorporarse a IMS con la ayuda del protocolo SIP. Los AS proporcionan los bloques básicos que ofrecen servicios de telecomunicaciones dentro de IMS y permiten cubrir el plano de servicio con una gama de nuevas herramientas (como mensajería, voz, video y presencia).

Servidor streaming Video

1. La transmisión multimedia dentro de un ambiente corporativo puede alcanzar niveles de audiencia ilimitadas gracias al método de transmisión Multicast.
2. La capacidad de enviar y de recibir IP Multicast es dependiente de la red. Los routers y switchs de la red deben soportar los paquetes Multicast.
3. Con el método Multicast el consumo de ancho de banda en una red lan es equivalente al de un único usuario.
4. Al momento de realizar un diseño de un Sistema Streaming a gran escala se debe considerar los conceptos de escalabilidad, tolerancia a fallos, costo, y balanceo de carga.
5. Darwin Streaming Server (DSS) es una solución de software para el streaming de video, desarrollada bajo la licencia pública general GNU (GPL). Está diseñado para generar flujos de videos sobre redes de alto ancho de banda. Soporta el transporte de datos a través de los protocolos IPv4 además de poder generar tráfico unicast o Multicast.
6. VLC media player, es un reproductor multimedia avanzado del proyecto VideoLAN, es software libre y código abierto que es distribuido bajo licencia GPL. Soporta multitud de códecs de audio y video, así como diferentes tipos de archivos, además soporta los formatos de DVD, VCD y varios protocolos streaming. También puede ser utilizado como servidor en unicast o Multicast, en IPv4 o IPv6, en redes de banda ancha.,
7. VLC es una herramienta que permite realizar un análisis académico respecto del funcionamiento de un servidor streaming de video, video bajo demanda, sin embargo no reúne la suficiente fortaleza como para hacer un uso comercial del mismo. El número de peticiones soportadas por el servidor es insuficiente para un uso masivo como servidor de VoD.

UCT IMS Client

1. La versión UCT IMS Client 1.012 tiene perfecta compatibilidad con el sistema operativo Ubuntu Linux 8.04. Existen nuevas versiones del cliente UCT IMS, sin embargo existen incompatibilidades entre VLC y el sistema operativo Ubuntu Linux 10.04.
2. El cliente UCT IMS Client 1.012 presentó 100% funcional para las pruebas de video bajo demanda.

Software Libre

1. Se destaca el uso de software Open Source para el desarrollo de este proyecto de investigación. Se utilizaron programas que exclusivamente no requieren pago de licencias y que se encuentran disponibles en el Internet. En este contexto toma mucha fuerza el concepto de desarrollo comunitario como lo es la iniciativa Open IMS Core, implementación pública y abierta para el testeo de la arquitectura IMS. Las aplicaciones abiertas desarrolladas por Fraunhofer Institute for Open Communication Systems (FOKUS), University of Cape Town (UCT), Darwin Streaming Server (DSS), VideoLan (VLC) fue fundamental para la realización de este trabajo.
2. El software libre se ha ido consolidando con el tiempo como alternativa, técnicamente viable y económicamente sostenible, respecto al software de licencia propietaria. Su proceso de corrección de errores es muy dinámico, ya los usuarios del programa en cualquier parte del mundo, gracias a que disponen del código fuente del programa, pueden detectar los posibles errores, corregirlos y contribuir así con las mejoras. Finalmente respecto a la seguridad y privacidad, el código fuente se puede auditar y verificar, de esta manera hay más dificultad para introducir código malicioso, entre otros problemas de seguridad.

6.2 Recomendaciones y trabajo futuro.

Esta Sección hace un listado de las recomendaciones para futuras líneas de trabajo para la maqueta IMS.

1. Migrar todos los componentes e infraestructura del prototipo IMS a una maqueta IPv6.
2. Desarrollar un cliente UCT IMS Client para la plataforma Windows y Linux. Una reproducción del cliente no aporta nada nuevo, por lo que se propone que el cliente sea capaz de correr sobre dispositivos celulares.
3. Una vez implementada en e prototipo IMS el S-CSCF y el HSS podemos definir una vía de desarrollo vertical que estaría encaminada a la creación de otros servicios servicios para ofrecer al abonado (Application Servers).
4. Habilitar servicios de roaming en el portotipo IMS.

