

**UNIVERSIDAD DE PANAMÁ
VICERRECTORÍA DE INVESTIGACIÓN Y POSTGRADO
FACULTAD DE INFORMÁTICA ELECTRÓNICA Y COMUNICACIÓN
PROGRAMA DE FORTALECIMIENTO A LOS POSTGRADOS NACIONALES
BAJO LA RESPONSABILIDAD DE LA DIRECCIÓN DE GESTIÓN DE CIENCIA
Y TECNOLOGÍA DE SENACYT**

***OPTIMIZACIÓN DE UNA RED DE MALLA (MESH) INALÁMBRICAS PARA
TRÁFICO MULTIMEDIA MEDIANTE CALIDAD DE SERVICIO***

AIDELÉN DANAIS CHUNG CEDEÑO

**TESIS PRESENTADA COMO UNO DE LOS REQUISITOS PARA OBTENER
EL GRADO DE MASTER EN CIENCIAS DE INGENIERÍA DE SISTEMAS DE
COMUNICACIONES CON ÉNFASIS EN REDES DE DATOS**

PANAMÁ, REPUBLICA DE PANAMÁ

2014

DEDICATORIA

Principalmente a Dios por permitirme alcanzar todas mis metas y brindarme su protección

A mi familia, padres y abuelos por ofrecer su amor seguridad, paciencia y apoyo incondicional a mis hermanos Maydeth y Víctor por brindarme su cariño

Mi hija Aidelis Evangeline Cedeño Chung, que es la razón de mi vida.

AGRADECIMIENTO

Deseo agradecer ante todo a Dios por permitirme alcanzar una etapa de mi vida profesional y por estar siempre conmigo en los momentos más importantes de mi vida.

Un agradecimiento especial a los profesores de la Maestría en Ciencias de Ingeniería en Comunicaciones con Énfasis en Redes de Datos por dedicar su tiempo en explicar los conceptos importantes de las telecomunicaciones. Agradecer al Dr. Iván Armuelles por su apoyo, asesoría y consejos durante la realización de este proyecto de tesis. Además, al Dr. Tomás Robles por recibirme en la Universidad Politécnica de Madrid (UPM) para realizar mi pasantía.

A la Secretaría Nacional de Ciencias, Tecnología e Innovación por el cofinanciamiento que ofrece el Programa de Maestría.

Al equipo de compañeros del programa de Maestría en Ciencias con Énfasis en Redes de Datos. Especialmente a Grace Gonzáles, Joaquín Chung, Edel Ortega por su apoyo y amistad.

A la Escuela de Electrónica y Comunicación por brindarme la consulta necesaria para la realización de este trabajo. Agradecemos al Centro de Investigación en Tecnologías de la Información y Comunicación (CITIC) de la Facultad de Informática y Comunicación de la Universidad de Panamá, por el uso de los equipos y sus instalaciones para iniciar y culminar el desarrollo de las investigaciones.

Agradezco a mi familia y a mis padres por todo su apoyo, consejos, oraciones y amor brindado.

ÍNDICE

RESUMEN	1
CAPÍTULO I INTRODUCCIÓN	4
1 ANTECEDENTES	4
1 1 DESCRIPCIÓN DEL PROBLEMA Y MOTIVACIÓN	10
1 2 OBJETIVOS	10
1 3 METODOLOGÍA	11
1 4 TRABAJOS RELACIONADOS	12
CAPÍTULO II FUNDAMENTOS TEÓRICOS	15
2 REDES INALAMBRICAS	15
2 1 1 REDES MESH INALÁMBRICAS	15
2 1 2 ARQUITECTURA DE LAS REDES MESH INALAMBRICAS	16
2 1 3 ENRUTAMIENTO EN 802 11S	22
2 1 4 SERVICIOS MULTIMEDIA	26
2 1 4 1 VoIP	27
2 1 4 2 Principales protocolos de VoIP	27
2 1 4 2 1 Protocolo de señalización de Llamadas	28
2 1 4 2 2 Protocolos de control de dispositivos	29
2 1 4 2 3 Otros protocolos de soporte para VoIP	29
2 1 4 3 Codec en Voip	30
2 1 4 4 Componentes de una red VoIP	31
2 1 5 Video Streaming	32
2 1 5 1 Contenedor de formato	34

2 1 6	Videoconferencia	37
2 1 6 1	Aplicaciones de Videoconferencia	37
2 1 7	CALIDAD DE SERVICIO	37
2 1 8	IntServ (Integrated Services)	40
2 1 9	DiffServ (Differentiated Services)	41
2 1 10	Arquitectura QoS en 802 11	42
2 1 10 1	Descripción del funcionamiento de la capa de acceso al medio del IEEE	
802 11		42
2 1 10 2	Descripción del Soporte de QoS segun el estándar IEEE 802 11e	49
CAPÍTULO III DISEÑO E IMPLEMENTACIÓN		56
3	DESCRIPCIÓN DE LA IMPLEMENTACIÓN	56
3 1	REQUISITOS BÁSICOS DEL DISEÑO	57
3 1 1	Economía y flexibilidad	57
3 1 2	Fiabilidad y versatilidad	58
3 1 3	Arquitecturas abiertas basadas en estándares	58
3 2	COMPONENTES DEL DISEÑO	59
3 3	IMPLEMENTACIÓN DE LA RED	62
CAPÍTULO IV ANALISIS DE LOS RESULTADOS		71
4	ANALISIS DE LOS RESULTADOS OBTENIDOS	71
4 1	CARACTERIZACIÓN DEL CANAL	71
4 1 1	Caracterización del canal mediante la inyección de tráfico TCP	72
4 1 2	Caracterización del canal mediante la inyección de tráfico UDP	73
4 2	Evaluación del mecanismo de soporte QoS en IEEE 802 11s	76

4 3	Evaluación de la capacidad de tratar conexiones de Video con soporte de QoS	79
4 3 1	Evaluación de la configuración de Video Streaming con QoS	80
4 3 2	Evaluación de la configuración de Video conferencia con QoS	84
4 4	Evaluación de la configuración de Voz sobre IP con QoS	90
4 4 1	Evaluación del Soporte de QoS para una llamada real de VoIP	90
4 4 2	Evaluación del Soporte de QoS para multiples conexiones de VoIP	94
CAPÍTULO V CONCLUSIONES Y TRABAJOS FUTUROS		99
5 1	CONCLUSIONES	99
5 2	TRABAJOS FUTUROS	104
BIBLIOGRAFÍA		105
ANEXO		109

INDICE DE CUADROS

CUADRO I	IEEE 802 11 Y SUS ENMIENDAS	7
CUADRO II	CODEC DE VOIP	31
CUADRO III	CONTENEDORES DE FORMATO	35
CUADRO IV	CONTINUACIÓN CONTENEDORES DE FORMATO	36
CUADRO V	RELACIÓN ENTRE LAS PRIORIDADES DE USUARIO Y LAS CATEGORIAS DE ACCESO IEEE 802 11	51
CUADRO VI	COMPONENTES DE SOFTWARE	64
CUADRO VII	CANTIDAD DE MUESTRAS CON TRAFICO TCP	72
CUADRO VIII	MUESTRAS CON EL TRÁFICO UDP	74
CUADRO IX	MAXIMO Y MÍNIMO CANAL	75
CUADRO X	EVALUACIÓN CUANTITATIVA DEL TRÁFICO UDP EN EL CANAL DE IEEE 802 11S SIN SOPORTE DE QOS	77
CUADRO XI	EVALUACIÓN CUANTITATIVA DEL TRAFICO UDP EN EL CANAL DE IEEE 802 11S SIN SOPORTE DE QOS DESVIACIÓN ESTANDAR	77
CUADRO XII	EVALUACIÓN CUANTITATIVA DEL TRÁFICO UDP EN EL CANAL DE IEEE 802 11S CON SOPORTE DE QOS	78
CUADRO XIII	EVALUACIÓN CUANTITATIVA DEL TRAFICO UDP EN EL CANAL DE IEEE 802 11S CON SOPORTE DE QOS DESVIACIÓN ESTANDAR	78
CUADRO XIV	SOPORTE DE QoS PARA VIDEO STREAMING	82
CUADRO XV	CUADRO PRUEBAS CON SERVIDOR DE VIDEO RED5	89
CUADRO XVI	MEDICIÓN DE CALIDAD DE LLAMADAS SIN QOS	92

INDICE DE FIGURAS

Fig 1	Extended Service Set (Carrano et al 2011)	5
Fig 2	Independent Basic Service Set(Carrano et al 2011)	6
Fig 3	Arquitectura de una Red WMN (Wireless Mesh Network)	18
Fig 4	WMN de Infraestructura /Backbone	19
Fig 5	WMN de Clientes	20
Fig 6	WMN Híbrida	21
Fig 7	Red VoIP	32
Fig 8	Cuatro colas para las categorías de las Acceso en IEEE 802 11e	53
Fig 9	Arquitectura general de soporte de QoS en IEEE 802 11	54
Fig 10	Estructura básica de la red de prueba de evaluación de QoS en Redes Mesh IEEE 802 11s	57
Fig 11	Modelo de Referencia de Red Mesh Inalámbrica	60
Fig 12	Red de Prueba de Soporte de QoS entre nodos de la Red Mesh	65
Fig 13	Implementación de la maqueta	66
Fig 14	Configuración del router a través de línea de comando	67
Fig 15	Configuración del router a través de interfaz Web	67
Fig 16	Activación de QoS en el router por línea de comando	68
Fig 17	Activación de QoS en el router por interfaz web	69
Fig 18	Canal total ocupado	73
Fig 19	Ancho de banda Máximo y Mínimo que puede ocupar el canal	75

CUADRO XVII	MEDICIÓN DE CALIDAD DE LLAMADAS CON QOS	93
CUADRO XVIII	DESCRIPCIÓN DE LAS CARACTERÍSTICAS DE CODECS DE VOZ	94
CUADRO XIX	EVALUACIÓN DEL SOPORTE DE QOS PARA MULTIPLES CONEXIONES DE VOIP SIN MECANISMO DE QOS	96
CUADRO XX	EVALUACIÓN DEL SOPORTE DE QOS PARA MULTIPLES CONEXIONES DE VOIP SIN MECANISMO DE QOS DESVIACION ESTANDAR	96
CUADRO XXI	EVALUACIÓN DEL SOPORTE DE QOS PARA MULTIPLES CONEXIONES DE VOIP CON MECANISMO DE QOS	97
CUADRO XXII	EVALUACIÓN DEL SOPORTE DE QOS PARA MULTIPLES CONEXIONES DE VOIP CON MECANISMO DE QOS DESVIACIÓN ESTANDAR	97
CUADRO XXIII	REQUISITO DE QoS PARA VoIP	98

Fig 20 Gráfica comparativa de la evaluación cuantitativa del tráfico UDP en el canal de IEEE 802.11s con y sin soporte de QoS 79

Fig 21 Comparativa de la evaluación del soporte de QoS para múltiples conexiones de VoIP con y sin mecanismo de QoS 98

LISTADO DE ACRÓNIMOS

AODV Ad hoc On Demand Distance Vector

AP Access Point

BSS Basic Service Set

CITIC Centro de Investigaciones de Tecnologías de Información y Comunicaciones

DHCP Dynamic Host Configuration Protocol

DiffServ Differentiated Services

ESS Extended Service Set

ETT Expected Transmission Time

ETX Expected Transmission Count

GISAI Grupo de Ingeniería de Sistemas Avanzados de Internet

HWMP Hybrid Wireless Mesh Protocol

IBSS Independent Basic Service Set

ICMP Internet Control Message Protocol

IEEE Institute of Electrical and Electronics Engineers

IETF Internet Engineering Task Force

IntServ Integrated Services

IP Internet Protocol

ITU International Telecommunications Union

MANET Mobile Ad Hoc Network

Mesh AP Mesh Access Point

Mesh STA Mesh Station

MP Mesh Point

MPP Mesh Portal Point

OLSR Optimized Link State Routing

PSTN Public Switched Telephone Network

QoS Quality of Service

RTCP RTP Control Protocol

RTP Real Time Protocol

SIP Session Initiation Protocol

STA Client or Station

TCP Transmission Control Protocol

UDP User Datagram Protocol

UPM Universidad Politécnica de Madrid

VoIP Voice Over Internet Protocol

WLAN Wireless LAN

WMN Wireless Mesh Networks

RESUMEN

Las Redes Mesh Inalámbricas (WMN) son redes de datos con topología de red de malla formadas por nodos que establecen y mantienen la conectividad de red automáticamente éstas surgieron como una tecnología capaz de brindar conectividad de banda ancha segura y de bajo costo Proporcionan una solución barata, rápida y eficaz para las redes inalámbricas de datos en zonas urbanas suburbanas y en los entornos rurales Son redes inalámbricas auto-organizadas auto-configurables y son fácilmente adaptables a diferentes necesidades de tráfico y los cambios de la red Otra de las ventajas del uso de WMN es su capacidad de reconstrucción de rutas al haber pérdida de enlace su escalabilidad y el manejo de múltiples interfaces de radio y antenas inteligentes Esta tecnología presenta aplicaciones interesantes por ejemplo en redes domésticas de banda ancha, redes comunitarias, de vecindario la automatización de edificios redes corporativas video vigilancia, para casos de emergencia y desastre

Para que las Redes Mesh Inalámbricas puedan trabajar se requiere de un protocolo de enrutamiento Utilizaremos el protocolo que propone el estándar para redes mesh (IEEE 802 11s) éste propone un protocolo de selección de rutas basado en direcciones de capa de enlace de datos (direcciones MAC) Este protocolo se conoce como HWMP (Hybrid Wireless Mesh Protocol) Es un protocolo híbrido basado en AODV (Ad hoc On Demand Distance Vector) AODV permite el enrutamiento dinámico autoarranque y multi salto entre todos los nodos móviles que participan en la red y utiliza la métrica de tiempo en el aire (air time-metric) que tiene en cuenta la velocidad de transmisión de datos, la sobrecarga y tasa de error en la trama.

La calidad de servicio (QoS) se refiere a la capacidad de la red para garantizar un cierto nivel de rendimiento para un flujo especialmente para el tráfico multimedia Las métricas de calidad de servicio son el rendimiento ancho de banda fiable jitter y el retraso Son requisitos básicos para un mejor tráfico en aplicaciones multimedia reales y de voz

Para la creación de redes basados en el protocolo IEEE 802 11s se puede utilizar software de código abierto en los router (encaminadores) inalámbricos y de descarga gratuita. Utilizaremos OpenWRT que es un firmware de software libre que permite personalizar completamente las aplicaciones en el router inalámbrico e implementa protocolos de enrutamiento se basa en el estándar 802 11s y tiene soporte a calidad de servicio

Un hecho importante de las Redes Mesh Inalámbricas es que proveen soporte a servicios multimedia reales Pero la transmisión de servicios multimedia reales presentan problemas de rendimiento inestable insuficiente ancho de banda, jitter y mayor pérdida de paquetes siendo difícil ofrecer garantías del servicio en una Red El propósito de este proyecto es ofrecer una solución mediante la implementación de los mecanismos de calidad de servicio en una red de prueba (testbed) inalámbrica multi saltos para las aplicaciones multimedia reales y garantizar que la transmisión de los datos sea más eficiente utilizando el estándar IEEE 802 11s con herramienta un firmware de software libre como OpenWRT

ABSTRACT

Wireless Mesh Networks (WMN) are data networks mesh network topology formed by nodes that establish and maintain the network connectivity automatically they emerged as a technology capable of providing secure broadband connectivity and low cost They provide a cheap quick and effective for wireless data networks in urban, suburban and rural environments solution They are self organized, self configurable and are easily adaptable to different traffic needs and changes in the network wireless networks Another advantage of using WMN is its ability to rebuild routes having link loss its scalability and management of multiple radio interfaces and smart antennas This technology has interesting applications for example in broadband home networking community networking and neighborhood, building automation, corporate networks video surveillance emergency and disaster

For wireless mesh networks can work requires a routing protocol We will use the proposed standard protocol for mesh networks (IEEE 802.11s) it proposes a route selection protocol based on address data link layer (MAC addresses) This protocol is known as HWMP (Hybrid Wireless Mesh Protocol) It is a hybrid protocol based on AODV (Ad hoc On Demand Distance Vector) AODV allows dynamic routing and multi hop boot among all the mobile nodes participating in the network and uses the metric of time in the air (air time metric) which takes into account the speed of data transmission, the overhead and error rate in the plot

The quality of service (QoS) refers to the ability of the network to guarantee a certain level of performance to a flow particularly for multimedia traffic The QoS metrics are performance reliable bandwidth, jitter and delay They are basic requirements for better traffic in real multimedia applications and voice

To create IEEE 802.11s based network protocol you can use open source software in the router (routers) and wireless free downloads We will use OpenWRT firmware which is free software that allows you to completely customize the applications on the wireless router and implements routing protocols is based on standard 802.11s supports quality of service

Importantly Wireless Mesh Network is providing real support multimedia services But the actual transmission of multimedia services presents problems of unstable performance insufficient bandwidth, jitter and packet loss greater being difficult to offer service guarantees in a Network The purpose of this project is to provide a solution by implementing mechanisms quality of service in a test network (testbed) for wireless multi hop real multimedia applications and ensure that the data transmission more efficient, using the IEEE 802.11s standard with tool free software firmware as OpenWRT

INTRODUCCIÓN

CAPÍTULO I

INTRODUCCIÓN

1 ANTECEDENTES

IEEE 802.11 ha sido ampliamente aceptado como el estándar para las redes de área local inalámbrica (WLAN) son soluciones en transmisión de datos de última milla para dispositivos portátiles. Las redes IEEE 802.11 se desarrollan para hacer posible la movilidad y servicios multimedia. Su principal objetivo es proveer servicios inalámbricos de banda ancha. Esto está motivado por su coste eficaz y su forma sencilla para establecer redes inalámbricas. La extensión de la cobertura WLAN sin embargo se enfrenta a algunos desafíos como la inter-conexión de redes cableadas, el desarrollo de redes multi-salto interconectando inalámbricamente los puntos de acceso con la mayor complejidad y el mayor costo de implementación en muchas situaciones. La norma 802.11 describe la capa física y la capa MAC para las frecuencias de comunicaciones inalámbricas en los rangos de 2.4GHz y 5GHz (Hertz et al. 2010).

Para una red WLAN el punto de acceso juega un papel central por la retransmisión de todo el tráfico entre los nodos participantes. Debido a estas razones necesitamos hacer que nuestro punto de acceso se conecte de manera inalámbrica a la infraestructura básica con otros puntos de acceso. IEEE 802.11 describe dos tipos distintos de redes o modos dependiendo de si hay o no la participación del nodo especial llamado Punto de Acceso (AP). La primera, se denomina modo de infraestructura, este hace referencia a la presencia de un punto de acceso que va a mediar con todas las comunicaciones entre los

nodos que están asociados a la misma. Normalmente el punto de acceso también está conectado a una red cableada a la que se va a extender el acceso a los nodos inalámbricos

(Fig 1) muestra una red local inalámbrica (WLAN) donde los puntos de acceso están conectados entre sí a través de un sistema de distribución por cable (DS) Un Extended Service Set (ESS) esta formado por dos conjuntos de servicios básicos y un sistema de distribución y un conjunto de servicios básicos independientes Un grupo de nodos conectados a un mismo punto de acceso se define como BSS (Basic Service Set) mientras que la unión de todos los puntos de acceso interconectados con un puente debido a la presencia del sistema de distribución, se llama ESS (Extended Service Set) (Carrano Magalhães, Saade & Albuquerque 2011)

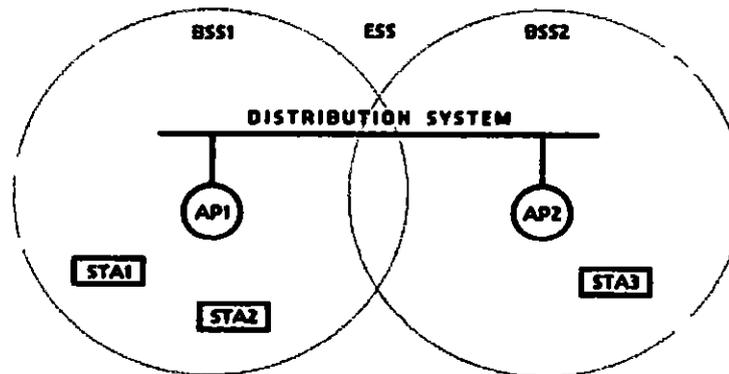


Fig 1 Extended Service Set (Carrano et al 2011)

El segundo tipo de redes IEEE 802.11 se compone sólo de estaciones (no hay puntos de acceso) que se conectan entre sí de un punto a otro este tipo de configuración

llamado modo ad hoc pero también es conocido como Independent Basic Service Set (IBSS) y se muestra en la figura (Fig 2) (Carrano et al 2011)

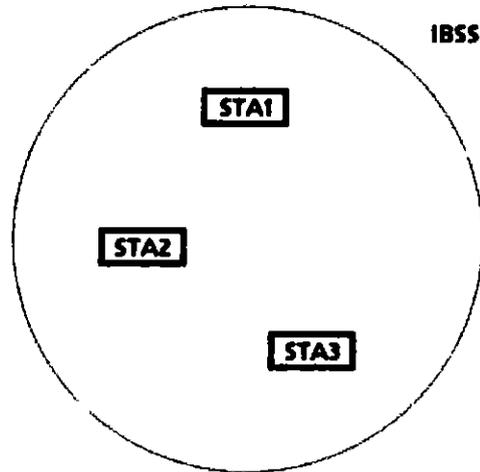


Fig 2 Independent Basic Service Set(Carrano et al 2011)

La norma IEEE 802.11 se subdivide en varios estándares (CUADRO I). El estándar IEEE 802.11 se amplió para mejorar el ancho de banda (IEEE 802.11a, b, g, el reciente proyecto n) para hacerlos más seguros (IEEE 802.11i) para mejorar el apoyo a la movilidad (proyecto r) o introducir la calidad de mecanismos de servicio (IEEE 802.11e). En esta investigación nos centramos principalmente en el estándar 802.11s (Wireless Mesh Network) y la calidad de servicio en este mismo estándar.

CUADRO I IEEE 802 11 Y SUS ENMIENDAS

Estándar o Enmienda	Descripción
802 11 1997	Original estandar (1997) que describe la capa MAC y las técnicas de modulación FHSS y DSSS (1 y 2 Mbps)
802 11a	Aprobado en 1999 presenta una nueva capa física OFDM (Orthogonal Frequency-division multiplexing)
802 11b	Approved in 1999 introduces a new physical layer HR/DSS (High Rate/Direct Sequence Spread Spectrum)
802 11g	Aprobado en 2003 introduce una nueva capa física ERP (Tarifa Extended PHY)
802 11d	Aprobado en 2001 introduce soporte de compatibilidad con regulaciones internacionales
802 11e	Aprobado en 2005 introduce la calidad de servicio (QoS)
802 11h	Aprobado en 2004 se adapta 802 11a para Regulaciones de la Unión Europea
802 11i	Aprobado en 2004 introduce nuevos mecanismos de seguridad
802 11j	Aprobado en 2004 se adapta al 802 11 norma japonés
802 11k	Permite a los conmutadores y puntos de acceso inalámbricos calcular y valorar los recursos de radiofrecuencia de los clientes de una red WLAN mejorando así su gestión
802 11 2007	Incorpora las enmiendas a b d e g h i e j para IEEE802 11 1997
802 11n (draft)	Task Group "n" (TGn) propone técnicas para lograr bandas superiores a 100 Mbps (MIMO o Multiple Input, Multiple Output es posiblemente el más popular de estas Técnicas)
802 11p	Es un estandar aprobado al IEEE 802 11 para agregar acceso inalámbrico en entornos vehiculares (de onda) Define mejoras a 802 11 necesarias para apoyar Sistemas Inteligentes de Transporte (ITS)
802 11r (draft)	Task Group "r" (TGr) trabaja en los mecanismos traspaso (handoff) en particular para los dispositivos de rápido movimiento (vehiculos por ejemplo)
802 11s (draft)	Task Group s (TGs) propone una red de malla para Dispositivos 802 11
802 11u	Task Group "u" (TGu) consiste en integrar las redes IEEE 802 11 de acceso con redes externas de forma genérica y estandarizada
802 11v	Task Group "v" (TGv) permite la configuración de los dispositivos del cliente mientras que está conectado con redes
802 11w	Task Group 'w' (TGw) está trabajando en mejorar la capa del control de acceso al medio de IEEE 802 11 para aumentar la seguridad de los protocolos de autenticación y codificación

Las redes mesh inalámbricas (WMN) se están convirtiendo en una tecnología prometedora y clave para la próxima generación de redes inalámbricas. Se espera que las redes mesh gradualmente sustituyan la funcionalidad de la infraestructura de red cableada. Proporcionar una solución barata, rápida y eficaz para las redes inalámbricas de datos en zonas urbanas, suburbanas e incluso los entornos rurales. La infraestructura de red está descentralizada y simplificada, ya que cada nodo necesita transmitir sólo hasta el siguiente nodo. Esta tecnología posee aplicaciones interesantes por ejemplo en redes domésticas de banda ancha, redes comunitarias de vecindario, la automatización de edificios, redes corporativas, video vigilancia, para casos de emergencia y desastre (Chung et al. 2014).

WMN se apoya del estándar IEEE 802.11s. En el estándar IEEE 802.11s todos los dispositivos que tienen funcionalidades de malla se denominan Estaciones de Malla (Mesh STA o MSTAs). El dispositivo puede participar en el protocolo de enrutamiento de malla y envían los datos en nombre de otros puntos de la malla de acuerdo con la propuesta de enmienda 802.11s (Carrano et al. 2011).

Para que las WMNs puedan trabajar se requiere de un protocolo de enrutamiento. Se utilizó el protocolo del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) en la norma 802.11s del 2012. éste propone un protocolo de selección de rutas basado en direcciones de capa de enlace de datos (direcciones MAC). El mismo se conoce como HWMP (Hybrid Wireless Mesh Protocol). Es un protocolo híbrido basado en AODV (Ad hoc On Demand Distance Vector). AODV (Perkins, Belding, Royer & Das 2003) permite el enrutamiento dinámico, autoarranque y multi salto entre todos los nodos.

móviles que participan en la red y utiliza la métrica de tiempo en el aire (air time metric) que tiene en cuenta la velocidad de transmisión de datos, la sobrecarga y tasa de error en la trama (Joaquin Chung et al. 2012)

Para la creación del testbed basado en IEEE 802.11s se utilizó el software de código abierto (gratuito) en los routers (encaminadores) inalámbricos que forman el núcleo de la red. Utilizamos el OpenWRT ('OpenWRT' n.d.) un firmware de código abierto basado en Linux, es una herramienta completa, ya que proporciona un sistema de ficheros totalmente modificable con la gestión de paquetes. Esto le libera de las restricciones de la selección y aplicación de configuración proporcionada por el vendedor y le permite utilizar los paquetes para personalizar un dispositivo integrado para adaptarse a cualquier aplicación. Permite personalizar las aplicaciones en el router e implementa protocolos de enrutamiento y tiene soporte a calidad de servicio. OpenWRT puede ser configurado a través de la línea de comandos o con una interfaz Web.

Para transmitir voz y video de una forma adecuada en una red mesh inalámbrica se necesita que estas redes soporten Calidad de Servicio (QoS) (Crawley, R. Nair, B. Rajagopalan, & H. Sandick, 1998) es un concepto técnico medido, expresado y comprendido en el nivel de red y de enlace.

En el presente proyecto de investigación, como trabajo de Tesis de la Maestría en Ciencias de Ingeniería de Sistemas de Comunicación con Énfasis en Redes de Datos, se plantea el diseño e implementación de WMN de bajo costo para la provisión de servicios multimedia con QoS utilizando el nuevo estándar IEEE 802.11s. Este nuevo estándar emergente del IEEE no se ha implementado de forma económica aun con el soporte de QoS.

1.1 DESCRIPCIÓN DEL PROBLEMA Y MOTIVACIÓN

Garantizar la provisión de calidad de servicio de extremo a extremo en redes inalámbricas multi saltos (Mesh, IEEE 802.11s). La calidad de servicio se refiere a la capacidad de la red para proporcionar un mejor servicio al tráfico seleccionado y así procurar que el tráfico de tiempo real de aplicaciones multimedia y de voz reciba la más alta prioridad, el mayor ancho de banda y el menor retardo.

Con esta investigación esperamos obtener los parámetros de calidad de servicio que garantizan el buen funcionamiento de las aplicaciones multimedia reales. Como resultado se podrán implementar en redes inalámbricas comunitarias y áreas rurales de difícil acceso para proporcionar servicios de voz, video y datos con buena calidad.

1.2 OBJETIVOS

Objetivo General

- Optimizar la prestación de Calidad de Servicios (QoS) en una red multi salto basada en el estándar IEEE 802.11s

Objetivo Específicos

- Implementar una red de prueba (testbed) inalámbrica multisaltos (Wireless Mesh Network) mediante el protocolo IEEE 802.11s
- Estudiar los mecanismos de provisión de calidad de servicio del estándar 802.11s
- Evaluar la capacidad de proveer calidad de servicio en la red de prueba (testbed) en aplicaciones multimedia reales

- Proponer configuraciones de los elementos de la red de prueba inalámbrica multisaltos para el soporte de QoS efectivo

1.3 METODOLOGÍA

La Metodología a utilizar en la investigación se apoya de actividades como las revisiones bibliográficas la recopilación de información, diseño de la arquitectura, configuración de los equipos pruebas análisis y resultados

- Se implementará una red de prueba mesh inalámbrica armada en el laboratorio con los recursos básicos necesarios Sobre la red de prueba se realizarán mediciones para determinar el desempeño y la fiabilidad de calidad de servicio en las aplicaciones multimedia.
- Con las mediciones de la pérdida de paquetes, el retardo la variación del retardo (jitter) y la capacidad máxima de transmisión de datos en la red se podrá determinar la calidad de servicio brindado a las aplicaciones y servicios multimedia probados
- Luego de esto se implementarán los mecanismos de calidad de servicios analizados y se evaluarán si las prestaciones son las adecuadas

El proyecto se realizará en los laboratorios del Centro de Investigaciones en Tecnologías de la Información y la Comunicación (CITIC) de la Facultad de Informática, Electrónica y Comunicación de la Universidad de Panamá utilizando equipamiento de bajo costo y software libre La utilización de software libre nos permitirá acceder a la configuración de los routers de la red mesh y realizar modificaciones en los mismos equipos

1 4 TRABAJOS RELACIONADOS

En esta sección se presentan los trabajos relacionados que tienen más relación con la provisión de calidad de servicio en redes mesh inalámbricas para aplicaciones multimedia reales. La mayor parte de los trabajos relacionados con calidad de servicio en las WMN se tratan de protocolos de enrutamiento con calidad de servicio.

(Kone Das, Zhao & Zheng, 2008) utilizan QUORUM, un protocolo de enrutamiento optimizado para WMN que proporciona propiedades de QoS precisas al predecir correctamente retraso y características de la pérdida de tráfico de datos. Lo que permite que siga con mayor precisión los requisitos de QoS y reducir al mínimo el mal comportamiento de los nodos egoístas.

(Liu, Shu, Zhang, Zhao & Li, 2009) diseñaron e implementaron un Campus de Pruebas (Red de Datos Experimental) para soportar el tráfico multimedia en Redes de Malla Inalámbricas e investigaron en detalle las mejoras en el número de capas para permitir la transmisión multimedia a través de redes inalámbricas con soporte QoS. Estudiaron una serie de mejoras de algunos protocolos de encaminamiento existentes para apoyar las transmisiones multimedia. Se aprendieron nuevos mecanismos de Control de Admisión y Control de la Tasa de Transmisión y sus ganancias de rendimiento mediante la experimentación.

(Cheng & Zhuang, 2009) proponen un enfoque de poca complejidad para la asignación de recursos a un nodo de WMN. Con la cooperación beneficiosa del nodo se demostró que el enfoque propuesto puede ser prometedor en el aprovisionamiento de QoS y el aumento del rendimiento del sistema.

(Kuklinski Radziszewski & Wytrębowicz, 2011) plantearon un nuevo protocolo de enrutamiento para las redes WMN llamado Wireless and Autonomic Routing Framework (WARF) En este protocolo pueden ser implementados o reconfigurados los nodos con respecto a las propiedades de la topología de red (estática y dinámica) el modo de emisión de los datos (unicast, broadcast) o algunas otras propiedades requeridas (movilidad, QoS etc)

Estas propuestas académicas con provisión de calidad de servicio en redes mesh inalámbricas no fueron tenidas en cuenta por los organismos de estandarización, por lo que se debe evaluar las oportunidades que provee el estándar emergente de la red mesh IEEE 802.11s. No encontramos en las literaturas una solución mediante la implementación de los mecanismos de calidad de servicio en una red de prueba (testbed) inalámbrica multi saltos para las aplicaciones multimedia reales utilizando el estándar IEEE 802.11s con un firmware de software libre como OpenWRT. Esta será mi aportación de tesis de investigación.

FUNDAMENTOS TEÓRICOS

CAPÍTULO II

FUNDAMENTOS TEÓRICOS

2 REDES INHALAMBRICAS

Las redes inalámbricas son aquellas que no poseen cables. Hacen uso de las ondas electromagnéticas para enviar señales a través de largas distancias. En síntesis, las redes inalámbricas a diferencia de sus antecesoras son más simples de instalar, productivas, seguras, escalables, menos complejas en su administración y de bajo costo. A través de esta tecnología, puede disponerse de conexión a Internet casi en cualquier lugar siempre que exista un punto de acceso enlazado con el punto de acceso a la Red de Datos (Andreu, Pellejero & Lesta, 2006). Las redes Mesh Inalámbricas se pueden clasificar en diferentes tipos en función de los elementos que participan en la red y el tamaño del área de cobertura. Nos enfocaremos en las Redes de Área Local Inalámbricas (WLAN) usada en el estándar IEEE 802.11. Las tecnologías WLAN permiten a los usuarios establecer conexiones inalámbricas dentro de un área local como un edificio o campus empresarial o en un espacio público.

2.1.1 REDES MESH INALÁMBRICAS

Las Redes Mesh Inalámbricas (WMN) son redes de datos con topología de red de malla formadas por nodos que establecen y mantienen la conectividad de red automáticamente mediante radio-enlaces. Las WMNs surgieron como una tecnología capaz de brindar conectividad de banda ancha segura y de bajo costo. Pueden ser instaladas en zonas urbanas, suburbanas y en entornos rurales. Son redes inalámbricas

auto-organizada, auto-configurables y son fácilmente adaptables a diferentes necesidades de tráfico y los cambios de la red (Akyildiz, Wang, & Wang 2005)

Entre las ventajas del uso de WMNs es su capacidad de reconstrucción de rutas al sufrir de pérdidas de enlace tiene una buena escalabilidad y el manejo de múltiples interfaces de radio y antenas inteligentes Esta tecnología presenta aplicaciones interesantes en redes domésticas de banda ancha, redes comunitarias, de vecindario la automatización de edificios, redes corporativas, video vigilancia, para casos de emergencia y desastre (Joaquin Chung et al 2012)

2 1.2. ARQUITECTURA DE LAS REDES MESH INALAMBRICAS

Hay dos tipos de nodos presentes en una Red Mesh Inalámbricas Enrutadores Mesh y Clientes Mesh.

Enrutadores Mesh tienen la capacidad de enrutamiento para las funciones de gateway / repetidor además contiene funciones adicionales de enrutamiento para apoyar a las redes mesh (Khan, Reza, & Moradi 2007) Están equipados con múltiples interfaces inalámbricas construidas sobre la misma o diferente tecnología de acceso inalámbrico Puede alcanzar la misma cobertura que un router convencional utilizando una potencia de transmisión mucho más baja a través de comunicaciones multi hop Los enrutadores mesh forman el backbone inalámbrico de la red (Akyildiz et al 2005)

Clientes Mesh se conectan a la red mesh a través de los enrutadores mesh. Sin embargo las funciones de puerta de enlace o repetidor no existen en estos nodos Sólo poseen una interfaz inalámbrica Como consecuencia de ello la plataforma de hardware y el software para los clientes mesh pueden ser mucho más simples que los enrutadores mesh

Los clientes mesh tienen una mayor variedad de dispositivos que pueden ser utilizados como clientes mesh tales como computadoras portátiles computadoras de escritorio Pocket PC PDA, teléfono IP lector RFID controlador BACnet, teléfonos inteligentes y muchos otros dispositivos

Los nodos en WMN se pueden dividir en diferentes categorías(Khan et al 2007) (Carrano et al 2011) se muestra en la (fig 3)

- **Client or Station (STA)** Es un nodo que solicita servicios pero no participa en mecanismos de descubrimiento de ruta.
- **Mesh Station (Mesh STA)** es un nodo que participa en generación/recepción de la formación y el funcionamiento de la nube de malla,
- **Mesh Point (MP)** Establece enlace de nodos con los vecinos MP y participa en los servicios WLAN mesh, pero puede tener una comunicación única de salto sólo con el vecino inmediato
- **Mesh Access Point (Mesh AP o MAP)** es un nodo de la malla o mesh que tiene un punto de acceso (AP) conectado para proveer servicios para los clientes (STA)
- **Mesh Portal Point (MPP)** Es un STA de acoplamiento con la funcionalidad adicional de actuar como un puente o puerta de enlace entre la nube de malla y las redes externas

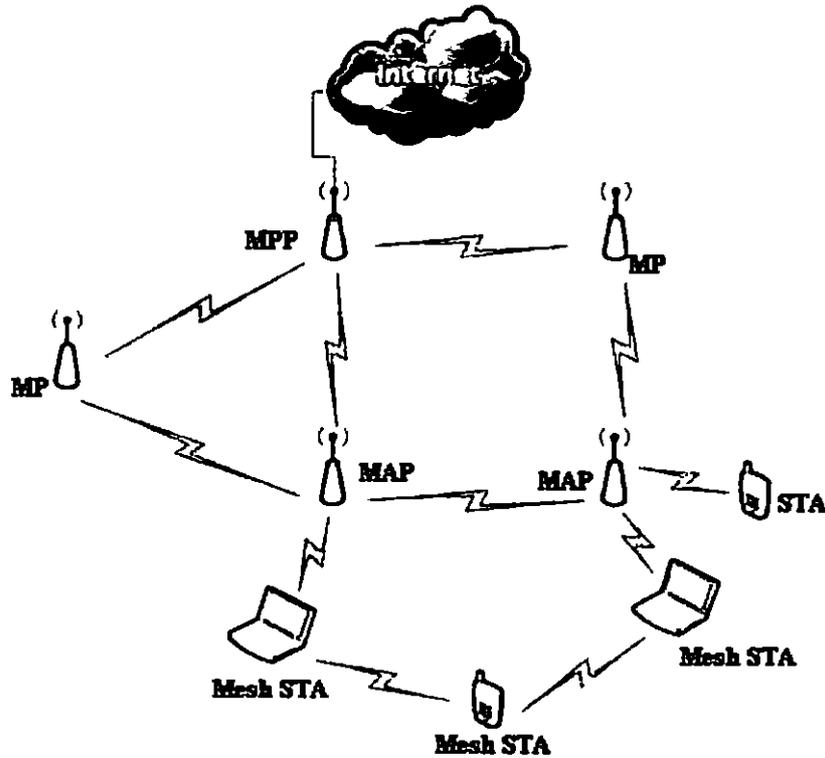


Fig 3 Arquitectura de una Red WMN (Wireless Mesh Network)

La topología de la WMN se puede clasificar en tres grupos principales basados en la funcionalidad de los nodos (Akyildiz et al 2005)

- WMN de Infraestructura / Backbone** Este tipo de WMN incluye los enrutadores mesh que forman una infraestructura para los clientes mesh que se conectan a ellos. Se muestra en la Fig 4. La infraestructura se puede construir utilizando diversos tipos de tecnologías inalámbricas. Los enrutadores mesh con la funcionalidad de gateway se pueden conectar a Internet y brindar acceso a Internet a los clientes mesh. Este tipo de infraestructura se utiliza por ejemplo en redes comunitarias.

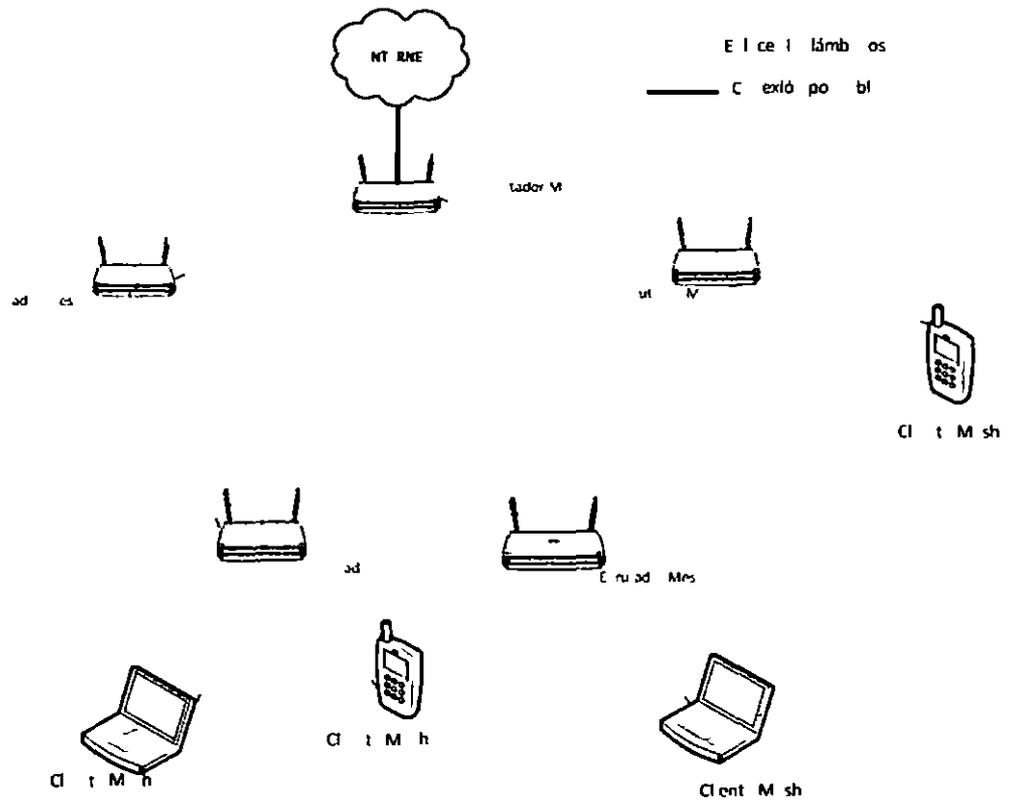


Fig 4 WMN de Infraestructura /Backbone

- WMN de Cliente** Esta topología proporciona conexiones punto a punto entre dispositivos cliente. Los nodos clientes llevan a cabo las funciones de enrutamiento y configuración. Por lo tanto, un enrutador mesh no se requiere para estos tipos de redes. La topología se muestra en la Fig 5

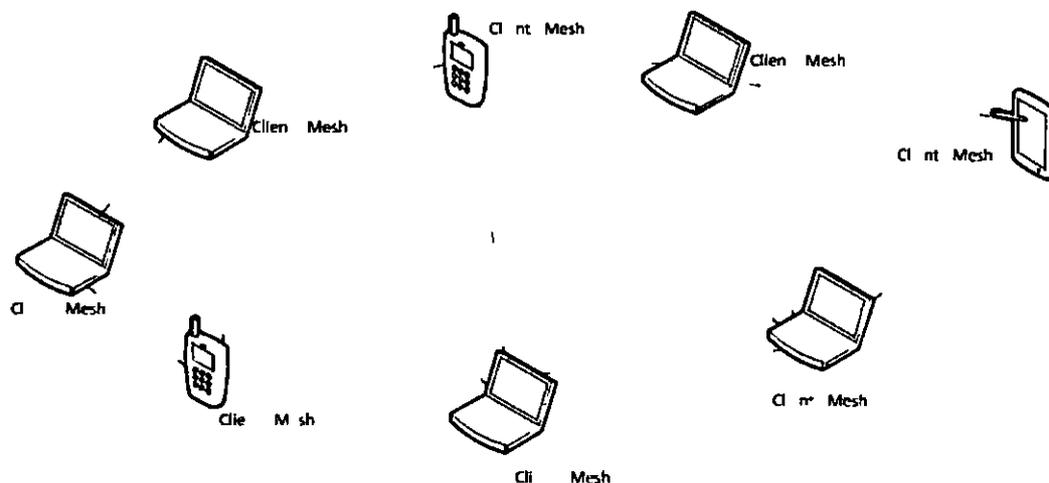


Fig 5 WMN de Clientes

- WMN Híbridas** Esta arquitectura es la combinación de WMN de infraestructura y las WMN de cliente como se muestra en la Fig 6. Los clientes mesh pueden acceder a la red a través de los enrutadores mesh y pueden conectarse con otros clientes mesh. Mientras que la infraestructura permite la conectividad con otras redes como Internet, Wi-Fi, WiMAX, celular y las redes de sensores, las capacidades de enrutamiento de los clientes proporcionan una mejor conectividad y cobertura dentro de la WMN.

