

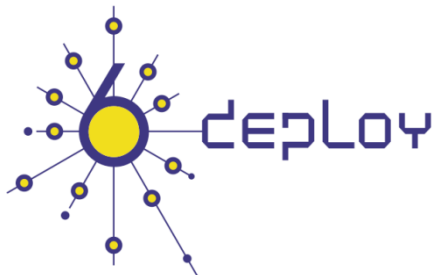
# IPv6 Startup

**Sta. Cruz, Bolivia**

**Septiembre, 2009**

**Jordi Palet, Consulintel**

**([jordi.palet@consulintel.es](mailto:jordi.palet@consulintel.es))**



# Agenda

1. Instalación de IPv6 en varias plataformas (Windows 2000/XP/2003/Vista/7, Linux, BSD, Mac OS X)
2. Configuración básica Stateless/Stateful, privacidad, rutas estáticas
3. Configuración de mecanismos de transición
4. Ejemplos de aplicaciones varias

# Parte 1

## Instalación de IPv6 en varias plataformas (2000/XP/2003/Vista/7, Linux, BSD, Mac OS X)

# IPv6 en Windows

- Soporte completo
  - Windows 7, Vista, XP SP1 y posteriores
  - Windows Server 2003 y 2008
- Technology preview
  - Windows XP sin SP
  - Windows 2000 (no compatible con SP2 o posteriores)
- Developer edition
  - Windows NT 4.0
- Productos de terceros pero sin soporte oficial
  - Windows 95/98/ME
  - Windows 2000 con SP2 y posteriores
- Características soportadas (últimas versiones)
  - Autoconfiguración, túneles 6in4, túneles 6to4, relay 6to4, túneles TEREDO, túneles ISATAP, IPsec (llaves manuales)



# Instalación de IPv6: 2000 (1)

- Pila no comercial (originalmente desarrollada por Microsoft Research)
- Descargar el “Microsoft IPv6 Technology Preview for Windows 2000”
  - Disponible en <http://www.ipv6tf.org/using/connectivity/guides.php?cid=1>
  - Téngase en cuenta que esta pila no tiene soporte comercial por parte de Microsoft
- Procedimiento de Instalación
  - **Se requieren permisos de administración**
  - Extraer los ficheros “IPv6 Technology Preview”
  - Seguir el procedimiento apropiado al SPn y IE6 fixed.txt para modificar /setup/hotfix.ini
  - Ejecutar setup.exe o hotfix.exe
  - Desde el escritorio ejecutar Inicio, Parámetros, Network y Dial-up Connections. Alternativamente, click-derecho sobre Entorno de Red y propiedades
  - Click-derecho sobre la conexión LAN a la que se quiere agregar IPv6, Propiedades, Instalar, componente de red, añadir y “Microsoft IPv6 Protocol”
- En una ventana DOS
  - **ipv6 if** para verificar que IPv6 ha sido instalado

# Instalación de IPv6: 2000 (2)

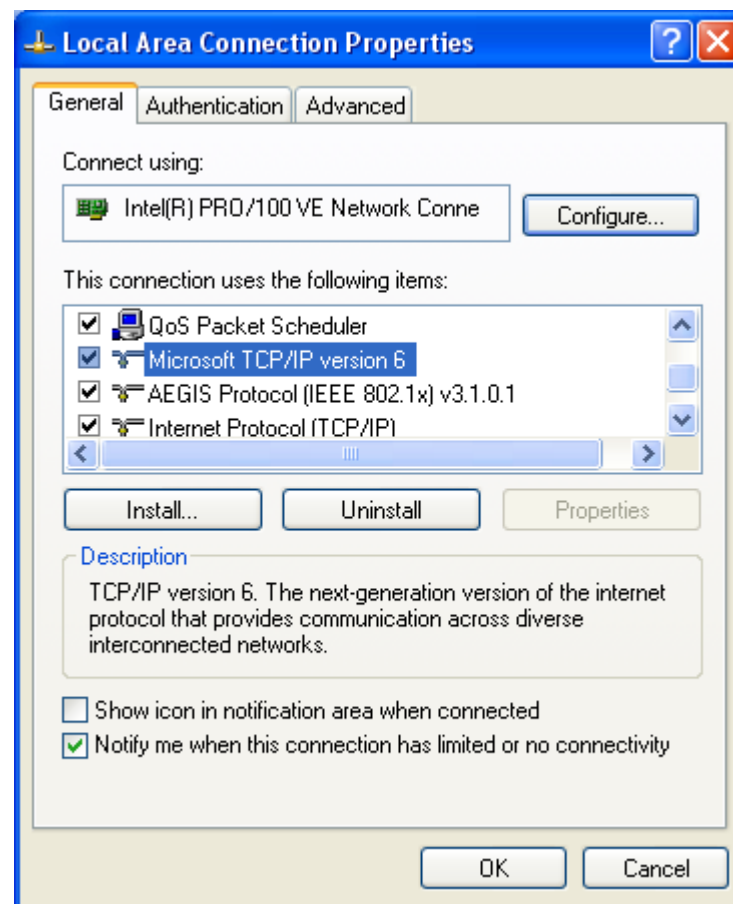
- Procedimiento de desinstalación
  - **Requiere permisos de administrador**
  - Desde el escritorio, ejecutar Inicio, Parámetros, Network y Dial-up Connections. Alternativamente, click-derecho en Entorno de Red y Propiedades
  - Click-derecho en la conexión de red donde se desea eliminar IPv6, y entonces Propiedades, IPv6, desinstalar
  - Reiniciar
- En una ventana DOS
  - **ipv6 if** para verificar si IPv6 ha sido desinstalado

# Instalación de IPv6: XP/2003 (1)

- En una ventana DOS
  - **ipv6 install** Instala IPv6 como protocolo de Red
  - **ipconfig** o **ipv6 if** para verificar si esta instalado IPv6

# Instalación de IPv6: XP/2003 (2)

- Otra opción para verificar si esta instalado IPv6
  - Network Connections > Local Area Coneccion > Properties
- También se puede instalar/desinstalar desde aquí





# Instalación de IPv6: XP/2003 (3)

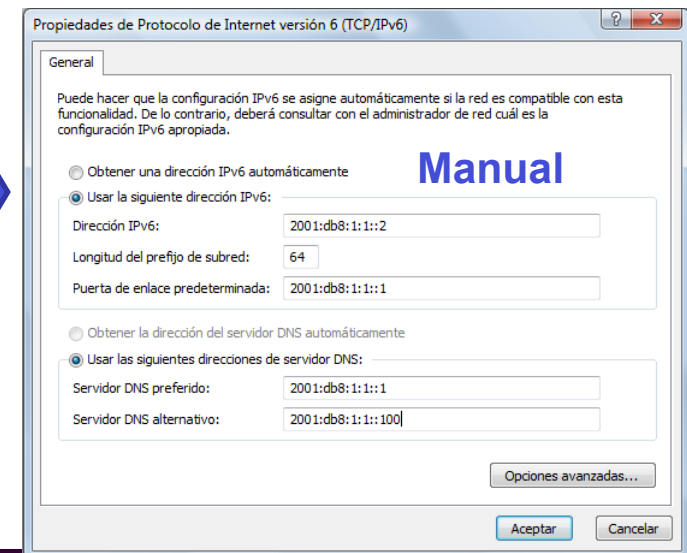
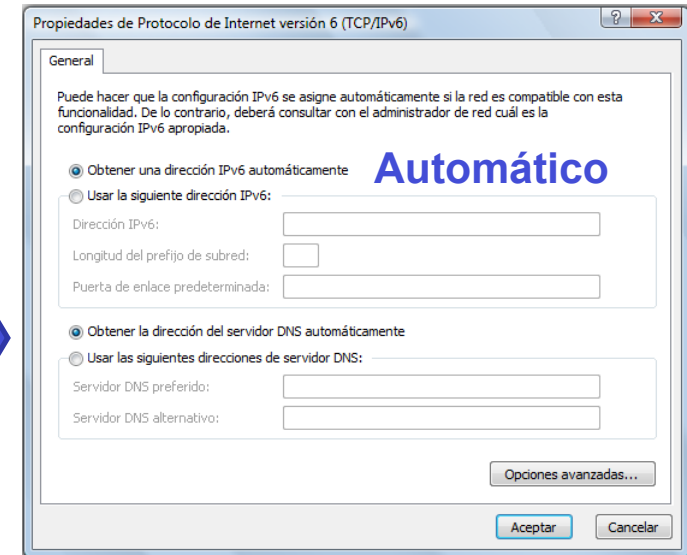
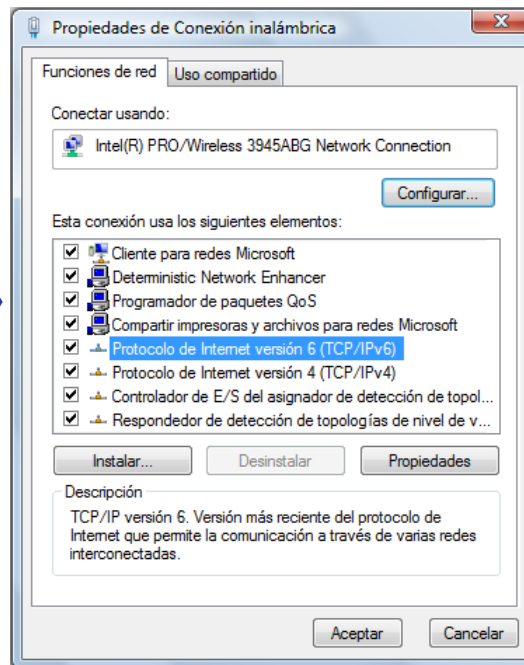
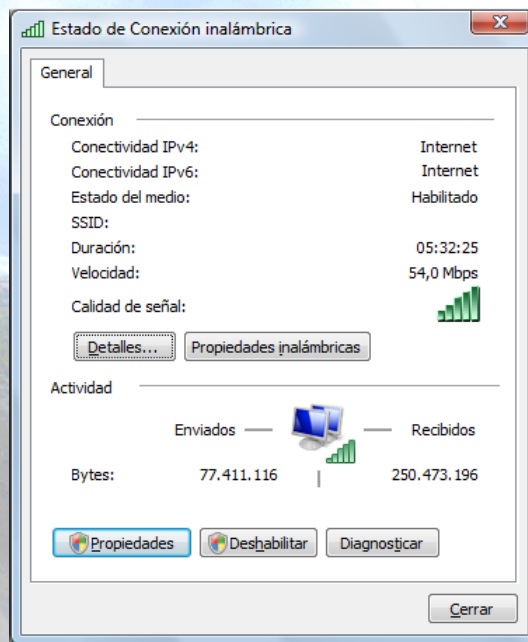
- Desinstalar
  - **ipv6 uninstall** Quita IPv6 como protocolo de Red
  - **ipconfig** o **ipv6 if** para verificar si esta desinstalado IPv6

# Instalación de IPv6: Vista (1)

- ¡No es necesario hacer nada!
  - IPv6 está instalado y habilitado por defecto ☺
  - La configuración esta basada en entorno gráfico y comandos netsh
- Otras características nuevas
  - Soporte completo IPsec
  - MLDv2
  - Link-Local Multicast Name Resolution (LLMNR)
    - No requiere un servidor DNS. Los nodos IPv6 en un segmento piden el nombre a una dirección IPv6 multicast. Similar al funcionamiento de NetBIOS.
  - Soporte de direcciones IPv6 en URLs
  - IPv6 Control Protocol (IPV6CP - RFC 5072)
  - IPv6 sobre PPP
  - DHCPv6, en el cliente y el servidor
  - Identificador de Interface aleatorio por defecto (RFC 3041)
  - Teredo soporta NATs simétricos
    - Activo por defecto. Solo se utiliza si la aplicación requiere soporte IPv6 y no esta disponible de forma nativa.

# Instalación de IPv6: Vista (2)

- Configuración en entorno gráfico



Deseleccionar la casilla para deshabilitar el protocolo por interfaz

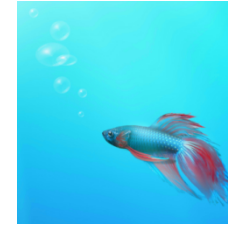
# Instalación de IPv6: Vista (3)

- Desinstalación
  - No puede ser desactivado ya que esta completamente integrado con IPv4
- Puede ser desactivado para una interfaz de red concreta
  - A través del GUI
- El uso de comandos netsh es posible. Requiere DOS con permisos de administrador
- Algunos componentes de IPv6 pueden ser definidos a través del registro
  - <http://technet.microsoft.com/en-us/library/bb878057.aspx>
  - Crear registro (tipo DWORD) : HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\tcpip\Parameters\DisabledComponents
  - El valor de DisabledComponents es una mascara de bits que controla los siguientes “flags”, empezando por el bit mas bajo (Bit 0 = activado, en todos los casos por defecto):
  - Bit 0 → 1 para desactivar IPv6 en todas las interfaces de túnel, incluyendo ISATAP, 6to4, y Teredo.
  - Bit 1 → 1 para desactivar todos los túneles 6to4.
  - Bit 2 → 1 para desactivar todos los túneles ISATAP.
  - Bit 3 → 1 para desactivar todos los túneles Teredo.
  - Bit 4 → 1 para desactivar IPv6 en todas las interfaces de tipo “no-tunnel”, incluyendo LAN, WLAN, PPP, etc.
  - Bit 5 → 1 para modificar la tabla de políticas de prefijos para que se prefiera IPv4 o IPv6. Default value is 0
  - Ejemplos de valores **DisabledComponents** para desactivar algunos componentes:

• Disable all tunnel interfaces	0x1
• Disable 6to4	0x2
• Disable ISATAP	0x4
• Disable Teredo	0x8
• Disable Teredo and 6to4	0xA
• Disable all LAN and PPP interfaces	0x10
• Disable all LAN, PPP, and tunnel interfaces	0x11
• Prefer IPv4 over IPv6	0x20
• Disable IPv6 over all interfaces and prefer IPv4 to IPv6	0xFF



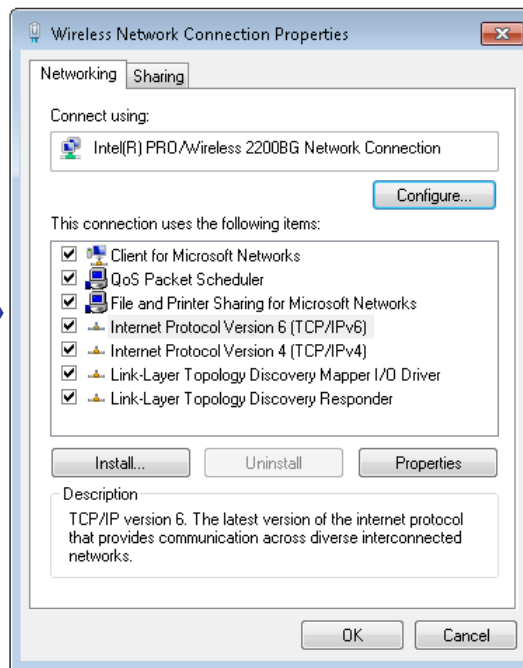
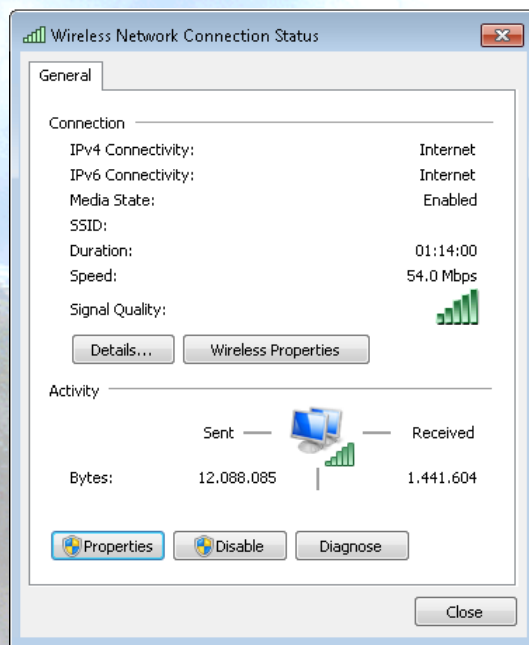
# Instalación de IPv6: 7 RC (1)



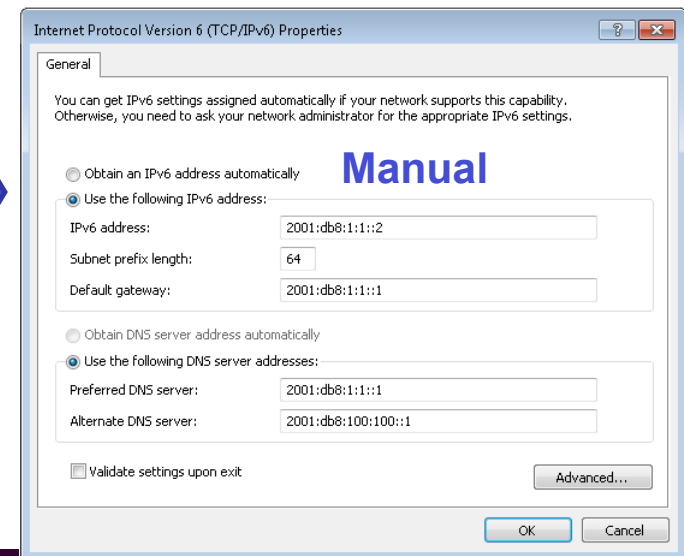
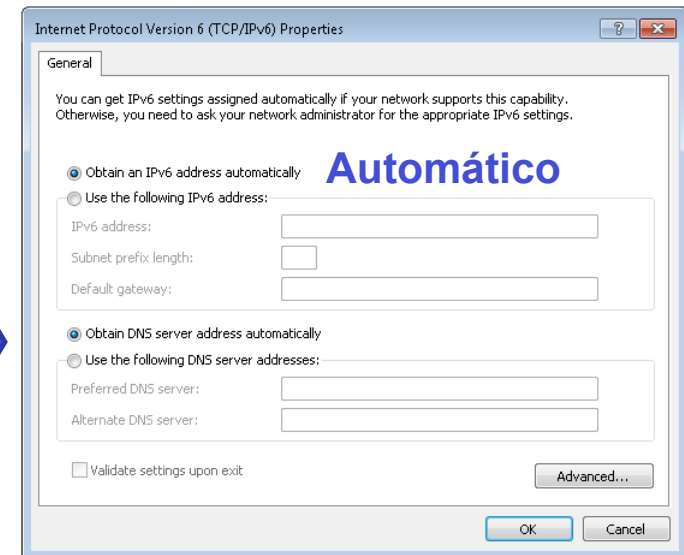
- Windows 7 Ultimate - Version RC 6.1.7100
- ¡No es necesario hacer nada!
  - IPv6 está instalado y habilitado por defecto ☺
  - La configuración esta basada en entorno gráfico y comandos netsh
- Soporte IPv6 similar al de Vista y Server 2008
  - IPsec, MLDv2, LLMNR, IPv6 en URLs, IPV6CP, IPv6 sobre PPP, DHCPv6, Teredo
  - Cambia: Identificador de Interface aleatorio por defecto (RFC 3041)
    - No usa EUI-64 por defecto para el identificador de interfaz en las direcciones autoconfiguradas.
      - netsh interface ipv6 set global [[randomizeidentifiers=]enabled|disabled]
- Pero con algunas nuevas mejoras
  - IP-HTTPS (IP over Secure HTTP)
    - permite a los hosts atravesar un servidor proxy o firewall y conectarse a redes privadas por medio de IPv6 dentro de un túnel HTTPS. HTTPS no provee seguridad a los datos, es necesario usar IPsec para dar seguridad a una conexión IP-HTTPS. Más información en <http://msdn.microsoft.com/en-us/library/dd358571.aspx>
  - DirectAccess
    - Permite a los usuarios conectarse de manera transparente a la red corporativa sin establecer específicamente una conexión VPN. También permite al administrador de red seguir en contacto con los host móviles fuera de la oficina, y poder hacer actualizaciones y dar soporte a dichos equipos. Se una arquitectura donde un cliente IPv6 se comunica con un servidor IPv6 en la red corporativa. También se pueden usar conexiones desde Internet IPv4 empleando 6to4, Teredo e ISATAP. También se puede usar IP-HTTPS. DirectAccess usa túneles IPsec para proveer seguridad a la autenticación y al acceso de recursos.
    - El cliente puede ser un Windows 7 o Server 2008. El servidor puede ser un Server 2008.

