

# IPv6 Startup

## IPv6 Tour

La Habana, Cuba  
6 de Octubre de 2005

**Miguel Angel Díaz (miguelangel.diaz@consulintel.es)**

**César Olvera (cesar.olvera@consulintel.es)**

**Jordi Palet (jordi.palet@consulintel.es)**

**Alvaro Vives (alvaro.vives@consulintel.es)**

# Agenda

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



# Parte 1

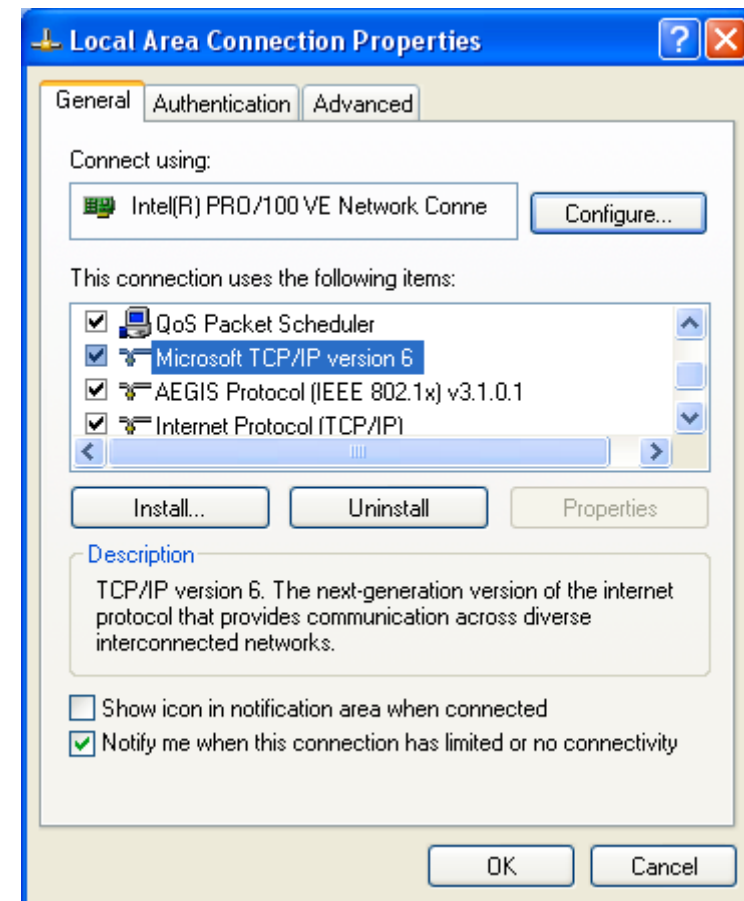
## Instalación de IPv6 en varias plataformas (XP/2003, Linux, BSD)

# Instalación de IPv6: XP/2003 (1)

- En una ventana de 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 Conection > Properties
- También se puede instalar/desinstalar desde aquí





# Instalación de IPv6: XP/2003 (3)

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

# Instalación de IPv6: Linux (1)

- 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

O

#/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 autoconfiguracion:
    # up echo 0 > /proc/sys/net/ipv6/conf/all/autoconf
    netmask 64
    # El router esta 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)

- Instalar Pila (Versiones 4.5+):
- Soporte muy bueno, la pila ya viene instalada





## Parte 2

# Configuración básica Stateless/Stateful, privacidad, rutas estáticas

# Configuración básica: XP/2003 (1)

- Comandos Básicos en XP/2003
- Sirven para obtener información sobre el estado y realizar la configuración de interfaces, direcciones, caches, rutas, etc.
- Dos grupos de comandos:
  - **ipv6.exe** (hasta Windows 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** (desde Windows XP SP2 y Server 2003)
    - Opcion de store=active|persistent para guardar cambios
- Equivalencias en:  
<http://www.microsoft.com/windowsserver2003/technologies/ipv6/ipv62netshtable.msp>

# Configuración básica: XP (2)

- **Comandos “ipv6”**
  - `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: XP/2003 (3)

- **Comandos “netsh interface ipv6”**
  - 6to4 - Changes to the `netsh interface ipv6 6to4' context
  - ? - Displays a list of commands
  - add - Adds a configuration entry to a table
  - delete - Deletes a configuration entry from a table
  - dump - Displays a configuration script
  - help - Displays a list of commands
  - install - Installs IPv6
  - isatap - Changes to the `netsh interface ipv6 isatap' context
  - renew - Restarts IPv6 interfaces
  - reset - Resets IPv6 configuration state
  - set - Sets configuration information
  - show - Displays information
  - uninstall - Uninstalls IPv6



# Configuración básica: XP/2003 (4)

- **Comandos “netsh interface ipv6 add”**
  - add 6over4tunnel - Creates a 6over4 interface.
  - add address - Adds an IPv6 address on an interface.
  - add dns - Adds a static DNS server address.
  - add prefixpolicy - Adds a prefix policy entry.
  - add route - Adds an IPv6 route over an interface.
  - add v6v4tunnel - **Creates an IPv6-in-IPv4 point-to-point tunnel.**
- **Comandos “netsh interface ipv6 set”**
  - set address - Modifies IPv6 address information.
  - set global - Modifies global configuration general parameters.
  - set interface - Modifies interface configuration parameters.
  - set mobility - Modifies mobility configuration parameters.
  - set prefixpolicy - Modifies prefix policy information.
  - set privacy - Modifies privacy configuration parameters.
  - set route - Modifies route parameters.
  - set state - Sets the state of deprecated functionality.
  - set teredo - Sets Teredo state.
- **Comandos “netsh interface ipv6 show”**
  - show address - Shows IPv6 addresses.
  - show bindingcacheentries - Shows binding cache entries.
  - show destinationcache - Shows destination cache entries.
  - show dns - Displays the DNS server addresses.
  - show global - Shows global configuration parameters.
  - show interface - Shows interface parameters.
  - show joins - Shows IPv6 multicast addresses.
  - show mobility - Shows mobility configuration parameters.
  - show neighbors - Shows neighbor cache entries.
  - show prefixpolicy - Shows prefix policy entries.
  - show privacy - Shows privacy configuration parameters.
  - show routes - Shows route table entries.
  - show siteprefixes - Shows site prefix table entries.
  - show state - Shows the state of deprecated functionality.
  - show teredo - Shows Teredo service state.

