

Responsabilidad Corporativa y Sostenibilidad

# Cuaderno Red de Cátedras Telefónica

Universidad Carlos III de Madrid

## NAT64/DNS64 para la transición a IPv6

Cátedra Telefónica de Internet del Futuro de la Universidad Carlos III de Madrid

El agotamiento inminente de las direcciones IPv4 hace necesaria la coexistencia de IPv6 e IPv4. Este artículo describe NAT64 y DNS64, herramientas que permiten la comunicación entre nodos únicamente con soporte IPv4 o IPv6.

Marcelo Bagnulo, Alberto García-Martínez, José Manuel Camacho.

Octubre 2011

## Biografía



### Marcelo Bagnulo

Marcelo es Profesor Titular del Departamento de Ingeniería Telemática de la Universidad Carlos III de Madrid. Es el Director de la Catedra Telefónica de Internet del Futuro para la productividad de la Universidad Carlos III de Madrid.



### Alberto García- Martínez

Alberto es Profesor Titular del Departamento de Ingeniería Telemática de la Universidad Carlos III de Madrid.



### José Manuel Camacho

José es Profesor Ayudante del Departamento de Ingeniería Telemática de la Universidad Carlos III de Madrid donde realiza sus estudios de doctorado.

# Índice

1. Introducción
2. Requisitos para la traducción IPv6-IPv4
3. NAT64
4. DNS64
5. RECORRIDO POR NAT64/DNS64
6. Conclusiones
7. Referencias

## 1. Introducción

A principios de los 90s empezó a ser evidente que los 32 bits de las direcciones IP no serían suficientes a largo plazo, lo que llevó al IETF a crear una nueva versión de IP con direcciones más largas, IPv6. Desafortunadamente, el protocolo IPv6 no es compatible con la versión anterior IPv4, por lo que son necesarias una serie de herramientas que permitan la transición y coexistencia de ambos protocolos. Existen tres mecanismos principales de transición entre IPv4 y el nuevo IPv6: tunelado, dual-stack y traducción. El tunelado consiste en el transporte o encapsulado de paquetes IPv6 dentro de paquetes IPv4 para comunicación entre nodos IPv6 a través de redes que sólo soportan IPv4. La utilización de dual-stack implica utilizar IPv4 e IPv6 en el mismo equipo simultáneamente. Un modelo interesante para la transición sería instalar todos los equipos con dual-stack y posteriormente eliminar el soporte para IPv4. Sin embargo, teniendo en cuenta el actual despliegue de servicios y equipos con dual-stack, parece poco probable que todos los nodos en Internet tengan dual-stack para cuando las direcciones IPv4 se hayan agotado. La traducción de paquetes IPv4 a IPv6 y viceversa es la única técnica de transición que permite la comunicación entre nodos IPv4 y nodos sólo con soporte IPv6.

NAT64/DNS64 [4] [5] es un conjunto de herramientas que habilitan las comunicaciones entre nodos IPv4 y nodos IPv6. NAT64 traduce paquetes IPv4 a paquetes IPv6 y viceversa generando estado y DNS64 crea registros DNS tipo AAAA para equipos IPv4 que solo cuentan con registros A en el servidor.

## 2. Requisitos para la traducción IPv6-IPv4

La tecnología de traducción de direcciones de red (Network Address Translation, NAT) para IPv4 fue inicialmente definida en el IETF en la RFC 1631 [7]. Esa RFC describe el funcionamiento global de un NAT, pero no contiene una especificación detallada que garantice el funcionamiento homogéneo de los NAT de diferentes fabricantes. El IETF era entonces reacio a proveer una especificación más detallada debido a que NAT era vista como una tecnología inferior que podría tener un impacto muy negativo en la arquitectura de Internet [3]. Este enfoque llevó a un extenso despliegue de una gran cantidad de diferentes implementaciones de NAT que se comportaban de una manera heterogénea

respecto a la manera de gestionar la traducción. Las características de una comunicación que pasaba a través de un equipo NAT eran difícilmente predecibles, por ejemplo la manera en que se asignan los puertos o el tiempo que un cierto mapeo se mantenía activo. Esto era especialmente problemático para las aplicaciones ejecutándose detrás del NAT que necesitaban crear y mantener estado en el NAT para permitir las comunicaciones iniciadas desde fuera del NAT. Con el fin de mitigar este fenómeno, el grupo de trabajo BEHAVE WG del IETF definió un conjunto de requisitos de comportamiento para los NATs IPv4 que cubría las potenciales interacciones con TCP [8], UDP [9] e ICMP [10].

