UNIVERSIDAD CARLOS III DE MADRID
Departamento de Ingeniería Telemática

TESIS DOCTORAL

Herramientas para la Conectividad IPv6 con Múltiples Proveedores

Autor: Marcelo Bagnulo Braun
Ingeniero Electricista

Tutor: Alberto García Martínez
Doctor Ingeniero de Telecomunicación

Leganés, mayo de 2005

Agradecimientos

El autor de la presente Tesis Doctoral quiere expresar su agradecimiento a las siguientes personas: Lili, Homero, Sylvia, Adriana, Homero O., Alberto, Tere, Marco, Arturo, Marifeli, Lulu, Yoli, Ignacio, Maria, Paco, David, Richi, Ivan, Carlos Jesús, Isaías, Juan Francisco, Albert, Carmen, Huw, Manolo, Jaime, Pablo, Jose Felix, Carlos G.G., Guillermo, Celeste, Carlos G.R. y el resto del Departamento de Ingeniería Telemática de la Universidad Carlos III de Madrid.

Resumen

La presente Tesis Doctoral propone una arquitectura arquitectura para el soporte escalable de multihoming en IPv6.
Las motivaciones para el desarrollo de dicha arquitectura vienen dadas por las limitaciones de escalabilidad identificadas en la solución de multihoming existente en IPv4. En la solución de multihoming en IPv4, cada sitio multihomed obtiene un bloque de direcciones propio y lo anuncia a través de todos sus proveedores usando el protocolo BGP. El resultado es que cada sitio multihomed contribuye con al menos una entrada en la tabla global de rutas. Dicho crecimiento es considerado incompatible con el número proyectado de usuarios multihomed en el futuro inmediato, donde la adopción del nuevo protocolo IPv6 y el despliegue universal de diversas tecnologías de acceso de banda ancha como pueden ser los distintos sabores de DSL o el acceso por cable habilitarán la adopción de configuraciones multihomed en ambientes de pequeñas oficinas e incluso domésticos.
Para garantizar la escalabilidad del sistema de rutas, se propone la adopción universal de direcciones agregables por proveedor (PA), que son asignadas por el proveedor, anunciando éste sólo su propio bloque en el sistema de rutas de interdominio. En éste escenario, un sitio multihomed obtiene tantos prefijos como proveedores tiene y los nodos dentro del sitio en cuestión deberán configurar múltiples direcciones por interfaz, una por prefijo disponible. Cabe notar que debido a que las direcciones son PA, una dirección es sólo alcanzable a través del proveedor que la ha asignado.
La configuración resultante preserva la escalabilidad del sistema de rutas, pero presenta una serie de dificultades que se presentan a continuación:

- Incompatibilidad con los filtros de ingreso, resultante del hecho que un sitio posee direcciones procedentes de distintos proveedores, y que los filtros de ingreso de dichos proveedores típicamente no aceptan paquetes cuya dirección origen no pertenezca al bloque delegado por ellos mismos.
- Imposibilidad de establecer comunicaciones nuevas después de un fallo. Después de que se produzca un fallo, para el establecimiento exitoso de nuevas comunicaciones será necesario el uso de aquellas direcciones asociadas a proveedores que no se han visto afectados por el fallo. Esto implica una inteligencia a la hora de la selección de direcciones que no se encuentra disponible actualmente.
- Imposibilidad de preservar las comunicaciones a través de fallos. Cuando se produce un fallo que afecta uno de los proveedores de sitio, las direcciones delegadas por dicho proveedores se vuelven inaccesibles. Esto implica que para continuar con el intercambio de paquetes necesario para preservar una comunicación establecida es necesario el uso de

una dirección alternativa, asociada a otro proveedor. El problema es que el uso de una dirección alternativa implica la ruptura de la comunicación, ya que las capas de transporte y aplicación existentes identifican los extremos involucrados en una comunicación por la dirección usada.

- Dificultades para la provisión de capacidades de ingeniería de tráfico. Actualmente, la solución de multihoming disponible permite la manipulación de los parámetros del protocolo BGP para obtener capacidades de ingeniería de tráfico. El uso de múltiples prefijos en los sitios multihomed implican que dichas técnicas de ingeniería de tráfico no estarán disponibles en la nueva configuración, por lo que será necesario contar con mecanismos alternativos para la provisión de dichas capacidades.

Para solventar las dificultades identificadas, la presente Tesis propone una arquitectura de multihoming que cuenta con los siguientes componentes:

- Un sistema de encaminamiento basado en dirección origen dentro del sitio multihomed para asegurar la compatibilidad con los filtros de ingreso. A través de dicho sistema de encaminamiento, los paquetes serán encaminados a través del proveedor que ha delegado el prefijo contenido en la dirección origen.
- Un sistema de selección de direcciones que tiene en cuenta si las direcciones usadas están alcanzables. La información de alcanzabilidad de una dirección dada es descubierta a través de un mecanismo de ensayo y error.
- Una nueva capa de identificación dentro de la capa de red que separa el identificador presentado a las capas de transporte y aplicación de los localizadores usados en los paquetes para transmitir los paquetes de un extremo a otro. Ésta separación permite el uso de distintas direcciones a lo largo de la vida de una comunicación para encaminar los paquetes, mientras que dicho paquetes se presentan a las capas superiores como provenientes siempre de una misma dirección, preservando la comunicación establecida.
- Mecanismos de configuración de la tabla de políticas definidas por el mecanismo de selección de direcciones por defecto para la provisión de capacidades de ingeniería de tráfico.

La presentación detallada de los mencionados mecanismos se realiza después de la realización de un análisis en profundidad que nos permite concluir que los mecanismos propuestos son los más idóneos para solventar los problemas identificados.

Palabras clave: IPv6, multihoming, tolerancia a fallos, escalabilidad, identificadores criptográficos.

Abstract

In this Thesis we propose an architecture for the provision of scalable IPv6 multihoming support.
In the multihoming solution currently deployed in the IPv4 Internet, the multihomed site announces a route to its address blocks through all the providers using BGP. The result is that multiple routes towards the multihomed site are available in the inter-domain routing system. While this solution provides the fault tolerance and path selection features required to a multihoming solution, it presents limited scalability, since each multihomed site contributes with at least one routing table entry in the already oversized inter-domain routing tables.
Because the support of the multihoming solution currently deployed in the IPv4 Internet is becoming challenging even for the current number of multihomed sites, this approach is deemed unsuitable for the expected number of multihomed sites in the future IPv6 Internet, especially when considering that the wide adoption of low-budget broadband access technologies such as ADSL or CATV will enable multihoming in SOHO environments. As a consequence, an alternative multihoming solution for IPv6 is needed. The requirements imposed to the new solution essentially include all the benefits provided by the incumbent solution, i.e. fault tolerance and traffic engineering capabilities, and also an enhanced scalability with respect to the number of multi-homed sites and other relevant Internet parameters. In order to preserve routing system scalability, aggressive route aggregation can be achieved through provider-based aggregation, precluding the injection of routes associated with individual multi-homed end-sites. When Provider Aggregatable (hereafter PA) addressing is used, multi-homed sites obtain one prefix per each one of their providers. Consequently, as each provider will only announce its own prefix to the rest of the Internet, a given provider will be used to reach the multihomed site only when the destination addresses used belong to the prefix associated with the provider. So, in order to be reachable through all of the providers of the site, each host within the multihomed site will have to configure multiple addresses, one per provider.

Even if this setup guarantees the scalability of the multihoming solution, such multi-addressed configuration is not without difficulties of its own when attempting to provide the additional features mentioned above. In particular, this configuration presents the following problems:

- Incompatibility with ingress filtering techniques: The incompatibility is caused by the lack of coordination between the IPv6 source address selection mechanism, performed by the host, and the path selection mechanism, performed by the intra-site routing system. As long as outgoing packets are routed through the provider that has delegated the prefix contained in the source address, packets will flow freely; but when those packets are routed through a different ISP, they will be discarded by the ingress filtering mechanism

due to source address incompatibility. It must be noted that because of this issue, packets may be discarded even in a scenario without failures.

- Difficulties when establishing new communications after an outage. The difficulties arise because not all of the addresses available for a multihomed host are reachable, so in order to be able to communicate, hosts need to properly discard unreachable addresses and select those addresses that are reacahable. Current address selection mechanisms are unable to cope with such situation.
- Difficulties when preserving established communications. In order to preserve established communications through outages, the endpoints of the communication have to adapt the addresses used during the lifetime of the communication according to the available providers. Moreover, this address replacement has to be performed in a transparent fashion with respect to transport and application layers, in order to actually preserve the established communication. Current applications and transport layers, such as TCP and UDP, identify the endpoints of a communication through the IP addresses of the nodes involved, implying that the IP addresses selected at the communication establishment time must remain invariant through the lifetime of the communication. But as it has been presented earlier, once that an outage has occurred in one of the available ISPs, the associated address becomes unreachable, so an alternative address has to be used in order to convey packets to the multi-homed host. These two constraints impose that after an outage, packets must carry a different address, corresponding to an available ISP, but they have to be presented to transport and application layers as if they contained the original address, in order to be recognized as belonging to the established communication. Such approach requires additional mechanisms in both ends of the communication in order to preserve a coherent mapping between the IP addresses presented to the transport and application layers and those addresses actually contained in the packets.
- Difficulties when providing traffic engineering capabilities. The usage of multiple prefixes pre multihomed site imply that those traffic engineering techniques will no longer apply, and alternative mechanisms that provide equivalent capabilities are required.

In this Thesis we describe an architecture for the provision of multihoming in IPv6 that deals with all the aforementioned concerns. The proposed IPv6 multihoming architecture introduces the following components:

- An intra-site routing paradigm that takes into account the source address, so that source hosts can determine through the selection of the source address, the exit path of the packets. Such feature provides ingress filtering compatibility.
- An address selection mechanism that takes into account address reachability information acquired through a trial and error procedure.
- A new Multihoming Sub-Layer within the IP layer that will perform the required mapping between the addresses that are presented to the upper layer protocols and the addresses that are actually used for exchanging packets in the network. Such layer allows the usage of different addresses for exchanging packets during the lifetime of a communication, while keeping unchanged the address presented to the upper layers, preserving the established communication.
- A mechanism for the configuration of the policy table defined in the default address selection procedure, for the provision of traffic engineering capabilities.

A detailed presentation of the aforementioned mechanisms is preceded by an exhaustive analysis of the solution space that justifies the selected approach.

Keywords: IPv6, multihoming, fault tolerance, scalability, cryptographic identifiers.

Índice general

Índice de figuras

Capítulo 1

Introducción

Internet se ha convertido en un recurso crítico para el funcionamiento de más y más instituciones de diversa naturaleza. Lejos están ya los días en que sólo las empresas relacionadas directamente con las tecnologías de la información eran las únicas para las cuales el acceso a Internet resultaba imprescindible para su operación. Hoy en día instituciones de toda naturaleza y tamaño requieren conectividad global ya sea para proveer servicios a través de Internet, para relacionarse con sus proveedores e incluso para el funcionamiento cotidiano de las operaciones internas. Esto implica que una interrupción en el acceso a Internet supone un alto coste, por lo que existe una fuerte demanda de mecanismos que brinden un alto nivel de tolerancia a fallos en la conexión a Internet.

Una opción cada vez más utilizada para ofrecer tolerancia a fallos consiste en contratar múltiples accesos a Internet a través de distintos proveedores de servicio de Internet, lo que se conoce normalmente como *multihoming*[1]. Esta configuración permite que si uno de los accesos sufre problemas, el dominio o *sitio* considerado pueda continuar conectado a Internet a través de los otros accesos existentes. Si bien las prestaciones ofrecidas por esta configuración son buenas, el coste de la misma es alto. En particular, la solución actualmente disponible en IPv4 para la

[1] A lo largo de este trabajo utilizaremos el término inglés multihoming para referirnos al problema aquí definido al no encontrar un término en castellano que sea a la vez correcto, elegante o conciso.

provisión de multihoming impone una ruta adicional en la tabla global de rutas por cada sitio multihomed, es decir que crece linealmente con el número de sitios que adoptan la solución. La experiencia nos ha mostrado en varias ocasiones que los sistemas de Internet que crecen linealmente con el número de sitios son difícilmente sustentables, ya que su coste se dispara cuando son adoptados masivamente. Este es efectivamente el caso de la solución actual de multihoming. A medida que los sitios adoptan esta configuración de multihoming, las tablas globales de rutas crecen proporcionalmente con el número de sitios, lo que implica dificultades operativas e inestabilidades en Internet. Con la adopción de IPv6 se espera que el número de dispositivos conectados a la red se dispare, por lo que es necesario que todos los mecanismos disponibles en IPv6 presenten buenas características de escalabilidad, y en particular que las presente la solución para multihoming. Es más, a medida que bajan los costes de las comunicaciones de banda ancha, se espera que en el futuro (con el uso de IPv6), más sitios sean multihomed, incluyendo no sólo a las empresas grandes que hoy pueden pagarlo, sino también a empresas y sitios mas pequeños. Los sitios más pequeños obtienen grandes beneficios del multihoming, ya que por su propio tamaño, no pueden acceder a servicios de tipo *premium* donde se garantice un nivel de servicio alto en cuanto a tolerancia a fallos. Por todo esto, es necesario encontrar una solución de multihoming en IPv6 que preserve la escalabilidad del sistema de rutas. En esta Tesis se propone investigar las distintas alternativas para brindar una solución escalable de multihoming en IPv6 y diseñar herramientas que la hagan posible.

Las herramientas propuestas constituyen una solución extremo a extremo para el soporte escalable de multihoming. La solución se basa en el uso de múltiples prefijos dentro del sitio multihomed, uno por cada uno de los proveedores disponibles en el sitio. Esta configuración permite, como veremos más adelante, reducir el número de entradas en la tabla global de rutas, gracias al uso general de la agregación por proveedor. El resultado de esta configuración es que cada nodo dentro del sitio multihomed poseerá múltiples direcciones, una por cada proveedor, y cada una de ellas estará asociada a un camino de salida. Esto implica que para cambiar el camino usado para cursar los paquetes, será necesario cambiar de dirección usada en la comunicación. Esto presenta una serie de dificultades, esencialmente debidas a que las aplicaciones y protocolos de transporte identifican los extremos mediante las direcciones IP involucradas en la comunicación, por lo que un cambio en las mismas implicaría la ruptura de la conexión establecida. Por ende, para poder efectivamente brindar una solución que preserve las comunicaciones establecidas a través de fallos en los caminos, será necesario crear un mecanismo que restaure las direcciones originales cuando los paquetes son entregados a las capas superiores (transporte y aplicación). El diseño de dicho mecanismo presenta a su vez una serie de dificultades, principalmente a nivel de seguridad. La solución diseñada en la presente Tesis ofrece una solución eficiente y elegante para garantizar la protección de los protocolos necesarios para el soporte de multihoming.

La presente Tesis está estructurada de la siguiente forma: En el siguiente capítulo presentaremos la solución de multihoming utilizada en IPv4 y sus limitaciones. En el capítulo 3 detallaremos los objetivos definidos para el diseño de una solución de multihoming para IPv6, para después en el capítulo 4 realizar un análisis en profundidad del espacio de soluciones. Dicho análisis estará compuesto de tres partes, a saber: un primer análisis arquitectónico, un segundo análisis de seguridad y finalmente un análisis funcional. Una vez explorado el espacio de soluciones e identificados los compromisos existentes. En el capítulo 5 presentaremos las grandes decisiones que se adoptarán para el diseño de la solución propuesta, para pasar en el capítulo 6 al detalle de las distintas herramientas que componen la solución planteada en la presente Tesis Doctoral. Finalmente, el presente estudio termina con un estudio de trabajos relacionados en el capítulo 7 y la presentación de las conclusiones y los trabajos futuros en el capítulo 8.

Capítulo 2

Multihoming en Sitios Finales: Antecedentes

2.1 Introducción

En esta sección presentaremos los conceptos fundamentales referentes al multihoming, así como las técnicas actualmente usadas para su provisión. Comenzaremos por definir el concepto de multihoming y las motivaciones que llevan a su adopción. A continuación presentaremos cómo ha evolucionado la arquitectura de direccionamiento en Internet y cómo esto ha influenciado en las soluciones de multihoming utilizadas. Finalmente, presentaremos los beneficios y limitaciones de la solución de multihoming actualmente adoptada así como posibles alternativas que han sido propuestas para sortear sus limitaciones.

2.2 Definiciones y motivaciones

En la presente sección introduciremos algunas definiciones relacionadas con el multihoming y exploraremos someramente las motivaciones que pueden existir para implantar multihoming.

Definición : Un *sitio final* es aquel en cuyas conexiones directas con otros sitios sólo fluyen paquetes dirigidos hacia el sitio considerado o paquetes originados en el sitio considerado.

Definición: Un *proveedor de servicio de Internet (ISP)* es un sitio que ofrece servicio de *tránsito* a sitios clientes, es decir que transporta paquetes originados en los sitios clientes hacia Internet, y paquetes desde Internet hacia los sitios clientes.

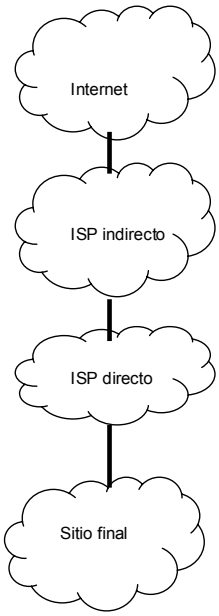

Figura 1: ISPs y sitios finales

Definición : Un *ISP directo* de un sitio final es un ISP con el que el sitio final tiene una conexión directa, i.e. existe un enlace directo entre el ISP directo y el sitio final.

Definición: Un *ISP indirecto* o *Upstream ISP* de un sitio final es un sitio que provee tránsito hacia Internet a uno de los ISPs directos del sitio final, o a otro ISP indirecto de sitio.

Definición: Un sitio es *multihomed* si obtiene acceso a Internet a través de dos o más ISPs directos.

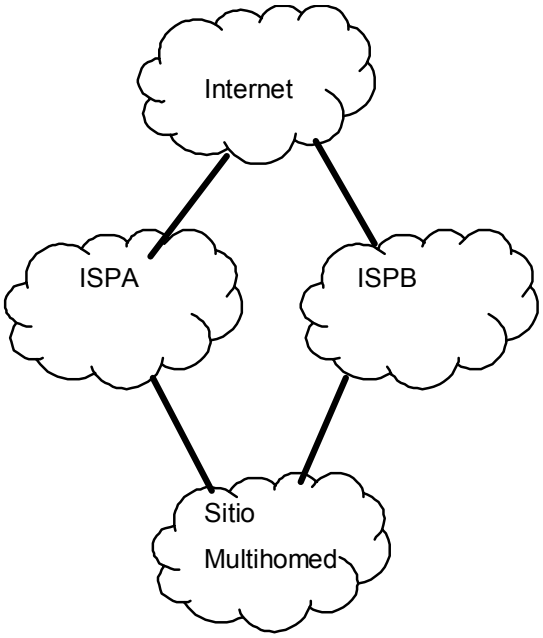

Figura 2: Sitio Multihomed

Existe una configuración similar que brinda un subconjunto reducido de los beneficios del multihoming, pero cuyo soporte es sensiblemente más simple, como explicaremos más adelante en este capítulo. Dicha configuración es llamada *multiconexión* y se define de la siguiente manera.

Definición: Un sitio es *multiconectado* si cuenta con dos o más enlaces a Internet a través del mismo ISP.

Como se puede apreciar, tanto un sitio multihomed como un sitio multiconectado cuentan con más de una conexión a Internet. La motivación fundamental para este tipo de configuraciones es mejorar la tolerancia a fallos de la conectividad global del sitio, de modo que si una de las conexiones falla, el sitio pueda obtener acceso a Internet a través de la otra conexión. Resulta claro que una configuración multihomed brinda una mayor tolerancia a fallos que una configuración multiconectada, ya que en esta última el sitio tiene un solo ISP, que se convierte en un punto simple de fallo. En cambio, en la configuración multihomed el sitio cuenta con dos ISPs, por lo que si uno de ellos falla completamente, el sitio aún puede contar con el otro. Como veremos posteriormente, una solución para soportar una configuración multiconectada es mucho más simple que la configuración requerida para soportar multihoming. Multihoming ofrece la posibilidad adicional de poder seleccionar el uso que da a sus ISPs. Dado que los distintos ISPs pueden brindar distintos tipos de servicios en lo que se refiere a calidad, ancho de banda, latencia,

coste, etc., el sitio deseará modelar los patrones de tráfico que envía y recibe a través de cada uno de sus ISPs. Esto se llama *ingeniería de tráfico* o *policing*.

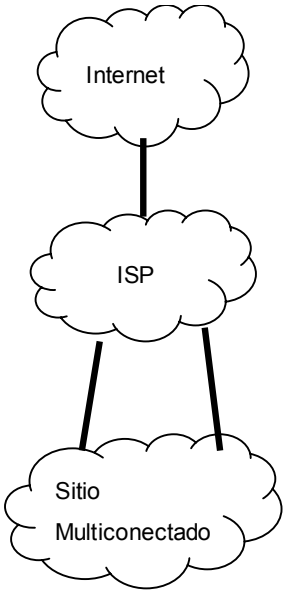

Figura 3: Sitio Multiconectado

2.3 Multihoming en la Internet con clases

Para conocer las motivaciones para las soluciones actuales, es conveniente adentrarse en la historia de Internet. En sus inicios, la arquitectura de direccionamiento de Internet estaba basada en clases de direcciones. Existían pues, tres clases de direcciones:

- Las direcciones de clase A, caracterizadas por tener el primer bit a 0, eran direcciones en las que los primeros 8 bits identificaban una red dentro de Internet y los últimos 24 bits identificaban una interfaz dentro de esta red. Existían 127 redes de clase A y cada una de esas redes podía tener un máximo de 16.777.214 direcciones.
- Las direcciones de clase B, con el primer bit a 1 y el siguiente bit a 0, eran direcciones en las que los primeros 16 bits identificaban una red dentro de Internet y los últimos 16 bits identificaban una interfaz dentro de la red. Existían 16.382 redes de clase B y cada una de esas redes podía tener un máximo de 65.536 direcciones.
- Las direcciones de clase C tenían el primer bit a 1, el segundo bit a 1 y el tercer bit a 0, y los primeros 24 bits identificaban una red dentro de Internet y los últimos 8 bits identificaban una interfaz dentro de la red. Existían 2.097.152 redes de clase A y cada una de esas redes podía tener un máximo de 256 direcciones.

En esta arquitectura de direccionamiento, cada nuevo sitio que deseaba conectarse a Internet solicitaba a la autoridad central, InterNIC en aquel momento, su propio rango de direccionamiento, ya fuera una clase A, B o C, y luego las anunciaba a sus vecinos a través del sistema de rutas de inter-dominio. De esta forma, cada nuevo sitio que se incorporara a Internet, contribuía con una entrada a la tabla global de rutas de Internet, independientemente del número de conexiones a Internet que tuviera. Por ende, no existían diferencias en el impacto en el sistema de rutas de la solución de conectividad para sitios multihomed y de la solución para sitios con un solo ISP, como se puede ver en la figura siguiente.

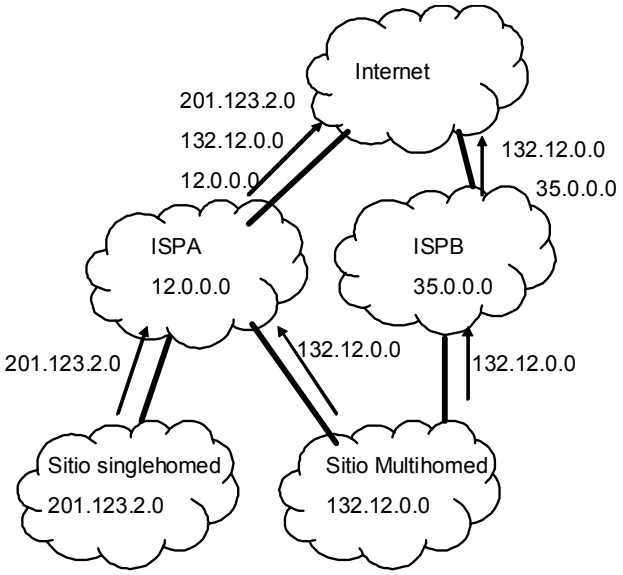

Figura 4: Multihoming en Internet con clases

2.4 Limitaciones de la arquitectura de direccionamiento de clases

A medida que fue pasando el tiempo y aumentó el número de sitios conectados a Internet, las limitaciones de la arquitectura de direccionamiento basada en clases se fueron haciendo cada vez más evidentes. Las limitaciones identificadas fueron fundamentalmente las siguientes:

Agotamiento del espacio de direcciones: Como hemos visto, los sitios que se conectaban a Internet podían obtener direcciones de clase A, B o C. Las clases A estaban reservadas para redes muy grandes, dado el elevando número de direcciones que contenían. Las clases C, por otra parte, eran poco deseadas debido a su limitado número de direcciones. Esto implicó una alta demanda

de clases B. Debe notarse que una clase B puede albergar hasta 65.536 nodos dentro de la red, un número considerablemente elevado y que era considerablemente mayor que el número de nodos contenidos en una buena parte de las redes existentes. Sin embargo, a falta de mejores opciones, las redes solicitaban y obtenían rangos de clase B. Todo esto implicó una alta tasa de consumo de direcciones IP, en particular de clases B, acompañado de una muy baja utilización del espacio de direcciones IP (del orden del 1% [Huston2003a]). Según la RFC 1519 [RFC1519], en febrero de 1992 se habían asignado 5.467 de las 16.382 clases B, quedando 10.915 todavía disponibles. Menos de un año mas tarde, en enero de 1993, se habían ya asignado 7.133 clases B. Si la tasa de crecimiento se mantenía, las clases B se agotarían en unos 15 meses. Como solución transitoria, a medida que las clases B fueron más escasas, los sitios comenzaron a solicitar múltiples clases C en lugar de una única clase B. Si bien este cambio mejoró la eficiencia en el uso del espacio de direcciones, ya que típicamente los sitios obtenían menos de las 255 clases C que equivaldrían a una clase B, a cambio agravó más aún el crecimiento desmedido de las tablas globales de rutas, como se presentará a continuación.

Crecimiento exponencial del tamaño de las tablas globales de rutas[2]: El tamaño en la tabla global de rutas[3] parecía crecer exponencialmente. Según la RFC 1519 [RFC1519], en enero de 1992, la tabla global de rutas del backbone de la NSF contenía unas 4.700 entradas. Datos históricos disponibles en ese momento mostraban que el tamaño de la tabla de rutas se había duplicado cada 10 meses en el período desde 1988 hasta 1991. Esto implicaba que las tablas llegarían a 30.000 entradas en menos de dos años. Si además se considera el efecto de la utilización de múltiples clases C en lugar de una clase B, se estimaba que las tablas se cuadruplicarían en lugar de duplicarse en el período considerado, implicando que las tablas de rutas llegarían a 20.000 entradas en menos de una año. La siguiente gráfica [Huston2001a] muestra el crecimiento exponencial de las tablas en el periodo considerado.

[2] Con tabla global de rutas nos referimos a lo que se conoce como *forwarding information base*, es decir la tabla que contiene UNA entrada para cada uno de los destinos conocidos por el nodo en cuestión. Adicionalmente, cada router posee una tabla de rutas BGP, la cual tiene todas las distintas rutas hacia un mismo destino a través de los distintos caminos conocidos por el protocolo.

[3] Téngase en cuenta que la tabla de rutas no es única, sino diversa para cada observador en la red, ya que en cada lugar está influida por las políticas aplicadas en los caminos desde cada originador del prefijo hasta el lugar de observación. No obstante, consideraremos para el análisis una observación típica o "media" de las ocurrencias de esta tabla.

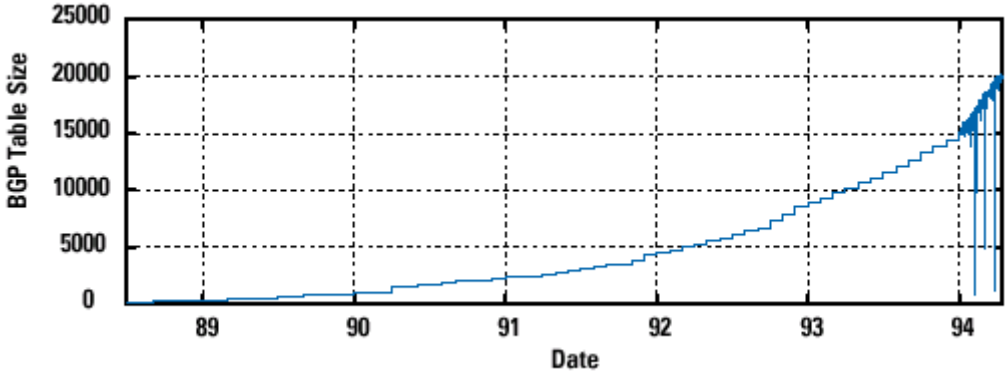

Figura 5: Crecimiento pre-CIDR (Figura publicada en [Huston2001a])

Este crecimiento se explica fácilmente ya que, como se ha presentado anteriormente, en el esquema de direccionamiento por clases, cada sitio contribuye con al menos una entrada en la tabla global de rutas, dependiendo del número de clases que haya obtenido. El número de rutas presentes en la tabla global de rutas puede estimarse de la siguiente forma:

$$N = \sum_{i}^{T} p_i$$

N número total de rutas en la tabla global de rutas

p_i número de clases asignadas al sitio i

T número total de sitios en Internet

Si dividimos los sitios presentes en Internet en sitios finales e ISPs, podemos decir que:

$$T = P + S$$

S número total de sitios finales

P número de ISPs

El número total de rutas en la tabla global de rutas queda:

$$N = \sum_{i}^{S} p_i + \sum_{j=S+1}^{S+T} p_j$$

Resulta entonces que podemos estimar el número total de rutas en la tabla global como:

$$N \approx p_S S + p_P P$$

p_S número medio de clases por sitio final

p_P número medio de clases por ISP

Considerando que cada sitio, ya sea sitio final o ISP, obtiene una clase o más, podemos concluir que el número de entradas en la tabla global de rutas de la Internet con clases crece al menos linealmente (con pendiente al menos igual a 1) con el número de sitios en Internet, en particular con el número de sitios finales y el número de ISPs. El agotamiento de las clases B, simplemente aumentó el número medio de clases por sitio, p_S y p_P, ya que si se utilizan clases B, un sitio requiere sólo una clase, es decir que $p_S = 1$ y $p_P = 1$, mientras que si se utilizan clases C, un sitio requiere múltiples clases, entre 4 y 16 de media según [RFC1519], por lo que $p_S \geq 1$ y $p_P \geq 1$.

$$N \propto (S, P)$$

Siendo:

N número total de rutas en la tabla global de rutas

S número total de sitios finales

P número de ISPs

Esta tasa de crecimiento era mayor de lo que podía aceptar Internet en ese momento (y probablemente también sea inaceptable para la arquitectura actual) ya que los routers disponibles no serían capaces de soportar esa tasa de crecimiento durante mucho tiempo ni en memoria requerida para almacenar los datos de encaminamiento, ni en ancho de banda para intercambiarlos, ni finalmente en capacidad de proceso para actualizarlos cuando hubiera cambios.

Una vez detectado el problema, el Internet Engineering Task Force (IETF) exploró diversas alternativas para resolver el problema a largo plazo y también para paliar sus efectos a corto plazo, ya que el peligro era inminente. Una de las estrategias adoptadas fue un cambio en la arquitectura de direccionamiento y en la forma de asignar los bloques de direcciones llamado Classless InterDomain Routing (CIDR), que presentaremos a continuación.

2.5 Classless InterDomain Routing (CIDR)

En la arquitectura de direccionamiento basada en clases, las clases de direcciones se asignaban a los sitios de forma arbitraria, sin consideraciones adicionales. El único nivel de abstracción existente era el provisto por el propio sistema de clases, que permitía tener una sola ruta en las tablas de encaminamiento indicando el próximo salto a seguir para alcanzar cualquiera de las direcciones pertenecientes a la clase en cuestión. La información referente a cuáles son las direcciones contenidas en una clase dada (o dada una dirección, determinar la clase a la cual pertenece) se encuentra codificada en las propias direcciones, en particular, en los primeros 3 bits. El resultado, como hemos visto, era que cada una de las clases debía incluirse en la tabla global de rutas para asegurar su alcanzabilidad global, implicando que cada sitio contribuía con una o más entradas en la tabla de rutas. La respuesta ofrecida por CIDR fue aumentar sustancialmente la capacidad de abstracción de la información de encaminamiento, para incrementar significativamente el número direcciones que eran alcanzables a través de una entrada en la tabla de rutas. En el sistema de clases dicho número venía determinado por las direcciones contenidas en la clase correspondiente, mientras que en CIDR ese número viene determinado por la topología de delegaciones, es decir por el número de direcciones que comparten un camino común en la topología.

Originalmente propuesto en la RFC 1338 [RFC1338] bajo el nombre de "*supernetting*" y luego modificado en la RFC 1519 [RFC1519] con el nombre de CIDR, y complementado por la RFC 1518 [RFC1518], CIDR consta de esencialmente dos elementos:

- un esquema alternativo de asignación de direcciones y
- un mecanismo para agregar información de encaminamiento.

En la arquitectura de direccionamiento con clases, los conjuntos de direcciones a los que hacía referencia una ruta estaban determinados en la dirección en sí misma, ya que cada ruta hacía referencia a la alcanzabilidad de todas las direcciones contenidas en la clase. Por ello, los protocolos de rutas solamente comunicaban una dirección, la dirección de la red, simbolizando así que la ruta era válida para acceder al conjunto completo de direcciones contenidas en la clase en cuestión.

Dado que las facilidades de abstracción ofrecidas por dicho esquema resultaban insuficientes para las necesidades de Internet, se adoptó la agregación de direcciones basada en la utilización de máscaras de red. En un esquema basado en máscaras, un par (dirección, mascara) hace referencia a un conjunto de direcciones que contiene a todas las direcciones que comparten la misma identificación de red. La parte de red de una dirección surge de hacer el AND lógico entre la dirección en cuestión y la máscara. Por ende, el conjunto de direcciones queda determinado por:

$$(a.b.c.d, w.x.y.z) = \{\forall e.f.g.h / [(e.f.g.h \wedge w.x.y.z) = (a.b.c.d \wedge w.x.y.z)]\}$$

Siendo:

$a.b.c.d$ una dirección del conjunto en cuestión

$w.x.y.z$ la máscara de red, interpretable en binario, con la propiedad de que si se empieza por la izquierda contiene un continuo de unos y luego un continuo de ceros, dejando fijos los bits más significativos y permitiendo variaciones dentro del conjunto de direcciones en los bits menos significativos.

De esta forma, en CIDR no basta con indicar una dirección para saber el conjunto al que pertenece, sino que es necesario también incluir la máscara de red para determinarlo. Esto obligó a cambiar los protocolos de rutas, ya que desde la adopción de CIDR, un destino queda especificado por un par dirección / máscara.

Esta notación, sin embargo, nos permite tener límites de abstracción variables, de forma que se puedan agregar direcciones de forma mucho más agresiva, a diferencia de la arquitectura de clases, donde los límites de agregación estaban predeterminados.

Con las máscaras que nos permiten definir los agregados de direcciones de una forma más flexible es posible mejorar los niveles de abstracción en el direccionamiento. En primera instancia, es posible expresar todo el bloque de direcciones asignado a un sitio a través de una única expresión, utilizando las máscaras, lo que redunda en la capacidad de poder expresar la alcanzabilidad hacia el sitio en cuestión a través de una única ruta en la tabla de rutas.

Como vimos anteriormente, al no haber más clases B disponibles, los sitios comenzaron a solicitar múltiples clases C. Esto implicaba que un sitio que obtenía N clases C impactaría con N entradas en la tabla global de rutas, ya que no existía forma de expresar el conjunto de direcciones asignado al sitio de forma más compacta. Una vez que las máscaras se encuentran disponibles, se propone la asignación de múltiples clases C contiguas, para poder expresar el conjunto de direcciones como una única expresión de la forma “dirección, máscara” (nótese que esto sólo es posible si el numero de clases C asignadas es una potencia de dos; en otro caso, será necesaria más de una expresión para expresar el conjunto asignado).

Supongamos que un sitio solicita 4 clases C: En el esquema de clases obtendría 4 clases C arbitrarias, por ejemplo, 193.1.2.0, 193.2.3.0, 194.45.32.0 y 197.12.32.0. Luego, tanto el sitio como su proveedor anunciaría las cuatro clases C, por lo que se crearían 4 entradas nuevas en la tabla global de rutas. En CIDR, el sitio obtendría 4 clases C contiguas, por ejemplo, 193.23.0.0, 193.23.1.0, 193.23.2.0 y 193.23.3.0, que pueden ser expresadas como 193.23.0.0 máscara

255.255.252.0, por lo que tanto el sitio como su proveedor inyectarán una única ruta hacia el agregado, creando así una única entrada en la tabla de rutas.

Adicionalmente, CIDR propone un nuevo esquema de asignación de direcciones que permite una agregación más poderosa aún. La propuesta es pasar de un modelo donde la asignación de direcciones es realizada por una autoridad central, sin otro criterio que no sea el tamaño del bloque a asignar, a un modelo distribuido, donde los proveedores obtienen grandes bloques de direcciones, y estos a su vez se los asignan a sus propios clientes. De esta forma, cada proveedor puede anunciar un solo bloque agregado de direcciones que contiene todos los bloques de direcciones que poseen sus clientes.

Este esquema de asignación de direcciones tiene un sentido topológico, ya que refleja la estructura jerárquica que existe en la topología donde un proveedor brinda acceso al resto de la red a sus clientes. El sentido intuitivo de este esquema esta basado en que todos los clientes del mismo proveedor comparten una misma ruta desde/hacia Internet, la cual pasa por el proveedor, de forma que obteniendo las direcciones del bloque del proveedor, es posible que el proveedor anuncie una sola ruta hacia el agregado a través de él mismo. Las direcciones asignadas de esta forma son llamadas *Agregables por Proveedor (PA, Provider Aggregatable)* y por oposición, las direcciones sin sentido topológico son llamadas *Independientes de Proveedor (PI, Provider Indenpendent)*.

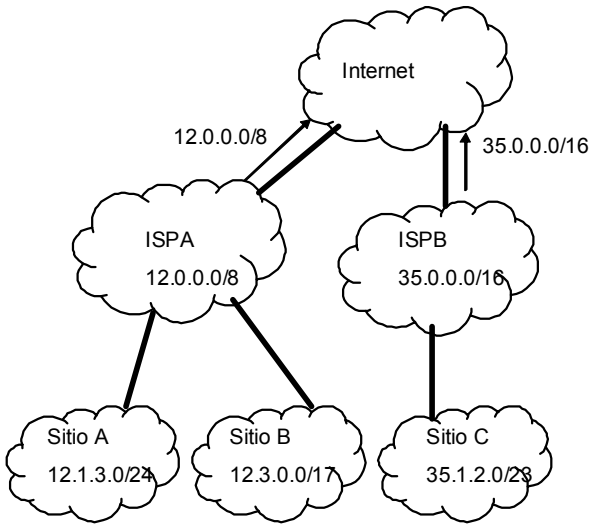

Figura 6: Asignación de direcciones PA

El presentado es un modelo jerárquico de agregación, por lo que los beneficios obtenidos por la agregación obtenida crecen exponencialmente a medida que bajamos de nivel. Es decir, que el mayor grado de agregación se obtiene cuando los sitios finales anuncian todas sus direcciones a través de un agregado. A continuación, la segunda mayor ganancia obtenida por la agregación es

conseguida cuando los proveedores directos anuncian un solo agregado que contiene todos sus sitios finales clientes. El próximo nivel de agregación se conseguiría cuando el proveedor indirecto agrega todos los bloques de los proveedores directos a los que brinda servicio en un solo agregado. En cierto punto, los beneficios de agregación obtenidos no justifican las dificultades adicionales que conllevan (por ejemplo en lo que se refiere a la dependencia del direccionamiento de proveedor de nivel superior), por lo que en alguna parte de la jerarquía, se deja de agregar en función de los proveedores.

En conclusión, la recomendación realizada fue adoptar el esquema de agregación topológica por proveedor en los dos niveles más bajos de la jerarquía, es decir agregar todas las direcciones de un sitio final y agregar los bloques de los sitios finales servidos por un mismo proveedor en el bloque del proveedor en cuestión. Esto reportaría los mayores beneficios de la agregación.

Como resultado de CIDR, el crecimiento del tamaño de la tabla de rutas globales se contuvo en los años posteriores a su adopción. Es más, entre los años 1994 y 1996, el tamaño de la tabla de rutas incluso disminuyó por momentos, como puede verse en la figura siguiente ([Huston2001a])

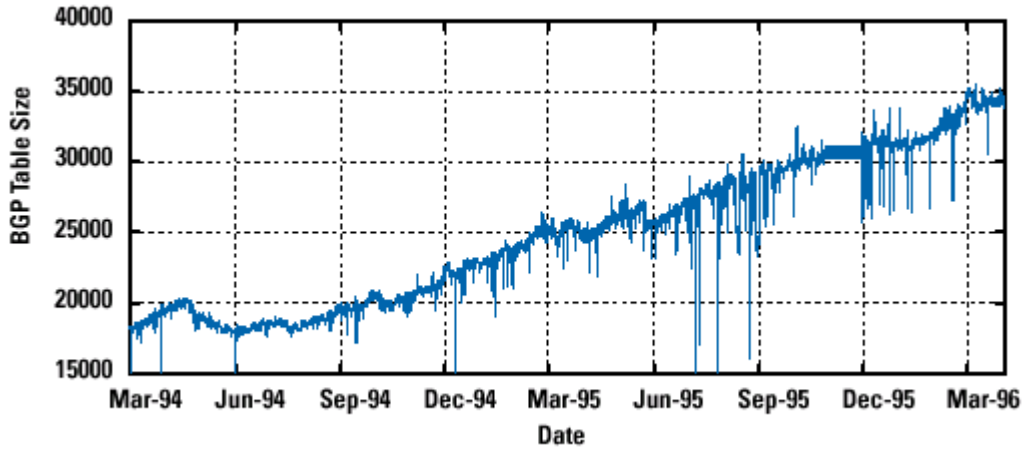

Figura 7: Adopción de CIDR (Figura publicada en [Huston2001a])

Algunos de los efectos observados se pueden explicar por las renumeraciones masivas realizadas para alinear el esquema de numeración a la topología jerárquica de la red.

En los años posteriores, el crecimiento con CIDR pasó a ser esencialmente lineal, como puede verse en la figura a continuación ([Huston2001a])

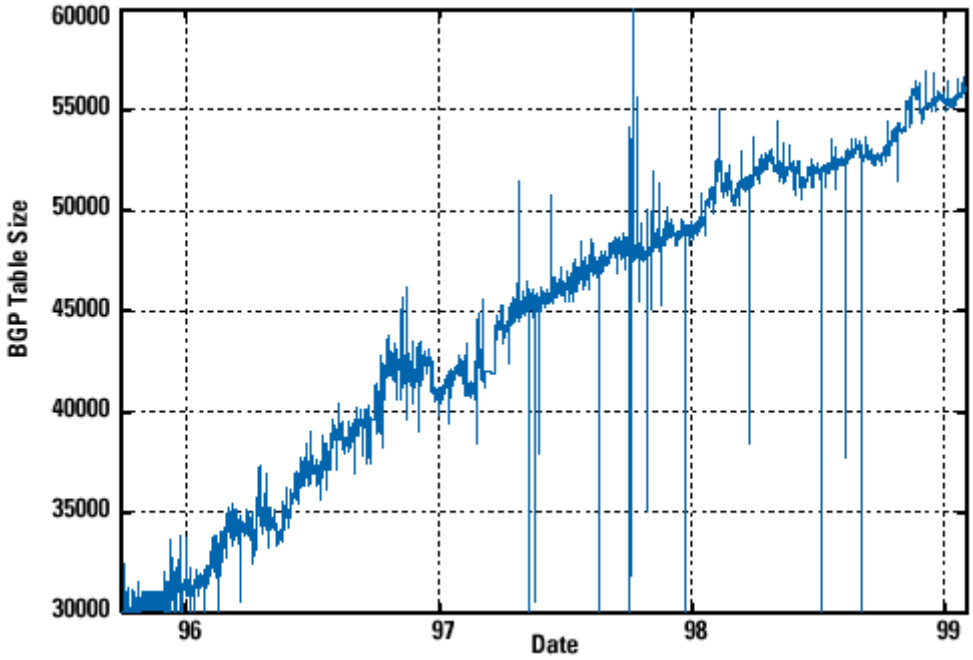

Figura 8: Crecimiento post-CIDR (Figura publicada en [Huston2001a])

Por lo tanto, como resultado de la adopción de CIDR, el crecimiento pasó de ser exponencial, antes de la adopción de CIDR, a ser prácticamente lineal, después de CIDR, con una tasa de crecimiento de unas 10.000 entradas por año, por lo que podemos decir que la estrategia de CIDR fue exitosa para contener el crecimiento de las tablas de rutas, al menos durante algunos años.

2.6 Multihoming en CIDR

Como hemos presentado, en CIDR los sitios finales obtienen un rango de direcciones de su proveedor, de forma tal que el proveedor sólo anuncia el agregado de direcciones, reduciendo así el tamaño de las tablas globales de rutas. En el caso de los sitios multihomed, estos tienen múltiples proveedores, por lo que obtendrán direcciones de uno de ellos. Sin embargo, en el esquema de agregación por proveedor, si las direcciones pertenecen al agregado de uno de los proveedores, éstas serán solamente alcanzables desde este proveedor, ya que sólo éste anuncia el agregado en el sistema de rutas interdominio. Por esto, es necesario que los otros proveedores del sitio multihomed anuncien también el rango de direcciones del sitio en el sistema de rutas, de forma que el sitio sea alcanzable a través de todos sus ISPs. El problema de este esquema es que el rango de direcciones del sitio multihomed pertenece al agregado de uno solo de los ISPs, y por ende, no al agregado de los otros ISPs. Como consecuencia estos últimos deberán anunciar una ruta específica al rango de direcciones del sitio multihomed, añadiendo así una ruta adicional a la

tabla global de rutas. Además, el propio ISP que ha delegado el rango de direcciones al sitio multihomed deberá anunciar una ruta más específica al sitio multihomed, ya que si no las rutas de los otros ISPs serían preferidas debido a la regla del longest prefix match (los otros proveedores anuncian una ruta más específica que el agregado del ISP).

La figura 9 ilustra los anuncios de rutas involucrados en una configuración multihomed con CIDR. En este caso, el sitio multihomed debe anunciar sus rutas a todos sus proveedores usando el protocolo de encaminamiento interdominio. La configuración de este protocolo determinará diversos aspectos del servicio obtenido por el sitio multihomed, por lo que a continuación presentaremos los aspectos más relevantes del mismo.

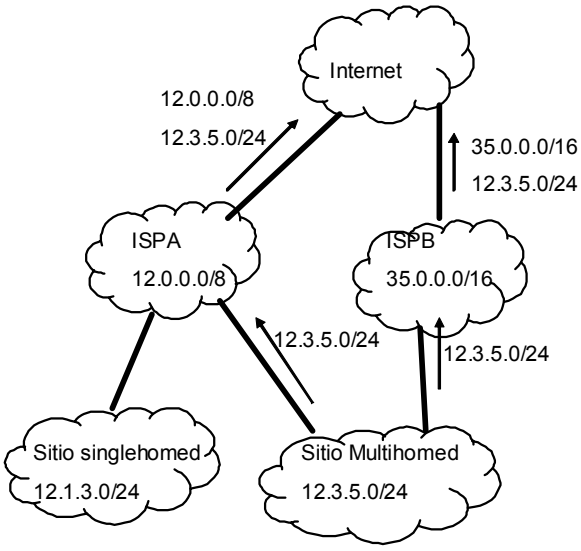

Figura 9: Multihoming con CIDR

2.6.1 Border Gateway Protocol

El *Border Gateway Protocol* en su versión 4 *(BGP-4),* definido en la RFC 1771 [RFC1771], es el protocolo utilizado para intercambiar información de encaminamiento entre *Sistemas Autónomos (AS, Autonomous Systems)*. Un AS es un conjunto de máquinas que pertenecen al mismo dominio administrativo, en el sentido de que comparten la misma política de encaminamiento. Es decir que todo el tráfico concerniente a dichas máquinas se encamina siguiendo las mismas políticas administrativas. Por ejemplo, un sitio que tiene una sola conexión a Internet no tiene una política propia de encaminamiento interdominio, ya que tiene una única opción de encaminamiento que es enviar y recibir todo el tráfico por el único enlace del que dispone. Sin embargo un sitio que tiene múltiple proveedores, puede elegir por cuál de ellos desea

encaminar los paquetes, definiendo así una política propia de encaminamiento interdominio, independiente de la de sus proveedores. El sitio multihomed es típicamente un AS y cada uno de los proveedores son ASs distintos.

BGP es un algoritmo de vector de caminos, de forma que cada AS informa a los demás de qué destinos son alcanzables a través de él mismo, incluyendo en dicho anuncio los ASs del trayecto por los que se ha transmitido el anuncio de este destino. El trayecto de ASs se utiliza para detectar bucles y también puede ser usado como método de selección entre múltiples rutas alternativas.

El protocolo BGP es utilizado por los routers de borde de dos ASs para intercambiar información de rutas. El protocolo BGP utiliza TCP como transporte para el intercambio de rutas, por lo que es necesario que una conexión TCP se establezca entre ambos routers. Una vez establecida, los routers intercambian los mensajes de inicialización de sesión BGP, en los que se definen los parámetros relevantes de la sesión en cuestión. Una vez establecida la sesión BGP, los routers utilizan mensajes `UPDATE` para intercambiar toda la información que quieren transmitir a ese vecino (un subconjunto de la tabla de rutas de cada router, filtrada según consideraciones de política de encaminamiento). Una vez que han intercambiado las tablas BGP, sólo se intercambian modificaciones a dichas tablas, manteniéndolas actualizadas y coherentes con la topología. Cuando no hay información topológica para intercambiar, los routers intercambian periódicamente mensajes `KEEPALIVE` para verificar que la sesión continúa establecida y que no hay un fallo en la comunicación o en las instancias que ejecutan el protocolo.

Adicionalmente, el protocolo BGP es también utilizado para intercambiar información de rutas exteriores entre los distintos routers de un mismo AS. A este uso del protocolo se le llama *I-BGP (Interior BGP)* mientras que al uso descrito en el párrafo anterior se le conoce como *E-BGP (Exterior BGP)*. Como el mecanismo de detección de bucles de BGP se basa en el uso de la lista de todos los ASs distintos por los que ha pasado un anuncio de ruta (el trayecto de ASs), este mecanismo no es efectivo cuando dos routers del mismo AS intercambian información. Para solucionar este inconveniente, los routers de un mismo AS que intercambien rutas usando I-BGP deberán hacerlo formando un malla completa y una información recibida de I-BGP no se reenviará a otros routers con los que también se utiliza I-BGP.

2.6.1.1 Atributos

Los mensajes de `UPDATE`, además de la información referente al destino de la ruta, es decir dirección IP y máscara, contienen información adicional en forma de *atributos*. Los atributos tienen un formato compuesto por `Tipo`, `Longitud` y `Valor` (codificación `TLV`). El tipo de

atributo consiste en 1 byte de `código de tipo de atributo` y 1 byte de `indicadores (flags)`, que especifican propiedades genéricas del atributo.

El primer bit del byte de `indicadores` define si el atributo es *opcional* o *bien conocido.* Los atributos bien conocidos deben ser soportados por todos los interlocutores BGP. Adicionalmente, hay ciertos atributos bien conocidos que son *obligatorios*, es decir que deben ser incluidos en todos los mensajes `UPDATE`.

El segundo bit del byte de `indicadores` define si el atributo es *transitivo* o *no transitivo*. Si un atributo es transitivo, este puede (y en el caso que sea obligatorio, debe) ser retransmitido cuando la ruta en cuestión sea informada a otro interlocutor BGP.

El tercer bit del byte de `indicadores` define si la información incluida en el atributo es *completa* o *incompleta*, permitiendo reconocer si un atributo opcional y transitivo ha sido procesado apropiadamente por todos los routers intermedios entre el que lo originó y el aquí considerado.

El cuarto bit del byte de `indicadores` define si el campo de `largo` del atributo es uno o dos bytes.

A continuación se describen los atributos más importantes:

`ORIGIN`: es un atributo bien conocido y obligatorio, que contiene información referente al mecanismo por el cual se obtuvo la información en el AS donde fue generada. Las posibilidades son "`IGP`", "`EGP`" o "`INCOMPLETE`".

`AS_PATH`: es un atributo bien conocido y obligatorio, que contiene la secuencia de AS a través de la cual se ha transmitido la ruta. Este atributo es utilizado por BGP para evitar bucles, además de ofrecer información muy valiosa para aplicar políticas.

