CAPITULO III

MECANISMOS DE TRANSICIÓN DE IPV4 A IPV6

3.1 Introducción

3.2 DSTM

3.3 Túneles

3.4 Traductores

3.1 Introducción

IPv6 e IPv4 coexistirán durante muchos años. Una amplia gama de técnicas se han definido que permiten la coexistencia y proporciona una transición fácil. Hay tres categorías principales que a continuación indicamos:

- Técnicas Dual-stack. Permiten a IPv4 y a IPv6 coexistir en los mismos dispositivos y redes.
- Técnicas de Tunneling. Permiten el transporte de tráfico de IPv6 a través de la infraestructura de IPv4 existente.
- Técnicas de traducción. Permiten comunicar solamente nodos IPv6 con nodos IPv4.

Estas técnicas pueden y probablemente se usarán combinandolas entre si. La migración a IPv6 puede hacerse paso a paso, empezando con un solo host o subnet. Se puede igualmente emigrar su red corporativa, o partes de la misma, mientras su ISP todavía trabaja sólo con IPv4. O su ISP puede actualizar a IPv6 mientras su red corporativa todavía ejecuta IPv4. Este capítulo describe las principales técnicas disponibles y factibles de implementar hoy en día. Conforme IPv6 siga creciendo en nuestras redes, se definirán nuevas herramientas y mecanismos para que la transición sea fácil de realizar. *[LIB012]*

A continuación vamos a describir brevemente cada técnica para luego pasar a analizar los métodos más importantes y que se usan con más frecuencia en el proceso de migración.

Técnicas Dual-stack

Un nodo dual-stack tiene el apoyo completo de ambas versiones protocolares. Este tipo de nodo es a menudo llamado un nodo IPv6/IPv4. En la comunicación con un nodo IPv6, este se comporta como un nodo IPv6 único, y en la comunicación con un nodo IPv4, este se comporta como un nodo IPv4 único. Las aplicaciones tienen un interruptor de configuración probablemente para habilitar o desactivar una de las pilas. Así que este tipo

de nodo puede tener tres modos de funcionamiento. Cuando la pila de IPv4 se habilita y la pila de IPv6 es desactivada, el nodo se comporta como un nodo IPv4 único. Cuando la pila de IPv6 se habilita y la pila de IPv4 es desactivada, se comporta como un nodo IPv6 único. Cuando se habilitan las pilas tanto en IPv4 y de IPv6, el nodo puede usar ambos protocolos. Un nodo IPv6/IPv4 tiene una dirección por lo menos para cada versión protocolar.

La desventaja de esta técnica es que se debe realizar una actualización de software de red para ejecutar las dos pilas del protocolo separadas. Esto significa que todas las tablas (por ejemplo, las tablas de ruteo) se guarda simultáneamente, mientras los protocolos de ruteo se configuran para ambos protocolos. Para la administración de red, se tiene comandos separados dependiendo del protocolo (por ejemplo, ping.exe para IPv4 y ping6.exe para IPv6) y esto consume más memoria y poder del CPU.

Técnicas de Tunneling

Los mecanismos de Tunneling pueden usarse para desplegar una infraestructura IPv6 mientras la infraestructura IPv4 todavía sea la base. El Tunneling puede usarse para llevar trafico IPv6 encapsulándolo en los paquetes IPv4 y además sobre una infraestructura de ruteo IPv4. Por ejemplo, si un proveedor todavía tiene una infraestructura IPv4 única, el tunneling permite tener una red IPv6 corporativa y un tunel a través de la red IPv4 de su ISP para localizar otro host o red IPv6. *[LIB013]*

Técnicas de traducción (NAT)

Estas técnicas ofrecen mecanismos de transición adicionales a dual-stack y tunneling. El objetivo de estas técnicas es mantener la asignación de rutas transparente a los nodos en las redes IPv6 para comunicarse con los nodos en las redes IPv4 y viceversa. El gateway NAT usa direcciones IPv4 globales unicas y las asocia a las direcciones IPv6. No es necesario realizar ningún cambio a los nodos finales

3.2 DSTM

El DSTM (Dual Stack Transition Mechanism) se compone de dos métodos en particular: AIIH (Assignment of IPv4 global addresses to IPv6 hosts) y DTI (Dynamic Tunneling Interface). AIIH es un método que permite asignar temporalmente direcciones IPv4 a hosts Dual Stack dentro de una red IPv6. ***DTI*** es una interfase diseñada para encapsular paquetes IPv4 dentro de paquetes IPv6. La unión de ambos métodos da como resultado el mecanismo DSTM, el cual tiene como objetivo que un host IPv6 obtenga una dirección IPv4 para establecer comunicación con hosts que manejen exclusivamente direcciones IPv4. DSTM permite también ejecutar aplicaciones IPv4 sin modificación alguna, y solo se puede aplicar dentro de una red IPv6. A continuación se muestra un esquema del mecanismo DSTM:

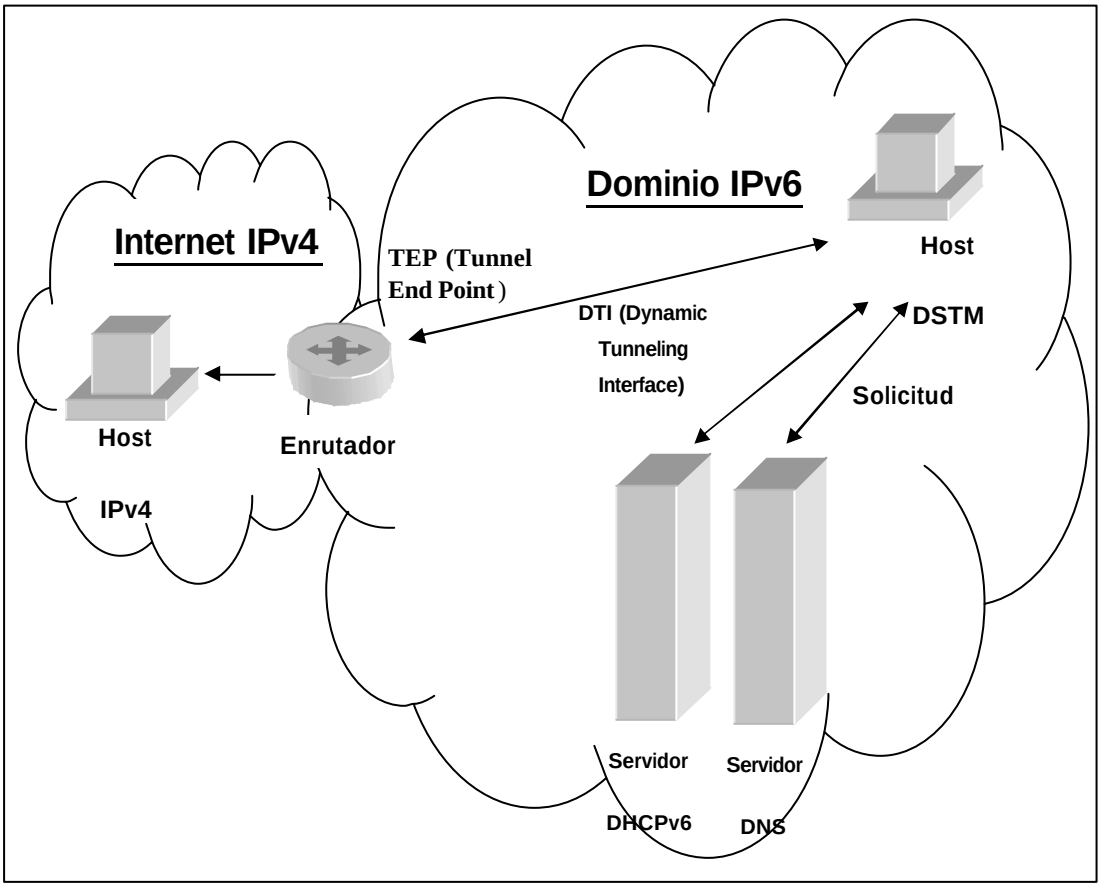


Figura 3.1 Entorno DSTM

El entorno DSTM trabaja solamente con hosts Dual Stack. Se necesita un servidor encargado de asignar temporalmente direcciones IPv4 a los hosts, el cual generalmente utiliza DHCPv6, ya que DHCPv4 no puede ser utilizado dentro de una red IPv6. También se necesita un servidor para resolución de DNS y un enrutador frontera con soporte Dual Stack para comunicar el dominio IPv6 a un dominio exterior o al Internet [7].

3.2.1 Funcionamiento de DSTM

El host DSTM hace una solicitud de dirección IPv4 al servidor de direcciones, para establecer comunicación con un host fuera del dominio. Para establecer comunicación con un host dentro del mismo dominio, no es necesario solicitar una dirección IPv4.

El servidor de direcciones le asigna una dirección IPv4 temporalmente al host DSTM. El tiempo de vida de esa asignación debe ser indicado en la respuesta del servidor al host, así como la dirección IPv6 del ***TEP*** (Tunnel End Point). Si un host requiere más tiempo, la dirección IPv4 tendrá que completar el tiempo y realizar una nueva solicitud al servidor de direcciones. El servidor de direcciones también se encargará de mapear las direcciones IPv4 asignadas a la dirección IPv6 correspondiente, es decir, relacionar ambas direcciones. Como éstas direcciones son asignadas temporalmente, se podrán guardar en una memoria cache. Como una extensión del proceso de asignación de direcciones, el servidor puede asignar un rango de puertos a utilizar por el host. Esto permite que una sola dirección IPv4 pueda ser utilizada por varios hosts al mismo tiempo, evitando que los puertos se traslapen.
Con la dirección IPv6 del TEP proporcionada por el servidor de direcciones, el host se encargará de configurar una interfase DTI hacia el TEP, encapsulando los paquetes IPv4 dentro de paquetes IPv6. Si la interfase no ha sido configurada, es decir, que no tiene asignada una dirección IPv4, el proceso deberá detenerse hasta obtener una dirección IPv4 del servidor de direcciones. Todo el tráfico IPv4 puede ser dirigido a esta interfase por medio de una entrada en la tabla de enrutamiento del host. Una vez que la dirección IPv4 ha sido asignada, es utilizada como dirección fuente para todos los paquetes que sean enviados desde esa interfase. *[WWW017]*
Por último, el host manda los paquetes encapsulados hacia el TEP, generalmente el enrutador frontera. Este se encarga de decapsular los paquetes y reenviarlos hacia la red exterior o el Internet, de modo que lleguen al host solicitado por el host Dual Stack.