7 BIBLIOGRAFÍA

- [1] Video sobre IP, Mireya S. García, Alejandro A. Ramírez. Centro de Investigación y Desarrollo de Tecnología Digital, perteneciente al Instituto Politécnico Nacional. México.
- [2] Servicio de Vídeo sobre redes Móviles de Nueva Generación. Rosa Rodríguez, Joaquín López, Antonio Ferreras, José García. Telefónica Móviles - Investigación y Desarrollo. España.
- [3] Subsistemas Multimedia IP (IMS) en 3GPP y 3GPP2, Rodrigo Lizana M., Student Member, IEEE. Departamento de Ingeniería Eléctrica Universidad de Santiago de Chile. Julio, 2008
- [4] The 3G IP Multimedia Subsystem (IMS). Gonzalo Camarilla, Miguel García. 414 Páginas. Second Edition. 2006.
- [5] Desarrollo de una aplicación de audio conferencia basada en Multicast para escenarios de red con soporte IMS, Mario Ramos Benito, Universidad Carlos III de Madrid. Marzo 2009.
- [6] Servicios Personalizados de Multimedia ofrecidos sobre la plataforma IP Multimedia Subsystem. Alton Kenneth Macdonald. Laboratorio de Redes Avanzadas Instituto Tecnológico de México. 125 Páginas. 2009.
- [7] A study of An Open Source IP Multimedia Subsystem Test. Jing Tang, Carol Davis, Yu Cheng.
- [8] Migration of VOIP/SIP Enterprise Solution towards IMS. Lian Wu-Anders Hagelskjaer Aagaard. Agder University. 96 Páginas. 2006.
- [9] Media Security in Open IMS Core. Arnaud Van Gelder. University of Groningen. 71 Páginas. 2009.
- [10] SIP – Supported Soft Handover for the Convergence of Wireless Enterprise Networks and 3GPP IP Multimedia Subsystem. Aalborg University. 67 Páginas. 2007
- [11] OpenIMS and Interoperability with Asterisk/Sip Express VOIP Enterprise Solutions. Fei Yao – Li Zhang. Agder University. Páginas 122. 2007.
- [12] Configuration and Deployment of an IMS Test Bed. Alton Kenneth Macdonald. Laboratorio de Redes Avanzadas Instituto Tecnológico de México. 61 Páginas. 2009.
- [13] Estudio comparativo de sistemas de difusión de video afluente. Juan Miguel Colores. Centro de Investigación y Desarrollo de Tecnología Digital. 2008.

- [14] An Open IMS Testbed for exploring Wireless Service Evolution and Network Architecture Evolution towards SAE and LTE. Niklas Blum, Fabricio Carvalho de Gouveia, Thomas Magedanz. Fraunhofer Institute for Open Communication Systems (FOKUS). Berlin-Germany.
- [15] Evolución de redes 3G. Universidad Rey Juan Carlos.
[Online]http://docencia.etsit.urjc.es/moodle/file.php/34/Teoria/Cursos_antiores/curso0708/B2.4_Evolucion3G_08.pdf
- [16] .FOKUS. Open IMS Core Installation Guide.
[Online]http://www.openimscore.org/installation_guide
[Online]http://uctimsclient.berlios.de/uctiptv_advanced_howto.html
- [17] Forum IP Multimedia Subsystem France. [Online]
<http://www.forum-ims.org/>
- [18] Networkworld. [Online] [http:// www.networkworld.es/articulo-178173](http://www.networkworld.es/articulo-178173)
- [19] The reference website for MPEG. [Online] <http://www.mpeg.org/>

8 ANEXOS

8.1 Anexo A. Protocolos Multicast

El envío eficiente de datos a múltiples usuarios de manera simultánea usando Multicast de red se basa en el soporte específico que la red ofrece a dicho envío. En primer lugar existe un soporte por parte de los equipos de red, como los routers de Internet, que han de contar con las capacidades de proceso y memoria suficientes. En segundo lugar se ha de poder establecer sobre la red un estado de encaminamiento Multicast para llevar a cabo la construcción de los árboles de distribución y el envío de datos en sesiones Multicast.

Para satisfacer el segundo punto hay protocolos específicos que operan a diversos niveles en el modelo TCP/IP para realizar las funciones requeridas para la distribución Multicast. Estas funciones incluyen la función principal de establecimiento de un estado de encaminamiento Multicast, así como funciones de soporte. Entre las funciones auxiliares se encuentran la distribución de información topológica de las redes, descubrimiento de fuentes de datos Multicast asociadas a un grupo, inclusión de mecanismos de redundancia, seguridad y control de admisión, interacción con los host receptores (miembros o potenciales miembros Multicast), etc.

Los requisitos de red fundamentales en cuanto a eficiencia, como son el ancho de banda y el retardo, son determinantes en la implementación y escalabilidad de las soluciones basadas en Multicast de red.

A continuación se detalla la descripción de algunos protocolos Multicast y se analiza su funcionalidad y validez para la transmisión de contenidos de video.

