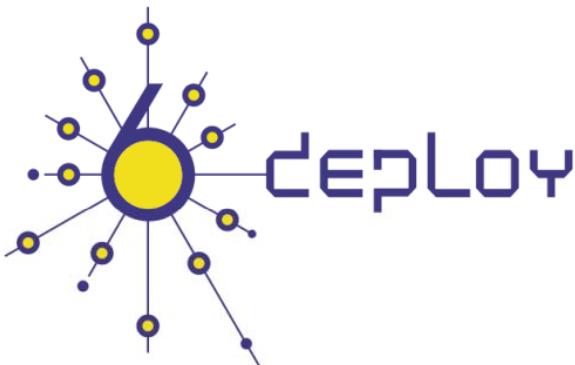


# IPv6 Startup



**Miguel A. Díaz ([miguelangel.diaz@consulintel.es](mailto:miguelangel.diaz@consulintel.es))**  
**César Olvera ([cesar.olvera@consulintel.es](mailto:cesar.olvera@consulintel.es))**  
**Jordi Palet ([jordi.palet@consulintel.es](mailto:jordi.palet@consulintel.es))**  
**Alvaro Vives ([alvaro.vives@consulintel.es](mailto:alvaro.vives@consulintel.es))**



# Agenda

1. IPv6 setup in several Platforms (Windows 2K/XP/2003/Vista, Linux, BSD)
2. Basic Configuration, Stateless/Stateful Autoconfiguration, Privacy, Static Routes
3. Transition Mechanisms Configuration
4. Examples of Applications
5. IPv6 DNS
6. IPv6 and PPP
7. Firewall IPv6
8. Enable IPv6 on Cisco Routers and IPv6 ACLs
9. SNMP over IPv6



# Part 1

## IPv6 Setup in several Platforms (Windows 2K/XP/2003/Vista, Linux, BSD)



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

# IPv6 Setup: W2K (1)

- Non-production stack available (originally developed by Microsoft Research)
- Download the “Microsoft IPv6 Technology Preview for Windows 2000”:
  - Available at <http://www.ipv6tf.org/using/connectivity/guides.php?cid=1>
  - Note that Windows 2000 IPv6 isn't supported anymore by Microsoft
- Install Procedure:
  - Log on to the Windows 2000 with local administrator privileges
  - Extract IPv6 Technology Preview files, for example in C:\IPv6Kit
  - Follow the procedure in SPn & IE6 fixed.txt in order to change /setup/hotfix.ini file
  - Run the Setup.exe or hotfix.exe
  - From the Windows 2000 desktop, click Start, point to Settings, and then click Network and Dial-up Connections. As an alternative, you can right-click My Network Places, and then click Properties
  - Right-click the Ethernet-based connection to which you want to add the IPv6 protocol, and then click Properties (typically, this connection is named Local Area Connection)
  - Click Install)
  - In the Select Network Component Type dialog box, click Protocol, and then click Add
  - In the Select Network Protocol dialog box, click Microsoft IPv6 Protocol and then click OK
  - Click Close to close the Local Area Connection Properties dialog box
- In a DOS Prompt:
  - **ipv6 if** to check if IPv6 has been installed



# IPv6 Setup: W2K (2)

- Uninstall Procedure:
  - Log on to the Windows 2000 with local administrator privileges
  - From the Windows 2000 desktop, click Start, point to Settings, and then click Network and Dial-up Connections. As an alternative, you can right-click My Network Places, and then click Properties
  - Right-click the connection to which you want to remove the Microsoft Research IPv6 protocol, and then click Properties (typically, this connection is named Local Area Connection)
  - Click MSR IPv6 Protocol and then click Uninstall
  - In the Uninstall MSR IPv6 Protocol dialog box, click Yes
  - In the Local Network dialog box, click Yes to restart your computer
- In a DOS Prompt:
  - **ipv6 if** to check if IPv6 was uninstalled