3.2.2 Comunicación Bidireccional

El mecanismo DSTM es bidireccional, es decir, permite que un host Dual Stack dentro de un dominio IPv6 se comunique con hosts exclusivamente IPv4, o en caso contrario, que un host exclusivamente IPv4 se pueda comunicar con un host Dual Stack dentro de un dominio IPv6.

En el primer caso, el host Dual Stack solicitará una resolución de dirección de tipo AAAA para el host con el que quiere establecer comunicación. Debido a que el host no es IPv6, el servidor DNS le devolverá un error de resolución. Es entonces cuando el host Dual Stack solicitará una dirección IPv4 para poder establecer la comunicación.

En el segundo caso, cuando un host exclusivamente IPv4 desea establecer comunicación con un host Dual Stack dentro de un dominio IPv6, el host IPv4 solicita una resolución de dirección de tipo A para el host Dual Stack al servidor DNS dentro de su red IPv4. El servidor DNS se comunica con el enrutador DSTM, el cual solicita al servidor de direcciones del dominio IPv6 que le asigne temporalmente una dirección IPv4 al host solicitado, para poder establecer la comunicación.

3.3 Túneles

3.3.1 6to4

Este método es también conocido como "Connection of IPv6 Domains via IPv4 Clouds" (conexión de dominios IPv6 por medio de nubes IPv4). Esencialmente, este método permite a sitios o hosts IPv6 comunicarse entre ellos a través de una red IPv4, sin necesidad de configuración manual de túneles, y permite que dichos sitios o hosts se comuniquen con el Internet IPv6 por medio de enrutadores 6to4 Relay. *[WWW020]*

Este método debe ser temporal, y se utilizará mientras se obtenga una conexión IPv6 nativa, es decir, mientras se lleva a cabo la transición de IPv4 a IPv6. No fue diseñado como una solución permanente.

El esquema de este método se muestra en la figura siguiente:

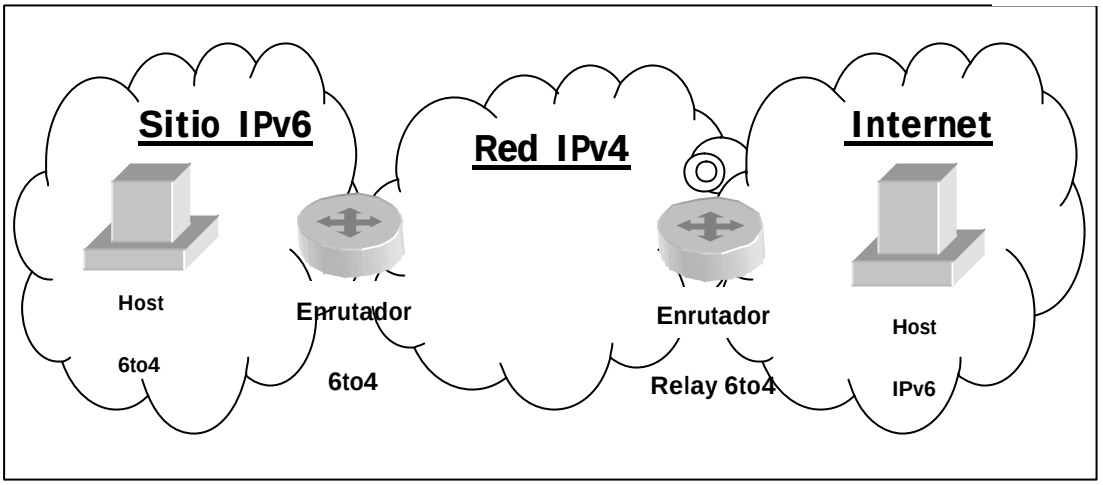


Figura 3.2 Entorno 6to4

Dentro de este entorno encontramos principalmente 3 elementos:

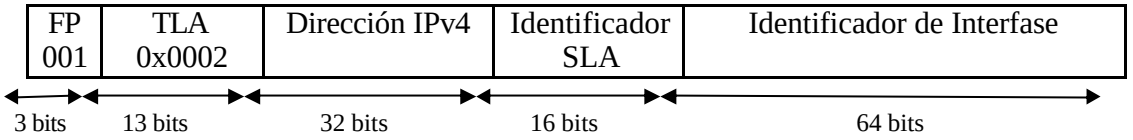
- Un host 6to4 es un host IPv6 que tiene configurada al menos una dirección de tipo 6to4. Estos hosts no requieren configuración manual, comúnmente cuentan con un mecanismo de autoconfiguración.

- Un enrutador 6to4 es un enrutador Dual Stack que soporta el uso de túneles 6to4, el cual sirve para intercambiar paquetes de tipo 6to4 entre enrutadores del mismo tipo y sitios o hosts IPv6. Estos enrutadores requieren configuración manual adicional, ya que son los encargados de encapsular y decapsular los paquetes.

- Un enrutador 6to4 Relay se puede definir como un enrutador 6to4 configurado para soportar el enrutamiento de tránsito entre direcciones 6to4 y direcciones IPv6 nativas. Este enrutador debe tener al menos una interfase 6to4 y una interfase IPv6 nativa, para poder establecer comunicación entre dominios IPv4 e IPv6.

3.3.1.1 Dirección 6to4

Una dirección de tipo *6to4*, esta conformada por distintas partes como se muestra en la siguiente figura:



Este tipo de dirección utiliza el prefijo 001, el cual identifica a las direcciones *Unicast* Globales Agregables, seguido por un identificador ***TLA*** de 13 bits asignado por IANA, cuyo valor es 0x0002. Después le sigue la dirección IPv4 del sitio, así como un identificador SLA yel identificador de interfase.

Todo esto se puede expresar 2002:DirecciónIPv4::/48. La forma en que se convierte la dirección IPv4 al formato para estas direcciones se muestra a continuación:

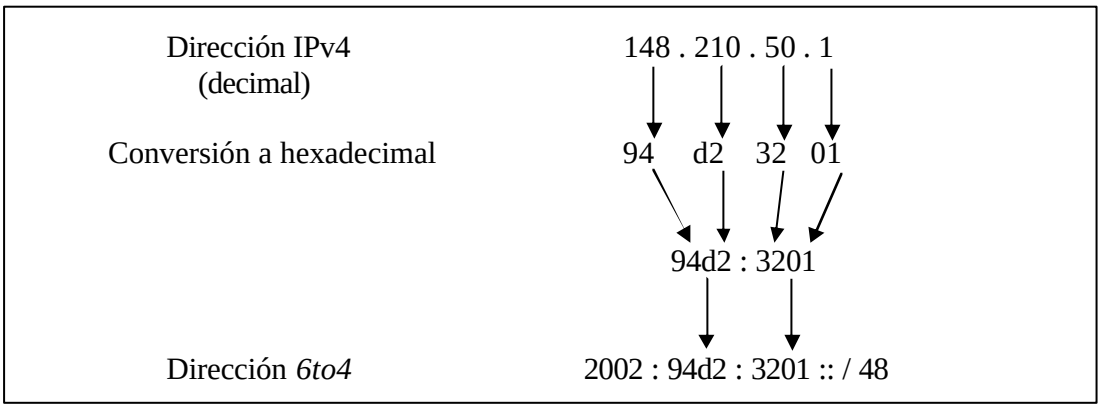


Figura 3.3 Conversión de dirección IPv4 a dirección 6to4

3.3.1.2 Selección de dirección

En caso de que un host tenga una dirección 6to4, y el host con el que quiere establecer comunicación tenga una dirección 6to4 y una dirección IPv6 nativa, es recomendable que ambos hosts establezcan la comunicación utilizando 6to4. En caso de que ambos hosts tengan direcciones 6to4 y direcciones IPv6 nativas, se puede establecer la comunicación siempre y cuando ambos hosts utilicen el mismo tipo de direcciones, aunque es recomendable que la comunicación se realice por medio de direcciones IPv6 nativas. *[WWW010]*

3.3.1.3 Encapsulación 6to4

En el método de 6to4 se utiliza la encapsulación de paquetes IPv6 dentro de paquetes IPv4. El campo "Protocolo" de la cabecera IPv4 debe ser igual a 41, que es el número asignado para este tipo de encapsulación o túneles. Las direcciones de destino y origen, ubicadas en la cabecera IPv4, pueden ser las mismas direcciones del campo que contiene la dirección IPv4 en el prefijo formado para las direcciones 6to4.

3.3.1.4 Tipos de comunicación

Los enrutadores IPv6 dentro de un mismo sitio publican prefijos 2002:DirecciónIPv4:IdentificadorSLA::/64 para permitirle a los hosts crear direcciones 6to4 autoconfiguradas. Los hosts o subredes individuales se configuran automáticamente con una ruta de 64 bits de una subred para intercambio directo entre hosts vecinos. Cualquier paquete IPv6 que no contenga un prefijo de 64 bits similar al de alguna de las subredes del sitio, será enviado al enrutador 6to4 colocado en la frontera del sitio.