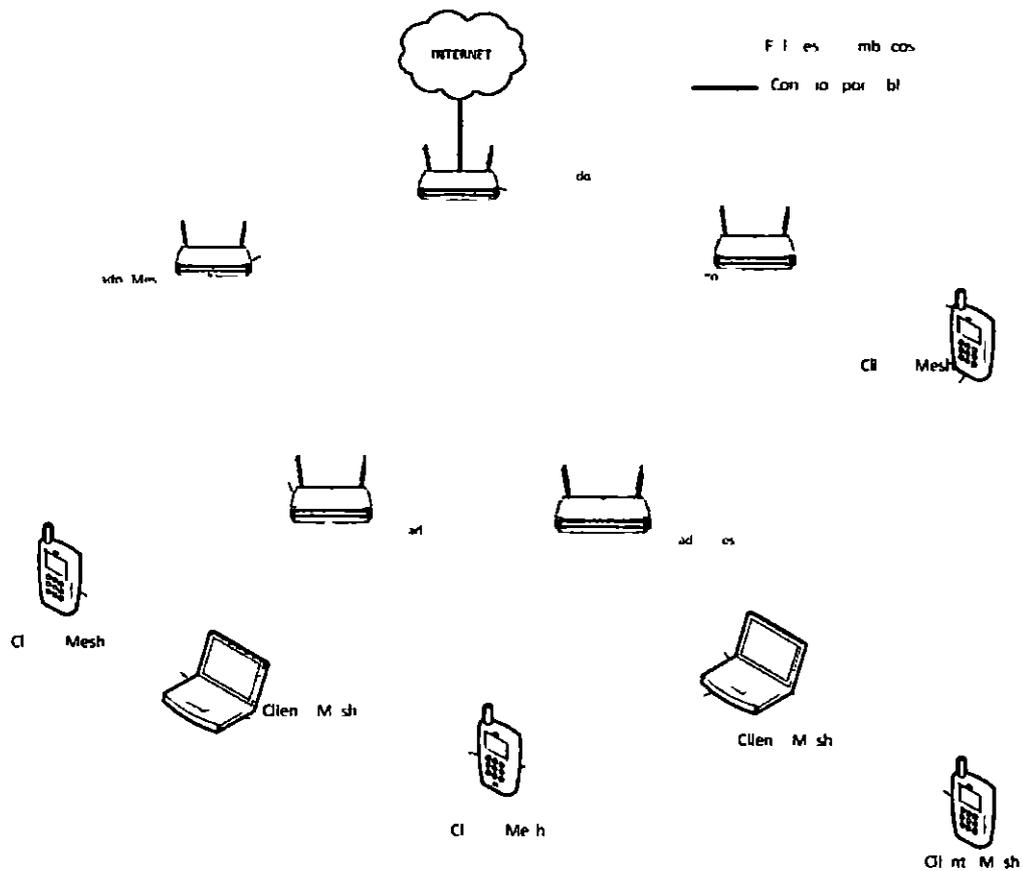


Fig 6 WMN Híbrida

Las características de WMN son (Akyıldız et al 2005)

- Es una red inalámbrica multisaltos en donde se extiende el rango de cobertura sin sacrificar la capacidad del canal no es necesario tener línea de vista con el destino.
- Soporte para redes ad hoc y capacidad de auto-formación, auto-recuperación y auto-organización. La red puede crecer gradualmente a medida que sea necesario

- Dependencia de la movilidad en los tipos de nodos de la red mesh. Enrutadores mesh por lo general tienen la movilidad mínima, mientras que los clientes mesh pueden ser nodos fijos o móviles
- Múltiples tipos de acceso a la red ya sea conectándose como WMN de Clientes o WMN de Infraestructura /backbone
- La dependencia del consumo de energía tiene que ver con el tipo de nodo de la red mesh. Enrutadores mesh por lo general no tienen restricciones estrictas sobre el consumo de energía. Sin embargo, los clientes mesh pueden requerir protocolos eficientes de energía.
- La compatibilidad y la interoperabilidad con otras redes inalámbricas existentes

2.1.3 ENRUTAMIENTO EN 802.11S

Redes de malla inalámbricas son redes multi salto. Por lo tanto se necesita un mecanismo para encontrar un camino entre la fuente y el destino. Este mecanismo se llama enrutamiento. Dependiendo del protocolo de enrutamiento, múltiples rutas entre los nodos locales y el MPP (Internet Gateway) pueden estar disponibles (Ksentini & Abassi, 2008). Una métrica es una propiedad de una ruta que consiste en cualquier valor utilizado por los algoritmos de enrutamiento para determinar si una ruta debería funcionar mejor que otra. Una métrica puede incluir la siguiente información: ancho de banda, retardo de la red, de paquetes, la tasa de pérdida, el número de saltos, coste de la ruta, carga, fiabilidad y comunicación (Chen, Tsai & Gerla, 2007). Cada protocolo de enrutamiento usa su propia métrica. Algunos de las métricas más utilizados para WMN son Expected Transmission Count (ETX) (De Couto, Aguayo, Bicket, & Morris, 2003) es la tasa de

pérdida de la emisión de paquetes entre par de nodos Round Trip Time (RTT) (Akyıldız et al 2005) es el retardo de ida y vuelta entre el par de nodos Una evolución del ETX es la métrica ETT (Expected Transmission Time) (Carrano et al 2011) es el tiempo que requiere un paquete de datos para ser transmitido de forma satisfactoria entre par de nodos

Los protocolos de enrutamiento se clasifican en tres categorías protocolos de enrutamiento proactivos reactivos y híbridos Los protocolos de enrutamiento proactivos también se conocen como no adaptativa (enrutamiento estático) se caracteriza porque determina las rutas solo cuando hay datos que enviar Protocolos de enrutamiento reactiva también se conocen como adaptativa (enrutamiento dinámico) se caracterizan por poseer una topología de red dinámica, intenta mantener las rutas a todos los destinos al mismo tiempo buscan la ruta bajo demanda Híbrido es un metodo que combina las dos categorías anteriores se apoya de las ventajas de los protocolos proactivos y reactivos Utiliza el enrutamiento proactivo cuando los nodos están cerca y utiliza el encaminamiento reactivo cuando los nodos están lejos También se utiliza el encaminamiento reactivo cuando los caminos son utilizados en pocas ocasiones

El estándar IEEE 802.11s no es la única manera de construir una WMN De hecho durante la última década, muchos protocolos de enrutamiento fueron diseñados para redes ad hoc y se pueden utilizar también en WMN Algunos de ellos son AODV (Ad hoc On Demand Distance Vector Routing) (Perkins et al 2003) OLSR (Optimized Link State Routing) (Clausen & Jacquet, 2003) DYMO (Dynamic MANET On-demand) (Chakeres & Perkins 2010) ZRP (Zone Routing Protocol) (Haas Pearlman, & Samar 2002) TBRPF (Topology Dissemination Based on Reverse-Path Forwarding) (Ogier Templin,

& Lewis 2004) y DSR (Dynamic Source Routing) (Johnson, Hu, & Maltz 2007) se han desarrollado como parte de la actividad IETF MANET WG. Las características de las redes mesh son diferentes de la red Mobile Ad hoc (MANET) en el que el MP (Mesh Point) es estacionario y el tráfico de malla se envía principalmente desde y hacia nodos gateway sobre líneas de cables que forman una topología de árbol lógico. Se ha adaptado el protocolo de enrutamiento MANET a WMN. También hay protocolos de enrutamiento de trayectos múltiples desarrollados para WMN: AOMDV (ad hoc on-demand multipath distance vector) (Marina & Das 2006), ROAM (Routing On-demand Acyclic Multi path), SMR (Split Multi path Routing) y MP DSR (Multi Path Dynamic Source Routing) (Alotaibi & Mukherjee 2012) pero se utilizan raramente. Desafortunadamente muchos de los protocolos de enrutamiento son sólo adecuados para escenarios de uso específicos (Kuklinski et al 2011).

El enrutamiento en 802.11s se hace mediante HWMP (Hybrid Wireless Mesh Protocol). Este protocolo debe ser implementado obligatoriamente por todos los nodos mesh, aunque se permite usar protocolos adicionales. Utilizamos el protocolo del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) en la norma 802.11s del 2012, el mismo se conoce como HWMP (Hybrid Wireless Mesh Protocol).

HWMP es el protocolo predeterminado para IEEE 802.11s WMN. Protocolo de selección de ruta que combina la flexibilidad de selección de ruta bajo demanda con las extensiones de topología de árbol proactivas. La combinación de elementos reactivos y proactivos de HWMP permite la selección de rutas eficiente en una amplia variedad de redes de mallas (con o sin acceso a la infraestructura) (Society 2012).

HWMP usa un conjunto en comun de elementos de protocolo generacion y reglas de procesamiento inspirados en el protocolo Ad Hoc On Demand Distance Vector (AODV) adaptado para la selección de caminos o rutas basada en dirección MAC y entendimiento de la métrica del enlace (Society 2012)

Es compatible con dos modos de funcionamiento dependiendo de la configuración Estos modos proporcionan diferentes niveles de funcionalidad como sigue

- **Modo On-demand** La funcionalidad de este modo siempre está disponible independiente de si una mesh STA está configurado en el MBSS o no Este modo permite que las estaciones mesh puedan comunicarse utilizando rutas peer to-peer (Society 2012) Se adopta para nodos de la malla que experimentan un entorno cambiante(Ksentini & Abassi 2008) Este modo funciona normalmente cuando no hay MP (Mesh Point) raíz Pero en algunos casos especiales se trabaja con MP raíz (Khan et al 2007)
- **Modo basado en árbol proactivo** En este modo se añade la funcionalidad de generación de arboles proactiva adicional para el modo on-demand (Society 2012) Es una opción eficiente para los nodos de la malla en una topología de red fija. Se aplica cuando un nodo raíz está configurado en la red de malla Con esta raíz, un árbol de vector de distancia se puede construir y mantener para otros nodos, lo que puede evitar la sobrecarga innecesaria de enrutamiento para el encaminamiento de descubrimiento y recuperación de la ruta (Ksentini & Abassi 2008) En este modo se utiliza el mecanismo de mensaje PREQ (Solicitud de ruta) y RANN (Anuncio de la raíz) (Khan et al 2007)

Estos modos no son excluyentes En la demanda y los modos proactivas se utilizan al mismo tiempo debido a que los modos proactivas son extensiones de la modalidad on demand

Todos los modos de operación HWMP utilizan reglas y primitivas de procesamiento comunes Elementos HWMP son la solicitud de ruta PREQ (Path Request) ruta de respuesta PREP (Path Reply) error de ruta de acceso PERR (Path Error) y anuncio de la raíz RANN (Root Announcement) El costo métrica de los enlaces determina qué rutas construye HWMP Con el fin de propagar la métrica de información entre las estaciones mesh, las métricas de campo que se utiliza son el PREQ PREP y elementos de RANN (Society 2012)

Selección Camino en HWMP utiliza un mecanismo de numero de secuencia para asegurar que las estaciones mesh puedan distinguir la información de la ruta actual de información de la ruta antigua en todo momento con el fin de mantener la conectividad sin bucles (Society 2012)

2 1 4 SERVICIOS MULTIMEDIA

El propósito de esta tesis es evaluar la capacidad de proveer calidad de servicio en aplicaciones multimedia reales a través de las redes mesh inalámbricas Nos enfocaremos en los siguientes tres servicios VoIP Video Streaming y Video Conferencia.

2 1 4 1 VoIP

VoIP (Voice Over Internet Protocol) es un estándar de la ITU. En su Informe de 2001 sobre la telefonía IP, la Unión Internacional de Telecomunicaciones (ITU) distinguió el término telefonía IP como refiriéndose a la voz sobre redes basadas en IP, se refiere a la emisión de voz en paquetes IP sobre redes de datos como puede ser voz sobre banda ancha (VoB), voz sobre línea de abonado digital (DSL), Voz sobre Internet (VOI), voz sobre la red inalámbrica de área local y la telefonía por Internet (ITU 2007b).

Es el mejor ejemplo de una tecnología convergente que se encuentran en el mercado de las telecomunicaciones de hoy. Durante la última década, impulsado por el crecimiento de las redes de banda ancha y la reducción de costes, los servicios de VoIP han revolucionado la economía de los servicios de voz de transmisión y provisión (Phillippa Biggs 2009).

VoIP es una tecnología y no un servicio que permite encapsular la voz en paquetes para poder ser transportados sobre redes de datos sin necesidad de disponer de la red pública (PSTN) la cual se utilizó antes para la transmisión de señales de voz. A diferencia de la Red Telefónica Pública Conmutada que utiliza conmutación de circuitos, la telefonía IP envía múltiples conversaciones a través del mismo canal codificadas en paquetes y en flujos independientes.

2 1 4 2 Principales protocolos de VoIP

Debe existir un protocolo para que los equipos puedan comunicarse y encontrarse mutuamente incorporados y cómo intercambiar la información con el

fin de permitir el flujo de paquetes entre los dispositivos de comunicacion Algunos de los principales protocolos para las distintas funciones son

2.1.4.2.1 Protocolo de señalización de Llamadas

Encuentra el dispositivo remoto y negocia los recursos por los cuales los medios de comunicacion pueden fluir entre los dos dispositivos para establecer la comunicación multimedia. Los dos protocolos más populares de VoIP son ITU T Recommendation H 323 y Session Initiation Protocol (SIP) (ITU 2007b)

- **ITU T Recommendation H.323** Es una Recomendación ITU T que se basa en los protocolos anteriores para la transmisión de voz y vídeo a través de PSTN ISDN y ATM Se inspira en gran medida de la herencia del sistema de comunicación y es un protocolo binario lo que significa que tiene una mejor interoperabilidad con PSTN y ofrece un mejor soporte para el transporte fiable fuera de banda de vídeo y DTMF Es el protocolo más ampliamente adoptado para la transmisión de VoIP en parte porque fue publicado por primera vez como un estándar desde el principio en 1996 No lo hace sin embargo abarcar todos los aspectos de las comunicaciones VoIP y cada vendedor puede tener sus propias variaciones de los algoritmos de compresión general de arquitectura de red y del habla.
- **Session Initiation Protocol (SIP)** es el otro protocolo principal de señalización de extremo a extremo Facilita la comunicación entre dos o más dispositivos soportados para SIP pero puede ser necesaria la utilización de más protocolos para realizar llamadas VoIP incluyendo extensiones SIP

para llevar la información necesaria y proporcionar la funcionalidad principal SIP es más popular para sistemas de mensajería instantánea (ITU 2007b)

2 1 4.2.2 Protocolos de control de dispositivos

Adminiistran la interfaz entre el Media Gateway Controller (MGC) que contiene la lógica de control de llamadas y la interfaz Media Gateway (MG) con la PSTN en la puerta de enlace dividida. Estos protocolos incluyen la Recomendación ITU T H 248 y Media Gateway Controller Protocol (MGCP) Más allá de estos protocolos también hay otros protocolos no estándar tales como Skype que han demostrado ser popular en el mercado(ITU 2007b)

2 1 4 2.3 Otros protocolos de soporte para VoIP

Estas arquitecturas incluirán protocolos tales como el Protocolo de Transporte en Tiempo Real (Real Time Transport Protocol RTP) (RFC 3550) (Schulzrinne Casner Frederick, & Jacobson, 2003) para el transporte de datos en tiempo real y proporciona retroalimentación QoS el protocolo de transmisión en tiempo real (Real Time Streaming Protocol RTSP) (RFC 2326) (Schulzrinne Rao & Lanphier 1998) para controlar el suministro de medios de transmisión el protocolo Media Gateway Control Protocol (MEGACO) (RFC 3015) (Cuervo et al 2000) para el control de los gateways a la red telefónica conmutada publica (Public Switched Telephone Network, PSTN) y el Session Description Protocol (SDP) (RFC 2327) (Handley & Jacobson, 1998) para describir sesiones

multimedia. Por lo tanto, el SIP se debe utilizar en combinación con otros protocolos con el fin de proporcionar servicios completos a los usuarios. Sin embargo, la funcionalidad básica y el funcionamiento de SIP no dependen de ninguno de estos protocolos. No hace falta señalar que SIP es un protocolo que funciona tanto sobre UDP (J. Postel, 1980) como TCP (Jon Postel, 1981).

2.1.4.3 Codec en VoIP

VoIP es un grupo de recursos que hacen posible que la señal de voz viaje a través de Internet empleando un protocolo IP. La voz debe codificarse para poder ser transmitida por la red IP. Para ello se hace uso de códecs que garanticen la codificación y compresión del audio o del video para su posterior decodificación y descompresión antes de poder generar un sonido o imagen utilizable. Según el códec utilizado en la transmisión, se utilizará más o menos ancho de banda. La cantidad de ancho de banda utilizada suele ser directamente proporcional a la calidad de los datos transmitidos.

Entre los códecs más utilizados en VoIP según la ITU están G.711, G.723.1 y el G.729.

Estos códecs tienen los siguientes anchos de banda de codificación ver (CUADRO II) (Lee, 2006).

CUADRO II CODEC DE VOIP

Año	Codec	Bit Rate
1972	G 711	64 kbps
1988	G 722	48 56 o 64 kbps
1988	G 726	13 24 32 40 kbps
1992	G 728	16 kbps
1995	G 723 1	5 3 o 6 3 kbps
1996	G 729	6 4 8 11 8 kbps
1999	G 722 1	24 32 kbps
2001	G 722 2	23 8 kbps
2006	G 729 1	8 o 32 kbps

2 1 4 4 Componentes de una red VoIP

Estructura básica de una red VoIP ver figura (Fig 7)

- **Terminales** Son los dispositivos que utilizarán los usuarios para comunicarse. Implementados tanto en hardware como en software realizan las funciones de los teléfonos tradicionales.
- **Gateways** De forma transparente se encargan de conectar las redes VoIP con las redes de telefonía tradicional. Podemos considerar al Gateway como una caja que por un lado tiene un interface LAN y por el otro dispone de uno o varias interfaces como POST T1/E1 ISDN E&M trunks.
- **Servidor** Proporciona el manejo y las funciones administrativas para soportar el enrutamiento de llamadas a través de la red IP.

- Red IP: Suministra la conectividad entre los terminales, esta puede ser una red IP privada, una Intranet o el propio Internet.



Fig 7. Red VoIP

2.1.5. Video Streaming

La transmisión en secuencias consiste en transmitir el contenido como paquetes que se van descargando a medida que el destinatario visualiza una pequeña parte del contenido ya descargado sin que éste tenga que esperar mucho tiempo. En la jerga tecnológica este mecanismo se conoce con el nombre de transmisión en secuencias (Streaming) (ITU, 2005).

La transmisión en secuencias permite ofrecer vídeo y audio por Internet o a través de diversas LAN (redes de área local) y WAN (redes de área extensa). La transmisión en secuencias de vídeo permite retransmitir en directo un acontecimiento o presentación, o reproducir "a la carta" vídeo almacenado.

Los usos más comunes de la transmisión de contenido de secuencias son los siguientes (ITU 2005)

- **Presentación, lanzamiento o actualización de productos o la presentación de servicios**
- **Comunicación con las sucursales de una empresa**
- **Actualización de ventas**
- **Programas de formación en línea**
- **Presentaciones de la empresa**
- **Reuniones de la Junta Directiva**
- **Conferencias con los clientes**
- **Noticias y programas de entretenimiento**

Tecnologías clásicas de Video Streaming son Real Media, QuickTime Windows Media. Entre las tecnologías emergentes mas conocidas y gratuitas que pueden utilizarse como servidor para transmitir Video Streaming en una red están

- **Flash Media(Adobe n.d) Es multiplataforma, posee código y protocolos propietarios**
- **Red 5 (Red5 n.d) Open Source Flash Server Proyecto OSFlash de software libre es multiplataforma, posee código abierto y protocolos no estándar Utilizado de forma profesional por algunas empresas**
- **VLC Media Player (VideoLAN n.d) Es un software libre multiplataforma tiene código abierto y protocolos estándar Reproduce la mayoría de archivos**

multimedia, así como DVD Audio CD VCD y diversos protocolos de transmisión

Características del software

- Reproductor multimedia simple rápido y potente
- Reproduce todo archivos discos cámaras de video dispositivos y flujos
- Reproduce la mayoría de códecs sin necesidad de paquetes de códecs MPEG 2 DivX, H 264 MKV WebM, WMV MP3 etc
- Se ejecuta en todas las plataformas Windows Linux Mac OS X, Unix
- Completamente libre 0 programas espía, 0 anuncios y sin seguimiento del usuario
- Puede convertir medios y emitirlos

2.1.5.1 Contenedor de formato

Un contenedor de formato es un tipo de formato de archivo que almacena información de video audio meta-datos e información de sincronización y corrección de errores siguiendo un formato preestablecido en su especificación técnica. Algunos contenedores multimedia son AVI, MPEG program stream (contenedor estándar para MPEG 1 and MPEG 2) MPEG 2 transport stream (MPEG TS) MOV (contenedor de QuickTime) ASF (contenedor de WMV y WMA) Ogg, OGM, RMVB WebM y Matroska, MP4 Flash (F4V) 3GP para celulares

Se presenta en el cuadro (CUADRO III y IV) todos los contenedores utilizados para el video Streaming

CUADRO III. CONTENEDORES DE FORMATO

Proprietor/ Developer	License	Variable bit rate audio	Variable frame rate	Chapters	Captions (Sub titles)	Video formats supported	Audio formats supported	Metadata	Menu Support (like DVD)	Streaming Support	Attachment Support	3D Support (codec)	3D Support at container level	Supported by hardware players
OGG	?	Yes	Yes	?	OGG Timed Text	H.263, MPEG-4 Part 2 and H.264/MPEG-4 AVC	AMR-NB, AMR-WB, AMR-WB+, AAC, HE-AAC and HE-AAC v2	?	No	?	?	?	?	?
OGG2	?	Yes	Yes	?	OGG Timed Text	H.263, MPEG-4 Part 2 and H.264/MPEG-4 AVC	AMR-NB, AMR-WB, AAC, HE-AAC, EVRC, EVRC-B, EVRC-WB, 13K (GOELP), SMV or VMR-WB	?	No	?	?	?	?	?
Microsoft	Proprietary, patent encumbered	Yes	Yes	Yes	Yes	Almost anything through VFW or OMO/H.264/AVC is problematic	Almost anything through ACM or DMO (Vorbis is problematic)	Yes	No	Yes	?	?	?	Yes
Microsoft	?	Yes	Yes	Yes, via third party modifications	Yes, via third party modifications	Almost anything through VFW	Almost anything through ACM, Vorbis problematic	Yes through RTT channels	No	No	No	?	No	Yes
DivX, Inc.	?	Yes	Yes	Yes	Yes	MPEG-4 Part 2 video encoded to meet DivX profiles	MP3, PCM, AC-3	?	Yes	Yes	?	?	?	Yes
MPEG	Patent encumbered	Yes	Yes	Yes	Yes	MPEG-2 Part 2, H.264/MPEG-4 AVC, VC-1	AC-3, E-AC-3, Dolby TrueHD, Linear PCM, DTS, DTS-HD, MPEG-2 Part 3	?	Yes	?	?	?	?	?
Adobe Systems	Patent encumbered	?	Yes	?	?	H.264/MPEG-4 AVC	MP3, AAC, HE-AAC	Yes	No	Yes	?	?	?	?
Adobe Systems	Patent encumbered	?	Yes	No	No	Sorenson, VP6 Video, H.264/MPEG-4 AVC	MP3, Nellymoser, A-AC3, Linear PCM, AAC, Speex	Yes	No	Yes	?	?	?	?
CoreCodec, Inc.	Freely licensed	Yes	Yes	Yes	Yes	Virtually anything	Virtually anything	Yes	Virtually anything	Yes	Yes	Yes	Yes	Yes
Community	BSD-style license for specs, GNU for tools	Yes	Yes	Yes	Yes	Not fully specified	Not fully specified	Yes	Possible	Yes	Yes	?	?	No

CUADRO IV. CONTINUACIÓN CONTENEDORES DE FORMATO

Proprietor / Developer	License	Variable bit rate audio	Variable frame rate	Chapters	Captions (Subtitles)	Video formats	Audio formats	Metadata	Menu Support (like DVD)	Streaming	Attachment	3D Support (codec)	3D Support at container level	Supported by hardware players
MPEG	Patent encumbered	Yes	Yes	In specially formatted text track (QuickTime) 2) In userdata atom (started by Hero Digital) can't interact with the sceneDescription or via segmentDescriptor	Text, VobSubs with private objectTypeIndicator not usable with the sceneDescription (started by Hero), BFS	MPEG-2 Part 2, MPEG-4 ASP, H.264/MPEG-4 AVC, H.263, VC-1, Divx, others	MPEG-2M (HE)-AAC, MPEG-1/2 Layers I, II, III (MP3), AC-3, Apple Lossless, ALS, SLS, Vorbis (with private objectTypeIndicator), others	Yes	Yes	Yes	Yes	Yes	?	Yes
MPEG	Patent encumbered	Yes	Yes	No	No	MPEG-1, MPEG-2	MPEG-1 Layers I, II (mp3), other formats only in private streams; LPCM	No	No	Yes	?	?	?	Yes
MPEG	Patent encumbered	Yes	?	Only in private streams — used in VOB files on DVDs	Only in private streams — used in VOB files on DVDs	MPEG-1, MPEG-2, MPEG-4 Part 2, VC-1, H.264	MPEG-1 Layers I, II (mp3), other formats only in private streams; AAC, AC-3, LPCM, DTS	No	Only in private streams — used in VOB files on DVDs	?	?	?	?	Yes
MPEG	Patent encumbered	Yes	Yes	No	possible via ETSI EN 300 743	MPEG-1, MPEG-2 Part 2, MPEG-4 ASP, H.264, other formats in private streams; VC-1, Divx	MPEG-1 Layers I, II (mp3), AAC, other formats in private streams; AC-3, LPCM, DTS	No	No	Yes	?	?	?	Yes
Blu-ray Disc Association	Proprietary	Yes	Yes	Yes	Yes	MPEG-2 Part 2, H.264, VC-1	AC-3, DTS, LPCM, Dolby Digital Plus, DTS HD, Dolby Lossless	?	?	?	?	Yes	?	Yes
SMPTE	?	Yes	Yes	?	Yes	Virtually anything	Virtually anything	Yes	No	?	?	?	?	No
Xiph.org	Patent-free software implementations available	Yes	Yes	Yes	Ogg, Vorbis, Ogg, AAC	Theora, Divx, Ogg (VTS, MNG) and others, and almost anything else via VTS	Vorbis, FLAC, Speex, Opus, Ogg PCM, and almost anything else via ACM	CMML, Ogg Skeleton, Vorbis comment	No	Yes	?	?	?	Yes
Apple	Proprietary	Yes	Yes	Yes	Yes	Limited to what is available to the QuickTime codec manager	Limited to what is available to Sound Manager and CoreAudio	Yes	Yes	Yes	Yes	Yes	Yes	Yes
RealNetworks	Proprietary	?	Yes	?	Yes	RealVideo 8, 9, 10	(HE)-AAC, Cook Codec, Vorbis, RealAudio, Lossless	?	No	Yes	?	?	?	?
DVD Forum	Proprietary	Yes	Yes	Yes	Yes	MPEG-2 Part 2, MPEG-1 Part 2	AC-3, Linear PCM, DTS, MPEG-2 Part 3, MPEG-1 Layer I	No	Yes	No	No	No	No	Yes
WebM Project	perpetual license grant to patents	Yes	Yes	No	Pending	VP8	Vorbis	Pending	No	Yes	No	No	Yes	Yes

2 1 6 Videoconferencia

Segun la ITU la definición de video conferencia es *“La videoconferencia es una forma interactiva de comunicación en tiempo real que utiliza sonido e imágenes Este tipo de comunicación similar a la televisión permite celebrar reuniones debates y presentaciones sin abandonar el lugar de trabajo por lo que se ahorra tiempo y dinero (ITU 2005)*

2 1 6 1 Aplicaciones de Videoconferencia

- **Comunicación uno a uno** Ésta es su forma más sencilla y es el tipo de conferencia que se utiliza más corrientemente en la conferencia entre computadoras Consiste sencillamente en una persona hablando con otra.
- **Comunicación uno a muchos** En este tipo de videoconferencia la información se transmite desde una localización a varias localizaciones Un buen ejemplo podria ser una presentación o la impartición de una clase a varios estudiantes
- **Comunicación muchos a muchos** Esta es la forma más compleja de videoconferencia y principalmente está relacionada con la tecnología de videoconferencias basada en salas En este caso varias partes (más de dos) participan en la conferencia y cada uno puede ver y hablar con los demás

2 1 7 CALIDAD DE SERVICIO

El objetivo de la calidad de servicio (QoS) es el soporte de las características y propiedades de aplicaciones específicas Sin embargo las necesidades de las distintas aplicaciones pueden ser muy diferentes Varias Recomendaciones ITU T trata la

QoS ITU T E 800 define la calidad de Servicio (QoS) como *el efecto colectivo de la calidad de funcionamiento del servicio que determina el grado de satisfacción del usuario del servicio*" La Rec ITU T E 800 considera que el soporte operabilidad posibilidad de ser servido (serviceability) y la seguridad son todos ellos parte de la calidad de funcionamiento del servicio Ampliando el concepto de QoS E 800 la Rec ITU T G 1000 divide la calidad de funcionamiento del servicio (o calidad de servicio) en componentes funcionales y la vincula a la calidad de funcionamiento de red tal como se define en las Recomendaciones ITU T I 350 Y 1540 e Y 1541 Como complemento a la Rec ITU T G 1000 la Rec ITU T G 1010 describe requisitos de aplicación centrados en el usuario de extremo en términos de amplias categorías (tales como interactivo tolerante a los errores etc) En lo que respecta a aplicaciones o parámetros de calidad de funcionamiento específicos entre las normas conexas la Rec ITU R M 1079 2 define los requisitos de calidad vocal y calidad de datos así como de calidad de funcionamiento de extremo a extremo para las redes de acceso IMT 2000 en tanto que la Rec ITU T G 114 especifica los límites para los tiempos de transmisión y las conexiones a través de una red digital

IETF RFC 2386 define calidad de servicio como un *conjunto de requisitos del servicio que debe cumplir la red en el transporte de un flujo*

Los parámetros de QoS son característicos del nivel de calidad de un determinado aspecto en la provisión de un servicio Deben cumplir los siguientes requisitos

- i Ser sencillos de utilizar
- ii Tener una correspondencia demostrable con las percepciones de los usuarios
- iii Haber sido aceptados como estándares

Algunos parámetros de QoS son

- i Éxito en el establecimiento de la conexión (Disponibilidad)**
- ii Retardo en el establecimiento de la conexión**
- iii Calidad conversacional y vocal**
- iv Calidad en transmisiones de video**
- v Parámetros de error en la capa de red de transporte**
- vi Parámetros de rendimiento en redes IP**

Las métricas de QoS son

- i La velocidad de transmisión**
- ii El ancho de banda fiable**
- iii El retardo de extremo a extremo (latencia) y**
- iv El jitter o variación del retardo**

Los parámetros enumerados son requisitos básicos para un mejor tráfico en aplicaciones multimedia reales y de voz

Nosotros definimos la Calidad de servicio (QoS) como la capacidad de la red para garantizar un nivel de rendimiento específico para un flujo especialmente para el tráfico multimedia.

QoS proporciona un servicio de entrega preferencial para las aplicaciones que lo necesiten, asegurando suficiente ancho de banda, control de latencia y jitter y la reducción de la pérdida de datos

The Internet Engineering Task Force (IETF RFC 2968) define dos modelos principales para la calidad de servicio en redes basadas en IP Servicios Integrados (Intserv) y servicios diferenciados (Diffserv) Estos modelos abarcan varias categorías

de mecanismos que permitan un trato preferencial al tráfico especificado. La ITU presenta cuatro métodos: servicios integrados (IntServ), servicios diferenciados (DiffServ), conmutación por etiqueta multiprotocolo (MPLS) y QoS dinámica. IPCablecom. En ambos organismos de estandarización los métodos están basados a nivel de la capa de red. A continuación explicaremos brevemente en qué consisten los modelos clásicos de QoS.

2.1.8 IntServ (Integrated Services)

Intserv o Servicios Integrados se enfoca en la reserva de recursos y mecanismos de control de tráfico para apoyar el manejo especial de *flujos de tráfico individuales*.

En el método IntServ (IETF RFC 1633) el objetivo principal es el soporte de QoS para aplicaciones en tiempo real sensibles al funcionamiento de la red. Se ha elaborado partiendo de la noción de que un flujo al que se da servicio a una velocidad algo mayor que su velocidad de datos tiene un retardo sujeto a un límite y que la red puede garantizar el límite de retardo del flujo. Según este método, una aplicación, antes de enviar datos primero señala a la red la petición del servicio deseado, incluyendo sus especificidades, tales como su perfil de tráfico y sus requisitos de ancho de banda y retardo. Después la red determina si puede asignar recursos adecuados (por ejemplo, ancho de banda o espacio de memoria) para proporcionar la calidad de funcionamiento deseada de la petición de servicio. Sólo después de haber sido aceptada la petición, la aplicación puede comenzar a enviar datos. Mientras la aplicación respeta su perfil de tráfico, la red cumple su compromiso de servicio manteniendo el estado del flujo y utilizando disciplinas avanzadas de puesta en cola o *queueing* (por ejemplo, puesta en cola ponderada sobre una

base justa) para compartición de enlace. Los bloques de construcción de importancia para el método IntServ incluyen control de admisión, puesta en cola, reserva de recurso, clasificación de tráfico y aplicación de políticas de tráfico. En particular, el protocolo de señalización RSVP se utiliza para reservar recursos. La red puede aceptar o rechazar una petición de reserva de recursos mediante control de admisión basado en la disponibilidad de los recursos. Cuando una petición de recursos es aceptada se instalan los estados pertinentes en los nodos sensibles a RSVP. Los bloques de construcción interactúan mediante el acceso a la información de estado y a otros objetos de datos aprovisionados (por lo que son relativamente estáticos).

2.1.9 DiffServ (Differentiated Services)

El modelo de DiffServ en cambio usa el control de tráfico para apoyar el manejo especial de flujos de tráfico agregados.

La noción en que se funda el método DiffServ es la de tratar un paquete en base a su clase de servicio tal como está codificada en su encabezamiento IP y no se enfoca en el flujo particular al que pertenece. El proveedor de servicio establece con cada usuario un acuerdo de nivel de servicio (o especificación de nivel de servicio) el cual, entre otras cosas, especifica qué cantidad de tráfico un usuario puede enviar dentro de una clase de servicio dada. El tráfico consiguiente se clasifica (paquete por paquete) en un flujo agregado o en una clase dentro de un pequeño número de flujos agregados o de clases y se somete a la aplicación de políticas en el borde de la red del proveedor de servicio. Una vez que el tráfico ha entrado en la red, los encaminadores le dan un tratamiento diferenciado. En contraste con el método IntServ como mencionamos el

tratamiento no se realiza por flujo sino únicamente por la clase de servicio indicada a cada paquete. La red global se establece de modo que se cumplan todos los acuerdos de nivel de servicio. Los bloques de construcción pertinentes (que incluyen gestión de memorias tampón, marcado de paquetes, acuerdo de nivel de servicio, metraje y registro de tráfico, aplicación de políticas de tráfico, conformación de tráfico y calendarización o scheduling) interactúan unos con otros de una manera relativamente estática, esencialmente mediante objetos de datos aprovisionados.

2.1.10 Arquitectura QoS en 802.11

Para entender cómo se provee QoS en el estándar IEEE 802.11s, primer explicaremos cómo funciona la capa de acceso al medio del estándar general IEEE 802.11 para comprender sus deficiencias al respecto de la provisión de calidad de servicio.

2.1.10.1 Descripción del funcionamiento de la capa de acceso al medio del IEEE 802.11

La capa de acceso al medio en 802.11 se encarga de proporcionar un servicio de datos fiable a los protocolos de capas superiores y al mismo tiempo permitir un acceso equitativo al medio inalámbrico compartido. A continuación se describe esta capa usando un Servicio Básico de ejemplo compuesto por un punto de acceso y diferentes estaciones asociadas al mismo. El estándar 802.11 proporciona un acceso confiable mediante un protocolo de intercambio de tramas de información. La secuencia mínima en este intercambio consiste en el envío de

una trama (marco o frame) de información del origen al destino y un asentimiento o reconocimiento (ACK Acknowledgment) enviado por el destino en el caso de que la primera trama haya sido recibida correctamente. Todas las tramas a nivel MAC incorporan un campo de control de errores (FCS Frame Check Sequence IEEE 32 bit CRC) que es comprobado en cada recepción. Si el origen no recibe el ACK o el campo de control falla, la trama es reenviada. Aunque este mecanismo consume ancho de banda, permite enfrentar los posibles errores provocados por el enlace inalámbrico.

Otro mecanismo básico de intercambio de tramas que se proporciona en el 802.11 otorga una mayor robustez al protocolo y permite afrontar el problema de los ‘nodos ocultos’. Éste es conocido por las siglas RTS/CTS que son las tramas que utiliza. Una estación que estuviese haciendo uso de este mecanismo debería mandar una trama RTS (Request To Send) al destino antes de transmitir cualquier trama de datos (también conocida como MSDU – MAC Service Data Unit). Cuando el destino recibe esta trama correctamente entonces éste debe responder con otra trama llamada CTS (Clear To Send). A partir de este momento la fuente podría comenzar a enviar sus tramas MSDU.

Todas las tramas, incluidas las FCS, ACK, RTS y CTS, contienen información sobre la duración de la transmisión de la MSDU/ACK. Así, todas las estaciones presentes pueden actualizar un contador interno llamado Vector de Adjudicación de Red (NAV Network Allocation Vector) y retrasar cualquier transmisión hasta que el contador expire. Aunque una estación oculta no pueda detectar la trama RTS enviada por la fuente, sí podrá recibir la trama CTS con la que responde el

destino de forma que puede actualizar el contador NAV adecuadamente Este mecanismo cuida la transmisión entre estaciones frente a inesperadas transmisiones de estaciones ocultas

El IEEE 802.11 define dos funciones para el control de acceso al canal la Función de Coordinación Distribuida (DCF – Distributed Coordination Function) y la Función de Coordinación Centralizada (PCF – Point Coordination Function) A continuación se describen ambas funciones

- Función de Coordinación Distribuida (DCF)

La DCF brinda un acceso compartido al enlace inalámbrico entre dispositivos con la misma capa física mediante el protocolo basado Acceso Multiple con Detección de Portadora (CSMA – Carrier Sense Multiple Access) con evasión de colisiones (CA – Collision Avoidance) Todas las estaciones deben incluir obligatoriamente este mecanismo

La detección de portadora se realiza de dos maneras una física y otra virtual La detección física se apoya en que toda estación, antes de intentar una transmisión, debe realizar una verificación de las condiciones del enlace y comprobar que el enlace está libre por un periodo de tiempo llamado Espacio Inter Trama (IFS – Inter Frame Space) La duración de este periodo varía, pero la utilizada justo antes de una transmisión en condiciones normales es llamada IFS de Función de Coordinación Distribuida (DIFS)

Para evitar una colisión entre dos estaciones que quieran transmitir simultáneamente se utiliza un algoritmo de espera (Backoff) así como la espera de un periodo DIFS Cuando existen peticiones de transmisiones pendientes y el

enlace se encuentra ocupado la estación espera hasta que el enlace se encuentre vacío por un periodo DIFS. Entonces la estación escoge un número aleatorio entre un rango determinado y usará ese valor como espera adicional antes de transmitir. El rango para elegir esta espera aleatoria es llamado Ventana de Contienda (CW – Contention Window) que varía de acuerdo con el número de retransmisiones previas. Si se detecta que el enlace pasa a estar ocupado durante el periodo de espera, el contador se detiene y se reanuda una vez el medio vuelva a estar vacío después del periodo DIFS.

La evasión de colisión, el mecanismo virtual se consigue a través del mecanismo de detección de portadora virtual. Cada estación mantiene el contador interno llamado NAV que indica cuando el enlace se encuentra ocupado. El valor de la duración se incluye en cada trama transmitida por cada estación e indica cuánto tiempo durará la transmisión incluyendo la de los asentimientos y fragmentos. Todas las estaciones que se encuentren próximas reciben esta trama y usan este valor para actualizar su contador NAV. De forma que cuando una estación quiere comenzar una emisión, primero comprueba que el contador NAV está a cero. Una vez que una estación consigue acceso al enlace ésta puede transmitir la trama de información (MSDU). Luego espera por un periodo de tiempo llamado IFS Corto (SIFS) para transmitir el asentimiento (ACK). La duración del periodo SIFS es más corta que en el caso de DIFS lo que proporciona al ACK mayor prioridad para acceder al medio. De esta manera se asegura que ninguna otra estación podrá comenzar una transmisión antes que el ACK. Si éste no es recibido justo después de un periodo SIFS se intenta una

retransmisión hasta que el número de retransmisiones supera un determinado umbral o el tiempo de vida de la MSDU expira. En este caso la trama de información MSDU sería descartada. Las tramas de información pueden ser fragmentadas para aumentar las probabilidades de éxito en la transmisión. Sin embargo dado que cada fragmento MSDU debe ser reconocido individualmente la fragmentación aumenta considerablemente la sobrecarga para la MSDU

- **Función de Coordinación Centralizada (PCF)**

El estándar 802.11 define un segundo mecanismo de acceso llamado Función de Coordinación Centralizada (PCF) opcional. El PCF está diseñado para ofrecer soporte de servicios con restricciones temporales (soporte de Calidad de Servicio QoS). Un nuevo elemento llamado punto de coordinación (PC – Point Coordinator) es el responsable de priorizar el acceso al medio de determinadas estaciones, y está situado en el punto de acceso (en lo que sigue del documento 'punto coordinador y punto de acceso son equivalentes')

El estándar 802.11 define dos periodos de tiempo entre el envío de dos mensajes de señalización de envío de tráfico (DTIM – Delivery Traffic Indication Message) el periodo de contienda (CP) y el periodo libre de contienda (CFP). En general el punto de acceso manda de forma periódica tramas de señalización (beacon) aunque estas tramas pueden ser retrasadas si el medio está ocupado y transportan información de red y sincronización. Las tramas de beacon (DTIM) son usadas por el Punto de Coordinación para indicar el comienzo del CFP. Durante el CP todas las estaciones compiten por el medio usando el mecanismo DCF. Durante el CFP el punto de acceso clasifica las transmisiones hacia o desde

determinadas estaciones usando un mecanismo de sondeo. No existe contienda entre las estaciones durante el ciclo CFP. Este periodo comienza cuando el punto de acceso consigue acceso al medio mediante el uso de un espacio de tiempo PIFS (IFS de función de coordinación centralizada) a la llegada de una trama de beacon. El tiempo PIFS es más corto que DIFS pero mayor que SIFS y de esta forma PCF logra mayor prioridad que DCF para el acceso pero no interrumpe ninguna comunicación DCF existente. Una vez que PCF consigue el acceso al medio se utiliza el periodo de tiempo SIFS para el intercambio de tramas durante el ciclo CFP.

El sistema de sondeo comienza cuando el Punto de Coordinación envía una trama CF Poll a una de las posibles estaciones. Si el Punto de Coordinación tiene alguna trama pendiente de envío, este podría utilizar una trama de datos incorporando una trama CF Poll (piggy backing). La estación sondeada puede responder con datos junto a una trama CF ACK, o simplemente con una trama CF ACK si no desea enviar más información. Una vez que el intercambio de tramas con una estación termina, el Punto de Coordinación envía el CF Poll a otra estación que estuviese en la lista de estaciones sondeables. Cuando el Punto de Coordinación ha terminado con todas las estaciones de la lista, o una vez que la duración del CFP ha expirado, el Punto de Coordinación transmite por difusión una trama CF End anunciando el final del ciclo CFP.

Cuando llega una trama de beacon, el contador NAV de todas las estaciones se inicializa al valor máximo para proteger el ciclo CFP de transmisiones no deseadas. Entonces el punto de acceso transmite por difusión la duración del ciclo

CFP en la trama de beacon, y el contador NAV se actualiza adecuadamente. Cuando finaliza el ciclo CFP todas las estaciones inicializan su contador NAV a cero cuando reciben la trama CF End, o cuando la duración del CFP termina. Desde entonces hasta la siguiente trama DTIM todas las estaciones compiten por el medio usando DCF. Este modo de funcionamiento permite que en una misma red coexistan estaciones con soporte PCF y DCF.

Sin embargo, el PCF definido en el estándar inicial del IEEE 802.11 presenta varios problemas entre los que está que el tiempo de transmisión de las estaciones en el periodo de contienda (CP) es desconocido. Una estación que ha sido sondeada por el punto de coordinación tiene la posibilidad de enviar una MSDU que podría ser fragmentada y de una longitud arbitraria, hasta un máximo de 2304 bytes. Además, al existir diferentes esquemas de modulación y codificación, la duración del envío de las MSDU después del sondeo no está bajo el control del punto de coordinación (PC), lo que reduce la Calidad de Servicio (QoS) proporcionada a otras estaciones durante el resto del periodo de CFP. De igual forma, dentro del estándar no se ha definido ninguna interfaz de gestión de forma que se pueda controlar las operaciones PCF. En consecuencia, es imposible configurar una política PCF según los requisitos de protocolos de capas superiores tales como DiffServ o IntServ del IETF. De forma que no existe ningún mecanismo para comunicar los requisitos de Calidad de Servicio de las estaciones al punto de acceso, lo que es esencial para optimizar el rendimiento del algoritmo PCF en el punto de coordinación.

2.1.10.2 Descripción del Soporte de QoS según el estándar IEEE 802.11e

Por los problemas mencionados en la sección anterior, el IEEE decidió crear un grupo (802.11e) para estudiar con mayor profundidad este tema e indicar posibles modificaciones sobre 802.11 para lograr cierto nivel de calidad de servicio (QoS). El grupo de trabajo 802.11e define un conjunto de novedades especificadas a la edición de 1999 del estándar IEEE 802.11. Estas mejoras distinguen entre estaciones con soporte de QoS (QSTA – QoS Enhanced Station) y las que no lo soportan (STA), punto de acceso con soporte de QoS (QAP) y sin soporte (AP). Dentro de 802.11e se pueden distinguir básicamente dos grupos funcionales. Se trataría de las funciones de acceso al canal y la gestión de especificación de tráfico o TSPEC.

La extensión 802.11e así lograda, define una función de coordinación llamada Función de Coordinación Híbrida (HCF – Hybrid Coordination Function) que se usa para el Conjunto de Servicios Básicos del IEEE 802 con soporte de QoS (QBSS). La función HCF define dos modos de operación:

- Acceso a Canal Distribuido Mejorado (EDCA – Enhanced Distributed Channel Access) que consiste en una función de acceso al canal que se basa en la contienda y que funciona de forma concurrente junto al segundo modo de operación.
- Acceso a Canal Controlado HCF (HCCA – HCF Controlled Channel Access) que se basa en un mecanismo de sondeo controlado por el Coordinador Híbrido (HC – Hybrid Coordinator). Este punto coordinador se encuentra situado junto al QAP.

Ambas funciones de acceso mejoran o extienden la funcionalidad de los métodos de acceso originales (DCF y PCF). La primera función de acceso EDCA, fue diseñada para soportar la priorización de tráfico tal como hace Diffserv mientras que HCCA soporta tráfico parametrizado de la misma forma que Intserv.

El concepto básico de estas funciones de acceso a canal es la Oportunidad de Transmisión (TXOP – Transmisión Opportunity). Un TXOP es un intervalo de tiempo limitado durante el cual una QSTA puede transmitir una serie de tramas. El periodo TXOP se define a través de un tiempo de inicio y una duración máxima. Si el periodo TXOP se obtiene usando el acceso a canal basado en contienda, entonces recibirá el nombre de EDCA TXOP. Si por el contrario se obtiene a través de HCCA se conocerá como HCCA TXOP. La duración del periodo EDCA TXOP se controla a través del QAP y se transmite al resto de estaciones QSTA en las tramas de beacon junto con otros parámetros relacionados con EDCA. La duración del periodo HCCA TXOP se transmite a las estaciones QSTA directamente por el HC como parte de la trama QoS CF Poll la cual garantiza el periodo HCCA TXOP.

Se menciona que el estándar original IEEE 802.11 obligaba el envío de tramas de asentimiento ACK para cada trama recibida correctamente. En 802.11e estas tramas de asentimiento pasaron a ser opcionales. De forma que cuando se utiliza una política basada en no utilizar ACK, la capa MAC no deberá enviar ACKs por cada trama recibida correctamente. Esto implica que la

fiabilidad de este tráfico se vería reducida, pero mejora el rendimiento general de la capa MAC para tráfico sensible a retardo, tal como sucede con VoIP donde la información tiene valor durante un periodo de tiempo muy corto.

EDCA es un nuevo mecanismo de acceso que mejora el original DCF para proporcionar soporte de calidad de servicio basado en priorización de tráfico. Esta priorización se consigue introduciendo cuatro Categorías de Acceso (AC – Access Categories), las cuales permiten el envío de tráfico asociado a prioridades de usuario, tal como define el estándar (Society, 2012) que describe el Control de Acceso al Medio de los Puentes o Bridges. En la siguiente tabla (CUADRO V) se resumen las prioridades relativas y la tabla de mapeo entre 802.1D y las categorías de acceso 802.11e. En la tabla se puede observar directamente el grado de prioridad otorgado a cuatro tipos de tráfico según los requisitos de QoS (VoIP con la mayor prioridad y el tráfico de descarga de archivos o navegación web con la menor, por ejemplo).

