Entramado
ISSN: 1900-3803
comunicacion.ayc.1@gmail.com
Universidad Libre
Colombia

Enríquez-Lenis, Andrés Eugenio; Agredo-Méndez, Guefry Léider
Análisis de rendimiento en redes IPv6
Entramado, vol. 11, núm. 1, enero-junio, 2015, pp. 214-229
Universidad Libre
Cali, Colombia

Disponible en: http://www.redalyc.org/articulo.oa?id=265440664016

Análisis de rendimiento en redes IPv6*

Andrés Eugenio Enríquez-Lenis
D.E.A. Ingeniería Telemática, Universidad de Vigo, España Estudiante Maestría en Electrónica y Telecomunicaciones, Universidad del Cauca. Instructor, Cisco Networking Academy, Universidad Libre Seccional Cali, Colombia
andresenriquez@unicauca.edu.co

Guefry Léider Agredo-Méndez
Magíster en Electrónica y Telecomunicaciones, Universidad del Cauca. Profesor Titular, Departamento de Telecomunicaciones, Universidad del Cauca, Popayán - Colombia
gagredo@unicauca.edu.co

Resumen

El objetivo principal de esta investigación es determinar el desempeño de diferentes servicios en Internet sobre una arquitectura de red IPv6 por medio de experimentación. El método utilizado fue el empírico, y la metodología seguida, el modelo en cascada, paradigma del ciclo de vida clásico en ingeniería que exige un enfoque sistemático y secuencial. El experimento se desarrolló en el laboratorio de telemática de la Facultad de Ingeniería, en la Universidad Libre en Cali, donde se dispuso de 8 enrutadores, 4 conmutadores y 5 computadores personales. Los instrumentos utilizados fueron el analizador de protocolos Wireshark, analizador de paquetes PRTG, SYSLOG y SNMP server para captura de alarmas y eventos, y herramientas para pruebas en la red: tracert/ traceroute, ping y telnet. Como resultados se observa que el rendimiento de una red IPv6 depende del grado de congestión y del tipo de tráfico que circula en la misma. La investigación permite concluir que aunque se disponga de mecanismos complejos de Calidad de Servicio y Diferenciación de Servicios en Internet, en condiciones de saturación, ninguno de estos mecanismos permite garantizar que los servicios y aplicaciones sensibles, funcionen adecuadamente.

Palabras clave

Calidad de servicio (QoS), DiffServ, IPTV, IPv4, IPv6, OSPFv3

IPv6 network performance analysis

Abstract

The main purpose of this research study is to determine the performance of various online services in an IPv6 network architecture by means of experimentation. An empirical approach was used following the cascade-model methodology, a paradigm of classic useful life in engineering which calls for a systematic, sequential approach. The experiment was conducted at the telematics laboratory at the School of Engineering at Universidad Libre in Cali where eight routers, four network switching devices, and five personal computers were available. A Wireshark protocol tester, a PRTG, SYSLOG, and SNMP server package tester for capturing alarms and events, and network testing tools (i.e. tracert/traceroute, ping, and telnet) were reviewed. The findings show that the performance of an IPv6 network depends on the level of network congestion and the kind of traffic circulating through the network. This research study makes it possible to conclude that, even when complex Internet Service Quality and Service Differentiation mechanisms are in place, none of these mechanisms is able to guarantee proper provision of services or correct operation of sensitive applications under saturation conditions.

Keywords

Quality of Service (QoS), DiffServ, IPTV, IPv4, IPv6, OSPFv3

Recibido: 15/10/2014 **Aceptado:** 09/12/2014

* Este trabajo es producto de una investigación "Análisis del desempeño de iptv sobre una arquitectura de red ipv6/ diffserv /mpls por medio de experimentación y analítica"

http://dx.doi.org/10.18041/entramado.2015v11n1.21118

Cómo citar este artículo: ENRÍQUEZ-LENIS, Andrés Eugenio;AGREDO-MÉNDEZ, Guefry Léider. Análisis de rendimiento en redes IPV6. En: Entramado. Enero - Junio, 2015 vol. 11, no. 1, p. 214-229, http://dx.doi.org/10.18041/entramado.2015v11n1.21118

Análise de desempenho em redes IPv6

Resumo

O principal objetivo dessa pesquisa é determinar o desempenho de diferentes serviços na Internet sobre uma arquitetura de rede IPv6 através da experimentação. O método utilizado foi o empírico, e a metodologia seguida, o modelo em cascata, paradigma do ciclo de vida clássico em engenharia que exige uma abordagem sistemática e sequencial. A experiência foi conduzida no laboratório de telemática da Faculdade de Engenharia, na Universidade Libre em Cali, onde se dispôs de 8 roteadores, 4 comutadores e 5 computadores pessoais. Os instrumentos utilizados foram o analisador de protocolos Wireshark, analisador de pacotes PRTG, SYSLOG e SNMP server para captura de alarmes e eventos, e as ferramentas para testes na rede: tracert/ traceroute, ping e telnet. Como resultado, foi observado que o rendimento de uma rede IPv6 depende do grau de congestionamento e do tipo de tráfego que circula na mesma. A investigação permite concluir que embora se disponha de mecanismos complexos de Qualidade do Serviço e Diferenciação de Serviços de Internet, em condições de saturação nenhum desses mecanismos permite garantir que os serviços e aplicativos sensíveis funcionem adequadamente.

Palabras-chave

Qualidade do serviço (QoS), DiffServ, IPTV, IPv4, IPv6, OSPFv3.

Introducción

Internet es la tecnología que ha revolucionado la forma como se comunica, como se interactúa con otras personas, como se educa, como se divierte, como se compra o como se vende. Ha afectado la sociedad, sin importar su estatus social, creencia religiosa o política. Sus orígenes datan del año 1969 cuando la agencia americana ARPA fundó el proyecto ARPANET, una red experimental conmutada de paquetes.

Durante el transcurso de los años, ARPANET evolucionó y se transformó en la red que hoy se conoce como Internet. En junio de 2014, Internet cumplió cuarenta y cinco años de funcionamiento. Tiempo en que un proyecto que nació de manera experimental, ha madurado y se ha expandido a todas los hemisferios del planeta, y ha impregnado a la sociedad sin importar su estatus o clase social. La "red de redes" como usualmente se denomina, se basa en el protocolo IP, el cual permite identificar cualquier dispositivo en la red. Se basa en el protocolo IPv4 que es la esencia del direccionamiento en Internet, pero que rápidamente quedó obsoleto y ha requerido en las últimas dos décadas redefiniciones y ajustes en distintas partes, para poder seguir en funcionamiento.

Para solucionar los problemas encontrados en IPv4, desde el año 1995 se propuso como su reemplazo el protocolo de la próxima generación IPv6. Sin embargo, dado que se dispone de diversos tipos de aplicaciones en Internet, que requieren recursos diferentes, se debe considerar la implementación de mecanismos de "calidad" a esos servicios.

El objetivo de esta investigación es desarrollar una arquitectura IPv6/DiffServ en un laboratorio real, con el fin de evaluar el comportamiento de la red de núcleo en IPv6 sin soporte alguno a IPv4, mapeada de acuerdo DiffServ. Se tendrán aplicaciones diversas de Internet, tales como IPTV y de mejor esfuerzo (FTP, HTTP), corriendo IPv6, que requieren diferente calidad de servicio.

Una de las implicaciones de esta investigación es encontrar un escenario lo más cercano posible a un entorno real, que permita evaluar el protocolo IPv6 sin el protocolo IPv4. Se estima que entre los años 2025 a 2035, IPv4 dejará de funcionar, por ello, el propósito final que se busca alcanzar con este trabajo es elaborar material investigativo y académico suficiente, que permita construir una línea base de conocimiento, y que la misma sirva para capacitar al capital humano, que tendrá la tarea de preparar las redes de telecomunicaciones futuras.

1. Contextualización del problema

Cuando se analiza el tráfico en una red, se encuentra gran cantidad de tipos de paquetes. Estos se pueden clasificar en forma general, dependiendo del tipo de recursos que demanda de la misma. De esta manera, la voz y la telefonía IP son consideradas como tráfico en tiempo real, y requieren de bajo retardo y una baja pérdida de paquetes en la transmisión extremo-a-extremo. El video, que puede ser en tiempo real o en demanda, requiere de un adecuado ancho de banda y garantía mínima de pérdida de paquetes para asegurar la Calidad de Servicio (QoS, *Quality of Service*). Aplicaciones multimedia tales como IPTV, requieren de ancho de banda considerable y una pérdida de paquetes baja. El resto de paquetes son clasificados como paquetes de datos (SMTP, FTP, HTTP, etc.,) y se tratan como tráfico de mejor esfuerzo o *"best-effort"* (Blake et al., 1998) (Braden, Clark y Shenker, 1994) (Firoui, Le Boudec, Towsley y Zhang, 2002).

La IETF (*Internet Engineering Task Force*) define QoS como "un conjunto de requerimientos de servicio a ser conseguidos por la red mientras se transporta un flujo". Un flujo se define como una corriente de paquetes IP desde un origen a un destino (unicast o multicast) con un nivel de QoS asociado. La IETF propone arquitecturas y protocolos para la comunidad de Internet, con el fin de solventar el problema de transportar distinto tipo de tráfico en el núcleo de la red, y darle a cada flujo, las características de QoS que requiere. A través de los RFC[1] propone estándares y arquitecturas para el diseño de Internet.