Con el método de 6to4 se pueden efectuar varios tipos de comunicación. A continuación se muestra un entorno para ejemplificar los diferentes tipos de comunicación:

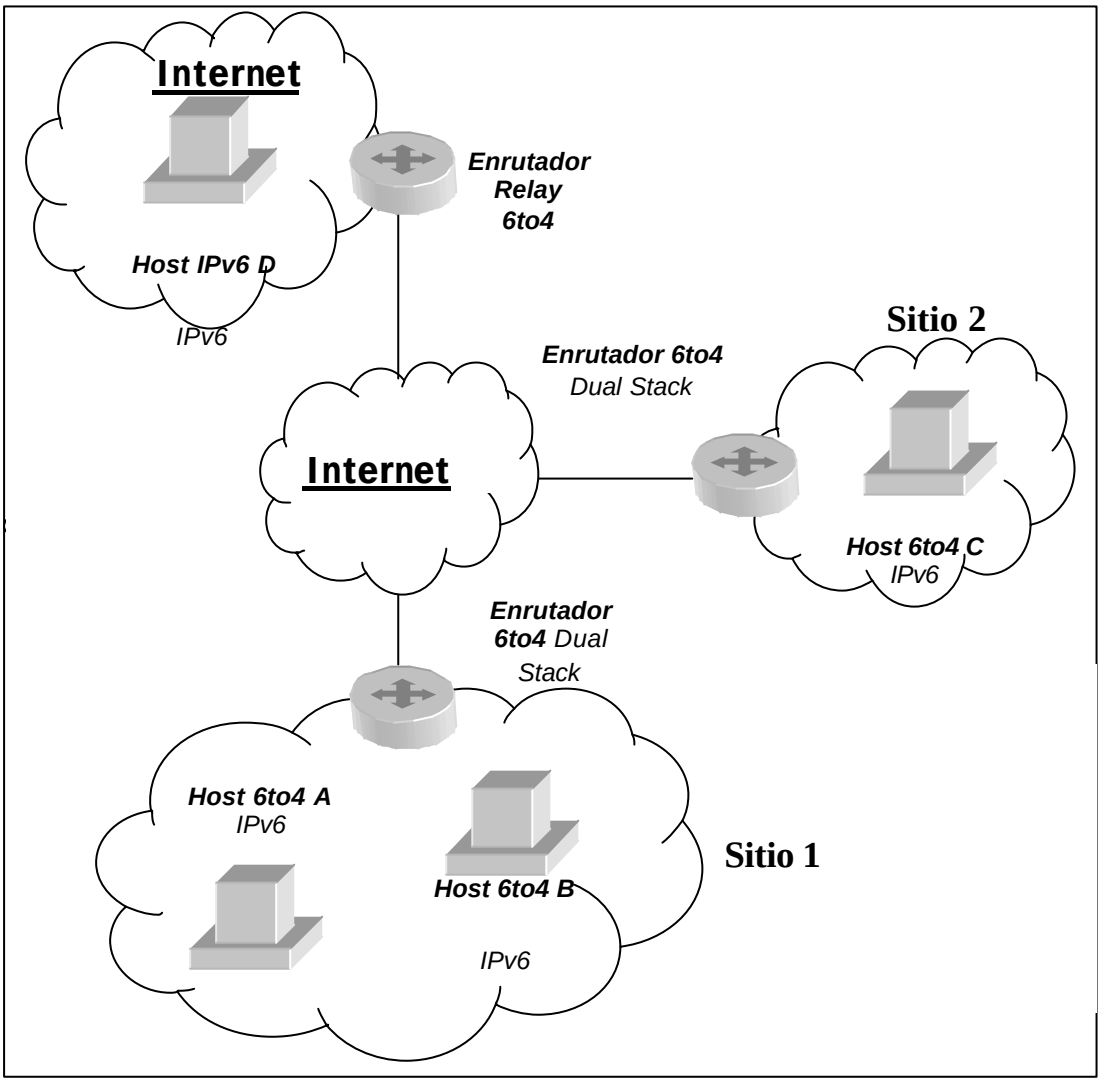


Figura 3.4 Comunicación 6to4

Comunicación del Host A al Host B. Un host 6to4 puede establecer comunicación con un host 6to4 dentro de su mismo sitio. El host origen (Host A) envía el paquete al host solicitado (Host B) utilizando la infraestructura del sitio local (Sitio 1).

Comunicación del Host A al Host C. Un host 6to4 puede establecer comunicación con hosts 6to4 en otros sitios. Lo primero que hace el host origen (Host A) es mandar el paquete al enrutador 6to4 local (Sitio 1). Después este se encarga de hacerlo llegar al enrutador 6to4 del sitio solicitado (Sitio 2) por medio de la creación de túneles en la infraestructura IPv4. Por último, el enrutador en el sitio destino (Sitio 2) se encarga de decapsular el paquete y entregarlo al host solicitado (Host C) utilizando la infraestructura IPv6 del sitio.

Comunicación del Host A al Host D. Un host 6to4 puede establecer comunicación con hosts en el Internet IPv6. Lo primero que hace el host origen (Host A) es mandar el paquete al enrutador 6to4 local (Sitio 1). Después este se encarga de hacerlo llegar a un enrutador 6to4 Relay, el cual tenga acceso a ambos entornos, Internet IPv4 e Internet IPv6. Por último, el enrutador 6to4 Relay se encarga de decapsular el paquete y entregarlo al host solicitado (Host D) utilizando la infraestructura IPv6 del sitio.

3.3.2 6over4

Este método permite que hosts IPv6 que se encuentren dentro de un dominio IPv4, y que no están conectados directamente a un enrutador Dual Stack, establezcan una comunicación con otros hosts IPv6 dentro del mismo dominio mediante la encapsulación de paquetes IPv6 dentro de paquetes IPv4. Si alguno de estos hosts IPv6 desea establecer comunicación con algún host ubicado en otro dominio IPv6, será necesario que exista de por medio un enrutador Dual Stack. *[LIB014]*

A este método se le conoce formalmente como "IPv6 over IPv4", pero comúnmente se le conoce como "6over4" o "Virtual Ethernet" (se entiende como una capa de ***enlace*** virtual). El esquema de este método se muestra en la figura siguiente:

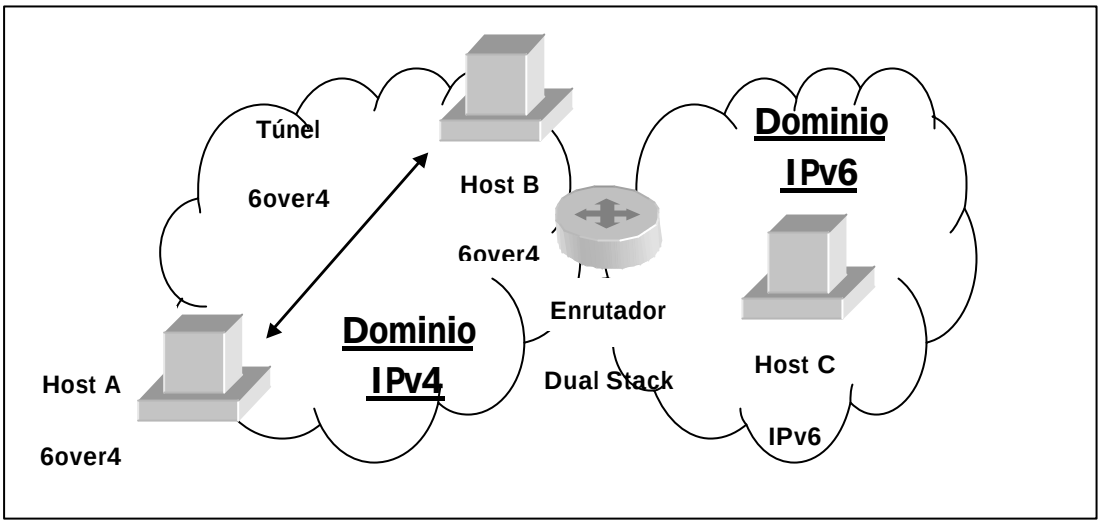


Figura 3.5 Entorno 6over4

3.3.2.1 Dirección 6over4

El dominio IPv4 debe ser multicast para que se puedan llevar a cabo algunos mensajes o procesos de descubrimiento de nodos vecinos (***Neighbor Discovery***). Para la traducción de direcciones IPv6 multicast a direcciones IPv4 multicast se ha definido el siguiente patrón:

239.192.[segundo byte más a la derecha de la dir. IPv6].[último byte de la dir. IPv6]

A continuación se muestran algunos ejemplos de direcciones IPv6 multicast traducidas a direcciones IPv4 multicast:

- FF02::1 (dirección *multicast* de enlace local con alcance a todos los *hosts*) se cambia por la dirección 239.192.0.1
- FF02::2 (dirección *multicast* de enlace local con alcance a todos los enrutadores) se cambia por la dirección 239.192.0.2
- *FF02::1:FF45:8C54 (dirección* multicast *de un nodo solicitado) se cambia por la dirección 239.192.140.84*

Cuando se utiliza 6over4, el dominio IPv4 hace uso de mensajes ***IGMP*** (Internet Group Management Protocol) para informar a los enrutadores IPv4 locales del tráfico multicast que esta siendo enviado. Los hosts que soportan 6over4 también registran direcciones MAC multicast adicionales para sus adaptadores de red, y estas son correspondientes a las direcciones IPv4 multicast. A continuación se muestran algunas direcciones MAC correspondientes a direcciones IPv4 multicast para un adaptador de tipo Ethernet:

- *La dirección MAC* multicast *correspondiente a la dirección 239.192.0.1 es 01-00-5E-40-00-01.*
- *La dirección MAC* multicast *correspondiente a la dirección 239.192.0.2 es 01-00-5E-40-00-02.*
- *La dirección MAC* multicast *correspondiente a la dirección 239.192.140.84 es 01-00-5E-40-8C-54* [5].