# Instalación de IPv6: 7 RC (2)

- Configuración en entorno gráfico



Deseleccionar la casilla para deshabilitar el protocolo por interfaz



# Instalación de IPv6: 7 RC (3)

- Desinstalación
  - No puede ser desactivado ya que esta completamente integrado con IPv4
- Puede ser desactivado para una interfaz de red concreta
  - A través del GUI
- El uso de comandos netsh es posible. Requiere DOS con permisos de administrador
- También como en Vista algunos componentes de IPv6 pueden ser definidos a través del registro
  - <http://technet.microsoft.com/en-us/library/bb878057.aspx>

# Instalación de IPv6: Linux (1)

- Soporte a partir de versión del kernel 2.4.x

- Comprobar si esta instalado:

```
#test -f /proc/net/if_inet6 && echo "Kernel actual soporta IPv6"
```

- Instalar módulo:

```
#modprobe ipv6
```

- Comprobar módulo:

```
#lsmod |grep -w 'ipv6' && echo "modulo IPv6 cargado"
```

- Carga/descarga automática del modulo (/etc/modules.conf o /etc/conf.modules ):

```
alias net-pf-10 ipv6 #habilita carga bajo demanda
```

```
alias net-pf-10 off #deshabilita carga bajo demanda
```



# Instalación de IPv6: Linux (2)

**# ifconfig** para verificar

```
eth0 Link encap:Ethernet HWaddr 00:E0:81:05:46:57
  inet addr:10.0.0.3 Bcast:10.0.0.255 Mask:255.255.255.0
  inet6 addr: fe80::2e0:81ff:fe05:4657/64 Scope:Link
  inet6 addr: 2001:800:40:2a05::3/64 Scope:Global
  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
  RX packets:2010563 errors:0 dropped:0 overruns:0 frame:0
  TX packets:1700527 errors:0 dropped:0 overruns:2 carrier:0
  collisions:0 txqueuelen:100
  RX bytes:205094215 (195.5 Mb) TX bytes:247063610 (235.6Mb)
  Interrupt:11 Base address:0xe000 Memory:f8201000-f8201038
lo Link encap:Local Loopback
  inet addr:127.0.0.1 Mask:255.0.0.0
  inet6 addr: ::1/128 Scope:Host
  UP LOOPBACK RUNNING MTU:16436 Metric:1
  RX packets:1675838 errors:0 dropped:0 overruns:0 frame:0
  TX packets:1675838 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:0
  RX bytes:659846244 (629.2 Mb) TX bytes:659846244 (629.2 Mb)
```

# Instalación de IPv6: Linux (3)

## Configuración permanente

- Red Hat (desde 7.1) y “clones”:

Añadir a `/etc/sysconfig/network`:

```
NETWORKING_IPV6=yes
```

Reiniciar la red:

```
# service network restart
```

○

```
#/etc/init.d/network restart
```

- SUSE:

Añadir en `/etc/sysconfig/network/ifcfg-<Interface-Name>`:

```
SUSE 8.0: IP6ADDR="<ipv6-address>/<prefix>"
```

```
SUSE 8.1: IPADDR="<ipv6-address>/<prefix>"
```

# Instalación de IPv6: Linux (4)

## Configuración permanente

- Debian:

Con el módulo IPv6 cargado se edita `/etc/network/interfaces`, por ejemplo:

```
iface eth0 inet6 static
    pre-up modprobe ipv6
    address 3ffe:ffff:1234:5::1:1
    # Elimina completamente la autoconfiguración:
    # up echo 0 > /proc/sys/net/ipv6/conf/all/autoconf
    netmask 64
    # El router está autoconfigurado y no tiene dirección fija.
    # Se encuentra gracias a
    # (/proc/sys/net/ipv6/conf/all/accept_ra).
    # Si no habrá que configurar el GW:
    # gateway 3ffe:ffff:1234:5::1
```

- Se reinicia o:  
# ifup --force eth0

# Instalación de IPv6: Linux (5)

- Herramientas:

1. net-tools package

```
# /sbin/ifconfig -? 2>& 1|grep -qw 'inet6' && echo "'ifconfig' soporta IPv6"
```

```
# /sbin/route -? 2>& 1|grep -qw 'inet6' && echo "'route' soporta IPv6"
```

2. iproute package

```
# /sbin/ip 2>&1 |grep -qw 'inet6' && echo "'ip' soporta IPv6"
```

3. iputils package contiene ping6,  
traceroute6 y tracepath6

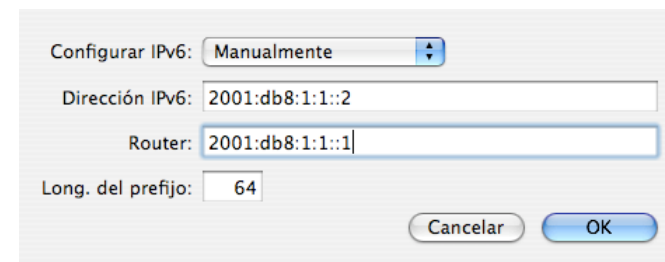
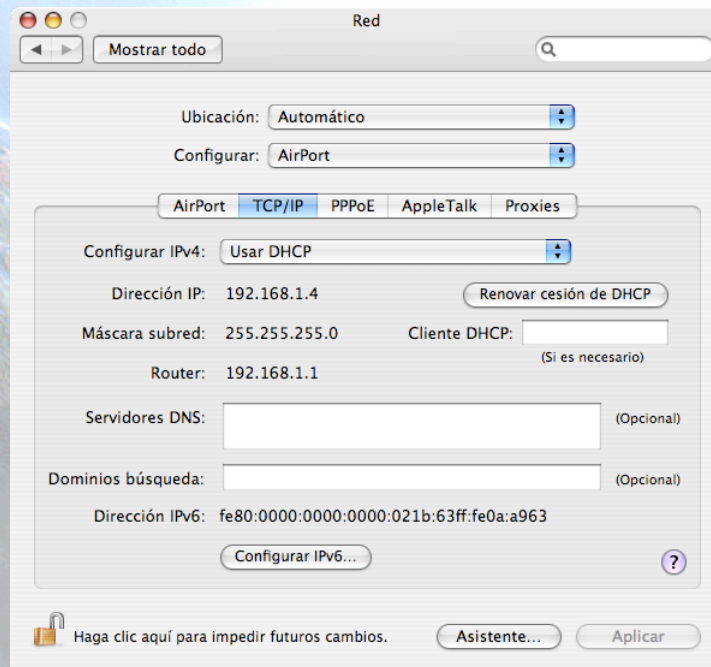


# Instalación de IPv6: BSD (1)

- Soporte a partir de versión 4.5+):
- Soporte muy bueno, la pila ya viene instalada

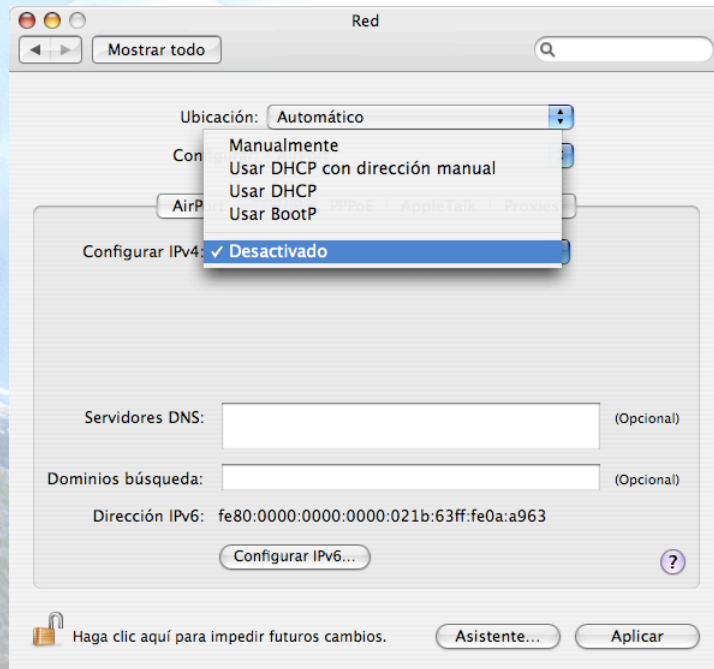
# Instalación de IPv6: Mac OS X (1)

- Soporte a partir de 10.2 Jaguar
- Habilitado por defecto
- A partir de 10.3 Panther es posible habilitar/deshabilitar y configurar IPv6 con una interfaz GUI. Preferencia del Sistema > Red
- Configurar IPv6

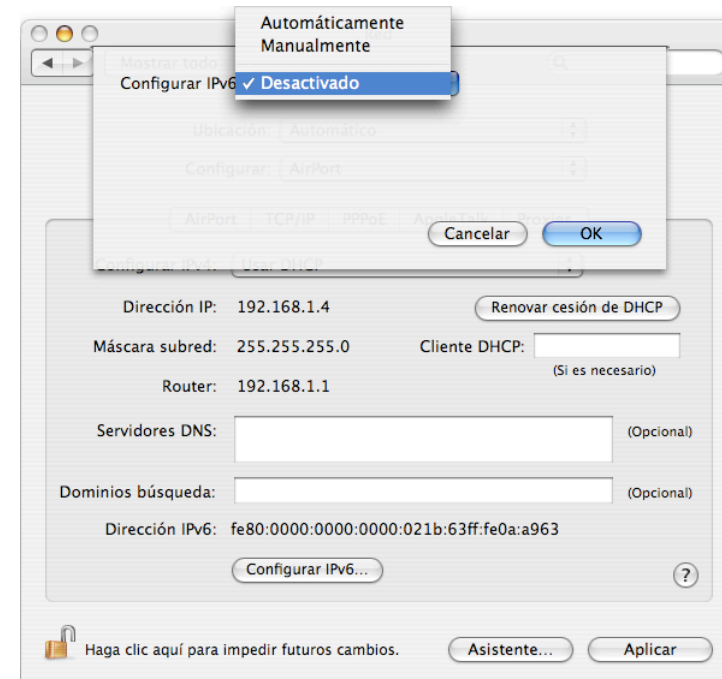


# Instalación de IPv6: Mac OS X (2)

- Habilitar/deshabilitar IPv4



- Habilitar/deshabilitar IPv6



- Habilitar/deshabilitar IPv6 en todas la interfaces
  - Habilitar # ip6 -a
  - Deshabilitar # ip6 -x

## Parte 2

# Configuración básica Stateless/Stateful, privacidad, rutas estáticas



# Configuración básica: Windows (1)

- Comandos de configuración básicos en Windows
- Sirven para obtener información sobre el estado y realizar la configuración de interfaces, direcciones, caches, rutas, etc.
- Dos grupos de comandos:
  - **ipv6.exe** (hasta XP SP1)
    - Algunos cambios no son permanentes y se pierden cuando se reinicia el PC. Se pueden ejecutar en cada inicio con un script .cmd
  - **netsh interface ipv6** (para XP SP2 y posteriores, 2003, Vista y 7)
    - Opción de store=active|persistent para guardar cambios
- Equivalencias en:  
<http://technet.microsoft.com/en-us/library/bb726950.aspx>

# Configuración básica: Windows (2)

- **Comandos “ipv6” (hasta XP SP1)**

(algunos requieren permisos de administración)

- ipv6 [-p] [-v] if [ifindex]
- ipv6 [-p] ifcr v6v4 v4src v4dst [nd] [pmlid]
- ipv6 [-p] ifcr 6over4 v4src
- ipv6 [-p] ifc ifindex [forwards] [-forwards] [advertises] [-advertises] [mtu #bytes] [site site-identifier] [preference P]
- ipv6 rlu ifindex v4dst
- ipv6 [-p] ifd ifindex
- ipv6 [-p] adu ifindex/address [life validlifetime[/preflifetime]] [anycast] [unicast]
- ipv6 nc [ifindex [address]]
- ipv6 ncf [ifindex [address]]
- ipv6 rc [ifindex address]
- ipv6 rcf [ifindex [address]]
- ipv6 bc
- ipv6 [-p] [-v] rt
- ipv6 [-p] rtu prefix ifindex[/address] [life valid[/pref]] [preference P] [publish] [age] [spl SitePrefixLength]
- ipv6 spt
- ipv6 spu prefix ifindex [life L]
- ipv6 [-p] gp
- ipv6 [-p] gpu [parameter value] ... (try -?)
- ipv6 renew [ifindex]
- ipv6 [-p] ppt
- ipv6 [-p] ppu prefix precedence P srclabel SL [dstlabel DL]
- ipv6 [-p] ppd prefix
- ipv6 [-p] reset
- ipv6 install
- ipv6 uninstall

# Configuración básica: Windows (3)

- **Comandos “netsh interface ipv6” (para XP SP2 y post., 2003, Vista y 7)**  
(algunos requieren permisos de administración)
  - 6to4 - Cambia al contexto `netsh interface ipv6 6to4'.
  - ? - Muestra una lista de comandos.
  - add - Agrega una entrada de configuración a una tabla.
  - delete - Elimina una entrada de configuración de una tabla.
  - dump - Muestra un script de configuración.
  - help - Muestra una lista de comandos.
  - isatap - Cambia al contexto `netsh interface ipv6 isatap'.
  - reset - Restablece las configuraciones de IP.
  - set - Establece la configuración de la información.
  - show - Muestra información.
  - install - Installs IPv6 (no disponible en Vista y 7)
  - uninstall - Uninstalls IPv6 (no disponible en Vista y 7)
  - renew - Restarts IPv6 interfaces (no disponible en Vista y 7)

# Configuración básica: Windows (4)

- **Comandos “netsh interface ipv6 add”**
  - add address - Agrega una dirección IP estática o puerta de enlace predeterminada a la interfaz especificada.
  - add dnsserver - Agrega una dirección estática del servidor DNS.
  - add neighbors - Agrega una dirección de vecino.
  - add potentialrouter - Agrega un enrutador a la lista de posibles enrutadores en una interfaz.
  - add prefixpolicy - Agrega una entrada de directiva de prefijo.
  - add route - Agrega una ruta a través de una interfaz.
  - add v6v4tunnel - Crea un túnel de punto a punto IPv6-in-IPv4.
  - add 6over4tunnel - Crea un túnel una interfaz 6over4 . (no disponible en Vista y 7)
- **Comandos “netsh interface ipv6 set”**
  - set address - Establece la dirección IP o puerta de enlace predeterminada para una interfaz.
  - set compartment - Modifica parámetros de configuración de compartimiento.
  - set dnsserver - Establece el modo y las direcciones del servidor DNS.
  - set dynamicportrange - Modifica el intervalo de puertos usado en la asignación de puertos dinámicos.
  - set global - Modifica parámetros generales de configuración global.
  - set interface - Modifica parámetros de configuración de interfaz para IP.
  - set neighbors - Establece una dirección de vecino.
  - set prefixpolicy - Modifica la información de directiva de prefijo.
  - set privacy - Modifica los parámetros de configuración de privacidad.
  - set route - Modifica parámetros de ruta.
  - set subinterface - Modifica parámetros de configuración de subinterfaz.
  - set teredo - Define el estado de Teredo.
  - set mobility - Modifica parámetros de configuración de movilidad. (no disponible en Vista y 7)
  - set state - Establece el estado de funcionalidades desaconsejadas (deprecated). (no disponible en Vista y 7)



# Configuración básica: Windows (5)

- **Comandos “netsh interface ipv6 show”**

- show addresses - Muestra direcciones IP actuales.
- show compartments - Muestra parámetros de compartimiento.
- show destinationcache - Muestra las entradas de caché de destino.
- show dnsservers - Muestra las direcciones del servidor DNS.
- show dynamicportrange - Muestra parámetros de configuración de intervalo de puertos dinámicos.
- show global - Muestra parámetros de configuración global.
- show interfaces - Muestra parámetros de interfaz.
- show ipstats - Muestra estadísticas IP.
- show joins - Muestra los grupos de multidifusión unidos.
- show neighbors - Muestra entradas en caché de vecinos.
- show offload - Muestra la información de descarga.
- show potentialrouters - Muestra los enrutadores posibles.
- show prefixpolicies - Muestra entradas de directiva de prefijo.
- show privacy - Muestra parámetros de configuración de privacidad.
- show route - Muestra entradas de tabla de rutas.
- show siteprefixes - Muestra entradas de la tabla de prefijos de sitios.
- show subinterfaces - Muestra parámetros de subinterfaz.
- show tcpstats - Muestra estadísticas TCP.
- show teredo - Muestra el estado de Teredo.
- show udpstats - Muestra estadísticas UDP.
- show bindingcacheentries – Muestra entradas de binding cache (no disponible en Vista y 7)
- show mobility - Muestra parámetros de configuración de movilidad. (no disponible en Vista y 7)
- show state - Muestra el estado de funcionalidades desaconsejadas (deprecated) (no disponible en Vista y 7)
- show routes - Muestra entradas de tabla de rutas. (no disponible en Vista y 7 - sustituido por show route)

# Configuración básica: Windows (6)

- Información de interfaces
  - **ipconfig [/all]**
  - **ipv6 [-v] if [IfIndex]**
  - **netsh interface ipv6 show interfaces [[interface=]<cadena>] [[rr=]<entero>] [[level=]normal|verbose] [store=]active|persistent]**
- Ejemplo: ipv6 if 5

```
Interface 5: Ethernet: Local Area Connection
Guid {F5149413-6E54-4FDA-87BD-24067735E363}
uses Neighbor Discovery
uses Router Discovery
link-layer address: 00-01-4a-18-26-c7
preferred global 2001:db8::2, life infinite (manual)
preferred global 2001:db8::4, life infinite (manual)
preferred global 2001:db8::fde7:a76f:62d5:3bb9, life 6d21h3m20s/21h33s (temporary)
preferred global 2001:db8::201:4aff:fe18:26c7, life 29d23h51m39s/6d23h51m39s (public)
preferred link-local fe80::201:4aff:fe18:26c7, life infinite
multicast interface-local ff01::1, 1 refs, not reportable
multicast link-local ff02::1, 1 refs, not reportable
multicast link-local ff02::1:ff18:26c7, 2 refs, last reporter
multicast link-local ff02::1:ffd5:3bb9, 1 refs, last reporter
multicast link-local ff02::1:ff00:4, 1 refs, last reporter
multicast link-local ff02::1:ff00:2, 1 refs, last reporter
link MTU 1500 (true link MTU 1500)
current hop limit 64
reachable time 29000ms (base 30000ms)
retransmission interval 1000ms
DAD transmits 1
default site prefix length 48
```

# Configuración básica: Windows (7)

- Ping6 hasta XP SP1/2003 SP1
  - **ping6 [-t] [-a] [-n cuenta] [-l tamaño] [-w tiempo\_espera] [-s srcaddr] [-r] nombre\_destino**
- Basta utilizar el comando **ping** en lugar **ping6** si la petición DNS devuelve un registro AAAA
- Ping
  - **ping [-t] [-a] [-n cuenta] [-l tamaño] [-f] [-i TTL] [-v TOS] [-r cuenta] [-s cuenta] [[-j lista-host] | [-k lista-host]] [-w tiempo\_espera] [-R] [-S srcaddr] [-4] [-6] nombre\_destino**
    - -4 Forzar el uso de IPv4
    - -6 Forzar el uso de IPv6

# Configuración básica: Windows (8)

- **Ejemplos de Ping**
- **ping www.ipv6tf.org**

Haciendo ping a [www.ipv6tf.org](http://www.ipv6tf.org) [2a01:48:1:0:2e0:81ff:fe05:4658] desde 2001:db8:0:0:2c0:26ff:fea0:a341 con 32 bytes de datos:

Respuesta desde 2a01:48:1:0:2e0:81ff:fe05:4658: tiempo<1m

Respuesta desde 2a01:48:1:0:2e0:81ff:fe05:4658: tiempo<1m

Respuesta desde 2a01:48:1:0:2e0:81ff:fe05:4658: tiempo<1m

Respuesta desde 2a01:48:1:0:2e0:81ff:fe05:4658: tiempo<1m

Estadísticas de ping para 2a01:48:1:0:2e0:81ff:fe05:4658:

Paquetes: enviados = 4, recibidos = 4, perdidos = 0  
(0% perdidos),

Tiempos aproximados de ida y vuelta en milisegundos:

Mínimo = 0ms, Máximo = 0ms, Media = 0ms



# Configuración básica: Windows (9)

- **Ejemplos de Ping**

- **ping ::1**

Haciendo ping a ::1 desde ::1 con 32 bytes de datos:

Respuesta desde ::1: tiempo<1m

Respuesta desde ::1: tiempo<1m

Respuesta desde ::1: tiempo<1m

Respuesta desde ::1: tiempo<1m

Estadísticas de ping para ::1:

Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos),

Tiempos aproximados de ida y vuelta en milisegundos:

Mínimo = 0ms, Máximo = 0ms, Media = 0ms

- **ping6 fe80::e8a7:b568:a076:6ba3 (link-local propia)**

Haciendo ping a fe80::e8a7:b568:a076:6ba3 desde fe80::e8a7:b568:a076:6ba3%5 con 32 bytes de datos:

Respuesta desde fe80::e8a7:b568:a076:6ba3: tiempo<1m

Respuesta desde fe80::e8a7:b568:a076:6ba3: tiempo<1m

Respuesta desde fe80::e8a7:b568:a076:6ba3: tiempo<1m

Respuesta desde fe80::e8a7:b568:a076:6ba3: tiempo<1m

Estadísticas de ping para fe80::e8a7:b568:a076:6ba3:

Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos),

Tiempos aproximados de ida y vuelta en milisegundos:

Mínimo = 0ms, Máximo = 0ms, Media = 0ms

# Configuración básica: Windows (10)

- Paréntesis 1: ¿Qué vecinos tengo?
  - **netsh interface ipv6 show neighbors**

...