Se pueden destacar el RFC 1349 *"Type of Service in the Internet Protocol Suite"* (Almquist, 1992), el cual define el tipo de servicio en la suite de protocolos de Internet[2]. Es actualizado por el RFC 2474 *"Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers"* (Nichols, Blake, Baker y Black, 1998) donde se define el campo DS en las cabeceras IPv4 e IPv6. A su vez, este RFC se complementó con el RFC 3168 *"The Addition of Explicit Congestion Notification-ECN to IP"* (Ramakrishnan, Floyd y Black, 2001) y con el RFC 3260 *"New Terminology and Clarifications for DiffServ"* (Grossman, 2002).

La Tabla 1 muestra a modo de resumen, los necesidades específicas de QoS para una clasificación muy general de aplicaciones en Internet. Vemos en esta tabla cómo las aplicaciones interactivas de voz y video son más sensibles al retardo, diferencia de retardo y pérdida de paquetes; el video en tiempo real requiere alto ancho de banda, mientras el video en demanda es sensible al *jitter* y la pérdida de paquetes; las transacciones interactivas son medianamente sensibles al retardo. Las aplicaciones de mejor esfuerzo no se consideran sensibles a estas métricas. (Ver Tabla 1).

Con estas necesidades de QoS por aplicación, se hace urgente revisar la integración de IPv6 con los protocolos de enrutamiento y aplicación de políticas de QoS.

1.1. Protocolo IPv6

El Protocolo de Internet versión 6 (IPv6), usualmente llamado Protocolo de Próxima Generación (*IP Next Generation Protocol*) o IPng, es un protocolo de la capa de red, recomendado inicialmente en el RFC 1752 *"The Recommendation for IP Next Generation Protocol"* en el año 1995 (Bradner y Mankin, 1995), fue especificado en el RFC 1883 (Deering,1995) , y posteriormente redefinido en el RFC 2460 (Deering y Hinden, 1998). Varias actualizaciones y ampliaciones posteriores han ocurrido hasta la fecha. IPv6 es la ampliación de su antecesor, el Protocolo de Internet versión 4 (IPv4) que se encuentra en uso desde los años 1980 (DARPA, 1981).

Las características principales de IPv6 son: (Deering y Hinden, 1998).

- **Capacidades de direccionamiento extendidas:** IPv6 incrementa el tamaño de las direcciones IP, pasando de 32 a 128 bits, y permite nuevas formas de autoconfiguración de nodos. Define tres tipos de direcciones IP: anycast, multicast y unicast.
- **Simplificación del formato de cabecera:** algunos campos de la cabecera IPv4 han sido eliminados o vueltos opcionales, para reducir el costo de procesamiento en el manejo de paquetes y así limitar el tamaño de la cabecera IPv6.
- **Capacidades de privacidad y autenticación:** se dispone de extensiones para soportar la autenticación, la integridad de datos y la confiabilidad.
- **Capacidades de marcación de flujos:** esta capacidad es adicionada para habilitar la marcación de paquetes pertenecientes a un flujo de tráfico particular.
- **Soporte mejorado a extensiones y opciones:** se modifican las opciones de la cabecera IP para permitir eficiencia en el reenvío, menos restricciones en el límite de la longitud de las opciones, y gran flexibilidad para introducir nuevas opciones futuras.

Tabla 1.
Sensitividad a métricas de QoS

Ejemplos de aplicaciones	REQUERIMIENTO DE QoS			
	Retardo (*delay*)	Ancho de banda	Diferencia de retardo (*jitter*)	Pérdida de paquetes
Voz en tiempo real (dos vías)	BAJO	BAJO	BAJO	BAJO
Video en tiempo real	BAJO	ALTO	BAJO	BAJO
Video en demanda	ALTO	ALTO	BAJO	BAJO
Aplicaciones interactivas misión crítica	MEDIO	VARIABLE TÍPICAMENTE MEDIO	MEDIO	MEDIO
Tráfico de mejor esfuerzo	ALTO	VARIABLE TÍPICAMENTE ALTO	ALTO	ALTO

Fuente: Adaptado de "IP Telephony Self-Study Cisco DQOS Exam Certification Guide" (Odom y Cavanaugh, 2004)

La Figura 1 muestra el formato de la cabecera de IPv6 (40 octetos):

En la actualidad, todas las redes IP a desplegar deben estar soportadas por IPv6 debido al agotamiento de direcciones públicas IPv4 desde inicios del año 2011[3], aspecto que hace evidente la necesidad de contar con trabajos de investigación experimental con este protocolo. La arquitectura presentada en próximos apartados, se basa en IPv6 explícitamente, deshabilitando IPv4, con el propósito de conocer su funcionamiento en forma exclusiva.

1.2. Servicios Integrados y Servicios Diferenciados

Los Servicios Integrados *"IntServ"* especifican una arquitectura diseñada para el envío de tráfico en tiempo real. Se basa en la premisa de predecir y garantizar el servicio antes que el mismo sea enviado en la red (Braden, Clark y Shenker, 1994). Esto implica que se debe realizar una "reservación de recursos" y "control de admisión" a la red. Significa, que las aplicaciones deben solicitar a la red una reserva de recursos extremo-a-extremo, y que si ésta se da, la red en este segmento deberá controlar qué servicios admite o no, para evitar que la aplicación que ha reservado unos recursos previamente, sufra deterioro en el QoS. *IntServ* se utiliza en la práctica en la reservación de recursos en la nube del ISP, cuando éste realiza Ingeniería de Tráfico (TE) en MPLS (*Multi-Protocol Label Switching*). En otros escenarios, *IntServ* no es empleado usualmente, debido a que se debe tener control de toda la red donde se desee reservar recursos.

Los servicios diferenciados *"DiffServ"* definen una arquitectura para implementar servicios escalables en Internet. Un servicio define alguna característica significante en la transmisión de paquetes en una dirección, a través de una o más rutas en la red. Estas características pueden ser especificadas en términos cuantitativos o estadísticos de retardo, diferencia de retardo (*jitter*), y cantidad de datos transmitidos (*throughput*), con o sin pérdida, y pueden ser detalladas en términos de alguna prioridad relativa para acceder a los recursos de la red.

La arquitectura está compuesta por elementos funcionales implementados en los nodos de la red, incluyendo formas rápidas de reenvío de paquetes al próximo salto (EF-PHB Expedited Forwarding Per-Hop Behavior), funciones de clasificación de paquetes y funciones de acondicionamiento de tráfico tales como medición, marcado, conformación y políticas.

Esta arquitectura logra escalabilidad implementando una clasificación compleja y funciones de acondicionamiento únicamente en los nodos de borde de la red, y aplicando comportamiento por salto para agregación de tráfico. Usualmente, este acondicionamiento de tráfico se realiza en los enrutadores del borde del ISP antes de entrar a su núcleo, empleando para ello el campo "clase de tráfico" de la cabecera IPv6 llamada DSFIELD o DiffServ (Nichols, Blake, Baker y Black, 1998).

En la arquitectura propuesta, se empleará DiffServ para realizar clasificación, marcación y envío de paquetes a la red del núcleo, que permitirá en conjunto, aplicar QoS a los diferentes servicios.

1.3. OSPFv3

Open Shortest Path First (OSPF) es un protocolo de enrutamiento desarrollado para redes IP por el grupo de trabajo de Internet, IETF. Este protocolo fue diseñado inicialmente en 1988 en el RFC 1131 como un protocolo IGP (*Interior Gateway Protocol*) basado en la ruta más corta, para lo cual emplea el algoritmo SPF (*Shortest Path First*), que se basa en el algoritmo de Dijkstra. Su desarrollo se debió principalmente a los problemas encontrados en el empleo de RIP (*Routing Information Protocol*) en redes de gran tamaño y heterogéneas.

OSPFv3 es especificado por la IETF en el RFC 2740 (Coltun, Ferguson y Moy, 1999), y RFC 5340 (Coltun, Ferguson, John y Lindem , 2008). El mecanismo fundamental de OSPF consiste en la selección de un DR (*Designated Router*) y un BDR (*Backup Designated Router*), un área de soporte para el enrutamiento y la búsqueda y selección de la ruta más corta. Para ello emplea el algoritmo SPF, una base de datos de topología y una tabla de enrutamiento. La métrica usada es el costo, el cual se encuentra asociado con la velocidad de la interfaz. Los anuncios y actualizaciones de enrutamiento se efectúan a través de los LSA (*Link-State Advertisement*).

Figura 1. Cabecera IPv6

Fuente: Adaptado de RFC 2460 (Deering y Hinden , 1998)

La arquitectura de red experimental se basa explícitamente en el enrutamiento OSPFv3. Este protocolo es requerido para la implementación de protocolos avanzados que permiten el soporte a *DiffServ* escalable, tales como MPLS e Ingeniería de Tráfico.