3.3.2.2 MTU

La unidad máxima de transmisión o MTU (Maximum Transmission Unit) por default de los paquetes IPv6 en un dominio IPv4 deberá ser de 1480 octetos (la MTU máxima normal es de 1500 octetos, pero se deben reservar 20 octetos para la cabecera IPv4). Este tamaño puede variar cuando se especifique alguna MTU en un Router Advertisement o por alguna configuración manual. En el caso de que la MTU sea muy grande para una red intermedia, esto asegurará una fragmentación, por lo que en este caso se debe asegurar que el bit de "DF" de la cabecera IPv4 no este activo. *[LIB013]*

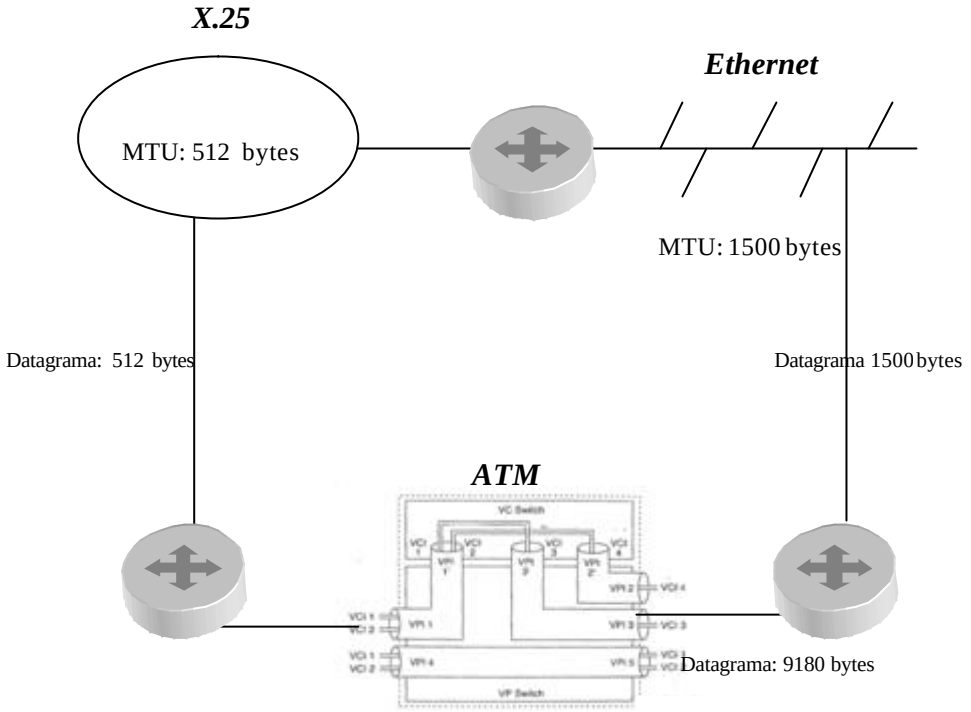


Figura3.6 Diferentes topologías de red con distintos MTU

3.3.2.3 Encapsulación 6over4

Los paquetes IPv6 serán transmitidos dentro de paquetes IPv4 con el campo de "Protocolo" igual a 41, que es el valor predefinido para túneles o encapsulación de paquetes IPv6 en paquetes IPv4. La cabecera IPv4 contiene las direcciones IPv4 de origen y destino. El cuerpo del paquete IPv4 contiene la cabecera IPv6, así como su carga. En caso de que el paquete IPv4 contenga opciones, estas deberán ser rellenadas hasta terminar todo el bloque de 32 bits, y que la cabecera IPv6 comience en un nuevo bloque de 32 bits.

3.3.3 Tunnel Broker

Desde el comienzo de IPv6 y durante su crecimiento, se ha tenido la necesidad de utilizar la infraestructura de red existente, es decir, la infraestructura IPv4. La mayoría del 6bone esta conectado por una variedad de túneles de distintos tipos, cada uno de ellos con un objetivo en especial, pero al mismo tiempo con ciertos problemas o limitaciones.

3.3.3.1 Tipos de túneles

Los siguientes son los principales tipos de túneles:

- Túneles automáticos con direcciones IPv4 compatibles. Útiles para conectar enrutadores o hosts que se encuentran aislados, pero trae consigo el problema de la escasez de direcciones IPv4 que prácticamente es el problema a solucionar. Además, las tablas de enrutamiento IPv4 seguirán creciendo cada vez más, y lo peor de todo es que estas direcciones se tendrán que almacenar también en las tablas de enrutamiento IPv6, lo cual creará un grande problema con respecto al tamaño de dichas tablas.

- Túneles de tipo 6to4. Permiten a dominios IPv6 aislados que cuenten con una conexión directa a una red IPv4 o al Internet, poder establecer comunicación con otros dominios IPv6 con una mínima configuración manual. Este tipo de túneles se utiliza comúnmente en redes aisladas o privadas.

- Túneles de tipo 6over4. Es un mecanismo a nivel de sitio que utiliza un dominio IPv4 Multicast, como una capa de enlace de datos virtual. Sin embargo necesita un ***enrutador*** extra si desea establecer comunicación con un dominio IPv6 externo.

Los túneles manualmente configurados han sido de gran ayuda hasta la fecha, pero requieren una estricta supervisión y mantenimiento de parte de los administradores de las redes, por lo que se pensó en crear un método que creara túneles configurados de una manera automática. Es aquí donde nació el concepto de Tunnel Broker (TB).

3.3.3.2 Descripción de TB

La idea principal de este método es tener servidores dedicados, llamados TB's, que se encarguen de configurar túneles de una manera automática en respuesta a requisiciones hechas por los usuarios de este servicio, y de esta manera aumentar el número de hosts que

se encuentran actualmente conectados a una red IPv6. Se espera que en un futuro existan varios tipos de TB's, de manera que el usuario pueda seleccionar de una lista el que mejor se acomode a sus necesidades, por ejemplo el más cercano, el más barato, etc.

El método de TB permite a hosts Dual Stack que cuenten con una conexión a una infraestructura IPv4, crear túneles automáticamente para poder establecer una comunicación con dominios IPv6. *[WWW006]*

A continuación se muestra el entorno de un TB de una manera gráfica:

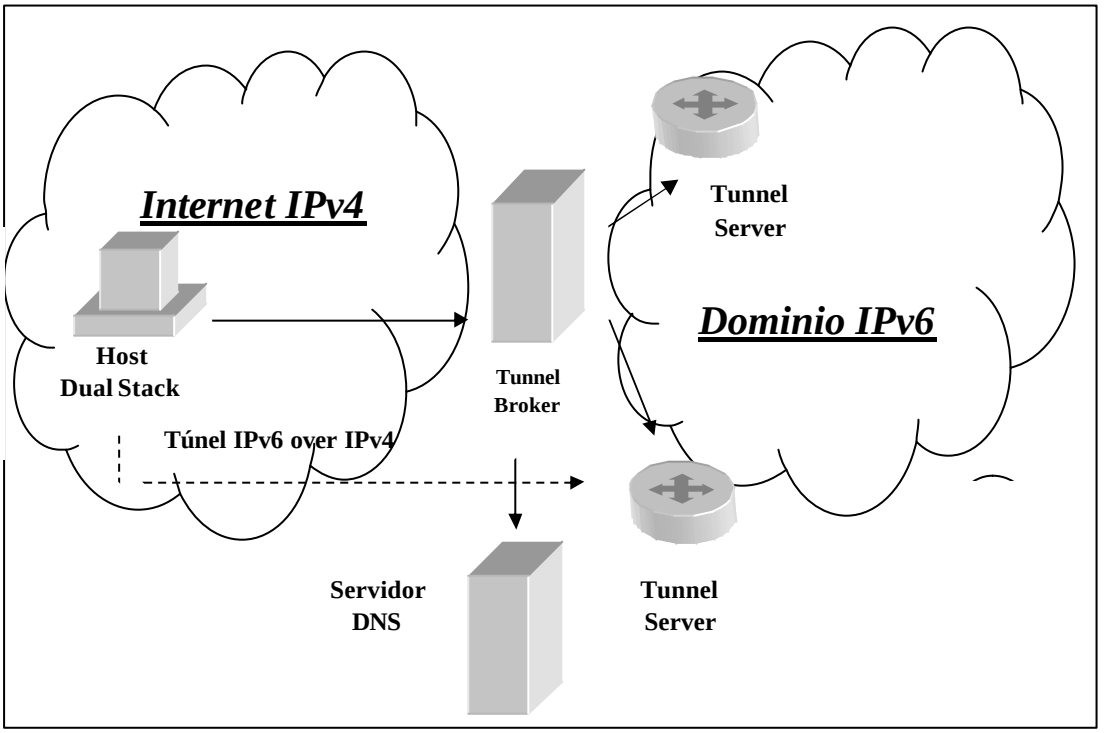


Figura 3.7 Entorno Tunnel Broker

El TB es en ocasiones una ***interfase*** de tipo Web, donde el usuario se registra para obtener un túnel. En esta interfase los túneles pueden ser creados, modificados o eliminados, de acuerdo a las necesidades del usuario. El TB puede repartir la carga entre varios Tunnel Servers (TS), e indicarles la configuración del túnel en cuestión. El TB también se encarga de dar de alta la dirección IPv6 del usuario, así como su nombre en el DNS. El TB debe

tener una dirección IPv4 a la cual pueda comunicarse el usuario. También puede tener una dirección IPv6, pero ésta es opcional. La comunicación entre el TB y el usuario se puede realizar por medio de IPv4 o IPv6.

Un TS es un enrutador de tipo Dual Stack que se encarga de crear, modificar o eliminar túneles, basándose en las órdenes recibidas por el TB. El TS también puede guardar una estadística del uso de los túneles que administra y enviársela al TB para la toma de decisiones sobre los distintos túneles. Un TS debe tener una conexión directa a un dominio IPv6 o al Internet IPv6.

3.3.3.3 Funcionamiento de TB

Como se había mencionado anteriormente, el usuario o cliente del TB es un nodo Dual Stack, ya sea un host o un enrutador.