Protocolo IP Multicast

El protocolo IP Multicast permite la transmisión de datagramas IP a un conjunto de anfitriones que forman parte de grupos de multidifusión. Se emplea básicamente el mismo concepto de entrega con el mejor esfuerzo, como en el caso de los datagramas IP Unicast. La pertenencia a un grupo de protocolo IP Multicast es un proceso dinámico. Un cliente puede unirse o abandonar un grupo en cualquier momento. Cada grupo IP Multicast tiene una dirección de multidifusión única (clase D). Así como en el caso de las direcciones IPv4 Clase A, B y C algunas direcciones de multidifusión IP Clase D son asignadas por la

Autoridad de Internet (IANA) y corresponden a grupos que siempre existen aun cuando actualmente no tengan miembros.

Otras direcciones de multidifusión están disponibles para usos temporales y corresponden a grupos transitorios de multidifusión que se crean cuando son necesarios y se desechan cuando el número de miembros llega a 0. Los mensajes IGMP son transmitidos dentro de datagramas IP, cuyo campo de protocolo de capa superior se establece con un valor de dos, y el campo de tiempo de vida con un valor de cero. A su vez, dicho paquete IP es encapsulado en tramas ethernet, en donde la dirección MAC destino contiene los 23 bits menos significativos de la dirección de grupo añadidos, al patrón hexadecimal 0x0100.5EXX.XXXX.

El protocolo IP Multicast puede utilizarse en una sola red LAN o a través de una red de redes. En este último caso, enrutadores de multidifusión especiales envían los datagramas de multidifusión. Para la realización de videoconferencias punto-multipunto bajo el estándar UIT H.323 , es posible utilizar el protocolo IP Multicast, y en el caso particular de redes LAN ethernet conmutadas en las que los equipos activos presentan funciones básicas correspondientes a las dos primeras capas del modelo OSI.

Existen tres protocolos de membresía de grupos IP Multicast (IGMP Snooping, GMRP y CGMP) que se pueden utilizar para el reconocimiento de los grupos de multidifusión, trabajando conjuntamente con el protocolo IGMP. Este último se emplea tanto por parte de los miembros de grupos así como por parte de un enrutador denominado consultor, para gestionar el proceso de adherencia y desincorporación.

Este proceso se realiza cuando un terminal de videoconferencia H.323 UIT manifiesta su intención de participar en una sesión previamente establecida entre dos terminales, e involucra tanto al conmutador conectado directamente, así como al enrutador consultor, ocasionando una degradación de la calidad de servicio de la sesión. Tal efecto dependerá de las características del protocolo de membresía de grupo utilizado.

Protocolo de gestión de grupos internet (IGMP)

IGMP v1 (Versión 1)

La primera versión del protocolo IGMP define procesos de consulta-respuesta, adherencia, desincorporación a grupos y un mecanismo de supresión de reporte.

Proceso de consulta-respuesta y mecanismo de supresión: permite al enrutador consultor determinar cuáles grupos de multidifusión están activos, es decir, si hay uno o más miembros interesados en la subred. El enrutador consultor envía cada 60 segundos un mensaje de consulta a los grupos presentes en una subred. De cada uno de ellos, un cliente miembro responde con un mensaje de reporte, representando al grupo, luego que expira un contador propio inicializado en 10 segundos por defecto. Se asocia entonces un contador distinto por cada grupo al cual pertenece, y es posible que dicho cliente no represente a los otros grupos a los que pertenece necesariamente.

Proceso de adherencia y desincorporación: si un cliente desea formar parte de algún grupo, puede enviar un mensaje de adherencia no solicitada de reporte al mismo. De esta forma, el anfitrión puede enviar y recibir información al grupo. Si se realiza una sesión de streaming de video dos terminales, un terminal adicional puede manifestar su intención de participar. Lamentablemente, si un cliente desea desincorporarse de algún grupo de multidifusión no tiene forma de manifestar intencionalmente su intención si utiliza la primera versión del protocolo IGMP.

El cliente sólo debe dejar de enviar mensajes de reporte al enrutador consultor de la subred, para dejar de recibir información al grupo. Como resultado de esto, un terminal dejaría de recibir tráfico de la sesión de streaming de video. El tiempo que tarda el enrutador en desincorporar al cliente de la sesión, es de 3 minutos, por lo que el terminal todavía recibe tráfico multimedia en ese intervalo de tiempo, teniendo como consecuencia una degradación de la calidad de servicio por congestión si inmediatamente desea formar parte de otra sesión, ya que habría un aumento en el grado de utilización de la red.

IGMPv2 (Versión 2)

Los mensajes de consulta y de reporte en IGMPv2 son idénticos a los de IGMPv1 con 2 excepciones: la primera es que los mensajes de consulta se dividen en dos subtipos: generales y específicos. Los mensajes generales de

consulta realizan las mismas funciones en ambas versiones. Los mensajes específicos de consulta, van dirigidos a grupos particulares.

La segunda excepción es que los mensajes de reporte tienen diferentes tipos de código. Un nuevo campo en los mensajes de consulta permiten al enrutador consultor especificar el máximo tiempo de consulta-respuesta para disminuir la latencia de desincorporación de anfitriones a grupos. En una sesión de videoconferencia, la disminución conlleva a un decremento en la latencia de salida.