# Configuración básica: XP/2003 (5)

- Información de interfaces
- **ipconfig [/all]**
- **ipv6 [-v] if [IfIndex]**
- 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 : XP/2003 (6)

- Ping en XP/2003
- **ping6 [-t] [-a] [-n count] [-l size] [-w timeout] [-s srcaddr] [-r] dest**
  - t Ping the specified host until interrupted
  - a Resolve addresses to hostnames
  - n count Number of echo requests to send
  - l size Send buffer size
  - w timeout Timeout in milliseconds to wait for each reply
  - s srcaddr Source address to use
  - r Use routing header to test reverse route also
- Basta utilizar el comando **ping** en lugar **ping6** si la petición DNS devuelve un registro AAAA

# Configuración básica : XP/2003 (7)

- **Ejemplos de Ping en XP/2003**
- **ping6 www.ipv6tf.org**

Pinging www.ipv6tf.org [2001:800:40:2a03::3]  
from 2001:800:40:2a05:9c4d:b1cd:98d5:5a32 with 32 bytes of data:

Reply from 2001:800:40:2a03::3: bytes=32 time<1ms

Reply from 2001:800:40:2a03::3: bytes=32 time<1ms

Reply from 2001:800:40:2a03::3: bytes=32 time<1ms

Reply from 2001:800:40:2a03::3: bytes=32 time<1ms

Ping statistics for 2001:800:40:2a03::3:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms



# Configuración básica : XP/2003 (8)

- **Ejemplos de Ping en XP/2003**

- **ping ::1**

Pinging ::1 from ::1 with 32 bytes of data:

Reply from ::1: bytes=32 time<1ms

Reply from ::1: bytes=32 time<1ms

Reply from ::1: bytes=32 time<1ms

Reply from ::1: bytes=32 time<1ms

Ping statistics for ::1:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms

- **ping6 fe80::201:4aff:fe18:26c7 (link-local propia)**

Pinging fe80::201:4aff:fe18:26c7 from fe80::201:4aff:fe18:26c7%5 with 32 bytes of data:

Reply from fe80::201:4aff:fe18:26c7%5: bytes=32 time<1ms

Reply from fe80::201:4aff:fe18:26c7%5: bytes=32 time<1ms

Reply from fe80::201:4aff:fe18:26c7%5: bytes=32 time<1ms

Reply from fe80::201:4aff:fe18:26c7%5: bytes=32 time<1ms

Ping statistics for fe80::201:4aff:fe18:26c7:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms

# Configuración básica : XP/2003 (9)

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

...

Interface 5: Local Area Connection

Internet Address	Physical Address	Type
-----	-----	-----
fe80::201:4aff:fe18:26c7	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 : XP/2003 (9)

- **Ejemplos de Ping en XP/2003**
- **ping fe80::200:87ff:fe28:a0e0%5 (link-local vecino en la interfaz 5)**  
Pinging fe80::200:87ff:fe28:a0e0%5 from fe80::201:4aff:fe18:26c7%5 with 32 bytes of data:  
Reply from fe80::200:87ff:fe28:a0e0%5: bytes=32 time<1ms  
Reply from fe80::200:87ff:fe28:a0e0%5: bytes=32 time<1ms  
Reply from fe80::200:87ff:fe28:a0e0%5: bytes=32 time<1ms  
Reply from fe80::200:87ff:fe28:a0e0%5: bytes=32 time<1ms  
Ping statistics for fe80::200:87ff:fe28:a0e0%5:  
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
Minimum = 0ms, Maximum = 0ms, Average = 0ms

# Configuración básica : XP/2003 (10)

- Traceroute en XP/2003
- **tracert6 [-d] [-h maximum\_hops] [-w timeout] [-s srcaddr] target\_name**
  - d Do not resolve addresses to hostnames
  - h max\_hops Maximum number of hops to search for target
  - w timeout Wait timeout milliseconds for each reply
  - s srcaddr Source address to use
  - r Use routing header to test reverse route also
- Basta utilizar el comando **tracert** en lugar **tracert6** si la petición DNS devuelve un registro AAAA



# Configuración básica : XP/2003 (11)

- **Ejemplos de Traceroute en XP/2003:**
- **tracert www.lacnic.net**

Tracing route to lacnic.net [2001:12ff:0:2::15] over a maximum of 30 hops:

1	1 ms	<1 ms	<1 ms	gr2000-00.consulintel.euro6ix.org [2001:800:40:2a05::1]
2	<1 ms	*	1 ms	2001:800:40:2f02::1
3	4 ms	1 ms	1 ms	2001:800:40:2f01::2
4	10 ms	4 ms	4 ms	data-to-tid.tid.euro6ix.org [2001:800:40:2f1a::2]
5	200 ms	189 ms	189 ms	3ffe:80a::1
6	388 ms	390 ms	388 ms	v6gw.isc.registro.br [2001:4f8:0:1::10:2]
7	396 ms	396 ms	387 ms	lacnic.net [2001:12ff:0:2::15]

Trace complete.

# Configuración básica : XP/2003 (12)

- Agregar una dirección:
- **netsh interface ipv6 add address**  
**InterfaceNameOrIndex IPv6Address**  
**[[type=]unicast|anycast]**  
**[[validlifetime=]Minutes|infinite]**  
**[[preferredlifetime=]Minutes|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 **ipv6 if 5**

# Configuración básica : XP/2003 (13)

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

# Configuración básica : XP/2003 (14)

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



# Configuración básica : XP/2003 (15)

- Agregar una ruta:
- **netsh interface ipv6 add route**  
**[[prefix=]IPv6Address/Integer**  
**[[interface=]String]**  
**[[nexthop=]IPv6Address]**  
**[[siteprefixlength=]Integer]**  
**[[metric=]Integer] [[publish=]{no | yes**  
**| immortal}] [[validlifetime=]{Integer |**  
**infinite}] [[preferredlifetime=]{Integer**  
**| 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 : XP/2003 (16)

- Mostrar rutas:
- **netsh interface ipv6 show routes**  
[[level=]{normal | verbose}]  
[[store=]{active | persistent}]
- Ejemplo: netsh interface ipv6 show routes

Querying active state...

Publish	Type	Met	Prefix	Idx	Gateway/Interface Name
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: XP/2003 (17)

- Eliminar una ruta:
- **netsh interface ipv6 delete route**  
**[prefix=]<IPv6 address>/<integer>**  
**[interface=]<string> [[nexthop=]<IPv6**  
**address>] [[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 routes**

# Configuración básica: XP/2003 (18)

- Añadir un Servidor DNS:
- **netsh interface ipv6 add dns**  
**[[interface=]String]**  
**[[address=]IPv6Address]**  
**[[index=]Integer]**
- Ejemplo: netsh interface ipv6 add dns “Local area network” 2001:7f9:1000:1::947c 1
- El “index” representa la posición del servidor DNS que se ha configurado en la lista de servidores DNS



# Configuración básica: XP/2003 (19)

- Mostrar servidores DNS:
- **netsh interface ipv6 show dns**  
**[[interface=]string]**
- Ejemplo: netsh interface ipv6 show dns

DNS servers in LAN interface

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

# Configuración básica: XP/2003 (20)

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

# Configuración básica: Linux (1)

## Comandos básicos (1)

- ifconfig
- ping6 <hostcondirlPv6>|<dirIPv6>|[-I <interfaz>] <link-local-ipv6address>
- traceroute6 <hostcondirlPv6>|<dirIPv6>
- tracepath6 <hostcondirlPv6>|<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: 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 XP/2003 esta activada por defecto
- **ipconfig** o **ipv6 if** 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)

- En hosts con Windows XP/2003 esta activada por defecto
- **ipconfig** o **ipv6 if** para revisar cual es la dirección autoconfigurada
- Existen dos formas de desactivarlo:
  1. **netsh interface ipv6 set privacy state=disabled store=persistent**
  2. **ipv6 [-p] gpu UseTemporaryAddresses no**
- Para revisar el cambio: “disable” y “enable” la interfaz física en Network Connection, entonces **ipconfig** o **ipv6 if**