El TB debe contar con algún tipo de verificación de autenticidad del cliente, para evitar el uso no autorizado del servicio. El cliente debe proveer cierta información al TB para que se pueda configurar el túnel. Además, debe indicarle su dirección IPv4, un nombre para asociar la dirección IPv6 y también le debe indicar si es un host o un enrutador.

En caso de ser un enrutador IPv6 que va a proporcionar servicio a varios hosts IPv6, debe indicar también la cantidad de hosts para que le sea asignado un prefijo de acuerdo a sus necesidades, en lugar de una sola dirección.

Con la información necesaria proveída por el cliente, el TB decide que TS asignarle, basándose en la carga de tráfico que tenga cada TS. Después decide el prefijo IPv6 que va a asignarle al cliente. Este prefijo puede ir desde 0 hasta 128 bits, los más comunes son 48 (prefijo de sitio), 64 (prefijo de subred) o 128 (prefijo de host). Las direcciones IPv6 asignadas a ambos extremos del túnel deben ser globales y pertenecer al espacio de direcciones del TB.

Otras de las funciones del TB son decidir el tiempo de vida del túnel, registrar el nombre asociado con la dirección IPv6 en el DNS, configurar el TS y notificar al cliente la configuración del túnel y su nombre de dominio en el DNS. *[WWW011]*

3.3.3.4 Mantenimiento de los túneles

Los túneles ocupan muchos recursos de los TS's, tales como memoria y tiempo de procesamiento, y por eso es indispensable contar con un mecanismo encargado de la administración y manejo de estos túneles.

En la mayoría de los casos esto puede ser controlado con el tiempo de vida que asigne el TB, pero el problema surge cuando el cliente esta utilizando una conexión en la que las direcciones se asignan dinámicamente. Por ejemplo, los usuarios que se conectan a su proveedor de Internet mediante módems, en donde un servidor DHCP se encarga de asignarles una dirección cada vez que accesan, la cual la mayoría de las veces es diferente a la que habían usado previamente.

En este caso es recomendable que el cliente utilice el túnel, y al terminar su conexión, el túnel sea eliminado, ya que al volver a acceder al Internet tendría una nueva dirección IPv4 y tendría que reconfigurar el túnel. Esta reconfiguración consumiría tal vez los mismo recursos que crear un nuevo túnel, y no es recomendable. Otra opción sería que los TS's informaran continuamente al TB acerca del estado del túnel, y contar con un mecanismo encargado de revisar el estado de la conexión del cliente, y tan pronto éste se desconecte, el túnel sea eliminado.

El mantenimiento de un túnel utiliza muchos recursos de los TS's, lo que implica un costo extra, pero evitaría algunos problemas. De esta manera, cuando el cliente se conecte al Internet, aunque sea con una dirección IPv4 diferente, solamente tendría que accesar al TB y proveer su nueva dirección IPv4 y crear el túnel de nuevo. Así, el cliente podría utilizar las mismas direcciones IPv6 asignadas e incluso el mismo nombre de dominio en el DNS.

Actualmente existen varios proveedores de este servicio, los cuales se listan en el "ANEXO A" de este documento.

3.4 Traductores

3.4.1 SIIT

El método de SIIT (Stateless IP/ICMP Translation algorithm) básicamente se encarga de traducir las cabeceras entre IPv4 e IPv6 (incluyendo las cabeceras ICMP), y permite la comunicación entre hosts exclusivamente IPv6 y hosts exclusivamente IPv4. El nodo IPv6 de alguna forma obtendrá una dirección IPv4 temporal y un medio de enrutamiento para los paquetes. La dirección IPv4 temporal será utilizada como una dirección IPv6 llamada IPv4-traducida. Después los paquetes pasarán por un traductor SIIT encargado de traducir las cabeceras de los paquetes IPv4 e IPv6, así como las direcciones en las cabeceras. Las direcciones utilizadas en este método pueden ser IPv4, IPv4-traducidas o IPv4-mapeadas.
Este método no especifica de que manera se obtendrá la dirección IPv4 temporal (se sugiere DHCP con algunas extensiones), ni como será registrada en el DNS. Tampoco especifica el tipo de enrutamiento de los paquetes.

El método de SIIT puede ser utilizado cuando se desea establecer comunicación entre redes IPv6 pequeñas o hosts IPv6 y hosts IPv4, como se muestra en la siguiente figura:

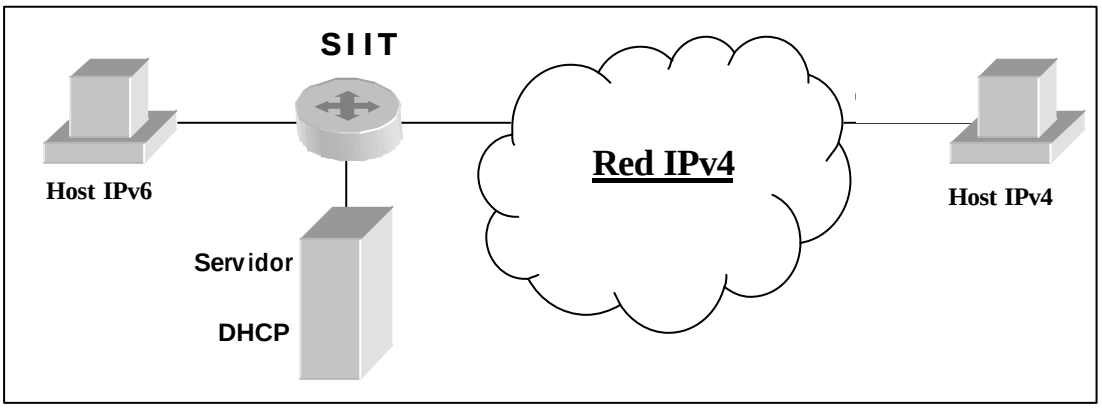


Figura 3.8 SIIT para redes pequeñas IPv6

También se puede utilizar cuando se desea establecer comunicación entre hosts IPv6 en una red Dual Stack y hosts IPv4, como se muestra en la siguiente figura:

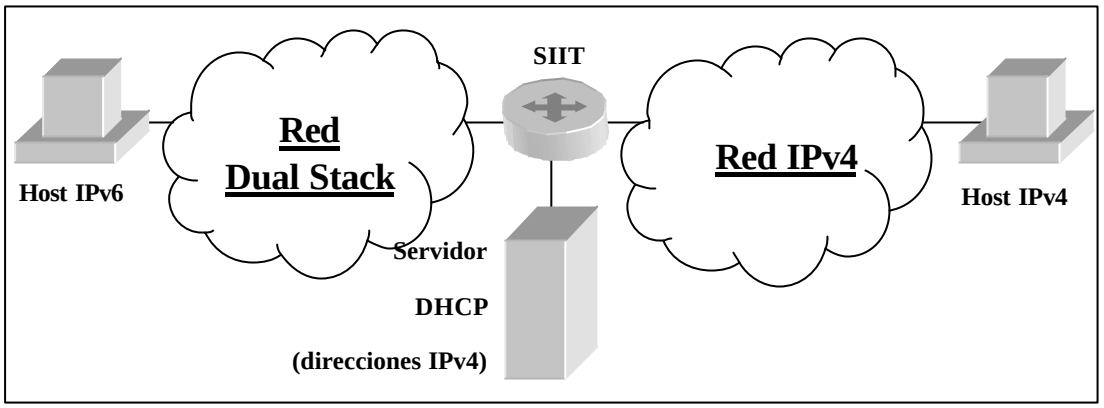


Figura 3.9 SIIT para redes Dual Stack

Este método no es recomendable después de la transición, ya que solo existirán algunas redes IPv4 pequeñas y los traductores se encontrarían en los límites de estas, lo que significa un largo recorrido de los paquetes provenientes de los *hosts* IPv6 para obtener una dirección IPv4 temporal, la cual les permitiría llevar a cabo la comunicación. *[WWW008]*

Las direcciones utilizadas por este método son las siguientes:

- IPv4-mapeada.- Una dirección de la forma 0::FFFF:a.b.c.d que identifica a un nodo que no soporta IPv6.
- IPv4-traducida.- Una dirección de la forma 0::FFFF:0:a.b.c.d que identifica a un nodo que soporta IPv6.

3.4.1.1 Traducción de IPv4 a IPv6

Cuando un traductor IPv4-IPv6 recibe un paquete IPv4 destinado a un host que se encuentra fuera de su dominio, debe traducir la cabecera IPv4 del paquete a una cabecera IPv6, para después reenviar ese paquete al exterior del dominio. La cabecera IPv4 es removida completamente y reemplazada por la cabecera IPv6. Los paquetes ICMP, así como la cabecera de transporte y la carga de datos no cambian. En IPv6 es necesario

realizar un Path MTU Discovery antes de enviar un paquete, pero en IPv4 no lo es. Esto significa que los enrutadores intermedios IPv6 nunca fragmentan paquetes, el único habilitado para hacer esto es el host que manda el paquete.

Cuando un host IPv4 realiza un Path MTU Discovery (habilitando el bit "DF" de la cabecera), puede realizar este proceso de host a host pasando por un traductor. En este proceso, enrutadores IPv4 o IPv6 podrán enviar mensajes ICMP de vuelta al host para indicarle que los paquetes son muy grandes ("packet too big"). Cuando estos paquetes sean enviados por un enrutador IPv6, estos deberán pasar por el traductor para que este se encargue de efectuar la traducción de paquetes ICMP a una forma en que el host IPv4 pueda entenderlos.

En caso contrario, cuando un host IPv4 no realiza un Path MTU Discovery, el traductor debe asegurar que los paquetes no rebasan el MTU máximo del enlace IPv6. Esto se logra mediante la fragmentación de los paquetes IPv4 de una manera que quepan en paquetes IPv6 de 1280 bytes, ya que con este tamaño se garantiza que los paquetes no tendrán que ser fragmentados.