1.4. IPTV

Existen cuatro tipos de servicios de video que son comúnmente enviados sobre redes IP: IPTV (Internet Protocol Television), IPVoD (Internet Protocol Video on Demand), Internet TV (Internet Television), e Internet video. La ITU define IPTV como "los servicios multimedia tales como la televisión, video, audio, texto, gráficos y envío de datos sobre una red gestionable basada en IP para proveer el nivel requerido de Calidad de Servicio (QoS), Calidad de Experiencia (QoE), seguridad, interactividad y confiabilidad" (Lee, 2007) (*International Telecommunication Union* (ITU), 2008), (*International Telecommunication Union* (ITU), 2009). En un informe de la ITU-D se muestra cómo en los últimos años la demanda creciente de servicios de banda ancha se ha aumentado y servicios IPTV[4] tales como televisión, video en demanda (VoD) de baja y alta calidad, y música en línea, son los servicios que más demandan los usuarios de Internet.

La arquitectura experimental ha sido probada con diferentes servicios, entre ellos IPTV, para revisar su comportamiento y efecto en el desempeño de la red. Este tipo de servicios son los que más recursos demandan de la red y deben competir con otros tipos de aplicaciones.

2. Trabajos relacionados

Se han encontrado varios trabajos previos, todos realizados en simulador por software, de los cuales se pueden citar los siguientes desarrollados en Network Simulator 2 (NS-2):

El artículo *"Integration of Protocols FHMIPv6/MPLS in Hybrid Networks"* (Ortiz, Perea, Santibáñez y Ortiz, 2011) presenta el resultado de la integración de los protocolos FHMIPv6/MPLS para proveer QoS en escenarios híbridos, cuando ocurre un handover. Los autores afirman que durante el handover, las métricas retardo, diferencia de retardo y volumen de trabajo, así como el nivel de calidad por defecto fueron mantenidos en el rendimiento, esperado. El resultado permitió identificar cuáles protocolos integrados fueron los mas apropiados para asegurar QoS en redes totalmente IPv6/MPLS. Una arquitectura para nueva generación de redes híbridas es propuesta. En resumen, la unión entre QoS y los protocolos de movilidad mencionados, son una excelente opción para proveer QoS en redes móviles y, especialmente, en redes híbridas móviles. Por otra parte, se puede decir que aunque no hay definido un estándar completo para las redes de próxima generación 4G, una arquitectura FHMIPv6/MPLS será crítica en las rede móviles de nueva generación, compatible con los estándares propuestos (WiMAX, advanced LTE/SAE, LTE/IMT, WiMAX/IMT). El tráfico empleado en la simulación fue CBR y FTP.

En "*AHRA: A routing agent in order to support the Hierarchical Mobile IPv6 protocol with Fast-Handover over mobile Adhoc network scenarios*", los autores proponen el desarrollo de un esquema de enrutamiento que permita soportar el protocolo Fast-Hierarchical Mobile IPv6 (F-HMIPv6) sobre redes ad-hoc móviles. Se describe un sistema que es el resultado de unir trabajo de movilidad de agentes que efectúan tareas de registro y descubrimiento de rutas, así como el protocolo de enrutamiento "NOAH", que emplea información de los agentes para enviar datos. El esquema AHRA (Ad-hoc Routing Agent) ha sido implementado y probado en NS-2 con buenos resultados. Esto ha involucrado el desarrollo de un nuevo agente de movilidad (FHAMIPv6) asignado a nodos intermedios para reenvío y procesamiento de mensajes de registro. También ejecuta modificaciones sobre el protocolo original NOAH, dándole capacidad para el reenvío de datos (Ortiz, González, Perea y López, 2011).

El artículo "*Integration of HMIPv6/MPLS*", presenta el efecto de la integración del protocolo *Hierarchical Mobile* IPv6 (HMIPv6) con el protocolo *Multiprotocol Label Switching* (MPLS) con respecto a QoS. La idea de esta integración parte del concepto "todo IPv6/MPLS" designadas para las redes de nueva generación 4G. El propósito de este trabajo es analizar los efectos en la QoS en la sección UDP. Se analiza el retardo, diferencia de retardo y el volumen de trabajo para proveer QoS de extremo-a-extremo. El protocolo RSVP es empleado como protocolo de señalización. Se demuestra con este trabajo que la integración de HMIPv6/MPLS es una buena opción para las redes de nueva generación (Ortiz, Perea, Ortiz y Santibañez, 2011).

Igualmente, se han encontrado un par de trabajos realizados con OPNET (*Optimized Network Engineering Tool*): En Aziz y Saiful (2011), Aziz, Saiful, Khan y Popescu (2012), Tesis de Máster y artículo, se presenta un estudio de rendimiento de QoS para aplicaciones en tiempo real tales como voz y videoconferencia sobre Diffserv, implementadas con y sin ingeniería de tráfico MPLS sobre redes IPv4 e IPv6. El trabajo muestra los resultados obtenidos y un esquema general de su implementación. El escenario de prueba se realiza con dos LAN conectadas, cada una con un enlace WAN a una nube de enrutadores. El tráfico generado pasará de una LAN a la otra, a través de la nube, por una o varias rutas. Sin embargo, la arquitectura de red no podía considerarse como una arquitectura de núcleo de un ISP genérica. Es un

buen trabajo realizado en simulación. Ninguna de las pruebas se efectuaron en un ambiente con equipos reales.

El trabajo que se presenta se diferencia de los anteriores, al emplear la experimentación con equipos reales en el núcleo de red IPv6/*DiffServ*, y con aplicaciones reales de Internet. Para efectos de medir el rendimiento de la red, identificar problemáticas en el despiegue, y evitar problemas de malas configuraciones o interpretaciones erróneas, el protocolo IPV4 se deshabilita por completo. Esto permitirá obtener conclusiones acerca de las variables retardo, diferencia de retardo, ancho de banda y pérdida de paquetes, cuando se compite por recursos, y se tienen aplicaciones IPTV que demandan un determinado QoS. El escenario diseñado e implementado es una aproximación al núcleo de red real que un ISP podría tener. No se tendrán en cuenta los efectos de handover, y se centrará en los efectos que se presentan en la red cuando se requiere un ajuste granular de QoS con servicios que demandan gran cantidad de recursos, tales como IPTV. El análisis se centrará en el desempeño del núcleo IPv6/*DiffServ* en condiciones de congestión y no congestión, para aplicaciones IPTV seleccionadas.

3. Arquitectura del laboratorio

La metodología empleada para la realización de esta investigación se soporta en el desarrollo de experimentos en un laboratorio real compuesto por enrutadores y conmutadores capa 2, y de equipos de cómputo de uso personal. La metodología seguida es el modelo en cascada, paradigma del ciclo de vida clásico en ingeniería que exige un enfoque sistemático y secuencial, definiendo diversas etapas para el desarrollo del sistema, las cuales han sido adaptadas para el avance de esta investigación. Las fases que define el modelo son ingeniería y análisis del sistema, diseño, codificación, pruebas y mantenimiento.

La arquitectura de núcleo diseñada se basó en documentación encontrada en Cisco Systems[5], que emula la red de un ISP. El núcleo está conformado por cuatro (4) enrutadores Cisco ISR 2811 conectados entre sí con interfases V.35 sincrónicas (estas interfases componen los enlaces WAN); cada enrutador del núcleo se conecta a su vez con un un enrutador Cisco ISR 2901 que emula el equipo terminal en la oficina del cliente. El cliente podría ser típicamente una empresa del tipo SOHO (*Small Office Home Office*). El objetivo de disponer de interfases seriales V.35 como conexiones WAN en el interior de la nube IPv6/OSPF, se debe a la facilidad de poder congestionar de "forma sencilla" las interfases de salida, poder aplicar mecanismos variados de QoS.

La red corre en forma nativa el protocolo IPv6, y para evitar configuraciones innecesarias o ambigüedades con IPv4, se deshabilita el enrutamiento IPv4 en forma explícita en los enrutadores. El núcleo de la red está configurado con el protocolo de enrutamiento OSPFv3 en una única área (área 0). Las redes de los clientes se conectan con rutas estáticas y se deshabilitar los anuncios de OSPF en sus interfases. Las redes terminales de los clientes se configuran como rutas por defecto, para permitir el enrutamiento a todos los puntos de la red. La Figura 2 muestra la arquitectura de red detallada con el direccionamiento IPv6 de las redes de clientes.