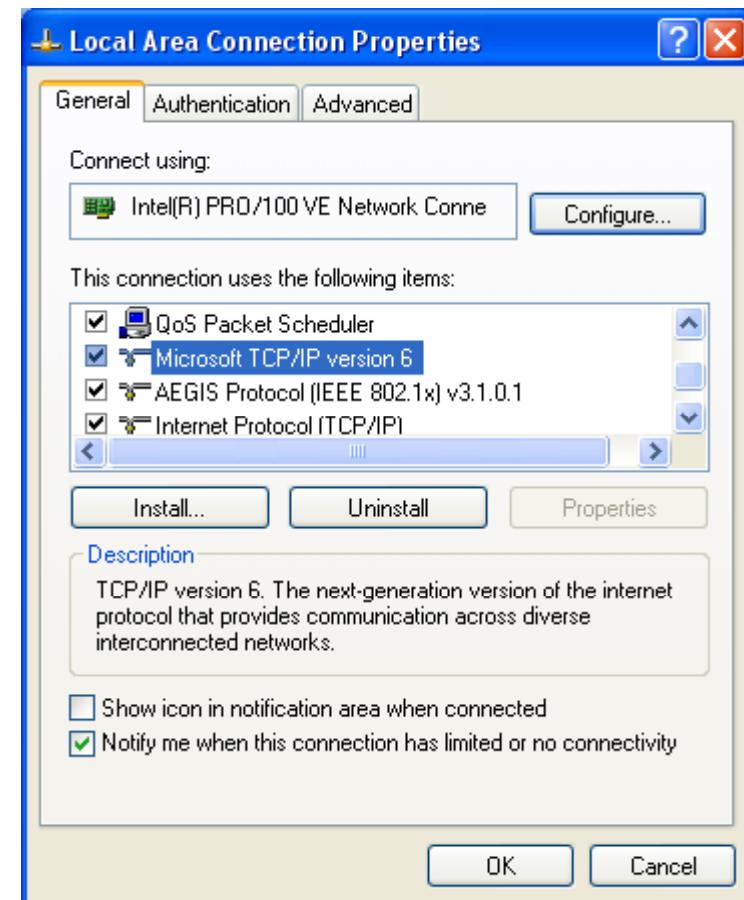
# IPv6 Setup: XP/2003 (1)

- In a DOS Prompt:
  - **ipv6 install** to install IPv6 as Network Protocol
  - **ipconfig** or **ipv6 if** to check if IPv6 was installed



# IPv6 Setup: XP/2003 (2)

- Another option to check if IPv6 was installed
  - Network Connections > Local Area Connection > Properties
- Also it is possible to install/uninstall IPv6 from here



# IPv6 Setup: XP/2003 (3)

In a Command Prompt:

- **ipv6 uninstall** to delete IPv6 as Network Protocol
- **ipconfig** or **ipv6 if** to check if IPv6 was uninstalled



# IPv6 Setup: Vista (1)

- Nothing to do!!!
  - IPv6 is installed by default ☺
  - The configuration is based on GUI ☺
- Other new features in Vista regarding IPv6
  - Full IPsec support
  - MLDv2
  - Link-Local Multicast Name Resolution (LLMNR)
    - It doesn't need DNS server. IPv6 nodes in a segment ask the name to a multicast IPv6 address. It's similar to the NetBIOS working.
  - IPv6 address in URLs support
  - IPv6 over PPP
  - DHCPv6, not only in the client but also in the server
  - Random IDs by default for the IPv6 address
    - Similar to the Privacy Extension Address but allows to be included in DNS
    - It prevents the user be tracked by using the EUI-64 part of the IPv6 address
  - Teredo supports symmetric NATs
    - It is enabled by default but no active. It becomes active automatically if any application needs IPv6 support and it is not natively available in the network



# IPv6 Setup: Vista (2)

- Un-installation
  - It cannot be un-installed because the IPv6 stack is completely integrated in the operating system as the IPv4 one
- It can be disabled for one specific network interface
  - Through the GUI “Network Connections” and disabling the IPv6 stack
- Different IPv6 components can be customized through the registry:
  - The following record (type DWORD) needs to be created:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\tcpip6\Parameters\DisabledComponents
  - The DisabledComponents registry value is a bit mask that controls the following series of flags, starting with the low order bit (Bit 0 = enabled):
    1. Bit 0 → Set to 1 to disable all IPv6 tunnel interfaces, including ISATAP, 6to4, and Teredo tunnels. Default value is 0
    2. Bit 1 → Set to 1 to disable all 6to4-based interfaces. Default value is 0
    3. Bit 2 → Set to 1 to disable all ISATAP-based interfaces. Default value is 0
    4. Bit 3 → Set to 1 to disable all Teredo-based interfaces. Default value is 0
    5. Bit 4 → Set to 1 to disable IPv6 over all non-tunnel interfaces, including LAN interfaces and Point-to-Point Protocol (PPP)-based interfaces. Default value is 0
    6. Bit 5 → Set to 1 to modify the default prefix policy table to prefer IPv4 to IPv6 when attempting connections. Default value is 0
  - Some values of DisabledComponents for disabling some components:

• 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



# IPv6 Setup: Linux (1)

- To check if IPv6 is installed:

```
#test -f /proc/net/if_inet6 && echo "Current Kernel supports IPv6"
```

- Module Installation:

```
#modprobe ipv6
```

- Module check:

```
#lsmod |grep -w 'ipv6' && echo "IPv6 module loaded"
```

- Automatic Load/Unload of Module  
(/etc/modules.conf o /etc/conf.modules ):

```
alias net-pf-10 ipv6 #enables load on demand
```

```
alias net-pf-10 off #disables load on demand
```



# IPv6 Setup: Linux (2)

# ifconfig to check

```
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)
```



# IPv6 Setup: Linux (3)

## Persistent Configuration

- Red Hat (from 7.1) and similar “distros”:

Add in /etc/sysconfig/network:

NETWORKING\_IPV6=yes

Network Restart:

```
# service network restart
```

Or

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

- SUSE:

Add in /etc/sysconfig/network/ifcfg-<Interface-Name>:

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

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



# IPv6 Setup: Linux (4)

## Persistent Configuration

- Debian/Ubuntu:

Once the IPv6 module is loaded, then edit /etc/network/interfaces, for example:

```
iface eth0 inet6 static
    pre-up modprobe ipv6
    address 3ffe:ffff:1234:5::1:1
    # unable autoconfiguration:
    # up echo 0 > /proc/sys/net/ipv6/conf/all/autoconf
    netmask 64
    # router is autoconfigured and doesn't have static address
    # it finds it because of
    # (/proc/sys/net/ipv6/conf/all/accept_ra).
    # if not, gateway must be configured:
    # gateway 3ffe:ffff:1234:5::1
```

- Reboot or:

```
# ifup --force eth0
```



# IPv6 Setup: Linux (5)

- Tools:

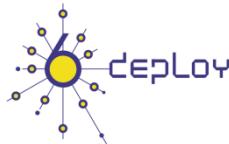
1. net-tools package

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

2. iproute package

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

3. iputils package contains ping6,  
traceroute6 and tracepath6



# IPv6 Setup: BSD (1)

- To install the Stack (Versions 4.5+)
- Good IPv6 support



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 16

## Part 2

# Basic Configuration Stateless/Stateful Autoconfiguration, Privacy, Static Routes



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 17

# Basic Configuration: W2K (1)

- Basic Commands in W2K
- Useful to obtain information about the status and to configure interfaces, addresses, caches, routes, and so on
- Two groups of commands:
  - **Net.exe**
    - Can be used to stop and start the IPv6 protocol
    - Restarting the IPv6 protocol causes it to reinitialize as if the computer were rebooting, which might change interface numbers
  - **ipv6.exe** (covers up to Windows XP SP2)
    - All Microsoft IPv6 protocol configuration is done with the ipv6.exe tool
    - Some changes are not persistent (values lost with each reboot). It is possible to execute a configuration in a .cmd script in each boot



# Basic Configuration: W2K (2)

- “Net” Commands
  - Net.exe has many subcommands, each with its own set of arguments and options. Only the following commands are directly relevant to IPv6:
    - net stop tcpip6: Stops the IPv6 protocol and unloads it from memory. This command fails if there are any open IPv6 sockets
    - net start tcpip6: Starts the IPv6 protocol if it was stopped. If a new Tcpip6.sys driver file is present in the %systemroot%\System32\Drivers directory, it is loaded
- “ipv6” Commands
  - ipv6.exe has many subcommands, each with its own set of arguments and options:
    - ipv6 if [if#]
    - ipv6 ifc if# [forwards] [advertisers] [-forwards] [-advertisers] [mtu #bytes] [site site-identifier]
    - ipv6 ifd if#
    - ipv6 nc [if# [address]]
    - ipv6 ncf [if# [address]]
    - ipv6 rc [if# address]
    - ipv6 rcf [if# [address]]
    - ipv6 bc
    - ipv6 adu if#/address [lifetime VL[/PL]] [anycast] [unicast]
    - ipv6 spt
    - ipv6 spu prefix if# [lifetime L]
    - ipv6 rt
    - ipv6 rtu prefix if#[/nexthop] [lifetime L] [preference P] [publish] [age] [spl site-prefix-length]
- Further information at:  
<http://msdn.microsoft.com/downloads/sdksp/Platform/tip6/start.asp>