`NEXT_HOP`: es un atributo bien conocido y obligatorio que contiene la dirección IP del router al que deben enviarse los paquetes para alcanzar el prefijo anunciado.

`MULTI_EXIT_DISC (MED)`: es un atributo opcional y no transitivo que se utiliza para influenciar la selección de camino cuando existe más de un camino entre dos ASs.

`LOCAL_PREF`: es un atributo bien conocido utilizado en I-BGP para informar sobre la preferencia entre múltiples rutas disponibles para un mismo prefijo.

`ATOMIC_AGGREGATE`: es un atributo bien conocido que informa de un suceso de agregación de la información de rutas a lo largo del camino, por lo que se ha eliminado parte de la información en el `AS_PATH`.

`AGGREGATOR`: es un atributo opcional y transitivo usado para identificar el AS que ha realizado la agregación de la información de ruta contenida.

`COMMUNITIES`: es un atributo opcional y transitivo definido posteriormente en la RFC 1997 [RFC1997] que se usa para marcar la información de rutas para su posterior procesamiento.

2.6.1.2 Selección de rutas en BGP

Cuando existen múltiples rutas para un mismo prefijo, BGP debe elegir una de ellas para incluirla en la tabla de rutas que efectivamente será utilizada por el router. Para ello se utilizan una serie de reglas que describiremos a continuación. Las reglas siguientes son aplicadas en orden y solamente hasta que una de ellas seleccione una ruta sobre otra.

1. Preferir la ruta con mayor `LOCAL_PREF`
2. Preferir la ruta con `AS_PATH` más corto
3. Preferir la ruta según el `ORIGIN`, prefiriendo primero `IGP`, luego `BGP` y luego `INCOMPLETE`
4. Preferir la ruta con menor `MED`
5. Preferir la ruta aprendida a través de E-BGP sobre una ruta aprendida a través de I-BGP
6. Preferir la ruta para la cual la distancia IGP hasta el `NEXT_HOP` sea menor
7. Preferir la ruta obtenida a través de un router con menor dirección IP

Cabe notar que algunas de las reglas arriba detalladas no están recogidas en la RFC 1771 que define BGP, pero sin embargo son habitualmente usadas en las implementaciones comerciales.

2.6.2 Configuración de un sitio multihomed

Como hemos visto en la figura anterior, un sitio multihomed inyecta información de encaminamiento a todos sus proveedores. Para ello utiliza el protocolo BGP. Adicionalmente, el sitio puede recibir información de rutas de los proveedores, para recibir información sobre cómo encaminar su tráfico. A continuación describiremos diversas configuraciones usadas para obtener distintos patrones de tráfico en sitios multihomed.

2.6.2.1 Tolerancia a fallos

Cuando el sitio multihomed inyecta una ruta hacia su propio prefijo a través de todos sus proveedores, este sitio es alcanzable a través de los mismos. De esta forma si ocurre un fallo en el camino a través de uno de ellos, se podrá utilizar un camino alternativo a través de otro proveedor. Todo esto funciona de forma automática gracias a BGP, ya que los sitios remotos recibirán los anuncios correspondientes a las distintas rutas hacia el sitio multihomed y elegirán la que más les conviene, basándose en criterios locales. Cuando ocurra algún fallo, la ruta afectada será retirada del sistema de rutas de BGP, por lo que ya no se utilizará más y se encaminarán los paquetes a través de las rutas alternativas que persistan.

Adicionalmente, esta configuración también provee tolerancia a fallos para el tráfico saliente del sitio multihomed. Esto se logra gracias a que los proveedores envían información de rutas al sitio multihomed. En la configuración más sencilla, los proveedores sólo anuncian una ruta por defecto al sitio multihomed. De esta forma, el sitio multihomed sabe si puede alcanzar al resto de destinos a través del proveedor. Cuando el sitio deja de recibir la ruta por defecto desde un proveedor, esto quiere decir que ha ocurrido un fallo y que el resto de destinos no son alcanzables a través de dicho proveedor. En ese caso, el sitio multihomed encaminará los paquetes a través de un proveedor alternativo del que sí que reciba una ruta por defecto.

La configuración anterior asume que los fallos son totales y que, o bien todos los destinos son alcanzables a través de un ISP, o ninguno lo es. En realidad, los fallos pueden afectar parcialmente la alcanzabilidad de los destinos a través de un proveedor, haciendo que sólo algunos destinos no sean alcanzables cuando hay un fallo. Es más, es posible que en caso de un fallo, ciertos destinos sean sólo alcanzables a través de un proveedor y el resto de destinos sea solamente alcanzable por el otro ISP. Un ejemplo de esta situación ocurre cuando el enlace de uno de los proveedores con el resto de Internet falla. En este caso, los clientes del ISP cuyo enlace con Internet ha fallado son solamente accesibles a través de dicho ISP, mientras que el resto de destinos son solamente alcanzables a través del otro ISP. Por lo tanto para obtener un mejor soporte de tolerancia a fallos, es posible recibir información de rutas más detallada de los proveedores. El caso más extremo es recibir de cada uno de los proveedores la tabla BGP completa, de forma tal que podamos saber cuáles son los destinos alcanzables por cada proveedor en cada instante, obteniendo así alcanzabilidad óptima. Sin embargo, hay que tener en cuenta que las tablas de BGP pueden ser excesivamente grandes, por lo que su almacenamiento y procesamiento puede requerir excesivos recursos en los routers involucrados. Para paliar esta situación, y si existe una ruta por defecto, es posible filtrar la información intercambiada (puede ser filtrada por el ISP, en cuyo caso no sería enviada al cliente, o bien puede ser filtrada por el cliente mismo, descartándola cuando la recibe del ISP). Por ejemplo, es posible filtrar la información recibida en función de la longitud del `AS_PATH`, como se sugiere en los manuales de BGP [Beijnum2002a]. Por ejemplo podemos descartar cualquier ruta cuyo `AS_PATH` contenga más de 8 ASs, asumiendo que es más factible que los fallos en enlaces más cercanos afecten más severamente al sitio.

2.6.2.2 Balanceo de carga

Además de tolerancia a fallos, esta configuración brinda capacidades de *balanceo de carga*, es decir que es posible distribuir el flujo de paquetes, tanto entrante como saliente, entre los distintos enlaces. Para el tráfico entrante, esto se logra gracias a la inyección mediante BGP de las rutas hacia el sitio a través de los múltiples enlaces hacia el resto del sistema de encaminamiento.

Los potenciales sitios fuente de mensajes recibirán las múltiples rutas y elegirán una de ellas, siguiendo criterios locales, por ejemplo basándose en la longitud del AS_PATH. El resultado será entonces que los paquetes serán encaminados por distintos caminos, balanceando el tráfico entre los distintos enlaces.

Para el tráfico saliente, el balanceo de carga se logra configurando el sitio para que distribuya el tráfico saliente entre los distintos proveedores. Esto se puede lograr de distintas formas, por ejemplo eligiendo distintas rutas por defecto asociadas a distintos proveedores en distintas subredes del sitio. Otra forma es encaminar el tráfico destinado a ciertas direcciones a través de un proveedor y el tráfico destinado a otras direcciones a través de otro ISP.

2.6.2.3 Configuración de políticas de encaminamiento

Un sitio multihomed puede preferir encaminar tráfico a través de un proveedor dado debido a políticas administrativas, por ejemplo por razones de seguridad o calidad de servicio. La configuración multihomed basada en BGP ofrece ciertas facilidades para implementar dichas políticas.

Para el tráfico entrante, es posible utilizar una técnica basada en el aumento artificial de la longitud del AS_PATH, por ejemplo incluyendo repetidas veces el AS propio en el AS_PATH. De esta forma la ruta inyectada por uno de los proveedores tendrá inicialmente un AS_PATH más largo que la ruta inyectada a través del otro proveedor. El resultado de esta técnica, es que la ruta con el AS_PATH más largo es menos atractiva, por lo que será menos usada, transfiriendo el tráfico hacia el otro ISP. Sin embargo debe notarse que la elección final de qué camino será usado para encaminar los paquetes no reside en el sitio multihomed, ya que las políticas locales en los sitios e ISPs remotos pueden preferir una ruta dada a pesar que tenga un AS_PATH más largo. Por ello, la solución de multihoming basada en BGP sólo permite influenciar el camino que seguirá el tráfico entrante, pero no permite forzar su camino. Adicionalmente, es posible anunciar rutas más especificas a través de uno de los proveedores, de forma que está ruta sea preferida sobre la ruta menos específica anunciada por el resto de proveedores. Esta técnica resulta en la desagregación artificial de la información de rutas con el fin de influenciar el encaminamiento.

Para el tráfico saliente, el sitio multihomed puede configurar sus routers de borde de modo que se prefieran las rutas recibidas por un ISP sobre las de otro. Esto se logra marcando las rutas con un LOCAL_PREF mayor. De esta forma, los routers incluirán en sus tablas de encaminamiento la ruta proveniente del ISP preferido, determinando así que los paquetes para el destino en cuestión serán encaminados por el mismo cuando éste se encuentre disponible. En este caso, el sitio multihomed sí puede forzar que los paquetes sean encaminados por el ISP seleccionado por su política.

2.7 Limitaciones en las capacidades de agregación de CIDR

A partir del año 1998 la tasa de crecimiento de la tabla global de rutas se tornó nuevamente exponencial como se puede ver en la figura siguiente tomada del estudio presentado en el IPJ [Huston2001a]

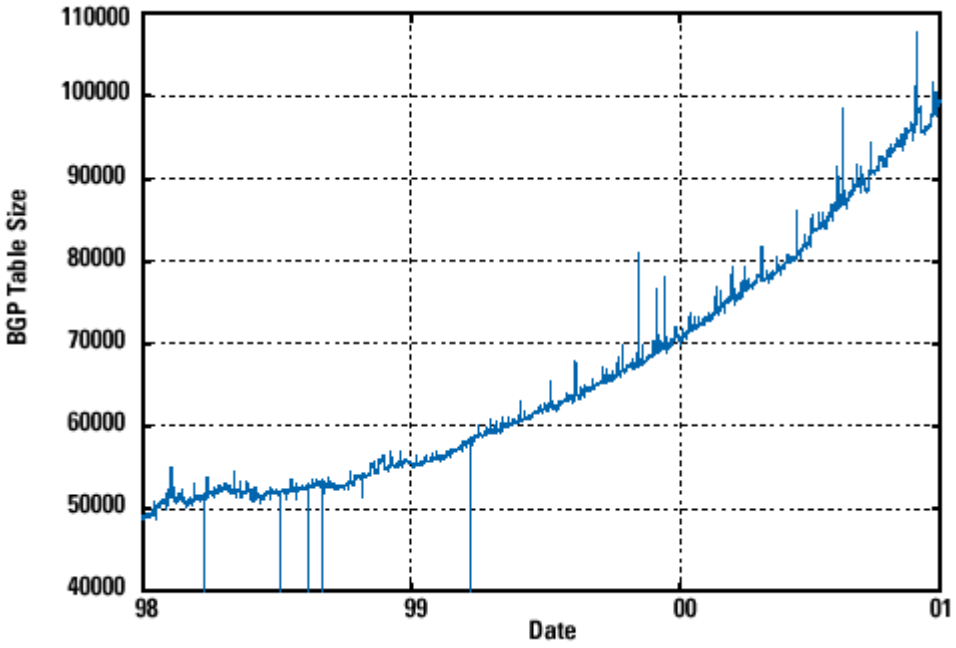

Figura 10: Crecimiento exponencial post-CIDR (Figura publicada en [Huston2001a])

Se han identificado los siguientes factores que contribuyen a dicho crecimiento:

- Agujeros en los agregados
- Sitios multihomed
- Políticas de encaminamiento

A continuación describiremos cómo afectan a la agregación en CIDR cada uno de los factores identificados.

2.7.1 Agujeros en los agregados

La agregación en CIDR se basa en que los clientes de un proveedor obtienen direcciones del bloque asignado al ISP de forma que el ISP sólo anuncia un agregado de direcciones en el sistema de rutas interdominio, en lugar de anunciar múltiples rutas, una por cliente. Por ende, para

preservar las capacidades de agregación del sistema, cada vez que un cliente cambia de proveedor también deberá cambiar de direcciones, pasando de las direcciones asignadas por el proveedor antiguo a las direcciones asignadas por el proveedor nuevo. En caso de que esto no ocurra y que el cliente se lleve consigo las direcciones, se genera lo que se llama un *agujero* en el bloque de direcciones del proveedor antiguo, ya que existe un subrango de direcciones del bloque correspondiente al proveedor que ya no le pertenece y que se deberá encaminar a través de otra ruta. Para ello es necesario, al menos, que el nuevo proveedor anuncie una ruta más específica, correspondiente al rango de direcciones que el cliente se ha llevado consigo. Nótese que este rango no puede agregarse dentro del bloque de direcciones del proveedor nuevo. Adicionalmente, es posible que el proveedor antiguo desee desagregar su rango de direcciones para así no ofrecer una ruta para el rango de su antiguo cliente. Cabe notar que si el proveedor antiguo continúa anunciando todo su rango, incluyendo las direcciones correspondientes a su antiguo cliente, en caso de que el nuevo proveedor deje de anunciar la ruta (por ejemplo por un fallo), todos los paquetes dirigidos al antiguo cliente serán encaminados al proveedor antiguo, consumiendo así ancho de banda de éste.

Las motivaciones del sitio para preservar sus direcciones son claras: el proceso de renumerar una red IP es costoso y proclive a errores. Debe notarse que dicho proceso no sólo se refiere a cambiar las direcciones IP de los ordenadores, routers y otros dispositivos conectados a la red. El proceso también afecta a todas aquellas configuraciones que utilicen direcciones IP, entre las que podemos encontrar el servidor DNS (local y de nivel superior), los cortafuegos, los filtros en los routers y las listas de control de acceso (ACLs), entre otros. Además es posible que sea necesario modificar la configuración de las aplicaciones (o incluso las propias aplicaciones), ya que muchas de ellas, sobre todo las de misión crítica, no utilizan nombres de dominio para hacer referencia a otra máquina, sino que hacen referencia directamente a su dirección IP con el objetivo de mejorar la velocidad de respuesta y la fiabilidad, omitiendo la consulta al DNS.

Diversos esfuerzos se han realizado para simplificar el proceso de renumeración. En particular, el uso de configuraciones dinámicas utilizando DHCP, tanto para IPv4 [RFC2131] como para IPv6 [RFC3315] simplifica notablemente el proceso de renumeración de los ordenadores en una red. Adicionalmente, en IPv6 se ha hecho especial énfasis en las facilidades de renumeración automática, lo que se ve reflejado en las capacidades disponibles de auto-configuración sin estado [RFC2462], que permiten tanto la auto configuración automática de la dirección IP como del router por defecto. Adicionalmente, se ha desarrollado una herramienta de renumeración de routers en IPv6 [RFC2894] que permite la renumeración de los router conectados en una red. Es más, nuevas opciones de delegación de prefijo utilizando DHCP para IPv6 [RFC3633] permiten variar automáticamente el prefijo utilizado por una red. Sin embargo, el proceso de renumeración todavía impone un esfuerzo de trabajo manual, especialmente en sitios grandes, para reconfigurar aplicaciones, routers y cortafuegos, como ya hemos mencionado,

por lo que los sitios siguen evitando la renumeración en la medida de lo posible. La consecuencia de esto es que la agregación se ve dañada. Para intentar remediarlo, las políticas de asignación de direcciones establecidas por los Regional Internet Registries (RIRs) definen que las direcciones IP se prestan (o alquilan) pero no se venden, por lo que un sitio debe devolverlas cuando ya no compra más el servicio de un proveedor. Sin embargo, la realidad es que esto depende esencialmente de acuerdos comerciales entre clientes y proveedores (antiguos y nuevos) y que algunos clientes preferirán alcanzar un acuerdo comercial que les permita llevarse sus direcciones antes de incurrir en el costo que implica una renumeración.

Todo esto nos lleva a concluir que cada uno de los sitios que cambie de proveedor y no renumere contribuirá al incremento del tamaño de las tablas globales de rutas en al menos una entrada y potencialmente algunas más. Para reducir este factor sería necesario proveer las herramientas necesarias que permitan renumerar de forma menos costosa.

2.7.2 Sitios multihomed

CIDR garantiza la eficiencia en la agregación siempre y cuando la topología de la red esté relacionada con la estrategia de asignación de direcciones. CIDR en particular, está diseñado para ofrecer una agregación máxima cuando la topología subyacente es un árbol, con los ISPs como nodos y los sitios finales como hojas. El problema se presenta cuando la topología real de la red se distancia de la topología de árbol. En particular, los sitios multihomed son una excepción a la topología de árbol, dando lugar a grafos cíclicos. Adicionalmente, dado que su objetivo es mejorar la alcanzabilidad global del sitio, la información referente a dicha excepción debe ser difundida a toda red. Como hemos presentado anteriormente, dicha información es difundida globalmente a través de la inyección de rutas específicas hacia el prefijo del sitio multihomed a través de BGP. Esta limitación en las capacidades de agregación de BGP era conocida cuando CIDR fue diseñado, como se expresa en la RFC 1519 [RFC1519]: “Since under this plan, multi-homed networks must continue to be explicitly advertised throughout the system […], the number multi-homed routes is expected to be the dominant factor in future growth of routing table size, once the supernetting plan is applied.”

Sin embargo, se esperaba que el número de sitios multihomed fuera a crecer de forma más controlada que el número de sitios en general, por lo que incluso con esta limitación CIDR brindaría una sustancial mejora en la agregación de las tablas de rutas. La hipótesis realizada ha demostrado ser cierta durante un largo periodo, desde 1994 hasta el 1998. Sin embargo, partir de 1998, las contribuciones de estos sitios al tamaño de la tabla global de direcciones comienzan a ser preocupantes, ya que se retoma el crecimiento exponencial en el número de entradas. Dentro de las causas posibles para este comportamiento podemos incluir la disminución de los costos de conexión a Internet, en particular de conexión permanente a Internet, con el avenimiento de

tecnologías como el cable o DSL con sus múltiples sabores. Dichas tecnologías de bajo coste hacen más posible una adopción masiva de multihoming. Es más, a medida que se encuentran disponibles opciones de conectividad a Internet de coste más bajo, puede resultar más económico para los sitios la contratación de múltiples accesos de bajo coste (y de baja calidad) en lugar de un único acceso de coste más alto (con la posibilidad de mayores garantías de disponibilidad), incrementando la adopción del multihoming.

El resultado final es que cada sitio multihomed contribuye con al menos una entrada en la tabla global de rutas.

2.7.3 Políticas de encaminamiento

Como ya hemos visto, ciertas políticas de encaminamiento requieren lograr diversos encaminamientos para ciertos rangos de direcciones dentro de un agregado, para lo cual es necesario desagregar el bloque de direcciones en múltiples bloques y anunciarlos por separado por distintos caminos. Por ejemplo, consideremos la configuración ilustrada en la figura siguiente:

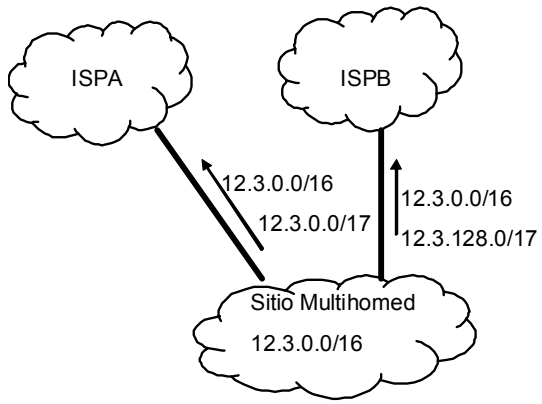

Figura 11: Políticas de encaminamiento en CIDR

En este caso, el sitio multihomed tiene un rango de direcciones 12.3.0.0/16 y dos proveedores, ISPA e ISPB. Supongamos que el proveedor ISPA brinda un servicio más caro, pero con mayores garantías de calidad de servicio y que ISPB ofrece un servicio más barato, pero sin garantías de calidad de servicio. Supongamos también que hay un grupo de máquinas de la red del sitio en cuestión que ejecutan aplicaciones que requieren un mejor servicio, mientras que otras máquinas son usadas para tareas menos críticas por lo que no es necesario garantizar el nivel de

servicio. Una solución es asignar a las maquinas de aplicaciones críticas direcciones de un subrango específico (12.3.0.0/17 en el ejemplo) y asignar direcciones de otro rango a las otras máquinas (12.3.128.0/17 en el ejemplo). Luego, se anuncia el rango reservado a las máquinas de aplicaciones críticas a través del ISPA y el otro rango a través del ISPB. Además, es deseable anunciar el agregado a través de ambos proveedores, de forma que se ofrezca un servicio de respaldo en caso de fallo de uno de los proveedores. En esta configuración, por la regla del longest prefix match, los paquetes destinados a direcciones dentro del rango 12.3.0.0/17 serán encaminados por el ISPA, siempre y cuando éste se encuentre disponible. Los paquetes destinados a direcciones dentro del rango 12.3.128.0/17 serán encaminados a través del ISPB. En caso de fallo de uno de los ISPs, todos los paquetes serán encaminados a través del ISP restante.

El resultado del empleo de esta técnica es un incremento en el número de entradas en la tabla de rutas. Cabe notar que no solamente los sitios finales multihomed utilizan estas técnicas de políticas, sino que también, y sobre todo, son utilizadas por los propios proveedores para dar forma a los patrones de tráfico que circulan por su red, por lo que el número de entradas en la tabla de rutas aportado por el uso de esta técnica es considerable. Una solución para paliar los efectos nocivos del uso de esta técnica consiste en filtrar los anuncios más específicos. Por ejemplo, es posible filtrar los anuncios cuyo prefijo sea más específico que un /24. Nótese que este filtrado no afecta a la alcanzabilidad del sitio que está tratando de imponer su política de encaminamiento, ya que por más que los anuncios más específicos sean filtrados, el anuncio del agregado probablemente no se vea afectado por el filtro. Sin embargo, el uso de filtros resulta problemático, ya que es posible que filtren completamente el anuncio de sitios que tienen asignaciones más pequeñas, por ejemplo un sitio multihomed.

2.7.4 Tamaño de las tablas de rutas en CIDR

En síntesis, el número de entradas presentes en la tabla global de rutas vendrá dado por:

$$N = N_{clases} + \sum_{i}^{I} d_i p_i + \sum_{m}^{M} d_m p_m + \sum_{r}^{R} d_r p_r$$

Siendo:

N: número total de entradas en la tabla global de rutas, sin considerar las rutas propias internas del AS

N_{clases} : número total de entradas en la tabla de rutas debido a los prefijos asignados mediante el sistema de clases. Se asume que este número está acotado y que su crecimiento efectivo es limitado. Adicionalmente es posible que disminuya debido a sitios que se reconviertan al sistema CIDR, devolviendo sus direcciones con clases y obteniendo rangos CIDR.

p_x : número de prefijos no agregables correspondientes al sitio x de la categoría en cuestión

d_x : coeficiente de desagregación del sitio x de la categoría en cuestión, el cual se define como el ratio entre el número de prefijos distintos anunciados por el AS y p_x

I : número de proveedores que han obtenido direcciones propias bajo CIDR

M : número de sitios multihomed

R : número de sitios que no han renumerado y se han llevado consigo las direcciones a otro proveedor

Resulta entonces que podemos estimar el número de entradas en la tabla de rutas como:

$$N \approx N_{clases} + d.p.(I + M + R)$$

Siendo:

d : la media de coeficiente de desagregación de un AS

p : el número medio de prefijos no agregables asignados a un AS

Por ende, el crecimiento del número de entradas en la tabla global de rutas N se verá afectado por:

$$N \propto (d, p, I, M, R)$$

Siendo

I : número de proveedores que han obtenido direcciones propias bajo CIDR

M : número de sitios multihomed

R : número de sitios que no han renumerado y se han llevado consigo las direcciones a otro proveedor

d : la media de coeficiente de desagregación de un AS

p : el número medio de prefijos no agregables asignados a un AS

Como puede verse, el número de entradas en la tabla global de rutas aumenta en proporción directa al número de sitios multihomed, el número de sitios que no renumeran cuando cambian de ISP, el número de proveedores, el número de prefijos asignado a cada AS y el nivel de desagregación que realicen los AS de sus prefijos.

Al diseñar IPv6, se ha hecho especial hincapié en la escalabilidad del sistema de rutas, por lo que es un objetivo explícito de IPv6 el contener lo más posible el crecimiento de las mismas. Para ello, se han incluido desde sus comienzos, herramientas que faciliten el renumerado de redes, intentando reducir al máximo el número de sitios R . Asimismo, se ha recomendado [RFC3177] la asignación de bloques de direcciones holgados (hasta podría decirse que exagerados) a los sitios finales e ISPs, de forma de asegurar que prácticamente todos los sitios tendrán suficientes direcciones con una sola asignación, haciendo que p sea prácticamente 1.

El objetivo de la Tesis propuesta es intentar minimizar y si es posible eliminar la contribución realizada por los sitios multihomed al crecimiento de las tablas de rutas, es decir eliminar la dependencia de M en la ecuación anterior.

En caso de lograr el objetivo propuesto en esta Tesis y si además consideramos que los otros objetivos perseguidos en el diseño de IPv6 se consiguen, resultaría que:

El impacto de p se ve prácticamente anulado ya que de acuerdo a la RFC 3221, $p = 1$

El impacto de R se minimiza gracias a las herramientas de renumeración disponibles y las políticas de asignación de direcciones.

El impacto de M se anula gracias a una nueva solución de multihoming a cuyo estudio se aboca la presente tesis

Lo anterior implica que:

$$N \propto (I, d)$$

Siendo

I : número de proveedores que han obtenido direcciones propias bajo CIDR

d : la media de coeficiente de desagregación de un AS

Es decir que el crecimiento de las tablas de rutas depende esencialmente de la mecánica de operación de los ISPs, de cómo obtienen direcciones, y de las herramientas utilizadas por los mismos para expresar sus políticas de encaminamiento.

Capítulo 3

Objetivos de diseño de una solución de multihoming para IPv6

3.1 Introducción

A continuación detallaremos los objetivos de diseño de una solución de multihoming para IPv6. Una lista inicial de objetivos puede encontrarse en el documento de objetivos de multihoming generado por el grupo de trabajo **multi6** del IETF [RFC3582]. Sin embargo, los objetivos presentados a continuación son considerablemente más concretos, ya que se basan en la conclusión de que la única arquitectura de direccionamiento escalable es aquella basada en la agregación por proveedor de direcciones (direcciones PA). En grandes líneas, una solución de multihoming deberá proveer funcionalidades similares (o mejores) a las ofrecidas a los sitios multihomed por la solución disponible hoy para IPv4, pero además deberá presentar mejores características de escalabilidad en términos del sistema de encaminamiento. En particular, el tamaño de la tabla de rutas no debe depender de forma lineal o superior con el número de sitios multihomed. Adicionalmente, la solución de multihoming no deberá generar dificultades en el resto de la arquitectura de Internet. A continuación detallaremos cada uno de estos puntos.

3.2 Tolerancia a fallos

Como hemos visto, la principal motivación para el multihoming es la mejora en la tolerancia a fallos. Por esto, una solución de multihoming debe brindar la máxima protección posible frente a fallos. Esto se traduce en que el sitio debe ser capaz de comunicarse con el exterior cuando hay un fallo en la red.

Por comunicarse, entendemos que el sitio debe ser capaz de:

- iniciar nuevas comunicaciones después del fallo
- recibir nuevas comunicaciones después del fallo
- preservar las comunicaciones ya establecidas después del fallo

Por *fallo* entendemos cualquier tipo de fallo en la red, siempre y cuando no afecte a todos los caminos disponibles hacia/desde el sitio. En particular dentro de los fallos frente a los que se debe brindar protección se incluyen:

- Fallo en la infraestructura física, como el fallo en la línea de comunicación o en los equipos de comunicación
- Fallos lógicos, como ser el comportamiento errático del equipamiento de comunicaciones
- Fallo en los protocolos de encaminamiento (BGP)
- Fallo total en el proveedor de servicio (directo o indirecto)
- Fallo en el punto neutro de interconexión utilizado por el proveedor.

En resumen, la solución debe ocultar cualquier tipo de fallo mientras exista al menos un camino disponible entre el sitio e Internet.

3.3 Balanceo de carga

Otra de las funcionalidades ofrecidas por la solución de multihoming actualmente disponible en IPv4 es la de balanceo de carga, por lo que una nueva solución debe ofrecer facilidades similares, es decir debe permitir distribuir la carga de tráfico entre los diversos proveedores disponibles.

3.4 Configuración de políticas de encaminamiento.

Adicionalmente, una solución de multihoming deberá proveer capacidades similares a las ofrecidas por la solución actual en cuanto a configuración de políticas de encaminamiento. Como hemos presentado anteriormente, la solución actualmente disponible permite:

- Definir el camino de los paquetes salientes de un modo bastante centralizado, ya que basta con configurar los routers de borde; además resulta bastante difícil sortear las políticas impuestas por el administrador del sitio, ya que el encaminamiento interno refleja dichas políticas
- Influenciar en el camino de los paquetes entrantes, si bien la elección última del camino está en manos de los sitios externos.

3.5 Escalabilidad del sistema global de rutas

La principal limitación de la solución de multihoming actualmente disponible es su contribución al crecimiento de las tablas globales de rutas. La nueva solución de multihoming deberá presentar mejores características de escalabilidad, en particular en lo que se refiere al sistema de rutas interdominio. Actualmente, la única arquitectura de direccionamiento que garantiza la escalabilidad del sistema de rutas es la basada en la agregación por proveedor presentada anteriormente, por lo que la nueva solución de multihoming deberá respetar la agregación por proveedor sin dañarla.

Como hemos visto, la solución actual no respeta la agregación por proveedor ya que el rango asignado a los sitios multihomed es inyectado por sus proveedores como una ruta más especifica, rompiendo la agregación. Si el objetivo es preservar la agregación, es necesario evitar la inyección de rutas más específicas, por lo que los proveedores sólo anunciarán sus propios agregados y no anunciarán direcciones que no pertenezcan a los mismos. Esto implica que sólo las direcciones pertenecientes a los rangos asignados a un proveedor serán accesibles a través del mismo. Por ende, si un sitio desea ser accesible a través de un proveedor, deberá configurar direcciones pertenecientes al rango del proveedor. En el caso de un sitio multihomed, el sitio deberá configurar múltiples direcciones en las interfaces conectadas a al red, una por cada proveedor contratado. De esta forma, las máquinas dentro del sitio son accesibles a través de todos los proveedores, siendo necesario utilizar distintas direcciones en función del proveedor que se quiera utilizar.

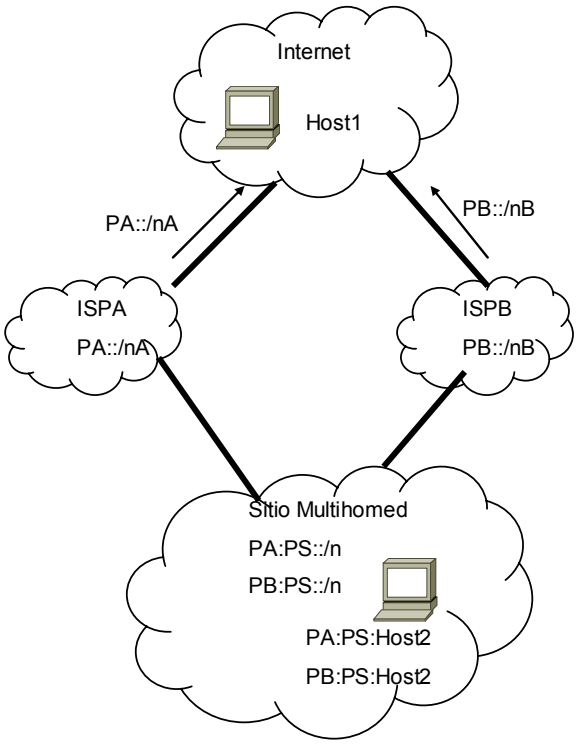

Figura 12: Sitio multihomed con direcciones PA

La configuración resultante implica que:

- Los proveedores sólo anuncian su propio agregado en el sistema de rutas interdominio, asegurando la escalabilidad del sistema de rutas
- Las máquinas del sitio multihomed son accesible a través de un ISP X sólo si tiene configurada una dirección perteneciente al rango asignado por el proveedor X. para

que las máquinas sean accesibles a través de todos los proveedores del sitio, se deberán configurar múltiples direcciones, una por ISP. Esta configuración se conoce como *multidireccionamiento*.

3.5.1 Consecuencias de la adopción del multidireccionamiento

Es trivial ver que la adopción del multidireccionamiento agrega una dificultad adicional en lo que se refiere a la selección de la dirección IP a utilizar en una comunicación. Cuando cada máquina cuenta con una única dirección IP, el algoritmo de selección es trivial. Si existen múltiples direcciones, deja de serlo. Si bien es cierto que en general en IPv6 se espera que los nodos posean múltiples direcciones, en este caso en particular las direcciones pueden tener distintas implicaciones en cuanto a las rutas escogidas para los paquetes que las utilicen. Nótese que en la configuración usada actualmente por los sitios multihomed en IPv4, con una única dirección IP por interfaz, esta dirección es accesible a través de todos los ISPs. Sin embargo, en una configuración con multidireccionamiento, cada una de las direcciones asignadas es solamente accesible a través de un único proveedor (el proveedor que asignó la dirección en cuestión al sitio). Si este proveedor sufriera un fallo, esta dirección queda inaccesible y el sitio sería solamente alcanzable usando otra dirección perteneciente a un ISP alternativo. Esto implica que una comunicación progresará correctamente si se utiliza una dirección IP correspondiente a un ISP que esté funcionando correctamente en ese momento, y fallará si se utiliza una dirección IP perteneciente a un ISP que no esté disponible en ese momento. Dependiendo de si la comunicación es iniciada por un nodo perteneciente al sitio multihomed o por una máquina externa, será necesario elegir correctamente la dirección origen o la dirección destino.

Consideremos el escenario ilustrado por la figura anterior. En el caso de que la comunicación sea iniciada por un nodo externo *Host1*, éste solicitará al DNS las direcciones correspondientes al nodo objetivo, es decir *Host2*. El DNS devolverá todas las direcciones disponibles para *Host2*, a saber, *PA:PS:Host2* y *PB:PS:Host2*. Entonces, para establecer la comunicación, *Host1* deberá elegir una de las direcciones y enviar paquetes a la misma. En caso de que los dos ISPs funcionen correctamente, no importa la dirección elegida (o al menos no influirá en el éxito de la comunicación). Sin embargo, si ha ocurrido un fallo en alguno de los proveedores, por ejemplo en el *ISPA*, la elección de la dirección se torna relevante, ya que si se intenta establecer la comunicación con la dirección correspondiente al ISP que ha sufrido el fallo (*PA:PS:Host2* en este caso) la comunicación fallará. Cabe notar, que si bien no es habitual el uso de configuraciones con multidireccionamiento en la Internet actual, sí es común que la consulta al DNS por un nombre de dominio devuelva múltiples direcciones y que algunas de ellas fallen, por

lo que es esperable que las aplicaciones actuales reintenten utilizando direcciones destino alternativas.

Cuando la comunicación sea iniciada por un nodo dentro del sitio multihomed *Host2*, éste solicitará al DNS las direcciones correspondientes al nodo objetivo, es decir Host1. Supongamos que el DNS devuelve una dirección única para *Host1*; en este caso la selección de la dirección destino es trivial. Sin embargo, para establecer la comunicación, *Host2* deberá elegir la dirección origen que utilizará para enviar paquetes. Cuando los dos ISPs funcionan correctamente, no importa la dirección elegida (o al menos no influirá en el éxito de la comunicación). Sin embargo, si ha ocurrido un fallo en alguno de los proveedores, por ejemplo en el *ISPA*, la elección de la dirección se torna relevante, ya que si se utiliza la dirección correspondiente al ISP que ha sufrido el fallo (*PA:PS:Host2* en este caso) la comunicación fallará. El fallo no afectará al paquete enviado por *Host2* ya que éste será encaminado a través del *ISPB* hacia *Host1*[4]. El problema ocurrirá con el paquete de vuelta desde *Host1* hacia *Host2*. *Host1* enviará los paquetes de respuesta a la dirección usada como dirección origen en el paquete inicial, es decir *PA:PS:Host2*. Debido al fallo ocurrido en el *ISPA*, este paquete no podrá ser entregado a *Host2* ya que no existe un camino disponible. Cabe notar que en este caso la comunicación fallará aunque exista un camino disponible entre *Host1* y *Host2* y que el fallo se debe no a una mala elección de la dirección destino, sino a una mala elección de la dirección origen. Actualmente, no es habitual que las aplicaciones reintenten con direcciones origen alternativas.

Además de los problemas presentados en el momento de establecer una nueva comunicación cuando ha ocurrido un fallo en un proveedor, la configuración con multidireccionamiento presenta dificultades para preservar las comunicaciones establecidas cuando hay un fallo en el proveedor que se está utilizando para encaminar los paquetes de la comunicación. Supongamos que se ha establecido una comunicación entre *Host1* y *Host2* utilizando la dirección IP *PA:PS:Host2* de *Host2* para intercambiar paquetes. En caso que ocurra un fallo en el *ISPA*, el flujo de paquetes de la comunicación en cuestión se detendrá, ya que no hay un camino para alcanzar la dirección *PA:PS:Host2*. La única forma de restablecer el flujo de paquetes es utilizar el proveedor alternativo, *ISPB*, para lo cual es necesario enviar paquetes a la dirección de *Host2* asociada a dicho proveedor, es decir *PB:PS:Host2*. Sin embargo, esto no sería suficiente para preservar la comunicación establecida ya que los paquetes destinados a la nueva dirección no serían reconocidos como pertenecientes a la comunicación establecida. En el caso de TCP, esto se debe a que las conexiones TCP son identificadas por la tupla (protocolo, dirección IP origen, dirección IP destino, puerto origen, puerto destino). Por ende, los paquetes enviados a la dirección alternativa del *Host2* no serían reconocidos como pertenecientes a la conexión TCP existente por llevar una dirección distinta. El mismo caso puede suceder en comunicaciones entre aplicaciones

[4] No Estamos considerando los efectos de los filtros de ingreso, que se estudiarán en la sección siguiente.

que utilicen la dirección IP del interlocutor para identificar a las partes involucradas en la comunicación.

En síntesis, la adopción de un esquema de multidireccionamiento reduce dramáticamente las capacidades de tolerancia a fallos de un sitio multihomed afectando:

- Al establecimiento de nuevas comunicaciones cuando hay un fallo
- A la preservación de las comunicaciones establecidas cuando hay un fallo

La nueva arquitectura de multihoming deberá proveer una solución a estos problemas, preservando las capacidades de tolerancia a fallos de una forma compatible con la arquitectura de direccionamiento basada en la agregación por proveedor.

3.6 Compatibilidad con los filtros de ingreso

Un filtro de ingreso esencialmente verifica que la dirección origen contenida en el paquete sea topológicamente correcta. En particular, los ISPs configuran filtros de ingreso en los routers que dan acceso a sus clientes de forma que sólo aceptan paquetes cuya dirección origen contenga el prefijo que el ISP ha asignado al cliente. Existen varias formas de implementar esto [RFC3704], desde la configuración manual de los prefijos que son aceptables por cada interfaz, hasta la utilización de *encaminamiento por camino de retorno para tráfico unicast (uRPF, unicast Reverse Path Forwarding)*. El mecanismo de uRPF se basa en determinar la interfaz de salida por la que se encaminaría un paquete destinado a la dirección usada como dirección origen en el paquete recibido. Si la interfaz de salida resultante es la interfaz por la que se recibió el paquete, esto quiere decir que el camino por el que le ha llegado el paquete es coherente con la visión de la topología que tiene el sistema de encaminamiento, por lo que el paquete será procesado. Pero si las interfaces son distintas, esto es interpretado como que la dirección origen usada no es coherente con la topología, por lo que el paquete es descartado.

La utilización de *filtros de ingreso* [RFC2827] se recomienda para dificultar la realización de ataques de denegación de servicio que utilizan direcciones origen falsas. El uso de direcciones de origen falsas ofrece diversos beneficios a los atacantes. Por un lado, el atacante puede ocultar su identidad, dificultando su posterior identificación. Adicionalmente, el uso de direcciones origen falsas puede ser usado por el atacante para dirigir un flujo de paquetes hacia la víctima, al utilizar la dirección de la víctima como dirección IP origen al solicitar el flujo de paquetes. Mediante el uso de filtros de ingreso, se limita el uso de direcciones origen falsas, dificultando estos ataques.

En cualquier caso, el resultado final de la adopción de filtros de ingreso es que un ISP sólo reenvía paquetes que contienen direcciones origen que a su entender son topológicamente correctas.

En el caso de un sitio multihomed al que se le han asignado varios prefijos, uno por proveedor, las técnicas de filtrado de ingreso añaden complicaciones adicionales como se detalla a continuación.

Como se ve en la figura 12, los nodos pertenecientes a un sitio multihomed con múltiples prefijos tienen múltiples direcciones, una por prefijo asignado al sitio, si es que estos nodos desean ser accesibles a través de todos los proveedores. Por ende, al enviar un paquete, el nodo deberá elegir qué dirección incluirá como dirección origen en el paquete. Nótese que esta decisión pertenece al nodo. Por ejemplo, en la figura anterior, supongamos que el *Host2* desea enviar paquetes al *Host1*. El *Host2* deberá elegir qué dirección usar como dirección origen. Supongamos que elige la dirección *PrefA:PS:Host2*. Una vez que el nodo (*Host2*) ha enviado el paquete, es el sistema de rutas del sitio multihomed quien decide qué camino de salida será utilizado para encaminar el paquete hacia Internet. Típicamente, el sistema de rutas toma esta decisión basándose en la dirección destino del paquete, independientemente de la dirección origen usada. En el caso en que el proveedor elegido por el sistema de rutas se corresponda con el proveedor que ha delegado el prefijo contenido el la dirección origen, el paquete será encaminado con éxito. Sin embargo, si este no es el caso, el paquete será descartado debido a los filtros de ingreso. En el ejemplo considerado, si el paquete es encaminado por el sistema de rutas interno del sitio multihomed hacia el *ISPA*, el paquete será enviado con éxito. Pero si es encaminado a través del *ISPB*, el paquete será descartado por el router de borde del *ISPB*, ya que éste no será compatible con los filtros configurados por el *ISPB*. Nótese que para cada ISP, el sitio multihomed sólo tiene el prefijo que el propio ISP le ha asignado y no es consciente de la existencia del otro prefijo en su cliente. En el ejemplo, el *ISPA* asume que el sitio tiene el prefijo *PA:PS::* por lo que configurará sus filtros para que sólo acepten paquetes que contengan direcciones origen con ese prefijo, e *ISPB* hará lo propio.

Una solución de multihoming deberá ser compatible con las técnicas de filtro de ingreso existentes.

3.7 Compatibilidad con equipos existentes

Es esperable que una solución al problema de multihoming requiera la adopción de nuevos mecanismos e incluso protocolos tanto en nodos finales como en los routers. Es necesario entonces preservar el correcto funcionamiento de los equipos actualmente existentes. Este

requerimiento tiene diversas implicaciones. Por un lado, si la solución requiere mecanismos y/o protocolos en los routers, los nuevos routers deben ser capaces de interoperar con aquellos routers que no soporten el nuevo mecanismo. Si consideramos los nodos finales, es posible que la solución requiera que estos implementen nuevos mecanismos y/o protocolos. Sin embargo, es necesario también dar algún nivel de soporte a nodos que no soporten los mismos.

En el caso de los nodos dentro del sitio multihomed que no soportan las nuevas funciones, es necesario que estos funcionen al menos igual de bien que funcionan en un sitio no multihomed. Es aceptable que estos nodos no se beneficien del multihoming, pero al menos deben recibir el nivel de servicio que recibirían en un sitio no multihomed. En otras palabras, que si conectamos un nodo de los que existen actualmente a un sitio multihomed, éste tiene que funcionar al menos como si estuviera en un sitio no multihomed. Nótese que actualmente no ocurre así, ya que ciertas comunicaciones fallarían en un sitio multihomed con múltiples prefijos debido a los filtros de ingreso, como se ha descrito en la sección anterior. En el caso de los nodos externos al sitio multihomed, es necesario que todos los nodos dentro del sitio multihomed, es decir los que soportan los nuevos mecanismos y los que no, puedan comunicarse normalmente con un nodo fuera del sitio multihomed que no soporte los nuevos mecanismos.

3.8 Otras condiciones

Impacto en el DNS: la solución deberá ser compatible con el funcionamiento actual del sistema de nombres de Internet. En particular no debe generar dependencias circulares con éste, es decir, el DNS utiliza el sistema de encaminamiento para transportar los paquetes vinculados a su protocolo. Una solución basada de alguna forma en el uso del DNS debe asegurar que su adopción no implica que el sistema de encaminamiento se base en el uso del DNS para su funcionamiento, ya que esto implicaría una dependencia circular entre ambos sistemas.

Modificaciones requeridas: las modificaciones requeridas por la solución en los equipos existentes deben poder implementarse como una función lógica independiente e implementable como una mecanismo que funcione paralelamente a las funciones actuales.

Interacción entre proveedores: No es deseable que una solución requiera cooperación entre los distintos proveedores de un sitio, ya que estos típicamente están en competencia, y no estarán predispuestos a la cooperación.

Simplicidad: una solución debe ser todo lo simple que sea posible, sin implicar complejas interacciones entre las partes y debe ser simple de administrar y operar.

Seguridad: la solución adoptada no debe introducir nuevas vulnerabilidades en Internet.

Capítulo 4

Análisis del espacio de soluciones

4.1 Introducción

Como hemos visto, con el objetivo de preservar la escalabilidad del sistema de rutas de inter-dominio, es necesario utilizar una arquitectura de direccionamiento basada en la agregación de direcciones por proveedor. Esto conlleva que los sitios multihomed deberán obtener un prefijo por cada uno de sus proveedores, y que los nodos deberán configurar múltiples direcciones en sus interfaces para poder ser alcanzables a través de los distintos proveedores. Considerando que:

- el objetivo de un sitio multihomed es ser accesible a través de todos sus proveedores, y
- que cuando introducimos el uso de direcciones PA, un nodo es accesible a través de un proveedor si y sólo si la dirección asociada al proveedor es usada, y
- que para ser accesible a través de otro proveedor es necesario cambiar la dirección usada por la dirección asociada al proveedor alternativo.

Resulta claro que **el problema fundamental a resolver es cómo vincular todas las direcciones que posee un mismo nodo de forma que se puedan usar distintas direcciones en la comunicación con el mismo según sea necesario.** En otras palabras, es necesario que exista al

menos una entidad que tenga conocimiento de las múltiples direcciones asociadas a un nodo multihomed y que sea capaz de elegir cuál usar en un contexto dado.

En el presente capítulo realizaremos un análisis del espacio de soluciones para el problema de multihoming. El análisis se dividirá en tres partes. Primero realizaremos un análisis arquitectónico, en el cual abstraeremos las decisiones fundamentales que implican los distintos enfoques planteados y evaluaremos las implicaciones arquitectónicas y operativas que conllevan. Después, realizaremos un análisis de seguridad, ya que el establecimiento de un vínculo entre distintas direcciones puede introducir riesgos de seguridad que deben ser analizados. Finalmente, realizaremos un análisis funcional que nos permita identificar cada una de las funciones necesarias para construir una solución de multihoming.

4.2 Análisis Arquitectónico

4.2.1 Los roles de las direcciones IP como motivación arquitectónica del problema

Antes de pasar al análisis del espacio de soluciones ahondaremos un poco más en el análisis de las causas del problema. El problema detectado es esencialmente que un nodo que tiene múltiples direcciones IP no puede usarlas indistintamente. A continuación analizaremos las causas de dichas dificultades, para concluir que éstas esencialmente residen en los múltiples roles que tienen las direcciones IP en la arquitectura TCP/IP.

En los trabajos realizados por Shoch en los años 70 [Shoch1978a] se distinguen tres conceptos esenciales en las redes de ordenadores con la terminología siguiente:

- El *nombre* de un recurso indica *qué* buscamos
- La *dirección* indica *dónde* se encuentra
- La *ruta* nos indica *cómo llegar*

Estas definiciones nos brindan una visión intuitiva de estos conceptos ya familiares. Algunos años más tarde, Saltzer [RFC1498] profundizó en el estudio realizado por Shoch para definir los siguientes elementos esenciales que requieren ser nombrados en una red:

- *Servicios y usuarios*: son las funciones que proveen servicios y los clientes que hacen uso de los mismos.
- *Nodos*: dispositivos que ejecutan las funciones anteriores
- *Puntos de acceso a la red*: puntos en la red donde los nodos obtienen conectividad

- *Caminos*: es el trayecto seguido por los paquetes para viajar entre dos puntos de acceso a la red.

Cabe notar que, a diferencia del trabajo de Shoch, el trabajo de Saltzer se centra en los elementos fundamentes de la red más que en los nombres de los mismos. Una vez identificadas las clases de elementos fundamentales de la red y considerando que en una red existirán múltiples instancias de cada una de las clases de elementos, es necesario definir un nombre para cada una de las instancias en cada una de las clases de elementos. En este enfoque, el término *nombre* es usado en un sentido más amplio que en el caso de Shoch, y se refiere simplemente al nombre o elemento identificativo (una cadena de caracteres) asignado a una instancia particular de una clase de elementos. Según este enfoque, habrá nombres de caminos, nombres de puntos de acceso a la red, nombres de nodos y nombres de servicios. Además, Saltzer nos brinda una definición general de la dirección de un elemento como "el nombre del elemento inferior al que esta unido"[5]. Es decir, que la dirección de un servicio es el nombre del nodo donde el servicio se ejecuta, la dirección de un nodo es el nombre del punto de acceso a la red donde está conectado, y la dirección de un punto de acceso a la red son los nombres de los caminos que llevan al mismo.

Trabajos posteriores realizados por Chiappa [Chiappa1999a] ahondan más aún en el concepto de nodo como elemento fundamental de la red, tomando como nodo no el dispositivo físico, sino el *extremo* involucrado en la comunicación. La diferencia entre estos dos conceptos es sutil, ya que en las situaciones más comunes el extremo coincide con el nodo. Sin embargo, en ciertas situaciones, como puede ser en procesos o agentes móviles, el extremo involucrado en la comunicación no es lo mismo que el nodo donde se ejecuta. Asimismo, en el caso de las granjas de servidores, donde un proceso se ejecuta en múltiples nodos simultáneamente, el extremo de la comunicación no coincide con un único dispositivo físico sino que está distribuido en varios. Un extremo se define entonces como la entidad que realiza una comunicación de extremo a extremo. Este concepto de *extremo* es también contemplado bajo el nombre de *stack* en el informe final del grupo de investigación *Namespace* del Internet Research Task Force (IRTF) [Lear2004a].

En síntesis, podemos definir las siguientes clases de elementos fundamentales en una red:

- **Servicios:** son las funciones que proveen servicios y los clientes que hacen uso de los mismos
- **Extremos:** entidad que realiza una comunicación confiable de extremo a extremo
- **Puntos de acceso a la red:** puntos en la red donde los nodos obtienen conectividad

[5] Inferior en este caso se refiere a la jerarquía definida por servicios y usuarios que residen en nodos los que a su vez están conectado a la red a través de puntos de acceso los cuales a su vez están unidos entre si a través de caminos. La jerarquía queda definida, de superior a inferior de la siguiente manera: Servicios y usuarios, nodos, puntos de acceso y caminos.

- **Caminos:** es el trayecto seguido por los paquetes para viajar entre dos puntos de acceso a la red.

Adicionalmente, podemos brindar las siguientes definiciones referentes a los distintos nombres:

- **Dirección de un elemento:** el nombre del objeto a la que se encuentra unido.
- **Identificador**: nombre de un extremo.
- **Localizador**: nombre de un punto de acceso a la red.
- **Ruta**: nombre de un camino.

4.2.1.1 Clases de elementos fundamentales y espacios de nombres en TCP/IP

Una vez que hemos identificado los elementos fundamentales de red, intentaremos identificar los nombres que les son asignados en la arquitectura TCP/IP. A simple vista, sólo parecen existir dos espacios de nombres en la arquitectura, a saber, el espacio de nombres de dominio [RFC1034] [RFC1035] y las direcciones IP. Es más, los nombres de dominio son tangenciales a la arquitectura, en el sentido de que es posible mantener una comunicación TCP/IP sin usar los nombres de dominio, ya que son esencialmente alias mnemotécnicos y amigables para los humanos de los nombres reales, es decir las direcciones IP.