CUADRO V. RELACIÓN ENTRE LAS PRIORIDADES DE USUARIO Y LAS CATEGORÍAS DE ACCESO IEEE 802.11

Priorities	User Priority (802.1D)	Description (802.1D)	Access Categories (802.11e)	Description (802.11e)
Lowest	1	Background	AC_BK	Best Effort
...	2	-	AC_BK	Best Effort
...	0	Best Effort	AC_BE	Best Effort
...	3	Excelent Effort	AC_BE	Video Test
...	4	Controlled Load	AC_VI	Video
...	5	Video	AC_VI	Video
...	6	Voice	AC_VO	Voice
Highest	7	Network Signaling	AC_VO	Voice

Cada categoría de acceso dispone de su propia cola de transmisión caracterizada por unos determinados parámetros (Fig 8) La priorización entre las diferentes categorías se consigue configurando adecuadamente los parámetros de cada cola de acceso

- **Numero de Espacio Arbitrario entre Tramas (AIFSN Arbitrary Inter Frame Space Number)** se corresponde con el intervalo mínimo desde que el medio físico se detecta como desocupado hasta que se comienza la transmisión
- **Ventana de Contienda (CW – Contention Window)** un numero aleatorio se escoge en este rango para lanzar el mecanismo de espera (backoff)
- **Limite de Oportunidad de Transmisión (TXOP limit)** es la duración máxima durante la cual una QSTA puede transmitir tras haber obtenido el TXOP

Cuando los datos llegan al Punto de Acceso de Servicio de información MAC (SAPMAC) la capa MAC de 802.11e se encarga de clasificar adecuadamente los datos y envía la MSDU a la cola correspondiente. Entonces los bloques de información (MSDU) de las diferentes colas (AC) compiten internamente por el EDCA TXOP. El algoritmo de contienda interno calcula la espera (backoff) independientemente para cada cola (AC) según los parámetros descritos AIFSN, CW y un numero aleatorio. El mecanismo de espera es similar al de DCF y la cola con el menor backoff ganará la competición interna. La cola (AC) vencedora competirá externamente por el acceso al medio inalámbrico. El algoritmo de contienda externo no se ha modificado significativamente comparado con DCF excepto que en DCF el backoff y tiempos de espera eran

fijos para un medio físico concreto mientras que en 802.11e estos son variables y se configuran adecuadamente según la cola (AC) correspondiente

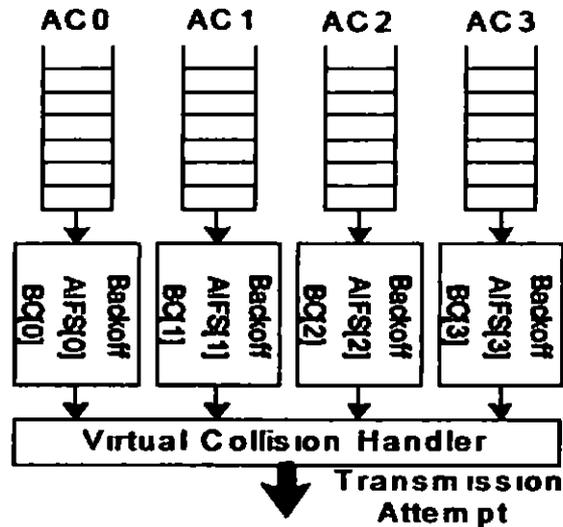


Fig 8 Cuatro colas para las categorías de las Acceso en IEEE 802.11e

A través de un ajuste adecuado de los parámetros de las colas (AC) el rendimiento del tráfico de diferentes colas puede ser ajustado y se puede lograr la priorización de tráfico. Esto requiere un punto de coordinación central (QAP) para mantener un conjunto común de parámetros en las colas y garantizar así un acceso justo entre las diferentes estaciones que componen la red (QBSS). De igual forma, para lograr ajustar la asimetría existente entre el tráfico de subida (QSTA a QAP) y de bajada (QAP a QSTA) un conjunto separado de parámetros EDCA se define exclusivamente para el QAP.

En la Figura (Fig 9) se muestra la arquitectura Genérica de QoS provista por el IEEE 802.11e que es la base fundamental provisión de QoS de la familia

IEEE 802.11 en general y por lo tanto en las redes mesh basadas en el IEEE 802.11s. En la enmienda

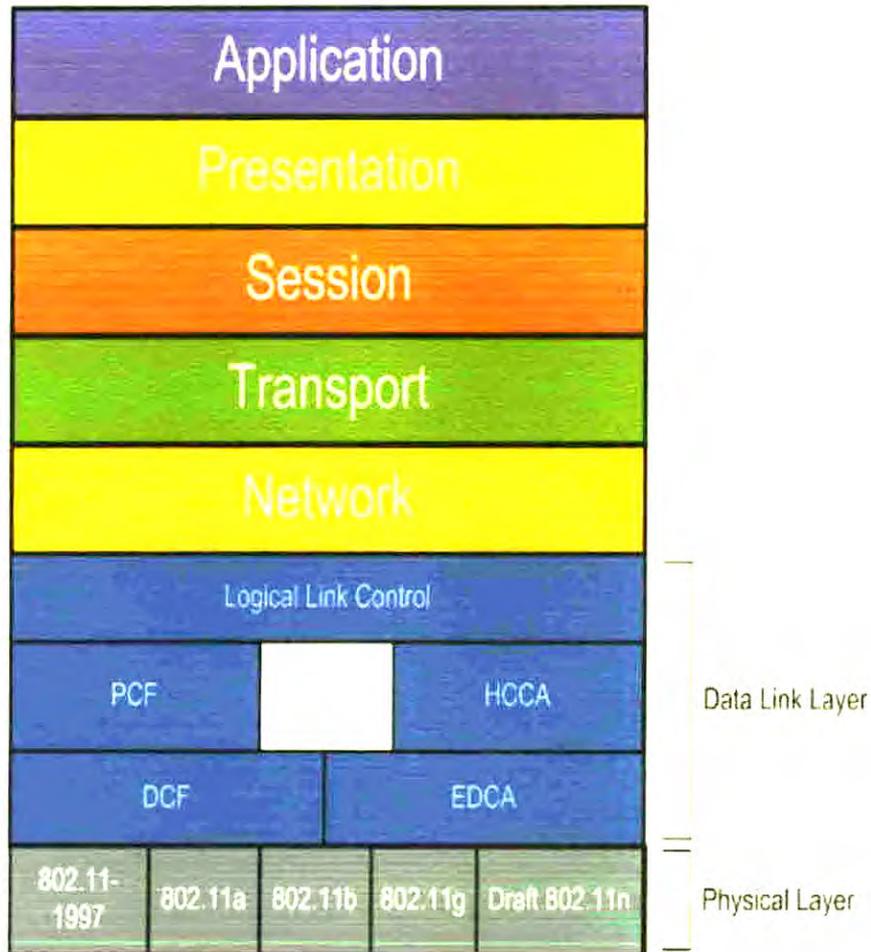


Fig 9. Arquitectura general de soporte de QoS en IEEE 802.11

En el documento del estándar IEEE del 802.11 del 2012 (Society, 2012) se incorporan todas las especificaciones de la familia incluidas el IEEE 802.11e y el IEEE 802.11s.

DISEÑO E IMPLEMENTACIÓN

CAPÍTULO III

DISEÑO E IMPLEMENTACIÓN

3 DESCRIPCIÓN DE LA IMPLEMENTACIÓN

Para poner a prueba el soporte de calidad de servicio QoS en redes mesh, se utilizó una red de prueba (testbed) sencilla que nos permitiera concentrarnos en los aspectos de la Calidad de Servicio QoS implementada en el núcleo de la red conformada por los Mesh STA. El desempeño de la red para servicios multimedia utilizando el estándar IEEE 802.11s ya fue evaluado en conjunto con los colegas del Grupo de Investigación de Tecnologías de Redes Mesh, estos son Joaquín Chung, Grace González, Iván Armuelles y Aidelen Chung. De este estudio se desprende la ventaja del uso del estándar IEEE 802.11s para la implementación de redes mesh de bajo costo en comparación con otras tecnologías como por ejemplo B A T M A.N Advanced. En esa evaluación, también, se tomaron en cuenta los parámetros básicos del desempeño de la red sin el apoyo de ningún mecanismo auxiliar como los vistos en la sección anterior (tasa de pérdidas, ancho de banda, jitter y latencia) para varios saltos (1, 2, 3 y 4). Los parámetros de desempeño evaluados fueron publicados en (J. Chung et al., n.d.) (J. Chung et al., 2013) (Voinov, Cedeño, Chung, & González, 2014). En la figura 10 se muestra el esquema básico del testbed usado en la evaluación de redes mesh por el grupo. Los enrutadores inalámbricos de la mesh (WM Routers o MSTAs) conforman un núcleo de red que se puede implementar para entornos que requieren pocos saltos como una red de interiores de una PYME, una red comunitaria pequeña o una red en entornos rurales con pocos nodos.

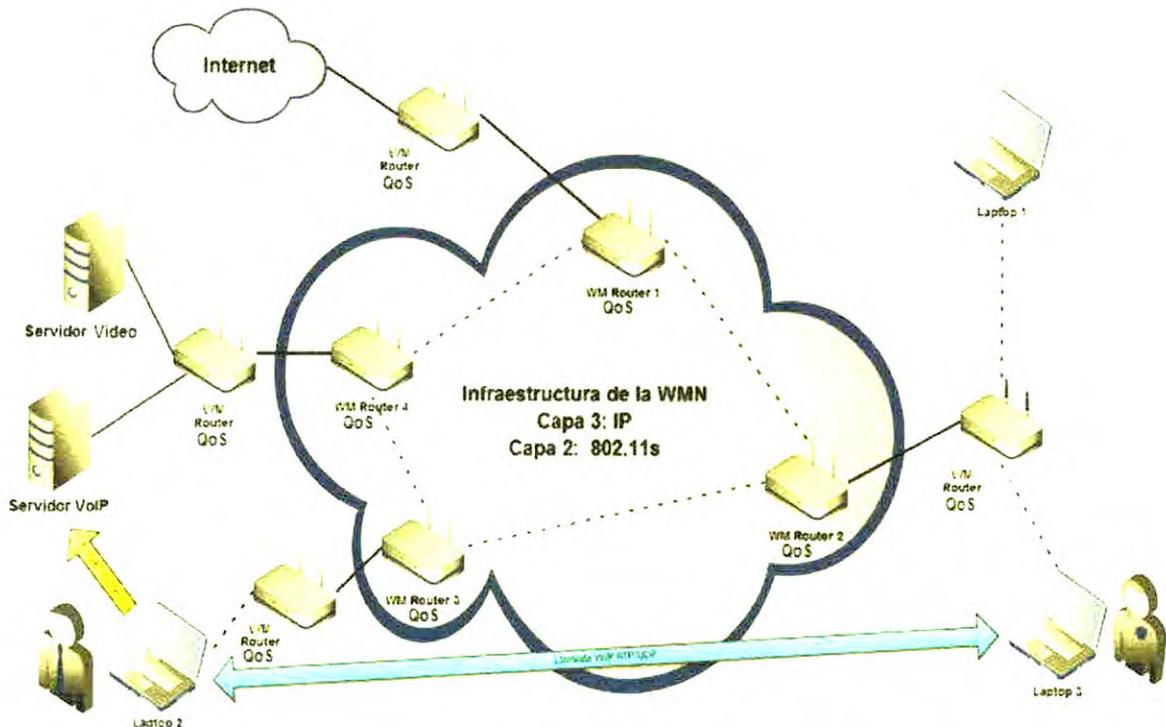


Fig 10. Estructura básica de la red de prueba de evaluación de QoS en Redes Mesh IEEE 802.11s

3.1. REQUISITOS BÁSICOS DEL DISEÑO

Economía y flexibilidad

La línea de investigación seguida en los proyectos del grupo se enfoca en el desarrollo de redes de comunicaciones de bajo costo para facilitar la inserción de la población de escasos recursos a la Sociedad de la Información y del Conocimiento, permitir que las empresas pequeñas y medianas (PYMES) sean más competitivas, o la rápida implementación de redes de comunicaciones ante desastres.

Por tal motivo el Proyecto de Tesis en particular se ha concentrado en la utilización de

- Redes Inalámbricas Multi saltos o de topología en Malla (Wireless Mesh Networks WMN) ya que cuentan con las características enunciadas en la introducción, relativas a la *fácil y rápida* implementación ya que no hay que invertir tiempo en las instalaciones de cableado y los procedimientos de orientación de antenas (sin no son omnidireccionales) son muchos más rápidos
- Equipos de comunicaciones deben ser de un *precio razonablemente bajo*

3 1 1 Fiabilidad y versatilidad.

Otro aspecto funcional en el diseño de las WMNs con los que se desean contar es la *fiabilidad* de los nodos o elementos a utilizar. Afortunadamente se encuentran en el mercado una oferta amplia de equipamiento que cumplen con certificaciones estándares de diseño e implementación que otorgan garantías de funcionamiento

Adicionalmente la *versatilidad* es un aspecto deseable muy importante en la experimentación. Los equipos a utilizar deben contar con la capacidad de ser configurable en varias de sus características seguridad, consumo de energía, parámetros del servicio que prestan, etc

3 1.2 Arquitecturas abiertas basadas en estándares

Para la investigación y la capacidad de desarrollos propios en comunidades de baja capacidad adquisitiva, es necesario trabajar con equipamiento configurable a la medida de las necesidades mediante software. Esto se consigue generalmente a la

disponibilidad de equipamiento con *arquitecturas abiertas* en el mercado tanto de software libre como de hardware abierto. El *hardware abierto* hace referencia a dispositivos con descripciones de diseño e implementación que están disponibles al público no sólo de las piezas que lo conforman, sino también de la programación a bajo nivel de sus dispositivos lógicos y programables por lo tanto está permitida la experimentación en éstos mediante cambios en su funcionamiento o diseño por parte de los usuarios que los adquieren.

El cumplimiento de *estándares* en el diseño de sus componentes su programación y los protocolos de comunicación implementados en los equipos también es una característica deseable para garantizar la compatibilidad entre sistemas.

Los requisitos básicos de diseño para la investigación y diseño de las redes mesh en el proyecto de Tesis (flexibilidad, precio, fiabilidad, versatilidad y arquitecturas abiertas y estándares) se han definido y justificado en esta sección, en la siguiente pasaremos a describir los componentes del diseño.

3.2 COMPONENTES DEL DISEÑO

El testbed general propuesto cuenta con cuatro componentes básicos:

- El terminal de comunicaciones del usuario (inalámbrico o cableado)
- *La red de acceso* conformado por los equipos que brindan la conectividad del terminal de comunicaciones de usuario al resto de la red. En el caso de las comunicaciones inalámbricas éstos deben facilitar una interfaz estándar de

radiocomunicaciones compartida entre el terminal y el nodo que brinda acceso (por ejemplo un punto de acceso inalámbrico).

- La *red de núcleo o backbone* que concentra la infraestructura de encaminamiento principal interconectando a todos los elementos de la red de acceso. En el caso de las redes mesh, el backbone utiliza radioenlaces entre los nodos que lo conforman.
- Los proveedores de *servicios* que incluyen a todas las organizaciones que brindan aplicaciones, contenidos y otros componentes de control de aplicaciones multimedia entre pares como Telefonía IP y Servidores de Videoconferencias, por ejemplo, centrales PBX, directorios, etc. En este último ejemplo, en los terminales de los usuarios se inician y finalizan los flujos de datos, pero requieren el apoyo de plataformas de control de sesión y facturación suministrados por los proveedores de servicios.

En la figura 10 se muestra un esquema que representa una abstracción de la red de prueba o tested representada en la (Fig. 11). Éste es el modelo de referencia que nos permite comprender fácilmente la interrelación entre todos los componentes del sistema.

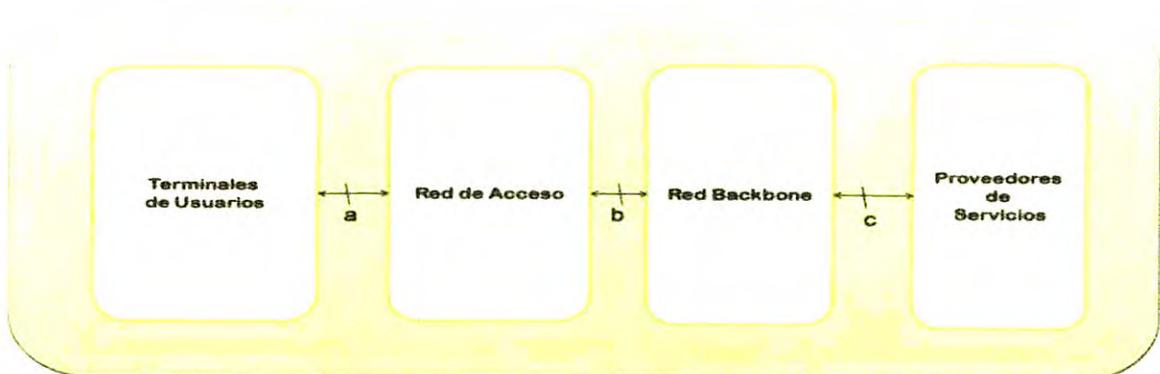


Fig 11. Modelo de Referencia de Red Mesh Inalámbrica

En la figura 11 se muestran un bloque funcional que representa al equipo terminal del usuario. A través de éste el usuario inicia las aplicaciones de red para distintas actividades según su contexto: trabajo, ocio, emergencia, etc. Cada una de estas aplicaciones tendrá requisitos especiales según la aplicación y el contexto del usuario.

Estos requisitos son expresados como parámetros cuantitativos y cualitativos. Como se ha expresado anteriormente en el capítulo anterior, los parámetros cuantitativos están bien definidos para las distintas aplicaciones multimedia ((ITU 2007a)

- Retardo medio en un sentido (latencia de un extremo al otro)
- Jitter o variación del retardo del paquete en un sentido
- Tasa de pérdidas de paquetes
- Velocidad de transmisión en un sentido

Estos requisitos son utilizados para planificar la transmisión de datos entre el terminal del usuario y los elementos de la red de acceso utilizando alguna tecnología de comunicaciones (cableada o inalámbrica) que cumple con estándares con soporte de calidad de servicio. A esta comunicación entre un terminal y el elemento de acceso a la red le hemos denominado "interfaz a". El elemento de la red de acceso establece por su parte mecanismos para el soporte de calidad de servicio al flujo recibido por el terminal, a para satisfacer sus requisitos cuando se traslada a los elementos de la red de núcleo o backbone. Este mecanismo en particular se implementa mediante una tecnología de comunicaciones que en el modelo denominamos "interfaz b". En el caso de la red mesh, los elementos del backbone (los MSTAs) se comunican con otros

MSTA o con los elementos de la red de acceso (STA) mediante canales inalámbricos según el estándar IEEE 802.11s. Finalmente los flujos llegan o parten de los elementos de provisión de servicio mediante diversas tecnologías que hemos denominado “interfaz c”. En nuestra investigación, nos enfocamos en la evaluación de los mecanismos de soporte de calidad de servicio entre los elementos del backbone de la red mesh basado en IEEE 802.11s.

3.3 IMPLEMENTACIÓN DE LA RED

Partiendo de los requisitos de diseño en la investigación se invirtió un tiempo importante evaluando distintos dispositivos de bajo costo y aplicaciones y servicios que pudieran cumplir con los criterios de interoperabilidad, flexibilidad, apertura, etc.

Entre el hardware evaluado para los nodos del backbone de la red mesh (WM Routers) se evaluaron routers inalámbricos tipo Small Office Small Home (SOHO) disponibles en el mercado local: Belkin, Cisco, D-link, TP Link, TRENDnet, y otros cuyos precios oscilan entre B/ 25 000 a B/ 250 000. Entre las diferencias más significativas entre éstos están las compatibilidades con las interfaces de radio (IEEE 802.11a/b/g/n), la capacidad de memoria RAM y Flash, y el tipo de kernel utilizado y su capacidad para ser modificado mediante un firmware de software libre.

De la selección de routers probados se eligió el Router Inalámbrico TP Link TL-WDR4300 con Chipset Atheros AR9344@560MHz, 5 puertos de switch Gigabit con soporte VLAN, RAM de 128MB y Flash de 8MB. Una de las características que nos pareció más conveniente de este equipo ha sido su capacidad de establecer conexiones por su interfaz de radio en dos bandas simultáneamente: en la de 2.4GHz a 300Mbps y

la de 5GHz a 450Mbps Se puede modificar el firmware de fábrica por firmwares de software libre y abierto cumple con los estándares de comunicaciones IEEE 802 11a, IEEE 802 11b IEEE 802 11g y IEEE 802 11n, otros protocolos de seguridad y algoritmos de cifrado que pueden ser implementados en éstos son WEP de 64 y 128 bit, WPA y WPA2 WPA PSK y WPA2 PSK Este modelo cumple con las normas y certificaciones CE FCC IC y RoHS También es compatible con IPv6 (posee la certificación IPv6 Ready Logo Committee del IPv6 Forum) Como se puede observar los routers TP Link TL WDR4300 ver 1.2 cumplen con los requisitos de diseño establecidos en la sección anterior

Para disminuir el ruido causado por equipos que estén usando la frecuencia de 2.4GHz en la zona de laboratorio se configuraron los routers inalámbricos del backbone con enlaces de 5GHz En éstos instalamos el firmware OpenWRT Backfire 10.03.1 que es de software libre permitiendo la personalización completa de las aplicaciones en el router inalámbrico e implementando protocolos de enrutamiento en este caso el estándar 802.11s, y tiene soporte de QoS (Fig. 10) Para probar los servicios multimedia se usaron servidores y aplicaciones clientes y herramientas de análisis de tráfico de software libre para mantener el coste de implementación bajo En el (CUADRO VI) se muestra un listado de las herramientas utilizadas

CUADRO VI. COMPONENTES DE SOFTWARE

#	Nombre	Tipo	Descripción
1	OpenWRT 10.03	Firmware	Firmware para routers inalámbricos basado en Linux con soporte para 802.11s
2	Ubuntu Server 12.04	Sistema Operativo	Sistema operativo Linux para los servidores de video
3	VMWare ESXi 5.1	Virtualización	Host para creación de máquinas virtuales
4	Red5 Server 1.0	Servidor de difusión de video	Servidor para difusión de video (video streaming) basado en flash. Soporta los codecs de video: ScreenVideo, On2 VP6, Sorenson H.263 y H264; y los audio codecs: ADPCM, NellyMoser, MP3, Speex y AAC.
5	VLC	Reproductor/server y «framework» multimedia de código abierto, multiplataforma, que reproduce la mayoría de archivos multimedia y realiza streaming en diversos protocolos de transmisión.	Soporta los siguientes video codecs: MPEG-1/2, DIVX (1/2/3), MPEG-4 ASP, DivX 4/5/6, H.261, H.263/H.263i, H.264/MPEG-4 AVC, MJPEG (A/B), WMV 1/2, WMV 3/WMV-9/VC-1, DV (Digital Video), On2 VP3/VP5/VP6 y otros. Audio codecs soportados: MPEG Layer 1/2, MP3 - MPEG Layer 3, AAC - MPEG-4 part3, Vorbis, AC3 - A/52 (Dolby Digital), E-AC-3 (Dolby Digital Plus), MLP / TrueHD">3, DTS, WMA 1/2,WMA 3 1, y otros. Protocolos de transmisión soportados: UDP/RTP Unicast, UDP/RTP Multicast, HTTP / FTP, MMS, TCP/RTP Unicast, DCCP/RTP Unicast
6	Iperf, Jperf	Herramientas para generar tráfico en la red.	Herramienta para medición de ancho de banda en redes de datos

En la figura 12 se tiene una representación de la maqueta final utilizada para evaluar las prestaciones de QoS entre los routers IEEE 802.11s en el canal de 5GHz.

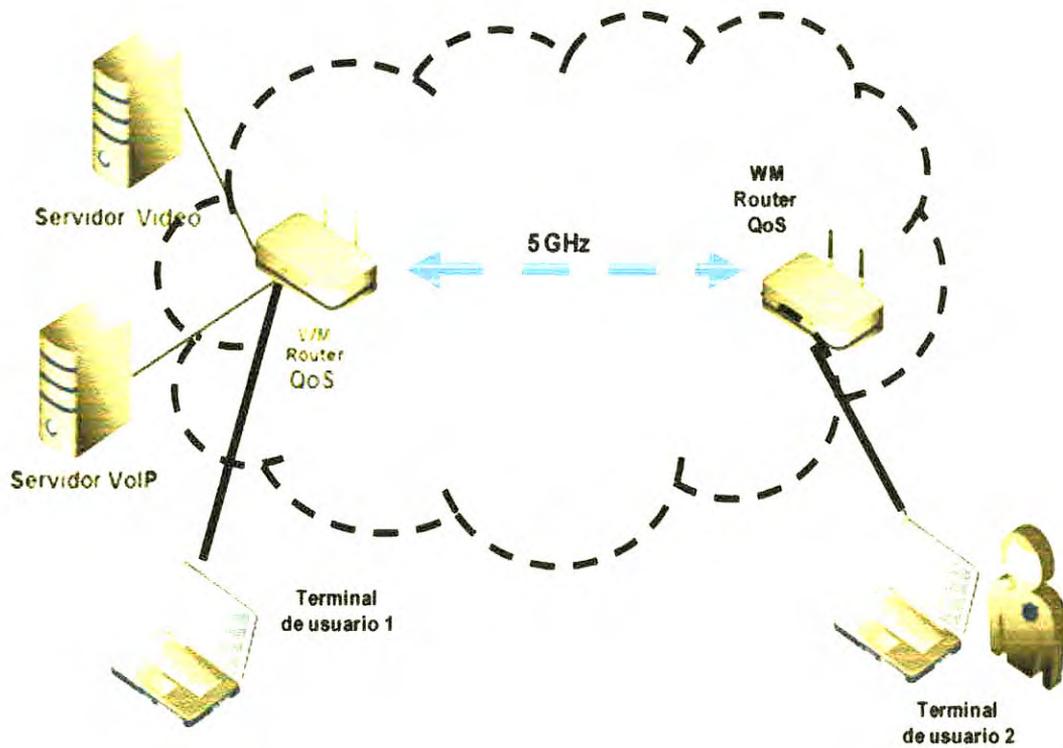


Fig 12. Red de Prueba de Soporte de QoS entre nodos de la Red Mesh

En la figura 13 muestra la implementación de la maqueta utilizada para evaluar las prestaciones de QoS entre los routers IEEE 802.11s en el canal de 5GHz.



Fig 13. Implementación de la maqueta

El análisis en el canal de 5GHz es el uno de los principales objetivos de nuestra investigación con el propósito de conocer cómo puede verse mejorada la transmisión de tráfico multimedia con soporte de QoS entre los nodos de un salto en una red mesh inalámbrica en comparación con el tráfico soportado sin mecanismos de QoS.

3.4. SOFTWARE IMPLEMENTADO EN LOS ROUTERS

3.4.1. OpenWRT:

Para la implementación de la red mesh inalámbrica se utilizó el firmware OpenWRT en los routers inalámbricos. Se prefiere OpenWRT sobre otros firmwares para routers inalámbricos por su estructura modular, que permite la instalación de diversos paquetes, personalizando la configuración del equipo.

3.4.2 Paquete qos-scripts

El paquete qos-scripts permite desplegar calidad de servicio en OpenWRT utilizando un sólo archivo de configuración. Este archivo permite configurar de forma sencilla las interfaces del router que aplican calidad de servicio (QoS) las políticas que se aplican a los paquetes y la prioridad en las colas de las interfaces. Debido a que OpenWRT es un sistema operativo basado en Linux también es posible configurar la calidad de servicio utilizando el paquete qos-scripts ya que simplifica las tareas de configuración y permite un despliegue rápido y eficiente de la calidad de servicio en la red mesh inalámbrica. Activamos la calidad de servicio en dos formas utilizando la línea de comando o la interfaz web conocida como Luci

```

arturo@arturo-HP-Compaq-dc5750-Small-Form-Factor: ~
● option packetsize 1000
● option avgrate 10

config class "Express"
  option packetsize 1000
  option max-size 800
  option avgrate 50
  option priority 10

config class "Normal"
  option packetsize 1500
  option packetdelay 100
  option avgrate 10
  option priority 5

config class "Normal_down"
  option avgrate 20

config class "Bulk"
  option avgrate 1
  option packetdelay 200
root@WHR-079-04:~# /etc/init.d/qos enable
root@WHR-079-04:~# /etc/init.d/qos start
root@WHR-079-04:~#

```

Fig 16 Activación de QoS en el router por línea de comando

ANÁLISIS DE LOS RESULTADOS

CAPÍTULO IV

ANÁLISIS DE LOS RESULTADOS

4 ANÁLISIS DE LOS RESULTADOS OBTENIDOS

Luego de haber implementado la red de prueba, configurando todos sus componentes de hardware y software procedimos a realizar distintas pruebas sobre las capacidades de soporte de QoS en redes mesh IEEE 802.11s implementado con el firmware

Las pruebas consistieron en lo siguiente

- Caracterización del canal entre dos nodos del núcleo de red IEEE 802.11s
- Verificación del QoS Script y mecanismo de QoS para IEEE 802.11s en el firmware OpenWRT
- Evaluación de la capacidad de tratar conexiones de VideoStreaming con soporte de QoS
- Evaluación de la capacidad de tratar conexiones simultáneas de flujos de VoIP con soporte de QoS

4.1 CARACTERIZACIÓN DEL CANAL

Para caracterizar el canal se requiere determinar los siguientes parámetros: ancho de banda, jitter y pérdidas de paquetes.

A continuación caracterizamos un canal utilizando dos métodos: inyección de tráfico tipo TCP e inyección de tráfico tipo UDP con la herramienta iperf.

4 1 1 Caracterización del canal mediante la inyección de tráfico TCP

Se realizó la medida del canal total ocupado tomando los datos con la herramienta iperf Utilizamos un flujo TCP con iperf para conocer el ancho de banda que ocupa el canal con este tipo tráfico El comportamiento de un flujo TCP debido al su forma de comportamiento norma, consiste en ocupar todo el canal disponible aunque sea con uno sólo flujo por lo que el ancho de banda es el unico parámetro que devuelve el iperf cuando se utiliza el tráfico TCP Para tomar 10 mediciones de un flujo de un minuto de duración Este fue el unico flujo monitoreado con detalle y sus resultados se encuentran en el Cuadro VII Esta mediada no presentó variaciones y se mantuvo constante Las pruebas adicionales siguientes concuerdan con estos resultados

CUADRO VII CANTIDAD DE MUESTRAS CON TRAFICO TCP

# Muestras	Ancho de Banda(Mbrrts/sec)
1	18 50
2	18 50
3	18 50
4	18 50
5	18 50
6	18 50
7	18 50
8	18 50
9	18 50
10	18 50
promedio	18.50

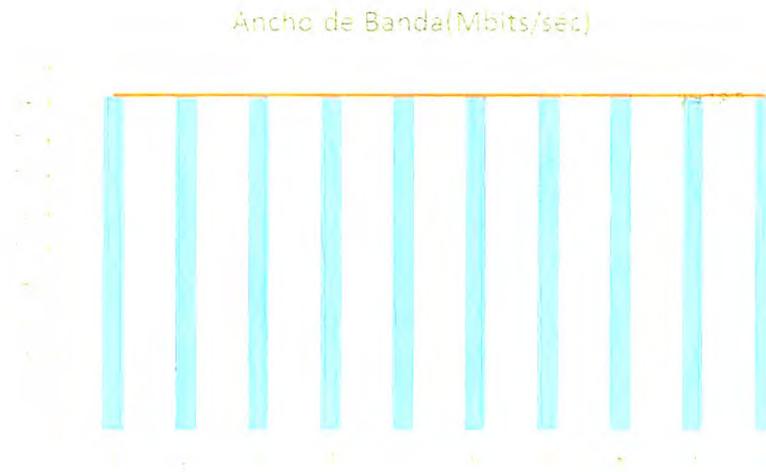


Fig 18. Canal total ocupado

Resultado: El ancho de banda que ocupa el canal es de 18.5 Mbits/sec.

4.1.2. Caracterización del canal mediante la inyección de tráfico UDP

Utilizamos un flujo UDP con iperf para conocer el ancho de banda máximo que ocupa el canal sin tráfico mediante este método. El iperf devuelve los siguientes parámetros con un tráfico UDP: ancho de banda, jitter y pérdida de paquete.

El iperf permite configurar una velocidad de transmisión (ancho de banda) del flujo UDP inyectado, así como también el tamaño del paquete (opcional) y la duración del flujo (opcional). Se realizaron las mediciones inyectando tráfico UDP con ancho de banda superior al evaluado mediante la inyección de tráfico TCP, específicamente a una velocidad de 80 Mbps (4.3 veces superior al ancho de banda del canal reportado en el experimento anterior). Al someter a este “estrés” el canal, se producen evidentemente altas pérdidas de paquetes UDP (que el iperf puede

detectar y reportar), pero se puede conseguir el ancho de banda máximo que soporta el canal. En el cuadro VIII se encuentran los resultados de este experimento.

CUADRO VIII. MUESTRAS CON EL TRÁFICO UDP

# Muestras	Ancho de Banda(Mbits/sec)	Jitter (ms)	Perdida de Paquete (%)
1	21.70	10.96	78.00
2	22.30	11.07	78.00
3	21.60	11.27	78.00
4	21.60	11.26	78.00
5	22.40	0.72	78.00
6	21.80	0.58	78.00
7	22.30	11.09	78.00
8	22.40	0.53	78.00
9	21.80	0.57	78.00
Promedio	21.99	6.45	78.00
Desviación estándar	0.35	5.23	0.00

Resultado: El ancho de banda máximo que puede ocupar el tráfico en el canal estudiado es de 21.99 Mbits/sec (± 0.35 Mbits/sec). Y el jitter promedio del canal utilizado, con una inyección de tráfico superior al que soporta el canal es de 6.45 ms, y el porcentaje de pérdida de paquetes máximo fue del 78% .

En la Figura 19 se muestra una comparación gráfica entre los datos recogidos en ambos experimentos para calcular la capacidad del canal entre los nodos del nodo del IEEE 802.11s. En la Cuadro IX se muestran los datos correspondientes a los graficos.

CUADRO IX. MÁXIMO Y MÍNIMO CANAL

	MIN (Mbits/sec)	MAX (Mbits/sec)	PROMEDIO (Mbits/sec)	Desviación Estándar
TCP	18.50	18.50	18.50	0.00
UDP	21.60	22.40	21.99	0.35

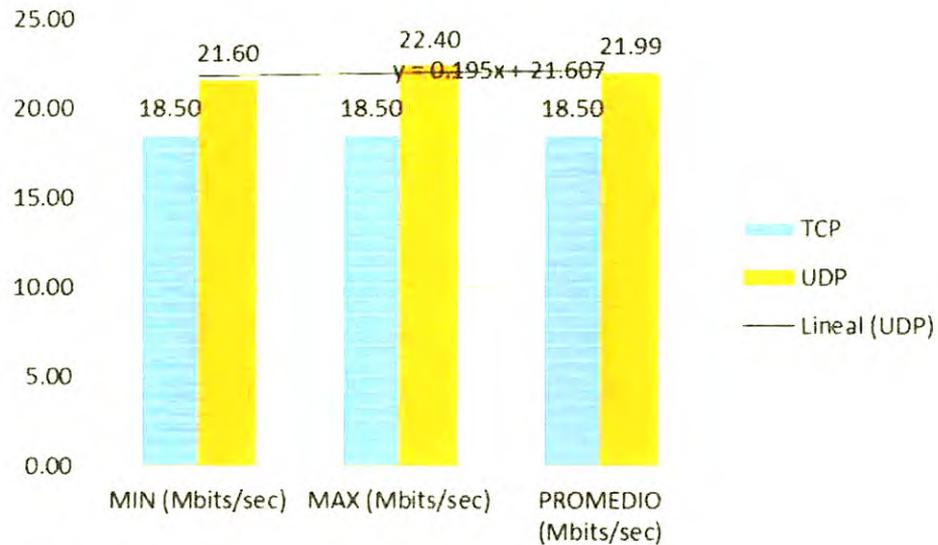


Fig 19. Ancho de banda Máximo y Mínimo que puede ocupar el canal.

Verificando un límite superior de capacidad del Ancho de Banda o Capacidad del Canal es de 21.99 Mbits/sec pero con pérdidas significativas de paquetes, definimos como **Capacidad del Canal regular los 18.5 Mbits/sec** detectados en el experimento anterior.

Este valor encontrado no es inconsistente. Recordemos que el módulo del IEEE 802.11s en el OpenWRT está configurado para IEEE 802.11g (velocidad teórica de 54 Mbits/sec) y que una tercera parte de la capacidad de ésta esta es la que realmente se registra en la práctica por la alta señalización del canal a bajo nivel. Como práctica de

ingeniería de tráfico se recomienda manejar un tráfico 10 % inferior a este máximo de 18.5 Mbits/sec (o sea 16.65 Mbits/sec)

4.2 Evaluación del mecanismo de soporte QoS en IEEE 802.11s

Para verificar funcionamiento del soporte de calidad de servicio brindada por el firmware OpenWRT que se configura mediante el archivo qos-scripts se utilizaron los mismos routers inalámbricos con éste firmware que son extremos de un enlace en la red de núcleo de mesh inalámbrica 802.11s. Así es que ideamos el siguiente experimento aprovechando que la herramienta de iperf nos permite configurar distintos parámetros de flujo UDP como la velocidad de transmisión, varianza del retardo (jitter) y pérdida de paquetes. Como se pueden controlar estos parámetros primero generamos un tráfico UDP incrementando poco a poco su velocidad de transmisión sin brindarle soporte de QoS y la segunda parte se desarrolló la misma actividad pero con el soporte de QoS. En ambos casos se transmitió un flujo de video streaming como tráfico de fondo pero no se le otorgó soporte de QoS en ninguno de las dos pruebas.

4.2.1 Evaluación cuantitativa del canal de IEEE 802.11s sin QoS

Primero realizamos las pruebas sin el soporte de calidad de servicio para tener un punto de referencia. Se realizó un video streaming que debía competir con tráfico adicional, mediante un flujo UDP. El UDP fue inyectado con velocidades de transmisión iguales a 50% 75% 90% 95% de la capacidad del canal calculado (18.5 Mbits/sec). El objetivo fue registrar cómo variaba el flujo UDP en el destino.

evaluando la velocidad de recepción del flujo, su variación del retardo (jitter) y el porcentaje de pérdidas registradas. La sesión de video streaming simultánea se realizó con el contenedor TS, además de RTP y UDP para su transporte. La sesión de video streaming funcionó como tráfico de fondo para competir con los recursos del canal. En el Cuadro X se muestran los valores recogidos de este experimento. El flujo UDP en cada ocasión tuvo una duración de 30 segundos y se tomaron muestras cada segundo. En la tabla se muestran los valores promedios.

CUADRO X. EVALUACIÓN CUANTITATIVA DEL TRÁFICO UDP EN EL CANAL DE IEEE 802.11S SIN SOPORTE DE QOS

Tráfico Inyectado (Mbits/sec)	Capacidad del Canal ocupado (%)	Velocidad de Transmisión medida en el destino (Mbits/sec)	Jitter (ms)	Paquetes Perdidos (%)
9.00	50	5.31	3.81	35.79
13.50	75	1.09	2,72	91.97
16.65	90	1.61	8.97	96.59
17.57	95	1.83	2.95	89.52

En el Cuadro XI se muestran los valores recogidos de este experimento para la desviación estándar.

CUADRO XI. EVALUACIÓN CUANTITATIVA DEL TRÁFICO UDP EN EL CANAL DE IEEE 802.11S SIN SOPORTE DE QOS. DESVIACIÓN ESTÁNDAR

Tráfico Inyectado (Mbits/sec)	Capacidad del Canal ocupado (%)	Velocidad de Transmisión medida en el destino (Mbits/sec)	Jitter (ms)	Paquetes Perdidos (%)
9.00	50	3.19	3.12	33.76
13.50	75	0.46	1.31	3.49
16.65	90	4.91	5.86	2.86
17.57	95	0.63	3.16	3.67

4.2.2. Evaluación cuantitativa del canal de IEEE 802.11s con QoS

Como prueba definitiva, se realizó el experimento anterior activando el mecanismo de QoS de OpenWRT. En el scrip de QoS que permite esta activación, se le dio prioridad al flujo UDP tipo “Express” identificando este flujo con el protocolo (UDP) y el puerto (“1234”). Al flujo de video en TS en RTP/UDP no se le dio ningún tratamiento. Los resultados se muestran en el cuadro XII.

CUADRO XII. EVALUACIÓN CUANTITATIVA DEL TRÁFICO UDP EN EL CANAL DE IEEE 802.11S CON SOPORTE DE QOS

Tráfico Inyectado (Mbits/sec)	Capacidad del Canal ocupado (%)	Velocidad de Transmisión medida en el destino (Mbits/sec)	Jitter (ms)	Paquetes Perdidos (%)
9.00	50	7.6	1.58	14.11
13.50	75	8.71	2.02	33.42
16.65	90	12.48	0.97	24.74
17.57	95	11.22	1.22	35.49

Los resultados de la desviación estándar se muestran en el cuadro XIII.

CUADRO XIII. EVALUACIÓN CUANTITATIVA DEL TRÁFICO UDP EN EL CANAL DE IEEE 802.11S CON SOPORTE DE QOS. DESVIACIÓN ESTÁNDAR

Tráfico Inyectado (Mbits/sec)	Capacidad del Canal ocupado (%)	Velocidad de Transmisión medida en el destino (Mbits/sec)	Jitter (ms)	Paquetes Perdidos (%)
9.00	50	1.99	1.09	21.07
13.50	75	3.59	1.55	26.66
16.65	90	2.61	0.40	15.69
17.57	95	3.37	0.61	18.87

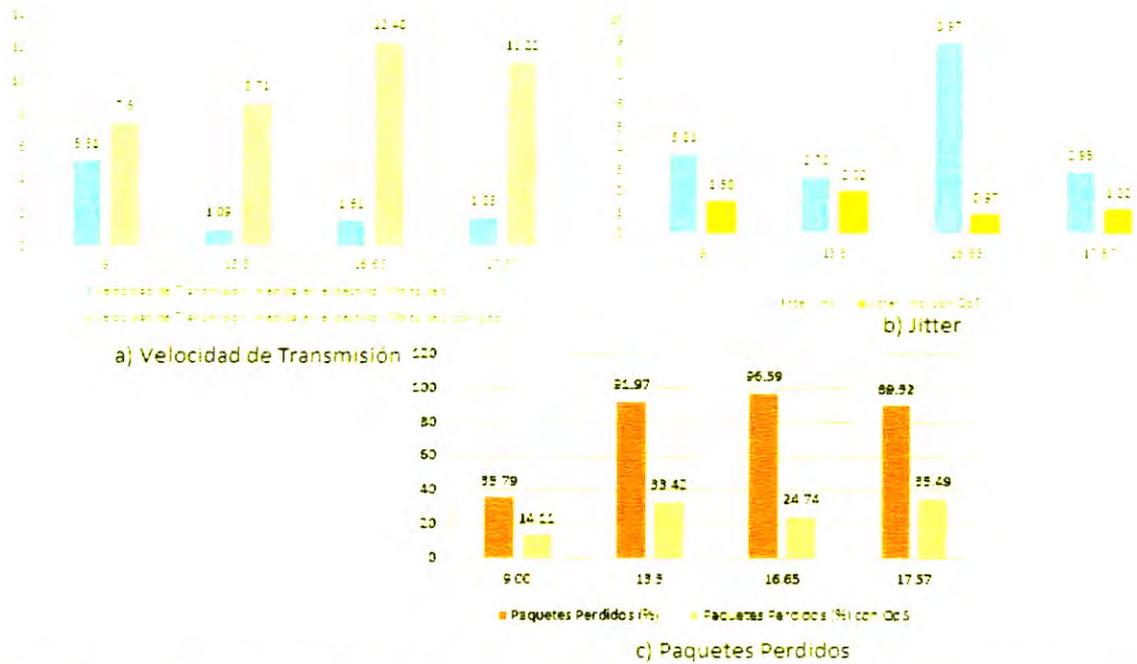


Fig 20. Gráfica comparativa de la evaluación cuantitativa del tráfico UDP en el canal de IEEE 802.11s con y sin soporte de QoS.

Resultado: Como se puede observar comparando los resultados de la figura 20 y las tablas X y XII, se hace evidente que la activación del mecanismo de QoS del estándar IEEE 802.11s hace un efecto positivo en el tratamiento del flujo UDP privilegiado. Al activar el soporte de QoS, la pérdida de paquetes del flujo UDP y el jitter disminuyeron sustancialmente, además de que la diferencia entre velocidad del flujo recibido y el flujo emitido es menor que en el caso en el que no se utiliza soporte de QoS alguno.

conferencias. Estos servicios son muy demandados en aplicaciones de tele educación, tele medicina, negocios y ocio entre otros.

4.3.1 Evaluación de la configuración de Video Streaming con QoS

Para la evaluación de transmisión de Video Streaming se procedió a realizar diversas pruebas basadas en experiencias previas que se realizaron sin QoS. Por ejemplo, en (Gonzalez, 2013) se calculó que la mejor combinación de Codecs de Video y Audio para la transmisión en redes mesh sin soporte de QoS es H.264/MPEG4 y AAC/MP3 del contenedor TS (MPEG transport stream o MPEG TS). Utilizando esta definición decidimos verificar cuáles son los parámetros de QoS que deben ser provistos por la red de prueba para brindar la mejor percepción de servicio.

De esta forma, se realizó el streaming de segmentos de un video comercial con características visuales muy dinámicas. El video de prueba fue la película 'Rápido y Furioso 2' (2 Fast 2 Furious¹). Esta fue configurada para ser transmitida por la red de prueba con el servidor VLC usando el contenedor TS mencionado. La resolución de las imágenes fue de 448x336 píxeles y se transmitió a una tasa constante de 23.97 cuadros por segundo (Figura 20). El codec de audio realizó el muestreo a 48KHz (según lo establecido por el estándar). Debido a la variabilidad de las características del video, la tasa de transmisión de bits fluctuaba los 1.0 Mbit/s (Figura 21). Esta configuración para la transmisión permaneció constante en las dos experiencias: primero transmitiendo el video sin soporte de QoS.

¹ "2 Fast 2 Furious" Ficha Técnica <http://www.imdb.com/title/tt0322259/>

simultáneamente con tráfico de fondo con velocidad incremental, y segundo, transmitiendo el video con soporte de QoS con el mismo tráfico de fondo.