# Basic Configuration: W2K (3)

- Ping in W2K
  - **ping6** destination-address
- Traceroute in W2K
  - **tracert6** destination-address



# Basic Configuration: W2K (4)

- Adding an Address:
  - **ipv6 add IfIndex/Address [life ValidLifetime[/PrefLifetime]] [anycast] [unicast]**
  - Example: ipv6 add 2/2001:db8::1
- Deleting an Address:
  - **ipv6 add IfIndex/Address [life ValidLifetime[/PrefLifetime]] [anycast] [unicast]**
  - Example: ipv6 add 2/2001:db8::1 life 0
- Check the configuration using
  - **ipv6 if 2**



# Basic Configuration: W2K (5)

- Adding a Static Route:
  - **ipv6 rtu Prefix IfIndex[/Address] [lifetimeValid[/Preferred]] [preference P] [publish] [age] splSitePrefixLength**
  - Example: **ipv6 rtu ::/0 2/::192.168.0.102**
    - Above, ::192.168.0.102 is the default gateway
- Showing Routes:
  - **ipv6 [-v] rt**



# Basic Configuration: W2K (6)

- Deleting a Static Route:
  - **ipv6 rtu Prefix IfIndex[/Address] [lifetimeValid[/Preferred]] [preference P] [publish] [age] splSitePrefixLength**
  - Example: `ipv6 rtu ::/0 2/::192.168.0.102 pub life 0`
    - Above, ::192.168.0.102 is the default gateway
- Check using
  - **ipv6 rt**



# Basic Configuration: W2K (7)

- Manual Tunnel
- Use **ipv6 adu** and **ipv6 rtu**
- Example:
- **ipv6 rtu ::/0 2/::200.20.20.20**
- **ipv6 adu 2/2001:db8:0a20:0011::2**
  - 200.20.20.20 is the remote endpoint address
  - 2001:db8:0a20:0011::2 is the local address
- Check using **ipv6 if 2** and **ipv6 rt**



# Basic Configuration: XP/2003 (1)

- Basic Commands in XP/2003
- Useful to obtain information about the status and to configure interfaces, addresses, caches, routes, and so on
- Two groups of commands:
  - **ipv6.exe** (covers up to Windows XP SP2)
    - Some changes are not persistent (values lost with each reboot). It is possible to execute a configuration in a script in each boot.
  - **netsh interface ipv6** (starting on Windows XP SP2 and Server 2003)
    - Option **store=active|persistent** to save changes
- Equivalences at:  
<http://www.microsoft.com/windowsserver2003/technologies/ipv6/ipv62netshtable.mspx>



# Basic Configuration: XP/2003 (2)

- “**ipv6**” Commands
  - `ipv6 [-p] [-v] if [ifindex]`
  - `ipv6 [-p] ifcr v6v4 v4src v4dst [nd] [pmld]`
  - `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`



# Basic Configuration: XP/2003 (3)

- “**netsh interface ipv6**” Commands
  - 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



# Basic Configuration: XP/2003 (4)

- “**netsh interface ipv6 add**” Commands
  - 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.
- “**netsh interface ipv6 set**” Commands
  - 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.
- “**netsh interface ipv6 show**” Commands
  - 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.