Por ello resulta que la arquitectura TCP/IP sólo cuenta con un espacio de nombres, el espacio de las direcciones IP. Si contrastamos esta conclusión con las clases de elementos fundamentales de una red identificados previamente, resulta que las direcciones IP son usadas como:

- Nombre de puntos de acceso a la red: lo que coincide con la definición intuitiva de la dirección de un nodo.
- Nombre de extremos, ya que son usadas por la capa IP, por las capas de transporte y las aplicaciones para identificar a los extremos de una comunicación.
- Parte del nombre de un servicio. Los servicios que se ejecutan en un extremo son identificados mediante la combinación de dirección IP, protocolo y puerto.
- Parte del nombre de un camino, llamado ruta, ya que las rutas vienen dadas por dirección IP destino, mascara, y dirección IP del próximo salto.

Adicionalmente, la dirección IP parece también jugar otros roles, como los de:

- Parte de la identificación de una comunicación en protocolos como TCP. Una comunicación es identificada mediante la combinación de dirección IP origen y destino, puerto origen y destino y protocolo.
- Etiqueta de encaminamiento. Los routers utilizan la dirección IP de destino contenida en los paquetes para encaminar el paquete hacia su destino final, ya que la dirección IP es

llevada en el paquete. Nótese que otras arquitecturas como puede ser ATM, la etiqueta de encaminamiento y la dirección son elementos distintos.

Sintetizando, los nombres usados en la arquitectura TCP/IP son los siguientes:

- Nombre de servicio: dirección IP, puerto y protocolo
- Identificador: Dirección IP
- Localizador: Dirección IP
- Ruta: dirección IP destino, máscara y dirección IP del próximo salto, es decir que, si esta dado el destino, el nombre de la ruta se limita a la dirección IP del próximo salto. Cabe notar que este es un nombre relativo al nodo donde se hace referencia y que el nombre de un mismo camino cambia de nodo en nodo.

4.2.1.2 Relaciones entre elementos y nombres en la arquitectura TCP/IP

Una vez identificados los elementos fundamentales de una arquitectura de red y los espacios de nombres existentes en la arquitectura de red, es relevante entender las distintas relaciones existentes entre los distintos elementos y los espacios de nombres para el caso concreto de las redes TCP/IP, por lo que se detallan a continuación.

4.2.1.2.1 Relaciones entre los elementos

- **Relación entre servicio y extremo**: la relación viene dada por el hecho de que el servicio se ejecuta sobre el extremo en cuestión. La relación es n a 1, es decir que múltiples servicios pueden ejecutarse sobre el mismo extremo, pero por definición de extremo, una instancia de un servicio se ejecuta sobre un solo extremo.
- **Relación entre punto de acceso a la red y extremo**: La relación se establece cuando el extremo se conecta a la red a través del punto de acceso, por ejemplo cuando conectamos un ordenador a una clavija de red. La relación es n a 1, ya que un extremo podría estar conectado a n puntos de acceso, mientras que podemos asumir que en un punto de acceso dado sólo se conecta un extremo (en el caso de WLAN, por ejemplo podemos asociar cada punto de acceso a la red con cada una de la direcciones Ethernet presentes)
- **Relación entre camino y punto de acceso a la red:** La relación entre dos puntos de acceso y el camino que los une viene definida por la topología física de la red. La relación es n a 1, ya que pueden existir muchos caminos entre dos puntos de acceso, pero el camino entre dos puntos de acceso no puede unir a otros dos puntos de acceso que no sean los considerados.

Otras posibles relaciones entre los elementos se derivan a partir de las mostradas aquí.

4.2.1.2.2 Relación entre los nombres de los elementos

- **Relación entre nombre de servicio e identificador**: el nombre del servicio contiene al identificador (dirección IP) en la arquitectura TCP/IP, ya que el nombre del servicio es un nombre jerárquico, formado por el nombre del extremo que provee el servicio, seguido del puerto y protocolo donde ubicar dicho servicio en el nodo en cuestión.
- **Relación entre identificador y localizador**: la relación es la identidad ya que ambos son la dirección IP del nodo en cuestión.
- **Relación entre localizador y ruta**: la relación entre el localizador de un destino y la ruta que identifica al camino que debemos seguir para llegar al mismo viene dada por la tabla de rutas. Es decir si tenemos el localizador de un destino en particular, la tabla de rutas nos devuelve la ruta (el nombre del camino) a seguir, determinada por la dirección IP del próximo salto. Cabe notar que esta relación es dinámica gracias a los protocolos de encaminamiento.

De forma análoga al caso anterior, estas son las únicas relaciones que tienen sentido de forma directa.

4.2.1.2.3 Relación entre los elementos y sus nombres

- **Relación entre los puntos de acceso a la red y sus nombres** (i.e. las direcciones IP). El proceso mediante el cual se asignan direcciones IP a los distintos puntos de acceso a la red se llama *asignación*. Este proceso ha cambiado durante la vida de Internet y actualmente se realiza de la siguiente forma: El responsable último de las direcciones IP es el ICANN[6], quien ha delegado la gestión del total de las direcciones a IANA[7] IANA a su vez, delega bloques (de gran tamaño) de direcciones a los Registros Regionales de Internet (RIRs, *Regional Internet Registries*), de los cuales actualmente existen cinco en el mundo: LACNIC[8] para América Latina y el Caribe, RIPE[9] para Europa, APNIC[10] para la región de Asia-Pacífico, ARIN[11] para Estados Unidos de América y Canadá y AfriNIC[12] para la región de Africa. Los RIRs delegan a su vez direcciones a los Registros Locales de Internet (LIRs, *Local Internet Registries*) que normalmente son ISPs que a su vez delegan bloques de direcciones a los sitios finales. Dentro del sitio

[6] http://www.icann.net

[7] http://www.iana.net

[8] http://www.lacnic.net

[9] http://www.ripe.net

[10] http://www.apnic.net

[11] http://www.arin.net

[12] http://www.afrinic.net

final, el administrador de red configura direcciones IP a las interfaces de los nodos que se conectan a la red. Cabe notar que en algunos casos los RIRs pueden delegar direcciones a Registros de Internet Nacionales (como en el caso de Brasil, México o Japón) quien a su vez delega bloques a los LIRs dentro del país en cuestión. Además los RIRs pueden en algunos casos delegar direcciones directamente a los usuarios finales por diversas razones, dependiendo de las políticas válidas en la región. En IPv4, en ciertas regiones como la del LACNIC, los usuarios con múltiples proveedores pueden obtener una asignación IPv4 directamente del RIR, independientemente del proveedor que utilicen para conectarse a Internet. Esto no es posible por ahora para IPv6 en ninguna de las regiones, ya que como hemos dicho anteriormente, se espera que en IPv6 los usuarios con múltiples proveedores utilicen espacio PA (como los demás sitios) con el objetivo de preservar la escalabilidad del sistema de rutas. Adicionalmente, tanto en IPv6 como en IPv4, la infraestructura crítica de Internet (puntos neutros de interconexión, servidores raíz de DNS, servidores ccTLD) pueden obtener asignaciones directas del RIR, con direcciones independientes del proveedor. Como conclusión, vemos que la relación entre los puntos de acceso a la red y sus respectivos nombres es un proceso complejo, que se esta viendo profundamente modificado en el caso de los sitios con múltiples proveedores en IPv6 con respecto a la relación usada para los sitios multihomed en IPv4, con el objetivo de garantizar la escalabilidad del sistemas de rutas. Podemos ver entonces que el requerimiento del uso de asignación de direcciones IP basada en proveedor que surge de la necesidad de preservar la escalabilidad del sistema de rutas responde a necesidades que surgen del rol de las direcciones IP como nombre de los puntos de acceso a la red, que tiene que estar relacionado con la topología de la conectividad física de la red. Para ello, es necesario utilizar una estructura de asignación de nombres que sintetice los puntos de acceso a la red en el mínimo posible de rutas. Para ello, todos los puntos de acceso a la red que sean accesible a través de una misma ruta deberán poseer nombres que sean susceptibles de ser agregados en una misma entrada. Para ello, se puede utilizar la notación dirección IP / máscara, que sintetiza mediante esta dupla un gran conjunto de direcciones. La consecuencia de esto es que todas las direcciones de los puntos de acceso a la red que sean accesibles a través de la misma ruta deberán obtener direcciones del mismo bloque. Dado que, por definición, un proveedor es la ruta hacia todos sus clientes, resulta natural asignar el bloque de direcciones al ISP, de forma que éste anuncie una única ruta hacia todo el agregado de direcciones. Esto implica que los clientes de un ISP deberán obtener direcciones del ISP, resultando en la arquitectura de direccionamiento PA ya descrita. Es importante enfatizar que la necesidad del uso de la agregación por proveedor responde al rol de las direcciones IP como nombre de los puntos de acceso a la red.

- **Relación entre los extremos y sus nombres**, (i.e. la dirección IP). En la arquitectura IP los extremos son nombrados a partir del punto de acceso a la red donde son conectados, es decir que un extremo tiene por nombre la dirección IP del punto de acceso a la red al cual se conecta (esto se materializa cuando las conexiones TCP entre dos extremos utilizan las direcciones IP de estos para identificar las partes involucradas en la comunicación). La relación entre extremo y nombre es entonces indirecta y se realiza a través del proceso de conexión de un extremo a un punto de acceso a la red y por el proceso de asignación de dirección IP a dicho punto de acceso a la red. Cabe notar entonces que si un extremo cambia de punto de acceso a la red, es necesario cambiar el nombre del extremo, lo cual genera problemas entre otros casos cuando se pretende soportar la movilidad de terminales. Adicionalmente, si un extremo tiene múltiples accesos a la red, tendrá entonces múltiples nombres. Finalmente en el caso particular de los extremos que forman parte de un sitio con múltiples proveedores, estos también obtienen múltiples nombres debido al uso de direccionamiento PA. Esto implica que esos extremos también poseerán múltiples nombres que los identifiquen. El problema es que, dado que el nombre usado se relaciona íntimamente con el ISP usado, el uso de un nombre u otro del extremo implicará el uso de un ISP u otro alternativo, por lo que el nombre del extremo vinculado al un ISP no será alcanzable a través del ISP alternativo. Como resultado el extremo deberá usar nombres distintos dependiendo del ISP usado, lo que dificulta que sea reconocido como una misma entidad al cambiar el ISP usado en la comunicación. Para lograr esto, es necesario hacer saber al otro extremo de la comunicación que los múltiples nombres usados corresponden todos al mismo extremo, proceso que dista mucho de ser trivial como veremos a lo largo de este análisis.
- **Relación entre el servicio y su nombre**. Como hemos dicho, el nombre de un servicio en la arquitectura IP viene dado por la dirección IP asignada al punto de acceso a la red donde se encuentra conectado el extremo que ejecuta el servicio, el número que identifica el puerto donde se ejecuta el proceso que ofrece el servicio y el protocolo de transporte usado para la comunicación con el servicio. La relación entre un servicio y su nombre, es entonces compleja y puede tomar diversas formas. Por un lado depende de la relación entre el punto de acceso a la red y su dirección IP, es decir del proceso de asignación. Adicionalmente, depende de la relación entre un extremo y su punto de acceso a la red, es decir de la conectividad física del extremo a la red, por lo que el nombre del servicio será dependiente, como el nombre del extremo, de la conectividad física del extremo a la red. Adicionalmente, el nombre del servicio depende del puerto donde se ejecute, el cual puede estar predeterminado en caso de que el servicio se ofrezca en un puerto bien conocido, puede estar prefijado, en caso que el servicio se

ofrezca en un puerto fijo pero fuera del rango de los puertos bien conocidos o puede ser un puerto efímero, el cual es usado solamente para la oportunidad en cuestión.

- **Relación entre los caminos y sus nombres**. Los caminos pueden ser nombrados a través de las direcciones de todos los puntos de acceso que atraviesan. Si bien este mecanismo absoluto de identificación de caminos es usado en algunas ocasiones, en general, en la arquitectura TCP/IP los nombres de los caminos, es decir las rutas, son nombres relativos al punto de acceso a la red donde se nombren. Como ya hemos visto, la tabla de rutas es la herramienta usada para relacionar un punto de acceso a la red con la ruta que lleva al mismo. En la tabla de rutas, una ruta es identificada por la dirección IP del próximo salto a seguir, por lo que una ruta es un nombre de un camino relativo al punto de acceso a la red donde se haga referencia a la misma. La traducción de la ruta al camino es realizada por el motor de reenvío del nodo, quien envía el paquete a través del camino identificado por el próximo salto.

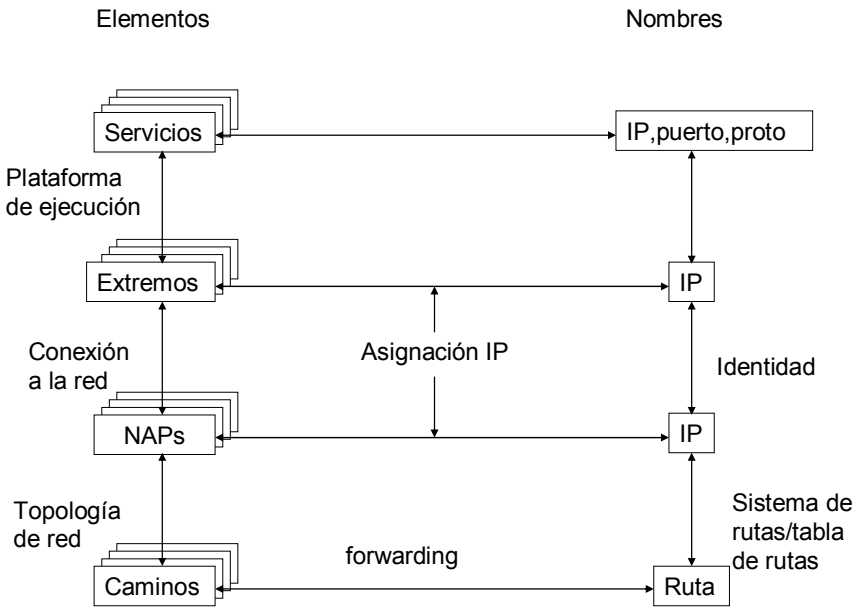

Figura 13: Relaciones entre elementos y nombres

Resulta entonces que la causa fundamental del problema del soporte de múltiples proveedores reside en los múltiples roles asignados a las direcciones IP en la arquitectura TCP/IP. El rol de localizador impone que la dirección IP esté íntimamente vinculada con la topología, lo que impone que si nodo se encuentra en distintos puntos de la topología simultáneamente, como ser el caso de un sitio multihomed, es necesario que éste reciba múltiples direcciones, vinculadas a cada uno de los puntos de conexión a la red. Análogamente, un nodo que cambia su posición en

la topología de la red deberá cambiar su localizador de forma acorde, lo que dificulta el soporte de la movilidad de nodos, como se puede ver en los esfuerzos para el soporte de movilidad existentes [RFC3775]. Por otra parte, el rol de identificador requiere que la dirección IP permanezca estable, incluso cuando cambie el punto de acceso a la red utilizado por el nodo, ya que el nodo continua siendo el mismo a pesar de que haya cambiado el punto en la topología usado para obtener conectividad. Cabe notar que mientras los nodos contaban con una sola dirección IP, los distintos roles no creaban incompatibilidades fundamentales por lo que no existían dificultades mayores.

4.2.2 Establecimiento del vínculo

Como hemos vistos, el vínculo entre las múltiples direcciones se puede realizar en diversas partes de la arquitectura, pero esencialmente existen dos criterios por los que se pueden clasificar los enfoques posibles: el nivel en la pila de protocolos donde se establece el vínculo y el elemento (o sistema) de la red que lo establece. En esta sección analizaremos las distintas opciones posibles para cada uno de los criterios presentados.

4.2.2.1 Clasificación del vínculo según el nivel en la pila de protocolos

El conocimiento referente a qué múltiples direcciones pertenecen a un mismo extremo puede residir en distintas partes de la pila de protocolos. De acuerdo a los principios de diseño de un sistema en capas, como lo es la pila de protocolos, el nivel de la pila que gestione el vínculo de las múltiples direcciones deberá resolver el problema haciéndolo transparente para las capas superiores. Es decir que si el vínculo es realizado por la capa enésima, las capas superiores no necesitarían tener conocimiento de la existencia de las múltiples direcciones ya que dicha capa resolvería este problema de forma transparente, mostrándoles siempre que en el otro extremo está el mismo interlocutor, independientemente de la dirección IP usada.

A continuación analizaremos la implementación del vínculo a nivel de aplicación, a nivel de una nueva capa de sesión, a nivel de transporte, y a nivel de una nueva capa de identificación por debajo de la capa de transporte.

4.2.2.1.1 Capa de aplicación

Una posibilidad es que las aplicaciones gestionen directamente el uso de múltiples direcciones y que sean ellas mismas quienes posean el conocimiento referente a las múltiples direcciones asociadas a un mismo extremo. Sin lugar a dudas existen y existirán aplicaciones que son capaces de manejar múltiples direcciones, incluso simultáneamente. Es más, dado que la

propia aplicación gestiona el uso de las direcciones, ésta puede hacerlo optimizando el objetivo perseguido por la misma.

Sin embargo, también parece claro que muchas de las aplicaciones existentes no son capaces de gestionar las múltiples direcciones por lo que resulta necesario ofrecer una solución genérica que no pase por actualizar todas las aplicaciones. Como consecuencia, aparte del soporte ofrecido por algunas aplicaciones se requerirá una solución más general, ubicada en un nivel inferior en la pila de protocolos que solucione el problema para las demás aplicaciones.

4.2.2.1.2 Nueva capa de sesión

Una posibilidad para brindar soporte a aquellas aplicaciones que no son capaces de gestionar las múltiples direcciones es la creación de una capa de *sesión* entre la capa de transporte y la capa de aplicación, de forma que esta nueva capa oculte todas las dificultades de la gestión de las múltiples direcciones a aquellas aplicaciones que son incapaces de hacerlo. Para ello, la nueva capa debería utilizar un identificador único para aquellos extremos que posean múltiples direcciones. Esto surge del hecho de que actualmente las aplicaciones utilizan las direcciones IP como identificadores de los extremos, como se ha presentado en la sección inicial. Por ello, las aplicaciones abren *sockets* hacia una dirección IP, pudiendo incluso elegir cuál es la dirección IP propia que desean utilizar para una comunicación en particular. Además las conexiones de transporte también utilizan la dirección IP para su identificación. Por ello, para hacer transparente la existencia de las múltiples direcciones, la nueva capa de sesión deberá presentar siempre el mismo identificador para la sesión establecida, independientemente de la dirección IP usada en los paquetes y de las direcciones IP usadas en las conexiones de transporte. Para esto sería necesario que la capa de sesión utilizara un nuevo identificador, que llamaríamos identificador de sesión, independiente de las direcciones IP usadas para transportar los paquetes. Un beneficio que presenta la adopción de una nueva capa de sesión es que resultaría posible la utilización de múltiples capas de transporte para una misma sesión. La dificultad que presenta este enfoque es que la nueva capa de sesión debería resolver muchos de los problemas que ya resuelven algunas de las capas de transporte existentes, en lo que se refiere a la perdida y reordenamiento de paquetes. En definitiva, este enfoque ofrece la ventaja de resolver el problema para todas las aplicaciones, y además es capaz de utilizar simultáneamente distintas tecnologías de transporte en una misma sesión, pero requiere la implementación de funcionalidades complejas ya existentes en otras partes de la pila de protocolos, como puede ser la capa de transporte TCP.

4.2.2.1.3 Capa de transporte

Otra posibilidad es modificar las capas de transporte de forma que múltiples direcciones puedan ser utilizadas en una misma conexión. Existen al menos dos capas de transporte que ofrecen algún tipo de soporte de múltiples direcciones, a saber SCTP [RFC2960] y DCCP

[Kohler2004a]. Estas capas de transporte son relativamente nuevas y están diseñadas para usar múltiples direcciones en una misma conexión. La dificultad que presenta este enfoque es que es necesario modificar todas y cada una de las capas de transporte para brindar soporte universal de multihoming. En particular es necesario brindar soporte a las capas existentes, como son TCP y UDP. En el caso de TCP, es necesario modificar la implementación y tal vez el protocolo en sí mismo para soportar las múltiples direcciones. En el caso de UDP, la complejidad es mayor ya que ni siquiera existe el concepto de conexión a nivel UDP, por lo que resulta difícil, que la propia capa UDP tenga conocimiento de cuándo es posible intercambiar las direcciones usadas. Sería posible tener una idea de conexión cuando se utilizan *connected sockets,* pero el soporte de *unconnected sockets* resulta complejo.

4.2.2.1.4 Capa de identificación

Considerando que la causa fundamental del problema radica esencialmente en los múltiples roles que juega la dirección IP en la arquitectura TCP/IP, en particular por la tensión generada entre los requerimientos de dependencia topológica impuesto por el rol de localizador y el requerimiento de invarianza con respecto al camino usado para alcanzar el extremo impuesto por el rol de identificador de extremo final, un enfoque posible es la creación de una nueva capa de identificación por encima de las funciones de reenvío de la capa IP que provea un identificador de extremo en la arquitectura IP. Esta capa esencialmente establecería una correspondencia entre todos los localizadores topológicamente dependientes de un extremo (usados para el encaminamiento y reenvío de paquetes) y un único identificador de extremo independiente de la topología (usados para la identificación del extremo remoto). La capa de identificación tendría conocimiento de todos los localizadores asociados a un extremo y los utilizaría para enviar/recibir paquetes mientras presenta siempre el mismo identificador a las capas superiores de la arquitectura. Este enfoque provee una solución única para todas las capas de transporte pero implica una modificación mayor de la arquitectura TCP/IP.

4.2.2.2 Clasificación del vínculo según el elemento de la red

El otro plano que afecta a la realización del vínculo entre las múltiples direcciones se refiere al elemento o sistema de la red que lo establece. Una posibilidad es que el vínculo se establezca en el extremo final de la comunicación. Este parece ser el enfoque más natural, ya que el extremo tiene conocimiento de sus propios localizadores y del conjunto de localizadores del otro extremo de la comunicación, y gestiona ambos conjuntos de acuerdo a sus necesidades. Este enfoque, sin embargo, implica la modificación de los nodos, lo que dificulta su adopción. Adicionalmente, es posible que el extremo no disponga de toda la información necesaria para elegir el mejor localizador a usar en cada paquete, ya que, por ejemplo, carece de información de

encaminamiento. Finalmente, este enfoque impone la exposición de los nodos internos del sitio con múltiples proveedores a localizadores que serán dependientes de la topología, implicando que si hay un cambio en la topología y los localizadores deben ser cambiados (por ejemplo debido a un cambio de proveedor), este cambio impactará en todos los nodos del sitio multihomed, requiriendo modificaciones en las configuraciones de los nodos internos al sitio.

Un enfoque alternativo es establecer el vínculo en un elemento intermedio, como pueden ser los routers de borde, de forma que se oculte al extremo final la existencia de las múltiples direcciones. Esto implica una reducción en el número de dispositivos que deben proveer el soporte de multihoming ya que sólo estos nodos intermedios deben ser modificados. Sin embargo, es necesario tener especial cuidado en el diseño de una solución de este tipo, ya que este enfoque puede presentar limitaciones similares a las del NAT [RFC2993] [RFC1631], ya que las direcciones son utilizadas no solamente en los encabezados de los protocolos sino también por las aplicaciones como identificadores de los extremos. Para evitar esto, es necesario que el paquete sea presentado al extremo que lo recibe tal y como fue generado por el extremo emisor, por lo que si, a modo de ejemplo, el router de borde en el extremo emisor modifica el paquete para incluir un localizador apropiado, otro dispositivo en el camino debe deshacer el cambio antes de que el paquete sea entregado al extremo. Otra dificultad que presenta este enfoque se refiere a la escalabilidad, ya que por la propia naturaleza de la solución, unos pocos dispositivos gestionarán las direcciones de un número mayor de extremos. Finalmente, otra dificultad potencial es la seguridad ya que será necesario que una tercera parte gestione diversos localizadores y modifique los paquetes en curso. Esto puede habilitar ataques, ya que esencialmente el router que manipula y modifica los paquetes está actuando como un atacante posicionado en el medio del camino. Será necesario entonces, incluir las medidas necesarias para poder distinguir las manipulaciones legítimas realizadas por los routers de los ataques.

Finalmente, es posible adoptar un enfoque híbrido, donde parte de los mecanismos residen en el extremo final, por ejemplo la gestión de la seguridad, mientras un nodo intermedio gestiona los distintos localizadores. Si bien esta solución anula ciertas ventajas de los enfoques anteriores, por ejemplo en lo que refiere al esfuerzo de adopción (este enfoque implica la modificación tanto de los extremo como de los routers de borde), este enfoque puede resultar óptimo en el sentido que cada elemento realizará la tarea para la cual es más idóneo.

4.2.2.3 Espacio de nombres

Como hemos visto en el análisis realizado anteriormente, las soluciones esencialmente consisten en que en alguna parte de la pila de protocolos de algún dispositivo de la red, una entidad establezca y gestione el vínculo existente entre las diversas direcciones asociadas a un mismo extremo de forma transparente. Para ello es necesario presentar los paquetes provenientes

de direcciones IP distintas como paquetes originados por el mismo extremo. Una posibilidad y probablemente la mas natural, es sustituir las distintas direcciones IP contenidas en los paquetes por otra cadena de 128 bits constante asociada al extremo origen. Esta sustitución puede realizarse en cualquiera de los niveles estudiados y por cualquiera de los elementos considerados en el análisis. En esencia, esto significa usar un espacio de nombres de identificadores de extremos de forma que a cada extremo se le asignaría uno o más identificadores. La opción de usar cadenas de 128 bits resulta natural ya que tanto las distintas capas de protocolos como las aplicaciones utilizan cadenas de 128 bits (las direcciones IP) directamente. También es posible utilizar otros espacios de nombres. Por ello, primero se analizará la posibilidad de usar espacios de 128 bits y luego otros enfoque alternativos.

4.2.2.3.1 Identificadores de 128 bits

El uso de identificadores de 128 bits resulta natural ya que la arquitectura actual utiliza la dirección IP como identificador de extremos, por lo que tanto las capas de transporte como las aplicaciones son capaces de manejarlos. Esto significa esencialmente que no se requieren cambios en las APIs o protocolos y que la adopción del nuevo espacio de nombres de identificadores puede limitarse a la propia entidad que hace la traducción de identificador a localizador.

A continuación describiremos distintas propiedades de los espacios de nombres que deben ser consideradas en el momento de la elección:

- **Identificadores estables versus identificadores efímeros**

Es posible considerar tanto el uso de identificadores *estables*, cuya vida útil es suficientemente larga como para que sea razonable esperar que en un futuro cercano el extremo siga siendo identificado por el mismo identificador, como el uso de identificadores *efímeros*, cuyo tiempo de vida esté bastante limitado, por ejemplo, al tiempo de vida de una comunicación. Cabe notar que la vida de un identificador estable no tiene porque ser infinita, ni siquiera ser comparable a la vida útil del dispositivo en cuestión, sino que simplemente debe ser válido durante un período compatible con usos futuros del identificador. Por ejemplo, los nombres de dominio son lo que llamaríamos nombres estables, ya que es esperable que permanezcan estables para su futuro uso (es posible también que un momento dado el nombre de dominio cambie). Un ejemplo claro de identificadores estables e identificadores inestables son los puertos de TCP. Por ejemplo, el puerto donde atiende el servidor SMTP (puerto 25) es un nombre estable ya que esta fijo y es esperable que el servidor continúe escuchando en ese puerto mucho tiempo después. Sin embargo, el puerto origen usado por una conexión cualquiera TCP será elegido al azar y sólo será válido durante la vida de la conexión TCP. No es esperable que el proceso que lo ha utilizado en una conexión en particular se encuentre disponible en dicho puerto en el futuro. El uso de identificadores estables simplifica el contacto inicial ya que es posible conocerlos a priori.

Una solución basada en identificadores efímeros requiere una negociación previa al uso del identificador para definir el identificador efímero a usar, lo que añade probablemente latencia en el establecimiento de la comunicación Es decir, supongamos que un nodo A desea comunicarse con un nodo B y se utilizan identificadores efímeros. En un primer paso, es necesario que los nodos se pongan en contacto y acuerden el identificador, para que luego ya sea posible establecer la comunicación sabiendo qué identificadores serán usados en la comunicación. Adicionalmente, los identificadores estables hacen posible las referencias, es decir que un nodo sea capaz de referirse a otro nodo sobre un tercero (o sobre sí mismo). Por ejemplo, en FTP (*File Transfer Protocol*) el nodo que inicia la comunicación informa al nodo que recibe la comunicación de la dirección IP en la que se encuentra, de forma que el nodo receptor inicie a su vez una comunicación de transferencia de datos. En este caso, el nodo que inicia la comunicación hace una referencia a sí mismo para el futuro (llamada *call-back*). Si el identificador es efímero, es posible que esto ya no funcione, ya que el identificador no será válido. Por esto, ciertos usos de los identificadores se ven simplificados si se usan identificadores estables. Sin embargo, esto trae aparejado un costo, ya que la seguridad de un identificador estable debe ser defendida, como se presenta a continuación.

En el caso de un identificador efímero, éste sólo tiene valor mientras esta siendo usado en una comunicación y una vez finalizada, es identificador carece de valor. Por ello, el identificador sólo requiere ser protegido de ataques o robos durante ese limitado periodo de tiempo, por lo que resulta simple brindar la seguridad necesaria. En el contexto de multihoming, la funcionalidad básica provista por este identificador es permitir reconocer a un mismo extremo a través de cambios en las direcciones IP a lo largo de una comunicación.

En el caso de un identificador estable, el identificador representa la identidad del extremo por lo que es necesario protegerlo de robos y falsificaciones, ya que es posible suplantar al extremo si se posee el identificador. Es decir un identificador estable será usado para iniciar el contacto con un extremo deseado, y también puede ser utilizado para reconocer a un extremo a lo largo de contactos sucesivos. Por ello, quien pueda probar que posee un identificador en particular puede hacerse pasar por el extremo que lo tiene asociado. Es decir, un identificador estable está asociado con una identidad de un extremo, por lo que es necesario disponer de las medidas de seguridad necesarias para protegerlo de posibles ataques. Cabe notar que es posible contar con los dos tipos de identificadores y utilizar en cada momento el que se adapte a las necesidades particulares de una comunicación concreta.

- **Alcance del identificador**

Existen distintos alcances de aplicación de los identificadores usados. En particular, es posible que el alcance de los identificadores sea *global*, es decir que el identificador sea único (para alguna de las definiciones de unicidad presentadas más abajo) en toda Internet, lo que implica que el identificador identifica a un único extremo en toda Internet. Por otro lado, podemos usar

identificadores de alcance *local*, donde el identificador sólo identifica un extremo dentro de los extremos involucrados en una comunicación en particular, es decir que sólo es único entre los elementos involucrados en la comunicación. Es posible utilizar alcances intermedios, como por ejemplo identificadores que son únicos dentro de un grupo de extremos, por ejemplo, dentro de los extremos conectados a un sitio en particular (como ocurre en IPv4 para las direcciones privadas). Queda claro que a medida que el alcance del identificador es mayor, más costoso es asegurar su unicidad. Por otro lado, dado que un identificador sólo tiene sentido dentro de su ámbito, tanto el dominio de referencia como el de búsqueda estarán acotados por el alcance del identificador.

- **Unicidad estadística versus unicidad administrativa**

Un identificador debe ser único en un momento dado y dentro de un alcance, como hemos visto en los puntos anteriores. Existen dos tipos de unicidad: la unicidad administrativa y la unicidad estadística. La unicidad *administrativa* se basa en la existencia de una entidad administradora del espacio de nombres que asegura que los nombres no se repiten. La obtención de un nombre implica entonces la solicitud de un nombre a la entidad en cuestión. La otra posibilidad es utilizar nombres con unicidad *estadística*, para lo que los identificadores son elegidos al azar de un espacio muy grande de nombres, de forma que la probabilidad de colisión sea menor que un valor dado y considerado aceptable [Bagnulo2002c]. En este caso, el identificador puede ser generado localmente sin requerir el contacto con ninguna entidad externa.

Cabe notar que tanto en el caso de la unicidad administrativa, como en el caso de la unicidad estadística, existe una probabilidad no nula de colisión. En el segundo caso esto es inherente al espacio de nombres y en el primer caso las colisiones se deben a errores realizados en la administración del espacio de nombres. La probabilidad de colisión en el caso de la unicidad estadística puede hacerse todo lo pequeño que se desee, pero esto implica utilizar un espacio de nombres mayor. Por otra parte, para asegurar la unicidad administrativa es necesario contar con una entidad administradora, lo que impone costos adicionales.

- **Identificadores agregables versus identificadores no agregables**

Un punto estrechamente vinculado al punto anterior se refiere a la posibilidad de agregar identificadores, es decir la posibilidad de denotar de forma compacta a un grupo de identificadores que cumplan con una cierta propiedad, como es posible hacer con las direcciones IP. En particular, la capacidad de agregación facilita el proceso de búsqueda de un localizador, ya que habilita la organización jerárquica del espacio de identificadores. Asimismo, la capacidad de agregación de identificadores simplifica la configuración de listas de acceso, filtros y otro tipo de mecanismos de selección basado en la pertenencia a un grupo dado. Por ejemplo, en el caso que el espacio de identificadores sea agregable, permitiría asignar un grupo de identificadores a una organización para que ésta los administrara internamente. Esto permite que la organización defina el acceso a sus recursos internos basándose en la pertenecía a dicho grupo de identificadores.

Adicionalmente, es posible que dicha organización se haga cargo de la búsqueda y resolución de la parte que le fuera asignada del espacio de nombres de identificadores, de forma que para buscar información asociada a cualquiera de los identificadores de dicho grupo, sería necesario contactar con la organización en cuestión, jerarquizando la búsqueda. Como ejemplo de un espacio de nombres que funciona organizado de forma jerárquica, podemos recordar el sistema de nombres de dominio (DNS).

- **Identificadores anónimos**

Otra propiedad que puede ser relevante en los identificadores es la capacidad de soportar anonimato. Es decir que sea posible obtener un identificador sin que ello implique revelar la identidad de quien lo solicita. Esta propiedad está relacionada con la posibilidad de crear el identificador localmente, ya que si esto fuera posible, resultaría más simple proveer el anonimato que en el caso donde se requiere solicitar los identificadores a una entidad central.

Una vez analizadas las distintas propiedades a tener en cuenta cuando se consideran los distintos espacios de identificadores posibles, pasaremos a evaluar algunos casos particulares que se han propuesto y que presentan características atractivas para el soporte de multihoming basado en multidireccionamiento.

- **Direcciones**

Una posibilidad es simplemente reservar una parte del espacio de direccionamiento de IPv6 para ser usado como identificadores, como por ejemplo un cierto prefijo. Cabe notar que la asignación de los identificadores no requiere tantos niveles de jerarquía como el espacio de localizadores, ya que probablemente sólo se requieran unos pocos niveles de agregación, por ejemplo, por organización, tal vez, por país, pero no tantos niveles como los que requieren las direcciones actuales que utilizan la agregación para facilitar el encaminamiento. Esto hace que la utilización del espacio sea mucho más eficiente que en el caso de las direcciones donde la utilización efectiva del espacio se ve afectada por la influencia de los múltiples niveles de jerarquía en la topología, como intentan recoger los criterios de dimensionamiento basados en la regla del 80% de utilización o el HD ratio [RFC3194] usados en las políticas actuales de asignación de direcciones en los distintos RIRs[13]. Cabe notar que, durante el periodo de adopción del nuevo espacio de identificadores, los extremos deberán lidiar tanto con los nuevos identificadores como con las direcciones IP (localizadores). Por ello, es posible que resulte atractivo e incluso necesario distinguir las direcciones IP que son usadas como localizadores de los nuevos identificadores de

[13] Las políticas de asignación de direcciones pueden encontrarse en los sitios web de cada uno de los RIRs detallados anteriormente.

128 bits. Para ello, será necesario reservar una fracción del espacio de direccionamiento de IP para alojar el nuevo espacio de identificadores.

- **Localizadores**

Otra posibilidad es utilizar localizadores como identificadores, tal y como lo hacemos hoy. En este caso, uno de los localizadores del extremo es elegido como el identificador (ya sea para una comunicación en particular o para siempre). De forma que independientemente de cuál sea el localizador contenido en el paquete IP, siempre se presenta a las capas superiores el localizador usado como identificador, de forma que éste permanezca constante. La posibilidad más natural probablemente sea utilizar los localizadores contenidos en el primer paquete como identificadores.

La ventaja de este enfoque es que mientras los localizadores no cambien, no es necesario realizar operaciones ni traducciones adicionales. Diversas propuestas utilizan esta técnica, como MIP [RFC3775] para el soporte de movilidad y NOID [Nordmark2004a] para el soporte de multihoming.

En la arquitectura de direccionamiento actual, existen localizadores de distintos alcances. Sin embargo, para nuestro estudio es relevante distinguir entre el dominio de encaminamiento del localizador y el dominio de unicidad del mismo. Existen direcciones, como pueden ser las link local, que tanto su dominio de encaminamiento como su dominio de unicidad esta limitado al enlace, por lo que no parecen atractivas para ser usadas como identificadores. Por ello, estas direcciones no resultan apropiadas para ser utilizadas como identificadores fuera de ese dominio. Sin embargo, las direcciones únicas locales [Hinden2005a], son direcciones cuyo dominio de encaminamiento es limitado, pero son únicas globalmente. Por ello, estas direcciones no pueden ser usadas como localizadores en comunicaciones que van más allá de su alcance natural de encaminamiento, pero sí pueden ser utilizadas como identificadores en comunicaciones globales, ya que son globalmente únicas. Finalmente, las direcciones globales que son globalmente únicas y globalmente encaminables pueden ser usadas como identificadores. Una diferencia relevante entre los localizadores globales y los locales es la dependencia con el proveedor de servicio. En el caso de las direcciones locales, estas son independientes del proveedor, por lo que si el sitio cambia de ISP, podría conservar sus direcciones locales. Sin embargo esto no es así en el caso de las direcciones globales, ya que si el sitio cambia de ISP, deberá también devolver las direcciones al ISP y deberá cambiar de localizadores y por ende los identificadores. Este proceso puede ser costoso, dependiendo de la forma de la solución final adoptada.

En síntesis, los localizadores globales presentan las siguientes características:

- Son estables
- Son de alcance global (si usamos direcciones globalmente únicas como pueden ser las direcciones globales o las direcciones locales únicas)

- Son administrativamente únicos, ya que son asignadas a través de la cadena IANA-RIR-LIR
- Son agregables, ya que son asignados por prefijos. Esto quiere decir que es posible hacer referencia al conjunto de los identificadores asignados a un sitio a través del prefijo común que estos comparten. Adicionalmente, existe ya un sistema que establece correspondencias entre estos localizadores y otros valores basándose en la resolución inversa de nombres. Esto permite que se descubran valores asociados al identificador a través de la búsqueda en el DNS inverso.
- No soportan anonimato

- **Identificadores aleatorios**

Otra posibilidad es generar los identificadores de forma aleatoria, es decir que cada extremo genere un número aleatorio de 128 bits y lo utilice como identificador. El tamaño del espacio de nombres, 2^128 es suficientemente grande como para poder asegurar unicidad estadística a nivel global, ya que de acuerdo a la resolución de la paradoja del cumpleaños, si tenemos una población de 9x10^6 [Moskowits2005a] que ha generado identificadores 128 bits de forma aleatoria, la probabilidad de colisión es de .00001, lo que parece un valor razonable. Por ello, se puede justificar la opción de que cada extremo utilice un número aleatorio de 128 bits como identificador. Las características de estos identificadores serán las siguientes:

- Pueden ser tanto estables como efímeros.
- Pueden ser de alcance global o local
- Son estadísticamente únicos
- No son agregables, ya que el los identificadores asignados a una organización no tienen ninguna correlación entre si.
- Son generados localmente, sin necesidad de contactar con una entidad exterior
- Soportan anonimato

- **Identificadores criptográficos**

Un caso particular de identificadores generados aleatoriamente son los *identificadores criptográficos*, en los que el identificador contiene información de naturaleza criptográfica.

Un ejemplo de este tipo de identificadores puede encontrase en SIM [Nordmark2003a]. En este caso, los identificadores presentados a las capas superiores corresponden a un hash de una clave pública de 128 bits. El proceso de generación del identificador en este caso sería: primero, el extremo genera un par de claves pública y privada y genera un hash de la clave pública de tamaño 128 bits. El resultado es el identificador de 128 bits. El identificador así generado es aleatorio, por lo que cumple con las propiedades descritas en el apartado anterior. Adicionalmente, el identificador posee una naturaleza criptográfica intrínseca que permite verificar la propiedad (en el sentido de derecho de posesión) del identificador. En este contexto, si deseamos verificar que quien está en el otro extremo es el propietario del identificador, lo que debemos hacer es solicitar

que dicho extremo cifre con la clave privada un desafío y nos envíe el resultado, junto con la clave pública usada para generar el identificador. Para verificar la propiedad del identificador, primero generamos el hash de la clave pública y verificamos que coincida con el identificador en cuestión y después verificamos que al descifrar el resultado recibido usando dicha clave pública, éste coincida con el desafío enviado. Esta verificación nos permite confirmar que el otro extremo posee la clave privada asociada a la clave pública que fue utilizada para generar el identificador. Como veremos en el análisis de seguridad realizado más adelante, esta propiedad puede ser útil para proveer la seguridad requerida. Los identificadores resultantes tienen las mismas propiedades que otros identificadores generados aleatoriamente, pero además ofrecen esta propiedad adicional en lo que se refiere a la seguridad.

- **Identificadores aleatorios jerárquicos**

Como hemos visto, una limitación importante de los identificadores generados aleatoriamente es que no es posible agregarlos, lo que dificulta su manejo. Por ejemplo, esto hace difícil realizar una búsqueda en el espacio de los identificadores. Dicha búsqueda es necesaria para encontrar los localizadores asociados a un identificador dado, entre otras cosas. Una posibilidad es entonces establecer una jerarquía, creando el identificador como una concatenación de múltiples cadenas de bits. Por ejemplo, el identificador podría estar formado como dos cadenas de 64 bits, donde la primera cadena identificaría la institución y la segunda cadena identificaría un extremo en particular dentro de esta institución. De esta forma, para realizar una búsqueda de un identificador primero resolvemos la parte de la institución, lo que nos llevaría al servicio de directorio propio de la institución en cuestión y después dentro del directorio de esta institución buscaremos el identificador deseado. El mecanismo es similar al actualmente usado por el sistema de nombres de domino [RFC1034]. Esto hace que no sea necesario crear un sistema de búsqueda en un espacio plano de grandes dimensiones, que podría ser particularmente difícil. En particular, se podrían generar identificadores criptográficos jerárquicos, donde los primeros 64 bits del identificador correspondan al hash de una clave pública asociada al sitio y los últimos 64 bits sean el hash de una clave asociada al extremo en cuestión. Esto permitiría disponer de una clave para el sitio, que permitiría al administrador realizar ciertas operaciones en nombre de todo el sitio, como por ejemplo insertar de forma autenticada registros correspondientes al sitio en una base de datos global.

4.2.2.3.2 Identificadores de 64 bits

La arquitectura de direccionamiento de IPv6 [RFC3513] define que todas las direcciones que no empiecen con los valores binarios 000 deben contener un Identificador de Interfaz construido usando el formato EUI modificado contenido en dicha especificación en los últimos 64 bits. Esto implica a efectos prácticos que cada dirección IPv6 contiene un localizador en los primeros 64 bits y un identificador en los últimos 64 bits. Por ello, una posibilidad es utilizar el identificador

de interfaz como identificador del extremo, como es propuesto por LIN [Teraoka2003a], GSE [Odell1997a]. Este enfoque presenta algunas ventajas, como que cada vez que se utiliza una dirección IP, ésta contiene tanto el identificador como un localizador asociado al mismo. Esto implica por ejemplo que los paquetes IPv6 contendrían tanto el localizador como el identificador, lo que permitiría tomar decisiones en función de cualquiera de ellos. Es más, sería posible modificar el localizador manteniendo el identificador de forma trivial, lo que puede ser una ventaja, aunque también una vulnerabilidad potencial en la seguridad como veremos más adelante.

Los identificadores de 64 bits pueden ser de diversa naturaleza, análogamente al caso de 128 bits. Una posibilidad es crear una entidad que administre este espacio y asigne bloques de identificadores del espacio existente. Otra posibilidad es utilizar identificadores estadísticamente únicos. Sin embargo la dificultad que presenta este enfoque es que el tamaño del espacio (2^64) no parece ser suficiente para asegurar una probabilidad de colisión suficientemente baja para una población de Internet alta, ya que, a modo de ejemplo, existe una probabilidad de colisión de .5 en una población de 2^32 cuando usamos un espacio de 2^64. Sin embargo cabe notar que es posible renunciar a la parte menos significativa del espacio destinado a la localización para extender el espacio del identificador; en particular, es posible incluir el prefijo de subred (16 bits menos significativos del prefijo /64), lo que elevaría a 80 el número de bits del espacio de nombres a considerar.

También se ha propuesto el uso de identificadores criptográficos de 64 bits como es el caso de CGA [RFC3972] y CB64 [Nordmark2003b], que sufren de esta misma limitación, con la dificultad adicional de que además son más vulnerables que los identificadores de 128 bits a ataques por fuerza bruta donde se puede descubrir una clave pública cuyo hash coincida con un identificador dado. No obstante, los ataques por fuerza bruta pueden ser contrarrestados imponiendo condiciones específicas al identificador que dificultan su generación, incrementando el coste de los ataques, tal y como se describe en [RFC3972].

4.2.2.3.3 Otros espacios de nombres

Es posible proponer el uso de nuevos espacios de identificadores de longitud distinta a 128 bits que sustituyan a las direcciones IP. Por ejemplo, se puede plantear el uso de los nombres de dominio [RFC1034] como identificadores. Está claro que esto implicaría cambios en las APIs y posiblemente en los protocolos, pero, también es patente que los nombres de dominio presentan múltiples ventajas como su fácil comprensión para los seres humanos como representación de un extremo

Otra posibilidad es utilizar directamente las claves públicas como identificadores y no utilizar un hash de las mismas, que es lo que propone HIP [Moskowits2005a] (siendo el hash una mera herramienta para la transición hacia el nuevo espacio de identificación). Este enfoque

provee mayor seguridad ya que el enfoque basado en el hash es susceptible a ataques sobre éste, que tiene un largo fijo.

Para soluciones a nivel de TCP, es posible pensar en el uso de los puertos como identificadores de las conexiones, independientemente de las direcciones IP usadas por los extremos. En este caso, el puerto pasa a ser un identificador de conexión efímero y de alcance local a los dos extremos involucrados en la conexión.

4.2.2.3.4 Usos de los identificadores

Como hemos visto, la motivación fundamental para adoptar un nuevo espacio de nombres es poder vincular los distintos localizadores disponibles en un extremo, de modo que se pueda mostrar a las capas superiores un único nombre del extremo a pesar de que los localizadores usados durante la comunicación cambien. Sin embargo, un identificador también es usado para otras múltiples funciones por lo que es necesario considerarlas cuando se evalúan los potenciales espacios de nombres. A continuación analizaremos los distintos usos que se hacen de las direcciones IP cuando son usadas como identificadores y evaluaremos la idoneidad de los distintos espacios de nombres posibles para ejecutarlas.

- **Continuidad de una comunicación establecida a través de cambio en los localizadores.** Como hemos mencionado, un objetivo fundamental del uso de los identificadores es preservar un nombre constante a lo largo de los cambios en los localizadores usados para encaminar los paquetes durante la vida de una comunicación dada. Para esto, es necesario sustituir los localizadores cambiantes por un identificador constante. El ejemplo más simple de esto es una aplicación cliente que se conecta a un servidor y a lo largo de la vida de la conexión el localizador usado por el cliente cambia. Es necesario que alguna entidad en el extremo del servidor sustituya los localizadores cambiantes por un identificador fijo que permita reconocer que la comunicación se mantiene siempre con el mismo extremo. Para esto solamente es necesario que el identificador sea constante durante el tiempo de vida de la comunicación, por lo que es suficiente con un identificador efímero con una vida mayor o igual al tiempo de la comunicación. El identificador puede ser cualquier cadena de bits y sólo es necesario que sea único para los nodos involucrados en la comunicación.
- **Identificadores para el contacto inicial.** Cuando una comunicación es iniciada, es necesario contar con un medio para identificar al interlocutor deseado. Es decir, cuando un extremo inicia una comunicación, desea iniciarla con un extremo dado. Por ello, es necesario contar con un identificador que permita especificar el interlocutor deseado para iniciar una comunicación. En otras palabras, un extremo que juega el rol de servidor necesita un identificador que permita a los potenciales clientes referirse a él. En este caso, el identificador representa la identidad del extremo. En la arquitectura actual, los

servidores se encuentran en direcciones IP conocidas a priori y los clientes utilizan esas direcciones IP para contactarlos. En este caso, un identificador efímero es insuficiente y es necesario un identificador estable que permita identificar el extremo a lo largo del tiempo.

Es interesante notar que ya se distingue en la actualidad entre estos dos tipos de roles del identificador. Es usual que los clientes obtengan identificadores (direcciones IP) efímeros a través de DHCP [RFC3315] cuya vida está limitada. Es más, los clientes que se encuentran detrás de un NAT [RFC1631] obtienen un identificador global que no tiene por qué durar más que el tiempo de vida de la comunicación en cuestión. Sin embargo, los servidores poseen identificadores estables (direcciones IP fijas) que les permite ser alcanzables por los clientes. Resulta claro entonces que los identificadores estables brindan funcionalidades adicionales y que son necesarias para ciertos usos esperados de los identificadores. Sin embargo, estas funcionalidades adicionales no vienen sin un costo asociado. El identificador estable representa la identidad del nodo, mientras que el identificador efímero es menos significativo. Por ello, es necesario defender los identificadores estables, de forma que un atacante no pueda apoderarse de la identidad que el identificador representa. Esta defensa es menos relevante en el caso del identificador efímero.

- **Llamadas inversas** (*Call back*). Ciertas aplicaciones realizan llamadas inversas, es decir que inicialmente un nodo A inicia una comunicación con el nodo B. El nodo A informa al nodo B de su identidad y luego, pasado un cierto tiempo, es el nodo B quien inicia la comunicación con el nodo A. Un ejemplo es el *File Transfer Protocol* (FTP). Para esto, es necesario que el identificador del nodo A (el nodo cliente) sea válido durante un tiempo, para que el nodo B pueda contactarlo usando el mismo identificador. En este caso, también es necesario disponer de identificadores estables y los identificadores efímeros no parecen ser suficientes para este uso.
- **Reconocimiento:** Análogamente, también es posible utilizar los identificadores para reconocer a un extremo con el que se ha mantenido una comunicación anteriormente. Este uso del identificador impone las mismas restricciones que el uso de llamadas inversas.
- **Referencias** (*Refferals*) Otro uso que hacen las aplicaciones actuales de los identificadores es lo que comúnmente se conoce como *referencias*. En este caso, tenemos un nodo A que se comunica con el nodo B y en esta comunicación, el nodo B se refiere a un nodo C. Por ejemplo, supongamos que el nodo A busca cierta información, y contacta con el nodo B. Ahora bien, el nodo B no tiene la información que busca el nodo A, pero sabe que el nodo C sí posee dicha información. Entonces, el nodo B informa al nodo A que la información buscada esta disponible en el nodo C. En este caso, el nodo B

enviará el identificador del nodo C al nodo A. Nuevamente, en este caso, el identificador del nodo C deberá tener una vida suficientemente larga como para seguir siendo válido cuando A quiera contactarlo. Para este uso, necesitamos identificadores estables y los identificadores efímeros no brindan las funcionalidades necesarias para este caso.

- **Coexistencia de los espacios de identificadores:** Cabe notar además que en el caso de una comunicación directa entre dos partes es posible determinar si ambas partes soportan un nuevo espacio de identificadores durante la fase de negociación inicial al comienzo de la comunicación, de forma que si uno de ellos no soporta el nuevo espacio de identificadores, se utilizarán direcciones IP de la forma tradicional. Sin embargo, en el caso de las referencias, un identificador usado en la comunicación entre dos nodos que soportan el nuevo espacio de identificadores puede terminar siendo enviado a un tercer nodo que no soporte el nuevo espacio de identificadores. En el ejemplo anterior en el punto de las referencias, si B y C soportan el nuevo espacio de identificadores, y A no lo soporta, cuando A obtenga el identificador de C a través de B, A no podrá utilizarlo, ya que no soporta el nuevo espacio de identificadores. Por ende, la única forma de soportar las referencias de forma compatible hacia atrás es que los identificadores sean localizadores validos (tal y como ocurre con los identificadores usados actualmente), de forma que si son recibidos por nodos que no soportan el nuevo espacio de identificadores, estos sean de todas formas capaces de enviar paquetes al nodo identificado por el identificador recibido, ya que será también un localizador válido (como lo es actualmente)

4.2.2.3.5 Propiedades de los identificadores asumidas por las aplicaciones

Actualmente, las aplicaciones asumen ciertas propiedades de los identificadores utilizados por ellas para la comunicación con otros extremos a través de la red. En particular, como hemos visto, muchas aplicaciones asumen que el identificador usado no cambia a lo largo de la vida de la comunicación. Adicionalmente, las aplicaciones que utilizan directamente direcciones IP para identificar a los nodos de la red, en lugar de usar los nombres de dominio, asumen que este identificador será siempre válido. Esto esencialmente supone dos cosas: primero, que el identificador es estable, es decir que el nodo no cambia de identificador con el tiempo (este punto en particular ha sido ilustrado en la discusión sobre identificadores efímeros). Segundo, esto también supone que las capas inferiores siempre serán capaces de enviar paquetes dirigidos al identificador usado. Esto implica no sólo que el identificador es válido, sino que las capas inferiores disponen de un conjunto de localizadores asociados al identificador. Las aplicaciones asumen esto ya que, en la arquitectura actual, donde el identificador coincide con un localizador válido, las capas inferiores (el particular la capa IP) siempre pueden enviar paquetes al extremo con los mismos identificadores usados por las aplicaciones.

