Pontificia Universidad Católica de Valparaíso

Facultad de Ingeniería

Escuela de Ingeniería Informática

IMPLEMENTACIÓN DE MOVILIDAD EN MIPv6

Autor:

Nikolaos Leiva Ugarte

Informe final del Proyecto para optar al Título profesional de

Ingeniero Civil en Informática

Profesor guía:

Iván Mercado Bermúdez

Profesor Co-referente:

Nibaldo Rodríguez Agurto

Julio de 2007

A Dios, mi familia y Universidad por darme las oportunidades y herramientas para salir adelante.

Resumen

Hoy en día, los distintos tipos de tecnologías inalámbricas tienden a integrarse desde el punto de vista del acceso a éstas, haciendo su disponibilidad posible en numerosas ubicaciones. En este ámbito, el trabajo se enfoca en establecer una arquitectura de acceso inalámbrico, que permita la selección del mejor medio de enlace a la red en cada momento, dependiendo de las características de las transmisiones, proveyendo un adecuado nivel de QoS a éstas, en un ambiente multihoming sobre una red IPv6. En forma simultánea, dado el actual periodo de transición desde redes bajo IPv4 a IPv6, se realiza un análisis sobre distintas situaciones y consideraciones ha abarcar durante este proceso de cambio las que serán desarrolladas en el proyecto.

Abstract

Nowadays, the different kind of wireless technologies tend to integrate from the kind of access to them, doing their availability possible in several locations. In this ambit, the work to be done is focused on establishing a wireless access architecture which allows the selection of the best network link medium at each moment, depending of the transmission characteristics, providing an adequate QoS level of them, in a multihoming environment over an IPv6 network. In a simultaneous way, giving the actual transition period from IPv4 to IPv6

networks, is done an analysis of several situations and considerations covered during this change process and that will be developed in the project.

CAPÍTULO 1
INTRODUCCIÓN Y DISCUSIÓN BIBLIOGRÁFICA.

1.1 INTRODUCCIÓN

Las comunicaciones inalámbricas han evolucionado enormemente, al estado de permitir establecer diferentes medios de comunicación entre dos o más dispositivos móviles mediante distintas tecnologías, tal como GPRS, WiMax, Bluetooth, entre otras. Además, la creciente apreciación de Internet como fuente de servicios y comunicaciones para distintas áreas e industrias, eleva el deseo de contar con acceso en cualquier lugar a cualquier hora. Este crecimiento es conducido por el desarrollo de nuevas tecnologías y aplicaciones multimedia en tiempo real, como VoIP, IPTV o videoconferencias, con nuevas funcionalidades y mayor valor agregado entregado por parte de los proveedores de servicios. A esto se añaden, por el lado de los usuarios, los requisitos de estar siempre conectados (conocido como “always on”) y el desarrollo de nuevos servicios a través de la Web, como e-learning; y, por el lado de los proveedores, una mayor capacidad disponible en los backbones de comunicaciones y la prestación de nuevas características, como el soporte de tráfico convergente y con atributos de calidad de servicio asociados en ambientes móviles.

Si sumamos a lo anterior el uso de Mobile IPv6 (MIPv6) [1], protocolo de movilidad IP para IPv6 (Internet Protocol version 6) [2], es posible expandir tanto el rango de beneficios como las prestaciones a obtener en los enlaces establecidos por el dispositivo. Hoy en día, Internet está basada principalmente en el protocolo de capa de red IPv4 [3], el cual permite identificar y comunicar a los dispositivos conectados a la red, el cual ha sido extendido para soportar la movilidad. Pero para asegurar una conectividad continua, con una deseable calidad de servicio (conocido como QoS), es preferible, para un nodo o red móvil, estar conectado vía varias interfaces, usando distintos medios de comunicación y distintos puntos de acceso a la vez, lo cual no es posible de realizar en forma transparente bajo IPv4. Varios proyectos han aparecido para acercar estos requerimientos a los usuarios de tecnologías móviles, con el fin de aumentar las prestaciones posibles de obtener, pero es difícil encontrar una propuesta que

junte estos puntos de forma de obtener una constante conectividad. Esto lleva a estudiar el cumplimiento de una serie de requisitos y objetivos para dar al usuario una transparencia e independencia total sobre el proceso de comunicación, dándole la apariencia de nunca haber abandonado su hogar.

Como el trabajo se enfoca en un ambiente IPv6, otra de las ideas a desarrollar corresponde a entregar distintos puntos a evaluar al momento de introducir servicios y redes sobre este tipo de protocolo en una organización, por lo que se da una explicación de temas como las consideraciones a tomar, la planeación necesaria para llevar a cabo una implementación y un nuevo direccionamiento de la red, el porte o convivencia de servicios y aplicaciones desde IPv4 a IPv6, además de abordar aspectos relacionados con la seguridad a nivel de la red.

El trabajo se encuentra dividido en dos grandes áreas. La primera parte (Capítulos 2 y 3) hace un análisis general de movilidad, IPv6 y otros conceptos estudiados en el proyecto. La segunda parte (Capítulos 4 al 6), se enfoca en los aspectos de diseño e implementación de la arquitectura y escenario de prueba creado. En forma específica, el Capítulo 2 menciona el desarrollo actual de las tecnologías a utilizar, en el Capítulo 3 del trabajo se encuentra una explicación de los distintos tópicos estudiados enfocándose en IPv6, en movilidad y en el uso de redes multihoming junto a la manera de controlar las transmisiones mediante el establecimiento de filtros y políticas. En este capítulo también son estudiados distintos efectos producto de la introducción del mundo IPv6 a una institución y las técnicas usadas actualmente para habilitar la transición y comunicación entre los dos tipos de protocolo. En el Capítulo 4, se explica en profundidad el trabajo realizado en la implementación de movilidad y de políticas de ruteo sobre una red IPv6, mostrándose los modelos y diagramas generados que explican el proceso completo de transmisión, además de mostrar las distintas pruebas llevadas a cabo, y en el Capítulo 5 se detallan las distintas pruebas que se han realizado sobre las implementaciones y arquitecturas desarrolladas. En el capítulo 6, se hace una revisión de los distintos objetivos que se plantean a lo largo del proyecto y de cómo estos son abordados en la arquitectura propuesta. Finalmente, en el apartado de Apéndices, se encuentra el código generado como los archivos de configuración necesarios para la ejecución de los escenarios de prueba.

1.2 DESARROLLO ACTUAL

La nueva generación de Internet, basada principalmente sobre el protocolo IPv6, se halla aún con puntos abiertos en su desarrollo. Entre estos, se encuentran dos relacionados a este proyecto: soporte a la movilidad y capacidad de multihoming. La movilidad se está convirtiendo en uno de los componentes principales de la nueva Internet, debido al gran desarrollo de tecnologías inalámbricas. Sin embargo, actualmente la movilidad no permite que cada dispositivo o equipo de una red sea identificado y ubicado sólo por su dirección, o que pueda mantener y manejar distintos enlaces en forma simultánea.

IPv6 es el siguiente paso en el desarrollo del protocolo IP, superando un gran número de puntos débiles que presenta el actual protocolo IP (IPv4) a la vez de incluir diversas mejoras, siendo una de las más relevantes el aumento del número de direcciones disponibles lo cual es consistente con el aumento de usuarios y dispositivos con acceso a la red que presenta hoy en día, como con las nuevas tecnologías de comunicación que permiten estar siempre conectados, requiriendo un punto fijo de localización. En cuanto a esto, IPv4 soporta 4.294.967.296 (2^{32}) direcciones de red diferentes (el número real de direcciones disponibles es mucho menor dado los distintos segmentos existentes en los cuales se divide), un número inadecuado para dar una dirección a cada persona del planeta (cantidad que está sobre los 6.600.000.000, aprox. 2^{33} habitantes), y mucho menos para cada vehículo, teléfono, PDA u otro dispositivo habilitado con conectividad; mientras que IPv6 soporta 340.282.366.920.938.463.463.374.607.431.768.211.456 (2^{128}) direcciones (aproximadamente $6{,}65 \times 10^{23}$ direcciones por metro cuadrado de la superficie de la Tierra). Aunque tampoco en IPv6 todo el espacio de direccionamiento es usado debido a distintos segmentos presentes (por ejemplo direcciones privadas, multicast o usadas en mecanismos de transición desde IPv4), se han definido los segmentos de red a asignar a las organizaciones con un largo de 48 bits, lo cual produce 2^{48} direcciones posibles de usar dentro de esta, lo cual soportaría que toda la población mundial trabajase en esa organización y cada uno tuviese una dirección asignada.

Hoy en día, las distintas implementaciones de movilidad en IPv4 permiten que los dispositivos móviles puedan conectarse a distintos puntos de enlace, seleccionar de estos uno y usarlo, entregando la conexión requerida al usuario. Sin embargo, existen distintos puntos en los que es posible una mejora, y que no es posible ser entregada por IPv4 en forma

transparente, o simplemente no es entregada. Por ejemplo, las redes a las que el dispositivo se enlaza, pueden entregar distintas direcciones de red, las que generalmente no son globales, lo que dificulta la mantención de las sesiones establecidas y reduce los servicios que son posibles de utilizar. El desarrollo de nuevos protocolos, como el protocolo SIP [4] para aplicaciones multimedia, usado frecuentemente en los dispositivos móviles, permite esconder este desplazamiento mediante un direccionamiento adicional entregado a cada usuario, e independiente del dispositivo, que permite ubicarlo en cualquier locación. Además, un área en el que se mantiene investigación hoy en día es el soporte de múltiples interfaces de conexión para un mismo nodo, lo que permitiría incluir varios beneficios al enlace, los que hoy en día no son posibles de obtener.

El punto de partida para transmisiones móviles en IPv6 es Mobile IPv6. Existen varias formas de abordar el problema de movilidad, entre las cuales se encuentran los protocolos MIPv6, SHIM6 [5], SIP y HIP [6] entre otras, pero se eligió MIPv6 debido por sus prestaciones, alcance, trabajos en desarrollo para extenderlo y el número de implementaciones que posee (lo que indica que existe suficiente trabajo dedicado en su desarrollo). En [7] se encuentra una comparación de MIPv6 con SHIM6 y HIP, mientras que en [8] es comparado con el protocolo de la capa de aplicación SIP.

El protocolo MIPv6, solución a nivel de red para otorgar movilidad en ambientes heterogéneos, permite a un nodo moverse de un enlace a otro sin perder su dirección original y, por lo tanto, sus comunicaciones desde y hacia la red, ya que este movimiento es transparente a la capa de transporte y superiores [1]. MIPv6 es extendido mediante el protocolo NEMO [9] (Network Mobility Basic Support) el cual permite que una red entera cambie su ubicación pero manteniendo su dirección y las comunicaciones establecidas. En Mobile IPv6, cada nodo móvil (MN por sus siglas en inglés, Mobile Node) tiene asignada 2 direcciones IP que permiten identificarlo. La primera es aquella otorgada por su lugar de origen o Home Agent, conocida como Home Address (HoA), y es la dirección permanente asignada al nodo. La otra dirección es conocida como dirección foránea o llamada también Care of Address (CoA), la cual permite identificar la localización actual del nodo, siendo obtenida en la red externa en donde se encuentra el dispositivo. Este proceso permite establecer y mantener una comunicación con el nodo móvil, independiente de sus desplazamientos.

Uno de los problemas presentes en las soluciones inalámbricas actuales, se encuentra en los casos donde las transmisiones tienen que mantener un nivel alto de calidad debido a la criticidad de éstas. Por ejemplo, en el contexto de aplicaciones de tiempo real como telemedicina, en la comunicación establecida desde un hospital a una ambulancia en terreno, es necesario que la información (consistente en datos, voz e imagen) llegue en forma correcta e ininterrumpida de un extremo a otro, situación que usando un sólo enlace de red de bajas prestaciones, difícilmente pueda ser lograrlo, ya que sólo segundos de pérdida afectan gravemente la transmisión. En especial, en un ambiente móvil, debido a los cambios de puntos de enlace, se producen demoras en la comunicación y pérdidas de paquetes, afectando negativamente la calidad final de la transmisión.

Multihoming [10,11] es un concepto relacionado con el uso de distintos enlaces de comunicación entre dos puntos, pudiendo generalmente ser logrado debido al uso de diferentes interfaces o por encontrarse el nodo asignado a distintas redes externas. Hoy en día, es ampliamente usado en soluciones empresariales, ya sea dentro de redes LAN, para incrementar el desempeño de la red y prestar redundancia a la estructura de ésta, como en forma externa, en conexiones con distintos ISP, de forma de hacer que la red sea resistente a distintas fallas que puedan presentarse. Actualmente, se encuentra en estudio para extender su uso a nodos móviles, debido al carácter dinámico y cambiante que presentan las conexiones realizadas en un ambiente inalámbrico. El uso de multihoming, aunque provee un alto nivel de movilidad, hace su administración más compleja, debido a la introducción de mecanismos de selección de interfaces, políticas de ruteo, detección de fallas, entre otros, los cuales deben ser desarrollados para su correcta aplicación en IPv6.

No obstante lo anterior, no existe un estándar que permita el uso de múltiples tecnologías de acceso en forma simultánea, pero el esfuerzo de desarrollo se encuentra enfocado hacia esa área. Por ejemplo, algunos progresos son:

En apoyo de Host Mobility (Sistemas que cambian su punto de enlace a la red): Mobile IP [1,12], HMIPv6 [13], FMIPv6 [14].

En apoyo de Network Mobility (Redes enteras que cambian su punto de enlace a la red): NEMO Basic Support [15].

En redes Ad-hoc (Protocolos de ruteo para redes sin infraestructura definida): MANET.

En Multihoming: Monami6 (Mobile Nodes and Multiple Interfaces in IPv6), Shim6.

Además de permitir el cambiar el punto de enlace a la red, es necesario que un dispositivo móvil sea capaz de estar adjunto a distintos medios. Algunos aportes en esta área son:

Extensiones del protocolo de Mobile IPv6 y NEMO para permitir el registro de múltiples Care of Addresses con una dirección Home Agent dada [16].

Políticas de intercambio de flujos y enlaces.

Filtros para Mobile IPv6 Bindings (NOMADv6), el cual introduce extensiones para el protocolo MIPv6 permitiendo el uso inteligente de múltiples puntos de acceso en forma simultánea.

A su vez, el desarrollo anterior es apoyado por distintas iniciativas de aplicación en este ámbito, los cuales se han incrementado los últimos años debido al mayor conocimiento relacionado a las nuevas tecnologías. Ejemplo de esto lo forman:

InternetCAR [17] (Investigación de los sistemas de comunicación).

InternetITS (Consorcio de industrias relacionadas de IT, como manufacturadores de autos, desarrolladores de servicios, vendedores de equipamiento, entre otros enfocados a establecer requerimientos del sistema de comunicación y de movilidad en la red, además de planear y ejecutar experimentos en situaciones reales).

Nautilus6 Working Group [18] (Enfocado en demostrar movilidad en la capa IP y en desarrollar nuevos procedimientos y escenarios en redes IPv6, como el descrito en [19]).

Proyecto IST-ANEMONE [20] (Proyecto que crea un testbed colaborativo, enfocándose en el uso de movilidad y características de multihoming bajo diferentes tecnologías actuales).

CAPÍTULO 2
ANÁLISIS DE OBJETIVOS Y MOTIVADORES.

2.1 MOTIVACIÓN

Para la realización del presente proyecto, se tienen los siguientes puntos como principales motivadores.

Constantemente aparecen nuevas áreas de desarrollo relacionadas con tecnología, por lo que es necesario estar enterado de cómo estas pueden ser usadas en forma correcta, obteniéndose los mayores beneficios de ellas. En este ámbito, se seleccionaron los temas de movilidad e IPv6 por ser actuales y por permitir la convergencia y la aparición de nuevas formas de servicios sobre las redes.

La introducción progresiva de IPv6 en los próximos años al ámbito productivo dará muchos beneficios y permitirá ampliar la gama de servicios a desarrollar sobre las redes. Sin embargo, también traerá nuevos desafíos, acerca de los cuales es importante tener conciencia de manera de hacer el proceso de transición seguro.

En Chile, el trabajo relativo a la introducción de IPv6 ha sido bajo, y en forma aislada, no existiendo investigación continua en esta área. Lo que se busca es incentivar el trabajo en esta materia, analizando los nuevos requerimientos y el crecimiento del mercado.

Entregar una visión integrada de IPv6 y movilidad en el ámbito productivo, enfocándose en el área de servicios a obtener sobre las redes, en un aspecto global, en vez de realizar un análisis específico en algunas áreas, debido al poco trabajo encontrado siguiendo esta orientación en la literatura actual.

2.2 OBJETIVO GENERAL

Definir e implementar una arquitectura multi-acceso mediante MIPv6, provocando cambios mínimos a las tecnologías e infraestructuras existentes.

2.3 OBJETIVOS ESPECÍFICOS

Análisis y diseño de alternativas de comunicación en función de los requerimientos planteados. Hoy en día se han propuesto varias formas de dar cumplimiento al objetivo planteado, como registro de múltiples interfaces de comunicación, división de tráfico, entre otras. Lo que se busca es estudiar y seleccionar aquellas que entreguen el mejor rendimiento, además de realizar las menores modificaciones a la arquitectura usada en la transmisión móvil, a la vez de estudiar formas de coexistencia con redes bajo IPv4.

Estudio de los potenciales beneficios y problemas de una arquitectura multiacceso bajo IPv6 en situaciones reales. Se busca el cumplimiento de beneficios en la transmisión como aumento del ancho de banda disponible, división y ordenamiento de las transmisiones según el tipo de tráfico específico y la confiabilidad en la comunicación, además de dar la apariencia al usuario de estar siempre conectado. Todo esto bajo IPv6, por lo que un análisis detallado respecto a este protocolo es también requerido.

Propuesta y diseño de un marco de routing basado en políticas de comunicación que soporte la arquitectura propuesta. Para permitir el envío efectivo de las transmisiones a través de los puntos de enlace establecidos a la red es necesario contar con un proceso de selección y de ordenamiento en las comunicaciones. El tema a estudiar se centra en el desarrollo de una estructura basada en políticas que permita a través de reglas y acciones un efectivo y eficiente comportamiento de la solución.

Implementación y prueba de las alternativas planteadas evaluándolas en cuanto a parámetros propios de las transmisiones. Existen distintas herramientas las cuales pueden ser usadas para implementar y probar la arquitectura desarrollada, además de trabajos ya realizados, los cuales pretenden dar una solución a los objetivos planteados. Con esto, se facilita el proceso de medición de los beneficios entregados por la solución propuesta como la tasa de error y de envío, la mantención del QoS en las transmisiones, entre otros.

2.4 REQUERIMIENTOS DE LA SOLUCIÓN

En base a los objetivos planteados, se establecieron los siguientes requerimientos para ser considerados en la etapa de diseño de la solución:

- **Preservar la transparencia en las comunicaciones**, permitiendo la comunicación entre un nodo que implemente la arquitectura a desarrollar con un nodo que no la implemente. También se debe permitir, en el caso que no implemente dicha arquitectura, funcione normalmente dentro de una estructura de multihoming, sin afectar su comportamiento.

- **No introducir vulnerabilidades adicionales** a la red resultante tras implementar la arquitectura a desarrollar. Se debe tratar que la comunicación permanezca en funcionamiento como lo es hasta antes de la implementación.

- **No romper las aplicaciones.** La arquitectura y posterior implementación de la arquitectura multihoming debe ser transparente al nivel de las aplicaciones.

- **Mantenerlo simple.** La solución debe estar enfocada a satisfacer los requerimientos realizando los cambios de la forma más simple posible sin afectar el proceso de comunicación que se establecería normalmente.

- **Seguimiento de estándares.** Los trabajos explicados en documentos RFC principalmente, o en DRAFTs que se encuentren en continuo desarrollo, serán tomadas como base para las soluciones planteadas en el presente trabajo.

CAPÍTULO 3
DISEÑO Y RESOLUCIÓN DEL TRABAJO REALIZADO.

3.1 IPv6

3.1.1 IPv6 y sus Beneficios

El presente desarrollo del protocolo IPv6, en reemplazo del protocolo IP versión 4 (ampliamente usado en la actualidad), permite la convergencia natural entre distintas tecnologías de acceso. De igual forma tiene presente en su estructura la acogida de requerimientos en distintos ámbitos, siendo los principales beneficios obtenidos: seguridad integrada en toda la red gracias a la incorporación de IPsec, permitiendo la implementación tanto de encriptación como de autentificación; confiabilidad; identificación de flujos y soporte de QoS; tolerancia y recuperación ante fallas; bajos costos de administración gracias a la autoconfiguración y al encaminamiento y direccionamiento jerárquico produciendo un ruteo más eficiente y rápido sobre la red y facilitando el proceso de migración; soporte de múltiples y nuevas generaciones de tecnologías de acceso, entre otras, además de ser transparente al uso de IPv4 y de entregar un modo sencillo de trasladar y migrar desde IPv4 a IPv6 permitiendo la coexistencia de ambos protocolos.

De los anteriores, los que más alcances tienen en el presente proyecto, son las características del espacio de direccionamiento, la habilidad de autoconfiguración y el soporte de movilidad.

En IPv6, las direcciones usadas son de 128 bits (formada de 8 campos de 16 bits), muy superior a los 32 bits utilizados en IPv4 (las cuales tienen una representación en el nuevo formato para facilitar la coexistencia), permitiendo más direcciones para distintas conexiones de dispositivos y usuarios, ayudando en forma adicional la incorporación y uso de nuevas tecnologías.

La autoconfiguración incluye crear una dirección en la capa de enlace y verificar su unicidad en el enlace, permitiendo determinar que información (por ejemplo dirección IP) es necesaria que sea configurada y la forma en que esta sea realizada, la que puede ser en forma automática (o stateless autoconfiguration) gracias a información disponible en el enlace y a la comunicada por los routers (incluido el prefijo de la red que abarca). Para formar la dirección IP global a usar, el equipo usa tanto la dirección local generada asociada a la interfaz del equipo como el prefijo informado por el router. En caso que este mecanismo automático no se encuentre soportado en la red o se requiera un mayor control sobre el proceso, se tiene como alternativa el usar un servidor (por ejemplo DHCPv6 [21]) en la red encargado de proveer (proceso conocido como stateful autoconfiguration) la información necesaria a los equipos conectados en ésta. Ambas formas de configuración pueden ser usadas en forma simultánea por un equipo para adquirir distintos tipos de información.

Ahora bien, relacionado con el soporte de movilidad, el uso de IPv6 tiene puntos importantes incorporados ya que en su diseño se incluyó esta área. Este tema se trata en profundidad en el punto 3.1.3.

De las funcionalidades anteriores, salen nuevos usos y servicios que pueden ser implementados sobre las redes. Por ejemplo, servicios como VoIP, los cuales no tienen un fácil desarrollo en ambientes IPv4 hoy en día, por motivos como el uso de NAT, debido a la necesidad del establecimiento de conexiones extremo a extremo. De igual forma, protocolos de capas superiores, como FTP, operarán de manera más fácil (transparente) debido a no existir capas NAT por la cual atravesar.

Otro uso favorecido corresponde a las conexiones a través de mensajes multicast. Multicast es soportado en IPv6 con el fin de transmitir paquetes a un conjunto de usuarios finales (pertenecientes a un mismo grupo multicast) a través de la red, brindando soporte a un amplio rango de aplicaciones como educación a distancia o comercio electrónico, servicios como televisión sobre IP (IPTV), además de abarcar dispositivos portátiles. El uso de IPv6 en multicast ayuda a reducir la complejidad de las aplicaciones que brindan algún tipo de solución en esta área, además de facilitar el control sobre los servicios que se están entregando. De igual forma, es posible ampliar en una forma más sencilla la manera de abarcar un mayor número de dispositivos, pudiendo lograr un hogar totalmente conectado en

forma transparente a Internet donde varios aparatos están conectados y manejados desde Internet en una forma más sencilla.

Estos nuevos servicios también son llevados a proyectos de gran escala para lograr convergencia en las comunicaciones, como es el caso del realizado en la ciudad de Harrisonburg, EEUU [22], permitiendo mezclar el acceso total a IPv6 en las ciudades, servicios como multicast IPTV, salones virtuales y el desarrollo de e-commerce y e-business en forma masiva, además de otros usos no posibles con IPv4, como servicios para las agencias de emergencias, teniendo como base la convergencia de información, brindando voz, datos e incluso servicios geográficos como mapas.

3.1.2 IPv6 en el mundo

De acuerdo a una predicción realizada por la empresa CISCO [23,24] en el año 2005 (Figura 3-1), a partir del año 2007, comenzará una adopción sostenida de IPv6 a nivel empresarial y de usuarios.

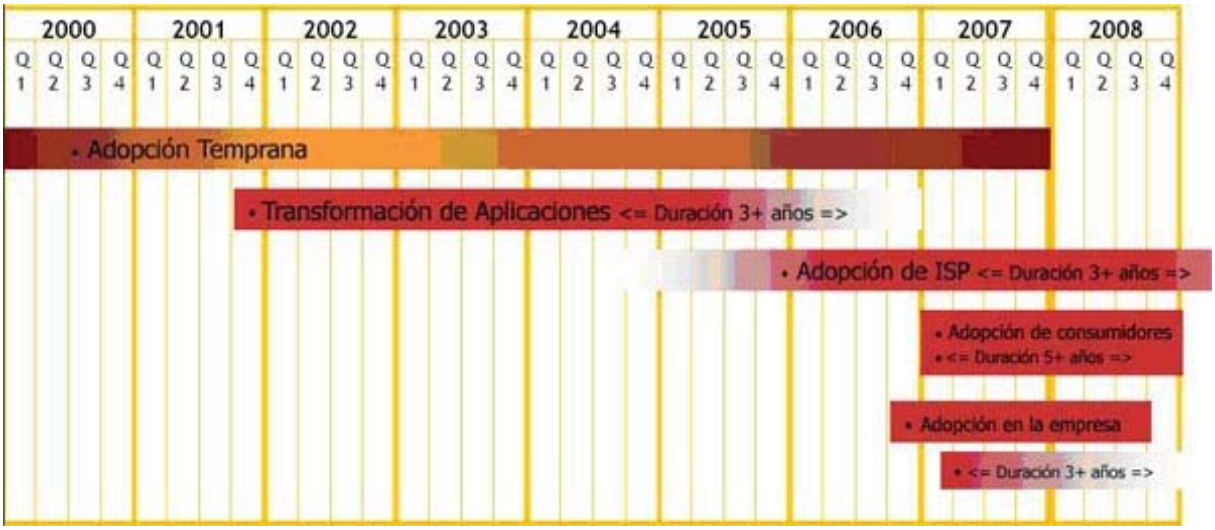


Figura 3-1. Proceso de adopción de IPv6. (Imagen obtenida en [23])

Hoy en día, en cuanto al proceso de adopción de IPv6 a nivel mundial, éste se encuentra encabezado por Asia, donde países como China, Japón y Taiwán tienen en funcionamiento IPv6 a nivel de ISPs y redes nacionales, junto a la prestación de servicios sobre ésta, estando su uso al alcance de todos. En la Figura 3-2 se encuentra un análisis actual de la topología (a nivel de sistemas autónomos, o AS) de ambas redes, IPv4 e IPv6, en la cual se visualiza que la red IPv6 se encuentra más dispersa y con muchos menos nodos que la red IPv4. En el

análisis se muestra que, aunque en la red IPv4 la mayor cantidad de puntos de acceso se encuentra en Estados Unidos; en IPv6, se encuentran en Asia y Europa. Principalmente en Asia, este gran desarrollo tecnológico sobre IPv6 se sustenta en un fuerte apoyo por parte del gobierno, el gran número de usuarios que utilizan servicios en línea, y una rápida adopción de redes y tecnologías inalámbricas (como teléfonos celulares, PDA, teléfonos IP, y consolas de juego) entre la población.

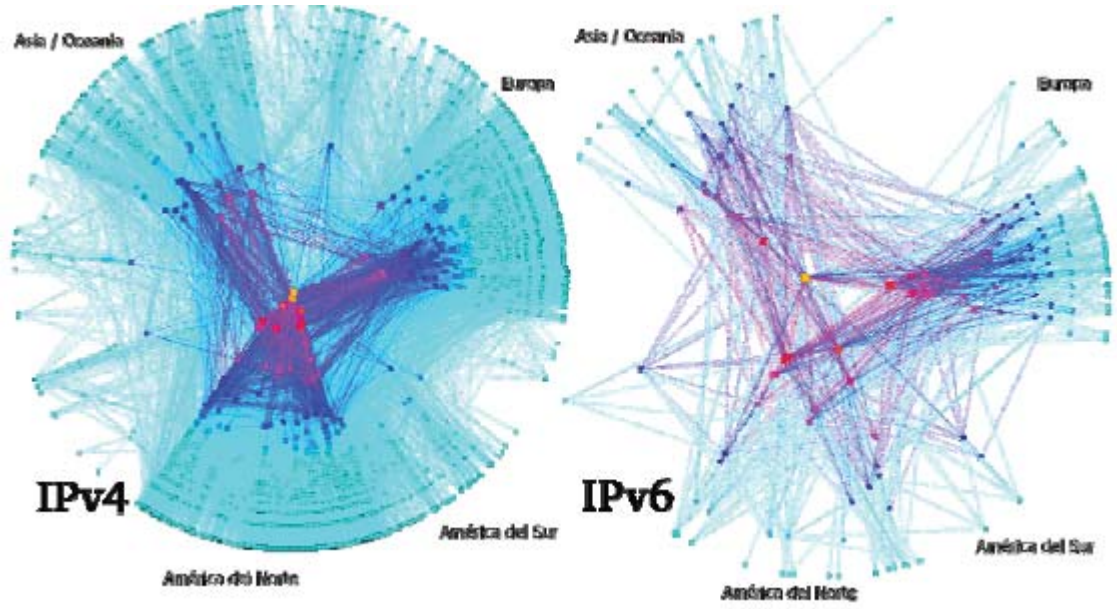


Figura 3-2. Extensión de nodos con soporte de IPv6 o IPv4 (situación a Marzo, 2005). (Imagen obtenida de CAIDA [25])

Por casos revisados [26], la adopción comienza generalmente a través de redes de investigación y educativas las cuales están enfocadas en promover el desarrollo de nuevas tecnologías de forma de convidar el uso de éstas a otras empresas, pero en varios casos, este proceso es promovido directamente por instituciones gubernamentales (Figura 3-3), como en el caso de Taiwán en donde el objetivo es potenciar el uso de tecnología en áreas como gobierno y comercio electrónico además en los servicios ofrecidos a la comunidad, los cuales deberían estar en funcionamiento el año 2008. Este incentivo ha llevado a las industrias del área de tecnología a desarrollar productos sobre redes en áreas como seguridad, multimedia y servicios como voz y video sobre IP. En forma adicional, ha permitido a Taiwán estar entre los primeros puestos en cuanto al desarrollo de productos con soporte de IPv6 (IPv6 Ready Logo Phase 1 y 2) [27]. Un caso similar ocurre en Estados Unidos [28], donde las

instituciones gubernamentales tienen como objetivo el poder manejar tráfico IPv6 a mediados del año 2008, proceso que es dirigido por el Departamento de Defensa de EEUU.

Figura 3-3. Estrategia de Activación de IPv6.

En Latinoamérica, se comenzó a trabajar en torno a IPv6 desde fines de los noventa. A fines del año 2005, en Latinoamérica existía cerca de un 4% del total de nodos IPv6 operando en el mundo. En Chile, en esa fecha se había completado la incorporación de IPv6 con soporte de multicast, uniéndose a la red global IPv6 (Figura 3-4), siendo los principales esfuerzos provenientes desde la Universidad Austral de Chile, la cual participó del proyecto global 6Bone, orientado al desarrollo y esparcimiento de IPv6.

Figura 3-4. Conexión a IPv6 en Latinoamérica.

Para incentivar, y también ordenar, el desarrollo de nuevas tecnologías (protocolos, equipamiento u otro) que funcionen bajo IPv6, el organismo IPv6 Forum [29] ha desarrollado el programa IPv6 Ready Logo Program [27], el cual consiste en un grupo de criterios que pueden ser usados para asegurar las características e interoperabilidad de productos IPv6. Este programa se divide en dos fases. La fase 1 revisa las funciones básicas y principales de los productos. La fase 2, apoyada sobre los resultados de la fase 1, agrega nuevas pruebas considerando el uso de IPsec y funciones de movilidad.

3.1.3 Mobile IPv6

El término movilidad abarca distintos puntos. Entre estos se encuentran que la entidad sea movible (capaz de moverse o ser movida), adaptable y versátil (cambiar su propósito) y migratoria (desplazarse habitualmente u ocasionalmente). La palabra movilidad estrictamente no significa inalámbrico, ya que redes cableadas pueden ofrecer movilidad. Además, la movilidad por si sola no representa un servicio, ya que su utilidad aparece cuando apoya a otros servicios. Hoy en día, este concepto está creciendo rápidamente en el ambiente de Internet, siendo principalmente apoyado por tecnologías como Mobile IPv4 y SIP, y próximamente por Mobile IPv6, pudiendo soportar innovadores conceptos como Personal Mobility, en la cual el usuario puede ser alcanzado en cualquier red y ubicación geográfica, redirigiendo sus comunicaciones a cualquiera de los dispositivos del usuario. De igual forma, los requerimientos para la movilidad se han puesto más exigentes. Entre estos destacan el no tener limitaciones geográficas (el dispositivo funciona en cualquier lugar), que sea independiente de la tecnología de red en uso (detectar un router local y conectarse a este) además de la habilidad de comunicarse con otros equipos que no soporten protocolos de movilidad (transparente a nivel de transporte y de aplicación). Sin embargo, el cambio frecuente en puntos de acceso producido en dispositivos móviles repercute directamente en las comunicaciones en curso, lo que hace poner un mayor esfuerzo en el desarrollo e implementación de movilidad en un ambiente real.

Mobile IPv6 (Figura 3-5) está diseñado para permitir al usuario el mantener las conexiones salientes mientras se mueve de una red a otra, siendo ubicable en su dirección IP original todo el tiempo. Funciona a nivel de la capa de red y fue desarrollado para enfrentar problemas

presentes en Mobile IPv4 como routing triangular e implementar el uso de rutas optimizadas a la vez de prestar mecanismos de seguridad por defecto [21].

En Mobile IP, cada nodo móvil tiene 2 direcciones IP. Una es la Home Address (HoA), la cual es usada para identificar al nodo móvil. La otra es una dirección foránea, la cual es usada para identificar la ubicación actual del nodo y manejar el tráfico entrante y saliente del dispositivo. La Home Address (HoA), dirección permanente del nodo móvil, es obtenida en su red de origen y es usada por las capas superiores de comunicación. La dirección foránea, también llamada, Care of Address (CoA), es obtenida en una red externa usando un mecanismo de auto configuración, mediante mensajes entregados por los routers, que permite entregar tanto la dirección IP, como otra información de configuración para la conexión en la red.

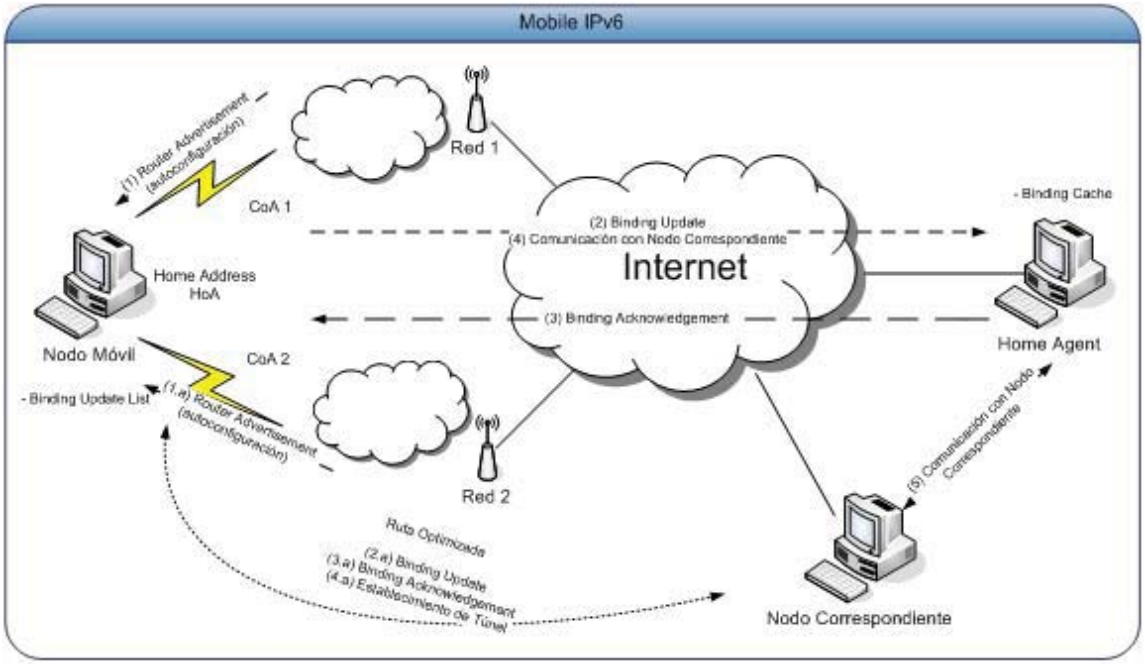


Figura 3-5. Mobile IP.

Cuando el nodo móvil entra en contacto con una infraestructura de acceso existente, este advierte la nueva red mediante los mensajes de Router Advertisement enviados por el router de acceso de ésta, por los cuales este es detectado y se inicia el proceso para adjuntarse, en forma temporal, a la red. El punto de salida o router original de la red, le asigna una dirección IP (o es autoconfigurada en el mismo nodo móvil) con la cual identificarlo y así permitirle realizar el proceso de registro con el HA, tras lo cual el nodo procede a establecer la comunicación a Internet a través de este nuevo enlace. Un mecanismo en desarrollo para

automatizar este proceso es DHCPv6, de forma que el dispositivo pueda ser configurado dinámicamente cuando el nodo se mueva a una red externa [30], el cual es un complemento a la propiedad de autoconfiguración presente en IPv6. El MN usa el proceso de Duplicate Address Detection (DAD) para asegurarse que la dirección asignada no está siendo usada por otro equipo en la red, el cual, aunque asegura una correcta configuración, incurre en un significativo retardo en la obtención de una nueva dirección y el comienzo de la transmisión. Técnicas como Optimistic Duplicate Address Detection (ODAD) tratan de mitigar completamente el efecto de la demora de la configuración de direcciones luego que el MN ha recibido un mensaje de aviso desde la nueva red.

La entidad dentro de la red de origen que realiza el manejo de las funciones de movilidad cuando el nodo esta lejos de su red, es conocido como Home Agent (HA). La presencia del nodo en su propia red es reconocida cuando el prefijo comunicado en la red, calza con el prefijo de la HoA perteneciente al nodo móvil. Generalmente, los prefijos advertidos para las redes internas de una organización son de un largo de 64 bits [31].

En Mobile IP, un agente Foráneo (FA, por su nombre en inglés: Foreign Agent) es conocido como la entidad en la red externa que ayuda al nodo móvil con las funciones de movilidad. Este concepto no es usado como tal en Mobile IPv6, en donde, el mismo nodo puede realizar este proceso. Ya sea usando al FA o no, tiene que ser necesario el registro con la HA enviando un mensaje llamado Binding Update (BU), de tal forma que el HA pueda rastrear la ubicación del nodo móvil (registro de la nueva CoA adquirida) y administrar la entrega de paquetes.

La entidad que se está comunicando con el nodo móvil es conocida como nodo correspondiente (o Correspondent Node, CN). Cuando el nodo móvil se encuentra fuera de su red, se comunica con esta mediante un túnel virtual bi-direccional creado entre el HA y la CoA adquirida en la nueva red. Esto permite ocultar el movimiento del nodo de los CN y de otros dispositivos vinculados con el nodo móvil, los cuales no necesitan tener capacidades adicionales para manejar la movilidad, ya que este proceso es manipulado completamente por el MN haciéndolo transparente para estos. En un principio toda comunicación entre el nodo móvil y el CN es realizado a través del HA, usando la HoA del dispositivo móvil, haciendo también transparente el movimiento a las capas de transporte y superiores.

Es posible establecer una vía de comunicación entre el MN y el CN principalmente mediante dos formas. En la primera, los paquetes enviados desde el CN son dirigidos en primer lugar hacia el HA y desde allí enviados al MN, ya que estos son interceptados mediante el proceso de Neighbor Discovery ejecutado por este dispositivo (HA) y luego enviados a la CoA respectiva que se tenga registrada para ese MN. El proceso inverso tiene la misma ruta. La segunda forma es usando la ruta optimizada (enfocada a resolver el problema ejemplificado con el caso de dos japoneses conectados en América) y requiere que el MN registre sus enlaces, además de registrarlos en el HA, en el CN, quien los guarda en su respectivo Binding Cache, lo que permite que los paquetes puedan ser dirigidos directamente a la CoA del MN. En la transmisión es agregado otro header al paquete conteniendo la HoA de forma de hacer transparente el ruteo para las capas superiores a la de red. De igual forma, el MN mantiene registrado las ubicaciones de los CN con los que ha establecido una comunicación de forma de enviar directamente los paquetes, sin pasar por el HA. En esta situación, es preciso que tanto el CN como el HA sean compatibles con MIPv6, ya que es necesario que manejen el registro de los enlaces. Como contraparte, es necesario mencionar que el uso de la ruta optimizada introduce una mayor latencia durante el periodo de handoff del nodo móvil, debido al proceso de registro de la nueva dirección. Existen extensiones para MIPv6, como HMIPv6 o FMIPv6 que proveen soporte para realizar handoff rápidos, reduciendo los tiempos de pérdida de señal y paquetes perdidos [32], mientras se tienen establecidas sesiones, pero que necesitan la introducción de nueva infraestructura a la red. Como en casos esta latencia introducida por los procesos de registro es tan amplia a nivel de transmisión, que para mitigar sus efectos, en el presente trabajo se utiliza la idea de registro de múltiples enlaces, los cuales pueden ser utilizados en forma simultanea.