Otra de las diferencias con respecto a IGMPv1, es que en IGMPv2 el anfitrión puede enviar al enrutador consultor de la subred un mensaje de desincorporación, produciendo igualmente una disminución en la latencia de salida. Al disminuir el máximo tiempo de respuesta a la consulta (Límite superior de contador individual), aumentan las ráfagas, y los picos de congestión, degradando fuertemente la calidad de servicio; de lo contrario, si se aumenta, se incrementa la latencia de salida. Debido a que IGMPv2 define que no necesariamente debe haber mensajes de desincorporación de grupos específicos, es posible que los conmutadores ethernet no identifiquen tal desincorporación, utilizando de forma equivocada algunos protocolos de membresía de grupos de red LAN conmutada.

IGMP v3 (Versión 3)

La versión 3 del protocolo IGMP amplía los mecanismos de adherencia y desincorporación de clientes. Permite a los miembros de un grupo seleccionar las fuentes de tráfico específicas de las cuales desean recibir tráfico aún dentro del mismo grupo. Se presentan mensajes de consulta y de reporte. Los mensajes de consulta pueden ser: generales, de grupo específico y de grupo-fuente específico. Los primeros son enviados por el enrutador en la subred para aprender acerca del estado de recepción de las interfaces. Los segundos son enviados para verificar el estado de recepción de un grupo específico. Los mensajes de grupo-fuente específicos son enviados por el consultor para verificar si alguno de los clientes vecinos desea recibir paquetes enviados a un grupo particular por parte de un conjunto de fuentes.

A continuación se realiza un resumen de protocolos con la confrontación de las características funcionales y estructurales de cada protocolo con las necesidades existentes en el ámbito de distribución de medios continuos en Internet.

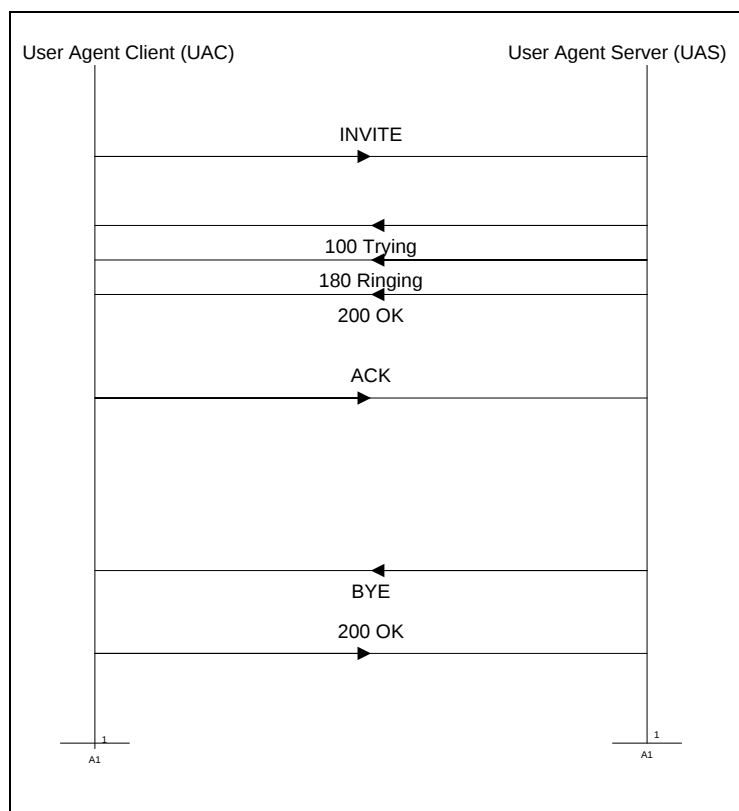
Función	Protocolo	IETF RFC	Comentarios
Establecimiento del estado de Encaminamiento Multicast	PIM-SM	4601	Protocolo de encaminamiento Multicast en el que se basa el estudio posterior, modelado y simulaciones de Multicast inter-dominio
	PIM-DM	3973	Sólo resulta útil en pequeñas redes en las que todos los hosts reciban los contenidos y se quiera reducir la información de control
	DVMRP	1075	Fue el primer protocolo de encaminamiento Multicast
	BGMP	3913	Es un protocolo específico para Multicast inter-dominio, aunque nunca se ha implementado o desplegado de forma comercial
Distribución de la información topológica	MP-BGP	4760	Permite distribuir información topológica entre múltiples dominios de red
	OSPF Multi-Topology Extensions	4915	Realiza una distribución de información topológica en el interior de un dominio de red
Descubrimiento de fuentes activas	SSM	4607	No necesita un protocolo de descubrimiento de fuentes. Las direcciones IP de éstas se configuran manual o estáticamente
	MSDP	3618	Permite intercambiar información de fuentes activas entre múltiples dominios de red
Interacción de los miembros Multicast	IGMPv3	3376	Protocolo de membresía estándar para IPv4
	MLDv2	3810	Protocolo de membresía estándar para IPv6

Tabla 12. Protocolos de encaminamiento y soporte Multicast

8.2 Anexo B. Protocolo SIP

Transacciones SIP

Una transacción SIP es una secuencia de mensajes entre dos elementos de Red. Una transacción corresponde a una petición y todas las respuestas a esa petición. En la figura 58 se muestra un ejemplo de los mensajes que pertenecen a una misma transacción dentro de una conversación SIP.