# Privacidad (3)

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



## Parte 3

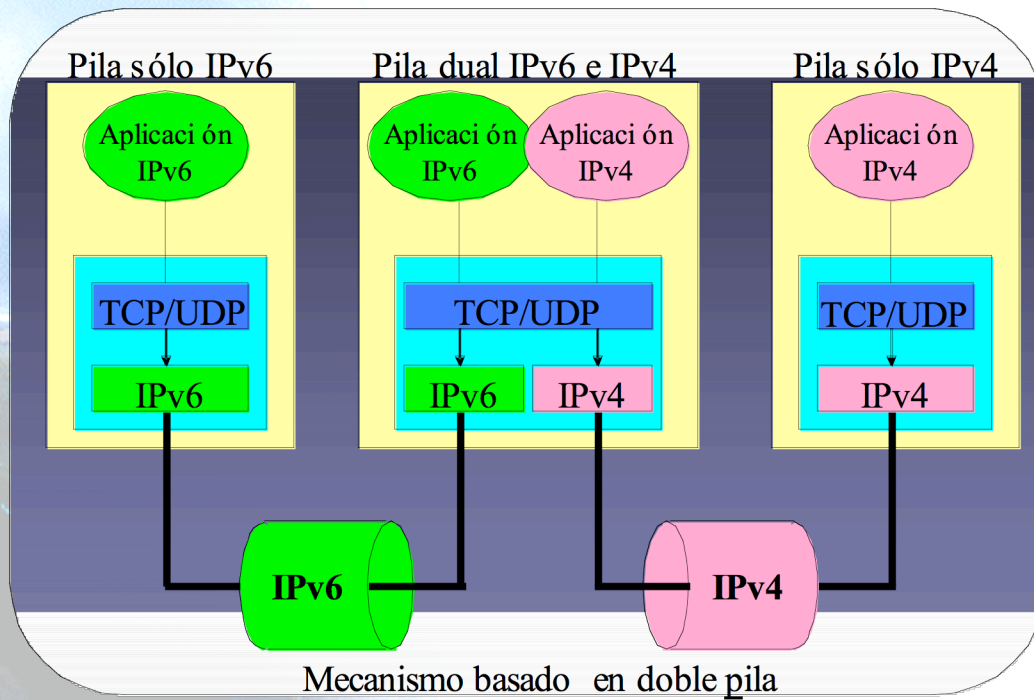
# Configuración de mecanismos de transición

# 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.
  - Traducción IPv4/IPv6, como último recurso, dado que no es perfecto.
  - Túneles, o encapsulado de IPv6 sobre IPv4 (y viceversa).
    - Son los más utilizados

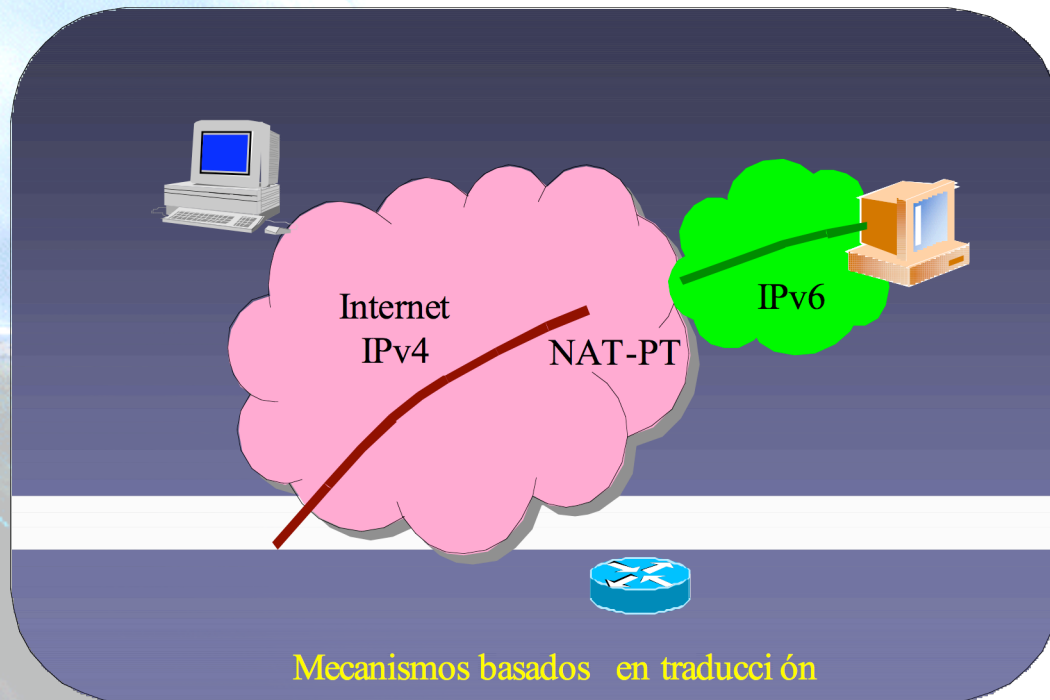


# Doble pila



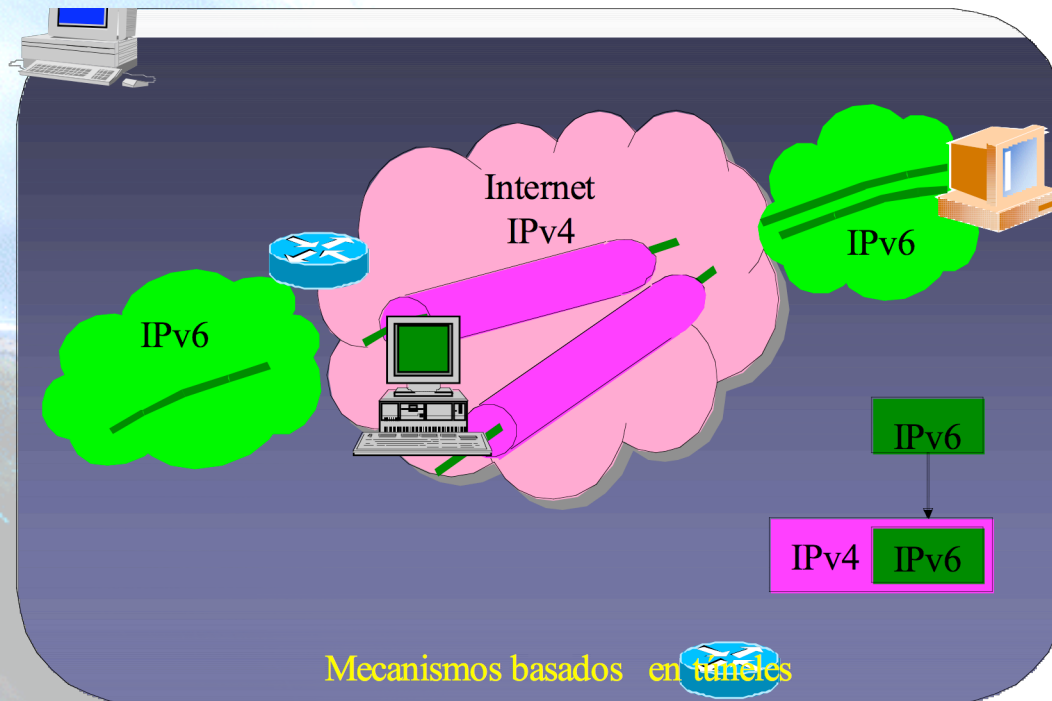
- Los nodos tienen implementadas las pilas IPv4 e IPv6
- Comunicaciones con nodos solo IPv6 ==> Pila IPv6, asumiendo soporte IPv6 en la red
- Comunicaciones con nodos solo IPv4 ==> Pila IPv4