De igual manera, existen trabajos de manera de juntar el protocolo SIP (Session Initiation Protocol) con MIPv6, de forma de extender las funcionalidades de manejo de sesiones multimedia a conexiones móviles, para adecuarse a nuevos requerimientos en las trasmisiones como la movilidad, ya sea personal o de dispositivos, conexiones a nivel global, independiente de la tecnología usada además de soportar aplicaciones en tiempo real.

3.1.4 Nodo o Router Móvil

En cuanto a las características que deben poseer para soportar la arquitectura a estudiar, ambas entidades deben tener las mismas propiedades. No obstante, el uso de un router móvil

(o MR por sus siglas en inglés) puede ofrecer características adicionales debido a su diseño, como por ejemplo, el mantener toda una red bajo sí (esto sucede en NEMO o Network Mobility), a la vez de necesitar que sólo la conexión IPv6 del MR cambie su enlace con el exterior, y no todos los puntos que conforman la red, ya que este equipo maneja la movilidad de todos los nodos interiores, los cuales no necesitan mayores capacidades de administración de movilidad para hacer uso del acceso a Internet.

Este dispositivo es el encargado de enviar la información necesaria al HA para que se mantenga la conexión correcta a través de los distintas redes, y CoAs, en las que se encuentre adjunto. Mantiene una estructura llamada Lista de Binding Update, o lista de enlaces, la cual registra toda la información enviada en los mensajes de BU, cuando el nodo se mueve, de forma de mantener sincronización en las comunicaciones a través del HA. Al momento de registro, debe enviar su identificación al HA en un mensaje de BU cuando se adjunte a una red externa, por lo que se podrá establecer un túnel entre ambos dispositivos. Cuando el MR recibe paquetes encapsulados desde el HA, este debe reenviarlos al nodo móvil adjunto a este router. De igual forma, si recibe paquetes de parte de los nodos de la red, el MR debe encapsularlos y enviarlos a través del túnel de acuerdo a las políticas definidas. Algunas funciones adicionales al proceso de los BU, incluyen el proceso de errores, re-establecimiento de túneles bidireccionales, descubrimiento de nuevas rutas, entre otras.

Es posible que distintos nodos se unan dinámicamente al espacio donde se encuentra el nodo/red móvil. Algunos de estos pueden tener su propio acceso a Internet, posiblemente a través de distintos tipos de acceso a la red o distintos tipos de tecnologías. Entonces, si se encontrase una forma de coordinar y comunicar estos nodos, sería posible el usar tal acceso a Internet, de forma de mejorar el acceso inicial que se posee. Una manera de coordinación entre distintas entidades actuando como nodo o router móviles se encuentra en [33], donde se explican los mecanismos de comunicación, estructuras y de enlaces adicionales, particularmente para ambientes bajo NEMO. En caso que sólo sea un dispositivo la entidad móvil, y no una red completa, el nodo crea enlaces con estos MR adicionales mediante las interfaces presentes en el MN.

Un caso de ejemplo lo constituye el acceso a Internet usando la conexión GPRS/EDGE/GSM de un teléfono móvil por un notebook a través de direccionamiento IP, pudiendo obtener las ventajas de este tipo de enlace, como por ejemplo, su amplio rango de cobertura que es a nivel

de kilómetros permitiendo el establecimiento de una conexión constante a través, de por ejemplo, una ciudad entera. En este caso, la conexión entre ambos dispositivos es a través del estándar Bluetooth, que permite la comunicación entre dispositivos dentro de un rango determinado sin establecer una conexión física entre ambos equipos. De esta forma, el enlace establecido por otro dispositivo móvil puede ser usado como una nueva vía de comunicación.

3.1.5 Home Agent

El HA debe mantener un Binding Cache para cada nodo móvil o MR que está actualmente registrado. El formato del Binding Cache es similar al de la lista de enlaces que mantiene el MR. Debe existir una tabla de direcciones de forma de poder prevenir el tener dos dispositivos que posean la misma denominación (dirección u otro prefijo). Esta tabla es revisada cuando se recibe un mensaje de Binding Update. Si el mensaje supera el proceso de revisión, el HA crea una nueva entrada en su Binding Cache de acuerdo a la información contenida en el mensaje recibido y envía de vuelta un mensaje de confirmación conocido como Binding Acknowledgement, estableciendo el túnel MN-HA para redirigir el tráfico.

Este proceso, permite al MN confirmar que el nodo destinatario de sus mensajes para el establecimiento de enlaces soporta el mismo protocolo de comunicación, en este caso Mobile IPv6. Así, que en el caso de no recibir respuestas a estos mensajes, u obtenerlos junto a un mensaje de error, el MN debe establecer una comunicación con las opciones básicas del protocolo y no usar nuevas opciones para el manejo del tráfico en los mensajes de registro enviados. En el caso que el HA reciba un paquete proveniente del CN, el HA debe comparar la dirección de destino con sus entradas en su Binding Cache. Si la dirección calza con alguna de las entradas, la CoA indicada será seleccionada. Como resultado, el paquete es enviado a esta dirección. Cuando un paquete es recibido, el HA remueve el encabezado exterior y reenvía el paquete a su dirección final.

3.1.6 Handoff

Se conoce como handoff al proceso cuando un nodo móvil (MN o MR) realiza un cambio en su punto de enlace a la red. Este efecto puede ocurrir principalmente en dos circunstancias. La primera, o hard handoff, es cuando se pierde contacto con su router de acceso local (AR), siendo inalcanzable en la CoA asignada, hasta que se conecta a un nuevo AR y obtiene una

nueva CoA. En el otro caso, o soft handoff, el nodo móvil se conecta a un AR cuando todavía tiene conexión con un anterior AR. El segundo caso es el que menos problemas trae en cuanto a pérdida de paquetes, ya que el tiempo sin conexión es mínimo y el cambio de CoA es manejada en el mismo equipo, ya que ambas se encuentran registradas. Aún más, en el caso que el dispositivo móvil disponga de más de una interfaz de comunicación, las pérdidas de paquetes pueden hasta ser eliminadas de la comunicación, ya que, por ejemplo, mientras una interfaz está activa, la otra puede estar en busca de nuevos puntos de acceso y adjuntarse a estos, entregando un respaldo a la transmisión. Este es uno de los beneficios a estudiar e implementar en el presente trabajo.

En general, la latencia (considerada como RTT más el tiempo de registro entre el HA con el MR y la obtención de una dirección válida) puede considerarse insignificante en el proceso de handoff entre redes externas donde se tiene acceso a tecnologías de gran alcance con altas tasas de transmisión como WiMax, ya que el tiempo (RTT) entre el MR y el HA permanece pequeño, cuando ambos dispositivos se encuentren bajo la cobertura de esta. Sin embargo, cuando no se cuente con este tipo de tecnología, o se trate de escenarios de mayor tamaño, el valor del RTT afectará el tiempo total de latencia, dados los distintos enlaces que se han de establecer.

3.2 MULTIHOMING

3.2.1 Explicación

Un nodo, o una red, se conocen como multihomed [34, 35] cuando este puede estar conectado vía varias interfaces o routers móviles (MR), varias tecnologías de acceso y varias redes de acceso. En el caso que se encuentren presentes varias interfaces, o varios MR, estos deben encontrarse en funcionamiento simultáneo. Esta estructura debe poder soportar los cambios de medio de acceso conocidos como handover. Estos pueden ser horizontales, si es cambiado el punto de acceso a la red externa, como verticales, que ocurren cuando es cambiado el medio de comunicación usado por una conexión.

El poseer distintos puntos de conexión a la red permite que los objetivos propuestos sean satisfechos correctamente, siendo este un requisito necesario para la arquitectura a construir. En [16], se ha desarrollado una forma de registro de múltiples enlaces, o CoAs, con una

misma entidad, lo que facilita el hecho de poder contar con conexiones de respaldo en ambientes de movimiento o que se pueda dividir el tráfico mediante algún criterio entre los múltiples registros entre el MN y el HA o CN. Este punto es importante, ya que el contar con múltiples puntos de enlace con la red, ya sea interna o externa a la organización, es la base para poder realizar la aplicación de políticas y distribución de tráfico sobre las distintas conexiones establecidas.

3.2.2 Beneficios

El uso de multihoming dentro de una red, ya sea cableada o inalámbrica, permite el obtener diversos beneficios. Los principales y más comentados en la literatura son:

Balanceo de carga: Con conexiones múltiples a Internet en forma simultánea, el nodo móvil puede recibir y enviar información por distintos caminos. De esto es posible obtener tanto distribución de carga (separando un flujo de información en distintas conexiones) como balanceo de carga (distribuir las distintas conexiones en las distintas conexiones disponibles). El tener información como el ancho de banda disponible o el nivel de congestión de la red es de relevancia al momento de decidir la forma de separar el tráfico.

Acceso permanente e ininterrumpido: El nodo, usando ya sea múltiples routers o interfaces, tiene la posibilidad de hacer uso de diferentes tecnologías de acceso en diferentes tiempos de forma de asegurar una conectividad continua.

Redundancia o tolerancia a fallas: El hecho de tener un nivel de redundancia, a través de distintas interfaces o MRs, el nodo móvil puede mantener un gran nivel de resistencia ante fallas en la red. Por ejemplo, es posible cambiar las transmisiones salientes de un router a otro, haciendo este cambio transparente a nivel de la aplicación. Es necesario tener distintos procedimientos que permitan completar este punto, como detección de caminos caídos, descubrimiento de nuevas rutas y registro de nuevas CoAs.

Aumento de ancho de banda: Dado el hecho de poder acceder a distintos medios de acceso, unos con mejores características que otros, es posible que la suma de la tasa de transferencia total sea mayor a la de una sola conexión.

Ruteo basado en políticas: Lo que permite que la ruta al nodo móvil sea decidida basada en métricas definidas por el usuario, como costo, eficiencia, calidad u otros. Esta política puede ser estática o dinámica, a la vez de poder ser iniciada tanto en el nodo móvil, en el MR, o por el HA, permitiendo que la optimización de las transmisiones sea en ambos sentidos.

3.2.3 Puntos Considerados

Preservación de sesiones: Si un túnel establecido falla, la ruta seguida por los paquetes debe ser cambiada sin afectar las sesiones salientes establecidas por las capas superiores a la capa IP. Esto puede lograrse encapsulando los paquetes antes de ser enviados, de una interfaz a otra, o de un MR a otro, si se detecta alguna falla en la ruta establecida. Lo anterior permite mantener las direcciones originales de origen y destino de los paquetes.

Ruteo por flujo en vez de por paquete: Esta última forma trae problemas en cuanto a la transmisión TCP debido, por ejemplo, a la congestión adicional debido al reordenamiento de paquetes dado que los enlaces generalmente tienen distintas características.

Establecimiento de múltiples caminos: Es necesario definir como las múltiples rutas entre la estación móvil y el HA serán definidas. Esto será tratado con el uso de tópicos como filtros, múltiples interfaces y MR, registro de múltiples CoAs y filtros con el HA.

3.3 FILTROS

El uso de filtros [36, 37] permite al nodo móvil optimizar y adecuar la forma de manejar el tráfico en la red, permitiendo asignar tráfico específico en una forma determinada mediante la aplicación de políticas a enlaces establecidos. Con esto se logra enlazar flujos o transmisiones particulares provenientes de una CoA registrada por el MN, sin afectar el comportamiento de los otros enlaces que se posean. Esto es logrado, ya que los filtros permiten identificar un flujo e identificar la correspondiente información de control que permite tomar una acción definida. Para esto es necesario contar con reglas o criterios de filtros los cuales podrán ser aplicados sobre el tipo de tráfico. De esta forma es posible distribuir, eliminar o manejar tráfico en una forma alternativa.

IPv6 presenta la capacidad de etiquetar tanto los flujos de transmisiones como las clases a las que pertenecen, a través de dos campos presentes en los paquetes que permite identificarlos, para adecuarse a requerimientos establecidos de QoS, ya sean referente los conceptos de int-serv (por flujos) o de diff-serv (por clase). Este campo es usado junto a la dirección de origen de los paquetes de manera de aplicar correctamente los filtros construidos.

Basado en el concepto de multihoming, los filtros (Figura 3-6) nos permiten distribuir el flujo entre los distintos puntos de enlace a la red, en una forma eficiente. Flujos diferentes pueden ser asignados, a nivel de aplicación, basados en las necesidades o prioridades requeridas, haciendo posible el tener distribuidos distintos flujos a través de distintas interfaces de comunicación.

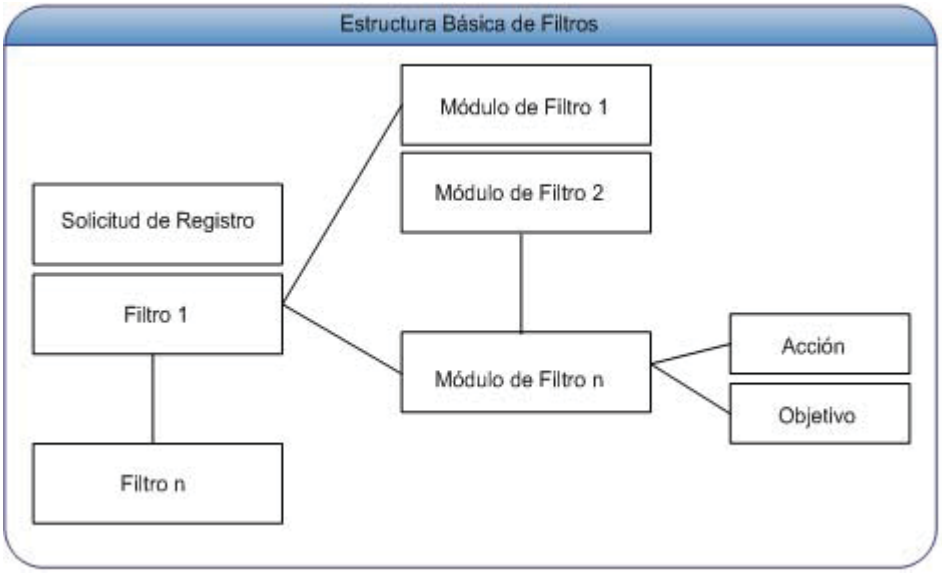


Figura 3-6. Estructura de filtros.

Básicamente los filtros están compuestos por 3 componentes:

Agente de filtrado: Cualquier agente que maneja criterios de filtrado sobre sus enlaces móviles.

Módulo de filtro: Es un criterio de filtrado que especifica la condición a chequear sobre el tráfico IP.

Filtro: Es una colección de módulos de filtro, el que además debe tener asociado un grupo de información para administrar el filtrado, como por ejemplo, una acción a

tomar si el tráfico satisface una cierta regla de filtrado, o una prioridad asignada a las reglas con tal de tener un criterio para seleccionar si varias aplican a un cierto caso.

Para poder lograr este cometido es necesario que el agente encargado de la movilidad en el nodo tenga definido reglas y políticas para decidir los criterios de filtro y los tipos de datos que deben ser comunicados entre las entidades participantes en la comunicación.

3.4 BINDING UPDATE

3.4.1 Explicación

a. Binding Update

El mensaje de Binding Update (BU) es usado por el MN para registrar con el HA nuevos CoAs adquiridos (Figura 3-7). La nueva CoA es enviada en el mensaje en el campo Mobility Options. Es necesario mencionar que en este campo es posible enviar distintos tipos de mensajes en el momento del registro, pudiendo no ser necesario que la entidad recibidora de estos, sea el HA, CN u otro, los reconozca como válidos. El formato del mensaje corresponde a la siguiente estructura, según ha sido definido en el soporte de movilidad para IPv6:

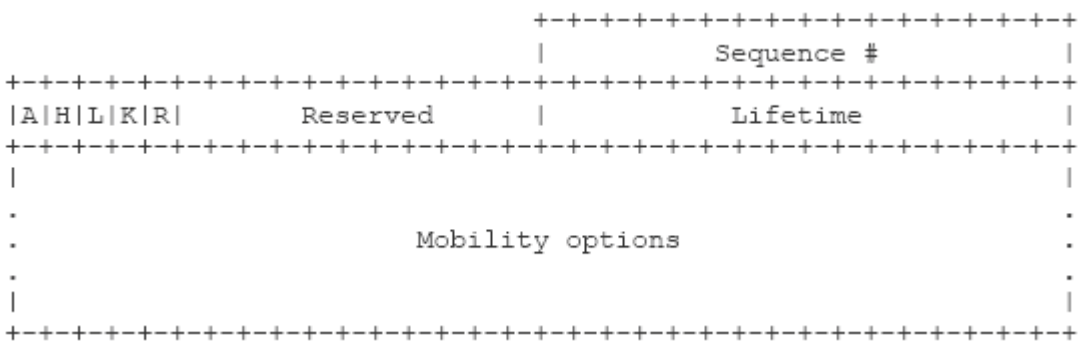


Figura 3-7. Estructura de BU.

El mensaje de Binding Acknowledgement (BA) indica la confirmación del registro del dispositivo móvil con su HA (Figura 3-8). En este mensaje, el bit R es también usado como en el caso de los mensajes de BU, para indicar si corresponde a un nodo o a un MR. Además el bit A, en el mensaje de BU, es activado si el MN requiere un mensaje de Binding Acknowledgement, para confirmar el registro. En el caso en que la opción adjunta en el mensaje de BU no sea reconocida, el mensaje de Acknowledgement no es devuelto o es devuelto un mensaje de error de forma de indicar al MN no seguir enviando este tipo de

mensajes. Por último el campo Lifetime es usado para indicar el tiempo que el registro es válido, y si es seteado a 0, significa la eliminación de éste, o parte de éste, por el HA.

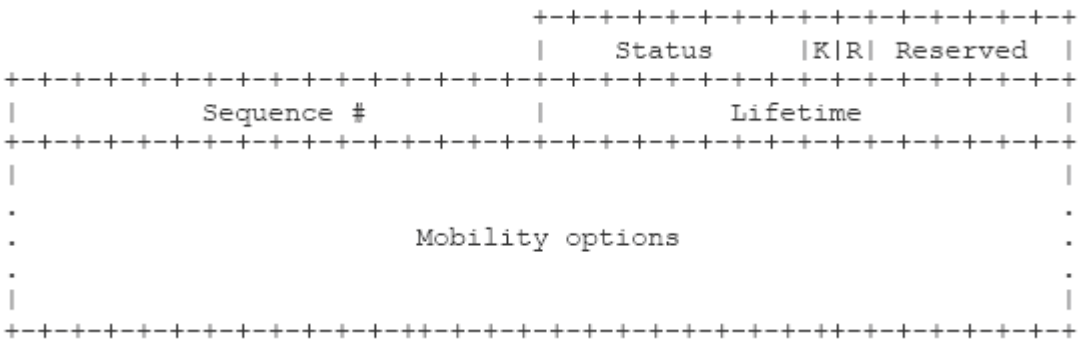


Figura 3-8. Estructura de BA.

Ahora, en el caso que distintas CoAs sean adquiridas, es necesaria la inclusión de un nuevo parámetro en las tablas de registro de los enlaces para poder utilizar y manejar las distintas conexiones establecidas. En los trabajos que se han venido realizando sobre el tema se propone el uso de un identificador especial llamado Binding Identifier (o BID) para cada entrada (CoA) que se registre **[16]**, el cual debe ser comunicado tanto al HA como al CN, si es posible usar un ruteo optimizado. Este punto se tratará en la sección 3.4.2.

b. Binding Cache

Es la estructura encargada de mantener los registros de los enlaces actualmente establecidos por el nodo móvil. Debe ser mantenido tanto por el CN como por el HA. Mantiene campos como:

CoA y HoA para los enlaces establecidos por un MN.
Lifetime para cada entrada.
Otra información relativa al enlace que será especificada en los siguientes puntos.

Los nodos móviles, a su vez mantienen una lista adicional de binding update, la cual permite optimizar los tiempos de handoff, debido a que es actualizada en forma más constante que el binding cache y contiene mayor información que esta.

3.4.2 Requerimientos para Multihoming

En el proceso de solicitud de registro ejecutado por el MN, este indica al HA y al CN, si existiese, que desea establecer múltiples enlaces, así que estos son agregados y el enlace actual existente no es actualizado.

Durante el periodo de uso del nodo móvil, las interfaces pueden moverse desde una red de acceso a otra. Cuando este cambio topológico de vínculo a la red se produce, la nueva CoA adquirida es enviada al HA y al CN, usando mensajes de BU. Además, es posible que varias interfaces puedan comunicarse simultáneamente. Cada interfaz tiene configurada una CoA propia. Para soportar esta estructura de transmisión, es necesaria que la forma de ruteo sea modificada para cumplir los requerimientos de movilidad en multihoming y de selección de interfaces.

Para establecer distintos flujos entre el MN y el HA, es necesario modificar el proceso de registro del binding update. También es necesario extender el almacenamiento de la información (binding cache) para almacenar estos nuevos datos. Esto es conocido también como registro de múltiples CoAs. En este proceso es necesario asociar un número (BID) a cada enlace que se desee registrar entre el MN (uso de COA o de interfaz) y HA. Estos valores son enviados por el MN incluidos como una opción en el mensaje de BU (Figura 3-9) y son almacenados tanto en el Binding List como en el Binding Cache. Si los valores a registrar ya se encuentran en el HA, estos deben ser actualizados según los nuevos valores, sino, una nueva entrada debe ser creada. En el mensaje del registro de la opción de Binding ID, algunos valores útiles a incluir son el BID, la prioridad del enlace registrado y la CoA a registrar.

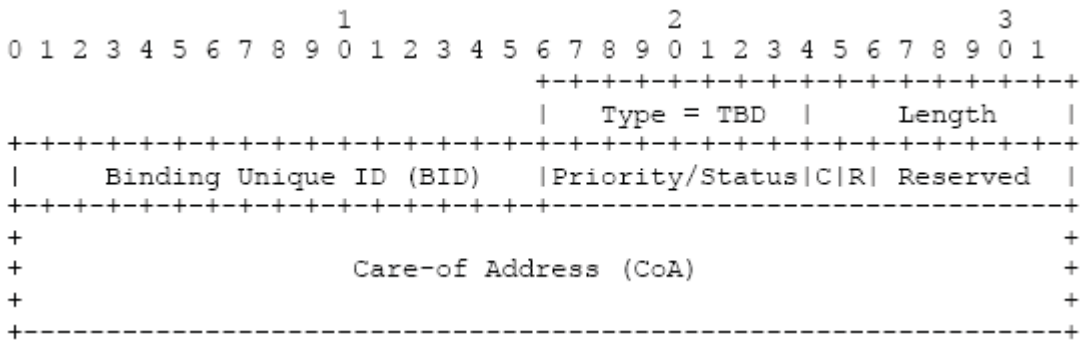


Figura 3-9. Registro para BID.

El mensaje puede ser usado para registrar una CoA como varias a la vez. Con esto se logra el registro de distintos enlaces para una HoA dada. Es posible manejar el uso de prioridades entre los distintos enlaces (campo en el mensaje) para indicar por cual devolver información desde el HA.

Por la anterior, surgen datos que son necesarios de mantener para un correcto control de los enlaces disponibles en un momento dado, y por lo que deben estar presentes de alguna manera en el registro de ese enlace. Por un lado, se tiene la información del BID como tal, que, junto a la dirección registrada, permite identificar un enlace en particular. Además, como se tienen varias alternativas de comunicación, el otro valor importante corresponde a la prioridad indicada en el mensaje de BU para un determinado BID. Estos valores son necesarios que queden tanto en el Binding List que posee el MN como en los Binding Cache de las entidades finales del enlace, es decir, el CN y el HA.

3.4.3 Requerimientos para Filtros

En la literatura **[36]**, se introduce el uso de la opción de identificador de filtros (Flow ID), la cual es incluida tanto en el mensaje de BU para su registro como en el mensaje de Binding Acknowledgement para la confirmación, y que permite identificar una conexión en particular. Esta opción permite al nodo móvil unir varios flujos de tráfico a una CoA mientras se mantiene la conexión de otros flujos en otras CoAs. Cuando no hay filtros definidos o políticas aplicables a un cierto caso, toda la información es enviada por la CoA por defecto que se encuentre registrada en el nodo.

La opción contiene información que permite al receptor del mensaje de BU el identificar los flujos de tráfico y rutearlos a una dirección dada. Varias de estas opciones pueden ser agregadas dentro de un solo mensaje de BU. En la solicitud de registro se envían extensiones relacionadas con los filtros específicos. Los filtros incluyen información como protocolo, puerto de origen, fuente, otros.

La opción de Flow ID (Figura 3-10) en el mensaje, propuesto en la literatura, tiene un formato flexible con tal de dar una mayor libertad a la forma de representar el flujo. Algunos campos posibles a incluir son información del origen y destino; información de la CoA por defecto a usar; la prioridad asignada al enlace establecido; el FID asignado; acción a tomar y tipo de

protocolo de la transmisión, entre otros. El mensaje debe ser enviado constantemente para actualizar el tiempo de vida de la asociación. De igual forma, este mensaje es compatible con los enviados para obtener el registro de múltiples CoAs y permite la asociación de los FID con los BID ya almacenados en el HA como en el CN, los cuales tienen que estar previamente registrados en estos dispositivos. Esto permite, por ejemplo, el poder cambiar todos los flujos (FID) registrados con una CoA a otra, en el caso que el MN se encuentre en desplazamiento.

Ya que el incluir los parámetros para medir y ordenar las conexiones, como ancho de banda, tiempo de demora o de ida y vuelta, no tiene un verdadero valor sino existe una forma de interpretarlo o sacarle utilidad, el parámetro o valor indicador de prioridad juega un papel importante al momento de seleccionar un enlace. Así que este parámetro es uno de los que permite obtener el beneficio de transmitir algún tipo de tráfico específico que se considere de importancia por el mejor enlace posible, dejando que los otros tipos de tráfico sigan otro camino de menores prestaciones.

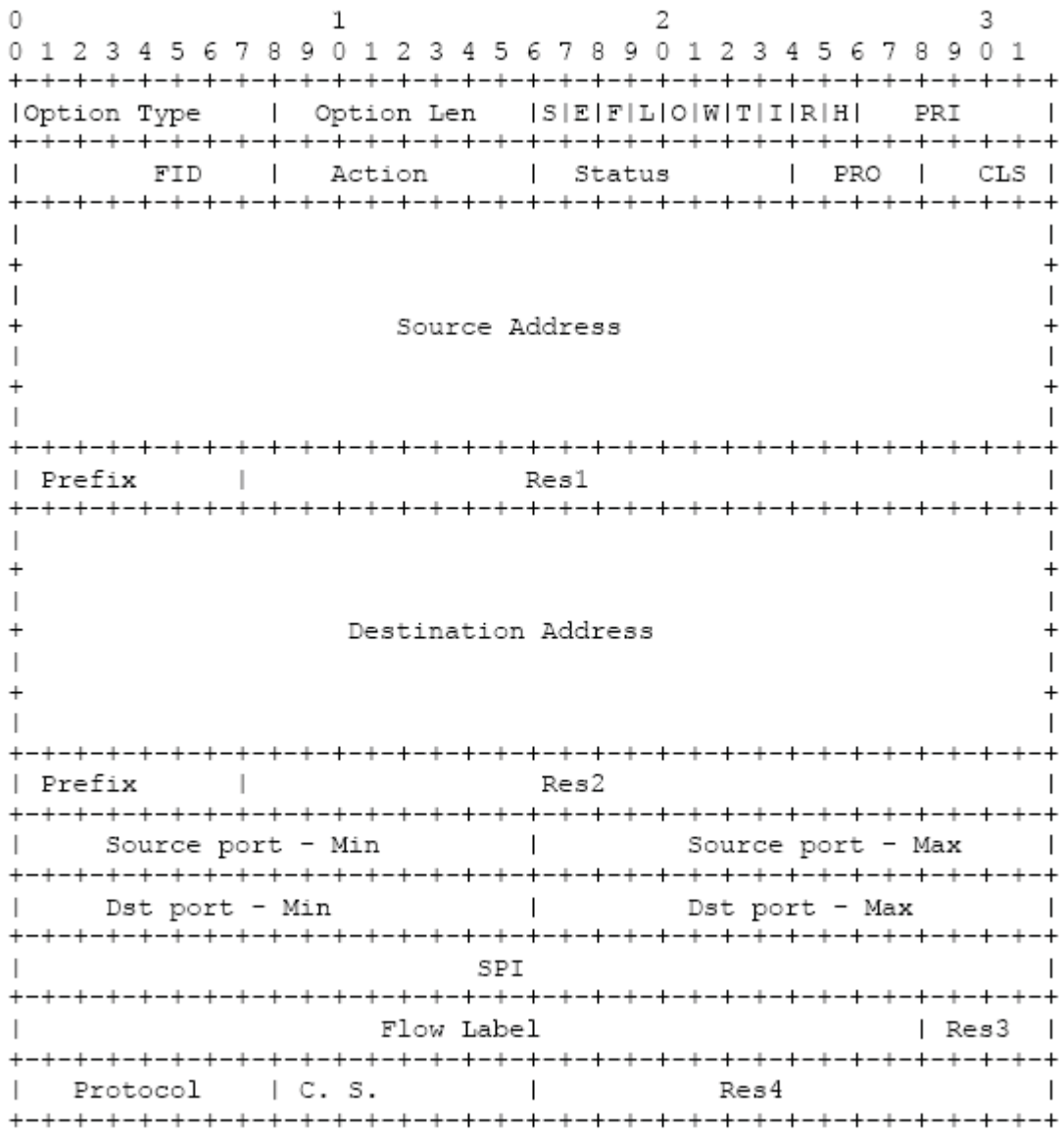


Figura 3-10. Registro para Filtro.

Tanto el Binding Cache como el Binding List tienen que ser adaptados para soportar distintos enlaces de un mismo dispositivo, quizás mismo CoA, con el mismo HA. En el caso de los flujos, se mencionó que en la opción usada en el mensaje de BU para su registro puede ir variada información la cual puede ser potencialmente utilizada por las políticas implementadas para mejorar el proceso de transmisión, por lo que deben ser almacenadas de alguna forma. Pero el elemento que debe estar claramente especificado en los registros de enlace es el FID que tiene que encontrarse relacionado con la HoA obtenida por el MN o MR que tiene el enlace, de forma que es posible que para la misma HoA se encuentren distintas combinaciones de flujos y CoAs. Otros campos que deben estar presentes para una correcta división son:

Prioridad de elección.
Dirección origen / destino.
Rango de Puertos usados.
Protocolo

3.5 POLÍTICAS

3.5.1 General

El uso de multihoming, especialmente en un ambiente inalámbrico, requiere administración y control de las conexiones establecidas para ser totalmente funcional. Las entradas para este control vienen de distintas fuentes. Por ej: del usuario, de la aplicación de la red, de las interfaces, entre otras. Cuando el equipo tiene múltiples rutas disponibles para enviar paquetes a sus pares, este debe de alguna manera tomar una decisión sobre que camino usar para tal conexión. En forma más específica, el equipo necesita una política de selección de la IP de origen (y posiblemente el puerto) y la interfaz de salida para enviar los paquetes desde y hacia el MN a través de los distintos enlaces establecidos **[38].**

El contenido de una política esta básicamente formada por un evento, condición y una acción. Las políticas pueden ser implementadas en varias capas, como la de IP, sesión o aplicación. Pueden ser usadas para asignar flujos, selección de interfaces, otros. Para reducir el proceso de administración de estas y mantener a las entidades participantes sincronizadas, es necesario contar con un mecanismo de distribución de políticas.

La operación del mecanismo de selección de interfaces tiene que tener en cuenta que la información de la red puede cambiar en cualquier momento. Por lo tanto, debe existir una estructura de políticas y mecanismos que permitan mantener las reglas actualizadas para el proceso de selección. Las políticas proveen a las diferentes entidades de red la posibilidad de controlar la ubicación de los flujos de tráfico del nodo móvil en diferentes interfaces. Las políticas que apoyan este proceso describen las preferencias de diferentes interfaces de red en varias situaciones. Basado en las políticas, el mecanismo de ruteo puede enviar los paquetes por distintas interfaces. Generalmente, la interfaz más preferida es siempre usada, y si esta no estuviese disponible, las conexiones pueden ser reubicadas en la interfaz que sigue en

preferencia. Adicional a esto, hay otros criterios de elección como por ejemplo del tipo de dirección o enlace adquirido en la interfaz (por ejemplo si se refiere a un HoA o CoA, si la dirección es privada o pública, entre otros criterios), condición que también debe influir al momento de seleccionar la ruta de salida del tráfico.

El usuario debe tener permitido actualizar una política basado en esta información o en criterios externos a los recolectados en la red. Esta actualización puede ser manual como automática, dependiendo del tipo de implementación a desarrollar.

Algunos puntos respecto a las políticas son:

Ejemplos de acciones a tomar: Según tipo de flujo, interfaz, tipo de protocolo.
Prioridad de acciones: Necesario establecer una por defecto e indicar el orden de búsqueda entre las restantes. Las acciones en la política deben tener alguna prioridad, la cual permite definir el orden en el cual las acciones serán buscadas y utilizadas.
Distribución de políticas: Ubicadas tanto en nodo móvil como en HA o en router intermedio. Es necesario indicar como estas son divididas y los componentes que intervienen en los repositorios de políticas. Hay que evitar los posibles choques de acciones.

3.5.2 Alternativas de Implementación

En la literatura existen dos referentes relacionados al manejo y almacenamiento de políticas, las cuales proporcionan el marco base para la construcción de la arquitectura de filtrado propuesta.

a. RFC 3484 [39].

Presenta una estructura de tabla de políticas para usarla para realizar búsquedas. Esta tabla contiene tres campos: el campo dirección, el campo precedencia y el campo de etiqueta. Dada una dirección IP, se accede a la tabla con una búsqueda que utiliza un cierto criterio, y retorna dos valores asociados con la dirección en cuestión, la precedencia y la etiqueta. El campo de precedencia se utiliza para elegir entre varias direcciones destino. El campo de etiqueta es usado para elegir una dirección origen para cierta dirección destino. Cuando la etiqueta de la

dirección destino coincide con la etiqueta de la dirección origen, el algoritmo selecciona la dirección origen en cuestión para ser incluida cuando se envían paquetes a la dirección de destino involucrada.

En su forma actual, la tabla de políticas permite definir políticas a nivel de dirección IP, pero es posible extender este nivel para tener en cuenta información adicional (a nivel de puerto a utilizar en comunicación TCP o UDP). Esta tabla puede ser incluida, mediante distribución, dentro de cada nodo que forma parte de la solución, de forma de ser más fácil la discriminación por política y aplicaciones. Este formato es una guía para la forma de ordenar los campos y las respectivas conexiones establecidas, ya que entrega una forma eficiente de realizar búsquedas y seleccionar políticas. Esto es logrado ya que proporciona guías para seleccionar las conexiones y direcciones de origen y destino en base a la información adquirida y criterios preformados (basados en el ámbito de las direcciones obtenidas, en el tipo de direccionamiento, entre otros criterios) además de permitir la inclusión de políticas definidas por el usuario. Como punto adicional permite el manejo de direccionamiento tanto en IPv6 como en IPv4.

b. Framework de Políticas (RFC 2753 [40], RFC 3198 [41]).

Las reglas de políticas descritas en la forma de modelos de políticas son transformadas en configuraciones en el equipo de red (nodo móvil, MR, otro). En este marco, dos elementos sobresalen en el control de las políticas: PEP (Policy Enforcement Point) y PDP (Policy Decision Point). PEP es el componente encargado de tratar directamente con los paquetes y es responsable de la ejecución de las acciones de las políticas, como por ejemplo, filtrado, marcado de paquetes, tasa de envío, entre otros. PDP es responsable para determinar cuales acciones son aplicables a que paquetes.

Otro componente importante es el Environment Detector o Detector de Ambiente que es responsable de administrar todos los tipos de información útiles, como monitoreo, actualización y almacenamiento de estos. Durante el periodo de administración, el detector de ambiente detecta todos los cambios que llevan a la creación de nuevos eventos y es posible enviar una notificación al PEP para la aplicación de una política especial el cual la envía posteriormente al PDP. Este proceso también puede ser llevado dinámicamente, como en el caso en que el PDP realice un cambio de forma que este informa automáticamente al PEP

para que actualice algún criterio, sin necesidad de que este inicie la comunicación. Existen protocolos de pregunta y respuesta para acceder a repositorios de políticas, como es el caso de Common Open Policy Service y de Lightweight Access Directory Protocol.

En este marco, tanto PDP y PEP como los detectores de ambiente pueden ser colocados en un dispositivo, o ubicados en forma distribuida en múltiples dispositivos.

El Repositorio de Políticas es la ubicación donde las políticas definidas son almacenadas. El repositorio, al igual que los otros componentes, puede ser ubicado en un sólo dispositivo, o replicado en varios dispositivos. Es necesario el establecimiento de formas de evitar conflictos entre políticas que pueden ser aplicadas al mismo tiempo debido a una respuesta a un evento en tiempo real.

Para lo anterior, resulta necesario contar con un mecanismo de distribución de tablas de políticas que permitan la configuración automática de las tablas de los nodos de un sitio multihomed. Algunas formas de realizar esto es mediante:

Router Advertisement (RA).
DHCP.

Este marco será usado para organizar la forma en que las políticas serán almacenadas y transmitidas para su aplicación, ahora vistas como componentes interrelacionados mediante mensajes de consultas y respuestas entre los distintos módulos presentados.

Uniendo los conceptos anteriores, las políticas se encuentran formadas por:

Entidades que definen acciones. Una entidad puede ser un usuario, un nodo, o un ente externo.
Acción es una operación que es definida por una entidad o es controlada por el sistema. Las acciones especifican interfaces que pueden ser usadas por las conexiones siguiendo lo requerido por las entidades. Las acciones pueden ser expuestas como estamentos condicionales. La cláusula condicional dentro de una acción consiste en un conjunto de atributos que son evaluados contra la información de la conexión. Si existiese un calce, la interfaz es buscada por orden según preferencia. Cualquier acción, excepto aquella por defecto, debe incluir información como: a) un número de acción, b) un parámetro

indicando si la acción debe ser forzada, c) conjunto de atributos que contienen información de la conexión, d) una lista de los tipos o características de las interfaces y el orden de preferencia. En el caso que ninguna acción calce con el estado de la red, existe una acción por defecto para las interfaces que es ejecutada en estos casos.

Políticas que rigen las acciones de una entidad. Sólo una acción puede tomar lugar en un cierto momento en una política. Un grupo de políticas contiene varias políticas posibles definidas por diferentes entidades (Figura 3-11). En la definición de las políticas debe estar especificada la prioridad entre estas y las acciones a tomar. Por ejemplo, si alguna de las interfaces en uso, se volviese inalcanzable o se encuentre fuera de los parámetros requeridos, el módulo de políticas tiene la misión de localizar la acción relacionada a este caso y seleccionar la siguiente mejor interfaz a usar con la consiguiente modificación de la tabla de ruteo.

Credenciales que son usadas para autorizar acciones que son definidas por entidades diferentes.

Mecanismos, los cuales evalúan acciones contra la información relacionada a la conexión y decide cual interfaz/puerto/tipo de protocolo tiene que ser usada con una conexión especifica. La evaluación de las políticas debe resultar siempre en la selección de una sola interfaz para un tipo de flujo o conexión. Esto es logrado por la prioridad especificada en las acciones. El mecanismo selecciona una interfaz basada en este orden de prioridades. Cuando una acción es actualizada durante la operación, esta afecta todas las conexiones activas relacionadas a esta acción específica.

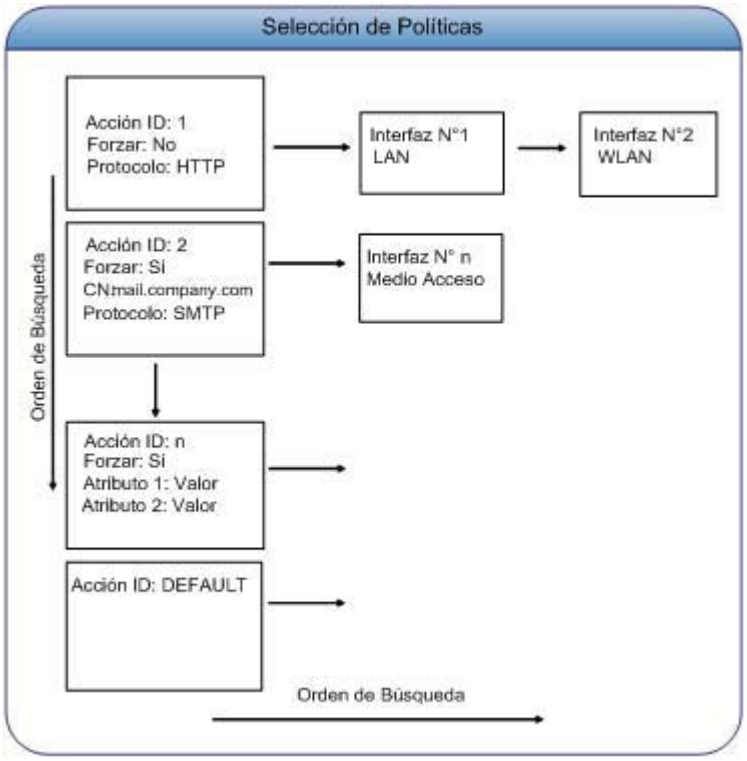


Figura 3-11. Selección de Políticas.

