



Desarrollo de Aplicaciones con soporte IPv6

Ing. Azael Fernández Alcántara
azael@ipv6.unam.mx



Universidad Nacional Autónoma de México, UNAM
Grupo de Trabajo de IPv6 en Internet2
Capítulo Mexicano del Foro IPv6
NETLab



Reunión de Otoño 2003
3 de octubre 2003
Cd. de Puebla, México.

Grupo de trabajo de IPv6 en **cudi**



AGENDA



1. Transición de IPv4 a IPv6
2. Conversión de aplicaciones para IPv6.
3. Referencias.

AGENDA

1. Transición de IPv4 a IPv6

- Tipos de nodos
- Mecanismos de transición
- Impacto de la transición en Capas Superiores.
- Traductores

TIPOS DE NODOS

- **Nodo IPv4:**
 - Solo tiene asignadas direcciones IPv4.
 - Este nodo no soporta IPv6.
 - Solo se pueden usar aplicaciones habilitadas para IPv4.
- **Nodo IPv6:**
 - Solo tiene asignadas direcciones IPv6.
 - Este nodo es capaz de comunicarse solamente con nodos y aplicaciones IPv6.
 - Solo se pueden usar aplicaciones habilitadas para IPv6.

TIPOS DE NODOS

- **Nodo IPv6/IPv4:**
 - Este nodo tiene las implementaciones tanto para IPv4 como para IPv6.
 - IPv6 habilitado, solo si este ha sido configurado.
 - Se pueden usar aplicaciones para IPv4 e IPv6.

MECÁNICISMOS DE TRANSICIÓN

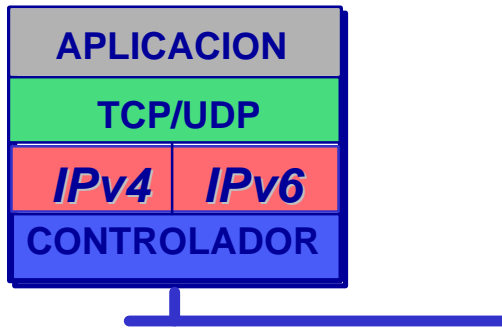
Dos principales:

- **Capa IP dual:** Los ruteadores y hosts soportan IPv4 y IPv6 simultaneamente.
- **Túneles de IPv6 sobre IPv4:** Los paquetes IPv6 se encapsulan con encabezados de IPv4 para transportarse por redes de IPv4. Existen dos tipo de de tuneles: configurados (manuales) y automáticos.

NODO DUAL STACK IPv6/IPv4

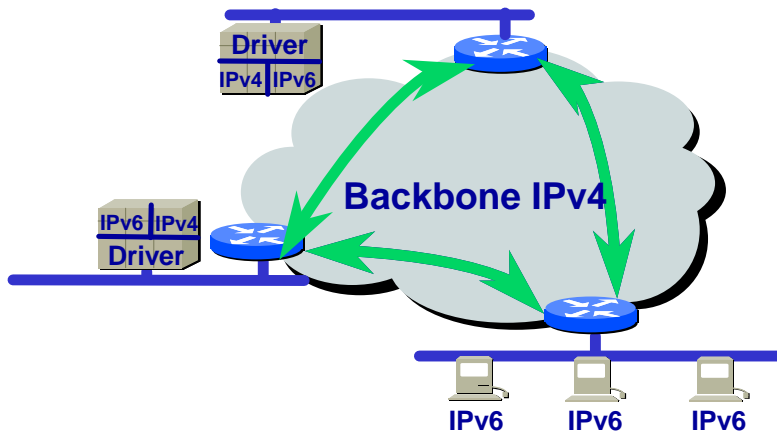
- Incluye librerías del resolver capaces de trabajar con registros A y AAAA/A6
- Cuando pregunta al DNS para un nodo dual, el orden de las respuestas normalmente definirá el protocolo usado (IPv6 antes).
- Las aplicaciones usan IPv6 o IPv4 dependiendo de las respuestas recibidas y su orden.