Fuente: Autor

Figura 62. Transacción SIP

SOLICITUDES SIP

Originalmente había un total de 6 SIP Request: INVITE, REGISTER, BYE, ACK, CANCEL y OPTIONS. Luego aparecieron las extensiones de SIP que se encuentran definidas en otras RFCs y que introducen 7 nuevas peticiones REFER, SUBSCRIBE, NOTIFY, MESSAGE, UPDATE, INFO y PRACK. A continuación se describe cada uno de los SIP Request.

INVITE

Se utiliza para establecer sesiones multimedia entre dos o más usuarios. Un INVITE normalmente posee un cuerpo dentro del mensaje que contiene la información multimedia del usuario que origina la llamada. Además este cuerpo también puede contener información de calidad de servicio (QoS) o información de seguridad. Una sesión se considera establecida cuando el UAC (User Agent

Client) recibe un mensaje de aceptación 200 OK por parte del servidor. En este momento se establece un diálogo entre los UAC que han acordado la sesión cuya duración expira en el momento que cualquiera de ellos decide abandonar la sesión enviando un BYE.

BYE

Este mensaje se utiliza para terminar una sesión previamente establecida. Para liberar la sesión el UAC ha de introducir en la primera línea del BYE la URI del usuario contra el que quiere terminar la sesión.

ACK

Se trata de un mensaje que se envía para asentar respuestas finales a una solicitud INVITE. Respuestas finales a otro tipo de respuestas finales nunca son asentadas. Las respuestas finales quedan definidas por los códigos 2xx, 3xx, 4xx, 5xx y 6xx.

NOTIFY

La solicitud NOTIFY es utilizada por un UAC para transmitir información sobre la ocurrencia de un cierto evento. El NOTIFY siempre se envía dentro de un diálogo cuando existe una suscripción entre el usuario y el que notifica. Dado que es un mensaje que se manda dentro de un diálogo estarán presentes las cabeceras *To*, *From*, y *Call-ID*. La respuesta al NOTIFY es un *200 OK* indicando que el mensaje ha sido recibido. Si se recibe un mensaje con el código *481 Dialog/Transaction Does Not Exist* la suscripción se termina automáticamente y no se envían más *NOTIFY*.

UPDATE

El mensaje UPDATE es un método que se utiliza para cambiar el estado de la sesión sin modificar el estado del diálogo. Posibles utilidades del UPDATE son silenciar la llamada, poner la llamada en espera y dejar de enviar el flujo de datos o realizar negociaciones de parámetros de calidad de servicio extremo a extremo antes de establecerse la sesión.

PRACK

El mensaje PRACK se utiliza para asentar la recepción de las respuestas fiables provisionales (con código 1xx). La fiabilidad de las respuestas con código 2xx, 3xx, 4xx, 5xx o 6xx se obtiene asintiendo al mandar un ACK.

En caso de tener respuestas provisionales tales como 180 Ringing, es crítico determinar el estado de la llamada y confirmar la recepción de la respuesta

provisional. Por tanto, el PRACK puede ser aplicado a todas las respuestas provisionales 1xx salvo para el 100 Trying. El PRACK lo genera el UAC cuando se recibe una respuesta provisional que contiene las cabeceras RSeq (Reliable Sequence Number) y la cabecera Supported:100rel. Entonces el UAC manda un PRACK asintiendo la respuesta provisional y añade una cabecera RACK con el número de secuencia que recibió en la respuesta temporal y el método que la generó. Además incrementa el número de secuencia CSeq.

Una vez que el PRACK llega al destinatario, este responde con un 200 OK para informar de que el mensaje ha llegado correctamente. La combinación del Call-ID, CSeq y RACK permite al UAC poder distinguir quien envía el 200 OK como respuesta al PRACK. Dentro del cuerpo de PRACK se puede añadir información relativa a la negociación de parámetros de calidad de servicio.