3.5.3 Ruteo Basado en Políticas

El ruteo basado en políticas provee un esquema de ruteo el cual va más allá de la tradicional estrategia del mejor esfuerzo de ciertos protocolos. Por ejemplo, esta estrategia permite a los routers el dirigir tráfico proveniente de distintas fuentes a través de diferentes conexiones a la red.

El uso de políticas permite el ruteo basado en la información del origen de los paquetes y no en el destino de estos. Permite además la incorporación de nuevos parámetros como QoS.

Algunos beneficios que pueden lograrse a través de la implementación de ruteo basado en políticas son:

Selección del tránsito basado en el usuario: Permite el ruteo del tráfico proveniente de distintos grupos de usuarios a través de diferentes conexiones a Internet.
Quality of Service (QoS): El QoS provisto puede ser aplicado según distintos tipos de tráfico, establecidos según la procedencia o valores de tipo de servicio adjuntos en las

cabeceras IP de los paquetes, realizado en los bordes de la red además de prestar mecanismos para priorizar el trafico en la red.

Ahorro de costo: Puede ser logrado por distribuir interactivamente el tráfico entre rutas permanentes de bajo ancho de banda/bajo costo y rutas mixtas de alto ancho de banda/alto costo.

Distribución de carga: Las políticas pueden ser implementadas para distribuir tráfico entre múltiples rutas basado solo en las características del trafico.

Las políticas pueden ser creadas desde diversos puntos de vista. Por un lado, el usuario puede preocuparse por características especiales del QoS de las aplicaciones; en el otro lado, los túneles tienen situaciones variantes de capacidades y de tiempo real. Las decisiones de ruteo deben ser realizadas en base de alguna de estas restricciones. No hay que olvidar que existe una contraparte entre la decisión correcta y la decisión eficiente. Por ejemplo, si se toman en cuenta todas las restricciones, la solución desarrollada puede ser muy consumidora de tiempo en cuanto a la toma de decisiones, lo que la hace ineficiente.

3.6 OBTENIENDO INFORMACIÓN

Las decisiones de ruteo, como de selección de interfaces, están basadas principalmente en la información proveniente de la red. Es posible, en un principio, obtener información desde la:

Capa de enlace: La información relacionada a la calidad del enlace es necesaria para una correcta toma de decisiones, ya que indica qué calidad de servicio y políticas pueden ser satisfechas y cuanto ancho de banda se encuentra disponible para usar. Con esto, es posible la obtención de valores estimados para el throughput, delay, entre otros existentes en el enlace. De igual forma, es posible detectar cuando la conexión sufre el proceso de handoff (cambio en el punto de conexión).

Capa de Red e IP: Obtención de parámetros como la dirección origen, destino, protocolo y encabezados de los paquetes.

Información de la red: Especificaciones técnicas de la red. Puede darse el caso que el proveedor del servicio entregue información como costo, ancho de banda, disponibilidad o tipos de accesos permitidos, entre otros.

Información desde los usuarios (adaptar la conexión a las características deseadas acorde a la definición de prioridades). Estos valores del usuario pueden influir directamente en la QoS para una cierta transmisión, modificando quizás su tasa de envío. Estas preferencias pueden ser presentadas directamente o guían la base para la construcción de políticas a seguir, ya que son requerimientos a cumplir.

Existen distintos tipos de información que pueden ser clasificados según:

a. Clase Uno: Información dinámica de la interfaz.

Este tipo de información describe la situación actual de la interfaz, la cual puede cambiar con factores como tiempo, ubicación, tráfico actual, etc. Algunos tipos posibles de información son:

Disponibilidad: Describe si las interfaces son o no usables en un momento de tiempo dado. Esta información es necesaria ya que solo la interfaz activada puede ser considerada durante el periodo de selección de interfaz. La mejor y más rápida forma para conocer la disponibilidad es preguntando a la capa de enlace por los datos detectados relacionados a la fuerza de la señal recibida. Una forma alternativa corresponde al uso de mensajes llamados “heartbeat” para revisar periódicamente los enlaces, pero los cuales pueden aumentar el costo de la transmisión.

RTT: Round Trip Time, es usado para medir la calidad de los túneles, como por ejemplo el nivel de congestión del tráfico. Cada túnel dado tiene su propio valor de RTT, el cual es determinado por su forma de conexión y la distancia entre los dos puntos finales del túnel.

Tasa de pérdida de paquetes: Este valor también es usado para medir la calidad del túnel. Los paquetes perdidos pueden ser resultado de distintas razones como congestión del tráfico, mala estabilidad de la red, etc. Un túnel con una tasa alta de pérdida de paquetes no es confiable.

Uso de buffer: Cada interfaz tiene un cierto número de buffers asociado. Si el buffer esta casi lleno, la interfaz no debe ser usada ya que el tráfico en el túnel es ya bastante alto como para ser manejado apropiadamente.

b. Clase Dos: Información estática de la capacidad de la interfaz.

Distintas interfaces usan diferentes tecnologías para acceder a la red. Cada tecnología tiene sus ventajas y desventajas comparadas con otras. Como resultado, estas características afectarán la decisión de ruteo por el bien de una transmisión eficiente. Algunos tipos de información incluyen:

Tasa máxima de transmisión: Cada tecnología de acceso tiene su propia capacidad de transmisión; por lo tanto, diferentes tipos de paquetes con requerimientos específicos de QoS deben ser calzados con diferentes tecnologías de acceso. Esta información es almacenada en el dispositivo de origen al comienzo.
Tasas de error: Dependiendo de los tipos de comunicación usados, las tasas de error debido a interferencia pueden variar.

c. Clase Tres: Información de Decisión del Usuario.

Como el usuario es el objetivo del servicio, este puede especificar algunas políticas, las cuales pueden sobrescribir a las políticas por defecto. Estas políticas pueden agregar poca o mucha complejidad como funcionalidad adicional al sistema. Algunos tipos de información incluyen:

Lista de equipos rechazados: Paquetes de ciertos lugares que no son aceptados.
Lista de tipos de conexiones rechazados: Tipos de paquetes específicos que no son aceptados por el usuario.

3.7 SELECCIÓN DE INTERFAZ

El proceso de Selección de Interfaz **[42]** está relacionado con la idea de efectuar un ruteo local (en el mismo equipo) de los paquetes a través de interfaces locales que constituyen el ambiente de multihoming. Este ruteo puede ser basado en una asociación de la conexión formada por la dirección IP, número de puerto, tipo de protocolo; pero pudiéndose agregar otros criterios como información de QoS. Este proceso define las rutas que serán usadas por los paquetes salientes. La información ya mencionada incluye tipo de conexión, disponibilidad de interfaces de redes y varias características de las redes en sí.

Ya sean los usuarios móviles, los propios nodos, aplicaciones u otros agentes, pueden definir preferencias y requerimientos relacionados con el uso de las interfaces presentes y del acceso a la red. Las decisiones para seleccionar las interfaces, se encuentran basadas, por lo tanto, en

esas preferencias y requerimientos, cuando estas pueden ser evaluadas contra la información obtenida sobre la red.

El mecanismo de selección tiene que estar presente en los componentes del enlace multihoming. Por ejemplo:

En el HA: La selección debe ser basada en la información almacenada en el Binding Cache.

En el MR: Debe ser seleccionada según información basada en los RA recibidos en sus interfaces.

En el MN: Debe ser seleccionado según la información recopilada de los enlaces.

3.8 TRANSMISIÓN

Como se mencionó, es necesario un mecanismo para esconder los cambios en la configuración de los sockets internos de las aplicaciones de forma de asegurar la transmisión correcta de los paquetes. La idea corresponde a re-estampar los paquetes entrantes, así que estos lucen como si ellos viniesen de su ubicación original y re-estampar todo los paquetes salientes, de forma de redirigirlos mediante redes externas. Este concepto es similar, aunque diferenciando en su base, al proceso de Network Address Translator (NAT).

En la transmisión hay 2 posibles problemas que son necesarios de considerar:

Filtrado de ingreso: Para beneficiarse del uso de nodos usando multihoming, es frecuente el dividir paquetes pertenecientes a la misma sesión a través de distintos túneles. Pero por hacer esto, hay que tener en cuenta el proceso de filtrado de ingreso, el cual ocurre cuando un nodo envía paquetes mediante túneles asignados a distintos HA, lo cual puede hacer que sean rechazados por el destinatario final por venir con un prefijo distinto de otra red.

Detección de fallas (ej. Router Advertisement): Un método posible es enviar mensajes de BU en forma más regular con valores de tiempo de vida (lifetime) más cortos. En forma similar, el HA puede regresar mensajes de binding acknowledgement con valores de lifetime mas pequeños, forzando a que el nodo móvil envíe sus mensajes de BU en forma más frecuente. Estos mensajes pueden ser usados en forma de heartbeats del

túnel. El nodo móvil puede también apoyarse en los mensajes de RA enviados por los routers de acceso u otro mecanismo similar para la detección de fallas. Es necesario considerar que las formas que incluyen el envío de mensajes adicionales también afectarán en la congestión de las rutas establecidas.

Los nodos móviles que emplean túneles bidireccionales con sus HA deben cambiar en forma dinámica sus puntos de salida, dependiendo del estado del enlace en el caso, por ejemplo, de detectarse que una interfaz se cayó. Esto genera la búsqueda de una ruta alternativa, la cual puede ser provista por otro punto de acceso a la red o por otra interfaz de comunicación. En este caso, el nodo debe reestablecer el túnel bidireccional usando esta ruta alternativa.

Por último, en la transmisión, el paso más importante es el de la selección de la nueva ruta, especialmente cuando la ruta primaria o por defecto ha dejado de funcionar. Para esto se tiene:

Detectar la presencia de una ruta alternativa: En primer lugar, el nodo móvil debe ser capaz de detectar rutas alternativas. Esto debe ser sencillo en el caso que la propiedad de multihoming esté dada por la presencia de múltiples interfaces de acceso, ya que el nodo debería conocer su propia configuración y sus formas de conexión a Internet. En el caso donde existan distintos routers móviles o puntos de acceso, cada router debe detectar la presencia de los otros routers móviles. Esto puede ser logrado, por ejemplo, a través de mensajes de Router Advertisement (RA) detectados en sus interfaces de ingreso. Cuando el router recibe estos mensajes debe darse cuenta que es posible crear una ruta alternativa de acceso a Internet a través de este dispositivo. Este proceso es similar al seguido por el nodo móvil para descubrir nuevas rutas.

Re-Establecimiento de Túneles Bidireccionales: Cuando el nodo móvil detecta que sus enlaces con su Home Agent se ha roto, debe establecer un nuevo túnel de acceso a través de la ruta alternativa detectada. Es posible distinguir dos casos. Primero, la ruta alternativa es provista por una interfaz de acceso que pertenece al nodo móvil. Segundo, que la ruta sea provista por otro router móvil, conectado a la red.

◦ Usando interfaces de salida alternativas: La idea es usar la dirección asignada (CoA) a la interfaz alternativa como la nueva CoA del nodo móvil para reestablecer los túneles de comunicación con su HA. La forma de hacer esto corresponde a enviar un mensaje

de BU al HA indicando la nueva CoA a usar para el nodo móvil. Después de recibir la respuesta, los paquetes son enviados a través del nuevo túnel establecido.
◦ Usando MRs alternativos: El router móvil puede utilizar una ruta entregada por un router móvil alternativo, si es que este existiese, para reestablecer la comunicación con su HA. Primero, el router móvil tiene que obtener su CoA con la información anunciada por el router móvil alternativo en su interfaz de ingreso. Entonces, este realiza el proceso de BU con su HA usando la nueva CoA. Luego de esto, es posible encapsular los paquetes salientes a través del túnel bidireccional establecido en la ruta alternativa.

3.9 UBICANDO LA SOLUCIÓN

Dadas las características de la arquitectura a diseñar y de la información que será necesaria procesar, la solución debe estar presente entre, o ser parte de dos capas: la de transporte y la de red. Principalmente por:

Capa de transporte: Para manejar los flujos y administrar y chequear las políticas. La información que realiza el proceso del sistema es ubicado en el nivel de usuario de forma de mantener las capas inferiores con las menores modificaciones posibles.

Capa de red: Encargada de transmitir los paquetes y manejar el direccionamiento de IPv6.

Con la información obtenida de estos lugares, se han de desarrollar una serie de componentes que permitirán el correcto funcionamiento de la estructura multihoming a crear **[43]**. Estos componentes deben incluir funciones como:

Administrador y selector de acceso: Su función es la de monitorear constantemente las interfaces del cliente por conectividad e información de estado de estas y permitir conectar o desconectar las interfaces según las reglas de transferencia creadas de forma de crear nuevos túneles de comunicación. Para la selección también son tomadas en cuenta las definiciones o requerimientos del usuario.

Administrador de movilidad: Este componente es el encargado de establecer los túneles de comunicación y manejar la movilidad del cliente.

Administrador de perfiles: Maneja tanto los requerimientos de las aplicaciones como del usuario. Incluye información como el modo de operación, las interfaces a usar, el

tipo de ordenamiento para seleccionar las interfaces y otras funcionalidades para manejar las conexiones.

Controlador de desempeño: Suministra información relacionada al desempeño de las transmisiones. Puede residir tanto en el equipo móvil como en el HA, permitiendo en forma conjunta el monitoreo de los enlaces establecidos.

3.10 OTROS PUNTOS

3.10.1 Comunicación entre Redes

La convergencia hacia IPv6 de las redes actuales es un hecho, debido a sus mayores prestaciones en rendimiento, optimización, ruteo, tráfico, direccionamiento entre otros. De hecho, la cuarta generación de redes celulares tiene a IPv6 como tecnología base a nivel de red. Sin embargo, hoy en día la mayor cantidad de redes se encuentran bajo IPv4. Esto hace necesario que exista una transparencia en la comunicación, especialmente a lo que redes móviles se refiere. A pesar de que la arquitectura a desarrollar se encuentre sobre el tema referente al establecimiento de comunicación (referido al paso de mensajes de los protocolos y creación de los correspondientes túneles), es necesario tenerlo en cuenta, ya que una vez creada la comunicación, las aplicaciones potencialmente podrán conversar con entidades ubicadas en ambas redes, lo que lleva a que el proceso de transmisión debe hacer este tema transparente a estas **[44]**.

Para que esto sea posible, las entidades en la comunicación, y principalmente el HA y el MN, deben ser capaces de intercambiar mensajes a través de las CoAs, independientemente si estos son IPv4 o IPv6, por lo que deben ser capaces de manejar dos tipos de registros de direcciones para ambos tipos de protocolos, conocido como Dual Stack (Figura 3-12 y 3-13), a la vez de comprender los distintos tipos de mensajes dependiendo en que tipo de red se encuentren. Generalmente son equipos que están al borde de la red los que poseen esta característica, prestando un servicio transparente a los equipos que están detrás de ellos. Para esta comunicación, existen esfuerzos para crear un nuevo mensaje **[45]** a incluir en el registro del HA que permita indicar la dirección en un formato correspondiente a la red, de forma de dirigir los mensajes por la red correspondiente.

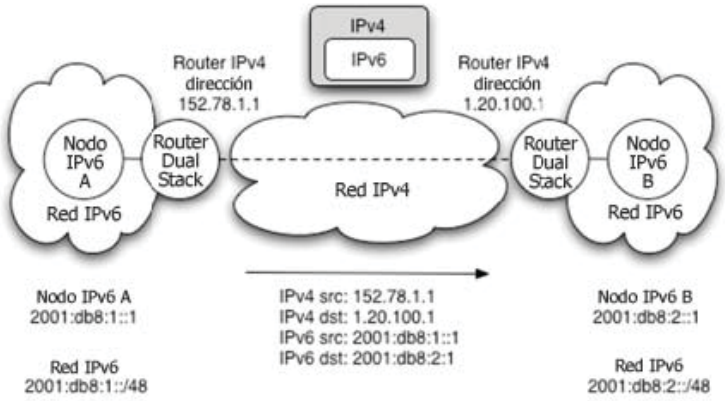


Figura 3-12. Conexión mediante Dual Stack.

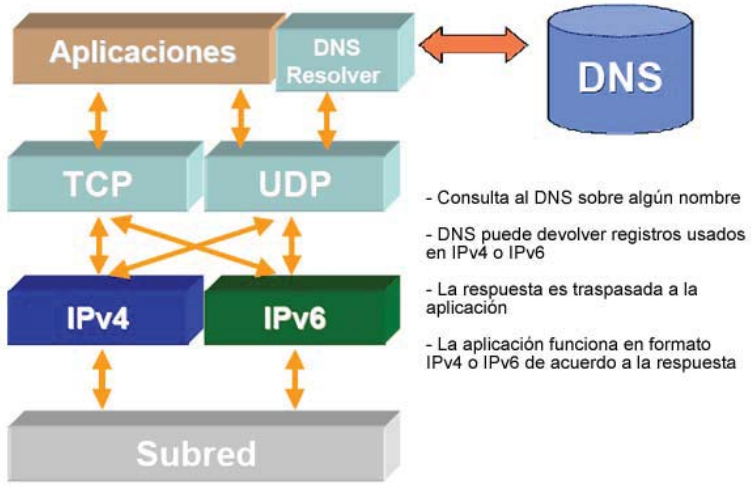


Figura 3-13. Resolución de nombres bajo Dual Stack.

En este sentido, la aplicación hace la elección de que protocolo usar, en caso de que ambos estén disponibles. En casos en que se use Dual Stack, se debe tener presente que sólo una buena conectividad y configuración IPv6 hace que IPv4 no se vea afectado en desempeño. A su vez, al estar recibiendo comunicaciones de ambos protocolos, las medidas de seguridad deben extenderse a ambos.

La comunicación entonces hace necesario que los mensajes sean encapsulados (Figura 314) de forma de poder ser entendidos por el CoA para poder ser dirigidos al HA, quien los entrega a su destinatario final, manteniendo una completa compatibilidad con IPv4, facilitando el proceso de transición de una a otra red **[46]**.

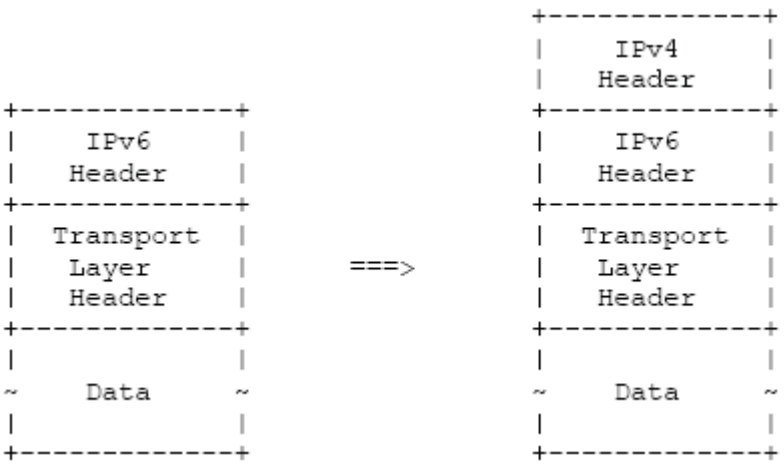


Figura 3-14. Encapsulación de un paquete IPv6 en un túnel IPv4.

De igual manera, es posible encapsular los paquetes IPv4 sobre un túnel IPv6 establecido, a medida que las comunicaciones IPv6 vayan predominando sobre la red.

3.10.2 Detección de Enlaces

Debido a la movilidad, el equipo puede presentar cambios relacionados al lugar de enlace a la red constantemente. Por esto, el equipo debe estar frecuentemente revisando cuando se produce este cambio a nivel de enlace, y decidir cuando es necesaria una nueva configuración ya sea a nivel de IP, como de la renovación de los registros de enlace que posee **[37]**.

Generalmente, las nuevas configuraciones son enviadas desde las redes en mensajes de RA, los que permiten al equipo móvil detectar algún cambio y poder adaptarse de forma de no provocar una interrupción prolongada en las transmisiones. De esta forma, los mensajes provenientes desde la capa de enlace del dispositivo móvil se convierten en actores importantes al momento de detectar cambios en la configuración de acceso **[35]**.

Los mensajes recibidos a nivel de la capa IP también son útiles para detectar algún cambio, pero estos son enviados con una cierta frecuencia lo que hace que las indicaciones incluidas no estén inmediatamente disponibles para tomar una acción relacionada al cambio.

3.11 CONEXIÓN AL MUNDO IPv6

3.11.1 Planeación

El análisis inicial corresponde a la definición de los recursos existentes, ya sea software (plataforma, aplicaciones y servicios) como hardware, que soportan IPv6 y definir que capacidades adicionales son requeridas. Tal como en IPv4, es necesario el adquirir un espacio de direcciones IPv6. Junto con esto, es necesario establecer conectividad con la red global IPv6. Una vez asignada, es necesario formular un nuevo plan de direccionamiento el cual ahora incluye direcciones globales (práctica recomendada en la literatura para IPv6 **[31]**), a diferencia que en IPv4, donde también se incluyen direcciones internas producto del uso de NAT.

Medidas similares se han de considerar de acuerdo a la seguridad, las cuales deben tener su base en las medidas usadas en la red IPv4. Estas deben ser revisadas al igual que las nuevas funcionalidades y características entregadas por IPv6, de forma de agregar consideraciones de IPv6 en los planes y pruebas existentes en la institución.

Este proceso debe ser llevado a cabo pensando en a interoperabilidad entre IPv4 e IPv6, y no en un migración completa.

3.11.2 Impacto en las Redes

La capa 2, capa de enlace, no debe significar mayores cambios en la infraestructura, salvo que existan en uso características especiales en la red que requieran actualizaciones de software y/o hardware de forma de soportar nuevas funcionalidades (nuevos tipos de mensaje a nivel de enlace) y configuraciones para el manejo correcto de IPv6. En la capa de red, los mecanismos de ruteo y manejo de tráfico deben soportar IPv6. Esto significa la implementación de protocolos de encaminamiento interior dentro de la red que permitan dirigir el tráfico por la ruta más eficiente, como RIPng, OSPF, u otros, que además tengan integrado el soporte de IPv6 junto al hardware necesario que permita el procesamiento de los paquetes en forma más eficiente. Una alternativa, en el caso que la topología de la red no varíe con frecuencia (o se trate de redes de tamaño pequeño donde no existan enlaces de comunicación redundantes), es usar rutas estáticas de configuración semiautomática y que pueden ser usadas en paralelo con los protocolos de red. A este nivel, es importante el manejo

de la asignación de los distintos rangos de direcciones (prefijos de red) respetando la estructura jerárquica entregada por el protocolo, como de las nuevas políticas de ruteo en los equipos. Las redes IPv6 han de ser de máscara 64 porque los mecanismos de autoconfiguración sin estado se basan en construir el sufijo de 64 bits a partir de la dirección física del dispositivo de red (o dirección MAC) que dispone de 48 bits.

En cuanto a los usuarios, estos deben contar con sistemas operativos con soporte de IPv6, de manera de tener acceso normal a la red y a los servicios de información ofrecidos sobre esta. De igual forma, las aplicaciones a nivel de empresa o de usuario deben ser portadas para manejar IPv6 en forma transparente y poder hacer uso de sus funcionalidades, como soporte de comunicaciones extremo a extremo.

En lo anterior, surgen puntos que una organización debe considerar al momento de decidir el desarrollo de IPv6. En general este desarrollo necesita actualizaciones en software y hardware además del desarrollo de nuevas aplicaciones, por lo que este proceso debe ser planeado cuidadosamente y testeado en redes piloto de manera de corroborar la estrategia a seguir. Como corresponde a un proyecto de cambio mayor, es necesario analizar principalmente el tipo de administración requerida, aplicaciones, infraestructura de seguridad requerida que estén presentes en la red final. En forma adicional puntos de interés corresponden a la protección de la inversión en equipamiento existente, el cual debe ser adecuado para convivir con IPv6, por lo que el proceso de migración debe considerar alternativas como el desarrollo independiente de IPv6 y de IPv4 en redes separadas como la coexistencia de ambas redes en ciertos sectores o funcionalidades que no sea posible portarlas completamente a IPv6 o que aún no se encuentren totalmente estables u ofrecidas por los proveedores de servicios, evitando problemas de interoperabilidad en los servicios. Hay que considerar, que actualmente está perfeccionándose el equipamiento de red que soporta IPv6, por lo que el desempeño obtenido se ve degradado, principalmente por que el nuevo protocolo es soportado en gran parte a nivel de software, y no de hardware.

Ya que son varios los segmentos, una estrategia recomendada es el migrar a IPv6 por etapas, comenzando a nivel de equipamiento de red de extremo junto a los usuarios finales, proveyendo funcionalidades básicas de IPv6, ya que en las redes existen actualmente sistemas y equipamiento funcionando en IPv4 y que van a pasar a formar parte de la nueva red y deben seguir siendo accesibles desde esta. En esto se incluyen servicios como SMTP, DNS, NTP,

NAC y DHCP, los cuales deben ser portados al nuevo protocolo pudiendo ser utilizados mediante ambos tipos de direcciones. Posteriormente, se continúa con los protocolos de la capa de enlace y de red de manera de permitir a los usuarios finales el comunicarse sobre la misma red y utilizar y evaluar funcionalidades extremo a extremo. En este proceso hay que tener clara la dependencia existente entre servicio y aplicaciones existentes, de manera de poseer una guía y poder priorizar y testear las distintas fases del porte.

Debido a la posible coexistencia con IPv4, una estrategia de traducción debe ser considerada de manera de permitir la interoperabilidad entre ambas redes. Generalmente, son dos las principales alternativas analizadas. Primero, que la red IPv4 e IPv6 convivan en el mismo segmento de red (red dual-stack), por lo que se comparte la misma red física (equipamiento de red). La desventaja principal de este método está en que ambas redes podrían llegar a interferir entre sí, sobre todo en los casos en que los recursos de red estén explotados al límite antes de introducir IPv6, o en los casos en que los routers implicados no tuviesen las capacidades necesarias para encaminar los paquetes de ambos niveles de red. Esta problemática, si bien será poco frecuente en redes corporativas de pequeña dimensión, sí será necesario analizarla en profundidad en las redes de los proveedores de servicio a terceros o en las que se deba garantizar estrictamente la calidad de servicio.

La segunda opción es utilizar distintos segmentos de red y equipamiento (red sólo IPv6 en paralelo), lo cual permite mantener el nivel de servicio en la red original o dar tratamiento diferente al tráfico de ambas redes, sin embargo, los recursos necesarios para esta opción, son mayores. Un mayor análisis se encuentra en **[47]** donde son presentados 3 escenarios y puntos necesarios de tratar al decidir comenzar la adopción de IPv6 abarcando áreas desde seguridad hasta aplicaciones.

3.11.3 Seguridad

Bajo los estándares de desarrollo referidos a las comunicaciones inalámbricas bajo IPv6, es posible brindar un cierto grado de seguridad a distintos niveles.

Por ejemplo, a nivel de establecimiento de enlaces, la seguridad se encuentra presente en la comunicación de mensajes de enlace con un HA o de registro de una nueva CoA emitidos por los equipos, ya que se manejan claves compartidas para la validación de este proceso en

ambos extremos. El enlace con la nueva red también incluye medidas de seguridad, enfocadas en el control de acceso, referidas a la asignación de las direcciones, las cuales son generadas criptográficamente. La seguridad también se presenta en los chequeos de prueba (Return Routability Procedure) con las entidades destino con tal de comprobar su correcta existencia y direccionamiento **[1]**. Este procedimiento es llevado a cabo una vez que el proceso de registro con la red hogar ha sido completado, lo que produce un retardo mayor en el reestablecimiento de la comunicación, sin embargo previene del robo y suplantación de la sesión establecida, haciendo las intrusiones más difíciles y aumentando la confiabilidad de las transmisiones sobre la red.

A nivel de las comunicaciones, establecidas por la capa de transporte o superiores, las transmisiones o el paso de mensajes pueden ser autentificados mediante diversas formas, ya sea usando un protocolo seguro como HTTPS, un camino asegurado mediante el uso de IPsec (soportado por defecto en IPv6), o mediante la encriptación de estos mediante claves. En cuanto a la integridad de las comunicaciones, el uso de IPsec permite reducir los ataques mediante sniffer a la red y las suplantaciones de identidad o de sesión, como ocurre en los ataques MiM (Man in the Middle). Sin el uso de IPsec, no habría mayor diferencia en cuanto a seguridad entre IPv4 e IPv6. Sin embargo, las mayorías de las vulnerabilidades existen a nivel de aplicación, y no de transporte o red, donde IPsec no es útil. **[48]**

Con IPv6 es posible el mejorar la seguridad y eficiencia en la red, ya sea a nivel fijo o móvil. A pesar de esto, IPv6 no es una solución global al tema de seguridad, ya que de todas formas el cómo es diseñada la arquitectura de seguridad (como el caso de políticas de filtrado y firewalls) y aplicaciones influye en el resultado final. Es de consideración que el uso de IPv6 agrega nuevas amenazas a la red como el escaneo de nuevos tipos de direcciones (por ejemplo del tipo multicast usada en los servidores DHCP o en aplicaciones multimedia como IPTV) detectando recursos importantes en la red y provocando ataques de denegación de servicio, atacando las debilidades de los protocolos y falencias en los mecanismos de transición (dado el paso por dos tipos de redes, las políticas de los firewall no están preparados para filtrar dos protocolos al mismo tiempo) y nuevas clases de virus y gusanos (creados específicamente para IPv6). Hay que agregar que recién están apareciendo productos para seguridad que soporten tanto IPv4 como IPv6 en forma nativa, ya sea hardware como software, por parte de los grandes proveedores de tecnología, lo que ha impedido que el proceso de transición sea continuo y seguro. Sin embargo, la mayoría de los productos actualmente en el mercado, no

soportan antispam, antivirus, filtrado de contenido o funciones de IDS o IPS, con la misma efectividad que sobre tráfico IPv4.

El soporte y uso de multihoming en redes IPv6 móviles introduce otra área de riesgos en las comunicaciones, ya que ahora cada equipo o dispositivo tiene grupos de direcciones IP las cuales cambian dinámicamente, lo cual no es considerado ampliamente en las soluciones tradicionales de seguridad, las que suponen topologías estáticas y direcciones permanentes durante las sesiones establecidas. En cambio, ahora la topología es dinámica, existiendo un aumento en el número de rutas que puede seguir el tráfico establecido. En forma adicional, los nuevos mensajes de coordinación utilizados (como los de Binding Update o Route Advertisement) y las múltiples rutas establecidas permiten introducir nuevas amenazas potenciales de ataques de DoS o de redirección del tráfico, robo de sesión o suplantación de dirección, intercepción de información, entre otros, principalmente producto del manejo de más de una dirección IP por parte del dispositivo **[49].**

Actualmente, uno de las áreas que posee más puntos relativos a seguridad es la relacionada con los mecanismos de transición y coexistencia entre IPv4 e IPv6, los cuales generalmente funcionan encapsulando el tráfico IPv6 sobre paquetes IPv4, haciendo la mayoría de las medidas de seguridad en uso (firewall o filtros entre otros) inservibles. Se profundiza este tema en el punto 3.11.5.

3.11.4 Aplicaciones Adaptables

Para un correcto uso de la solución a desarrollar bajo la arquitectura propuesta, han salido a la luz estudios para la creación de aplicaciones adaptables, es decir, que poseen múltiples comportamientos para una misma acción, dependiendo de los recursos disponibles, en este caso, relacionados con la conexión. Actualmente, aplicaciones como QuickTime o Reproductor de Windows permiten detectar el estado de la conexión a Internet y de acuerdo a eso ofrecer una lista de reproducciones que se adecuen de mejor forma a los recursos presentes. De igual forma que el presente trabajo, hacen uso de políticas de adaptación según las características que se vayan detectando durante su uso.

Una de las áreas en que este punto tiene mayor relevancia es sobre las transmisiones multimedia, en especial a las relativas a transmisión de video sobre la red, como IPTV. En

general, esta transmisión es realizada sobre escenarios que cuentan con mecanismos de monitoreo de la red y canales de control por el cliente o de retroalimentación de la señal, los cuales usan información proveniente de la capa de enlace y/o física, permitiendo entregar una transmisión continua al destinatario, ajustando el formato y la calidad de transmisión, según las nuevas condiciones. Basado en esta información, el servidor de video puede adecuar la tasa o el formato de la transmisión. Por ejemplo, en una videoconferencia o en una transmisión con video es posible usar distintos tipos de comprensión los que afectan tanto la calidad como los recursos (ancho de banda) utilizados. El objetivo de estos trabajos es poder extender esta funcionalidad presente a una forma dinámica, es decir, que permitan cambiar los tipos de reproducción de acuerdo a los cambios presentes durante la transmisión.

3.11.5 Mecanismos de Transición

Hoy en día, la primera estrategia para unirse a IPv6, es el desarrollar un ambiente sólo IPv6, que tenga equipamiento específico a este protocolo y pueda hacer uso de los diversos beneficios que entregue este entorno. Sin embargo, será necesario durante el periodo de transición entre ambos redes, aunque sólo sea para conectarse con otros ambientes IPv6, atravesar redes con el protocolo IPv4, por lo que debe existir un método de realizar la conversión entre ambos lenguajes de forma de permitir la comunicación sin problemas entre los extremos (Figura 3-15). A favor se tiene que IPv6 fue desarrollado teniendo en mente la coexistencia y migración desde IPv4. El problema es que en la transición, las comunicaciones deben atravesar áreas sin soporte de la nueva versión del protocolo, a la vez de atravesar las medidas y políticas de seguridad definidas para IPv4, y que, en su mayoría, son incapaces de revisar el protocolo IPv6 encapsulado junto a IPv4.

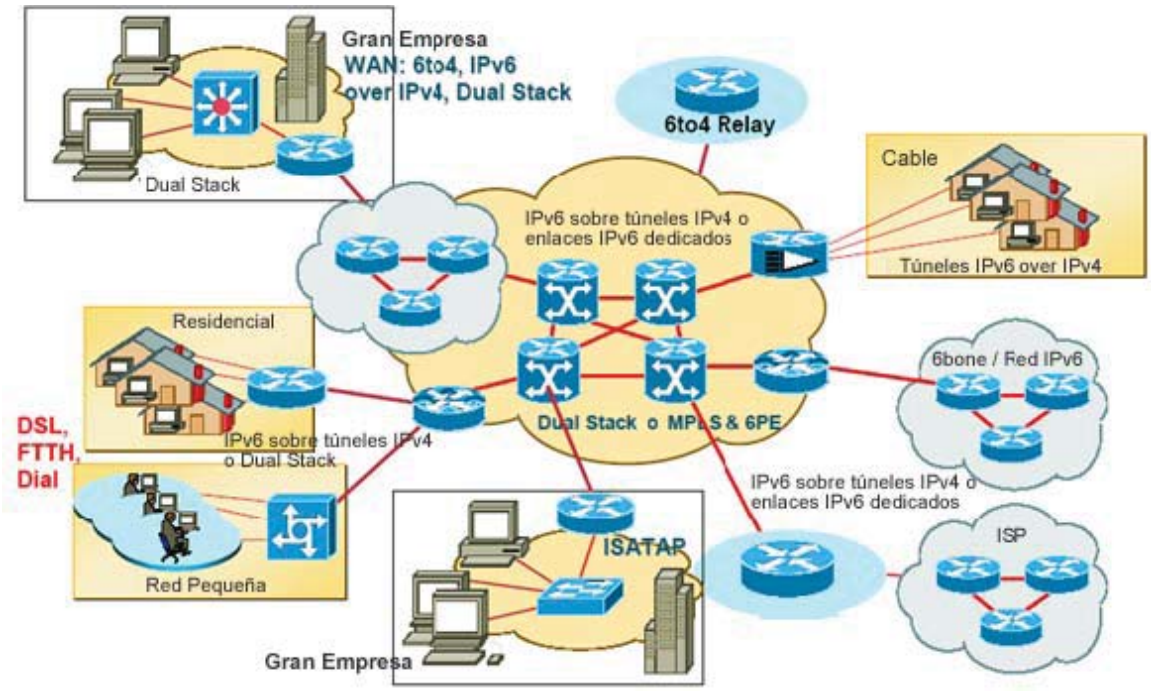


Figura 3-15. Conexiones a IPv6.

Principalmente, la puesta a punto de IPv6 dentro de un ambiente de uso, es dividida en dos fases. La primera tiene relación con la elección de la implementación a usar, incluyendo la definición de protocolos y aplicaciones (como VoIP, tráfico multimedia, entre otros) a soportar sobre la red. La segunda, consiste en agregar soporte IPv6 al backbone de la empresa, y enfocarse en el soporte doble de IPv4 e IPv6.

En el año 2006, distintos proyectos de alcance global, entre ellos 6BONE **[50]** (backbone experimental para ayudar al estudio y desarrollo de IPv6) y M6BONE (servicio de multicast sobre IPv6), permitieron establecer una comunidad de enlaces para conectarse a la red IPv6. Dado que la gran mayoría de los usuarios de esta red hacían su acceso desde conexiones IPv4, surgieron distintas formas de establecer enlaces a la red IPv6, diferenciándose hacia quien están enfocadas, que pueden ser principalmente un usuario, empresa o un proveedor de servicios.

Estas técnicas de traslado hacen uso de túneles que permiten encapsular los paquetes IPv6 en paquetes IPv4 para atravesar redes sin migrar. En este proceso, los paquetes pueden ser fragmentados debido a la información adicional que es necesaria comunicar, pudiendo afectar el desempeño de la transmisión debido al mayor trabajo de procesamiento presente en los extremos de la comunicación. En estos casos, es necesario que los extremos de los túneles

posean la característica de ser dual stack, de forma de llevar a cabo la transformación de la información.

Existen distintas formas de conexión que varían según factores como seguridad, tipo de configuración, manejo de direcciones, escalabilidad, entre otros. Las más usadas son:

Túneles configurados manualmente (Tunnel Broker): Es necesario que una dirección IPv6 sea configurada en la interfaz del tipo túnel, junto a direcciones IPv4 en ambos extremos. El Túnel Broker sigue un modelo cliente-servidor. El cliente es un dispositivo IPv6 dual-stack conectado a la red IPv4. El servidor está formado por un túnel broker y por uno o más servidores de túneles, los cuales son IPv6 dual-stack y están conectados a la red global y a backbones IPv6. Este tipo de configuración puede ser usado entre dos routers de borde o entre estos y un equipo cliente. El inconveniente se produce ya que es necesaria la configuración en ambos extremos del túnel, generando saturación en la administración de estos cuando crece el número de túneles. A favor, tiene el hecho de que los túneles son generados manualmente, haciendo más seguras las conexiones y servicios ofrecidos sobre estos, a la vez de ser fáciles de establecer y configurar ya que la administración de estos queda en los ISP que brindan el servicio.

Túneles 6to4: En este caso, el extremo final del túnel es determinado transformando la dirección IPv4 de éste, en una dirección IPv6 y uniéndolo con el prefijo 2002::/16. Este tipo de configuración puede ser usado entre dos routers de borde o entre estos y un equipo cliente. Es necesario contar con equipos conocidos como 6to4 relay routers, que permitan dirigir el tráfico y realizar la transformación de IPv4 a IPv6, y viceversa, entre ambos extremos del túnel. Por el hecho de que no siempre se conoce la integridad de los extremos de los túneles establecidos, este método no es tan utilizado tomando en cuenta la seguridad que ofrece. Es posible obtener automáticamente respuesta de equipos configurados como relay routers, como está especificado en [51,52]. Un ejemplo de la configuración básica necesaria está en la Figura 3-16.

```
ip tunnel add tun6to4 mode sit remote any local 158.251.10.23 ttl 64
ip link set dev tun6to4 up
ip -6 addr add 2002:9efb:0a17::1582:5101:0023/128 dev tun6to4
ip -6 route add 2000::/3 via ::192.88.99.1 dev tun6to4 metric 1
```

Figura 3-16. Configuración para acceso IPv6 usando 6to4 (ambiente Linux).

Una comparación entre estos dos tipos de túneles es:

Característica	Túneles Manuales	6to4
Seguridad	Soporta Autentificación	Potencialmente Peligroso
Configuración	Manual	Automática
Escalabilidad	Buena	Muy Buena
Descubrimiento de Servicio de Túnel	Configuración Manual	Automática
Concentración de Tráfico	En servidor de túneles	En 6to4 relay

Tabla 3-1. Comparación entre Túneles Manuales y 6to4.

TEREDO: Es una variación de los tipos anteriores, permitiendo establecer los túneles a través de NAT. Esto es realizado a través del uso de paquetes UDP.

Otros mecanismos para realizar esta transición, basados en túneles son:

ISATAP: Protocolo de direccionamiento automático de túneles entre sitios sin necesidad de routers IPv6 gracias al uso de direcciones ISATAP especialmente construidas.