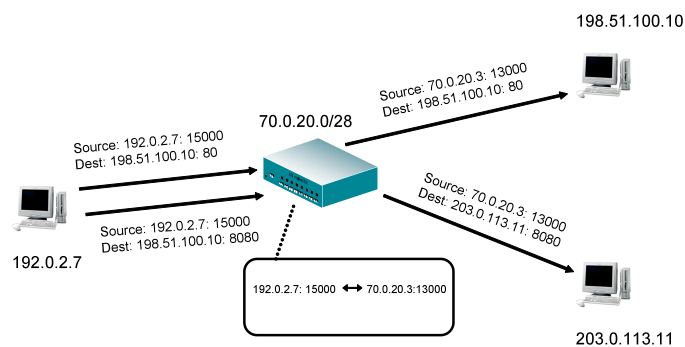


Figura 1. Mapeo independiente del destino.

Análogamente al caso de los NAT en IPv4, para las traducciones entre IPv4 e IPv6, es necesario definir un conjunto de requisitos para la traducción IPv6-IPv4. En el resto de esta sección, se subrayan los principales aspectos de comportamiento de los NATs IPv4 y su extensión para la traducción IPv6-IPv4.

De cara a entender el funcionamiento de un NAT, es importante distinguir entre el comportamiento del mapeo y del filtrado. En líneas generales, un NAT IPv4 es un dispositivo que conecta dos dominios con direcciones IPv4, uno de ellos utiliza direcciones privadas (no alcanzables desde fuera del NAT) y otro dominio que típicamente es el dominio público de Internet. Al recibir un paquete enviado desde el dominio privado, el NAT crea y almacena estado del mapeo entre el par dirección y puerto origen (privado) y el par dirección y puerto público asignado por el NAT. Definimos como dirección de transporte al par dirección/puerto. El NAT entonces traduce la dirección de transporte

origen de los paquetes por la seleccionada para ese mapeo en concreto y reenvía el paquete al dominio público.

El comportamiento del NAT define como se realiza el mencionado mapeo. Tres tipos de comportamiento han sido definidos:

- Mapeo independiente del destino (Fig. 1): El mapeo queda determinado en exclusiva por la dirección de transporte del equipo detrás del NAT. Los paquetes con la misma dirección de transporte privada se mapean a la misma dirección de transporte del repositorio del NAT independientemente de su dirección pública de destino.
- Mapeo por dirección: El mapeo en este caso queda determinado por la dirección de transporte del equipo detrás del NAT y la dirección IP del equipo externo. Los paquetes con la misma dirección privada de transporte y la misma dirección de red pública son mapeados a la misma dirección de transporte elegida por el NAT de su repositorio independientemente del puerto utilizado por el equipo externo al NAT.
- Mapeo por dirección y puerto: El mapeo en este caso depende tanto de la dirección de transporte interna como de la dirección de transporte del destino externo.

Los diferentes tipos de mapeos determinan como los equipos detrás del NAT son vistos por los equipos externos. Las diferentes conexiones iniciadas por el mismo proceso ejecutándose en un equipo detrás del NAT, que utiliza mapeo independiente del destino, serán vistas por el equipo externo con la misma dirección de transporte origen. Técnicas NAT-traversal optimizadas, como STUN [6], requieren de este comportamiento. Los requerimientos de funcionamiento de los NAT, tanto para TCP como UDP, dictan que se debe usar mapeo independiente del destino.

No todos los paquetes son reenviados por el equipo NAT, dado que además se definen reglas de filtrado. El filtrado sucede cuando un equipo NAT recibe un paquete en una de sus interfaces y aplica las reglas de filtrado para determinar si el paquete se reenvía o no en base a la información de direcciones y/o puertos. En función del comportamiento del filtrado, se definen los siguientes tipos:

- Filtrado independiente del destino: las reglas de filtrado sólo dependen de la dirección de transporte del equipo interno. Esto implica que un paquete es reenviado o eliminado dependiendo en exclusiva de la dirección del equipo detrás del NAT (ya sea la dirección de transporte privada o la pública asignada por el NAT en el mapeo).
- Filtrado por dirección: las reglas de filtrado se crean en base a la dirección de transporte del equipo interno y la dirección de red del equipo externo.