Interface 5: Local Area Connection

Internet Address	Physical Address	Type
fe80::e8a7:b568:a076:6ba3	00-01-4a-18-26-c7	Permanent
fe80::200:87ff:fe28:a0e0	00-00-87-28-a0-e0	Stale (router)
2001:db8::201:4aff:fe18:26c7	00-01-4a-18-26-c7	Permanent
2001:db8::fde7:a76f:62d5:3bb9	00-01-4a-18-26-c7	Permanent
2001:db8::2a03::3	00-e0-81-05-46-57	Stale
2001:db8::1	00-00-87-28-a0-e0	Stale
2001:db8::2	00-01-4a-18-26-c7	Permanent
2001:db8::4	00-01-4a-18-26-c7	Permanent

- Paréntesis 2: La referencia a una interfaz se hace con %
  - %5 se refiere a la interfaz 5

# Configuración básica: Windows (11)

- Ejemplos de Ping
- **ping fe80::200:87ff:fe28:a0e0%5 (link-local vecino en la interfaz 5)**

Haciendo ping a fe80::200:87ff:fe28:a0e0%5 desde fe80::201:4aff:fe18:26c7%5 con 32 bytes de datos:

Respuesta desde fe80::200:87ff:fe28:a0e0%5: tiempo<1ms

Respuesta desde fe80::200:87ff:fe28:a0e0%5: tiempo<1ms

Respuesta desde fe80::200:87ff:fe28:a0e0%5: tiempo<1ms

Respuesta desde fe80::200:87ff:fe28:a0e0%5: tiempo<1ms

Estadísticas de ping para fe80::200:87ff:fe28:a0e0%5:

Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos),

Tiempos aproximados de ida y vuelta en milisegundos:

Mínimo = 0ms, Máximo = 0ms, Media = 0ms

# Configuración básica: Windows (12)

- Traceroute hasta XP SP1/2003 SP1
  - **tracert6 [-d] [-h saltos\_máximos] [-w tiempo\_de\_espera] [-s srcaddr] nombre\_destino**
- Basta utilizar el comando **tracert** en lugar **tracert6** si la petición DNS devuelve un registro AAAA
- Traceroute
  - **tracert [-d] [-h saltos\_máximos] [-j lista\_de\_hosts] [-w tiempo\_de\_espera] [-R] [-S srcaddr] [-4] [-6] nombre\_destino**
    - -4 Forzar el uso de IPv4
    - -6 Forzar el uso de IPv6



# Configuración básica: Windows (13)

- **Ejemplos de Traceroute**
- **tracert www.lacnic.net**

Traza a la dirección lacnic.net [2001:13c7:7002:4000::10]  
sobre un máximo de 30 saltos:

```
1  <1 ms  <1 ms  <1 ms  2a01:48:1::ff0
2  29 ms  25 ms  7 ms  2a01:48::d5ac:227d
3  53 ms  60 ms  35 ms  tunnel105.tserv17.lon1.ipv6.he.net [2001:470:14:69::1]
4  75 ms  109 ms  34 ms  gige-g4-18.core1.lon1.he.net [2001:470:0:a3::1]
5  63 ms  43 ms  73 ms  10gigabitethernet1-1.core1.ams1.he.net
   [2001:470:0:3f::2]
6  447 ms  163 ms  112 ms  2001:7f8:1::a500:3549:2
7  297 ms  325 ms  319 ms  2001:450:2002:7f::2
8  303 ms  313 ms  656 ms  ar01.bb2.registro.br [2001:12ff:2:1::244]
9  297 ms  315 ms  313 ms  gw01.lacnic.registro.br [2001:12ff:1:3::212]
10 302 ms  320 ms  320 ms  www.lacnic.net [2001:13c7:7002:4000::10]
```

Traza completa.

# Configuración básica: Windows (14)

- Agregar una dirección:
- **netsh interface ipv6 add address**  
[interface=]<cadena (nombre de interfaz o índice)> [address=]<dirección IPv6>[/<entero>]  
[[type=]unicast|anycast] [[validlifetime=]<entero>|infinite] [[preferredlifetime=]<entero>|infinite]  
[[store=]active|persistent]
- Ejemplo: netsh interface ipv6 add address 5 2001:db8::2 type=unicast validlifetime=infinite preferredlifetime=10m store=active
- Revisar configuración con **netsh interface ipv6 show address 5**

# Configuración básica: Windows (15)

- Modificar opciones de una dirección existente:
- **netsh interface ipv6 set address**  
[interface=]<cadena> [address=]<dirección IPv6> [[type=]unicast|anycast]  
[[validlifetime=]<entero>|infinite]  
[[preferredlifetime=]<entero>|infinite]  
[[store=]active|persistent]
- Ejemplo: netsh interface ipv6 set address 5 2001:db8::2 preferredlifetime=infinite
- Revisar configuración con **netsh interface ipv6 show address 5**

# Configuración básica: Windows (16)

- Eliminar una dirección:
- **netsh interface ipv6 delete address**  
[interface=]<cadena> [address=]<dirección IPv6> [[store=]active|persistent]
- Ejemplo: netsh interface ipv6 delete address 5  
2001:db8::2 store=persistent
- Revisar configuración con **netsh interface ipv6 show address 5**



# Configuración básica: Windows (17)

- Agregar una ruta:
- **netsh interface ipv6 add route**  
**[prefix=]<dirección IPv6>/<entero>**  
**[interface=]<cadena> [[nexthop=]<dirección**  
**IPv6>] [[siteprefixlength=]<entero>**  
**[[metric=]<entero>] [[publish=]no|yes|immortal]**  
**[[validlifetime=]<entero>|infinite]**  
**[[preferredlifetime=]<entero>|infinite]**  
**[[store=]active|persistent]**
- Ejemplo: netsh interface ipv6 add route  
2002::/16 5 fe80::200:87ff:fe28:a0e0  
store=persistent
  - Arriba, fe80::200:87ff:fe28:a0e0 es la puerta de enlace

# Configuración básica: Windows (18)

- Mostrar rutas:
- **netsh interface ipv6 show route**  
[[level=]normal|verbose] [[store=]active|persistent]

- Ejemplo: netsh interface ipv6 show route

Publicar	Tipo	Mét	Prefijo	Índ	Puerta enl./Nombre int.
No	Manual	8	::/0	13	Conexión de área local* 7
no	Manual	0	2002::/16	5	fe80::200:87ff:fe28:a0e0
no	Autoconf	8	2001:db8::/64	5	Local Area Connection
no	Autoconf	256	::/0	5	fe80::200:87ff:fe28:a0e0

# Configuración básica: Windows (19)

- Eliminar una ruta:
- **netsh interface ipv6 delete route**  
[prefix=]<dirección IPv6>/<entero>  
[interface=]<cadena> [[nexthop=]<dirección IPv6>] [[store=]active|persistent]
- Ejemplo: netsh interface ipv6 delete route  
2002::/16 5 fe80::200:87ff:fe28:a0e0  
store=persistent
- Revisar con **netsh interface ipv6 show route**

# Configuración básica: Windows (20)

- Añadir un Servidor DNS:
- **netsh interface ipv6 add dnsserver**  
[name=]<cadena> [address=]<dirección IPv6>  
[[index=]<entero>]
- En XP SP1/2003 SP1 se usa **dns** en lugar de **dnsserver**
- Ejemplo: netsh interface ipv6 add dnsserver  
“Local area network” 2001:7f9:1000:1::947c 1
- El “index” representa la posición (preferencia) del servidor DNS que se configura en la lista de servidores DNS



# Configuración básica: Windows (19)

- Mostrar Servidores DNS:
- **netsh interface ipv6 show dnsservers**  
[[name=]cadena]
- Ejemplo: netsh interface ipv6 show dnsservers

DNS servers in LAN interface

Index	DNS server
1	2001:7f9:1000:1::947c
2	2001:7f9:1000:1::947c

# Configuración básica: Windows (20)

- Borrar un Servidor DNS:
- **netsh interface ipv6 delete dnsserver**  
[name=]<cadena> [[address=]<dirección IPv6>|all]
- Ejemplo: netsh interface ipv6 delete dnsserver “Local area network” all
- Verificar mediante **netsh interface ipv6 show dnsservers**

# Configuración básica: Linux (1)

## Comandos básicos (1)

- ifconfig
- ping6 <hostcondirIPv6>|<dirIPv6>|[-I <interfaz>] <link-local-ipv6address>
- traceroute6 <hostcondirIPv6>|<dirIPv6>
- tracepath6 <hostcondirIPv6>|<dirIPv6>
- tcpdump

# Configuración básica: Linux (2)

```
# ping6 ::1
```

```
PING ::1(::1) 56 data bytes
```

```
64 bytes from ::1: icmp_seq=1 ttl=64 time=0.047 ms
```

```
64 bytes from ::1: icmp_seq=2 ttl=64 time=0.039 ms
```

```
64 bytes from ::1: icmp_seq=3 ttl=64 time=0.042 ms
```

```
64 bytes from ::1: icmp_seq=4 ttl=64 time=0.020 ms
```

```
--- ::1 ping statistics ---
```

```
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
```

```
rtt min/avg/max/mdev = 0.020/0.037/0.047/0.010 ms
```

```
# ping6 -I eth0 fe80::2e0:81ff:fe05:4657
```

```
PING fe80::2e0:81ff:fe05:4657(fe80::2e0:81ff:fe05:4657) from ::1 eth0: 56 data bytes
```

```
64 bytes from fe80::2e0:81ff:fe05:4657: icmp_seq=1 ttl=64 time=0.056 ms
```

```
64 bytes from fe80::2e0:81ff:fe05:4657: icmp_seq=2 ttl=64 time=0.055 ms
```

```
64 bytes from fe80::2e0:81ff:fe05:4657: icmp_seq=3 ttl=64 time=0.048 ms
```

```
64 bytes from fe80::2e0:81ff:fe05:4657: icmp_seq=4 ttl=64 time=0.128 ms
```

```
--- fe80::2e0:81ff:fe05:4657 ping statistics ---
```

```
4 packets transmitted, 4 received, 0% packet loss, time 2997ms
```

```
rtt min/avg/max/mdev = 0.048/0.071/0.128/0.034 ms
```



# Configuración básica: Linux (3)

## Comandos básicos (2)

- Añadir una dirección IPv6

```
# /sbin/ip -6 addr add <ipv6address>/<prefixlength> dev <interface>
```

```
# /sbin/ifconfig <interface> inet6 add <ipv6address>/<prefixlength>
```

- Eliminar una dirección IPv6

```
# /sbin/ip -6 addr del <ipv6address>/<prefixlength> dev <interface>
```

```
# /sbin/ifconfig <interface> inet6 del <ipv6address>/<prefixlength>
```

# Configuración básica: Linux (4)

## Rutas estáticas

- Ver rutas IPv6

```
# /sbin/ip -6 route show [dev <device>]
```

```
# /sbin/route -A inet6
```

- Añadir ruta a través de una puerta de enlace

```
# /sbin/ip -6 route add <ipv6network>/<prefixlength> via <ipv6address>  
[dev <device>]
```

```
#!/sbin/route -A inet6 add <ipv6network>/<prefixlength> gw <ipv6address>  
[dev <device>]
```

# Configuración básica: Linux (5)

- Eliminar ruta a través de una puerta de enlace

```
# /sbin/ip -6 route del <ipv6network>/<prefixlength> via <ipv6address>  
[dev <device>]
```

```
# /sbin/route -A inet6 del <network>/<prefixlength> [dev <device>]
```

- Añadir ruta a través de una interfaz

```
# /sbin/ip -6 route add <ipv6network>/<prefixlength> dev <device> metric 1
```

```
# /sbin/route -A inet6 add <network>/<prefixlength> dev <device>
```

# Configuración básica: Linux (6)

- Eliminar ruta a través de una interfaz

```
# /sbin/ip -6 route del <ipv6network>/<prefixlength> dev <device>
```

```
# /sbin/route -A inet6 del <network>/<prefixlength> dev <device>
```

- Visualizar tabla de “vecinos”

```
# ip -6 neigh show [dev <device>]
```

- Añadir entrada a la tabla de “vecinos”

```
# ip -6 neigh add <IPv6 address> lladdr <link-layer address> dev <device>
```

- Eliminar entrada a la tabla de “vecinos”

```
# ip -6 neigh del <IPv6 address> lladdr <link-layer address> dev <device>
```



# Configuración básica: BSD (1)

## Comandos básicos:

- Añadir una dirección IPv6

```
#>ifconfig <interface> inet6 add <dir. IPv6>
```

- Eliminar una dirección IPv6

```
#>ifconfig <interface> inet6 del <dir. IPv6>
```

# Configuración básica: BSD (2)

- **Configuración permanente:**

Se hace en el fichero `/etc/rc.conf`:

```
ipv6_enable="YES"
```

```
ipv6_ifconfig_rlo="2001:618:10:4::4 prefixlen 64"
```

En `/etc/defaults/rc.conf` se pueden consultar las posibles opciones existentes y las que se usan por defecto.

- Para aplicar cambios en `rc.conf` habrá que reiniciar

# Configuración básica: BSD (3)

## Rutas estáticas

- Añadir ruta por defecto

```
#>route -n add -inet6 default <dir. IPv6>
```

- Eliminar ruta por defecto

```
#>route -n del -inet6 default
```

# Configuración básica: Mac OS X (1)

## Comandos básicos (1)

- ifconfig, ifconfig en1
- ping6 <hostcondirIPv6>|<dirIPv6>|[-I <interfaz>] <link-local-ipv6address>
- traceroute6 <hostcondirIPv6>|<dirIPv6>
- tracepath6 <hostcondirIPv6>|<dirIPv6>
- tcpdump



# Configuración básica: Mac OS X (2)

```
$ ping6 ::1
```

```
PING6(56=40+8+8 bytes) ::1 --> ::1
```

```
16 bytes from ::1, icmp_seq=0 hlim=64 time=0.122 ms
```

```
16 bytes from ::1, icmp_seq=1 hlim=64 time=0.127 ms
```

```
16 bytes from ::1, icmp_seq=2 hlim=64 time=0.117 ms
```

```
16 bytes from ::1, icmp_seq=3 hlim=64 time=0.118 ms
```

```
16 bytes from ::1, icmp_seq=4 hlim=64 time=0.128 ms
```

```
--- ::1 ping6 statistics ---
```

```
5 packets transmitted, 5 packets received, 0% packet loss
```

```
round-trip min/avg/max = 0.117/0.122/0.128 ms
```

```
$ ping6 fe80::2e0:81ff:fe05:4657%en1
```

# Configuración básica: Mac OS X (3)

## Comandos básicos (2)

- Añadir una dirección IPv6

```
# ifconfig <interface> inet6 2001:db8:1:1::2/64
```

- Eliminar una dirección IPv6

```
#>ifconfig <interface> inet6 del 2001:db8:1:1::2/64
```

- Ver configuración

```
# ifconfig | grep inet6
```

```
# ifconfig <interface> | grep inet6
```

# Configuración básica: Mac OS X (4)

## Rutas estáticas

- Añadir ruta por defecto

```
# route -n add -inet6 -prefixlen 0 default 2001:db8:1:1::2
```

- Eliminar ruta por defecto

```
#>route del -inet6 default
```

- Ver rutas IPv6

```
# netstat -r -f inet6
```

# Configuración básica: Ejercicios 1

- Tratar de hacer ping6 a la dirección link-local de otra máquina
- A la vez, tratar de capturar paquetes con tcpdump:

```
# tcpdump -t -n -i eth0 -s 512 -vv ip6 or proto ipv6
```

- Otras formas de ver direcciones:

```
# /sbin/ip -6 addr show dev eth2
```

```
# ifconfig eth0
```

- Añadir y quitar la dirección

```
2001:800:40:2a09:1:2:3:4 a la interfaz eth0
```



# Configuración básica: Ejercicio 2

## Linux

- Añadir y eliminar una ruta a través de una puerta de enlace
- Añadir y eliminar una ruta a través de una interfaz
- Visualizar tabla de vecinos
- Añadir y eliminar un vecino

## BSD

- Añadir y eliminar una ruta a través de un puerta de enlace

# Autoconfiguración Stateless (1)

- RFC 2462: IPv6 Stateless Address Autoconfiguration
- [STATELESS] Proporciona información sobre:
  - Prefijo de red
  - Enrutamiento
- Direcciones globales se forman con la unión de dos elementos
  - Identificador de interfaz (de 64 bits basado en EUI-64, y usualmente obtenido de una dirección IEEE 48 bit MAC)
  - Prefijo obtenido de la opciones de Prefix Information contenida en los Router Advertisements
- Facilita la autoconfiguración
  - El usuario no necesita introducir ningún parámetro de red para que el nodo final tenga conectividad IPv6 nativa

# Autoconfiguración Stateless (2)

- En hosts con Windows esta activada por defecto
- Usar **ipconfig**, **ipv6 if** o **netsh interface ipv6 show addresses** para revisar cual es la dirección autoconfigurada
- Ejemplo: **2001:db8:10:10:201:4aff:fe18:26c7**
  - Identificador de interfaz EUI-64 obtenido de una dirección MAC: 4aff:fe18:26c7
  - Prefijo dado por el router: **2001:db8:10:10**

# Stateless: Ejercicio 1 (1)

- Configurar un router linux para enviar paquetes RA a la red
- Obtener la implementación del daemon 'radvd' adecuada a la distribución Linux
  - <http://www.rpmfind.net/linux/rpm2html/search.php?query=radvd&submit=Search+...>
- Instalarla
- Habilitar la funcionalidad de routing
  - `echo 1 > /proc/sys/net/ipv6/conf/all/forwarding`
- Generar el fichero `/etc/radvd.conf` con el siguiente contenido



# Stateless: Ejercicio 1 (2)

```
interface eth00
{
    AdvSendAdvert on;

    MinRtrAdvInterval 3;
    MaxRtrAdvInterval 5;

    AdvHomeAgentFlag off;

    prefix 2001:8500:40:2a30::/64
    {
        AdvOnLink off;
        AdvAutonomous on;
        AdvRouterAddr off;
    };
};
```

# Stateless: Ejercicio 1 (3)

- Arrancar el demonio radvd
  - radvd
- Comprobar las direcciones obtenidas en otros PC conectados a la misma red

# Autoconfiguración Stateful (1)

- [STATEFUL] Similar al funcionamiento DHCP en IPv4
- Se proporciona una dirección IPv6 que puede ser diferente cada vez que se conecta un nodo
- Proporciona información complementaria a la proporcionada por stateless
  - Servidor DNS (puede ser IPv6)
  - Nombre dominio
  - Servidor NTP (puede ser IPv6)
  - Servidor SIP (puede ser IPv6)
  - Nombre dominio SIP
  - Prefix delegation
  - Etc.
- Las implementaciones de DHCPv6 no están aún disponibles en los Sistemas Operativos habituales
  - Necesario realizar la instalación específica de una aplicación que implemente la funcionalidad DHCPv6 (server y/o client)
    - <http://klub.com.pl/dhcpv6/>
    - <http://sourceforge.net/projects/dhcpv6-linux/>