RESPUESTAS SIP

Una respuesta SIP es un mensaje originado por el UAS (User Agent Server) o proxy que contesta a una solicitud realizada por un UAC. Esta respuesta puede contener cabeceras adicionales que pueden proporcionar información extra o pueden ser simples asentimientos a solicitudes anteriores. Existen 5 clases de respuestas SIP. Las primeras 5 clases fueron copiadas del protocolo HTTP mientras que la sexta es propia de SIP. Las clases de respuestas se listan en la tabla 12

Clase	Descripción	Acción
1xx	Información	Indica el estado de la llamada antes de que se complete. Es la primera información que se recibe y es provisional.
2xx	Éxito	La solicitud se ha completado con éxito
3xx	Redirección	El servidor ha cambiado de localización. El cliente ha de dirigir la solicitud a otro servidor.
4xx	Error en el cliente	La solicitud ha fracasado por fallo en el cliente. El cliente puede intentar hacer otra solicitud.
5xx	Error en el servidor	La solicitud ha fracasado por fallo en el servidor. La solicitud puede ser redirigida a otro servidor
6xx	Fallo global	La solicitud no debe volverse a mandar a ningún servidor.

Tabla 13. Clases de respuesta SIP

Cada código de cada respuesta SIP no es entendida por el UAC, es decir, el UAC no entiende el código en particular, tan solo la clase.

100 TRYING

Se trata de una respuesta provisional que se propaga solo de un salto a otro y nunca se reenvía. Además nunca contiene ninguna información en el cuerpo del mensaje. Esta respuesta puede ser generada tanto por un Server (cuando por ejemplo recibe un INVITE) como por un UAC.

180 RINGING

Esta respuesta se manda para indicar que el INVITE ha sido recibido por el UAC y alertar al iniciador de la llamada. Cuando el UA acepta la llamada, se deja de enviar el RINGING y se manda un 200 OK

El cuerpo del mensaje puede llevar información de calidad de servicio, información de seguridad, puede llevar adjunto un tono de llamada o cualquier tipo de animación que mande el UAS al UAC.

183 SESSION IN PROGRESS

Esta respuesta proporciona información sobre el progreso de la llamada. Al contrario que el 100 TRYING, el 183 SESSION IN PROGRESS es un mensaje extremo a extremo dentro de un diálogo por lo que las cabeceras To y Contact han de estar presentes en el mensaje. Este mensaje se puede utilizar para permitir al UAC escuchar el tono de llamada, tono ocupado, o escuchar cualquier tipo de anuncio o mensaje de la PSTN. También este mensaje se puede utilizar para enviar repuestas en modo fiable. Cuando el UAC inicia la llamada enviando el INVITE, el destino de la llamada puede contestarle con un 183 SESSION IN PROGRESS para indicarle sobre el progreso de la llamada. Esta respuesta actualizará el valor de la cabecera RSeq incrementando el valor de este campo si fuera necesario. Además añadirá la cabecera Require que incluirá el parámetro 100rel para denotar que la respuesta es fiable. El destinatario de la llamada no enviará más repuestas provisionales en este diálogo hasta que no reciba el PRACK de la parte que inició la llamada.

El 183 SESSION IN PROGRESS puede contener un cuerpo SDP en el mensaje con parámetros de calidad de servicio que se negocian durante el establecimiento de la llamada.

200 OK

El uso de este mensaje tiene dos acepciones. Cuando se envía para aceptar una sesión (aceptar el INVITE) lleva un cuerpo en el mensaje con información sobre las capacidades de la parte llamada. Cuando se utiliza para dar respuesta a otras solicitudes indica que la solicitud se recibió y ha sido completada.

8.3 Anexo C. Atributos del Protocolo UDP

Los atributos sirven para extender tanto la descripción de sesión como las descripciones multimedia dentro de la carga SDP. Los atributos pueden ser definidos para ser utilizados tanto a nivel de sesión como a nivel multimedia o ambos. Por tanto, se pueden encontrar los atributos en dos formatos: los de propiedad que se muestran de la forma **a=<atributo>** para indicar si se soporta una determinada propiedad, o los de valor, que extienden a los anteriores de la forma **a=<atributo>:<valor>**.

Dentro del primer grupo podemos encontrar los que definen la dirección del flujo multimedia que se pretende establecer en una descripción multimedia y que son cuatro:

a=sendonly: sólo se desea enviar flujo multimedia

a=recvonly: solo se desea recibir flujo multimedia

a=sendrecv: se desea enviar y recibir flujo multimedia

a=inactive: no se desea ni enviar ni recibir flujo multimedia

Para controlar la calidad de servicio tanto en el lado local como en el otro extremo se utilizan los atributos con las precondiciones. En ellos se indican el estado actual de la calidad de servicio en ambos extremos y la que se desea para que se establezca la sesión multimedia. A continuación se muestra un ejemplo con posibles combinaciones de precondiciones tanto en el extremo local como en remoto:

a=curr:qos local none

a=curr:qos remote none

Estas dos líneas indican que actualmente (curr) no hay calidad de servicio garantizada (none) tanto en local como en remoto

a=des:qos mandatory local send

a=des:qos none remote sendrecv

Este par de líneas manifiestan que se desea (des) calidad de servicio obligatoria para mandar en local (qos mandatory local send) y todavía sin especificar en remoto (qos none remote sendrecv).

a=conf:qos remote sendrecv

Esta última línea expresa que el extremo llamado (remote) deberá mandar un mensaje de confirmación (conf) en el momento que los recursos (qos) hayan sido reservados en ambas direcciones de envío y recepción (sendrecv).