- Filtrado por dirección y puerto: el filtrado depende de ambos, la dirección de transporte del equipo interno y del externo.

La recomendación para los NAT es implementar filtrado independiente del destino y en caso de necesitar mayor seguridad, utilizar filtrado por direcciones.

De cara a extender estos requisitos para el caso del NAT64, es necesario equiparar los roles de las direcciones IPv4 e IPv6 a los roles de las direcciones públicas y privadas que encontramos en los NAT para IPv4. Puesto que un mapeo se crea cuando el traductor recibe un paquete del dominio IPv6, es evidente que el dominio IPv6 en NAT64 debe mapearse al dominio privado en el caso de NATs para IPv4, y el dominio IPv4 en NAT64 al dominio público para NATs IPv4. La gran similitud entre NAT IPv4 y NAT64 permite derivar inmediatamente los requisitos para el mapeo y el filtrado para NAT64, soportando mapeo independiente de destino y dos tipos de filtrado, filtrado independiente del destino y por dirección.

Existen otros requisitos que han sido definidos por el IETF para obtener un comportamiento adecuado de los NAT y con los que NAT64 debe ser compatible. En particular, el tiempo mínimo de vida recomendado para mapeos de UDP y TCP es de 5 minutos y 2 horas respectivamente. Hay otros requisitos ([8], [9] y [10]) relacionados con varios aspectos del funcionamiento de los NAT, como la asignación de puertos, fragmentación y soporte para ICMP, entre otros.

### 3. NAT64

NAT64 traduce paquetes IPv6 en paquetes IPv4 y viceversa. Cuenta principalmente con dos mecanismos, el de traducción de direcciones y el de traducción de protocolos. Este último se encarga de traducir campos que no son direcciones y trabaja sin necesidad de estado para realizar las traducciones, intentando en la manera de los posible mantener la semántica de los campos originales. Originalmente fue definido en [1] y ha sido actualizado en [11].

La traducción de direcciones mapea direcciones de transporte IPv6 a direcciones de transporte IPv4 y viceversa. Para crear los diferentes mapeos, un equipo NAT64 cuenta con dos repositorios de direcciones, un repositorio de direcciones IPv6 (para representar direcciones IPv4 en el dominio IPv6) y un repositorio de direcciones IPv4 (para representar direcciones IPv6 en el dominio IPv4).

NAT64 crea los mapeos de IPv4 al repositorio IPv6 utilizando un prefijo IPv6 (denotado como Pref64::*n*). Cada dirección IPv4 se mapea en una dirección IPv6 simplemente

concatenando el prefijo Pref64::*n* con la dirección IPv4 mapeada y en caso de que el valor de *n* sea menor que 96, un sufijo adicional [2]. Pref64::*n* puede ser un prefijo especialmente definido a tal efecto (64:ff9b::*n*, llamado Prefijo *Well-Known*) [2] o un prefijo local asignado manualmente dentro del bloque de direcciones unicast IPv6 con las que cuenta un dominio para ese fin. En cualquier caso, el mapeo es estable a lo largo del tiempo dado que no existe la necesidad de re-utilizar direcciones IPv6, dado que el repositorio de direcciones disponibles es suficientemente grande. Si el prefijo *Well-Known* es utilizado, la representación de direcciones IPv4 en IPv6 toma sentido a nivel global. De esta manera, cualquier receptor en Internet que reciba esa dirección, es capaz de interpretar que se trata de una traducción de una dirección IPv4 a IPv6 e incluso comunicar con esa dirección IPv4 en presencia de un servicio NAT64 local. El uso de un prefijo conocido es recomendado en caso de que no exista uno configurado manualmente.

El repositorio de direcciones IPv4 es típicamente un bloque pequeño de direcciones asignadas a la interfaz externa del NAT64. Debido al tamaño del espacio de direcciones IPv4, el repositorio de direcciones IPv4 del NAT64 no es suficiente para realizar un mapeo uno a uno con las direcciones IPv6. Por tanto, los mapeos utilizando direcciones IPv4 son creados y liberados dinámicamente.

Un equipo IPv6 que inicie la comunicación aprenderá la dirección IPv6 que representa la dirección IPv4 final por medio de DNS64, como se describe en la siguiente sección, o por otro medio.