# Stateful: Ejercicio 1 (1)

- Configurar un servidor DHCPv6 en Linux
  - Obtener la implementación de DHCPv6 para linux de: <http://klub.com.pl/dhcpv6/dibbler/dibbler-0.4.0-linux.tar.gz>
  - Descomprimir el fichero
    - `tar -xvzf dibbler-0.4.0-linux.tar.gz`
  - Crear los directorios
    - `/var/lib/dibbler`
    - `/etc/dibbler`



# Stateful: Ejercicio 1 (2)

- Modificar el contenido del fichero server.conf
  - log-level 7
  - log-mode short
  
  - iface eth0 {
  - T1 1000
  - T2 2000
  - class {
  - pool 2001:3820:40:2a03::10-2001:3820:40:2a03:ffff:ffff:ffff:ffff
  - }
  
  - option dns-server 2001:800:40:2a03::2, 2001:800:40:2a04::2
  - option domain example.com, test1.example.com
  
  - }
- Las direcciones proporcionadas estarán en el rango 2001:3820:40:2a03::/64 a partir de la 2001:3820:40:2a03::10
- Copiar el fichero server.conf en el directorio /etc/dibbler
- Arrancar el servidor dhcpv6
  - dhcpv6-server run

# Stateful: Ejercicio 2 (1)

- Configurar un cliente DHCPv6 en Linux
  - Obtener la implementación de DHCPv6 para linux de: <http://klub.com.pl/dhcpv6/dibbler/dibbler-0.4.0-linux.tar.gz>
  - Descomprimir el fichero
    - `tar -xvzf dibbler-0.4.0-linux.tar.gz`
  - Crear los directorios
    - `/var/lib/dibbler`
    - `/etc/dibbler`

# Stateful: Ejercicio 2 (2)

- Modificar el contenido del fichero client.conf
  - log-mode short
  - iface eth0
  - {
  - IA
  - option dns-server
  - option domain
  - }
- La configuración es para obtener
  - una dirección IPv6
  - servidores dns
  - nombre del dominio
- Copiar el fichero client.conf en el directorio /etc/dibbler
- Arrancar el client dhcpv6
  - dhcpv6-client run
- Con 'ifconfig eth0' se puede observar la dirección obtenida
- En el fichero /etc/resolv se puede observar los servidores dns obtenidos
- No se obtiene información de routing, por lo que no se puede hacer ping
  - Esta información es proporcionada por la autoconfiguración stateless (RA)

# Privacidad (1)

- RFC 3041: Privacy Extensions for Stateless Address Autoconfiguration in IPv6
- Extensión de Autoconfiguración Stateless
- Para generar una dirección global que cambie con el tiempo
- Dificulta recolectar información para identificar que transacciones corresponden a un nodo



# Privacidad (2)

- Desde XP/2003 esta activada por defecto
- **netsh interface ipv6 show privacy** para ver el estado
- **ipconfig, ipv6 if o netsh interface ipv6 show addresses** para ver la dirección autoconfigurada
- Para desactivarlo:
  - **netsh interface ipv6 set privacy state=disabled store=persistent**
  - **ipv6 [-p] gpu UseTemporaryAddresses no**
- Para revisar el cambio **ipconfig, ipv6 if o netsh interface ipv6 show addresses**
- En XP/2003 es posible que sea necesario “disable” y “enable” la interfaz física en Network Connection para ver el cambio en la configuración de la privacidad

# Privacidad (3)

- **Más opciones en comandos netsh:**
- netsh interface ipv6 set privacy  
[[state=]enabled|disabled]  
[[maxdadattempts=]<entero>]  
[[maxvalidlifetime=]<entero>]  
[[maxpreferredlifetime=]<entero>]
- [[regeneratetime=]<entero>  
[[maxrandomtime=]<entero>] [[store=]active|  
persistent]



## Parte 3

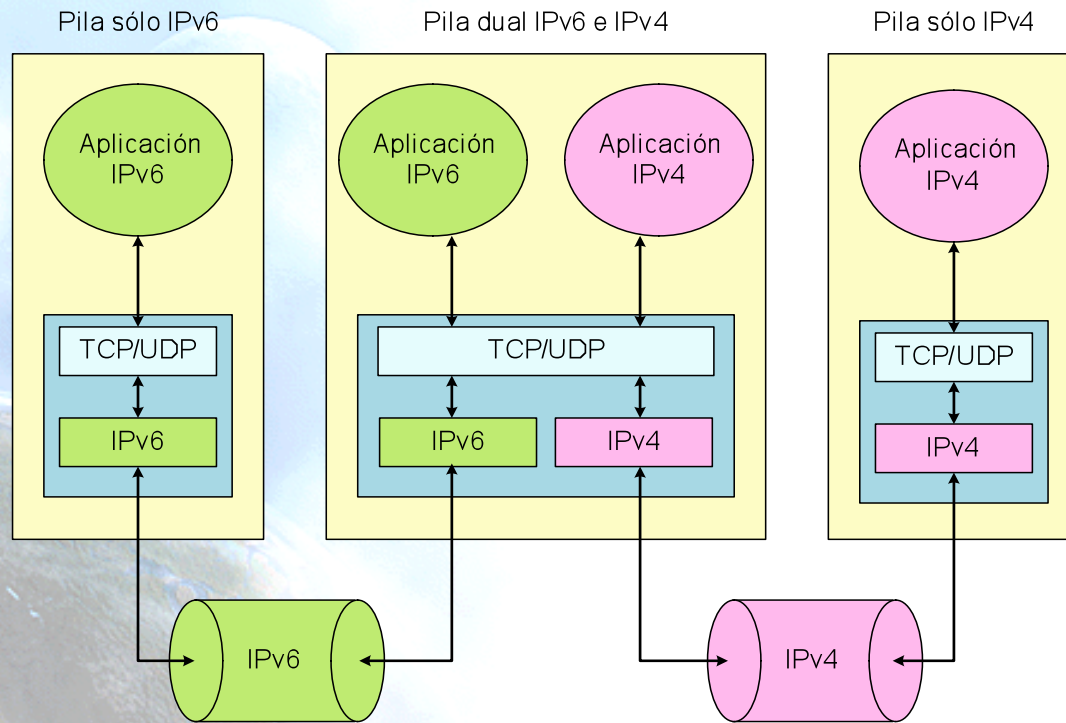
# Configuración de mecanismos de transición

# Mecanismos de transición

- IPv6 ha sido diseñado de tal forma que se facilite la transición y coexistencia con IPv4
- Se han diseñado diferentes estrategias para la coexistencia con redes/nodos IPv4
  - Doble pila, o soporte simultáneo de IPv4 e IPv6
  - Túneles, o encapsulado de IPv6 sobre IPv4 (y viceversa)
    - Son los más utilizados
  - Traducción IPv4/IPv6, como último recurso, dado que no es perfecto



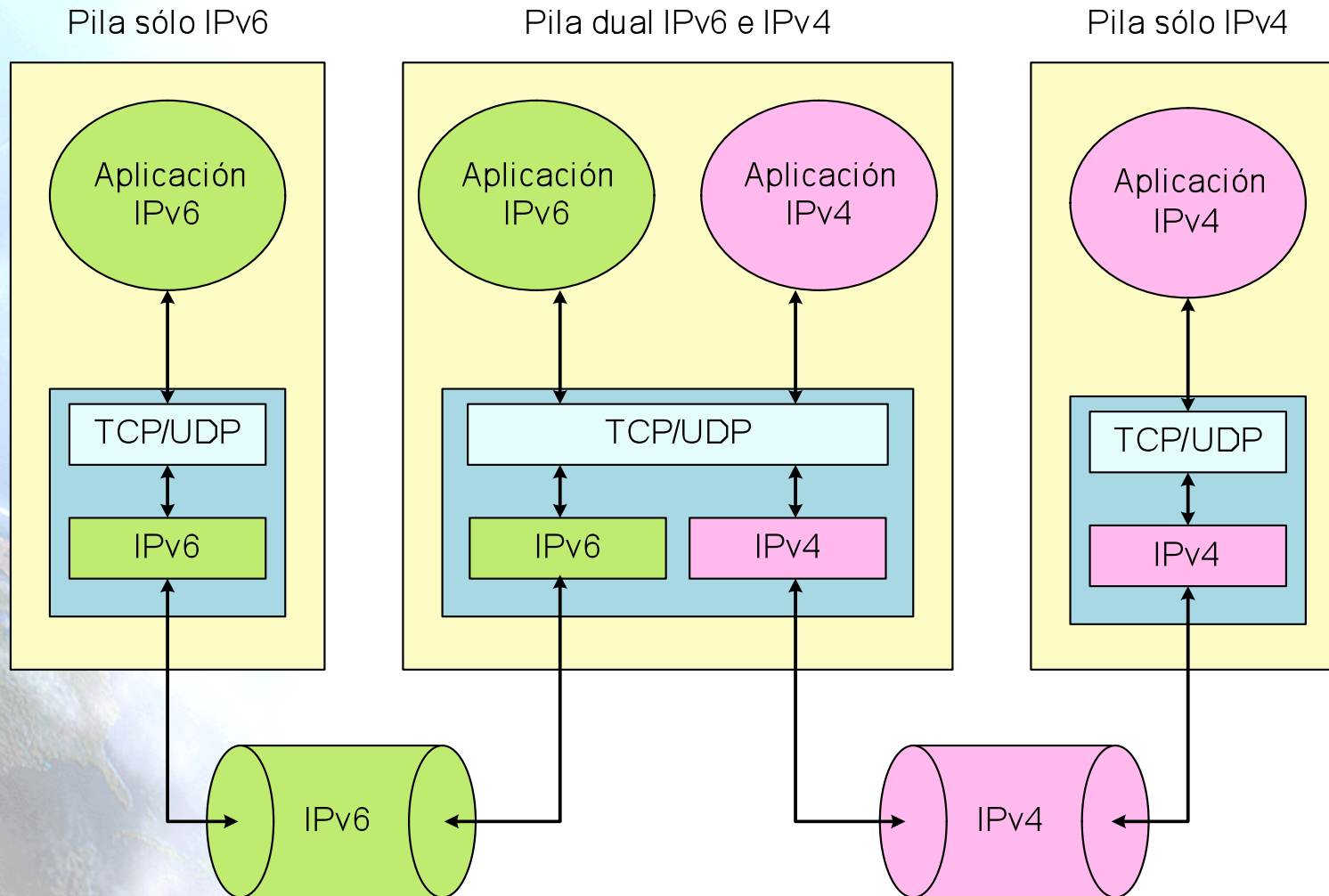
# Doble pila (1)



Mécanismo basado en doble pila

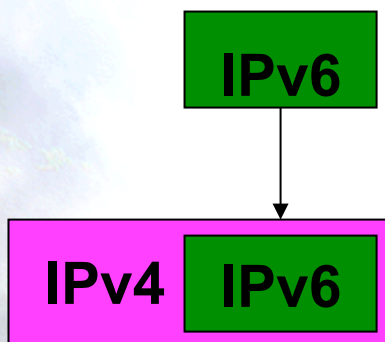
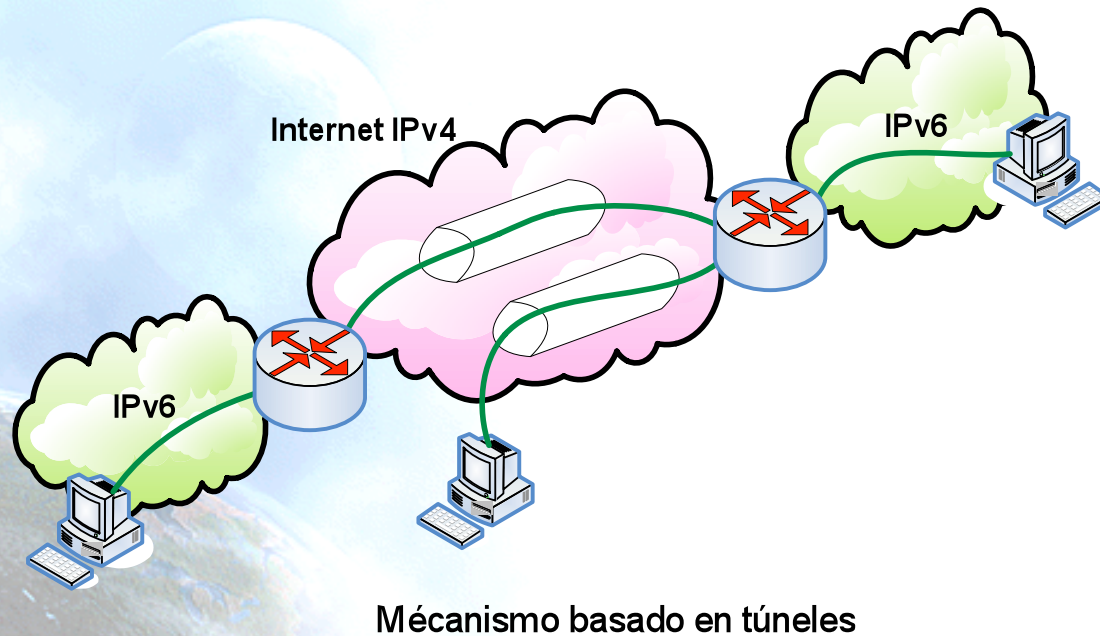
- Los nodos tienen implementadas las pilas IPv4 e IPv6
- Comunicaciones con nodos solo IPv6 ==> Pila IPv6, asumiendo soporte IPv6 en la red
- Comunicaciones con nodos solo IPv4 ==> Pila IPv4

# Doble pila (2)



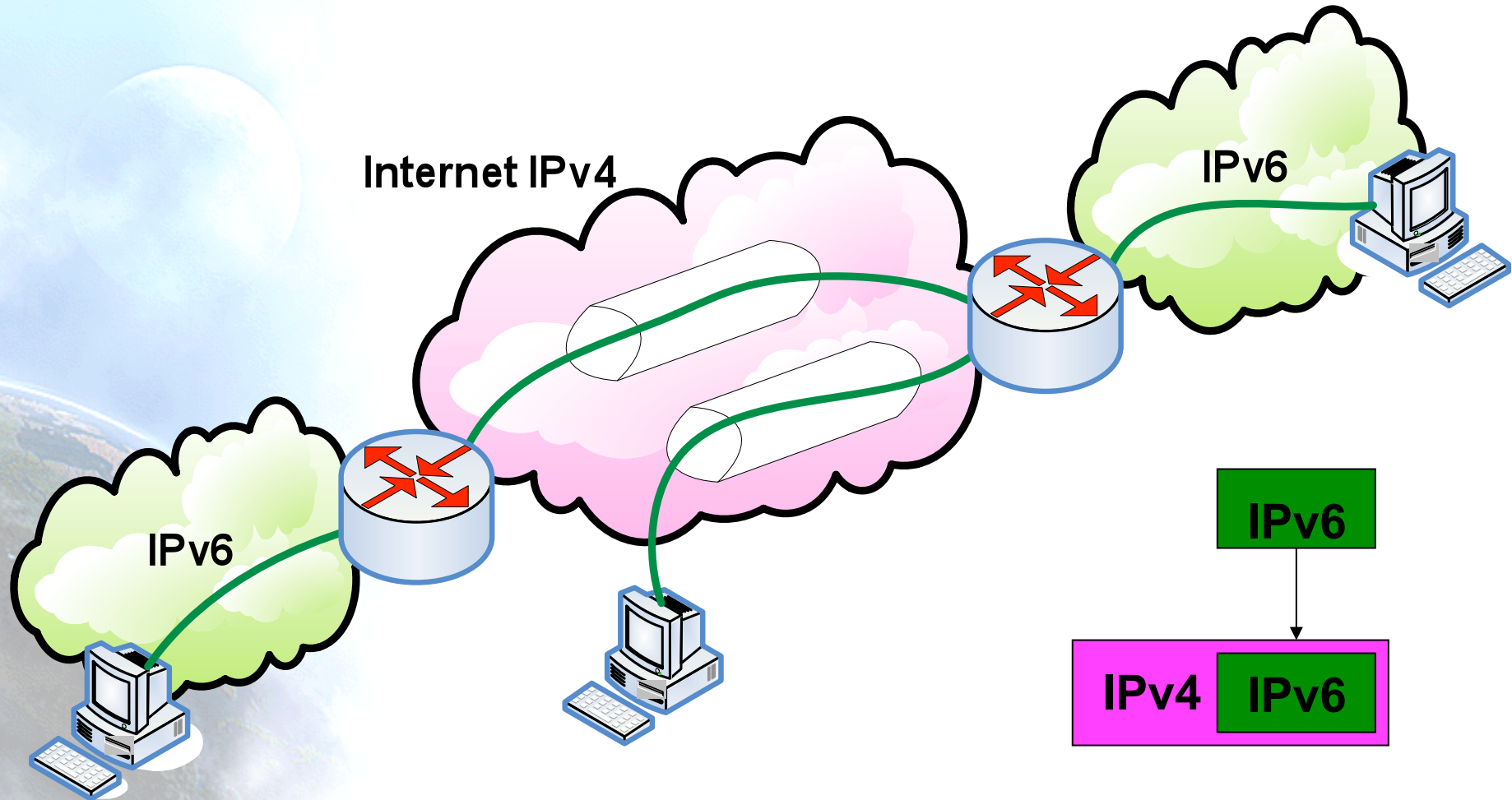
Mécanismo basado en doble pila

# Túneles IPv6 en IPv4 (1)



- Usado para proporcionar conectividad IPv6 en redes que solo tiene soporte IPv4
- Se encapsulan paquetes IPv6 dentro de paquetes IPv4
- Los paquetes resultantes viajan por redes IPv4

# Túneles IPv6 en IPv4 (2)

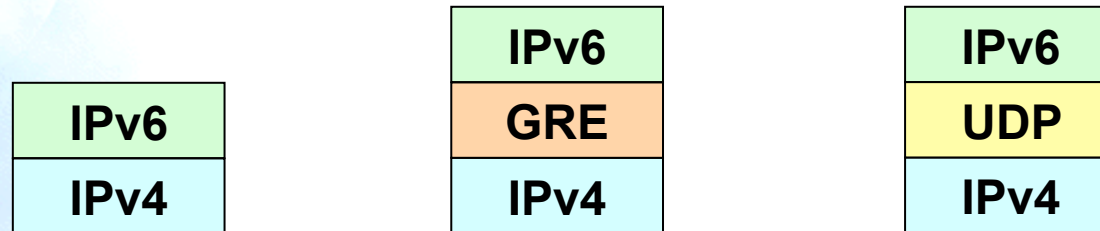


Mécanismo basado en túneles



# Túneles IPv6 en IPv4 (2)

- Existen diversas formas de encapsular los paquetes IPv6



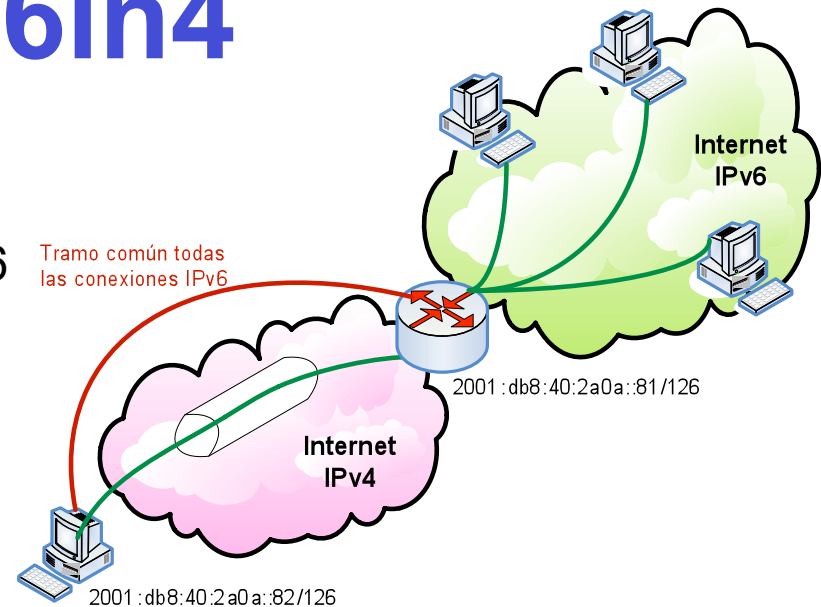
- Existen diversos mecanismos de transición basados en túneles, cada uno con una forma diferente de encapsulación