# 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.

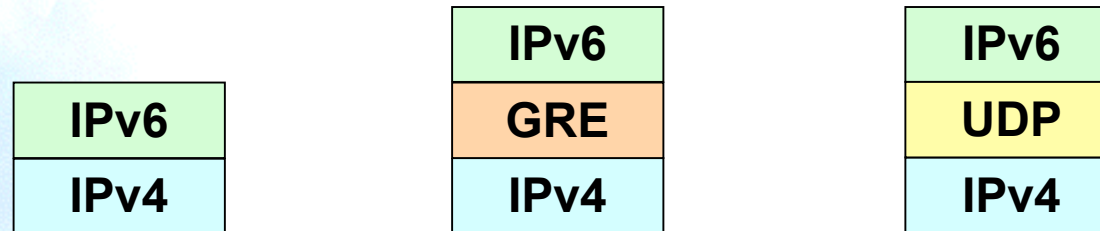
# Túneles IPv6 sobre IPv4 (1)



- 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 sobre 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

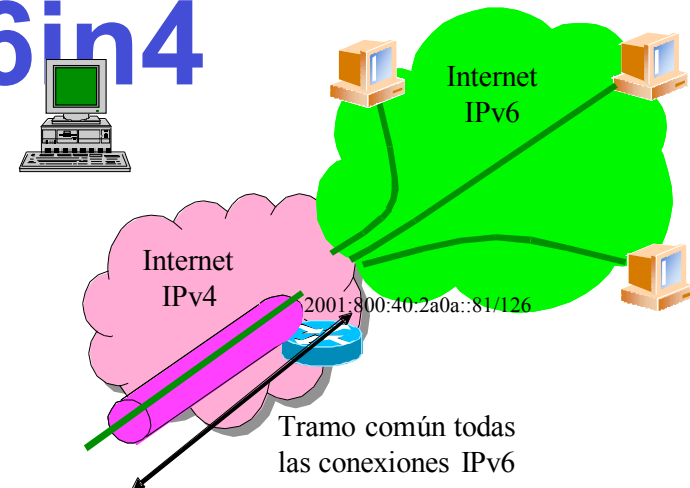


# Túneles IPv6 sobre IPv4 (3)

- 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]

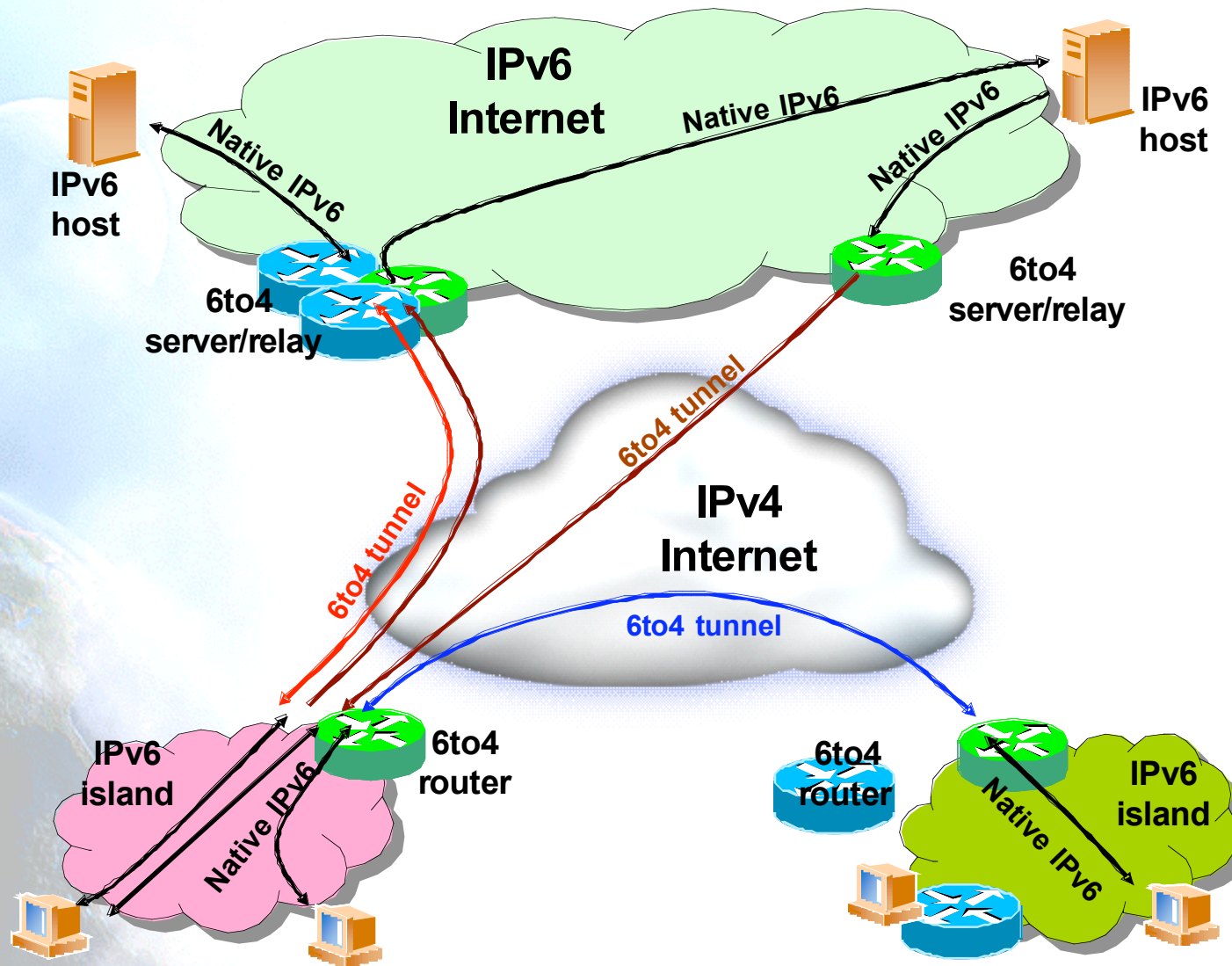


2001:800:40:2a0a::82/126

# Tunnel Broker

- 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 esta basado en un interfaz web sino en un aplicación cliente que se instala el cliente y se conecta con un servidor, aunque el concepto es el mismo.