Los paquetes enviados por el equipo IPv6 origen son interceptados por el dispositivo NAT64. El NAT64 asocia una dirección de transporte IPv4 de su repositorio a la dirección IPv6 transporte del nodo interno, guardando estado de la asociación, de tal manera que los paquetes de respuesta son traducidos de vuelta al dominio IPv6 interno y reenviados al nodo inicial. El estado de la asociación se mantiene mientras exista un flujo de paquetes. Una vez el flujo se detiene, y tras la expiración de un temporizador, la dirección de transporte pública (IPv4) es devuelta al repositorio de direcciones disponibles.

## 4. DNS64

DNS64 es una pasarela de nivel de aplicación para el protocolo DNS que genera registros de tipo AAAA (AAAA RRs) a partir de registros A (A RRs). DNS64 permite a equipos que utilizan únicamente IPv6 utilizar nombres de dominio de equipos IPv4 para iniciar la comunicación.

Cuando un equipo IPv6 inicia la comunicación, realiza directamente una consulta por un registro AAAA con el fin de obtener la dirección IPv6 de destino. Para permitir a un equipo

IPv6 iniciando la comunicación obtener la dirección del destino, DNS64 es utilizado para crear un registro AAAA a partir de un registro tipo A (que contiene la dirección IPv4 real donde el destino puede ser contactado). DNS64 está diseñado como una funcionalidad adicional de un DNS resolver recursivo. Como tal, cuando un servidor DNS64 recibe una petición por un registro AAAA generada por un iniciador IPv6, busca un registro tipo AAAA. Si el registro AAAA no existe para el nodo contactado (que es el caso habitual con nodos que utilicen sólo IPv4), el DNS64 realiza una búsqueda del registro de recurso tipo A. Si un registro A es encontrado, DNS64 crea un registro AAAA sintético añadiendo el prefijo Pref64::*n* del NAT64 a la dirección IPv4 del nodo a contactar (y en caso de que *n* es menor que 96, adicionalmente un sufijo, al igual que para NAT64). El registro AAAA sintético es devuelto al nodo IPv6 origen, el cual inicia una comunicación IPv6 con la dirección IPv6 asociada a la dirección IPv4 destino.

El paquete es enrutado hacia el NAT64 local, el cual realiza el mapeo de direcciones, tanto origen como destino, IPv6-a-IPv4 descrito anteriormente. Es importante observar que el DNS64 y el NAT64 no comparten ninguna información de estado. En particular, cuando DNS64 genera una respuesta sintética, no se guarda ningún tipo de estado en el NAT64. La única información compartida en ambos es el prefijo Pref64::*n*, que es definido por dominio. Por defecto, ambos NAT64 y DNS64 utiliza el prefijo *Well-Known* mencionado con anterioridad, por lo que no es necesaria una configuración manual en ninguno de los dos.

## 5. RECORRIDO POR NAT64/DNS64

Para este recorrido consideramos la topología de la Fig. 2. El NAT64 utiliza el prefijo *Well-Known* 64:ff9b::*n*/96 para mapear direcciones IPv4 a IPv6 y tiene asignada la dirección T a su interfaz IPv4 hacia la red exterior. El servidor DNS local implementa la funcionalidad DNS64 y utiliza el prefijo *Well-Known* para la síntesis de los registros AAAA. Los equipos IPv6 realizan búsquedas recursivas contra el servidor local de DNS.

A continuación describimos como H1 inicia la comunicación con H2:

1. H1 realiza una búsqueda DNS para la dirección IPv6 de H2 enviando una petición DNS de un registro AAAA al servidor DNS/DNS64 local.
2. El servidor local de DNS/DNS64 resuelve la petición, y descubre que no existe registro tipo AAAA para H2.

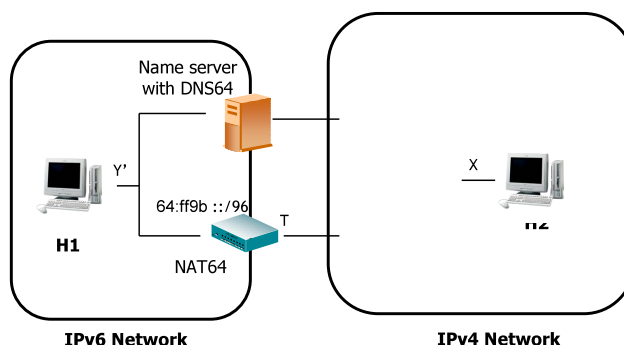


Figura 2. Escenario del recorrido

3. El servidor DNS/DNS64 busca un registro A para H2, obteniendo la dirección IPv4 X.

4. El servidor DNS/DNS64 sintetiza un registro AAAA añadiendo el prefijo 64:ff9b::/96 a la dirección IPv4 X, e incluye la dirección en la respuesta a H1.