En el caso que el identificador y el localizador sean distintos, es necesario entonces que la capa que implementa la relación entre el identificador y los localizadores preserve dicha relación durante el tiempo en el que la aplicación use el identificador. El problema es que este tiempo no es acotado y es imposible para las capas inferiores saber hasta cuándo la aplicación planea enviar paquetes a un cierto identificador. Es cierto que si la aplicación utiliza conexiones TCP por ejemplo, el hecho que la conexión se cierre puede ser un indicio de que la aplicación ha terminado de comunicarse con cierto identificador. Sin embargo, esto no es más que un indicio y no un hecho concluyente. Además existen aplicaciones que utilizan UDP, que no incorpora el concepto de conexión. Por otra parte, resultaría imposible mantener información sobre todos los identificadores y sus respectivos localizadores de forma indefinida, ya que esto implicaría un gasto excesivo de memoria. Por ende, es necesario que la capa que relaciona los identificadores con sus localizadores posea mecanismos de recolección de basura que borren la información concerniente a identificadores que no están siendo usados. Esta tarea no es trivial, ya que no es posible que esta capa tenga conocimiento referente al uso futuro de un identificador por parte de las aplicaciones. En el caso en el que es posible obtener la información de localizadores a partir del identificador, la situación no parece muy crítica, ya que el estado puede ser recuperado a partir del identificador. Sin embargo, si no es posible obtener el conjunto de localizadores a partir del identificador, es imposible recuperar el estado borrado, por lo que no será posible establecer una comunicación con el identificador. En este caso, el resultado es que la aplicación fallará si se descarta prematuramente (es decir, en tiempo menor al tiempo de uso de identificador por parte de la aplicación, es decir un tiempo desconocido y no acotado) la información de localizadores asociada al identificador. En conclusión, para preservar las asunciones realizadas por las aplicaciones actuales, será necesario que el identificador sea estable y que además sea posible obtener los localizadores asociados al mismo.

4.2.2.3.6 Resolución de identificadores en localizadores

Si bien el papel fundamental de un identificador es, como su propio nombre indica, identificar de forma única un extremo dentro de un dominio dado (temporal y espacial), el objetivo final de quien requiere el identificador de un tercero es en muchos casos la comunicación con este. Por ello, no basta con poder identificar al extremo, sino que es necesario obtener el conjunto de localizadores válidos para alcanzar al extremo identificado. Por ello, es necesario un mecanismo que nos permita obtener los localizadores a partir del identificador. En la arquitectura actual, el identificador y el localizador son uno, por lo que cuando sabemos uno automáticamente sabemos el otro. Sin embargo, si separamos la función de identificación de la función de localización, la situación requiere un análisis más detallado

A continuación analizaremos las posibilidades para la resolución de los distintos tipos de identificadores que han sido considerados.

Localizadores

Consideremos el caso en donde un localizador válido del extremo es usado como identificador. Dado que el identificador es un localizador válido, quien conoce el identificador también conoce un localizador. El problema es que ese localizador en cuestión puede que no esté disponible en un momento dado, y sea necesario descubrir otros localizadores a partir de éste. Dicho sistema de búsqueda no se encuentra actualmente disponible. Una posible solución pasa por usar el sistema de búsqueda inverso del sistema de nombres. En este caso, cuando se recibe un localizador, se realiza una búsqueda inversa en el DNS para obtener el nombre de dominio correspondiente a dicho localizador. Luego se puede hacer una búsqueda directa para obtener el conjunto completo de localizadores asociados al extremo, siempre y cuando los registros asociados a dicho nombre se correspondan a un solo extremo. De esta forma sería posible usar un localizador como identificador y a partir de él obtener todo el conjunto de localizadores asociados al extremo. Sin embargo, este sistema puede tener una velocidad de respuesta y robustez no compatible con las necesidades de algunas aplicaciones.

Otros identificadores

Si se usa otro espacio de nombres de identificadores, es necesario que sea posible obtener los localizadores asociados a un identificador dado para poder realizar el contacto inicial, las referencias y las llamadas inversas. Para el contacto inicial, es posible recurrir al sistema de nombres de dominio [RFC1034] para poder obtener el identificador y los localizadores de un nodo. Además una vez establecida la comunicación, es posible disponer de un canal de señalización entre los nodos que permita el intercambio de localizadores adicionales. Sin embargo, para soportar correctamente las llamadas inversas y las referencias es necesario crear un mecanismo para obtener los localizadores asociados a un identificador. Esto es más simple si el identificador tiene una estructura jerárquica, ya que sería entonces posible realizar un sistema similar al DNS para realizar esta función. Un sistema jerárquico tiene también como ventaja que cada cual es responsable de administrar y mantener la información asociada a sus propios identificadores, lo que hace que el sistema tenga sentido práctico. En el caso de que el espacio de identificadores no sea jerárquico, será necesario realizar búsquedas en un espacio plano de 128 bits. Esto es un problema complejo pero que podría ser resuelto usando, por ejemplo, tecnologías de DHT (Distributed Hash Tables, [Wiley2003a]) Sin embargo, existen dudas de la robustez y viabilidad de un sistema de este tipo.

4.3 Análisis de seguridad

Como hemos visto, una solución de multihoming pasa por poder asociar múltiples localizadores a un mismo identificador. En la arquitectura actual, el identificador es el localizador. El localizador es usado por el sistema de encaminamiento para alcanzar el destino solicitado, lo que provee un mínimo nivel de seguridad, ya que si confiamos en el sistema de encaminamiento, sabemos que el paquete será entregado al localizador seleccionado, que a su vez es el identificador del extremo deseado. En síntesis, la cadena de confianza en la arquitectura actual cuenta con los siguientes elementos:

- Confianza en el sistema de encaminamiento. Un paquete dirigido a una dirección destino dada será entregado a dicha dirección. Cabe notar que es natural confiar en el sistema de encaminamiento, ya que, gracias a éste los paquetes son entregados en la red. Cabe notar que en el diseño del sistema de encaminamiento (en particular en BGP) hace difícil que un equipo final pueda introducir información que altere las configuraciones generadas por los administradores de las redes. Es decir que cuando un nodo envía un paquete, éste confía en todos los nodos intermedios ubicados a lo largo del camino por el que se transporta el paquete.
- Confianza en la dirección IP como identificador. La confianza otorgada por el hecho de ser un localizador válido, es decir, la garantía provista por el sistema de encaminamiento de que sólo entregará los paquetes al extremo que posea la dirección IP contenida en el paquete, es automáticamente traspasada al identificador, ya que identificador y localizador coinciden en la dirección IP. Esto ofrece una garantía sobre la identidad del extremo con el que se establece la comunicación. La cadena de confianza resultante es entonces:
    - El extremo es identificado por su dirección IP
    - La dirección IP es también el localizador
    - El sistema de encaminamiento garantiza entregar el paquete al extremo al que fue asignado el localizador incluido en el paquete.
- Confianza en el mecanismo para obtener las direcciones IP en el momento de iniciar la comunicación. En particular si el DNS es usado, confianza en la información obtenida a través de la consulta al DNS.

La asociación de múltiples localizadores a un mismo identificador rompe la cadena de confianza anteriormente presentada, como veremos analizando los nuevos ataques posibles en un entorno donde el localizador y el identificador están separados.

Esencialmente, los nuevos ataques pasan por asociar a un identificador uno o varios localizadores que el extremo no posee. Nótese que en la arquitectura actual esto no es posible gracias a que el localizador y el identificador son el mismo, y constante a lo largo de una comunicación.

4.3.1 Escenario de los ataques

Supongamos un escenario donde el nodo A tiene como identificador a IdA y como localizadores LocA1, LocA2,.., LocAn y que el nodo B tiene como identificador a IdB y como localizadores LocB1, LocB2,.., LocBm. Para que la comunicación entre ambos se beneficie del multihoming, es necesario que exista un estado referente al conjunto de localizadores asociados a cada identificador involucrado en cada uno de los extremos de la comunicación. Es decir, que en el extremo A (o en algún dispositivo relacionado con éste), es necesario que exista un estado que asocie el IdB a LocB1, LocB2,.., LocBm y análogamente para el extremo B e IdA y LocA1, LocA2,.., LocAn. Los ataques se basan entonces en introducir el localizador deseado por el atacante LocX, dentro de los localizadores asociados a alguno de los identificadores de los extremos y lograr que dicho localizador sea usado para intercambiar los paquetes de la comunicación. En el ejemplo, el resultado del ataque sería que el extremo A creyera que el identificador IdB tiene asociado el conjunto de localizadores LocB1, LocB2,.., LocBm y LocX. Adicionalmente, para llevar a cabo el ataque es necesario que el extremo A seleccione LocX para enviar paquetes a IdB. De esta forma, la comunicación dejará de realizarse entre el nodo A y el nodo B y se realizará entre el nodo A y el nodo seleccionado por el atacante, que está ubicado en LocX, todo esto sin que el nodo A se percate de ello.

4.3.2 Alcance de los ataques

Como primera aproximación podemos decir que los ataques posibles pueden tener diversos alcances:

- **Secuestro de una comunicación particular**. En este caso, el atacante logra redirigir los paquetes de una comunicación dada entre dos nodos cambiando uno de los extremos involucrados en la comunicación. El ataque no afecta a paquetes relacionados con otras comunicaciones, ya sean futuras, o en otro sentido, ni siquiera a otras comunicaciones simultáneas entre las partes.

- Robo de identidad. El ataque presentado en el párrafo anterior sólo afecta a los paquetes de una comunicación en particular. Sin embargo es posible ampliar el efecto del ataque a un grupo más extenso de comunicaciones de un nodo, como detallaremos a continuación.

- **Robo de identidad en un nodo en concreto**. Una posible extensión de ataque pasa por hacer creer al nodo A que nodo B identificado por el identificador IdB se encuentra a todos los efectos en LocX (y no en su localización real). El efecto de esto es que cada vez que el nodo A crea que se comunica con el identificador IdB, se estará en realidad comunicando con el nodo ubicado en LocX. El resultado de este ataque es que la identidad del nodo B es robada por otro nodo, en lo que el nodo A se refiere. Como consecuencia, todas las comunicaciones que el nodo A establece con el nodo B, entrantes o salientes, presentes o futuras, se estarán realizando con el nodo ubicado en LocX, y no con el nodo B. Adicionalmente, este ataque puede también afectar a otros nodos en caso que se realicen referencias.
- **Robo global de identidad**. Es posible concebir una forma más general del ataque presentado anteriormente, donde el alcance del ataque son todos los nodos de la red. En este caso, todos los nodos que desean establecer una comunicación con el nodo B creerán que este tiene como localizador el LocX (en lugar de LocB1, LocB2,.., LocBn), de forma que todas las comunicaciones que cualquier nodo realiza con en nodo B serán establecidas con el nodo ubicado en LocX.

La distinción entre los distintos alcances de los ataques propuesta en la presente Tesis Doctoral ha sido recogida en el documento de análisis de riesgos del grupo de trabajo multi6 [Nordmark2005a].

4.3.3 Objetivos de los ataques

A continuación analizaremos cuáles son los posibles objetivos del atacante una vez ha logrado redirigir los paquetes enviados a un localizador dado hacia otro localizador (elegido por el atacante).

- Suplantar a un extremo. Un objetivo del atacante puede ser hacerse pasar por uno de los extremos, el extremo B en el ejemplo. En este caso, el localizador LocX asociado al identificador de la victima es un localizador donde el atacante puede recibir los paquetes. De esta forma, cuando el nodo A crea intercambiar paquetes con el nodo B, en realidad estará intercambiando paquetes con el atacante.

- Inspeccionar el tráfico. Otro objetivo posible de atacante es redirigir el tráfico hacia una localización donde pueda inspeccionar el tráfico para luego reenviarlo hacia el destino final. Para esto es necesario lograr que ambos extremos envíen sus paquetes hacia el atacante, para que así

éste inspeccione los paquetes y luego los reencamine hacia el destino final. En este caso, el LocX es también un localizador al cual tiene acceso el atacante.

- **Hombre en el medio.** En adición a lo presentado en el punto anterior, el atacante puede desear modificar los paquetes antes de reenviarlos hacia los extremos involucrados en la comunicación.

- **Denegación de servicio a los extremos de la comunicación.** Otra posibilidad es que atacante desee realizar un ataque de denegación de servicio. En este caso el LocX corresponde a una dirección donde los paquetes serán descartados.

- **Inundación.** En este caso el objetivo del atacante es inundar de paquetes a una víctima. Para ello, el LocX introducido por atacante es el correspondiente a la víctima. De esta forma, el atacante establece una comunicación con el nodo A y solicita, por ejemplo, un gran volumen de información. Luego, una vez que el flujo de paquetes ha comenzado, introduce el localizador de la víctima LocX, como el localizador al cual el nodo A debe enviar los paquetes. El resultado es que el nodo A enviara el flujo de paquetes a la víctima, inundándola.

Se han realizado diversos análisis de los riesgos involucrados en las arquitecturas de separación de identificadores y localizadores, dentro de los cuales caben destacar [Nordmark2005a], [Nikander2004a].

4.3.4 Consideraciones sobre la privacidad

Cuando consideramos la adopción de un nuevo espacio de nombres para identificadores debemos también tener en cuenta los temas relacionados con la privacidad de los nodos de la red. Actualmente, los nodos utilizan direcciones IPv6 dentro de los paquetes para identificar las partes involucradas en la comunicación. Sin embargo, dependiendo del caso, las direcciones pueden ser asignadas por períodos más o menos largos. En el caso en que la dirección se asigne por un periodo largo, un nodo externo podría identificar las distintas comunicaciones realizadas por un mismo nodo. Esto puede brindar información valiosa al nodo observador referente a los gustos y conducta del nodo en particular. Dicha información puede tener valor para campañas publicitarias y otras actividades de marketing. Es más, los riesgos a la privacidad de las personas físicas se incrementan cuando consideramos la proliferación de dispositivos personales conectados a Internet, como pueden ser teléfonos móviles, PDAs, etc. En este caso, un dispositivo está asociado a un solo usuario, por lo que la información obtenida refiere directamente al comportamiento de una persona física en particular. En el caso de IPv6, es habitual que la dirección IP contenga un Identificador de Interfaz [RFC3513] formado a partir de la dirección MAC de la tarjeta Ethernet, identificador que es globalmente único. Esto hace que el mismo

dispositivo sea identificable incluso cuando cambia de prefijo de red (es decir cuando cambia de red). Esto implica que es posible rastrear un dispositivo móvil cuando éste se mueve en la red.

Para hacer frente a estos problemas, se han definido las Extensiones de Privacidad [RFC3041] que se basan en la creación periódica de Identificadores de Interfaz temporales (y por ende la generación periódica de direcciones IP). En este caso, el nodo genera los Identificadores de Interfaz de forma aleatoria y los va reemplazando periódicamente, por lo que un observador externo no es capaz de correlacionar las distintas comunicaciones realizadas por el nodo en cuestión.

Una solución de multihoming debe al menos brindar un soporte equivalente a la privacidad que el ofrecido por las Extensiones de Seguridad [RFC3041].

4.3.5 Pruebas de correspondencia entre identificadores y localizadores

En una sección anterior hemos visto que existen distintos espacios de nombres que pueden ser utilizados para los identificadores. En el momento en que un identificador representa la identidad de un extremo, es necesario poder probar la relación entre el identificador propuesto para la comunicación y el conjunto de localizadores disponibles para este identificador. Dicha verificación es necesaria para evitar posibles suplantaciones y ataques de inundación. A continuación estudiaremos distintas formas de demostrar la relación entre identificadores y localizadores. En particular estudiaremos tres casos: cuando uno de los localizadores es usado como identificador (el caso actual y posibles extensiones al caso con múltiples localizadores disponibles en un extremo), cuando se utiliza una tercera parte de confianza para verificar la relación entre identificadores y los localizadores (el caso del DNS), y el caso de los identificadores criptográficos.

Localizadores como identificadores.

Como hemos visto anteriormente, cuando un extremo envía un paquete, éste confía en el sistema de encaminamiento, ya que cualquiera de los nodos intermedios que lo componen es capaz de redirigir el paquete hacia otro destino o simplemente borrarlo. Es posible utilizar mecanismos de seguridad como IPSec [RFC2401], que permitan evitar que un nodo intermedio modifique un paquete, o que inspeccione su contenido, e incluso es posible detectar si se ha perdido algún paquete, pero lo que parece claro es que no hay nada que los extremos puedan hacer para evitar que los nodos intermedios redirijan los paquetes hacia otro lado o que simplemente los descarten. El modelo actual de confianza se basa en la suposición de que el sistema de encaminamiento llevará el paquete hasta el destino final especificado por la dirección destino contenida en el mismo.

Por ello, una forma de verificar que un extremo es propietario de un localizador es el llamado procedimiento de encaminamiento inverso (*Return Routability procedure RR*) [RFC3775], que consiste en verificar si el extremo que dice poseer un localizador dado es capaz de recibir paquetes enviados al mismo. En este esquema, un nodo A puede verificar que un nodo B posee el localizador LocB, simplemente enviando un desafío (por ejemplo un número aleatorio) dirigido al nodo B y destinado a LocB. Si el nodo B es capaz de responder a dicho paquete, incluyendo en su respuesta el desafío inicial, esto quiere decir que el nodo B posee el LocB, ya que es capaz de recibir paquetes destinados al mismo. Cabe notar, que cualquier nodo que se encuentre en el camino entre el nodo A y el nodo B es capaz de responder satisfactoriamente es este desafío, pero como mencionamos antes, este nodo intermedio es también capaz de descartar todos los paquetes que fluyen entre el nodo A y el nodo B, por lo que ya existe una confianza implícita entre los extremos y los nodos intermedios.

El mecanismo de encaminamiento inverso es satisfactorio (dentro de las limitaciones ya descritas) para verificar la posesión de un localizador por parte de un extremo, sin embargo ¿es posible aplicarlo también para verificar la relación entre el identificador y el conjunto de localizadores?

Consideremos el caso en el que el nodo A tiene múltiples localizadores, LocA1, LocA2,..., LocAn. Para una comunicación dada, utilizará uno solo de estos como identificador, para permitir al otro extremo que reconozca su identidad a lo largo de los cambios de localizadores. Esto quiere decir que uno de estos localizadores, por ejemplos el LocAi, será la identidad de nodo A. Mientras el nodo A utilice el LocAi también como localizador para la comunicación, se estará (implícitamente) utilizando el mecanismo de RR, ya que el nodo A estará respondiendo a paquetes enviados al LocAi, por lo que estará implícitamente probando que puede recibir paquetes al LocAi, y estará demostrando así la relación entre el localizador y el identificador. Sin embargo, en el caso que el nodo A deje de recibir los paquetes en el LocAi y reciba los paquetes en otro localizador, por ejemplo el LocAj, el mecanismo de RR no estará demostrando que el nodo A es dueño del LocAi sino que estaría demostrando que es el dueño del LocAj. Es decir que mientras que el nodo A recibe paquetes en LocAk, sólo demuestra que es dueño de este localizador y no de otro. Por ello, si el nodo A comienza una comunicación usando el LocAi como identificador, sólo estará demostrando su propiedad mientras reciba paquetes en éste, por lo que si desea utilizar otro localizador durante la vida de esta comunicación, en el momento que empiece a hacerlo dejaría de demostrar la propiedad de su identificador.

Una posibilidad en este caso, es verificar la relación entre el identificador y el localizador utilizados en la comunicación mediante el procedimiento de RR a través del intercambio periódico de desafíos-respuestas con ambas direcciones, como se hace en la solución de Optimización de Rutas definida en el soporte de movilidad en IPv6 [RFC3775]. En este caso, cuando el nodo A desea utilizar el LocAi como identificador y el LocAj como localizador en la

comunicación con el nodo B, el nodo B envía los paquetes de datos de la comunicación al LocAj, pero periódicamente verifica la identidad del nodo A intercambiando paquetes de desafío-respuesta a través del LocAi. La verificación esencialmente está dirigida a confirmar que el nodo que con el que se intercambia paquetes dirigidos al LocAj también es dueño de la identidad que reclama, es decir el LocAi. Para esto el nodo B envía desafíos tanto al LocAi como al LocAj, y espera una respuesta que contenga ambos, de forma que se verifique que mismo nodo posee ambos localizadores. Es necesario realizar dicha verificación de forma periódica para evitar ataques desplazados en el tiempo, como se detalla a continuación.

En principio, uno podría pensar que con una verificación inicial de la relación entre el localizador y el identificador sería suficiente y que luego se podría continuar la comunicación utilizando el localizador alternativo. Sin embargo esto abre la puerta a ciertos ataques llamados *ataques desplazados en el tiempo*, descritos en [Nikander2004a]. En estos ataques, el atacante se instala a lo largo del camino de forma que intercepte los paquetes de desafío. Una vez respondido el desafío el atacante abandona su posición a lo largo del camino y pasa a ocupar una posición más cómoda, en un localizador que le resulte más accesible. Si el procedimiento de RR para el identificador se realiza una sola vez, es suficiente que el atacante ocupe una posición a lo largo del camino el tiempo necesario para responder satisfactoriamente ese único desafío y luego continuar usurpando el identificador desde otra posición, de forma indefinida. Esto simplifica la tarea del atacante, ya que en la configuración actual, con un solo identificador/localizador, el atacante debe permanecer a lo largo del camino para realizar su ataque y no puede seguir con el ataque una vez abandonado el camino. Para evitar este ataque, es necesaria la ejecución periódica del procedimiento RR, de forma que se obligue al atacante a permanecer en el camino para poder responder a los desafíos periódicos. Con este mecanismo, la situación resultante es similar a la actual, donde el atacante debe permanecer en el camino para realizar un ataque.

Todo lo anterior implica que cuando se utiliza un localizador como identificador y se utiliza el procedimiento de RR para verificar la relación entre ambos, el localizador utilizado como identificador debe permanecer alcanzable mientras éste sea usado, ya que es necesario que responda a los desafíos periódicos enviados al este localizador. Este mecanismo parece entonces poco idóneo para demostrar la relación entre identificadores y localizadores en una solución de tolerancia a fallos como es el multihoming, donde los localizadores pueden no estar alcanzables a lo largo de la vida de una conexión [Bagnulo2003a], [Huston2005a], [Bagnulo2003d].

Tercera parte de confianza. Otra posibilidad para demostrar la relación entre localizadores e identificadores es recurrir a una tercera parte de confianza. Un ejemplo claro de este tipo de mecanismos es el DNS. El DNS asocia un identificador de extremo, el nombre de dominio, a un conjunto de localizadores. El sistema de nombres de dominio actual es susceptible a un conjunto de ataques [RFC3833] pero a su vez existen diversas herramientas para mejorar su seguridad [RFC4033]. Es posible extender el mecanismo de DNS u otra tercera parte de confianza para

asociar un identificador distinto del nombre de dominio para vincular el identificador a un conjunto de localizadores. Por ejemplo, es posible pensar en un sistema basado en el DNS, que al consultar por un nombre de dominio, devuelva un identificador y un conjunto de localizadores posibles.

Identificadores criptográficos. Una tercera posibilidad para poder probar la relación entre identificadores y localizadores es hacer uso de las propiedades de los identificadores criptográficos

Por ejemplo, en el caso en que el identificador es la clave pública de un par de claves, cuya clave privada es conocida sólo por el propietario del identificador. De esta forma, demostrar la relación entre el identificador y sus localizadores es trivial, y sólo requiere la firma del conjunto de localizadores con la clave privada que corresponde al identificador en cuestión. Es decir, si el nodo A desea probarle al nodo B que es propietario del identificador PuA que es la clave pública asociada a la clave privada, sólo debe cifrar un mensaje que contenga el conjunto de localizadores. Este sistema permite ofrecer protección frente a los ataques de suplantación de identidad, ya que el poseedor de la clave puede asociar el conjunto de localizadores. Sin embargo, este esquema no provee protección frente a ataques de inundación. Esto se debe a que el poseedor de la clave privada puede asociar cualquier localizador al identificador asociado a la clave privada. Esto permite asociar el localizador del objetivo del ataque de inundación, habilitando así el susodicho ataque.

El análisis de propiedad de localizadores e identificadores anterior planteado por el autor en la presente Tesis Doctoral ha sido recogido en el Apéndice B del documento de análisis de riesgos del grupo de trabajo multi6 [Nordmark2005a].

Adicionalmente, algunas de las reflexiones presentadas en este análisis, han sido planteadas por el autor de la presente Tesis doctoral y recogidas en los siguientes documentos del grupo multi6: “Architectural Approaches to Multi-Homing for IPv6” [Huston2005a], “Things MULTI6 Developers should think about [LEAR2005a] y en otros Internet Drafts como “Choices for Multiaddressing” [Crocker2003a]

4.4 Análisis funcional

La presente sección incluye un análisis funcional del problema de multihoming. El objetivo del presente análisis es identificar las distintas funciones que son necesarias para la construcción de una solución al problema en cuestión. A través de esta identificación de funciones se pretende

aislar los distintos componentes que son necesarios en una solución de multihoming y analizar su interacción. La descomposición funcional planteada en la presente Tesis Doctoral es la siguiente:

- Mecanismo de descubrimiento de identificadores
- Mecanismo de descubrimiento de localizadores
    - Para el contacto inicial
    - Una vez establecido el contacto
- Mecanismo de validación de la relación entre identificadores y localizadores
    - Para el contacto inicial
    - Una vez establecido el contacto
- Mecanismo de selección de camino/localizadores
- Mecanismo de detección de fallos durante una comunicación
- Mecanismo de recolección de basura

4.4.1 Mecanismo de descubrimiento de identificadores

Como hemos visto en el análisis arquitectónico realizado, las distintas soluciones requieren del uso de identificadores para aislar las capas superiores de los cambios en los localizadores usados. Estos identificadores pueden ser usados solamente por las aplicaciones, en el caso de las soluciones de capa de transporte o capa de sesión, o pueden ser utilizados por las capas de transporte y las capas de aplicación, en el caso de las soluciones implementadas por debajo de la capa de transporte. En cualquier caso, es necesario disponer de un mecanismo para el descubrimiento de los identificadores a usar en la comunicación. Debido a las potenciales diferencias, estudiaremos por separado el caso del descubrimiento del identificador del nodo receptor por parte del nodo iniciador, y el descubrimiento del identificador del nodo iniciador por parte del nodo receptor.

4.4.1.1 Descubrimiento del identificador por parte del nodo iniciador de la comunicación.

Cuando hablamos de descubrimiento, estamos implícitamente considerando un punto de partida, alguna referencia al nodo en cuestión, a partir de la cual podemos obtener el identificador correspondiente. En general, el nodo iniciador de la comunicación necesita ser capaz de descubrir el identificador del nodo receptor partiendo del FQDN (*Fully Qualified Domain Name*, nombre DNS) del nodo receptor, es decir que el iniciador desea saber cuál es el identificador asociado a un cierto FQDN. El mecanismo más directo para esto es simplemente guardar la información del

identificador correspondiente en el propio DNS, pero es posible que se requieran cambios en el mismo para soportar esta funcionalidad.

También es posible que sea necesario descubrir el identificador asociado a un cierto localizador, en aquellas situaciones donde la información del localizador sea el punto de partida (un caso para esto puede ser cuando se recibe una referencia por parte de un nodo que no soporta el nuevo espacio de identificadores). En este caso, el nodo iniciador sólo dispone de la información de un localizador, y debe obtener la información del identificador del nodo receptor a partir de este localizador. Para esto, es posible pensar dos tipos de mecanismos: Por una parte, un sistema de directorio que devuelva el identificador asociado a un localizador. Tal sistema podría basarse, por ejemplo, en el uso del DNS inverso. Por otra parte, es posible considerar un sistema de descubrimiento en el que nodo iniciador establece una comunicación de señalización con el nodo receptor y descubre el identificador del nodo receptor a través de la misma.

4.4.1.2 Descubrimiento de identificador por parte del nodo receptor.

En este caso, el nodo receptor ha sido contactado por el nodo iniciador, por lo que es necesario que el nodo receptor descubra el identificador del nodo que ha iniciado la comunicación. El punto de partida viene dado aquí por la información contenida en el paquete inicial enviado por el iniciador y recibido por el receptor.

Es posible pensar que el identificador estará contenido directamente en el paquete inicial. Esto ocurre cuando el identificador es un localizador válido, ya que este puede estar contenido en el campo dirección origen del paquete IPv6. En el caso que el identificador no sea un localizador válido, el identificador puede estar contenido en el paquete, por ejemplo en un *Extension Header.*

También es posible que el identificador no se encuentre disponible en el paquete inicial y que sea necesario descubrirlo a partir de información contenida en éste, por ejemplo a partir del localizador usado. Para ello, será necesario utilizar un directorio para obtener el identificador a partir del localizador contenido en el paquete inicial (por ejemplo el DNS inverso).

4.4.2 Mecanismo de descubrimiento de localizadores

Para hacer uso de las capacidades del multihoming, es necesario que los nodos involucrados en la comunicación descubran el conjunto de localizadores disponibles para el otro nodo. A continuación analizaremos los distintos mecanismos posibles para realizar esta tarea. Estudiaremos por separado el caso del descubrimiento de los localizadores por parte del iniciador de la comunicación y por parte del receptor. Además distinguiremos el caso del descubrimiento de localizadores para el contacto inicial y cuando la comunicación ya se encuentra en curso.

4.4.2.1 Descubrimiento del conjunto de localizadores por parte del nodo iniciador de la comunicación

4.4.2.1.1 Contacto inicial.

En este caso, el iniciador puede obtener el conjunto de localizadores a partir de:

- El FQDN del receptor
- El identificador
- Uno de los localizadores

El descubrimiento del conjunto de localizadores a partir del FQDN es análogo a la situación actual, donde el nodo iniciador obtiene los localizadores mediante una consulta al DNS, obteniendo los múltiples localizadores asociados a un nombre de dominio por cada consulta. Es decir que cuando el iniciador consulte en el DNS por el FQDN del receptor, el DNS devolverá el conjunto de localizadores asociados a dicho nombre. Cabe notar que el conjunto de localizadores devueltos no tiene porqué pertenecer a un único extremo y por ende, no estar asociados al mismo identificador. Por ejemplo, el sistema de nombres se puede utilizar para proveer una forma primaria de balanceo de carga, asociando localizadores de los distintos servidores al mismo nombre de dominio (por ejemplo cuando hay múltiples servidores web para un dominio muy concurrido, es común que existan múltiples registros asociando www.foo.net a diversas direcciones de distintos servidores físicos). La solución a este problema pasa por explicitar cuales de los localizadores están asociados a cada identificador, posiblemente creando un nuevo registro de DNS que ofrezca esta semántica.

En el caso de que sea necesario descubrir el conjunto de localizadores partiendo del identificador viene dado, por ejemplo, en situaciones en donde se utilizan referencias que contienen identificadores. En este caso es necesario obtener el conjunto de localizadores a partir del identificador. Para ello, es necesario un sistema de directorio que contenga esa información. Como ya hemos visto, la existencia de dicho directorio depende de la naturaleza del espacio de identificadores usado. Si los identificadores usados son los FQDNs, entonces estamos en el caso anterior. Si los identificadores usados son localizadores válidos, podemos obtener el conjunto completo usando estos mismos localizadores de dos formas, a saber: usando el DNS inverso y luego el DNS directo con la respuesta obtenida; o contactar con el nodo propietario del identificador utilizando el identificador como localizador y obtener de él el conjunto de localizadores (esto implica que el localizador que disponemos se encuentra alcanzable).

4.4.2.1.2 Comunicación en curso

Una vez realizado el contacto inicial, el nodo iniciador puede descubrir el conjunto preciso de localizadores asociados con el nodo receptor, si es que el mecanismo usado para el contacto inicial no le ha brindado dicha información (las causas para esta situación pueden ser que la información contenida en el DNS no sea completa o ajustada a la realidad, por ejemplo en el caso en que un FQDN esta asociado a múltiples nodos con el objetivo de brindar una forma precaria de distribución de carga). Los mecanismos para este caso son los mismos que puede utilizar el nodo receptor para descubrir los localizadores del nodo iniciador, y se describen a continuación.

4.4.2.2 Descubrimiento del conjunto de localizadores por parte del receptor.

En este caso, por definición de receptor, no es necesario descubrir los localizadores para el contacto inicial ya que el nodo receptor ha sido previamente contactado por el nodo iniciador. El nodo receptor debe descubrir el conjunto de localizadores a partir de:

- La información contenida en el paquete inicial
- el localizador contenido en el paquete inicial
- el identificador

La opción más simple parece ser que el nodo iniciador incluya en el paquete inicial (u otros paquetes subsiguientes) la información referente al conjunto de localizadores de él mismo.

Si esta opción no es usada, otra posibilidad es obtener el conjunto de localizadores a partir del localizador incluido en los paquetes entrantes. Para ello, es posible usar por ejemplo una combinación del DNS inverso y del DNS directo aunque es posible que se requieran modificaciones adicionales.

Finalmente, sería posible obtener el conjunto de localizadores a partir del identificador contenido en el paquete inicial. Para esto es necesario que exista un servicio de directorio que asocie identificadores a localizadores. Si el identificador es un localizador válido, es posible usar el DNS como servicio de directorio.

- Cabe notar que estos mecanismos de descubrimiento de localizadores pueden ser usados por cualquiera de los nodos involucrados en la comunicación en cualquier momento de la vida de la comunicación, y no es requerido que sea en el primer paquete intercambiado.

4.4.3 Mecanismo de validación de la relación entre identificadores y localizadores

Como se ha presentado en la sección 4.3.5 existen los siguientes mecanismos para validar la relación entre localizadores e identificadores:

- Encaminamiento inverso en el caso que exista una relación univoca entre identificador y localizador
- Tercera parte de confianza
- Identificadores criptográficos

4.4.4 Mecanismo de selección de caminos y localizadores

4.4.4.1 Relación entre localizadores y caminos

Como se ha estudiado en varias ocasiones a lo largo de la presente Tesis Doctoral, existe una estrecha relación entre localizadores y caminos en los entornos derivados del uso de múltiples rangos de direccionamiento agregables por proveedor (PA) como el propuesto para los sitios multihomed. En particular los localizadores usados mantienen una estrecha relación con los proveedores a través de los cuales se encaminarán los paquetes desde y hacia el sitio multihomed. Esto es esencialmente consecuencia de dos factores: la naturaleza de los bloques de direcciones agregables por proveedor y los filtros de ingreso.

En una comunicación entre dos sitios multihomed, podemos distinguir esencialmente tres tramos del camino que une ambos sitios, a saber: el tramo que va desde el sitio origen hasta su LIR, el tramo que une el LIR del sitio origen con el LIR del sitio destino y el tramo que va desde el LIR del sitio destino hasta este sitio (recordemos que el LIR de un sitio es el proveedor que obtiene el rango de direcciones PA directamente del RIR, y que a su vez se lo delega al sitio o a sub-LIRs que a su vez se lo delegan al sitio multihomed). En una arquitectura de direccionamiento basado en proveedor, todos los sitios y proveedores que obtienen conectividad a través de un LIR obtienen direcciones del bloque del LIR, y el LIR sólo anuncia su propio bloque a los demás proveedores, por lo que sólo recibe paquetes dirigidos a destinos contenidos en su propio bloque. La situación se ilustra en la figura siguiente.

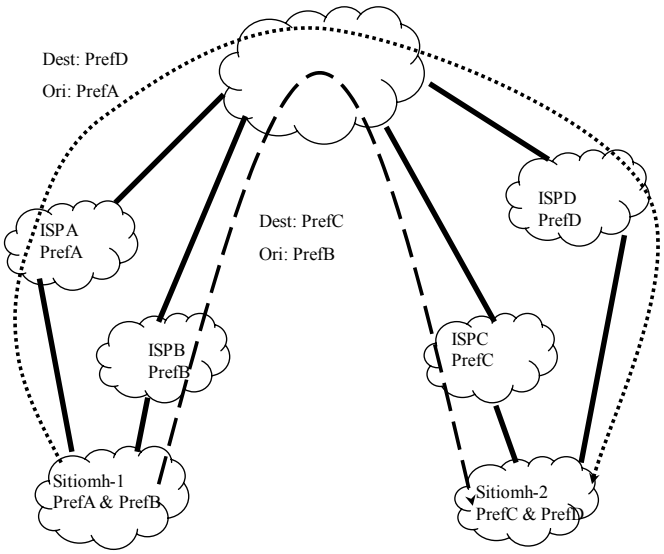

Figura 14: Alcanzabilidad y prefijos PA

En este escenario, cuando un paquete fluye del sitio multihomed 1 hacia el sitio multihomed 2, existe una estrecha relación entre los localizadores contenidos en el paquete y ciertos tramos del camino seguido. En primer lugar, por la propia naturaleza del direccionamiento PA, el LIR del sitio destino queda determinado por la dirección de destino usada, ya que si usamos una dirección de otro LIR del sitio destino, este otro LIR será el utilizado para llevar los paquetes hasta el sitio destino.

Adicionalmente, debido a la existencia de filtros de ingreso, debe existir una estrecha relación entre el LIR del sitio del sitio multihomed usado para cursar los paquetes y la dirección origen del paquete, ya que si estos dos elementos no son compatibles, los filtros de origen descartarán el paquete.

En síntesis:

- La dirección destino del paquete está estrechamente vinculada al LIR del sitio destino.
- La dirección origen del paquete está estrechamente vinculada al LIR del sitio origen.

Además, en general, la dirección origen de un paquete será utilizada como dirección destino de los paquetes generados en respuesta al paquete original, y la dirección destino del paquete original en general será utilizada como dirección origen en los paquetes respuesta, por lo que podemos agregar que:

- La dirección origen del paquete inicial está estrechamente vinculada con el LIR del sitio origen usado para recibir los paquetes respuesta del paquete original
- La dirección destino del paquete inicial está estrechamente vinculada con el LIR del sitio destino usado para enviar los paquetes respuesta del paquete original.

Nota: estas dos ultimas conclusiones asumen que las direcciones origen y destino del paquete original serán utilizadas como direcciones destino y origen en los paquetes respuesta, lo que puede no ser siempre cierto, dependiendo de la solución de multihoming adoptada

Por ende, resulta claro que los mecanismos de selección de caminos y los mecanismos de selección de localizadores estarán estrechamente vinculados. Sin embargo, hay que notar que en la arquitectura actual para IPv4, los mecanismos de selección de camino residen esencialmente en el sistema de encaminamiento mientras que los mecanismos de selección de localizadores residen en los nodos finales. Por ello, para permitir la gestión apropiada de la relación entre selección de camino y selección de localizadores será necesario realizar alguna modificación al comportamiento actual, como se estudia a continuación.

4.4.4.2 Mecanismos de selección de caminos y localizadores

Como hemos visto, es necesario que exista una relación entre los mecanismos de selección de localizadores y de caminos. Actualmente, la selección de localizadores es realizada por los nodos, que asocian ciertos localizadores a las comunicaciones establecidas. Por otra parte, la selección de caminos es realizada por el sistema de encaminamiento, el cual posee información de los caminos disponibles. Para crear una relación entre la selección de caminos y la selección de localizadores debemos alterar alguno de los mecanismos actuales.

4.4.4.2.1 Mecanismos basados en el sistema de encaminamiento

Una posibilidad es que el sistema de encaminamiento realice ambas tareas, es decir selección de caminos y selección de localizadores. Esto implica que los localizadores seleccionados por el nodo sean reemplazados por el sistema de encaminamiento, de forma que se incluyan localizadores acordes a la información de camino disponible en el sistema de encaminamiento.

Para el caso de la dirección origen, es posible configurar los prefijos disponibles en el sitio multihomed en el sistema de encaminamiento, de modo que el propio sistema de encaminamiento elija el prefijo mas adecuado para el paquete en cuestión. El prefijo incluido en la dirección origen debe ser compatible con los filtros de ingreso de los proveedores (es decir que debe ser compatible con el proveedor usado para transportar el paquete) y además debe estar asociado a un proveedor a través del cual exista un camino hacia (y desde) el destino. Por ende, el mecanismo para la selección del localizador origen por parte del sistema de encaminamiento deberá seguir los pasos siguientes:

- Determinar los caminos existentes hacia el destino. Esta información puede estar disponible en el sistema de encaminamiento
- Determinar el prefijo asociado con el proveedor involucrado.
- Seleccionar el localizador correspondiente a dicho proveedor
- Encaminar el paquete a través de dicho proveedor

Para el caso del localizador destino, la situación es más compleja, ya que no es posible configurar con anterioridad el conjunto de prefijos disponible para cada uno de los destinos posibles. Por esto, el sistema de encaminamiento deberá descubrir el conjunto de localizadores susceptibles a ser utilizados como localizador destino, para luego determinar cuál es el localizador más apropiado, en particular cuáles de ellos son alcanzables. Entonces, en un mecanismo de selección de camino y locador destino basado en el sistema de encaminamiento, éste realizaría las siguientes tareas:

- El sistema de encaminamiento descubre el conjunto de localizadores disponibles para el destino
- El sistema de encaminamiento determina cuál de los localizadores es alcanzable
- El sistema de encaminamiento selecciona el localizador apropiado
- El sistema de encaminamiento encamina el paquete a través del camino seleccionado

Resulta claro que la dificultad esencial radica en el descubrimiento de los localizadores disponibles para un destino dado y en determinar cuales de ellos están alcanzables.

Cabe notar que los mecanismos de selección de localizador destino y origen están naturalmente relacionados y que la elección de uno influye en la del otro. Esto se debe a que un cierto prefijo destino puede estar alcanzable desde un cierto prefijo origen y no desde otro. Esto implica que el mecanismo final deberá interrelacionar la selección de prefijo origen y destino. También es relevante notar en este punto que la información de caminos de la que dispone el sistema de encaminamiento se refiere a agregados y no a nodos en particular. Esto implica que es posible (y frecuente) que el sistema de encaminamiento disponga de una ruta hacia un agregado, pero que en realidad, el destino deseado no se encuentre alcanzable a través de esta ruta.

4.4.4.2.2 Mecanismos basados en los nodos

En este caso, los nodos tienen natural acceso a la información del conjunto de localizadores, pero no tienen por defecto acceso a la información de qué caminos se encuentran disponibles. Adicionalmente, los nodos no seleccionan el camino a seguir por el paquete, ya que esto es tarea del sistema de encaminamiento.

Entonces para hacer un mecanismo de selección de localizador destino basado en los nodos, debemos permitir que los nodos realicen las siguientes tareas:

- El nodo debe descubrir el conjunto de localizadores del destino
- El nodo debe descubrir cuáles de dichos localizadores son alcanzables. Para esto se puede pensar en mecanismo de ensayo y error o en la obtención de la información a partir del sistema de encaminamiento
- El nodo debe enviar el paquete al localizador destino elegido a través del camino disponible.

En el caso de la selección del localizador origen, el nodo posee de forma natural la información de los posibles localizadores a ser usados, y debe entonces descubrir cuál es el localizador origen más idóneo, por lo que las tareas que deberá realizar el nodo son:

- El nodo debe descubrir el conjunto de localizadores origen
- El nodo debe descubrir qué prefijos corresponden a proveedores que tienen un camino disponible hacia el destino deseado. Esto es posible obtenerlo a través de ensayo y error u obteniendo información del sistema de rutas. Ambos enfoques presentan dificultades. En el caso el ensayo y error, se requiere que el nodo sea capaz de forzar la selección de camino, lo cual es actualmente potestad del sistema de rutas. En el caso de hacer uso de la información disponible a través del sistema de encaminamiento, la dificultad reside en que esta no incluye información del proveedor a través del cual es accesible un destino dado.
- El nodo selecciona el localizador origen a utilizar
- El nodo envía el paquete con el localizador origen seleccionado a través del proveedor asociado. Para esto es necesario que el nodo sea capaz de forzar la selección del camino, ya que el encaminamiento por defecto no se ve influido por el localizador origen.

Cabe notar que, como hemos observado anteriormente, la selección del localizador origen y la selección del localizador destino están relacionadas, por lo que el proceso final de selección deberá contemplar la información referente a los posibles localizadores destino y origen.

4.4.4.2.3 Mecanismo híbrido

Es posible también considerar un mecanismo híbrido, donde la selección del localizador origen es realizada por el sistema de encaminamiento y la selección del localizador destino es realizada por el nodo, o viceversa. En este caso, cada una de las partes realizaría sólo uno de los mecanismos descritos anteriormente y la otra parte realizaría el otro mecanismo de selección.

4.4.4.3 Políticas

Además de la información de disponibilidad, la información sobre políticas de encaminamiento deberá tenerse en cuenta para la selección de caminos, y por ende también para

la selección de localizadores. Los mecanismos de selección deberán entonces incorporar la información de políticas cuando realicen la selección.

4.4.5 Mecanismos de detección de fallos durante una comunicación

Una vez establecida una comunicación, es necesario disponer de un mecanismo de detección de fallos, para poder iniciar un proceso de cambio de camino.

La información de fallos puede obtenerse a través de diversos medios como:

- Información del sistema de encaminamiento que indica que cierto destino no está alcanzable
- Información de señalización entre la red y el nodo utilizando por ejemplo paquetes ICMP para informar de que cierto destino es inalcanzable
- Sistema dedicado extremo a extremo que verifica periódicamente la disponibilidad del camino en uso
- Las capas superiores pueden detectar fallos en la comunicación e informar de los mismos. Por ejemplo, las aplicaciones que esperan información con una cierta cadencia o capas de transporte que poseen temporizadores que informan si no se ha recibido los paquetes enviados hasta el momento.

4.4.6 Mecanismo de recolección de basura

Una vez finalizada la comunicación entre dos nodos, es necesario que la información referente a la contraparte en dicha comunicación sea eliminada de los nodos para recuperar recursos que ya no son usados. Existen esencialmente dos enfoques posibles para disparar el proceso de eliminación de información que ya no está siendo utilizada, a saber, un enfoque coordinado mediante un protocolo de finalización de la comunicación y un enfoque discrecional basado en un tiempo de vida de la información.

En el primer enfoque, la capa que dispone de la información de estado sobre la comunicación (por ejemplo los localizadores, identificadores, información de seguridad) ejecuta un protocolo de finalización de la comunicación cuando ésta se termina. Como consecuencia de este protocolo, toda la información referente a la comunicación es borrada de los nodos involucrados. La dificultad que presenta este mecanismo es cómo la capa en cuestión descubre que la comunicación ha terminado, especialmente en los casos en que esta capa es no orientada a conexión. Es posible que la capa en cuestión decida ejecutar el protocolo de finalización cuando ha pasado un cierto tiempo desde la última vez que la información relacionada con una cierta

comunicación fue utilizada. Sin embargo, este tiempo puede variar fuertemente de una aplicación a otra, y en caso de que la información sea eliminada antes de tiempo, la aplicación puede experimentar problemas de comunicación. Adicionalmente, cabe notar que en este enfoque será necesario contar con un mecanismo discrecional de recolección de basura como el que se presenta a continuación, para poder lidiar con situaciones en las cuales una de las partes pierde inesperadamente la información relativa a comunicación, sin ejecutar el protocolo de finalización (considerar por ejemplo el caso en que un nodo se reinicia debido a un error interno).

El segundo caso es que cada uno de los nodos involucrados decida por sí mismo eliminar la información relacionada con una cierta comunicación. Este proceso puede dispararse a partir de una notificación explícita por parte de las capas superiores (función no disponible en la arquitectura actual de TCP/IP) o bien porque la información relacionada con una comunicación no ha sido utilizada durante un cierto tiempo. Esta última opción presenta las mismas dificultades que en el caso anterior.

En resumen, la información relacionada con una comunicación puede eliminarse de forma discrecional por parte de cada uno de los nodos involucrados en la comunicación o de forma coordinada. Las ventajas de la forma discrecional es la simplicidad y las que se obtienen de la forma coordinada es la coherencia en la información disponible en cada nodo (lo que puede ser una ventaja en términos de seguridad). La dificultad principal existente es cómo saber cuándo se puede eliminar la información. Cabe notar que esta dificultad es menor si la información eliminada es recuperable de alguna forma automática.

Capítulo 5

Consideraciones de diseño para soluciones de multihoming

5.1 Introducción

En el presente capítulo presentaremos consideraciones referentes al diseño de soluciones para proveer soporte de multihoming en IPv6. Comenzaremos por describir los criterios de diseño que consideramos relevantes para el caso en consideración para luego pasar a presentar una serie de decisiones de diseño que a nuestro entender son apropiadas para el problema en cuestión. En particular, las decisiones presentadas a continuación son las que a nuestro entender se ajustan más adecuadamente a los objetivos del diseño presentado en el capítulo **¡Error! No se encuentra el origen de la referencia.**

5.2 Criterios de diseño

A continuación presentaremos un reducido conjunto de criterios de diseño fundamentales en los que basaremos las decisiones de diseño para la definición de soluciones para el soporte de multihoming en IPv6.

5.2.1 No comprometer la funcionalidad actual

Una consideración esencial para el desarrollo de una nueva solución en Internet es que la nueva tecnología no rompa nada de lo que está en funcionamiento actualmente. Este argumento se aplica a distintos aspectos de la red, dentro de los cuales son relevantes al menos los siguientes:

- **Preservar las comunicaciones posibles actualmente**, incluyendo
    - Permitir la comunicación entre un nodo (sitio) que implemente la solución con un nodo (sitio) que no implemente la solución (es decir que sólo implemente las especificaciones vigentes al día de hoy),
    - Permitir que un nodo que no implemente la solución funcione correctamente dentro de un sitio (multihomed) que sí implementa la solución. Por funcionamiento correcto entendemos que pueda funcionar como si estuviera dentro de un sitio no-multihomed que no implementa la solución.
- **No introducir vulnerabilidades adicionales**. La red resultante después de la adopción de la solución no debe tener más vulnerabilidades que la red actual (sin la solución). En particular, los nodos que implementan la solución no deben tener más vulnerabilidades que los nodos actuales.
- **No romper las aplicaciones**. Las aplicaciones actuales hacen un cierto uso de las direcciones IP. Si se modifica el uso de las direcciones IP o se introduce un nuevo espacio de identificadores para ser usado por las aplicaciones, los supuestos realizados por las aplicaciones sobre las propiedades de las direcciones IP (o de la cadena de caracteres que estas manejan) deben mantenerse.

5.2.2 Despliegue

Cualquier solución que deba ser desplegada en una red existente y vasta como es Internet, debe considerar los aspectos referentes a su adopción y despliegue en este ambiente. En particular consideramos que es relevante considerar los siguientes puntos:

- **Impacto reducido**: minimizar el número de cambios requeridos. Como ejemplo a no seguir, podemos considerar RSVP [RFC2205], cuya adopción requería la modificación tanto de nodos finales como de routers.
- **Despliegue incremental**. La solución en cuestión debe poder desplegarse incrementalmente. Esto implica que no debe ser necesario realizar todas las modificaciones para obtener los beneficios deseados. En particular, la solución debe proveer beneficios (tal vez reducidos) cuando algunos nodos adoptan la solución.
- **Incentivos para la adopción**. Para que una solución sea adoptada, los nodos que la implementan deben obtener beneficios de ésta. Es decir, es necesario que si un sitio adopta la solución de multihoming, éste obtenga algún beneficio (tal vez no todos) de la solución, independientemente de que la solución no sea adoptada por otros sitios. Por otra parte, es también deseable que los costos de la solución sean afrontados por quienes obtienen beneficios de la misma, ya que si estos deben ser afrontados por partes que no obtienen beneficios, el despliegue de la solución tendrá menos incentivos.

5.2.3 Otros criterios

Adicionalmente, se considera relevante considerar criterios generales de diseño dentro de los cuales cabe destacar:

- **Simplicidad**.
- **Valor añadido a la arquitectura**. En este apartado nos referimos a cuestiones como si la arquitectura diseñada para multihoming ofrece ventajas para la movilidad, el cambio de proveedor de un sitio, la seguridad de las comunicaciones, etc.