# Túneles 6to4 (1)

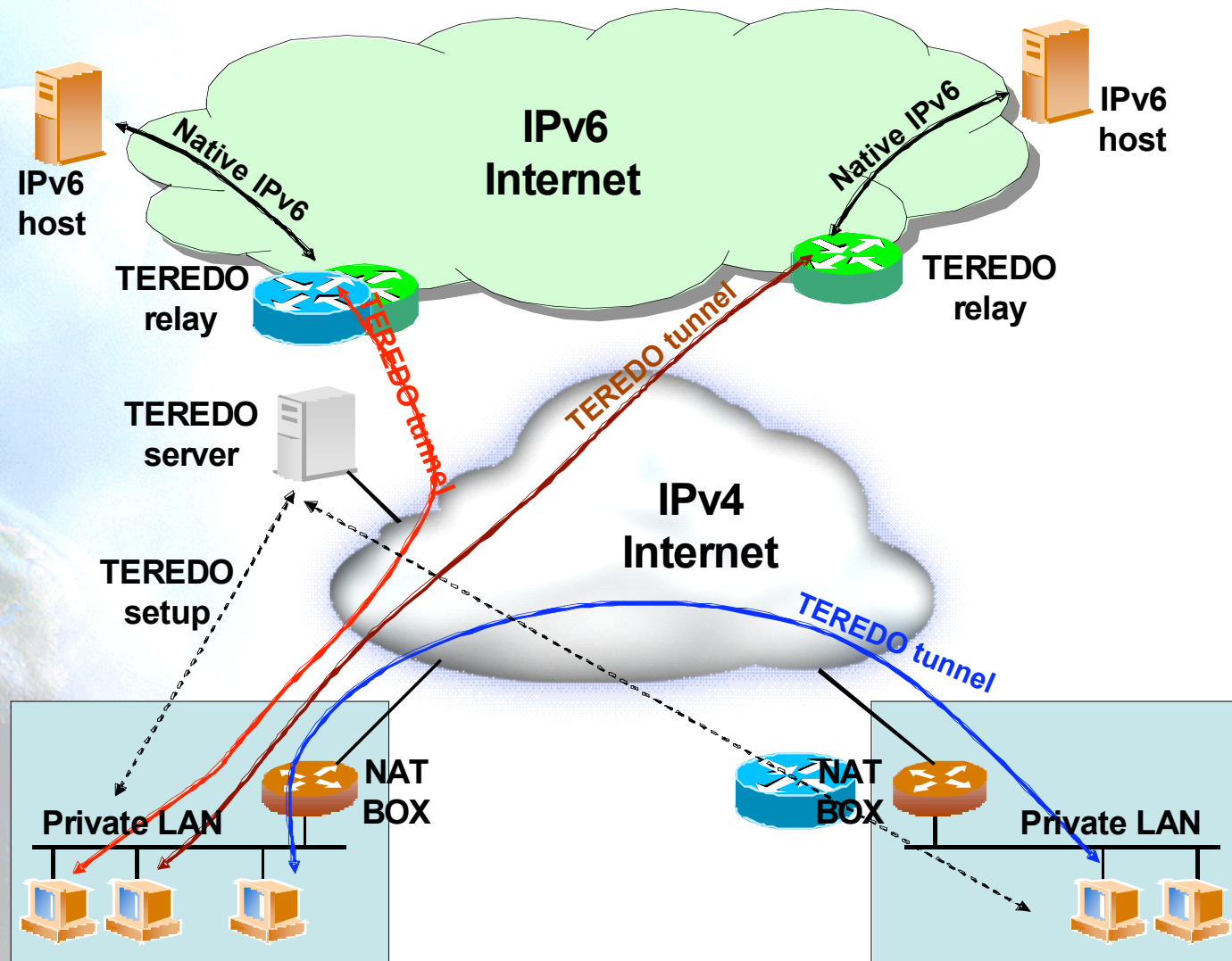




# Túneles 6to4 (2)

- 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.

# Teredo (1)



# Teredo (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
  - Simetric
- Intervienen diversos agentes:
  - Teredo Server
  - Teredo Relay
  - Teredo Client
- El cliente configura un Teredo Server que le proporciona una dirección IPv6 del rango 3FFE:831F::/32 basada en la dirección IPv4 publica 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

# 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 XP/2003 (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 XP/2003 (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 XP/2003 (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
    - 22 -> 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 (Windows 2003)

- La configuración de un 6to4 Relay es muy fácil en el caso de Windows 2003
  - 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 direccion 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 ping6 y tracerout6 (ping y tracer en Windows XP/2003) para comprobar conectividad



# E6: Configurar un Cliente Teredo (Windows XP/2003)

- Existen otras implementaciones de Teredo para otros Sistemas Operativos:
  - Linux: <http://www.simpahalempin.com/dev/miredo/>
  - FreeBSD: <http://www-rp.lip6.fr/teredo/>
- Windows XP/2003 incorpora una implementación del Teredo Client
- 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 Teredo Client
  - Comprobar haciendo ping a la dirección de otro Teredo Client de otro alumno




# 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)
  - Windows XP/2003

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

- Implementar un Proxy IPv4/IPv6 en Windows XP/2003
  - 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
  - netsh int port set v4tov6 8220  
2001:200:0:8002:203:47ff:fea5:3085 80 Direccion\_IPv4\_local
  - Comprobar con http://direccion\_IPv4\_local
- Implementar un Proxy IPv6/IPv4 en Windows XP/2003
  - Puerto 8330 IPv6 redirigirlo al puerto 80 de www.kame.net (203.178.141.194)
  - netsh int port set v6tov4 8330 203.178.141.194 80  
Direccion\_IPv6\_local



# Parte 4

## Ejemplos de aplicaciones varias

# Aplicaciones IPv6 (1)

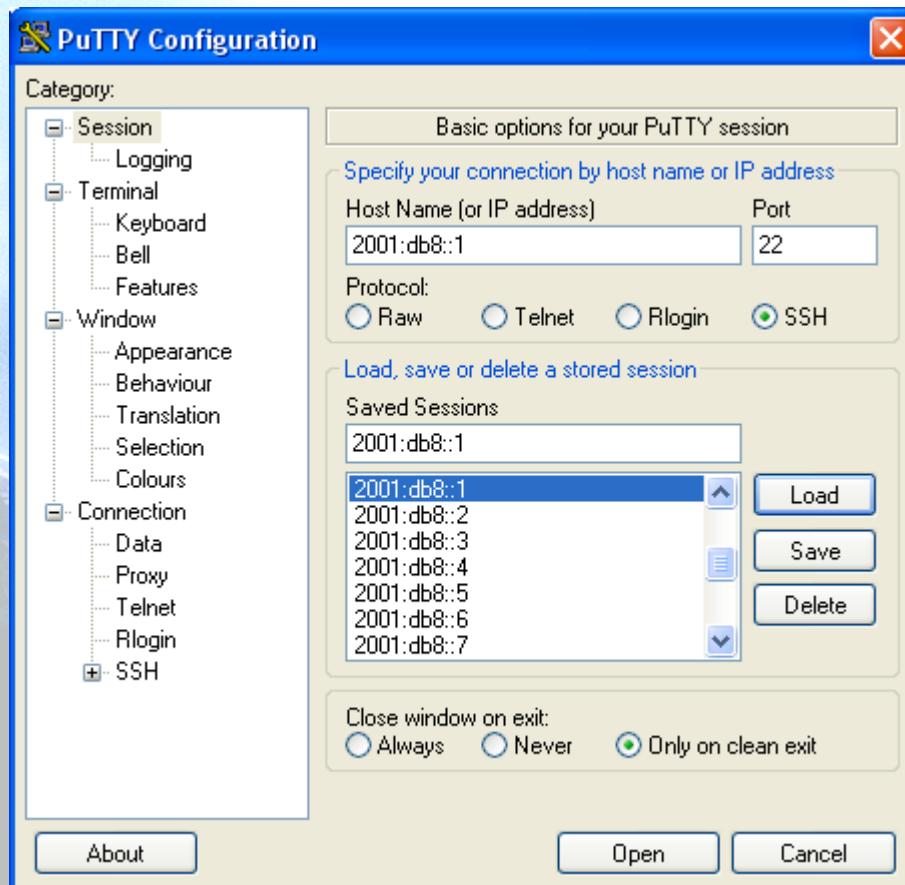
- Modelo Cliente-Servidor implica que se pueden tener aplicaciones clientes y/o servidores que sean:
  - Sólo IPv4
  - Sólo IPv6
  - IPv4 + IPv6
- Esto proporciona un conjunto de combinaciones que deben tenerse en cuenta conjuntamente con la existencia o no de conectividad IPv4 y/o IPv6