# Túneles IPv6 en IPv4 (3)

- Algunos mecanismos de transición basados en túneles
  - 6in4 (\*) [6in4]
  - TB (\*) [TB]
  - TSP [TSP]
  - 6to4 (\*) [6to4]
  - Teredo (\*) [TEREDO], [TEREDOC]
  - Túneles automáticos [TunAut]
  - ISATAP [ISATAP]
  - 6over4 [6over4]
  - AYIYA [AYIYA ]
  - Silkroad [SILKROAD ]
  - DSTM [DSTM]
- (\*) Más habituales y explicados en detalle a continuación

# Túneles 6in4

- Encapsula directamente el paquete IPv6 dentro de un paquete IPv4
- Se suele hacer entre
  - nodo final ==> router
  - router ==> router
- Aunque también es posible para
  - nodo final ==> nodo final
- El túnel se considera como un enlace punto-a-punto desde el punto de vista de IPv6
  - Solo un salto IPv6 aunque existan varios IPv4
- Las direcciones IPv6 de ambos extremos del túnel son del mismo prefijo
- Todas las conexiones IPv6 del nodo final siempre pasan por el router que está en el extremo final del túnel
- Los túneles 6in4 pueden construirse desde nodo finales situados detrás de NAT
  - Imprescindible que la implementación de NAT soporte “proto-41 forwarding” [PROTO41]

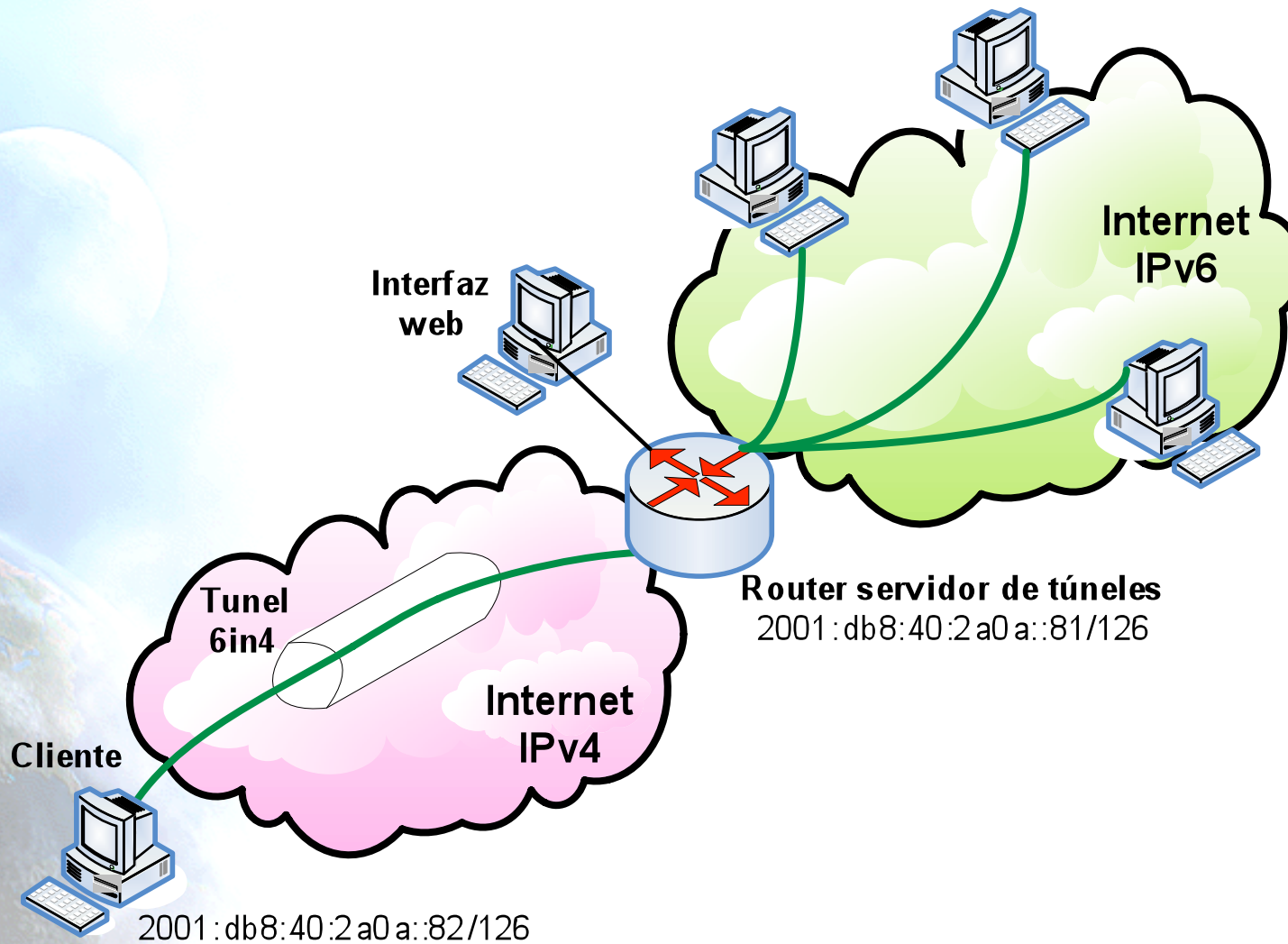


# Tunnel Broker (RFC3053)

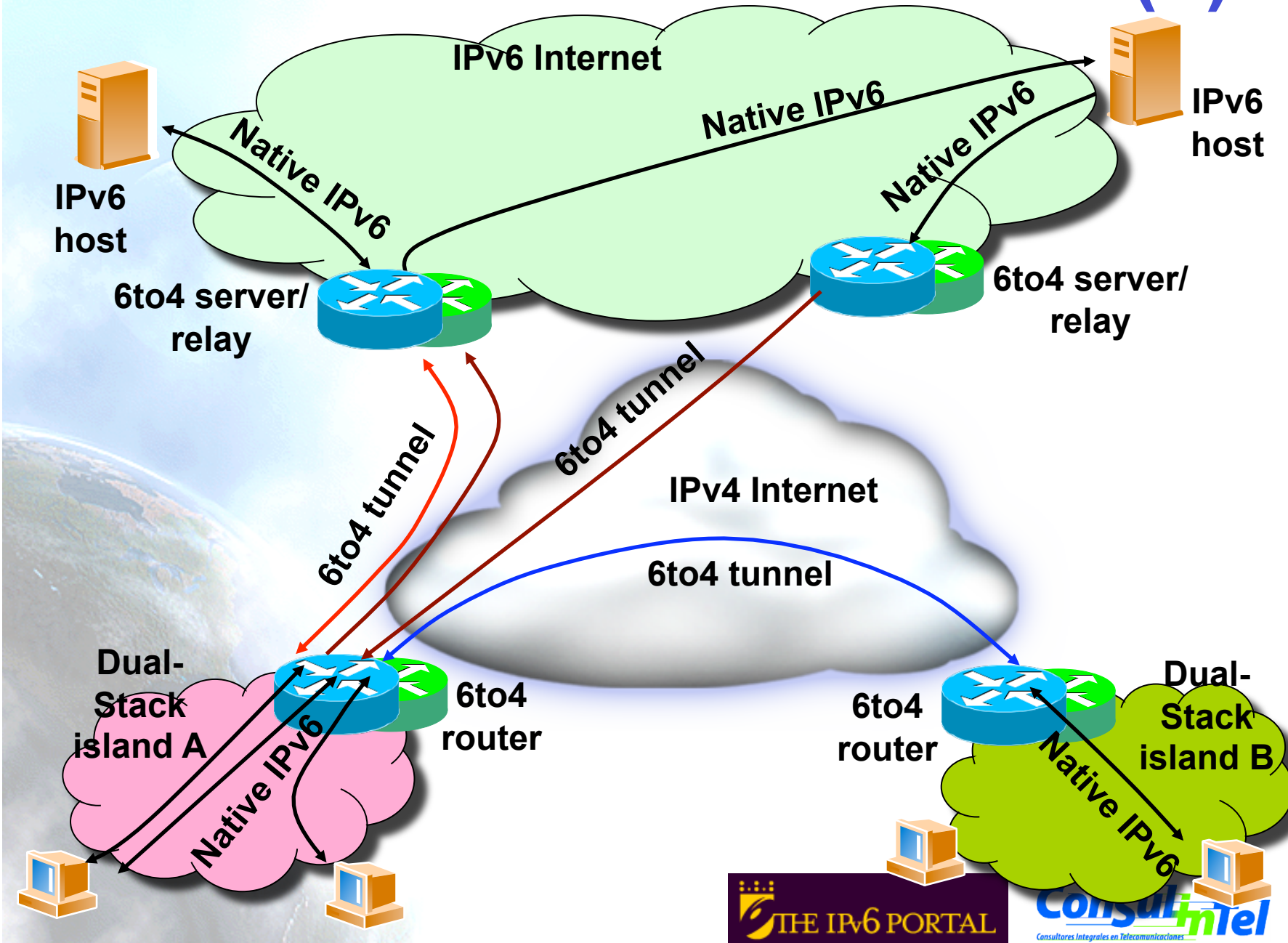
- Los túneles 6in4 requieren la configuración manual de los equipos involucrados en el túnel
- Para facilitar la asignación de direcciones y creación de túneles IPv6, se ha desarrollado el concepto de Tunnel Broker (TB).
  - Es un intermediario al que el usuario final se conecta, normalmente con un interfaz web
- El usuario solicita al TB la creación de un túnel y este le asigna una dirección IPv6 y le proporciona instrucciones para crear el túnel en el lado del usuario
- El TB también configura el router que representa el extremo final del túnel para el usuario
- En <http://www.ipv6tf.org/using/connectivity/test.php> existe una lista de TB disponibles
- TSP [TSP] es un caso especial de TB que no está basado en un interfaz web sino en un aplicación cliente que se instala en el cliente y se conecta con un servidor, aunque el concepto es el mismo.



# Tunnel Broker (RFC3053)



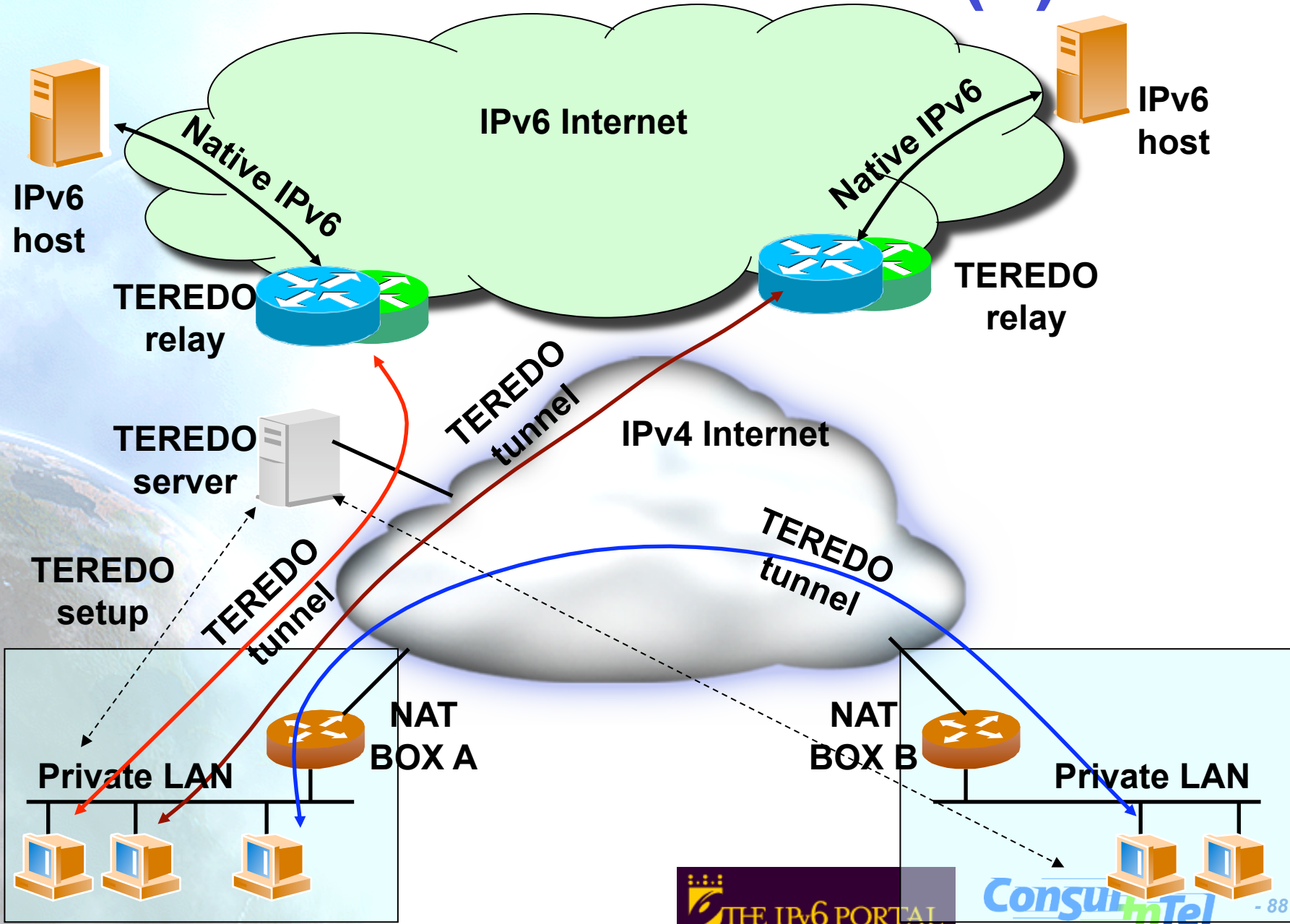
# Túneles 6to4: RFC3056 (1)



# Túneles 6to4: RFC3056 (2)

- Se trata de un encapsulado de paquetes IPv6 en paquetes IPv4, similar a 6in4
- Diferencias:
  - La dirección IPv6 del cliente no depende del router al que se conecta sino de la dirección IPv4 pública
    - Rango 2002::/16
  - Los paquetes IPv6 de salida del cliente siempre son enviados al mismo “6to4 relay”, sin embargo los paquetes IPv6 de entrada al cliente pueden provenir de otros “6to4 relay” diferentes.
- Prefijo IPv4 anycast (RFC3068):
  - 192.88.99.1/24

# Teredo: RFC4380 (1)





# Teredo: RFC4380 (2)

- Teredo [TEREDO] [TEREDOC] está pensado para proporcionar IPv6 a nodos que están ubicados detrás de NAT que no son “proto-41 forwarding”.
  - Encapsulado de paquetes IPv6 en paquetes UDP
- Funciona en NAT de tipo [STUN]
  - Full Cone
  - Restricted Cone
- No funciona en NATs de tipo
  - Symmetric (solventado a partir de Windows Vista)
- Intervienen diversos agentes:
  - Teredo Server
  - Teredo Relay
  - Teredo Client
- El cliente configura un Teredo Server que le proporciona una dirección IPv6 del rango 2001:0000::/32 basada en la dirección IPv4 pública y el puerto usado
  - Si el Teredo Server configurado es además Teredo Relay, el cliente tiene conectividad IPv6 con cualquier nodo IPv6
  - De lo contrario solo tiene conectividad IPv6 con otros clientes de Teredo
- Actualmente Microsoft proporciona Teredo Servers públicos y gratuitos, pero no Teredo Relays

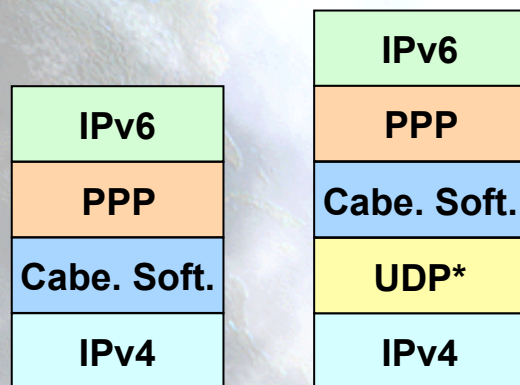
# Softwires: RFC4925

- Protocolo que esta siendo discutido en el grupo de trabajo Softwire del IETF. Presenta las siguientes características:
  - Mecanismo de transición “universal” basado en la creación de túneles
    - IPv6-en-IPv4, IPv6-en-IPv6, IPv4-en-IPv6, IPv4-en-IPv4
    - Permite atravesar NATs en las redes de acceso
    - Proporciona delegación de prefijos IPv6 (/48, /64, etc.)
    - Autenticación de usuario para la creación de túneles mediante la interacción con infraestructura AAA
    - Posibilidad de túneles seguros
    - Baja sobrecarga en el transporte de paquetes IPv6 en los túneles
    - Fácil inclusión en dispositivos portátiles con escasos recursos hardware
  - Softwires posibilitará la provisión de conectividad IPv6 en dispositivos como routers ADSL, teléfonos móviles, PDAs, etc. cuando no exista conectividad IPv6 nativa en el acceso
  - También posibilita la provisión de conectividad IPv4 en dispositivos que solo tienen conectividad IPv6 nativa
- En realidad Softwires no es un nuevo protocolo, sino la definición de cómo usar de una forma diferente protocolos ya existentes con el fin de proporcionar conectividad IPv6 en redes IPv4 y viceversa
- Softwires se basa en:
  - L2TPv2 (RFC2661)
  - L2TPv3 (RFC3991)

# Encapsulamiento de Softwires basado en L2TPv2

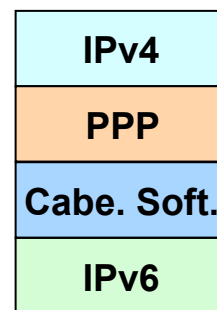
- El funcionamiento se especifica en draft-ietf-softwire-hs-framework-l2tpv2
- Existen dos entidades:
  - Softwires Initiator (SI): agente encargado de solicitar el túnel
  - Softwires Concentrator (SC): agente encargado de crear el túnel (tunnel end point)
- Se utiliza PPP para transportar paquetes IPx (x=4, 6) en paquetes IPy (y=4, 6)
  - Opcionalmente se puede encapsular los paquetes PPP en UDP en caso de que haya que atravesar NATs