5.3 Decisiones de diseño

En esta sección presentaremos una serie de decisiones de diseño mediante las cuales se realiza una selección entre las múltiples opciones que se han identificado durante la etapa de análisis presentada en el capítulo anterior.

5.3.1 Capa elegida para implementar un mecanismo para preservar las comunicaciones establecidas

Como hemos visto, el mecanismo para preservar comunicaciones establecidas a través de cambios en los localizadores usados puede realizarse en distintas capas de la arquitectura, a saber, en la capa de aplicación, en una capa de sesión, en las capas de transporte o por debajo de la capa de transporte.

La opción preferida para el desarrollo de esta Tesis es por debajo de la capa de transporte, ya sea dentro de la capa IP o en una nueva capa de identificación. La razón para preferir esta opción sobre las otras es esencialmente la complejidad añadida por las otras opciones, complejidad que ya se presentó en el Capítulo 4. En particular, la adopción de una solución en la capa de aplicación requiere la modificación de todas las aplicaciones para que implementen los mecanismos necesarios. Esto sería claramente un esfuerzo redundante además de una complejidad añadida. Similarmente, una adopción en la capa de transporte requiere la modificación de todas las capas de transporte existentes, a saber TCP y UDP, pero también requiere que las nuevas capas de transporte que surjan en el futuro implementen las funciones necesarias para el soporte de multihoming. El caso de la capa de sesión presenta la dificultad de que, al estar más arriba en la pila de protocolos que la capa de transporte, será necesario re-implementar diversas funciones que son implementadas por algunas capas de transporte. Es decir, por ejemplo, la capa TCP provee un servicio fiable basado en la implementación de mecanismo de detección de fragmentos perdidos y la retransmisión de los mismos. Si ahora implementamos por encima de la capa TCP las funciones de recuperación de fallos en la comunicación, será entonces necesario implementar nuevamente los mecanismos necesarios para la comunicación fiable, lo que es claramente una duplicación de esfuerzos y complejidad añadida. Una solución por debajo de la capa de transporte se beneficiaría de estos mecanismos ya implementados en estas capas.

Por ende, la opción elegida es: **implementar los mecanismos para preservar comunicaciones establecidas en un nivel inferior a la capa de transporte, ya sea una nueva capa de identificación o dentro de la capa IP.**

5.3.2 Espacio de identificadores elegido

Como hemos visto, existen múltiples espacios nombres de los cuales se puede extraer los identificadores que usarán las capas superiores (transporte y aplicación). Sin embargo, sólo el uso de identificadores que sean también localizadores válidos preserva las propiedades asumidas por

las aplicaciones actuales y permite la compatibilidad con nodos que no implementan la solución de multihoming. En particular, si consideramos las aplicaciones que utilizan referencias (es decir que intercambian identificadores dentro del la información de aplicación), podemos ver que solamente el uso de identificadores que son a su vez localizadores preserva su correcto funcionamiento. Esto se debe a que una aplicación que utiliza referencias puede enviar la información de localizador a otro nodo que no soporte la solución. Si el identificador usado no es también un localizador, el nodo que recibe (que no implementa la solución) no podrá establecer comunicación con el identificador recibido, ya que no dispondrá de localizadores válidos para este identificador (además de no conocer el uso de identificadores y localizadores separados), por lo que la aplicación fallará. Por esto, la opción elegida en este punto es: **utilizar identificadores que sean también localizadores válidos.**

El análisis realizado anteriormente presenta dos opciones para la validación de identificadores que son localizadores, a saber, el uso de una tercera parte de confianza (como el DNS) o el uso de identificadores que contengan un prefijo que sirva de localizador y un sufijo que contenga información criptográfica. Una solución basada en el uso de una tercera parte de confianza sería específica para el problema de multihoming y requeriría de la administración de la tercera parte de confianza (en el caso del DNS, una solución basada en él impone que todo nodo en el sitio multihomed tenga una entrada directa e inversa en el DNS), lo cual puede ser un requerimiento fuerte para sitios pequeños no administrados como pueden ser las redes domesticas. Creemos entonces, que una solución basada en localizadores criptográficos brinda una solución más atractiva para todo el rango de usuarios posibles y presenta menores costes de gestión.

Por esto, la opción elegida es: **el uso de identificadores que contengan un localizador válido en el prefijo y un sufijo de naturaleza criptográfica.**

5.3.3 Mecanismo de selección de localizadores/caminos

Como hemos visto en el análisis funcional, existen dos enfoques posibles a la selección de caminos y localizadores, a saber, que dicha función sea realizada por los nodos finales o que sea realizada por los routers. En el caso de que la selección de localizadores sea realizada por los routers, es necesario que ambos nodos involucrados en la comunicación soporten la solución para que el enfoque funcione adecuadamente, ya que es necesario que los nodos involucrados en la comunicación reconozcan a los paquetes cuyos localizadores han sido modificados por el sistema de encaminamiento como pertenecientes a la comunicación original asociada a los identificadores usados por los extremos. Esto implica que no sólo es necesario que los nodos internos soporten la solución, sino que los nodos externos (no pertenecientes a un sitio multihomed) también la implementen. Nótese que es necesario que los nodos externos implementen la solución para poder

comunicarse con los nodos del sitio multihomed, incluso si no hay fallos durante la comunicación, ya que la correcta selección de localizadores es necesaria para proveer la compatibilidad con los filtros de ingreso. Una solución basada en la selección de localizadores por parte de los routers requiere su adopción por parte de todos los nodos de Internet (incluso aquellos que no obtienen beneficios directos ya que no son multihomed) para habilitar la comunicación básica (sin tolerancia a fallos) requerida. Claramente este tipo de solución no preserva la comunicación con nodos que no soportan la solución, ni internos, ni externos al sitio multihomed.

Por otra parte, una solución basada en la selección de localizadores por parte de los nodos es compatible con el funcionamiento actual de los nodos, por lo que puede proveer una mejor compatibilidad hacia atrás. En este tipo de solución, el localizador elegido por el nodo final permanece inalterado durante la comunicación, a menos que el nodo mismo decida cambiarlo, por ejemplo si se implementa una solución que así lo requiera para tolerar fallos (en este caso el nodo puede controlar cuándo se cambian los localizadores, por lo que puede asegurarse que si el otro extremo de la comunicación no soporta la solución, los localizadores no cambien). Adicionalmente al mecanismo de selección de localizadores es necesario ofrecer un mecanismo para proveer compatibilidad con los filtros de ingreso, de forma que el localizador seleccionado por el nodo sea compatible con el camino utilizado para transportar el paquete, como se estudia a continuación.

Por las consideraciones de despliegue presentadas, la opción elegida para la selección de localizadores es: **selección de localizadores basada en nodos finales**.

5.3.3.1 Mecanismo para la compatibilidad con los filtros de ingreso

Como se ha presentado en la Sección 3.6, una solución de multihoming debe contar con un mecanismo para asegurar la compatibilidad con los filtros de ingreso que son habitualmente configurados en los ISPs.

A continuación se detallan las distintas opciones para proveer la compatibilidad requerida con los filtros de ingreso que son compatibles con la selección del localizador origen por parte de los nodos finales. Los mecanismos presentados a continuación pueden encontrarse descritos también en [Huitema2004a] [Huitema2004b]

5.3.3.1.1 Relajar los filtros de ingreso

La opción más directa parece simplemente relajar los filtros de acceso para que no descarten los paquetes del sitio multihomed. Existen distintas posibilidades de cómo realizar dicha relajación. Una posibilidad es configurar los filtros para que acepten no solamente el prefijo delegado por el proveedor al cliente, sino además el prefijo delegado por los otros proveedores al

cliente en cuestión. Esta opción requiere una considerable cantidad de administración manual, ya que el ISP deberá configurar a mano los prefijos adicionales permitidos para cada sitio. Nótese que actualmente, el ISP no debe realizar una configuración manual ya que existe la opción de usar el uRPF para definir los filtros de ingreso. El problema es que los filtros generados por el uRPF no contemplan los prefijos adicionales del sitio multihomed.

Otra opción posible es que el ISP confíe al sitio la realización filtrado, es decir que el ISP no configure filtros de ingreso y que estos sean configurados en los routers de salida del sitio multihomed. Esto requiere un considerable nivel de confianza entre el ISP y el sitio, tanto confianza en el buen comportamiento del sitio como en el conocimiento técnico del sitio. Si bien este nivel de confianza existirá en un cierto número de situaciones, en otras no será así, por lo que parece claro que es necesaria una solución alternativa.

5.3.3.1.2 Encaminamiento basado en dirección origen

Otra posibilidad es adaptar el sistema de encaminamiento intra-dominio del sitio multihomed para que considere la dirección origen seleccionada por el nodo en el momento de elegir el camino de salida del sitio hacia Internet. Para ello, es necesario crear un dominio de encaminamiento que no sólo considere la dirección destino, sino también la dirección origen incluida en el paquete. Esto se traduce en que cada router debe mantener múltiples tablas de encaminamiento, una por cada prefijo asignado al sitio multihomed. Cabe notar que el dominio de encaminamiento basado en dirección origen no debe cubrir todo el sitio multihomed y es suficiente con que sea un domino conexo que contenga todos los routers de salida del sitio como se ilustra en la figura 15. De esta forma, el paquete será encaminado por el sistema de encaminamiento usual hasta llegar a cualquier punto del dominio de encaminamiento basado en dirección origen, donde será encaminado hacia el router de salida basándose en la dirección destino y origen.

Configuraciones simplificadas

El dominio de encaminamiento basado en dirección origen requerido toma su mínima expresión cuando las conexiones a todos los ISPs son terminadas en un mismo router. En este caso, sólo este router debe realizar encaminamiento basado en dirección origen, de forma que cada paquete sea encaminado a través del proveedor que ha delegado el prefijo contenido en la dirección origen del paquete.

Otra situación bastante simple es cuando todos los routers de salida están directamente conectados, por ejemplo a una misma LAN. En este caso, sólo es necesario que los routers de salida soporten el encaminamiento basado en dirección origen.

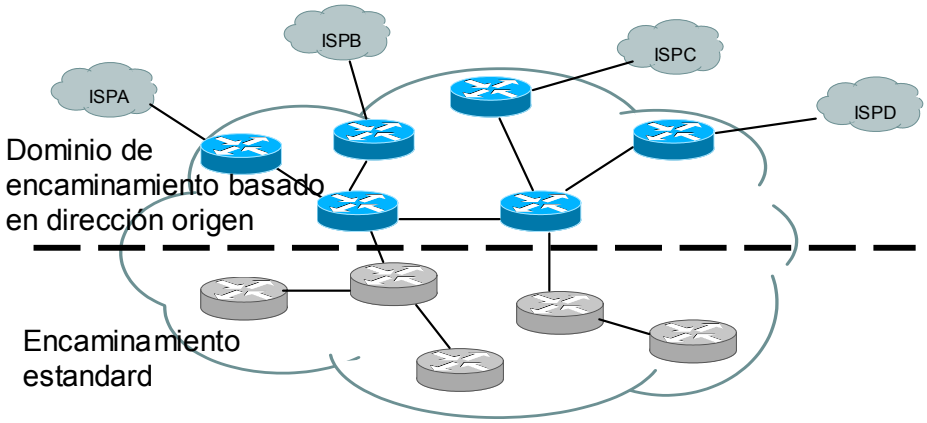

Figura 15: Dominio de encaminamiento basado en dirección origen

Redes Overlay

Cuando existen múltiples routers de salida y estos no comparten un mismo enlace, el dominio de encaminamiento basado en dirección origen necesario será más complejo. Una posibilidad es crear una malla virtual que conecte todos los routers de salida usando túneles IP sobre IP. Esta opción solamente requiere que los routers de salida soporten el encaminamiento basado en dirección origen, lo que simplifica su adopción, pero impone encaminamiento sub-óptimo y disminuye la MTU del camino elegido. La ventaja de este enfoque es que el soporte de encaminamiento basado en dirección origen está limitado a los routers de salida.

Otra posibilidad es que el nodo seleccione el router de salida compatible con la dirección de origen que ha seleccionado y envíe el paquete directamente hacia este router de salida a través de un túnel.

Caso general

Finalmente el caso general es cuando existen múltiples routers de salida y estos no están directamente conectados entre sí. Si no deseamos emplear la malla de túneles descrita en el párrafo anterior, deberemos crear un dominio de encaminamiento basado en dirección origen que contenga todos los routers de salida y todos los routers adicionales necesarios para que el dominio sea conexo. Dependiendo de cuál sea la tecnología de encaminamiento usada en el sitio, el enfoque usado para implantar el encaminamiento basado en dirección origen puede variar.

5.3.3.1.3 Mecanismos seleccionados para la compatibilidad con filtros de ingreso

Como se puede apreciar, existen diversos mecanismos que presentan diferentes características. El enfoque de relajar los filtros de ingreso no se considera apropiado como solución general ya que en muchas situaciones no existirá el nivel de confianza requerido para su aplicación. En los casos donde no se cuente con una topología simplificada como las identificadas

previamente, será necesario utilizar otro de los enfoques presentados. El enfoque basado en la creación de túneles entre los routers de salida permite una adopción rápida y restituir la mínima funcionalidad requerida, es decir que los nodos actuales puedan comunicarse satisfactoriamente al menos cuando no haya fallos. Sin embargo, este enfoque puede sufrir de encaminamiento sub-óptimo y además impone el uso de túneles que incrementan el tamaño de los paquetes transportados y reducen la MTU. Por ello, creemos que este mecanismo puede ser usado transitoriamente hasta el despliegue de una solución más idónea. La solución definitiva propuesta se basa en el despliegue de un encaminamiento basado en dirección origen. Esta opción es superior al uso de túneles entre los nodos y los routers de salida ya que evita las limitaciones de los túneles en términos de reducción de carga útil de los paquetes.

Por esto la opción propuesta es:

- **Encaminamiento basado en dirección origen**
- **Túneles entre los routers de salida como medida transitoria de rápida adopción**

-

Capítulo 6

Solución Propuesta

6.1 Introducción

En este capítulo presentaremos la solución propuesta para la provisión del soporte de multihoming para sitios finales en IPv6. Como ya hemos destacado dentro de los criterios de diseño presentados anteriormente, un aspecto crítico de una solución son las consideraciones para el despliegue y adopción de la misma. Por ello, la solución propuesta en este capítulo es una solución incremental, es decir que la solución consiste en múltiples mecanismos que se despliegan progresivamente y cada uno de ellos brinda funcionalidades adicionales a los anteriores.

El primer mecanismo propuesto permite restablecer las funcionalidades perdidas por los nodos actuales cuando estos son conectados a un sitio multihomed. Como ya hemos visto anteriormente, si conectamos un nodo de los actualmente disponibles a un sitio multihomed, es posible que éste sufra problemas de conectividad con destinos externos al sitio debido a incompatibilidades con los filtros de ingreso. El primer mecanismo necesario es entonces uno que restablezca la funcionalidad obtenida por estos nodos con las mínimas modificaciones posibles,

de forma que se brinde una respuesta de fácil adopción a este problema. Para ello, se propone adoptar una solución basada en la creación de una malla de túneles entre los routers de salida. Una vez subsanado este problema mediante este mecanismo de rápida adopción, presentaremos mecanismos adicionales que permitan mejorar el rendimiento dentro del sitio multihomed, eliminando el encaminamiento sub-óptimo y el incremento en el tamaño de los paquetes intercambiados introducido por el uso de túneles. Para ello se propone la adopción de un sistema de encaminamiento basado en dirección origen dentro del sitio multihomed. Esta opción presenta un rendimiento superior al mecanismo basado en túneles, pero su adopción tiene un coste de implantación superior, como se verá más adelante.

Una vez restaurada la funcionalidad perdida, es posible adoptar mecanismos que brinden algunos de los beneficios del multihoming. Los mecanismos que sólo requieren modificaciones dentro del sitio multihomed serán más fácilmente adoptados. Las razones para esto son: por una parte, éstos mecanismos requieren la modificación de un numero reducido de equipos (los equipos del sitio multihomed) y por otra parte, las modificaciones requeridas por éstos mecanismos sólo deben ser realizadas por quienes obtienen un beneficio de las mismas, generándose así un incentivo para que los involucrados asuman los costos. Mediante modificaciones en los nodos del sitio multihomed es posible permitir que los nodos del sitio en cuestión puedan establecer y recibir nuevas comunicaciones después de un fallo en la conectividad. Para lograr esto, es necesario modificar los mecanismos de selección de localizadores de los nodos del sitio multihomed e implementar mecanismos para que los nodos del sitio multihomed puedan detectar los caminos disponibles y los localizadores asociados a dichos caminos. Mediante modificaciones en los nodos del sitio multihomed podemos proveer cierto grado de tolerancia a fallos (iniciar nuevas comunicaciones después de un fallo) y capacidades de ingeniería de tráfico.

Sin embargo, para preservar las comunicaciones establecidas a través de fallos será necesario no sólo modificar los nodos internos al sitio multihomed, sino también será necesario modificar los nodos externos al sitio en cuestión. Esta última etapa de la solución claramente requiere un esfuerzo mucho mayor de despliegue ya que es necesario modificar todos los nodos externos para lograr el soporte universal, y dado que dichos nodos no obtienen un beneficio directo de los nuevos mecanismos, estos no tienen un incentivo tan claro para su adopción. Por ello, este es el último mecanismo propuesto y se espera que su adopción requiera más tiempo que los mecanismos anteriores.

A continuación brindaremos una descripción detallada de los mecanismos mencionados en el párrafo anterior.

6.2 Primera etapa: Mecanismos para restablecer la funcionalidad perdida

Como ya hemos mencionado anteriormente, en la situación actual, si se conecta un nodo IPv6 en un sitio multihomed que ha obtenido prefijos de sus proveedores, el nodo puede experimentar pérdida de paquetes debido a incompatibilidad con los filtros de ingreso. Para solucionar este problema se propone la adopción de dos soluciones: primero, una solución de muy rápido despliegue pero de rendimiento limitado para solucionar el problema en el plazo inmediato y segundo una solución cuyo coste de despliegue es mayor, pero cuyo rendimiento lo es también, por lo que puede ser adoptada como solución definitiva en el mediano plazo. A continuación se incluye una descripción detallada de ambas.

6.2.1 Fase 1: Solución basada en túneles

Cuando existen múltiples routers de salida es posible crear una malla virtual que conecte todos los routers de salida usando túneles IP sobre IP. Esta opción solamente requiere que sean los routers de salida los que soporten el encaminamiento basado en dirección origen. Para ello, es necesario que cada router de salida tenga conocimiento de la dirección IP del router de salida asociado a cada uno de los prefijos disponibles. Un enfoque posible para esto es asignar a los routers de salida correspondientes a un proveedor dado una dirección *anycast* generada automáticamente a partir del prefijo asociado su proveedor correspondiente, por ejemplo la dirección resultante de concatenar el prefijo /48 asignado por el proveedor con una ristra de 80 bits a 1 (esto asume que el sitio multihomed obtiene un prefijo /48 de cada proveedor como ha recomendado el IETF a los RIRs [RFC3177]). Los routers de salida conectados a un ISP deberán entonces inyectar una ruta hacia la dirección *anycast* correspondiente al proveedor en cuestión a través del IGP. De esta forma, cuando un router de salida recibe un paquete cuya dirección no coincide con el prefijo del ISP que tiene directamente conectado, encapsula el paquete dentro de un túnel hacia la dirección *anycast* del router de salida asociado al prefijo contenido en la dirección origen, asegurando así la compatibilidad con los filtros de ingreso.

El mecanismo resultante es el siguiente:

Tenemos un sitio multihomed con *n* ISPs: ISP1,…, ISPn, y cada ISP ha delegado un prefijo al sitio multihomed, a saber Perf1/48,..., Prefn/48

El sitio multihomed tiene *n* routers de salida, R1,…,Rn, cada uno de los cuales está conectado al ISP correspondiente.

En el primer paso, cada router de salida genera la dirección *anycast*, que le corresponde, concatenando una ristra de 80 bits a 1 al prefijo delegado por el ISP al que está directamente conectado, a saber, Pref*i*: FFFF: FFFF: FFFF: FFFF: FFFF para *i*=1,..,n

Después, cada router de salida anuncia una ruta de host hacia la dirección *anycast* que le corresponde. Es decir que el R*i* anunciará una ruta hacia Prefi:FFFF: FFFF: FFFF: FFFF: FFFF /128 para todo *i*=1,…, *n*.

Entonces cuando un paquete cuya dirección origen contiene Prefj/48 es recibido por el router de salida R*i*, el R*i* realizará las siguientes operaciones:

- Si *i=j*, entonces encamina el paquete a través del router que tiene directamente conectado, ya que los prefijos coinciden.
- Si *i≠j*, entonces R*i* buscará si existe una ruta hacia Prefj: FFFF: FFFF: FFFF: FFFF: FFFF /128
    - Si la ruta existe, entonces encapsula el paquete recibido en un túnel, cuya dirección de origen es la de R*i* y la dirección destino es Prefj: FFFF: FFFF: FFFF: FFFF: FFFF y manda este paquete
    - Si la ruta no existe, descarta el paquete e informa al nodo origen del fallo a través de un paquete de error ICMP (Cabe notar que la nueva versión de la especificación de ICMP [Conta2004a] define un nuevo código de error 5 el cúal corresponde a un problema con los filtros de ingreso (*Destination Unreachable with Code 5* (*Source Address Failed Ingress Policy*))

Esta solución es de rápida adopción ya que los routers disponibles actualmente soportan las funciones requeridas para su implementación. Sin embargo, esta solución presenta diversas limitaciones que la hacen poco recomendable como solución definitiva. Dentro de las limitaciones identificadas podemos mencionar:

- Reducida tolerancia a fallos. Un paquete pasa por más de un router para salir del sitio, por lo que es sensible a fallos en cada uno de estos.
- Encaminamiento sub-optimo. El camino usado para salir del sitio en la mayoría de los casos no será el óptimo, debido a que el paquete recorrerá más de un router de salida. Esto es especialmente grave en situaciones donde los router de salida están muy dispersos.
- Reducción en la MTU. Debido al uso de túneles en la solución, la MTU se reduce. Esto puede implicar dificultades si los paquetes tienen un tamaño superior al de la MTU permitida menos 40 bytes (tamaño del encabezado IPv6)

Por esto, es necesario proponer otro mecanismo como solución definitiva. La solución presentada en esta sección será entonces adoptada como una medida paliativa transitoria, hasta el desarrollo de la solución definitiva.

6.2.2 Fase 2: Encaminamiento basado en dirección origen

Cuando existen múltiples routers de salida debemos crear un dominio de encaminamiento basado en dirección origen que contenga todos los routers de salida y todos los routers adicionales necesario para que el dominio de encaminamiento basado en dirección origen sea conexo. Dependiendo de cuál sea la tecnología de encaminamiento usada en el sitio, el enfoque usado para implantar el encaminamiento basado en dirección origen puede variar. A continuación estudiaremos tres escenarios, a saber, el uso de rutas estáticas en el intra-dominio, el uso de BGP sin redistribución de rutas en el IGP y el uso de un IGP para seleccionar el camino de salida.

6.2.2.1 Rutas Estáticas

En el caso de que dentro del sitio multihomed haya *m* proveedores y se utilicen rutas estáticas, es necesario configurar rutas adicionales por cada router contenido en el dominio de encaminamiento basado en dirección origen. Si asumimos que solamente se han configurado filtros de ingreso en los ISPs y que los paquetes pueden circular libremente dentro del sitio, sólo es necesaria la configuración de rutas adicionales para los destinos que estén fuera del sitio multihomed. Esto implica que si tenemos:

- Un sitio multihomed que tiene *m* proveedores
- Un dominio de encaminamiento basado en dirección origen que contiene *r* routers
- El número medio de rutas externas por router es *e*

Entonces, será necesario configurar $m*r*e$ rutas en todo el dominio considerado para el correcto funcionamiento del encaminamiento basado en dirección origen. Esto implica que será necesario configurar $(m-1)*r*e$ rutas adicionales respecto al encaminamiento actual basado solamente en la dirección destino. Cabe notar que actualmente los sitios multihomed no utilizan rutas estáticas, ya que estas no ofrecen las capacidades de tolerancia a fallos buscadas con dicha configuración. Sin embargo, como veremos más adelante, el uso de rutas estáticas sí es posible dentro del marco de una solución de multihoming para IPv6 debido a que la selección de rutas puede ser realizada directamente por los nodos, sin requerir el uso de un protocolo de encaminamiento dinámico para modificar el camino a seguir por los paquetes.

6.2.2.2 BGP sin redistribución en el IGP

Otro caso a considerar es cuando el sitio multihomed utiliza BGP con sus proveedores para obtener la información de encaminamiento de cada uno de sus ISPs. Es habitual que por razones

operativas los sitios no redistribuyan la información de rutas a sus IGPs[14] [Beijnum2002a]. Considerando que el IGP no posee información completa de todas las rutas externas, es necesario entonces incluir routers internos en la malla I-BGP además de los routers de borde (incluidos por defecto en I-BGP). En particular, es necesario que la malla de routers I-BGP forme un dominio conexo que contenga todos los routers de borde. Esto permite que una vez que un paquete se encuentra en este dominio, el paquete pueda ser encaminado dentro de este dominio hacia el router de salida más apropiado para alcanzar el destino final del paquete, basándose en la información de rutas ofrecida por BGP. La malla I-BGP formará entonces una dominio conexo que une el resto del sitio (que no tiene información de rutas externas) con Internet, como se ilustra en la figura siguiente.

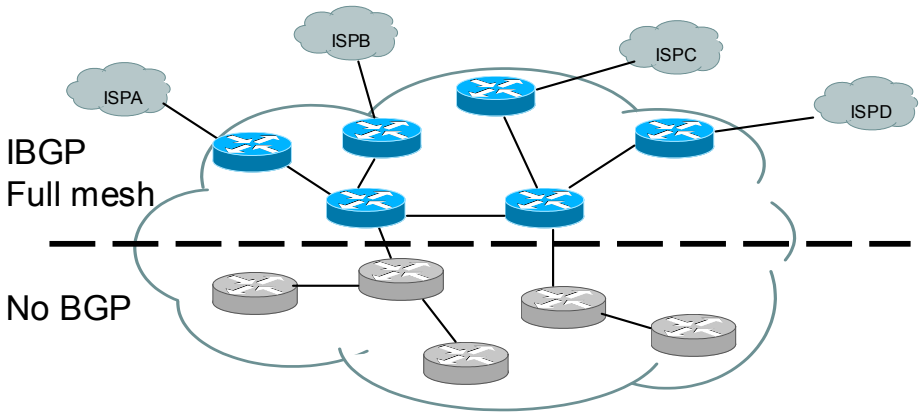

Figura 16: Dominio iBGP

Cabe notar que sólo el dominio I-BGP debe ser conexo y que el dominio no-BGP puede estar formado por múltiples dominios no conexos (conectados a través del dominio I-BGP)

Resulta claro entonces que para adoptar el encaminamiento basado en dirección origen es suficiente con que éste sea soportado por los routers contenidos en el dominio I-BGP. Por ello, es posible utilizar las capacidades de BGP para proveer configuración dinámica de rutas al encaminamiento basado en dirección origen. Para ello es necesario que cada router de borde *coloree* la información de rutas recibida del ISP directamente conectado con un color asociado al ISP en cuestión. La información de color puede estar contenida en el atributo `community`

[14] Por redistribuir la información obtenida por BGP dentro del IGP se entiende que la información obtenida por BGP se inyecta en el IGP, por lo que todos los destinos aprendidos a través de BGP son conocidos por el IGP. Las dificultades operativas son debidas al gran volumen de información manejado por BGP, así como a la inestabilidad de dicha información, que al inyectarla en el IGP, generan a su vez inestabilidades en el dominio local

[RFC1997] disponible en BGP. Mediante este mecanismo, es posible que los demás routers del dominio de encaminamiento basado en dirección origen sean capaces de identificar al proveedor a través del cual se ha aprendido dicha información y por ende, poder deducir el prefijo a incluir en la dirección origen al enviar paquetes por la ruta en cuestión. Para esto es necesario entonces asociar a cada color usado uno de los prefijos disponibles en el sitio multihomed. Si se ha establecido una relación entre colores usados y prefijos a utilizar en la dirección origen, y los routers de borde colorean la información intercambiada por I-BGP cuando es recibida, es posible que los routers actualicen las distintas tablas de encaminamiento asociadas a cada uno de los prefijos disponibles con la información de rutas con el color correspondiente a dicho prefijo.

En resumen, en un entorno donde un sitio multihomed tiene *n* proveedores, cada uno de los cuales ha asignado el prefijo Pref_*i* *i*=1,…,*n* al sitio multihomed, el funcionamiento de distribución de información de rutas para el encaminamiento basado en dirección origen usando el atributo de `community` en BGP sería el siguiente:

- Primero se asigna un valor de `community` del rango reservado para uso privado a cada uno de los ISPs. De esta forma, el valor Com_*i* se liga a las rutas anunciadas por el ISP_*i*, siendo *i*=1,…,*n*
- Segundo, se crean *n* tablas de rutas en cada uno de los routers pertenecientes al dominio de encaminamiento basado en dirección origen, y se relaciona cada una de las tablas con uno de los prefijos Pref_*i* (*i*=1,…,*n*) existentes en el sitio multihomed. Adicionalmente, se configuran los routers de forma que encaminen los paquetes que contienen el prefijo Pref_*i* (*i*=1,…,*n*) en la dirección origen usando la tabla asociada a dicho prefijo.
- Tercero, se configuran las reglas de procesamiento de rutas de BGP de forma que las rutas recibidas que contienen el atributo `community` con el valor Com_*i* afecten solamente a la tabla de encaminamiento asociada al prefijo Pref_*i* (para *i*=1,…,n)
- Finalmente, se configuran los routers de borde de forma que el router de borde que conecta al sitio con el ISP*i*, marque las rutas recibidas con un atributo `community` con valor Com_*i* cuando las anuncia por I-BGP.

El comportamiento resultante es que cada router dentro del dominio I-BGP tendrá múltiples tablas de rutas, cada una de las cuales estará asociada a uno de los prefijos de dirección origen disponible en el sitio, y cada tabla contendrá la información de rutas aprendida a través del ISP que ha delegado el prefijo en cuestión.

6.2.2.3 IGP para la selección del camino de salida

En este caso, el sitio multihomed utiliza el IGP para distribuir información sobre rutas internas y externas. El IGP puede aprender la información de rutas externas por tres mecanismos:

- rutas manualmente configuradas e importadas al IGP,
- redistribución de la información aprendida por BGP en el IGP, o
- intercambiando información de rutas con sus proveedores usando el mismo IGP.

Como en el caso anterior, no es necesario que todo el sitio adopte el encaminamiento basado en dirección origen, sino que basta con que un dominio conexo que contenga todos los routers de salida lo implemente. La dificultad en este escenario estriba en las escasas facilidades de coloreado de información que suelen presentar los IGPs comúnmente usados. Una solución posible es correr múltiples instancias del IGP, una por prefijo presente en el sitio multihomed. De esta forma, cada una de las instancias del IGP actualizará la tabla de rutas correspondiente al prefijo en cuestión. Una dificultad adicional que se presenta es cómo distinguir las diferentes instancias del protocolo que se ejecutan simultáneamente en la misma red. Algunos protocolos como OSPFv6 [RFC2740] soportan la ejecución de múltiples instancias simultáneas mientras que otros como RIP son más limitados en ese aspecto. El comportamiento resultante es similar al descrito en el punto anterior, solo que el lugar del atributo `community` lo ocupará la instancia del protocolo IGP usado.

6.2.3 Situación resultante

Una vez que se ha adoptado cualquiera de estas soluciones, la funcionalidad perdida queda restablecida. Esto quiere decir que es posible conectar con el exterior del dominio un nodo que no cuente con soporte de multihoming en un sitio que implemente cualquiera de estas soluciones, y este nodo obtendrá un servicio similar al que obtendría si es conectado en un sitio single-homed. Es decir que las incompatibilidades con los filtros de ingreso introducidas por el multihoming son subsanadas mediante estas soluciones. En las secciones siguientes veremos cómo es posible implementar mecanismos adicionales que permitan a los nodos del sitio multihomed beneficiarse de facilidades adicionales provistas por el multihoming.

Estos mecanismos se encuentran especificados en [Huitema2004a] y [Huitema2004b]. El autor de la presente tesis ha desarrollado los mecanismos de SADR (*Source Address Dependent Routing*, Encaminamiento Dependiente de la Dirección Fuente) y estos estudios han sido publicados en [Bagnulo2004a] y en [Bagnulo2005b]

6.3 Segunda Etapa: Mecanismos intra-sitio para soporte de multihoming

Una vez desplegados los mecanismos de la primera etapa, es posible conectar un nodo a un sitio multihomed y que éste obtenga un servicio similar al que obtendría en un sitio no-multihomed. Si bien esto restaura las funcionalidades perdidas, el objetivo buscado en el momento de la adopción del multihoming por parte del sitio es más ambicioso que simplemente obtener un servicio similar a un sitio single-homed. Como ya hemos presentado anteriormente, los objetivos del sitio que adopta el multihoming incluyen una mejora en la tolerancia a fallos y ciertas capacidades de ingeniería de tráfico que permitan encaminar los paquetes a través de los caminos preferidos. En esta sección presentaremos un conjunto de mecanismos cuyo objetivo es la provisión de estas funcionalidades adicionales. El conjunto de mecanismos presentados en esta sección afectan solamente al sitio multihomed y no requieren modificaciones al comportamiento de los nodos exteriores al sitio multihomed. Esto hace que sólo sea necesario modificar los nodos que obtienen un beneficio directo del mecanismo, por lo que es esperable que la adopción de estos mecanismos sea considerablemente simple y rápida. En las secciones siguientes presentaremos primero los mecanismos necesarios para mejorar la tolerancia a fallos del sitio y luego los mecanismos para la provisión de facilidades de ingeniería de tráfico. Los mecanismos presentados asumen la existencia de al menos una de las soluciones presentadas en la sección anterior. Es decir, se asume que la dirección origen contenida en un paquete determina el ISP usado para encaminar dicho paquete, ya que se supone la existencia de algún mecanismo de encaminamiento basado en la dirección origen. Esto implica que los nodos del sitio multihomed disponen de un mecanismo para forzar el camino de salida y que estos pueden cambiar de camino de salida si cambian la dirección origen del paquete en cuestión.

6.3.1 Mecanismos intra-sitio para la mejora de la tolerancia a fallos

Existen distintos niveles de tolerancia a fallos que pueden ser alcanzados en un sitio multihomed, a saber:

- Establecimiento de nuevas comunicaciones entrantes después de un fallo.
- Establecimiento de nuevas comunicaciones salientes después de un fallo.
- Preservar comunicaciones establecidas a través de fallos

Como veremos en esta sección, es posible diseñar mecanismos para establecer nuevas comunicaciones entrantes y salientes que estén basados en las especificaciones existentes y que no requieran cambios en los nodos exteriores. Por el contrario, los mecanismos requeridos para preservar conexiones establecidas, requiere modificaciones en ambos extremos de la comunicación, por lo que la presentación de estos mecanismos será postergada hasta una tercera etapa.

A continuación presentaremos los mecanismos para el establecimiento de nuevas comunicaciones entrantes y luego pasaremos a los necesarios para el establecimiento de nuevas comunicaciones salientes.

6.3.1.1 Mecanismos para el establecimiento de nuevas comunicaciones entrantes al sitio después de un fallo

En este caso, tenemos un sitio multihomed que tiene n proveedores, ISP1,…, ISPn cada uno de los cuales ha delegado un prefijo Perf1::/48,…,Prefn::/48 al sitio multihomed. Por ende, el sitio multihomed es accesible a través del ISPi si la dirección usada contiene el prefijo Prefi::/48.

Asumimos que los nodos del sitio multihomed desean ser accedidos a través de todos los proveedores, por lo que incluirán múltiples direcciones (una por prefijo disponible) asociadas a su FQDN en el DNS.

Cuando un nodo externo NE, desea establecer una comunicación con un nodo mhN del sitio multihomed, el nodo NE realizará una búsqueda en el DNS por el nombre de mhN. Esta búsqueda devolverá al nodo NE todas las direcciones del nodo mhN, una por cada proveedor.

Una vez que el nodo NE tiene las múltiples direcciones del nodo mhN, el algoritmo de selección de direcciones definido en la RFC 3484 [RFC3484] ordenará la lista de direcciones y se la pasará a la aplicación que desea establecer una comunicación con el nodo mhN. Los criterios usados para ordenar las direcciones son variados, pero en principio es posible que si ha habido un fallo recientemente, la dirección con más prioridad en la lista no se encuentre alcanzable.

La aplicación debe utilizar la primera dirección de la lista para iniciar la comunicación. Si la comunicación falla, la RFC 3484 especifica que la aplicación debe reintentar establecer la comunicación con la próxima dirección incluida en la lista. Entonces, según las especificaciones vigentes, en particular la RFC 3484, una aplicación debe intentar establecer comunicación con todas las direcciones disponibles. Esto significa que si se produce un fallo, este mecanismo existente permitirá la comunicación a través de un proveedor alternativo, ya que se intentará con direcciones (es decir prefijos) alternativos. Por ende, para la permitir el establecimiento de nuevas comunicaciones entrantes después de un fallo, sólo es necesario que el sitio multihomed publique en el DNS las direcciones generadas a partir de los distintos prefijos delegados por los distintos

proveedores. Una vez que las mismas se encuentran disponibles en DNS, los nodos externos las utilizarán en los distintos intentos de establecimiento de comunicación.

6.3.1.2 Mecanismos para el establecimiento de nuevas comunicaciones salientes del sitio después de un fallo

En este caso, el nodo que inicia la comunicación pertenece al sitio multihomed, de forma que tendrá disponibles múltiples direcciones origen y deberá elegir una de ellas, considerando que, en caso de fallo, se deberá evitar el uso de la dirección correspondiente al proveedor asociado al camino que ha sufrido un fallo.

El escenario es entonces un sitio multihomed con n proveedores, ISP1,…nISPn, cada cual ha asignado un prefijo al sitio multihomed, Pref1::/48,…, Prefn::/48.

Un nodo del sitio multihomed mhN tendrá entonces n direcciones disponibles, Pref1::mhN,…,Prefn::mhN

Supongamos entonces que mhN desea iniciar una comunicación con un nodo NE externo al sitio. Asumiremos que este nodo NE cuenta con una sola dirección para simplificar la explicación y consideraremos el caso general posteriormente.

El nodo mhN debe entonces seleccionar la dirección origen a utilizar para iniciar esta comunicación. El algoritmo para la selección de dirección origen esta especificado en la RFC 3484, y esencialmente define una serie de reglas a través de las cuales se prefiere una dirección origen sobre otras, siguiendo criterios que detallaremos más adelante pero que debido a la limitada información que un nodo final tiene sobre el estado de la red, no puede considerar el estado de la red y los posibles fallos ocurridos. Por ende es posible que el nodo mhN seleccione una dirección origen Prefi::mhN que corresponda a un proveedor que no disponga de un camino hacia NE debido a un fallo reciente. En este caso, el nodo mhN seleccionará la dirección Prefi::mhN y enviará el paquete. El sistema de encaminamiento basado en dirección origen encaminará el paquete hacia el proveedor ISPi, quien al no contar con un camino disponible hacia el destino, deberá descartar el paquete. Al descartar el paquete, el ISPi generará un mensaje ICMP de error, informando del paquete descartado. Sin embargo, debido al frecuente filtrado de los mensajes de ICMP, no es posible confiar en la recepción de dicho mensaje para la detección de un problema, por lo que son necesarios mecanismos alternativos de detección. Adicionalmente, cabe notar que en este caso, será necesario cambiar la dirección origen usada y no la dirección destino, como en el caso anterior. La especificaciones actuales no requieren que las aplicaciones reintenten con una dirección origen alternativa, es más, habitualmente, la dirección origen usada es transparente a las aplicaciones. Por ello, cuando una comunicación como la descrita

anteriormente falla, y no hay direcciones destino alternativas, la aplicación no reintentará establecer la comunicación.

En resumen, para solventar la situación es necesario brindar las siguientes dos funcionalidades adicionales, a saber:

- un mecanismo de detección de caminos con fallos, y
- un mecanismo de retransmisión.

Dichas funcionalidades debe ser brindadas de forma transparente a las aplicaciones ya que en caso contrario sería necesaria la modificación de las aplicaciones, lo que dificultaría la adopción de la solución.

Para proveer estas funcionalidades, uno puede pensar en un mecanismo basado en el ensayo y error. Es decir, cuando el nodo mhN desea iniciar una comunicación con ME, escoge una dirección origen según el algoritmo definido en la RFC 3484. Si la comunicación es exitosa, sigue utilizando esta dirección. Si la comunicación falla, intenta con otra dirección origen, imponiendo así el uso de un proveedor alternativo. El problema fundamental con este mecanismo es que no es siempre posible saber cuándo una comunicación ha sido exitosa. Un mecanismo para la detección de una comunicación exitosa podría basarse en identificar el tráfico en sentido inverso. Es decir si el nodo mhN recibe paquetes en respuesta al paquete inicial, esto quiere decir que el paquete inicial ha llegado a destino. Sin embargo, este mecanismo no permite distinguir flujos unidireccionales y caminos con fallos. Es decir, cuando una aplicación genera un flujo unidireccional, ningún paquete de respuesta será recibido por el nodo mhN, lo que llevará al mecanismo a diagnosticar que existe un fallo, por lo que intentará la retransmisión por un camino alternativo. Éste no es el comportamiento deseado. Por otra parte, este mecanismo requiere que la capa que lo implementa (posiblemente la capa de red) almacene una copia de los paquetes enviados, para que pueda reenviarlos después con una dirección origen alternativa.

Otro mecanismo puede basarse en el envío simultáneo de tantos paquetes como direcciones origen se encuentran disponibles en el nodo. Es decir que cuando una aplicación corriendo sobre el nodo mhN quiere iniciar un el nodo NE, ésta pasa un paquete a la capa de red. La capa de red genera tantas copias del paquete como direcciones origen existen en mhN y las envía. Esto implica que se envía una copia por cada uno de los proveedores disponibles. La primera respuesta recibida (en caso de los flujos bidireccionales) es la dirección origen a usar. Este mecanismo parece brindar un buen soporte de tolerancia a fallos, aunque su coste es considerable, ya que es necesario enviar múltiples copias de los paquetes iniciales. Cabe notar que sólo es necesario enviar múltiples paquetes de los paquetes iniciales, ya que los paquetes posteriores utilizarán la dirección origen identificada como óptima tras el envío del paquete inicial. El tratamiento de los flujos unidireccionales requiere medidas adicionales que veremos a continuación, pero este mecanismo permite un soporte razonable de dichos flujos, ya que no basa su estrategia de recuperación de errores en la recepción de paquetes de respuesta como era el caso del mecanismo

anterior. Este mecanismo, no requiere el almacenamiento del paquete, ya que los paquetes no se retransmiten. Finalmente, el tiempo de respuesta del mecanismo es óptimo, ya que no impone latencia adicional cuando existen caminos con fallos, ya que todos los caminos son usados simultáneamente. Es más, el mecanismo selecciona el camino con menor retardo hacia el destino. Finalmente, cabe mencionar que el nodo NE en general recibirá múltiples copias del paquete inicial. Sin embargo, esto no parece ser un problema, ya que los paquetes subsiguientes a la llegada del primer paquete serán tratados como duplicados y descartados. Por todo esto, consideramos que este último mecanismo es el más idóneo para brindar las funcionalidades requeridas. A continuación incluimos una descripción detallada del funcionamiento del mecanismo.

El mecanismo propuesto utiliza una estructura de datos que llamaremos Caché de Camino de Salida, la cual almacenará por cada dirección destino una dirección origen correspondiente y un tiempo de vida de esta información.

El proceso de creación de las entradas de la Caché de Camino de Salida es el siguiente:

Cuando el nodo mhN recibe un paquete con dirección destino DA y dirección origen SA, se realiza una búsqueda en la Caché de Camino de Salida por una entrada que contenga SA como dirección destino.

- Si existe una entrada que contenga SA como dirección destino, se verifica la dirección contenida como dirección origen en dicha entrada.
    - Si la dirección contenida en el campo dirección origen de la entrada en cuestión de la Caché de Camino de Salida es DA, entonces el tiempo de vida de dicha entrada es extendido.
    - Si la dirección contenida en el campo dirección origen de la entrada en cuestión de la Caché de Camino de Salida es distinta a DA, entonces la entrada es actualizada y DA es almacenada como dirección origen y el tiempo de vida es extendido.
- Si no se encuentra una entrada que tenga SA como dirección destino, se crea una nueva entrada en la tabla con SA como dirección destino y DA como dirección origen. Esta entrada es bloqueada durante un cierto tiempo, de forma que la recepción de múltiples paquetes consecutivos no afecten a la entrada.

El mecanismo para iniciar comunicaciones por parte del nodo mhN será entonces el siguiente:

El nodo mhN desea iniciar una comunicación (típicamente, abriendo un *socket*) con el nodo NE.

1. El nodo mhN verifica la Caché de Camino de Salida en busca de una entrada que contenga la dirección NE en el campo dirección destino.

2. Si dicha entrada existe, el nodo mhN utiliza la dirección contenida en el campo dirección origen de dicha entrada
3. Si dicha entrada no existe, el nodo mhN genera tantos paquetes como direcciones disponibles existan en mhN y lanza un temporizador.
4. Si se recibe un paquete de respuesta a alguno de los paquete enviados, mhN genera una entrada en la Caché de Camino de Salida como se detalla en la sección anterior, por lo que los paquetes siguientes hacia ese destino utilizarán la dirección origen incluida en la Caché de Camino de Salida.
5. Si el temporizador expira y aún no se ha recibido un paquete de respuesta, esto puede significar que no existe ningún camino disponible hacia el destino o que se trate de un flujo unidireccional. En cualquier caso, esta situación no puede ser resuelta con la información disponible en este nivel, por lo que el comportamiento propuesto es que el nodo mhN seleccione la última dirección origen utilizada con la dirección destino en cuestión (aunque haya expirado) y en caso que no se encuentre disponible información sobre está dirección, seleccione una dirección origen al azar. Esta dirección así elegida se utiliza para enviar paquetes. Para evitar el envío de múltiples paquetes cada vez que esta situación ocurre, es necesario crear una entrada en la Caché de Camino de Salida con la dirección origen elegida. El mecanismo de detección de caminos disponibles se ejecutará entonces nuevamente cuando esta entrada expire.

Este mecanismo ha sido una contribución del autor de esta tesis a [Huitema2004a] y a [Huitema2004c].

6.3.1.2.1 Optimizaciones posibles

Una optimización posible que puede simplificar el mecanismo de selección de dirección origen consiste en deprecar los prefijos asociados a aquellos proveedores con los que se ha perdido conectividad. Es decir, supongamos que se ha perdido la conectividad con el proveedor ISPi. En este caso, es deseable que los nodos del sitio multihomed no utilicen direcciones que contengan el prefijo Prefi::/48 como dirección origen, ya que este proveedor no se encontrará disponible. Para ello, es posible deprecar este prefijo en el sitio. El mecanismo completo para esto cuenta de los siguientes componentes:

- Mecanismo de detección de ISPs no disponibles
- Mecanismo para deprecar prefijos.

El mecanismo para detectar cuándo un ISP no se encuentra disponible puede basarse en distintas herramientas. En el caso en que se ejecute BGP entre el sitio multihomed y sus proveedores, BGP permite la detección de perdida de conectividad entre dos pares BGP. Si no se utiliza BGP, es

posible utilizar otras herramientas como pueden ser BFD [Katz2005a] o incluso un ping entre el router de salida del sitio y el router de borde del proveedor.

Una vez que se ha detectado un fallo en un ISP, es necesario un mecanismo para deprecar el prefijo asociado a dicho ISP. Para esto es posible utilizar los mensajes de Router Advertisement [RFC2461] para los nodos directamente conectados al router de salida en cuestión, y para los demás nodos y routers del sitio será necesario el uso de herramientas de alcance superior a un enlace, como el protocolo de Router Renumbering o la opción de delegación de prefijos de DHCP [RFC3315] [RFC3633].

Finalmente cabe notar que este último mecanismo no es más que una optimización y no es suficiente para cubrir el caso general, ya que existen modos de fallo en donde aunque el proveedor se encuentra disponible, no existe un camino hacia el destino deseado a través del mismo. Un ejemplo de este tipo de fallos es cuando un proveedor tiene problemas con su conectividad a Internet. En este caso, el resto de clientes de este proveedor serán solamente accesible a través de este proveedor, mientras que el resto de Internet será accesible a través de los proveedores alternativos del sitio multihomed.

6.3.1.3 Situación resultante

Una vez adoptados los mecanismos descritos en las secciones anteriormente presentadas por parte del sitio multihomed, el sitio en cuestión disfrutará de capacidades de tolerancia a fallos mejoradas, permitiendo el establecimiento de nuevas comunicaciones bidireccionales después de un fallo a través de los caminos alternativos disponibles (en el caso de los flujos unidireccionales, el nivel de soporte es limitado ya que la capa de red no tiene realimentación del progreso satisfactorio de la comunicación). Cabe resaltar que los nuevos mecanismos necesarios sólo tienen que ser adoptados por el sitio multihomed para que éste pueda obtener los beneficios de los mismos. A continuación veremos los mecanismos necesarios para la provisión de capacidades de ingeniería de tráfico.

6.3.2 Mecanismos intra-sitio para la provisión de ingeniería de tráfico

En esta sección presentaremos un conjunto de mecanismos que permiten al sitio multihomed realizar ingeniería de tráfico, es decir encaminar los paquetes desde/hacia su sitio según unos ciertos criterios que van mas allá de la selección por defecto realizada por los protocolo de encaminamiento y que tienen en cuenta otras consideraciones. Empezaremos recapitulando las funcionalidades de ingería de tráfico disponibles en la solución basada en BGP usada en IPv4 y

luego presentaremos un conjunto de mecanismos cuyo objetivo es brindar funcionalidades similares a las ahora existentes.

6.3.2.1 Capacidades de ingeniería de tráfico disponibles en la solución basada en BGP usada en IPv4

El escenario considerado en esta sección es un sitio multihomed mhS que tiene un solo prefijo asignado Pref y lo anuncia a través de sus múltiples proveedores ISP1,…,ISPn usando BGP. A su vez, estos proveedores lo anuncian mediante BGP a sus respectivos proveedores, y así sucesivamente hasta llegar a la zona libre de rutas por defecto de Internet. A continuación describiremos las capacidades de ingeniería de tráfico existentes en este escenario. Empezaremos por describir las posibilidades para influir el tráfico entrante y luego describiremos las posibilidades para el tráfico saliente.

6.3.2.1.1 Tráfico entrante

Para influir en tráfico entrante, el sitio multihomed mhS puede utilizar las capacidades de los anuncios de rutas de BGP. A partir de lo que se ha descrito en la presentación realizada del protocolo BGP, podemos ver que existen dos posibilidades para influir en el tráfico entrante:

- Anuncios más específicos. Debido a que el encaminamiento IP se rige por la regla del *longest prefix match,* podemos anunciar prefijos más específicos a través de uno de los proveedores, de forma que esta ruta sea elegida debido a que es más específica.
- *AS path prepending*: Como hemos presentado anteriormente, BGP considera el largo del camino de ASs cuando tiene que elegir entre dos rutas que tienen iguales valores para otras características más prioritarias. Esto permite que si el sitio multihomed añade ASs de forma artificial en la ruta anunciada a través de uno de los proveedores (por ejemplo su propio número de AS repetido una cierta cantidad de veces), la ruta con longitud del camino de ASs más corta será preferido sobre la ruta con camino de ASs más largo anunciada por el otro proveedor. El efecto resultante es que el tráfico es encaminado a través de la ruta por el proveedor cuya ruta tiene un camino de ASs más corto[15].

[15] Nótese que el mecanismo permite regular el flujo que entra por cada proveedor, ya que si el número de ASs añadidos artificialmente es bajo, puede haber sitios para los que la ruta más corta se encuentre por el proveedor por el que se propaga la ruta incrementada. Al añadir ASs al camino propagado, aumenta el porcentaje de tráfico que se recibe por el proveedor considerado como preferido. Nótese que la granularidad con la que se puede realizar el ajuste de tráfico por cada proveedor depende de la configuración de los sistemas autónomos externos y de la topología de la red.