MECÁNISMOS DE TRANSICIÓN



< Capa IP dual

v Tuneles de IPv6 sobre IPv4



DATOS	Encabezado Capa de Transporte	Encabezado IPv6
-------	-------------------------------	-----------------

DATOS	Encabezado Capa de Transporte	Encabezado IPv6	Encabezado IPv4
-------	-------------------------------	-----------------	-----------------

IMPACTO DE LA TRANSICIÓN EN CAPAS SUPERIORES

- La arquitectura de red TCP/IP no está perfectamente dividida.
- Las aplicaciones identifican al nodo destino:
 - Usando la dirección IP.
 - Usando el nombre DNS.
- Las aplicaciones deben ser revisadas en ambos casos:
 - IPv6 maneja otro formato.
 - Cambia la interfaz de la capa de transporte.

IMPACTO DE LA TRANSICIÓN EN CAPAS SUPERIORES

- Durante la transición será necesario soportar tanto los nodos de IPv4 como los de IPv6.
- Se requieren traductores entre la interfaz de red IPv6 y la interfaz de programación IPv4.

TRADUCTORES

- **De capa de Red:**
 - SIIT (Stateless IP/ICMP Translator)
 - NAT-PT (Network Address Translation - Protocol Translation)
 - BIS (Bump in the Stack)
 - MBIS (Extensiones Multicast para BIS)
- **De capa de Transporte:**
 - TRT (Transport Relay Translator).
- **De capa de Aplicación:**
 - BIA (Bump in the API).

Usando una aplicación de IPv4

- Pueden conectarse nodos IPv4 o los duales.
- Los nodos dual Stack pueden conectarse usando la red IPv6
- Los nodos IPv6 no pueden usar aplicaciones IPv4 (si es posible con un traductor).

Usando aplicaciones IPv6 e IPv4

- Los nodos IPv6 y los duales pueden conectarse usando la red IPv6.
- Una aplicación IPv6 puede usarse sobre la red IPv4:
 - Si se usa una dirección compatible con IPv4 (:::a.b.c.d)
- Un nodo IPv4 puede conectarse con un nodo IPv6 si usa un traductor o por túnel.



AGENDA

1. Transición de IPv4 a IPv6
- 2. Conversión de aplicaciones para IPv6.**
3. Referencias.



AGENDA



2. Conversión de aplicaciones para IPv6

- Escenarios
- Consideraciones y Cambios
- Herramientas
- Recomendaciones

3. Referencias

- **Convirtiendo las redes existentes:**
 - Aplicaciones solamente para IPv4.
 - Proveer dos aplicaciones diferentes.
 - Aplicaciones duales (IPv4 e IPv6).

- **Dando de alta redes nuevas de IPv6:**
 - Aplicaciones duales (IPv4 e IPv6).
 - Las aplicaciones pueden ser solo para IPv6.
 - Si son independientes del protocolo

ESCENARIOS

- Usando una aplicación existente de IPv4:
 - Mediante traductores (NAT-PT , SIIT , BIS)
 - Válido solamente con limitaciones.
- Convirtiendo una aplicación existente:
 - Aplicable sólo si el código fuente está disponible.
 - Convirtiendo las librerías de comunicaciones.
 - Ejemplo: Java net library
- Desarrollando una nueva aplicación:

ESCENARIOS

- Desarrollando una nueva aplicación:
 - Independiente del protocolo .
 - Dependiente del protocolo.
 - No recomendado.
 - Desarrollar un código dual IPv4/IPv6.

CONSIDERACIONES

- Los códigos fuente y binario deden ser compatibles con los códigos existentes y las aplicaciones:
 - Los binarios existentes (IPv4) seguirán ejecutándose.
- Cambios mínimos en la API (<0,1%).
 - La conversión a IPv6 debe ser sencilla.
 - Mismas llamadas de sockets.
 - Pocas nuevas funciones.
 - Localizables en el código.

CONSIDERACIONES

- Del lado del servidor:
 - Cambiar las funciones “socket”
 - Ajustar la función de registro para manejar direcciones IP más grandes.
 - Incrementar todos los datos de los miembros que guarden direcciones IP (BD).
- Del lado del cliente:
 - Cambiar las funciones “socket”
 - Ajustar las funciones de registro.
 - Ajustar la función interfaz del teclado y de despliegue para manejar direcciones IP más grandes.

CONSIDERACIONES

- Algunas aplicaciones usan los dos puntos “.” para distinguir el puerto de la dirección.
 - Ejemplo: En los URLs.
 - En IPv6 las direcciones IPv6 se representan con paréntesis cuadrados:
 - [http://\[3ffe:8070::1\]/index.html](http://[3ffe:8070::1]/index.html)

CONSIDERACIONES

- Dependencias en la aplicación.
 - Porciones del código no afectadas
 - Porciones del código afectadas
- Naturaleza de la aplicación.
- Espacio de la aplicación.
- Arquitectura.
- No se ve afectada la secuencia de código típica.