Capa de Enlace dedicada a IPv6: Esta opción requiere infraestructura adicional para manejar las conexiones.

En el caso en que sólo se cuente con un entorno IPv4, y se requiera comunicarse con un entorno IPv6, existe un proceso alternativo de traducción que puede realizarse a distintos niveles de comunicación, el cual actúa en forma similar a un NAT, pero mapeando direcciones IPv4 a IPv6. Dependiendo de la situación, este proceso puede ser llevado a nivel de red, transporte o de aplicación. Por ejemplo, cuando la aplicación no puede ser portada a IPv6 o el sistema o hardware sólo funciona en IPv4. Uno de las técnicas más usadas en esta área es NAT-PT el cual permite la comunicación bidireccional entre ambientes sólo IPv6 y ambientes sólo IPv4 a través de Internet. En NAT-PT, los equipos IPv4 se encuentran

representados con direcciones IPv6 formadas por un prefijo preconfigurado (el cual es conocido en la red) más la dirección IPv4 (por ejemplo, <PREFIX::/96><IPv4>), la cual es formada automáticamente cuando llegan solicitudes a equipos IPv4. Una de las desventajas de este proceso, es que el equipo borde en la red debe tener conocimiento de las direcciones IPv4 siendo usadas, las cuales deben ser públicas.

3.12 ESCENARIOS DE APLICACIÓN

Escenarios que se podrán ver beneficiados por el uso de IPv6 y, particularmente por Mobile IPv6, pueden ser empresas que posean instalaciones en distintas ubicaciones (por ejemplo, extracción de minerales); empresas que posean un gran número de personal en terreno (por ejemplo, distribuidoras de alimentos); o empresas que tengan unidades móviles que tengan requerimientos altos en cuanto a su calidad de conexión (por ejemplo, carabineros o instituciones de salud). Como base del presente proyecto se tuvo a este último grupo, ejemplificado por las unidades de emergencia, con ambulancias y hospitales como actores y la telemedicina como servicio, debido a los requerimientos que serían necesarios para un correcto desempeño de este escenario (Figura 3-17).

En este marco, lo que se busca es la vialidad de transmitir información de signos vitales e imágenes en tiempo real desde una ambulancia en el lugar de los hechos hacia un departamento en el hospital donde se creará una imagen del estado del paciente que está siendo trasladado. Este proceso es realizado a través de comunicaciones inalámbricas **[53]** de forma de poder mantener la conectividad a Internet mientras el móvil se encuentra en movimiento, pudiendo recibir tanto datos, voz o video sin mayores inconvenientes.

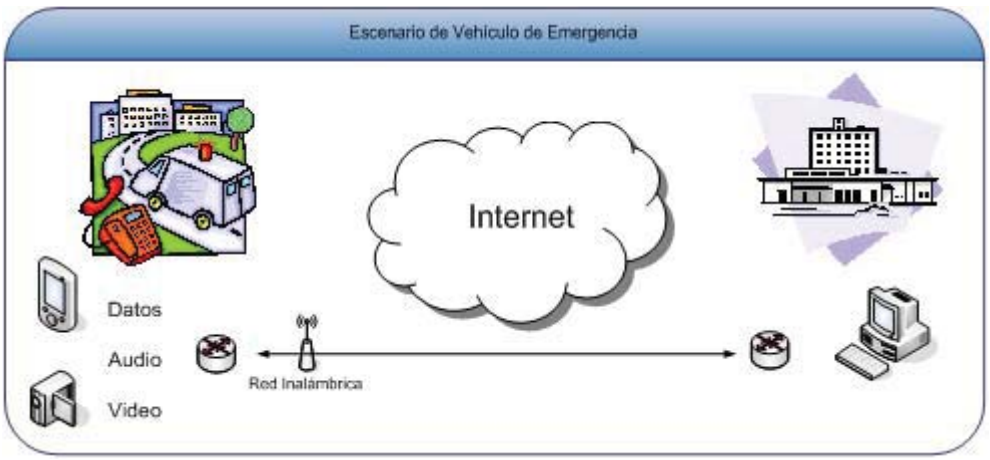


Figura 3-17. Telemedicina.

El principal objetivo es mejorar la calidad y el tiempo de cuidado durante el transporte hacia el hospital, inmediatamente desde el momento de daño y proveer mejor información al departamento de emergencia antes que el paciente llegue al hospital. Para obtener esto, lo primordial es estar siempre conectado desde y hacia el hospital, manteniendo todos los tipos de comunicaciones establecidos, ya sean voz, audio o video.

Con el uso del sistema de telemedicina móvil para transmitir información de diagnóstico en camino se trata de acortar el tiempo para recibir tratamiento adecuado, aumentando los tipos posibles de estos además de mejorar la captura de las reacciones físicas del paciente.

Algunos sub-objetivos a cumplir en esta área son:

- Demostrar que un sistema de telemedicina móvil puede ser implementado mediante tecnología inalámbrica entre una ambulancia y su correspondiente centro de atención.
- Demostrar que tanto la información de signos vitales como imagines o video pueden ser transmitidos y usados correctamente en tal sistema.
- Demostrar que la grabación electrónica del paciente puede ser implementada efectivamente al comienzo del proceso de cuidado.

De igual forma, es posible rescatar ciertas mejoras en el proceso (Figura 3-18), como lo son:

- Mejoramiento del diagnostico y administración del cuidado de traumas y de otras emergencias médicas.
- Mejorar los recursos del hospital.
- Proveer secuencia de audio del cuidado.
- Mejor preparación del hospital para recibir a pacientes.

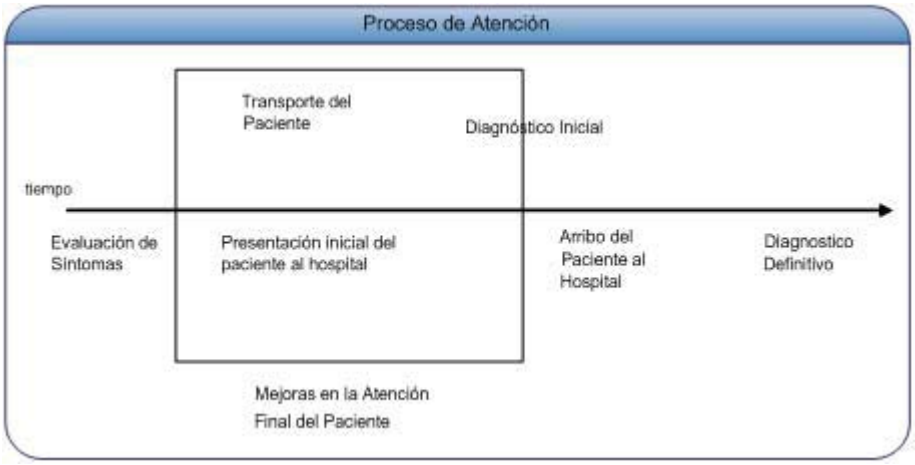


Figura 3-18. Mejoras del Proceso de Atención.

Estos mismos requerimientos y objetivos mencionados pueden ser llevados a otros vehículos de emergencia, como carabineros, bomberos, u otros a quienes también les beneficiaría el hecho de contar con comunicaciones de distinta índole, y no sólo de voz como comúnmente son mantenidas, pudiéndose ampliar tanto el rango de servicios y a la vez de mejorar el tiempo en que estos son aplicados.

Implementaciones en este ámbito ya se han llevado a la práctica, como es el caso de Emergency Unit Usage: Mobile ER Pilot Test, el cual hace uso de MIPv6 para establecer la conexión con hospitales o centros de emergencia, pudiendo hacer uso de distintas tecnologías como WiMax para tener disponible una mayor calidad de transmisión **[54].**

3.13 CONSIDERACIONES PARA LA SOLUCIÓN

Teniendo en consideración los temas tratados y explicados anteriormente, la propuesta a desarrollar consiste en:

Establecer y mantener una comunicación entre un nodo móvil y su red de origen. Para conseguir el fin de crear un nodo multihoming mediante distintos MRs, o distintas interfaces, es necesaria la creación de distintos componentes en ambos lados de la comunicación (nodo móvil como HA) que permitan el correcto funcionamiento de esta idea. Algunos de estos componentes incluyen un mecanismo de selección de interfaces y uno de asignación de transmisiones.

Para establecer esta comunicación, el nodo tendrá una interfaz conectada a Internet, la cual será usada por defecto. Además, tendrá que existir la posibilidad de agregar nuevas fuentes de comunicación al nodo (routers o puntos de acceso), las cuales podrán ser usadas como interfaces de acceso.

La solución deberá poder unirse a las redes (Foreign Network) que sea posible mientras se encuentra fuera de su red origen y obtener la respectiva CoA.

La configuración considerada corresponde al caso de "Múltiple MRs, un solo HA y un solo MN", referida como (n,1,1). En este caso, la configuración se traduce como múltiples túneles bidireccionales establecidos entre cada par (HoA,CoA) formado, trayendo tanto ventajas y temas a solucionar consigo.

Si es posible establecer distintos caminos de comunicación entre el emisor y el receptor, se dispondrá del uso de filtros para dividir las cadenas a transmitir entre los enlaces disponibles.

Posibles requerimientos.

◦ La división mencionada será realizada asignando prioridades a las transmisiones y un nivel de calidad (criterio de elección) a los enlaces disponibles. Para esto, será necesario el estableces políticas que ayuden al proceso de asignación y al de envío y recepción del tráfico en la red.
◦ El registro de diferentes interfaces de comunicación entre el emisor y receptor, deben ser agregadas al proceso de registro de interfaces (CoA) con el home agent a través de los mensajes de BU. Esto es opcional para el registro de los filtros o políticas, las cuales podrán ser comunicadas en otras formas.
◦ En el caso en que sean usados distintas fuentes como interfaces, es necesario establecer una coordinación entre ellos a través de un protocolo de ruteo basado en políticas.

Es necesario encapsular doblemente los paquetes. Primero desde el MR primario al secundario, y luego de este al HA. Esto evita reconfigurar los nodos, en caso que el MR primario pierda conectividad.

De igual manera, cuando los mensajes son recibidos desde el HA, es necesario que este encapsule doblemente los paquetes. El primer envoltorio indica que estos deben dirigirse primeramente al MR, y luego, desde ahí al MN.

Uso de mensaje de RA para dar a conocer su existencia al router primario. También puede usarse una modificación a los mensajes ICMP entre equipos, para conocer el estado del MR (fallas, desconexión u otros)

Para establecer y compartir las rutas, es posible el uso de BGP entre los routers de las distintas redes presentes en la solución.

Pasar de un punto de enlace a otro, sin perder la conectividad, haciendo este proceso transparente a las transmisiones.

Es posible usar una idea del fast handover para evitar que las desconexiones sean lentas. (Esto incluye modificación en cada router en el camino. El MR debe ser capaz de asignar los flujos a aquellas conexiones que estén funcionando correctamente).

La división por filtros de acuerdo a la aplicación de políticas debe poder realizarse en ambas partes que intervienen en la comunicación de forma que la optimización se produzca tanto en el tráfico entrante como saliente.

Selección de tipos de conexiones a usar. Estas pueden ser cadenas TCP o UDP. De igual forma, es posible el probar mediante secuencias de video de forma de ver el desempeño de la estructura diseñada en transmisiones de tiempo real.

En el escenario de aplicación propuesto, el MN pertenece a un HA (en este caso debido a una misma organización) que permite que ambos pertenezcan a una misma topología (preferentemente IPv6, debido a los beneficios ya mencionados, especialmente en el área de Mobile IP) lo que hace que ambos puedan entender el mismo protocolo de comunicación, eso si dependiendo sólo del estándar soportado por las redes foráneas en los que se encuentre el nodo móvil.

CAPÍTULO 4

CONECTÁNDOSE CON MIPv6.

4.1 ACOTAMIENTO DEL PROBLEMA

Dado que pueden existir distintas combinaciones para que una entidad se encuentre en un ambiente multihoming [35], ya sea por tener distintas interfaces, obtener distintas IPs o CoAs en sus interfaces, encontrarse asociado a distintos HAs u otra situación, la arquitectura a desarrollar se enfocará en el caso correspondiente a estar asociado a un HA (y tener 1 HoA), tener disponibles múltiples interfaces (adquisición de múltiples CoAs) y habilitar un prefijo para una red móvil (MNP) interna (Figura 4-1). Sin embargo, esto no impide que la arquitectura pueda incluir a los demás casos mediante extensiones y resolución de problemas puntuales que aparezcan. Por ejemplo, en el caso en que se poseen distintas HoAs y se realiza un cambio de interfaz en la comunicación, es posible que se presenten problemas debido al filtrado de ingreso de las transmisiones en un HA, situación que, con la simplificación del problema, ha sido eliminada.

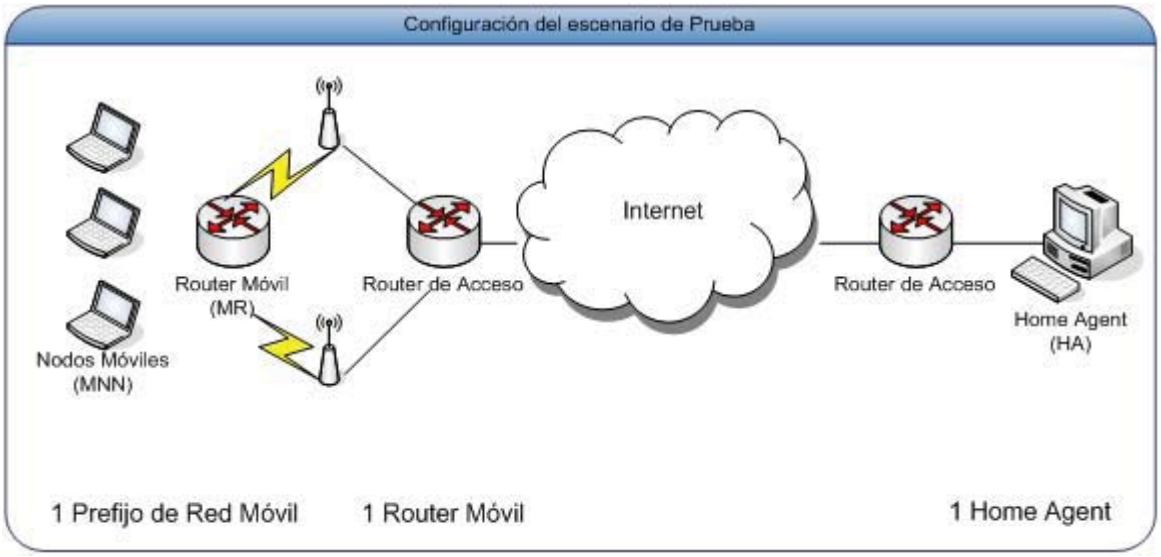


Figura 4-1. Escenario de Prueba.

A pesar de la limitación, este caso claramente corresponde a un ambiente multihoming (si múltiples túneles son mantenidos al mismo tiempo), además de ser posible el lograr diversos beneficios como balanceo y distribución de carga, obtención de una mayor confiabilidad, aplicación de políticas y filtros en la transmisión, junto a un aumento del ancho de banda disponible en la comunicación.

A su vez, los mecanismos de detección de fallas, exploración y selección de rutas deben ser provistos por ambos extremos de los túneles establecidos, es decir, el MR (en caso de NEMO) y el HA.

4.2 METODOLOGÍA

Para poder examinar los aspectos de calidad de servicio de los protocolos MIPv6 y NEMO, es necesario observar su comportamiento en un ambiente real. Para ello se desarrolló un escenario de pruebas con terminales y otros elementos de red (equipos y routers) con los que se implementaron los protocolos de movilidad bajo estudio. El escenario en cuestión debe ser lo suficientemente fiable y robusto como para poder controlar todos los aspectos que atañen al buen funcionamiento de los protocolos de movilidad, para que se puedan realizar las modificaciones pertinentes en caso necesario y se garantice la validación de los datos obtenidos.

Una vez construido el escenario de pruebas, el paso siguiente corresponde a diseñar una metodología de pruebas que permita caracterizar determinados parámetros de calidad de servicio. Sin embargo, la metodología de pruebas desarrollada será totalmente válida para realizar el análisis de otros protocolos de movilidad existentes o que puedan reemplazar los presentes, o bien otro tipo de protocolos de redes bajo IPv6 que nada tengan que ver con movilidad, mediante modificaciones pertinentes al escenario de prueba para cada protocolo. De esta forma se podrá tomar este proyecto como referente para posibles líneas de trabajo futuras.

Esta metodología de pruebas esta fundamentada en dos tipos de medidas de tráfico distintas pero complementarias. La caracterización del tráfico se realizará en primer lugar en base a la observación de parámetros temporales vinculados al tráfico que esté cursando la red en un determinado momento, el cual se relaciona con el proceso de handover, de una forma

totalmente transparente y sin influir en el comportamiento de la red, lo que se conoce como aplicación de medidas pasivas. Para capturar este tráfico cursado por los elementos de la red participantes en el proceso de handover, se diseñarán los métodos y aplicaciones que permitan realizar la captura en tiempo real y que realicen su procesado de forma automatizada. De forma complementaria a ésta, se hará uso de los métodos de medida activa, las cuales pueden modificar el comportamiento de la red introduciendo tráfico en determinados puntos de la red y capturándolo en otros distintos, de forma que se pueda analizar como se ha demorado dicho tráfico y qué políticas se han de tomar.

4.3 PROCESO DE COMUNICACIÓN

4.3.1 Tunnel Broker y Dual Stack

Para realizar la conexión a IPv6, se usó el concepto de Tunnel Broker, que corresponden a túneles configurados en forma manual, y que son ampliamente utilizados por usuarios IPv6. En estos sistemas, generalmente el usuario se registra con el proveedor, entregándole su dirección IPv4 de forma de que éste pueda configurar su extremo del túnel y entregar la información necesaria al usuario de forma de establecer el túnel en su extremo. La principal desventaja con este tipo de servicios es el hecho de que si el proveedor se encuentra lejano, los tiempos de conexión (Round Trip Times) se ven afectados.

En particular, se eligió al proveedor BTExact, ubicado en Inglaterra, debido a la facilidad de configuración (especialmente cuando se usan direcciones IP dinámicas) y a la estabilidad de la conexión, además de la amplia documentación ofrecida por su comunidad de usuarios. Junto con esto, el proveedor entrega una dirección IPv6 junto a un segmento de 64 bits (/64), y soporta tráfico ilimitado con un ancho de banda del túnel de 8000 kbps. Este servicio se usa tanto en la red de origen como en la red foránea.

En forma adicional, y con propósito de analizar los resultados de distintas pruebas bajo diferentes conexiones, también se realizó una conexión a IPv6 bajo el servicio de túnel broker provisto por Freenet6, ubicado en Estados Unidos. Este servicio entrega un prefijo de 48 bits (/48), y automatiza el proceso de manejo de túneles usando el protocolo llamado Tunnel Setup Protocol (TSP), el cual permite la configuración automática del túnel bajo una arquitectura cliente-servidor. Este túnel se encuentra en la red de prueba foránea.

La conexión con el servicio de túneles se realiza desde un router que fue configurado para soportar tanto el protocolo IPv4 como el protocolo IPv6.

4.3.2 Direccionamiento

Dada la característica de multihoming, el nodo puede contener distintas direcciones IP debido a sus múltiples interfaces o debido a que una interfaz posee distintas identidades. En el caso de estar dentro de una red IPv6, es factible el utilizar los 128 bits para establecer la identidad y ubicación del nodo, de forma de establecer un direccionamiento a esta IP tanto para enviar como para recibir paquetes.

Es posible establecer dentro del nodo, dos tipos de direcciones. Una que va a ser utilizada por las aplicaciones y sesiones, que identifica al nodo como tal usada sobre la capa de transporte, y otro tipo que será asociada con las interfaces de comunicaciones presentes y usadas para la transmisión de los paquetes. Esta idea permite que las sesiones serán establecidas usando las direcciones del nivel de la capa de transporte, por lo que estas no verán un cambio si estas, a un nivel más bajo, son redireccionadas con el uso de otra dirección la que si tiene un enlace a la red asociado. La contraparte, es el uso de un nivel de encapsulación adicional a agregar en el paquete.

En los escenarios de prueba utilizados, el espacio de direccionamiento disponible fue entregado por los proveedores de los túneles de acceso.

4.3.3 Enlaces y Establecimiento de la Comunicación

Mobile IPv6 se basa en el establecimiento de túneles de comunicación entre el dispositivo móvil y el Home Agent, cuando el proceso de registro ha sido exitoso, permitiendo la mantención de la dirección como la transparencia de las comunicaciones.

a. Binding

En el proceso de binding, el MR o MN envía un mensaje de BU al HA con el propósito de registrar su nuevo enlace con la red y actualizar su ubicación y nueva identificación (por ej. su dirección IP). Este mensaje debe incluir valores como la CoA adquirida, el HA al cual se registra, la correspondiente HoA, el largo de la dirección o prefijo que el dispositivo móvil

haya adquirido. A este mensaje el HA tiene que responder con un Binding Acknowledgment, que es un mensaje de BU con el flag binding ack activado. En el caso que el enlace establecido se invalide (por ejemplo, se pierda la conectividad), este debe ser borrado de las listas de binding. Esto es logrado, seteando el campo denominado *lifetime* a 0 de los mensajes de BU a un valor corto de forma de solicitar constantemente actualizaciones o que el MN envíe nuevos mensajes de BU cuando realice un cambio en sus conexiones y de esta forma se actualicen los CoAs activos en el HA y CN.

De esta forma, una vez terminado el registro, el MN contará con dos direcciones IP: una el HoA y otra la CoA. Estas son manejadas por la capa de red.

El enlace establecido permite dirigir el tráfico desde el dispositivo o red móvil, guardando la CoA por el cual el tráfico deberá ser enviado, estableciendo un vínculo entre la IP o prefijo del nodo con una CoA perteneciente a este dispositivo. Esto permite que los paquetes puedan ser enviados siguiendo el camino definido por MR o MN → CoA → HA o CN. Ya que el paquete no es enviado directamente por una ruta predefinida entre origen y destino, se usa tunneling o encapsulación IP – IP de los paquetes, de forma de guiarlos en el camino, creando un aumento del overhead en los paquetes debido al aumento de información en estos [55]. El mismo caso ocurre cuando los paquetes deben transmitirse desde un dispositivo perteneciente a una red IPv6 a una IPv4, requiriendo la existencia de una forma de traducción que permita la comunicación sin romper las sesiones establecidas. Esta debería estar ubicada en el punto de conexión del MN o MR a Internet [56]. Este ruteo indirecto hace que los mensajes sean encapsulados antes de llegar al CoA respectivo y desde allí son desencapsulados y enviados a su destino original.

Un caso especial merece el registro de múltiples CoAs bajo una misma HoA, caso que no es soportado en el documento Mobility Support [1], que trata este tema. Este punto debe ser registrado mediante un nuevo mensaje sub-Option en el BU, enviado desde el MN, y que será almacenado en el nodo receptor del mensaje de BU. Estas direcciones se manejan a nivel de red, y no deben afectar el normal funcionamiento de las aplicaciones, quienes sólo ven la IP destino a quien transmitir o que la IP por la que transmiten es su HoA. Para el registro de estos enlaces es necesario asignar un identificador especial, denominado en la literatura como BID (Binding Identifier), el cual permite el registro de distintos enlaces (entiéndase CoAs)

con un mismo HoA y debe ser almacenado en las listas de BU junto a la información adicional enviada en los mensajes de Sub-Option.

En el caso de existir distintos enlaces de comunicación establecidos posibles de elegir, es necesario un método de diferenciarlos al momento de utilizarlos. Una solución es almacenarlas junto a una prioridad dinámica, que puede depender de parámetros como el tipo de conexión, tasa de errores, entre otros, de forma que se adapte a las condiciones cambiantes del entorno en que se encuentra el dispositivo.

Lo importante en este paso es procesar y obtener de forma adecuada la información de registro enviada en estos mensajes, de forma de facilitar el proceso de selección de la CoA y HoA adecuada en cada caso.

b. Envío de Paquetes

Los nodos, ya sea el CN o el HA, usan información almacenada en el Binding Cache para conocer la CoA del dispositivo móvil a cual dirigirse.

En el caso de registro de múltiples enlaces, por defecto una sola interfaz es usada, y las restantes se mantienen de respaldo. Sin embargo, mediante la aplicación de políticas sobre las transmisiones es posible la utilización de todas las interfaces simultáneamente. La forma de separar el tráfico es mediante separación de flujos o conexiones, y no por paquetes, lo que evitará generar un proceso adicional de reordenamiento de paquetes que demoraría el proceso de comunicación. Para mejorar tanto el proceso de envío como recepción, los flujos se separarán preferentemente tanto en el nodo emisor como en el receptor, de forma que ambos deberán entender los mensajes de registro de flujos y deberán estar capacitados para almacenarlos.

Un punto es la construcción de una función que permita monitorear los vínculos de forma de tomar una acción de cambio de ruta a tiempo. Por ejemplo, tomando en cuenta los valores de variables de la conexión como demora, sincronización de tcp o tiempo de ida y vuelta, es posible ir midiendo la calidad de los enlaces constantemente y tener una base para compararlos en general. El objetivo es evitar la perdida de paquetes y disminuir la demora en la conexión [57].

c. Módulos de la Arquitectura

En el proceso de comunicación, es necesario tener almacenada información como la

siguiente:

- Lista de los múltiples HA con las que se esta registrado.

- Lista de las CoA adquiridas. Estas son almacenadas con su respectiva prioridad de uso. La más alta, tienen la preferencia de ser usada como la predeterminada.

- Lista de los mensajes de Binding Update, manteniendo un registro de los CN con los cuales se tiene comunicación. -Binding Cache. Almacena los destinos conocidos a los cuales enviar mensajes.

Como se ha mencionado, la arquitectura permite la comunicación entre las capas que intervienen en el envío de paquetes a la red. Esto permite que el manejo de los datos de las distintas capas (de la de aplicación hacia abajo) que tengan relevancia en el proceso de transmisión se maneje globalmente, reduciendo el overhead que se produciría si la información tuviese que ser transmitida en todo momento e informando a partes que no la utilizarían, lo cual ralentizaría el proceso [58]. Esta debe contener:

- Información de estado.- Obtenida en forma dinámica desde el ambiente, siendo utilizada para tomar decisiones (por ejemplo: fuerza de señal, nivel de tráfico).

- Información de recursos.- Esta permite configurar parámetros y recursos que están disponibles en todas las capas (por ejemplo: dirección IP, tablas de ruteo y tasa de transmisión entre otros).

Junto a la información disponible, es necesario definir la forma en que esta será modificada. Para lograr esto es posible definir, en forma general, un proceso basado en 3 métodos principales: Método Capturar () el cual permite escuchar continuamente cambios en la información y tomar acciones dinámicamente; Método Actualizar () utilizado para guardar o actualizar información; Método Obtener () el cual permite la recuperación de información. Por ejemplo, cuando el MN recibe una IP desde la red foránea, este envía un mensaje Actualizar () a la interfaz, de forma de registrar este hecho.

Ya que la información es de alcance de todas las capas, se deben manejar un grupo de permisos de forma de no afectarla por accesos indebidos por otras capas del sistema. Tanto el paso de mensajes como el manejo de los permisos, puede ser manejado, como alternativa, a través de sockets.

En cuanto a los cambios de interfaz, es necesario capturar el estado de los enlaces, las direcciones IP que se han obtenido, las entradas de ruteo, para formar una visión global de la conectividad de Internet en cada dispositivo de conexión. Principalmente, la información para detectar los cambios va a ser entregada por la capa 2.

Para esto, la información que será recolectada será usada en:

- Recolectada en Capa 2 (Enlace) y Capa 3 (Red)
    - Política de Selección
    - Información de Ruteo
    - Selección de CoA
    - Mantención de Binding Cache y HoAs.

- Recolectada en Capa 4 (Transporte/Sesión/Presentación)
    - Información de Sesiones
    - Origen y Destino de las comunicaciones
    - Direcciones y Puertos

Cuando se descubre una nueva conexión, esta provoca un cambio en el estado de las interfaces y de la información almacenada relacionada a esta. A la vez, puede generar la creación de una nueva política o la actualización de alguna existente, redefiniendo sus prioridades o su criterio de selección.

En algunos casos, donde la disponibilidad de una conexión dependa de una acción a tomar por parte del usuario, como la activación de una nueva interfaz, o el uso de una conexión pagada, que requiera de un acceso especial (por ejemplo especificando una clave) o que este prefiera una interfaz a otra, es relevante que este cuente con una forma de interactuar con la

forma de transmisión. Por esto, algunos datos que pueden ser útiles que la arquitectura domine son:

Lista de interfaces actuales o disponibles
Detalle de las interfaces
Información del sistema
Menú de acciones disponibles (interacción con la transmisión)

Debido a que las decisiones de ruteo influyen directamente en el tiempo y en la forma en que los paquetes son transmitidos, no puede existir mucho tiempo en el proceso de comunicación entre estos. Pero tomando la consideración de que el entorno no cambia demasiado rápido, o las veces en que lo hace son las menos, es posible pensar que una buena alternativa es el dividir la arquitectura en un conjunto de módulos interrelacionados. Puntos adicionales a favor, producto de una composición modular, corresponden al hecho de poder crear o agregar nuevos componentes a la solución en caso de poner nuevas funcionalidades, de crear compatibilidad con otro tipo de tecnología o de establecer un estándar de comunicación para intercambiar información para la toma de decisiones con otros dispositivos, tema que ya ha sido mencionado como posible. En un principio el sistema está dividido principalmente en 3 grandes módulos (Figura 4-2) que pueden comunicarse entre si, en los cuales es posible mapear los métodos de control de información definidos anteriormente.

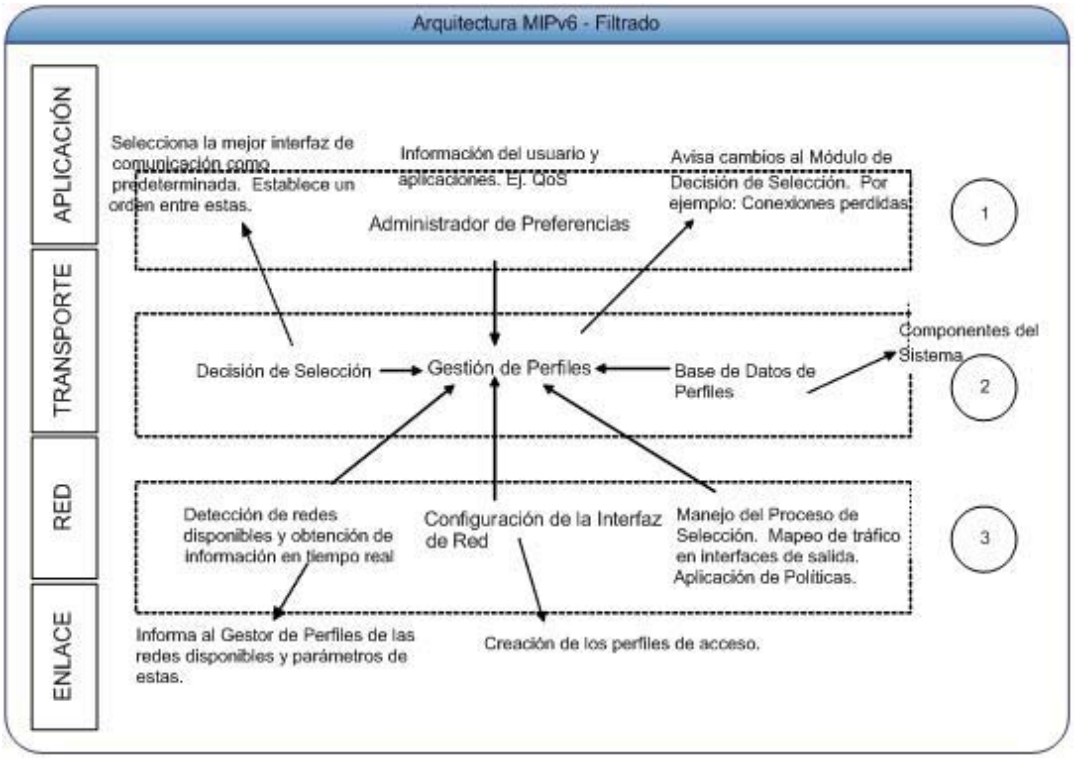


Figura 4-2. Módulos de la Arquitectura.

- Módulo 1. Encargado de:

  ◦ Que los recursos disponibles sen ocupados según las preferencias establecidas por el usuario o administrador del sistema o por las características necesitadas por las aplicaciones.

  ◦ Manejar los parámetros requeridos por las transmisiones como es el caso de una QoS dada.

  ◦ Realizar seteos y configuraciones necesarias en las capas 2 y 3 para establecer la conexión y lograr el acceso a la red.

  ◦ Especificar las características de las interfaces de red disponibles.

- Módulo 2. Encargado de:

  ◦ Generación de políticas y prioridades según la información que va siendo recolectada.

  ◦ Ordenamiento de las interfaces.

- Módulo 3. Encargado de:
    - Redirección del tráfico en las interfaces según los parámetros y políticas establecidas.
    - Manejo de Binding Cache, y listas de enlace.
    - Cambios en las características de los enlaces (Figura 4-3). Chequeos constantes.

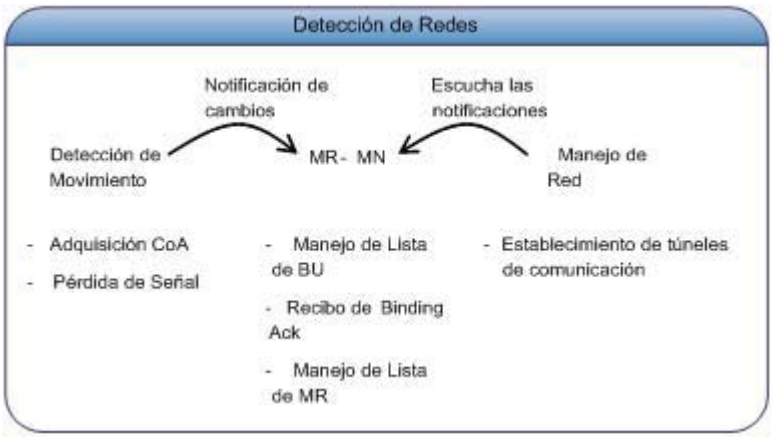


Figura 4-3. Sub sección de Detección de Redes.

Los perfiles facilitan el manejo de los recursos conocidos en tiempo real, permitiendo la evaluación de parámetros provenientes de la red como las obtenidas de las especificaciones propias de la conexión, como otras adicionales, como el throughput y/o delay. De igual forma, maneja el perfil de acceso a las distintas tecnologías disponibles, ya sean claves o proveyendo información necesaria para la correcta configuración de éstas **[59].**

En el módulo de Decisión de Selección (Módulo 2) se recomiendan las interfaces acorde a los perfiles definidos y a los valores que son obtenidos continuamente de los parámetros de las comunicaciones establecidas. Estas interfaces pueden ser las preferidas, o por defecto, que se encuentran configuradas como interfaces que pueden estar activadas y en uso, o aquellas que son agregadas durante la comunicación.

Para llevar a cabo lo anterior, en el proyecto se propone la construcción de una arquitectura actuando como middleware entre la capa de transporte y de red, que permita manejar la información mencionada y que se encuentre encargada del proceso de asignación de flujos de transmisión a cada interfaz del dispositivo.

d. Políticas

Para un correcto uso de la información que fue recolectada, es necesario el uso de políticas de ruteo. Un punto importante es la selección de la interfaz apropiada por la cual transmitir. En este sentido, referirse a la selección de la interfaz es lo mismo que la selección de la CoA adecuada por la cual enviar los paquetes. Esta debe ser elegida según las políticas definidas y la disponibilidad de las interfaces, siendo monitoreado en tiempo real.

Con la distinta información recolectada, las políticas se pueden implementar haciendo uso de:

Tipo de protocolo.
Puerto (origen – destino).
IP (Origen – Destino).
Identificación del flujo.
Tipo de dirección (pública o privada) y tiempo de vida de ésta.
Fuente del enlace (uso de HoA o de CoA, ya que generalmente se prefiere un enlace directo).

La información debe ser almacenada en una estructura, actuando como una base de datos de políticas en el MN. Esta información, potencialmente puede ser transmitida al HA y/o CN para que el recibir paquetes también se realice de forma optimizada en caso que estos dispositivos conozcan múltiples rutas hacia el MN. En el caso que las políticas deseen ser transmitidas, estas, por ejemplo, se pueden adjuntar a los mensajes de BU estableciendo un mensaje de sub-option de políticas (Figura 4-4), las cuales deben corresponder a todas las políticas que se deseen transmitir por interfaz. Estas políticas deberán ser almacenadas con el correspondiente binding del MN en el binding cache de cada entidad, o haciendo referencia a esta información, en especial a la relativa a las interfaces en uso. Si se quieren eliminar o actualizar, siguen el proceso común de actualización de un mensaje Binding Update, haciendo referencia a los identificadores de estas. En este ámbito, los componentes principales referente a políticas, el repositorio, como el ejecutor de estas, se encontrarán en cada punto y sólo intercambiaran información cuando se detecten cambios en los enlaces aplicables a ambos extremos y no a cada momento para consultar por cada evento surgido en la transmisión, reduciendo de esta forma los tiempos de aplicación de las políticas. Es decir, según los términos del framework de políticas, tanto el PDP como el PEP permanecerán

específicos a cada equipo móvil, y transmitiéndose entre los PDP sólo las políticas que se consideren aplicables de ámbito global (que permitan mejorar la recepción de paquetes).

Figura 4-4. Proceso de Ruteo por Políticas.

En los mensajes de BU hay que tener especial cuidado en el número de sub-options utilizados y enviados, ya que el tamaño de estos puede afectar el desempeño de la red si se cuenta con una conexión lenta. De igual forma, es necesario considerar el efecto del overhead adicional provocado por el uso de encapsulación al momento de pasar entre redes o al dirigir los paquetes a través de CoAs.

Una forma alternativa, es transmitir las políticas mediante una transmisión normal, por ejemplo mediante el uso de sockets bajo TCP, y enviar toda la información en un formato determinado de forma de facilitar su almacenamiento. Esta opción permite sólo enviar los mensajes de actualización de políticas cuando se detecten cambios, y no tenerlos asociados al proceso de registro de BU, reduciendo el tamaño de los mensajes y frecuencia de estos.

En ambos casos, ya sea mediante la sub-option de políticas o mediante una transmisión, se debe incluir información como:

Tipo y largo. Usado para almacenar y codificar las políticas.
Tipo de políticas. El cual define a que tipo de transmisión habrá de afectar, especificando parámetros como protocolo, dirección, puerto y flujo.
Prioridad. Valor de prioridad de la política.

Tiempo de vida. De forma de conocer cada cuanto tiempo es necesario actualizarlas.

Enlace. Ya que existen múltiples enlaces registrados, es necesario identificar el BID del enlace sobre el cual se emplea una política.

Home Address. En el caso que las políticas sean transmitidas, es necesario asociarlas con un valor que ayude en la aplicación de las políticas a una determinada transmisión.

Según el tipo de negocio en que se encuentre operando esta arquitectura, se tendrán distintas reglas que aplicar, propias a las comunicaciones mantenidas por esta organización. De esta forma, en algunas puede privilegiarse la transmisión de video (teleambulancia), mientras que en otras es importante la descarga de información en texto e imágenes (carabineros) lo que conlleva a que la arquitectura, en lo que a políticas se refiere, debe ser adaptable en forma sencilla de manera de convertir los requerimientos propios de la organización en políticas implementables y aplicables sobre las transmisiones establecidas. Es decir, se debe contar con una forma de modelar y mapear los parámetros o reglas del negocio requeridas en la comunicación, como por ejemplo demora, tasa de pérdida u otros en una forma consistente y de sencilla interpretación por los componentes de las políticas PDP y PEP, mencionados anteriormente. Un ejemplo de esto, es lo propuesto en **[60]** que además trata de proveer un marco general para modelar políticas para ser utilizado por los fabricantes de equipos móviles de forma de poder ser compartido entre dispositivos.

En forma adicional, es necesario establecer una forma de lenguaje común en cuanto a la estructura de las políticas en sí, de forma de que sean interpretadas fácilmente para ser almacenadas o actualizadas en el equipo móvil. De igual forma, la notación elegida debe ser lo suficientemente clara como precisa, de forma de no sobrecargar los mensajes intercambiados y por lo tanto la conexión. Alternativas en este punto son usar un formato establecido y que sea necesario pasar parámetros como en la notación EBNF **[61]**, o un formato un poco más detallado como XML que permita mayor flexibilidad en cuanto al ordenamiento e inclusión nuevos parámetros permitiendo la extensión de los tipos de políticas a aplicar, pero que causará un mayor tamaño de los mensajes transmitidos.

Otros puntos que son necesarios de considerar para incluirlos en el proceso de construcción de las políticas son:

Conciencia de la velocidad del medio de conexión usado como del enlace por el cual se estableció la conexión al igual que al hecho de que los enlaces generalmente no presentan la mismas prestaciones para tráfico saliente como al entrante, por lo que hay que tener en cuenta esto al momento de seleccionar los túneles mediante las políticas y hace recomendable que los dispositivos presentes en la comunicación, MN o HA o CN puedan también efectuar un proceso de selección al momento de enviar información por los distintos caminos disponibles hacia el destino.