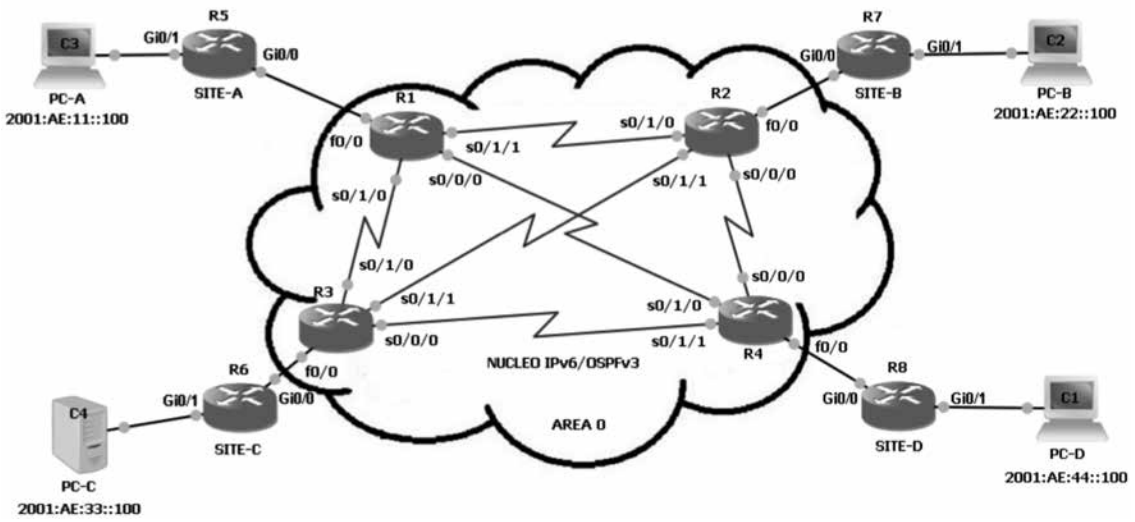


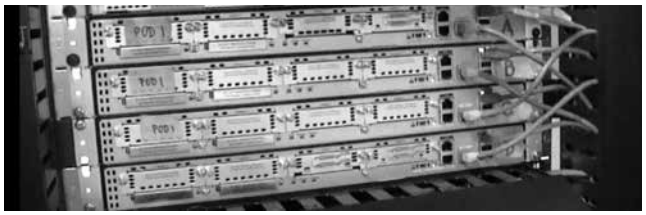
Figura 2. Topología de red y direccionamiento IPv6 de clientes.
Fuente: Los autores.

3.1. Hardware empleado

Las características resumidas de los enrutadores Cisco ISR 2811 son: IOS Advance IP Service: c2800nm-advipservicesk9-mz.124-24.T, 512 MB RAM, 32 MB memoria flash, dos interfases FastEthernet 10/100 UTP (LAN) y cuatro interfases seriales V.35 sincrónicas (WAN). Los enrutadores Cisco ISR 2901 tienen la siguiente configuración: IOS Advance IP Service: c2900-universalk9-mz.SPA.151-1, 512 MB RAM, 256 MB memoria flash; dos interfases GigabitEthernet 10/100/1000 UTP (LAN).

Las Fotografías 1 a la 3 muestran una vista general del laboratorio utilizado, así como los equipos empleados para el núcleo de la red (R1 a R4) y los enrutadores de acceso que conectan las redes de clientes (R5 a R8). Los equipos de cliente están conformados por PCs conectados a los enrutadores terminales. La red de servidores está integrada por un servidor WEB, un servidor de video en demada, un servidor FTP, un servidor VoIP y un servidor de video en vivo. En los equipos de cliente PC-A y PC-B se ha implementado un cliente de FTP, un navegador (HTTP y FTP) y Wireshark como analizador de protocolos. El cliente PC-D se emplea como gestor de la red y tiene implementado adicionalmente un administrador de SNMP, un analizador de tráfico SNMP/Netflow y un servidor SYSLOG.

Fotografía 1. Vista general del laboratorio.
Fuente: Los autores.



Fotografía 2. Vista del laboratorio - Enrutadores de acceso (clientes).
Fuente: Los autores.

Fotografía 3. Vista del laboratorio - Enrutadores del núcleo.
Fuente: Los autores.

3.2. Tablas de enrutamiento en IPv6

Las Tablas 2 y 3 (Ver pág. 221) muestran el direccionamiento IPv6, diseñado específicamente para esta arquitectura.

3.3. Detalle de la implementación de IPv6

La Tabla 4 (Ver pág. 222) presenta la configuración básica del enrutador R1.

Esta configuración habilita explícitamente IPv6 en todas las interfaces y desactiva IPv4. Los comandos IPv6 *unicast-routing* e IPv6 cef, se requieren para el enrutamiento OSPF. La dirección FE80::1 se emplea como dirección de enlace local, para facilitar los anuncios entre vecinos; las direcciones con el prefijo 2001:AE::/40 se utilizan como direcciones unicast públicas en Internet. Las interfaces seriales se configuraron con una velocidad de reloj de 512Kbps; el parámetro *bandwidth* define el ancho de banda que se usará para configuración de QoS y para las actualizaciones de enrutamiento OSPF. El parámetro *no fair-queue* configura la interfaz de salida como FIFO explícitamente. Todas las interfaces se configuraron dentro del área 0 en el proceso 1 de OSPF.

La Tabla 5 (ver pág.222) muestra la configuración del enrutamiento estático IPv6 y ajustes a OSPF en R1.

El comando IPv6 route 2001:AE:11::/64 FastEthernet0/0 define una ruta estática hacia la red LAN de SITE-A. En OSPFv3, el comando passive-interface desactiva los anuncios LSAs hacia las interfaces FastEthernet para mejorar el ancho de banda, evitar inundaciones de enrutamiento, y prevenir ataques de denegación de servicio. El comando *auto-cost reference-bandwidth 1000* coloca una nueva referencia del costo para el cálculo de la ruta más corta en OSPF (1 Gigabit/seg.). Esto es necesario, ya que OSPF fue definido inicialmente para interfaces menores o iguales a 100Mbps.

La configuracion anterior conforma la red del núcleo con OSPFv3 e IPv6 y conecta las redes externas con rutas está-

Tabla 2.
Direccionamiento detallado IPv6 del núcleo

Equipo	Interfaz	Dirección IPv6	Conecta con
R1	Fa0/0	2001:AE:1::1/64 ; FE80::1 link-local	Fa0/0 SITE-A
	S0/1/1 (DCE)	2001:AE:12::1/64; FE80::1 link-local	S0/1/0 R2
	S0/1/0 (DCE)	2001:AE:13::1/64; FE80::1 link-local	S0/1/0 R3
	S0/0/0 (DCE)	2001:AE:14::1/64; FE80::1 link-local	S0/1/0 R4
R2	Fa0/0	2001:AE:2::1/64; FE80::2 link-local	Fa0/0 SITE-B
	S0/1/0	2001:AE:12::2/64; FE80::2 link-local	S0/1/1 R1
	S0/1/1	2001:AE:23::2/64; FE80::2 link-local	S0/1/1 R3
	S0/0/0 (DCE)	2001:AE:24::2/64; FE80::2 link-local	S0/0/0 R4
R3	Fa0/0	2001:AE:3::1/64; FE80::3 link-local	Fa0/0 SITE-C
	S0/0/0 (DCE)	2001:AE:34::3/64; FE80::3 link-local	S0/1/1 R4
	S0/1/1 (DCE)	2001:AE:23::3/64; FE80::3 link-local	S0/1/1 R2
	S0/1/0	2001:AE:13::3/64; FE80::3 link-local	S0/1/0 R1
R4	Fa0/0	2001:AE:4::1/64; FE80::4 link-local	Fa0/0 SITE-D
	S0/0/0	2001:AE:24::4/64; FE80::4 link-local	S0/0/0 R2
	S0/1/1	2001:AE:34::4/64; FE80::4 link-local	S0/0/0 R3
	S0/1/0	2001:AE:14::4/64; FE80::4 link-local	S0/0/0 R1

Fuente: Los autores

Tabla 3.
Direccionamiento detallado IPv6 de las redes de acceso

Equipo	Interfaz	Dirección IPv6	Conecta con	Puerta Enlace
SITE-A	Gi0/0	2001:AE:1::2/64 ; FE80::1:11 link-local	Fa0/0 R1	2001:AE:1::1
	Gi0/1	2001:AE:11::1/64 ; FE80::1:11 link-local	PC-A	N.A.
SITE-B	Gi0/0	2001:AE:2::2/64, FE80::2:22 link-local	Fa0/0 R2	2001:AE:2::1
	Gi0/1	2001:AE:22::1/64, FE80::2:22 link-local	PC-B	N.A.
SITE-C	Gi0/0	2001:AE:3::2/64, FE80::3:33 link-local	Fa0/0 R3	2001:AE:3::1
	Gi0/1	2001:AE:33::1/64, FE80::3:33 link-local	PC-C (SERVER)	N.A.
SITE-D	Gi0/0	2001:AE:4::2/64, FE80::4:44 link-local	Fa0/0 R4	2001:AE:4::1
	Gi0/1	2001:AE:44::1/64, FE80::4:44 link-local	PC-D	N.A.
PC-A	NIC	2001:AE:11::100/64	Gi0/1 SITE-A	2001:AE:11::1
PC-B	NIC	2001:AE:22::100/64	Gi0/1 SITE-B	2001:AE:22::1
Server	NIC	2001:AE:33::100/64	Gi0/1 SITE-C	2001:AE:33::1
PC-D	NIC	2001:AE:44::100/64	Gi0/1 SITE-D	2001:AE:44::1

Fuente: Los autores

ticas. Como se puede revisar en las pruebas efectuadas, se logra conectividad extremo-a-extremo en todos los puntos de la red, aunque no se ha efectuado ningún ajuste o configuración para el soporte de QoS.

En este escenario, todo el tráfico se comporta como de mejor esfuerzo, y compite de igual forma entre sí. Configuraciones similares se aplican a los diferentes enrutadores del núcleo de la red (R2, R3 y R4).