Túnel IPv6-en-IPv4

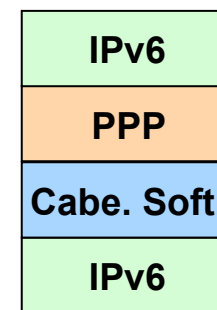


\* Opcional

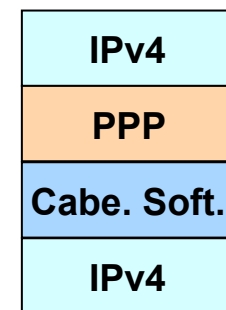
Túnel IPv4-en-IPv6



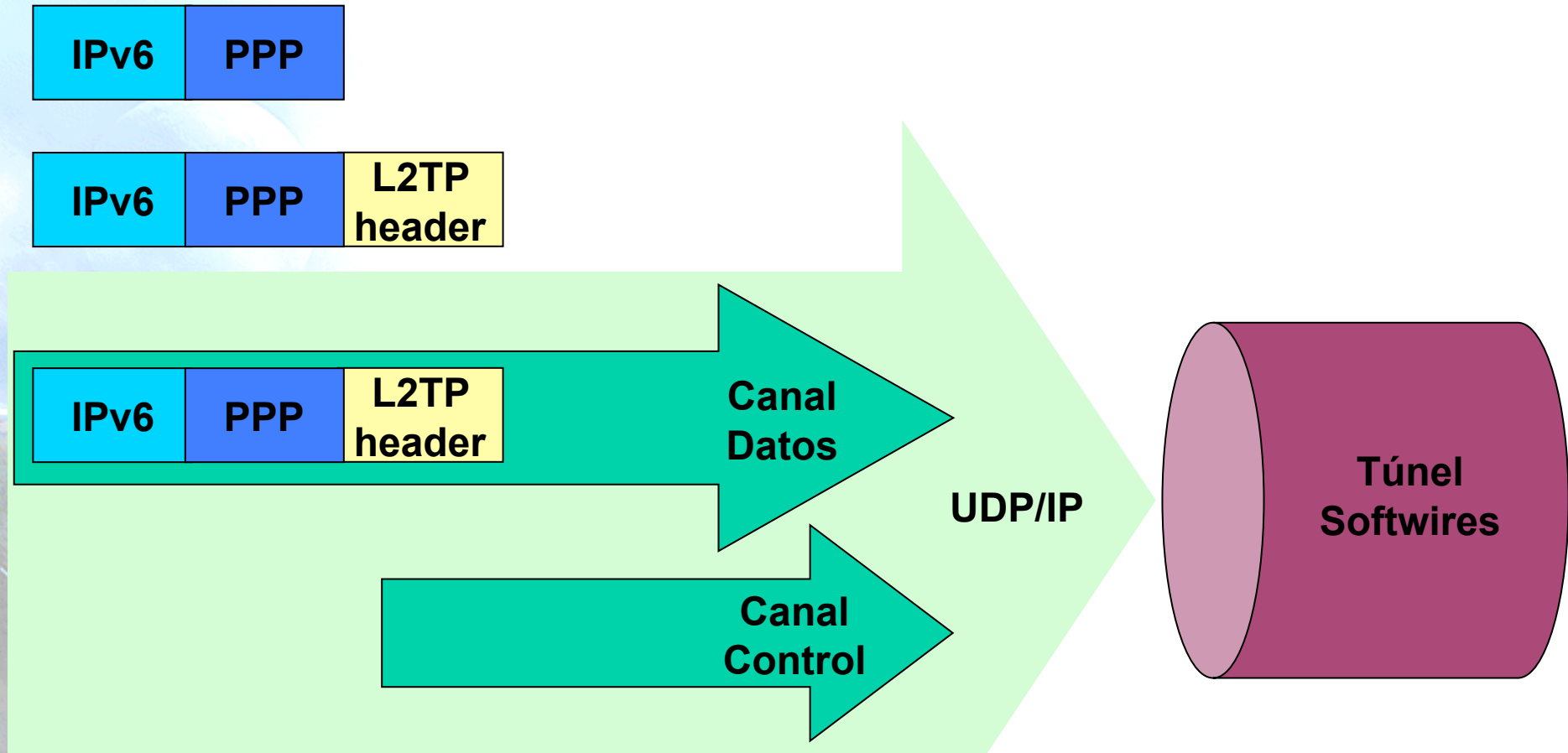
Túnel IPv6-en-IPv6



Túnel IPv4-en-IPv4



# Softwires basado en L2TPv2

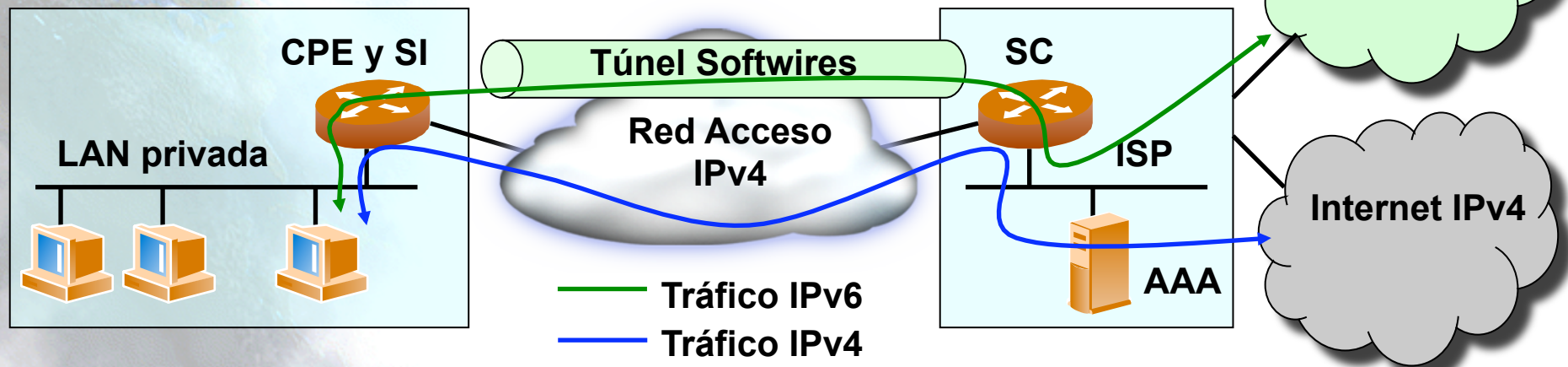


- Existe un plano de control y otro de datos
- Se usa PPP como protocolo de encapsulamiento



# Ejemplo de uso de Softwires

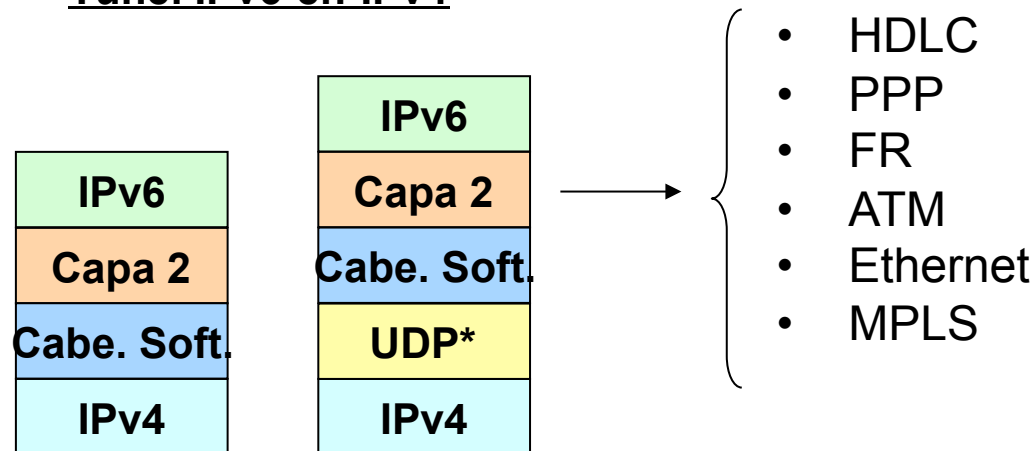
- Un uso típico previsible de Softwires es la provisión de conectividad IPv6 a usuarios domésticos a través de una red de acceso solo-IPv4
  - El SC está instalado en la red del ISP
    - DSLAM, Router de agregación u otro dispositivo
  - El SI está instalado en la red del usuario
    - CPE típicamente. También es posible otro dispositivo diferente en la red del usuario
  - El SC proporciona conectividad IPv6 al SI, y el SI hace de encaminador IPv6 para el resto de la red de usuario
  - Se usa delegación de prefijo IPv6 entre el SC y el SI para proporcionar un prefijo (típicamente /48) a la red del usuario
    - DHCPv6 PD
- Otros usos son también posibles
  - VPNs sobre IPv6 o IPv4
  - Conectividad IPv4 en red de acceso solo IPv6, etc.



# Encapsulamiento de Softwires basado en L2TPv3

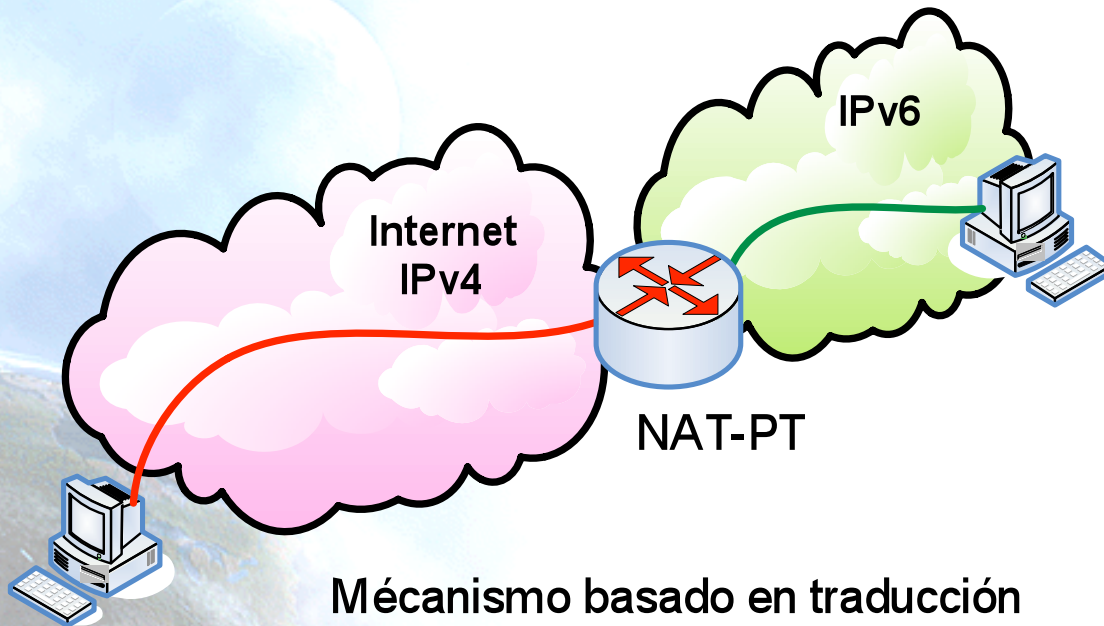
- Misma filosofía y componentes que con L2TPv2, pero con las particularidades de L2TPv3
  - Transporte sobre IP/UDP de otros protocolos de capa 2 diferentes a PPP
    - HDLC, PPP, FR, ATM, Ethernet, MPLS, IP
  - Formato de cabeceras mejorado para permitir un tratamiento más rápido en los SC
    - Permite velocidades del rango de T1/E1, T3/E3, OC48
  - Mínimo overhead en los paquetes encapsulados (solo de 4 a 12 bytes extra)
  - Otros mecanismos de autenticación diferentes a CHAP y PAP
    - EAP

## Túnel IPv6-en-IPv4



\* Opcional

# Traducción IPv4/IPv6



- Diferentes soluciones, pero tiene en común que tratan de traducir paquetes IPv4 a IPv6 y viceversa
  - [SIT], [BIS], [TRT], [SOCKSv64]
- La más conocida es NAT-PT [NATPT], [NATPTIMPL]
  - Un nodo intermedio (router) modifica las cabeceras IPv4 a cabeceras IPv6
  - El tratamiento de paquetes es complejo
- Es la peor solución puesto que la traducción no es perfecta y requiere soporte de ALGs, como en el caso de los NATs IPv4
  - DNS, FTP, VoIP, etc.

# Configuración de mecanismos de transición: Ejercicios

- E1: Establecer túnel 6in4 entre las máquinas de dos alumnos
- E2: Eliminar túnel 6in4
- E3: Obtener conectividad IPv6 mediante un túnel 6in4 usando un TB
  - Ver path a diferentes sitios web IPv6
  - Ver path a la dirección IPv6 desde un looking glass
- E4: Obtener conectividad IPv6 mediante un túnel 6to4
  - Ver path a diferentes sitios web IPv6
  - Ver path a la dirección IPv6 desde un looking glass
- E5: Configurar un 6to4 relay (Windows 2003)
- E6: Configurar un Teredo Cliente (Windows XP/2003)
- E7: Uso de proxy IPv4/IPv6
  - 46Bouncer
  - Windows XP/2003



# E1: Establecimiento túnel 6in4 (1)

1. Ejercicio para ser realizado entre dos alumnos (\*)
    - Alumno A ==> DIR\_IPv4\_A
    - Alumno B ==> DIR\_IPv4\_B
  2. El alumno A realiza la configuración de su extremo del túnel con los siguientes datos
    - Dirección IPv4 local ==> DIR\_IPv4\_A
    - Dirección IPv4 remota ==> DIR\_IPv4\_B
    - Dirección IPv6 ==> 2001:10:20:30::12/126
    - Dirección puerta de enlace IPv6 ==> 2001:10:20:30::11/126
  3. El alumno B realiza la configuración de su extremo del túnel con los siguientes datos
    - Dirección IPv4 local ==> DIR\_IPv4\_B
    - Dirección IPv4 remota ==> DIR\_IPv4\_A
    - Dirección IPv6 ==> 2001:10:20:30::11/126
    - Dirección puerta de enlace IPv6 ==> 2001:10:20:30::12/126
  4. Comprobar conectividad IPv6 entre ambos
    - Alumno A ==> ping6 Direccion\_IPv6\_Alumno\_B
    - Alumno B ==> ping6 Direccion\_IPv6\_Alumno\_A
- (\*) Este ejercicio no proporciona conectividad IPv6 global, solo entre los alumnos A y B

# E1: Establecimiento túnel 6in4 (2)

- Scripts de creación de túneles 6in4
  - Windows con netsh desde ventana comandos
    - netsh interface ipv6 add v6v4tunnel "Tunel01" Direccion\_IPv4\_local Direccion\_IPv4\_remota
    - netsh interface ipv6 add address "Tunel01" Direccion\_IPv6
    - netsh interface ipv6 add route ::/0 "Tunel01" Direccion\_gateway\_IPv6 publish=yes
  - Linux/UNIX (desde ventana de comandos)
    - modprobe ipv6
    - ip tunnel add Tunel01 mode sit remote Direccion\_IPv4\_remota local Direccion\_IPv4\_local ttl 255
    - ip link set Tunel01 up
    - ip addr add Direccion\_IPv6/126 dev Tunel01
    - ip route add 2000::/3 dev Tunel01
  - FreeBSD
    - gifconfig gif0 Direccion\_IPv4\_local Direccion\_IPv4\_remota
    - ifconfig gif0 inet6 Direccion\_IPv6 Direccion\_gateway\_IPv6 prefixlen 128
    - route -n add -inet6 default Direccion\_gateway\_IPv6

# E1: Establecimiento túnel 6in4 (3)

- Scripts de creación de túneles 6in4
  - FreeBSD >= 4.4
    - ifconfig gif0 create
    - ifconfig gif0 tunnel Direccion\_IPv4\_local Direccion\_IPv4\_remota
    - ifconfig gif0 inet6 Direccion\_IPv6 Direccion\_gateway\_IPv6 prefixlen 128
    - route add -inet6 default Direccion\_gateway\_IPv6
  - NetBSD
    - ifconfig gif0 Direccion\_IPv4\_local Direccion\_IPv4\_remota
    - ifconfig gif0 inet6 Direccion\_IPv6 Direccion\_gateway\_IPv6 prefixlen 128
    - route -n add -inet6 default Direccion\_gateway\_IPv6
  - OpenBSD
    - ifconfig gif0 giftunnel Direccion\_IPv4\_local Direccion\_IPv4\_remota
    - ifconfig gif0 inet6 Direccion\_IPv6 Direccion\_gateway\_IPv6 prefixlen 128
    - route -n add -inet6 default Direccion\_gateway\_IPv6

# E2: Eliminación túnel 6in4 (1)

- Ejercicio para ser realizado por cada alumno
- El alumno elimina el túnel creado anteriormente según el script de configuración de su Sistema Operativo
- Comprobará que ya no existe el túnel usando:
  - ipconfig en Windows XP/2003
  - Ifconfig en Unix/Linux/\*BSD



# E2: Eliminación túnel 6in4 (2)

- Scripts de eliminación de túneles 6in4
  - Windows con netsh desde ventana comandos
    - netsh interface ipv6 del route ::/0 "Tunel01" Direccion\_gateway\_IPv6
    - netsh interface ipv6 del address "Tunel01" Direccion\_IPv6
    - netsh interface ipv6 del int "Tunel01"
  - Linux/UNIX (desde ventana de comandos)
    - ip route del 2000::/3 dev Tunel01
    - ip addr del Direccion\_IPv6/126 dev Tunel01
    - ip link set Tunel01 down
    - ip tunnel del Tunel01 mode sit remote Direccion\_IPv4\_remota local Direccion\_IPv4\_local ttl 255
  - FreeBSD
    - route delete -inet6 default
    - ifconfig gif0 inet6 delete Direccion\_IPv6
    - ifconfig gif0 down

# E2: Eliminación túnel 6in4 (3)

- Scripts de eliminación de túneles 6in4
  - FreeBSD >= 4.4
    - route delete -inet6 default Direccion\_gateway\_IPv6
    - ifconfig gif0 inet6 Direccion\_IPv6 prefixlen 128 delete
    - ifconfig gif0 delete
  - NetBSD
    - route delete -inet6 default
    - ifconfig gif0 inet6 delete Direccion\_IPv6
    - ifconfig gif0 down
  - OpenBSD
    - ifconfig gif0 inet6 delete Direccion\_IPv6
    - ifconfig gif0 deletetunnel
    - ifconfig gif0 down
    - route delete -inet6 default

# E3: Conectividad IPv6 con un TB

1. Elegir un TB de <http://www.ipv6tf.org/using/connectivity/test.php>
  - Se recomienda <http://tb4.consulintel.euro6ix.org>
2. Seguir los pasos necesarios para obtener conectividad global IPv6
3. Comprobar que se tiene conectividad IPv6
  - ping6, traceroute6 (ping y tracert en windows)
    - [www.kame.net](http://www.kame.net), [www.6net.org](http://www.6net.org), [www.ipv6.org](http://www.ipv6.org)
  - Navegación web a los mismos sitios
4. Comprobar path a la dirección IPv6 asignada desde un looking glass externo
  - <http://www.ipv6.udg.mx/lg.php>
  - [http://www.ipv6tf.org/using/connectivity/looking\\_glass.php](http://www.ipv6tf.org/using/connectivity/looking_glass.php)
  - <http://www.v6.dren.net/lg/>

# E4: Conectividad IPv6 con 6to4 (1)

1. Elegir un 6to4 relay de <http://www.ipv6tf.org/using/connectivity/6to4.php>
  - Se recomienda [6to4.autotrans.consulintel.com](http://6to4.autotrans.consulintel.com)
2. Seguir el script de configuración en función del Sistema Operativo
3. Comprobar que se tiene conectividad IPv6
  - ping6, traceroute6 (ping y tracert en windows)
    - [www.kame.net](http://www.kame.net), [www.6net.org](http://www.6net.org), [www.ipv6.org](http://www.ipv6.org)
  - Navegación web a los mismos sitios
4. Comprobar path a la dirección IPv6 asignada desde un looking glass externo
  - <http://www.ipv6.udg.mx/lg.php>
  - [http://www.ipv6tf.org/using/connectivity/looking\\_glass.php](http://www.ipv6tf.org/using/connectivity/looking_glass.php)
  - <http://www.v6.dren.net/lg/>



# E4: Conectividad IPv6 con 6to4 (2)

- Scripts de eliminación de túneles 6to4
  - Windows con netsh desde ventana comandos
    - netsh int ipv6 6to4 set relay Direccion\_6TO4\_RELAY enabled 1440
  - Linux/UNIX (desde ventana de comandos)
    - ip tunnel add tun6to4 mode sit ttl 80 remote any local Direccion\_publica\_IPv4\_local
    - ip link set dev tun6to4 up
    - ip -6 addr add 2002:XXYY:ZZUU::1/16 dev tun6to4
    - ip -6 route add 2000::/3 via ::192.88.99.1 dev tun6to4 metric 1
  - Note que XXYY:ZZUU es la notación hexadecimal para Direccion\_publica\_IPv4\_local (la direccion IPv4 pública) según lo siguiente:
    - Direccion\_publica\_IPv4\_local = 60.172.21.22 -> 60 -> 3C
    - 172 -> AC
    - 21 -> 15
    - 222 -> DE
  - 60.172.21.22 -> XXYY:ZZUU = 3CAC:15DE

# E4: Conectividad IPv6 con 6to4 (3)

- Scripts de eliminación de túneles 6in4
  - \*BSD
    - Asegurese de que hay al menos un interfaz stf(4) configurado en el kernel
      - En <http://www.netbsd.org/Documentation/kernel/> puede encontrar información sobre ello
    - ifconfig stf0 inet6 2002:XXYY:ZZUU::1 prefixlen 16 alias
    - route add -inet6 default 2002:c058:6301::1
  - Note que XXYY:ZZUU es la notación hexadecimal para Direccion\_publica\_IPv4\_local (la direccion IPv4 publica) segun lo siguiente:
    - Direccion\_publica\_IPv4\_local = 60.172.21.22 -> 60 -> 3C
    - 172 -> AC
    - 21 -> 15
    - 222 -> DE
  - 60.172.21.22 -> XXYY:ZZUU = 3CAC:15DE