Continuo chequeo de las interfaces y del estado de las conexiones existentes. Es decir, cuando una conexión aparece o desaparece como cuando una interfaz es agregada o eliminada del MN.

Cambios de preferencias provenientes del usuario o externos al sistema. Ya sean en selección de interfaces o relacionado a las comunicaciones con CN específicos.

Detección de correcto funcionamiento de las interfaces, por ejemplo pérdida de comunicación, tasa de error demasiada alta o similar.

Comparar constantemente las prestaciones de las comunicaciones por las interfaces con las preferencias predefinidas ya sea que los sobrepasan o que entran dentro de los límites establecidos.

Cualquier cambio en estos criterios o en otros que puedan ser agregados, produce un reprocesamiento de los costos y prioridades asignadas a las políticas y un reordenamiento para su selección.

e. Evaluación

Principalmente es enfocada en dos parámetros: en tiempo (como el empleado en el envío de paquetes) como costo (como procesamiento adicional para manejar las políticas) requerido para obtener la información como el tiempo de detección y adaptación ante cambios. Como base para estos chequeos se usa el envío de ráfagas de mensajes del tipo ICMPv6 (PING), repetidas con una cierta frecuencia, para chequear estado de los canales accesibles por las diferentes interfaces, debido a los distintos tipos de información que entrega.

Las extensiones a los mensajes de binding pueden no causar mayor impacto en los tiempos involucrados. Esto considerando que para almacenar las políticas puede usarse alguna

codificación como algoritmos Hash o similares, de manera de hacer más liviano su almacenamiento como procesamiento. Otra alternativa es utilizar un lenguaje XML de forma de facilitar su portabilidad entre distintos dispositivos. Para almacenarlos y buscarlos es posible la utilización de listas enlazadas. Esta estructura también es posible de utilizar para el registro de las conexiones establecidas. Por ejemplo, en el caso de las conexiones y/o interfaces siendo utilizadas, información importante a almacenar puede ser:

tipo de tecnología
operador/costo
estado
información/identificación de la interfaz
preferencia/prioridad

Con la información almacenada de las interfaces es posible poblar las políticas en forma dinámica (Figura 4-5), resultando con un esquema como el siguiente:

Figura 4-5. Estructura de Almacenamiento de Políticas.

Lo anterior permite que exista un orden tanto en la selección de políticas como de las interfaces, de forma que si alguna de ellas falla (ya no se encuentra disponible o es imposible aplicarla) se pasa a la siguiente o a una acción por defecto.

En cuanto a herramientas a usar en la implementación y en la posterior simulación y prueba de la arquitectura, se evaluaron aquellas que poseyeran un alto y constante nivel de desarrollo y que se adecuaran a los estándares más actuales desarrollados, por ejemplo, documentos RFC o DRAFT que se encuentren vigentes.

Entre las alternativas que se evaluaron se encuentran:

- Mobile IPv6
  - Linux MIPL HUT
  - Linux NEPL (Proyecto Nautilus)
  - FreeBSD KAME

- Filtros:
  - IPFilter en BSD
  - Netfilter en Linux.

CAPÍTULO 5
IMPLEMENTACIÓN.

5.1 IMPLEMENTACIÓN ELEGIDA

5.1.1 Sistema Operativo e Implementación de MIPv6

Hoy en día, existe un gran desarrollo en torno a IPv6 y MIPv6, por lo que diversos sistemas ofrecen soporte de estos protocolos, entre ellos Windows, Linux, BSD, HP UX, Sun Solaris y MacOS.

A su vez, existen distintas implementaciones de MIPv6 en desarrollo. Entre ellas MIPL (Mobile IP for Linux), NEPL, KAME, SHISA y SHIM6 son las más recurrentes. De estas, se eligió NEPL (NEMO Platform for Linux), la cual está basada en MIPL (poseen grupos de desarrollos relacionados) a la que se le agrega el soporte de NEMO (movilidad de redes además de un sólo equipo) y MCoA (registro de distintos enlaces por los cuales trasmitir), dando soporte completo a los RFC 3775 (Mobile IP) y RFC 3963 (Network Mobility). A favor de esta elección se encuentra que es aquella, junto a MIPL, que más soporte y documentación relativa posee y que se encuentra en constante desarrollo, siendo la última versión desarrollada en enero del año 2007. Entrega soporte a las distintas entidades de la red, ya sean HA, MN, MR o CN. Por otro lado, como es un proyecto de código abierto, no existen restricciones en cuanto a su uso o modificación.

La versión usada en esta etapa corresponde a nepl-0.2-mcoa-beta3-20070118 [62]. En esta versión, es permitido realizar filtrado y aplicación de políticas mediante la herramienta ip6tables además de soportar el registro de múltiples CoAs. Sin embargo, no es soportado el uso de IPsec ni el uso de la ruta optimizada en la comunicación entre el MN y el HA, cuando se maneja el registro de múltiples enlaces.

La elección de esta solución permite abordar una serie de objetivos que se han planteado a desarrollar, específicamente los relacionados a los mensajes de registro y soporte de múltiples

interfaces. NEPL funciona a nivel de la capa de red, soportando además las funciones relacionadas al control de las distintas interfaces del dispositivo móvil. Sin embargo, el área relacionada al manejo y control de políticas no se encuentra desarrollada, así que los mayores esfuerzos de diseño e implementación se enfocan en ese tema, y en su coexistencia con el protocolo de movilidad.

Una vez elegida esta, se procedió a elegir el sistema operativo Linux, que es donde la implementación seleccionada puede ser instalada.

5.1.2 Red de Prueba Inicial

a. Red

Uno de los criterios que guiaron la creación de este escenario de pruebas fue el hacerlo lo más real posible, es decir, usar características de diseño que tendrán que ser estudiadas por cualquier organización que decida la introducción de IPv6 a sus redes.

Como se mencionó, se dispuso de un segmento de 64 bits, entregado por el servicio de tunnel broker. El segmento utilizado corresponde a 2001:618:400:69ef::/64 para la red de origen, y al segmento 2001:618:400:2e29::/64 para la red foránea.

En cuanto al ancho de banda disponible en las redes, la red origen tiene una capacidad de 10 Mbps y la red foránea, de 512 Kbps, para conexión con Internet bajo IPv4; y de 8000 kbps para conexión bajo IPv6. Referente a la conexión interna, ambos equipos usados como HA se encuentran conectados con los routers internos mediante interfaces de 10/100 Mbps. En cuanto a los equipos móviles, estos tienen interfaces bajo 802.11g (con una tasa de transmisión hasta 54 Mbps) e interfaces 10/100 Mbps para conectarse a su red móvil propia. Ambas redes soportan tanto IPv4 e IPv6, debido a que el uso de este último protocolo no será tan masivo como para afectar las conexiones o recursos asignados a IPv4.

Como se muestra en la Figura 5-1, la red de origen estuvo conformada por un router, y dos equipos, uno actuando como HA y otro como MN. Lo mismo sucede en la red foránea. Ambas soportan el protocolo NEMO y registro de MCoA, además de permitir unirse a la red IPv6 varios equipos, ya sean fijos como móviles.

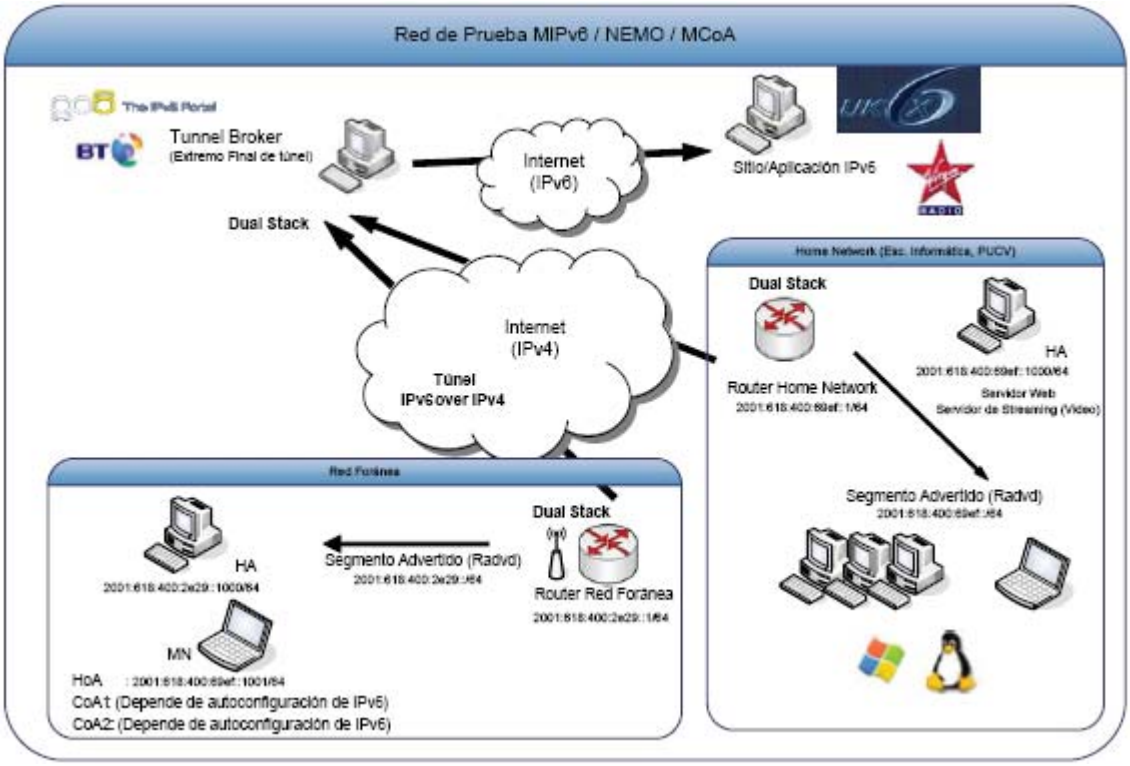


Figura 5-1. Red de prueba MIPv6.

Una vez adquirida una conexión IPv6, es posible hacer uso de aplicaciones que ya han sido portadas a IPv6 y acceder a servicios sobre esta red. Para probar la correcta conexión y configuración de IPv6 se hizo uso de aplicaciones como SSH (Acceso Seguro), Firefox (Navegador Web), XMMS (Reproductor Multimedia) y VLC (Reproductor y Servidor de Streaming de Video), con los cuales se pudo acceder a servicios en línea bajo IPv6 como reproducción de videos y televisión [63, 64], como de música [65].

Uno de los objetivos principales para usar este tipo de escenario, es el poder probar el desempeño del protocolo en un ambiente entre-sitios (conectados a través de Internet) y no dentro de un sólo laboratorio, permitiendo el análisis de distintas variables y situaciones que influyen normalmente en una transmisión, además de un mayor número de servicios posibles de brindar sobre IPv6, los que pueden ser probados juntos, bajo condiciones reales de conectividad.

En forma adicional, debido a la facultad de autoconfiguración de IPv6, la red de prueba puede ser usada por cualquier equipo conectado a estas redes, facilitando la experimentación con este protocolo.

b. Router

Es el equipo encargado de conectar la red local IPv6 con Internet. El equipo utilizado es un router modelo Linksys WRT54G. Este equipo posee capacidades de Access Point, Switch, Router y Firewall. Una de sus principales características es que está construido sobre un sistema Linux, lo que permite agregarle nuevos módulos y extender sus funcionalidades.

Como se ha mencionado, un equipo que tenga que convivir con los dos tipos de protocolos, necesita contar con la modalidad dual stack. Esta es posible de obtener gracias a la distribución OpenWRT [66], la cual puede ser instalada en el equipo por estar bajo la licencia GPL. OpenWRT es una distribución basada en Linux que permite instalar en el router sólo los paquetes que uno desee, además de permitir al usuario acceder a la configuración y al sistema de archivos del equipo, entregando una alta flexibilidad.

Los paquetes necesarios a instalar (además de la instalación básica por defecto) son:

- *kmod-ipv6* para el soporte de IPv6.

- *wl* para la configuración de la interfaz inalámbrica.

- *ip* para la configuración de direcciones y rutas IP.

- *ip6tables* para la definición de políticas de ruteo.

- *Radvd* para el envío de mensajes de router advertisement con el prefijo de la red.

c. Home Agent

Este equipo corre sobre un sistema Linux, que posee la implementación de MIPV6, NEPL. Está conectado por una interfaz cableada con el router y posee asignada la dirección 2001:618:400:2e29::1000/64.

Para monitorear su desempeño, se instalaron las herramientas MRTG [67] y NTOP [68], las cuales permiten seguir el funcionamiento de la red y las conexiones en el tiempo. En la sección de Apéndices, punto C, se encuentran capturas relativas al desempeño de la red IPv6 en funcionamiento. En el escenario de prueba sólo se consideró un MN como miembro de la red, además de los equipos fijos (sin soporte de movilidad) pertenecientes a la red del Home Agent. Las herramientas usadas para la medición constante del desempeño permiten medir el tráfico de la red a nivel de protocolo, la carga del procesador y la memoria usada en el Home Agent. En forma adicional, se desarrollaron distintas pruebas enfocadas en la medición de la demora adicional incurrida en las transmisiones realizadas bajo IPv6 en comparación con IPv4.

Adicionalmente, se desarrolló un sitio Web sobre el equipo (Figura 5-2), con acceso por resolución de nombre sólo bajo IPv6, con acceso a distintas utilidades y recursos utilizados como al material relacionado con presente proyecto, como los documentos RFC y DRAFT que se tuvieron como base en la investigación desarrollada.

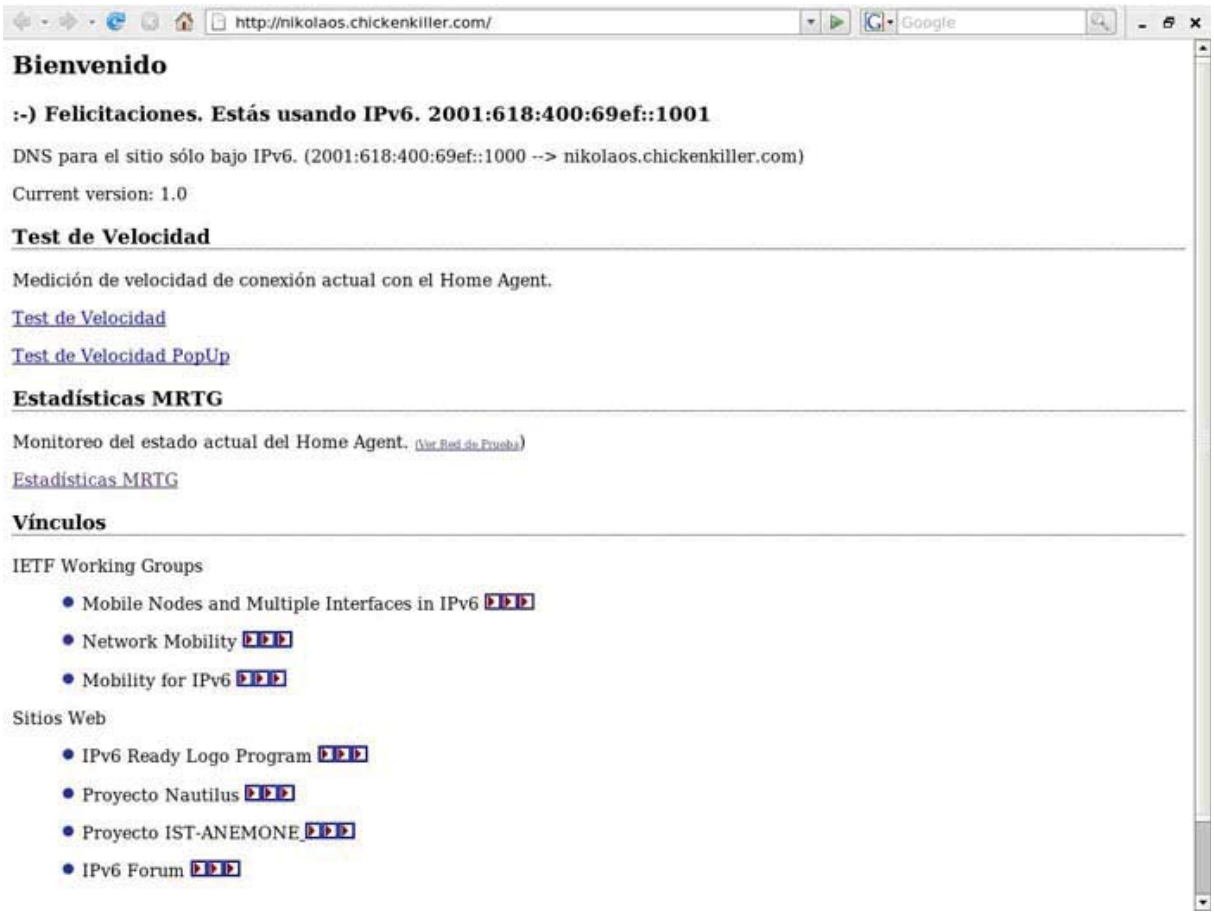


Figura 5-2. Página de inicio del sitio Web desarrollado.

Para las pruebas de archivos de video, el equipo de HA actuó como servidor de streaming, permitiendo la conversión de las opciones de transmisión, resolución y parámetros de los videos.

En la Figura 5-3 se muestra una captura del proceso de recepción del mensaje de BU enviado por el dispositivo móvil al HA, y en la Figura 5-4, el registro de las interfaces que queda almacenado en el binding cache de esta entidad, con el respectivo valor de BID que será usado por las políticas para identificarlas.

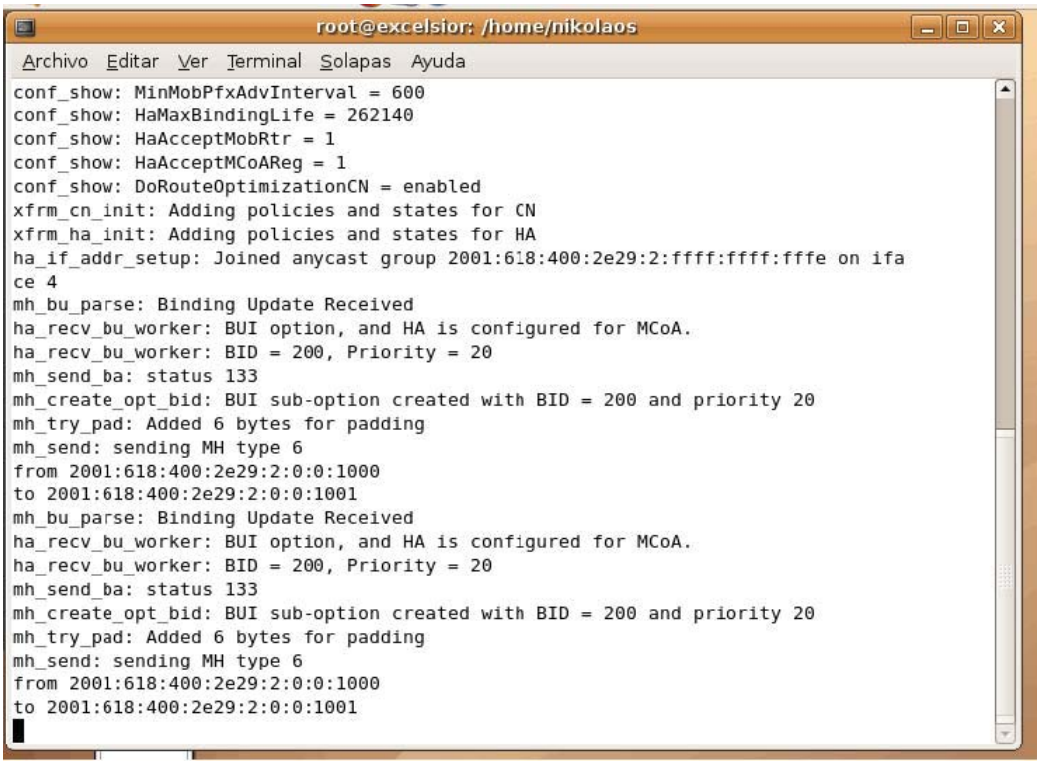


Figura 5-3. BU recibido en el HA.

```
root@nleiva-ucv:/var/www/archivos# telnet localhost 7777
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
mip6d> hal
eth0 2001:618:400:69ef:0:0:0:1000
 preference 10 lifetime 1800
mip6d> bc
hoa 2001:618:400:69ef:0:0:0:1001 status registered
 coa 2001:5c0:9738:0:0:0:0:4321 BID 200 BidPriority 20 flags AH--
 local 2001:618:400:69ef:0:0:0:1000
 lifetime 12 / 60 seq 60515 unreach 0 / 134775712 retry -2
hoa 2001:618:400:69ef:0:0:0:1001 status registered
 coa 2001:5c0:9738:0:0:0:0:1234 BID 100 BidPriority 10 flags AH--
 local 2001:618:400:69ef:0:0:0:1000
 lifetime 37 / 60 seq 23617 unreach 0 / 134775712 retry -2
mip6d>
```

Figura 5-4. Registro de múltiples interfaces en HA.

d. Nodo Móvil

El equipo de prueba usado (notebook) se encuentra configurado como MR, sobre un sistema Linux. Cuenta con una interfaz cableada para conectar los equipos clientes (en su

red móvil propia) y con dos interfaces inalámbricas (de 54 Mbps, correspondiente al estándar 802.11b/g), las cuales son usadas para tener comunicación a través de los puntos de acceso. Cuando se encuentra en la red de origen, tiene asignada la HoA 2001:618:400:2e29::1001/64. En caso contrario, genera una CoA mediante autoconfiguración según el prefijo recibido en los mensajes de RA. Esto se ve reflejado en las Figuras 5-5, 5-6 y 5-7.

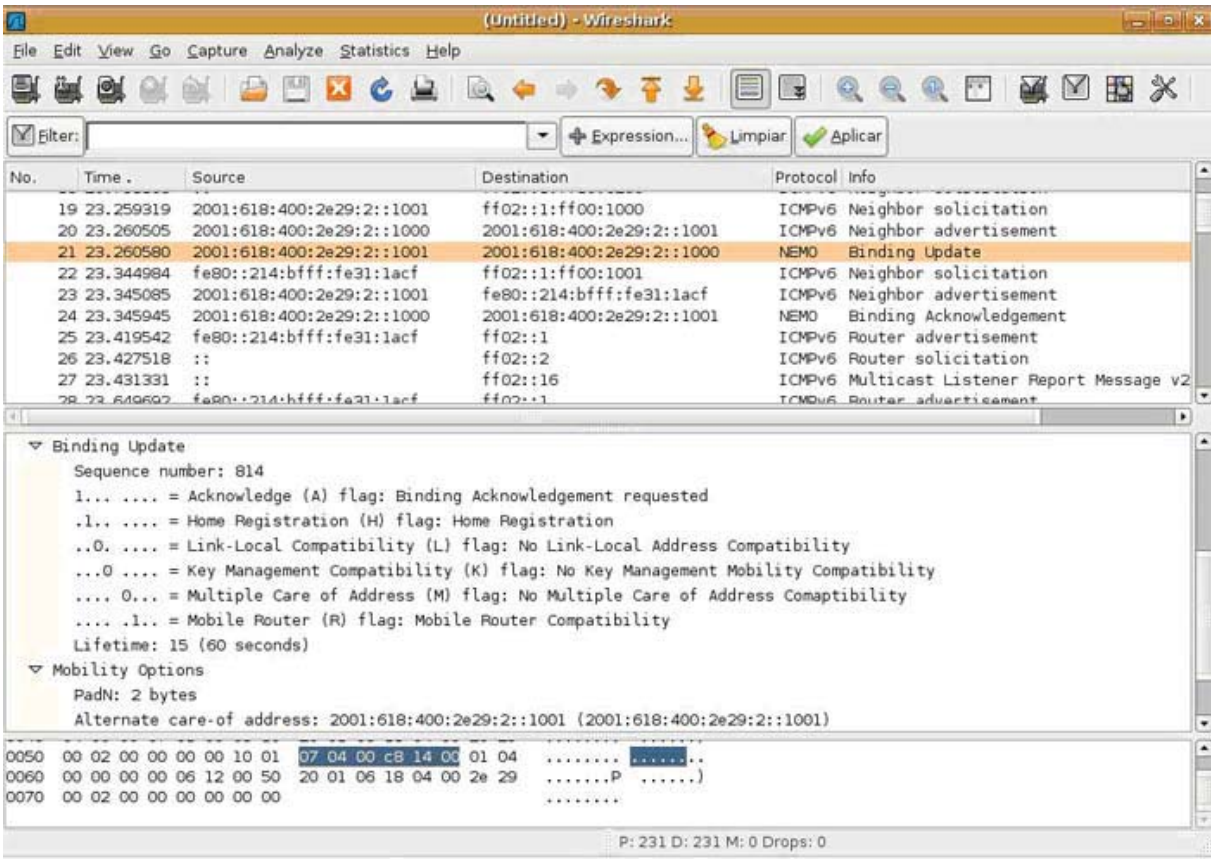


Figura 5-5. Secuencia de mensajes para completar un nuevo registro de CoA.

```
eth1      Link encap:Ethernet  HWaddr 00:14:A5:10:62:D0
          inet addr:192.168.1.151  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::214:a5ff:fe10:62d0/64 Scope:Link
          inet6 addr: 2001:618:400:2e29:214:a5ff:fe10:62d0/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:543 errors:0 dropped:0 overruns:0 frame:0
          TX packets:550 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:339342 (331.3 KiB)  TX bytes:127789 (124.7 KiB)
          Interrupt:209 Memory:d0204000-d0206000

ip6tnl1   Link encap:UNSPEC  HWaddr 20-01-06-18-04-00-2E-29-00-00-00-00-00-00-00-00
          inet6 addr: 2001:618:400:69ef::1001/64 Scope:Global
          inet6 addr: fe80::214:a5ff:fe10:62d0/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP  MTU:1460  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:440 (440.0 b)
```

Figura 5-6. Túnel creado tras el registro de nueva CoA.

```
root@notebook:/home/nikolaos# telnet localhost 7777
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
mip6d> bul
== BUL_ENTRY ==
Home address    2001:618:400:69cf:0:0:0:1001
Care-of address 2001:5c0:9738:0:0:0:0:4321
CN address      2001:618:400:69ef:0:0:0:1000
 lifetime = 60,  delay = 57000
 BID = 200, Priority = 20
 flags: IP6_MH_BU_HOME IP6_MH_BU_ACK
 ack ready
 lifetime 33 / 60 seq 60517 resend 0 delay 57(after 31s)
 mps -183 / 0
== BUL_ENTRY ==
Home address    2001:618:400:69ef:0:0:0:1001
Care-of address 2001:5c0:9738:0:0:0:0:1234
CN address      2001:618:400:69ef:0:0:0:1000
 lifetime = 60,  delay = 57000
 BID = 100, Priority = 10
 flags: IP6_MH_BU_HOME IP6_MH_BU_ACK
 ack ready
 lifetime 58 / 60 seq 23619 resend 0 delay 57(after 56s)
 mps -183 / 0
mip6d>
```

Figura 5-7. Estado del Binding Cache en el MN.

5.1.3 Política de Filtrado

Una sección faltante a la implementación de protocolo elegida, NEPL, es el control de las interfaces y la asignación de tráfico a ellas en forma ordenada. Un esquema se encuentra en la Figura 5-8, y el código fuente se encuentra en el punto D, de la sección de Apéndices. Esta implementación actúa como intermediaria entre la capa de transporte y de red, permitiendo un manejo más inteligente de las transmisiones establecidas y de las interfaces disponibles.

Básicamente, el proceso consiste en usar las interfaces activadas por el protocolo de movilidad, priorizando el envío de ciertas transmisiones, calificadas como prioritarias por el usuario, por aquellas interfaces que poseen una mejor calidad. En esta parte, calidad se entiende como factor de baja latencia, baja tasa de paquetes perdidos y cercanía con el otro extremo de la transmisión. Esto se logra monitoreando constantemente las interfaces en uso a nivel de las capas de enlace y de red, como detectando las nuevas que han logrado adjuntarse a una red, de manera de poder priorizarlas. Una vez que estas se encuentran clasificadas, en el caso de contar con más de una conexión activa, se procede a asignar los tráficos establecidos

a las interfaces según su prioridad, acorde a los valores medidos y previamente definidos, las que son posteriormente transmitidas al HA de la red.

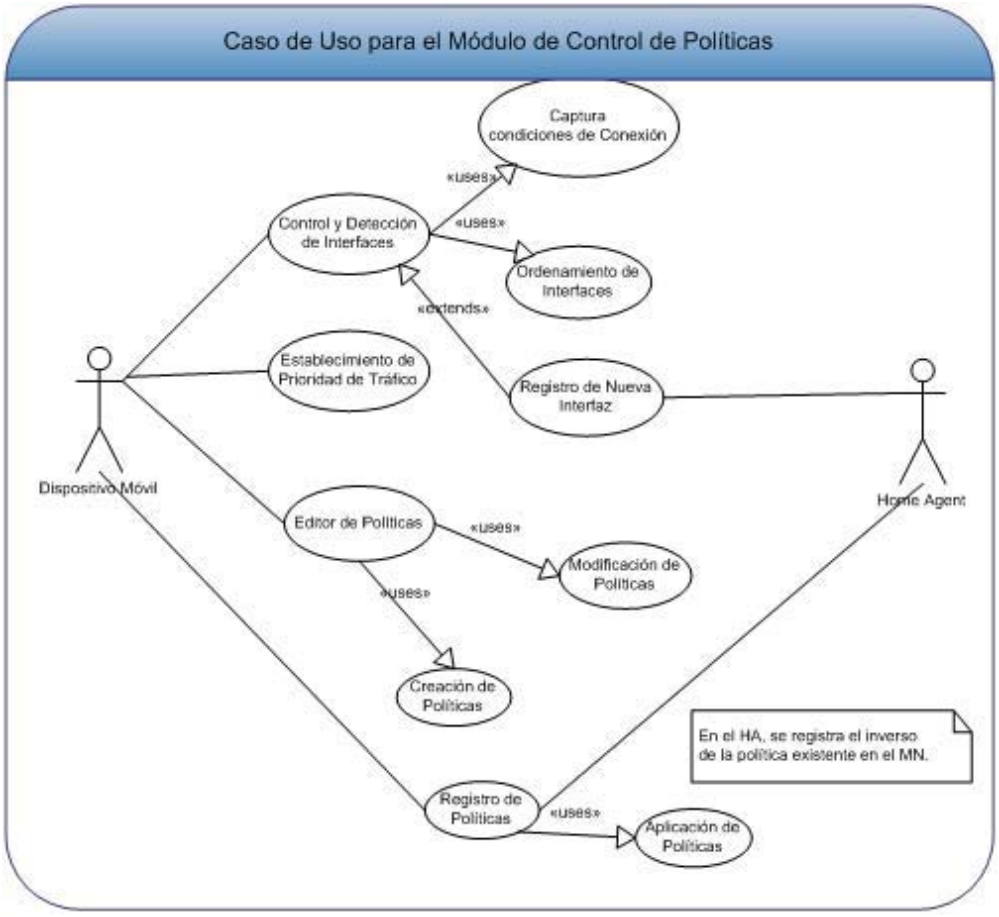


Figura 5-8. Módulo de Políticas implementado.

Dado el carácter de movilidad, uno de los parámetros elegidos para la medición de la calidad de las interfaces corresponde a la latencia existente en la red a través de una cierta interfaz. Este valor nos da una medida de la demora que afectará al tráfico saliente, ya que se ve influido por factores como la velocidad de conexión y la demora en la comunicación producida entre los routers existentes en el camino de la transmisión. Otro valor usado es el correspondiente al TTL, que nos entrega información acerca de los distintos saltos en los que debe incurrir un paquete transmitido en llegar a su destino. La elección de este criterio se basa en que mientras más equipos existan entre los extremos de la comunicación, existen más puntos de riesgo que pueden fallar y afectar la calidad de la transmisión.

Un tercer criterio considerado en la medición, corresponde al porcentaje de los paquetes perdidos, y que es una medida directa de la calidad del canal, indicando que al menos una

parte de este se encuentra congestionado, lo que puede influir significativamente en el desempeño de transmisiones orientadas a la conexión, como TCP. Particularmente, en conexiones inalámbricas, los paquetes pueden verse afectados por distintas razones como interferencias, ruido o errores en los canales usados.

Estos tres criterios son integrados en una función de ordenamiento (Ecuación 5-1) a través del uso de factores de importancia de cada uno de ellos, por ejemplo:

$$Factor\_elección(I) = \alpha * Paquetes\_Perdidos(I) + \beta * RTT(I) - \chi * TTL(I),$$

donde I es la interfaz a medir.

$$\alpha, \beta, \chi \in [0,1], \alpha + \beta + \chi = 1$$

Ecuación 5-1

lo que entrega un valor cuantificable que permite ordenar las interfaces de menor a mayor, donde la que posee el menor valor dado por Factor_eleccion() corresponde a la mejor interfaz disponible en cuanto a su consistencia y estabilidad, sin embargo, los valores también son manejados en forma separada, asignándose a cada interfaz. Las restricciones impuestas en la ecuación, son para ajustar los valores a los usados en la etapa de pruebas, pero pueden ser obviadas o modificadas sin mayor problema. En el caso de los paquetes perdidos y el tiempo de ida y vuelta, mientras menor sean estos valores indican una mejor eficiencia en la red utilizada. En cambio, el valor de TTL se considera mejor, mientras más cercano se encuentre al valor original colocado en los paquetes salientes. El hecho de que los valores medidos varíen constantemente entre mediciones puede indicar que un enlace tiene un comportamiento impredecible, por lo que su uso debe ser limitado. Para el caso de las pruebas, estos valores fueron ajustados según experimentos sobre distintos tipos de transmisiones, especialmente, de video, conexión segura (SSH) y transmisión vía Web, estableciendo rangos de corte, como por ejemplo un máximo de 5% de tasa de error en la transmisión de video. En forma general, los factores pueden tomar cualquier valor numérico, según la influencia que se quiera dar a cada parámetro. Sin embargo, en los casos de prueba se puso la restricción de que la suma de los factores debe ser igual a 1. Los valores elegidos se encuentran en el código adjunto, presente en el punto D, de la sección de Apéndices.

Para el chequeo de estos tres criterios, se usan ráfagas de mensajes tipo ICMPv6, los que permiten medir, transmitiendo pequeñas cantidades de información, los valores de tiempo de retorno, números de saltos y la fiabilidad del canal (considerada como la tasa de pérdida de paquetes o si los nodos son inalcanzables entre sí), debido a tratarse de mensajes del tipo

solicitud y respuesta. A pesar de que estos mensajes pueden ser filtrados o no mostrar los tiempos reales de demora, se optó por ellos debido a su extenso uso en pruebas relativas a redes y que son ampliamente entendidos por distintos tipos de sistemas operativos y por equipos de red, evitando problemas de compatibilidad y el incluir aplicaciones adicionales que no se encuentren integradas en estos equipos.

Esto permite, en el caso que se tenga algún conocimiento anterior del estado de red en el cual se establece la conexión, adecuar los factores a esta información, entregándonos una variación continua acerca del desempeño de la red, y como será el funcionamiento de ciertos tipos de protocolos o transmisiones sobre estos enlaces, visualizando tendencias de comportamiento. Sin embargo, para hacer más efectiva esta medición de las condiciones de la red, otro valor necesario de considerar es el ancho de banda, tanto de envío como de descarga, que se maneja en el dispositivo. En este ámbito, la consideración tomada es el medir el ancho de banda actual siendo utilizado. A pesar de que esta forma no permite calcular toda la capacidad disponible, se evita el agregar nuevo tráfico sobre la red y así no dañar las transmisiones establecidas.

Un acotamiento realizado corresponde al hecho de que todas las transmisiones probadas fueron a través del túnel establecido entre el dispositivo móvil y el HA, es decir, sin el uso de una ruta optimizada con los destinatarios de la conexión. Esto permite simplificar el manejo de las transmisiones, ya que todos los enlaces son establecidos a través de este equipo, por lo que las mediciones como las políticas son pensadas en esta comunicación. En el caso de extender este modelo para soportar transmisiones directas entre el MN y el CN, es necesario almacenar en forma adicional parámetros específicos para cada una de ellas, y establecer las políticas a nivel de dirección IP y puerto de destino, además de información propia de la transmisión, como el tipo de protocolo usado.

Con lo anterior es posible el formar una tabla de asignación entre transmisiones e interfaces, donde es permitido, en el caso de tener más de una interfaz activa, asignar varias transmisiones a ésta. Las prioridades de los tipos de transmisiones son asignadas por el usuario, y son almacenadas en un formato que pueda ser rescatado por el módulo de control de políticas. Además, como la información es medida continuamente, la tabla de reglas es mantenida actualizada para un correcto uso del proceso de selección y aplicación de políticas. Se ha optado por el criterio de enviar el tráfico de más alta prioridad por la mejor interfaz, y

los demás, dividirlos en las otras, permitiendo contar con una transmisión dedicada para aquellos flujos que más se requieren. No obstante, se realiza una medición periódica del ancho de banda siendo usado en las interfaces de forma de no subutilizar los recursos debido a una mala asignación de las transmisiones. En la actual implementación el usuario sólo posee la habilidad de modificar la tabla de tráfico y sus prioridades, sin embargo, no le es posible el actualizar las políticas generadas por el sistema en base a información externa, como el costo económico de un enlace, medida por éste desde la red.

El nodo móvil está encargado tanto de determinar las acciones a aplicar sobre los paquetes (a través de la tabla de políticas generada) como de la ejecución de éstas. A su vez, el Home Agent sólo posee la capacidad de ejecutar las políticas definidas y que son recibidas desde el dispositivo móvil. En cuanto al proceso de registro de las políticas con el HA, se envía tanto la HoA como el BID de la interfaz referenciada, junto a la información propia de la política, con lo cual es posible identificar en forma precisa el destino de la transmisión, permitiendo que el HA maneje registros pertenecientes a distintas entidades móviles. El proceso de control, detección de nuevas interfaces y registro de políticas se muestra resumido en la Figura 5-9. En este proceso, cuando una nueva interfaz es detectada, esto puede generar en la creación de una nueva política o en la actualización de alguna existente, redefiniendo sus prioridades o su criterio de selección.

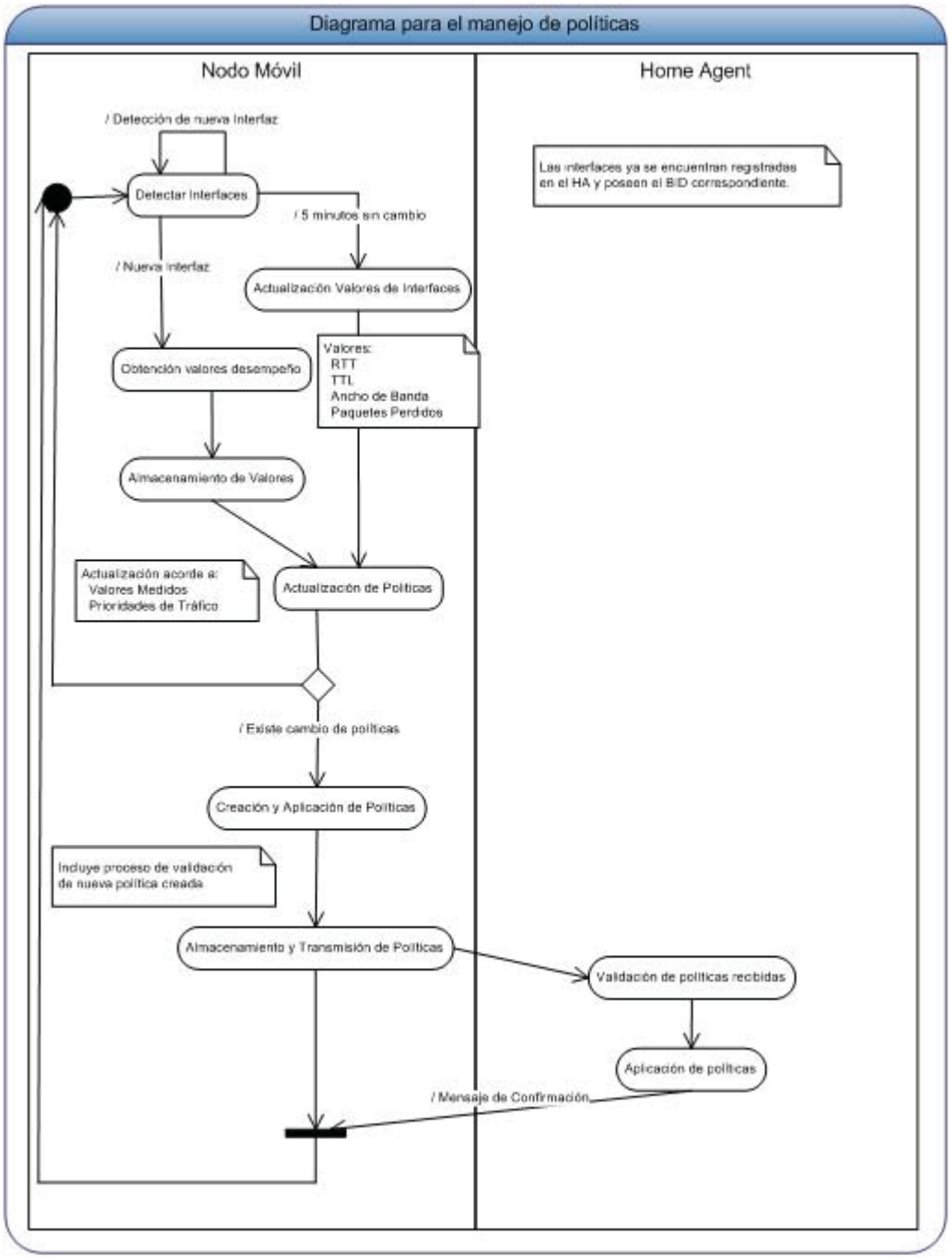


Figura 5-9. Módulo de control de interfaces y políticas.