# Basic Configuration: XP/2003 (5)

- Interface Information
- **ipconfig [/all]**
- **ipv6 [-v] if [IfIndex]**
- Example: **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



# Basic Configuration: XP/2003 (6)

- Ping in 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
- ping command default to IPv6 if available



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 30

# Basic Configuration: XP/2003 (7)

- Examples of Ping in XP/2003
- ping6 www.ipv6tf.org

Pinging www.ipv6tf.org [2a01:48:1:0:2e0:81ff:fe05:4658]

from 2001:800:40:2a05:9c4d:b1cd:98d5:5a32 with 32 bytes of data:

Reply from 2a01:48:1:0:2e0:81ff:fe05:4658: bytes=32 time<1ms

Ping statistics for 2a01:48:1:0:2e0:81ff:fe05:4658:

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

Approximate round trip times in milli-seconds:

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



# Basic Configuration: XP/2003 (8)

- **Examples of Ping in XP/2003**

- **ping ::1**

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

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 (own link-local)**

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

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



# Basic Configuration: XP/2003 (9)

- Which are my neighbors?
  - **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

- The reference to specific interface is done with “%”
  - %5 is about interface 5



# Basic Configuration: XP/2003 (10)

- Examples of Ping in XP/2003
- **ping fe80::200:87ff:fe28:a0e0%5 (link-local neighbor in interface 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

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



# Basic Configuration: XP/2003 (11)

- Traceroute in 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
  - tracert command defaults to IPv6 when available
- **Example of traceroute in 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.



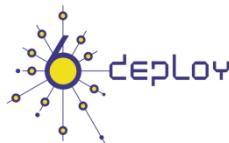
# Basic Configuration: XP/2003 (12)

- Adding an Address:
- **netsh interface ipv6 add address**  
**InterfaceNameOrIndex IPv6Address**  
**[[type=]unicast|anycast]**  
**[[validlifetime=]Minutes[infinite]]**  
**[[preferredlifetime=]Minutes[infinite]]**  
**[[store=]active|persistent]**
- Example: netsh interface ipv6 add address 5  
2001:db8::2 type=unicast validlifetime=infinite  
preferredlifetime=10m store=active
- Check the configuration using **ipv6 if 5**



# Basic Configuration: XP/2003 (13)

- Modifying the options of an already configured address:
- **netsh interface ipv6 set address**  
[interface=<string> [address=<IPv6 address>  
[[type=unicast|anycast]  
[[validlifetime=<integer>|infinite]  
[[preferredlifetime=<integer>|infinite]  
[[store=active|persistent]
- Example: netsh interface ipv6 set address 5  
2001:db8::2 preferredlifetime=infinite
- Check the configuration using **ipv6 if 5**



# Basic Configuration: XP/2003 (14)

- Deleting an Address:
- **netsh interface ipv6 delete address**  
[interface=<string> [address=<IPv6 address>  
[[store=]active|persistent]
- Example: netsh interface ipv6 delete address 5  
2001:db8::2 store=persistent
- To check the configuration using **ipv6 if 5**



# Basic Configuration: XP/2003 (15)

- Adding a Static Route:
- **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}]**
- Example: netsh interface ipv6 add route 2002::/16 5 fe80::200:87ff:fe28:a0e0 store=persistent
- Above, fe80::200:87ff:fe28:a0e0 is the default gateway



# Basic Configuration: XP/2003 (16)

- Showing Routes:
- **netsh interface ipv6 show routes**  
[[**level**=]{normal | verbose}]  
[[**store**=]{active | persistent}]
- Example: 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



# Basic Configuration: XP/2003 (17)

- Deleting a Static Route:
- **netsh interface ipv6 delete route**  
**[prefix=]<IPv6 address>/<integer>**  
**[interface=]<string> [[nexthop=]<IPv6**  
**address>] [[store=]active|persistent]**
- Example: netsh interface ipv6 delete route  
2002::/16 5 fe80::200:87ff:fe28:a0e0  
store=persistent
- Check using **netsh interface ipv6 show routes**



# Basic Configuration: XP/2003 (18)

- Adding a Static DNS Server:
- **netsh interface ipv6 add dns  
[[interface=]String]  
[[address=]IPv6Address]  
[[index=]Integer]**
- Example: netsh interface ipv6 add dns “Local area network” 2001:DB8:1000:1::947c 1
- The index represent the position of the DNS server just configured in the DNS servers lists