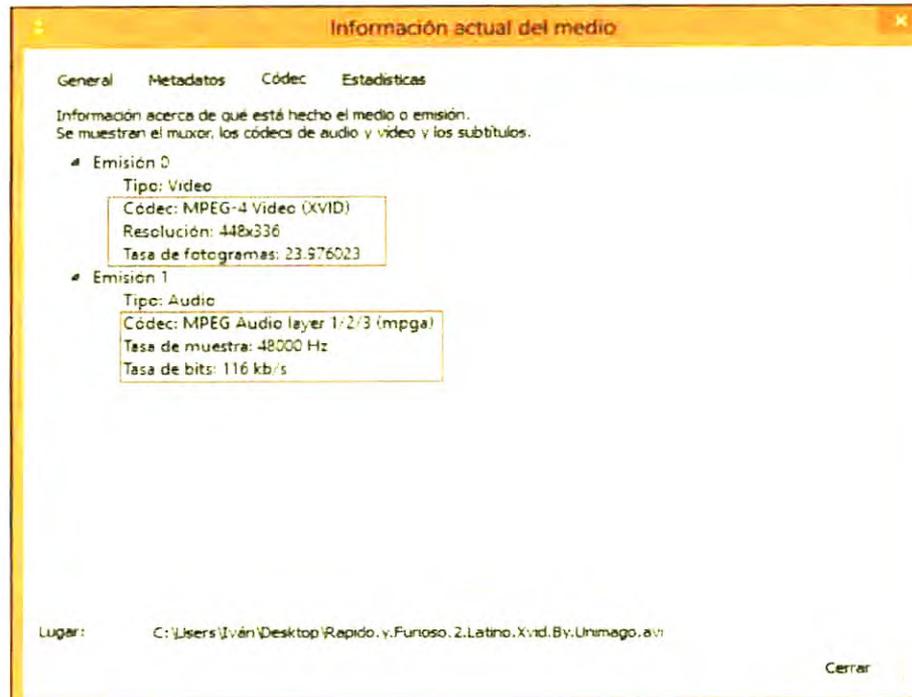


Fig. 20. Información de los codecs de video streaming de prueba suministrada por VLC

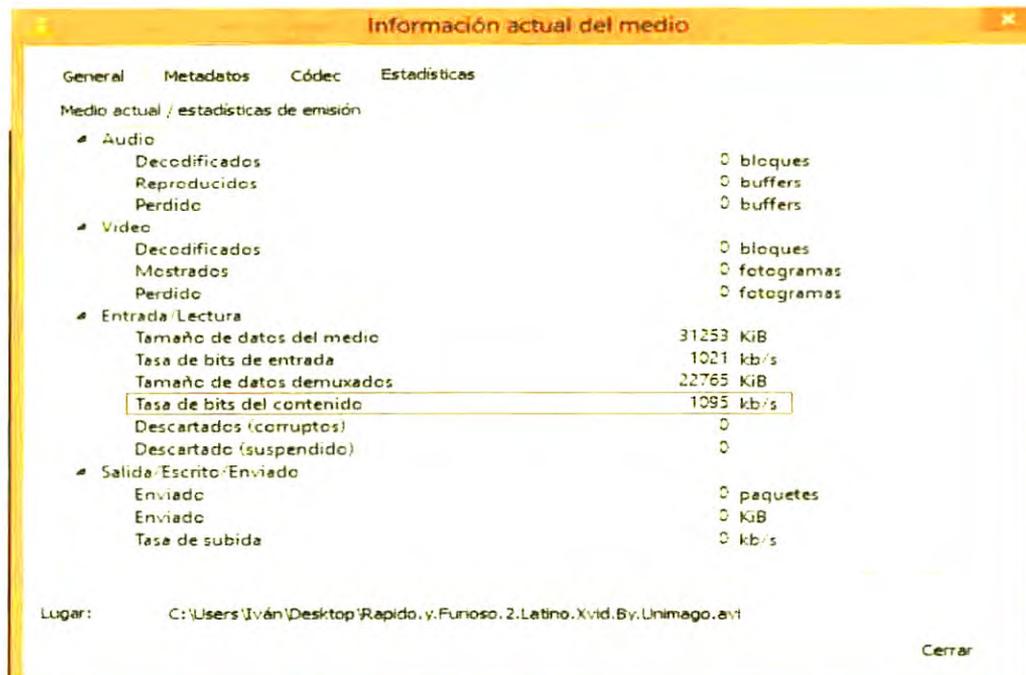
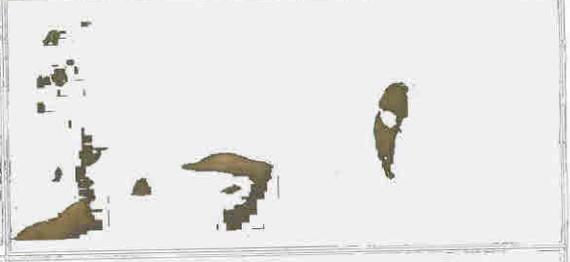
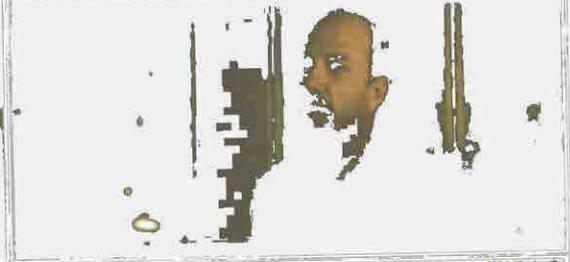
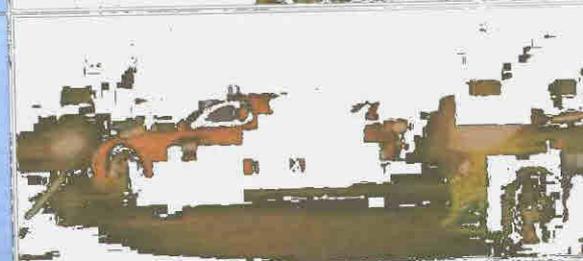
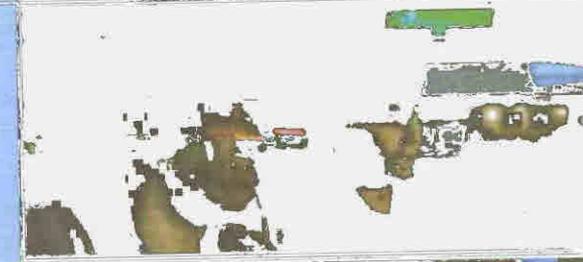
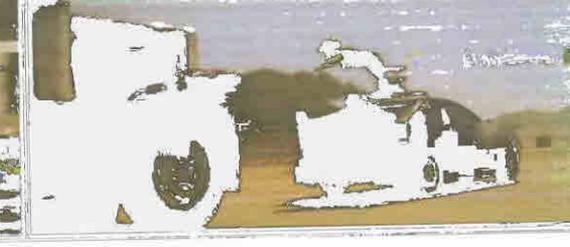


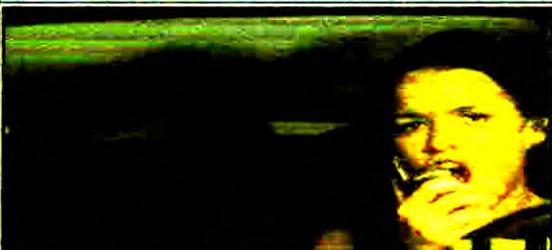
Fig. 21. Información de la tasa de transmisión del video streaming de prueba suministrada por VLC

En la Cuadro XIV se muestran los resultados comparativos de la percepción de ambos pruebas de video streaming sin y con soporte de QoS. El video fue emitido a la red con los protocolos especificados en (Gonzalez, 2013), estos son, sobre Real-time Transport Protocol (RTP) sobre User Datagram Protocol (UDP), o sea RTP/UDP/IP. En cada fila se muestra el aspecto del video con un tráfico de fondo tipo UDP cuyo ancho de banda en el canal aumentaba en 200 Kbit/s. La primera fila (con tráfico de fondo de 100 Kbit/s) sirve de referencia de base. El scrip de QoS se configuró para dar soporte tipo “Express” según la clasificación de prioridades de soporte de QoS del script de OpenWRT.

CUADRO XIV. SOPORTE DE QoS PARA VIDEO STREAMING

Tráfico de Fondo (Kpbs)	Video Streaming Sin QoS	Video Streaming Con QoS
100		
200		

Tráfico de Fondo (Kbps)	Video Streaming Sin QoS	Video Streaming Con QoS
400		
800		
2000		
10000		
12000		
14000		

Tráfico de Fondo (Kpbs)	Video Streaming Sin QoS	Video Streaming Con QoS
1600		
1800		

Resultado: Como se puede observar comparando los resultados de Cuadro XIV, se hace evidente que la activación del mecanismo de QoS del estándar IEEE 802.11s hace un efecto positivo en el tratamiento del flujo video Streaming privilegiado cuando éste debe competir por el canal con tráfico de fondo UDP con valores iguales o superiores a los 600Kbps. Esta es un advertencia del especial cuidado que se debe tener al configurar redes mesh IEEE 802.11s según la versión actual del OpenWRT con tráfico multimedia con las características elegidas (TS/RTP/UDP/IP/IEEE802.11s). Estos resultados pueden variar significativamente al cambiar el códec o la resolución de las imágenes.

4.3.2. Evaluación de la configuración de Video conferencia con QoS

La siguiente prueba de video streaming consistió en la utilización de servicio de video conferencia utilizando Red5. Este servidor fue instalado en una máquina virtual del servidor mostrado en la Figura 22. En la Figura 23 se muestra una captura de la interfaz gráfica del servidor VMware donde se instalaron los

servidores virtuales Elaxtix (para el experimento con VoIP) y para el actual experimento con Red 5. La Figura 24 muestra una captura de la interfaz del servidor Red5 y la Figura 25 muestra una captura de un cliente recibiendo el video streaming de prueba. Finalmente, en la Figura 26 se muestra uno de los clientes de una videoconferencia de prueba con soporte de QoS y en la Figura 27 se muestra el mismo cliente de la videoconferencia de prueba sin calidad de servicio.

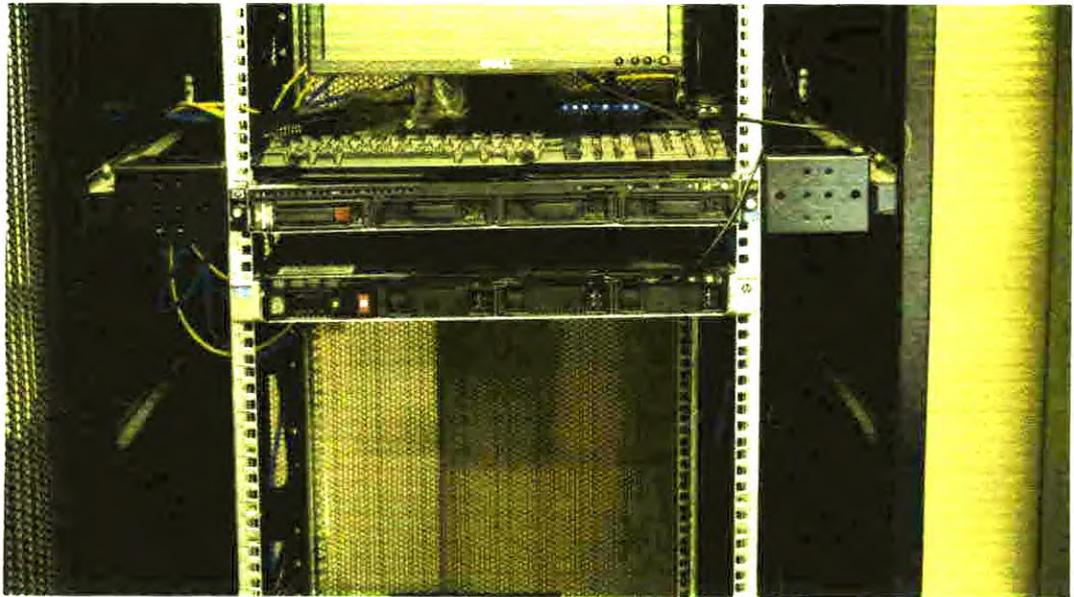


Fig. 22 Gabinete de Servidores de virtualización

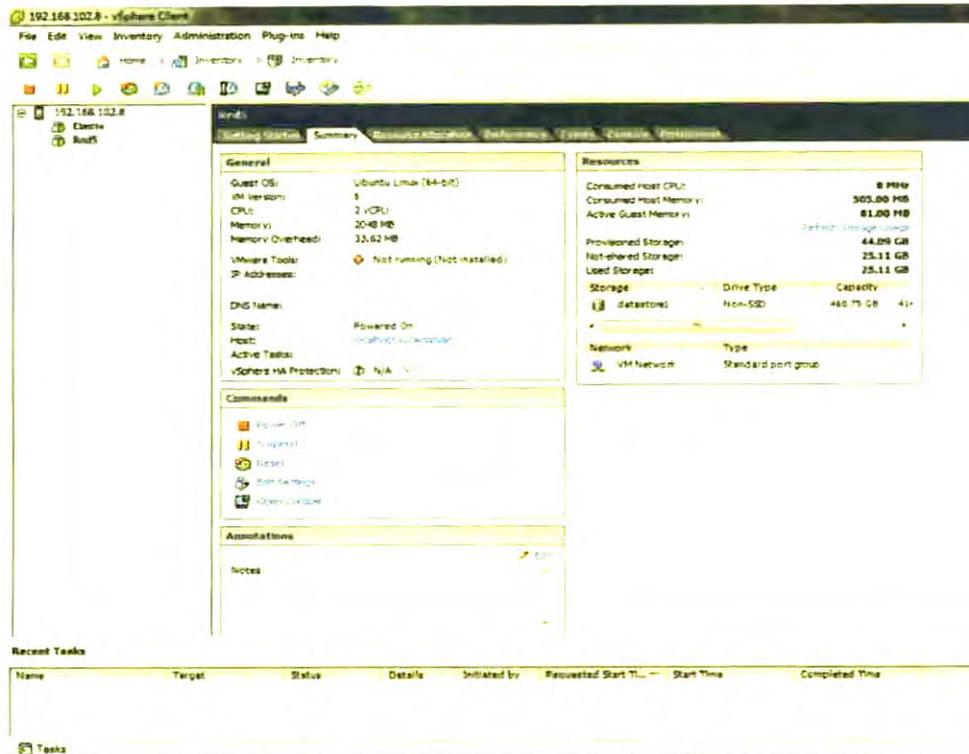


Fig. 23. Servidor donde se instalaron los servidores virtuales de VoIP (Elastix) y video streaming y video conferencias (Red5).

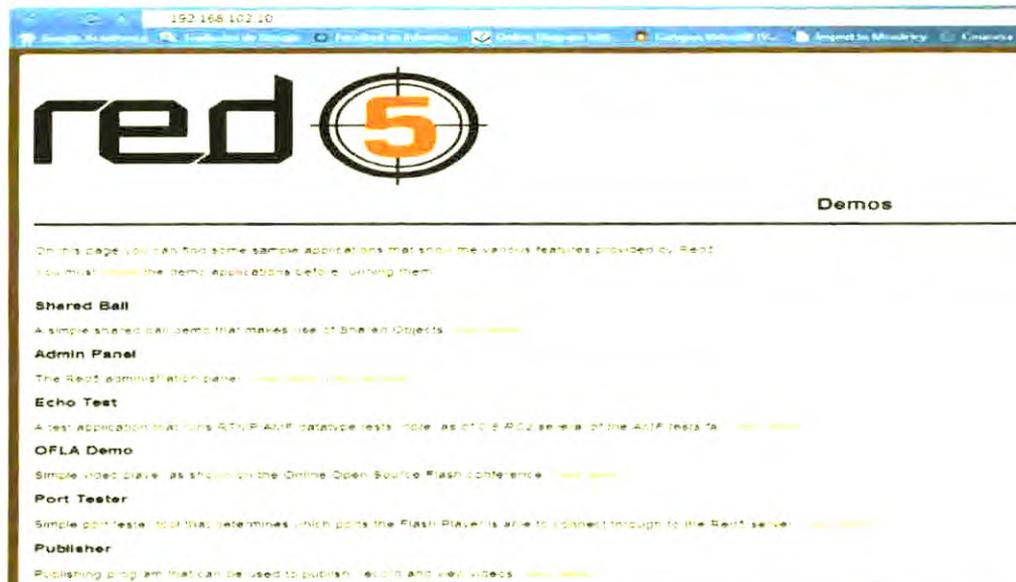


Fig 24. Captura de pantalla de Servidor virtual Red5

192.168.102.10

Google Académico Traductor de Google Facultad de Inform... Online Diagram Soft... Campus Virtual@ V...

red5

[links]

size	name	last modified
2244285	movie_red5.flv	24/12/2012 11:11
2137497	sample-connection.flv	24/12/2012 11:11

[links]

[download]

File Name: file.flv #

File Size: 151.116 KB

Time: 12/12/2012 11:11:11

File Type: application/x-flv

[output]

```

red5: Connection successful
177017: connection successful
177018: lastConnectionStatus
mode = 1
fps = 25
capabilities = 31
data =
  version = 10.0.1101
  type = h264
description = Connection successful
code = lastConnection.Connected.Success
level = status
177019: connectionSuccessful
177020:

```

Fig 25. Captura de pantalla de Cliente recibiendo video streaming

192.168.102.10

Google Académico Traductor de Google Facultad de Inform... Online Diagram Soft... Campus Virtual@ V... Signal to Marketing... Cursos... Reportar Error...

Monitor View Publish

LOG

```

2012-12-24 11:11:11 [INFO] [192.168.102.10] [192.168.102.10] [192.168.102.10] [192.168.102.10]

```

stream079

Stop Pause

Settings Video Audio

192.168.102.10

192.168.102.10

480p

128kbps

Audio VideoData NetStream

Video Conference NetConnection

Fig. 26 Captura de pantalla de video conferencia con Calidad de servicio.



Fig. 27 Captura de pantalla de video conferencia sin Calidad de servicio.

El contenedor utilizado por Red5 es FLV (Flash Video). FLV es el contenedor de facto utilizado para el web streaming en Internet. Éste usualmente contiene material codificado con formato de video compresión Sorenson Spark o Compresión de Video VP6. Las versiones más recientes de Flash Player, producto de Adobe Systems y MainConcept, soporta también H.264 para el video streaming y HE-AAC para la distribución del audio. VP6 provee una calidad visual mejor que Sorenson Spark, especialmente cuando se usan tasas de transmisión muy bajas. Por otra parte, la carga computación en los sistemas finales es mayor y sus algoritmos son más complejos. VP6 produce una salida que tienen una Relación Señal a Ruido Pico (PSNR) que son mejores que los provistos por H.264 sobre un amplio rango de velocidades desde el dial-up (28.8 Kbit/s) hasta tasas de DVD y HD (de 6 a 8 Mbit/s).

Se estableció una videoconferencia y se tomaron las medidas de la misma en el receptor o destino del flujo mientras se inyectaba tráfico UDP con velocidades

incrementales obteniendo resultados positivos para esta herramienta como los que se muestran en el Cuadro XV.

CUADRO XV. CUADRO PRUEBAS CON SERVIDOR DE VIDEO RED5

Prueba	BW UDP iperf (Mbits/s)	Sin QoS	Con QoS
1	12	Video se ve bien para imágenes estáticas, se pixela levemente con el movimiento	Video se ve bien para imágenes estáticas, se pixela levemente con el movimiento
2	15	Video se ve bien para imágenes estáticas, se pixela levemente con el movimiento	Video se ve bien para imágenes estáticas, se pixela levemente con el movimiento
3	18	La imagen se pixela cuando está estática, se pierde la mitad de la pantalla cuando hay movimientos	Video se ve bien para imágenes estáticas, se pixela levemente con el movimiento
4	20	La imagen se pixela cuando está estática, se pierde la mitad de la pantalla cuando hay movimientos	La imagen se pixela cuando está estática, se pierde la mitad de la pantalla cuando hay movimientos
5	22	Se pierde la mitad de un frame constantemente	Se pierde la mitad de un frame constantemente

Resultado: Como se puede observar comparando los resultados de Cuadro XV, se hace evidente que la activación del mecanismo de QoS del estándar IEEE 802.11s hace un efecto positivo en el tratamiento del flujo video privilegiado cuando éste debe competir por el canal con tráfico de fondo UDP. Usando un códec de video más eficiente, como el VP6, se mejoró el soporte de QoS con respecto a ensayos anteriores. Sin embargo, los resultados de esta prueba de videoconferencia no pueden compararse directamente con el de Video Streaming, debido a las diferencias sustanciales de resolución de imagen y

variabilidad de las imágenes. Es por ello que en esta última experiencia el soporte de Calidad de Servicio parece ser también eficiente contra mayores velocidades de tráfico. Los resultados muestran que se logran alcanzar los 18 Mbit/s. Hay que señalar que la percepción actual de la calidad de los servicios de video streaming (youtube, justing tv, UStream, etc) en las redes de datos sin QoS se debe a que se utilizan códecs muy conservadores en el uso del canal (como VP6), técnicas de buffering para amortiguar el jitter y TCP como protocolo de transporte que tiene soporte de pérdidas de paquetes.

4.4 Evaluación de la configuración de Voz sobre IP con QoS

En esta sección describimos las interacciones del flujo de VoIP con y sin soporte de QoS para una llamada real en VoIP y para varios flujos simulados de UDP. Esto último para verificar la capacidad del canal para dar soporte a múltiples llamadas reales.

4.4.1 Evaluación del Soporte de QoS para una llamada real de VoIP

Para evaluar los efectos del soporte de QoS en la red de prueba de IEEE 802.11s con el firmware OpenWRT sobre los servicios de VoIP se utilizó el servidor Elastix 2.3.0. Ésta es la versión actual estable basada en Asterisk 1.8. Es una imagen ISO de un servidor de Asterisk pre-configurado lo que significa que Elastix contiene una versión completa de Asterisk y fue creada para el establecimiento de comunicaciones unificadas, además del servidor VoIP integra otros servicios que no son de interés para nuestros experimentos. Entre las características que nos interesan, Elastix es un software de código abierto y la licencia es gratis. El servidor de Elastix hace las funciones de central telefónica realizando todas las funciones

relativas a señalización (basada en SIP u otros estándares) localización de abonados y terminales, contabilidad y facturación de las llamadas. En la Figura 23 se mostró una captura de la interfaz gráfica del servidor VMware donde se instaló una máquina virtual con Elastix. En la Figura 28 se muestra la interfaz gráfica del servidor Elastix configurado.

Como terminales se usaron teléfonos basados en software o Softphones. Jitsi también de código abierto y gratuito que soporta el protocolo de señalización SIP y se pueden instalar en distintas plataformas como Microsoft Windows. Una vez configurado el servicio de VoIP se realizaron pruebas de valoración cualitativa de las llamadas basadas en el Codec G 711 mediante el Promedio de Valoración por Opinión o MOS (*Mean Opinion Score*) que califica la calidad de una llamada del 1 al 5 siendo 5 una calidad perfecta y la calificación de 1 se obtiene cuando la conversación es imposible de entender. Así para evaluar una llamada de VoIP se realizó una llamada de voz sobre IP y se llenó el canal de datos con tráfico extra hasta que la calidad de la llamada se deterioró considerablemente como se realizó con el servicio de video streaming. Se utilizó `iperf` como generador del tráfico extra basado en UDP.

Luego se repitieron las pruebas con el soporte de QoS activado en los routers del núcleo de red. Los resultados deben ser que en el punto de referencia tomado anteriormente la calidad de la llamada debe ser mejor. En el cuadro XVI se muestran las valoraciones de la llamada sin soporte de QoS en la red y en el cuadro XVII se muestran las evaluaciones de las llamadas con el Soporte de QoS.

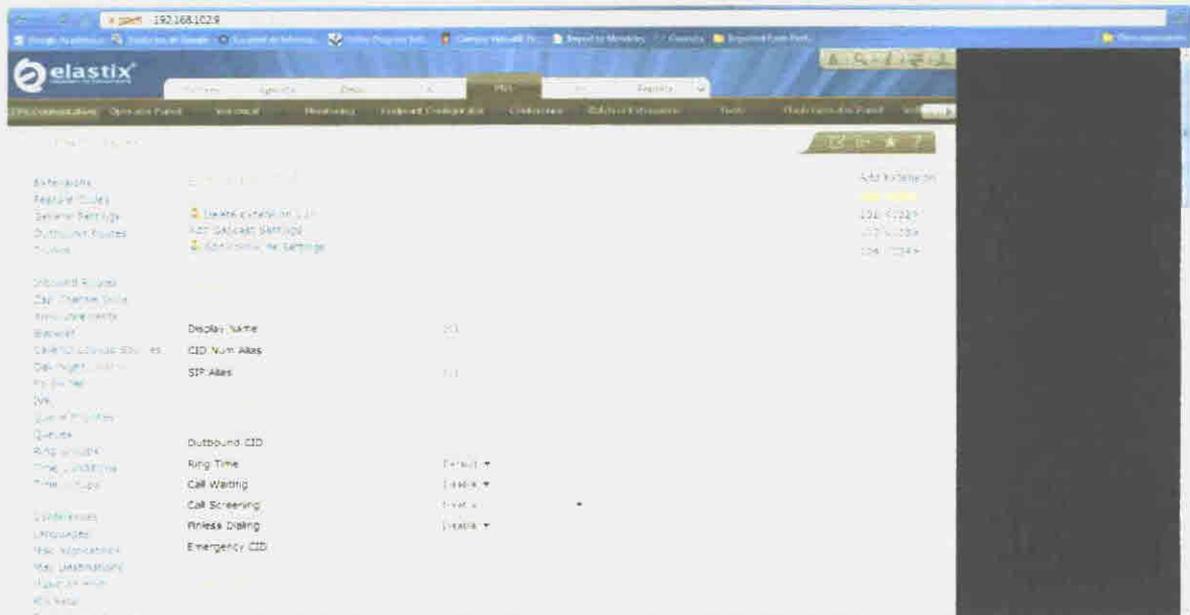


Fig. 28 Interfaz gráfica de servidor Elastix instalado.

CUADRO XVI. MEDICIÓN DE CALIDAD DE LLAMADAS SIN QOS

Prueba	BW (Mbits/s)	Calidad de la llamada	MOS	% Pérdidas
1	10	Muy Buena	5	0.0
2	11	Buena	5	0.0
3	12	Aceptable (voz robotizada)	4	5.9
4	13	Aceptable (voz robotizada)	3	11
5	14	Mala	2	16
6	15	Mala (audio en una vía)	1	22
7	16	Muy Mala	1	28
8	17	Muy Mala	1	32
9	18	Muy Mala	1	34

CUADRO XVII MEDICIÓN DE CALIDAD DE LLAMADAS CON QOS

Prueba	BW (Mbits/s)	Calidad de la llamada	MOS	% Pérdidas
1	10	Muy Buena	5	0 025
2	11	Muy Buena	5	0
3	12	Muy Buena	5	3 4
4	13	Buena	4	11
5	14	Buena	4	24
6	15	Aceptable	4	27
7	16	Aceptable	3	27
8	17	Aceptable	3	31
9	18	Mala	3	39
10	19	Mala	3	38
11	20	Mala	3	42
12	21	Mala	3	44
13	22	Mala	3	80
14	23	Mala	3	52
15	24	Mala	3	76
16	25	Muy Mala	2	68

Resultado Como se puede observar comparando ambos cuadros (XVI y XVII) se hace evidente que la activación del mecanismo de QoS del estándar IEEE 802.11s hace un efecto positivo en el tratamiento del flujo VoIP privilegiado cuando éste debe competir por el canal con tráfico de fondo UDP. El Codec G.711 ocupa un canal de 67 a 86 Kbit/s dependiendo del uso o no de la compresión de cabeceras RTP. Es por este hecho que el deterioro de una llamada sin soporte de QoS ocurre cuando el tráfico de fondo alcanza los 14 Mbit/s (alrededor de un 75% del ancho de banda del canal). Sin embargo, gracias al soporte de QoS activado, el punto donde la calidad de la llamada se deteriora debe trasladarse a un ancho de banda mayor, cerca del límite calculado de 18.5 Mbit/s (exactamente a los 18 Mbit/s o 97.2% del canal).

4.4.2. Evaluación del Soporte de QoS para múltiples conexiones de VoIP

Para conocer las posibilidades de transmitir tráfico simultáneo de VoIP en un canal de IEEE 802.11s entre dos MSTA o nodos del núcleo de red mesh, hemos configurado flujos UDP con las características similares a las que produce el códec de VoIP con mayor consumo de ancho de banda que es el G.711 en comparación con otros que se muestran en el cuadro XVIII. El códec G.711 se resalta en amarillo.

CUADRO XVIII. DESCRIPCIÓN DE LAS CARACTERÍSTICAS DE CODECS DE VOZ

Información sobre el Codec			Cálculos de Ancho de Banda				
Codec & Bit Rate (Kbps)	Tamaño de Muestra del Codec (Bytes)	Duración de Muestra del Codec (ms)	Tamaño de la Carga Útil (Voz) (ms)	Tamaño de la Carga Útil (Voz) (Bytes)	Paquetes por Segundo (PPS)	Ancho de Banda sin cRTP (Kbps)	Ancho de Banda con cRTP (Kbps)
G.711 (64 Kbps)	80 Bytes	10 ms	20 ms	160 Bytes	50	82.8 Kbps	67.6 Kbps
G.729 (8 Kbps)	10 Bytes	10 ms	20 ms	160 Bytes	50	26.8 Kbps	11.6 Kbps
G.723.1 (6.3 Kbps)	24 Bytes	30 ms	30 ms	24 Bytes	33.3	18.9 Kbps	8.8 Kbps
G.723.1 (5.3 Kbps)	20 Bytes	30 ms	30 ms	20 Bytes	33.3	17.9 Kbps	7.7 Kbps
G.726 (32 Kbps)	20 Bytes	5 ms	20 ms	80 Bytes	50	50.8 Kbps	35.6 Kbps
G.726 (24 Kbps)	15 Bytes	5 ms	20 ms	60 Bytes	50	42.8 Kbps	27.6 Kbps
G.728 (16 Kbps)	10 Bytes	5 ms	30 ms	60 Bytes	33.3	28.5 Kbps	18.4 Kbps
G722_64k (64 Kbps)	80 Bytes	10 ms	20 ms	160 Bytes	50	82.8 Kbps	67.6 Kbps
ilbc_mode_20 (15.2Kbps)	38 Bytes	20 ms	20 ms	38 Bytes	50	34.0Kbps	18.8 Kbps
ilbc_mode_30 (13.33Kbps)	50 Bytes	30 ms	30 ms	50 Bytes	33.3	25.867 Kbps	15.73Kbps

Fuente: Cisco²

² <http://www.cisco.com/c/en/us/support/docs/voice/voice-quality/7934-bwidth-consume.html>

Todos los paquetes de VoIP están compuestos por dos componentes el muestreo de la voz y las cabeceras IP/UDP/RTP Aunque las muestras de voz son comprimidas mediante tratamiento digital de la señal (DSP) y pueden variar en tamaño basado en el códec usado las cabeceras tienen un tamaño constante de 40 bytes de longitud Al comparar los 20 bytes de muestras de voz por defecto en una llamada con códec G 729 estas cabeceras realizan una considerable cantidad de sobrecarga. Con cRTP o compresión de cabeceras RTP estas cabeceras son comprimidas de dos a 4 bytes Esta compresión ofrece suficiente ahorro de ancho de banda. Por ejemplo en una llamada configurada por defecto en G 729 esta consume 24 Kbps sin cRTP y sólo 12 Kbps con la cRTP habilitada.

Utilizando la descripción de un paquete de códec G 711 con una carga útil de 160 Bytes y compresión de cabeceras se tiene que el ancho de banda consumido promedio es de 67.6 Kbps El `iperf` nos permite inyectar varios flujos UDP simultáneamente en un sentido programando el ancho de banda. Configuramos el `iperf` con 64 kpbs, según los datos generales del códec G 711 porque el valor del ancho de banda que se genera es variable oscilando alrededor de 64 kpbs En la tabla XVIII se mostró resaltado los parámetros seleccionados La cantidad de flujos en el canal se iban incrementando de la siguiente forma 1 3 5 10 15 20 30 40 y 60 flujos simultáneos Primero se realizó la prueba sin el mecanismo de QoS activado y luego activando el mecanismo de QoS en el canal verificando las ventajas en los parámetros de QoS En la Cuadro XIX se muestran los resultados de la evaluación del soporte de QoS para múltiples conexiones de VoIP sin mecanismo de QoS activado

CUADRO XIX. EVALUACIÓN DEL SOPORTE DE QOS PARA MULTIPLES CONEXIONES DE VOIP SIN MECANISMO DE QOS

Cantidad de Flujos	Ancho de Banda (Kbits/sec)	Jitter	% Paquetes Perdidos
1	64	0 29	0 00
3	64	0 39	0 00
5	63 67	0 90	0 00
10	63 83	0 90	0 00
15	61 96	1 17	0 0
20	61 79	0 83	0 00
25	59 64	0 89	0 00
30	58 87	0 81	0 00
40	53 43	1 48	0 54
60	37 06	1 99	0 59

En la Cuadro XX se muestran los resultados de la desviación estándar de las multiples conexiones de voip sin mecanismo de QoS activado

CUADRO XX EVALUACIÓN DEL SOPORTE DE QOS PARA MULTIPLES CONEXIONES DE VOIP SIN MECANISMO DE QOS DESVIACION ESTÁNDAR

Cantidad de Flujos	Ancho de Banda (Kbits/sec)	Jitter(ms)	% Paquetes Perdidos
1	0	0 06	0 00
3	0 43	0 14	0 00
5	2 15	0 60	0 00
10	4 55	0 26	0 00
15	5 39	0 85	0 0
20	9 82	0 22	0 00
25	16 50	0 53	0 00
30	17 68	0 39	0 00
40	31 70	1 39	2 75
60	44 05	1 70	2 77

En el Cuadro XXI se exponen los resultados de la evaluación del soporte de QoS para multiples conexiones de voip con mecanismo de QoS activado

CUADRO XXI EVALUACIÓN DEL SOPORTE DE QOS PARA MULTIPLES CONEXIONES DE VOIP CON MECANISMO DE QOS

Cantidad de Flujos	Ancho de Banda (Kbits/sec)	Jitter(ms)	% Paquetes Perdidos
1	64	0 27	0 04
3	63 89	0 52	0 00
5	63 94	1 19	0 00
10	63 83	1 13	0
15	60 8	1 56	0 00
20	62 97	0 91	0 00
25	51 59	2 55	0 00
30	58 87	0 81	0 00
40	42 43	2 89	0 00
60	36 42	1 98	0 02

En la Cuadro XXII se muestran los resultados de la desviación estándar de las multiples conexiones de voip con mecanismo de QoS activado

CUADRO XXII EVALUACIÓN DEL SOPORTE DE QOS PARA MULTIPLES CONEXIONES DE VOIP CON MECANISMO DE QOS DESVIACIÓN ESTÁNDAR

Cantidad de Flujos	Ancho de Banda (Kbits/sec)	Jitter	% Paquetes Perdidos
1	0	0 03	0 20
3	1 20	0 21	0 00
5	1 79	1 00	0 00
10	4 30	0 41	0
15	9 37	1 48	0 00
20	6 72	0 24	0 00
25	18 14	2 64	0 00
30	17 38	0 38	0 00
40	27 63	3 62	0 00
60	45 63	2 94	0 06

CUADRO XXIII. REQUISITO DE QoS PARA VoIP

Ancho de Banda (Kbits/sec)	Jitter (ms)	Delay (ms)	% Perdidas
85 a 60	<50	<150	1%

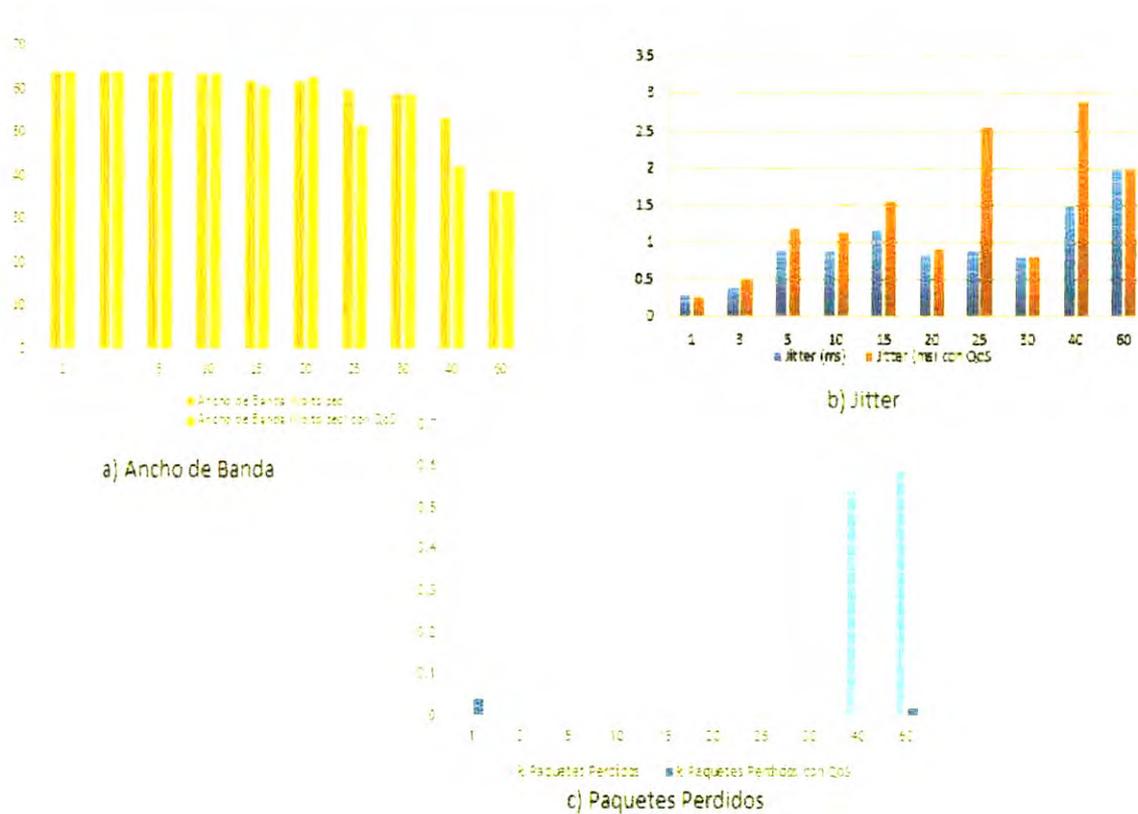


Fig 21. Comparativa de la evaluación del soporte de QoS para múltiples conexiones de VoIP con y sin mecanismo de QoS

Resultado: Como se puede observar comparando ambos cuadros (XIX y XXI.) se hace evidente que la activación del mecanismo de QoS del estándar IEEE 802.11s hace un efecto positivo en el tratamiento del flujo VoIP privilegiado con múltiples flujos simultáneos.

CAPÍTULO V

CONCLUSIONES Y TRABAJOS FUTUROS

5 1 CONCLUSIONES

Aunque las Redes Multi saltos Inalámbricos han sido objeto de estudio desde el inicio del Internet a través de los proyectos iniciales de desarrollo del DARPA Net, no ha sido hasta la última década en la que se ha impreso una mayor atención y dinamismo gracias a las mejoras en las tecnologías de acceso al medio inalámbrico (MIMO OFDM, CDMA, FHSS DSSS etc) mejoras al hardware de comunicaciones y la digitalización de la mayor parte de los sistemas de comunicaciones mediante DSP y otros factores. Se han desarrollado así proyectos de investigación en redes con topología en forma de malla con estas tecnologías, lo que en la actualidad se conoce como Redes Mesh Inalámbricas (WMN). Estas WMN han empezado a ser implementadas en zonas urbanas y en donde los recursos capitales son mayores. Se empiezan a realizar ensayos en zonas urbanas y bajo nivel económico gracias a las iniciativas de Open Hardware y Open Software. Así nuestra Universidad de Panamá, a través de la Facultad de Informática, Electrónica y Comunicación en el Marco del Programa de Maestría en Ciencias de la Ingeniería de Sistemas de Comunicaciones con Énfasis en Redes de Datos se formó un grupo de investigación en WMN de bajo coste con el fin de ayudar a diferentes estratos sociales a acceder a la Sociedad de la Información.

El proyecto revela que las Redes Mesh Inalámbricas puede proveer soporte a servicios multimedia reales con Calidad de Servicio (QoS) óptimo para los usuarios. La transmisión de servicios multimedia reales presenta problemas de rendimiento

insuficiente ancho de banda, jitter y mayor pérdida de paquetes siendo difícil ofrecer garantías de servicio. Como aporte innovador de esta investigación es la implementación de la red piloto en la que se desarrollan pruebas de servicios multimedia de interés para el desarrollo de la Sociedad de la Información en entornos de escasos recursos y bajo desarrollo humano. Dichos servicios se verifican con y sin soporte de QoS comprobándose la viabilidad de los beneficios perseguidos.

Los resultados de esta Tesis de Investigación se constituyen en aportes particulares entre los trabajos desarrollados en el Grupo de Investigación. Realizadas la red de prueba, el diseño de un sistema de monitorización de flujos, el análisis de la configuración apropiada para servicios multimedios WMN mediante conexiones unicast y multicast, los objetivos de la Tesis de Investigación apuntan al estudio de la configuración de los nodos del núcleo de red de la WMN para el soporte de QoS. Todos los objetivos específicos fueron realizados o logrados.

- 1 Se investigó el estado del arte de las Redes Mesh Inalámbricas y el Soporte de Calidad de Servicio en general y en las Redes Mesh basadas en el estándar IEEE 802.11s en particular.
- 2 Se implementó una red específica de prueba IEEE 802.11s para el soporte de QoS de los flujos que por éste transitan.
- 3 Se evaluó la capacidad de proveedor calidad de servicio a distintos tipos de tráfico multimedia como VoIP, Streaming de Video y Video Conferencia.
- 4 Se han propuesto configuraciones de soporte de QoS para los nodos del núcleo de red mesh IEEE 802.11s.

Sobre el primer objetivo se estableció una comprensión fundamental sobre los retos que se presentan ante en las redes WMN. Entre esto la cantidad de señalización adicional que se debe introducir en la red para poder evitar colisiones entre los nodos que desean transmitir al mismo tiempo y el efecto de nodo oculto (acceso al medio mediante CSMA/CA). Otro de los retos, es el otorgamiento de privilegios a los flujos mediante categorías, la calendarización (scheduling) y el control de tráfico a la salida de los nodos terminales y entre los nodos de la red nuclear. Debido a que el estándar IEEE 802.11s se basa en el encaminamiento a través de la Capa de Enlace los mecanismos de QoS en la WMN no dependen del tipo de protocolo de Capa de Red como IPv4 o IPv6.

Sobre el segundo objetivo se definió una red mesh inalámbrica de prueba basadas en las experiencias previas del grupo y se incorporaron servicios adicionales para la generación de tráfico multimedia de interés (VoIP, Video Streaming y Video Conferencia) se configuraron además los nodos del núcleo de la red mesh aprovechando los módulos de soporte de QoS que se configuran mediante el QoS Script del OpenWRT comprendiendo los aspectos de programación del mismo (principalmente con las categorías "priority" y "express"). La red mesh fue implementada siguiendo principios de diseño establecido por requisitos bien fundamentados: economía y flexibilidad, fiabilidad y versatilidad, y utilización de estándares.

Para el logro del tercer objetivo se realizaron pruebas evaluativas del funcionamiento del mecanismo de QoS entre dos nodos de la red mesh IEEE 802.11s (MSTAs) implementados con el Firmware OpenWRT instalados en routers inalámbricos de bajo costo de la marca TP Link TL WDR4300. Se aprovechó que los nodos se comunican en una frecuencia de 5GHz, banda que no se utiliza por otros dispositivos de banda no

licenciada en el entorno de pruebas eliminando así posibles interferencias que afectarían las mediciones. Además de la configuración de servicios multimedia con una gama considerable de ofertas concentrados en tres aplicaciones (video streaming, video conferencia y VoIP) se analizaron también los flujos y se revisaron sus características para modelar el tráfico mediante herramientas de sniffing (Wireshark). Conocer las características del flujo nos permitió configurar los scripts de QoS según nuestras necesidades.

El análisis de los resultados de los ensayos desarrollados en la red de prueba configurada nos ha permitido conocer los beneficios que aporta al canal de comunicaciones en un solo salto el mecanismo de QoS. Se comprendió que el firmware (software) que gobierna el funcionamiento del hardware no siempre aprovecha las facilidades que éste último brinda. Por ejemplo, los routers inalámbricos usados TP Link TL-WDR4300 pueden comunicarse a 300 Mbit/s según el estándar IEEE 802.11n, pero usando el OpenWRT aún se sincronizan en 54 Mbit/s según el estándar IEEE 802.11g. Se espera en el futuro que actualice el firmware para cumplir estas especificaciones.

Se determinó que de esta velocidad teórica de 54 Mbit/s en realidad la capacidad del enlace en condiciones de estabilidad, cercanía y carencia de interferencias sólo es de una tercera parte, o sea, 18.5 Mbit/s.

Con respecto al soporte de QoS de los servicios multimedia, se determinó que la transmisión de video streaming de alta calidad (con un flujo promedio de 1.0 Mbit/s) codificado mediante los codificadores MPEG 4 para el video y MP3/AAC para el audio transportado por el juego de protocolos (RTP/UDP/IP) puede lograrse siempre que el tráfico de fondo sin QoS no supere los 600 Kbit/s. Esto nos motivó a probar con otros

esquemas de codificación más recientes como el flv o Flash Movie que utiliza el códec de video VP más eficientes para enlaces ruidosos o de poca capacidad. Es así que al probar este codificador implementado en el servidor Red5 la capacidad de mantenimiento de

QoS para una video-conferencia regular siendo su tolerancia un tráfico de carga que alcanza los 15 Mbit/s o más

Aunque se ha utilizado Iperf como inyector de tráfico para estimar cuantitativamente los valores la evaluación cuantitativa mediante el MOS es de mucho valor ya que el usuario final se implica en esta valoración y se conocen mejor sus percepciones del servicio

En cuanto a la audio-conferencia o llamadas mediante VoIP se encontró que el codificador G 711 permite la transmisión de una llamada con calidad aceptable cuando el canal está ocupado con tráfico de fondo de hasta 17 Mbit/s (cerca del 92% de ocupación del canal) al programar apropiadamente el QoS Script

Los resultados de esta Tesis de Investigación se han podido divulgar en las siguientes publicaciones

- 1 Aidelen Danais Chung Iván Armuelles Voinov Optimización de una Red Mesh Inalámbrica para Tráfico Multimedia Mediante Calidad de Servicio Centro de Investigación en Tecnologías de la Información y Comunicación Facultad de Informática, Electrónica y Comunicación. XXVI Congreso Científico Nacional Universidad de Panamá. 21 al 25 de octubre de 2013
- 2 Iván Armuelles Aidelen Chung Joaquín Chung Grace González. "A Performance Analysis of Wireless Mesh Networks Implementations Based on Open Source Software Proceedings of the International Conference on Open Source Systems Springer International San José Costa Rica.