Debido a las consideraciones tomadas en el diseño, el mayor procesamiento de las políticas se produce en el nodo móvil, dejando al HA sólo la obligación de revisarlas y aplicarlas cuando estas son recibidas, de manera de producir la optimización sobre el tráfico entrante como saliente.

Para producir la asignación de tráfico a las distintas interfaces, se traduce la tabla en la capa de transporte, usando la aplicación Netfilter, y específicamente el comando ip6tables, que permite manejar el control de tráfico, en este caso, gracias a la función de marcado de paquetes con la opción MARK, función que es ampliamente soportada en ambientes Linux, no provocando problemas de compatibilidad en la comunicación entre distintos equipos.

Para el control de las políticas a crear se usa el valor BID asignado a las interfaces en uso, por el protocolo de movilidad, para marcar los paquetes salientes. Esto es, los paquetes marcados con un valor X, serán enviados por aquella interfaz que tenga la interfaz con BID X, constituyendo la acción tomada por la política. Con esto se logra forzar que un cierto tipo de tráfico sea enviado por una interfaz específica. Si esta interfaz no se encuentra disponible, los paquetes serán enviados a través de la interfaz disponible con mayor prioridad (mejor valoración entregada por la función de ordenamiento). En la situación por defecto, es decir, sin políticas definidas y que se cuente con varias interfaces activas, sólo es usada una interfaz manteniendo a las otras en forma de respaldo. Por una limitación de la implementación de movilidad elegida, sólo los paquetes reenviados por el MR (pero no los generados en forma local) pueden usar estas políticas de asignación.

Para ejemplo, si se tiene definida la interfaz eth1 con un BID 100 (Figura 5-10):

```
.
.
Interface "eth1" {
     Bid 100;
     BidPriority 10;
     Reliable true;
}
.
.
```

Figura 5-10. Definición de Interfaz.

que acorde al proceso de calificación de interfaces se ha evaluado como la “mejor” interfaz disponible, y se ha definido el siguiente tráfico, o condición a chequear en la transmisión (Figura 5-11):

```
.
.
tcp 22 1
.
.
```

Figura 5-11. Definición de tráfico.

entendido como protocolo tcp sobre el puerto destino 22 (conocido como SSH) posee una prioridad 1 (mientras menor la prioridad, más crítica es la transmisión). Entonces es posible la definición de la siguiente orden (Figura 5-12),

```
ip6tables -A PREROUTING -t mangle
        -p tcp  --dport 22
        -j MARK --set-mark 100
```

Figura 5-12. Política de Filtrado.

a nivel del MR, y a nivel de HA (Figura 5-13),

```
ip6tables -A PREROUTING -t mangle
        -p tcp  --sport 22
        -j MARK --set-mark 100
```

Figura 5-13. Definición de tráfico.

como regla simétrica. Esto es hecho, ya que en el caso en que no se cuente con la posibilidad de usar el ruteo optimizado, todo el tráfico, hacia y desde el MR, pasa por el HA.

Acorde a la definición de política realizada en la sección 3.5, es posible mapear en la solución a la presencia de una cierta transmisión como el evento, a la presencia de una regla existente asociada a ese tipo de tráfico como la condición, y a la aplicación de la política de ruteo como la acción tomada.

Para la definición de las políticas a ser aplicadas en el mismo dispositivo, como las que son enviadas al HA, se usa el formato presentado en **[38]** debido a su simplicidad para incorporar nuevos parámetros, facilidad de procesamiento y ligereza en cuanto al tamaño final de la política. En forma adicional permite que los mensajes sean portables y que ambas entidades

en la comunicación no sean necesariamente totalmente compatibles en cuanto a las opciones de política que manejen. Se prefirió esta forma de estructurar la política sobre al uso de XML debido al tamaño de los mensajes que este método requeriría. En este sentido, en su forma básica, la política manejada contiene la siguiente información:

HoA del nodo.
Nombre del protocolo y puerto.
BID de la interfaz.

A nivel del sistema operativo, es utilizada la tabla MANGLE ya que en esta es posible modificar los parámetros de los paquetes IPv6, permitiendo el marcado de los paquetes y la posterior priorización de estos. El marcado se produce en la etapa de pre ruteo (cadena PREROUTING) de los paquetes (ya sean entrantes o salientes), sin afectar la cabecera IP de éstos, sino sólo a nivel de control de kernel, para su posterior clasificación y envío.

Una vez que la política es construida y trasmitida a los nodos que la aplicarán, ésta es almacenada en una estructura interna específica para ella, de donde serán recuperadas acorde a las transmisiones que se manejen. Este proceso es realizado mediante una comunicación bajo el esquema de cliente (nodos móviles) y servidor (el home agent de la red), usando una transmisión basada en sockets. En forma adicional, para brindar mayor seguridad a este proceso, es recomendable el usar esta transmisión bajo un mecanismo seguro, pudiendo asegurar la autenticidad como integridad de ambos extremos.

El proceso completo, junto a su dependencia al protocolo NEMO, es esquematizada en la Figura 5-14.

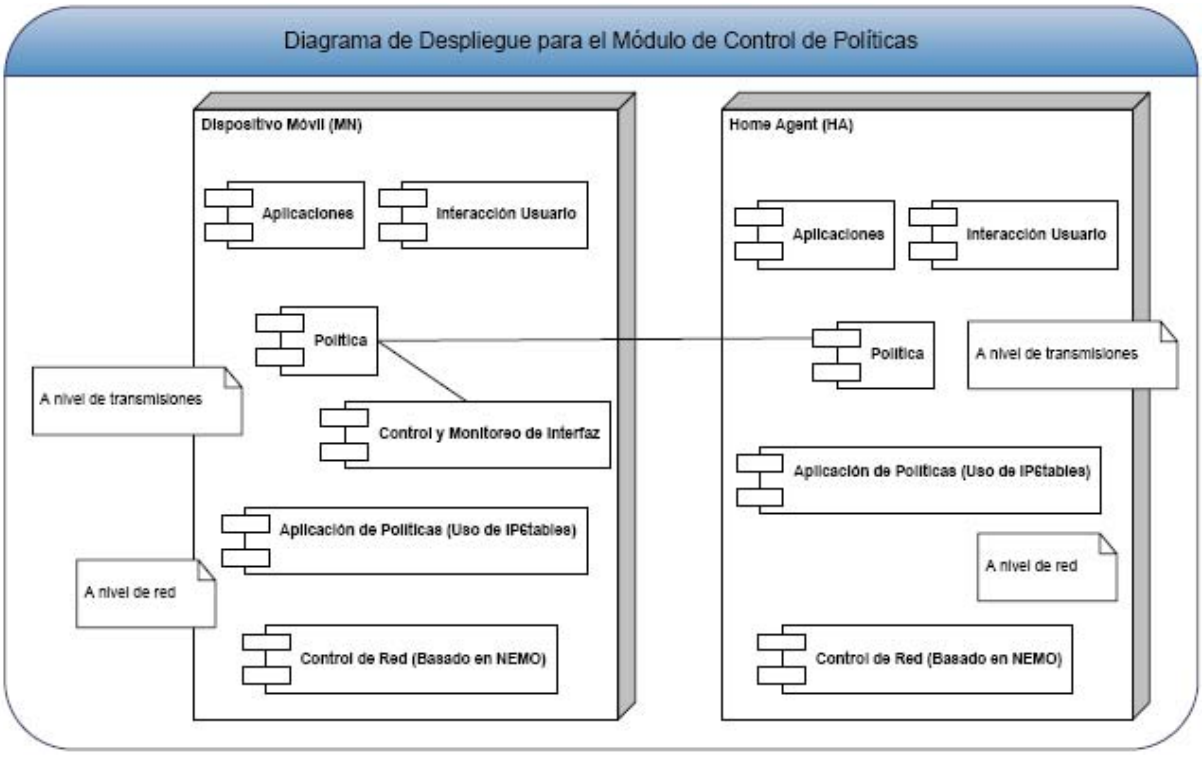


Figura 5-14. Componentes de la solución.

5.2 PRUEBAS

En esta parte del proyecto se busca principalmente validar la metodología de conexión y políticas de transmisión propuestas usando una implementación real y encontrándose acorde a las técnicas y desarrollos actuales. De igual manera, se busca ganar experiencia en esta área, permitiendo obtener conclusiones para ser usadas en nuevos experimentos y creación de nuevos trabajos en el área, como el poder definir variables y/o nuevas métricas para el establecimiento de indicadores de monitoreo y análisis de trasmisiones móviles, particularmente pensando en la etapa de transición desde IPv4 a IPv6, en ambientes móviles.

La etapa de pruebas tiene como base la medición de los siguientes conceptos:

Alcance. Comprobar que los equipos presentes en la prueba puedan ser ubicados en cada momento, mediante comunicaciones normales o encapsuladas.

Desempeño. Medición de las métricas definidas en la transmisión, evaluadas constantemente. Esta se encuentra principalmente referida a:

- Pérdida de Paquetes. Considerados como aquellos enviados y no confirmados o recibidos.

◦ Tiempo de demora. Entendido como el intervalo de tiempo entre el envío y la recepción de los paquetes.

Disponibilidad. Criterio relacionado a los enlaces establecidos por las interfaces, y se refiere a que en cada momento que se realice una prueba sobre estos, los valores de las métricas medidas referentes al desempeño deben encontrarse dentro de un rango considerado aceptable (valor definido dentro del contexto de los tipos de comunicación establecidos). La aceptabilidad se da, en el caso que existan, respecto a los distintos enlaces activos. A su vez, los valores de las métricas son los que permiten, junto a la medición del desempeño, el ordenamiento y selección de los enlaces disponibles.

Las principales herramientas de medición usadas en esta etapa correspondieron a:

IPERF. Herramienta para generación de tráfico TCP/UDP y medición de éste.

TCPDump. Analiza el tráfico de la red, pudiéndose usar para interfaces como protocolos específicos.

MIPv6Tester. Herramienta para medir transmisión y proceso de handover bajo conexiones TCP/UDP en redes IPv6 móviles.

Wireshark. Analizador de conexiones, con facultades de filtrado y de gráfico.

En el escenario de prueba, mientras el MN se encuentra en su red de origen, la autoconfiguración fue suficiente para que este pudiese adquirir una dirección IPv6, estableciese conectividad y comenzase a generar tráfico. En el caso que el MN se encuentre en una red externa, es necesario que el dispositivo tenga activas sus funcionalidades de MIPv6, que la autoconfiguración asigne una dirección local en el MN y que se registre esta CoA con su HA, para que el nodo pudiese enviar tráfico a través de su red de origen.

Principalmente en los casos probados, tanto la red de origen como la red foránea, se encuentran conectadas a través de un mismo Tunnel Broker (BTExact, ubicado en Inglaterra). A pesar que los puntos de acceso se encuentran “cerca” (misma ciudad), se produce una considerable diferencia en los tiempos de RTT en la entrega directa (mediante la red local a la que se encuentra asociado el MN) y en los provenientes de una red externa generados usando el túnel establecido con el HA (Figura 5-15). Para experimentar con esta situación, posteriormente se utiliza un segundo enlace establecido con un servidor de túneles ubicado en Estados Unidos (Freenet6), notándose cambios principalmente en los tiempos de ida y vuelta

de la red (RTT), además del número de saltos presentes entre ambos extremos (TTL), cuando se accede a un equipo fuera de las redes de prueba.

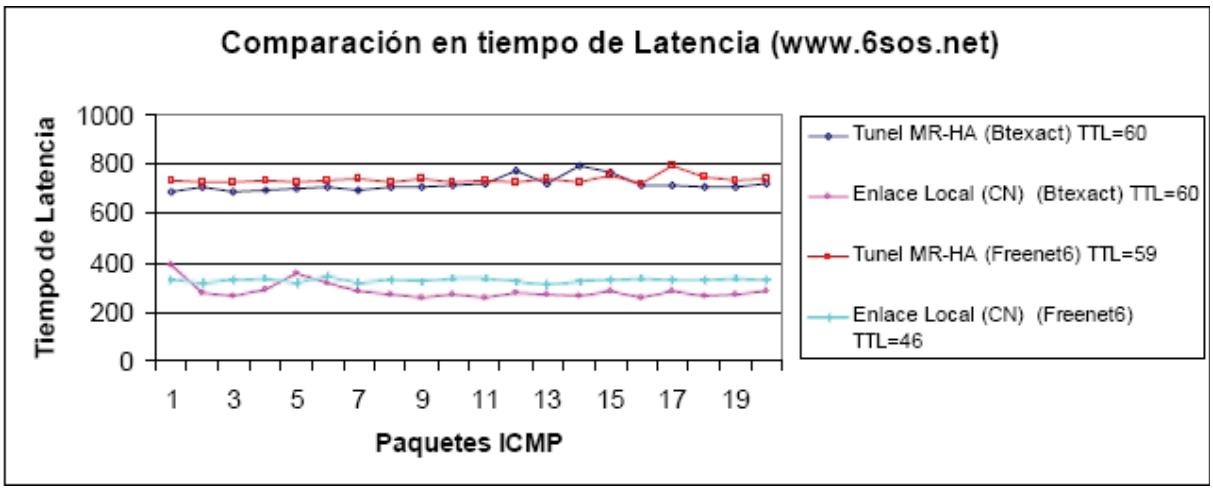


Figura 5-15. Comparación en tiempos de latencia.

El tiempo extra proviene de la encapsulación adicional necesaria por el MN para enviar el tráfico generado, la cual agrega un nuevo encabezado a los paquetes a enlutar y un mayor tiempo de procesamiento, además de la necesidad de realizar un doble viaje al servidor del túnel broker, debido al paso por el HA. Resalta la disminución en el tiempo de transmisión en el caso en que se use la ruta optimizada, es decir MN-CN, lo cual evita el paso por el HA; y también el hecho de que no necesariamente la combinación (servicio de túnel y encapsulación) con menor tiempo de latencia corresponde al menor número de saltos, lo que aumenta la importancia de poder asignar correctamente un enlace a un tipo transmisión específica.

La ejecución del proceso de tracepath6 (Figura 5-16) en el MN permite ilustrar la diferencia de la ruta utilizada,

```
1?: [LOCALHOST]                      pmtu 1432
 1:  no reply
 2: tb-exit.ipv6.btexact.com         705.678ms
 3: 2001:618:400::c84a:6a9d          972.835ms
 4: 2001:618:400:2e29::1000          973.663ms reached
     Resume: pmtu 1432 hops 4 back 4
```

Figura 5-16. Tiempos de latencia y saltos usando servicio de BTExact.

donde, encontrándose ubicado en la red 2001:618:400:2e29::/64 (red foránea), pero perteneciendo a otro segmento de red (red de origen), se visualiza el viaje al HA (punto 3 de la Figura 5-16) para luego dirigirse al equipo objetivo. En la Figura 5-17 se muestra la misma ruta anterior, pero usando otro servicio de túneles, lo que provoca que para llegar al destino, se incremente el número de saltos necesarios.

```
1?: [LOCALHOST]                          pmtu 1280
 1:  2001:5c0:9738::1                       3.354ms
 2:  2001:5c0:8fff:fffe::6e16             214.763ms
 3:  2001:5c0:0:5::114                    213.666ms
 4:  2001:5a0:300::5                      216.761ms
 5:  2001:5a0:300:200::2                  asymm  6 252.461ms
 6:  2001:5a0:300:100::2                  asymm  7 225.231ms
 7:  2001:5a0:400:200::1                  asymm  8 218.836ms
 8:  2001:5a0:600:200::1                  225.916ms
 9:  2001:5a0:600:200::5                  asymm  7 229.508ms
10:  2001:5a0:600::5                      asymm  8 250.306ms
11:  2001:1900:4:3::25                    asymm  9 232.659ms
12:  2001:1900:6:1::1                     asymm 13 368.776ms
13:  2001:1900:5:2::62                    295.581ms
14:  2001:618:1::6                        asymm 15 313.349ms
15:  2001:618:400:69ef::1000              508.387ms reached
     Resume: pmtu 1280 hops 15 back 15
```

Figura 5-17. Tiempos de latencia y saltos usando servicio de Freenet6.

Tomando como ejemplo el largo de un paquete ICMP transmitido (de 64 bytes), es posible observar la diferencia producto de esta encapsulación (Tabla 5-1).

Protocolo	Tamaño total enviado	Encapsulación de Datos
ipv4	98 bytes	eth:ip:icmp:data
ipv6 (uso de CN)	118 bytes	eth:ipv6:icmpv6:data
ipv6 (uso de túnel MN-HA)	158 bytes	eth:ipv6:ipv6:icmpv6:data

Tabla 5-1. Aumento en el tamaño producido debido a la encapsulación.

En las comparaciones anteriores se deja ver que tanto la encapsulación adicional como el tiempo influenciado debido al paso por el HA, principalmente, inciden directamente en las

transmisiones establecidas. Como es de suponer, la ubicación de los servidores de túnel, tanto de la red foránea como de la red origen, influyen sobre la conexión, como se refleja en la Tabla 5-2. En esta prueba, se establece un enlace a nivel de aplicación, entre el dispositivo móvil y el HA, por el cual se transmite una cantidad fija de información (con límite de 3 MB), midiendo el tiempo requerido para recibir estos datos.

Tipo de Conexión	Protocolo	Servicio	Ancho de Banda	Tasa de Descarga
Usando Túnel HA-MN	IPv6	Freenet6	216 kbps	27.09 KB/sec
Usando Túnel HA-MN	IPv6	BTExact	262 kbps	32.83 KB/sec
Enlace Directo	IPv6	BTExact	360 kbps	45.05 KB/sec
Enlace Directo	IPv6	Freenet6	402 kbps	50.31 KB/sec
Conexión Normal	IPv4	Telefónica	517 kbps	64.69 KB/sec

Tabla 5-2. Influencia en Ancho de Banda en la conexión entre redes.

En cambio, dentro de una red interna, con soporte tanto de IPv4 como IPv6, no existe mayor diferencia en las respectivas tasas de transmisión, alcanzando el máximo de la capacidad de envío de las interfaces del dispositivo móvil, en forma independiente del protocolo usado.

En las Figura 5-18 es posible visualizar el cambio producido por la elección de la forma de conexión utilizada, al descargar un archivo de tamaño de 1.9 MB, desde un equipo ubicado en un servidor externo a ambas redes de prueba, logrando un promedio de 54 KB/seg para la conexión mediante túnel con el HA, y de 61 KB/seg en el caso de enlace directo. El uso de una encapsulación extra en la comunicación produce un aumento en el número de paquetes perdidos (en el experimento corresponden casi al doble comparado con una conexión directa) y en el número de acknowledgements duplicados enviados al origen en la transmisión.

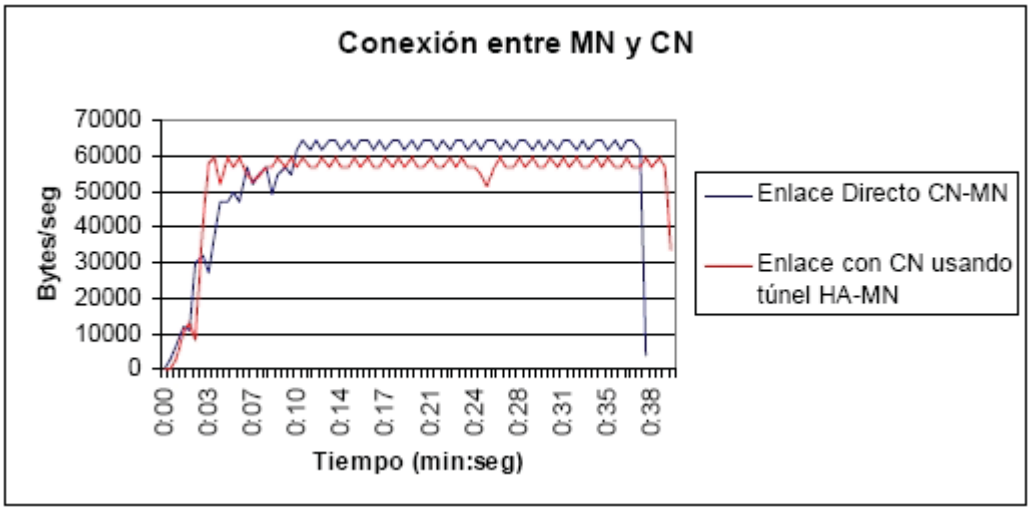


Figura 5-18. Influencia de tipo de conexión bajo MIPv6.

Uno de los principales aspectos definidos en este proyecto es el uso de varias interfaces de conexión en forma simultánea. Para el caso específico de las pruebas, se usaron dos interfaces, como se muestra en la Figura 5-19, lo que permitió la aplicación de políticas de distribución sobre las transmisiones.

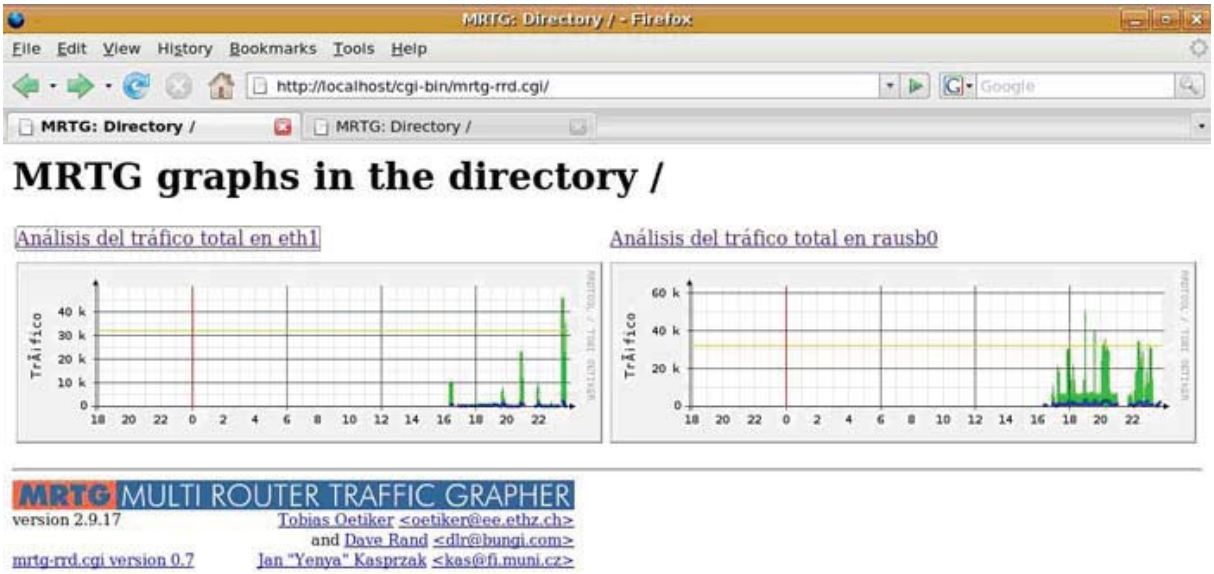
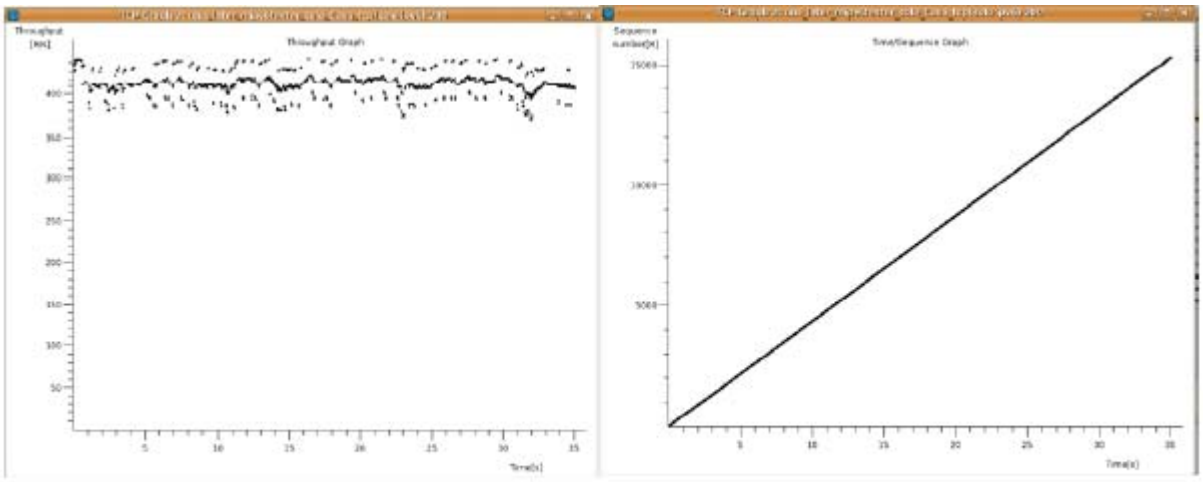


Figura 5-19. Uso simultáneo de interfaces en el nodo móvil.

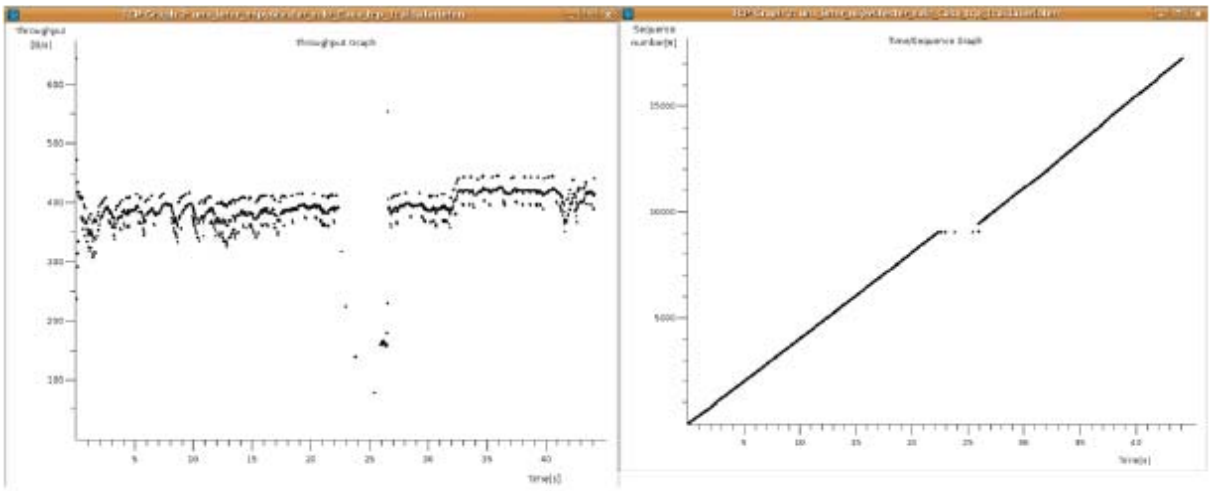
En los casos en que una sola interfaz se encuentra disponible (Figura 5-20 a) y b)), y se produce una caída (Figura 5-21 a) y b)), el tiempo de restablecimiento del enlace es

aproximadamente de 3 segundos bajo NEMO (producto del tiempo requerido para obtener una nueva dirección y realizar el proceso de registro con el HA), lo que en comunicaciones de tiempo real afecta considerablemente la calidad de la recepción.



(a) (b)

Figura 5-20 (a) y (b) en una transmisión normal (protocolo TCP; tasa de 400 B/s).



(a) (b)

Figura 5-21 (a) y (b) en una transmisión normal con una caída (protocolo TCP; tasa de 400 B/s)

En el caso de disponer de varios enlaces funcionando en el dispositivo móvil, estos actúan de respaldo, reduciendo los tiempos caída de la señal. Este proceso se muestra en la Figura 5-22.

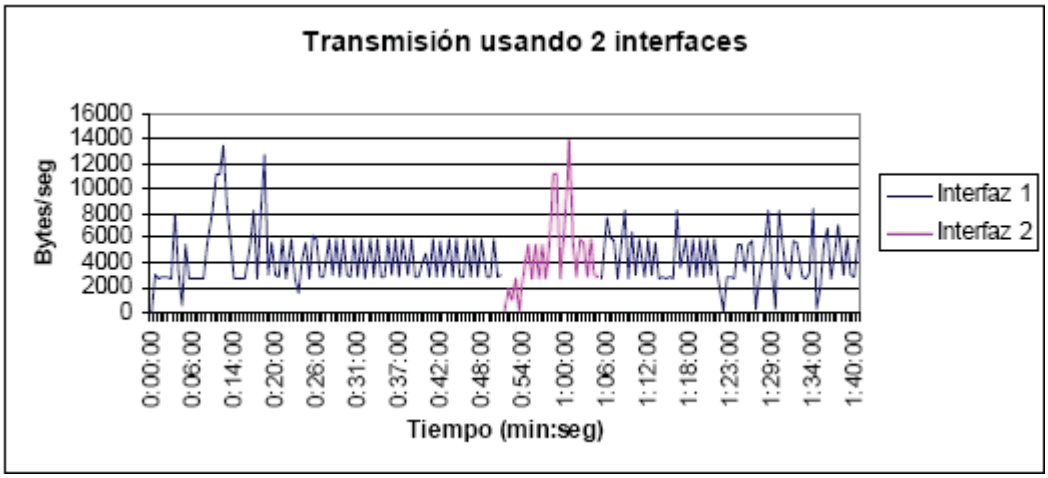
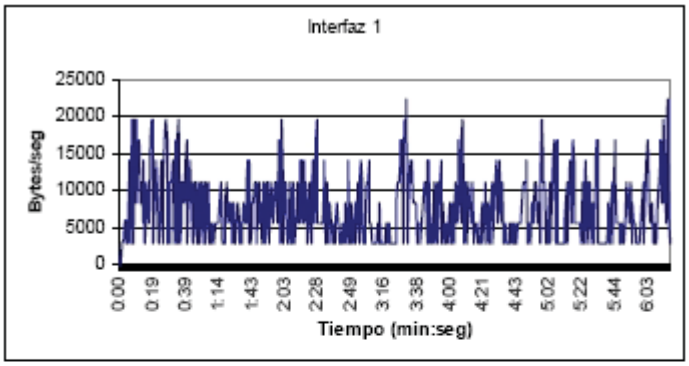


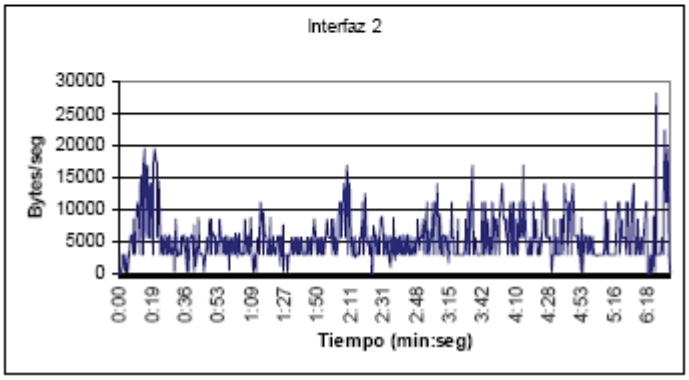
Figura 5-22. Uso de dos interfaces durante la transmisión.

Las constantes caídas o handover en las transmisiones TCP producen la aparición del comienzo lento en las transmisiones, haciendo que la tasa de envío permanezca en niveles bajos, afectando el desempeño del protocolo.

Sin embargo, el uso de las restantes interfaces sólo en forma de respaldo no es óptimo, en especial si se dispone de un ancho de banda de conexión superior a lo soportado por cada interfaz en forma individual. A lo anterior puede ser sumado el hecho de diferencias en las interfaces como el soporte de distintas tecnologías de conexión o medidas de seguridad existentes en ellas. Esto hace requerido el poder distribuir el tráfico, según prioridades del tráfico transmitido, por los distintos enlaces, mediante el registro de políticas. Lo anterior permite que el tráfico especificado vaya por la mejor interfaz (posee mejor desempeño), asignando el tráfico no prioritario por las otras interfaces disponibles. El caso en que el dispositivo posea dos interfaces activas, se muestra en la Figura 5-23:



(a) Tráfico Video



(b) Tráfico Web y audio

Figura 5.23 (a) y (b). Distribución de tráfico entre las interfaces activas.

lo que asociado a un continuo monitoreo del estado de estas interfaces, permite que la transmisión no se vea afectada producto de distorsiones en la comunicación, siendo especialmente útil cuando es requerido un alto nivel de calidad de servicio.

Actualmente, no es posible sobre la plataforma elegida realizar un control más avanzado sobre el tráfico IPv6, como el manejo de colas o control de ancho de banda. Sin embargo, dado las condiciones cambiantes de los entornos móviles, el manejo de transmisiones propuesto es acorde a estos cambios y permite un manejo rápido y eficiente de los paquetes, no cayendo en el cambio continuo de diversos parámetros a nivel del dispositivo móvil.

Otro aspecto introducido en el proyecto es el de la construcción de aplicaciones adaptables a los distintos escenarios en ambientes móviles. Uno de los más importantes, en el ámbito de comunicaciones en tiempo real, es el relativo a las transmisiones de video, donde una buena decisión de la forma de comprensión, ya sea en audio y video, permite una buena utilización de los recursos disponibles a la vez de mantener la calidad de la transmisión. En la Figura 5-

24 se realiza un a comparación de tres esquemas de transmisión, usando un mismo video con tasa de video de 192 bits y de tasa de audio de 64 bits. El resultado puede visualizarse en la Figura 5-25 a) y b), comparando la transmisión del video original, y sobre UDP, la cual demostró reducir el uso del ancho de banda a la vez de mantener una tasa de transmisión, sin afectar la calidad final percibida en el nodo extremo de la comunicación. En cuanto a las transmisiones sobre TCP, la que presentó una menor cantidad de segmentos perdidos y mejor tasa de transmisión fue el video codificado en el formato WMV2.

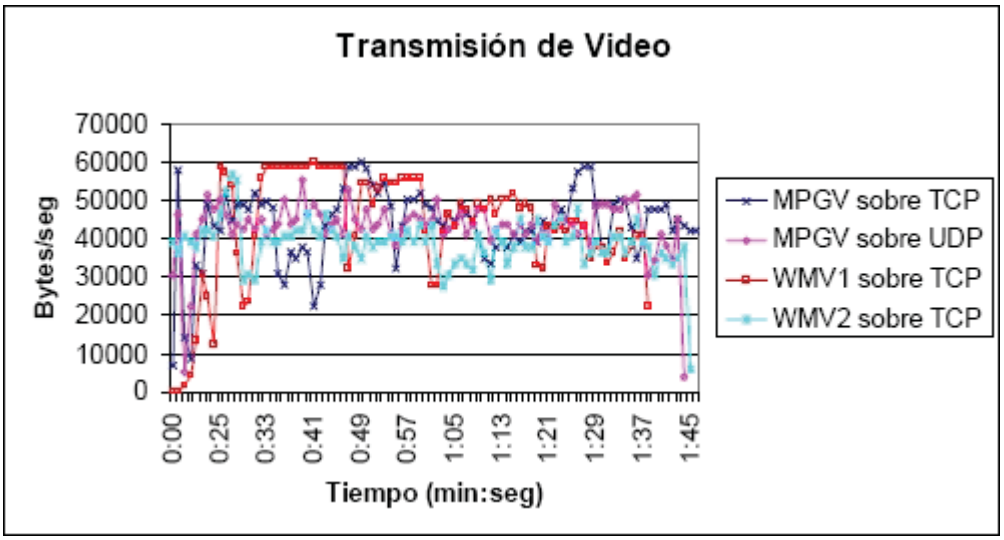


Figura 5-24. Comparación de esquemas de transmisión de video.

(a) (b)

Figura 5-25. Comparación de esquemas de transmisión de video.

A pesar que en ambientes móviles la pérdida de paquetes es más propensa a ocurrir, es necesario notar que se produce un aumento de ésta cuando la comunicación se lleva a cabo sobre una transmisión IPv6 establecida a través del HA (debido, por ejemplo, a la encapsulación adicional presente). Esta situación se encuentra visualizada en los resultados de distintos tipos de transmisiones mostrados en la Figura 5-26.

Como se ha mencionado en el proyecto, la situación anterior también produce que se tenga que estar consiente de los distintos entornos por los que atraviesa la comunicación, en especial cuando se usan servicios de túneles (como los tunnel broker) debido a que se está propenso al incremento de la fragmentación de los paquetes, introduciendo una mayor demora en el proceso de transmisión normal, afectando principalmente las transmisiones en tiempo real como radio o video.

Radio sobre TCP

Característica	Total de Paq.	Segm. Perdidos (%)	Ack duplicados (%)	KB/seg prom.	Tiempo (seg)	Bytes env.	Observaciones
Normal	1644	0,55	0,12	19	57	1083	Servicio VirginRadio 128 kbps sobre IPv6
Túnel	1710	6,73	0,76	19	60	1140	

Video sobre TCP

Característica	Total de Paq.	Segm. Perdidos (%)	Ack duplicados (%)	KB/seg prom.	Tiempo (seg)	Bytes env.	Observaciones
Normal	2697	3,52	0,52	61	38	2318	Video: 1.9 MB, 1min 33 seg, 105 bps, 176x144
Túnel	2652	5,54	0,34	53	43	2279	Produce fragmentacion de los paqueles

Video

Característica	Total de Paq.	Segm. Perdidos (%)	Ack duplicados (%)	KB/seg prom.	Tiempo (seg)	Bytes env.	Observaciones
Original (mpgv2)	4194	6,18	0,36	41	97	3977	Caract.: 352x288, 3min 2seg, 25 fps, 3.8 MB
wmv1 (TCP)	2954	2,98	0,54	32	88	2816	video 192 kbits, audio 64 kbits, 25 fps
mpgv (TCP)	4499	4,38	0,76	38	110	4180	video 192 kbits, audio 64 kbits, 25 fps
wmv2 (TCP)	4131	2,95	0,27	40	99	3960	video 192 kbits, audio 64 kbits, 25 fps
mpgv (UDP)	3804	0,00	0,00	46	112	5152	video 192 kbits, audio 64 kbits, 25 fps

Nota: - Segmentos perdidos (origen--> destino)
- Ack duplicados (destino-->origen)

Figura 5-26. Desempeño de transmisiones sobre una red IPv6.

Particularmente, en las transmisiones sobre TCP, es importante que el equipo emisor posea la habilidad de adecuar el valor del tamaño de la ventana, o la cantidad de información enviada por el dispositivo móvil (mediante la utilización del proceso de descubrimiento de MTU incluida en el protocolo IP o la implementación de una función similar) según las condiciones cambiantes de los distintos puntos de acceso con los cuales se establecen enlaces, para hacer un uso correcto de estos (como se grafica en la Figura 527 y Figura 5-28), y es un requerimiento del cual el dispositivo tendrá que estar consciente, de forma de poder regular la información enviada y compartir los recursos de transmisión que se posean.

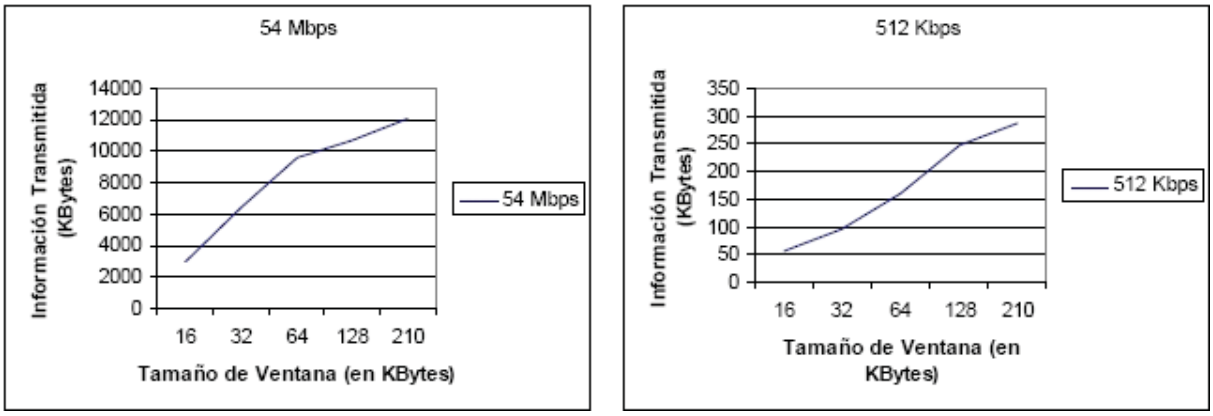


Figura 5-27. Variación de la tasa de transmisión en distintos enlaces.