La Tabla 6 (ver pág. 222) presenta una variación a la arquitectura inicial, cambiando el encolamiento de las interfaces de salida de los enrutadores del núcleo. El comando *fair-queue* configura WFQ (*Weighted Fair Queuing*). Este algoritmo de encolamiento permite compartir el ancho de banda de forma justa entre los flujos, reduciendo tiempo de respuesta para flujos interactivos, programándolos al inicio de la cola. Previene que flujos con alto volumen de datos monopolicen una interfaz, tal como lo hace el tráfico FTP.

Tabla 4.
Configuración básica en R1

```
ipv6 unicast-routing
ipv6 cef
!
interface FastEthernet0/0
 description Conexión SITE-A
 no ip address
 ipv6 address FE80::1 link-local
 ipv6 address 2001:AE:1::1/64
 ipv6 ospf 1 area 0
!
interface Serial0/0/0
 description WAN --> R4
 bandwidth 512
 no ip address
 ipv6 address FE80::1 link-local
 ipv6 address 2001:AE:14::1/64
 ipv6 ospf 1 area 0
 no fair-queue
 clock rate 512000
interface Serial0/1/0
 description WAN --> R3
 bandwidth 521
 no ip address
 ipv6 address FE80::1 link-local
 ipv6 address 2001:AE:13::1/64
 ipv6 ospf 1 area 0
 no fair-queue
 clock rate 512000
!
interface Serial0/1/1
 description WAN --> R2
 bandwidth 512
 no ip address
 ipv6 address FE80::1 link-local
 ipv6 address 2001:AE:12::1/64
 ipv6 ospf 1 area 0
 no fair-queue
 clock rate 512000
```

Fuente: Los autores

Tabla 5.
Configuración de enrutamiento en R1

```
!
ipv6 route 2001:AE:11::/64 FastEthernet0/0
!
ipv6 router ospf 1
passive-interface fa0/0
passive-interface fa0/1
auto-cost reference-bandwidth 1000
!
```

Fuente: Los autores

Tabla 6.
Configuración WFQ

```
! APLICACION WFQ
interface Serial0/1/0
  fair-queue
interface Serial0/1/1
  fair-queue
interface Serial0/0/0
  fair-queue
!
```

Fuente: Los autores

La Tabla 7 revela la configuración genérica de un enrutador terminal de cliente. En este caso, se muestra la configuración de SITE-A, que se conecta a la nube del ISP a través de R1. En esta configuración se observa que se tiene una nueva dirección de link-local (*FE80::1:11*) exclusiva para este enrutador. Al ser un enrutador terminal se configura una ruta por defecto (*ipv6 route ::/0 2001:AE:1::1*), que permite enrutar todo el tráfico saliente a la nube del ISP.

3.4. Configuración de DiffServ

Para configurar diferenciación de servicios se utilizará como base la Tabla 8 (ver pág. 223). La configuración presentada se aplica a los enrutadores de acceso a la red (ej. SITE-A).

En este escenario se emplea una de muchas técnicas descritas en Cisco System (Cisco Systems, 2005). En particular, se usan los *class-map* para definir las clases de tráfico. La sección *police-map QoS-Policy* permite crear la política a aplicar dependiendo del tipo de tráfico. El comando *random-detect dscp-based* se usa para aplicar el algoritmo RED (*Random Early Detection*) al tráfico de video interactivo, que permite un descarte selectivo de paquetes en el momento de congestión. La sección *policy-map Marcacion* permite efectuar la marcación del campo DS a los paquetes IPv6. El apartado *policy-map Shaping-Police* crea la política de "suavizado" para tráfico *best-effort* y comprimir la cabecera TCP. Una vez definidas las directivas de QoS, se aplican las política de tráfico a las interfases, lo cual se muestra en la Tabla 9. (Ver pág. 223).

Tabla 7.
Configuración básica en SITE-A

```
hostname SITE-A
 !
 ipv6 unicast-routing
 ipv6 cef
 !
interface Serial0/0/0
 no ip address
 no fair-queue
 shutdown
!
interface Serial0/0/1
 no ip address
 no fair-queue
 shutdown
!
ipv6 route ::/0 2001:AE:1::1
!
interface GigabitEthernet0/1
 description CONEXION LAN
 no ip address
 duplex auto
 speed auto
 ipv6 address FE80::1:11 link-local
 ipv6 address 2001:AE:11::1/64
no shutdown
!
interface GigabitEthernet0/0
 description CONEXION WAN
 no ip address
 duplex auto
 speed auto
 ipv6 address FE80::1:11 link-local
 ipv6 address 2001:AE:1::2/64
 no shutdown
```

Fuente: Los autores

Tabla 8.
Configuración de DiffServ

```
! Clasificación                    ! Creación de políticas          ! Marcación
class-map match-any CRITICAL       policy-map QoS-Policy            policy-map MARCACION
 match ip precedence 7             class CRITICAL                    class CRITICAL
class-map match-any VoIP             bandwidth percent 25            set ip dscp 56
 match ip precedence 5             class VoIP                        class VoIP
class-map match-any VideoInteractivo  priority percent 10            set ip dscp EF
 match ip precedence 4             class VideoInteractivo            class VideoInteractivo
class-map match-any TraficoControl   bandwidth percent 30            set ip dscp AF43
  match ip precedence 3             random-detect dscp-based        class TraficoControl
class-map match-any BD               class TraficoControl             set ip dscp AF31
  match ip precedence 2              bandwidth percent 10            class BD
class-map match-any WEB-FTP          class BD                         set ip dscp AF21
 match protocol http                  bandwidth percent 10            class WEB-FTP
 match protocol ftp                  class WEB-FTP                    set ip dscp AF12
 match ip precedence 1                bandwidth percent 10           class class-default
class-map match-any class-default    class class-default               set ip dscp 00
 match ip precedence 0                fair-queue
!                                    !
```

Fuente: Los autores

Tabla 9. Configuración de DiffServ (continuación)

```
! Política de trafico saliente
policy-map Shaping-Police
 class class-default
  shape peak percent 10
  compress header ip tcp
!
! Aplicación qos a las interfases
interface FastEthernet0/0
  service-policy input marcación
interface Serial0/1/0
   service-policy output QoS-Policy
interface Serial0/1/1
   service-policy output QoS-Policy
interface Serial0/0/0
   service-policy output Shaping-Police
```

Fuente: Los autores

3.5. Servicios implementados

A continuación se describen los servicios implementados, sin dar detalles técnicos. Si se desea información al respecto pueden consultar con el autor principal a través de su email.

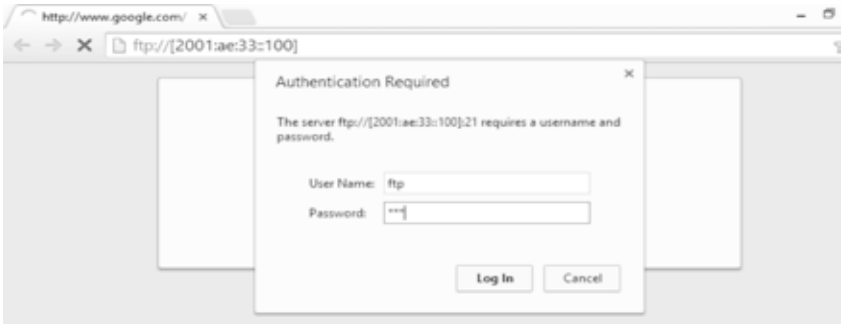
La Fotografía 4 muestra la página principal del servidor WEB. Esta se implementó en Apache 2, corriendo en Linux Debian núcleo 2.4.

En la Fotografía 5a se muestra el servidor FTP implementado y el acceso a través de Google Chrome. La Fotografía 5b revela el listado de archivos del servidor FTP. (Ver pág 224)

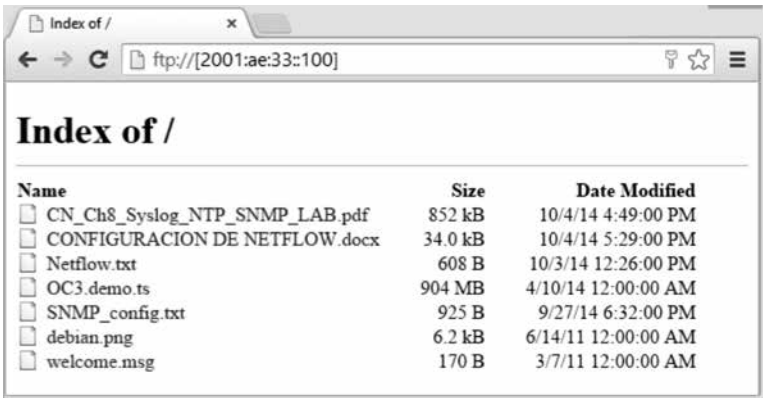
La Fotografía 6 muestra la página empleada para video en demanda. La página se elaboró en Camtasia Studio 8, y en la configuración de salida, se generaron videos en 480p y 720p. en condiciones de no congestión, la página carga sin retrasos ni congelamientos. (Ver pág 224)

Las fotografías 7a y 7b (Ver pág 224)muestran el momento en que ocurre una congestión en la red. El tiempo de respuesta del equipo remoto tarda demasiado *(time out)*; el video se detiene y empieza a mostrar el *buffer* intermedio. En general, todos los servicios se ven afectados por la congestión de la red.