5 2 TRABAJOS FUTUROS

En cuanto al estudio de la implementación de red basado en el estándar IEEE 802 11s se ha realizado muchos avances y se han logrado varias publicaciones sobre la comparación de la implementación del estándar en OpenWRT con respecto a otros tipos de mecanismos de enrutamiento la configuración apropiadas de servicios multimedia, el monitoreo de los flujos y el soporte de la Calidad de Servicio Sobre los trabajos de investigación que aun se pueden realizar están

- 1 Investigar la integración de mecanismos de control de acceso al tráfico a la red mesh de tal forma que se pueda proveer calidad de servicio extremo a extremo para la implementación de una red real
- 2 Investigar y aplicar el soporte de Movilidad de los terminales de usuario en la red mesh sin la pérdida de la conexión de los flujos basándonos en el estándar IEEE 802 11r
- 3 Investigar la interoperabilidad con otros servicios de comunicaciones móviles tal y como lo recomienda el IEEE 802 11u

Se espera que estas actividades sean desarrolladas en el futuro próximo por otros participantes del grupo de investigación

BIBLIOGRAFÍA

- Adobe (n.d.) Flash Media. Retrieved from <http://www.adobe.com/es/products/flash-media/playback.html>
- Akyıldız, I F Wang, X. & Wang, W (2005) Wireless mesh networks a survey *Computer Networks* 47(4) 445–487 doi 10.1016/j.comnet.2004.12.001
- Alotaibi E & Mukherjee, B (2012) A survey on routing algorithms for wireless Ad Hoc and mesh networks *Computer Networks* 56(2) 940–965 doi 10.1016/j.comnet.2011.10.011
- Andreu, F Pellejero I & Lesta, A (2006) *Fundamentos y aplicaciones de seguridad en redes WLAN* (p 160) España MARCOMBO S A doi ISBN 84-267 1405 6
- Carrano R. C Magalhães, L C S Saade D C M & Albuquerque C V N (2011) IEEE 802.11s Multihop MAC A Tutorial *IEEE COMMUNICATIONS SURVEYS & TUTORIALS* 13(1) 52–67
- Chakeres, I & Perkins, C (2010) Dynamic MANET On-demand (DYMO) Routing *DRAFT IETF MANET WG* 1–41
- Cheng H T & Zhuang W (2009) QoS Driven Node Cooperative Resource Allocation for Wireless Mesh Networks with Service Differentiation *GLOBECOM 2009 2009 IEEE Global Telecommunications Conference* 1–6 doi 10.1109/GLOCOM.2009.5425462
- Chung, J González, G Armuelles, I Robles, T Alcarria, R. & Morales, A. (2012) Experiences and Challenges in Deploying OpenFlow over a Real Wireless Mesh Network. *IEEE* 0–4
- Chung, J González, G I A Robles, T Alcarria, R. & Morales, A (n.d.) Characterizing the Multimedia Service Capacity of Wireless Mesh Networks for Rural Communities *The 8th IEEE International Conference on Wireless and Mobile Computing Networking and Communications (WiMob 2012 Barcelona, Spain)*
- Chung, J González, G I A Robles, T Alcarria, R. & Morales, A. (2013) Experiences and Challenges in Deploying OpenFlow over a Real Wireless Mesh Network *Latin America Transactions IEEE (Revista IEEE America Latina)* 2(3) doi ISSN 1548-0992
- Clausen, T & Jacquet, P (2003) Optimized Link State Routing Protocol (OLSR) *IETF RFC 3626* 1–69

- Crawley E, R.Nair B Rajagopalan, & H Sandick. (1998) A Framework for QoS based Routing in the Internet *IETF RFC 2386* 1–35
- Cuervo F, Greene, N, Rayhan, A, Huitema, C, Rosen, B & Segers, J (2000) MEGACO Protocol *IETF RFC 3015* 1–180
- De Couto D S J, Aguayo D, Bicket, J & Morris, R. (2003) A high throughput path metric for multi hop wireless routing *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking MobiCom 03* 134
doi 10.1145/938998.939000
- Gonzalez, G (2013) *Análisis de Técnicas para la Transmisión de Tráfico Multimedia a Través de Redes Mesh Inalámbricas* Maestría en Ciencias de la Ingeniería de Sistemas de Comunicación con Énfasis en Redes de Datos. Facultad de Informática, Electrónica y Comunicación. Universidad de Panamá.
- Haas, Z, J, Pearlman, M R. & Samar P (2002) The Zone Routing Protocol (ZRP) for Ad Hoc Networks *MANET INTERNET DRAFT* 1–16
- Handley M & Jacobson, V (1998) SDP Session Description Protocol *IETF RFC 2327* 1–39
- Hertz, G R, Denteneer D, Max, S, Taori R, Cardona, J, Berlemann, L & Walke B (2010) IEEE 802.11S THE WLAN MESH STANDARD *IEEE Wireless Communications* (February) 104–111
- ITU (2005) Manual sobre redes basadas en el Protocolo Internet (IP) y asuntos conexos
- ITU (2007a) ITU T Rec Y 1543 Measurements in IP networks for inter-domain performance assessment ITU [En línea] 2007
- ITU (2007b) The Future of Voice The Status of Voice Over Internet Protocol (VoIP) Worldwide 2006 doi FoV/04
- Johnson, D, Hu, Y & Maltz, D (2007) The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4 *IETF RFC 4728* 1–108
- Khan, M, A, Reza, S M & Moradi H (2007) A Brief Overview of Wireless Mesh Networks with Focus on Routing 1–10
- Kone V, Das, S, Zhao B Y & Zheng, H (2008) QUORUM—Quality of Service in Wireless Mesh Networks *Mobile Networks and Applications* 12(5-6) 358–369
doi 10.1007/s11036-008-0050-8

- Ksentini A. & Abassi O (2008) A Comparison of VoIP Performance Over Three Routing Protocols for IEEE 802.11s based Wireless Mesh Networks (WLAN Mesh) *MobiWac 08* 147–150 doi 10.1145/1454659.1454687 ISBN 978-1-60558-055-5
- Kuklinski S, Radziszewski, P & Wytrębowicz, J (2011) WARF Component Based Platform for Wireless Mesh Networks *The Smart Computing Review* 1(2) 125–138 doi 10.6029/smarterc.2011.02.004
- Lee C S (2006) Global Standards for Converged Communication Services In *ITU Workshop Future of Voice* Geneva.
- Liu, C, Shu, Y, Zhang, L, Zhao Z & Li, X. Y (2009) A multi layer experimental study of multimedia and QoS communication in wireless mesh networks *Pervasive and Mobile Computing* 5(1) 19–36 doi 10.1016/j.pmcj.2008.04.013
- Marina, M K. & Das, S R. (2006) Ad hoc on-demand multipath distance vector routing *Wireless Communications and Mobile Computing* 6(7) 969–988 doi 10.1002/wcm.432
- Ogier R, Templin, F & Lewis, M (2004) Topology Dissemination Based on Reverse-Path Forwarding (TBRPF) *IETF RFC 3684* 1–47
- OpenWRT (n.d.) Retrieved from <https://openwrt.org/>
- Perkins, C, Belding Royer E & Das, S (2003) Ad hoc On Demand Distance Vector (AODV) Routing *IETF RFC 3561* 1–38
- Phillippa Biggs I (2009) GSR 2009 *ITU*
- Postel J (1980) User Datagram Protocol *IETF RFC 768* 9–11
- Postel J (1981) Transmission Control Protocol (TCP) *IETF RFC 793*
- Red5 O (n.d.) Red 5 Retrieved from <http://www.red5.org/>
- Schulzrinne H, Casner S, Frederick, R. & Jacobson, V (2003) RTP A Transport Protocol for Real Time Applications *IETF RFC 3550* 1–96
- Schulzrinne H, Rao A & Lanphier R (1998) Real Time Streaming Protocol (RTSP) *IETF RFC 2326*
- Society I C (2012) *IEEE Standard for Information technology Telecommunications and information exchange between systems Local and metropolitan area networks Specific requirements Part 11 Wireless LAN Medium Access Control (MAC) and*

Physical Layer (PHY) Specifications (The Institute of Electrical and Electronics Engineers Ed) (Vol 2012 p 2793) United States of America.

VideoLAN (n d) VLC Retrieved from [http //www videolan org/vlc/](http://www.videolan.org/vlc/)

Voinov I A Cedeño A. C Chung, J & González, G (2014) A Performance Analysis of Wireless Mesh Networks Implementations Based on Open Source Software *10th IFIP WG 2.13 International Conference on Open Source Systems OSS 2014 Book. Mobile Open Source Technologies San Jose Costa Rica Mayo 6-9 2014*

ANEXO

Prueba de caracterización de ancho de banda con TCP en el servidor sin QoS

Datos regidos en el servidor de iperf

Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation Reservados todos los derechos

```
C:\Users\CITIC_2>iperf -s
```

```
-----  
Server listening on TCP port 5001  
TCP window size 64 0 KByte (default)  
-----  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51238  
[ ID] Interval      Transfer      Bandwidth  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51239  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51240  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51241  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51242  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51243  
[ 4] 0 0-60 1 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51244  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51245  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51246  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79  
port 51247  
[ 4] 0 0-60 2 sec   133 MBytes   18.5 Mbits/sec
```

Caracterización del ancho de banda utilizando UDP sin QoS

Datos regidos en el servidor de iperf

Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation Reservados todos los derechos

C:\Users\CITIC_2>iperf -u -s

```
-----  
Server listening on UDP port 5001  
Receiving 1470 byte datagrams  
UDP buffer size 64 0 KByte (default)  
-----  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 43876  
[ ID] Interval      Transfer      Bandwidth      Jitter    Lost/Total  
[ 4] 0 0-30 2 sec  78 1 MBytes  21 7 Mbits/sec  10 957 ms 200678/256409 (78%)  
[ 4] 0 0 30 2 sec  1 datagrams received out-of-order  
[ 3] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 49584  
[ 3] 0 0-30 2 sec  80 4 MBytes  22 3 Mbits/sec  11 069 ms 199030/256406 (78%)  
[ 3] 0 0-30 2 sec  1 datagrams received out-of order  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 34752  
[ 4] 0 0 30 2 sec  77 9 MBytes  21 6 Mbits/sec  11 266 ms 200828/256410 (78%)  
[ 4] 0 0 30 2 sec  28 datagrams received out-of-order  
[ 3] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 54182  
[ 3] 0 0-30 2 sec  78 0 MBytes  21 6 Mbits/sec  11 256 ms 200779/256406 (78%)  
[ 3] 0 0-30 2 sec  13 datagrams received out-of-order  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 49953  
[ 4] 0 0-30 1 sec  80 4 MBytes  22 4 Mbits/sec  0 723 ms 199066/256401 (78%)  
[ 4] 0 0 30 1 sec  17 datagrams received out-of-order  
[ 3] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 60838  
[ 3] 0 0-30 1 sec  78 1 MBytes  21 8 Mbits/sec  0 579 ms 200714/256402 (78%)  
[ 3] 0 0-30 1 sec  11 datagrams received out-of order  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 46661  
[ 4] 0 0-30 2 sec  80 5 MBytes  22 3 Mbits/sec  11 091 ms 198953/256406 (78%)  
[ 4] 0 0 30 2 sec  1 datagrams received out-of-order  
[ 3] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 39821  
[ 3] 0 0-30 1 sec  80 4 MBytes  22 4 Mbits/sec  0 531 ms 199048/256410 (78%)  
[ 3] 0 0-30 1 sec  1 datagrams received out-of-order  
[ 4] local 192 168 18 78 port 5001 connected with 192 168 18 79 port 38140  
[ 4] 0 0-30 1 sec  78 1 MBytes  21 8 Mbits/sec  0 571 ms 200698/256402 (78%)  
[ 4] 0 0 30 1 sec  1 datagrams received out-of-order
```

Estado de la interfaz inalámbrica mesh

```
root@WMMR-079-04 ~# iw dev wlan1 station dump
```

```
Station 64 70 02 8f 35 56 (on wlan1)
  inactive time      400 ms
  rx bytes           2669219116
  rx packets         1981535
  tx bytes           1677257904
  tx packets         1078945
  tx retries         30930
  tx failed          0
  signal             -38 [-43 -44 -41] dBm
  signal avg         -37 [-42 -43 -41] dBm
  Toffset            -24311378427 us
  tx bitrate         54 0 MBit/s
  rx bitrate         6 0 MBit/s
  mesh llid          14938
  mesh plid          52374
  mesh plink         ESTAB
  authorized         yes
  authenticated      yes
  preamble           long
  WMM/WME            yes
  MFP                no
  TDLS peer         no
```

```
Station 64 70 02 8f 33 f8 (on wlan1)
  inactive time      300 ms
  rx bytes           2082421805
  rx packets         1555148
  tx bytes           89636
  tx packets         149
  tx retries         23
  tx failed          0
  signal             -49 [-53 -55 -54] dBm
  signal avg         -48 [-53 -53 -53] dBm
  Toffset            5546386765040 us
  tx bitrate         24 0 MBit/s
  rx bitrate         36 0 MBit/s
  mesh llid          50457
  mesh plid          62916
  mesh plink         ESTAB
  authorized         yes
  authenticated      yes
  preamble           long
  WMM/WME            yes
  MFP                no
  TDLS peer         no
```

```

root@WMR-079-04 ~# iw dev wlan1 mpath dump
DEST ADDR          NEXT HOP          IFACE   SN    METRIC  QLEN
  EXPTIME          DTIM DRET  FLAGS
64 70 02 8f 35 56 64 70 02 8f 35 56 wlan1  196  152  0    0
  100  0    0x14
64 70 02 8f 33 f8 64 70 02 8f 33 f8 wlan1  1796 342  0    0    0
  0    0x14

```

Archivo de configuración de la red mesh

```

config interface loopback
  option ifname lo
  option proto static
  option ipaddr 127 0 0 1
  option netmask 255 0 0 0

config interface lan
  option ifname eth0 1
  option type bridge
  option proto static
  option netmask 255 255 255 0
  option ipaddr 192 168 18 40

config interface wan
  option ifname eth0 2
  option proto dhcp

config switch
  option name eth0
  option reset 1
  option enable_vlan 1

```

Archivo de configuración wireless

```

config wifi-device radio0
  option type mac80211
  option channel 11
  option macaddr 64 70 02 8f 34 bb
  option hwmode 11ng
  option htmode HT20
  list ht_capab LDPC
  list ht_capab SHORT-GI-20
  list ht_capab SHORT-GI-40
  list ht_capab TX-STBC
  list ht_capab RX-STBC1
  list ht_capab DSSS_CCK-40
  option disabled 1

```

```

config wifi-iface
    option device radio0
    option mode ap
    option ssid OpenWrt
    option encryption none
    option network lan

config wifi-device radio1
    option type mac80211
    option channel 36
    option macaddr 64 70 02 8f 34 bc
    list ht_capab LDPC
    list ht_capab SHORT-GI-20
    list ht_capab SHORT-GI-40
    list ht_capab TX-STBC
    list ht_capab RX-STBC1
    list ht_capab DSSS_CCK-40

config wifi-iface
    option device radio1
    option network lan
    option mode mesh
    option mesh_id mesh079
    option encryption none
    option disabled 0

```

Ejemplo de Propiedades de configuración del router

```

root@WMR-079-04 ~# ifconfig
br-lan    Link encap Ethernet  HWaddr 64 70 02 8F 34 BA
          inet addr 192 168 18 40  Bcast 192 168 18 255
Mask 255 255 255 0
          UP BROADCAST RUNNING MULTICAST  MTU 1500  Metric 1
          RX packets 1840820 errors 0 dropped 3053 overruns 0
frame 0
          TX packets 23060 errors 0 dropped 0 overruns 0 carrier 0
          collisions 0 txqueuelen 0
          RX bytes 2448800369 (2 2 GiB)  TX bytes 9860993 (9 4 M1B)

eth0     Link encap Ethernet  HWaddr 64 70 02 8F 34 BA
          UP BROADCAST RUNNING MULTICAST  MTU 1500  Metric 1
          RX packets 2645003 errors 0 dropped 4 overruns 0 frame 0
          TX packets 1864086 errors 0 dropped 0 overruns 0 carrier 0
          collisions 0 txqueuelen 1000
          RX bytes 3979129769 (3 7 GiB)  TX bytes 2500877526 (2 3
GiB)
          Interrupt 4

```

```
eth0 1   Link encap Ethernet HWaddr 64 70 02 8F 34 BA
UP BROADCAST RUNNING MULTICAST MTU 1500 Metric 1
RX packets 2644996 errors 0 dropped 0 overruns 0 frame 0
TX packets 1854518 errors 0 dropped 0 overruns 0 carrier 0
collisions 0 txqueuelen 0
RX bytes 3931517735 (3 6 GiB) TX bytes 2489574846 (2 3
GiB)

eth0 2   Link encap Ethernet HWaddr 64 70 02 8F 34 BA
UP BROADCAST RUNNING MULTICAST MTU 1500 Metric 1
RX packets 0 errors 0 dropped 0 overruns 0 frame 0
TX packets 9568 errors 0 dropped 0 overruns 0 carrier 0
collisions 0 txqueuelen 5
RX bytes 0 (0 0 B) TX bytes 3846336 (3 6 MiB)

1fb0     Link encap Ethernet HWaddr 66 05 53 44 FF AA
UP BROADCAST RUNNING NOARP MTU 1500 Metric 1
RX packets 0 errors 0 dropped 0 overruns 0 frame 0
TX packets 0 errors 0 dropped 0 overruns 0 carrier 0
collisions 0 txqueuelen 5
RX bytes 0 (0 0 B) TX bytes 0 (0 0 B)

lo       Link encap Local Loopback
inet addr 127 0 0 1 Mask 255 0 0 0
UP LOOPBACK RUNNING MTU 16436 Metric 1
RX packets 229120 errors 0 dropped 0 overruns 0 frame 0
TX packets 229120 errors 0 dropped 0 overruns 0 carrier 0
collisions 0 txqueuelen 0
RX bytes 15580480 (14 8 MiB) TX bytes 15580480 (14 8 MiB)

wlan1    Link encap Ethernet HWaddr 64 70 02 8F 34 BC
UP BROADCAST RUNNING MULTICAST MTU 1500 Metric 1
RX packets 1835123 errors 0 dropped 0 overruns 0 frame 0
TX packets 1081827 errors 0 dropped 0 overruns 0 carrier 0
collisions 0 txqueuelen 32
RX bytes 2480879165 (2 3 GiB) TX bytes 1680780806 (1 5
GiB)
```

Evaluación del flujo Streaming sin QoS

1. Datos regidos en el servidor de iperf.

Con una utilización del 50% = 9 Mbps



Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

```
C:\Users\CITIC_2>iperf -s -u -i1 -p 1234
```

```
-----  
Server listening on UDP port 1234  
Receiving 1470 byte datagrams  
UDP buffer size: 64.0 KByte (default)  
-----
```

```
[ 3] local 192.168.18.78 port 1234 connected with 192.168.18.79 port 51388  
[ ID] Interval      Transfer    Bandwidth    Jitter    Lost/Total Datagram  
  
[ 3] 0.0- 1.0 sec   195 KBytes  1.60 Mbits/sec  6.325 ms   0/ 136 (0%)  
[ 3] 1.0- 2.0 sec   267 KBytes  2.19 Mbits/sec  6.487 ms  584/ 770 (76%)  
[ 3] 2.0- 3.0 sec   227 KBytes  1.86 Mbits/sec  5.089 ms  502/ 660 (76%)  
[ 3] 3.0- 4.0 sec   207 KBytes  1.69 Mbits/sec  7.099 ms  566/ 710 (80%)  
[ 3] 4.0- 5.0 sec   189 KBytes  1.55 Mbits/sec  7.287 ms  536/ 668 (80%)  
[ 3] 5.0- 6.0 sec   211 KBytes  1.73 Mbits/sec  11.549 ms 734/ 881 (83%)  
[ 3] 6.0- 7.0 sec   212 KBytes  1.74 Mbits/sec  12.342 ms 601/ 749 (80%)  
[ 3] 7.0- 8.0 sec   273 KBytes  2.23 Mbits/sec  4.449 ms  750/ 940 (80%)  
[ 3] 8.0- 9.0 sec   217 KBytes  1.78 Mbits/sec  7.262 ms  467/ 618 (76%)  
[ 3] 9.0-10.0 sec   0.00 Bytes  0.00 bits/sec   7.262 ms   0/ 0 (nan%)  
[ 3] 10.0-11.0 sec  102 KBytes   835 Kbits/sec  4.982 ms 1685/ 1756 (96%)  
[ 3] 11.0-12.0 sec  903 KBytes  7.40 Mbits/sec  2.377 ms  417/ 1046 (40%)  
[ 3] 12.0-13.0 sec  564 KBytes  4.62 Mbits/sec  3.196 ms  405/ 798 (51%)  
[ 3] 13.0-14.0 sec  748 KBytes  6.13 Mbits/sec  1.055 ms  230/ 751 (31%)  
[ 3] 14.0-15.0 sec  1.08 MBytes  9.04 Mbits/sec  1.228 ms  121/ 890 (14%)  
[ 3] 15.0-16.0 sec  1.08 MBytes  9.02 Mbits/sec  1.345 ms   0/ 767 (0%)  
[ 3] 16.0-17.0 sec  1.10 MBytes  9.23 Mbits/sec  1.193 ms   0/ 785 (0%)  
[ 3] 17.0-18.0 sec  1.06 MBytes  8.89 Mbits/sec  1.551 ms   0/ 756 (0%)  
[ 3] 18.0-19.0 sec  891 KBytes  7.30 Mbits/sec  2.077 ms  37/ 658 (5.6%)
```

```

[ 3] 19.0-20.0 sec 814 KBytes 6.67 Mbits/sec 1.994 ms 152/ 719 (21%)
[ 3] 20.0-21.0 sec 973 KBytes 7.97 Mbits/sec 2.292 ms 192/ 870 (22%)
[ 3] 21.0-22.0 sec 932 KBytes 7.63 Mbits/sec 2.217 ms 57/ 706 (8.1%)
[ 3] 22.0-23.0 sec 867 KBytes 7.10 Mbits/sec 1.270 ms 212/ 816 (26%)
[ 3] 23.0-24.0 sec 1.05 MBytes 8.80 Mbits/sec 1.103 ms 77/ 825 (9.3%)
[ 3] 24.0-25.0 sec 899 KBytes 7.36 Mbits/sec 1.642 ms 13/ 639 (2%)
[ 3] 25.0-26.0 sec 739 KBytes 6.06 Mbits/sec 1.274 ms 324/ 839 (39%)
[ 3] 26.0-27.0 sec 937 KBytes 7.68 Mbits/sec 1.243 ms 136/ 789 (17%)
[ 3] 27.0-28.0 sec 1.09 MBytes 9.11 Mbits/sec 1.372 ms 9/ 784 (1.1%)
[ 3] 28.0-29.0 sec 1.07 MBytes 8.96 Mbits/sec 2.116 ms 4/ 766 (0.52%)
[ 3] 29.0-30.0 sec 396 KBytes 3.25 Mbits/sec 3.671 ms 402/ 678 (59%)
[ 3] 0.0-30.4 sec 19.2 MBytes 5.30 Mbits/sec 12.152 ms 9276/22970 (40%)
[ 3] 0.0-30.4 sec 1 datagrams received out-of-order
read failed: Connection reset by peer

```

2. Datos regidos en el servidor de iperf.

Con una utilización del 75% = 13.5 Mbps



Microsoft Windows [Versión 6.1.7601]
 Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

```
C:\Users\CITIC_2>iperf -s -u -i 1 -p 1234
```

```
-----
Server listening on UDP port 1234
Receiving 1470 byte datagrams
UDP buffer size: 64.0 KByte (default)
-----
```

```

[ 3] local 192.168.18.78 port 1234 connected with 192.168.18.79 port 37297
[ ID] Interval      Transfer    Bandwidth    Jitter    Lost/Total Datagrams

[ 3] 0.0- 1.0 sec  54.6 KBytes  447 Kbits/sec  2.475 ms 1138/ 1176 (97%)
[ 3] 1.0- 2.0 sec   181 KBytes  1.48 Mbits/sec  2.771 ms 1012/ 1138 (89%)
[ 3] 2.0- 3.0 sec   179 KBytes  1.47 Mbits/sec  1.356 ms 1054/ 1179 (89%)

```

[3]	3.0- 4.0 sec	145 KBytes	1.19 Mbits/sec	1.528 ms	1130/	1231	(92%)
[3]	4.0- 5.0 sec	87.6 KBytes	717 Kbits/sec	3.140 ms	932/	993	(94%)
[3]	5.0- 6.0 sec	51.7 KBytes	423 Kbits/sec	5.755 ms	1032/	1068	(97%)
[3]	6.0- 7.0 sec	57.4 KBytes	470 Kbits/sec	4.295 ms	1150/	1190	(97%)
[3]	7.0- 8.0 sec	80.4 KBytes	659 Kbits/sec	2.710 ms	1153/	1209	(95%)
[3]	8.0- 9.0 sec	243 KBytes	1.99 Mbits/sec	1.505 ms	1021/	1190	(86%)
[3]	9.0-10.0 sec	109 KBytes	894 Kbits/sec	2.244 ms	1012/	1088	(93%)
[3]	10.0-11.0 sec	134 KBytes	1.09 Mbits/sec	2.262 ms	1091/	1184	(92%)
[3]	11.0-12.0 sec	103 KBytes	847 Kbits/sec	2.877 ms	1169/	1241	(94%)
[3]	12.0-13.0 sec	158 KBytes	1.29 Mbits/sec	3.060 ms	1053/	1163	(91%)
[3]	13.0-14.0 sec	220 KBytes	1.80 Mbits/sec	1.485 ms	1026/	1179	(87%)
[3]	14.0-15.0 sec	192 KBytes	1.58 Mbits/sec	2.079 ms	850/	984	(86%)
[3]	15.0-16.0 sec	112 KBytes	917 Kbits/sec	2.270 ms	1110/	1188	(93%)
[3]	16.0-17.0 sec	87.6 KBytes	717 Kbits/sec	3.495 ms	1092/	1153	(95%)
[3]	17.0-18.0 sec	198 KBytes	1.62 Mbits/sec	3.015 ms	1016/	1154	(88%)
[3]	18.0-19.0 sec	131 KBytes	1.07 Mbits/sec	1.856 ms	1044/	1135	(92%)
[3]	19.0-20.0 sec	144 KBytes	1.18 Mbits/sec	1.792 ms	1177/	1277	(92%)
[3]	20.0-21.0 sec	175 KBytes	1.43 Mbits/sec	2.443 ms	1016/	1138	(89%)
[3]	21.0-22.0 sec	152 KBytes	1.25 Mbits/sec	2.896 ms	948/	1054	(90%)
[3]	22.0-23.0 sec	80.4 KBytes	659 Kbits/sec	2.804 ms	1064/	1120	(95%)
[3]	23.0-24.0 sec	178 KBytes	1.46 Mbits/sec	1.821 ms	1016/	1140	(89%)
[3]	24.0-25.0 sec	77.5 KBytes	635 Kbits/sec	2.842 ms	1002/	1056	(95%)
[3]	25.0-26.0 sec	50.2 KBytes	412 Kbits/sec	7.675 ms	1046/	1081	(97%)
[3]	26.0-27.0 sec	116 KBytes	953 Kbits/sec	1.869 ms	1195/	1276	(94%)
[3]	27.0-28.0 sec	234 KBytes	1.92 Mbits/sec	2.362 ms	986/	1149	(86%)
[3]	28.0-29.0 sec	113 KBytes	929 Kbits/sec	2.234 ms	1077/	1156	(93%)

3. Datos regidos en el servidor de iperf.

Con una utilización del 90% = 16.65 Mbps





Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

```
C:\Users\CITIC_2>iperf -s -u -i 1 -p 1234
```

```
-----  
Server listening on UDP port 1234  
Receiving 1470 byte datagrams  
UDP buffer size: 64.0 KByte (default)  
-----
```

```
[ 3] local 192.168.18.78 port 1234 connected with 192.168.18.79 port 41602  
[ ID] Interval      Transfer      Bandwidth      Jitter    Lost/Total Datagrams  
  
[ 3] 0.0- 1.0 sec   197 KBytes    1.61 Mbits/sec  1.950 ms  1458/ 1595 (91%)  
[ 3] 1.0- 2.0 sec   280 KBytes    2.29 Mbits/sec  1.705 ms  1302/ 1497 (87%)  
[ 3] 2.0- 3.0 sec   277 KBytes    2.27 Mbits/sec  2.053 ms  1287/ 1480 (87%)  
[ 3] 3.0- 4.0 sec   218 KBytes    1.79 Mbits/sec  1.434 ms  1305/ 1457 (90%)  
[ 3] 4.0- 5.0 sec   205 KBytes    1.68 Mbits/sec  1.642 ms  1391/ 1534 (91%)  
[ 3] 5.0- 6.0 sec   266 KBytes    2.18 Mbits/sec  1.893 ms  1270/ 1455 (87%)  
[ 3] 6.0- 7.0 sec   287 KBytes    2.35 Mbits/sec  1.655 ms  1343/ 1543 (87%)  
[ 3] 7.0- 8.0 sec   317 KBytes    2.60 Mbits/sec  2.295 ms  1192/ 1413 (84%)  
[ 3] 8.0- 9.0 sec   198 KBytes    1.62 Mbits/sec  1.833 ms  1386/ 1524 (91%)  
[ 3] 9.0-10.0 sec   227 KBytes    1.86 Mbits/sec  14.938 ms 1297/ 1455 (89%)  
[ 3] 10.0-11.0 sec  74.6 KBytes    612 Kbits/sec  11.027 ms 1343/ 1395 (96%)  
[ 3] 11.0-12.0 sec  93.3 KBytes    764 Kbits/sec  5.989 ms  1560/ 1625 (96%)  
[ 3] 12.0-13.0 sec  100 KBytes     823 Kbits/sec  5.534 ms  1348/ 1418 (95%)  
[ 3] 13.0-14.0 sec  113 KBytes     929 Kbits/sec  3.486 ms  1496/ 1575 (95%)  
[ 3] 14.0-15.0 sec  89.0 KBytes    729 Kbits/sec  6.622 ms  1439/ 1501 (96%)  
[ 3] 15.0-16.0 sec  162 KBytes    1.33 Mbits/sec  2.345 ms  1374/ 1487 (92%)  
[ 3] 16.0-17.0 sec  154 KBytes    1.26 Mbits/sec  2.067 ms  1403/ 1510 (93%)  
[ 3] 17.0-18.0 sec  281 KBytes    2.30 Mbits/sec  1.306 ms  1308/ 1504 (87%)  
[ 3] 18.0-19.0 sec  294 KBytes    2.41 Mbits/sec  1.542 ms  1283/ 1488 (86%)  
[ 3] 19.0-20.0 sec  167 KBytes    1.36 Mbits/sec  1.408 ms  1375/ 1491 (92%)  
[ 3] 20.0-21.0 sec  306 KBytes    2.50 Mbits/sec  1.425 ms  1232/ 1445 (85%)  
[ 3] 21.0-22.0 sec  274 KBytes    2.25 Mbits/sec  1.859 ms  1385/ 1576 (88%)  
[ 3] 22.0-23.0 sec  281 KBytes    2.30 Mbits/sec  1.549 ms  1306/ 1502 (87%)  
[ 3] 23.0-24.0 sec  339 KBytes    2.78 Mbits/sec  1.290 ms  1268/ 1504 (84%)  
[ 3] 24.0-25.0 sec  228 KBytes    1.87 Mbits/sec  1.781 ms  1293/ 1452 (89%)  
[ 3] 25.0-26.0 sec  217 KBytes    1.78 Mbits/sec  1.523 ms  1361/ 1512 (90%)  
[ 3] 26.0-27.0 sec  293 KBytes    2.40 Mbits/sec  1.222 ms  1301/ 1505 (86%)  
[ 3] 27.0-28.0 sec  247 KBytes    2.02 Mbits/sec  1.356 ms  1334/ 1506 (89%)  
[ 3] 28.0-29.0 sec  301 KBytes    2.47 Mbits/sec  0.845 ms  1313/ 1523 (86%)
```

Archivo de configuración utilizado para el ancho de banda en QoS

```
# QoS configuration for OpenWrt

# INTERFACES
config interface wan
    option classgroup Default
    option enabled 1
    option upload 128
    option download 1024

# RULES
config classify
    option target Priority
    option ports 22 53
config classify
    option target Normal
    option proto tcp
    option ports 20 21 25 80 110 443 993 995
config classify
    option target Express
    option ports 1234
config default
    option target Express
    option proto udp
    option pktsize -500
config reclassify
    option target Priority
    option proto icmp
config default
    option target Bulk
    option portrange 1024-65535
config reclassify
    option target Priority
    option proto tcp
    option pktsize -128
    option mark Bulk
    option tcpflags SYN
config reclassify
    option target Priority
    option proto tcp
    option pktsize -128
    option mark Bulk
    option tcpflags ACK

# Don t change the stuff below unless you
# really know what it means )

config classgroup Default
    option classes Priority Express Normal Bulk
    option default Normal
```

```
config class Priority
    option packetsize 400
    option maxsize 400
    option avgrate 10
    option priority 20
config class Priority_down
    option packetsize 1000
    option avgrate 10

config class Express
    option packetsize 1000
    option maxsize 800
    option avgrate 50
    option priority 10

config class Normal
    option packetsize 1500
    option packetdelay 100
    option avgrate 10
    option priority 5
config class Normal_down
    option avgrate 20

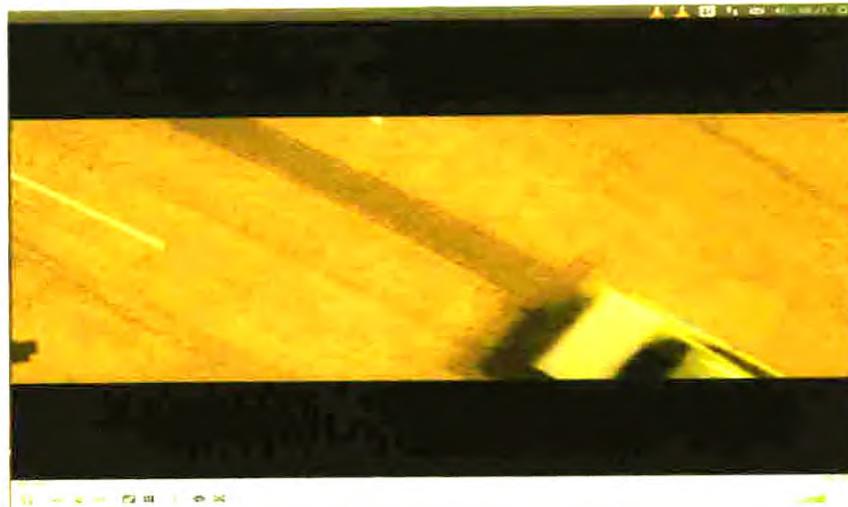
config class Bulk
    option avgrate 1
    option packetdelay 200
```

Evaluación del flujo Streaming con QoS

Datos regidos en el servidor de iperf.

1. Datos regidos en el servidor de iperf.

Con una utilización del 50% = 9 Mbps



```
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.
```

```
C:\Users\CITIC_2>iperf -s -u -i 1 -p 1234
```

```
-----
Server listening on UDP port 1234
Receiving 1470 byte datagrams
UDP buffer size: 64.0 KByte (default)
-----
```

```
[ 3] local 192.168.18.78 port 1234 connected with 192.168.18.79 port 53939
[ ID] Interval          Transfer          Bandwidth          Jitter    Lost/Total
Datagrams
[ 3] 0.0- 1.0 sec    1.08 MBytes    9.04 Mbits/sec    1.074 ms     0/ 769 (0%)
[ 3] 1.0- 2.0 sec    1.07 MBytes    9.01 Mbits/sec    0.831 ms     0/ 766 (0%)
[ 3] 2.0- 3.0 sec   1013 KBytes    8.30 Mbits/sec    1.354 ms     0/ 706 (0%)
[ 3] 3.0- 4.0 sec    1.06 MBytes    8.87 Mbits/sec    1.258 ms     0/ 754 (0%)
[ 3] 4.0- 5.0 sec    1.16 MBytes    9.70 Mbits/sec    0.996 ms     0/ 825 (0%)
[ 3] 5.0- 6.0 sec    1.07 MBytes    8.95 Mbits/sec    1.161 ms     0/ 761 (0%)
[ 3] 6.0- 7.0 sec    1.10 MBytes    9.20 Mbits/sec    0.945 ms     0/ 782 (0%)
[ 3] 7.0- 8.0 sec    1.00 MBytes    8.41 Mbits/sec    1.093 ms     0/ 715 (0%)
[ 3] 8.0- 9.0 sec     916 KBytes    7.50 Mbits/sec    1.006 ms    30/ 668
(4.5%)
[ 3] 9.0-10.0 sec    1.11 MBytes    9.30 Mbits/sec    1.191 ms     5/ 796
(0.63%)
[ 3] 10.0-11.0 sec   1.12 MBytes    9.36 Mbits/sec    1.044 ms     0/ 796 (0%)
[ 3] 11.0-12.0 sec   1.14 MBytes    9.58 Mbits/sec    1.069 ms     0/ 815 (0%)
[ 3] 12.0-13.0 sec   1005 KBytes    8.23 Mbits/sec    0.995 ms     0/ 700 (0%)
```

```

[ 3] 13.0-14.0 sec  1.08 MBytes  9.09 Mbits/sec  0.806 ms    0/   773 (0%)
[ 3] 14.0-15.0 sec  1.03 MBytes  8.66 Mbits/sec  1.058 ms    0/   736 (0%)
[ 3] 15.0-16.0 sec   725 KBytes  5.94 Mbits/sec  2.624 ms    0/   505 (0%)
[ 3] 16.0-17.0 sec   224 KBytes  1.83 Mbits/sec  4.543 ms   230/   386 (60%)
[ 3] 17.0-18.0 sec   233 KBytes  1.91 Mbits/sec  5.798 ms   621/   783 (79%)
[ 3] 18.0-19.0 sec   877 KBytes  7.19 Mbits/sec  1.533 ms   781/  1392 (56%)
[ 3] 19.0-20.0 sec   952 KBytes  7.80 Mbits/sec  1.275 ms    84/   747 (11%)
[ 3] 20.0-21.0 sec  1019 KBytes  8.35 Mbits/sec  1.567 ms    45/   755 (6%)
[ 3] 21.0-22.0 sec   979 KBytes  8.02 Mbits/sec  1.020 ms    74/   756
(9.8%)
[ 3] 22.0-23.0 sec   973 KBytes  7.97 Mbits/sec  1.353 ms   108/   786 (14%)
[ 3] 23.0-24.0 sec   952 KBytes  7.80 Mbits/sec  1.410 ms    65/   728
(8.9%)
[ 3] 24.0-25.0 sec   593 KBytes  4.86 Mbits/sec  2.635 ms   156/   569 (27%)
[ 3] 25.0-26.0 sec   759 KBytes  6.22 Mbits/sec  2.110 ms   359/   888 (40%)
[ 3] 26.0-27.0 sec   1.00 MBytes  8.42 Mbits/sec  1.192 ms   153/   869 (18%)
[ 3] 27.0-28.0 sec   716 KBytes  5.87 Mbits/sec  1.699 ms   197/   696 (28%)
[ 3] 28.0-29.0 sec   754 KBytes  6.17 Mbits/sec  1.198 ms   265/   790 (34%)
[ 3] 29.0-30.0 sec   788 KBytes  6.46 Mbits/sec  1.478 ms   198/   747 (27%)
[ 3] 0.0-30.2 sec  27.4 MBytes  7.60 Mbits/sec  1.330 ms 3430/22971 (15%)
[ 3] 0.0-30.2 sec  1 datagrams received out-of-order
read failed: Connection reset by peer

```

2. Datos regidos en el servidor de iperf.

Con una utilización del 75% = 13.5 Mbps



Microsoft Windows [Versión 6.1.7601]
 Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

```
C:\Users\CITIC_2>iperf -s -u -i 1 -p 1234
```

```
-----
Server listening on UDP port 1234
Receiving 1470 byte datagrams
UDP buffer size: 64.0 KByte (default)
-----
```

```
[ 3] local 192.168.18.78 port 1234 connected with 192.168.18.79 port 45330
```

[ID]	Interval	Transfer	Bandwidth	Jitter	Lost/Total	Datagrams
[3]	0.0- 1.0 sec	636 KBytes	5.21 Mb/s	2.056 ms	135/ 578	(23%)
[3]	1.0- 2.0 sec	482 KBytes	3.95 Mb/s	4.252 ms	950/ 1286	(74%)
[3]	2.0- 3.0 sec	1.03 MBytes	8.61 Mb/s	1.447 ms	580/ 1312	(44%)
[3]	3.0- 4.0 sec	1.10 MBytes	9.21 Mb/s	1.159 ms	420/ 1203	(35%)
[3]	4.0- 5.0 sec	1.58 MBytes	13.3 Mb/s	0.953 ms	186/ 1315	(14%)
[3]	5.0- 6.0 sec	1.40 MBytes	11.8 Mb/s	1.140 ms	34/ 1036	(3.3%)
[3]	6.0- 7.0 sec	1.07 MBytes	8.95 Mb/s	1.133 ms	316/ 1077	(29%)
[3]	7.0- 8.0 sec	1.51 MBytes	12.7 Mb/s	0.918 ms	152/ 1231	(12%)
[3]	8.0- 9.0 sec	1.39 MBytes	11.7 Mb/s	1.117 ms	146/ 1138	(13%)
[3]	9.0-10.0 sec	1.55 MBytes	13.0 Mb/s	3.422 ms	31/ 1136	(2.7%)
[3]	10.0-11.0 sec	851 KBytes	6.97 Mb/s	1.444 ms	514/ 1107	(46%)
[3]	11.0-12.0 sec	1.18 MBytes	9.93 Mb/s	0.653 ms	382/ 1226	(31%)
[3]	12.0-13.0 sec	1.60 MBytes	13.4 Mb/s	0.809 ms	30/ 1168	(2.6%)
[3]	13.0-14.0 sec	1.53 MBytes	12.9 Mb/s	1.282 ms	29/ 1122	(2.6%)
[3]	14.0-15.0 sec	1.64 MBytes	13.7 Mb/s	0.903 ms	13/ 1182	(1.1%)
[3]	15.0-16.0 sec	1.28 MBytes	10.7 Mb/s	0.937 ms	210/ 1120	(19%)
[3]	16.0-17.0 sec	1.40 MBytes	11.7 Mb/s	0.847 ms	141/ 1138	(12%)
[3]	17.0-18.0 sec	883 KBytes	7.23 Mb/s	3.708 ms	28/ 643	(4.4%)
[3]	18.0-19.0 sec	256 KBytes	2.09 Mb/s	5.658 ms	622/ 800	(78%)
[3]	19.0-20.0 sec	277 KBytes	2.27 Mb/s	6.129 ms	1015/ 1208	(84%)
[3]	20.0-21.0 sec	444 KBytes	3.63 Mb/s	2.272 ms	1449/ 1758	(82%)
[3]	21.0-22.0 sec	1.21 MBytes	10.1 Mb/s	0.817 ms	443/ 1306	(34%)
[3]	22.0-23.0 sec	1.03 MBytes	8.68 Mb/s	1.041 ms	417/ 1155	(36%)
[3]	23.0-24.0 sec	1.03 MBytes	8.66 Mb/s	1.316 ms	328/ 1064	(31%)
[3]	24.0-25.0 sec	932 KBytes	7.63 Mb/s	3.160 ms	258/ 907	(28%)
[3]	25.0-26.0 sec	650 KBytes	5.33 Mb/s	1.486 ms	896/ 1349	(66%)
[3]	26.0-27.0 sec	535 KBytes	4.39 Mb/s	4.856 ms	454/ 827	(55%)
[3]	27.0-28.0 sec	459 KBytes	3.76 Mb/s	3.242 ms	1074/ 1394	(77%)
[3]	28.0-29.0 sec	949 KBytes	7.77 Mb/s	1.149 ms	631/ 1292	(49%)
[3]	29.0-30.0 sec	1.45 MBytes	12.1 Mb/s	1.176 ms	167/ 1200	(14%)
[3]	0.0-30.2 sec	31.4 MBytes	8.70 Mb/s	1.557 ms	12065/34440	(35%)

3. Datos regidos en el servidor de iperf.

Con una utilización del 90% = 16.65 Mbps



Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation Reservados todos los derechos

C:\Users\CITIC_2>iperf s -u 1 1 -p 1234

Server listening on UDP port 1234
Receiving 1470 byte datagrams
UDP buffer size 64 0 KByte (default)

[ID]	Interval	Transfer	Bandwidth	Jitter	Lost/Total Datagrams
[3]	local 192 168 18 78 port 1234 connected with 192 168 18 79 port 33913				
[3]	0 0- 1 0 sec	1 77 MBytes	14 9 Mbits/sec	0 633 ms	9/ 1272 (0 71%)
[3]	1 0- 2 0 sec	1 57 MBytes	13 1 Mbits/sec	0 834 ms	268/ 1386 (19%)
[3]	2 0- 3 0 sec	1 75 MBytes	14 6 Mbits/sec	0 557 ms	197/ 1442 (14%)
[3]	3 0- 4 0 sec	1 85 MBytes	15 5 Mbits/sec	0 982 ms	58/ 1377 (4 2%)
[3]	4 0- 5 0 sec	1 62 MBytes	13 6 Mbits/sec	0 899 ms	268/ 1421 (19%)
[3]	5 0- 6 0 sec	1 42 MBytes	11 9 Mbits/sec	0 843 ms	407/ 1423 (29%)
[3]	6 0- 7 0 sec	972 KBytes	7 96 Mbits/sec	1 614 ms	669/ 1346 (50%)
[3]	7 0- 8 0 sec	930 KBytes	7 62 Mbits/sec	1 576 ms	538/ 1186 (45%)
[3]	8 0- 9 0 sec	1013 KBytes	8 30 Mbits/sec	1 136 ms	904/ 1610 (56%)
[3]	9 0-10 0 sec	1 01 MBytes	8 50 Mbits/sec	0 976 ms	711/ 1434 (50%)
[3]	10 0-11 0 sec	1 48 MBytes	12 5 Mbits/sec	0 600 ms	443/ 1502 (29%)
[3]	11 0-12 0 sec	1 47 MBytes	12 4 Mbits/sec	0 842 ms	364/ 1416 (26%)
[3]	12 0-13 0 sec	1 88 MBytes	15 7 Mbits/sec	0 718 ms	100/ 1438 (7%)
[3]	13 0-14 0 sec	1 75 MBytes	14 7 Mbits/sec	0 884 ms	125/ 1375 (9 1%)
[3]	14 0-15 0 sec	1 34 MBytes	11 3 Mbits/sec	0 957 ms	470/ 1427 (33%)
[3]	15 0-16 0 sec	1 45 MBytes	12 1 Mbits/sec	1 163 ms	380/ 1413 (27%)
[3]	16 0-17 0 sec	1 35 MBytes	11 3 Mbits/sec	0 787 ms	436/ 1401 (31%)
[3]	17 0-18 0 sec	1 29 MBytes	10 8 Mbits/sec	1 164 ms	474/ 1395 (34%)
[3]	18 0-19 0 sec	1 89 MBytes	15 8 Mbits/sec	0 738 ms	143/ 1488 (9 6%)
[3]	19 0-20 0 sec	1 76 MBytes	14 8 Mbits/sec	0 578 ms	155/ 1412 (11%)
[3]	20 0-21 0 sec	1 92 MBytes	16 1 Mbits/sec	0 583 ms	70/ 1439 (4 9%)
[3]	21 0-22 0 sec	1 16 MBytes	9 74 Mbits/sec	2 183 ms	499/ 1327 (38%)
[3]	22 0-23 0 sec	1 01 MBytes	8 48 Mbits/sec	1 920 ms	603/ 1324 (46%)
[3]	23 0-24 0 sec	1 48 MBytes	12 4 Mbits/sec	0 628 ms	543/ 1596 (34%)
[3]	24 0-25 0 sec	1 83 MBytes	15 4 Mbits/sec	0 751 ms	92/ 1399 (6 6%)
[3]	25 0-26 0 sec	1 75 MBytes	14 7 Mbits/sec	0 769 ms	165/ 1415 (12%)
[3]	26 0-27 0 sec	1 76 MBytes	14 8 Mbits/sec	1 150 ms	83/ 1338 (6 2%)
[3]	27 0-28 0 sec	1 32 MBytes	11 1 Mbits/sec	1 016 ms	496/ 1441 (34%)
[3]	28 0-29 0 sec	1 30 MBytes	10 9 Mbits/sec	0 923 ms	444/ 1372 (32%)
[3]	29 0-30 0 sec	1 60 MBytes	13 4 Mbits/sec	0 637 ms	390/ 1530 (25%)
[3]	0 0-30 1 sec	44 8 MBytes	12 5 Mbits/sec	0 632 ms	10504/42493 (25%)
[3]	0 0 30 1 sec	1 datagrams received out of-order			