6.3.2.1.2 Tráfico saliente

Para influir en el proveedor usado para transportar el tráfico saliente, el sitio multihomed debe configurar sus routers BGP para preferir las rutas anunciadas por un proveedor sobre aquellas anunciadas por los demás proveedores. Para ello, basta con configurar que las rutas anunciadas por el proveedor preferido sean marcadas con una mayor preferencia local (`Local Preference`). De esta forma si existen múltiples rutas anunciadas por distintos proveedores, la ruta anunciada por el proveedor elegido será la seleccionada en todos los routers gracias al atributo de `Local Preference`

6.3.2.1.3 Consideraciones iniciales sobre los mecanismos para ingeniería de tráfico en IPv6

Como ya hemos visto en repetidas ocasiones, en la configuración de multihoming que estamos considerando, un sitio multihomed obtiene un prefijo de cada uno de sus proveedores y la elección del prefijo al que pertenezca la dirección de nodo del sitio multihomed usada para la comunicación determinará el proveedor del sitio multihomed usado durante la misma (asumiendo que la dirección usada permanece inalterada durante la comunicación). Esto implica que quien elija la dirección del nodo del sitio multihomed será quien determine el proveedor usado para dicha comunicación, tanto para los paquetes entrantes como para los paquetes salientes de esta comunicación. Ésta es la primera diferencia con el caso anteriormente presentado: en el uso de BGP existen distintos mecanismos para el tráfico saliente y para el tráfico entrante. En este caso, existirán mecanismos para influir en el proveedor usado en comunicaciones iniciadas por nodos externos y otros mecanismos para influir en el proveedor usado en comunicaciones iniciadas por nodos del sitio multihomed.

6.3.2.1.4 Mecanismos para ingeniería del tráfico de comunicaciones iniciadas por nodos externos

Desde el momento en que el proveedor queda determinado por la selección de la dirección de nodo del sitio multihomed a usar durante la comunicación, parece claro que en el caso en que la selección de direcciones la realiza un nodo externo, no es fácil influir en la misma, y primarán las políticas del propio nodo o sitio. No obstante, a través del DNS se puede influir en la selección de direcciones cuando ésta es realizada por nodos externos.

El mecanismo de selección de dirección destino definido en la RFC 3484 incluye una serie de reglas que ordenarán a las direcciones obtenidas a través de una consulta al DNS. Sin embargo, cuando un nodo multihomed tiene múltiples direcciones globales, es esperable que en muchos

casos ninguna de las reglas afecte al orden en el cual el DNS ha devuelto la lista de direcciones. Por esto, el orden en el que el DNS devuelve las direcciones puede efectivamente afectar la dirección finalmente utilizada en los casos en que la lista de direcciones permanece inalterada por el mecanismo de selección de direcciones destino del nodo externo. Es posible programar el servidor DNS del sitio multihomed para que devuelva las distintas direcciones disponibles en el sitio para lograr los patrones de tráfico deseado. Por ejemplo en el caso en que el sitio disponga de dos proveedores, es posible configurar el DNS para que devuelva en primer lugar direcciones del primer proveedor un x% de las veces y que el restante (100-x)% de las veces devuelva primero la dirección correspondiente al otro ISP. Otra herramienta que puede ser utilizada para lograr un efecto similar son los registros SRV [RFC2782]. Este mecanismo puede ser bastante preciso, pues como ya hemos dicho, es esperable que un gran porcentaje de los nodos externos no modifiquen el orden en que el DNS les entrega las direcciones.

Está claro que el orden de las direcciones sí puede ser alterado debido a políticas locales al nodo externo. Pero esto no constituye ninguna novedad respecto a la solución basada en BGP, donde el sistema externo puede hacer valer sus propias políticas al elegir el camino a usar para llegar al destino cuando existen múltiples caminos disponibles (recuérdese que independientemente del largo del camino de ASs incluido en una ruta, el router remoto puede elegir enviar paquetes a través del camino con camino más largo).

6.3.2.1.5 Mecanismos para ingeniería del tráfico de comunicaciones iniciadas por nodos internos

Cuando es el nodo dentro del sitio multihomed quien inicia la comunicación, es posible definir políticas que afecten a la selección de direcciones, en particular a la de la dirección origen (que es lo que determina el proveedor a usar). Concretamente, el algoritmo de selección de direcciones origen definido en la RFC 3484 define una tabla de políticas que establece preferencias entre las direcciones a usar, como se detalla a continuación.

Tabla de políticas de la RFC 3484

La tabla de políticas definida en la RFC 3484 utiliza la regla del *longest prefix match* para realizar búsquedas. La tabla tiene tres campos, a saber: el campo *dirección*, el campo *precedencia* y el campo de *etiqueta*. Dada una dirección D, se accede a la tabla con una búsqueda que utiliza como criterio el *longest prefix match*, y retorna dos valores asociados con la dirección en cuestión, a saber la precedencia y la etiqueta.

El campo de precedencia se utiliza para elegir entre varias direcciones destino, por lo que no afectará directamente el proveedor usado (asumiendo que todos los proveedores pueden acceder a toda la Internet).

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. Por ello, a través de la configuración de los valores de la etiqueta es posible definir qué direcciones origen serán preferidas para alcanzar un destino dado, lo que implica determinar el proveedor usado para comunicarse con este destino.

La tabla configurada por defecto presentada en la especificación es la siguiente:

Prefix	Precedencia	Etiqueta
::1/128	50	0
::/0	40	1
2002::/16	30	2
::/96	20	3
::ffff:0:0/96	10	4

Ingeniería de tráfico mediante la tabla de políticas

Como se puede apreciar, es posible utilizar la tabla de políticas para hacer que un nodo prefiera una dirección origen para comunicarse con una dirección destino. En el momento en el que la dirección origen determina el proveedor usado para transportar el paquete en cuestión, es posible utilizar la tabla de políticas para determinar el proveedor usado para alcanzar un destino dado.

En su forma actual, la granularidad de la tabla de políticas permite definir políticas a nivel de dirección IP. Es posible extender esa granularidad para tener en cuenta información adicional. Por ejemplo en el trabajo [Bagnulo2001b] se ha extendido la tabla de políticas para tener en cuenta la información de puerto de destino, de forma que se pueda definir el proveedor a usar cuando se realiza una comunicación con un puerto TCP o UDP dado. Para ello basta con incluir un campo de protocolo/puerto en la tabla y tenerlo en cuenta en el momento de hacer la búsqueda.

La funcionalidad obtenida mediante este mecanismo es muy poderosa, ya que permite configurar una política de selección de proveedor con una granularidad muy fina, dado que cada nodo tiene su propia tabla de política y dentro de cada nodo es posible discriminar entre distintas aplicaciones

Distribución de políticas

Considerando que cada nodo dentro del sitio multihomed tiene su propia tabla de políticas y que para lograr los patrones de tráficos deseados es necesaria la configuración de estas tablas de políticas, es necesario contar con un mecanismo para configurar las entradas necesarias en las mismas. La especificación actual sólo define el mecanismo de configuración manual de las tablas.

Si bien eso puede ser apropiado para un número reducido de nodos, este mecanismo es claramente inapropiado para la configuración de políticas en un sitio con un número no reducido de nodos. Resulta necesario entonces 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. Es posible identificar un par de herramientas que pueden ser utilizadas para esta tarea, a saber:

- Router Advertisement (RA)
- DHCP

Un mecanismo basado en RA tiene la ventaja de que RA es generalmente usado en la redes IPv6, por lo que no sería necesario la adopción de mecanismos adicionales. En el caso de DHCP, existen sitios que no lo utilizan y configuran sus direcciones usando la autoconfiguración de direcciones sin estado, por lo que si el mecanismo se basa en el uso de DHCP, estos sitios deberían desplegar DHCP, requisito que dificultaría su adopción. Sin embargo, un mecanismo basado en RA presenta múltiples dificultades, como se detalla a continuación. En primer lugar, los mensajes RA tienen un alcance limitado al enlace en cuestión, por lo que es necesario que al menos un router por cada enlace propague la información de políticas. Esto implica que es necesario configurar al menos un router por enlace. En el caso de DHCP es posible configurar un único servidor DHCP para todo un sitio, lo que resulta en una administración simplificada y centralizada de las políticas. En segundo lugar, la distribución de la tabla de políticas debe ser atómica, es decir que la tabla debe ser distribuida y configurada en el nodo de forma completa. Si esto no es así, el nodo podría dar lugar a resultados no deseados por el administrador. En el caso de RA, esto significa que la tabla de políticas debe estar contenida dentro de un solo mensaje de RA, lo que acota a 50 el número máximo de entradas en la tabla distribuida (no es posible partir la tabla en múltiples mensajes ya que la especificación de Neighbour Discovery no permite que el procesado de un mensaje RA dependa del procesado contenido de otro mensaje RA, limitación que no existe en el caso de DHCP, ya que en el tamaño máximo de paquete permitido por la especificación DHCP es mayor y soporta fragmentación).

Dado que el enfoque de DHCP no cuenta con las relevantes desventajas identificadas para RA, creemos que es más apropiado para la distribución de tablas de políticas.

Los mecanismos para ingeniería de tráfico presentados en este capítulo pueden encontrarse documentados también en [Bagnulo2005d]

6.4 Tercera Etapa: mecanismo para preservar las comunicaciones establecidas a través de fallos

En esta sección se presenta el mecanismo para preservar las comunicaciones establecidas a través de fallos, mecanismo que se corresponde con la tercera etapa de implantación de la solución de multihoming. El mecanismo presentado debe ser adoptado tanto por los nodos dentro del sitio multihomed como por los nodos fuera del sitio multihomed, por lo que su adopción será más costosa que los mecanismos de las etapas anteriores. El mecanismo propuesto a continuación es un mecanismo extremo a extremo, es decir que sólo los nodos involucrados en la comunicación participan del mismo. La arquitectura de la solución puede encontrase también descrita en [Nordmark2005b].

La solución propuesta se basa en una nueva capa de identificación que realiza una separación entre el identificador del extremo involucrado en la comunicación y el localizador del mismo. Esta nueva capa de identificación la ubicaremos dentro de la capa de red IP, por encima de la sub-capa de reenvío[16] (forwarding) y por debajo de la sub-capa de extremo de IP[17], como se ilustra en la figura siguiente. De esta forma, las funciones realizadas en la sub-capa de extremo de IP como son fragmentación o IPSec, percibirán un identificador único representando al otro extremo de la comunicación, preservando así el correcto funcionamiento de estas funciones. Por otra parte, la sub-capa de reenvío de IP recibirá paquetes con distintos localizadores, dependiendo de la alcanzabilidad de los mismos. La nueva capa de identificación adaptará entonces los localizadores usados para alcanzar el otro extremo de la comunicación a las condiciones de alcanzabilidad de cada uno de los localizadores disponibles, y esto lo realizará de una forma transparente a todas las capas superiores, quienes sólo percibirán el identificador del otro extremo, independientemente del localizador efectivamente usado para la comunicación. De esta forma, un extremo puede utilizar distintos localizadores en función de la disponibilidad de cada uno de ellos, y preservar la comunicación presentando a las capas superiores un identificador constante a lo largo de la vida de la comunicación.

La capa de identificación establecerá correspondencias entre los identificadores y los diversos localizadores asociados a cada extremo. Los identificadores usados serán, como hemos concluido en el capítulo de Diseño de la solución, localizadores válidos que contengan

[16] La capa de reenvío de IP es la capa que determina el próximo salto al cuál debe enviarse un paquete, dada la dirección destino.

[17] La capa de extremo de IP es la capa que realiza las funciones extremo a extremo de IP, como son la fragmentación o IPSec.

información criptográfica que permita proveer la seguridad necesaria a la relación entre el identificador y los localizadores que tiene asociados. La naturaleza del espacio de nombres elegido para los identificadores se detallará en las secciones siguientes.

Adicionalmente al nuevo espacio de nombres para los identificadores, la solución contará con un plano de usuario que ejecuta las funciones relacionadas con el intercambio de paquetes de datos de usuario. Estas funciones incluyen la identificación y traducción de los paquetes que contienen localizadores que difieren de los identificadores asociados a dicha comunicación.

Finalmente, la solución cuenta con un plano de control que ejecuta las funciones relacionadas con el establecimiento de las sesiones y el intercambio de la información asociada a estas, como ser los identificadores y los conjuntos de localizadores de cada uno de los extremos involucrados. En las siguientes secciones describiremos en detalle cada uno de los componentes de la solución. La solución presentada se encuentra también descrita en [Nordmark2005b], [Bagnulo2004c], [Bagnulo2004b], [Bagnulo2005c] y [Bagnulo2005b].

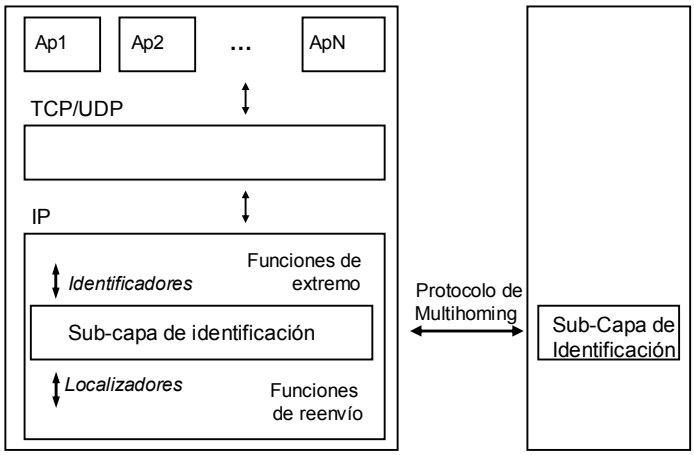

Figura 17: Arquitectura de capas – Sub-capa de identificación

6.4.1 Espacio de nombres para los identificadores

Como ya hemos visto, un aspecto fundamental de un mecanismo para preservar comunicaciones establecidas es la naturaleza de los identificadores presentados a las capas superiores a la capa de identificación. Como ya hemos expuesto en el capítulo de Diseño de la solución, en la presente Tesis Doctoral se propone el uso de identificadores que sean a la vez localizadores válidos y que además sean de naturaleza criptográfica. A estos identificadores les llamaremos genéricamente Direcciones Criptográficamente Generadas (DGCs). Las razones para esta elección, que ya han sido extensamente presentadas anteriormente, son ofrecer un nivel de seguridad suficiente frente a ataques que actualmente no pueden realizarse en el modelo de

encaminamiento actual, incluyendo en particular protección frente a los ataques desplazados en el tiempo, y el continuar siendo un localizador válido (lo que permite por un lado soportar aplicaciones que realizan referencias y por el otro preservar la seguridad existente brindada por el sistema de rutas).

A continuación presentaremos una forma de DGCs existente, conocida como las *Cryptographically Generated Addresses* (CGAs, *Direcciones Generadas Criptográficamente*). Posteriormente, analizaremos las limitaciones de este esquema y propondremos una forma alternativa de DGCs que supera las limitaciones identificadas.

6.4.1.1 Antecedentes: DGCs existentes

El uso de las DGCs ya ha sido propuesto para solucionar otros problemas en la arquitectura IP. En particular, existe una especificación de DGCs llamadas Cryptographically Generated Addresses (de ahora en adelante CGAs) [RFC3972] que define el formato y los mecanismos para la generación y verificación de CGAs para su aplicación en el nuevo protocolo de seguridad para el Descubrimiento de Vecinos Seguro [RFC3971].

Las CGAs asumen el Identificador de Interfaz de 64 bits de largo y utilizan 62 bits del mismo para contener información de las CGAs. Los 2 bits restantes son los bits “u” y “*g*” definidos por la Arquitectura de Direccionamiento de IPv6 [RFC3513].

De los 62 bits restantes, 3 bits son utilizados para contener el parámetro *Sec* que representa el nivel de defensa contra ataques de fuerza bruta que posee la dirección en cuestión. Los restantes 59 bits son el hash de los parámetros de la DGC que se presentarán a continuación.

De forma que las CGAs están definidas como direcciones que en sus 64 bits menos significativos contienen:

- 2 bits definidos por la arquitectura de direccionamiento de IPv6
- 3 bits codificando el Sec
- 59 bits contienen el resultado de lo que a continuación definiremos como Hash1

6.4.1.1.1 Parámetros de las CGAs y valores de Hash

Se define una Estructura de Datos de CGA (Figura18) que contiene los siguientes parámetros de la CGAs definidos de la siguiente manera:

- Modificador: un valor aleatorio de 128 bits usado para la generación de la CGA
- Prefijo de Red: el prefijo de la dirección en cuestión.
- Contador de colisiones: un valor que indica el número de colisiones encontradas en el momento de la generación de la dirección

- Clave Pública: la clave pública asociada a la CGA
- Extensiones: otra información a ser incluida en el Hash, en forma de extensión.

Adicionalmente se definen los siguientes valores de Hash:

- Hash1: son los 64 bits más significativos del Hash SHA-1 de la estructura de parámetros de la CGA definida anteriormente.
- Hash2: son los 112 bits más significativos del Hash SHA-1 de la estructura de parámetros de la CGA, solo que los campos de prefijo de subred y contador de colisiones deberán estar a cero.

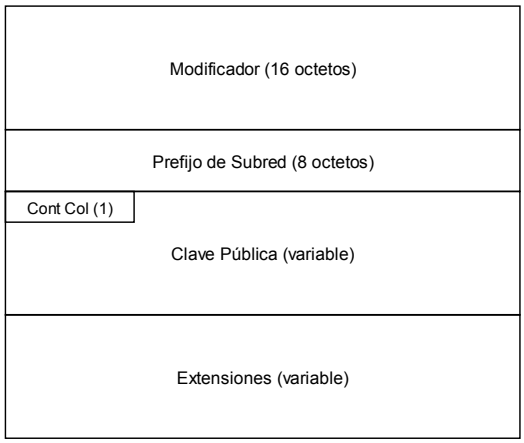

Figura 18: Estructura de Datos de CGA

6.4.1.1.2 Procedimiento de generación de las CGAs

El procedimiento de generación de una CGA se detalla a continuación:

Las entradas al procedimiento son las siguientes:

- La clave pública asignada a la CGA
- El prefijo de subred a la que pertenece la CGA
- El parámetro de seguridad Sec que se ha fijado en función de las necesidades de protección requeridas

El procedimiento cuenta con los siguientes pasos:

1- Se genera aleatoriamente un valor de 128 bits para el modificador.

2- Se genera la estructura de parámetros necesaria para generar Hash2, es decir la concatenación del modificador, el prefijo de red (puesto a cero para el cálculo del

Hash2), el contador de colisiones (también a cero para Hash2) y la clave pública. El hash SHA-1 se genera a partir de esta estructura de datos y los 112 bits más significativos son Hash2.

3- Se comparan los 16*Sec bits más significativos de Hash2 con cero. Si son todos cero (o si Sec=0) se pasa el paso 4. Si no se incrementa el modificador y se vuelve al paso 2.
4- El contador de colisiones se pone a cero.
5- Se obtiene la estructura de parámetros para la generación de Hash1, es decir se concatena el modificador vigente, con el contador de colisiones vigente, el prefijo de subred y la clave pública en cuestión. Se genera el hash SHA-1 de esta estructura de datos. Los 64 bits más significativos son Hash1.
6- Se genera el identificador de interfaz de la CGA, sustituyendo los tres primeros bits de Hash1 por el valor de Sec y poniendo a cero los bits "u" y "g".
7- Se genera la CGA concatenando el prefijo con el Identificador de Interfaz obtenido.
8- Se realiza la detección de duplicados [RFC2461]. Si existe una colisión, se aumenta el contador de colisión y se vuelve al paso 5. Si hay más de tres colisiones, se aborta.

Como puede verse, es posible que sea necesario iterar varias veces para obtener un Hash2 correcto cuando Sec>0. Estudiaremos este punto más en detalle a continuación cuando analicemos las propiedades de seguridad de esta solución.

6.4.1.1.3 Procedimiento de verificación de las CGAs

El procedimiento de verificación de una CGA toma como entradas una dirección IPv6 y una estructura de parámetros de CGA y realiza los pasos detallados a continuación:

1. Verificar que el contador de colisiones sea 0, 1 o 2. Si es un valor distinto, la verificación falla.
2. Verifica que el prefijo de red incluido en la estructura de datos de la CGA sea igual al incluido en la CGA. Si es distinto, la verificación falla.
3. Se ejecuta el Hash SHA-1 sobre la estructura de datos de la CGA. Los 64 bits más significativos son Hash1.
4. Se compara Hash1 con el Identificador de Interfaz de la CGA sin tener en consideración los bits de Sec y los bits "u" y "g". Si los valores difieren, la verificación falla.
5. Obtener el valor de Sec de la CGA.

6. Generar el Hash2 tomando los 112 bits más significativos del Hash SHA-1 de la cadena de bits formada por 9 octetos a cero, la clave pública y los campos de extensión.
7. Comparar los 16*Sec bits más significativos de Hash2 con cero. Si algún bit difiere de cero, la verificación falla. En caso contrario la verificación es exitosa.

6.4.1.1.4 Firmas y verificación de firmas usando CGA

A continuación detallaremos cómo se firma utilizando la información criptográfica de las CGA. Para firmar un mensaje utilizando CGAs, un nodo necesita: la CGA en cuestión, la estructura de parámetros asociada, el mensaje a firmar y la clave privada de la CGA.

La firma del mensaje se realiza cifrando con RSA [RFC3447] el mensaje en cuestión utilizando la clave privada asociada a la CGA.

La firma resultante, así como los parámetros de la CGA y el mensaje son enviados al receptor. Para la verificación, el receptor realiza los pasos siguientes:

1. Verifica la CGA como se detallado anteriormente.
2. Verifica la firma RSA utilizando el algoritmo RSA [RFC3447] y SHA-1. Las entradas para la verificación de esta firma son: la clave pública, el mensaje y la firma recibida.

6.4.1.1.5 Análisis de seguridad

6.4.1.1.5.1 Ataques posibles

El punto más vulnerable de las CGAs es la longitud del Hash contenido en el Identificador de Interfaz. Esto se debe a que este es el único parámetro que tiene un largo fijo, por lo que no es posible mejorar la seguridad añadiendo más bits. La firma de los mensajes se realiza utilizando la clave privada completa por lo que es posible mejorar la seguridad de la firma utilizando claves más largas. Por ello, la mayor vulnerabilidad que presentan las CGAs son las colisiones de Hash, es decir que un atacante pueda encontrar una estructura de datos de CGAs con claves pública y privada generadas por el atacante, cuyo hash coincida con el de una víctima. Esto permite al atacante iniciar una comunicación pretendiendo ser el dueño de la CGAs de la victima, lo que puede aparejar ataques de secuestro de la identidad y negación de servicio.

6.4.1.1.5.2 Dificultad de los ataques:

Primero consideraremos el caso en que Sec=0 y luego extenderemos el análisis para Sec>0.

Como hemos visto, un atacante deberá encontrar una estructura de datos cuyo hash coincida con el identificador de interfaz de una victima escogida. Para ello, el atacante puede variar el valor del modificador de entrada, de modo que se obtengan distintos valores de salida hasta obtener uno que coincida con el de la víctima.

El número esperado de intentos necesario para encontrar una pre-imagen de un valor dado en una función de hash ideal de n bits es 2^n [Menezes1997a]. Esto implica que el número de intentos necesario para poder encontrar un valor de modificador que resulte en el identificador de interfaz deseado por el atacante es igual a 2^{59}, ya que el identificador de interfaz contiene 59 bits de información de hash.

A continuación, estimaremos el tiempo necesario para realizar este ataque:

El largo esperado de la estructura de datos de la CGA que debe usar el atacante será:

- 16 octetos de modificador
- 8 octetos de prefijo
- 128 octetos (1024 bits) de clave pública RSA

Resulta en un total de 152 octetos por intento.

Según la herramienta `openssl speed`[18], un ordenador con un procesador Pentium 4 a 2.66 Ghz y 440 MB de RAM, puede hacer operaciones de hash a 66631 kB por segundo, con bloques de 152 octetos como entrada.

Despreciando el tiempo de comparación y el de sumar uno al modificador, resulta que el tiempo necesario para encontrar un modificador que corresponda con el hash de la victima es:

$T= (2^{59}) * 152B /(66.631.000B/s) = 42\ 000$ años (aprox)

Si bien este mecanismo parece ofrecer una protección suficiente para la tecnología actual, es posible que a medida que la velocidad de cálculo mejore, el tiempo necesario para realizar este ataque disminuya considerablemente. Para proteger de estos avances, la especificación utiliza el parámetro Sec.

En caso que Sec sea superior a 0, no solamente será necesario encontrar un modificador con el cual el Hash resultante coincida con el identificador de interfaz de la CGA de la victima, sino que además será necesario que los Sec*16 bits del Hash2 sean iguales a cero. Esto impone búsquedas adicionales en el momento de la generación de la CGA, tanto para el atacante como para el propietario de la CGA.

[18] `openssl speed` es un commando del OpenSSL Project, www.openssl.org.

La influencia del parámetro Sec puede ser analizada como si la salida de la función de hash se extendiera en Sec*16 bits. Esto implica, que el número de bits considerados de la salida de la función de hash es igual a 59 + Sec*16

Esto implica que el número de intentos para encontrar una pre-imagen de una imagen dada es de $2^{(59+Sec*16)}$

El costo de esta mejora en la seguridad es un mayor esfuerzo para poder generar una CGA, siendo la esperanza del número de intentos necesarios para generar una CGA igual a $2^{(16*Sec)}$

6.4.1.2 Nuevas DGCs propuestas: Hash Based Addresses (HBAs)

6.4.1.2.1 Limitaciones de las CGAs

Como hemos visto, las CGAs son una poderosa herramienta que permite ofrecer la seguridad necesaria de un protocolo para preservar las comunicaciones establecidas a través de fallos en sitios multihomed. Sin embargo, el hecho de que la operación de las CGAs esté basada en el uso extensivo de la criptografía de clave pública impone un considerable coste de operación en el protocolo. Esto se debe a que para establecer una sesión de multihoming entre dos nodos, es necesario realizar al menos una verificación de una firma basada en criptografía RSA. A medida que el nodo del sitio multihomed mantiene comunicaciones con más nodos remotos, este coste crece proporcionalmente. Esto nos lleva a buscar alternativas más económicas que ofrezcan un nivel equivalente de seguridad. A continuación proponemos un nuevo tipo de direcciones generadas criptográficamente, llamadas *Hash Based Addresses* (HBAs, *Direcciones Basadas en Hash*). Veremos que las HBAs proveen el mismo nivel de seguridad a un coste reducido. Una descripción de las HBAs puede encontrarse también en [Bagnulo2004a] y [Bagnulo2005c].

6.4.1.2.2 Conceptos fundamentales de las Hash Based Addresses

El objetivo fundamental de las HBAs es proteger la relación entre el grupo de direcciones disponibles en un nodo ubicado en sitio multihomed al cual se le han delegado múltiples prefijos agregables por proveedor. Es decir, que la información criptográfica contenida en una dirección HBA deberá permitir verificar el conjunto de direcciones alternativas asociadas a la dirección en cuestión.

Para ello, el esquema HBA propone generar el identificador de interfaz de las todas las direcciones asociadas a un mismo nodo del sitio multihomed como un hash del conjunto de prefijos disponibles para dicho nodo (y otros parámetros). De esta forma, la información de los múltiples prefijos disponibles (es decir, la información referente a las diferentes direcciones disponibles) estará codificada dentro del identificador de interfaz de cada una de las direcciones asignadas al nodo, permitiendo así su verificación posterior.

6.4.1.2.3 Relación entre las HBAs y las CGAs

Como podemos observar, tanto las CGAs como las HBAs utilizan los bits del identificador de interfaz para almacenar información criptográfica. Esto implica que si no se diseña de forma adecuada, puede resultar imposible utilizar las HBAs y las CGAs de forma simultánea. Existen al menos dos razones que hacen que el soporte simultáneo de ambas sea necesario:

1. Las CGAs son necesarias para la provisión de SEcure Neighbour Discovery [RFC3971]. Si las HBAs son incompatibles con las CGAs, será imposible hacer uso simultáneo de los protocolos que se basen en CGAs y los protocolos que se basen en HBAs. Por ejemplo, si el protocolo de multihoming se basa en el uso de HBAs, será imposible usarlo simultáneamente con Secure Neighbour Discovery, lo que es claramente indeseable.
2. Las CGAs son, como veremos más adelante, más costosas en su operación que las HBAs, pero proveen de ciertas funcionalidades que no son soportadas por las HBAs. Esto hace que en ciertas ocasiones sea deseable que un nodo posea direcciones que son simultáneamente HBA y CGA, de modo que pueda utilizar las funcionalidades de la HBA de reducido coste cuando estas proveen el soporte necesario, y utilizar las funcionalidades más costosas de las CGAs cuando éstas son necesarias.

Por las razones presentadas, el objetivo es diseñar las HBAs de forma que sean compatibles con la especificación de las CGAs. Esto quiere decir que es necesario generar direcciones que sean simultáneamente CGA y HBA, donde el identificador de interfaz contenga información criptográfica de la CGA y del conjunto de prefijos disponibles en el nodo. La forma más natural de hacerlo es definir una nueva extensión para las CGAs que contenga la información de prefijos disponible en el nodo multihomed.

A pesar de que el enfoque elegido para la definición de las HBAs sea una extensión de las CGAs, cabe notar que las CGAs y las HBAs son conceptos diferentes. En particular, las CGA están inherentemente unidas a una clave pública, mientras que las HBAs están inherentemente unidas a un conjunto de prefijos. Esto quiere decir que para la generación de las HBAs, no es necesario contar con una clave pública, al igual que para generar una CGA no es imprescindible contar con múltiples prefijos. Por ello, no parece razonable exigir una clave pública para la generación de las HBAs, cuando las funcionalidades de las CGAs no son requeridas. Parece entonces razonable proponer la existencia de tres tipos de direcciones:

- Direcciones CGA: Direcciones generadas como define la especificación de CGA [RFC3872] sin incluir la nueva extensión de múltiples prefijos. Estas direcciones está unidas a una clave pública y a un único prefijo. Estas direcciones pueden ser usadas para ejecutar el mecanismo de Secure Neighbour Discovery y para soporte de multihoming si éste es basado en el uso de clave pública.

- Direcciones híbridas HBA/CGA: Estas direcciones son direcciones CGA cuya estructura de datos CGA contiene la extensión de múltiples prefijos. Estas direcciones están unidas a una clave pública y a un conjunto de prefijos y soportan las funciones de las CGAs y las de las HBAs simultáneamente. Estas direcciones pueden ser utilizadas para Secure Neighbour Discovery, y para multihoming tanto basado en CGA como basado en HBA.
- Direcciones HBA: Estas direcciones están unidas solamente a un conjunto de prefijos y no están unidas a una clave pública. Con el objetivo de preservar la compatibilidad con CGAs, se utilizará la estructura de parámetros de CGA (con la extensión de múltiples prefijos) para su generación, pero un número aleatorio se incluirá en el campo de clave pública. Estas direcciones pueden ser utilizadas para los protocolos basados en HBAs, pero no para aquellos que requieran funcionalidades de las CGAs.

6.4.1.2.4 Extensión de Múltiples Prefijos para las CGAs

El formato de la Extensión de Múltiples Prefijos se define a continuación:

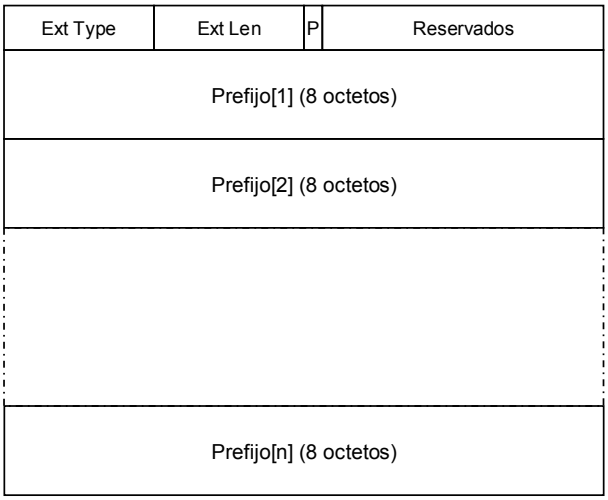

Figura 19: Extensión de Múltiples Prefijos

Siendo:

Ext Type: un identificador de 8 bits de la Extensión de Múltiples Prefijos (se sugiere el valor 0x12)

Ext Len: 8 bits que expresan el largo de la Extensión en octetos

Indicador P: Bit que está a uno si el campo de clave pública de la Estructura de Parámetros de la CGA contiene una clave pública y está a cero si el campo contiene un número aleatorio.

Reserved: 15 bits reservados para uso futuro. Inicializados a cero.

Prefix[1…n]: Vector de prefijos de 64 bits, numerados de 1 a n.

6.4.1.2.5 Generación de los conjuntos de HBAs

Con el objetivo de preservar la compatibilidad con las CGAs, el proceso de generación de la HBAs debe basarse en el proceso de generación de generación de las CGAs presentado anteriormente. Como hemos visto, el proceso de generación de las CGA tiene tres entradas: un prefijo de subred de 64 bits, una clave pública y un parámetro de seguridad Sec.

La principal diferencia entre el proceso de generación de las CGAs y de las HBAs es que todas las HBAs asociadas a un conjunto de prefijos deben ser generadas usando el mismo conjunto de parámetros, por lo que su generación debe realizarse de forma coordinada, sino conjunta. A continuación describiremos un mecanismo para generar todas las direcciones de un conjunto de HBAs asociadas a un conjunto de prefijos. El proceso de generación de cada una de ellas es muy similar al proceso de generación de una CGA, sólo que se incluye la Extensión de Múltiples Prefijos y los procesos de generación de las múltiples direcciones utilizan los mismos parámetros.

El proceso de generación de un conjunto de HBAs asociadas a un conjunto de prefijos se describe a continuación.

Las entradas al proceso de generación de un conjunto de HBAs son:

- Un vector con n prefijos de 64 bits
- Un valor del parámetro Sec
- En el caso de la generación de direcciones híbridas HBA/CGA, se requiere también una clave pública. (Esta entrada no es necesaria para generar direcciones HBA)

La salida del proceso de generación es:

1. Un conjunto de direcciones HBA
2. Sus respectivas estructuras de parámetros CGA

Los pasos del proceso de generación son los siguientes:

1. Generación de la Extensión de Múltiples Prefijos: Generar la Extensión de Múltiples Prefijos con el formato definido en la sección 6.4.1.2.4. Incluir el vector de n prefijos de 64 bits de

entrada en los campos Prefix[1…n]. El valor del campo Ext Len es n*8. Si una clave pública ha sido provista como entrada, el indicador P es puesto a 1; si no éste es puesto a 0.

2. Generación del Modificador. Generar el Modificador como un número aleatorio de 128 bits. Si no se ha provisto una clave pública como entrada, entonces generar un Modificador Extendido como un número aleatorio de 384 bits. Codificar el Modificador Extendido como una clave RSA en una estructura ASN.1 de tipo `SubjectPublicKeyInfo` codificada en DER, como es definida en los perfiles de certificados X.509 [RFC3280].
3. Concatenar de izquierda a derecha, el Modificador, 9 octetos a cero, la clave pública o el Modificador Extendido codificado (dependiendo si se ha provisto o no clave pública como entrada) y la Extensión de Múltiples Prefijos. Realizar el hash SHA-1 sobre la cadena resultante de la concatenación y extraer los 112 bits más significativos. El resultado es Hash2.
4. Verificar si los 16*Sec bits más significativos son 0. Si la verificación es exitosa, continuar en el paso (5). Si la verificación falla, incrementar el Modificador en 1 y volver al paso (3).
5. Poner el Contador de Colisiones a 0.
6. Para i=1 a n, realizar:
   6.1. Concatenar de izquierda a derecha, el Modificador resultante, el Prefijo[i], un octeto con el contador de colisión, la clave pública o el Modificador Extendido codificado (dependiendo si se ha provisto o no clave pública como entrada) y la Extensión de Múltiples Prefijos. Realizar el hash SHA-1 sobre el cadena resultante de la concatenación y extraer los 64 bits más significativos. El resultado es Hash1[i].
   6.2. Crear el identificador de interfaz iid[i] a partir de Hash1[i] sobre-escribiendo el valor del parámetro Sec en los tres bits más significativos y poniendo los bits “u” y “g” a 0 como define [RFC2460].
   6.3. Generar la dirección HBA[i] concatenando de izquierda a derecha Prefix[i] con iid[i].
   6.4. Realizar la detección de direcciones duplicadas. Si una colisión es detectada, incrementar el contador de colisión y volver al paso (6). Después de tres colisiones, abortar y reportar el error.
   6.5. Crear la Estructura de Parámetros de CGA correspondiente a HBA[i] concatenando de izquierda a derecha, el Modificador resultante, el Prefijo[i], un octeto con el contador de colisión usado, la clave pública o el Modificador Extendido codificado (dependiendo si se ha provisto o no clave pública como entrada) y la Extensión de Múltiples Prefijos.

6.4.1.2.6 Procedimiento de verificación de la HBA

Como las HBAs han sido definidas como una extensión de las CGAs, una HBA y su correspondiente Estructura de Parámetros de CGA (incluyendo la Extensión de Múltiples Prefijos) podrán efectuar exitosamente el procedimiento de verificación de CGA definido en

[RFC3972] y descrito en la sección 6.4.1.1.3. Esta verificación permite determinar si una cierta dirección HBA dada se corresponde a una estructura de parámetros de CGA propuesta.

Sin embargo, para los usos potenciales de las HBAs, también es relevante determinar si una cierta HBA pertenece a un conjunto de HBAs dado. Un conjunto de HBAs queda definido a través de una estructura de datos de CGA que contenga una Extensión de Múltiple Prefijos. Esto es así porque a partir de ésta, es posible generar el conjunto de direcciones del conjunto HBA solamente variando el prefijo incluido en el campo de Prefijo de Subred de la estructura de parámetros de CGA usada para la generación.

A continuación definiremos el procedimiento para verificar si una HBA dada pertenece al conjunto de HBAs asociado a una estructura de datos de CGA.

Las entradas al proceso son:

- Una dirección HBA
- Una estructura de datos de CGA (que contiene una extensión de Múltiples Prefijos)

Los pasos para realizar la verificación son:

1. Verificar que el prefijo de 64 bits incluido en la dirección HBA se encuentra en el conjunto de prefijos de la extensión de Múltiples Prefijos de la estructura de datos de CGA. Si el prefijo no se encuentra, el proceso de verificación falla. Si el prefijo está incluido, reemplazar el prefijo contenido en el campo de Prefijo de Subred de la estructura de datos de la CGA por el prefijo de 64 bits de la HBA.
2. Ejecutar el proceso de verificación de la CGA como está descrito en la sección 6.4.1.1.3.
   2.1. Verificar que el contador de colisiones sea 0,1 o 2. Si es un valor distinto, la verificación falla.
   2.2. Verificar que el prefijo de red incluido en la estructura de datos de la CGA es igual al incluido en la CGA. Si es distinto, la verificación falla.
   2.3. Se ejecuta el Hash SHA-1 sobre la estructura de datos de la CGA. Los 64 bits más significativos son Hash1.
   2.4. Se compara Hash1 con el Identificador de Interfaz de la CGA sin tener en consideración los bits de Sec y los bits “u” y “g”. Si los valores difieren, la verificación falla.
   2.5. Obtener el valor de Sec de la CGA.
   2.6. Generar el Hash2 tomando los 112 bits más significativos del Hash SHA-1 de la cadena de bits formada por el Modificador, 9 octetos a cero, la clave pública y los campos de extensión.

2.7. Comparar los 16*Sec bits más significativos de Hash2 con cero. Si alguno difiere de cero, la verificación falla. En caso contrario la verificación es exitosa.

6.4.1.2.7 Análisis de Seguridad

Como hemos mencionado anteriormente, el objetivo de las HBAs es crear conjuntos de direcciones que se encuentren inherentemente unidos de forma segura, de forma que puedan ser usados indistintamente para comunicarse con el nodo multihomed a quien corresponden. El principal ataque que previenen las HBAs son los anteriormente mencionados ataques de redirección, en los que los paquetes dirigidos a una dirección son redirigidos hacia otra dirección seleccionada por el atacante, secuestrando así la comunicación. El uso de las HBAs previene dichos ataques ya que solamente las direcciones incluidas en el conjunto de HBAs pueden ser usadas para la comunicación, limitando así el conjunto de direcciones a un conjunto predeterminado que pertenece al nodo original. De esta forma, una comunicación establecida con la dirección A podrá ser redirigida a través del protocolo de multihoming a una dirección B si y sólo si la dirección B pertenece al mismo conjunto HBA de la dirección A.

En caso que un atacante desee redirigir una comunicación establecida con una dirección HBA1 a una dirección IPX, el atacante deberá crear una estructura de parámetros de CGA que resulte en un conjunto de HBA que contenga a ambas direcciones HBA1 y IPX.

Dicha estructura de datos debe cumplir las siguientes condiciones:

- El prefijo de HBA1 está contenido en el conjunto de prefijos incluidos en la Extensión de Múltiples Prefijos
- El prefijo de IPX está contenido en el conjunto de prefijos incluidos en la Extensión de Múltiples Prefijos
- HBA1 está contenida en el conjunto de HBAs resultante.

Estas condiciones asumen que es suficiente para el atacante redirigir la comunicación a cualquiera de las direcciones del prefijo de la dirección IPX, i.e. que el atacante tiene acceso no sólo a una dirección del prefijo en cuestión, sino a todo el prefijo.

Los demás campos de la estructura de datos de la CGA que pueden ser cambiados a voluntad por el atacante son: el Modificador, los otros prefijos, la clave pública y otras extensiones.

Para encontrar los parámetros de la estructura de datos de CGA que produzcan el conjunto de HBA deseado, el atacante deberá realizar un ataque de fuerza bruta, variando los parámetros que le son permitidos. Análogamente al caso de las CGAs, el número esperado de intentos que deberá realizar depende del número de bits de hash incluidos en el identificador de interfaz y del parámetro Sec y tiene un valor de $2^{(59+16*Sec)}$

6.4.1.2.8 Comparación HBAs y CGAs

Hemos argumentado que la principal limitación presentada por las CGAs era el elevado coste de computación que suponía su uso masivo. Hemos presentado a las HBAs como una alternativa más económica. Una comparación del coste computacional que supone el enfoque de basado en los principios de las HBAs con el coste asociado al uso de una solución basada en los principios de las CGAs puede encontrarse en [Bagnulo2005c]. Dicho estudio, compara los costes asociados a una técnica basada en los mismos principios de las HBAs, pero que utiliza una estructura de datos diferente, con los costes asociados al uso de una técnica basada en los principios de las CGAs (pero con una estructura de datos distinta). El estudio considera dos funciones críticas para la operación de un protocolo de multihoming, a saber: la creación de direcciones y el establecimiento de la sesión de multihoming, donde los localizadores alternativos disponibles para una comunicación son intercambiados y validados. Las conclusiones son que:

- Para la creación de direcciones, el tiempo de procesamiento requerido por la técnica basada en los principios de las HBAs es cuatro órdenes de magnitud menor al tiempo requerido por la técnica basada en los principios de las CGAs
- Para el establecimiento de sesión, el tiempo de procesamiento requerido por la técnica basada en los principios de las HBAs es dos órdenes de magnitud menor al tiempo requerido por la técnica basada en los principios de las CGAs

Además del coste computacional, existen otras diferencias relevantes entre las CGAs y las HBAs. En particular, es importante notar que por su propia naturaleza, el conjunto de direcciones contenidas en un conjunto de HBAs queda determinado en el momento de su creación y no puede ser modificado a posteriori. Esto implica que no es posible agregar nuevas direcciones al conjunto una vez creado. De forma que si hay un cambio en el conjunto de prefijos disponibles en el nodo, será necesario crear un nuevo conjunto de HBAs que contenga todos los prefijos disponibles. Esta limitación no parece muy importante en el caso de un sitio multihomed, ya que el conjunto de prefijos disponibles en un sitio no es muy dinámico, y sólo cambia cuando hay un evento de renumerado, cuya frecuencia es baja. Sin embrago, cuando consideramos escenarios móviles, donde el nodo cambia su punto de conexión frecuentemente, la dificultad de soportar conjuntos fuertemente dinámicos de direcciones en un nodo es un problema.

En conclusión, las HBAs son menos costosas pero no soportan conjuntos dinámicos de direcciones, mientras que las CGAs son más costosas pero soportan conjuntos dinámicos de direcciones. Sin embargo, cabe notar que gracias al diseño realizado, las HBAs son compatibles con las CGas y es posible generar direcciones que sean híbridas HBA/CGA. Estas direcciones permiten utilizar las técnicas de HBA que son menos costosas mientras el conjunto de direcciones sea estable, y utilizar las funciones de CGA cuando el conjunto de direcciones sea dinámico, obteniendo lo mejor de los dos mundos.

6.4.2 El plano de datos

El plano de datos de la capa de identificación realizará las funciones de traducción entre los identificadores y los localizadores usados en el intercambio de paquetes de datos. Para dicha tarea, es necesaria la capacidad de identificar los paquetes que deben ser sometidos al proceso de traducción. Esto es debido a que no todos los paquetes requieren procesamiento por parte de la capa de identificación. En particular, los paquetes de aquellas comunicaciones para las que no se haya creado una sesión de multihoming no deben ser procesados por la capa de identificación. Por ejemplo, considere aquellas comunicaciones establecidas con nodos que no implementan el protocolo de multihoming o aquellas comunicaciones para las que por su escasa duración no compensa el coste asociado al establecimiento de la sesión de multihoming. Es más, no todos los paquetes intercambiados en una comunicación para la que se ha establecido una sesión de multihoming requieren que sus direcciones sean traducidas por la capa de multihoming. En particular, aquellos paquetes que contengan como localizadores los identificadores usados para establecer la comunicación no requieren traducción alguna.

El comportamiento de la capa de identificación de multihoming es el siguiente: Cuando se establece una comunicación, la aplicación que la inicia selecciona una de las direcciones destino disponibles. Es posible que también seleccione la dirección origen; si esto no es así, la capa IP seleccionará una de las direcciones posibles para esa comunicación. Si esas direcciones seleccionadas inicialmente se encuentran operativas, i.e. los paquetes que contienen dichas direcciones son cursados satisfactoriamente por la red, esas direcciones serán utilizadas tanto como identificadores como localizadores para el intercambio de los paquetes asociados a la comunicación. En este caso, no es necesario que la capa de identificación realice traducción alguna, como hemos mencionado anteriormente.

Sin embargo, en caso que, posteriormente al establecimiento de la sesión de multihoming a través del protocolo de multihoming definido en la sección siguiente, se produzca un fallo en el camino usado para intercambiar los paquetes de la comunicación, será necesario utilizar localizadores alternativos para preservar la comunicación establecida. En este caso, los localizadores utilizados para el intercambio de paquetes diferirán de los identificadores asignados a la comunicación. Para poder realizar correctamente esta traducción, la capa de identificación debe poder identificar los paquetes cuyas direcciones deben ser traducidas. Esto se debe a que todas las direcciones disponibles en un nodo pueden ser usadas tanto como localizadores como identificadores, por lo que es posible, y probable que una misma dirección sea utilizada simultáneamente como identificador y localizador. Esto implica que las direcciones contenidas en un paquete no brindan información suficiente para determinar si el paquete debe ser sometido al proceso de traducción. Para solventar esta dificultad, son necesarias consideraciones adicionales que permitan la correcta identificación de los paquetes que requieren procesamiento. Existen

diversas alternativas para solventar la situación, que se presentarán a continuación. Los enfoques posibles pueden clasificarse en dos categorías: enfoques que evitan las ambigüedades y enfoques que utilizan una etiqueta de contexto.

6.4.2.1 Enfoques que evitan las ambigüedades

6.4.2.1.1 Identificador predeterminado

La opción más simple en esta categoría es predeterminar una de las direcciones disponibles como el identificador a utilizar en todas las comunicaciones, mientras que todas las otras direcciones serán utilizadas solamente como localizadores. Esto implica que las capas superiores sólo percibirán esta dirección seleccionada como identificador en todas las comunicaciones del nodo y el resto de las direcciones permanecerán invisibles a ellos. La traducción en este caso es trivial ya que todos lo paquetes que no posean la dirección usada como identificador deben ser traducidos. Sin embargo, éste enfoque requiere que sólo la dirección usada como identificador sea publicada en el DNS, ya que ésta es la única dirección que puede ser devuelta a las aplicaciones para ser usada como identificador. Esta configuración resulta en una reducción de las capacidades de tolerancia a fallos de la solución, ya que sólo una de las direcciones estará disponible en el DNS para realizar el contacto inicial. Una posible solución a esta limitación pasa por definir un nuevo registro de DNS para publicar la información referente a cuál es el identificador elegido. El problema de este enfoque es que requiere soporte de ambos extremos y que no funciona con nodos que no implanten el protocolo de multihoming, lo que la hace inaceptable en un entorno real en donde es imposible considerar una transición inmediata al protocolo de multihoming.

6.4.2.1.2 N cuadrado direcciones

Para solucionar las limitaciones del enfoque anterior es posible crear direcciones adicionales que tengan un rol predeterminado. En este enfoque, cada nodo con n prefijos disponibles creará n^2 direcciones, formando n conjuntos de n direcciones. Cada conjunto contiene una dirección de cada prefijo. Dentro de cada conjunto, se le asigna a una de las direcciones el rol de identificador y a las demás direcciones el rol de localizador. El resultado es que existirán n direcciones a las que se le asigna el rol de identificador y que poseen un prefijo distinto en cada uno de los n conjuntos. Las direcciones a las que se les asigna el rol de identificador se publican en el DNS mientras que las direcciones a las que se les asigna el rol de localizador no se publican en el DNS para evitar que sean usada como identificadores por las capas superiores. La configuración resultante provee capacidades completas de tolerancia a fallos, ya que direcciones con todos los prefijos disponibles pueden ser usadas para el contacto inicial.

El funcionamiento según este enfoque es el siguiente: Si un paquete contiene direcciones a las que se les ha asignado el rol de identificador, entonces no es necesario realizar la operación de traducción. Sin embargo, si el paquete contiene direcciones a las que se les ha asignado el rol de localizador, entonces se traducen a las direcciones-identificadores del conjunto al que pertenecen.

6.4.2.2 Enfoques basados en una etiqueta de contexto

Los enfoques de esta categoría se basan en incluir información adicional, una etiqueta de contexto, en los paquetes que necesitan ser traducidos por la capa de identificación. A continuación se analizan diversas opciones donde transportar dicha información. Sólo son necesarias tantas direcciones como prefijos tenga el nodo multihomed, y cualquiera de estos prefijos puede utilizarse como identificador y localizador.

6.4.2.2.1 Flow Label

Una opción es llevar la etiqueta de contexto en el campo de Flow Label del encabezado IPv6. La principal dificultad que presenta este enfoque es que el campo de Flow Label ha sido diseñado para otros usos, en particular para poder identificar flujos de datos y así poder ofrecerles un tratamiento diferenciado cuando transitan por la red. Cabe notar que los bits del Flow Label son preciosos, ya que son de los escasos bits que pertenecen al encabezado principal de IPv6 y que son inspeccionados por todos los nodos del camino transitado por el paquete. Las funciones de la capa de identificación son realizadas extremo a extremo, por lo que no parece necesario usar bits del encabezado principal. Por ende, si bien es posible utilizar los bits del Flow Label, es necesario que su uso no interfiera con posibles usos futuros que se le quieran dar a dicho campo. En particular, es razonable exigir que todos los paquetes intercambiados entre dos direcciones en una comunicación contengan el mismo valor de Flow Label.

Además, es importante notar que el Flow Label a utilizar en los paquetes intercambiados en una comunicación entre dos direcciones IP es determinado por el iniciador de la comunicación, cuando envía el primer paquete. Cabe notar que el protocolo de control para el establecimiento de las sesiones de multihoming es un protocolo asíncrono con respecto a los datos, y que no necesariamente se ejecutará al comienzo de la comunicación. Esto es así porque el establecimiento de la sesión de multihoming implica un coste, y parece razonable permitir que políticas internas del nodo determinen qué comunicaciones requieren el soporte del protocolo de multihoming y a partir de cuándo se provee dicho soporte. Por ende, no es posible asumir que existe una señalización previa al establecimiento de la comunicación.

Los dos puntos anteriores nos llevan a concluir que el valor de Flow Label usado en los paquetes de una comunicación entre dos direcciones IP quedará determinado por el iniciador de la comunicación, sin contar con ningún tipo de señalización por parte del protocolo de multihoming.

Es interesante ver que quien asigna el valor del Flow Label (iniciador/transmisor) no es quien lo utilizará para identificar los paquetes a ser traducidos (receptor).

Dado que el Flow Label es asignado antes de ningún tipo de señalización de multihoming, el nodo iniciador que asigna el Flow Label no puede saber el conjunto de direcciones disponibles para el nodo contraparte, ni si tiene otras comunicaciones establecidas con éste nodo. Esto es un problema ya que si se asigna el mismo valor de Flow Label a dos sesiones con el mismo nodo contraparte, pero que tengan asociadas distintos identificadores, es posible que se den situaciones donde la información contenida en el paquete no sea suficiente para identificar la sesión a la que pertenece.