Dentro de los atributos de valor también tenemos aquellos que mapean un determinado número con un formato de codificación (a=rtpmap:). Por ejemplo tenemos:

```
a=rtpmap: 0 PCMU
```

En este caso se produce un mapeo entre el payload 0 con la codificación PCMU.

La línea media contiene información sobre el flujo multimedia que se va a transmitir. La línea tiene el siguiente formato:

```
m=<media> <puerto> <transporte> <lista_de_formatos>
```

- El parámetro media puede tener varios valores dependiendo del flujo a transmitir: audio, video, text, application y message.
- El parámetro puerto indica el puerto por el que se envía el flujo multimedia.
- El campo transporte designa el protocolo de transporte sobre el que se va a enviar el flujo multimedia. Puede tomar distintos valores: UDP cuando no especifica ningún protocolo en concreto, RTP/AVP cuando el protocolo es RTP para el envío de audio y/o video o RTP/SAVP cuando el protocolo es RTP para el envío de audio y/o video en modo seguro.
- Finalmente tenemos el listado de formatos que muestran todos los formatos en que el flujo multimedia se puede codificar. Es una cadena de caracteres numéricos separados por un espacio en el que cada número se corresponde a un formato de codificación.

Una posible línea m podría ser la siguiente:

```
m=audio 3458 RTP/AVP 0 96 97 98
```

Esta línea especifica que se puede enviar información de audio al puerto 3458 y que los formatos soportados para codificar la voz son 0, 96, 97 y 98.

8.4 Anexo D. Configuración IGMP Snooping en switch Enterasys.

Un switch de L2 sin IGMP Snooping trata las tramas de Multicast como si fuera de broadcast, El switch se satura con información innecesaria bajando notablemente las prestaciones de la comunicación de cada puerto.

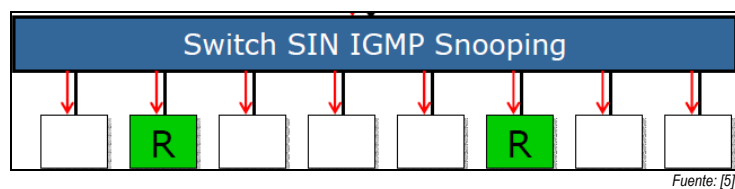


Figura 63. Switch SIN IGMP Snooping

Un switch con IGMP Snooping, escucha las peticiones de registro y abandono de grupos de Multicast:

1. Envía tramas solo por los puertos en lo que hay algún equipo que ha lanzado tramas de participación en grupos Multicast,
2. Aunque exista muchos equipos enviando hacia el router tramas de participación IGMP (de petición de inicio o de respuesta a los queries periódicos que lanza el router para comprobar la existencia de participantes en el grupo Multicast) el switch solo envía una de ellas hacia el router, ahorrando ancho de banda.

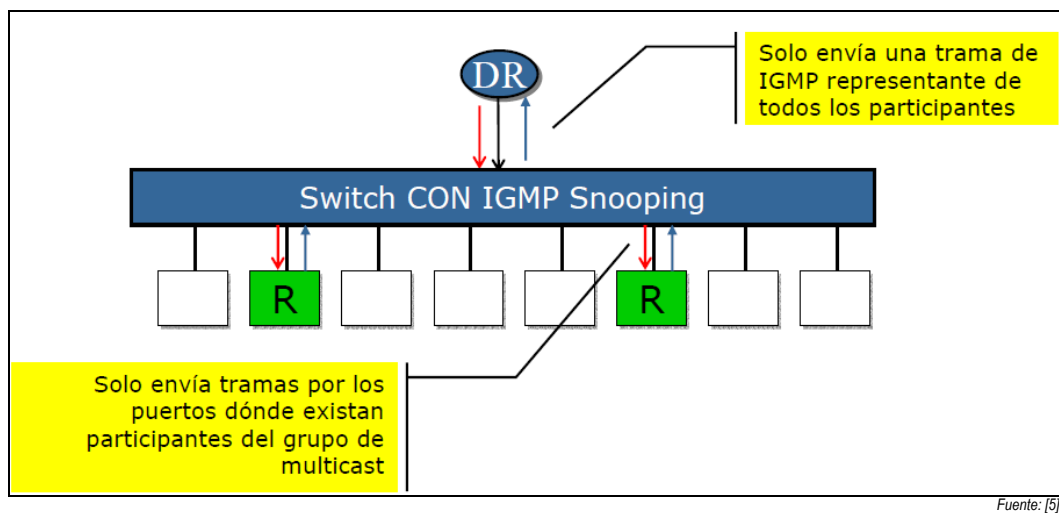


Figura 64. Switch con IGMP Snooping

En el switch Enterasys utilizado en el proyecto, es necesario habilitar el servicio IGMP Snooping :

```
show igmpsnooping
set igmpsnooping adminmode enable
set igmpsnooping interfacemode ge.1.1-24 enable
show igmpsnooping
```