# 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

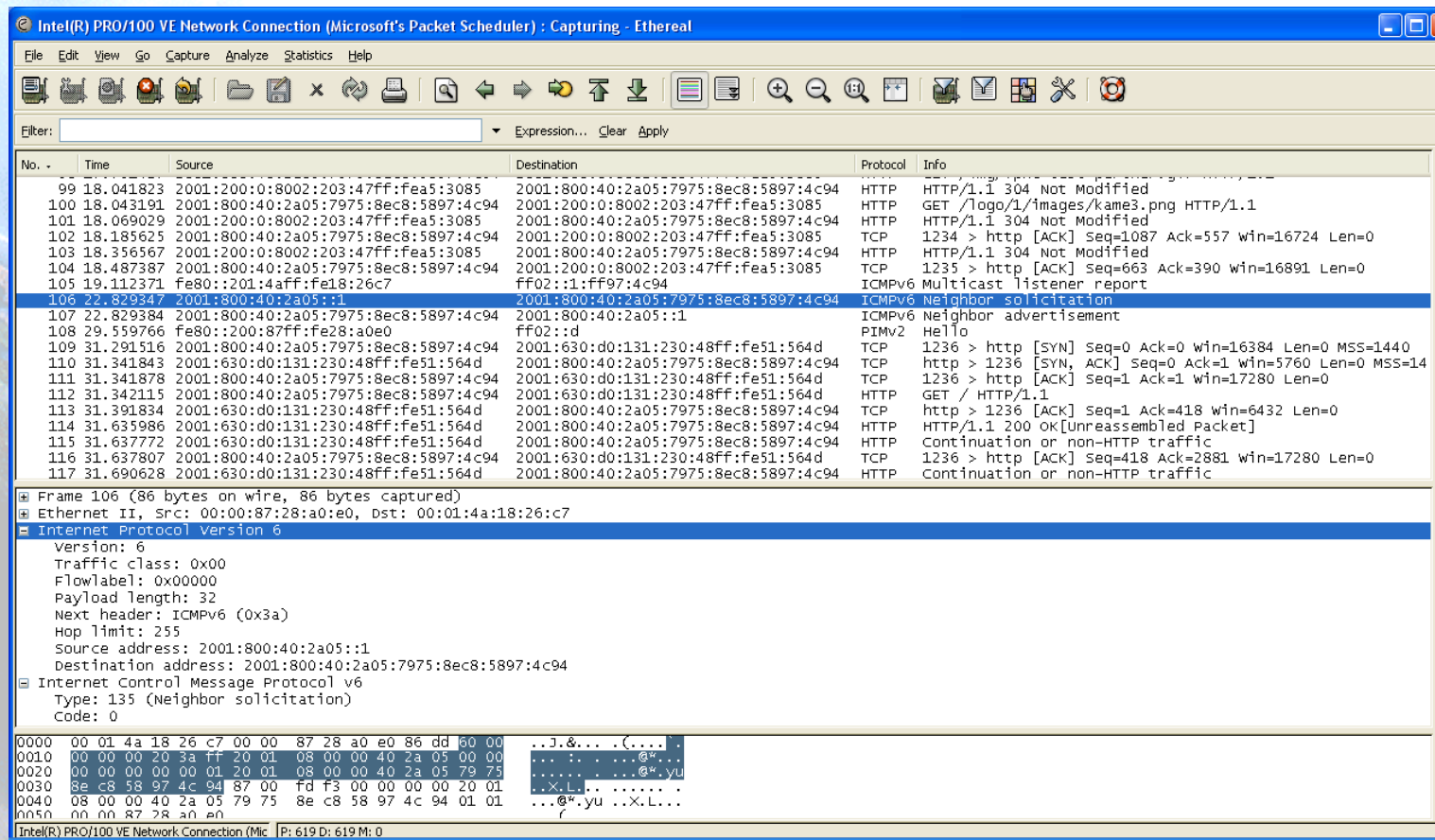
# Aplicaciones IPv6 (3)



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

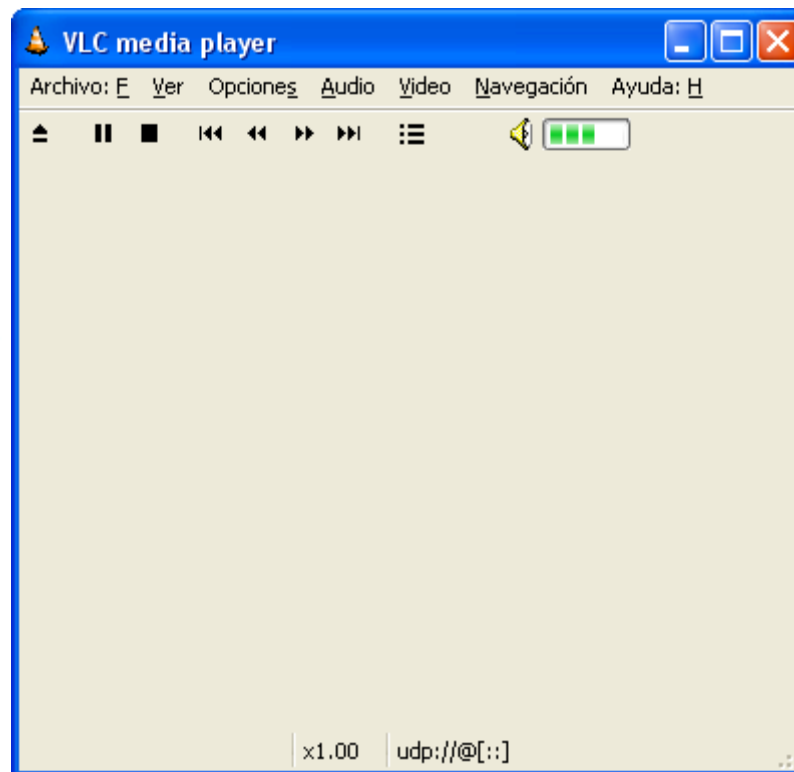
# Aplicaciones IPv6 (4)