4 Datos regidos en el servidor de iperf

Con una utilización del 95% = 17 57 Mbps



Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

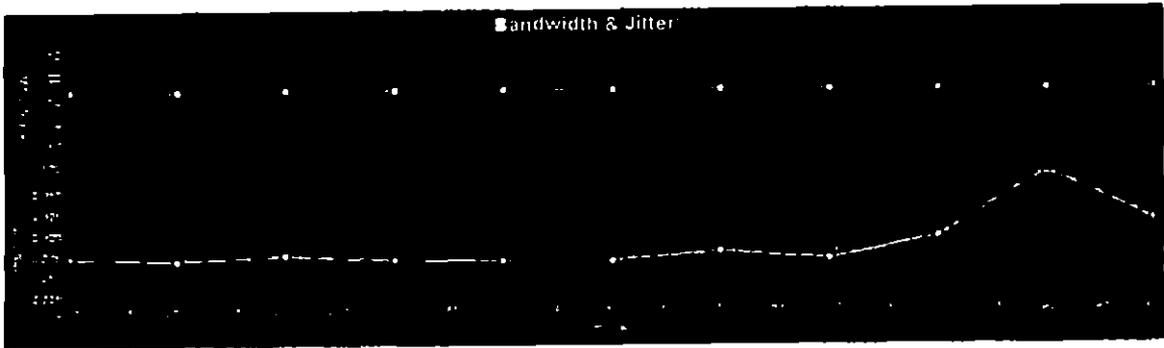
```
C:\Users\CITIC_2>iperf -s -u -i 1 -p 1234
```

```
-----  
Server listening on UDP port 1234  
Receiving 1470 byte datagrams  
UDP buffer size: 64.0 KByte (default)  
-----
```

```
[ 3] local 192.168.18.78 port 1234 connected with 192.168.18.79 port 44649  
[ ID] Interval          Transfer          Bandwidth          Jitter    Lost/Total Datagrams  
  
[ 3] 0.0- 1.0 sec      927 KBytes       7.60 Mbits/sec     1.705 ms   474/ 1120 (42%)  
[ 3] 1.0- 2.0 sec      627 KBytes       5.14 Mbits/sec     2.943 ms   910/ 1347 (68%)  
[ 3] 2.0- 3.0 sec      643 KBytes       5.27 Mbits/sec     2.374 ms 1099/ 1547 (71%)  
[ 3] 3.0- 4.0 sec      696 KBytes       5.70 Mbits/sec     2.840 ms 1033/ 1518 (68%)  
[ 3] 4.0- 5.0 sec      824 KBytes       6.75 Mbits/sec     1.857 ms   996/ 1570 (63%)  
[ 3] 5.0- 6.0 sec     1.23 MBytes      10.3 Mbits/sec     0.907 ms   792/ 1668 (47%)  
[ 3] 6.0- 7.0 sec     1.42 MBytes      11.9 Mbits/sec     1.283 ms   431/ 1442 (30%)  
[ 3] 7.0- 8.0 sec     1.29 MBytes      10.8 Mbits/sec     1.379 ms   496/ 1417 (35%)  
[ 3] 8.0- 9.0 sec     1.28 MBytes      10.7 Mbits/sec     0.989 ms   666/ 1576 (42%)  
[ 3] 9.0-10.0 sec     1.43 MBytes      12.0 Mbits/sec     0.930 ms   496/ 1518 (33%)  
[ 3] 10.0-11.0 sec    1.47 MBytes      12.4 Mbits/sec     1.242 ms   466/ 1518 (31%)  
[ 3] 11.0-12.0 sec    1.84 MBytes      15.4 Mbits/sec     0.709 ms   237/ 1550 (15%)  
[ 3] 12.0-13.0 sec    2.08 MBytes      17.5 Mbits/sec     0.717 ms     7/ 1491 (0.47%)  
[ 3] 13.0-14.0 sec    1.51 MBytes      12.6 Mbits/sec     1.090 ms   351/ 1425 (25%)  
[ 3] 14.0-15.0 sec    1.45 MBytes      12.2 Mbits/sec     1.397 ms   451/ 1487 (30%)  
[ 3] 15.0-16.0 sec    1.56 MBytes      13.1 Mbits/sec     0.615 ms   479/ 1590 (30%)  
[ 3] 16.0-17.0 sec    2.06 MBytes      17.2 Mbits/sec     0.649 ms    32/ 1498 (2.1%)  
[ 3] 17.0-18.0 sec    2.04 MBytes      17.1 Mbits/sec     0.693 ms    40/ 1494 (2.7%)  
[ 3] 18.0-19.0 sec    1.14 MBytes      9.58 Mbits/sec     1.256 ms   535/ 1350 (40%)  
[ 3] 19.0-20.0 sec    1.25 MBytes      10.5 Mbits/sec     0.839 ms   681/ 1572 (43%)  
[ 3] 20.0-21.0 sec    1.47 MBytes      12.3 Mbits/sec     1.080 ms   434/ 1484 (29%)  
[ 3] 21.0-22.0 sec    1.29 MBytes      10.8 Mbits/sec     0.803 ms   573/ 1490 (38%)  
[ 3] 22.0-23.0 sec    1.36 MBytes      11.4 Mbits/sec     0.623 ms   599/ 1567 (38%)  
[ 3] 23.0-24.0 sec    2.05 MBytes      17.2 Mbits/sec     0.644 ms    37/ 1498 (2.5%)  
[ 3] 24.0-25.0 sec    1.46 MBytes      12.2 Mbits/sec     1.025 ms   327/ 1365 (24%)  
[ 3] 25.0-26.0 sec    1.21 MBytes      10.2 Mbits/sec     0.894 ms   694/ 1560 (44%)  
[ 3] 26.0-27.0 sec    1.23 MBytes      10.3 Mbits/sec     0.976 ms   594/ 1474 (40%)  
[ 3] 27.0-28.0 sec    1.25 MBytes      10.5 Mbits/sec     1.081 ms   636/ 1531 (42%)  
[ 3] 28.0-29.0 sec    1.19 MBytes      10.0 Mbits/sec     1.656 ms   427/ 1277 (33%)  
[ 3] 29.0-30.0 sec    962 KBytes       7.88 Mbits/sec     1.443 ms   852/ 1522 (56%)  
[ 3] 0.0-30.4 sec    40.3 MBytes      11.1 Mbits/sec    12.132 ms 16072/44842 (36%)  
[ 3] 0.0-30.4 sec    1 datagrams received out-of-order
```

Pruebas de VoIP sin QoS

1 Datos regidos en el servidor de jperf con 1 flujo



```
bin/iperf.exe s u P 0 i 1 p 1234 -l 200 0B -f k
```

```
-----  
Server listening on UDP port 1234  
Receiving 200 byte datagrams  
UDP buffer size 8 00 KByte (default)  
-----
```

```
OpenSCManager failed - Acceso denegado (0x5)  
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 2137  
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams  
[112] 0 0- 1 0 sec  7 81 KBytes  64 0 Kbits/sec  0 246 ms  1348221507/ 40  
(3 4e+009%)  
[112] 1 0- 2 0 sec  7 81 KBytes  64 0 Kbits/sec  0 286 ms  0/ 40 (0%)  
[112] 2 0- 3 0 sec  7 81 KBytes  64 0 Kbits/sec  0 274 ms  0/ 40 (0%)  
[112] 3 0- 4 0 sec  7 81 KBytes  64 0 Kbits/sec  0 274 ms  0/ 40 (0%)  
[112] 4 0- 5 0 sec  7 81 KBytes  64 0 Kbits/sec  0 274 ms  0/ 40 (0%)  
[112] 5 0- 6 0 sec  7 81 KBytes  64 0 Kbits/sec  0 274 ms  0/ 40 (0%)  
[112] 6 0- 7 0 sec  7 81 KBytes  64 0 Kbits/sec  0 274 ms  0/ 40 (0%)  
[112] 7 0- 8 0 sec  7 81 KBytes  64 0 Kbits/sec  0 274 ms  0/ 40 (0%)  
[112] 8 0- 9 0 sec  7 81 KBytes  64 0 Kbits/sec  0 267 ms  0/ 40 (0%)  
[112] 9 0-10 0 sec  7 81 KBytes  64 0 Kbits/sec  0 275 ms  0/ 40 (0%)  
[112] 10 0-11 0 sec  7 81 KBytes  64 0 Kbits/sec  0 267 ms  0/ 40 (0%)  
[112] 11 0-12 0 sec  7 81 KBytes  64 0 Kbits/sec  0 266 ms  0/ 40 (0%)  
[112] 12 0-13 0 sec  7 81 KBytes  64 0 Kbits/sec  0 273 ms  0/ 40 (0%)  
[112] 13 0-14 0 sec  7 81 KBytes  64 0 Kbits/sec  0 273 ms  0/ 40 (0%)  
[112] 14 0-15 0 sec  7 81 KBytes  64 0 Kbits/sec  0 274 ms  0/ 40 (0%)  
[112] 15 0-16 0 sec  7 81 KBytes  64 0 Kbits/sec  0 561 ms  0/ 40 (0%)  
[112] 16 0-17 0 sec  7 81 KBytes  64 0 Kbits/sec  0 288 ms  0/ 40 (0%)  
[112] 17 0-18 0 sec  7 81 KBytes  64 0 Kbits/sec  0 268 ms  0/ 40 (0%)  
[112] 18 0-19 0 sec  7 81 KBytes  64 0 Kbits/sec  0 327 ms  0/ 40 (0%)  
[112] 19 0-20 0 sec  7 81 KBytes  64 0 Kbits/sec  0 271 ms  0/ 40 (0%)  
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams  
[112] 20 0-21 0 sec  7 81 KBytes  64 0 Kbits/sec  0 267 ms  0/ 40 (0%)  
[112] 21 0-22 0 sec  7 81 KBytes  64 0 Kbits/sec  0 273 ms  0/ 40 (0%)  
[112] 22 0-23 0 sec  7 81 KBytes  64 0 Kbits/sec  0 267 ms  0/ 40 (0%)  
[112] 23 0-24 0 sec  7 81 KBytes  64 0 Kbits/sec  0 266 ms  0/ 40 (0%)  
[112] 24 0-25 0 sec  7 81 KBytes  64 0 Kbits/sec  0 266 ms  0/ 40 (0%)  
[112] 25 0-26 0 sec  7 81 KBytes  64 0 Kbits/sec  0 277 ms  0/ 40 (0%)  
[112] 26 0-27 0 sec  7 81 KBytes  64 0 Kbits/sec  0 267 ms  0/ 40 (0%)
```

```

[112] 27 0-28 0 sec 7 81 KBytes 64 0 Kbits/sec 0 294 ms 0/ 40 (0%)
[112] 28 0-29 0 sec 7 81 KBytes 64 0 Kbits/sec 0 371 ms 0/ 40 (0%)
[112] 29 0-30 0 sec 7 81 KBytes 64 0 Kbits/sec 0 313 ms 0/ 40 (0%)
[112] 0 0-30 0 sec 235 KBytes 64 0 Kbits/sec 0 275 ms 0/ 1202 (0%)

```

2 Datos regidos en el servidor de jperf con 3 flujos



```
bin/iperf exe -s -u -P 0 -i 1 -p 1234 -l 200 0B -f k
```

```

-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----

```

```

OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 2140
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec 7 81 KBytes 64 0 Kbits/sec 0 312 ms 1348221507/ 40
(3 4e+009%)
[112] 1 0- 2 0 sec 7 81 KBytes 64 0 Kbits/sec 0 999 ms 0/ 40 (0%)
[112] 2 0- 3 0 sec 7 81 KBytes 64 0 Kbits/sec 0 551 ms 0/ 40 (0%)
[112] 3 0- 4 0 sec 7 81 KBytes 64 0 Kbits/sec 0 484 ms 0/ 40 (0%)
[112] 4 0- 5 0 sec 7 81 KBytes 64 0 Kbits/sec 0 361 ms 0/ 40 (0%)
[112] 5 0- 6 0 sec 7 81 KBytes 64 0 Kbits/sec 0 344 ms 0/ 40 (0%)
[112] 6 0- 7 0 sec 7 81 KBytes 64 0 Kbits/sec 0 303 ms 0/ 40 (0%)
[112] 7 0- 8 0 sec 7 81 KBytes 64 0 Kbits/sec 0 306 ms 0/ 40 (0%)
[112] 8 0- 9 0 sec 7 81 KBytes 64 0 Kbits/sec 0 387 ms 0/ 40 (0%)
[112] 9 0-10 0 sec 7 81 KBytes 64 0 Kbits/sec 0 441 ms 0/ 40 (0%)
[112] 10 0-11 0 sec 7 81 KBytes 64 0 Kbits/sec 0 364 ms 0/ 40 (0%)
[112] 11 0-12 0 sec 7 81 KBytes 64 0 Kbits/sec 0 308 ms 0/ 40 (0%)
[112] 12 0-13 0 sec 7 81 KBytes 64 0 Kbits/sec 0 284 ms 0/ 40 (0%)
[112] 13 0-14 0 sec 7 62 KBytes 62 4 Kbits/sec 0 244 ms 0/ 39 (0%)
[112] 14 0-15 0 sec 8 01 KBytes 65 6 Kbits/sec 0 291 ms 0/ 41 (0%)
[112] 15 0-16 0 sec 7 81 KBytes 64 0 Kbits/sec 0 367 ms 0/ 40 (0%)
[112] 16 0-17 0 sec 7 81 KBytes 64 0 Kbits/sec 0 308 ms 0/ 40 (0%)
[112] 17 0-18 0 sec 7 81 KBytes 64 0 Kbits/sec 0 288 ms 0/ 40 (0%)
[112] 18 0-19 0 sec 7 81 KBytes 64 0 Kbits/sec 0 275 ms 0/ 40 (0%)
[112] 19 0-20 0 sec 7 81 KBytes 64 0 Kbits/sec 0 327 ms 0/ 40 (0%)
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 20 0-21 0 sec 7 81 KBytes 64 0 Kbits/sec 0 446 ms 0/ 40 (0%)
[112] 21 0-22 0 sec 7 81 KBytes 64 0 Kbits/sec 0 498 ms 0/ 40 (0%)
[112] 22 0-23 0 sec 7 81 KBytes 64 0 Kbits/sec 0 284 ms 0/ 40 (0%)
[112] 23 0-24 0 sec 7 81 KBytes 64 0 Kbits/sec 0 418 ms 0/ 40 (0%)
[112] 24 0-25 0 sec 7 81 KBytes 64 0 Kbits/sec 0 366 ms 0/ 40 (0%)
[112] 25 0-26 0 sec 7 81 KBytes 64 0 Kbits/sec 0 487 ms 0/ 40 (0%)

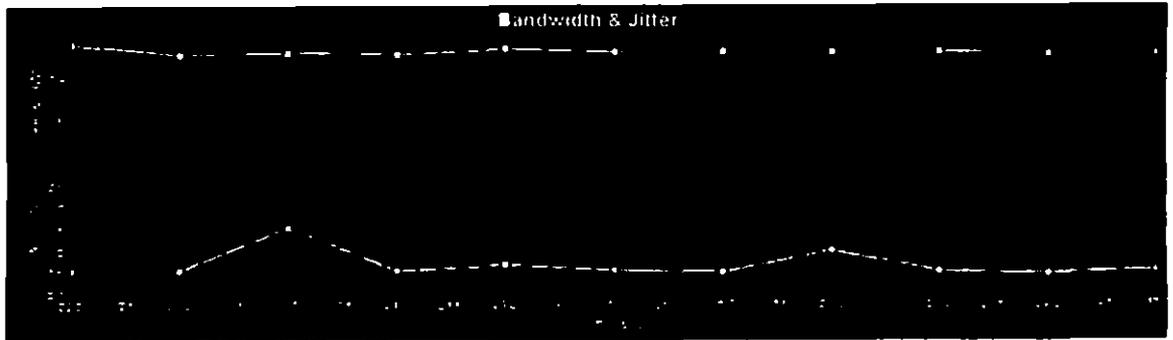
```

```

[112] 26 0-27 0 sec 7 81 KBytes 64 0 Kbits/sec 0 324 ms 0/ 40 (0%)
[112] 27 0-28 0 sec 7 81 KBytes 64 0 Kbits/sec 0 414 ms 0/ 40 (0%)
[112] 28 0-29 0 sec 7 81 KBytes 64 0 Kbits/sec 0 364 ms 0/ 40 (0%)
[112] 29 0-30 0 sec 7 81 KBytes 64 0 Kbits/sec 0 357 ms 0/ 40 (0%)
[112] 0 0 30 1 sec 235 KBytes 64 0 Kbits/sec 0 704 ms 0/ 1202 (0%)

```

3 Datos regidos en el servidor de jperf con 5 flujos



```
bin/iperf exe -s -u P 0 i 1 -p 1234 1 200 0B -f k
```

```

-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
--

```

```

OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 2145
[ ID] Interval          Transfer          Bandwidth          Jitter    Lost/Total Datagrams
[112] 0 0- 1 0 sec 7 62 KBytes 62 4 Kbits/sec 0 482 ms 1348221507/ 39
(3 5e+009%)
[112] 1 0- 2 0 sec 7 81 KBytes 64 0 Kbits/sec 0 717 ms 0/ 40 (0%)
[112] 2 0- 3 0 sec 7 42 KBytes 60 8 Kbits/sec 1 595 ms 0/ 38 (0%)
[112] 3 0- 4 0 sec 7 42 KBytes 60 8 Kbits/sec 0 949 ms 0/ 38 (0%)
[112] 4 0- 5 0 sec 7 81 KBytes 64 0 Kbits/sec 2 672 ms 0/ 40 (0%)
[112] 5 0- 6 0 sec 8 01 KBytes 65 6 Kbits/sec 0 932 ms 0/ 41 (0%)
[112] 6 0- 7 0 sec 7 62 KBytes 62 4 Kbits/sec 2 591 ms 0/ 39 (0%)
[112] 7 0- 8 0 sec 8 20 KBytes 67 2 Kbits/sec 0 665 ms 0/ 42 (0%)
[112] 8 0- 9 0 sec 7 42 KBytes 60 8 Kbits/sec 0 629 ms 0/ 38 (0%)
[112] 9 0-10 0 sec 8 01 KBytes 65 6 Kbits/sec 1 254 ms 0/ 41 (0%)
[112] 10 0-11 0 sec 7 62 KBytes 62 4 Kbits/sec 0 612 ms 0/ 39 (0%)
[112] 11 0-12 0 sec 7 81 KBytes 64 0 Kbits/sec 1 877 ms 0/ 40 (0%)
[112] 12 0-13 0 sec 7 62 KBytes 62 4 Kbits/sec 0 789 ms 0/ 39 (0%)
[112] 13 0-14 0 sec 8 20 KBytes 67 2 Kbits/sec 0 599 ms 0/ 42 (0%)
[112] 14 0-15 0 sec 7 62 KBytes 62 4 Kbits/sec 0 608 ms 0/ 39 (0%)
[112] 15 0-16 0 sec 8 01 KBytes 65 6 Kbits/sec 0 493 ms 0/ 41 (0%)
[112] 16 0-17 0 sec 7 62 KBytes 62 4 Kbits/sec 0 723 ms 0/ 39 (0%)
[112] 17 0-18 0 sec 8 01 KBytes 65 6 Kbits/sec 0 642 ms 0/ 41 (0%)
[112] 18 0-19 0 sec 7 23 KBytes 59 2 Kbits/sec 0 320 ms 0/ 37 (0%)
[112] 19 0-20 0 sec 8 40 KBytes 68 8 Kbits/sec 0 532 ms 0/ 43 (0%)
[ ID] Interval          Transfer          Bandwidth          Jitter    Lost/Total Datagrams
[112] 20 0-21 0 sec 7 62 KBytes 62 4 Kbits/sec 0 520 ms 0/ 39 (0%)
[112] 21 0-22 0 sec 7 81 KBytes 64 0 Kbits/sec 1 518 ms 0/ 40 (0%)
[112] 22 0-23 0 sec 7 62 KBytes 62 4 Kbits/sec 0 540 ms 0/ 39 (0%)
[112] 23 0-24 0 sec 8 01 KBytes 65 6 Kbits/sec 0 658 ms 0/ 41 (0%)
[112] 24 0-25 0 sec 7 81 KBytes 64 0 Kbits/sec 0 542 ms 0/ 40 (0%)

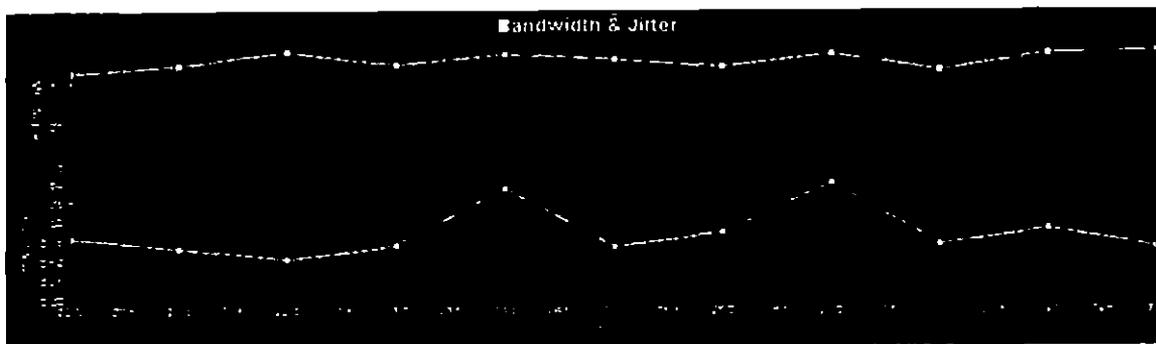
```

```

[112] 25 0-26 0 sec 7 81 KBytes 64 0 Kbits/sec 0 490 ms 0/ 40 (0%)
[112] 26 0-27 0 sec 7 81 KBytes 64 0 Kbits/sec 0 991 ms 0/ 40 (0%)
[112] 27 0-28 0 sec 7 81 KBytes 64 0 Kbits/sec 0 516 ms 0/ 40 (0%)
[112] 28 0-29 0 sec 7 62 KBytes 62 4 Kbits/sec 0 453 ms 0/ 39 (0%)
[112] 29 0-30 0 sec 7 62 KBytes 62 4 Kbits/sec 0 535 ms 0/ 39 (0%)
[112] 0 0-30 0 sec 233 KBytes 63 7 Kbits/sec 0 741 ms 0/ 1195 (0%)

```

4 Datos regidos en el servidor de jperf con 10 flujos



```
bin/iperf.exe -s -u -P 0 -i 1 -p 1234 1 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

```
OpenSCManager failed Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 2235
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec 7 42 KBytes 60 8 Kbits/sec 0 954 ms 1348221507/ 38
(3 5e+009%)
[112] 1 0- 2 0 sec 7 23 KBytes 59 2 Kbits/sec 1 157 ms 0/ 37 (0%)
[112] 2 0- 3 0 sec 8 40 KBytes 68 8 Kbits/sec 0 943 ms 0/ 43 (0%)
[112] 3 0- 4 0 sec 7 23 KBytes 59 2 Kbits/sec 1 181 ms 0/ 37 (0%)
[112] 4 0- 5 0 sec 8 40 KBytes 68 8 Kbits/sec 0 686 ms 0/ 43 (0%)
[112] 5 0- 6 0 sec 7 42 KBytes 60 8 Kbits/sec 1 383 ms 0/ 38 (0%)
[112] 6 0- 7 0 sec 8 40 KBytes 68 8 Kbits/sec 1 010 ms 0/ 43 (0%)
[112] 7 0- 8 0 sec 7 03 KBytes 57 6 Kbits/sec 1 002 ms 0/ 36 (0%)
[112] 8 0- 9 0 sec 8 40 KBytes 68 8 Kbits/sec 0 650 ms 0/ 43 (0%)
[112] 9 0-10 0 sec 7 81 KBytes 64 0 Kbits/sec 1 022 ms 0/ 40 (0%)
[112] 10 0-11 0 sec 7 23 KBytes 59 2 Kbits/sec 0 667 ms 0/ 37 (0%)
[112] 11 0-12 0 sec 7 81 KBytes 64 0 Kbits/sec 0 568 ms 0/ 40 (0%)
[112] 12 0-13 0 sec 8 40 KBytes 68 8 Kbits/sec 0 982 ms 0/ 43 (0%)
[112] 13 0-14 0 sec 7 23 KBytes 59 2 Kbits/sec 0 871 ms 0/ 37 (0%)
[112] 14 0-15 0 sec 8 59 KBytes 70 4 Kbits/sec 0 795 ms 0/ 44 (0%)
[112] 15 0-16 0 sec 7 42 KBytes 60 8 Kbits/sec 0 944 ms 0/ 38 (0%)
[112] 16 0-17 0 sec 7 62 KBytes 62 4 Kbits/sec 0 772 ms 0/ 39 (0%)
[112] 17 0-18 0 sec 7 62 KBytes 62 4 Kbits/sec 0 873 ms 0/ 39 (0%)
[112] 18 0-19 0 sec 8 59 KBytes 70 4 Kbits/sec 1 227 ms 0/ 44 (0%)
[112] 19 0-20 0 sec 6 84 KBytes 56 0 Kbits/sec 0 775 ms 0/ 35 (0%)
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 20 0 21 0 sec 7 42 KBytes 60 8 Kbits/sec 0 645 ms 0/ 38 (0%)
[112] 21 0 22 0 sec 8 40 KBytes 68 8 Kbits/sec 0 504 ms 0/ 43 (0%)
[112] 22 0-23 0 sec 7 42 KBytes 60 8 Kbits/sec 0 682 ms 0/ 38 (0%)
[112] 23 0-24 0 sec 8 20 KBytes 67 2 Kbits/sec 1 414 ms 0/ 42 (0%)
[112] 24 0 25 0 sec 7 81 KBytes 64 0 Kbits/sec 0 657 ms 0/ 40 (0%)
[112] 25 0-26 0 sec 7 23 KBytes 59 2 Kbits/sec 0 842 ms 0/ 37 (0%)

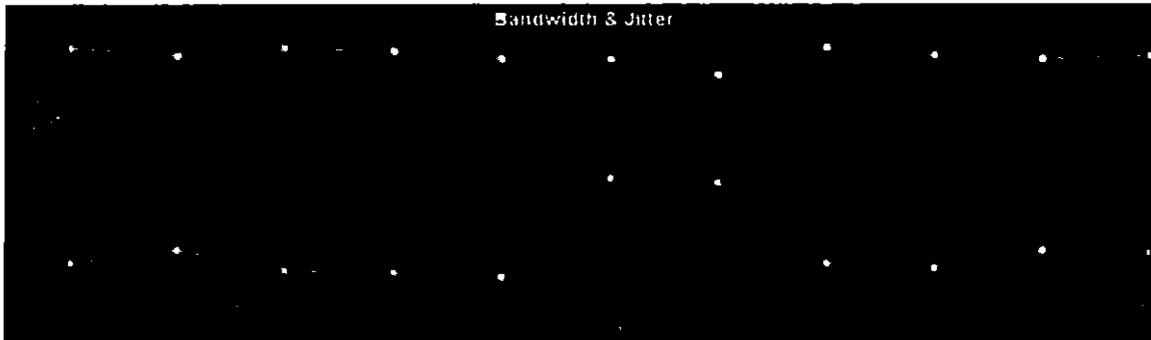
```

```

[112] 26 0-27 0 sec  8 20 KBytes  67 2 Kbits/sec  1 485 ms  0/  42 (0%)
[112] 27 0-28 0 sec  7 03 KBytes  57 6 Kbits/sec  0 684 ms  0/  36 (0%)
[112] 28 0-29 0 sec  8 20 KBytes  67 2 Kbits/sec  0 886 ms  0/  42 (0%)
[112] 29 0-30 0 sec  8 40 KBytes  68 8 Kbits/sec  0 664 ms  0/  43 (0%)
[112]  0 0-30 2 sec 234 KBytes  63 5 Kbits/sec  1 084 ms  0/ 1197 (0%)

```

5 Datos regidos en el servidor de jperf con 15 flujos



```
bin/iperf exe -s -u -P 0 -i 1 -p 1234 -l 200 0B f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size  8 00 KByte (default)
-----
```

```
OpenSCManager failed - Acceso denegado (0x5)
```

```

[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 2476
[ ID] Interval          Transfer      Bandwidth      Jitter    Lost/Total Datagrams
[112] 0 0- 1 0 sec  7 42 KBytes  60 8 Kbits/sec  0 780 ms  1348221507/  38
(3 5e+009%)
[112] 1 0- 2 0 sec  6 25 KBytes  51 2 Kbits/sec  1 634 ms  0/  32 (0%)
[112] 2 0- 3 0 sec  7 42 KBytes  60 8 Kbits/sec  1 029 ms  0/  38 (0%)
[112] 3 0- 4 0 sec  7 62 KBytes  62 4 Kbits/sec  0 626 ms  0/  39 (0%)
[112] 4 0- 5 0 sec  7 81 KBytes  64 0 Kbits/sec  0 880 ms  0/  40 (0%)
[112] 5 0- 6 0 sec  7 62 KBytes  62 4 Kbits/sec  0 686 ms  0/  39 (0%)
[112] 6 0- 7 0 sec  8 59 KBytes  70 4 Kbits/sec  0 720 ms  0/  44 (0%)
[112] 7 0- 8 0 sec  7 62 KBytes  62 4 Kbits/sec  0 483 ms  0/  39 (0%)
[112] 8 0- 9 0 sec  8 01 KBytes  65 6 Kbits/sec  0 824 ms  0/  41 (0%)
[112] 9 0-10 0 sec  6 45 KBytes  52 8 Kbits/sec  0 921 ms  0/  33 (0%)
[112] 10 0-11 0 sec  6 45 KBytes  52 8 Kbits/sec  0 666 ms  0/  33 (0%)
[112] 11 0-12 0 sec  8 01 KBytes  65 6 Kbits/sec  2 989 ms  0/  41 (0%)
[112] 12 0-13 0 sec  6 84 KBytes  56 0 Kbits/sec  2 686 ms  0/  35 (0%)
[112] 13 0-14 0 sec  8 01 KBytes  65 6 Kbits/sec  0 826 ms  0/  41 (0%)
[112] 14 0-15 0 sec  8 01 KBytes  65 6 Kbits/sec  0 517 ms  0/  41 (0%)
[112] 15 0-16 0 sec  8 40 KBytes  68 8 Kbits/sec  0 718 ms  0/  43 (0%)
[112] 16 0-17 0 sec  6 45 KBytes  52 8 Kbits/sec  0 741 ms  0/  33 (0%)
[112] 17 0-18 0 sec  7 81 KBytes  64 0 Kbits/sec  0 668 ms  0/  40 (0%)
[112] 18 0-19 0 sec  8 01 KBytes  65 6 Kbits/sec  0 728 ms  0/  41 (0%)
[112] 19 0-20 0 sec  8 20 KBytes  67 2 Kbits/sec  0 953 ms  0/  42 (0%)
[ ID] Interval          Transfer      Bandwidth      Jitter    Lost/Total Datagrams
[112] 20 0-21 0 sec  7 62 KBytes  62 4 Kbits/sec  1 355 ms  0/  39 (0%)
[112] 21 0-22 0 sec  8 20 KBytes  67 2 Kbits/sec  0 757 ms  0/  42 (0%)
[112] 22 0-23 0 sec  8 01 KBytes  65 6 Kbits/sec  0 691 ms  0/  41 (0%)
[112] 23 0-24 0 sec  7 42 KBytes  60 8 Kbits/sec  0 562 ms  0/  38 (0%)
[112] 24 0-25 0 sec  7 42 KBytes  60 8 Kbits/sec  3 484 ms  0/  38 (0%)
[112] 25 0-26 0 sec  6 25 KBytes  51 2 Kbits/sec  3 346 ms  0/  32 (0%)

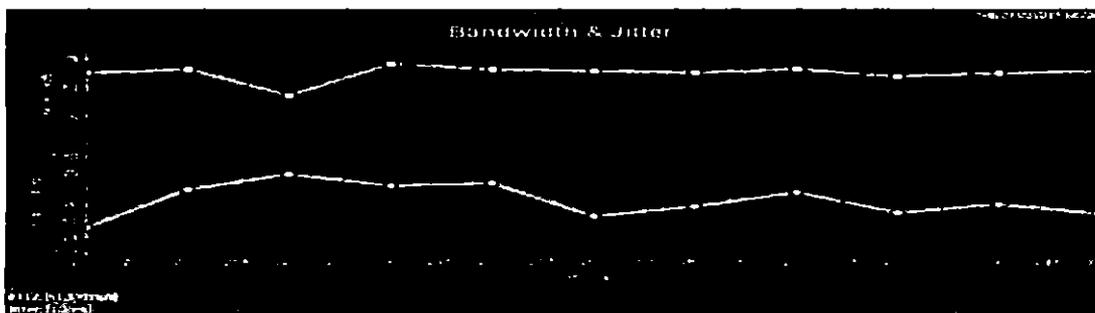
```

```

[112] 26 0-27 0 sec 8 20 KBytes 67 2 Kbits/sec 0 962 ms 0/ 42 (0%)
[112] 27 0-28 0 sec 7 62 KBytes 62 4 Kbits/sec 0 832 ms 0/ 39 (0%)
[112] 28 0-29 0 sec 7 42 KBytes 60 8 Kbits/sec 1 352 ms 0/ 38 (0%)
[112] 29 0-30 0 sec 7 62 KBytes 62 4 Kbits/sec 1 294 ms 0/ 39 (0%)
[112] 0 0-30 1 sec 227 KBytes 61 9 Kbits/sec 1 518 ms 0/ 1163 (0%)

```

6 Datos regidos en el servidor de jperf con 20 flujos



```
bin/jperf exe -s -u -P 0 -i 1 -p 1234 -l 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

```
OpenSCManager failed - Acceso denegado (0x5)
```

```

[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1056
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec 7 81 KBytes 64 0 Kbits/sec 0 842 ms 1852796269/ 40
(4 6e+009%)
[112] 1 0- 2 0 sec 3 13 KBytes 25 6 Kbits/sec 0 710 ms 0/ 16 (0%)
[112] 2 0- 3 0 sec 6 84 KBytes 56 0 Kbits/sec 0 663 ms 0/ 35 (0%)
[112] 3 0- 4 0 sec 7 23 KBytes 59 2 Kbits/sec 0 829 ms 0/ 37 (0%)
[112] 4 0- 5 0 sec 7 42 KBytes 60 8 Kbits/sec 0 953 ms 0/ 38 (0%)
[112] 5 0- 6 0 sec 8 20 KBytes 67 2 Kbits/sec 0 713 ms 0/ 42 (0%)
[112] 6 0- 7 0 sec 8 59 KBytes 70 4 Kbits/sec 0 837 ms 0/ 44 (0%)
[112] 7 0- 8 0 sec 7 62 KBytes 62 4 Kbits/sec 1 144 ms 0/ 39 (0%)
[112] 8 0- 9 0 sec 8 40 KBytes 68 8 Kbits/sec 0 760 ms 0/ 43 (0%)
[112] 9 0-10 0 sec 6 84 KBytes 56 0 Kbits/sec 0 913 ms 0/ 35 (0%)
[112] 10 0-11 0 sec 7 42 KBytes 60 8 Kbits/sec 0 587 ms 0/ 38 (0%)
[112] 11 0-12 0 sec 7 23 KBytes 59 2 Kbits/sec 1 463 ms 0/ 37 (0%)
[112] 12 0-13 0 sec 7 62 KBytes 62 4 Kbits/sec 0 786 ms 0/ 39 (0%)
[112] 13 0-14 0 sec 8 01 KBytes 65 6 Kbits/sec 0 789 ms 0/ 41 (0%)
[112] 14 0-15 0 sec 8 20 KBytes 67 2 Kbits/sec 0 759 ms 0/ 42 (0%)
[112] 15 0-16 0 sec 7 03 KBytes 57 6 Kbits/sec 0 795 ms 0/ 36 (0%)
[112] 16 0-17 0 sec 8 98 KBytes 73 6 Kbits/sec 0 617 ms 0/ 46 (0%)
[112] 17 0-18 0 sec 6 45 KBytes 52 8 Kbits/sec 1 036 ms 0/ 33 (0%)
[112] 18 0-19 0 sec 9 96 KBytes 81 6 Kbits/sec 0 776 ms 0/ 51 (0%)
[112] 19 0-20 0 sec 7 62 KBytes 62 4 Kbits/sec 0 404 ms 0/ 39 (0%)
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 20 0-21 0 sec 8 01 KBytes 65 6 Kbits/sec 0 987 ms 0/ 41 (0%)
[112] 21 0-22 0 sec 5 27 KBytes 43 2 Kbits/sec 1 227 ms 0/ 27 (0%)
[112] 22 0-23 0 sec 8 59 KBytes 70 4 Kbits/sec 1 040 ms 0/ 44 (0%)
[112] 23 0-24 0 sec 8 01 KBytes 65 6 Kbits/sec 1 082 ms 0/ 41 (0%)
[112] 24 0-25 0 sec 7 81 KBytes 64 0 Kbits/sec 0 566 ms 0/ 40 (0%)
[112] 25 0-26 0 sec 7 62 KBytes 62 4 Kbits/sec 0 724 ms 0/ 39 (0%)
[112] 26 0-27 0 sec 8 01 KBytes 65 6 Kbits/sec 0 939 ms 0/ 41 (0%)

```

```

[112] 27 0-28 0 sec 7 23 KBytes 59 2 Kbits/sec 0 628 ms 0/ 37 (0%)
[112] 28 0-29 0 sec 7 62 KBytes 62 4 Kbits/sec 0 748 ms 0/ 39 (0%)
[112] 29 0-30 0 sec 7 81 KBytes 64 0 Kbits/sec 0 612 ms 0/ 40 (0%)
[112] 0 0-30 2 sec 227 KBytes 61 5 Kbits/sec 1 345 ms 0/ 1162 (0%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1073
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0 0 8 sec 6 64 KBytes 64 5 Kbits/sec 1 098 ms 1138/ 1172 (97%)

```

7 Datos regidos en el servidor de jperf con 25 flujos



```
bin/jperf exe s u -P 0 -l 1 -p 1234 -l 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

```

OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1077
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0- 1 0 sec 7 62 KBytes 62 4 Kbits/sec 0 498 ms 1852796269/ 39
(4 8e+009%)
[112] 1 0 2 0 sec 0 20 KBytes 1 60 Kbits/sec 0 545 ms 0/ 1 (0%)
[112] 2 0- 3 0 sec 4 30 KBytes 35 2 Kbits/sec 0 789 ms 0/ 22 (0%)
[112] 3 0- 4 0 sec 7 03 KBytes 57 6 Kbits/sec 3 400 ms 0/ 36 (0%)
[112] 4 0- 5 0 sec 6 45 KBytes 52 8 Kbits/sec 1 179 ms 0/ 33 (0%)
[112] 5 0 6 0 sec 6 64 KBytes 54 4 Kbits/sec 0 666 ms 0/ 34 (0%)
[112] 6 0 7 0 sec 7 42 KBytes 60 8 Kbits/sec 0 920 ms 0/ 38 (0%)
[112] 7 0- 8 0 sec 7 42 KBytes 60 8 Kbits/sec 0 963 ms 0/ 38 (0%)
[112] 8 0- 9 0 sec 6 64 KBytes 54 4 Kbits/sec 0 525 ms 0/ 34 (0%)
[112] 9 0-10 0 sec 9 77 KBytes 80 0 Kbits/sec 0 525 ms 0/ 50 (0%)
[112] 10 0-11 0 sec 6 45 KBytes 52 8 Kbits/sec 0 490 ms 0/ 33 (0%)
[112] 11 0 12 0 sec 9 77 KBytes 80 0 Kbits/sec 0 470 ms 0/ 50 (0%)
[112] 12 0-13 0 sec 6 64 KBytes 54 4 Kbits/sec 1 474 ms 0/ 34 (0%)
[112] 13 0-14 0 sec 9 38 KBytes 76 8 Kbits/sec 0 637 ms 0/ 48 (0%)
[112] 14 0-15 0 sec 6 45 KBytes 52 8 Kbits/sec 0 808 ms 0/ 33 (0%)
[112] 15 0-16 0 sec 6 84 KBytes 56 0 Kbits/sec 0 871 ms 0/ 35 (0%)
[112] 16 0-17 0 sec 9 96 KBytes 81 6 Kbits/sec 0 522 ms 0/ 51 (0%)
[112] 17 0-18 0 sec 6 84 KBytes 56 0 Kbits/sec 0 896 ms 0/ 35 (0%)
[112] 18 0-19 0 sec 6 45 KBytes 52 8 Kbits/sec 1 156 ms 0/ 33 (0%)
[112] 19 0-20 0 sec 8 98 KBytes 73 6 Kbits/sec 0 844 ms 0/ 46 (0%)
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 20 0 21 0 sec 6 25 KBytes 51 2 Kbits/sec 0 736 ms 0/ 32 (0%)
[112] 21 0-22 0 sec 9 57 KBytes 78 4 Kbits/sec 0 531 ms 0/ 49 (0%)
[112] 22 0 23 0 sec 6 84 KBytes 56 0 Kbits/sec 0 825 ms 0/ 35 (0%)
[112] 23 0-24 0 sec 9 77 KBytes 80 0 Kbits/sec 0 801 ms 0/ 50 (0%)
[112] 24 0 25 0 sec 6 05 KBytes 49 6 Kbits/sec 0 693 ms 0/ 31 (0%)

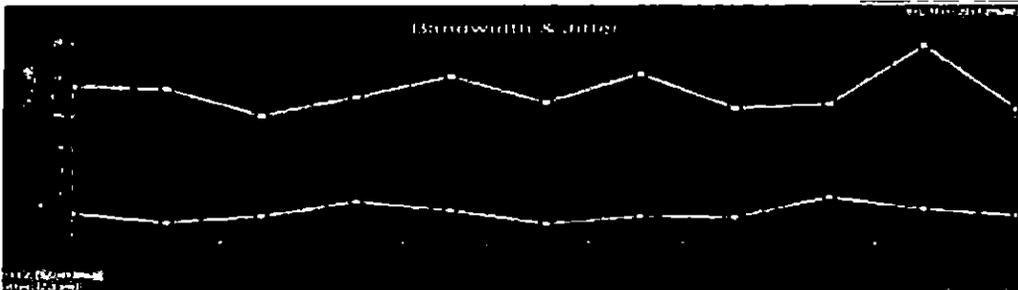
```

```

[112] 25 0-26 0 sec 9 57 KBytes 78 4 Kbits/sec 0 495 ms 0/ 49 (0%)
[112] 26 0-27 0 sec 6 64 KBytes 54 4 Kbits/sec 0 958 ms 0/ 34 (0%)
[112] 27 0-28 0 sec 6 25 KBytes 51 2 Kbits/sec 1 021 ms 0/ 32 (0%)
[112] 28 0-29 0 sec 9 77 KBytes 80 0 Kbits/sec 0 924 ms 0/ 50 (0%)
[112] 29 0 30 0 sec 6 84 KBytes 56 0 Kbits/sec 1 051 ms 0/ 35 (0%)
[112] 0 0-30 2 sec 219 KBytes 59 3 Kbits/sec 1 568 ms 0/ 1122 (0%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1091
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0 0 5 sec 1 76 KBytes 28 8 Kbits/sec 0 159 ms 1162/ 1171 (99%)

```

8 Datos regidos en el servidor de jperf con 30 flujos



```
bin/iperf.exe -s -u -P 0 -i 1 -p 1234 1 200 0B -f k
```

```

-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----

```

```

OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1412
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0- 1 0 sec 7 03 KBytes 57 6 Kbits/sec 0 420 ms 1852796269/ 36
(5 1e+009%)
[112] 1 0 2 0 sec 0 20 KBytes 1 60 Kbits/sec 0 549 ms 0/ 1 (0%)
[112] 2 0- 3 0 sec 0 00 KBytes 0 00 Kbits/sec 0 549 ms 0/ 0 ( 1 %)
[112] 3 0 4 0 sec 6 45 KBytes 52 8 Kbits/sec 0 464 ms 0/ 33 (0%)
[112] 4 0 5 0 sec 7 42 KBytes 60 8 Kbits/sec 0 547 ms 0/ 38 (0%)
[112] 5 0- 6 0 sec 7 81 KBytes 64 0 Kbits/sec 0 539 ms 0/ 40 (0%)
[112] 6 0 7 0 sec 7 81 KBytes 64 0 Kbits/sec 0 711 ms 0/ 40 (0%)
[112] 7 0- 8 0 sec 7 62 KBytes 62 4 Kbits/sec 0 379 ms 0/ 39 (0%)
[112] 8 0 9 0 sec 8 40 KBytes 68 8 Kbits/sec 1 215 ms 0/ 43 (0%)
[112] 9 0-10 0 sec 8 20 KBytes 67 2 Kbits/sec 0 511 ms 0/ 42 (0%)
[112] 10 0-11 0 sec 7 03 KBytes 57 6 Kbits/sec 0 430 ms 0/ 36 (0%)
[112] 11 0-12 0 sec 7 81 KBytes 64 0 Kbits/sec 0 846 ms 0/ 40 (0%)
[112] 12 0-13 0 sec 7 62 KBytes 62 4 Kbits/sec 0 503 ms 0/ 39 (0%)
[112] 13 0-14 0 sec 7 81 KBytes 64 0 Kbits/sec 1 453 ms 0/ 40 (0%)
[112] 14 0-15 0 sec 8 01 KBytes 65 6 Kbits/sec 0 973 ms 0/ 41 (0%)
[112] 15 0-16 0 sec 7 81 KBytes 64 0 Kbits/sec 1 296 ms 0/ 40 (0%)
[112] 16 0-17 0 sec 7 81 KBytes 64 0 Kbits/sec 0 843 ms 0/ 40 (0%)
[112] 17 0-18 0 sec 7 23 KBytes 59 2 Kbits/sec 0 776 ms 0/ 37 (0%)
[112] 18 0-19 0 sec 7 42 KBytes 60 8 Kbits/sec 0 706 ms 0/ 38 (0%)
[112] 19 0 20 0 sec 8 59 KBytes 70 4 Kbits/sec 0 597 ms 0/ 44 (0%)
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 20 0 21 0 sec 6 64 KBytes 54 4 Kbits/sec 1 593 ms 0/ 34 (0%)
[112] 21 0-22 0 sec 7 81 KBytes 64 0 Kbits/sec 0 974 ms 0/ 40 (0%)
[112] 22 0-23 0 sec 10 4 KBytes 84 8 Kbits/sec 1 088 ms 0/ 53 (0%)

```

```

[112] 23 0-24 0 sec 7 81 KBytes 64 0 Kbits/sec 0 595 ms 0/ 40 (0%)
[112] 24 0-25 0 sec 4 69 KBytes 38 4 Kbits/sec 0 871 ms 0/ 24 (0%)
[112] 25 0-26 0 sec 7 62 KBytes 62 4 Kbits/sec 0 470 ms 0/ 39 (0%)
[112] 26 0-27 0 sec 8 59 KBytes 70 4 Kbits/sec 0 884 ms 0/ 44 (0%)
[112] 27 0-28 0 sec 8 20 KBytes 67 2 Kbits/sec 1 975 ms 0/ 42 (0%)
[112] 28 0-29 0 sec 7 62 KBytes 62 4 Kbits/sec 0 611 ms 0/ 39 (0%)
[112] 29 0-30 0 sec 8 01 KBytes 65 6 Kbits/sec 0 518 ms 0/ 41 (0%)
[112] 0 0-30 2 sec 216 KBytes 58 5 Kbits/sec 1 076 ms 0/ 1105 (0%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1427
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 0 5 sec 3 52 KBytes 63 1 Kbits/sec 0 792 ms 998/ 1016 (98%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1439
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 0 7 sec 5 86 KBytes 64 5 Kbits/sec 1 266 ms 978/ 1008 (97%)

```

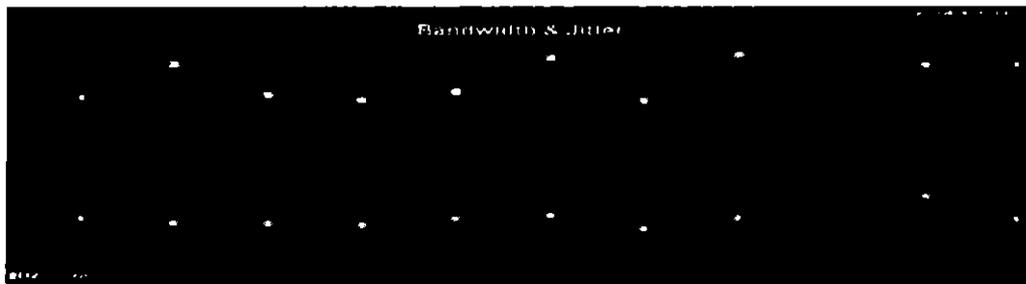
/Total Datagrams

```

[112] 20 0-21 0 sec 9 96 KBytes 81 6 Kbits/sec 0 581 ms 0/ 51 (0%)
[112] 21 0-22 0 sec 5 66 KBytes 46 4 Kbits/sec 0 991 ms 0/ 29 (0%)
[112] 22 0-23 0 sec 5 27 KBytes 43 2 Kbits/sec 1 523 ms 0/ 27 (0%)
[112] 23 0-24 0 sec 10 9 KBytes 89 6 Kbits/sec 0 635 ms 0/ 56 (0%)
[112] 24 0-25 0 sec 6 84 KBytes 56 0 Kbits/sec 0 909 ms 0/ 35 (0%)
[112] 25 0-26 0 sec 5 66 KBytes 46 4 Kbits/sec 1 498 ms 0/ 29 (0%)
[112] 26 0-27 0 sec 12 1 KBytes 99 2 Kbits/sec 0 868 ms 0/ 62 (0%)
[112] 27 0-28 0 sec 4 30 KBytes 35 2 Kbits/sec 0 552 ms 0/ 22 (0%)
[112] 28 0-29 0 sec 11 1 KBytes 91 2 Kbits/sec 0 983 ms 0/ 57 (0%)
[112] 29 0-30 0 sec 7 42 KBytes 60 8 Kbits/sec 0 397 ms 0/ 38 (0%)
[112] 0 0-30 3 sec 221 KBytes 59 8 Kbits/sec 1 545 ms 0/ 1132 (0%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1137
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 0 0 sec 0 20 KBytes 1 Kbits/sec 0 000 ms 1177/ 1178 (1e+002%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1169
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 0 5 sec 4 10 KBytes 71 7 Kbits/sec 0 320 ms 1181/ 1202 (98%)

```

9 Datos regidos en el servidor de jperf con 40 flujos