Secuencia de Código Típica (IPv4 IPv6)

- Del lado del servidor:
 - socket – se abre un socket
 - bind - de la dirección local al socket
 - listen – se escucha en un puerto
 - accept – espera conexiones
 - “read” y/o “write” si es TCP
 - “recvfrom” y/o “sendto” si es UDP
- Del lado del cliente:
 - socket - se abre un socket
 - connect – se conecta al servidor
 - “read” y/o “write” si es TCP
 - “recvfrom” y/o “sendto” si es UDP

Cambios requeridos en la API

- A través de los Sockets.
- En las partes de la API donde se muestre el tamaño de la dirección IP. (se requieren nuevas estructuras de datos).
- En las partes de aplicación que manipule la dirección IP.

Cambios requeridos en la API

- Funciones socket() del núcleo.
- Estructuras de datos para direcciones.
- Funciones de traducción de Nombre –Dirección.
- Funciones de conversión de direcciones.

Cambios requeridos en la API

- Funciones socket() del núcleo.
 - En IPv4
s = socket (PF_INET, SOCK_STREAM, 0);
 - En IPv6
s = socket (PF_INET6, SOCK_STREAM, 0);

PF (Familia del Protocolo)

Cambios requeridos en la API

- Funciones socket() del núcleo.
 - Longitud de dirección.
 - Espacio para nuevos campos en la cabecera.
 - Mecanismos para poner nuevos valores de campo:
 - Determinar la clase de trafico (QoS).
 - Poner opciones de seguridad (AH y ESP).
 - Requerimientos de espacio y memoria.

Cambios requeridos en la API

- Estructuras de datos para direcciones.
 - Nueva Familia de Dirección **AF_INET6**.
 - **ssockaddr_in** para IPv4
 - **sockaddr_in6** de 128 bits para IPv6
 - **sockaddr_storage** independiente del protocolo

 - sin_port ✍ sin6_port
 - sin_family ✍ sin6_family

Cambios requeridos en la API

- Funciones de traducción de Nombre –Dirección.
 - En IPv4
 - gethosbyname () y gethosbyaddr ()**
 - En IPv6
 - getipnodebyname() y getipnodebyaddr()**
- La norma POSIX 1003.g especifica funciones independientes del protocolo.
 - Nuevas funciones: **getnameinfo() getaddrinfo()**

Cambios requeridos en la API

- Funciones de conversión de direcciones.
 - En IPv4:
 - Cadena -> Binario `inet_aton ()` y `inet_addr ()`
 - Binario -> Cadena `inet_ntoa ()`
 - En IPv6 e IPv4:
 - Cadena -> Binario `inet_pton ()`
 - Binario -> Cadena `inet_nton ()`

Cambios requeridos en la API

	IPv4	IPv6
Estructuras de Datos	AF_INET	AF_INET6
	in_addr sockaddr_in	in6_addr sockaddr_in6
Funciones de Conversión de Direcciones	inet_aton() inet_addr()	inet_pton()
	inet_ntoa()	inet_ntop()
Funciones Nombre a Dirección	gethostbyname() gethostbyaddr()	getipnodebyname() getipnodebyaddr()
	getnameinfo() getaddrinfo()	getnameinfo() getaddrinfo()

- Algunas disponibles:
 - Socket Scrubber de Sun
 - Socks v4 a v6
 - Checkv4 de Microsoft
- Ayudan a encontrar e identificar las líneas de código (fuente) que requieren cambiarse o actualizarse.

RECOMENDACIONES

- Desarrollar aplicaciones independientes de la familia de direcciones:
 - La mejor manera de conversión para tener la mayor portabilidad posible.
 - Esconder el código dependiente del protocolo mediante el uso de las funciones:
getnameinfo() y getaddrinfo()
- Habilitar la aplicación para usar las características de IPv6.