5. Después de recibir el registro sintético AAAA, H1 manda un paquete hacia H2 desde la dirección de transporte origen (Y', y) a la dirección de transporte destino (64:ff9b::X, x), donde y y x son puertos elegidos por H2.

6. El paquete se enruta hacia el interfaz IPv6 del NAT64 (dado que 64:ff9b::/96 ha sido asociado a esta interfaz), y el NAT64 realiza las siguientes acciones:

- Selecciona un puerto libre t y crea una entrada (Y', y) <-> (T, t)
- Traduce la cabecera IPv6 en la cabecera IPv4 utilizando traducción sin estado.
- Incluye en el paquete (T, t) como dirección de transporte origen y (X, x) como dirección de transporte destino.
- El NAT64 envía el paquete a la red IPv4.

7. El nodo H2 recibe el paquete y contesta enviando un paquete con destino la dirección de transporte (T, t) y la dirección de transporte (X,x) como origen.

8. El paquete se enruta hacia el NAT64 a través de la red IPv4. El NAT64 busca la entrada conteniendo (T,t). Cuando la entrada se encuentra,

- El NAT64 traduce el paquete IPv4 a un paquete IPv6 usada la traducción sin estado.
- El NAT64 incluye en el paquete la dirección de transporte (Y',y) como dirección de origen y (Pref64:X, x) como dirección transporte destino.

El paquete traducido se reenvía finalmente hacia H1.

## 6. CONCLUSIONES

NAT64 es un traductor de direcciones de red y protocolos que permite la comunicación entre nodos IPv6 e IPv4. DNS64 es una pasarela a nivel de aplicación que sintetiza registros tipo AAAA a partir de la información de registros tipo A para un nombre de dominio dado. Se espera que estas herramientas jueguen un papel crucial en la transición hacia IPv6 en el futuro. Están ya disponibles varias implementaciones comerciales y de código libre de estas herramientas.

Se soportan varios modelos de despliegue aunque es de esperar que dos de ellos predominen: El primero permite que nodos de una red final únicamente con IPv6 sean capaces de comunicarse con direcciones IPv4 de Internet. En este caso la funcionalidad NAT64/DNS64 puede ser llevada a cabo en la propia red IPv6 o por su proveedor directo, por ejemplo en un Carrier Grade NAT. En el otro escenario, un dominio final sólo con IPv4 sirve a clientes en Internet IPv6. En este caso, la funcionalidad NAT64 puede ser provista por la red final IPv4, y el servicio DNS64 no es necesario ya que el servidor DNS del dominio IPv4 tiene autoridad para gestionar la información local.

## 7. Referencias

- [1] E. Nordmark, "Stateless IP/ICMP Translation Algorithm (SIIT)", RFC2765, 2000.
- [2] C. Bao, C. Huitema, M. Bagnulo, M. Boucadair, X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC6052, 2010.
- [3] T. Hain, "Architectural Implications of NAT", RFC2993, 2000.
- [4] M. Bagnulo, P. Matthews, I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC6146, July 2010.
- [5] M. Bagnulo, A. Sullivan, P. Matthews, I. van Beijnum, "DNS64: DNS extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", RFC6147, July 2010.
- [6] VJ. Rosenberg, R. Mahy, P. Matthews, D. Wing. "Session Traversal Utilities for NAT (STUN)". RFC5389, 2008.
- [7] K. Egevang, P. Francis. "The IP Network Address Translator (NAT)". RFC1631, 1994.
- [8] S. Guha, Ed., K. Biswas, B. Ford, S. Sivakumar, P. Srisuresh. "NAT Behavioral Requirements for TCP". RFC5382, 2008.
- [9] F. Audet, Ed., C. Jennings. "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP". RFC4787, 2007.

- [10] P. Srisuresh, B. Ford, S. Sivakumar, S. Guha. "NAT Behavioral Requirements for ICMP". RFC5508, 2009.
- [11] X. Li, C. Bao, F. Baker, "IP/ICMP Translation Algorithm", RFC6144, 2010.