```
bin/jperf exe s -u -P 0 -i 1 -p 1234 -l 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

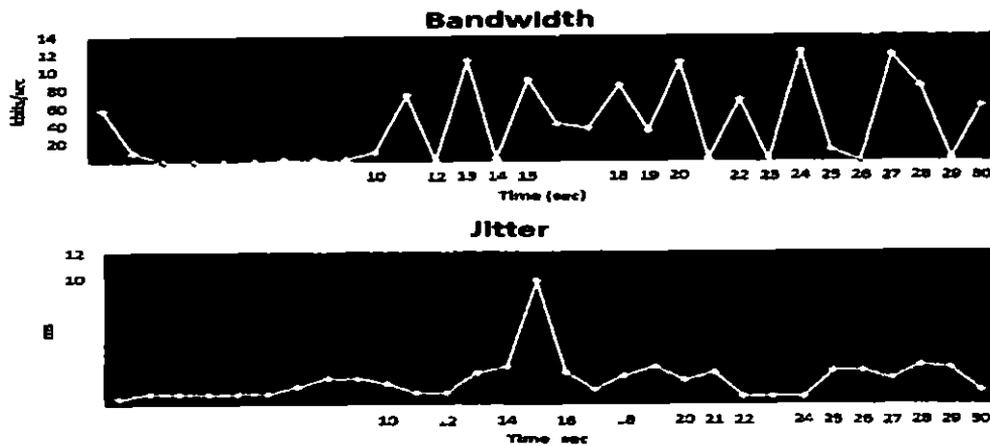
```
OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1212
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
```

```

[112] 0 0- 1 0 sec 7 42 KBytes 60 8 Kbits/sec 1 145 ms 1852796268/ 37
(5e+009%)
[112] 1 0 2 0 sec 0 20 KBytes 1 60 Kbits/sec 1 151 ms 0/ 1 (0%)
[112] 2 0- 3 0 sec 0 00 KBytes 0 00 Kbits/sec 1 151 ms 0/ 0 ( 1 $%)
[112] 3 0- 4 0 sec 0 00 KBytes 0 00 Kbits/sec 1 151 ms 0/ 0 (-1 $%)
[112] 4 0- 5 0 sec 2 54 KBytes 20 8 Kbits/sec 0 999 ms 0/ 13 (0%)
[112] 5 0- 6 0 sec 7 23 KBytes 59 2 Kbits/sec 0 854 ms 0/ 37 (0%)
[112] 6 0- 7 0 sec 8 59 KBytes 70 4 Kbits/sec 0 603 ms 0/ 44 (0%)
[112] 7 0- 8 0 sec 4 69 KBytes 38 4 Kbits/sec 0 540 ms 0/ 24 (0%)
[112] 8 0 9 0 sec 4 88 KBytes 40 0 Kbits/sec 0 933 ms 0/ 25 (0%)
[112] 9 0-10 0 sec 9 77 KBytes 80 0 Kbits/sec 0 809 ms 0/ 50 (0%)
[112] 10 0-11 0 sec 5 08 KBytes 41 6 Kbits/sec 0 780 ms 0/ 26 (0%)
[112] 11 0-12 0 sec 5 27 KBytes 43 2 Kbits/sec 0 822 ms 0/ 27 (0%)
[112] 12 0-13 0 sec 13 5 KBytes 110 Kbits/sec 1 784 ms 0/ 69 (0%)
[112] 13 0-14 0 sec 4 69 KBytes 38 4 Kbits/sec 1 161 ms 0/ 24 (0%)
[112] 14 0-15 0 sec 5 47 KBytes 44 8 Kbits/sec 1 000 ms 0/ 28 (0%)
[112] 15 0-16 0 sec 6 84 KBytes 56 0 Kbits/sec 0 486 ms 0/ 35 (0%)
[112] 16 0 17 0 sec 9 18 KBytes 75 2 Kbits/sec 7 842 ms 0/ 47 (0%)
[112] 17 0-18 0 sec 4 30 KBytes 35 2 Kbits/sec 2 242 ms 0/ 22 (0%)
[112] 18 0-19 0 sec 11 3 KBytes 92 8 Kbits/sec 2 809 ms 0/ 58 (0%)
[112] 19 0 20 0 sec 5 66 KBytes 46 4 Kbits/sec 1 459 ms 0/ 29 (0%)
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 20 0 21 0 sec 10 9 KBytes 89 6 Kbits/sec 1 006 ms 0/ 56 (0%)
[112] 21 0 22 0 sec 6 05 KBytes 49 6 Kbits/sec 0 960 ms 0/ 31 (0%)
[112] 22 0-23 0 sec 5 27 KBytes 43 2 Kbits/sec 0 844 ms 0/ 27 (0%)
[112] 23 0-24 0 sec 6 64 KBytes 54 4 Kbits/sec 1 373 ms 0/ 34 (0%)
[112] 24 0-25 0 sec 11 9 KBytes 97 6 Kbits/sec 1 702 ms 0/ 61 (0%)
[112] 25 0-26 0 sec 5 27 KBytes 43 2 Kbits/sec 0 491 ms 0/ 27 (0%)
[112] 26 0 27 0 sec 12 5 KBytes 102 Kbits/sec 1 502 ms 0/ 64 (0%)
[112] 27 0 28 0 sec 0 00 KBytes 0 00 Kbits/sec 1 502 ms 0/ 0 (-1 $%)
[112] 28 0-29 0 sec 10 7 KBytes 88 0 Kbits/sec 3 488 ms 9/ 64 (14%)
[112] 29 0-30 0 sec 10 7 KBytes 88 0 Kbits/sec 1 421 ms 0/ 55 (0%)
[112] 0 0-30 8 sec 197 KBytes 52 3 Kbits/sec 1 619 ms 8/ 1017 (0 79%)
[112] 0 0-30 8 sec 1 datagrams received out-of-order
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1241
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0- 1 0 sec 8 59 KBytes 70 4 Kbits/sec 0 813 ms 1066/ 1104 (97%)
[112] 0 0- 1 0 sec 42 datagrams received out of order
[112] 0 0- 1 0 sec 8 98 KBytes 73 2 Kbits/sec 1 012 ms 1063/ 1106 (96%)
[112] 0 0 1 0 sec 45 datagrams received out-of-order
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1242
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0- 0 2 sec 1 56 KBytes 81 3 Kbits/sec 1 214 ms 1153/ 1161 (99%)

```

10 Datos regidos en el servidor de jperf con 60 flujos



```
bin/iperf.exe -s -u -P 0 -i 1 -p 1234 1 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

OpenSCManager failed - Acceso denegado (0x5)

```
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 2386
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec    7 03 KBytes   57 6 Kbits/sec  0 504 ms    1348221506/ 35
(3 9e+009%)
[112] 1 0- 2 0 sec    1 17 KBytes   9 60 Kbits/sec  0 865 ms    0/ 6 (0%)
[112] 2 0- 3 0 sec    0 00 KBytes   0 00 Kbits/sec  0 865 ms    0/ 0 ( 1 $%)
[112] 3 0- 4 0 sec    0 00 KBytes   0 00 Kbits/sec  0 865 ms    0/ 0 (-1 $%)
[112] 4 0- 5 0 sec    0 00 KBytes   0 00 Kbits/sec  0 865 ms    0/ 0 ( 1 $%)
[112] 5 0- 6 0 sec    0 00 KBytes   0 00 Kbits/sec  0 865 ms    0/ 0 (-1 $%)
[112] 6 0- 7 0 sec    0 20 KBytes   1 60 Kbits/sec  1 363 ms    0/ 1 (0%)
[112] 7 0- 8 0 sec    0 20 KBytes   1 60 Kbits/sec  2 097 ms    0/ 1 (0%)
[112] 8 0- 9 0 sec    0 20 KBytes   1 60 Kbits/sec  2 005 ms    0/ 1 (0%)
[112] 9 0-10 0 sec   1 17 KBytes   9 60 Kbits/sec  1 639 ms    0/ 6 (0%)
[112] 10 0-11 0 sec   8 98 KBytes   73 6 Kbits/sec  0 948 ms    0/ 46 (0%)
[112] 11 0-12 0 sec   0 00 KBytes   0 00 Kbits/sec  0 948 ms    0/ 0 ( 1 $%)
[112] 12 0-13 0 sec  13 7 KBytes   112 Kbits/sec   2 491 ms    0/ 70 (0%)
[112] 13 0-14 0 sec   0 20 KBytes   1 60 Kbits/sec  2 962 ms    0/ 1 (0%)
[112] 14 0 15 0 sec  11 1 KBytes   91 2 Kbits/sec   9 700 ms    0/ 57 (0%)
[112] 15 0-16 0 sec   5 08 KBytes   41 6 Kbits/sec   2 511 ms    0/ 26 (0%)
[112] 16 0-17 0 sec   4 30 KBytes   35 2 Kbits/sec   1 196 ms    0/ 22 (0%)
[112] 17 0-18 0 sec  10 4 KBytes   84 8 Kbits/sec   2 264 ms    0/ 53 (0%)
[112] 18 0-19 0 sec   4 10 KBytes   33 6 Kbits/sec   2 953 ms    0/ 21 (0%)
[112] 19 0-20 0 sec  13 5 KBytes   110 Kbits/sec   1 859 ms    0/ 69 (0%)
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 20 0-21 0 sec   0 20 KBytes   1 60 Kbits/sec   2 447 ms    0/ 1 (0%)
[112] 21 0-22 0 sec   8 40 KBytes   68 8 Kbits/sec   0 608 ms    0/ 43 (0%)
[112] 22 0-23 0 sec   0 00 KBytes   0 00 Kbits/sec   0 608 ms    0/ 0 (-1 $%)
[112] 23 0-24 0 sec  14 8 KBytes   122 Kbits/sec   0 513 ms    0/ 76 (0%)
[112] 24 0-25 0 sec   1 37 KBytes   11 2 Kbits/sec   2 649 ms    1/ 8 (13%)
[112] 25 0-26 0 sec   0 00 KBytes   0 00 Kbits/sec   2 649 ms    0/ 0 ( 1 $%)
```

```

[112] 26 0-27 0 sec 14 5 KBytes 118 Kbits/sec 1 976 ms 0/ 74 (0%)
[112] 27 0-28 0 sec 10 2 KBytes 83 2 Kbits/sec 2 989 ms 0/ 52 (0%)
[112] 28 0-29 0 sec 0 20 KBytes 1 60 Kbits/sec 2 882 ms 0/ 1 (0%)
[112] 29 0-30 0 sec 7 42 KBytes 60 8 Kbits/sec 1 118 ms 0/ 38 (0%)
[112] 30 0-31 0 sec 0 00 KBytes 0 00 Kbits/sec 1 118 ms 0/ 0 (-1 %)
[112] 0 0-31 2 sec 139 KBytes 36 4 Kbits/sec 2 527 ms 0/ 710 (0%)
[112] 0 0-31 2 sec 1 datagrams received out-of-order
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 2422
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0- 1 0 sec 6 05 KBytes 49 6 Kbits/sec 0 303 ms 875/ 905 (97%)
[112] 1 0- 2 0 sec 7 62 KBytes 62 4 Kbits/sec 1 042 ms 0/ 39 (0%)
[112] 0 0- 2 1 sec 14 1 KBytes 54 1 Kbits/sec 1 229 ms 874/ 946 (92%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 2434
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0- 0 4 sec 2 54 KBytes 55 5 Kbits/sec 0 254 ms 1044/ 968
(1 1e+002%)
[112] 0 0 0 4 sec 12 datagrams received out of order

```

Archivo de Configuración de para VoIP con QoS

```

root@WMR-079-04 ~# vi /etc/config/qos
# QoS configuration for OpenWrt

```

INTERFACES

```

config interface wan
    option classgroup Default
    option enabled 1
    option upload 128
    option download 1024

```

RULES

```

config classify
    option srchost 192 168 18 78 192 168 18 79
    option dsthost 192 168 18 78 192 168 18 79
    option proto udp
    option target Priority

```

```

config classify
    option target Express
    option ports 22 53

```

```

config classify
    option target Normal
    option proto tcp
    /etc/config/qos 1/92 1%

```

```

config reclassify

```

```

    option target    Priority
    option proto     icmp

config default
    option target    Bulk
    option portrange 1024-65535

config reclassify
    option target    Priority
    option proto     tcp
    option pktsize   128
    option mark      'Bulk
    option tcpflags  SYN

config reclassify
    option target    Priority
    option proto     tcp
    option pktsize   128
    option mark      'Bulk
    option tcpflags  ACK

# Don't change the stuff below unless you
# really know what it means )

config classgroup Default
    option classes   Priority Express Normal Bulk
    option default   "Normal

config class Priority
    option packetsize 400
    option maxsize    400
    option avgrate    10
    option priority    20
config class Priority_down
    option packetsize 1000
    option avgrate    10

config class Express
    option packetsize 1000
    option maxsize    800
    option avgrate    50
    option priority    10

```

```

config class "Normal
  option packetsize 1500
  option packetdelay 100
  option avgrate 10
  option priority 5
config class "Normal_down
  option avgrate 20

```

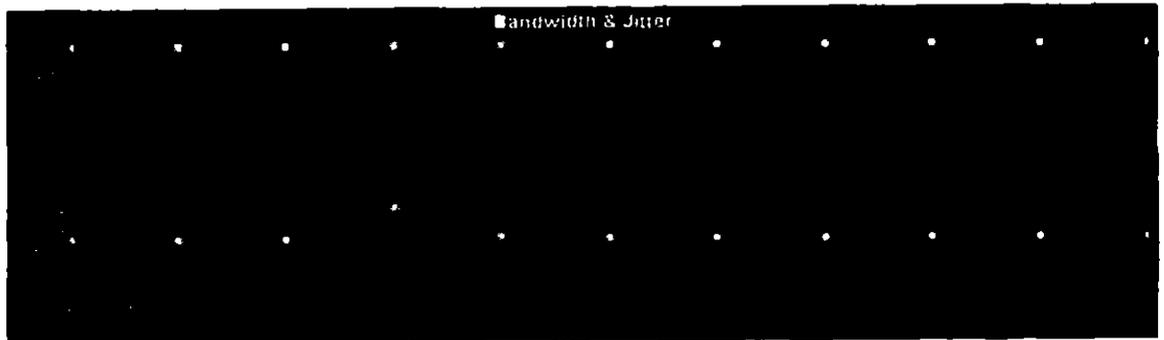
```

config class Bulk
  option avgrate 1
  option packetdelay 200

```

Pruebas de VoIP con QoS

1 Datos regidos en el servidor de jperf con 1 flujo



```
bin/iperf exe -s -u -P 0 -i 1 -p 1234 -l 200 0B -f k
```

```

-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----

```

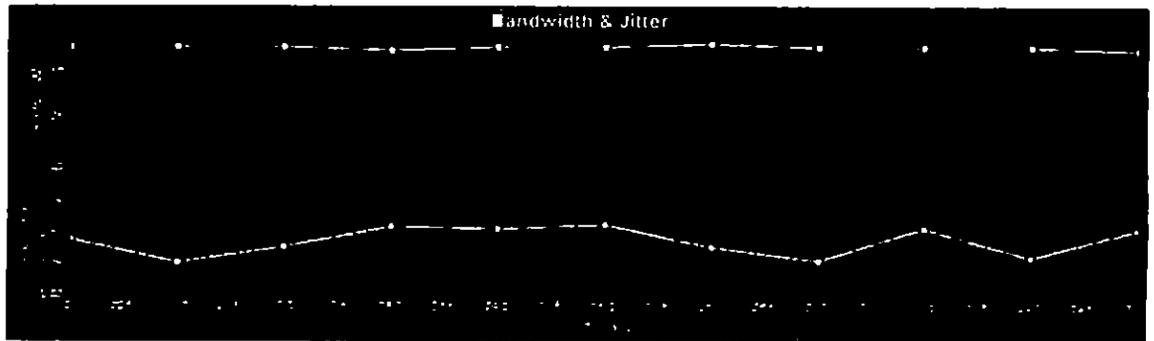
```

OpenSCManager failed Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1256
[ ID] Interval      Transfer      Bandwidth      Jitter    Lost/Total Datagrams
[112] 0 0- 1 0 sec    7 81 KBytes   64 0 Kbits/sec  0 536 ms  1852796282/ 53
(3 5e+009%)
[112] 1 0- 2 0 sec    7 81 KBytes   64 0 Kbits/sec  0 302 ms   0/ 40 (0%)
[112] 2 0- 3 0 sec    7 81 KBytes   64 0 Kbits/sec  0 262 ms   0/ 40 (0%)
[112] 3 0- 4 0 sec    7 81 KBytes   64 0 Kbits/sec  0 259 ms   0/ 40 (0%)
[112] 4 0- 5 0 sec    7 81 KBytes   64 0 Kbits/sec  0 259 ms   0/ 40 (0%)
[112] 5 0- 6 0 sec    7 81 KBytes   64 0 Kbits/sec  0 259 ms   0/ 40 (0%)
[112] 6 0- 7 0 sec    7 81 KBytes   64 0 Kbits/sec  0 259 ms   0/ 40 (0%)
[112] 7 0- 8 0 sec    7 81 KBytes   64 0 Kbits/sec  0 259 ms   0/ 40 (0%)
[112] 8 0- 9 0 sec    7 81 KBytes   64 0 Kbits/sec  0 319 ms   0/ 40 (0%)
[112] 9 0-10 0 sec    7 81 KBytes   64 0 Kbits/sec  0 263 ms   0/ 40 (0%)

```

[112]	10	0	11	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)
[112]	11	0-12	0	sec	7	81	KBytes	64	0	Kbits/sec	0	302	ms	0/	40	(0%)	
[112]	12	0-13	0	sec	7	81	KBytes	64	0	Kbits/sec	0	262	ms	0/	40	(0%)	
[112]	13	0	14	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)
[112]	14	0-15	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[112]	15	0-16	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[112]	16	0-17	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[112]	17	0-18	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[112]	18	0	19	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)
[112]	19	0-20	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[ID]	Interval					Transfer			Bandwidth		Jitter		Lost/Total		Datagrams		
[112]	20	0-21	0	sec	7	81	KBytes	64	0	Kbits/sec	0	258	ms	0/	40	(0%)	
[112]	21	0-22	0	sec	7	81	KBytes	64	0	Kbits/sec	0	405	ms	0/	40	(0%)	
[112]	22	0-23	0	sec	7	81	KBytes	64	0	Kbits/sec	0	270	ms	0/	40	(0%)	
[112]	23	0-24	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[112]	24	0-25	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[112]	25	0	26	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)
[112]	26	0-27	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[112]	27	0	28	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)
[112]	28	0-29	0	sec	7	81	KBytes	64	0	Kbits/sec	0	259	ms	0/	40	(0%)	
[112]	0	0-29	7	sec	232	KBytes			64	0	Kbits/sec	0	234	ms	13/	1202	(1 1%)

2 Datos regidos en el servidor de jperf con 3 flujos



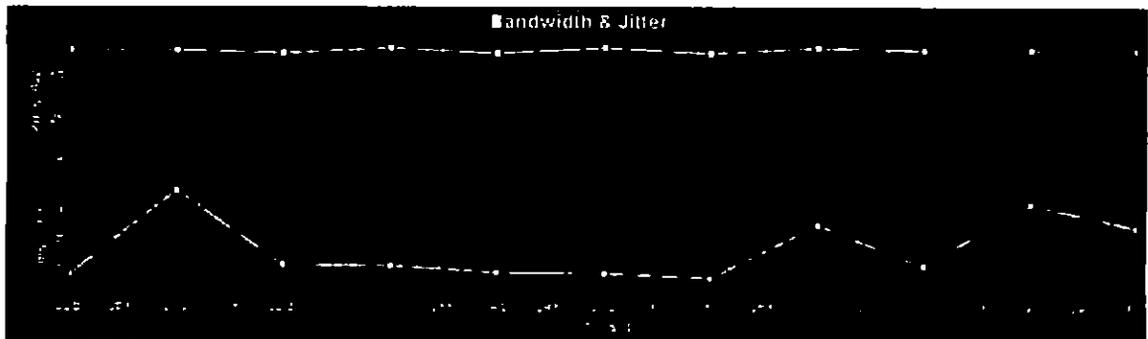
```
bin/ perf exe -s -u P 0 i 1 p 1234 -l 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

```
OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1259
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec    7 81 KBytes   64 0 Kbits/sec  0 713 ms    1852796269/ 40
(4 6e+009%)
[112] 1 0- 2 0 sec    7 42 KBytes   60 8 Kbits/sec  0 852 ms    0/ 38 (0%)
[112] 2 0- 3 0 sec    7 81 KBytes   64 0 Kbits/sec  0 502 ms    0/ 40 (0%)
[112] 3 0- 4 0 sec    7 81 KBytes   64 0 Kbits/sec  0 939 ms    0/ 40 (0%)
[112] 4 0 5 0 sec    8 01 KBytes   65 6 Kbits/sec  0 617 ms    0/ 41 (0%)
[112] 5 0- 6 0 sec    8 01 KBytes   65 6 Kbits/sec  0 750 ms    0/ 41 (0%)
[112] 6 0- 7 0 sec    7 62 KBytes   62 4 Kbits/sec  0 956 ms    0/ 39 (0%)
[112] 7 0 8 0 sec    7 62 KBytes   62 4 Kbits/sec  0 648 ms    0/ 39 (0%)
[112] 8 0- 9 0 sec    8 01 KBytes   65 6 Kbits/sec  0 262 ms    0/ 41 (0%)
```

[112]	9	0-10	0 sec	7 81 KBytes	64 0 Kbits/sec	0 381 ms	0/	40 (0%)
[112]	10	0-11	0 sec	7 81 KBytes	64 0 Kbits/sec	0 275 ms	0/	40 (0%)
[112]	11	0-12	0 sec	7 81 KBytes	64 0 Kbits/sec	0 670 ms	0/	40 (0%)
[112]	12	0-13	0 sec	7 62 KBytes	62 4 Kbits/sec	0 889 ms	0/	39 (0%)
[112]	13	0-14	0 sec	8 01 KBytes	65 6 Kbits/sec	0 403 ms	0/	41 (0%)
[112]	14	0-15	0 sec	7 81 KBytes	64 0 Kbits/sec	0 509 ms	0/	40 (0%)
[112]	15	0-16	0 sec	7 62 KBytes	62 4 Kbits/sec	0 663 ms	0/	39 (0%)
[112]	16	0-17	0 sec	8 01 KBytes	65 6 Kbits/sec	0 278 ms	0/	41 (0%)
[112]	17	0-18	0 sec	7 81 KBytes	64 0 Kbits/sec	0 492 ms	0/	40 (0%)
[112]	18	0-19	0 sec	7 81 KBytes	64 0 Kbits/sec	0 284 ms	0/	40 (0%)
[112]	19	0-20	0 sec	7 81 KBytes	64 0 Kbits/sec	0 452 ms	0/	40 (0%)
[ID]	Interval	Transfer	Bandwidth	Jitter	Lost/Total	Datagrams		
[112]	20 0-21 0 sec	7 81 KBytes	64 0 Kbits/sec	0 259 ms	0/	40 (0%)		
[112]	21 0-22 0 sec	7 81 KBytes	64 0 Kbits/sec	0 390 ms	0/	40 (0%)		
[112]	22 0-23 0 sec	7 62 KBytes	62 4 Kbits/sec	0 552 ms	0/	39 (0%)		
[112]	23 0-24 0 sec	7 81 KBytes	64 0 Kbits/sec	0 537 ms	0/	40 (0%)		
[112]	24 0-25 0 sec	7 81 KBytes	64 0 Kbits/sec	0 573 ms	0/	40 (0%)		
[112]	25 0-26 0 sec	8 01 KBytes	65 6 Kbits/sec	0 388 ms	0/	41 (0%)		
[112]	26 0-27 0 sec	7 81 KBytes	64 0 Kbits/sec	0 281 ms	0/	40 (0%)		
[112]	27 0-28 0 sec	7 81 KBytes	64 0 Kbits/sec	0 540 ms	0/	40 (0%)		
[112]	28 0 29 0 sec	7 81 KBytes	64 0 Kbits/sec	0 305 ms	0/	40 (0%)		
[112]	29 0 30 0 sec	7 62 KBytes	62 4 Kbits/sec	0 536 ms	0/	39 (0%)		
[112]	0 0-30 0 sec	234 KBytes	64 0 Kbits/sec	0 507 ms	0/	1200 (0%)		

3 Datos rigidos en el servidor de jperf con 5 flujos



```
bin/jperf exe -s -u P 0 -i 1 -p 1234 -l 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

```
OpenSCManager failed - Acceso denegado (0x5)
```

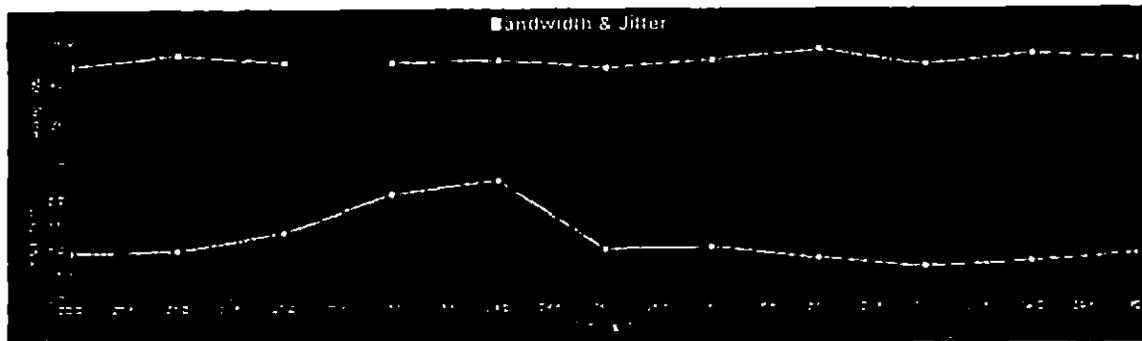
[112]	local	192 168 18 78	port 1234	connected with	192 168 18 79	port 1276
{ ID]	Interval	Transfer	Bandwidth	Jitter	Lost/Total	Datagrams
[112]	0 0- 1 0 sec	7 81 KBytes	64 0 Kbits/sec	0 915 ms	1852796269/	40 (4 6e+009%)
[112]	1 0 2 0 sec	7 62 KBytes	62 4 Kbits/sec	1 017 ms	0/	39 (0%)
[112]	2 0- 3 0 sec	8 01 KBytes	65 6 Kbits/sec	0 783 ms	0/	41 (0%)
[112]	3 0- 4 0 sec	7 42 KBytes	60 8 Kbits/sec	4 140 ms	0/	38 (0%)
[112]	4 0- 5 0 sec	7 81 KBytes	64 0 Kbits/sec	0 963 ms	0/	40 (0%)
[112]	5 0 6 0 sec	8 20 KBytes	67 2 Kbits/sec	0 675 ms	0/	42 (0%)
[112]	6 0 7 0 sec	7 81 KBytes	64 0 Kbits/sec	0 617 ms	0/	40 (0%)

```

[112] 7 0 8 0 sec 7 42 KBytes 60 8 Kbits/sec 0 952 ms 0/ 38 (0%)
[112] 8 0 9 0 sec 7 81 KBytes 64 0 Kbits/sec 0 528 ms 0/ 40 (0%)
[112] 9 0 10 0 sec 7 81 KBytes 64 0 Kbits/sec 0 714 ms 0/ 40 (0%)
[112] 10 0-11 0 sec 8 01 KBytes 65 6 Kbits/sec 0 511 ms 0/ 41 (0%)
[112] 11 0-12 0 sec 8 01 KBytes 65 6 Kbits/sec 0 881 ms 0/ 41 (0%)
[112] 12 0-13 0 sec 7 81 KBytes 64 0 Kbits/sec 0 594 ms 0/ 40 (0%)
[112] 13 0 14 0 sec 7 42 KBytes 60 8 Kbits/sec 0 605 ms 0/ 38 (0%)
[112] 14 0-15 0 sec 8 20 KBytes 67 2 Kbits/sec 0 741 ms 0/ 42 (0%)
[112] 15 0-16 0 sec 7 42 KBytes 60 8 Kbits/sec 0 738 ms 0/ 38 (0%)
[112] 16 0 17 0 sec 7 81 KBytes 64 0 Kbits/sec 0 578 ms 0/ 40 (0%)
[112] 17 0-18 0 sec 8 01 KBytes 65 6 Kbits/sec 0 802 ms 0/ 41 (0%)
[112] 18 0-19 0 sec 7 81 KBytes 64 0 Kbits/sec 0 623 ms 0/ 40 (0%)
[112] 19 0-20 0 sec 7 81 KBytes 64 0 Kbits/sec 0 735 ms 0/ 40 (0%)
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 20 0-21 0 sec 7 81 KBytes 64 0 Kbits/sec 3 752 ms 0/ 40 (0%)
[112] 21 0-22 0 sec 7 62 KBytes 62 4 Kbits/sec 1 090 ms 0/ 39 (0%)
[112] 22 0-23 0 sec 8 01 KBytes 65 6 Kbits/sec 1 063 ms 0/ 41 (0%)
[112] 23 0-24 0 sec 7 62 KBytes 62 4 Kbits/sec 0 839 ms 0/ 39 (0%)
[112] 24 0-25 0 sec 8 01 KBytes 65 6 Kbits/sec 0 781 ms 0/ 41 (0%)
[112] 25 0-26 0 sec 7 62 KBytes 62 4 Kbits/sec 0 622 ms 0/ 39 (0%)
[112] 26 0-27 0 sec 8 01 KBytes 65 6 Kbits/sec 2 516 ms 0/ 41 (0%)
[112] 27 0-28 0 sec 7 81 KBytes 64 0 Kbits/sec 1 060 ms 0/ 40 (0%)
[112] 28 0-29 0 sec 7 81 KBytes 64 0 Kbits/sec 3 259 ms 0/ 40 (0%)
[112] 29 0-30 0 sec 7 81 KBytes 64 0 Kbits/sec 2 453 ms 0/ 40 (0%)
[112] 0 0-30 1 sec 235 KBytes 63 9 Kbits/sec 2 740 ms 0/ 1201 (0%)

```

4 Datos regidos en el servidor de jperf con 10 flujos



```
bin/jperf exe -s -u P 0 -i 1 -p 1234 1 200 0B -f k
```

```

-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----

```

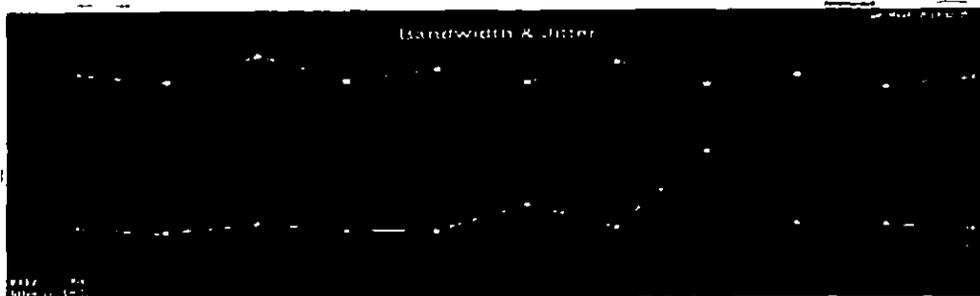
```

OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1274
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0- 1 0 sec 7 42 KBytes 60 8 Kbits/sec 1 121 ms 1852796269/ 38
[4 9e+009%)
[112] 1 0- 2 0 sec 8 01 KBytes 65 6 Kbits/sec 1 174 ms 0/ 41 (0%)
[112] 2 0- 3 0 sec 8 01 KBytes 65 6 Kbits/sec 1 241 ms 0/ 41 (0%)
[112] 3 0 4 0 sec 7 03 KBytes 57 6 Kbits/sec 0 884 ms 0/ 36 (0%)
[112] 4 0- 5 0 sec 7 81 KBytes 64 0 Kbits/sec 1 182 ms 0/ 40 (0%)
[112] 5 0 6 0 sec 8 01 KBytes 65 6 Kbits/sec 0 782 ms 0/ 41 (0%)

```

[ID]	Interval	Transfer	Bandwidth	Jitter	Lost/Total	Datagrams
[112]	6 0 7 0 sec	8 01 KBytes	65 6 Kbits/sec	0 897 ms	0/	41 (0%)
[112]	7 0- 8 0 sec	8 20 KBytes	67 2 Kbits/sec	1 665 ms	0/	42 (0%)
[112]	8 0- 9 0 sec	6 64 KBytes	54 4 Kbits/sec	1 796 ms	0/	34 (0%)
[112]	9 0-10 0 sec	8 79 KBytes	72 0 Kbits/sec	1 132 ms	0/	45 (0%)
[112]	10 0-11 0 sec	7 81 KBytes	64 0 Kbits/sec	1 188 ms	0/	40 (0%)
[112]	11 0-12 0 sec	6 45 KBytes	52 8 Kbits/sec	1 181 ms	0/	33 (0%)
[112]	12 0 13 0 sec	8 59 KBytes	70 4 Kbits/sec	1 079 ms	0/	44 (0%)
[112]	13 0-14 0 sec	7 81 KBytes	64 0 Kbits/sec	0 735 ms	0/	40 (0%)
[112]	14 0-15 0 sec	7 81 KBytes	64 0 Kbits/sec	1 425 ms	0/	40 (0%)
[112]	15 0 16 0 sec	7 42 KBytes	60 8 Kbits/sec	1 119 ms	0/	38 (0%)
[112]	16 0 17 0 sec	7 62 KBytes	62 4 Kbits/sec	1 005 ms	0/	39 (0%)
[112]	17 0-18 0 sec	8 01 KBytes	65 6 Kbits/sec	0 612 ms	0/	41 (0%)
[112]	18 0-19 0 sec	8 20 KBytes	67 2 Kbits/sec	0 901 ms	0/	42 (0%)
[112]	19 0 20 0 sec	7 42 KBytes	60 8 Kbits/sec	0 903 ms	0/	38 (0%)
[112]	20 0 21 0 sec	8 20 KBytes	67 2 Kbits/sec	0 928 ms	0/	42 (0%)
[112]	21 0-22 0 sec	7 62 KBytes	62 4 Kbits/sec	1 296 ms	0/	39 (0%)
[112]	22 0-23 0 sec	7 62 KBytes	62 4 Kbits/sec	2 076 ms	0/	39 (0%)
[112]	23 0-24 0 sec	7 81 KBytes	64 0 Kbits/sec	2 359 ms	0/	40 (0%)
[112]	24 0-25 0 sec	7 23 KBytes	59 2 Kbits/sec	0 964 ms	0/	37 (0%)
[112]	25 0-26 0 sec	7 81 KBytes	64 0 Kbits/sec	1 009 ms	0/	40 (0%)
[112]	26 0-27 0 sec	8 59 KBytes	70 4 Kbits/sec	0 777 ms	0/	44 (0%)
[112]	27 0-28 0 sec	7 42 KBytes	60 8 Kbits/sec	0 604 ms	0/	38 (0%)
[112]	28 0-29 0 sec	8 20 KBytes	67 2 Kbits/sec	0 719 ms	0/	42 (0%)
[112]	29 0-30 0 sec	7 81 KBytes	64 0 Kbits/sec	0 863 ms	0/	40 (0%)
[112]	0 0 30 2 sec	234 KBytes	63 5 Kbits/sec	0 795 ms	0/	1197 (0%)

5 Datos regidos en el servidor de jperf con 15 flujos



```
bin/jperf exe -s -u -P 0 -i 1 p 1234 -l 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

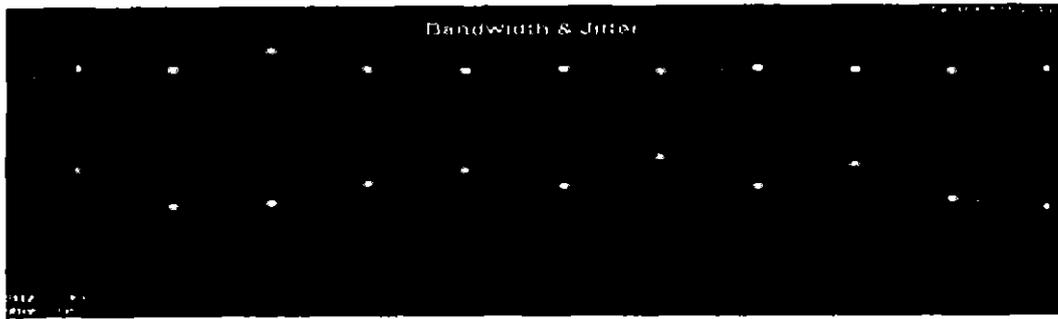
```
OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1323
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec    7 03 KBytes   57 6 Kbits/sec  1 439 ms    1852796269/ 36
(5 1e+009%)
[112] 1 0- 2 0 sec    4 69 KBytes   38 4 Kbits/sec  1 031 ms     0/ 24 (0%)
[112] 2 0- 3 0 sec    8 20 KBytes   67 2 Kbits/sec  0 874 ms     0/ 42 (0%)
[112] 3 0 4 0 sec    8 20 KBytes   67 2 Kbits/sec  1 143 ms     0/ 42 (0%)
[112] 4 0 5 0 sec    6 45 KBytes   52 8 Kbits/sec  4 070 ms     0/ 33 (0%)
[112] 5 0 6 0 sec    8 40 KBytes   68 8 Kbits/sec  2 018 ms     0/ 43 (0%)
[112] 6 0 7 0 sec    7 23 KBytes   59 2 Kbits/sec  0 848 ms     0/ 37 (0%)
```

```

[112] 7 0 8 0 sec 7 03 KBytes 57 6 Kbits/sec 1 131 ms 0/ 36 (0%)
[112] 8 0- 9 0 sec 6 84 KBytes 56 0 Kbits/sec 0 636 ms 0/ 35 (0%)
[112] 9 0-10 0 sec 9 96 KBytes 81 6 Kbits/sec 2 418 ms 0/ 51 (0%)
[112] 10 0-11 0 sec 7 42 KBytes 60 8 Kbits/sec 1 817 ms 0/ 38 (0%)
[112] 11 0-12 0 sec 7 42 KBytes 60 8 Kbits/sec 1 063 ms 0/ 38 (0%)
[112] 12 0-13 0 sec 8 79 KBytes 72 0 Kbits/sec 1 116 ms 0/ 45 (0%)
[112] 13 0-14 0 sec 6 64 KBytes 54 4 Kbits/sec 1 250 ms 0/ 34 (0%)
[112] 14 0-15 0 sec 6 64 KBytes 54 4 Kbits/sec 1 449 ms 0/ 34 (0%)
[112] 15 0-16 0 sec 7 81 KBytes 64 0 Kbits/sec 0 746 ms 0/ 40 (0%)
[112] 16 0-17 0 sec 7 23 KBytes 59 2 Kbits/sec 1 292 ms 0/ 37 (0%)
[112] 17 0-18 0 sec 8 01 KBytes 65 6 Kbits/sec 0 854 ms 0/ 41 (0%)
[112] 18 0-19 0 sec 8 79 KBytes 72 0 Kbits/sec 0 732 ms 0/ 45 (0%)
[112] 19 0-20 0 sec 7 42 KBytes 60 8 Kbits/sec 1 049 ms 0/ 38 (0%)
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 20 0-21 0 sec 6 45 KBytes 52 8 Kbits/sec 0 596 ms 0/ 33 (0%)
[112] 21 0-22 0 sec 9 57 KBytes 78 4 Kbits/sec 1 474 ms 0/ 49 (0%)
[112] 22 0-23 0 sec 6 64 KBytes 54 4 Kbits/sec 0 814 ms 0/ 34 (0%)
[112] 23 0-24 0 sec 8 01 KBytes 65 6 Kbits/sec 0 728 ms 0/ 41 (0%)
[112] 24 0-25 0 sec 6 45 KBytes 52 8 Kbits/sec 3 259 ms 0/ 33 (0%)
[112] 25 0-26 0 sec 8 79 KBytes 72 0 Kbits/sec 1 045 ms 0/ 45 (0%)
[112] 26 0-27 0 sec 6 25 KBytes 51 2 Kbits/sec 8 132 ms 0/ 32 (0%)
[112] 27 0-28 0 sec 7 23 KBytes 59 2 Kbits/sec 1 446 ms 0/ 37 (0%)
[112] 28 0-29 0 sec 5 86 KBytes 48 0 Kbits/sec 1 286 ms 0/ 30 (0%)
[112] 29 0-30 0 sec 6 84 KBytes 56 0 Kbits/sec 0 816 ms 0/ 35 (0%)
[112] 0 0-30 0 sec 223 KBytes 60 8 Kbits/sec 1 608 ms 0/ 1140 (0%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1319
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 0 4 sec 2 93 KBytes 67 6 Kbits/sec 1 982 ms 1133/ 1148 (99%)

```

6 Datos regidos en el servidor de iperf con 20 flujos



```
bin/iperf.exe -s -u -P 0 -i 1 -p 1234 -l 200 0B -f k
```

```

-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----

```

```

OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1360
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0 1 0 sec 7 03 KBytes 57 6 Kbits/sec 1 019 ms 1852796269/ 36
(5 1e+009%)
[112] 1 0- 2 0 sec 4 88 KBytes 40 0 Kbits/sec 1 052 ms 0/ 25 (0%)
[112] 2 0 3 0 sec 7 42 KBytes 60 8 Kbits/sec 0 590 ms 0/ 38 (0%)
[112] 3 0 4 0 sec 7 62 KBytes 62 4 Kbits/sec 0 885 ms 0/ 39 (0%)

```

```

[112] 4 0- 5 0 sec 7 81 KBytes 64 0 Kbits/sec 1 316 ms 0/ 40 (0%)
[112] 5 0- 6 0 sec 7 42 KBytes 60 8 Kbits/sec 0 928 ms 0/ 38 (0%)
[112] 6 0 7 0 sec 7 42 KBytes 60 8 Kbits/sec 0 593 ms 0/ 38 (0%)
[112] 7 0- 8 0 sec 7 42 KBytes 60 8 Kbits/sec 0 497 ms 0/ 38 (0%)
[112] 8 0- 9 0 sec 8 01 KBytes 65 6 Kbits/sec 1 066 ms 0/ 41 (0%)
[112] 9 0-10 0 sec 9 96 KBytes 81 6 Kbits/sec 0 742 ms 0/ 51 (0%)
[112] 10 0-11 0 sec 7 42 KBytes 60 8 Kbits/sec 0 771 ms 0/ 38 (0%)
[112] 11 0-12 0 sec 8 01 KBytes 65 6 Kbits/sec 0 829 ms 0/ 41 (0%)
[112] 12 0-13 0 sec 8 01 KBytes 65 6 Kbits/sec 1 221 ms 0/ 41 (0%)
[112] 13 0-14 0 sec 7 62 KBytes 62 4 Kbits/sec 0 592 ms 0/ 39 (0%)
[112] 14 0-15 0 sec 7 62 KBytes 62 4 Kbits/sec 0 757 ms 0/ 39 (0%)
[112] 15 0-16 0 sec 7 42 KBytes 60 8 Kbits/sec 0 973 ms 0/ 38 (0%)
[112] 16 0-17 0 sec 7 23 KBytes 59 2 Kbits/sec 0 891 ms 0/ 37 (0%)
[112] 17 0-18 0 sec 7 62 KBytes 62 4 Kbits/sec 0 827 ms 0/ 39 (0%)
[112] 18 0-19 0 sec 7 23 KBytes 59 2 Kbits/sec 0 928 ms 0/ 37 (0%)
[112] 19 0-20 0 sec 8 01 KBytes 65 6 Kbits/sec 1 202 ms 0/ 41 (0%)
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 20 0 21 0 sec 7 81 KBytes 64 0 Kbits/sec 0 682 ms 0/ 40 (0%)
[112] 21 0-22 0 sec 9 77 KBytes 80 0 Kbits/sec 0 718 ms 0/ 50 (0%)
[112] 22 0-23 0 sec 7 81 KBytes 64 0 Kbits/sec 1 000 ms 0/ 40 (0%)
[112] 23 0-24 0 sec 7 62 KBytes 62 4 Kbits/sec 1 200 ms 0/ 39 (0%)
[112] 24 0 25 0 sec 7 81 KBytes 64 0 Kbits/sec 0 952 ms 0/ 40 (0%)
[112] 25 0-26 0 sec 7 42 KBytes 60 8 Kbits/sec 1 383 ms 0/ 38 (0%)
[112] 26 0-27 0 sec 7 81 KBytes 64 0 Kbits/sec 0 948 ms 0/ 40 (0%)
[112] 27 0 28 0 sec 7 62 KBytes 62 4 Kbits/sec 1 273 ms 0/ 39 (0%)
[112] 28 0-29 0 sec 7 42 KBytes 60 8 Kbits/sec 0 765 ms 0/ 38 (0%)
[112] 29 0 30 0 sec 7 62 KBytes 62 4 Kbits/sec 0 649 ms 0/ 39 (0%)
[112] 0 0-30 1 sec 230 KBytes 62 6 Kbits/sec 0 827 ms 0/ 1179 (0%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1354
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 0 1 sec 0 78 KBytes 68 3 Kbits/sec 0 071 ms 1194/ 1198 (1e+002%)

```

7 Datos regidos en el servidor de jperf con 25 flujos



```
bin/iperf.exe -s -u -P 0 1 1 -p 1234 -l 200 0B -f k
```

```

-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----

```

```

OpenSCManager failed - Acceso denegado (0x5)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1411
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec 7 81 KBytes 64 0 Kbits/sec 0 635 ms 1852796269/ 40
(4 6e+009%)

```

```

[112] 1 0- 2 0 sec 0 20 KBytes 1 60 Kbits/sec 0 713 ms 0/ 1 (0%)
[112] 2 0- 3 0 sec 5 08 KBytes 41 6 Kbits/sec 0 419 ms 0/ 26 (0%)
[112] 3 0- 4 0 sec 9 38 KBytes 76 8 Kbits/sec 1 909 ms 0/ 48 (0%)
[112] 4 0- 5 0 sec 6 25 KBytes 51 2 Kbits/sec 1 027 ms 0/ 32 (0%)
[112] 5 0- 6 0 sec 6 84 KBytes 56 0 Kbits/sec 1 609 ms 0/ 35 (0%)
[112] 6 0- 7 0 sec 7 03 KBytes 57 6 Kbits/sec 1 596 ms 0/ 36 (0%)
[112] 7 0- 8 0 sec 9 77 KBytes 80 0 Kbits/sec 1 594 ms 0/ 50 (0%)
[112] 8 0- 9 0 sec 6 25 KBytes 51 2 Kbits/sec 1 556 ms 0/ 32 (0%)
[112] 9 0-10 0 sec 9 57 KBytes 78 4 Kbits/sec 0 874 ms 0/ 49 (0%)
[112] 10 0-11 0 sec 6 25 KBytes 51 2 Kbits/sec 1 657 ms 0/ 32 (0%)
[112] 11 0-12 0 sec 6 45 KBytes 52 8 Kbits/sec 0 742 ms 0/ 33 (0%)
[112] 12 0 13 0 sec 5 27 KBytes 43 2 Kbits/sec 13 486 ms 0/ 27 (0%)
[112] 13 0 14 0 sec 4 69 KBytes 38 4 Kbits/sec 3 284 ms 0/ 24 (0%)
[112] 14 0 15 0 sec 8 79 KBytes 72 0 Kbits/sec 2 702 ms 0/ 45 (0%)
[112] 15 0 16 0 sec 4 10 KBytes 33 6 Kbits/sec 1 557 ms 0/ 21 (0%)
[112] 16 0 17 0 sec 7 81 KBytes 64 0 Kbits/sec 1 265 ms 0/ 40 (0%)
[112] 17 0-18 0 sec 6 64 KBytes 54 4 Kbits/sec 0 867 ms 0/ 34 (0%)
[112] 18 0 19 0 sec 2 34 KBytes 19 2 Kbits/sec 1 982 ms 0/ 12 (0%)
[112] 19 0-20 0 sec 6 64 KBytes 54 4 Kbits/sec 0 920 ms 0/ 34 (0%)
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 20 0-21 0 sec 9 77 KBytes 80 0 Kbits/sec 5 542 ms 0/ 50 (0%)
[112] 21 0-22 0 sec 6 64 KBytes 54 4 Kbits/sec 4 937 ms 0/ 34 (0%)
[112] 22 0-23 0 sec 3 52 KBytes 28 8 Kbits/sec 3 853 ms 0/ 18 (0%)
[112] 23 0 24 0 sec 7 42 KBytes 60 8 Kbits/sec 3 862 ms 0/ 38 (0%)
[112] 24 0-25 0 sec 7 42 KBytes 60 8 Kbits/sec 1 544 ms 0/ 38 (0%)
[112] 25 0-26 0 sec 4 69 KBytes 38 4 Kbits/sec 1 568 ms 0/ 24 (0%)
[112] 26 0-27 0 sec 7 23 KBytes 59 2 Kbits/sec 0 736 ms 0/ 37 (0%)
[112] 27 0-28 0 sec 7 23 KBytes 59 2 Kbits/sec 1 248 ms 0/ 37 (0%)
[112] 28 0-29 0 sec 5 27 KBytes 43 2 Kbits/sec 6 683 ms 0/ 27 (0%)
[112] 29 0-30 0 sec 4 10 KBytes 33 6 Kbits/sec 4 169 ms 0/ 21 (0%)
[112] 0 0-30 3 sec 191 KBytes 51 5 Kbits/sec 3 928 ms 0/ 977 (0%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1388
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[112] 0 0 0 7 sec 5 66 KBytes 63 2 Kbits/sec 0 739 ms 1161/ 1190 (98%)