3.4.1.2 Traducción de cabeceras IPv4 a cabeceras IPv6

Los campos de la cabecera IPv6 se traducen como sigue:

Versión	Version	6
Clase de Tráfico	Traffic Class	Se copia del campo Type of Service de la cabecera IPv4.
Etiqueta de flujo	Flow Label	0 (todos los bits en cero).
Longitud de carga útil	Payload Length	Longitud total de la cabecera IPv4, menos el tamaño de esta y sus opciones, si existen.
Proxima Cabecera	Next Header	Se copia del campo Protocol de la cabecera IPv4.

Limite de saltos	Hop Limit	Se copia del campo Time to Live de la cabecera IPv4. Como el traductor es un enrutador, este debe decrementar el campo Time to Live de la cabecera IPv4 o Hop Limit de la cabecera IPv6 antes de reenviar el paquete. Después de decrementar el valor también debe verificar que este no esté en cero. Si se encuentra en cero, deberá mandar un mensaje de error ("ttl exceeded").
Dirección de origen	Source Address	Los 32 bits más a la derecha son la dirección fuente IPv4. Los 96 bits anteriores son el prefijo para las direcciones IPv4-mapeadas (::FFFF:0:0/96).
Dirección de destino	Destination Address	Los 32 bits más a la derecha son la dirección destino IPv4. Los 96 bits anteriores son el prefijo para las direcciones IPv4-traducidas (0::FFFF:0:0:0/96).

Tabla 3.1 Traducción de cabeceras IPv4 a cabeceras IPv6

Si se encuentran opciones en la cabecera IPv4, estas deberán ser ignoradas, no deberán ser traducidas. Si fuera necesario agregar una cabecera de fragmentación (el bit de "DF" no está activo o el paquete es un fragmento), los campos se deben traducir como sigue:

Campos de la cabecera IPv6:

Longitud de carga útil	Payload Length	Longitud total de la cabecera IPv4, mas 8 de la cabecera de fragmentación, menos el tamaño de la cabecera IPv4 y sus opciones, si existen.
Proxima Cabecera	Next Header	Cabecera de fragmentación (44).

Campos de la cabecera de fragmentación IPv6:

Proxima Cabecera	Next Header	Se copia del campo Protocol de la cabecera IPv4.
Compensación de fragmento de cabecera	Fragment Offset	Se copia del campo Fragment Offset de la cabecera IPv4.
Bandera M	M Flag	Se copia el bit del campo More Fragments de la cabecera IPv4.
Identificación	Identification	Se copian los 16 bits más a la derecha del campo Identification de la cabecera IPv4. Los 16 bits restantes se rellenan con ceros.

3.4.1.3 Traducción de IPv6 a IPv4

Cuando un traductor IPv6-IPv4 recibe un paquete IPv6 destinado a una dirección IPv6 de tipo IPv4-mapeada, debe traducir la cabecera IPv6 a una cabecera IPv4, para después reenviar ese paquete a su destino. La cabecera IPv6 original es removida completamente y reemplazada por la cabecera IPv4. Los paquetes ICMP, así como la cabecera de transporte y la carga de datos no cambian.

Un enlace IPv6 debe tener un MTU de 1280 bytes o más, mientras que IPv4 debe tener un MTU de 68 bytes. Entre IPv4 e IPv6 existen diferencias que afectan la traducción, tales como la fragmentación y el MTU de los enlaces. No es posible realizar un Path MTU Discovery de host a host cuando existe un traductor IPv6-IPv4, debido a que el host IPv6 puede recibir mensajes ICMP de error, indicándole que los paquetes son muy grandes originados por un enrutador IPv4 que reporte un MTU menor de 1280 bytes.

Los host IPv6 responden a estos mensajes de error de ICMP reduciendo el MTU del enlace a 1280 bytes, e incluyen una cabecera de fragmentación IPv6 a cada paquete, indicando que este puede ser fragmentado. Esto permite que se realice el proceso de Path MTU Discovery a través del traductor mientras el MTU del enlace sea de 1280 bytes o menor. Cuando el MTU sea menor de 1280 bytes, el nodo enviará paquetes de 1280 bytes que serán fragmentados por enrutadores IPv4 a lo largo del enlace, antes de ser traducidos a IPv4.

3.4.1.4 Traducción de cabeceras IPv6 a cabeceras IPv4

Los campos de la cabecera IPv4 se traducen como sigue:

Versión	Version	4
Longitud de la cabecera de Internet	Internet Header Length	5 (sin opciones IPv4)
Tipo de servicio	Type of Service	Se copia del campo Traffic Class de la cabecera IPv6.
Longitud total	Total Length	Longitud de la carga de datos de la cabecera IPv6, mas el tamaño de la cabecera IPv4.
Identificación	Identification	Todos en cero.
Banderas	Flags	La bandera de More Fragments se pone en cero. La bandera de Don't Fragment se pone en uno.

Compensación de fragmento de cabecera	Fragment Offset	Todos en cero.
Tiempo de vida	Time to Live	Se copia del campo Hop Limit de la cabecera IPv6. Como el traductor es un enrutador, este debe decrementar el campo Time to Live de la cabecera IPv4 o Hop Limit de la cabecera IPv6 antes de reenviar el paquete. Después de decrementar el valor también debe verificar que este no esté en cero. Si se encuentra en cero, deberá mandar un mensaje de error ("ttl exceeded").
Protocolo	Protocol	Se copia del campo Next Header de la cabecera IPv6.
Suma de verificación de cabecera	Header Checksum	Se calcula una vez que la cabecera IPv4 ha sido creada.
Dirección de origen	Source Address	Si la dirección origen IPv6 es una dirección de tipo IPv4-traducida entonces se copian los 32 bits más a la derecha. De otra manera, la dirección origen se cambia a 0.0.0.0, evitando que los paquetes sean descartados.
Dirección de destino	Destination Address	Los paquetes IPv6 que son traducidos deben tener una dirección destino IPv6 de tipo IPv4-mapeada. Se copian los 32 bits más a la derecha.

Tabla 3.2 Traducción de cabeceras IPv6 a cabeceras IPv4

Si el paquete IPv6 contiene una cabecera de fragmentación, entonces los campos se traducen como se indicó anteriormente, con las siguientes excepciones:

Longitud total	Total Length	Longitud de la carga de datos de la cabecera IPv6, menos 8 de la cabecera de fragmentación, mas el tamaño de la cabecera IPv4.
Identificación	Identification	Se copian los 16 bits más a la derecha del campo Identification de la cabecera de fragmentación.
Banderas	Flags	Se copia la bandera M de la cabecera de fragmentación a la bandera More Fragments. La bandera Don't Fragment se pone en cero, permitiendo que el paquete sea fragmentado por enrutadores IPv4.
Compensación de fragmento de cabecera	Fragment Offset	Se copia del campo Fragment Offset de la cabecera de fragmentación.
Protocolo	Protocol	Se copia del campo Next Header de la cabecera de fragmentación.

3.4.2 NAT-PT

El método de NAT-PT (Network Address Translator – Protocol Translator) es similar al método de NAT utilizado en IPv4, pero no idéntico. El NAT utilizado en IPv4 consiste en traducir una dirección IPv4 a otra dirección IPv4, mientras que NAT-PT consiste básicamente en la traducción de direcciones IPv4 a direcciones IPv6 y viceversa. NAT-PT utiliza un grupo de direcciones IPv4 (se asume que son únicas globalmente y no privadas) para asignar dinámicamente a los nodos IPv6 cuando estos inicien una sesión para establecer comunicación con algún otro nodo. Todos los paquetes pertenecientes a una misma sesión deberán pasar por el mismo enrutador NAT-PT. Este método utiliza SIIT para la traducción de protocolos, con algunas modificaciones que se explicarán más adelante. *[WWW016]*

Una parte fundamental de NAT-PT son los ALG's (Application Level Gateways). Estos se utilizan cuando se manejan direcciones IP dentro de la carga de datos del paquete para realizar la traducción, ya que NAT-PT no revisa esa parte del paquete, y por lo tanto no traduce las direcciones en la carga de datos. El ALG más importante es el DNS-ALG, el cual se encarga de mapear las direcciones IPv4 asignadas a un host IPv6.

3.4.2.1 NAT-PT Tradicional

El NAT-PT tradicional permite a nodos dentro de un dominio IPv6 establecer comunicación con nodos en un dominio IPv4, pero solamente en un sentido, es decir, solamente paquetes al exterior del dominio IPv6. Este se divide en NAT-PT Básico y NAPT-PT.

3.4.2.1.1 NAT-PT Básico

A continuación se muestra un diagrama para explicar los distintos tipos de NAT-PT:

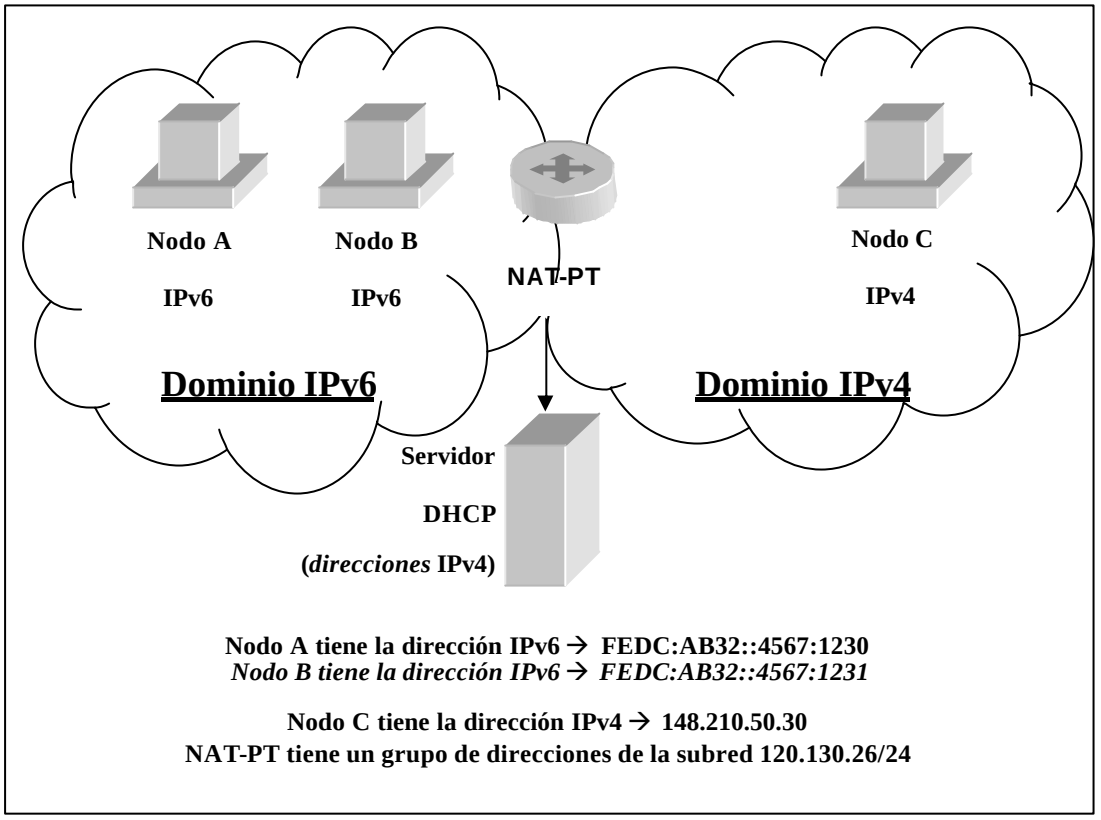


Figura 3.10 Entorno NAT-PT

Si el grupo de direcciones IPv4 es igual ó mayor que el número de nodos IPv6, entonces se podrá asignar una dirección IPv4 a cada nodo IPv6. Si el grupo de direcciones IPv4 es menor, entonces se tendrán que asignar las direcciones IPv4 dinámicamente.

Supongamos que el nodo A quiere establecer comunicación con el nodo C y crea un paquete con dirección fuente FEDC:AB32::4567:1230 y dirección destino PREFIJO::148.210.50.30. El prefijo PREFIJO::/96 es publicado en el dominio por el NAT-PT, por lo cual todos los paquetes con este prefijo serán direccionados al NAT-PT. Si el paquete no es un inicio de sesión, entonces el NAT-PT deberá conocer el estado de la sesión, así como las direcciones previamente asignadas y el mapeo entre direcciones IPv4 e

IPv6. Si el paquete es un inicio de sesión, entonces el NAT-PT le asigna una dirección del grupo de direcciones IPv4 (por ejemplo 120.130.26.1) y traduce el paquete a IPv4. Los parámetros y el mapeo de las direcciones IPv4 e IPv6 se guardan en el NAT-PT el tiempo que dure la sesión. El paquete traducido tiene como dirección fuente 120.130.26.1 y como dirección destino 148.210.50.30. Cualquier paquete de respuesta que sea reconocido como perteneciente a la misma sesión será traducido utilizando la información guardada previamente. La dirección fuente sería PREFIJO::148.210.50.30 y la dirección destino FEDC:AB32::4567:1230.

3.4.2.1.2 NAPT-PT

NAPT-PT (Network Address Port Translation – Protocol Translation) permite que múltiples nodos IPv6 se comuniquen con nodos IPv4 utilizando una sola dirección IPv4. Esto se logra especificando los puertos TCP/UDP además de la dirección IPv4, lo que permite establecer un gran número de sesiones utilizando una sola dirección IPv4.
Una ventaja de utilizar NAPT-PT por encima de NAT-PT es que soluciona el problema de la escasez de direcciones IPv4. Si se terminaran las direcciones IPv4 utilizadas por NAT-PT, este mecanismo dejaría de funcionar, ya que los nodos IPv6 nuevos no podrían establecer sesiones con redes exteriores.

Supongamos que tenemos NAPT-PT en un ***enrutador frontera*** (en vez de NAT-PT) y todas las direcciones IPv6 pueden ser mapeadas a una dirección IPv4 única (120.130.26.1). Cuando el nodo A desea establecer una sesión TCP con el nodo C, crea un paquete con dirección fuente FEDC:AB32::4567:1230, puerto TCP fuente 3017, dirección destino PREFIJO::148.210.50.30 y puerto TCP destino 23. Cuando el paquete llega al NAPT-PT, este le asigna una dirección IPv4 respetando el número del puerto de la dirección IPv6, por lo tanto la dirección fuente sería 120.130.26.1, el puerto TCP fuente 1025, la dirección destino 148.210.50.30 y el puerto TCP destino 23.

Cualquier paquete proveniente de la dirección 148.210.50.30 con puerto TCP 23 será reconocido como perteneciente a la misma sesión, y la dirección fuente sería

PREFIJO::148.210.50.30, el puerto TCP fuente 23, la dirección destino FEDC:AB32::4567:1230 y el puerto TCP destino 3017.

3.4.2.2 NAT-PT Bidireccional

El NAT-PT Bidireccional, como lo indica su nombre, permite que las sesiones sean iniciadas por hosts dentro del dominio IPv6 o hosts en un dominio IPv4. NAT-PT Bidireccional debe ser utilizado en conjunto con un DNS-ALG para facilitar el mapeo entre direcciones y nombres. Al iniciarse una sesión, la asignación de direcciones IPv4 a paquetes dirigidos al interior o al exterior del dominio IPv6 se manejan de una manera distinta, como se explica a continuación.

3.4.2.2.1 Sesiones al interior del dominio (IPv4 – IPv6)

Cuando un nodo en el dominio IPv4 envía una requisición de búsqueda de nombre para un nodo dentro del dominio IPv6, esta requisición es enviada al servidor DNS en el dominio IPv6. Como NAT-PT se encuentra en la frontera de ambos dominios, este intercepta la requisición y el DNS-ALG se encarga de traducir la requisición, modificando lo siguiente:

- Cambia el tipo de requisición de A al tipo de requisición AAAA.
- Reemplaza la cadena “IN-ADDR.ARPA” por la cadena “IP6.INT”, así como la dirección IPv4 precedente a la cadena “IN-ADDR.ARPA” con la dirección IPv6 correspondiente (si se ha efectuado previamente un mapeo) en orden inverso.

En caso contrario, cuando se envía una respuesta del servidor DNS del dominio IPv6 al nodo IPv4, NAT-PT intercepta la requisición y el DNS-ALG se encarga de la traducción, modificando lo siguiente:

- Cambia el tipo de respuesta DNS de tipo A al tipo AAAA.
- Reemplaza la dirección IPv6 devuelta por el servidor DNS del dominio IPv6 por la dirección IPv4 asignada previamente por el enrutador NAT-PT. Si la dirección IPv4 no ha sido asignada previamente, se asignará en ese momento.

Supongamos que el nodo C desea establecer una sesión con el nodo A, y solicita una requisición de búsqueda de nombre a su servidor DNS. Este reenvía esta requisición al servidor DNS en el dominio IPv6, pero NAT-PT intercepta esta requisición y solicita al DNS-ALG efectúe la traducción del tipo A al tipo AAAA. Después, es enviada al servidor DNS en el dominio IPv6, quien responde de la siguiente manera:

NodoA AAAA FEDC:AB32::4567:1230

Cuando se envía esta respuesta al dominio IPv4, NAT-PT intercepta la respuesta y solicita al DNS-ALG efectué la traducción correspondiente, quedando de la siguiente manera:

NodoA A 120.130.26.1

Esta respuesta de tipo A es enviada al nodo C en el dominio IPv4, y este puede entonces establecer una sesión con el nodo A.

3.4.2.2.2 Sesiones al exterior del dominio (IPv6 – IPv4)

Los nodos IPv6 pueden obtener las direcciones de nodos IPv4 de un servidor en el dominio IPv4 o del servidor dentro del dominio IPv6. Si el servidor DNS en el dominio IPv6 almacena registros para direcciones de nodos IPv6, así como registros para direcciones de nodos IPv4, entonces las requisiciones hechas por nodos IPv6 para direcciones IPv4 no deberán salir del dominio IPv6 y por lo tanto no serán interceptadas por el NAT-PT. Si el servidor DNS en el dominio IPv6 almacena únicamente registros para direcciones de nodos IPv6, entonces las requisiciones hechas por nodos IPv6 para direcciones IPv4 deberán salir del dominio en busca de un servidor DNS en el dominio IPv4. Esto significa que serán interceptadas por el NAT-PT, y el DNS-ALG se encargará de la traducción. *[WWW013]*

Supongamos que el nodo A quiere establecer una sesión con el nodo C, por lo que hace una requisición de búsqueda de nombre de tipo AAAA para el nodo C. Como el nodo C puede tener direcciones IPv4 ó IPv6, la requisición se envía sin cambio alguno al servidor DNS en el dominio IPv4, así como una requisición de tipo A.

Si existe un registro de tipo AAAA para el nodo C, este se devuelve al NAT-PT, quien lo envía al nodo A. Si existe un registro de tipo A para el nodo C, este también se devuelve al NAT-PT, entonces el DNS-ALG traduce la respuesta agregando el prefijo correspondiente y lo envía al nodo A, de la siguiente manera:

NodoC A 148.210.50.30

es traducido a

NodoC AAAA PREFIJO:: 148.210.50.30

El nodo A puede entonces utilizar esta dirección para establecer una sesión con el nodo C.