Fotografía 4. Página inicio servidor WEB
Fuente: Los autores.



Fotografía 5a. Acceso desde un cliente FTP
Fuente: Los autores



Fotografía 5b. Listado obtenido del servidor FTP
Fuente: Los autores

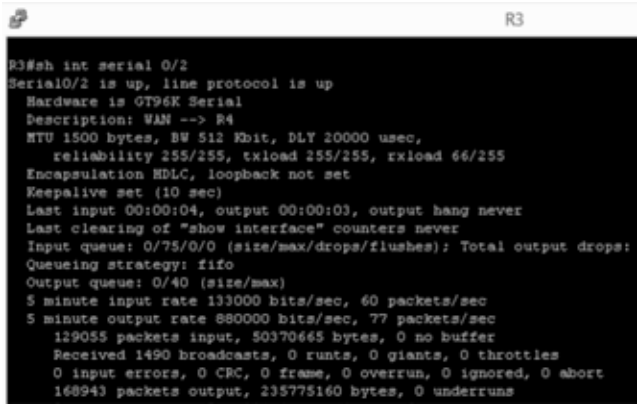
Fotografía 6. Video en demanda
Fuente: Los autores

Fotografía 7a. Indicios de congestión en la red
Fuente: Los autores

Fotografía 7b. Video en demanda indicando congestión en la red
Fuente: Los autores

La Fotografía 8 muestra el estado de la interfaz serial 0/2 en R3, la cual se emplea como ruta desde PC-C hasta PC-D. OSPFv3 utiliza como métrica el costo de la interfaz, sin tener en cuenta otras métricas, tales como la congestión o el retardo.

Se puede observar en el estado de esta interfaz, que la carga de transmisión (*txload*) es de 255/255, mientras que la carga de recepción (*rxload)* es 66/255. El valor máximo a utilizar debería ser 75% del ancho de banda disponible (384 kbps equivalente a 191/255).



Fotografía 8. Estadísticas de la una interfaz congestionada
Fuente: Los autores

4. Resultados y análisis

A través de diferentes pruebas (ping y traceroute), se logró comprobar la conectividad extremo-a-extremo en toda la red. También se pudo verificar que al emplear en conjunto con OSPFv3, rutas estáticas y rutas por defecto, se logró tener los menores tiempos de retardo en la red de extremo-a-extremo (<=10 ms). Se inyectaron en forma simultánea paquetes desde los cuatro extremos de las redes de clientes, inicialmente como ICMPv6 de 64, 500, 1000 y 1500 bytes. También se inyectaron paquetes de 5000 bytes y posteriormemte de 50000 y 55000 bytes. Para estos tres últimos casos, debido al tamaño de los mismos, se supondría que los paquetes serían rechazados por la red. Esto no fue así.

Debido a nuevas caracteristicas de IPv6, los paquetes de más de 1500 bytes se fragmentan en paquetes de máximo 1500 bytes (en el origen) y se re-ensamblan en el destino. Así, un paquete de 50000 bytes es equivalente a 34 paquetes de 1448 y uno de 144 bytes.

La Figura 3 muestra la captura de tráfico ICMPv6 en el instante de tiempo 1700 a 1950 segundos empleando el analizador de paquetes Wireshark versión 1.10.3. Se ha filtrado el tráfico, dejando únicamente ICMPv6. La interfaz de captura es la LAN del SITE-C.

Se observa que los protocolos empleados son IPv6 (Internet Protocol Version 6) e ICMPv6 (Internet Control Message Protocol v6). La dirección de origen es PC-B [2001:ae:22:0:997b:17a4:a468:b8be] y la de destino es PC-C [2001:ae:33::100]. Por defecto, Wireshark emplea la dirección IPv6 definida por autoconfiguración y no la dirección configurada manualmente. El campo de fragmentación (Fragmentation Header), seguido de los fragmentos, indica que se tienen 35 fragmentos IPv6 en los que se ha dividido el paquete original. Se puede observar en el campo Data el tamaño del paquete original (50000 bytes).

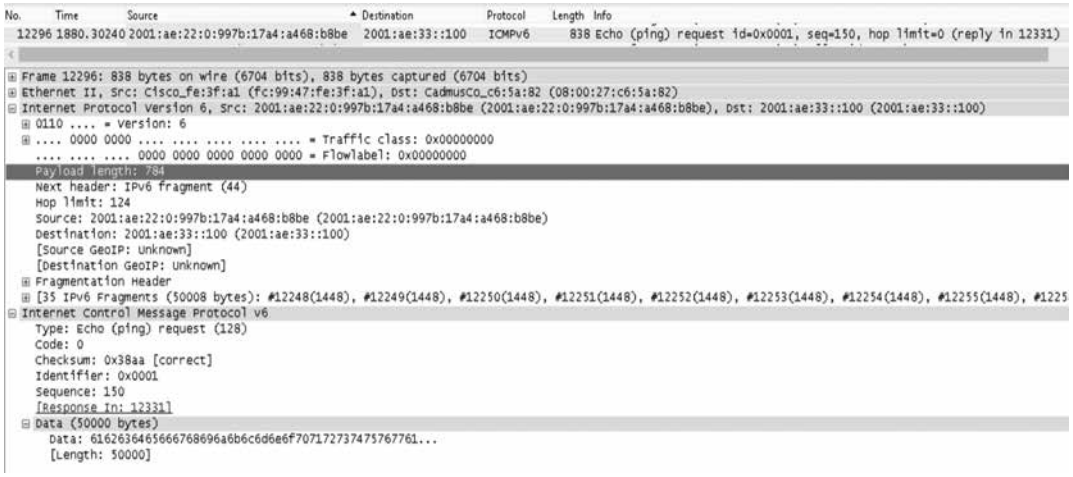


Figura 3. Captura de tráfico ICMPv6 con paquetes de 50000 bytes
Fuente: Los autores

La Figura 4a presenta el análisis de tráfico en el periodo 1700 a 1950 segundos. En el lapso de 1740 a 1930 segundos, se corrieron varias pruebas de ICMPv6 con paquetes de 50000 y 55000 bytes a todos los clientes en la red. El periodo a analizar puntualmente inicia en el tiempo 1879.06 y finaliza en 1879.35. El valor en el punto 1880 es de 31104 bytes/tick (un tick en esta gráfica corresponde a 10 segundos y 10 pixeles por tick). Esto es, el valor del tráfico en ICMPv6 en la interfaz LAN del enrutador SITE-C.
La Figura 4b muestra parcialmente el tráfico ICMPv6 capturado por Wireshark correspondiente a este tiempo.

La Figura 5 muestra el análisis de tráfico en el lapso de 0 a 250 segundos. Se muestra tráfico HTTP, ICMPv6 y FTP. Se

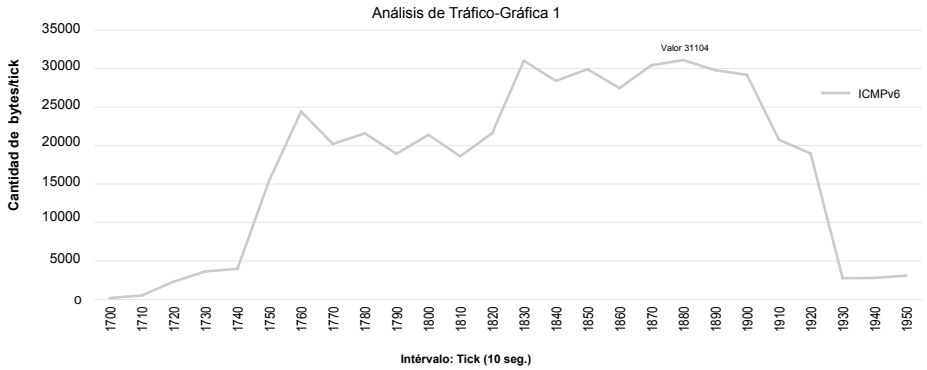


Figura 4a. Comportamiento de la red con paquetes de 50000 bytes (ICMPv6)
Fuente: Los autores