Por ejemplo, consideremos la situación siguiente: tenemos dos nodos A y B con múltiples direcciones cada uno, IPA1,…, IPAn, y IPB1,…, IPBm respectivamente. Ahora bien, el nodo A establece dos comunicaciones: una primera comunicación entre su dirección IPA1 y la dirección IPB1 y una segunda comunicación entre IPA2 e IPB2. Si bien estas dos comunicaciones han sido establecidas entre los mismo nodos, ni A ni B lo saben, ya que no han ejecutado aún ningún protocolo de señalización que les permita conocer el conjunto de localizadores asociados a los identificadores que han usado para establecer la comunicación. Sin embargo, los valores de Flow Label asociados a cada una de las comunicaciones ya han sido determinado en el instante previo al inicio de las comunicaciones. Ahora bien, si se ha elegido el mismo valor de Flow Label para ambas comunicaciones, es posible que se creen los problemas que se describen a continuación. Supongamos que se ha utilizado el mismo valor de Flow Label (esta es una situación razonable, ya que ninguno de los nodos sabe que ambas comunicaciones están establecidas entre las mismas partes). Después de transcurrido un tiempo, ambas partes deciden establecer una sesión de multihoming para proteger la primera de las comunicaciones. Durante el establecimiento de la sesión, se intercambiarán los conjuntos de localizadores alternativos para la sesión. El resultado será una sesión con la siguiente información

Sesión asociada a la primera comunicación:

Identificadores: IPA1 y IPB1

Localizadores alternativos (IPA2,…, IPAn) y (IPB2,…,IPBm)

Flow Label: el mismo que para la segunda comunicación.

Ahora bien, supongamos que se produce un fallo en la comunicación y que se utilizan direcciones alternativas, Supongamos que se eligen IPA2 e IPB2 para esta comunicación. El resultado es que se intercambiarán paquetes con las mismas direcciones IPA2 e IPB2, con el mismo valor de Flow Label, pero que corresponden a comunicaciones distintas, lo que llevará a errores en el procesamiento.

Una forma de solventar esta situación es asignar los valores de Flow Label de forma única en cada nodo. Esto quiere decir que un nodo utilizará valores distintos de Flow Label para cada una

de las comunicaciones que establezca. De esta forma se soluciona el problema presentado. Sin embargo, el campo de Flow Label tiene 20 bits, lo que permite 1.048.576 comunicaciones simultáneas. Si bien es un número considerable, es posible que ciertos servidores requieran un número más elevado de sesiones simultáneas, sobre todo si consideramos que las sesiones de multihoming pueden tener un tiempo de vida mayor que el de una comunicación. En este caso, los valores de Flow Label se repetirían inexorablemente. La solución para esta situación pasa por evitar que los conjuntos de localizadores disponibles para una sesión resulten en combinaciones de dirección destino y origen que también pertenezcan a otras sesiones a las cuales se les ha asignado el mismo valor de Flow Label. El resultado final es que estas sesiones dispondrán de un conjunto de localizadores reducido, viendo así reducida su tolerancia de fallos.

6.4.2.2.2 Destination Option

Otra posible solución es llevar la etiqueta de contexto en una Opción de Destino. Esta parece una opción razonable, ya que la información relacionada con el protocolo de multihoming es claramente de extremo a extremo. Sin embargo, esta opción presenta una serie de dificultades debidas a la especificación de las opciones de destino. En particular, en la especificación de IPv6 [RFC2460] en donde se definen las opciones de destino no se define un orden en el caso en que existan múltiples direcciones de destino. Considerando que el procesamiento asociado a la opción de destino que contiene la etiqueta de contexto resulta en un cambio de las direcciones con las que se presenta el paquete, el orden en que ubiquen las opciones de destino puede afectar al comportamiento final del nodo. Otro problema relacionado con éste ocurre cuando existen múltiples entidades que contribuyen con opciones de destino para un mismo paquete (considérese por ejemplo el caso de un nodo que utiliza tanto multihoming como Mobile IP [RFC3775]). En éste caso el orden en que distribuyan las opciones de destino también afectará el comportamiento final del nodo.

Adicionalmente, la interacción con IPSec [RFC2401] también es problemática, ya que como hemos dicho, la capa de identificación se ubica por debajo de IPSec en la pila de protocolos. Las opciones de destino pueden ubicarse tanto antes o después del encabezado IPSec (siendo antes, más cerca del encabezado IPv6). La opción de poner antes IPSec y después la opción de destino es incompatible con la propuesta de arquitectura. La opción de poner antes la opción de destino y después el encabezado IPSec, implica que la opción de destino debe ser procesada por todos los nodos que se encuentren en el Routing Header, lo que significa que no es posible utilizar el Routing Header en estos casos.

6.4.2.2.3 Extensión Header

Otra opción es el uso de un nuevo encabezado de extensión para transmitir la etiqueta de contexto. Esta opción no presenta ninguna de las dificultades de las opciones anteriores, ya que el

orden de los encabezados de extensión se encuentra perfectamente especificado. Además por ser un encabezado específico para esta aplicación, no se estaría sobrecargando ningún campo existente con funciones ya asignadas.

El inconveniente de esta opción es el coste en bytes a transmitir que genera. Sin embargo, éste coste es relativo, ya que el nuevo encabezado de extensión sólo se incluiría en aquellos paquetes que requieren traducción, es decir aquellos paquetes que pertenecen a comunicaciones para las que se ha establecido un sesión de multihoming y para las que además se ha producido un fallo.

6.4.2.3 Enfoque adoptado

Una vez evaluadas todas las opciones, parece claro que la solución más elegante consiste en el uso de un nuevo encabezado de extensión para el transporte de una etiqueta de contexto que permita identificar la sesión a la que pertenecen los paquetes intercambiados.

En síntesis, el plano de datos de la capa de identificación deberá realizar la correspondencia entre localizadores e identificadores. Para los paquetes entrantes, deberá identificar los paquetes que deben ser traducidos mediante la etiqueta de contexto contenida en un nuevo encabezado de extensión definido con éste propósito. Para los paquetes salientes, la capa de identificación deberá determinar si es necesario utilizar un localizador alternativo al identificador, y caso de que así sea, deberá reemplazar los identificadores por los localizadores correspondientes e incluir el encabezado de extensión con la etiqueta de contexto apropiada.

Para realizar todo este procesamiento, la capa de identificación necesita disponer de información de estado asociada a al sesión de multihoming. Dicha información incluye los identificadores y conjuntos de localizadores asociados a la sesión, así como la etiqueta de contexto de la misma. Esta información es adquirida a través de un protocolo de señalización de multihoming del plano de control de la capa de identificación que será presentado en la sección siguiente.

6.4.3 El plano de control

En esta sección presentaremos el protocolo de señalización de multihoming. Para ello, primero especificaremos el escenario de aplicación del mismo, para luego pasar al protocolo en si mismo. Éste cuenta con tres partes principales, a saber, un protocolo de establecimiento de sesión, un mecanismo para cambiar el localizador utilizado para la comunicación y un mecanismo de finalización de sesión. Cada uno de ellos serán presentados a continuación. El protocolo de

control presentado en esta sección (o componentes del mismo) también ha sido descrito en [Bagnulo2004c] [Bagnulo2005b] [Bagnulo2005c]

6.4.3.1 Escenario de aplicación

Como hemos presentado anteriormente, la solución propuesta utilizará DGCs como identificadores. Como hemos visto, las DGCs utilizadas pueden ser de tres tipos: HBAs, CGAs o híbridas HBA/CGA. Para presentar el caso más general, consideraremos el caso de las direcciones híbridas HBA/CGA.

El escenario de aplicación esta compuesto por un sitio multihomed que se comunica con otro sitio que puede o no ser multihomed. Para presentar el caso más general, asumiremos que ambos sitios involucrados en la comunicación son multihomed, siendo el caso de la comunicación de un sitio multihomed con un sitio single-homed un caso particular del caso anterior en donde el número de proveedores de uno de los sitios es uno.

De esta forma, cada uno de los sitios multihomed tiene múltiples proveedores y por ello, múltiples prefijos disponibles. En éste escenario, cada nodo de un sitio multihomed, generará, un conjunto de HBAs (de direcciones HBA/CGA) utilizando un par de claves pública y privada y el conjunto de prefijos de /64 disponibles en el enlace en el que se encuentra conectado, siguiendo el procedimiento descrito en la sección 6.4.1.2.5.

Cabe notar que debido a que el prefijo incluido en el campo de Prefijo de Subred de la parte principal de la estructura de datos de la CGA cambia, los Identificadores de Interfaz de cada una de las distintas DGC serán distintos.

De esta forma, la configuración de un escenario de aplicación del protocolo consiste en dos nodos, A y B, ubicados en distintos sitios multihomed que establecen una comunicación.

El nodo A está ubicado en el sitio servido por n proveedores, ISPA1, ISPA2,…, ISPAn de forma que en el enlace donde está conectado el nodo A se encuentran disponibles n prefijos PerfA1::/64, PrefA2::/64,…, PrefAn::/64. El nodo A genera entonces n DGCs, PrefA1:A1, Pref2:A2, …, Prefn:An..

Análogamente, el nodo B está ubicado en el sitio servido por m proveedores, ISPB1, ISPB2,…, ISPBm. En el enlace del nodo B se encuentran disponibles entonces m prefijos, PrefB1, PrefB2,…, PrefBm. El nodo B genera entonces m DGCs, PrefB1:B1, PrefB2:B2,…, PrefBm:Bm.

El escenario se ilustra en la figura siguiente:

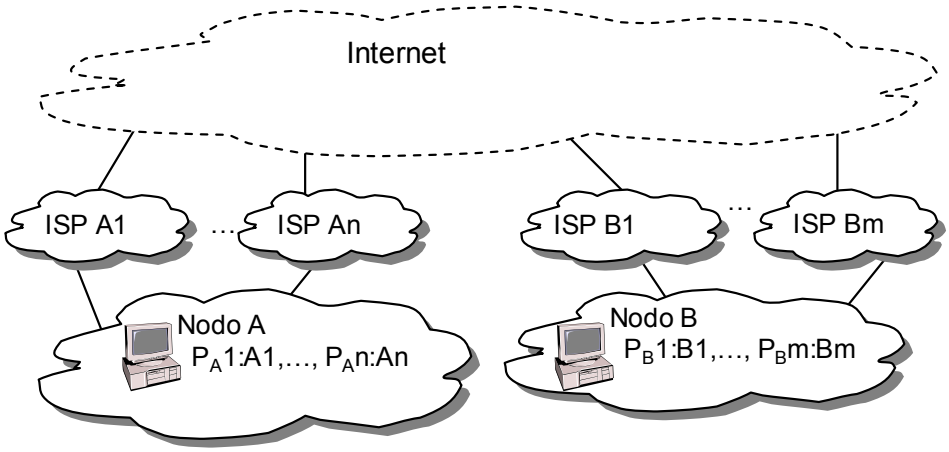

Figura 20: Escenario de aplicación

6.4.3.2 Protocolo de establecimiento de sesión

En el escenario anterior, suponemos que dos nodos, uno de cada sitio establecen una comunicación. Para ello, ejecutan la selección de direcciones para el correcto establecimiento de la comunicación como se detalla en la sección 6.3.1. Una vez que la comunicación está en curso, alguno de los nodos involucrados en la misma puede considerar que la comunicación debería beneficiarse de las facilidades de tolerancia a fallos provista por el protocolo de multihoming, por lo que desea crear una sesión de multihoming asociada a esta comunicación para brindar protección a la misma. Cabe notar que el establecimiento de la sesión de multihoming no tiene porque realizarse en el inicio de la comunicación, sino que es asíncrono a la misma, y permite que la decisión de proteger una comunicación se realice en cualquier momento de la vida de una comunicación. Cuando uno de los nodos involucrados en la comunicación decide establecer la sesión de multihoming, ejecuta el protocolo de establecimiento de sesión como se detalla a continuación.

Mediante el protocolo de establecimiento de sesión se intercambia información básica asociada a la sesión en cuestión, a saber:

- Identificadores de cada uno de los extremos
- Parte del conjunto de localizadores de cada uno de los nodos (al menos uno por extremo) y su correspondiente información de seguridad
- Etiquetas de contexto

El intercambio propuesto consta de cuarto fases para limitar ciertos ataques de negación de servicio. Como resultado del establecimiento de sesión, cada nodo invierte recursos, para

almacenar el estado correspondiente a la misma, lo que puede habilitar ataques de negación de servicio basados en el consumo de memoria. El intercambio en cuatro fases permite limitar dichos ataques mediante dos técnicas: Primero, el extremo que juega el rol de receptor en el intercambio no creará ningún estado asociado a dicha sesión hasta que el iniciador del intercambio no pruebe fehacientemente que ya ha creado su estado correspondiente. De esta forma, si un atacante (iniciador) desea consumir la memoria de una víctima (receptor), el atacante deberá también consumir sus propios recursos para realizar el ataque, ya que la víctima sólo creará estado después que el atacante. Segundo, para llevar a buen término un intercambio en cuatro fases se requiere que ambas partes sean capaces de recibir información, es decir que el localizador usado en el intercambio debe efectivamente corresponder a las partes involucradas (al menos durante el tiempo en el que se realiza en intercambio). Esto facilita la identificación de los atacantes, ya que es posible determinar desde qué dirección IP ha sido lanzado el ataque.

A continuación describiremos el intercambio de establecimiento en detalle.

El primer paquete P1 de intercambio, es esencialmente una solicitud de iniciar el intercambio por parte del iniciador al receptor.

El receptor responde enviando el paquete P2, en el cual se incluye una Prueba de Contacto Previo, que es esencialmente una cadena de caracteres que permite al receptor verificar que el iniciador ha enviado previamente el paquete P1 y que ha creado el correspondiente estado necesario. Dicha Prueba de Contacto Previo se genera como un hash construido a partir de una información privada al receptor y de las direcciones contenidas en el paquete P1. De esta forma, el receptor puede utilizar la misma información privada para todas las solicitudes de inicio de comunicación P1 que reciba, y puede verificarlas simplemente calculando un hash. Esto permite al receptor no guardar ningún estado específico a cada solicitud. El receptor puede responder a este paquete sin generar ningún estado interno, ni realizar ninguna operación criptográfica. Adicionalmente a la Prueba de Contacto Previo, el receptor incluye en el paquete P2 el identificador a utilizar en la sesión, un conjunto de localizadores asociados a dicho identificador y la información de seguridad para verificar la correspondencia entre los localizadores y el identificador. Dado que los identificadores propuestos son HBA/CGAs, la información de seguridad consta de la estructura de parámetros de la CGA usada como identificador. En caso de estar usando las funciones HBA del identificador, se deberá incluir adicionalmente la Extensión de Múltiples Prefijos. En caso de estar usando las funciones de CGA del identificador, se deberá incluir una firma del conjunto de localizadores realizada con la clave privada asociada a la CGA, tal y como se describe en la sección 6.4.1.1.4.

Al recibir P2, el iniciador verifica la validez de la información recibida. Para ello, verifica que la estructura de parámetros se corresponde con el identificador propuesto. Esto se realiza

mediante el procedimiento descrito en la sección 6.4.1.2.6[19]. Si la verificación es exitosa, el iniciador genera el paquete P3, en el que se incluye la Prueba de Contacto Previo, el identificador del iniciador usado en la sesión, una etiqueta de contexto para los paquetes entrantes, el conjunto de localizadores asociados a dicho identificador e información de seguridad que permita validar la correspondencia entre el identificador y el conjunto propuesto de localizadores del iniciador. Análogamente al caso de P2, dicha información consta de la estructura de parámetros de la CGA. En el caso de estar usando las funciones de HBA, también se debe incluir la extensión de Múltiples Prefijos. En caso de usar las funciones de CGA, se deberán incluir la firma del conjunto de localizadores realizada con la clave privada asociada al identificador.

Al recibir el paquete P3, el receptor verifica la Prueba de Contacto Previo y si ésta es correcta, verifica la información de seguridad incluida. Para ello realiza un procesamiento similar al realizado por el iniciador para validar P2. Es decir que verifica que la estructura de parámetros CGA recibida corresponde con el identificador propuesto[20].

Si esta verificación es exitosa, el receptor envía el paquete P4, en el que se incluye una etiqueta de contexto para los paquetes entrantes.

Cabe notar que en el protocolo se utilizaran dos etiquetas de contexto distintas, una para los paquetes entrantes al receptor y otra para los paquetes entrantes al iniciador. También se destaca que estas etiquetas son asignadas por el nodo que va a recibir los paquetes. Esto permite que la etiqueta en cuestión identifique unívocamente la sesión a la que corresponden los paquetes entrantes a cada uno de los nodos involucrados en la comunicación.

6.4.3.3 Tolerancia a fallos

Una vez que se ha establecido la sesión de multihoming, es posible preservar las comunicaciones asociadas a dicha sesión de fallos en los caminos usados para el intercambio de los paquetes. Esto se logra cambiando los localizadores usados para intercambiar paquetes en el momento en que se detecte un fallo en la comunicación. Para preservar la comunicación establecida, es necesario:

1. Detectar el fallo
2. Explorar caminos alternativos

[19] La verificación de la correspondencia entre el identificador y el conjunto propuesto de localizadores se realiza en el momento previo a la utilización de cada localizador en cuestión por motivos de eficiencia que se detallarán después. De momento asumimos que el localizador usado para el intercambio es el identificador, por lo que sólo es necesario verificar el identificador.

[20] A éste caso se aplican las mismas consideraciones recogidas en la ntoa anterior

3. Mover la comunicación del camino que ha fallado al nuevo camino disponible identificado.

A continuación describiremos cada uno de estos pasos.

6.4.3.3.1 Detección de fallos

Existen múltiples mecanismos que brindan información sobre fallos que afectan a las comunicaciones en curso. Para nuestro estudio, consideraremos que los caminos son bidireccionales y decretaremos que hay un fallo cuando al menos el camino en una de las direcciones no se encuentra disponible.

Está claro que en muchos casos las aplicaciones que se estén comunicando serán capaces de detectar fallos en la comunicación. Sin embargo, las aplicaciones actuales no informan sobre los problemas en la comunicación ya que las capas inferiores no necesitan dicha información, Sería posible adaptar las aplicaciones para que brinden dicha información, pero esto no se encuentra actualmente disponible, por lo no se considerará en el análisis.

Por otro lado, ciertas capas de transporte, como TCP, también son capaces de detectar cuándo hay fallos en la comunicación. Adicionalmente, la recomendación de Descubrimiento de Vecinos [RFC2461] requiere que TCP informe sobre el estado de la comunicación, de forma que se evite el procedimiento de *detección de no alcanzabilidad de vecinos* (NUD). Así la información de la que TCP dispone puede ser utilizada para confirmar que el par de localizadores todavía funciona correctamente o que existe un fallo potencial.

En capas más bajas, la propia capa IP puede tener una idea de la alcanzabilidad gracias a dos indicaciones: por un lado, el propio flujo de paquetes informa de que la comunicación fluye normalmente. Para esto es necesario que los paquetes fluyan en ambas direcciones. (Cuando el nodo no envía paquetes, no parece necesario verificar la alcanzabilidad, ya que el nodo no tiene nada que enviar, por lo que la información de alcanzabilidad no parece relevante en ese instante). Por otro lado, la capa IP dispone de información de señalización que puede servir para determinar la disponibilidad de un par de localizadores. En particular, los mensajes de Router Advertisement, que pueden deprecar un prefijo, son relevantes para este caso. Asimismo, los mensajes de ICMP, en particular los de Destino Inalcanzable, también proveen información relevante. Adicionalmente, el par de localizadores usado por el otro extremo para enviar paquetes puede servir como indicio para detectar un posible fallo, ya que puede indicar que el otro nodo considera que ha habido un problema en la comunicación. Finalmente, también resulta relevante para la detección de fallos información local al nodo tal como el fallo de una interfaz,.

A pesar que existe mucha información sobre la alcanzabilidad del otro extremo, hay casos en los que ésta no es suficiente. Por ejemplo, si consideramos flujos unidireccionales sobre UDP, el extremo emisor no tiene información de vuelta, por lo que difícilmente podrá detectar un fallo. En esos casos, será necesario recurrir a un protocolo de verificación de alcanzabilidad propio del

mecanismo de multihoming. Dicho protocolo consta de dos paquetes, una solicitud y una respuesta, y permite determinar si el camino que une un par de direcciones estás disponible. La información incluida en el protocolo de verificación de alcanzabilidad se limita a una cadena de caracteres aleatorios, que permita verificar que efectivamente el nodo al otro extremo ha recibido el paquete de solicitud. El protocolo de verificación de alcanzabilidad será utilizado también para explorar los caminos alternativos disponibles como veremos a continuación.

6.4.3.3.2 Exploración de caminos alternativos

Una vez detectado un fallo, es necesario explorar caminos alternativos a través de los cuales encaminar los paquetes de la comunicación establecida. Para ello, el nodo que ha detectado el fallo intentará utilizar localizadores alternativos del conjunto de localizadores que tiene para cada uno de los identificadores de la sesión. Esto significa que se intentará restablecer la comunicación utilizando ya sea un localizador alternativo para el identificador del propio nodo[21], un localizador alternativo para el identificador del nodo remoto, o ambos.

La correspondencia entre el identificador del propio nodo y los localizadores alternativos para el mismo no requiere verificación alguna ya que dicha correspondencia es del ámbito estrictamente local al nodo. Sin embargo, la correspondencia entre el identificador del nodo remoto y los localizadores alternativos ha de verificarse usando la información de seguridad recibida durante el intercambio inicial, ya que el conjunto de localizadores asociados al nodo remoto ha sido adquirida mediante el protocolo de establecimiento de sesión de multihoming, el cual, como hemos visto, es susceptible a ataques. Cabe recordar que esta verificación no se ha realizado durante el proceso de establecimiento de sesión por motivos de eficiencia, ya que no era necesario hacerlo en esa instancia.

Para realizar la verificación, tomamos la información de seguridad que ha sido recibida en el intercambio de establecimiento de sesión. El procedimiento para verificar la correspondencia entre los localizadores y el identificador del nodo remoto depende de si la seguridad de la correspondencia entre el identificador y el conjunto de localizadores se basa en las funciones de HBA o en las de CGA. Si se basa en HBA, la verificación se realiza mediante el procedimiento presentado en la sección 6.4.1.2.6. Si se basa en CGA, la verificación se realiza mediante el procedimiento presentado en la sección 6.4.1.1.4.

Una vez que se encuentran disponibles localizadores alternativos tanto para el nodo local como para el nodo remoto, es posible explorar si los caminos asociados a las posibles combinaciones de localizador destino y localizador origen son alcanzables. Para ello, se utilizar el protocolo de verificación de alcanzabilidad definido en la sección anterior, con cada una de las combinaciones de localizador destino, localizador origen disponibles. Cabe notar que el número

[21] Ya que esto implica un cambio en el proveedor propio usado para encaminar los paquetes.

de combinaciones localizador destino localizador origen crece como el producto de ambos, lo que lleva a un alto número de posibilidades con un bajo número de localizadores. Esto puede significar un tráfico considerable de señalización para la verificación de la alcanzabilidad. Por ello, se sugiere limitar el número de pares de localizadores verificados simultáneamente.

Cabe notar que el procedimiento de verificación de alcanzabilidad no sólo permite explorar caminos disponibles, sino que también es un mecanismo de seguridad que previene los ataques de inundación, como hemos visto en la sección 4.3.

6.4.3.3.3 Cambio de localizadores usados para la comunicación

Una vez identificado un par de localizadores alternativos asociados a un camino disponible, para cambiar la comunicación hacia este nuevo camino, el nodo simplemente envía los siguientes paquetes con los nuevos localizadores. Adicionalmente, el nodo debe incluir en los paquetes la cabecera de extensión de multihoming conteniendo la etiqueta de contexto correspondiente. De esta forma, la comunicación es preservada a través de fallos.

6.4.3.4 Finalización de sesión

Una vez terminada la comunicación es necesario finalizar también la sesión existente de mutihoming, para así poder liberar los recursos que ésta implica. Dado que el soporte de multihoming brindado por la solución propuesta es transparente para las capas superiores, no es posible esperar que las aplicaciones informen a la capa de identificación cuando terminen la comunicación asociada a la sesión de multihoming. Por ello, es necesario utilizar heurísticos para determinar cuándo finalizar la sesión de multihoming.

El enfoque propuesto en la presente Tesis se basa en la filosofía de “soft state”. Esto implica que cuando el estado es creado, se le asocia un tiempo de vida determinado que se va decrementando con el tiempo. Dicho tiempo de vida puede ser extendido por ciertos eventos de refresco de estado, que se definirán a continuación. En caso de ausencia de tales eventos de refresco, el tiempo de vida expira y el estado es descartado y la sesión es finalizada.

Para ello, definiremos los siguientes eventos como válidos para extender el tiempo de vida de una sesión:

- La recepción de paquetes de datos asociados a la sesión en cuestión.
- La recepción de paquetes de señalización asociados a la sesión en cuestión

De esta forma, pasado un tiempo de vida T sin recibir paquetes de ningún tipo para una sesión dada, el estado asociado a una sesión será descartado[22].

[22] Cabe notar que en el caso de flujos unidireccionales, el comportamiento del protocolo de control implicará la recepción periódica de los mensajes asociados al protocolo de verificación de alcanzabilidad,

Cabe notar que es posible reestablecer la sesión simplemente ejecutando el protocolo de establecimiento de sesión nuevamente.

6.4.3.5 Maquina de estados

6.4.3.5.1 Estados posibles de los pares de localizadores

Como ya hemos presentado anteriormente, para el inicio de la comunicación, un localizador destino puede ser alcanzable cuando se utiliza un localizador origen dado, pero inalcanzable cuando se utiliza otro localizador origen. Por esto, el estado de alcanzable o no alcanzable se refiere a un par de localizadores origen/destino y no a un localizador destino.

Los estados posibles que pueden estar un par de localizadores origen/destino son los siguientes:

- Alcanzable: cuando se tiene información positiva que el par de localizadores es alcanzable
- No alcanzable: cuando se tiene información positiva que el par de localizadores es no alcanzable
- Indeterminado: cuando no hay información reciente de la alcanzabilidad del par de localizadores.
- Test: cuando se está ejecutando el protocolo de verificación de alcanzabilidad para el par de localizadores.

Adicionalmente, uno de los pares de localizadores será marcado como preferido ya que será el que se utilice para enviar paquetes. Los localizadores preferidos pueden estar en estado “alcanzable”, “indeterminado” y “test”, por lo que podemos representar esto como tres estados adicionales, “preferido alcanzable”, “preferido indeterminado” y “preferido test”.

6.4.3.5.2 Transiciones entre los estados

A continuación ilustramos los estados y las transiciones entre los mismos en un diagrama de estados.

---

los cuales extenderán el tiempo de vida de la sesión, incluso cuando no se reciben paquetes de datos para dicha sesión.

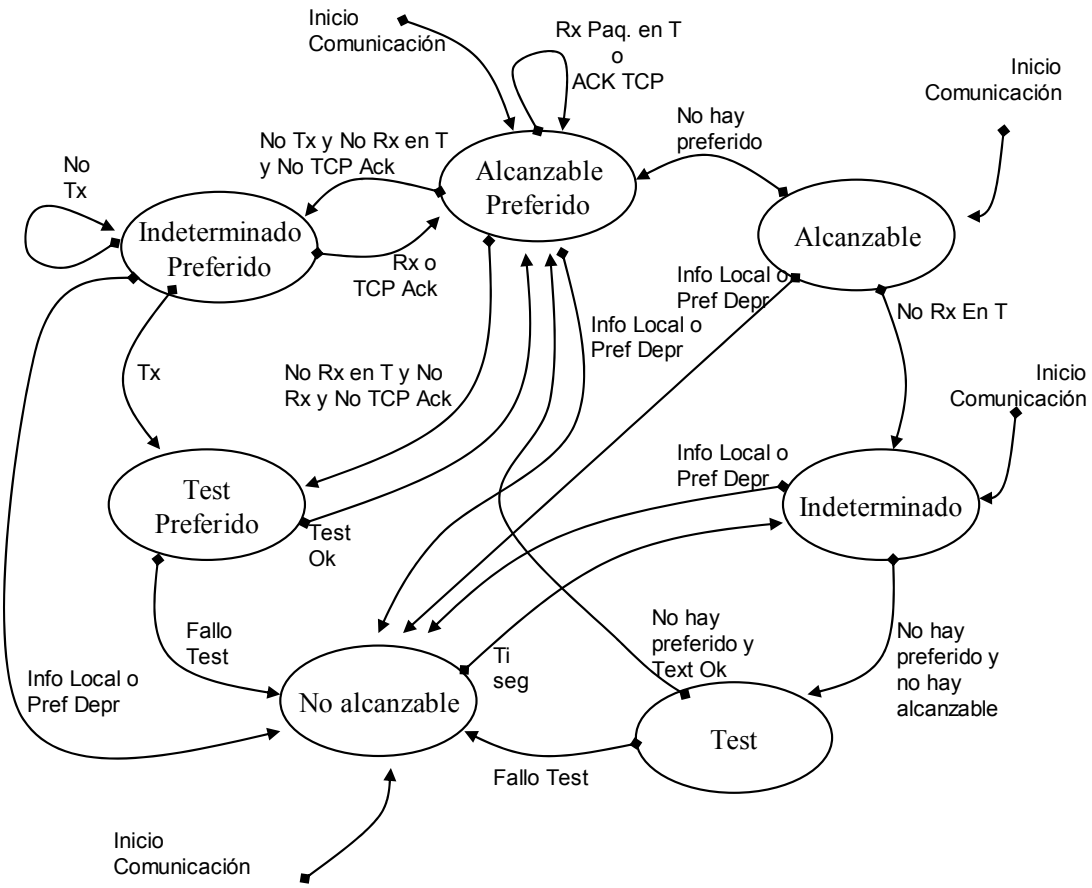

Figura 21: Diagrama de estados

La transición entre los distintos estados vendrá dada por los siguientes criterios:

- Cuando una comunicación entre dos nodos se inicia, se ha realizado el intercambio de localizadores y la alcanzabilidad de ciertos pares de localizadores se ha verificado. Para aquellos que la verificación de alcanzabilidad ha sido exitosa, el estado será de "alcanzable". Para los que la verificación de alcazabilidad ha fallado, su estado será "inalcanzable". Para aquellos que no se ha realizado la verificación, su estado será "indeterminado". En particular, el par de direcciones usado como identificadores para la capa superiores debe ser de "alcanzable" al comienzo de la comunicación y es elegido como "preferido", por lo que estará en estado "preferido alcanzable".
- Si se han recibido paquetes en los últimos T segundos, y estos paquetes contienen el par de localizadores preferido, este par de localizadores queda en estado "preferido alcanzable".
- Si existe una realimentación positiva por parte de TCP u otro protocolo superior, el par de localizadores usado queda como "preferido alcanzable" aunque no se reciban paquetes en los últimos T segundos.

- Si no se han recibido paquetes en los últimos T segundos y no hay una confirmación por parte de los protocolo superiores y no hay paquetes para enviar, el par de localizadores pasa a un estado de “preferido indeterminado”.
- Si no se han recibido paquetes en los últimos T segundos y no hay una confirmación por parte de los protocolo superiores y hay paquetes para enviar, el par de localizadores pasa a un estado de “preferido test”.
- Si no se han recibido paquetes en los últimos T segundos y no hay una confirmación por parte de los protocolo superiores y no hay paquetes para enviar en los últimos T segundos, el par de localizadores pasa a un estado de “preferido indeterminado”.
- Mientras no fluyan paquetes, el par de localizadores en estado “preferido indeterminado” queda en dicho estado.
- La recepción de paquetes con el par de localizadores en cuestión, este par de localizadores pasa de “preferido indeterminado” a “preferido alcanzable”.
- Información positiva de las capas superiores pasan el par de localizadores de “preferido indeterminado” a “preferido alcanzable”.
- Cuando se empieza a enviar paquetes usando el par de localizadores en estado “preferido indeterminado” y no se reciben paquetes al cabo de T segundos, el par de localizadores pasa al estado “preferido test”.
- Cuando un par de localizadores entra en estado “preferido test”, se ejecuta el protocolo de verificación da alcanzabilidad, enviando un mensaje de solicitud. Si la verificación es exitosa, el par de localizadores pasa al estado “preferido alcanzable”. Si la verificación falla, el par de localizadores pasa a estado “inalcanzable” y se elige un nuevo par de localizadores como “preferido”. Los localizadores con estado “alcanzable” tiene mayor prioridad, pero si no existe ningún par de localizadores en este estado, se elige un par de localizadores en estado “indeterminado”.
- Un par de localizadores permanece en el estado “inalcanzable” durante Ti segundos y luego es enviado al estado “indeterminado”.
- Información local al nodo, como un fallo en una interfaz, pone los pares de localizadores que contienen los localizadores locales afectados en estado “inalcanzable” por un tiempo indeterminado, hasta la reparación del fallo.
- Cuando un prefijo es deprecado, los pares de localizadores que contienen los localizadores locales afectados se ponen en estado “inalcanzable” por un tiempo indeterminado, hasta que el prefijo sea valido nuevamente.
- Un mensaje de ICMP de Destino inaccesible mueve un par de localizadores “alcanzables” hasta un estado de “indeterminado”.

6.4.3.6 Recorrido por el protocolo

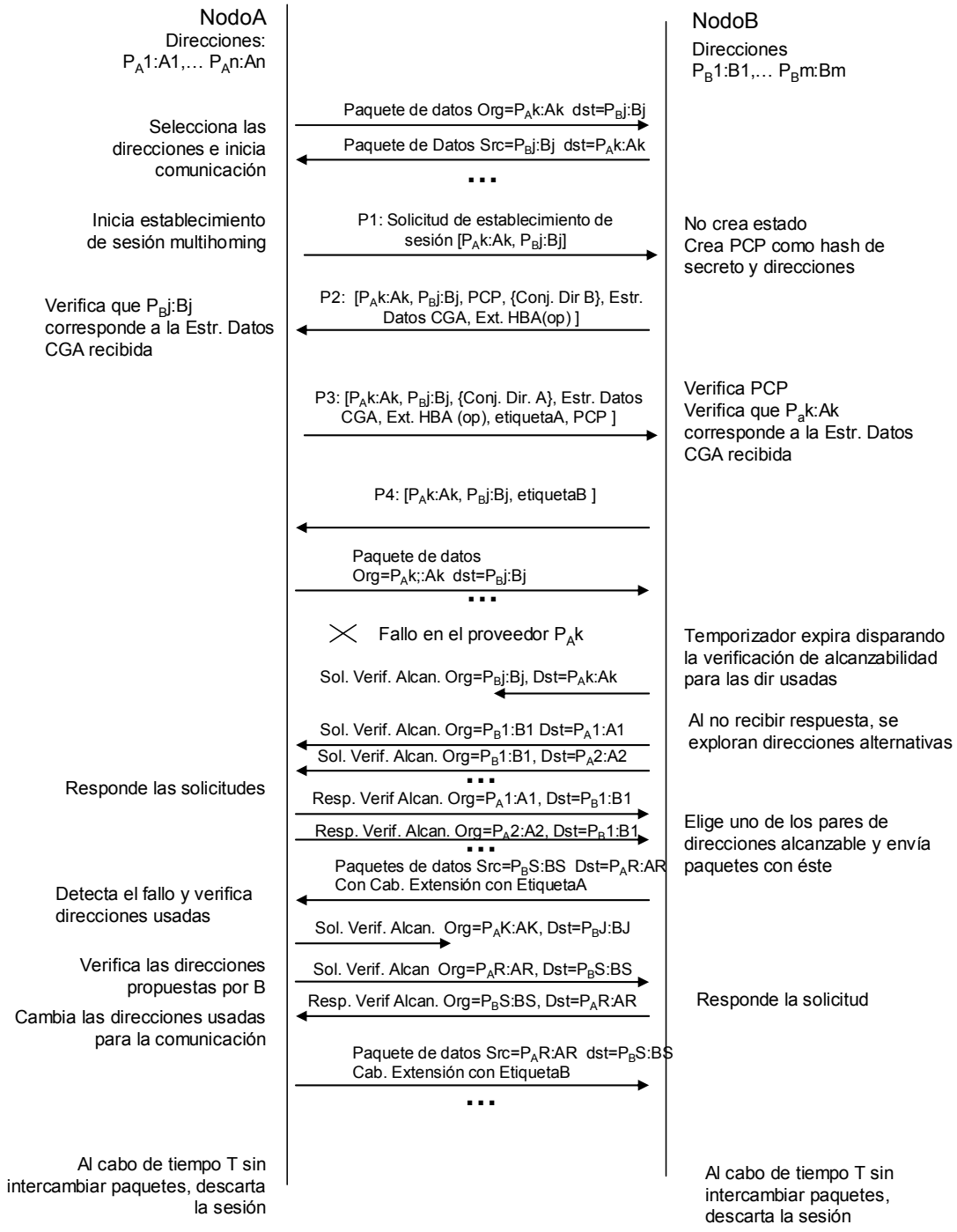

Figura 22: Recorrido por el protocolo

En esta sección ilustraremos el funcionamiento del protocolo de control presentado. En la figura se puede observar como dos nodos ubicados en dos sitios multihomed establecen una

comunicación. Pasado un tiempo, deciden ejecutar el protocolo de establecimiento de sesión para proteger la comunicación frente a posibles fallos en el camino. Un tiempo más tarde, ocurre un fallo que afecta la comunicación. Cuando éste es detectado, se lanza el procedimiento de exploración de caminos presentado anteriormente y se identifica un camino alternativo. La comunicación se mueve a dicho nuevo camino, reestableciéndose el intercambio de paquetes de datos. Una vez que finaliza el intercambio de paquetes de datos entre las partes, el tiempo de vida de la sesión expira en ambos extremos, lo que causa que el estado asociado sea descartado.

Capítulo 7

Trabajo Relacionado

7.1 Introducción

En el presente capítulo presentaremos otros mecanismos propuestos para brindar soporte de multihoming. Cabe notar que los problemas relacionados con el soporte de multihoming usando direccionamiento basado en proveedor ya habían sido identificados en el año 1992 cuando el esquema CIDR fue propuesto, por lo que existe una extensa lista de propuestas para abordar este problema. En el momento de la redacción de la presente Tesis Doctoral hay más de 30 propuestas que han sido presentadas en el grupo de trabajo multi6. A continuación presentaremos algunas de las propuestas más relevantes que han sido consideradas en los últimos cuatro años. Por otros mecanismos adicionales, el lector puede referirse al compendio de soluciones [Bagnulo2001a], [Bagnulo2005a], [Bagnulo2002a].

Para la presentación de los mecanismos, se ha realizado una taxonomía con cuatro categorías, a saber: mecanismos basados en el sistema de encaminamiento, mecanismos basados en la capa de transporte, mecanismos basado en la separación de identificador y localizador a nivel de red, y mecanismos que proveen soluciones parciales al problema. Cada una de estas categorías se presenta a continuación.

7.2 Mecanismos basados en el sistema de encaminamiento inter-dominio

Existen esencialmente dos tipos de propuestas basadas en el sistema de encaminamiento, a saber, las soluciones del tipo de las utilizadas actualmente en IPv4, y las soluciones basadas en alguna variante de agregación geográfica. Los detalles de las propuestas se presentan a continuación.

7.2.1 Solución tipo IPv4

Existen ciertas propuestas que plantean la utilización de la solución actualmente utilizada en IPv4 para el caso de IPv6. Estas propuestas tienen presentes las limitaciones de escalabilidad de este tipo de soluciones, por lo que no proponen dicha solución como una solución definitiva, sino como una solución transitoria mientras se desarrolla y adopta una solución más escalable. En particular, la solución propuesta en [Lindqvist2002a] plantea adoptar la solución usada en IPv4 en esta etapa inicial durante la cual las tablas de rutas tienen aún un tamaño reducido, por lo que es posible soportar el coste de las entradas adicionales supuesto por la solución de multihoming. En el caso de la propuesta presentada en [Savola2004a], la idea es que aquellos sitios que disponen un número de sistema autónomo propio accedan a un prefijo independiente del proveedor (generado a partir del número de sistema autónomo), y que puedan inyectar dicho prefijo en el sistema de rutas de interdominio. De esta forma, sólo aquellos sitios que tienen un número de sistema autónomo pueden acceder a un prefijo independiente del proveedor, lo cual actuaría como una restricción al número de sitios que pueden acceder a un prefijo de estas características. Sin embargo queda la duda de si la adopción de esta solución no generaría una sobre demanda de números de sistema autónomos, lo que implicaría no sólo un alto numero de rutas en las tablas de rutas debidas a la solución de multihoming, sino también un consumo excesivo de números de sistema autónomo.

Nuestra opinión es que es inevitable que los grandes sitios de Internet accedan a sus propios bloques independientes de sus proveedores y que los inyecten a través del sistema de rutas de interdominio. Aunque esto no es deseable en general, no resulta un gran problema si el número de sitios no es muy grande. Lo que sí es necesario es evitar que todos los sitios multihomed inyecten sus propios prefijos, como sucede actualmente, ya que el sistema de rutas no soportaría tal carga. Para ello, es necesario que exista otra solución alternativa para los sitios más pequeños, que no está contemplada en las soluciones propuestas en esta sección. A medida que la solución alternativa es más atractiva, mayor será el número de sitios que la adopte.

7.2.2 Agregación geográfica

Otro enfoque propuesto es el uso de direccionamiento geográfico. En este caso, cada sitio obtiene un bloque de direcciones correspondientes a la zona geográfica donde se encuentra. De esta forma, cada sitio tiene un solo prefijo, independiente del proveedor. La agregación ya no se basa en los proveedores sino en las zonas geográficas. Existen distintas propuestas de cómo asignar las direcciones dentro de las zonas geográficas, dentro de las cuales cabe destacar la generación del prefijo a partir de la longitud y latitud [Hain2004a] y la asignación de prefijos en función de los países [Py2002a]. La agregación geográfica sufre de dos problemas importantes, a saber:

- La topología de la red puede no tener (y de facto no tiene) una relación directa con la geografía, por lo que la agregación resultante puede ser reducida. Es decir, sólo es posible agregar una zona geográfica que sea interiormente conexa, es decir que todos los proveedores que proveen servicio dentro de la zona se conecten entre sí dentro de la zona geográfica. Si esto no es así, es necesario romper la agregación e inyectar rutas más específicas que separen el tráfico entre las distintas partes de no conexas que componen la zona. La agregación se logra entonces sólo a partir de esas regiones conexas. Estas zonas pueden ser realmente muy extensas y es posible que existan problemas de escalabilidad incluso dentro de ellas mismas, debido al gran numero de rutas necesarias para esta zona. Hay quienes proclaman que a medida que la topología se hace mas mallada, la conectividad aumenta y este problema disminuye. Sin embargo, no es evidente que la conectividad aumente dentro de la zona geográfica porque, por ejemplo, es posible que la conectividad aumente entre zonas geográficas distintas, como puede ocurrir con los cables transatlánticos que unen América con Europa.
- Incluso si una zona geográfica es conexa, esto presenta problemas con el modelo de negocio. Para lograr la agregación deseada, es necesario que todos los proveedores que prestan servicio en una zona geográfica anuncien el prefijo de dicha zona en el sistema de encaminamiento de interdominio. Esto implica que todos los proveedores que cubren una zona geográfica deben estar dispuestos a aceptar tráfico dirigido a clientes de otros proveedores que cubren esa misma zona. Esto puede presentar problemas en el modelo de negocio, ya que, por ejemplo pueden existir proveedores de muy distinto tamaño, de forma que la distribución de tráfico sea muy desigual, por no entrar en que el servicio recibido por los clientes de un ISP puede verse afectado por el tratamiento recibido por otro proveedor.

Además de los esquemas de direccionamiento geográfico generales discutidos anteriormente, se han planteado esquemas de tipo de direccionamiento geográfico pero de alcance mas reducido, como pueden ser:

- Agregación basada en puntos neutros [Fernández2004a] en este caso, se asigna un bloque de direcciones al punto neutro, de forma que los clientes de los ISPs presentes en el punto neutro obtengan direcciones del bloque del punto neutro y no de los bloques de los ISPs. Esto permite al sitio cambiar de ISP sin necesidad de cambiar de direcciones, siempre y cuando el nuevo ISP también tenga presencia en el punto neutro. También permite que un sitio multihomed a dos de los ISPs pertenecientes al punto neutro sólo tenga un prefijo perteneciente a las direcciones del punto neutro. La ventaja de este enfoque, es que por definición el direccionamiento es coherente con la topología física, ya que todos los ISPs pertenecen al punto neutro. La limitación que presenta este esquema es que el punto neutro es un punto simple de fallos. Adicionalmente, este enfoque presenta una dificultad desde el punto de vista del modelo de negocio, ya que todos los proveedores conectados al punto neutro deben anunciar a través de BGP el bloque agregado que ha sido asignado al ISP. Esto resulta en que los ISPs recibirán y deberán cursar tráfico de clientes de otros ISPs del punto neutro. Si bien esto es posible técnicamente, surgen dificultades al momento de hacer esta configuración compatible con el modelo de negocio actual de los ISPs. Existen alternativas para la tarificación basada en el tráfico intercambiado entre los ISPs, pero esto al menos impone un cambio en el modelo de negocio actual.
- Otra propuesta basada en direccionamiento geográfico se basa en utilizar la agregación geográfica de forma interna a los sitios [Beijnum2004b]. En este caso, el intercambio de rutas a nivel interdominio sería similar al actual, sólo que dentro del sitio se agregará basándose en la geografía. Dado que la agregación es interna al sitio, este esquema de agregación no presenta la dificultad en el modelo de negocio que sufre el esquema genérico de agregación geográfica presentada anteriormente. Sin embargo, debido también a que la agregación es interna, este enfoque no reduce la cantidad de información de rutas intercambiada y propagada entre los proveedores, por lo que los problemas relacionados con dicho volumen de información persisten.

7.3 Mecanismos basados en la capa de transporte

Como es evidente, los mecanismos basados en la capa de transporte son específicos de la capa de transporte en cuestión, por lo que deberán existir tantos mecanismos como capas de transporte. Existen múltiples propuestas de soluciones de multihoming para las distintas capas de transporte existentes. Las propuestas pueden dividirse en propuestas para capas de transporte que

naturalmente soportan múltiples direcciones por conexión de transporte y las propuestas que modifican una capa de transporte existente que no tiene soporte de múltiples direcciones por conexión. A continuación se presentan estas dos categorías.

7.3.1 Capas de transporte que soportan múltiples direcciones por conexión.

Existen dos capas de transporte que soportan nativamente múltiples direcciones por conexión, a saber SCTP [RFC2960] y DCCP [Kohler2003a]. Para estos dos niveles de transporte sería posible que las conexiones sobrevivieran fallos en las comunicaciones ya que es posible utilizar una dirección alternativa en la conexión establecida. Se ha realizado un esfuerzo considerable en detallar como sería posible soportar conexiones SCTP en entornos multihomed [Coene2003a], [Coene2004a]. Como ya se ha comentado anteriormente, el problema de esta solución es que no podría ser utilizada por las aplicaciones actuales que utilizan TCP o UDP.

7.3.2 Capas de transporte que no soportan múltiples direcciones por conexión

Si bien ciertas capas de transporte soportan múltiples direcciones en su diseño inicial, está claro que las capas de transporte más usadas actualmente no poseen esta propiedad, por lo que para brindar un soporte universal de multihoming a nivel de transporte será necesario adaptar estas capas de transporte existentes para que soporten múltiples direcciones por conexión. Existen diversas propuestas para dotar a TCP de esta capacidad de multidirección, dentro de los cuales podemos destacar [Matsumoto2004a], [Huitema1995a].

Estos enfoques esencialmente definen extensiones para llevar múltiples direcciones o prefijos en los paquetes de SYN de establecimiento de conexión TCP. La información puede ser llevada en extensiones TCP o en Extension Headers de IPv6.

La seguridad se basa en el uso de claves que son incluidas en los paquetes cuando una nueva dirección es usada. Cabe notar que los ataques sólo afectan a una conexión y no a toda la identidad del nodo, por lo que el reducido nivel de seguridad es aceptable, como ya mencionamos en el análisis de seguridad.

Hasta el momento, el autor de la presente Tesis Doctoral no tiene conocimiento de ninguna iniciativa para dotar a UDP de la capacidad de soportar múltiples direcciones. Esto puede deberse a que UDP es un protocolo no orientado a conexión, por lo que UDP no tiene conocimiento de las distintas comunicaciones existentes. Esto implica que el tráfico UDP no tendrá soporte de multihoming a nivel de transporte y el soporte deberá ser brindado a nivel de aplicación.

7.4 Mecanismos basados en la separación de identificador y localizador a nivel de red

Existen múltiples propuestas basadas en la separación de identificador y localizador a nivel de red o en un nuevo nivel de identificador. A continuación se presentará una selección de las propuestas existentes. Las propuestas están agrupadas en función de la naturaleza del identificador usado. Se presentarán propuestas que utilizan los siguientes tipos de identificadores: un localizador preferido como identificador, identificadores criptográficos, identificadores efímeros, identificadores extraídos de un espacio reservado del espacio de direccionamiento IPv6, y direcciones criptográficamente generadas.

7.4.1 Localizadores preferidos como identificadores

7.4.1.1 Solución basada en MIPv6

Durante estos últimos años, la comunidad de Internet ha desarrollado el protocolo MIPv6 [RFC3775] que brinda soporte de movilidad a nivel IP. Esto significa que preserva las comunicaciones establecidas a través de cambios en el punto de conexión a la red debidos al movimiento. El problema de la movilidad parece tener parecido con ciertos aspectos del soporte de multihoming, en particular con preservar las comunicaciones establecidas a través de cambios en la conectividad debidos a fallos. Parece entonces interesante investigar la posibilidad de aplicar el protocolo MIPv6 para brindar soporte de multihoming, como se realiza en [Bagnulo2003a].

Esta propuesta esencialmente explora el uso de los mensajes de Binding Update definidos en el protocolo MIPv6 para informar sobre las direcciones alternativas a utilizar cuando se produce un fallo. Para ello, el nodo multihomed asume el rol de nodo móvil y el nodo exterior asume el rol de nodo correspondiente. De esta forma, cuando se establece la comunicación la dirección del nodo multihomed usada adopta el rol de HoA. Mientras que no ocurra un fallo que afecte a la dirección inicialmente elegida, la comunicación utiliza la misma. Sin embargo, si se produce un fallo en la comunicación, se envía un paquete de Binding Update al otro extremo de la comunicación, conteniendo la dirección inicial como HoA y una de las direcciones alternativas como CoA, de forma que la comunicación continúe a través de la dirección alternativa. Dicho mensaje de Binding Update debe ser validado mediante información obtenida a través del procedimiento de encaminamiento inverso definido por MIPv6. Sin embargo, las asociaciones

entre HoAs y CoAs validadas por dicho mecanismo tienen una validez de 7 minutos para evitar ataques desplazados en el tiempo [Nikander2004a], de forma que después de 7 minutos dicha asociación se terminará y la comunicación fallará. Esto implica que el protocolo MIPv6 en su estado actual no es suficiente para brindar soporte de multihoming. Es posible realizar ciertas modificaciones que permitan extender la vida de las asociaciones entre la HoA y la CoA en el nodo correspondiente, como la propuesta en [Bagnulo2003b] [Bagnulo2003c] [Bagnulo2003d].

Sin embargo, estas modificaciones requieren modificaciones en los nodos correspondientes, lo que hace que pierda el atractivo principal de la reutilización de mecanismos ya existentes.

7.4.1.2 MEX

Las ideas que recoge MEX [Bagnulo2002d], [Bagnulo2003e] en su propuesta son las siguientes: a) la información que relaciona múltiples direcciones (o prefijos) como pertenecientes a un mismo nodo no es suficientemente agregable como para poder ser almacenada en el sistema de rutas de interdominio, por lo que debe ser almacenada en los nodos finales y b) una vez que los paquetes son entregados al sistema de encaminamiento, sólo este puede reencaminarlos cuando se produce un fallo. Por ende, puede ser beneficioso que la información de múltiples direcciones alternativas disponibles (almacenada en los nodos finales) sea comunicada al sistema de encaminamiento para que este reencamine los paquetes cuando se produce un fallo. Para esto se define un Extensión Header que transporte todas las direcciones alternativas para la dirección destino contenida en un paquete. De esta forma, si ocurre un fallo, los encaminadores sólo deben extraer una dirección alternativa del Extension Header, y reencaminar el paquete hacia esta nueva dirección. De esta forma, la información de direcciones alternativas no es almacenada en los encaminadores, pero puede ser utilizada por estos cuando se produce un fallo. El compromiso adquirido es una disminución del ancho de banda frente a una liberación de memoria en los routers y una reducción en la carga de la propagación de rutas, supuesto que hay muchos prefijos específicos no se tienen que inyectar para obtener beneficios de multihoming. Es decir que en lugar de utilizar memoria en los routers para almacenar la información de direcciones alternativas, esta información es contenida en los propios paquetes. El mayor problema de esta solución es el esfuerzo de despliegue, ya que sería necesaria la modificación tanto de los nodos finales como de encaminadores. La propuesta plantea también un mecanismo para limitar los encaminadores que requieren modificación mediante la inyección de rutas por defecto dentro de los distintos dominios administrativos, que podría simplificar el despliegue. Otro problema que presenta esta solución, es que la detección de fallos es realizada por los encaminadores, lo que si bien permite que los paquetes en curso sean reencaminados, impone limitaciones en los tiempos de respuesta. Esto se debe a que la estabilidad es más relevante que precisión en la topología para sistema de encaminamiento interdominio. Es decir que es mejor que el sistema de

encaminamiento interdominio tenga un conocimiento de la topología que no se encuentre completamente actualizado a que el sistema de interdominio sea inestable. Esto se refleja en los tiempos utilizados para determinar un fallo en un enlace. El valor por defecto en los encaminadores comerciales es de 3 minutos para la detección de una falla. Esto quiere decir que un enlace debe estar caído por 3 minutos para que BGP considere que éste ha fallado. Esto implica que sólo después de esto los paquetes serán reencaminados hacia la dirección alternativa y mientras, los paquetes serán descartados, por lo que existirá perdida de paquetes en cualquier caso.