3.4.2.3 Traducción de Protocolo

En NAT-PT se utilizan los mismos métodos especificados en SIIT, a excepción de algunas modificaciones, debido a que NAT-PT también traduce direcciones. A continuación se muestran estas modificaciones:

Dirección de origen	Source Address	Los 32 bits más a la derecha son la dirección IPv4. Los 96 bits restantes son el PREFIJO. El prefijo es publicado por el NAT-PT, y todos los paquetes con este prefijo, serán direccionados al NAT-PT.
Dirección de destino	Destination Address	La dirección destino IPv4 es reemplazada por la dirección destino IPv6, basándose en el mapeo previamente establecido.
Dirección de origen	Source Address	La dirección fuente IPv6 es reemplazada por la dirección fuente IPv4, basándose en el mapeo previamente establecido.
Dirección de destino	Destination Address	Los paquetes IPv6 que son traducidos tienen una dirección destino de la forma PREFIJO ::IPv4/96. Los 32 bits más a la derecha de la dirección destino IPv6 se copian la dirección destino IPv4.

3.4.3 BIS

Actualmente existen muy pocas aplicaciones IPv6 en comparación con aplicaciones IPv4. El objetivo en un futuro es que el número de aplicaciones IPv6 sea igual o mayor que el número de aplicaciones IPv4, pero mientras esto ocurre, se necesitan traductores. Un traductor capaz de ejecutar este tipo de traducciones es BIS (Bump-In-the-Stack), que permite a hosts Dual Stack comunicarse con hosts IPv6 utilizando aplicaciones IPv4.

El método de BIS permite a los hosts convertirse en traductores autónomos, sin necesidad de un traductor externo. BIS se encuentra en el área de seguridad IP, y se encarga de verificar los datos que pasan entre TCP/IPv4 y la interfase de red, además de traducirlos a IPv6 y viceversa.

3.4.3.1 Componentes de BIS

Los hosts Dual Stack necesitan contar con aplicaciones, módulos de TCP/IP y direcciones, tanto para IPv4 como para IPv6. El método de BIS sustituye las aplicaciones IPv6 por tres módulos que le permiten al host comunicarse con otros hosts utilizando aplicaciones IPv4. Estos tres módulos reciben los nombres de Extension Name Resolver, Address Mapper y Translator.

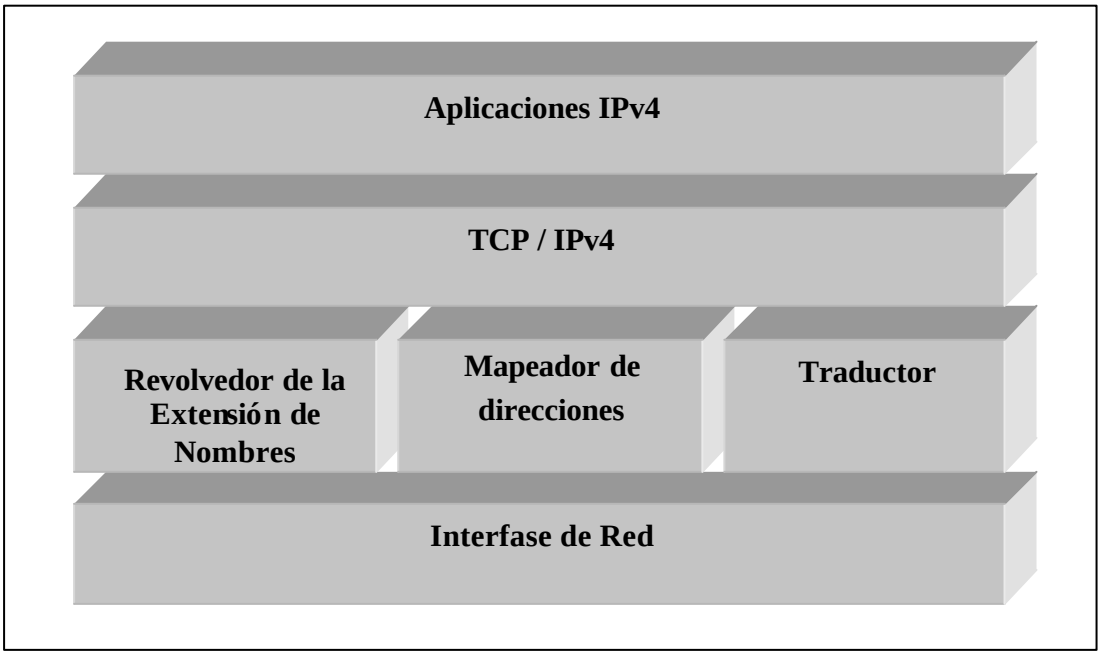


Figura 3.11 Componentes de BIS

3.4.3.1.1 Resolvedor de la extensión de nombres

Se encarga de responder las requisiciones de nombres de la aplicación IPv4. La aplicación IPv4 hace una requisición de tipo A para obtener la dirección del host con el que quiere establecer comunicación. El modulo de Extension Name Resolver se encarga de crear otra requisición de tipo AAAA y envía ambas requisiciones al servidor DNS. Si este responde a la requisición A, entonces esta es enviada a la aplicación IPv4 y el proceso se lleva a cabo normalmente, es decir, no hay necesidad de una traducción de paquetes. Si el servidor DNS responde a la requisición AAAA, entonces esta es enviada al módulo de Address Mapper para que este le asigne una dirección IPv4 correspondiente y realice el mapeo de las direcciones. Después, el módulo de Extension Name Resolver crea una respuesta a la requisición A con la nueva dirección IPv4 asignada por el Address Mapper y la envía a la aplicación IPv4.

3.4.3.1.2 Mapeador de direcciones

Cuenta con un grupo de direcciones IPv4 (pueden ser direcciones privadas). También se encarga de mantener una tabla que consiste en pares de direcciones IPv4 e IPv6. Cuando el módulo de Extension Name Resolver o Translator requieren una dirección IPv4, le notifican a este módulo para que asocie una dirección IPv4 del grupo de direcciones a la dirección IPv6 que se esté utilizando. Esta asociación se debe registrar en la tabla. Los casos en que se efectúa el registro son los siguientes:

- Cuando el módulo de Extension Name Resolver obtiene respuesta solamente para la requisición de tipo AAAA, y no existe una entrada en la tabla que involucre a la dirección obtenida.
- Cuando el módulo de Translator recibe un paquete IPv6 y no existe una entrada en la tabla que involucre la dirección fuente del paquete.

Existe una excepción en el registro de entradas de la tabla, y esta se presenta al principio de la creación de la tabla, cuando registra un par de direcciones IPv4 e IPv6 propias, de una manera estática.

3.4.3.1.3 Traductor

Se encarga de efectuar la traducción entre IPv4 e IPv6, utilizando el método de SIIT. Cuando recibe paquetes IPv4 de aplicaciones IPv4, convierte las cabeceras IPv4 en cabeceras IPv6, y después fragmenta los paquetes IPv6 debido a que las cabeceras IPv6 son dos veces más grandes que las cabeceras IPv4. Cuando recibe paquetes IPv6 de redes IPv6, convierte las cabeceras IPv6 en cabeceras IPv4, pero en este caso no efectúa ninguna fragmentación, ya que no es necesaria.

3.4.3.2 Comunicación en BIS

En el método de BIS, cuando se desea establecer comunicación entre un host Dual Stack (el cual maneja BIS) y un host IPv6, la comunicación puede ser iniciada por cualquiera de los dos hosts. A continuación se presentan ambos casos.

3.4.3.2.1 Comunicación iniciada por host Dual Stack

La aplicación IPv4 hace una requisición de tipo A a su servidor DNS para obtener la dirección del host IPv6. El módulo Extension Name Resolver crea una requisición de tipo AAAA para el host IPv6, y envía ambas requisiciones al servidor DNS. En este caso, como la comunicación esta dirigida al host IPv6, solamente se obtendrá respuesta de la requisición tipo AAAA.

El módulo Extension Name Resolver solicita al módulo Address Mapper que le asigne una dirección IPv4 de su grupo de direcciones para asociarla con la dirección IPv6 del host IPv6. El Extension Name Resolver crea una respuesta a la requisición de tipo A, pero con la nueva dirección IPv4 y la envía a la aplicación IPv4.

Luego la aplicación envía un paquete IPv4 al host IPv6. El paquete llega al módulo

Translator, el cual intenta traducir dicho paquete en un paquete IPv6, pero no sabe traducir las direcciones IPv4 fuente y destino. Es entonces cuando el módulo Translator le solicita al módulo Address Mapper que le provea las direcciones IPv6. El Address Mapper revisa su tabla, encuentra las direcciones IPv6 asociadas con las direcciones IPv4 y las envía al módulo Translator. Es entonces cuando este módulo traduce el paquete IPv4 a un paquete IPv6, fragmenta el paquete IPv6 si es necesario y lo envía al host IPv6.

El paquete IPv6 llega al host IPv6 el cual responde enviando paquetes IPv6 al host Dual Stack, que son interceptados por el módulo Translator de este mismo. El Translator obtiene las direcciones IPv4 asociadas con las direcciones IPv6, traduce el paquete IPv6 a un paquete IPv4 y lo envía a la aplicación IPv4.

3.4.3.2.2 Comunicación iniciada por host IPv6

El host IPv6 hace una requisición de tipo AAAA a su servidor DNS para el host Dual Stack, y envía un paquete IPv6 a la dirección IPv6 obtenida. El paquete IPv6 es interceptado por el módulo Translator del host Dual Stack.
El módulo Translator intenta traducir el paquete IPv6 en un paquete IPv4, pero no sabe traducir las direcciones IPv6 fuente y destino. Es entonces cuando el módulo Translator le solicita al módulo Address Mapper que le provea las direcciones IPv4.

El módulo Address Mapper revisa su tabla y encuentra únicamente la dirección IPv6 destino, la cual fue registrada en la creación de la tabla, pero no encuentra la dirección IPv6 fuente.
Después, el módulo Address Mapper asocia una dirección IPv4 de su grupo a la dirección IPv6 fuente, y envía las direcciones IPv4 fuente y destino al módulo Translator. Es entonces cuando este módulo traduce el paquete IPv6 a un paquete IPv4 y lo envía a la aplicación IPv4 del host Dual Stack. La aplicación le envía un paquete IPv4 al host IPv6 como respuesta y sigue el proceso descrito en la sección anterior.