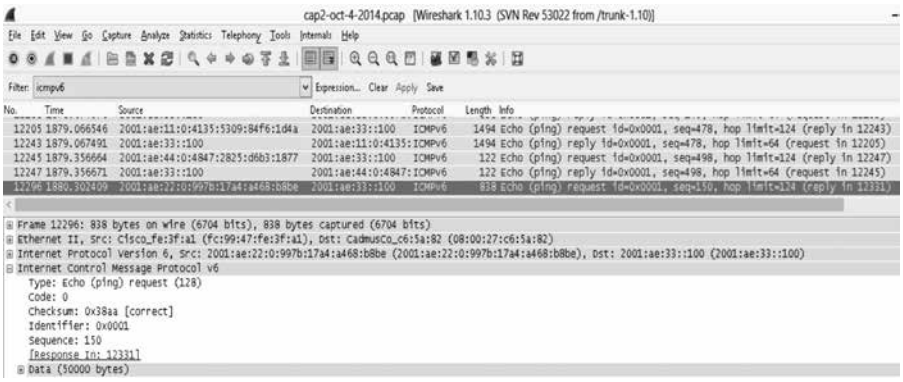


Figura 4b. Tráfico capturado por Wireshark (ICMPv6)
Fuente: Los autores

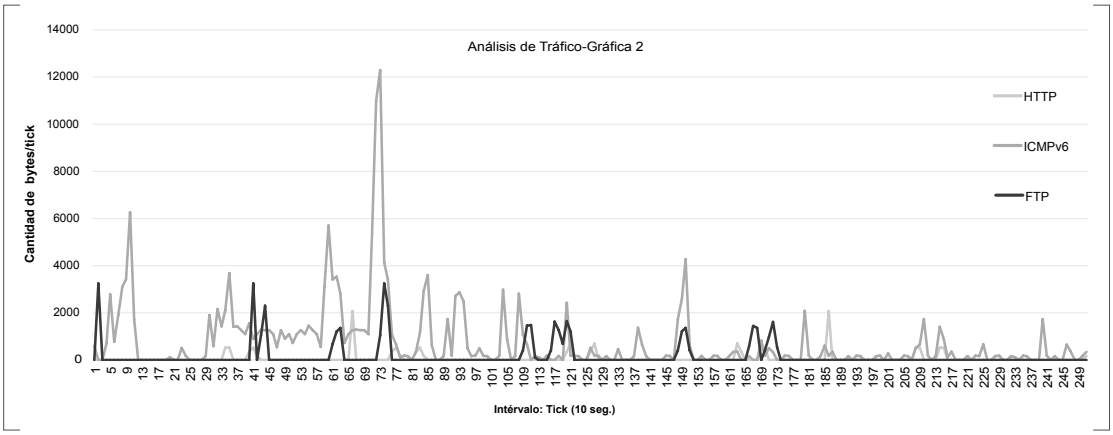


Figura 5. Análisis de tráfico HTTP, ICMPv6 y FTP
Fuente: Los autores

observa que el tráfico que más relevancia tiene es el ICMPv6, ya que se están inyectando paquetes de gran tamaño. Sin embargo, al hacer el comparativo con el tráfico FTP-data, este tráfico parece insignificante.

La Figura 6 presenta el tráfico FTP-data en el mismo periodo, mientras la Figura 7 hace el comparativo de HTTP, ICMPv6, FTP y FTP-data.

La Figura 7 muestra de forma casi desapercibida el tráfico HTTP, ICMPv6 y FTP. La flecha en la gráfica revela el pico de tráfico ICMPv6 en el segundo 73.

5. Conclusiones

La arquitectura de red anterior presenta de forma genérica el núcleo de la red de un proveedor de servicios de Internet (ISP), y busca obtener información que permita tomar conclusiones acerca del comportamiento de la red cuando funciona de manera real en forma exclusiva con IPv6. En esta primera etapa se muestra cómo se ha configurado el escenario genérico con el protocolo intradominio OSPFv3, para permitir, a partir de él, realizar nuevas configuraciones y pruebas de la red.

El escenario presentado es sólo una porción de las diferentes alternativas que se están evaluando en este trabajo. Se realizaron tres pruebas diferentes: la primera sin DiffServ y con interfaces de salida configuradas como FIFO. En la segunda, se configuraron las interfaces de salida como WFQ y se realizaron las mismas pruebas. En este último escenario se comprobó cómo se mejoró el desempeño de la red para casi todas las aplicaciones, con excepción a VoIP, que requiere adicionalmente configurarle una cola de salida del tipo LLC y una asignación de ancho de banda garantizado. El tercer escenario incluye la aplicación de DiffServ, desarrollando la clasificación de paquetes, creación de políticas y marcación. También se incluye un tratamiento especial para el tráfico best-effort (suavizado y compresión de la cabecera TCP), y aplicación del algoritmo de descarte RED para la clase VideoInteractivo, que permite mejorar significati-

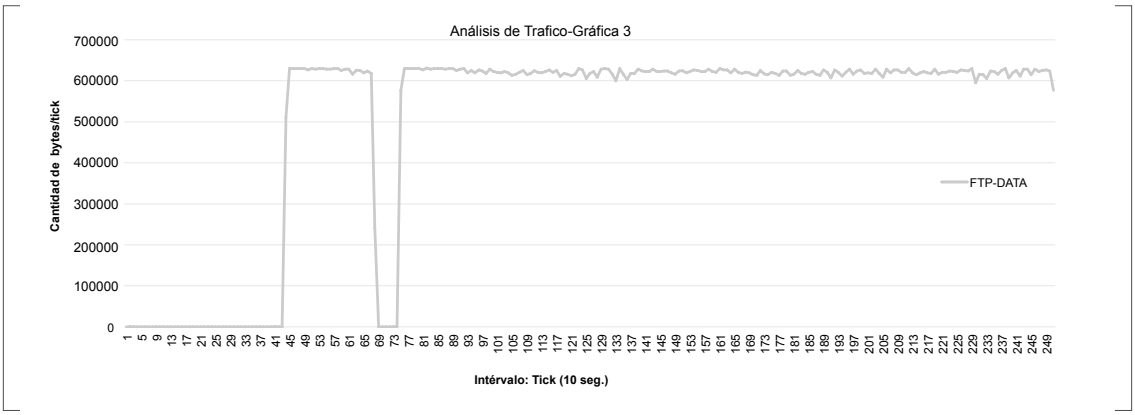


Figura 6. Análisis de tráfico FTP-data
Fuente: Los autores

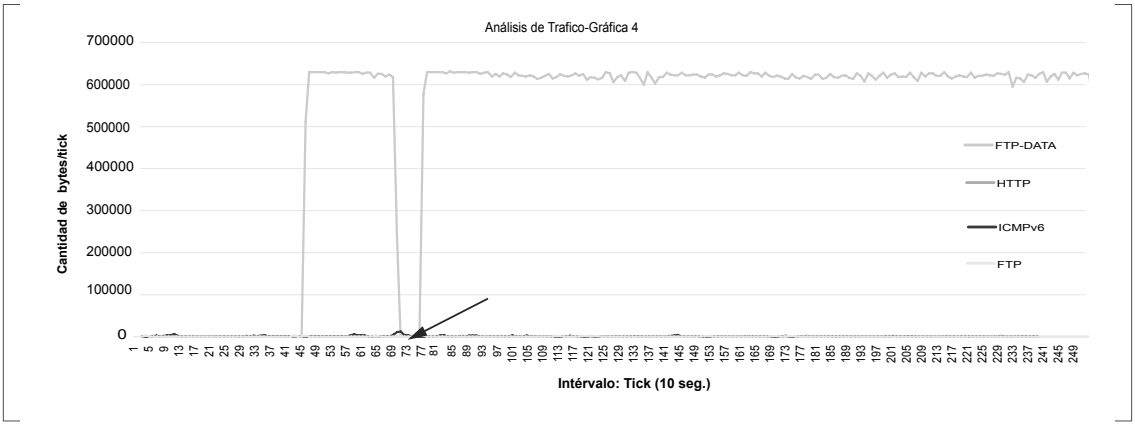


Figura 7. Análisis de tráfico HTTP, ICMPv6, FTP y FTP-data
Fuente: Los autores

vamente la forma de descarte de paquetes en momentos de congestión. Para el desarrollo de los servicios, se configuraron varias máquinas virtuales (MV) en VirtualBox[6] para soportar los servidores y clientes en la red.

A pesar de tener enlaces WAN de baja velocidad (512 kbps), las interfases no se congestionaron con facilidad. Cuando ocurre la congestión, el retardo se incrementa considerablemente. Para el caso de ICMPv6 con inyección de paquetes de 50000 bytes, el retardo en momentos de congestión fue superior a 3800 mseg. Al incluir en las pruebas aplicaciones del tipo IPTV, WEB y FTP, se puedo comprobar la degradación casi de inmediato en las interfaces involucradas.

Las aplicaciones VoIP utilizadas emplean el códec PCM G.711 muestreados a 20 ms, presentan un consumo de 85.6 kbps total (carga útil y cabeceras). Aunque no parece mucho ancho de banda, estas aplicaciones requieren un envío de paquetes garantizado, el retardo extremo-a-extremo inferior a 150ms y una diferencia de llegadas de paquetes menor a 30ms. Al competir estos paquetes en la red, sufren deterioro significativo en los recursos que demandan. Por esta razón, deben ser marcados en el campo DS como EF y ser tratados con una cola de salida de los enrutadores como LLC, aparte de garantizar un ancho de banda mínimo. Estas características (con excepción de LLC) se implementaron en la política de QoS.

Se puedo comprobar cómo las aplicaciones inelásticas (VoIP, video interactivo y video en demanda) son las que más se deterioran cuando compiten por recursos y cuando no se tiene una política de QoS aplicada. Para aplicaciones IPTV tales como el video en demanda, se requiere adicionalmente un considerable ancho de banda que depende de la calidad del video. El retardo y diferencia de retardo también afectan considerablemente la funcionalidad de estas aplicaciones.

Se puede concluir que aunque se dispone de políticas de QoS debidamente aplicadas, las mismas no tienen ninguna funcionalidad cuando el tráfico que se demanda es superior a la capacidad de la red. Por más políticas de QoS que se tengan definidas y aplicadas, estas no funcionan, y la única solución es aumentar el ancho de banda disponible.