7.4.1.3 NOID

NODI [Nordmark2004a] es un mecanismo extremo a extremo que obtiene el conjunto de direcciones disponibles para ambos extremos a partir del DNS. De esta forma, es posible obtener el conjunto de direcciones disponibles con un nivel de seguridad equivalente al disponible actualmente.

El protocolo esencialmente se basa en que el nodo que inicia la comunicación obtiene a través de una búsqueda en el DNS las direcciones disponibles para el otro extremo (y conocimiento sobre si este soporta el protocolo NOID). Una vez que obtiene las direcciones, establece la comunicación eligiendo una de las direcciones disponibles como identificador para las capas superiores. Cuando el extremo receptor recibe un paquete NOID, realiza una búsqueda inversa en el DNS, y luego una búsqueda directa por el nombre obtenido en la búsqueda inversa, obteniendo así todas las direcciones disponibles para el nodo iniciador. De esta forma, ambos extremos descubren las direcciones disponibles para el otro extremo. La selección de la dirección origen es realizada por el router de salida del sitio origen, de forma que sea topológicamente correcta. Esto sirve para lidiar con los filtros de ingreso, pero también como un mecanismo de detección de fallos, ya que la dirección origen indicará un prefijo para el cual el camino se encuentra disponible (asumiendo caminos bidireccionales).

Las limitaciones identificadas en NOID son por una parte la necesidad de mantener el árbol inverso del DNS, actualmente muy pobremente mantenido incluso para sitios grandes, por lo que es mucho más difícil de mantener para redes domésticas o para redes móviles. Adicionalmente, el soporte de las extensiones de privacidad definidas en la RFC 3041 es difícil en esta solución, ya que requiere la modificación dinámica de los registros de DNS por parte de los nodos finales.

7.4.1.4 Otras propuestas

Existen muchas otras propuestas basadas en la creación de una capa de identificación y en el uso de uno de los localizadores como identificador para las capas superiores. Dentro de estas se

pueden mencionar ODT, MAST, y MHTP. ODT [Beijnum2004b] no considera que los ataques desplazados en el tiempo sean un problema para aquellas comunicaciones que no estén protegidas mediante criptografía (IPSec, TLS) por lo que asume como suficiente el uso de claves intercambiadas al principio de la comunicación. MAST [Crocker2003b] utiliza Purpose Build Keys [Bradner2003a] para la protección, que también son vulnerables a ataques desplazados en el tiempo. MHTP [Py2004a] crea túneles entre los routers de salida de los sitios involucrados en la comunicación.

7.4.2 Identificadores criptográficos

Existen varias propuestas basadas en el uso de identificadores puramente criptográficos. Dentro de éstas podemos encontrar la aplicación de HIP [Moskowitz2005a] al problema de multihoming [Nikander2005a]. HIP utiliza como identificadores los hashes de las claves públicas que tiene cada nodo HIP. De esta forma cada nodo es capaz de probarse dueño de su identificador mediante criptografía de clave pública. El protocolo HIP es utilizado para generar una Asociación de Seguridad IPSec entre los extremos para que el tráfico intercambiado entre los extremos se encuentre protegido por IPSec. La mayor dificultad que presenta esta propuesta es su alto costo computacional y de ancho de banda, ya que exige que todos los paquetes de la comunicación utilicen ESP y además exige que se realice un conjunto de operaciones costosas al comienzo de la comunicación (Diffie-Hellman, verificación de claves) cuando es posible que esto no sea necesario para el propósito de multihoming.

SIM [Nordmark2003a] es una propuesta que recoge los aspectos esenciales de HIP, pero que subsana estos inconvenientes presentados por HIP para su aplicación al multihoming. SIM protege solamente los paquetes de señalización y deja como opcional la protección de los paquetes de datos. Adicionalmente SIM permite que las verificaciones de claves se realicen en el momento donde sea necesario introducir un localizador alternativo al que se encuentra en uso. Sin embargo, SIM no resuelve todos los problemas ya que persisten dificultades en el manejo de los identificadores por parte de las aplicaciones. En particular, las aplicaciones que realizan referencias no son soportadas por este mecanismo, porque si la referencia sólo incluye el identificador, no es posible para el extremo que recibe la referencia obtener los localizadores asociados al mismo. Un problema similar ocurre con las aplicaciones que desean utilizar el identificador después de un largo periodo de inactividad del mismo, ya que los mecanismos de recolección de basura pueden haber borrado la información referente al conjunto de localizadores asociados a un identificador criptográfico.

7.4.3 Identificadores efímeros

WIMP [Ylitalo2004a] es una solución extremo a extremo basada en el uso de identificadores efímeros para el extremo iniciador de la comunicación. En este mecanismo, el nodo iniciador genera su identificador de forma aleatoria y el tiempo de vida de este está limitado a una comunicación. De esta forma, el identificador del nodo iniciador sólo representa la identidad del nodo durante la vida de la comunicación, por lo que la seguridad tiene menos valor que para el caso de un identificador estable. En particular no parece necesario brindar protección frente a ataques desplazados en el tiempo, ya que en un tiempo futuro, el identificador efímero habrá perdido su valor. La protección de redirección de la comunicación se realiza mediante cadenas de hash inversas, que permiten verificar que el nodo en el nuevo localizador es el mismo nodo que estaba comunicando a través de localizador anterior. Como identificador del receptor, para lo que se necesita un identificador estable para recibir comunicaciones entrantes, se utiliza el hash del FQDN, y los localizadores asociados a éste son obtenidos del DNS, lo que brinda un nivel de confianza similar al actual.

La selección de localizador origen se realiza por los router de salida del sitio origen al igual que en NOID y también se utiliza el mismo sistema de detección de fallos.

La principal dificultad que presenta WIMP es la falta de un mecanismo para obtener los localizadores asociados a un identificador, como en HIP o SIM, agravado por el hecho de que en WIMP el identificador es efímero, por lo que es posible que el identificador en cuestión ya no sea válido. Esto implica que las aplicaciones que hacen referencias o que tienen comunicaciones con largos periodos de inactividad no son soportadas.

7.4.4 Espacio IPv6 reservado para los identificadores

LIN6 [Teraoka2003a] es una propuesta para el soporte conjunto de movilidad y multihoming. La idea fundamental se basa en reservar parte del espacio IPv6 como espacio de identificadores y utilizar identificadores únicos en los 64 bits del Identificador de Interfaz. Este mecanismo presenta diversas dificultades, dentro de las cuales podemos destacar la reducida seguridad, ya que esta se basa en cookies intercambiadas por los nodos, y la dificultad en asegurar la unicidad de los Identificadores de Interfaz de 64 bits.

7.5 Soluciones parciales

Como hemos visto, la solución de multihoming para IPv4 se basa esencialmente en propagar las múltiples rutas existentes hacia el sitio multihomed hasta todos los routers del sistema de encaminamiento de inter-dominio. Debido a que el número de rutas propagadas es proporcional al número de sitios multihomed, la escalabilidad del sistema es limitada. Un posible paliativo a este problema de escalabilidad es limitar el número de routers que tienen conocimiento de la existencia de las múltiples rutas hacia el sitio multihomed. En particular es posible limitar dicho conocimiento a los routers de los ISPs que proveen servicio al sitio multihomed, que tienen un incentivo económico para hacerse cargo de las complicaciones adicionales que esto implica. Este enfoque presenta mejor escalabilidad que la propagación de rutas a todo el sistema de encaminamiento, ya que los routers ISPs sólo conocen las rutas alternativas a los sitios multihomed que prestan servicio. Dicho de otra forma, se limita el alcance topológico de la información de rutas alternativas, ya que en la solución para IPv4 el alcance de la información de rutas alternativas es toda la Internet y en este caso dicho alcance está limitado a los routers de los ISPs del sitio multihomed.

La consecuencia natural de este enfoque es una reducción en las capacidades de tolerancia a fallos de la solución con respecto al caso de la solución usada para IPv4. Esto se debe a que sólo los routers que tienen conocimiento de la existencia de rutas alternativas son capaces de tolerar fallos en las rutas que están usando. En el caso de la solución usada en IPv4, estos son todos los routers de Internet, por lo que la solución es capaz de tolerar fallos en toda Internet, siempre y cuando exista un camino alternativo. En el caso del nuevo enfoque propuesto, sólo los routers de los ISPs del sitio multihomed tienen conocimiento de las rutas alternativas, por lo que esta solución solo es capaz de tolerar fallos que afectan a dichos ISPs. A continuación presentaremos la solución presentada en la RFC 2260 [RFC2260], basada en este enfoque. Luego, identificaremos otras limitaciones de esta solución para después proponer una herramienta propuesta por el autor para solventar la limitación identificada.

7.5.1 Una solución de dominio restringido: RFC 2260

A continuación describiremos la solución presentada en la RFC 2260 [RFC2260], cuya versión para IPv6 se encuentra descrita en la RFC 3178 [RFC3178]. Supongamos que tenemos un sitio multihomed con dos ISPs, como se presenta en la figura a continuación (la solución se puede extender a escenarios con más de dos ISPs fácilmente, pero nos limitaremos a sólo dos proveedores por razones pedagógicas).

En este caso, el sitio multihomed utiliza direcciones PA, por lo que es necesaria la configuración de múltiples prefijos (uno por ISP) en los nodos del sitio multihomed. Esto implica que el sitio es alcanzable a través del ISPA (ISPB) sólo si se utilizan direcciones conteniendo el prefijo PrefA:Site:: (PrefB.:Site::). Así, si ocurre un fallo en el Enlace A (Enlace B), los paquetes dirigidos a una dirección que contenga el prefijo PrefA:Site:: (PrefB.:Site::) no encontrarán un camino disponible hacia su destino.

La solución planteada se basa en la configuración de túneles entre los routers de borde de los ISPs y los routers de salida del sitio multihomed, como se muestra en la figura. De esta forma, en caso de un fallo en uno de los enlaces de acceso del sitio multihomed, los paquetes pueden ser reencaminados a través del túnel. En esta configuración, los paquetes dirigidos a una dirección que contiene el prefijo PrefA:Site:: (PrefB:Site::) siempre son encaminados hasta el ISPA (ISPB), ya que el resto de Internet no tiene conocimiento de que tanto PrefA:Site:: como Pref.B:Site:: se refieren al mismo sitio. Una vez dentro del ISPA (ISPB) el paquete es encaminado a través del enlace A (enlace B) si éste funciona correctamente, y en caso de fallo en dicho enlace, el paquete es encaminado a través del proveedor alternativo ISPB (ISPA) a través del túnel.

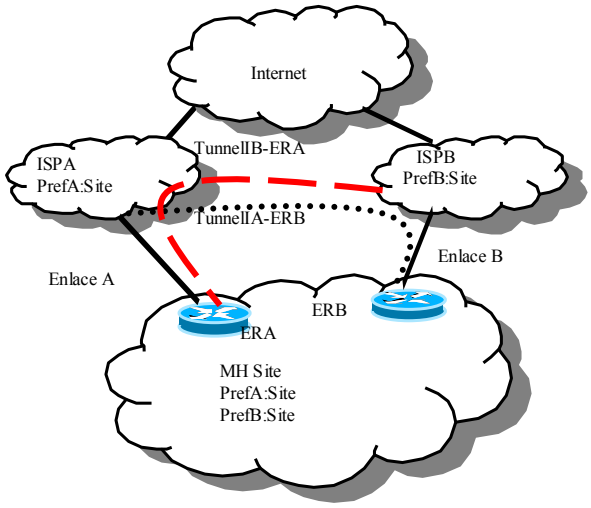

Figura 23: Solución planteada en RFC 2260

Esta solución ofrece tolerancia frente a fallos parciales dentro de los ISPs del sitio multihomed. En caso de un fallo, los paquetes serán encaminados a través del túnel vía el ISP alternativo. De esta forma, tanto las nuevas comunicaciones como las conexiones establecidas pueden tolerar el fallo ocurrido.

Como hemos visto, por su propia naturaleza, esta solución ofrece un dominio de tolerancia a fallos limitado al conjunto de los ISPs del sitio multihomed. Es decir, que si ocurre un fallo fuera

de este dominio, la solución no es capaz de ofrecer caminos alternativos para encaminar los paquetes. En particular, si se produce un fallo en el camino entre el ISPA (ISPB) e Internet, los paquetes dirigidos a PrefA:Site:: no encontrarán un camino hacia el sitio. Análogamente, la solución no ofrece tolerancia a un fallo total de uno de los ISPs del sitio multihomed. Ésta es una limitación de diseño de la solución ya que, como se ha presentado anteriormente, este enfoque ofrece escalabilidad limitando el dominio de difusión de la información de rutas alternativas a los ISPs del sitio multihomed. Consecuentemente, esta limitación es intrínseca a la solución y determina el dominio de aplicabilidad de la misma. Es decir que esta solución será aplicable a sitios cuyo principal requisito sea la protección frente a fallos en la última milla y no frente a fallos más generales. Cabe notar que, dependiendo del escenario considerado, los fallos en la última milla pueden ser mucho más frecuentes que los fallos en otras partes de la topología. En particular, en el caso de los sitios pequeños, que no pueden acceder a servicios provistos de forma fiable con un SLA (Service Level Agreement) y que obtienen redundancia a través de la contratación de múltiples accesos de baja calidad, los fallos en la última milla son mucho más probables que los fallos otras partes de la topología (los accesos de los ISPs son probablemente de mejor calidad, por ejemplo), por lo que la solución presentada colma las necesidades de este tipo de sitio.

Sin embargo, hemos detectado una limitación adicional que no es inherente a la solución pero dificulta la operación de la misma y limita su adopción. Como hemos visto, la solución requiere la configuración de túneles entre el sitio multihomed y sus ISPs. El proceso de configuración de túneles es un proceso manual, lo que lo hace costoso. Es más, debido a que la solución parece idónea para sitios pequeños, es esperable que el número de sitios que la adopten pueda ser alto, por lo que el costo de administración de los túneles puede ser alto también, lo que hace que la solución sea poco atractiva para los ISPs. Una opción para mejorar esta limitación, es el uso de un *gestor de túneles para multihoming*, una herramienta para la automatización del proceso de creación de túneles que permitirá reducir el costo de la solución que se presenta a continuación.

7.5.2 Gestión de túneles

Los túneles son una herramienta muy usada en Internet, en particular en la migración de IPv4 a IPv6. Tanto es así que la dificultad presentada anteriormente ya ha surgido en otros contextos, por lo que se ha definido un modelo para un gestor de túneles [RFC3053]. Si bien los detalles difieren, el modelo general es aplicable por lo que basaremos la herramienta requerida en el mismo.

7.5.2.1 Modelo para el gestor de túneles

El Gestor de túneles tiene dos componentes esenciales, como se ilustra en la figura: el gestor en sí mismo y el servidor de túneles.

- El Gestor de túneles es el elemento con el que los usuarios se comunican para crear y gestionar los túneles. A su vez, el Gestor se comunica con un Servidor de túneles para configurar el túnel requerido por el usuario.
- El Servidor de túneles es el extremo del túnel con el usuario, que es creado y borrado en función de las solicitudes del Gestor de túneles.

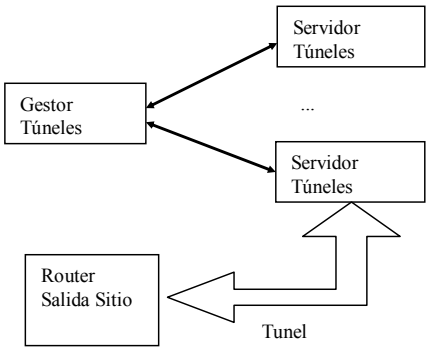

Figura 24: gestor de túneles

7.5.2.2 Gestor de túneles para multihoming

El gestor de túneles para multihoming gestionará túneles IPv6 en IPv6 con el objetivo de mejorar la tolerancia a fallos de sitios multihomed. Los clientes del gestor serán los sitios cliente del ISP que sean multihomed. Para ello, el Gestor cuenta con una interfaz a través de la cual los clientes solicitan la creación, la modificación y el borrado de los túneles. Asumimos que existe una relación comercial entre los ISPs y sus clientes, de forma que los ISPs son capaces de identificar a sus clientes y también conocen el prefijo que le han delegado. También se asume que el ISP ha creado los medios necesarios para habilitar una comunicación segura y autenticada con el cliente a través de la red, como puede ser usuario y contraseña, o certificado digital. El cliente del ISP solicitará al gestor la creación de un túnel. El Gestor podrá aceptar solicitudes a través de diversas interfaces, como por ejemplo una página Web. Para el envío de una solicitud, el cliente

deberá identificarse usando los medios disponibles para ello. Las solicitudes deberán contener la siguiente información:

- Identificación del cliente.
- Dirección IPv6 del extremo del túnel del lado del cliente (el router del sitio multihomed conectado al otro ISP).
- Prefijo IPv6 para el que se creará una ruta alternativa a través del túnel (este prefijo debe estar contenido o ser igual al prefijo delegado por el ISP al cliente).
- Información de validación de la solicitud, usando los medios disponibles, como por ejemplo una firma digital de la solicitud.

Una vez que el Gestor recibe la solicitud, éste deberá realizar las siguientes acciones:

- Verificar la identidad del cliente.
- Verificar la información de validación de la solicitud.
- Verificar que el prefijo incluido en la solicitud coincide o está contenido dentro del prefijo delegado al cliente en cuestión.
- Verificar si se dan las circunstancias técnicas o administrativas para proceder al establecimiento del túnel, si estas están determinadas (número máximo de túneles, número máximo de túneles por cliente, etc.).
- Enviar la orden de configuración al Servidor de túneles conteniendo la información necesaria para realizar la operación, en particular, la dirección IPv6 del extremo del túnel del lado del cliente.
- Informar al cliente de la dirección IPv6 del extremo del ISP del túnel, para que éste lo configure en su router de salida.

7.5.2.3 Servidor de túneles

El Servidor de túneles puede ubicarse en distintos puntos de la red del ISP.

Una opción es que el propio router que brinda acceso al sitio cliente sea el Servidor de túneles que establezca el túnel hacia el sitio, como se ilustra en la figura.

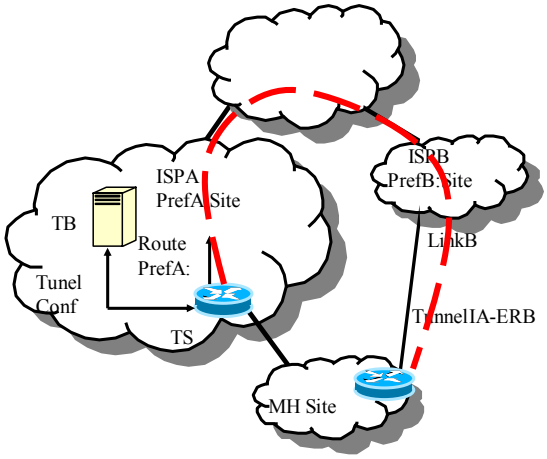

Figura 25: Servidor de túneles en el router de salida

En este caso, bien cuando los paquetes se encaminan a través del enlace directo entre el sitio y el ISP o cuando se encaminan a través del túnel, los paquetes deben llegar previamente al router del ISP que brinda acceso al cliente en cuestión. Esto que simplifica el encaminamiento interno del ISP, ya que la elección del camino a usar es interna a dicho router. Sin embargo, esta solución cuenta con un punto simple de fallos en este router, ya que si éste falla, ninguno de los dos caminos podrá ser utilizado.

Una solución más robusta puede obtenerse ubicando el Servidor de túneles en otro dispositivo distinto, como se ilustra en la próxima figura.

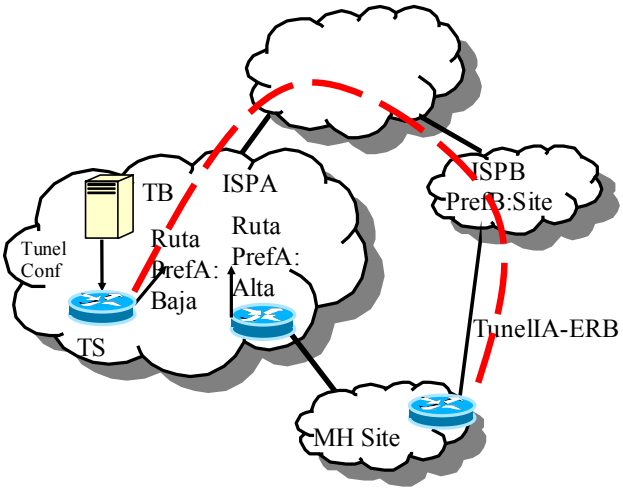

Figura 26: Servidor de túneles en punto arbitrario de la red

En este caso, una ruta hacia el prefijo del cliente a través del camino directo entre el ISP y el cliente es anunciada por el router de salida con una alta prioridad, cuando ésta se encuentra disponible. Adicionalmente, el Servidor de túneles anuncia una ruta hacia el prefijo del cliente a través del túnel con una prioridad más baja. El resultado es que cuando la ruta directa está disponible, los paquetes son encaminados al router de salida y después directamente al sitio final ya que esta es preferida por el sistema interno de rutas debido a su prioridad. Cuando la ruta directa no está disponible, la ruta a través de la misma no se anunciada, por lo que el sistema interno de rutas encaminará los paquetes hacia el Servidor de túneles que los encapsulará y los enviará a través del túnel.

Capítulo 8

Conclusiones

A continuación detallaremos las conclusiones que se pueden extraer del análisis en detalle del problema de la provisión de una solución de multihoming escalable y de la exploración de distintos enfoques. La primera observación que podemos hacer es que la solución actualmente disponible en IPv4 para el soporte de multihoming basada en la inyección de rutas de sitio en el sistema global de rutas impone una carga que crece linealmente con el número de sitios multihomed, lo que limita su escalabilidad y las posibilidades de crecimiento. Esto impone la búsqueda de soluciones alternativas que garanticen la escalabilidad del sistema global de rutas. El uso de direcciones agregables por proveedor (PA) resulta en un crecimiento de las tablas de rutas proporcional al número de proveedores de servicio e independiente del número de sitios multihomed, lo que garantiza la salud del sistema de rutas para el futuro, al menos en lo que a sitios multihomed se refiere. Sin embargo, el uso de direcciones PA en los sitios multihomed impone una serie de desafíos. Estos desafíos son consecuencia de que en una configuración basada en direcciones PA, un sitio multihomed obtiene tantos bloques de dirección como proveedores tiene. Es más, un nodo dentro de un sitio multihomed deberá configurar una dirección de cada uno de los prefijos disponibles para poder intercambiar tráfico a través de todos los proveedores. Dicha configuración plantea las siguientes dificultades:

- Compatibilidad con los filtros de ingreso

- Iniciar una nueva comunicaciones después de un fallo
- Preservar las comunicaciones ante la ocurrencia de un fallo
- Ofrecer capacidades de ingeniería de tráfico

Tras explorar cada uno de los problemas mencionados y analizar diferentes enfoques, se han generado las siguientes recomendaciones:

- La compatibilidad con los filtros de ingreso debe brindarse a través del encaminamiento basado en dirección origen.
- La capacidad de establecer nuevas comunicaciones después de fallos debe proveerse a través de mecanismos de selección de direcciones origen y reintento con distintas direcciones origen.
- Para preservar las comunicaciones establecidas será necesaria la adopción de una nueva capa de identificación dentro de la capa IP que realice una correspondencia entre los identificadores presentados a las capas superiores y los localizadores usados para intercambiar paquetes. Dicha capa de identificación hará uso de un protocolo de multihoming para la creación y la gestión del estado referente a las sesiones de multihoming asociadas a las distintas comunicaciones en curso. La adopción de dicho protocolo requiere el uso de herramientas de seguridad que protejan a los nodos involucrados de las vulnerabilidades resultantes. La herramienta propuesta para ofrecer la protección necesaria son las direcciones generadas criptográficamente, en particular, las Hash Based Addresses y las Cryptographically Generated Addresses.
- Las capacidades de ingeniería de tráfico necesarias son provistas mediante la configuración de registros del DNS y el uso de las tablas de políticas especificadas en el algoritmo de selección de direcciones por defecto.

Como se puede observar, al finalizar el estudio realizado, se han definido un gran número de los puntos necesarios para la provisión del soporte de multihoming para IPv6.

8.1 Contribuciones

La generación de conocimiento es un fenómeno social, es decir, que normalmente no es atribuible a un individuo en particular, sino que es el resultado de una maduración colectiva de la comunidad. Si bien esto es cierto en todos los casos, este fenómeno es especialmente acentuado en el entorno en el que fue realizado el trabajo relacionado con la presente Tesis Doctoral, a saber los grupos de trabajo del Internet Engineering Task Force. En dicho ambiente, el intercambio de ideas es constante y al final del proceso es difícil poder identificar las contribuciones individuales, ya que normalmente el resultado final es un collage de partes de ideas diversas relacionadas.

En cualquier caso, en esta sección intentaremos identificar aquellas partes donde la contribución del autor ha sido, a nuestro juicio, más relevante. A continuación presentaremos entonces las contribuciones principales de la presente Tesis Doctoral.

Probablemente, la contribución más relevante de la presente Tesis Doctoral sean las Hash Based Addresses para proteger la correspondencia entre las direcciones disponibles para un nodo en un sitio multihomed [Bagnulo2004b] [Bagnulo2005c]. Este formato direcciones ofrecen un mecanismo novedoso para la vinculación entre distintos localizadores asociados a un nodo multihomed, que resulta tener bajo coste en comparación con otras soluciones anteriormente propuestas como CGAs.

Adicionalmente, la presente Tesis ha contribuido en el diseño y definición de los demás aspectos de la solución de multihoming [Bagnulo2005b]. En particular, se han realizado aportaciones en:

- La arquitectura resultante de la adopción de una nueva capa de identificación que separe los localizadores de los identificadores usados por capas superiores [Nordmark2005b].
- El protocolo de control necesario para el manejo de sesiones para preservar comunicaciones establecidas a través de fallos [Bagnulo2004c].
- Los mecanismos para proveer compatibilidad con los filtros de ingreso [Bagnulo2004a] [Huitema2004a] [Huitema2004b].
- Los mecanismos de selección de direcciones para el establecimiento de nuevas comunicaciones después de producirse un fallo [Huitema2004a] [Huitema2004c].
- Las herramientas para la provisión de capacidades de ingeniería de tráfico en sitios multihomed [Bagnulo2001b] [Bonaventure2003a] [Bagnulo2005d].

Además de las contribuciones en forma de elementos constitutivos de una arquitectura para la provisión del soporte de multihoming en sitios finales, creemos que los siguientes análisis realizados en la presente Tesis también han contribuido a la comprensión del problema y al diseño de la solución resultante:

- El análisis del espacio de soluciones [Bagnulo2005a] [Bagnulo2002a] [Crocker2003a]
- El análisis de seguridad (Apéndice B de [Nordmark2005a]).
- El análisis realizado sobre la unicidad estadística del espacio de identificadores y su aplicación a los identificadores de interfaz [Bagnulo2002b] [Bagnulo2002c]
- Análisis de las posibilidades y limitaciones para la aplicación de los protocolos de movilidad MIPv6 al problema de multihoming: [Bagnulo2003a], (Sección 4.2 de [Huston2005a]), y alternativas posibles [Bagnulo2003b] [Bagnulo2003c] [Bagnulo2003d].

Queremos destacar que parte de las aportaciones realizadas en esta Tesis [Bagnulo2004b], [Bagnulo2004c], [Normark2005b] han sido adoptadas como documentos de trabajo del grupo de trabajo multi6 del IETF, en particular el modelo de direccionamiento de HBA.

8.2 Trabajos futuros

En esta sección presentaremos posibles líneas futuras de trabajo, que a nuestro parecer son necesarias para la continuación del trabajo en el área de multihoming.

8.2.1 Afinación de los parámetros de la solución

Si bien los elementos básicos de la solución pueden considerarse definidos, es necesario afinar en funcionamiento de la misma en entornos reales En particular, es importante definir los tiempos y heurísticos que gobiernen las decisiones de establecer una sesión de multihoming para dos nodos que están comunicando. Igualmente, es importante definir los elementos que determinarán un cambio de camino, es decir los tiempos involucrados en la detección de fallos ocurridos durante una comunicación. También deben ser investigados los mecanismos para determinar qué caminos entre dos nodos son válidos. Finalmente, otro elemento relevante que debe ser explorado es la interacción de la solución propuesta y los mecanismos de control de congestión, en particular, el control de congestión de TCP.

8.2.2 API para multihoming

La solución propuesta en la presente Tesis Doctoral asume que las aplicaciones no tienen conocimiento alguno de la existencia de los mecanismos para el multihoming ni de las herramientas implementadas para su soporte. Sin embargo, cabe notar que a medida que la solución de multihoming se despliegue, es posible que las aplicaciones obtengan beneficios adicionales si son concientes de la existencia de los mecanismos de multihoming. Esto permitiría a las aplicaciones y capas de transporte tener una mejor interacción con la sub-capa de identificación y optimizar el comportamiento resultante.

En particular, un caso claro donde se podría obtener una mejora es en el tratamiento de las referencias. En el caso en que la aplicación no tenga conciencia de la existencia de múltiples localizadores, la aplicación incluirá solamente el localizador utilizado como identificador en las referencias. Sin embargo, si la aplicación tiene conocimiento del conjunto de localizadores, podría incluirlos en la referencia, de forma que quien recibe la referencia tenga información sobre

todo el conjunto de localizadores, resultando así en una mejora en la tolerancia a fallos del sistema de referencias. Nótese que en el caso en que la referencia solo incluye un localizador, la comunicación fallará si este no se encuentra disponible.

Otro beneficio que podrían obtener las aplicaciones a través de una API para multihoming sería el de poder definir exactamente cuál es el tiempo aceptable de espera para la detección de un fallo. Como hemos visto, este tiempo depende fuertemente de la aplicación en cuestión. De forma que si no hay información de la aplicación en cuestión, se utilizará un valor por defecto. Sin embargo, si la aplicación puede definir el valor, el comportamiento resultante será más ajustado a sus necesidades.

Adicionalmente, una API para multihoming podría permitir que las aplicaciones tengan acceso a la clave pública asociada al nodo, y así disponer de un identificador criptográfico más fuerte y habilitar otros usos, como ser la autenticación/autorización basada en la clave pública.

8.2.3 Transición hacia una separación completa de identificador y localizador

Como hemos visto, la principal razón para utilizar identificadores que también sean localizadores válidos (como son las DGCs) es el soporte de aplicaciones existentes que realizan referencias o tienen largos periodos de inactividad. Sin embargo, a medida que pasa el tiempo, es posible migrar dichas aplicaciones para que hagan uso de la nueva API de multihoming, de forma que sean concientes de los múltiples localizadores asociados a un identificador. De esta forma, la aplicación será capaz de soportar identificadores que no sean localizadores, ya que simultáneamente dispondrán de herramientas que les ofrecerán conocimiento sobre el conjunto de localizadores. En este caso, es posible migrar a identificadores puramente criptográficos como los definidos en HIP.

Por ende, es posible ver la solución de multihoming planteada como un primer paso hacia una solución de largo plazo de separación total de identificadores y localizadores.

8.2.4 Interacción con mecanismos de movilidad

La interacción entre los mecanismos de movilidad y multihoming tiene múltiples aspectos. Por un lado, es necesario evaluar si la solución propuesta para multihoming satisface las necesidades de nodos móviles multihomed y redes móviles multihomed. Después, es necesario estudiar si es posible aplicar la solución desarrollada a estos entornos y qué cambios son necesarios para adaptar la solución propuesta.

Por otra parte, existen propuestas para el soporte de la movilidad basado en el uso de la DGCs [Haddad2005a]. Resulta interesante ver cuan compatibles son estas propuestas con la solución aquí desarrollada para poder explotar posible sinergias entre ambas.

Referencias

[Bagnulo2001a] M., Bagnulo, A., García-Martinez, D., Larrabeiti, A., Azcorra. *Survey on proposed IPv6 multi-homing network level mechanisms.*, Internet Draft (Work in progress), Internet Engineering Task Force, 2001

[Bagnulo2001b] M., Bagnulo, A., García-Martinez, D., Larrabeiti, A., Azcorra, *A QoS-Driven ISP Selection Mechanism for IPv6 Multi-Homed Sites*, PROMS 2001: 6th International Conference on Protocols for Multimedia Systems, pps 23-34. Enschede. Lecture Notes in Computer Science, 2213. Ed. Springer-Verlag. ISBN 3-540-42708-2. Oct 2001.

[Bagnulo2002a] M., Bagnulo, A., García-Martínez, D., Larrabeiti, A., Azcorra, *Primeros pasos hacia multi-homing en IPv6*. Comunicaciones World. Mar. 2002, pags. 14-15. Ed.: IDG Comunications, ISSN 1139-0867

[Bagnulo2002b] M., Bagnulo, I., Soto, A., García-Martinez, A., Azcorra, *Random generation of interface identifiers*, Internet Draft (Work in progress), Internet Engineering Task Force, 2002.

[Bagnulo2002c] M., Bagnulo, I., Soto, IA, García-Martinez, A., Azcorra, *Avoiding DAD for Improving Real-Time Communication in MIPv6 Environments*. PROMS 2002: 7th International Conference on Protocols for Multimedia Systems, pp. 73-79. Lecture Notes in Computer Science, 2515. Ed. Springer-Verlag. ISBN 3-540-00169-7. ISSN 0302-9743. Oct. 2002

[Bagnulo2002d] M., Bagnulo, A., García-Martinez, *Extension Header for Site-Multi-homing Support*, Internet Draft (Work in progress), Internet Engineering Task Force, 2002.

[Bagnulo2003a] M., Bagnulo, A., García-Martínez, I., Soto, *Application of the MIPv6 Protocol to the Multi-Homing Problem.*, Internet Draft (Work in progress), Internet Engineering Task Force, 2003.

[Bagnulo2003b] M., Bagnulo, A., García-Martínez, I., Soto, *Extending BCE Lifetime to Enable the Application of MIPv6 to Multi-Homing.*, Internet Draft (Work in progress), Internet Engineering Task Force 2003.

[Bagnulo2003c] M., Bagnulo, A., García-Martínez, I., Soto, *Preserving MIPv6 communications when the HoA becomes unreachable*, Internet Draft (Work in progress), Internet Engineering Task Force, 2003.

[Bagnulo2003d] M., Bagnulo, A., García-Martinez, I., Soto, A., Azcorra. A., *A MIPv6-based Multi-Homing Solution*. EUNICE 2003: 9th Open European Summer School and IFIP Workshop on Next Generation Networks (co-sponsored by IEEE). Sep 03.

[Bagnulo2003e] M., Bagnulo, A., García-Martinez, I., Soto, A., Azcorra. J., Rodriguez Hervella, *Preserving Established Communications in IPv6 Multi-homed Sites with MEX*, Proceedings of International Workshop on Multimedia Interactive Protocols and Systems MIPS'2003. Nápoles, Págs 54-66. November 2003. Lecture Notes in Computer Science, 2856. Ed. Springer-Verlag. ISBN 3-540-20534-9. ISSN 0302-9743.

[Bagnulo2004a] M., Bagnulo, A., García-Martínez, J., Rodríguez-Hervella, A., Azcorra, *The Case for Source Address Dependent Routing in Multihoming*, 1st International Workshop on QoS Routing. Barcelona. Pps 237-246. October 2004. Lecture Notes in Computer Science, 3266. Ed. Springer-Verlag. ISBN 3-540-20534-9. ISSN 0302-9743.

[Bagnulo2004b] M., Bagnulo, *Hash Based Addresses (HBA)*, Internet Draft (Work in progress), Internet Engineering Task Force, 2004.

[Bagnulo2004c] M., Bagnulo, J., Arkko, *Functional decomposition of the M6 protocol*, Internet Draft (Work in progress), Internet Engineering Task Force, 2004.

[Bagnulo2004d] M., Bagnulo, J., Rodríguez-Hervella, A., García-Martínez, A., Azcorra, *Multi-Homing Tunnel Broker*, IEEE Euromicro 2004, Rennes, Francia. Págs 282-289. September 2004. IEEE Computer Society. ISBN 0-7695-2199-1. ISSN 1089-6503.

[Bagnulo2005a] M., Bagnulo, A., García-Martínez, A., Azcorra, *IPv6 Site Multihoming*, IPv6 and Broadband. Edited by 6LINK with support from the European Union and the IPv6 Cluster. Págs 39-43. ISBN 3-00-013801-3. 2005.

[Bagnulo2005b] M., Bagnulo, A., García Martínez, A., Azcorra, C., de Launois, C., *An Incremental Approach to IPv6 Multihoming*, a publicarse en Elsevier Computer Communication, 2005.

[Bagnulo2005c] M., Bagnulo, A., García Martínez, A., Azcorra, *Efficient Security for IPv6 Multihoming*, a publicarse en ACM Computer Communication Review, 2005.

[Bagnulo2005d] M., Bagnulo, A., García Martínez, C., Bernardos, A., Azcorra, *Traffic Engineering in Multihomed Sites*, a publicarse en Proceeding of 10th IEEE Symposium on Computers and Communications, 2005.

[Beijnum2002a] I. Van Beijnum, *BGP*, Oreilly, 2002.

[Beijnum2003a] I. van Beijnum, *Provider-Internal Aggregation based on Geography to Support Multihoming in IPv6,* Internet Draft, (work in progress) Internet Engineering Task Force, 2003.

[Beijnum2004b] I. van Beijnum, *On Demand Tunneling For Multihoming,* Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[Bejinum2004a] I. van Beijnum, *Crypto Based Host Identifiers,* Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[Bonaventure2003a] O., Bonaventure, P., Trimintzios, G., Pavlou, B., Quoitin, A., Azcorra, M., Bagnulo, P., Fegkas, A., García-Martínez, P., Georgatsos, L., Georgiadis, C., Jacquenet, L., Swinnen, S., Tandel, S., Uhlig, *Internet Traffic Engineering*, Chapter in Quality of Future Internet Services. Cost Action 263 Final Report. Págs 118-180. 2003. Lecture Notes in Computer Science, 2856. Ed. Springer-Verlag. ISBN 3-540-20193-9. ISSN 0302-9743.

[Bradner2003a] S. Bradner, A. Mankin, J. Schiller, *A Framework for Purpose-Built Keys (PBK),* Internet Draft, (work in progress) Internet Engineering Task Force, 2003.

[Chiappa1999a] J. N. Chiappa, *Endpoints and Endpoints names*, Internet Draft (Work in progress), Internet Engineering Task Force, 1999.

[Coene2003a] L. Coene, *Multihoming issues in the Stream Control Transmission Protocol,* Internet Draft, (work in progress) Internet Engineering Task Force, 2003.

[Coene2004a] L. Coene, J. Loughney, *Multihoming: the SCTP solution,* Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[Cohen1978a] D. Cohen, *On Names, Addresses and Routings*, Internet Experiment Note #23, University of Southern California, Information Sciences Institute, Marina Del Rey, Calif., 1978.

[Conta2004a] A. Conta, S. Deering, M. Gupta, Nokia, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification,* Internet Draft, (work in progress) Internet Engineering Task Force, 2005.

[Crocker2003a] D., Crocker, *Choices for Multiaddressing*, Internet Draft (Work in progress), Internet Engineering Task Force, 2003.

[Crocker2003b] D. Crocker, *Multiple Address Service for Transport (MAST): an extended proposal,* Internet Draft, (work in progress) Internet Engineering Task Force, 2003.

[Fernández2004a] D. Fernández, T. de Miguel, F. Galán, *Study and Emulation of IPv6 InternetExchange-Based Addressing Models,* IEEE Communications Magazine, January 2004.

[Francis1994a] P. Francis, *Addressing in Internetwork Protocols*, PhD Thesis University College of London, 1994.

[Haddad2005a] W., Haddad, L., Madour, J., Arkko, F. Dupont, *Applying Cryptographically Generated Addresses to Optimize MIPv6 (CGA-OMIPv6)*, Internet Draft (work in progress) Internet Engineering Task Force, 2005.

[Hain2004a] T. Hain, *An IPv6 Provider-Independent Global Unicast Address Format,* Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[Hinden2005a] R. Hinden, B. Haberman, *Centrally Assigned Unique Local IPv6 Unicast Addresses,* Internet Draft, (work in progress) Internet Engineering Task Force, 2005.

[Huitema1995a] C. Huitema, *Multi-homed TCP,* Internet Draft, (work in progress) Internet Engineering Task Force, 1995.

[Huitema2004a] C., Huitema, D., Draves, M., Bagnulo, *Host-Centric IPv6 Multihoming*, Internet Draft (Work in progress), Internet Engineering Task Force, 2004.

[Huitema2004b] C., Huitema, R., Draves, M., Bagnulo, *Ingress filtering compatibility for IPv6 multihomed sites*, Internet Draft (Work in progress), Internet Engineering Task Force, 2004.

[Huitema2004c] C., Huitema, R., Draves, M., Bagnulo, *Address selection in multihomed environments*, Internet Draft (Work in progress), Internet Engineering Task Force, 2004.

[Huston2001a] G. Huston, *Analyzing the Internet BGP Routing Table*, Internet Protocol Journal, Vol. 4, N. 1, 2001.

[Huston2003a] G. Huston, *IPv4-How long do we have?*, ISP column, 2003.

[Huston2005a] G., Huston, *Architectural Approaches to Multi-Homing for IPv6*, Internet Draft (Work in progress), Internet Engineering Task Force, 2005

[Katz2005a] D. Katz, D. Ward, *Bidirectional Forwarding Detection*, Internet Draft, (work in progress) Internet Engineering Task Force, 2005.

[Kohler2004a] E. Kohler et al., *Datagram Congestion Control Protocol (DCCP)*, Internet Draft (Work in progress), Internet Engineering Task Force, 2004.

[Labovitz2000a] C. Labovitz, A. Ahuja, A. Bose, *Delayed Internet Routing Convergence*, Proceedings of Sigcomm, 2000.

[Lear2004a] E. Lear, R. Droms, *What's In A Name: Thoughts from the NSRG,* Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[Lear2005a] E. Lear, *Things MULTI6 Developers should think about*, Internet Draft, (work in progress) Internet Engineering Task Force, 2005.

[Lindqvist2002a] K. Lindqvist, *Multihoming in IPv6 by multiple announcements of longer prefixes*, Internet Draft, (work in progress) Internet Engineering Task Force, 2002.

[Matsumoto2004a] A. Matsumoto et al., *TLC-FM : Transport Layer Common Framework for Multihoming ,* Internet Draft, (work in progress) Internet Engineering Task Force, 2004

[Menezes1997a] A. Menezes, *Handbook of applied cryptography,* Cap. 8, pag.336, CRC Pres, 1997.

[Montenegro2001a] G. Montenegro, C. Castelluccia, *SUCV Identifiers and Addresses*", Internet Draft, (work in progress) Internet Engineering Task Force, 2001.

[Morley2001a] Z. Morley et al., *Route Flap Damping Exacerbates Internet Routing Convergence*, Proceeding of Sigcomm, 2001.

[Moskowitz2005a] R. Moskowitz et al., *Host Identity Protocol*, Internet Draft (Work in progress), Internet Engineering Task Force, 2005.

[Nikander2004a] P., Nikander, J., Arkko, T., Aura, G., Montenegro, E. Nordmark, *Mobile IP version 6 Route Optimization Security Design Background*, Internet Draft (work in progress), Internet Engineering Task Force, 2004.

[Nikander2005a] P. Nikander, J. Arkko, and T. Henderson, *End-Host Mobility and MultiHoming with Host Identity Protocol*, Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[Nordmark2003a] E. Nordmark, *Strong Identity Multihoming using 128 bit Identifiers (SIM/CBID128)*, Internet Draft, (work in progress) Internet Engineering Task Force, 2003.

[Nordmark2003b] E., Nordmark, *Multihoming using 64-bit Crypto-based IDs*, Internet Draft, (work in progress) Internet Engineering Task Force, 2003.

[Nordmark2004a] E. Nordmark, *Multihoming without IP Identifiers*, Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[Nordmark2005a] E., Nordmark, T., Li, *Threats relating to IPv6 multihoming solutions*, Internet Draft (Work in progress), Internet Engineering Task Force, 2005

[Nordmark2005b] E., Nordmark, M., Bagnulo, *Multihoming L3 Shim Approach*, Internet Draft (Work in progress), Internet Engineering Task Force, 2005.

[Odell1997a] M. O'Dell, *GSE - An Alternate Addressing Architecture for IPv6*, Internet Draft (Work in progress), Internet Engineering Task Force, 1997.

[Py2002a] M. Py, I. van Beijnum, *GAPI: A Geographically Aggregatable Provider Independent Address Space to Support Multihoming in IPv6,* Internet Draft, (work in progress) Internet Engineering Task Force, 2002.

[Py2004a] M. Py, *Multi Homing Aliasing Protocol (MHAP) intro*, Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[RFC1034] P. Mockapetris, *DOMAIN NAMES - CONCEPTS AND FACILITIES,* RFC 1034, Internet Engineering Task Force, 1987.

[RFC1035] P. Mockapetris, *DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION,* RFC 1035, Internet Engineering Task Force, 1987.

[RFC1338] V. Fuller, T. Li, J. Yu, K. Varadhan, *Supernetting: an Address Assignment and Aggregation Strategy*, RFC 1338, Internet Engineering Task Force, 1992.

[RFC1498] J. Saltzer, *On the Naming and Binding of Network Destinations*", RFC 1498, Internet Engineering Task Force, 1993.

[RFC1518] Y. Rekhter, T. Li, *An Architecture for IP Address Allocation with CIDR*, RFC 1518, Internet Engineering Task Force, 1993.

[RFC1519] V. Fuller, T. Li, J. Yu, K. Varadhan, *Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy*, RFC 1519, Internet Engineering Task Force, 1993.

[RFC1631] K., Egevang, P. Francis, *The IP Network Address Translator*, RFC 1631, Internet Engineering Task Force,1994.

[RFC1661] W. Simpson, *The Point-to-Point Protocol (PPP),* RFC 1661, Internet Engineering Task Force, 1994.

[RFC1771] Y. Rekhter, T. Li, *A Border Gateway Protocol 4 (BGP-4)*, RFC 1771, Internet Engineering Task Force, 1995.

[RFC1997] R. Chandra, P. Traina, T. Li, *BGP Communities Attribute* , RFC 1997, Internet Engineering Task Force, 1996.

[RFC2131] R. Droms, *Dynamic Host Configuration Protocol*, RFC 2131, Internet Engineering Task Force, 1997.

[RFC2205]. R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, *Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification* September, RFC 2250, Internet Engineering Task Force, 1997.

[RFC2260] T. Bates, Y. Rekhter, *Scalable Support for Multi-homed Multi-provider Connectivity*, RFC 2260, Internet Engineering Task Force, 1998.

[RFC2401] S. Kent, R. Atkinson, *Security Architecture for the Internet Protocol*, RFC 2401, Internet Engineering Task Force, 1998.

[RFC2460] S. Deering, R. Hinden, *2460 Internet Protocol, Version 6 (IPv6) Specification,* RFC 2460, Internet Engineering Task Force, 1998.

[RFC2461] T. Narten, W. Simpson, E. Nordmark, *Neighbor Discovery for IP Version 6 (IPv6),* RFC 2461, Internet Engineering Task Force, 1998

[RFC2462] S. Thomson, T. Narten, *IPv6 Stateless Address Autoconfiguration*, RFC 2462, Internet Engineering Task Force, 1998.

[RFC2740] R. Coltun, D. Ferguson, J. Moy, *OSPF for IPv6,* RFC 2740, Internet Engineering Task Force,1999.

[RFC2782] A. Gulbrandsen, P. Vixie, L. Esibov, *A DNS RR for specifying the location of services (DNS SRV),* RFC 2782, Internet Engineering Task Force, 2000.

[RFC2827] P. Ferguson, D. Senie, *Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing*, RFC 2827, Internet Engineering Task Force, 2000.

[RFC2894] M. Crawford, *Router Renumbering for IPv6*, RFC 2894, Internet Engineering Task Force, 2000.

[RFC2960] R. Stewart et al., *Stream Control Transmission Protocol*, RFC 2960, Internet Engineering Task Force, 2000.

[RFC2993] T. Hain, *Architectural Implications of NAT,* RFC 2993, Internet Engineering Task Force, 2000.

[RFC3041] T., Narten, R. Draves, *Privacy Extensions for Stateless Address Autoconfiguration in IPv6*, RFC 3041, Internet Engineering Task Force, 2001.

[RFC3053] A., Durand , P., Fasano , I., Guardini, D. Lento , *IPv6 Tunnel Broker*, RFC 3053, Internet Engineering Task Force, 2001.

[RFC3177] IAB, IESG, *IAB/IESG Recommendations on IPv6 Address Allocations to Sites*, RFC 3177, Internet Engineering Task Force, 2001.

[RFC3178] J. Hagino, H. Snyder, *IPv6 Multihoming Support at Site Exit Routers*, RFC 3178, Internet Engineering Task Force, 2001.

[RFC3194] A. Durand, C. Huitema, *The Host-Density Ratio for Address Assignment Efficiency: An update on the H ratio*, RFC 3194, Internet Engineering Task Force, 2001.

[RFC3280] R., Housley, W., Polk, W., Ford, D. Solo, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, RFC 3280, Internet Engineering Task Force, 2002.

[RFC3315] R. Droms et al., *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*, RFC 3315, Internet Engineering Task Force, 2003.

[RFC3447] J. Jonson, B. Kaliski, *Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1*, RFC 3447, Internet Engineering Task Force, 2003.

[RFC3484] R. Draves, *Default Address Selection for Internet Protocol version 6 (IPv6)*, RFC 3484, Internet Engineering Task Force, 2003.

[RFC3513] R., Hinden, S. Deering, *Internet Protocol Version 6 (IPv6) Addressing Architecture*", RFC 3513, Internet Engineering Task Force, 2003.

[RFC3582] J. Abley, B. Black, V. Gill, *Goals for IPv6 Site-Multihoming Architectures*, RFC 3582, Internet Engineering Task Force, 2003.

[RFC3633] O. Troan, R. Droms, *IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6*, RFC 3633, Internet Engineering Task Force, 2003.

[RFC3704] F. Baker, P. Savola., *Ingress Filtering for Multihomed Networks*, RFC 3704, Internet Engineering Task Force, 2004.

[RFC3775] D. Johnson, C. Perkins, J. Arkko, *Mobility Support in IPv6,* RFC 3775, Internet Engineering Task Force, 2004.

[RFC3833] D. Atkins, R. Austein, *Threat Analysis of the Domain Name System (DNS),* RFC 3833, Internet Engineering Task Force, 2004.

[RFC3971] J., Arkko, J., Kempf, B., Sommerfeld, B., Zill, P. Nikander, *SEcure Neighbor Discovery (SEND)*, RFC 3972, Internet Engineering Task Force, 2005.

[RFC3972] T., Aura, *Cryptographically Generated Addresses (CGA)*, RFC 3972, Internet Engineering Task Force, 2005.

[RFC4033] R. Arends, R. Austein, M. Larson, D. Massey, S. Rose, *DNS Security Introduction and Requirements*, RFC4033, Internet Engineering Task force, 2005

[Savola2004a] P. Savola, *Multihoming Using IPv6 Addressing Derived from AS Numbers,* Internet Draft, (work in progress) Internet Engineering Task Force, 2004.

[Shoch1978] John F. Shoch, *Inter-Network Naming, Addressing and Routing*, Internet Experiment Note #19, University of Southern California, Information Sciences Institute, Marina Del Rey, Calif., 1978.

[Teraoka2003a] F. Teraoka et al., *LIN6: A Solution to Mobility and Multi-Homing in IPv6*, Internet Draft (Work in progress), Internet Engineering Task Force, 2003.

[Wiley2003a] B. Wiley, *Distributed hash tables, Part I,* Linux Journal, Volume 2003 , Issue 114.

[Ylitalo2004a] J. Ylitalo, V. Torvinen, and E. Nordmark, *Weak Identifier Multihoming Protocol (WIMP)*, Internet Draft, (work in progress) Internet Engineering Task Force, 2004.