8.5 Anexo E. WireShark

Wireshark es un capturador/analizador de paquetes de red (llamado a veces, sniffer o esnifer). Wireshark permite ver, aun nivel bajo y detallado, qué está pasando en la red. Además es gratuito, open source, y multiplataforma. Posee una interfaz gráfica y muchas opciones de organización y filtrado de información. Así, permite ver todo el tráfico que pasa a través de una red

El instalador y los archivos binarios de Wireshark pueden ser descargados en <http://www.wireshark.org/download.html>. WireShark soporta múltiples plataforma entre ellas UNIX, LINUX y Windows, a continuación se describe la instalación en Linux.

Instalación Linux Ubuntu

```
sudo apt-get install wireshark
```

Panel de paquetes capturados

Cada línea corresponde a un paquete capturado al seleccionar una de estas, ciertos detalles son desplegados en el resto de los paneles (Detalles y bytes). Y las columnas muestran datos del paquete capturado, Wireshark dispone de una gran cantidad de detalles que pueden agregarse en estas columnas desde el menú Edit->Preferences, por defecto se tienen:

- No.: posición del paquete en la captura.
- Time: muestra el Timestamp del paquete. Su formato puede ser modificado desde el menú View->Time Display Format.
- Source: dirección origen del paquete.
- Destination: dirección destino del paquete.
- Protocol: nombre del protocolo del paquete.
- Info: información adicional del contenido del paquete.

Panel para detalles de paquetes capturados

Contiene el protocolo y los campos correspondientes del paquete previamente seleccionado en el panel de paquetes capturados. Seleccionando una de estas líneas con el botón secundario del Mouse se tiene opciones para ser aplicadas según las necesidades.

Panel de paquetes capturados en Bytes

En este panel se despliega el contenido del paquete en formato hexadecimal.

```
0000 00 0f 34 cd 72 c3 00 16 35 74 23 59 08 00 45 00  .4.r... 5t#Y..E.
0010 00 3c 55 e5 00 00 80 01 91 66 ac 11 01 51 ac 11  .<U.....f...Q..
0020 fa 01 08 00 e3 5b 02 00 68 00 61 62 63 64 65 66  .....[. h.abcdef
0030 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76  ghijklmn opqrstuv
0040 77 61 62 63 64 65 66 67 68 69                    wabcdefg hi
```

De izquierda a derecha se muestra el offset del paquete seguidamente se muestra la data del paquete y finalmente se muestra la información en caracteres ASCII si aplica o “.” (Sin comillas) en caso contrario.

Captura de Paquetes

Una de las principales funciones de WireShark es capturar paquetes con la finalidad de que los administradores puedan hacer uso de estos y realizar el análisis necesario para tener una red segura y estable. Como requisito para el proceso de capturar datos es ser administrador y contar con estos privilegios y es necesario identificar exactamente la interfaz que se quiere analizar.

8.6 Anexo F. Monitor de recursos servidor. Conky

Conky es una herramienta de control y motorización del sistema, es altamente configurable y personalizable que nos muestra información sobre el hardware, el software, del estado de ejecución de programas. Es muy ligero y no usa casi recursos del sistema ni ralentiza el rendimiento del Servidor.

Instalación

Conky está en los repositorios oficiales de Ubuntu así que para instalarlo tan solo tendremos que ejecutar en el terminal:

```
$ sudo aptitude install conky
```

8.7 Anexo G. Control de ancho de banda. Wondershaper

El mejor limitador de ancho de banda para Linux es Wondershaper. Funciona de una manera muy sencilla, abrimos una Terminal e instalamos el programa:

```
$sudo apt-get install wondershaper
```

Luego con el comando `$ifconfig` vemos cual es la tarjeta de red que estamos utilizando (eth0, eth1, wlan0 .. etc). Una vez que tenemos identificada nuestra salida, en la Terminal:

```
$sudo wondershaper ethX velocidad_bajada velocidad_subida
```

```
$sudo wondershaper wlan0 300 60 (esto pone un limite de 30 Kb de subida y 6 de bajada).
```

Para eliminar el límite, escribimos:

```
$sudo wondershaper wlan0 clear
```