Otro de los problemas encontrados se relaciona con el protocolo de enrutamiento OSPF. Este protocolo emplea como ruta entre dos redes, la ruta de menor costo. Esta ruta, como se puede evidenciar, se congestiona fácilmente por exceso de tráfico. Al verificar, aunque todos los enrutadores del núcleo tenían interfaces sin congestión, ninguna de ellas se empleó como interface para el envío de tráfico. Ello debido a que OSPF determina que tales interfases (aunque mejores, por no presentar congestión), no son las que muestran un menor costo a la red de destino.

Se ha podido comprobar que poner a punto una red del tipo de un ISP es algo complejo, y que aparte de los conocimientos teóricos, se deben realizar muchas pruebas y ajustes en la marcha, hasta lograr los objetivos propuestos. Para alcanzar la saturación de los enlaces WAN del ISP, se debe generar mucho tráfico de distinto tipo, y con diferentes patrones. También se encontró que una sola aplicación de IPTV no consigue congestionar los enlaces, y que se requiere poner a competir dicho tráfico con otro tipo de aplicaciones, tales como FTP, WEB e ICMPv6.

Como resultado de las pruebas, se pudo medir y evaluar el rendimiento de una red de núcleo que funciona con IPv6 sin soporte alguno a IPv4, alcanzando el objetivo propuesto de la investigación. También se pudo establecer un método de implementación de diferentes servicios de Internet y se logró integrar todo en una misma arquitectura. El trabajo permitió realizar un análisis de rendimiento de tráfico en la red, en un escenario lo más cercano a un entorno real, y en particular, a la red que podría disponer un ISP. Como resultado académico, se escribieron varios documentos escritos que han servido de base para la realización del primer seminario sobre servicios de IPv6 en la Universidad Libre y la sustentación de tres tesis de pregrado en Ingeniería de Sistemas. También se ha incluido bastante material en los diplomados de extensión CCNA de Cisco Networking Academy.

El trabajo desarrollado se diferencia de otros, en que la arquitectura se ha diseñado y probado en un laboratorio con equipos reales, equivalentes a los que tendría un ISP, mientras los trabajos correlacionados identificados a la fecha, han implementado el trabajo únicamente en simuladores por software. A la fecha, no se conocen trabajos experimentales de este tipo en IPv6 en la comunidad científica internacional.

Los trabajos en desarrollo y futuros sobre esta arquitectura, incluyen la simulación completa de la red empleando el simulado de redes GNS-3 y la conexión con MV. Aunque este escenario parecería sencillo, consume mucho recurso de hardware y requiere de varios ajustes para que el comportamiento sea similar al del laboratorio experimental. Otros trabajos futuros incluyen la integración con MPLS con y sin TE, así como adicionar generadores de tráfico sintético, lo que permitirá más opciones para congestionar la red. El empleo de analizadores de protocolos y red permitirá evaluar el tráfico y generar ecuaciones matemáticas y gráficas que describan su comportamiento.

Agradecimientos

Se agradece el apoyo de la Universidad Libre Seccional Cali, por facilitar el laboratorio de telemática de la Facultad de Ingeniería, donde se realizaron las pruebas de laboratorio. Igualmente, a la Universidad del Cauca, Facultad de Ingeniería Electrónica y Telecomunicaciones, y en especial al grupo de Investigación Nuevas Tecnologías en Telecomunicaciones – GNTT.

Conflicto de intereses

Los autores declaran no tener ningún conflicto de intereses.

Notas

1. RFC: Request For Comments. Documentos desarrollados por la IETF para el desarrollo y avance de Internet.
2. Este RFC actualiza los RFCs 1248, RFC 1247, RFC 1195, RFC 1123, RFC 1122, RFC 1060 y RFC 791, definidos para IPv4.
3. Según informe de ARIN (American Registry for Internet Numbers), el 3 de febrero de 2011 se entregaron todos los 256 bloques de direcciones IPv4 /8 disponibles en el mundo. Esto significa el agotamiento de las direcciones públicas IPv4 y la necesidad inminente de utilizar las direcciones IPv6. Fuente: https://www.arin.net/knowledge/ip_address_pools.pdf
4. Next Generation IPTV, ITU. Fuente: http://www.slideshare.net/RockyS11/next-generation-iptv
   NGN and IPTV, ITU. Fuente: http://www.itu.int/
5. Cisco Systems es un fabricante multinacional que soporta los principales ISPs en el mundo, y es precursor de la tecnología de multietiquetado MPLS.
6. VirtualBox: https://www.virtualbox.org/

Referencias bibliográficas

1. ALMQUIST, Philip. Type of Service in the Internet Protocol Suite, RFC 1349. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc1349 >
2. AZIZ, Tariq y SAIFUL, Mohammad. Performance Evaluation of Real-Time Applications over Diffserv/MPLS in IPv4/IPv6 Networks. Karlskrona, Sweden: Blekinge Institute of Technology, Tesis Master, 2011.
3. AZIZ, Tariq, SAIFUL, Mohammad, ISLAM, Nazmul y POPESCU, Adrian. Effect of packet delay variation on video/voice over Diffserv-MPLS in IPv4/IPv6 Networks. En: International Journal of Distributed and Parallel Systems (IJDPS), 2012, P 27-47.
4. BLAKE, Steven, et al. An Architecture for Differentiated Services, RFC 2475. 1998. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc2475>
5. BRADEN, Bob, CLARK, David, y SHENKER, Scott. Integrated services in the Internet Architecture: An overview, RFC 1633, 1994. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc1633>
6. BRADNER, Scott y MANKIN Allison. The Recommendation for the IP Next Generation Protocol. RFC 1752, 1995. Disponible desde Internet <URL: http://tools.ietf.org/rfc/rfc1752>
7. Cisco Systems. Cisco AVVID QoS Design Guide, 2005. Disponible desde Internet <URL: http://www.cisco.com/web/AT/assets/docs/qosdg_new.pdf>
8. COLTUN, Rob, FERGUSON, Dennis, y MOY, John. OSPF for IPv6, 1999. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc2740>
9. COLTUN, Rob, FERGUSON, Dennis, MOY, John y LINDEM , Acee. OSPF for IPv6, 2008. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc5340>
10. DEERING Stephen, HINDEN, Robert. Internet Protocol, Version 6 (IPv6), RFC 1883, 1995. Disponible desde Internet <URL: http://tools.ietf.org/rfc/rfc1883>
11. DEERING Stephen, HINDEN, Robert. Internet Protocol, Version 6 (IPv6), RFC 2460, 1998. Disponible desde Internet <URL: http://tools.ietf.org/rfc/rfc2460>
12. DARPA, Defense Advanced Research Projects Agency, Internet Protocol, RFC 791, 1981. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc791>
13. FIROUI, Victor, LE BOUDEC, Jean-Yves, TOWSLEY, Don, y ZHANG, Zhi-Li. Theories and Models for Internet Quality of Service. En: Proceedings of the IEEE, Vol.90, 2002, P 1565-1591.
14. GROSSMAN, Dan. New Terminology and Clarifications for Diffserv, RFC 3260, 2002. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc3260>
15. International Telecommunication Union (ITU). Recommendation ITU-T Y.1901, 2009. Disponible desde Internet <URL: http://www.itu.int/ITU-T/recommendations/index.aspx>
16. International Telecommunication Union (ITU). IPTV vocabulary of terms, 2008. Disponible desde Internet <URL: http://www.itu.int/md/T05-FG.IPTV-DOC-0082>
17. LEE, Chae-Sub. IPTV over Next Generation Networks, En: Broadband Convergence Networks, 2007. BcN '07. 2nd IEEE/IFIP International Workshop on. Disponible desde Internet <URL: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4238824>
18. NICHOLS, Kathleen, BLAKE, Steven, BAKER, Fred, y BLACK, David. Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, RFC 2474, 1998. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc2474>
19. ODOM, Wendell y CAVANAUGH, Michael. IP Telephony Self-Study Cisco DQOS Exam Certification Guide. Indianapolis, Cisco Press, 2004.
20. ORTIZ, Jesús, PEREA, Jorge, ORTIZ, Alejandro, y SANTIBAÑEZ, David. Integration of HMIPv6/MPLS. En: International Journal of Research and Reviews in Computer Science IJRRCS, Vol.2, No.1, 2011, P 238-241.
21. ORTIZ, Jesús, PEREA, J., SANTIBAÑEZ, David y ORTIZ, Alejandro. Integration of Protocols FHMIPv6/MPLS in Hybrid Networks. En: Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT), 2011, P 32-41.
22. ORTIZ, Jesús, GONZALES, Santiago, PEREA, Jorge, y LÓPEZ, Juan. AHRA: A routing agent in order to support the Hierarchical Mobile IPv6 protocol with Fast-Handover over mobile Ad-hoc network scenarios. En: International Journal of Research and Reviews in Computer Science IJRRCS, Vol 2, No.1, 2011, P 232-237.
23. RAMAKRISHNAN, K.K., FLOYD, Sally, y BLACK, David. The Addition of Explicit Congestion Notification (ECN) to IP, RFC 3168, 2001. Disponible desde Internet <URL: http://tools.ietf.org/html/rfc3168>