AGENDA



1. Transición de IPv4 a IPv6
2. Conversión de aplicaciones para IPv6.
3. Referencias.



3. Referencias



PAGINAS WEB

Windows: <http://www.research.microsoft.com/msripv6/>

Solaris: <http://www.sun.com/solaris/ipv6>

Linux: <http://www.bieringer.de/linux/IPv6>

3Com: <http://www.3com.com/nsc/ipv6.html>

Nortel: <http://www.nortelnetworks.com>

Cisco: <http://www.cisco.com/go/ipv6>

- Base de Datos de Aplicaciones y Parches IPv6:

http://6net.iif.hu/ipv6_apps

- <http://playground.sun.com/pub/ipng/html/ipng-implementations.html>



PAGINAS WEB

Otros Sitios:

<http://www.6bone.net>

<http://www.ipv6forum.com>

<http://www.kame.net/>

DOCUMENTOS

- Sun's porting guide:

<http://www.sun.com/software/solaris/ipv6>

- Porting IPv4 applications to IPv6:

[http://uw7doc.sco.com/SDK_netapi/sockC.PortIPv4app
IPv6.html](http://uw7doc.sco.com/SDK_netapi/sockC.PortIPv4appIPv6.html)

RFC

- (2373) IP Version 6 Addressing Architecture.
- (2893) Transition mechanisms for IPv6 Hosts and Routers.
- (3484) Default Address Selection for IPv6.
- (2732) Format for Literal IPv6 Addresses in URL.

RFC

- (3493) Basic Socket Interface Extensions for IPv6. substituye (RFC 2553)
- (3542) Advanced Sockets API for IPv6. substituye (RFC 2292)
- (2767) Dual Stack Hosts using the Bump-In-The-Stack technique (BIS).
- (3022) Traditional IP Network Address Translator (Traditional NAT).
- (2766) Network Address Translation – Protocol Translation (NAT-PT).
- (2765) Stateless IP/ICMP Translator (SIIT)

- Basic Socket Interface Extensions for IPv6. <draft-ietf-ipngwgRFC2553bis-05> 02/02
- Advanced Sockets API for IPv6. < draft-ietf-ipngwg-rfc2292bis-09.txt > Marzo 2003.
- Dual Stack Hosts using Bump-In-The-API (BIA). <draft-ietfngtrans-bia-01.txt>.
- Application Aspects of IPv6 Transition. <draft-shinngtransapplication-transition-00.txt> 05/02
- Dual Stack Transition Mechanism (DSTM). <draft-ietf-ngtransdstm-08.txt>

LIBROS

- **IPv6 Essentials** Silvia Hagen O'Reilly & Associates
Primera edición (Julio 2002)
- **Understanding IPv6** Joseph Davies Microsoft Press;
(Noviembre 2002)
- **Migrating to IPv6** Marc Blanchet John Wiley & Sons;
Primera edición (Noviembre 2002)
- **Programming IPv6** Sean E. Walton, Addison Wesley
Professional Primera edición (Noviembre 2002)
- **Linux Socket Programming** Sean Walton Sams Primera
edición (Enero 2001)

LIBROS

- **IPv6: The New Internet Protocol**, by Christian Huitema, Prentice Hall, 1997.
- **IPv6 Clearly Explained**, Pete Loshin, AP Professional, 1999.
- **IPv6 Networks**, Marcus Goncalves, Kitty Niles, McGraw-Hill, 1998.
- **Implementing IPv6**, Mark A. Miller, IDG Books, 1998 (2nd edition Julio 1999)
- **IP Addressing and Subnetting, Including IPv6**, Syngress Media, Octubre 1999.
- **Understanding IPv6 Addressing**, Peter H. Salus, AP Professional, 1999.

LIBROS

- **Internetworking IPv6 With Cisco Routers**, Silvano Gai, McGraw-Hill, 1998.
- **IPv6 : The Next Generation Internet Protocol**, Digital Press, 1997.
- **TCP/IP : Architecture, Protocols, and Implementation With IPv6 and IP Security**, Sidnie Feit, McGraw-Hill, 1998.
- **IPng and the TCP/IP Protocols**, Stephan Thomas, Wiley, 1996.
- **Advanced Internet Technologies**, Uyles Black, Prentice-Hall, 1999, (includes VoIP (H.323), IP Multicast, RSVP, RTP/RTCP, IPv6, Mobile IP, and others).



GRACIAS

A: Rogelio Morales

azael@ipv6.unam.mx

staff_ipv6@ipv6.unam.mx

Tel. (+52) 55 56 22 88 57