```

8 Datos regidos en el servidor de jperf con 30 flujos



```
bin/iperf.exe -s -u -P 0 -i 1 p 1234 -l 200 0B -f k
```

```
-----
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
-----
```

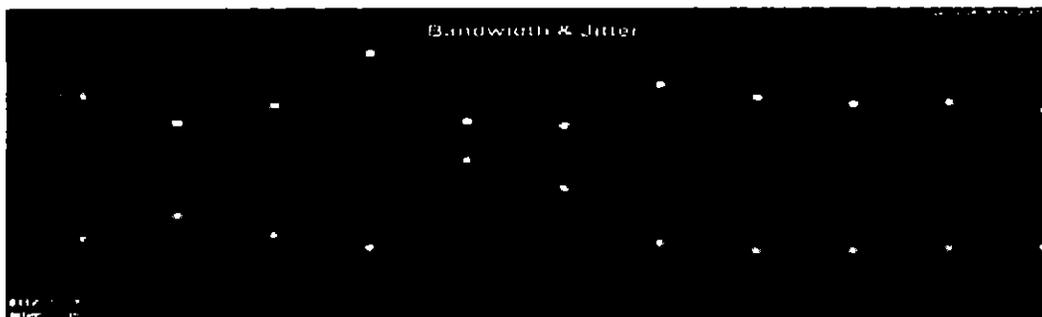
```
OpenSCManager failed Acceso denegado (0x5)
```

```

[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1412
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec    7 03 KBytes   57 6 Kbits/sec  0 420 ms    1852796269/ 36
(5 1e+009%)
[112] 1 0- 2 0 sec    0 20 KBytes   1 60 Kbits/sec  0 549 ms    0/ 1 (0%)
[112] 2 0- 3 0 sec    0 00 KBytes   0 00 Kbits/sec  0 549 ms    0/ 0 (-1 %)
[112] 3 0- 4 0 sec    6 45 KBytes   52 8 Kbits/sec  0 464 ms    0/ 33 (0%)
[112] 4 0- 5 0 sec    7 42 KBytes   60 8 Kbits/sec  0 547 ms    0/ 38 (0%)
[112] 5 0- 6 0 sec    7 81 KBytes   64 0 Kbits/sec  0 539 ms    0/ 40 (0%)
[112] 6 0- 7 0 sec    7 81 KBytes   64 0 Kbits/sec  0 711 ms    0/ 40 (0%)
[112] 7 0- 8 0 sec    7 62 KBytes   62 4 Kbits/sec  0 379 ms    0/ 39 (0%)
[112] 8 0- 9 0 sec    8 40 KBytes   68 8 Kbits/sec  1 215 ms    0/ 43 (0%)
[112] 9 0-10 0 sec   8 20 KBytes   67 2 Kbits/sec  0 511 ms    0/ 42 (0%)
[112] 10 0-11 0 sec   7 03 KBytes   57 6 Kbits/sec  0 430 ms    0/ 36 (0%)
[112] 11 0-12 0 sec   7 81 KBytes   64 0 Kbits/sec  0 846 ms    0/ 40 (0%)
[112] 12 0-13 0 sec   7 62 KBytes   62 4 Kbits/sec  0 503 ms    0/ 39 (0%)
[112] 13 0-14 0 sec   7 81 KBytes   64 0 Kbits/sec  1 453 ms    0/ 40 (0%)
[112] 14 0-15 0 sec   8 01 KBytes   65 6 Kbits/sec  0 973 ms    0/ 41 (0%)
[112] 15 0-16 0 sec   7 81 KBytes   64 0 Kbits/sec  1 296 ms    0/ 40 (0%)
[112] 16 0-17 0 sec   7 81 KBytes   64 0 Kbits/sec  0 843 ms    0/ 40 (0%)
[112] 17 0-18 0 sec   7 23 KBytes   59 2 Kbits/sec  0 776 ms    0/ 37 (0%)
[112] 18 0-19 0 sec   7 42 KBytes   60 8 Kbits/sec  0 706 ms    0/ 38 (0%)
[112] 19 0-20 0 sec   8 59 KBytes   70 4 Kbits/sec  0 597 ms    0/ 44 (0%)
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 20 0-21 0 sec   6 64 KBytes   54 4 Kbits/sec  1 593 ms    0/ 34 (0%)
[112] 21 0-22 0 sec   7 81 KBytes   64 0 Kbits/sec  0 974 ms    0/ 40 (0%)
[112] 22 0-23 0 sec   10 4 KBytes   84 8 Kbits/sec  1 088 ms    0/ 53 (0%)
[112] 23 0-24 0 sec   7 81 KBytes   64 0 Kbits/sec  0 595 ms    0/ 40 (0%)
[112] 24 0-25 0 sec   4 69 KBytes   38 4 Kbits/sec  0 871 ms    0/ 24 (0%)
[112] 25 0-26 0 sec   7 62 KBytes   62 4 Kbits/sec  0 470 ms    0/ 39 (0%)
[112] 26 0-27 0 sec   8 59 KBytes   70 4 Kbits/sec  0 884 ms    0/ 44 (0%)
[112] 27 0-28 0 sec   8 20 KBytes   67 2 Kbits/sec  1 975 ms    0/ 42 (0%)
[112] 28 0-29 0 sec   7 62 KBytes   62 4 Kbits/sec  0 611 ms    0/ 39 (0%)
[112] 29 0-30 0 sec   8 01 KBytes   65 6 Kbits/sec  0 518 ms    0/ 41 (0%)
[112] 0 0-30 2 sec   216 KBytes   58 5 Kbits/sec  1 076 ms    0/ 1105 (0%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1427
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 0 5 sec    3 52 KBytes   63 1 Kbits/sec  0 792 ms    998/ 1016 (98%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1439
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 0 7 sec    5 86 KBytes   64 5 Kbits/sec  1 266 ms    978/ 1008 (97%)

```

9 Datos regidos en el servidor de jperf con 40 flujos



```
bin/iperf.exe -s u P 0 i 1 p 1234 -l 200 0B -f k
```

```
-----  
Server listening on UDP port 1234  
Receiving 200 byte datagrams  
UDP buffer size 8 00 KByte (default)  
-----
```

```
OpenSCManager failed - Acceso denegado (0x5)
```

```
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1591
```

```
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams  
[112] 0 0- 1 0 sec    7 81 KBytes   64 0 Kbits/sec  0 752 ms    1852796269/ 40  
(4 6e+009%)
```

```
[112] 1 0- 2 0 sec    0 20 KBytes   1 60 Kbits/sec  0 705 ms    0/ 1 (0%)
```

```
[112] 2 0- 3 0 sec    0 00 KBytes   0 00 Kbits/sec  0 705 ms    0/ 0 (-1 $%)
```

```
[112] 3 0- 4 0 sec    0 20 KBytes   1 60 Kbits/sec  0 898 ms    0/ 1 (0%)
```

```
[112] 4 0- 5 0 sec    1 37 KBytes   11 2 Kbits/sec  0 838 ms    0/ 7 (0%)
```

```
[112] 5 0- 6 0 sec    6 25 KBytes   51 2 Kbits/sec  1 849 ms    0/ 32 (0%)
```

```
[112] 6 0- 7 0 sec    11 7 KBytes   96 0 Kbits/sec  0 862 ms    0/ 60 (0%)
```

```
[112] 7 0- 8 0 sec    5 66 KBytes   46 4 Kbits/sec  0 875 ms    0/ 29 (0%)
```

```
[112] 8 0- 9 0 sec    4 10 KBytes   33 6 Kbits/sec  0 983 ms    0/ 21 (0%)
```

```
[112] 9 0-10 0 sec    2 73 KBytes   22 4 Kbits/sec  7 037 ms    0/ 14 (0%)
```

```
[112] 10 0-11 0 sec    8 01 KBytes   65 6 Kbits/sec  3 020 ms    0/ 41 (0%)
```

```
[112] 11 0-12 0 sec    5 27 KBytes   43 2 Kbits/sec  0 935 ms    0/ 27 (0%)
```

```
[112] 12 0-13 0 sec    3 52 KBytes   28 8 Kbits/sec  0 994 ms    0/ 18 (0%)
```

```
[112] 13 0-14 0 sec    11 5 KBytes   94 4 Kbits/sec  1 499 ms    0/ 59 (0%)
```

```
[112] 14 0-15 0 sec    8 59 KBytes   70 4 Kbits/sec  0 915 ms    0/ 44 (0%)
```

```
[112] 15 0-16 0 sec    0 98 KBytes   8 00 Kbits/sec  7 193 ms    0/ 5 (0%)
```

```
[112] 16 0-17 0 sec    10 7 KBytes   88 0 Kbits/sec  1 872 ms    0/ 55 (0%)
```

```
[112] 17 0-18 0 sec    5 08 KBytes   41 6 Kbits/sec  0 637 ms    0/ 26 (0%)
```

```
[112] 18 0-19 0 sec    6 05 KBytes   49 6 Kbits/sec  1 608 ms    0/ 31 (0%)
```

```
[112] 19 0-20 0 sec    6 05 KBytes   49 6 Kbits/sec  3 034 ms    0/ 31 (0%)
```

```
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
```

```
[112] 20 0-21 0 sec    2 73 KBytes   22 4 Kbits/sec  6 980 ms    0/ 14 (0%)
```

```
[112] 21 0-22 0 sec    4 88 KBytes   40 0 Kbits/sec  3 525 ms    0/ 25 (0%)
```

```
[112] 22 0-23 0 sec    11 3 KBytes   92 8 Kbits/sec  1 571 ms    0/ 58 (0%)
```

```
[112] 23 0-24 0 sec    2 93 KBytes   24 0 Kbits/sec  16 233 ms    0/ 15 (0%)
```

```
[112] 24 0-25 0 sec    2 34 KBytes   19 2 Kbits/sec  11 421 ms    0/ 12 (0%)
```

```
[112] 25 0-26 0 sec    7 42 KBytes   60 8 Kbits/sec  2 254 ms    0/ 38 (0%)
```

```
[112] 26 0-27 0 sec    5 86 KBytes   48 0 Kbits/sec  1 063 ms    0/ 30 (0%)
```

```
[112] 27 0-28 0 sec    5 08 KBytes   41 6 Kbits/sec  1 063 ms    0/ 26 (0%)
```

```
[112] 28 0-29 0 sec    5 27 KBytes   43 2 Kbits/sec  1 566 ms    0/ 27 (0%)
```

```
[112] 29 0-30 0 sec    4 30 KBytes   35 2 Kbits/sec  1 610 ms    0/ 22 (0%)
```

```
[112] 0 0-30 2 sec    158 KBytes   43 0 Kbits/sec  2 075 ms    0/ 811 (0%)
```

```
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1571
```

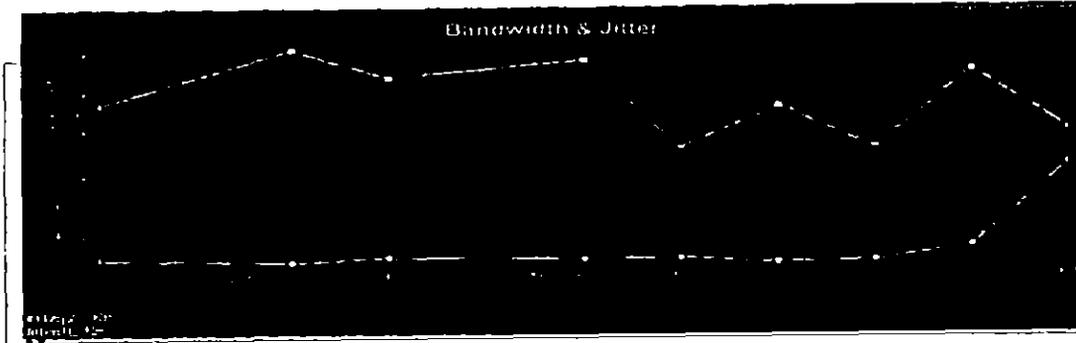
```
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
```

```
[112] 0 0- 1 0 sec    8 98 KBytes   73 6 Kbits/sec  1 120 ms    844/ 890 (95%)
```

```
[112] 1 0- 2 0 sec    5 86 KBytes   48 0 Kbits/sec  3 445 ms    0/ 30 (0%)
```

```
[112] 0 0- 2 0 sec    15 0 KBytes   60 7 Kbits/sec  3 269 ms    844/ 921 (92%)
```

10 Datos rigidos en el servidor de jperf con 60 flujos



```
bin/iperf.exe s u P 0 1 1 p 1234 -l 200 0B -f k
```

```
Server listening on UDP port 1234
Receiving 200 byte datagrams
UDP buffer size 8 00 KByte (default)
```

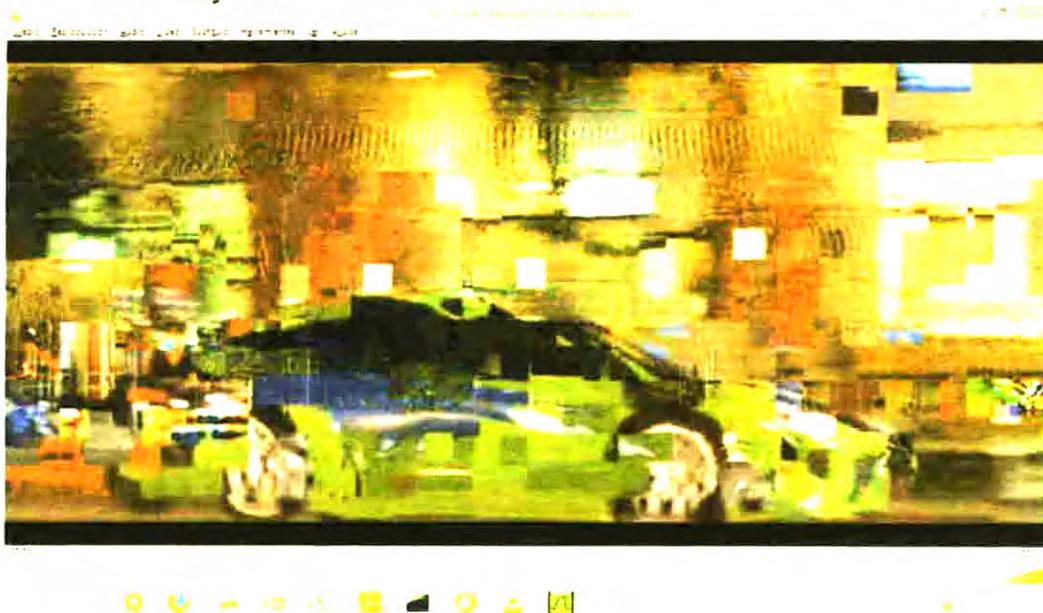
```
OpenSCManager failed - Acceso denegado (0x5)
```

```
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1593
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0- 1 0 sec    6 84 KBytes   56 0 Kbits/sec  0 832 ms    1852796269/ 35
(5 3e+009%)
[112] 1 0- 2 0 sec    0 20 KBytes   1 60 Kbits/sec  0 897 ms    0/ 1 (0%)
[112] 2 0- 3 0 sec    0 00 KBytes   0 00 Kbits/sec  0 897 ms    0/ 0 (-1 $%)
[112] 3 0- 4 0 sec    0 00 KBytes   0 00 Kbits/sec  0 897 ms    0/ 0 (-1 $%)
[112] 4 0- 5 0 sec    0 00 KBytes   0 00 Kbits/sec  0 897 ms    0/ 0 (-1 $%)
[112] 5 0- 6 0 sec    0 00 KBytes   0 00 Kbits/sec  0 897 ms    0/ 0 (-1 $%)
[112] 6 0- 7 0 sec    0 20 KBytes   1 60 Kbits/sec  0 885 ms    0/ 1 (0%)
[112] 7 0- 8 0 sec    0 20 KBytes   1 60 Kbits/sec  0 985 ms    0/ 1 (0%)
[112] 8 0- 9 0 sec    0 20 KBytes   1 60 Kbits/sec  1 041 ms    0/ 1 (0%)
[112] 9 0-10 0 sec    1 95 KBytes   16 0 Kbits/sec  1 669 ms    0/ 10 (0%)
[112] 10 0-11 0 sec    9 57 KBytes   78 4 Kbits/sec  2 027 ms    0/ 49 (0%)
[112] 11 0-12 0 sec    0 00 KBytes   0 00 Kbits/sec  2 027 ms    0/ 0 (-1 $%)
[112] 12 0-13 0 sec    13 5 KBytes   110 Kbits/sec  1 357 ms    0/ 69 (0%)
[112] 13 0-14 0 sec    0 00 KBytes   0 00 Kbits/sec  1 357 ms    0/ 0 (-1 $%)
[112] 14 0-15 0 sec    10 4 KBytes   84 8 Kbits/sec  0 546 ms    0/ 53 (0%)
[112] 15 0-16 0 sec    0 00 KBytes   0 00 Kbits/sec  0 546 ms    0/ 0 (-1 $%)
[112] 16 0-17 0 sec    8 40 KBytes   68 8 Kbits/sec  3 354 ms    16/ 59 (27%)
[112] 17 0-18 0 sec    0 00 KBytes   0 00 Kbits/sec  3 354 ms    0/ 0 (-1 $%)
[112] 18 0-19 0 sec    12 9 KBytes   106 Kbits/sec  2 459 ms    0/ 66 (0%)
[112] 19 0-20 0 sec    7 23 KBytes   59 2 Kbits/sec  1 440 ms    0/ 37 (0%)
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 20 0-21 0 sec    0 00 KBytes   0 00 Kbits/sec  1 440 ms    0/ 0 (-1 $%)
[112] 21 0-22 0 sec    16 0 KBytes   131 Kbits/sec  0 761 ms    0/ 82 (0%)
[112] 22 0-23 0 sec    11 5 KBytes   94 4 Kbits/sec  1 582 ms    0/ 59 (0%)
[112] 23 0-24 0 sec    0 00 KBytes   0 00 Kbits/sec  1 582 ms    0/ 0 (-1 $%)
[112] 24 0-25 0 sec    14 3 KBytes   117 Kbits/sec  1 261 ms    0/ 73 (0%)
[112] 25 0-26 0 sec    0 20 KBytes   1 60 Kbits/sec  1 415 ms    0/ 1 (0%)
[112] 26 0-27 0 sec    6 84 KBytes   56 0 Kbits/sec  0 618 ms    0/ 35 (0%)
[112] 27 0-28 0 sec    0 20 KBytes   1 60 Kbits/sec  0 893 ms    0/ 1 (0%)
[112] 28 0-29 0 sec    12 3 KBytes   101 Kbits/sec  3 294 ms    0/ 63 (0%)
[112] 29 0-30 0 sec    2 93 KBytes   24 0 Kbits/sec  16 973 ms    0/ 15 (0%)
[112] 0 0-30 8 sec    136 KBytes   36 2 Kbits/sec  15 808 ms    16/ 713 (2 2%)
[112] local 192 168 18 78 port 1234 connected with 192 168 18 79 port 1631
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[112] 0 0 1 0 sec    6 05 KBytes   49 6 Kbits/sec  1 021 ms    790/ 821 (96%)
[112] 0 0 1 6 sec    12 3 KBytes   62 7 Kbits/sec  1 025 ms    790/ 853 (93%)
```

```
[112] local 192.168.18.78 port 1234 connected with 192.168.18.79 port 1644
[ ID] Interval      Transfer    Bandwidth   Jitter    Lost/Total Datagrams
[112] 0.0- 1.0 sec   4.49 KBytes 36.8 Kbits/sec 2.161 ms 758/ 781 (97%)
[112] 0.0- 1.6 sec   8.98 KBytes 45.7 Kbits/sec 4.373 ms 758/ 804 (94%)
```

Prueba de emisión del video streaming sin QoS

- Prueba de streaming sin QoS usando MPEG4 de video y MPEG3 para audio en la transmisión. VLC transcodifica de Divx a esta configuración.
- Se usa tráfico cruzado de un flujo udp creciente de 200 en 200 Kbytes/s hasta los 2000 Kbytes/s





Prueba de emisión del video streaming con QoS

- Prueba de streaming con QoS usando MPEG4 de video y MPEG3 para audio en la transmisión. VLC transcodifica de Divx a esta configuración.
- Se usa tráfico cruzado de un flujo udp creciente de 200 en 200 Kbytes/s hasta los 2000 Kbytes/s.



Figure 1: A photograph of a woman and a dog in a room.





UNIVERSIDAD DE PANAMA
Vicerrectoría de Investigación y Postgrado



C E I J R E I I

AIDELÉN DANAIS CHUNG

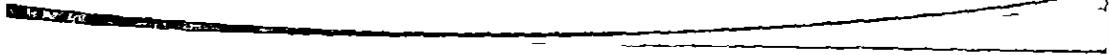
TRABAJO DE INVESTIGACIÓN
ANEXO C E I N I T I O I E R T I I D

“Optimización de una red Mesh inalámbrica para tráfico multimedia mediante calidad de servicio”

R E I d I C p D I I m I A M d J I I I I d b I I O I

[Handwritten signature]

Juan Carlos Chung



Optimización de una Red Mesh Inalámbrica para Tráfico Multimedia Mediante Calidad de Servicio

Aidelen Danaís Chung¹ Iván Armuelles Voinov²

Centro de Investigación en Tecnologías de la Información y Comunicación
Facultad de Informática, Electrónica y Comunicación
Universidad de Panamá

¹aidelen.chung@citicup.org

²ivan.armuelles@citicup.org

Las Redes Mesh Inalámbricas (WMN) son redes de datos con topología de red de malla formadas por nodos que establecen y mantienen la conectividad de red automáticamente éstas surgieron como una tecnología capaz de brindar conectividad de banda ancha segura y de bajo costo. Proporcionan una solución barata, rápida y eficaz para las redes inalámbricas de datos en zonas urbanas, suburbanas y en los entornos rurales. Son redes inalámbricas auto-organizada, auto-configurables y son fácilmente adaptables a diferentes necesidades de tráfico y los cambios de la red. Otra de las ventajas del uso de WMN es su capacidad de reconstrucción de rutas al haber pérdida de enlaces, su escalabilidad y el manejo de múltiples interfaces de radio y antenas inteligentes. Esta tecnología presenta aplicaciones interesantes, por ejemplo en redes domésticas de banda ancha, redes comunitarias de vecindario, la automatización de edificios, redes corporativas, video vigilancia, para casos de emergencia y desastre.

Para que las redes malladas inalámbricas puedan trabajar se requiere de un protocolo de enrutamiento. Utilizaremos el protocolo que propone el estándar para redes mesh (IEEE 802.11s) éste propone un protocolo de selección de rutas basado en direcciones de capa de enlace de datos (direcciones MAC). Este protocolo se conoce como HWMP (Hybrid Wireless Mesh Protocol). Es un protocolo híbrido basado en AODV (Ad hoc On Demand Distance Vector). AODV permite el enrutamiento dinámico, autoarranque y multi salto entre todos los nodos móviles que participan en la red y utiliza la métrica de tiempo en el aire (air time-metric) que tiene en cuenta la velocidad de transmisión de datos, la sobrecarga y tasa de error en la trama.

La calidad de servicio (QoS) se refiere a la capacidad de la red para garantizar un cierto nivel de rendimiento para un flujo, especialmente para el tráfico multimedia. Las métricas de calidad de servicio son el rendimiento, ancho de banda, fiable, jitter y el retraso. Son requisitos básicos para un mejor tráfico en aplicaciones multimedia reales y de voz.

Para la creación de redes basados en el protocolo IEEE 802.11s se puede utilizar software de código abierto en los routers (encaminadores) inalámbricos y de descarga gratuita. Utilizaremos OpenWRT que es un firmware de software libre que permite personalizar completamente las aplicaciones en el router inalámbrico e implementa protocolos de enrutamiento se basa en el estándar 802.11s y tiene soporte a calidad de servicio.

Un hecho importante de las Redes Mesh Inalámbricas es que proveen soporte a servicios multimedia reales. Pero la transmisión de servicios multimedia reales presenta problemas de rendimiento inestable, insuficiente ancho de banda, jitter y mayor pérdida de paquetes siendo difícil ofrecer garantías del servicio en una Red. El propósito de este proyecto es ofrecer una solución mediante la implementación de los mecanismos de calidad de servicio en una red de prueba (testbed) inalámbrica multi saltos para las aplicaciones multimedia reales y garantizar que la transmisión de los datos sea más eficiente utilizando el estándar IEEE 802.11s con un firmware de software libre como OpenWRT.

Comunicaciones Orales

CP-CNT-036

OPTIMIZACIÓN DE UNA RED MESH INALÁMBRICA PARA TRÁFICO MULTIMEDIA MEDIANTE CALIDAD DE SERVICIO

Aldelán Daniela Chung¹ Iván Armuelles Volnov²

Centro de Investigación en Tecnologías de la Información y Comunicación

Facultad de Informática, Electrónica y Comunicación, Universidad de Panamá

Palabras clave: redes mesh, calidad de servicio, HWMP

Las Redes Mesh inalámbricas (WMN) son redes de datos con topología de red de malla formadas por nodos que establecen y mantienen la conectividad de red automáticamente; éstas surgieron como una tecnología capaz de brindar conectividad de banda ancha segura y de bajo costo. Proporcionan una solución barata, rápida y eficaz para las redes inalámbricas de datos en zonas urbanas, suburbanas y en los entornos rurales. Son redes inalámbricas auto-organizadas, auto-configurables y son fácilmente adaptables a diferentes necesidades de tráfico y los cambios de la red. Otra de las ventajas del uso de WMN, es su capacidad de reconstrucción de rutas al haber pérdida de enlace, su escalabilidad y el manejo de múltiples interfaces de radio y antenas inteligentes. Esta tecnología presenta aplicaciones interesantes, por ejemplo, en redes domésticas de banda ancha, redes comunitarias, de vecindario, la automatización de edificios, redes corporativas, video vigilancia, para casos de emergencia y desastre. Para que las redes mallas inalámbricas puedan trabajar se requiere de un protocolo de enrutamiento. Utilizaremos el protocolo que propone el estándar para redes mesh (IEEE 802.11s); éste propone un protocolo de selección de rutas basado en direcciones de capa de enlace de datos (direcciones MAC). Este protocolo se conoce como HWMP (Hybrid Wireless Mesh Protocol). Es un protocolo híbrido, basado en AODV (Ad hoc On-Demand Distance Vector). AODV permite el enrutamiento dinámico, autoarranque y multi-salto entre todos los nodos móviles que participan en la red y utiliza la métrica de tiempo en el aire (air-time-metric), que tiene en cuenta la velocidad de transmisión de datos, la sobrecarga y tasa de error en la trama.

La calidad de servicio (QoS) se refiere a la capacidad de la red para garantizar un cierto nivel de rendimiento para un flujo, especialmente para el tráfico multimedia. Las métricas de calidad de servicio son: el rendimiento, ancho de banda fiable, jitter y el retraso. Son requisitos básicos para un mejor tráfico en aplicaciones multimedia reales y de voz.

Para la creación de redes basadas en el protocolo IEEE 802.11s, se puede utilizar software de código abierto en los routers (encaminadores) inalámbricos y de descarga gratuita. Utilizaremos OpenWRT que es un firmware de software libre que permite personalizar completamente las aplicaciones en el router inalámbrico e implementa protocolos de enrutamiento, se basa en el estándar 802.11s y tiene soporte a calidad de servicio. Un hecho importante de las Redes Mesh Inalámbricas es que proveen soporte a servicios multimedia reales. Pero la transmisión de servicios multimedia reales presenta problemas de rendimiento inestable, insuficiente ancho de banda, jitter y mayor pérdida de paquetes siendo difícil ofrecer garantías del servicio en una Red. El propósito de este proyecto es ofrecer una solución mediante la implementación de los mecanismos de calidad de servicio en una red de prueba (testbed) inalámbrica multi-saltos para las aplicaciones multimedia reales y garantizar que la transmisión de los datos sea más eficiente, utilizando el estándar IEEE 802.11s con herramienta un firmware de software libre como OpenWRT.



“Optimización de una Red Mesh Inalámbrica para Tráfico Multimedia Mediante Calidad de Servicio”

Adrián Chong, Juan Armando Sosa
Centro de Investigación en Tecnología de la Información y Comunicación
Facultad de Informática, Electrónica y Comunicación
Universidad de Panamá
Email: a.chong@up.edu.pa, javasosa@up.edu.pa

I. Introducción

Las Redes Mesh Inalámbricas (WMIs) son un tipo de red con topología de malla de redes formadas por nodos que establecen y mantienen la conexión entre los nodos entre sí para intercambiar los datos. Las WMIs se diferencian de las redes de acceso de banda ancha, como las redes de telefonía móvil, en que los nodos de la red pueden ser tanto clientes como estaciones de base. Este tipo de red permite un mayor ancho de banda y una mayor flexibilidad en la configuración de la red.

Una de las ventajas de las WMIs es su capacidad de reconfiguración de sus nodos de acuerdo a las necesidades de la red. Esto permite un mayor ancho de banda y una mayor flexibilidad en la configuración de la red. Sin embargo, esta flexibilidad también puede ser utilizada para realizar ataques de denegación de servicio (DDoS) a través de la red. Por lo tanto, es necesario implementar medidas de seguridad para proteger la red de estos tipos de ataques.

En el presente proyecto, se investigó y se implementó la red de la Universidad de Panamá en un entorno de simulación de red. El objetivo principal de este proyecto es optimizar la red de la Universidad de Panamá para soportar el tráfico multimedia. Para ello, se implementó un sistema de gestión de la red que permite configurar y controlar los recursos de la red de acuerdo a las necesidades de los usuarios.

A. Objetivos

Objetivo General
Optimizar la red de la Universidad de Panamá para soportar el tráfico multimedia mediante la implementación de un sistema de gestión de la red.

- Implementar una red de acceso inalámbrico (WLAN) mediante el protocolo IEEE 802.11n en un entorno de simulación de red.
- Realizar un análisis de tráfico de la red de acuerdo a los requisitos de servicio de los usuarios.
- Analizar los requisitos de los dispositivos de acceso inalámbrico para determinar los requisitos de la red.
- Realizar la configuración de la red de acuerdo a los requisitos de los dispositivos de acceso inalámbrico.
- Implementar un sistema de gestión de la red para controlar los recursos de la red.



Figura 1. Esquema del Tectbed implementado

II. Metodología

Para que las WMIs puedan brindar los servicios de un sistema de comunicación, se debe tener en cuenta los requisitos de los usuarios. En este caso, se investigó y se implementó un sistema de gestión de la red que permite configurar y controlar los recursos de la red de acuerdo a las necesidades de los usuarios. El sistema de gestión de la red se implementó mediante el uso de un software de simulación de red. Este software permite configurar y controlar los recursos de la red de acuerdo a las necesidades de los usuarios. El sistema de gestión de la red se implementó mediante el uso de un software de simulación de red. Este software permite configurar y controlar los recursos de la red de acuerdo a las necesidades de los usuarios.

Tabla 1. Componentes de software

Nombre del Componente	Versiones
Entorno de Simulación de Red	OpenFlow 13.0.1
Generador de Tráfico	iperf
Plataforma de Virtualización	Proxmox VE 5.1
Análisis de Tráfico	Wireshark
Cliente y Servidor de Video Streaming	VLC
Servidor de VoIP	Gate / Asterisk
Cliente SIP para VoIP	Issi / Gnu
Cliente SIP para Video Conferencia	Jitsi



Figura 2. Funcionamiento de OpenWRT

IV. Conclusiones

El presente artículo muestra que la red de acceso inalámbrico (WLAN) implementada en un entorno de simulación de red puede soportar el tráfico multimedia mediante la implementación de un sistema de gestión de la red. El sistema de gestión de la red se implementó mediante el uso de un software de simulación de red. Este software permite configurar y controlar los recursos de la red de acuerdo a las necesidades de los usuarios. El sistema de gestión de la red se implementó mediante el uso de un software de simulación de red. Este software permite configurar y controlar los recursos de la red de acuerdo a las necesidades de los usuarios.



UNIVERSIDAD DE PANAMÁ
FACULTAD DE INFORMÁTICA, ELECTRÓNICA
Y COMUNICACIÓN

Otorga el presente CERTIFICADO a:

Aidelen Chung

Por su Participación en el **Primer Congreso Internacional
Informática, Electrónica y Comunicación**, desarrollado del
24 al 26 de Agosto del 2011.

Dado en la Ciudad Universitaria a los 29 días del mes de agosto de dos mil once



Dra. Diana Chen
Decana





Ing. Félix Morenc
Secretario Administrativo

Se extiende el presente certificado:

A _____

Por su participación en la
Primera Conferencia de Directores de Tecnologías de Información y Comunicación de Instituciones de
Educación Superior (TICAL 2011)
realizada el 20 y 21 de junio en Ciudad del Saber, Ciudad de Panamá, Panamá


Florencio Utreras Díaz
Director Ejecutivo
RedCLARA

Ciudad de Panamá, 21 de junio de 2011



A Performance Analysis of Wireless Mesh Networks Implementations Based on Open Source Software

Iván Armuelles Voinov, Aidelen Chung Cedefío, Joaquín Chung, Grace González

Research Center for Information and Communication Technologies,
University of Panama, Republic of Panama
(iarmuelles, achung, jchung, ggonzalez)@citicup.org

Abstract. Wireless mesh networks (WMNs) have emerged as a promising technology, capable of provide broadband connectivity at low cost. Implementations based on Open Source Software of these networks offer advantages for providing broadband networking communications in scenarios where cabling is too expensive or prohibitive such as rural environments. In this paper we evaluate the performance of small scale wireless mesh WMN routing protocols for WMNs: B.A.T.M.A.N. Advanced and the 802.11s standard. We also compare an OpenFlow controller implemented over the WMN, verifying their bandwidth, datagram loss and jitter.

Keywords: Open Source Software for research and innovation, Wireless Mesh Networks, OpenFlow, OpenWRT, network performance.

1 Introduction

Providing telecommunication services to difficult access areas (such as rural environments) is still difficult due to the lack of appropriate or inexpensive infrastructure. In this context, Wireless Mesh Networks (WMN) are an attractive solution for these scenarios due to their lower deployment costs and

ease of expansion WMNs could be used in community networks, home networking, video surveillance and emergency/disaster situations [1] However WMNs face several restrictions such as low-end equipment, single wireless channel and interferences, which degrade the overall performance of the network, and impose many drawbacks to even current and standard Internet's services In this paper we evaluate the performance of WMN implementations based on Open Source Software for multimedia service transport. In our case WMNs based on layer 2 routing protocol raise as a suitable and optimal connectivity choice, as any layer 3 addressing protocol could be used on top either IPv4 or IPv6 Hence we compare IEEE 802.11s standard [2] against Better Approach To Mobile Adhoc Networking Advanced protocol (B.A.T.M.A.N.) [3] Also we compare these two protocols with the OpenFlow protocol [4] which is used to control the forwarding tables of switches, routers and access points from a remote server leveraging innovative services over the network such as access control, network virtualization, mobility network management and visualization. The paper is organized as follows in Section 2 we describe the implementation of our testbed. In Section 3 we describe the test and show the results obtained. Finally in Section 4 we describe our conclusions

2 Testbed Implementation

The testbed consisted of a small scale WMN composed by four wireless routers. The experiments were conducted inside a small laboratory because this indoor environment is very similar to the conditions of a real home networking scenario (Fig 1) All the wireless routers used in this testbed were TP Link TL-WR1043ND v1.8, with four LAN ports of 1 Gbit/s Ethernet, one WAN port of 1 Gbit/s Ethernet, and one 802.11n wireless interface that works in the 2.4 GHz frequency band. We replaced the firmware of the wireless routers with the open firmware OpenWRT [5] which is a very well-known Linux distribution for embedded devices OpenWRT supports WMN with routing protocols like OLSR, B.A.T.M.A.N. and the new standard for WMN networks 802.11s For the experiments we used the OpenWRT Backfire 10.03.1 which comes with 802.11s support by default. To evaluate B.A.T.M.A.N. Advanced, was necessary to install the batman-adv package For the experiments with OpenFlow we compiled OpenWRT with a package called Pantou that supports the OpenFlow protocol The OpenFlow controller used in the experiments was POX, which is a controller based on NOX for rapid deployments of SDNs using Python A list of all the components of the testbed and used tools based on Open Source Software is shown in the table 1

3 Performance Evaluation

Our experiment had three scenarios, the first one was a WMN composed by four MP configured with the 802.11s standard. The second scenario was the same four MP but using the batman-adv protocol The last one was the four MP connected to an OpenFlow controller using an out of band network for control and a data network i.e. a wired network for the control signaling, due to a limitation of the hardware selected The tests were conducted in a low interference environment, which maintains the optimum conditions for VoIP traffic (packet loss should not exceed 1%, the maximum delay should <150 ms and jitter must be kept below 20 ms) For our study we took measurements using Iperf either in UDP and TCP mode Every UDP test was maintained for 300 seconds and repeated ten times, while the TCP test where maintained for 180 seconds and repeated five times Every test was made for one hop two hops, three hops and four hops, for both scenarios.

In the *UDP* test we found that for one and two hops the maximum *throughput* was higher in the Batman-adv scenario This is because in the Batman-adv implementation, the wireless interfaces synchronize at 802.11n (transmission rates up to 300 Mbit/s) On the other hand, the implementation of 802.11s only synchronizes at

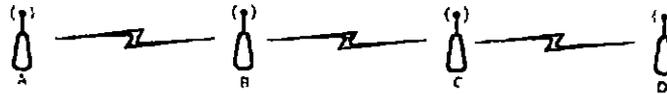


Fig. 1 WMN implementation

Table 1 Tools based on Open Source Software for WMN implementation and their evaluation

Name	Supported Platform	Features
OpenWRT	Wide variety of wireless routers	Allows you to customize the applications on the wireless router. It implements routing protocols such as OLSR and BATMAN. It can be adapted to work with IPv6 and supports 802.11s. OpenWrt is the framework to build an application without having to build a complete firmware around it.
Pantou	Linksys and TP Link Wireless Router	Implementation of the OpenFlow protocol for the OpenWRT firmware
NOX/POX	Linux	OpenFlow controller based on Python and C++
Mininet	Linux	Allows you to create scalable software defined networks within a single PC
Insider	Linux and Windows	Locate wireless networks and measures the intensity of their signals.
Iperf	Linux and Windows	Creates TCP and UDP data flows to measure the behavior of the network with respect to some QoS parameters
Wireshark	Linux and Windows	Protocol analyzer used for analyzing and solving problems in communication networks

802.11g (54 Mbit/s) For the OpenFlow scenario the data network is based on a 802.11s WMN where the mesh interfaces are controlled by OpenFlow. We did not get results for OpenFlow at three and four hops because Iperf did not show the report.

The reason is the default behavior of the OpenFlow switch, it cannot send a packet through the incoming port. However, in a WMN this behavior is valid in a node acting as a relay. This behavior caused a large amount of errors and Iperf did not show a report. As a matter of fact, batman-adv is able to achieve higher throughput at three hops, but at the expense of a higher percentage of packet loss and jitter. Finally, at four hops the throughput of batman-adv is again greater than the throughput of 802.11s, and OpenFlow was not able to pass traffic correctly. Batman-adv has an overall greater throughput, albeit at three hops 802.11s shows a better performance. The control signaling of OpenFlow has a low impact in the performance of the WMN routing protocol.

In the jitter measurements we found that for all hops Batman-adv had a greater jitter than the 802.11s standard, however the maximum value is still below the 20 ms permitted for a good VoIP call. The results for three and four hops are missed because of the same reason as the UDP throughput test. The jitter for one hop was greater than the jitter for two hops, for all the three protocols. This is because at one hop the test was conducted using a PC for the Iperf server and one of the wireless routers as the Iperf client, which have a lower computing power.

With respect to the loss, it was determined that batman-adv has a higher percentage of lost datagrams than 802.11s until the third hop while sending UDP traffic. At the fourth hop 802.11s had too many errors and duplicated packets, so the results reported by Iperf were unreliable. The same behavior was observed for OpenFlow at three and four hops. We made tests sending TCP traffic to measure the maximum throughput allowed. Batman-adv obtained greater throughput than the 802.11s standard and OpenFlow. Since TCP has mechanisms for detecting and correcting errors, the throughput of batman-adv at three hops is greater than the throughput of 802.11s and OpenFlow in this case regarding the results obtained in UDP. Besides, for all the cases batman-adv showed the best performance of the three protocols under study.

4 Conclusion

From the results of this experience we can conclude that both layer 2 routing protocols implemented with Open Source Software for WMNs have advantages and disadvantages. The B.A.T.M.A.N. Advanced protocol achieves higher transmission rates than 802.11s, but at the expense of a higher percentage of datagram loss. However, the throughput of the WMN is not an impediment for services such as videoconference; the current video codecs allow high quality videos with lower bandwidth requirements. Besides, the 802.11s showed a lower jitter than batman-adv, which is better for real-time communications. 802.11s is an IEEE standard, consequently many equipments in the future will support this protocol. Also, 802.11s is more secure because it does not have a SSID field in the frame, so it cannot be easily sniffed. Finally, 802.11s has support for multicast inherently. Regarding OpenFlow, the architecture based on a control network separated from the data network (as proposed by Dely et al. in [4]) shows an acceptable performance compared to the 802.11s standard. The in-band control approach is not recommended for WMN deployments for rural communities, due to its bad performance.

5 References

- [1] I. F. Akyildiz, "A survey on wireless mesh networks," *IEEE Communications Magazine*, vol. 43, no. 9, pp. S23-S30, Sep. 2005.
- [2] IEEE Standard for Information Technology--Telecommunications and information exchange between systems--Local and metropolitan area networks--Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Am, *IEEE Std 802.11s-2011*, pp. 1-372, 2011.
- [3] D. Sether, A. König, and M. Hollick, "Routing performance of Wireless Mesh Networks: A practical evaluation of BATMAN advanced," *Local Computer Networks (LCN)*, 2011 IEEE 36th Conference on, pp. 897-904, 2011.
- [4] P. Dely, A. Kassler, and N. Bayer, "OpenFlow for Wireless Mesh Networks," in *Computer Communications and Networks (ICCCN)*, 2011 Proceedings of 20th International Conference on, 2011, pp. 1-6.
- [5] OpenWrt, <https://openwrt.org/>

X

Se otorga a:

AIDELÉN CHUNG

el presente

Certificado de Expositor

en el CONESCAPAN XXX



El Sr. Ing. Fernando Bouche, Presidente de la Sección Panamá, del IEEE, me ha otorgado el presente certificado de expositor.

F. Chung

ING. FERNANDO BOUCHE
PRESIDENTE
IEEE SECCIÓN PANAMÁ

F. Chung

HEATHER GÓMEZ
PRESIDENTE
IEEE CONESCAPAN XXX

Este documento es propiedad de IEEE y puede ser reproducido en el idioma de todos los países.



**UNIVERSIDAD DE PANAMÁ
VICERRECTORÍA DE INVESTIGACIÓN Y POSTGRADO
DIRECCIÓN DE POSTGRADO**

**VIP-DP-222 14
24 de febrero de 2014**

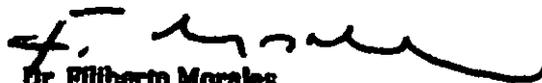
**Prof. Gustavo Díaz
Director de Investigación y Postgrado
Maestría en Ciencias de Ingeniería de Sistemas de Comunicaciones
con énfasis en Redes de Datos
Facultad de Informática, Electrónica y Comunicación
Universidad de Panamá
E. S. D.**

Estimada Señor Director

Atendiendo su solicitud de inscripción del Proyecto de Tesis adjunto copia de la misma con su respectivo código para los trámites pertinentes.

NOMBRE DEL ESTUDIANTE	TÍTULO DE LA TESIS	CÓDIGO
Aidelea Chung	Optimización de Calidad de Servicio para Tráfico Multimedia en redes Mesh inalámbricas en la Universidad de Panamá.	CE-PI-327 17-02 14-06
Yader Mendoza	Propuesta Tecnológica para Mejorar el proceso de Enseñanza Aprendizaje en una escuela del área rural de la República de Panamá, considerando los últimos avances de la Tecnología Web.	CE-PI 327 17-02 14-07
Karina Muñoz	Sistema de almacenamiento con Localización RFID-WIPL	CE-PI 327 17-02 14-08

Atentamente


**Dr. Filiberto Morales
Director de Postgrado**

c.c. Dr. Tomas Diaz, Director de Investigación, VIP

**Ante
2013: "14to de la Consolidación de la Investigación, Innovación y Desarrollo Tecnológico"
CIUDAD UNIVERSITARIA OCTAVIO MÉNDEZ FERRERA
Edificio Universitario, Panamá, República de Panamá
Tel: (507) 523-6328/ 523-6319 Fax: (507) 523-6319
Correo electrónico: postgrado@up.edu.pa**





**UNIVERSIDAD DE PANAMÁ
VICERRECTORÍA DE INVESTIGACIÓN Y
POSTGRADO**

ACTA DE SUSTENTACIÓN DE TESIS

CÓDIGO CE-FI-327 17-02-14-06

**PROGRAMA DE MAESTRÍA EN MAESTRÍA EN CIENCIAS EN INGENIERÍA DE
COMUNICACIONES CON ÉNFASIS EN REDES DE DATOS**

**Título de la Tesis "OPTIMIZACIÓN DE CALIDAD DE SERVICIO PARA TRÁFICO
MULTIMEDIA EN REDES MESH INALÁMBRICAS EN LA UNIVERSIDAD DE
PANAMÁ**

Nombre del Participante AIDELÉN CHUNG

Cédula 8-797 2374

Miembros del Jurado

Calificación otorgada

<u>Gustina Diéguez</u>	<u>98.5</u>
<u>Iván Armuelle</u>	<u>98.5</u>
<u>Alvaro Matute</u>	<u>98.5</u>

Firma de los Miembros del Jurado	Trabajo Escrito	Defensa	Promedio
<u>[Signature]</u>	100	97	98.5
<u>[Signature]</u>	100	97	98.5
<u>[Signature]</u>	100	97	98.5
NOTA FINAL			98.5

Recomendaciones del Jurado _____

Firma del Director de I P & Coordinador del Programa de _____
[Signature]

Firma del Representante de la Vicerrectoría de Investigación y Postgrado (VIP) _____
[Signature]

Firma del Estudiante _____
[Signature]

Fecha. _____
25/02/2014