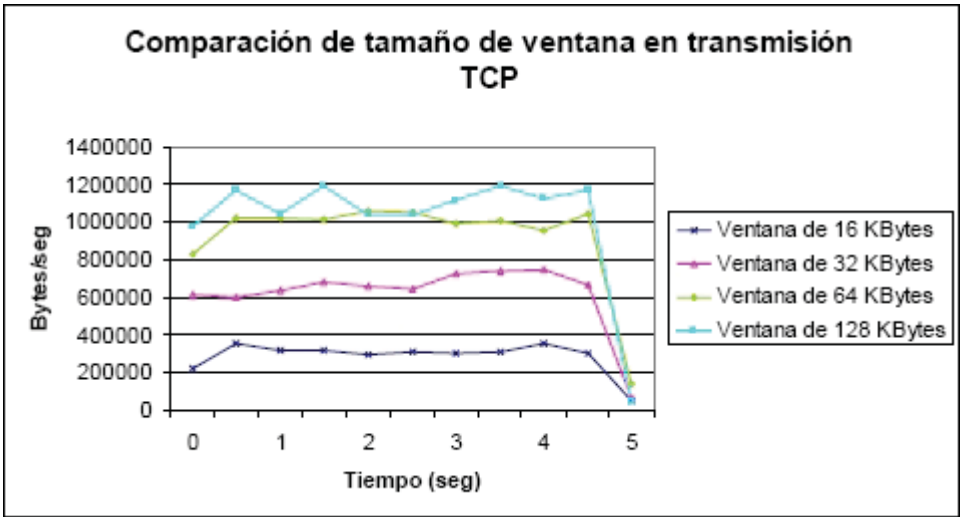


Figura 5-28. Comparación de tamaño de ventana bajo conexión de 54 Mbps.

CAPÍTULO 6
RESULTADOS A OBJETIVOS PROPUESTOS.

6.1 COMPARACIÓN CON OBJETIVOS

Dos fueron los principales objetivos abarcados durante este proyecto. El primero correspondió al estudio de IPv6 bajo el ámbito de su introducción al ambiente productivo en una organización, haciendo un estudio de distintas consideraciones necesarias, como el desarrollo de aplicaciones, medidas de seguridad y de transición, para una correcta migración desde IPv4 a IPv6. En forma paralela, el otro objetivo fue estudiar el uso de IPv6 en un ambiente de movilidad, haciendo uso de nuevas facultades, en especial, crear un ambiente multihoming de comunicación mediante el uso de varias interfaces de conexión.

En relación a los resultados obtenidos para los objetivos específicos propuestos al inicio del presente trabajo, se tiene:

- Análisis y diseño de alternativas de comunicación en función de los requerimientos planteados.

  De las distintas implementaciones disponibles, se eligió el uso de NEPL con soporte de registro de múltiples CoA, principalmente debido a la documentación disponible y su constante desarrollo. Esta implementación soporta la comunicación entre un nodo móvil y su red origen, acorde a MIPv6. Sin embargo, como no soporta el manejo del tráfico en forma eficiente por todos los enlaces establecidos, fue diseñado un marco de control de políticas de tráfico, el cual funciona en paralelo a esta implementación, y que incluye mecanismos de selección de interfaces y de asignación de transmisiones a éstas acorde a las políticas creadas. Con esto se logra mejorar el rendimiento adecuándose en mejor forma a las condiciones cambiantes de los entornos móviles. En este punto, debido al uso requerido de IPv6, se abarca en forma amplia el proceso de transición, y coexistencia, desde IPv4 a IPv6, centrándose preferentemente en ambientes de producción, y los requerimientos tanto del hardware como software necesario para este propósito.

- Estudio de los potenciales beneficios y problemas de una arquitectura multiacceso IPv6 en situaciones reales.
  La implementación de distintos enlaces de comunicación desde el dispositivo móvil, permitió la obtención de beneficios como la división y priorización de flujos, el control sobre el ancho de banda disponible y sobre la calidad existente en los distintos enlaces. Otro resultado es el balanceo y transferencia de carga, entre las interfaces activas en el dispositivo, permitiendo dar la apariencia al usuario de estar siempre conectado, debido a la reducción del tiempo sin transmisión por fallas en la conexión o producto del proceso de handoff en los enlaces, lo que se logra mediante la redundancia producto del uso de varias interfaces a la vez, permitiendo que, si una interfaz falla o pierde conectividad, el tráfico sea traspasado y dividido entre las otras.

- Propuesta y diseño de un marco de routing basado en políticas de comunicación que soporte la arquitectura propuesta.
  Uno de los aspectos no incluidos en la implementación del protocolo de movilidad elegido es el contar con un proceso de selección y de ordenamiento en las comunicaciones. Para esto se desarrolló y se probó una arquitectura de control de políticas de filtrado que permitiese, a través de reglas y acciones, un efectivo y eficiente comportamiento de la solución, basada en los distintos enlaces manejados e identificados por el BID asignado. Esta estructura permite el control entre ambos extremos de la comunicación, transmitiéndose las políticas necesarias, en un formato liviano (sin consumir muchos recursos de la red), lográndose mejoras en el tráfico entrante como saliente. En su diseño también se consideró el hecho de su extensión en cuanto a opciones manejadas y a módulos de control.

- Implementación y prueba de las alternativas planteadas evaluándolas en cuanto a parámetros propios de las transmisiones.
  En la etapa de pruebas se evaluaron distintos parámetros de conexión como funcionalidades y servicios bajo IPv6, bajo un escenario compuesto de equipos fijos,

routers y equipos móviles, todos con capacidad de dual stack. Debido a que el tamaño del escenario era reducido, se usaron rutas fijas, en lugar de algún protocolo de ruteo con mayores funcionalidades. Se buscó el formar un marco de introducción de la movilidad bajo IPv6, y sus nuevas funciones, estableciendo qué criterios considerar en el control de las transmisiones. Es especial, se tuvieron como base (y usadas en el control de la arquitectura de políticas) el tiempo de latencia, la tasa de errores, el ancho de banda disponible y el número de saltos presentes en la conexión. Como escenario de pruebas durante el proyecto, se tuvo a los vehículos de emergencia, por lo que ésta tuvieron centradas en la transmisión de información entre el dispositivo móvil (MN) y la entidad de origen (HA), como archivos de datos, audio, y transmisiones de video (tanto bajo TCP como UDP). Con esto se evaluó el desempeño y se establecieron criterios y consideraciones para un mejor control de este tipo de comunicaciones, como lo es, por ejemplo, la adaptación de las transmisiones acorde a las características actuales de los enlaces de comunicación. Otro punto en el que se hizo énfasis en la creación del escenario de pruebas es que se tratase de un ambiente entre redes, es decir, que no se encontrasen todas las partes involucradas en la comunicación en el mismo lugar, conectándose a IPv6 por medios disponibles actualmente, permitiendo el estudiar cómo estos afectan en la transmisión. Como factor adicional, se ha construido un sitio Web, con información relativa al proyecto además de algunas utilidades, como un test de velocidad que funciona bajo IPv4 como IPv6, usado como forma de difusión del trabajo realizado.

A lo largo del proyecto se establecieron distintos requerimientos que eran necesarios de considerar, tanto en la etapa de diseño como de construcción, ya sea en el escenario de pruebas o en la arquitectura de políticas. Entre los principales se encuentran:

- Preservar la transparencia en las comunicaciones, permitiendo la comunicación entre un nodo que implemente la arquitectura a desarrollar con un nodo que no la implemente. También se debe permitir, en el caso que no implemente dicha arquitectura, funcione normalmente dentro de una estructura de multihoming, sin afectar su comportamiento. Esto también se tomó en cuenta en la construcción de la arquitectura de políticas, ya que en el caso en que no se encuentre implementada, o no tengan la misma versión en uso, sólo se ejecutan las funciones y opciones que se tengan en común.

- No introducir vulnerabilidades adicionales a la red resultante tras implementar la arquitectura a desarrollar. Se debe tratar que la comunicación permanezca en funcionamiento como lo es hasta antes de la implementación. Esta área tiene relación con el mecanismo de transición que se utilice para funcionar con IPv6 y con las características tanto de hardware como software que posean los equipos que forman parte de la comunicación. En el caso del escenario de pruebas, se usó el mecanismo de dual stack para soportar tanto IPv4 como IPv6, y el uso de túneles IPv6 sobre IPv4 (encapsulación de los paquetes) para la conexión con IPv6, por ser una de las formas de transición más seguras y transparentes en cuanto a la coexistencia con redes bajo IPv4, aunque no completamente en lo relacionado a la aplicación de medidas de seguridad sobre el tráfico que viaja en estos túneles. Por esto, también es analizado cómo la seguridad es abordada sobre las transmisiones en IPv6 a distintos niveles, considerando principalmente los temas de confidencialidad e integridad sobre éstas. Además, el uso de una encapsulación adicional introduce una fragmentación adicional sobre el tráfico. El uso de la política de filtrado diseñada no supone requerimientos especiales a soportar por parte de los dispositivos. También, para evitar el desarrollo de un trabajo sobre requerimientos muy específicos o sin mucho uso hoy en día, el trabajo se basó sobre documentos RFC y DRAFT actuales y con amplio soporte en la comunidad.

- No romper las aplicaciones. La arquitectura y posterior implementación de la arquitectura de multihoming como de políticas es transparente al nivel de las aplicaciones, y la aplicación de políticas se realiza a nivel de flujos, no afectando su transmisión ni la experiencia para el usuario. En forma adicional, son estudiados distintas características que las nuevas aplicaciones que funcionen bajo entornos de conectividad tendrán que estar consientes, como la tasa de transmisión recursos del enlace, de manera de adaptarse a los entornos cambiantes por los que atraviesan.

- Establecimiento de una metodología de pruebas. Los criterios de disponibilidad y desempeño, en especial considerando el quiebre de las conexiones y la pérdida de paquetes en la comunicación, estuvieron como meta tanto en la construcción del escenario de pruebas, como en la elección de los servicios sobre los cuales aplicarlas.

6.2 TRABAJO FUTURO

En forma general, el proyecto ha cumplido con los objetivos establecidos en relación a los puntos a estudiar e implementaciones propuestas. No obstante, el área estudiada permite que se continúe trabajando en ella, y uno de los objetivos es el fomentar nuevas áreas de interés relativas a la introducción de IPv6. Principalmente se identifican tres grandes sectores para extender el área de alcance del proyecto.

El primero, corresponde a profundizar el estudio sobre el traspaso o migración completa de una organización desde IPv4 a IPv6. Actualmente son pocos los casos existentes, y la tecnología todavía se encuentra en perfeccionamiento, a nivel de especificación como de implementación (tanto lógica como física). La complejidad de las redes que se encuentran actualmente en operación, principalmente en grandes compañías, hará este proceso más complicado, teniendo que considerar muchas variables, especialmente en el porte de los servicios soportados (como por ejemplo, ERP, Correo, Web, entre otros) sobre la red IPv4.

Otra área corresponde en el mejoramiento del proceso y estudio de nuevas formas para la distribución de políticas. En este proyecto se propone el uso de una estructura cliente-servidor en la cual se intercambian mensajes estructurados correspondientes a acciones a tomar sobre cierto protocolo. Nuevos desarrollos hacen énfasis en incluir este registro como una sub-opción dentro de los mensajes de BU manejados entre el MN y el HA, y en la aplicación de medidas de seguridad adicionales sobre este proceso.

Por último, la construcción de un escenario más real que incluya los esquemas desarrollados en este proyecto se convierte en un objetivo más ambicioso, pero más práctico, pudiendo adaptarlo a la realidad de una organización específica.

CAPÍTULO 7
CONCLUSIONES.

El uso de tecnologías inalámbricas ha evolucionado en gran manera, y de igual forma lo han hecho aquellas aplicaciones de la vida real que se benefician de este avance, las que a su vez imponen nuevos requerimientos y características a las tecnologías en desarrollo. El concepto de “estar siempre conectado” está cada vez más cerca a su implementación, lo que llevará a que dejar nuestra oficina o empresa sea totalmente transparente al usuario. La combinación mostrada en el trabajo, telemedicina (como servicio de emergencia) y redes multihoming inalámbricas, es un claro ejemplo de ello, razón por la que se tuvo como motivadora para el estudio de los beneficios a lograr con la introducción de nuevas tecnologías móviles.

IPv6 surge como el paso siguiente indicado a dar para acceder a nuevos beneficios y al desarrollo de servicios en un gran abanico de temas. A pesar de que el desarrollo de este protocolo y de MIPv6 se ha dado principalmente en ciertas áreas del globo, este tiende a expandirse de forma de conectar al mundo bajo esta nueva red. Este tema se convirtió en un punto a favor en el desarrollo del presente proyecto dado el poco trabajo, en comparación con el globo, desarrollado en el área de IPv6 en Latinoamérica, además de ser enfocado bajo un ámbito global de estudio aplicado al cambio de tecnología y a las distintas consideraciones que tendrán que abordar las organizaciones en cuanto a la incorporación de IPv6 y sus nuevos beneficios, especialmente abarcando las ventajas en torno a la movilidad.

Bastó darse cuenta que hoy en día los dispositivos están siendo adecuados para poder asociarse a distintos puntos de acceso para estimar que era necesario el diseño de una forma de permitir a los dispositivos móviles el usar todas, y en forma eficiente, aquellas conexiones que fuesen posibles. En este sentido, se cumplió con los objetivos propuestos en el presente trabajo en cuanto a la implementación de un escenario de prueba basado en movilidad sobre IPv6, que pudiese hacer uso, acorde a políticas dinámicamente definidas, de las distintas conexiones establecidas.

En cuanto a resultados, este proyecto ha permitido el hacer converger distintos temas relativos a transmisiones móviles y a IPv6, entregando análisis sobre métricas y puntos de consideración en el proceso de transición, con incidencia sobre el desempeño de tipos específicos de comunicaciones, como es el caso de aplicaciones conscientes sobre las condiciones de los enlaces y que pueden modificar su tasa de transmisión entregando al usuario una experiencia de continuidad. También se ha desarrollado un marco extensible para mejorar el proceso de transmisiones bajo las ventajas del nuevo protocolo, de manera de incorporar variaciones para el desempeño de nuevas aplicaciones o uso de la red. Junto a esto, el hecho de desarrollar una metodología de prueba con la particularidad de incorporar distintas redes y servicios bajo IPv6, permite obtener resultados sobre un mayor número de situaciones, como nuevos escenarios de uso o aplicaciones que se deseen evaluar.

A su vez, la construcción de un escenario real bajo IPv6 permitió el análisis de nuevas situaciones y recomendaciones que extendieron el alcance del proyecto, abarcando nuevas áreas, lo que hizo que tuviese un alcance más práctico y real, con posibilidades de aplicación dentro de los escenarios de ejemplo propuestos a lo largo del presente proyecto. Lo anterior permite que el proyecto quede abierto, en el sentido que presenta formas de extender y profundizar ciertas áreas abordadas en las cuales es necesario realizar un mayor estudio. En forma adicional, el hecho de presentar varias ramas de trabajo futuro que extienden el alcance del presente trabajo, indica que tanto la movilidad como la introducción de IPv6 son áreas que tienen bastante que ser desarrollado y estudiado de forma de brindar y aprovechar al máximo las capacidades que estos temas ofrecen.

REFERENCIAS.

[1] Johnson, D., Perkins, C., Arkko, J.: *RFC 3775: Mobility Support in IPv6.* Junio, 2004.

[2] Deering, S.,Hinden, R.: *RFC 2460: Internet Protocol, Version 6 (IPv6) Specification.* Diciembre, 2008.

[3] Postel, J.: *RFC 791: Internet Protocol. Darpa Internet Program. Protocol Specification.* Septiembre,1981.

[4] Rosenberg, J., Schulzrinne, H. Camarillo, G. et al.:*RFC 3261: SIP: Session Initiation Protocol*. Junio, 2002.

[5] Nordmark , E., Bagnulo , M.: *Shim6: Level 3 Multihoming Shim Protocol for IPv6*. draft-ietf-shim6-proto. Mayo, 2007.

[6] Moskowitz, R., Nikander, P.: *RFC 4423: Host Identity Protocol (HIP) Architecture*. Mayo, 2006.

[7] Thaler, D.: *A Comparison of IP Mobility-Related Protocols*. draft-thaler-mobilitycomparison. Octubre, 2006.

[8] Dutta, A. et al.: *Comparative Analysis of Network Layer and Application Layer IP mobility protocols for IPv6 Networks.* Wireless Personal Multimedia Communications (WPMC) 2006, San Diego, CA, USA. Septiembre, 2006.

[9] Devarapalli, V., Wakikawa, R., Petrescu, A. et al.: *RFC 3963: Network Mobility (NEMO) Basic Support Protocol.* Enero, 2005.

[10] Montavont, N., Wakikawa, R., Ernst, T. et al.: *Analysis of Multihoming in Mobile IPv6.* draft-ietf-monami6-mipv6-analysis. Febrero, 2007.

[11] Proyecto 6net. *Deliverable D4.5.3 Evaluation of Multihoming Solutions*. Enero, 2005.

[12] Perkins, C.: *RFC 3344: IP Mobility Support for IPv4*. Agosto, 2002.

[13] Soliman, H., Castelluccia, C., El Malki, K. et al.: *RFC 4140: Hierarchical Mobile IPv6 Mobility Management (HMIPv6).* Agosto, 2005.

[14] McCann, P.: *RFC 4260: Mobile IPv6 Fast Handovers for 802.11 Networks.* Noviembre, 2005.

[15] Devarapalli, V., Wakikawa, R., Petrescu, A. et al.: *RFC 3963: Network Mobility (NEMO) Basic Support Protocol.* Enero, 2005.

[16] Wakikawa, R., Ernst, T., Nagami, K. et al.: *Multiple Care-of Addresses Registration.* draft-ietf-monami6-multiplecoa. Marzo, 2007.

[17] Proyecto InterCar. http://www.sfc.wide.ad.jp/InternetCAR. Realizado en Keio University SFC, Japón. Última visita: Septiembre, 2007.

[18] Proyecto Nautilus. http://www.nautilus6.org. Última visita: Septiembre, 2007.

[19] Boutet, A., Kuntz, R., Montavont, J. et al.: *E-Bike – Demonstration of the IPv6 Network Mobility*. Francia. 2007.

[20] Proyecto IST-ANEMONE. http://www.ist-anemone.eu. Comunidad Europea. Última visita: Septiembre, 2007.

[21] Droms, R., Bound, J., Volz, B. et al.: *RFC 3315: Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*. Julio, 2003.

[22] Harz, C.: *IPv6 CE Demonstration Area: The City of the Future Project*. 2007.

[23] Baker, F.: *IPv6 Transition Thoughts*. Cisco Systems. 2005.

[24] Baker, F.: *Global Trends in Supporting IPv6 Services and Applications*. Cisco Systems. 2007.

[25] Cooperative Association for Internet Data Analisis (CAIDA). http://www.caida.org. Última visita: Septiembre, 2007.

[26] Lee., H.: *IPv6 Deployment Strategies and Status in Taiwan*. Taiwan Network Inforrmation Center. Taiwan. 2006.

[27] IPv6 Ready Logo Program. http://www.ipv6ready.org. Última visita: Septiembre, 2007.

[28] North American IPv6 Task Force. http://www.nav6tf.org. Última visita: Septiembre, 2007.

[29] The IPv6 Forum. http://www.ipv6forum.org. Última visita: Septiembre, 2007.

[30] Thomson, S., Narten, T.: *RFC 2462: IPv6. Stateless Address Autoconfiguration*. Diciembre, 1998.

[31] RIPE (Réseaux IP Européens). *IPv6 Address Allocation and Assignment Policy. APNIC, ARIN, RIPE NCC*. Julio, 2007.

[32] Bokor, L.: *Performance Evaluation of IPv6 Mobility Implementations*. 16th IST Mobile & Wireless Communications Summit. IST-ANEMONE Workshop on IPv6 mobility networking testbeds. Budapest, Hungary. Julio, 2007.

[33] Tsukada, M., Ernst, T., Wakikawa, R. eth al.: *Dynamic Management of Multiple Mobile Routers*. IEEE International Malaysia Conference on Communications and IEEE International Conference in Networks (MICC-ICON), Kuala Lumpur, Malaysia. Noviembre, 2005.

[34] Montavont, N., Wakikawa, R., Ernst, et al.: *Analysis of Multihoming in Mobile IPv6.* draft-ietf-monami6-mipv6-analysis. Febrero, 2007.

[35] Ng, C., Paik, E., Ernst, T. et al.: *Analysis of Multihoming in Network Mobility Support*. draft-ietf-nemo-multihoming-issues. Febrero, 2007.

[36] Kuladinithi, K., Nikouras, F., Könsgen, A. et al.: *Enhanced Terminal Mobility through the use of Filters for Mobile IP*. In Proceedings of the Summit on Mobile and Wireless Communications (IST Summit), Aveiro, Portugal. Junio, 2003.

[37] Soliman, H., Montavont, N., Fikouras, N. et al.: *Flow Bindings in Mobile IPv6*. draftsoliman-monami6-flow-binding. Octubre, 2006.

[38] Mitsuya, K.., Tasaka, K., Wakikawa R. et al.: *A Policy Data Set for Flow Distribution.* draft-mitsuya-monami6-flow-distribution-policy. Febrero, 2007.

[39] Draves, R.: *RFC 3484 - Default Address Selection for Internet Protocol version 6 (IPv6).* Febreero, 2003.

[40] Yavatkar, R., Pendarakis, D., Guerin, R.: *RFC 2753 - A Framework for Policy-based Admission Control*. Enero, 2000.

[41] Westerinen, A., Schnizlein, J., Strossner, J. et al.: *RFC 3198 - Terminology for Policy-Based Management*. Noviembre, 2001.

[42] Ylitalo, J., Jokikyyny, T., Kauppinen, T.et al.: *Dynamic Network Interface Selection in Multihomed Mobile Hosts.* Proceedings of the 36th Hawaii International Conference on System Sciences. Hawaii, Estados Unidos. 2003.

[43] Chan-Wah Ng, Ernst, T.: *Multiple Access Interfaces for Mobile Nodes and Networks.* IEEE International Conference on Networks ICON, Singapore. Noviembre, 2004.

[44] Gilligan , R., Nordmark, E.: *RFC 2893: Transition Mechanisms for IPv6 Hosts and Routers*. Agosto, 2000.

[45] Soliman, H.: *Mobile IPv6 support for dual stack .Hosts and Routers (DSMIPv6)*. draft-ietf-mip6-nemo-v4traversal. Marzo, 2007.

[46] Conta, A., Deering, S.: *RFC 2473: Generic Packet Tunneling in IPv6 Specification*. Diciembre, 1998.

[47] Bound, J.: *RFC 4057: IPv6 Enterprise Network Scenarios*. Junio, 2005.

[48] O'Hanlon, P.: *IPv6 Security*. Central American IPv6 Training Workshop. Guatemala. Febrero, 2007.

[49] Faigl, Z.: *Security threats in systems supporting IPv6 mobility and state-of-the art security solutions*. Proyecto ANEMONE. Junio, 2007.

[50] Proyecto 6BONE. *Testbed for deployment IPv6.* http://www.6bone.net. Última visita: Septiembre, 2007.

[51] Carpenter, B., Moore, K.: *RFC 3056: Connection of IPv6 Domains via IPv4 Clouds.* Febrero, 2001.

[52] Huitema, C.: *RFC 3068: An Anycast Prefix for 6to4 Relay Routers*. Junio, 2001.

[53] Gállego, J., Hernández-Solana, A., Canalesm M. et al.: *Performance Analysis of Multiplexed Medical Data Transmission for Mobile Emergency Care Over the UMTS Channel.* IEEE Transactions on Information Technology in Biomedicine. vol. 9, no. 1, pp. 13-22. Marzo, 2005.

[54] *Mobile ER (Emergency Room) an IPv6-based Video System for Emergency Care Support.* Nara Institute of Science and Technology. http://www.ipv6style.jp/en/special/20051031_2/index.shtml. Última visita: Septiembre, 2007.

[55] Kim, J., Kim, D., Hong, C. et al.: *A Network Mobility Management Architecture for IPv4 and IPv6 Environments.* The 1st International Workshop on Broadband Convergence Networks, Canada. Abril, 2006.

[56] Shima, K., Kuntz, R., Mitsuya, K.: *Nautilus6 mobile technology demonstrations at the First IPv6 Summit in Thailand*. Julio, 2006.

[57] Yong Liu, A., Narasimha, R.: *Multihoming Route Control among a Group of Multihomed Stub Networks*. Febrero, 2006.

[58] Wakikawa, R., Uehara, K., Teraoka, F. et al.: *MibSocket. An integrated mechanism to manipulate general network information in mobile communications*. IEICE Transactions on Communication. Vol. E84-B, No 8, pp. 2001-2010. Japon, 2001.

[59] Suciu, L., Bonnin, J., Ernst, T. et al.: *Multiple Network Interfaces Management for Mobile Routers*. 3rd 5th International Conference on ITS Telecommunications (ITST), Brest, Francia. Junio, 2005.

[60] Snir, Y., Ramberg Y., Strossner, J. et al.: *RFC 3644: Policy Quality of Service (QoS) Information Model*. Noviembre, 2003.

[61] Waldbusser, S., Saperia, J., Hongal, T.: *RFC 4011: Policy Based Management MIB.* Marzo, 2005.

[62] *Multiple Care-of Addresses Registration for NEPL.* Proyecto Nautilus. http://software.nautilus6.org/MCoA. Última visita: Septiembre, 2007.

[63] Cool IPv6 Stuff. SixXS. http://www.sixxs.net/misc/coolstuff/. Última visita: Septiembre, 2007.

[64] Video on Demand. UK6x. http://www.uk6x.com/applicationservices/videos.html. Última visita: Septiembre, 2007.

[65] Virgin Radio over IPv6 Unicast. http://www.ipv6.ecs.soton.ac.uk/virginradio. Última visita: Septiembre, 2007.

[66] IPv6 Howto. OpenWRT. http://wiki.openwrt.org/IPv6_howto. Última visita: Septiembre, 2007.

[67] The Multi Router Traffic Grapher (MRTG). http://www.mrtg.org. Última visita: Septiembre, 2007.

[68] Network Traffic Probe (NTOP). http://www.ntop.org. Última visita: Septiembre, 2007.

APÉNDICES.

APÉNDICES.

APÉNDICE A. ACRÓNIMOS

BU	:	Binding Update
CN	:	Correspondent Node
CoA	:	Care of Address
GPRS	:	General Packet Radio Service
HA	:	Home Agent
HoA	:	Home Address
IPSEC	:	Internet Protocol Security
IPTV	:	Internet Protocol Television
MCoA	:	Multiple Care of Addresses
MIPL	:	Mobile IP for Linux
MIPv6	:	Mobile IP version 6
MN	:	Mobile Node
MR	:	Mobile Router
NEMO	:	Network Mobility
QoS	:	Quality of Service
RA	:	Router Advertisement
RADVD	:	Router Advertisement Daemon
RFC	:	Request For Comment
SIP	:	Session Initiation Protocol
VOIP	:	Voice Over Internet Protocol

APÉNDICE B. ARCHIVOS DE CONFIGURACIÓN

ROUTER

Configuración de Túnel entre Router y BTExact

```
ip tunnel add bt mode sit remote 213.121.24.85 local 200.74.106.132 ttl 255 \
&& ip link set bt up \
&& ip -6 addr add 2001:618:400::c84a:6a84/128 dev bt \
&& ip route add ::/0 dev bt \
&& ip -f inet6 addr \
&& echo 1 > /proc/sys/net/ipv6/conf/all/forwarding \
&& echo 0 > /proc/sys/net/ipv6/conf/all/accept_ra \
&& /etc/init.d/S51radvd start \
&& echo "Configuración IPv6 completa" || \
{ echo "Configuración IPv6 errónea!" 1>&2; exit 1; }
```

```
root@OpenWrt:/etc/init.d# ifconfig br0
br0       Link encap:Ethernet  HWaddr 00:14:BF:31:1A:CF
          inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: 2001:618:400:2e29::1/64 Scope:Global
          inet6 addr: fe80::214:bfff:fe31:1acf/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1074 errors:0 dropped:0 overruns:0 frame:0
          TX packets:756 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:79236 (77.3 KiB)  TX bytes:132779 (129.6 KiB)

root@OpenWrt:/etc/init.d# ifconfig bt
bt        Link encap:UNSPEC  HWaddr C8-4A-6A-ED-00-00-00-00-00-00-00-00-00-00-00-00
          inet6 addr: 2001:618:400::c84a:6aed/128 Scope:Global
          inet6 addr: fe80::c84a:6aed/128 Scope:Link
          UP POINTOPOINT RUNNING NOARP  MTU:1472  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:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Figura B-1. Direccionamiento a túnel broker (Interfaz bt).

HA

Configuración HA

```
# Archivo de Configuración Mobile IPv6: Home Agent
# filename: /etc/mip6d.conf

NodeConfig HA;

## Nivel de Información mostrada
DebugLevel 10;

## Lista de Interfaces donde está activo el HA
```

```
Interface "eth0";

HaAcceptMobRtr enabled;
HaAcceptMCoAReg enabled;

##
## IPsec configuración
## No activo.
#UseMnHaIPsec enabled;
UseMnHaIPsec disabled;
KeyMngMobCapability disabled;

# MNP configuración
HaServedPrefix 2001:618:400:69ef::/64;
BindingAclPolicy 2001:618:400:69ef::1000 (2001:618:400:69ef::/64) MCoAReg allow;
DefaultBindingAclPolicy allow;
```

Configuración RADVD en HA

```
# cat /etc/radvd.conf
interface eth0
{
        AdvSendAdvert on;
        MaxRtrAdvInterval 3;
        MinRtrAdvInterval 1;
        AdvIntervalOpt off;
        AdvHomeAgentFlag on;
        HomeAgentLifetime 10000;
        HomeAgentPreference 20;
        AdvHomeAgentInfo on;
        prefix 2001:618:400:69ef::1000/64
        {
                AdvRouterAddr on;
                AdvOnLink on;
                AdvAutonomous on;
                AdvPreferredLifetime 10000;
                AdvValidLifetime 12000;
        };
};
```

Script de inicio de NEMO en HA

```
#!/bin/bash
IF1=eth0

RADVD=/usr/src/radvd-1.0/radvd
NEMOD=/usr/local/sbin/nemod

echo 0 > /proc/sys/net/ipv6/conf/all/accept_ra
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding

ifconfig $IF1 inet6 add 2001:618:400:69ef::1000/64
#ip -6 route add default via 2001:618:400:69ef::1/64 dev eth0
```

$RADVD -C /etc/radvd.conf
$NEMOD -c /etc/nemod.conf

#EOF

```
eth0      Link encap:Ethernet  HWaddr 00:01:29:FD:08:BF
          inet addr:158.251.10.108  Bcast:158.251.10.255  Mask:255.255.255.0
          inet6 addr: 2001:618:400:69ef::1000/64 Scope:Global
          inet6 addr: fe80::201:29ff:fefd:8bf/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:14820 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3224 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1934753 (1.8 MiB)  TX bytes:401144 (391.7 KiB)
          Interrupt:185 Base address:0xc000

ip6tnl1   Link encap:UNSPEC  HWaddr 20-01-06-18-04-00-69-EF-00-00-00-00-00-00-00-00
          inet6 addr: fe80::201:29ff:fefd:8bf/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP  MTU:1440  Metric:1
          RX packets:47 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:5822 (5.6 KiB)  TX bytes:1440 (1.4 KiB)
```

Figura B-2. Direccionamiento de túnel virtual entre HA y MN (Interfaz ip6tnl1).

```
root@nleiva-ucv:~# ip -6 route show
2001:618:400:69ef::1001 from 2001:618:400:69ef::1000 dev eth0  proto 15  metric 128  expires 21334258sec mtu 1500
 advmss 1440 hoplimit 4294967295
2001:618:400:69ef::1001 dev ip6tnl1  proto 15  metric 192  expires 21334258sec mtu 1440 advmss 1380 hoplimit 4294
967295
2001:618:400:69ef:1::/80 dev ip6tnl1  proto 15  metric 192  expires 21334258sec mtu 1440 advmss 1380 hoplimit 429
4967295
unreachable default dev lo  proto unspec  metric -1  error -101 hoplimit 255
unreachable default dev lo  proto unspec  metric -1  error -101 hoplimit 255
2001:618:400:69ef::/64 dev eth0  metric 256  expires 21328617sec mtu 1500 advmss 1440 hoplimit 4294967295
fe80::/64 dev eth0  metric 256  expires 21328119sec mtu 1500 advmss 1440 hoplimit 4294967295
fe80::/64 via :: dev bt  metric 256  expires 21328625sec mtu 1480 advmss 1420 hoplimit 4294967295
fe80::/64 dev ip6tnl1  metric 256  expires 21334258sec mtu 1440 advmss 1380 hoplimit 4294967295
ff00::/8 dev eth0  metric 256  expires 21328119sec mtu 1500 advmss 1440 hoplimit 4294967295
ff00::/8 dev bt  metric 256  expires 21328625sec mtu 1480 advmss 1420 hoplimit 4294967295
ff00::/8 dev ip6tnl1  metric 256  expires 21334258sec mtu 1440 advmss 1380 hoplimit 4294967295
default dev bt  metric 1024  expires 21328625sec mtu 1480 advmss 1420 hoplimit 4294967295
unreachable default dev lo  proto unspec  metric -1  error -101 hoplimit 255
```

Figura B-3. Salida del comando IP route, estableciendo la ruta hacia el MN mediante el túnel creado.

Configuración de MRTG en HA

```
##############################################################################
# Multi Router Traffic Grapher
##############################################################################

# Configuración Global
WorkDir: /var/www/mrtg
WriteExpires: Yes
Interval: 1
EnableIPv6: Yes
```

```
LogFormat: rrdtool
PathAdd: /usr/bin
IconDir: /images

Target[eth0]: `/usr/bin/mrtg-ip-acct eth0`
MaxBytes[eth0]: 1250000
Title[eth0]: Análisis de Tráfico total para Interfaz: eth0
PageTop[eth0]: <H1>Análisis de Tráfico para eth0 (private network)

 </H1>
 <TABLE>
   <TR><TD>System:</TD><TD>Router IPv6</TD></TR>
   <TR><TD>Interface:</TD><TD>eth0</TD></TR>
   <TR><TD>IP:</TD><TD>2001:618:400:69ef::1000</TD></TR>
   <TR><TD>Velocidad Máxima de Descarga:</TD>
      <TD>1250.0 kBytes/s</TD></TR>
  </TABLE>

Target[web]: `/home/nikolaos/portstat.sh`
Title[web]: Análisis de Tráfico TCP bajo IPv6 para Interfaz: eth0
PageTop[web]: <h1>Tráfico TCP</h1>
MaxBytes[web]: 1250000
#YLegend[web]: Bytes/s
#ShortLegend[web]: B/s

Target[radio]: `/home/nikolaos/radiostat.sh`
Title[radio]: Análisis de Tráfico UDP bajo IPv6 para Interfaz: eth0
PageTop[radio]: <h1>Tráfico UDP</h1>
MaxBytes[radio]: 1250000
#YLegend[radio]: Bytes/s
#ShortLegend[radio]: B/s

Target[localhost-cpu]: `/home/nikolaos/cpu.sh`
#Options[localhost-cpu]: noinfo, nopercent, growright, nobanner, noi
Title[localhost-cpu]: Consumo de CPU
PageTop[localhost-cpu]: <h1>Consumo de CPU</h1>
MaxBytes[localhost-cpu]: 9999999999
YLegend[localhost-cpu]: CPU %
ShortLegend[localhost-cpu]: %
Legend1[localhost-cpu]: CPU
Legend2[localhost-cpu]: CPU % Activo
LegendI[localhost-cpu]: Active

Title[mem]: Consumo de Memoria
PageTop[mem]: <H1>Consumo de Memoria</H1>
Target[mem]: `/home/nikolaos/mrtg-data mem`
#Options[mem]: growright, transparent, absolute, gauge, noinfo, nopercent
#MaxBytes[mem]: 167772160
MaxBytes[mem]: 461553664
AbsMax[mem]: 534742144
YLegend[mem]: RAM, Swap
ShortLegend[mem]: --
Legend1[mem]: Memoria RAM
Legend2[mem]: Memoria Swap
```

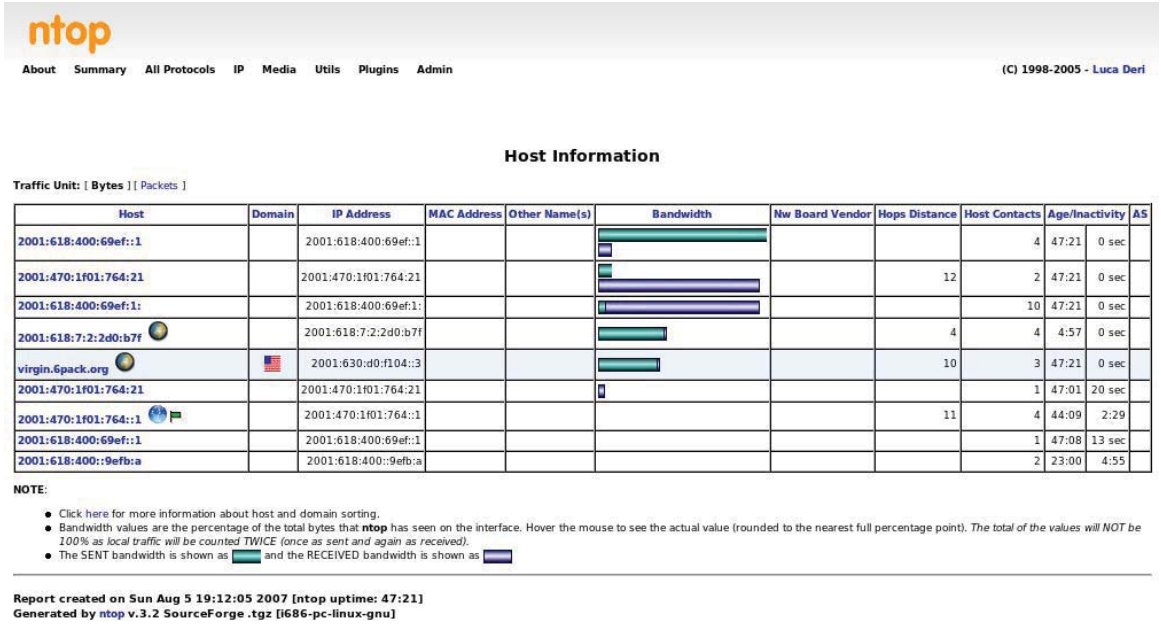
LegendI[mem]: Memoria RAM
LegendO[mem]: Memoria Swap

MR

Configuración del Mobile Router

```
# Mobile IPv6 configuration file: Home Agent
# filename: /etc/mip6d.conf

NodeConfig MN;

# Nivel de Información Mostrada
DebugLevel 10;

DoRouteOptimizationCN disabled;
DoRouteOptimizationMN disabled;
SendMobPfxSols enabled;
UseCnBuAck disabled;
MobRtrUseExplicitMode enabled;
OptimisticHandoff enabled;

# Tiempo de vida de los enlaces
MnMaxHaBindingLife 60;

# Interfaz de Egreso
# Conectado a la red externa

Interface "eth1" {
    #MnIfPreference 2;
    Bid 200;
    BidPriority 20;
    Reliable true;
}

Interface "rausb0" {
    #MnIfPreference 2;
    Bid 100;
    BidPriority 10;
    Reliable true;
}