# E5: Configurar un 6to4 Relay (1)

## Windows 2003

- La configuración de un 6to4 Relay en el caso de Windows 2003 se hace con comandos **netsh interface ipv6 set**
  - netsh interface ipv6 set interface interface="Conexión de área local" forwarding=enabled
  - netsh interface ipv6 set state state=enabled undoonstop=disabled
  - netsh interface ipv6 set relay name=192.88.99.1 state=enabled interval=1440
    - Se puede sustituir la dirección anycast 192.88.99.1 por cualquiera de <http://www.ipv6tf.org/using/connectivity/6to4.php>
  - netsh interface ipv6 set routing routing=enabled sitelocals=enabled
- Cualquier paquete 6to4 recibido por el interfaz “Conexión de área local” será reenviado al destino IPv6 adecuado
- Para comprobar la configuración del 6to4 relay se puede establecer un túnel 6to4 en una máquina diferente y poner como 6to4 server la dirección IPv4 del 6to4 relay recién configurado
  - Hacer ping y tracert para comprobar conectividad

# E5: Configurar un 6to4 Relay (2)

## Windows Vista y 7

- La configuración de un 6to4 Relay en el caso de Windows Vista y 7 se hace con comandos **netsh interface ipv6 6to4 set**
  - netsh interface ipv6 6to4 set interface "Conexión de área local" routing=enabled
  - netsh interface ipv6 6to4 set state state=enabled undoonstop=disabled
  - netsh interface ipv6 6to4 set relay name=192.88.99.1 state=enabled interval=1440
    - Se puede sustituir la dirección anycast 192.88.99.1 por cualquiera de <http://www.ipv6tf.org/using/connectivity/6to4.php>
  - netsh interface ipv6 6to4 set routing routing=enabled sitelocals=enabled
- Cualquier paquete 6to4 recibido por el interfaz “Conexión de área local” será reenviado al destino IPv6 adecuado
- Para comprobar la configuración del 6to4 relay se puede establecer un túnel 6to4 en una máquina diferente y poner como 6to4 server la dirección IPv4 del 6to4 relay recién configurado
  - Hacer ping y tracert para comprobar conectividad



# E6: Configurar un Cliente Teredo Windows XP/2003

- Desde XP/2003, Windows incorpora implementaciones de cliente Teredo
- En Windows Vista y 7 el cliente Teredo ya viene activado por defecto
- En XP/2003 hay que activarlo según se describe abajo
- Desde una ventana DOS hay que introducir el comando
  - set teredo client teredo.ipv6.microsoft.com. 60 34567
  - se emplea un Teredo Server público de microsoft
    - teredo.ipv6.microsoft.com
- Existen otros Teredo Server/Relays experimentales, (sin servicio garantizado)
  - teredo.ipv6.vol.cz
  - teredo.ipv6.wind.com
  - teredo.via.ecp.fr
- Comprobar la dirección IPv6 obtenida
  - Ipconfig
- Comprobar datos del interfaz Teredo
  - netsh int ipv6 show teredo
  - netsh int ipv6 show int teredo
- No se tiene conectividad global IPv6 porque Microsoft no proporciona ningún Teredo Relay
- Sí se tiene conectividad IPv6 con otro cliente Teredo
  - Comprobar haciendo ping a la dirección de otro Teredo Client de otro alumno
- Implementaciones de clientes Teredo para otros Sistemas Operativos
  - Linux: <http://www.simpnalempin.com/dev/miredo/>
  - FreeBSD: <http://www-rp.lip6.fr/teredo/>

# E7: Uso de Proxy IPv4/IPv6 (1)

- Un Proxy IPv4/IPv6 no es lo mismo que un mecanismo de traducción IPv4/IPv6 (NAT-PT)
- El Proxy es un intermediario que trabaja en el nivel de aplicación
  - Recibe una conexión TCP sobre un protocolo (IPv4/IPv6) y extrae los datos del nivel de aplicación
  - Establece conexión TCP (IPv6/IPv4) sobre el destino e introduce los datos del nivel de aplicación extraídos en el paso anterior
- Por tanto permite la conexión entre
  - Cliente IPv4 ==> Proxy IPv4/IPv6 ==> Servidor IPv6
  - Cliente IPv6 ==> Proxy IPv6/IPv4 ==> Servidor IPv4
- Existen dos Proxys bien conocidos
  - 46Bouncer (Windows y Linux)
  - Proxy de Windows disponible en XP/2003 y posteriores

# E7: Uso de Proxy IPv4/IPv6 (2)

- Configurar un Proxy en Windows con comandos netsh
- Proxy IPv4/IPv6
  - Puerto 8220 IPv4 redirigirlo al puerto 80 de www.kame.net (2001:200:0:8002:203:47ff:fea5:3085)
  - **netsh int port set v4tov6** Puerto\_v4\_TCP\_local Direccion\_IPv6\_remota Puerto\_v6\_TCP\_remoto Direccion\_IPv4\_local
  - Ejemplo: netsh int port set v4tov6 8220 2001:200:0:8002:203:47ff:fea5:3085 80 Direccion\_IPv4\_local
  - Comprobar con
    - netsh int port show all
    - http://direccion\_IPv4\_local:8220
- Proxy IPv6/IPv4
  - Puerto 8330 IPv6 redirigirlo al puerto 80 de www.kame.net (203.178.141.194)
  - Ejemplo: netsh int port set v6tov4 8330 203.178.141.194 80 Direccion\_IPv6\_local
  - Comprobar con
    - netsh int port show all
    - http://[direccion\_IPv6\_local]:8330



# Parte 4

## Ejemplos de aplicaciones varias



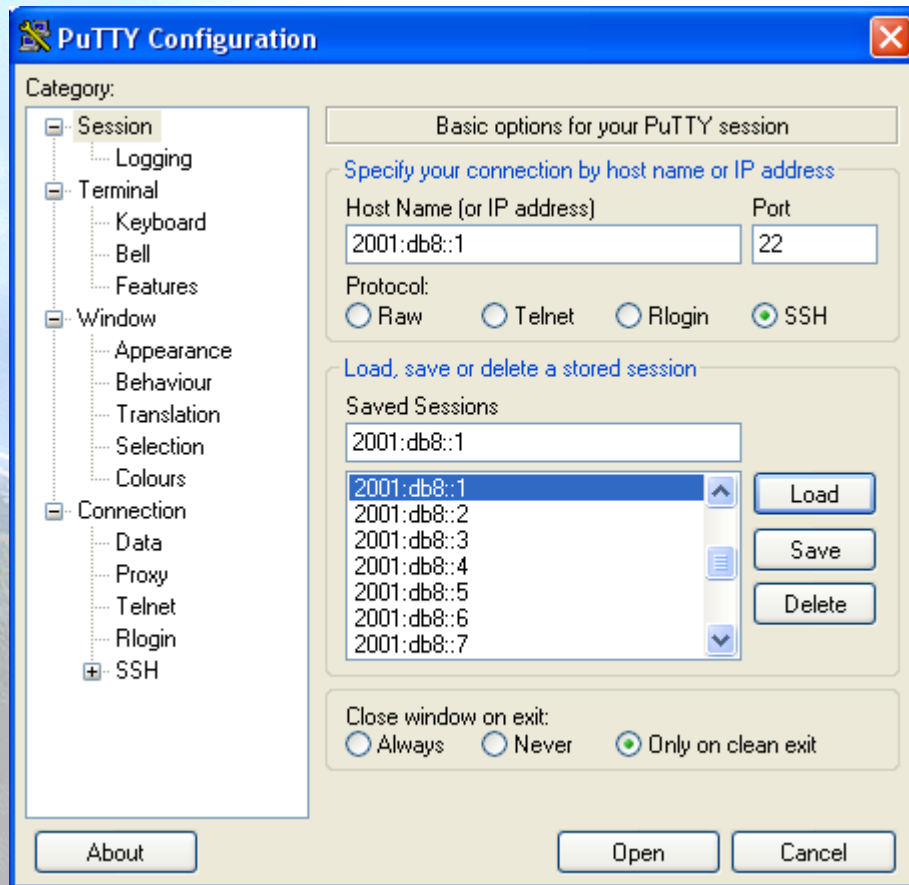
# Aplicaciones IPv6 (1)

- Modelo Cliente-Servidor implica que se pueden tener aplicaciones clientes y/o servidores que sean:
  - Sólo IPv4
  - Sólo IPv6
  - IPv4 + IPv6
- Esto proporciona un conjunto de combinaciones que deben tenerse en cuenta conjuntamente con la existencia o no de conectividad IPv4 y/o IPv6

# Aplicaciones IPv6 (2)

- Para diferenciar o indicar la accesibilidad de un servicio mediante IPv4 y/o IPv6 se utiliza la **resolución DNS**
- Cuando un cliente quiere conectar con servicio.ejemplo.com al resolver el nombre puede obtener una dirección IPv4, IPv6 o ambas.
- En este último caso es decisión del cliente elegir el protocolo (v4/v6) usado para comunicarse. El caso general es intentar v6 por defecto primero

# Aplicaciones IPv6 (3)



- **Putty**
- Cliente IPv4/IPv6 de Telnet y SSH
- Muy útil para Gestión y Administración de equipos
- <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

# Aplicaciones IPv6 (4)

- **Ethereal y Wireshark**
- Captura y Decodifica Trafico IPv4/IPv6
- Muy útil validación de conexiones y solución de problemas
- <http://www.ethereal.com> y <http://www.wireshark.org>

Marvell Yukon Ethernet Controller: Capturing - Wireshark

Filter: Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
1876	145.774985	Hitachi_28:a0:e1	Broadcast	ARP	who has 10.0.0.248? Tell 10.0.0.4
1877	147.761561	2a01:48:1:1::ff1	2a01:48:1:1:2110:7263:8cb7:1ab3	ICMPv6	Neighbor solicitation
1878	147.761743	2a01:48:1:1:2110:7263:8cb7:1ab3	ff02::1:ff00:ff1	ICMPv6	Neighbor solicitation
1879	147.763405	2a01:48:1:1::ff1	2a01:48:1:1:2110:7263:8cb7:1ab3	ICMPv6	Neighbor advertisement
1880	147.763427	2a01:48:1:1:2110:7263:8cb7:1ab3	2a01:48:1:1::ff1	ICMPv6	Neighbor advertisement
1881	147.774932	Hitachi_28:a0:e1	Broadcast	ARP	who has 10.0.0.248? Tell 10.0.0.4
1882	148.629369	fe80::3572:2b60:e9dd:aeca	fe80::200:87ff:fe28:a0e1	ICMPv6	Neighbor solicitation
1883	148.630528	fe80::200:87ff:fe28:a0e1	fe80::3572:2b60:e9dd:aeca	ICMPv6	Neighbor advertisement
1884	149.775047	Hitachi_28:a0:e1	Broadcast	ARP	who has 10.0.0.248? Tell 10.0.0.4

Frame 1879 (86 bytes on wire, 86 bytes captured)

Ethernet II, Src: Hitachi\_28:a0:e1 (00:00:87:28:a0:e1), Dst: QuantaCo\_45:7c:68 (00:1e:68:45:7c:68)

Internet Protocol Version 6

0110 .... = Version: 6  
.... 0000 0000 .... = Traffic class: 0x00000000  
.... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000  
Payload length: 32  
Next header: ICMPv6 (0x3a)  
Hop limit: 255  
Source: 2a01:48:1:1::ff1 (2a01:48:1:1::ff1)  
Destination: 2a01:48:1:1:2110:7263:8cb7:1ab3 (2a01:48:1:1:2110:7263:8cb7:1ab3)

Internet Control Message Protocol v6

Type: 136 (Neighbor advertisement)  
Code: 0  
Checksum: 0x93f7 [correct]  
Flags: 0xe0000000  
Target: 2a01:48:1:1::ff1  
ICMPv6 option (Target link-layer address)  
Type: Target link-layer address (2)  
Length: 8  
Link-Layer address: 00:00:87:28:a0:e1

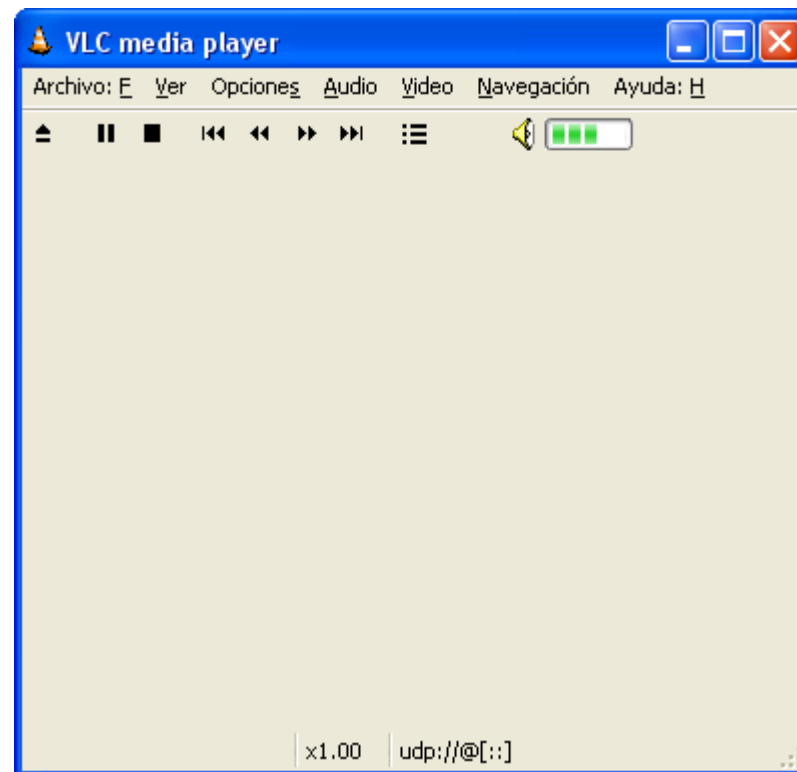
```
0000 00 1e 68 45 7c 68 00 00 87 28 a0 e1 86 dd 60 00  ..he|h.. (. ...
0010 00 00 00 20 3a ff 2a 01 00 48 00 01 00 01 00 00  *..H...!
0020 00 00 00 00 0f f1 2a 01 00 48 00 01 00 01 21 10  *..H...!
0030 72 63 8c b7 1a b3 88 00 93 f7 e0 00 00 00 2a 01  .c...*...*
0040 00 48 00 01 00 01 00 00 00 00 00 00 0f f1 02 01  .H...*...
0050 00 00 87 28 a0 e1  ..(.
```

Internet Protocol Version 6 (ipv6), 40 bytes | Packets: 2084 Displayed: 2084 Marked: 0 | Profile: Default



# Aplicaciones IPv6 (5)

- **VLC**
- Cliente y Servidor Multimedia
- Soporta Unicast y Multicast
- <http://www.videolan.org/vlc/>



# Aplicaciones IPv6 (6)

- **VNC**
  - Conexión remota a un PC sobre IPv6
  - Entorno gráfico
- **Modelo cliente/servidor**
  - Servidor en la máquina a la que se pretende acceder
  - Cliente en la maquina local que se conecta a la remota
- **Sistemas Operativos soportados**
  - Windows XP
  - Linux
- **Descargar de**
  - <http://jungla.dit.upm.es/~acosta/paginas/vncIPv6.html>

# Aplicaciones IPv6 (7)

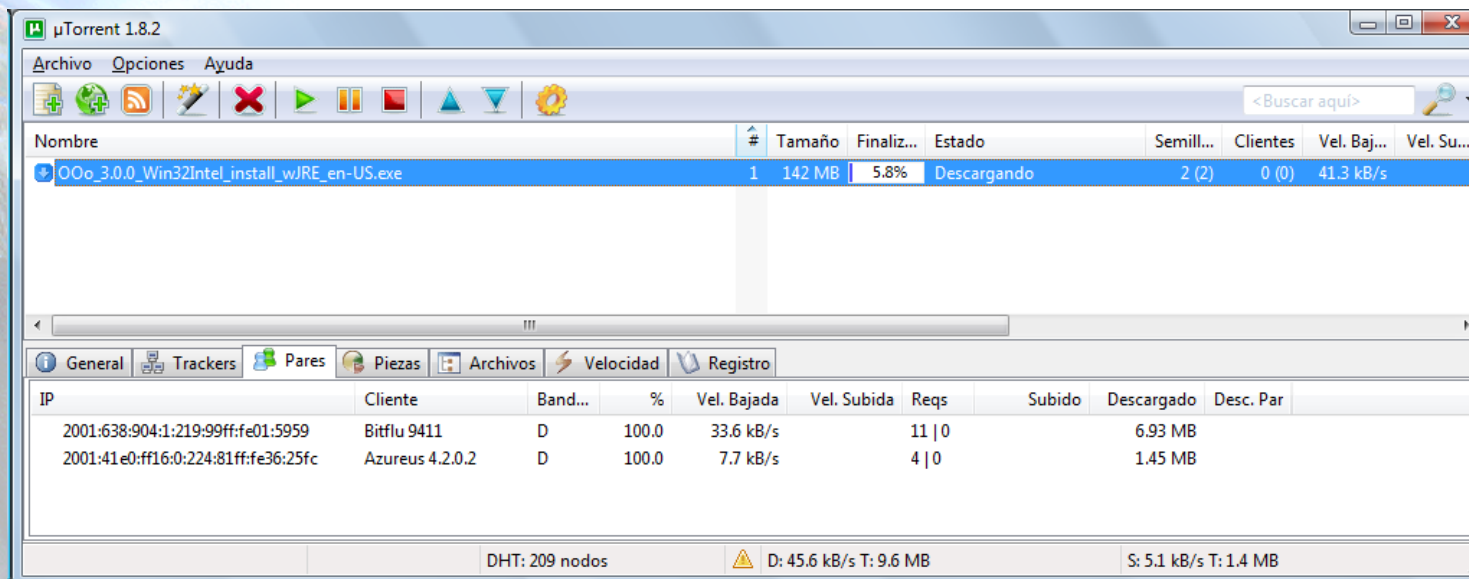
- **Web**
- **Cientes:**  
prácticamente todos:  
Firefox, IE,  
Konqueror, Opera,  
Safari
- **Servidores:** Apache,  
ISS

The screenshot shows the 'The IPv6 Portal' website in a Mozilla Firefox browser window. The browser's address bar displays 'http://www.ipv6tf.org/news/newsroom.php'. The website header features the IPv6 logo and navigation links for 'meet IPv6', 'IPv6 news', 'using IPv6', 'IPv6 guide', 'the portal', and 'your IPv6 portal'. A large yellow '2' is prominently displayed on the left side of the page. The main content area is titled 'newsroom' and includes a 'Main Headlines' section with several news items, such as 'Barcelona 2005 Global IPv6 Summit slides available' and 'The Catalanian Telematic Educational Network Demonstrates IPv6 Deployment'. A sidebar on the right contains a search box, a 'who's online?' section, and a 'LOGIN' field. The browser's status bar at the bottom shows 'Done'.

# Aplicaciones IPv6 (8)

- **BitTorrent**
- Clientes IPv6: Azureus, BitTornado, uTorrent, etc.
- Trackers IPv6:
  - NIIF/HUNGARNET <http://ipv6.niif.hu/index.php?mn=3&sm=6&lg=en>
  - SixXS <http://www.sixxs.net/tools/tracker/>
  - The Pirate Bay <http://thepiratebay.org/>

3.612.575 usuario registrados. Ultimo subido 10:34:04.  
IPv4 17.480.629 peers (8.996.836 seeders + 8.483.793 leechers) en 1.677.348 torrents en el tracker.  
IPv6 8.434 peers (3.783 seeders + 4.651 leechers) en 92.270 torrents en el tracker.



The screenshot shows the uTorrent 1.8.2 application window. At the top, there is a menu bar with 'Archivo', 'Opciones', and 'Ayuda'. Below the menu is a toolbar with various icons. The main area displays a download progress bar for the file 'OOo\_3.0.0\_Win32Intel\_install\_wJRE\_en-US.exe', which is 142 MB in size and 5.8% downloaded. Below the progress bar, there is a table showing the details of the peers connected to the tracker.