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



# Aplicaciones IPv6 (5)

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



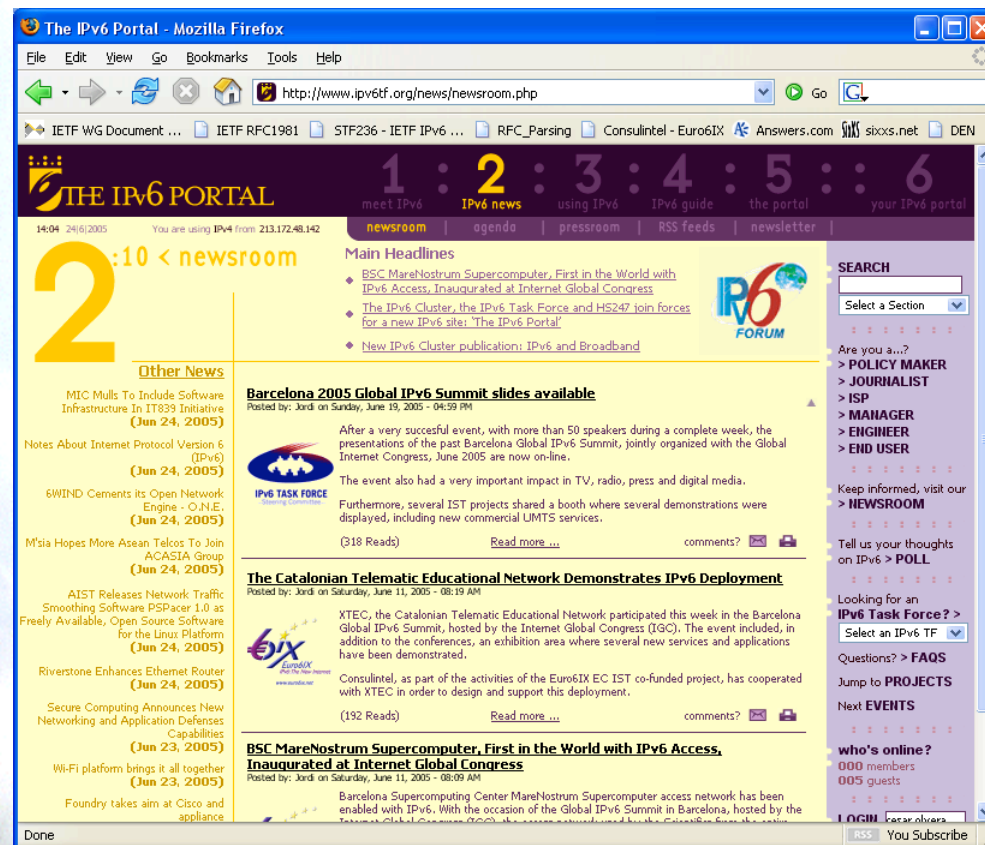


# Aplicaciones IPv6 (6)

- **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 máquina 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**
- Clientes: Los más usuales: Firefox, IE, Konqueror, Opera, Safari
- Servidores: Apache 2 soporta IPv6



# Aplicaciones IPv6 (8)

- **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)
  - Etherreal
  - VLC
  - VNC

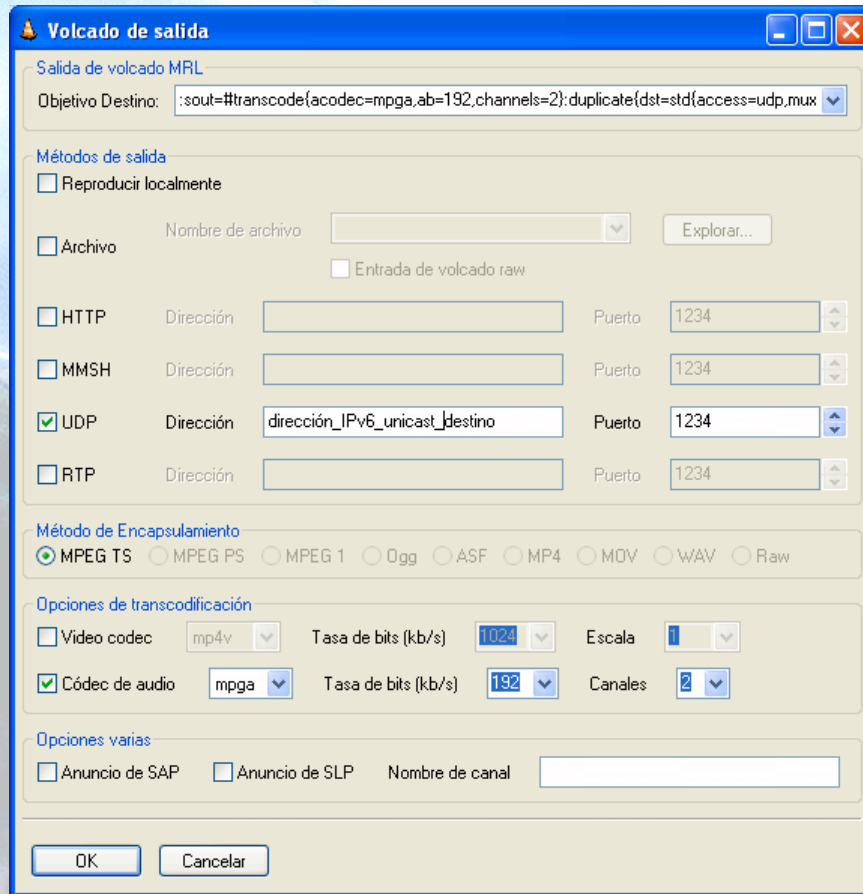
# Aplicaciones IPv6: Ejercicio 3

- Acceder a los distintos servicios mientras se capturan paquetes con Ethereal (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)