# Interfaz de egreso a registrar con el HA
MnHomeLink "eth1" {
    IsMobRtr enabled;
    HomeAgentAddress 2001:618:400:2e29::1000;
    HomeAddress 2001:618:400:2e29::1001/64 (2001:618:400:2e29::/64);
    RegMultipleCoA enabled;
    IfMultipleCoA "eth1","rausb0";
}
## IPsec configuración
##
UseMnHaIPsec disabled;
KeyMngMobCapability disabled;
```

APÉNDICE C. CAPTURAS NTOP



ntop

About Summary All Protocols IP Media Utils Plugins Admin (C) 1998-2005 - Luca Deri

Host Information

Traffic Unit: [Bytes] [Packets]

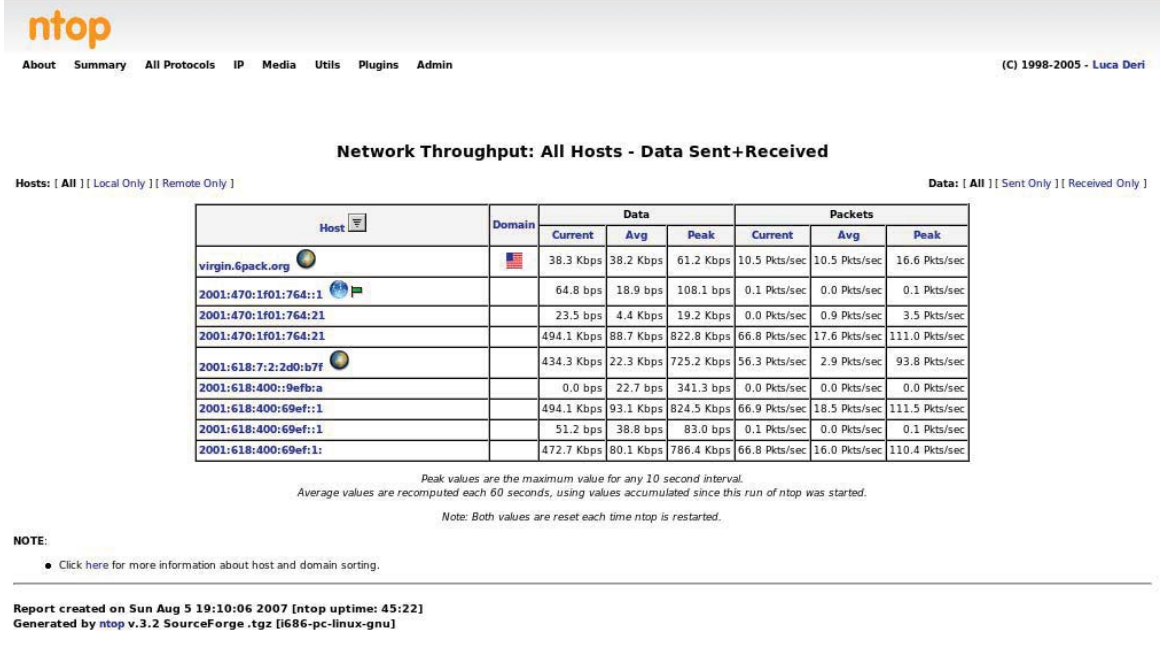
Host	Domain	IP Address	MAC Address	Other Name(s)	Bandwidth	Nw Board Vendor	Hops Distance	Host Contacts	Age/Inactivity		AS
2001:618:400:69ef::1		2001:618:400:69ef::1						4	47:21	0 sec	
2001:470:1f01:764:21		2001:470:1f01:764:21					12	2	47:21	0 sec	
2001:618:400:69ef:1:		2001:618:400:69ef:1:						10	47:21	0 sec	
2001:618:7:2:2d0:b7f		2001:618:7:2:2d0:b7f					4	4	4:57	0 sec	
virgin.6pack.org		2001:630:d0:f104::3					10	3	47:21	0 sec	
2001:470:1f01:764:21		2001:470:1f01:764:21						1	47:01	20 sec	
2001:470:1f01:764::1		2001:470:1f01:764::1					11	4	44:09	2:29	
2001:618:400:69ef::1		2001:618:400:69ef::1						1	47:08	13 sec	
2001:618:400::9efb:a		2001:618:400::9efb:a						2	23:00	4:55	

NOTE:

- Click here for more information about host and domain sorting.
- Bandwidth values are the percentage of the total bytes that **ntop** has seen on the interface. Hover the mouse to see the actual value (rounded to the nearest full percentage point). *The total of the values will NOT be 100% as local traffic will be counted TWICE (once as sent and again as received).*
- The SENT bandwidth is shown as and the RECEIVED bandwidth is shown as

Report created on Sun Aug 5 19:12:05 2007 [ntop uptime: 47:21]
Generated by ntop v.3.2 SourceForge .tgz [i686-pc-linux-gnu]

Figura C-1. Captura con los distintos enlaces establecidos por el HA.



ntop

About Summary All Protocols IP Media Utils Plugins Admin (C) 1998-2005 - Luca Deri

Network Throughput: All Hosts - Data Sent+Received

Hosts: [All] [Local Only] [Remote Only] Data: [All] [Sent Only] [Received Only]

Host	Domain	Data			Packets		
		Current	Avg	Peak	Current	Avg	Peak
virgin.6pack.org		38.3 Kbps	38.2 Kbps	61.2 Kbps	10.5 Pkts/sec	10.5 Pkts/sec	16.6 Pkts/sec
2001:470:1f01:764::1		64.8 bps	18.9 bps	108.1 bps	0.1 Pkts/sec	0.0 Pkts/sec	0.1 Pkts/sec
2001:470:1f01:764:21		23.5 bps	4.4 Kbps	19.2 Kbps	0.0 Pkts/sec	0.9 Pkts/sec	3.5 Pkts/sec
2001:470:1f01:764:21		494.1 Kbps	88.7 Kbps	822.8 Kbps	66.8 Pkts/sec	17.6 Pkts/sec	111.0 Pkts/sec
2001:618:7:2:2d0:b7f		434.3 Kbps	22.3 Kbps	725.2 Kbps	56.3 Pkts/sec	2.9 Pkts/sec	93.8 Pkts/sec
2001:618:400::9efb:a		0.0 bps	22.7 bps	341.3 bps	0.0 Pkts/sec	0.0 Pkts/sec	0.0 Pkts/sec
2001:618:400:69ef::1		494.1 Kbps	93.1 Kbps	824.5 Kbps	66.9 Pkts/sec	18.5 Pkts/sec	111.5 Pkts/sec
2001:618:400:69ef::1		51.2 bps	38.8 bps	83.0 bps	0.1 Pkts/sec	0.0 Pkts/sec	0.1 Pkts/sec
2001:618:400:69ef:1:		472.7 Kbps	80.1 Kbps	786.4 Kbps	66.8 Pkts/sec	16.0 Pkts/sec	110.4 Pkts/sec

Peak values are the maximum value for any 10 second interval.
Average values are recomputed each 60 seconds, using values accumulated since this run of ntop was started.

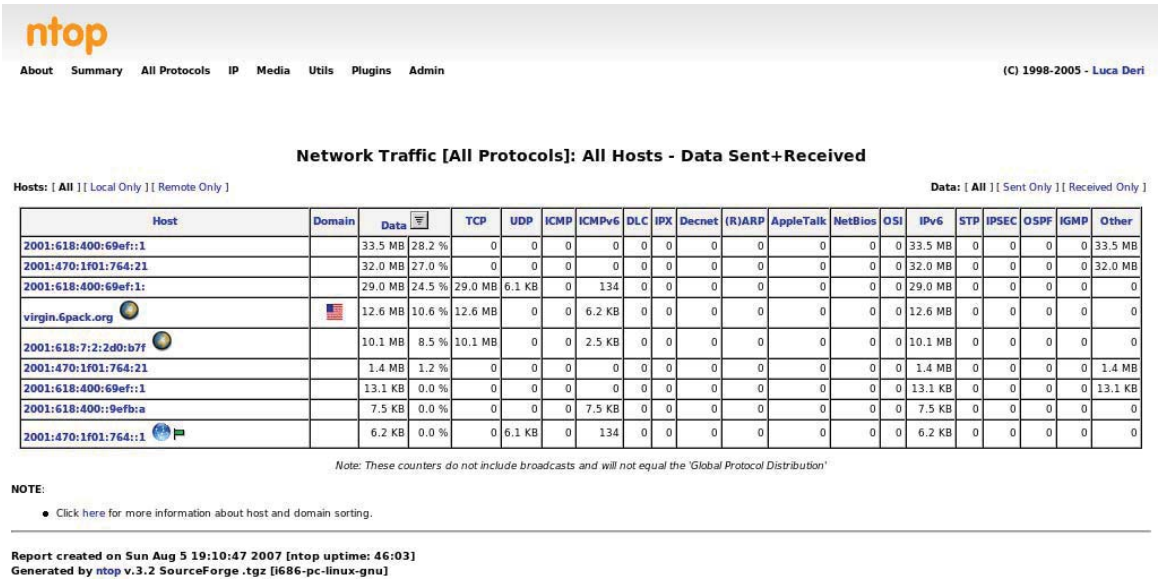
Note: Both values are reset each time ntop is restarted.

NOTE:

- Click here for more information about host and domain sorting.

Report created on Sun Aug 5 19:10:06 2007 [ntop uptime: 45:22]
Generated by ntop v.3.2 SourceForge .tgz [i686-pc-linux-gnu]

Figura C-2. Captura con el throughput alcanzado con cada servicio accesado.



ntop

About Summary All Protocols IP Media Utils Plugins Admin (C) 1998-2005 - Luca Deri

Network Traffic [All Protocols]: All Hosts - Data Sent+Received

Hosts: [All] [Local Only] [Remote Only] Data: [All] [Sent Only] [Received Only]

Host	Domain	Data		TCP	UDP	ICMP	ICMPv6	DLC	IPX	Decnet	(R)ARP	AppleTalk	NetBios	OSI	IPv6	STP	IPSEC	OSPF	IGMP	Other
2001:618:400:69ef::1		33.5 MB	28.2 %	0	0	0	0	0	0	0	0	0	0	0	33.5 MB	0	0	0	0	33.5 MB
2001:470:1f01:764:21		32.0 MB	27.0 %	0	0	0	0	0	0	0	0	0	0	0	32.0 MB	0	0	0	0	32.0 MB
2001:618:400:69ef:1:		29.0 MB	24.5 %	29.0 MB	6.1 KB	0	134	0	0	0	0	0	0	0	29.0 MB	0	0	0	0	0
virgin.6pack.org		12.6 MB	10.6 %	12.6 MB	0	0	6.2 KB	0	0	0	0	0	0	0	12.6 MB	0	0	0	0	0
2001:618:7:2:2d0:b7f		10.1 MB	8.5 %	10.1 MB	0	0	2.5 KB	0	0	0	0	0	0	0	10.1 MB	0	0	0	0	0
2001:470:1f01:764:21		1.4 MB	1.2 %	0	0	0	0	0	0	0	0	0	0	0	1.4 MB	0	0	0	0	1.4 MB
2001:618:400:69ef::1		13.1 KB	0.0 %	0	0	0	0	0	0	0	0	0	0	0	13.1 KB	0	0	0	0	13.1 KB
2001:618:400::9efb:a		7.5 KB	0.0 %	0	0	0	7.5 KB	0	0	0	0	0	0	0	7.5 KB	0	0	0	0	0
2001:470:1f01:764::1		6.2 KB	0.0 %	0	6.1 KB	0	134	0	0	0	0	0	0	0	6.2 KB	0	0	0	0	0

Note: These counters do not include broadcasts and will not equal the 'Global Protocol Distribution'

NOTE:

- Click here for more information about host and domain sorting.

Report created on Sun Aug 5 19:10:47 2007 [ntop uptime: 46:03]
Generated by ntop v.3.2 SourceForge .tgz [i686-pc-linux-gnu]

Figura C-3. Captura con el tráfico total de la de red.

APÉNDICE D. CÓDIGO FUENTE

READ_INF_MN.PL

```
#!/usr/bin/perl

#################################################################
#read_inf_mn.pl                                                 #
#Version: 1.0                                                   #
#Descripción: Obtiene información desde el archivo              #
#de configuración de Mobile IPv6 (Interfaces, BID asignados,    #
#HoA y HA)                                                      #
#                                                               #
#################################################################

sub local_add{
	#Recupera la dirección IPv6 local asignada a una interfaz,
	#posibilitando la comparación de una interfaz virtual con una
	#interfaz real

	$inter = $_[0];
	@array = qx{/sbin/ifconfig $inter};
	$largo = @array;
	$conta_a = 0;
	while ($conta_a < $largo){
		if($array[$conta_a] =~ m/(\s|\t)+fe80::(.*)(\s|\t)+Scope:Link/){
			$dir = "fe80::".$2;
		}
		$conta_a++;
	}
	return $dir;
}

open (CHECKBOOK, "/etc/mip6d.conf") || die "El archivo no existe.";

while ($record = <CHECKBOOK>) {
	if($record =~ m/(^(\s|\t)*Interface(\s|\t)*)\"(.*)\"/){
		#La información obtenida, se almacena en el archivo ha.txt
		#para facilitar su acceso

		open (MYFILE, '>ha.txt');

		print MYFILE "Interface ".$4." ";
		$interf = $4;
		$bid = 0;
		$dir = local_add($interf);
		print MYFILE "lla ".$dir." ";
		while (($bid == 0) && ($record = <CHECKBOOK>)){
			if($record =~ m/(^(\s|\t)*Bid(\s|\t)*)(.*);/){
				print MYFILE "bid ".$4."\n";
				$bid = 1;
			}
		}
		close (MYFILE);
	}
	if($record =~ m/(^(\s|\t)*HomeAgentAddress(\s|\t)*)(.*);/){
		open (MYFILE, '>>ha.txt');
		print MYFILE "HaA ".$4."\n";
		close (MYFILE);
```

```
	}
	if($record =~ m/(^(\s|\t)*HomeAddress(\s|\t)+)(.+)(\s|\t)+.+/){
		open (MYFILE, '>>ha.txt');
		print MYFILE "HoA ".$4;
		close (MYFILE);
	}
}

close(CHECKBOOK);

system("perl read_dev.pl");

exit 0;
```

READ_DEV.PL

```
#!/usr/bin/perl

#################################################################
#read_dev.pl                                                    #
#Version: 2.0                                                   #
#Descripción: Obtiene información desde las interfaces          #
#permitiendo su ordenamiento en cuanto a la calidad de estas#
#Se basa en RTT, % de paquetes perdidos y TTL presente en la#
#ruta                                                           #
#                                                              #
#################################################################

sub ping {
	#Ejecuta el comando ping sobre las interfaces

	$host = "2001:618:400:69ef::1000"; #Es el HA de la red.
	$interval = 0.001;
	$inter = $_[0];
	$pingcount = 10;
	my @array = ();

	@array = qx{/bin/ping6 -n -c$pingcount -i$interval -I$inter $host};

	$array[$pingcount + 4] =~ /\/(\d*\.\d*)\//;
	$avg = $1;

	$array[$pingcount + 3] =~ /(\d*)%/;
	$packet_los = $1;

	$array[1] =~ /ttl=(\d*)\s+/;
	$packet_ttl = $1;

	return($avg,$packet_los,$packet_ttl);

}

sub bandwidth {
	#Ejecuta el comando ping sobre las interfaces
	$inter = $_[0];
	open (CHECKBOOK, "prio.txt") || die "El archivo no existe.";
	foreach $dat(<CHECKBOOK>){
		if($record =~ m/$inter\s(\d+)\s(\d+)/){
			$ban1 = $1;
			$ban2 = $2;
		}
```

```
	}
	close(CHECKBOOK);
	return($ban1,$ban2);
}

	open (CHECKBOOK, "/proc/net/dev") || die "El archivo no existe.";

	$count = 0;
	while ($record = <CHECKBOOK>) {
		#Se descartan las interfaces definidas por defecto
		#eth0 sólo se usa para conectar con la red interna (MNN) y no como
una
		#interfaz de egreso.
		if($record                                                    !~
(m/^(\s*ip6tnl.*|\s*lo.*|\s*sit.*|\s*Inter.*|\s*face.*|\s*eth0.*)/)){

			@valor = split(/[\s*:]/,$record);

			$inter[$count] = $valor[2];
			@resp_ping = ping($inter[$count]);
			$rtt[$count] = $resp_ping[0];
			$p_los[$count] = $resp_ping[1];
			$p_ttl[$count] = $resp_ping[2];
			@resp_band = bandwidth($inter[$count]);
			$band_act = @resp_band[0];
			$band_lim = @resp_band[1];
			$count++;
		}
	}

	#Archivo prio.txt mantiene los flujos indicados como críticos para el
sistema y
	#su prioridad asignada

	$conta = 0;
	open (CHECKBOOK, "prio.txt") || die "El archivo no existe.";
	foreach $dat(<CHECKBOOK>){
		$dat =~ /(\w+)\s+(\w+|-)\s+(\d+)\s+(\w+)/;
		$prior[$conta] = $3; #Prioridad
		$tipo[$conta] = $4; #Tipo flujo
		if($conta == 0){
			$prio_l[0] = $prior[$conta];
			$prio_l[1] = $prior[$conta];
			$flujo = $tipo[$conta];
		}else{
			if($prior[$conta]<$prio_l[0]){
				$prio_l[0] = $prior[$conta];
				$flujo = $tipo[$conta];
			}
			if($prior[$conta]>$prio_l[1]){
				$prio_l[1] = $prior[$conta];
			}
		}
		$conta++;
	}
	close (CHECKBOOK);

	#Parámetros de importancia elegidos según tipo de tráfico

	if ($flujo =~ m/ssh/){
		#Ajustado para preferencia con el tiempo de respuesta
```

```
              $f_p_los = 0.3;
              $f_rtt = 0.5;
              $f_p_ttl = 0.2;
       } elsif ($flujo =~ m/video/){
              #Ajustado para preferencia con el número de paquetes perdidos y el
              #número de saltos necesarios
              $f_p_los = 0.5;
              $f_rtt = 0.1;
              $f_p_ttl = 0.4;
       } elsif ($flujo =~ m/www/){
              #Ajustado para preferencia en tiempo de respuesta
              $f_p_los = 0.3;
              $f_rtt = 0.4;
              $f_p_ttl = 0.3;
       } else {
              #Otro tráfico ajustado acorde al número de paquetes perdidos
              $f_p_los = 0.4;
              $f_rtt = 0.3;
              $f_p_ttl = 0.3;
       }

       #Cálculo de métrica para ordenar las interfaces
       $conta_a = 0;
       while ($conta_a < $count){
              if($conta_a == 0){
                     $mejor_i = $inter[$conta_a];
                     $inter_el = $conta_a;
                     $metrica  =  $f_p_los*$p_los[$conta_a]+$f_rtt*$rtt[$conta_a]-
$f_p_ttl*$p_ttl[$conta_a];
              }else{
                     if($f_p_los*$p_los[$conta_a]+$f_rtt*$rtt[$conta_a]-
$f_p_ttl*$p_ttl[$conta_a]<$metrica){
                            $mejor_i = $inter[$conta_a];
                            $inter_el = $conta_a;
                            $metrica                                                =
$f_p_los*$p_los[$conta_a]+$f_rtt*$rtt[$conta_a]-$f_p_ttl*$p_ttl[$conta_a];
                     }
              }
              $conta_a++;
       }

   close(CHECKBOOK);

push(@param,$mejor_i);
if($inter_el == 0){
       push(@param,$inter[1]);
}else{
       push(@param,$inter[0]);
}
$par = join(" ",@param);

system("perl adm_flow.pl $par");

exit 0;
```

ADM_FLOW.PL

```
#!/usr/bin/perl

#############################################################
#adm_flow.pl                                                #
```

```
#Version: 1.0                                                  #
#Descripción: Asigna flujos a interfaces según el estado de#
#éstas, pudiendo determinar el añadir o eliminar políticas #
#                                                              #
################################################################

#Los parámetros recibidos corresponden a las interfaces (en orden de
preferencia)
#disponibles

$parametros = $#ARGV;
print $parametros;
$cont = 0;

if($parametros > 0){
	#Existe más de una interfaz disponible

	$conta = 0;

	#Archivo prio.txt mantiene los flujos indicados como críticos para el
sistema y
	#su prioridad asignada
	#Menor prioridad, más importante.
	#Se manejan dos niveles de prioridad (alto = 1 ; bajo = 2)

	open (CHECKBOOK, "prio.txt") || die "El archivo no existe.";
	foreach $dat(<CHECKBOOK>){
		$dat =~ /(\w+)\s+(\w+|-)\s+(\d+)/;

		$prot[$conta] = $1;  #Protocolo
		$port[$conta] = $2;  #Puerto
		$prior[$conta] = $3; #Prioridad
		if($conta == 0){
			$prio_l[0] = $prior[$conta];
			$prio_l[1] = $prior[$conta];
		}else{
			if($prior[$conta]<$prio_l[0]){
				$prio_l[0] = $prior[$conta];
			}
			if($prior[$conta]>$prio_l[1]){
				$prio_l[1] = $prior[$conta];
			}
		}
		$conta++;
	}
	close (CHECKBOOK);

	if($prio_l[0] != $prio_l[1]){
		#Distintas prioridades en los flujos indicados.  Posibilidad de
		#asignarlas a distintas interfaces
		$conta2 = 0;
		while($conta2 < $conta){
			@param = @borra;
			push(@param,"add");
			push(@param,"-opt A");
			if ($prior[$conta2] == $prio_l[0]){
				push(@param,"-inter",$ARGV[0]);
			}else{
				push(@param,"-inter",$ARGV[1]);
			}
			push(@param,"-prot",$prot[$conta2]);
```

```
                        push(@param,"-port",$port[$conta2]);
                        $par = join(" ",@param);

                        system("perl valida_ip.pl $par");
                        $conta2++;
                }
        }else{
                #Misma prioridad requerida, se envia todo por la mejor interfaz
disponible.
                $conta2 = 0;
                while($conta2 < $conta){
                        push(@param,"add");
                        push(@param,"-opt A");
                        push(@param,"-inter",$ARGV[0]);
                        push(@param,"-prot",$prot[$conta2]);
                        push(@param,"-port",$port[$conta2]);
                        $par = join(" ",@param);

                        system("perl valida_ip.pl $par");
                        $conta2++;
                }
        }

}else{
        #Una sola interfaz disponible.  No es posible dividir el tráfico.
        #Se eliminan las politicas creadas.

        $readfile="policies.txt";
        if ( (-e $readfile) && (-r $readfile) )
        {
                #Revisar por presencia de politica
                #Se eliminan las politicas actuales.

                open  (CHECKBOOK,  "policies.txt")  ||  die  "Algun  error  en  el
proceso.";
                foreach $dat(<CHECKBOOK>){
                        @param = @borra;
                        push(@param,"delete");
                        push(@param,"-opt D");
                        push(@param,"-inter",$ARGV[0]);
                        if($dat =~ m/.+tcp\s(\d+).*/){
                                $port = $1;
                                $proto = "tcp";

                                push(@param,"-prot",$proto);
                                push(@param,"-port",$port);
                        }
                        if($dat =~ m/.+udp\s(\d+).*/){
                                $port = $1;
                                $proto = "udp";

                                push(@param,"-prot",$proto);
                                push(@param,"-port",$port);
                        }
                        if($dat =~ m/.+icmpv6.*/){
                                $port = "0";
                                $proto = "icmpv6";

                                push(@param,"-prot",$proto);
                                push(@param,"-port",$port);
                        }
```

```
                    $dat =~ m/.*bid\s(\d+).*/;
                    $bid = $1;

                    push(@param,"-bid",$bid);

                    $dat =~ m/.*idaddr\s(.+)\/64.*/;
                    $add = $1."/64";

                    push(@param,"-idaddr",$add);

                    $par = join(" ",@param);

                    system("perl valida_ip.pl $par");

             }
             close (CHECKBOOK);
      }
      #Se inicializa archivo de políticas
      open (POLITICA, '>policies.txt');
      close(POLITICA);
}

exit 0;
```

VALIDA_IP.PL

```
#!/usr/bin/perl

#################################################################
#valida_ip.pl                                                   #
#Version: 1.0                                                   #
#Descripción: Valida la posibilidad de crear o eliminar la      #
#política indicada.                                             #
#                                                               #
#################################################################

#Los parámetros recibidos corresponden a una posible acción
#(creación o eliminación de política) sobre el tráfico

die "\n" unless $ARGV[0];

$param = $#ARGV;
print $param;
$cont = 0;

$_=shift(@ARGV);
if (m/^add$/){
      #Caso para agregar nueva política.

      while ($cont < $param){

            $_=shift(@ARGV);
            if (m/^-inter$/){
                  $inte=shift(@ARGV);

                  #Archivo ha.txt mantiene información local necesaria
                  #para funcionamiento de Mobile IPv6

                  open (INTER, "ha.txt") || die "El archivo no existe.";
                  foreach $int(<INTER>){
```

```
                        if($int =~ m/^HoA\s+(.*)/){
                                $hoa = $1;
                                push(@param,"-idaddr",$hoa);
                        }
                        if($int =~ m/^HaA\s+(.*)/){
                                $haa = $1;
                                push(@param,"-haa",$haa);
                        }
                        if($int =~ m/^Interface\s$inte.+bid\s(.*)/){
                                $bid = $1;
                                push(@param,"-bid",$bid);
                        }
                }
                close (INTER);
        } elsif (m/^-prot$/) {
                $proto=shift(@ARGV);
                push(@param,"-".$proto);
        } elsif (m/^-port$/) {
                $dsport=shift(@ARGV);
                push(@param,$dsport);
        } elsif (m/^-opt$/) {
                $opt=shift(@ARGV);
                push(@param,"-opt",$opt);
        }
        $cont++;
}

$conta = 0;
$readfile="policies.txt";
if ( (-e $readfile) && (-r $readfile) )
{
        #Se valida que la política no exista previamente

        open (CHECKBOOK, "policies.txt") || die "Algun error en el
proceso.";
        foreach $dat(<CHECKBOOK>){
                if(($dat =~ m/.+$proto\s.*$dsport.*/) && ($dat =~
m/.*bid\s$bid.*/)){
                        last;
                        $conta = 1;
                }
        }
        close (CHECKBOOK);
}

if($conta == 0){
        #Política no existe.  Se procede a crearla.

        $readfile="policies.txt";
        if ( (-e $readfile) && (-r $readfile) )
        {
                open (POLITICA, ">>policies.txt") || die "Algun error en el
proceso.";
        }else
        {
                open (POLITICA, '>policies.txt');
        }
        #Registro de la política en el archivo de políticas: policies.txt
        print POLITICA "idaddr ".$hoa." ";
        print POLITICA $proto." ".$dsport." ";
        print POLITICA "bid ".$bid." ";
```

```
                print POLITICA "\n";
                close (POLITICA);

                $par = join(" ",@param);

                #Envío de política para su ejecución
                system("perl send_ip.pl $par");
        }
}else{
        #Caso para eliminar política.

        while ($cont < $param){

                $_=shift(@ARGV);
                if (m/^-inter$/){
                        $inte=shift(@ARGV);
                } elsif (m/^-prot$/) {
                        $proto=shift(@ARGV);
                        push(@param,"-".$proto);
                } elsif (m/^-port$/) {
                        $dsport=shift(@ARGV);
                        if($proto ne "icmpv6"){
                                push(@param,$dsport);
                        }
                } elsif (m/^-bid$/) {
                        $bid=shift(@ARGV);
                        push(@param,"-bid",$bid);
                } elsif (m/^-idaddr$/) {
                        $add=shift(@ARGV);
                        push(@param,"-idaddr",$add);
                } elsif (m/^-opt$/) {
                        $opt=shift(@ARGV);
                        push(@param,"-opt",$opt);
                }
                $cont++;
        }

        $par = join(" ",@param);

        #Envío de política para su ejecución
        system("perl send_ip.pl $par");

}

exit 0;
```

SEND_IP.PL

```
#!/usr/bin/perl

#################################################################
#send_ip.pl                                                     #
#Version: 1.0                                                   #
#Descripción: Ejecuta una política                              #
#                                                               #
#                                                               #
#################################################################

#Los parámetros recibidos corresponden a una posible acción
#(creación o eliminación de política) sobre el tráfico
```

```
while(@ARGV && $ARGV[0] =~ m/^-/) {
	#Se forma la política en formato entendible por el comando ip6tables
	#ej. ip6tables -A PREROUTING -t mangle -p icmpv6 -j MARK --set-mark 100

	$_=shift(@ARGV);
	if (m/^-idaddr$/) {
		$mn=shift(@ARGV);

	} elsif (m/^-tcp$/) {
		$dsport=shift(@ARGV);

		push(@param,"-p tcp");
		push(@param,"--dport",$dsport);
		push(@param_s,"-p tcp");
		push(@param_s,"--sport",$dsport);
	} elsif (m/^-udp$/) {
		$dsport=shift(@ARGV);

		push(@param,"-p udp");
		push(@param,"--dport",$dsport);
		push(@param_s,"-p udp");
		push(@param_s,"--sport",$dsport);
	} elsif (m/^-icmpv6$/) {
		push(@param,"-p icmpv6");
		push(@param_s,"-p icmpv6");
	} elsif (m/^-bid$/) {
		$bid=shift(@ARGV);

		push(@param,"-j MARK --set-mark",$bid);
		push(@param_s,"-j MARK --set-mark",$bid);
	} elsif (m/^-haa$/) {
		$haa=shift(@ARGV);
	} elsif (m/^-opt$/) {
		$opt=shift(@ARGV);
		push(@param,"-$opt PREROUTING");
		push(@param_s,"-$opt PREROUTING");
	}
}


push(@param,"-t mangle");
push(@param_s,"-t mangle");

$par = join(" ",@param);
$par_s = join(" ",@param_s);

#Ejecución del comando ip6tables en el dispositivo local
system("/sbin/ip6tables $par");

#Envío de la política para su ejecución en el dispositivo central
system("./cliente $haa $par_s");

exit 0;
```

BANDWIDTH.SH

```
#!/usr/bin/perl -w

################################################################
#bandwidth.pl                                                  #
#Version: 3.0                                                  #
```

```
#Descripción: Obtiene información del ancho de banda (subida#
#y bajada) disponible en las distintas interfaces, calculado#
#según las transmisiones establecidas.  Se ejecuta          #
#continuamente.                                             #
###############################################################

use strict;

use Time::HiRes qw( gettimeofday );

select(STDOUT); $| = 1;

my $TIMEVAL_T = "LL";

my $delay;
$delay = 1;

# Leer /proc/net/dev
open(PROC, "</proc/net/dev") || die "Error al abrir /proc/net/dev\n";

$_ = <PROC>;
if ($_ !~ m#^[^|]+\|\s*Receive\s*\|\s*Transmit\s*$#) {
       die "Secuencia no reconocida en /proc/net/dev";
}
$_ = <PROC>;
my ($r_fields, $t_fields) = m#^[^|]+\|([^|]+)\|([^|]+)$#;

$r_fields = $r_fields." mayor_bw menor_bw prom_bw";
$t_fields = $t_fields." mayor_bw menor_bw prom_bw";
my @fields = map { "r_" . $_ } split(' ', $r_fields);
push(@fields, map { "t_" . $_ } split(' ', $t_fields));
print "**fields ".@fields."\n";
close(PROC);

my %prev_stats;

my $actual_delay;

my $vez_ej = 1;
my $activado = 0;
my $prom_t = 0;
my $prom_r = 0;

while (1) {
       open(PROC, "</proc/net/dev") || die "Error al abrir /proc/net/dev\n";

       # Cabecera del archivo
       $_ = <PROC>;
       $_ = <PROC>;

       my $non_zero_iface = 0;
       my %totals;
       $totals{'r_bytes'} = 0;
       $totals{'t_bytes'} = 0;

       while (<PROC>) {

              my ($iface, $rest) = m#^\s*(\S+):(.*)$#;

              if($iface !~ (m/^(\s*ip6tnl.*|\s*lo.*|\s*sit.*|\s*eth0.*)/)){
```

```
                        my @data = split(' ', $rest);

                        my %delta;
                        my $non_zero_field = 0;

                        foreach my $i (@fields) {
                                if (!defined($prev_stats{$iface}{$i})) {
                                        $prev_stats{$iface}{$i} = 0;
                                }
                                #$x --> actual lectura desde archivo

                                if($i !~
(m/^(.*mayor_bw.*|.*menor_bw.*|.*prom_bw.*)/)){
                                        my $x = shift @data;
                                        #diferencia con lectura anterior
                                        $delta{$i} = $x - $prev_stats{$iface}{$i};
                                        if ($delta{$i} < 0) {
                                                $delta{$i} += 2*(1<<31);
                                        }
                                        if ($delta{$i} != 0) {
                                                #Si existe algun cambio
                                                $non_zero_field = 1;
                                                $non_zero_iface = 1;
                                        }
                                        $prev_stats{$iface}{$i} = $x;
                                }
                        }

                        next unless $non_zero_field;

                        if (defined($actual_delay)) {
                                #Ancho de banda actual en Kbits

                                $totals{'r_bytes'} += int($delta{'r_bytes'} /
$actual_delay + 0.5)*8;
                                $totals{'t_bytes'} += int($delta{'t_bytes'} /
$actual_delay + 0.5)*8;
                                printf "%6s %9u %9u", $iface,
                                        int($delta{'r_bytes'} / $actual_delay + 0.5)*8,
                                        int($delta{'t_bytes'} / $actual_delay + 0.5)*8;

                                foreach my $i (@fields) {
                                        if($i !~
(m/^(.*mayor_bw.*|.*menor_bw.*|.*prom_bw.*)/)){
                                                next if $i eq 'r_bytes';
                                                next if $i eq 'r_packets';
                                                next if $i eq 't_bytes';
                                                next if $i eq 't_packets';
                                                next if $delta{$i} == 0;
                                                print " *** $i $delta{$i}";
                                        }
                                        if(($i =~ (m/^(.*r_mayor_bw.*)/)) &&
($prev_stats{$iface}{$i}<$totals{'r_bytes'})){

        $prev_stats{$iface}{$i}=$totals{'r_bytes'};
                                                if($vez_ej == 1){

        $prev_stats{$iface}{"r_menor_bw"}=$totals{'r_bytes'};
                                                }
                                        }
```

```
                                if(($i =~ (m/^(.*r_menor_bw.*)/)) &&
($prev_stats{$iface}{$i}>$totals{'r_bytes'})){

      $prev_stats{$iface}{$i}=$totals{'r_bytes'};
                                }

                                if(($i =~ (m/^(.*t_mayor_bw.*)/)) &&
($prev_stats{$iface}{$i}<$totals{'t_bytes'})){

      $prev_stats{$iface}{$i}=$totals{'t_bytes'};
                                      if($vez_ej == 1){

      $prev_stats{$iface}{"t_menor_bw"}=$totals{'r_bytes'};
                                            $activado = 1;

                                      }
                                }
                                if(($i =~ (m/^(.*t_menor_bw.*)/)) &&
($prev_stats{$iface}{$i}>$totals{'t_bytes'})){

      $prev_stats{$iface}{$i}=$totals{'t_bytes'};
                                }
                                if($activado == 1){
                                      $vez_ej = 0;
                                }
                                if($i =~ (m/^(.*r_prom_bw.*)/)){

      $prev_stats{$iface}{"r_prom_bw"}=int(($prev_stats{$iface}{"r_prom_bw"}+$t
otals{'r_bytes'})/2);
                                }
                                if($i =~ (m/^(.*t_prom_bw.*)/)){

      $prev_stats{$iface}{"t_prom_bw"}=int(($prev_stats{$iface}{"t_prom_bw"}+$t
otals{'t_bytes'})/2);
                                }
                          }

                          open (MYFILE, '>prom_iface.txt');
                          print MYFILE $iface."
".$prev_stats{$iface}{"r_prom_bw"}." ".$prev_stats{$iface}{"t_prom_bw"}."\n";
                          close(MYFILE);
                          print "-->".$prev_stats{$iface}{"r_mayor_bw"}."
".$prev_stats{$iface}{"t_mayor_bw"}." ";
                          print "++>".$prev_stats{$iface}{"r_prom_bw"}."
".$prev_stats{$iface}{"t_prom_bw"}." ";
                          print "-->".$prev_stats{$iface}{"r_menor_bw"}."
".$prev_stats{$iface}{"t_menor_bw"}."\n";
                    }
              }
      }
      close(PROC);

      my @start = gettimeofday;
      sleep($delay);
      my @done = gettimeofday;
      $actual_delay = ($done[0] + $done[1] / 1_000_000) - ($start[0] +
$start[1] / 1_000_000);

      my $delta = $actual_delay - $delay;

}
```

CLIENTE.C

```
/*****************************************************************************/
/*** cliente.c                                                             ***/
/*** Version: 1.0                                                          ***/
/*** Cliente para conexiones IPv6.                                         ***/
/*** Parámetros: - Dirección IPv6 de Home Agent                            ***/
/***             - Política a transmitir                                   ***/
/*****************************************************************************/

#include <stdio.h>
#include <errno.h>
#include <resolv.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <ctype.h>
#include <stdarg.h>

void panic(char* msg, ...)
{   va_list ap;

    va_start(ap, msg);
    vprintf(msg, ap);
    va_end(ap);
    abort();
}

int main(int count, char *strings[])
{     int sd,sent = 0;
      int portnum = 9999,cont = 0;
      struct sockaddr_in6 addr;
      char line[100],*parametros;

    if ( count <= 1 ){
      printf("Parámetros insuficientes\n");
    }else

      /*Inicialización de la conexión bajo sockets IPv6 con el Home Agent
indicado*/

      bzero(&addr, sizeof(addr));
      if ( (sd = socket(PF_INET6, SOCK_STREAM, 0)) < 0 )
            panic("Error en socket");
      addr.sin6_family = AF_INET6;
      addr.sin6_port = htons(portnum);
      if ( inet_pton(AF_INET6, strings[1], &addr.sin6_addr) == 0 )
            panic("Error al inicializar");
      if ( connect(sd, (struct sockaddr*)&addr, sizeof(addr)) != 0 )
            panic("Error al conectar en v6");

      for(cont=2;cont<count;cont++){
            send(sd, strings[cont], strlen(strings[cont])+1, 0);
            recv(sd, line, sizeof(line), 0);
            printf("client=%s", line);
      }
      strcpy(line,"end");
      sent = send(sd, line, strlen(line)+1, 0);

      close(sd);
}
```

SERVIDOR.C

```
/*****************************************************************************/
/*** server.c                                                              ***/
/*** Version: 2.0                                                          ***/
/*** Servidor para servicios IPv6                                          ***/
/*****************************************************************************/

#include <stdio.h>
#include <errno.h>
#include <resolv.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <ctype.h>
#include <stdarg.h>
#include <stdlib.h>
#include <string.h>

void panic(char* msg, ...)
{     va_list ap;

      va_start(ap, msg);
      vprintf(msg, ap);
      va_end(ap);
      abort();
}

int main(int count, char *strings[])
{     int sd, portnum;
      struct sockaddr_in6 addr;
      char parametros[200];

      int sent, size=sizeof(addr);
      int client,count1 = 0,count2 = 0,numLineas = 100, flag = 0;
      char line[100];
      char *arrayIndices[50];

      if ( count == 2 )
            portnum = atoi(strings[1]);
      else
            portnum = 9999;

      /*Inicialización de la conexión bajo sockets IPv6 en el Home Agent*/

      bzero(&addr, sizeof(addr));
      if ( (sd = socket(PF_INET6, SOCK_STREAM, 0)) < 0 )
            panic("Error en socket");
      addr.sin6_family = AF_INET6;
      addr.sin6_port = htons(portnum);
      if ( inet_pton(AF_INET6, "0::0", &addr.sin6_addr) == 0 )
            panic("Error en inicialización");
      if ( bind(sd, (struct sockaddr*)&addr, sizeof(addr)) != 0 )
            panic("Error en enlace");
      if ( listen(sd, 15) != 0 )
            panic("Error en conexión");

      while (1)
      {
            for(count2=0; count2<50; count2++ )
            {
                arrayIndices[count2] = malloc(100*sizeof(char));
```

```
            }

            /*Se establece la comunicación con el dispositivo móvil*/
            client = accept(sd, (struct sockaddr*)&addr, &size);
            count1 = 0;
            do
            {
                  /*Se reciben los parámetros y se forma la sentencia a
ejecutar
                  usando ip6tables*/

                  sent = send(client, line, recv(client, line, sizeof(line),
0), 0);
                  strcpy(arrayIndices[count1],line);

                  count1++;

                  if(strcmp(line,"end") != 0){
                        strcat(parametros,line);
                        strcat(parametros," ");
                  }
            }
            while (strcmp(line,"end") != 0 );
            close(client);

            count2=0;
            flag = 0;
            /*Se revisa que la cadena recibida se encuentre en un formato
válido*/
            while((count2<(count1-1)) && (flag == 0)){
                  if((strcmp(arrayIndices[count2],"-A")      ==      0)      ||
(strcmp(arrayIndices[count2],"-D") == 0)){
                        count2++;
                        if((strcmp(arrayIndices[count2],"PREROUTING") != 0) &&
(strcmp(arrayIndices[count2],"POSTROUTING") != 0)){
                              flag = 1;
                        }
                        count2++;
                  }else{
                        if(strcmp(arrayIndices[count2],"-t") == 0){
                              count2++;
                              if((strcmp(arrayIndices[count2],"mangle")    !=
0)){
                                    flag = 1;
                              }
                              count2++;
                        }else{
                              if(strcmp(arrayIndices[count2],"-p") == 0){
                                    count2++;
                                    if((strcmp(arrayIndices[count2],"tcp")
== 0) || (strcmp(arrayIndices[count2],"udp") == 0)){
                                          count2++;
                                          if((strcmp(arrayIndices[count2],"-
-sport ") == 0) && (isdigit(arrayIndices[count2++]) != 0)){
                                                flag = 0;
                                                count2++;
                                          }else
                                                flag = 1;
                                    }else{

      if((strcmp(arrayIndices[count2],"icmpv6") != 0)){
```

```
                                                flag = 1;
                                          }else{
                                                count2++;
                                          }
                                    }
                              }else{
                                    if(strcmp(arrayIndices[count2],"-j")   ==
0){
                                          count2++;

      if((strcmp(arrayIndices[count2],"MARK") == 0)){
                                                count2++;

      if((strcmp(arrayIndices[count2],"--set-mark") == 0)){
                                                      count2++;

      if((atoi(arrayIndices[count2])  >=  0)  &&  (atoi(arrayIndices[count2])  <=
512)){
                                                            flag = 0;
                                                            count2++;
                                                      }else
                                                            flag = 1;

                                                }else{
                                                      flag = 1;
                                                }
                                          }else{
                                                flag = 1;
                                          }
                                    }else{
                                          flag = 1;
                                    }
                              }
                        }
                  }
            }
            if(flag == 0){
                  /*Se ejecuta la secuencia recibida en el Home Agent*/
                  if(!fork()){

                        execv("/sbin/ip6tables",parametros,NULL);

                        for(count2=0; count2<50; count2++ )
                        {
                            free(arrayIndices[count2]);
                        }

                        for(count2=0; count2<200; count2++ )
                        {
                            parametros[count2]=0;
                        }
                        printf("Regla Ejecutada");
                        return -1;
                  }else{
                        printf("Error al ejecutar comando");
                  }
            }else{
                  printf("Cadena Erronea. No ejecutada");
            }
      }
}
```