IP	Cliente	Band...	%	Vel. Bajada	Vel. Subida	Reqs	Subido	Descargado	Desc. Par
2001:638:904:1:219:99ff:fe01:5959	Bitflu 9411	D	100.0	33.6 kB/s		11   0		6.93 MB	
2001:41e0:ff16:0:224:81ff:fe36:25fc	Azureus 4.2.0.2	D	100.0	7.7 kB/s		4   0		1.45 MB	

At the bottom of the window, there is a status bar showing 'DHT: 209 nodos', 'D: 45.6 kB/s T: 9.6 MB', and 'S: 5.1 kB/s T: 1.4 MB'.



# Aplicaciones IPv6 (9)

- **IPv6 to Standard**

- Listado de dispositivos, aplicaciones y servicios que soportan IPv6
- <http://www.ipv6-to-standard.org>

**IPv6 to Standard**

The IETF [IPv6](#) and [IPv6 Maintenance](#) working groups have started the process to advance the core IPv6 specifications to the last step in the IETF standardization process (e.g., Standard). IETF protocols are elevated to the Internet Standard level when significant implementation and successful operational experience has been obtained. Vendors with IPv6 products are encouraged to participate in this process by identifying their IPv6-enabled products by means of this web page.

Check IPv6 RFCs Status [here](#).

Type: All  
Subtype: All  
Product or Application or Service: All  
Vendor or Author / Name: All  
Free search:

If you can't find your Product, Service or Application in this page, please, [submit it](#).

**Total (1907)**

Applications(702)			
<i>End User Applications(260)</i>			
Audio and Video Client	<a href="#">Apple / iTunes</a>	<a href="#">Edit</a>	<a href="#">View</a>
Audio and Video Client	<a href="#">Apple / QuickTime</a>	<a href="#">Edit</a>	<a href="#">View</a>
Audio and Video Client	<a href="#">Microsoft / Windows Media Player</a>	<a href="#">Edit</a>	<a href="#">View</a>
Audio and Video Client	<a href="#">MPlayer Team and Kame / MPlayer</a>	<a href="#">Edit</a>	<a href="#">View</a>
Audio and Video Client	<a href="#">Wide / DVTS</a>	<a href="#">Edit</a>	<a href="#">View</a>

# Aplicaciones IPv6 (10)

- **FreeBSD**

- Se pueden usar los ports de FreeBSD:

```
#>cd /usr/ports
```

```
#>make search key="ipv6"
```

- Aparecerá una lista de aplicaciones que soportan IPv6. Entre la información de cada aplicación se encuentra path, que será el directorio a donde nos moveremos y desde donde podemos instalar la aplicación:

```
#>cd path
```

```
#>make install
```

- Esto hará que se comience a buscar en una lista de servidores el código fuente, que se descargará, se compilará y se instalará.
- Se puede sólo descargar el código fuente, que se colocará en /usr/ports/distfiles, haciendo, en vez de make install, make fetch.

# Aplicaciones IPv6: Ejercicio 1 (1)

- **Linux:**

```
# dig a www.ipv6tf.org
```

```
:: QUESTION SECTION:
```

```
;www.ipv6tf.org.      IN      A
```

```
:: ANSWER SECTION:
```

```
www.ipv6tf.org.     172800 IN A    213.172.48.141
```

- **# dig aaaa www.ipv6tf.org**

```
:: QUESTION SECTION:
```

```
;www.ipv6tf.org.      IN      AAAA
```

```
:: ANSWER SECTION:
```

```
www.ipv6tf.org.     172800 IN AAAA 2001:800:40:2a03::3
```

# Aplicaciones IPv6: Ejercicio 1 (2)

- **Linux:**

```
#dig aaaa www.kame.net @2001:800:40:2a03::3
;; QUESTION SECTION:
;www.kame.net.      IN      AAAA
;; ANSWER SECTION:
www.kame.net. 86400 IN AAAA
                2001:200:0:8002:203:47ff:fea5:3085
;; Query time: 400 msec
;; SERVER:
                2001:800:40:2a03::3#53(2001:800:40:2a03::3)
;; WHEN: Fri Jun 24 13:49:41 2005
;; MSG SIZE rcvd: 107
```



# Aplicaciones IPv6: Ejercicio 1 (3)

- **Windows**

```
C:\>nslookup
```

```
>set type=a
```

```
>www.ipv6tf.org
```

```
Name: www.ipv6tf.org
```

```
Address: 213.172.48.141
```

```
>set type=aaaa
```

```
>www.ipv6tf.org
```

```
www.ipv6tf.org AAAA IPv6 address =  
2001:800:40:2a03::3
```

# Aplicaciones IPv6: Ejercicio 2

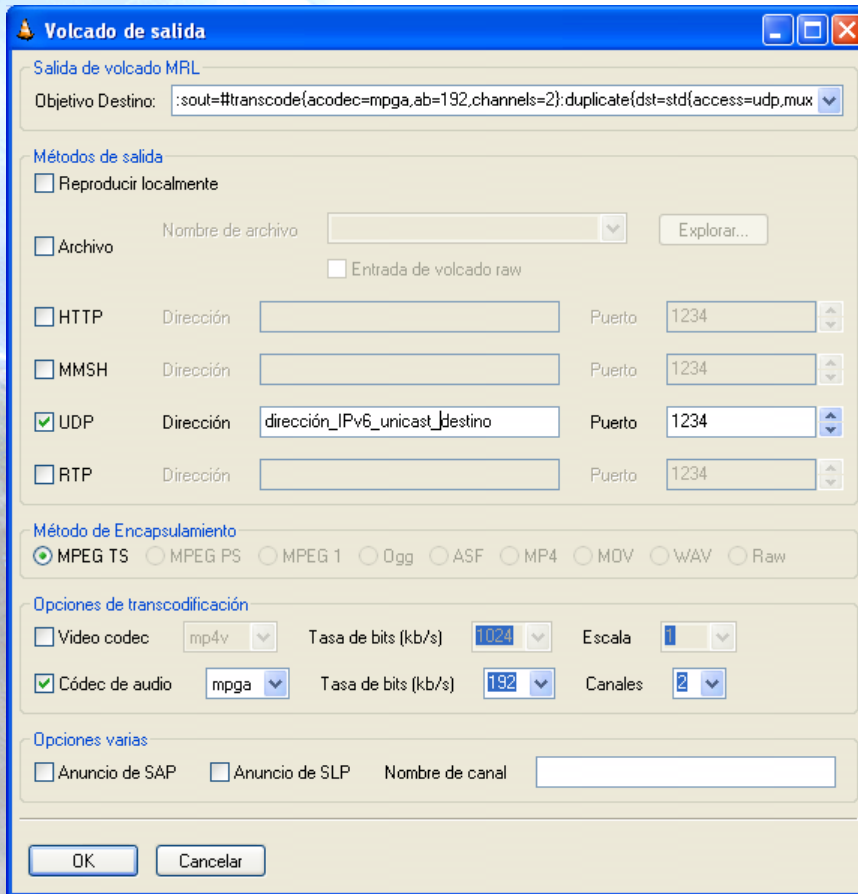
- Instalar (si no lo están ya):
  - Cliente SSH con soporte IPv6 (Putty)
  - Cliente FTP (Línea de comandos en BSD, Linux, Windows)
  - Navegador Web (Firefox, IE, Safari, etc.)
  - Wireshark
  - VLC
  - VNC
  - BitTorrent

# Aplicaciones IPv6: Ejercicio 3

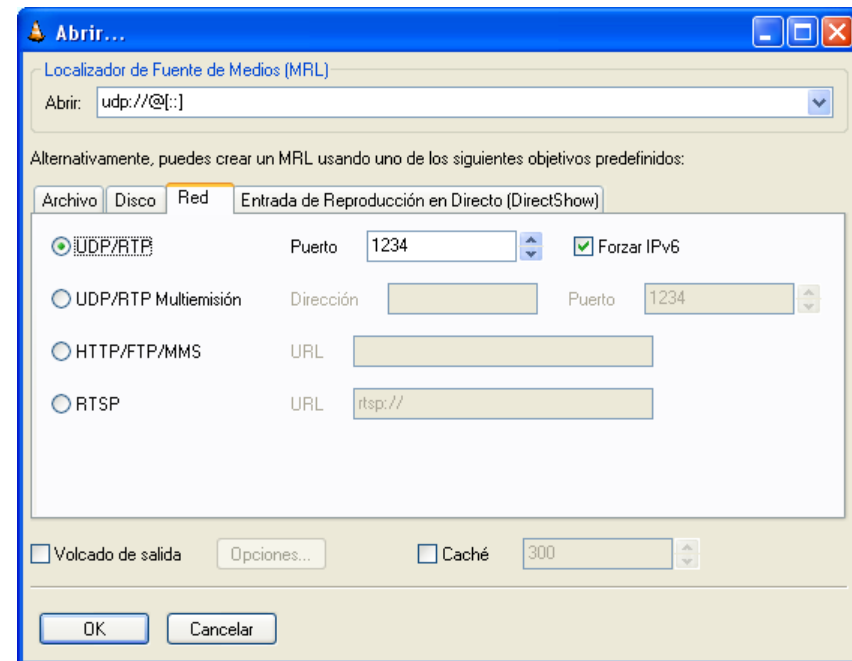
- Acceder a los distintos servicios mientras se capturan paquetes con Wireshark (en su defecto usar tcpdump).
- Usar el cliente ssh para acceder mediante v4 o v6 diferenciando mediante DNS
- Usar el cliente ssh para acceder mediante v4 o v6 diferenciando mediante un parámetro de la aplicación (linux: #ssh -6|-4) (XP: ping -6|-4)

# Aplicaciones IPv6: Ejercicio 3

- VLC con Unicast



Servidor

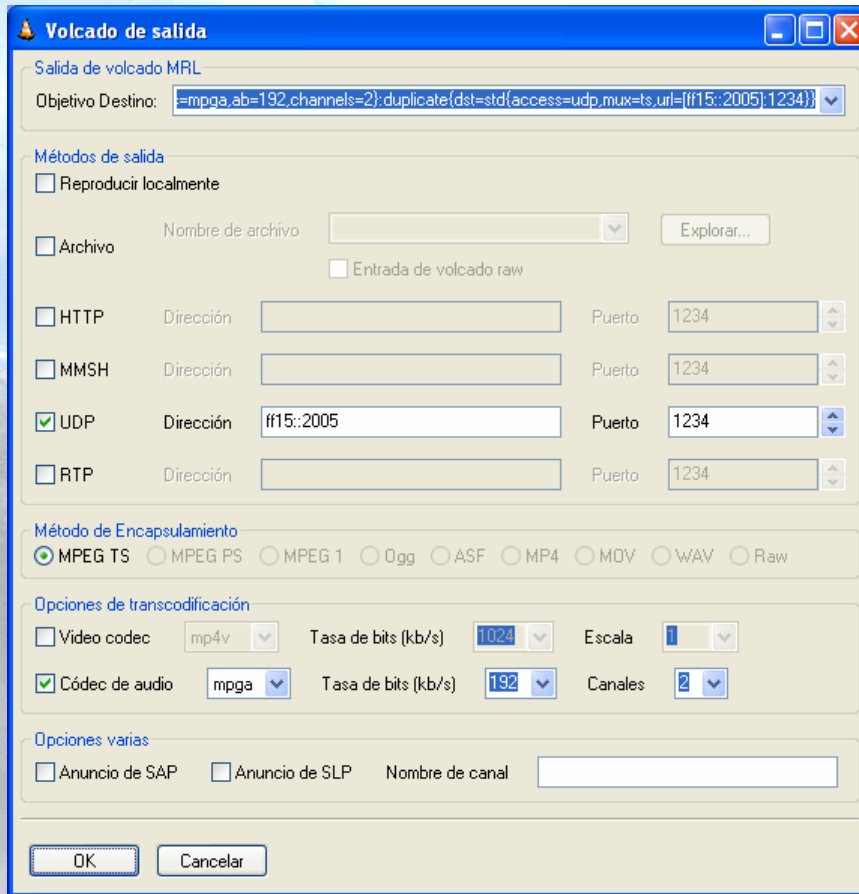


Cliente

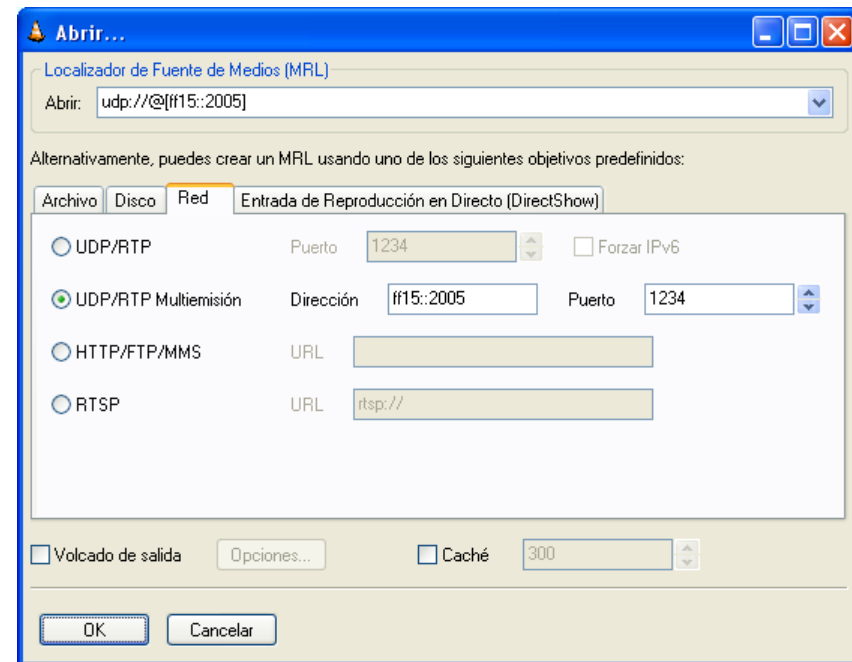


# Aplicaciones IPv6: Ejercicio 4

- VLC con Multicast

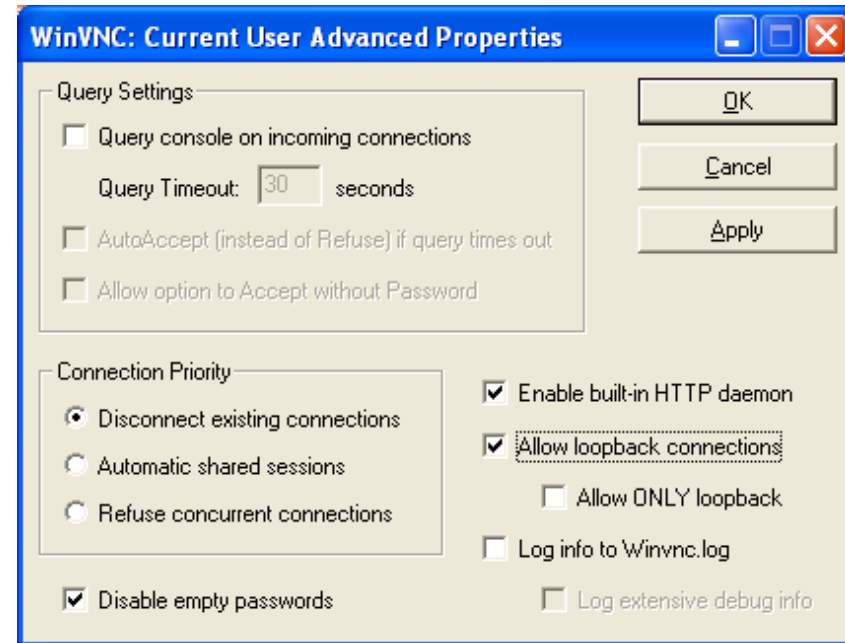
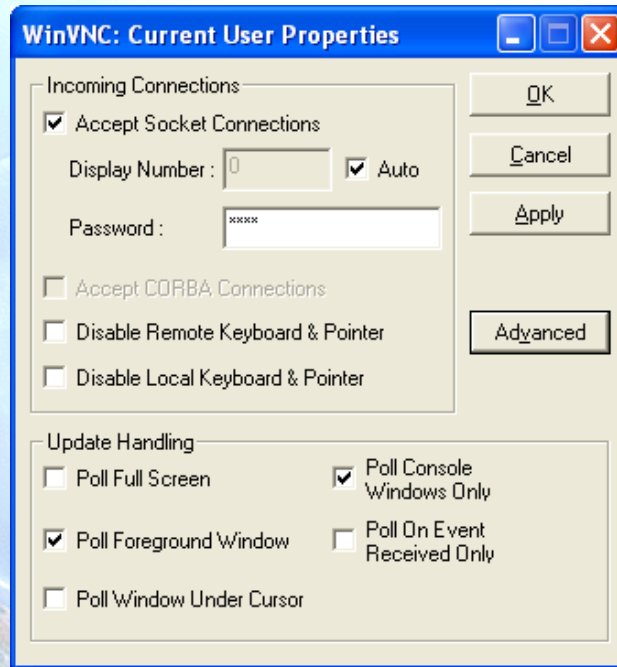


Servidor



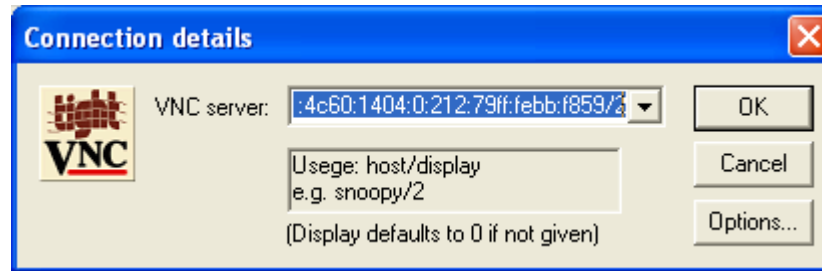
Cliente

# Aplicaciones IPv6: Ejercicio 5 (1)



- **VNC Server Properties**
  - Se configura el número de display para recibir conexiones
    - Por defecto es 0
  - También el password
- VNC Server Properties ==> Advanced
  - Habilitar “allow loopback connections”

# Aplicaciones IPv6: Ejercicio 5 (2)



- VNC client
  - Se especifica el VNC server
    - Directamente con la dirección IPv6
    - Un nombre DNS
  - El display se añade a continuación del VNC server
    - Se especifica con un número separado del VNC server con el carácter ‘/’

# Referencias (1)

- [6in4] RFC1933
- [TunAut] RFC1933
- [6to4] RFC3056
- [6over4] RFC2529
- [TB] RFC3053
- [TSP] draft-vg-ngtrans-tsp-01, <http://www.hexago.com/index.php?pgID=step1>
- [TEREDO] draft-huitema-v6ops-teredo-05
- [TEREDOC] <http://www.microsoft.com/technet/prodtechnol/winxpro/maintain/teredo.msp>
- [ISATAP] draft-ietf-ngtrans-isatap-24
- [AYIYA ] draft-massar-v6ops-ayiya-02
- [SILKROAD ] draft-liumin-v6ops-silkroad-02
- [DSTM] draft-ietf-ngtrans-dstm-10
- [SIIT] RFC2765
- [NATPT] RFC2767
- [BIS] RFC2767
- [TRT] RFC3142
- [SOCKSv64 ] RFC3089



# Referencias (2)

- [PROTO41] draft-palet-v6ops-proto41-nat-04
- [STUN] RFC3489
- [NATPTIMPL]
  - <http://www.ipv6.or.kr/english/download.htm> ==> Linux 2.4.0
  - [http://www.ispras.ru/~ipv6/index\\_en.html](http://www.ispras.ru/~ipv6/index_en.html) ==> Linux y FreeBSD
  - <http://research.microsoft.com/msripv6/napt.htm> Microsoft
  - <ftp://ftp.kame.net/pub/kame/snap/kame-20020722-freebsd46-snap.tgz> ==> KAME snapshot (22.7.2002)
  - <http://ultima.ipv6.bt.com/>
- [STATELESS] RFC2462
- [STATEFULL] RFC3315
- [PRIVACY] RFC3041
- Windows IPv6
  - [http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/sag\\_ip\\_v6\\_add\\_utils.msp](http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/sag_ip_v6_add_utils.msp)
  - <http://www.microsoft.com/technet/community/columns/cableguy/cg0902.msp>.