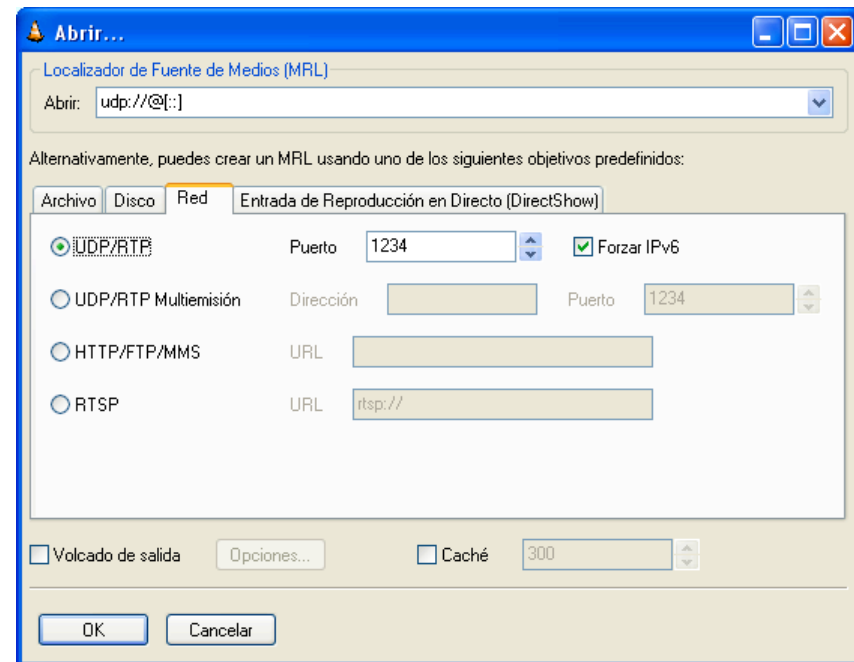


# Aplicaciones IPv6: Ejercicio 3

- VLC con Unicast



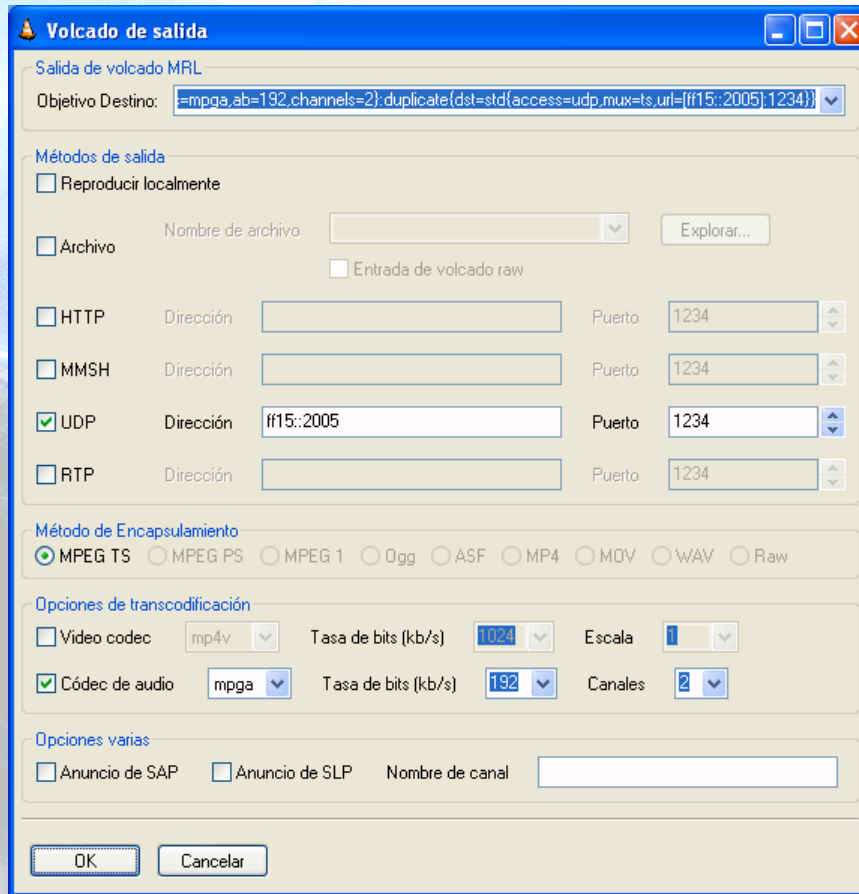
Servidor



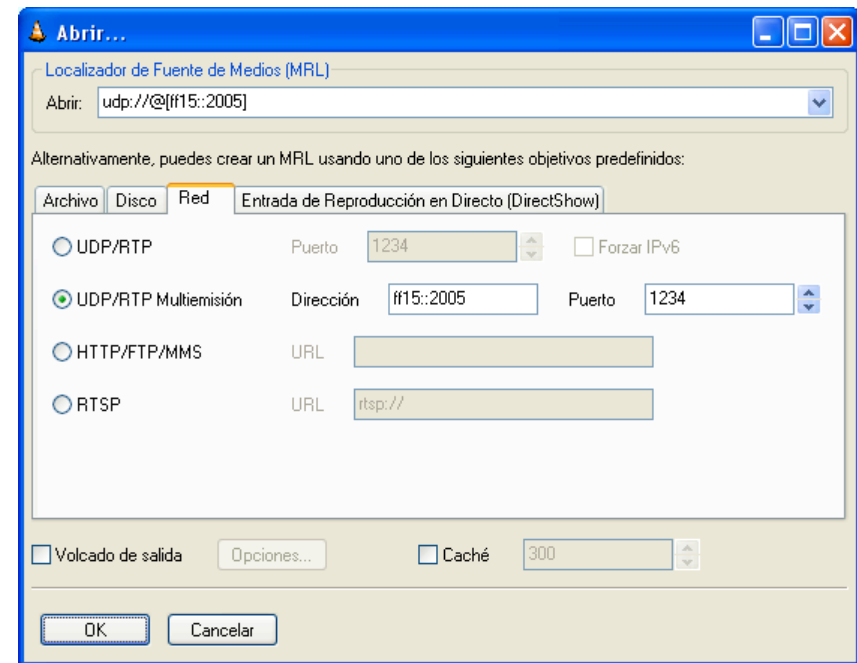
Cliente

# Aplicaciones IPv6: Ejercicio 4

- VLC con Multicast

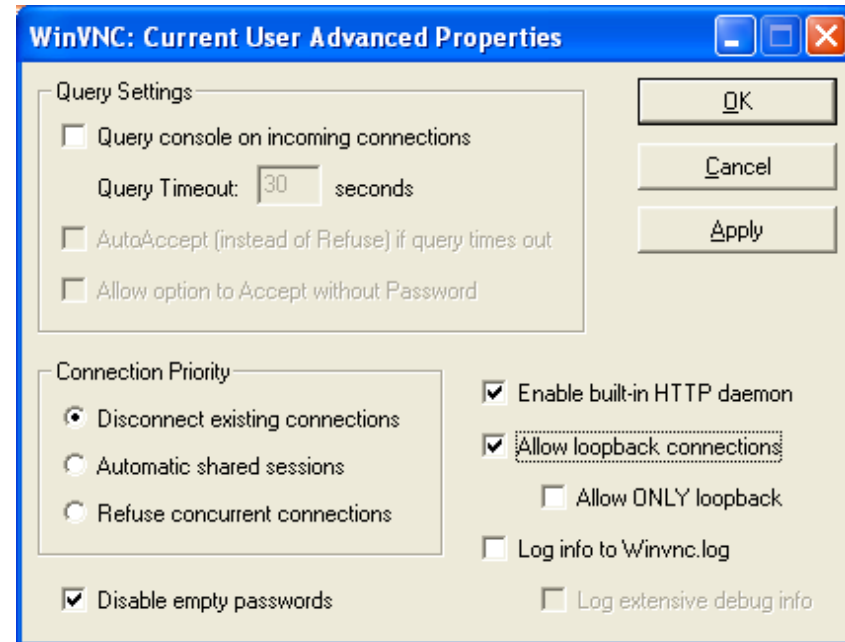
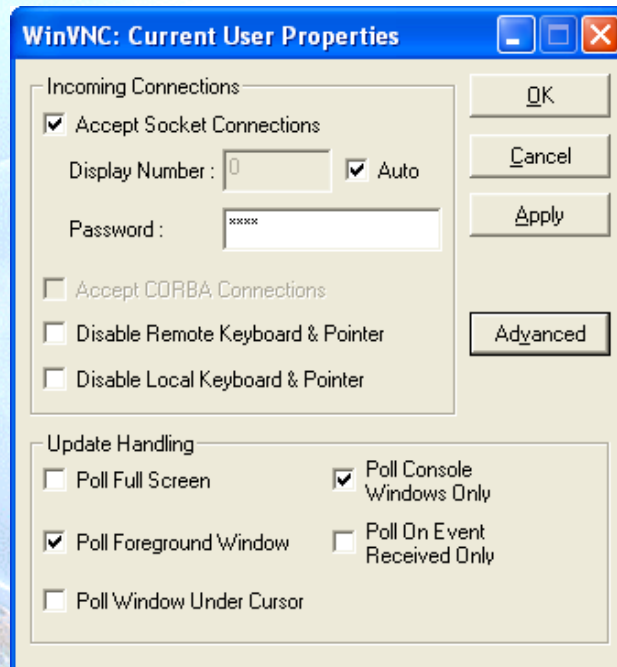


Servidor



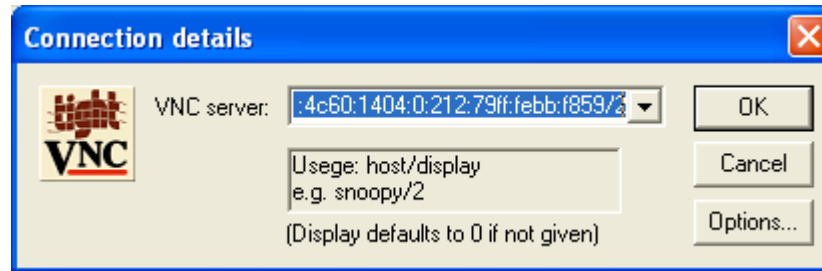
Cliente

# 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/winxppro/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.mspx](http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/sag_ip_v6_add_utils.mspx)
  - <http://www.microsoft.com/technet/community/columns/cableguy/cg0902.mspx>.