# Basic Configuration: XP/2003 (19)

- Showing DNS servers:
  - **netsh interface ipv6 show dns [[interface=]string]**
  - Example: 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
```

- Deleting a Static DNS server:
  - **netsh interface ipv6 delete dns [interface=<string> [[address=<IPv6 address>|all]]**
  - Example: netsh interface ipv6 delete dns “Local area network” all
  - Check using **netsh interface ipv6 show dns**

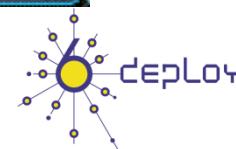
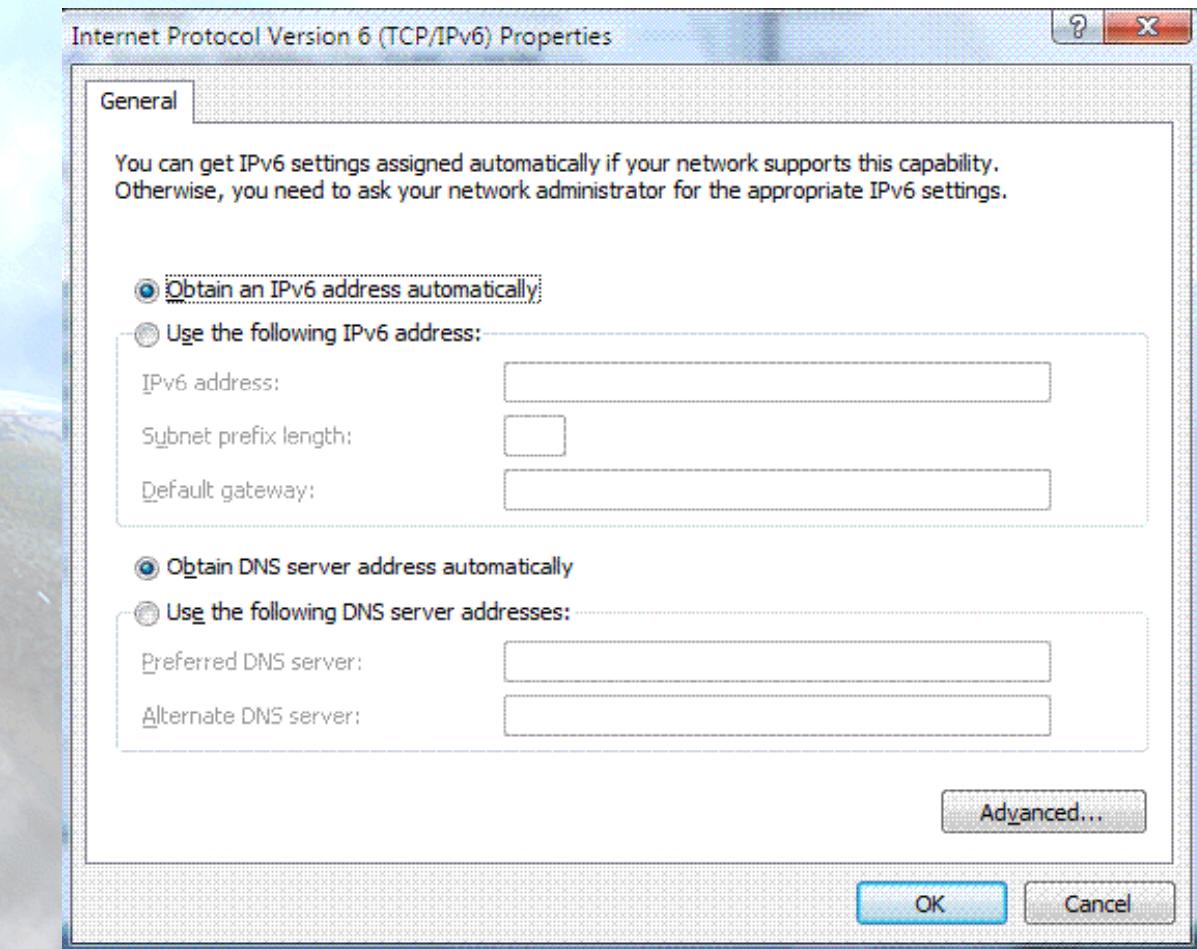
# Basic Configuration: Vista (1)

- There exists two ways of configuration
  - It supports configuration based on GUI (new) ☺
  - Basic commands based on DOS
    - Same that XP/2003.
    - Valid all the before mentioned regarding XP/2003 configuration
      - **netsh interface ipv6** (like in Windows XP and Server 2003)

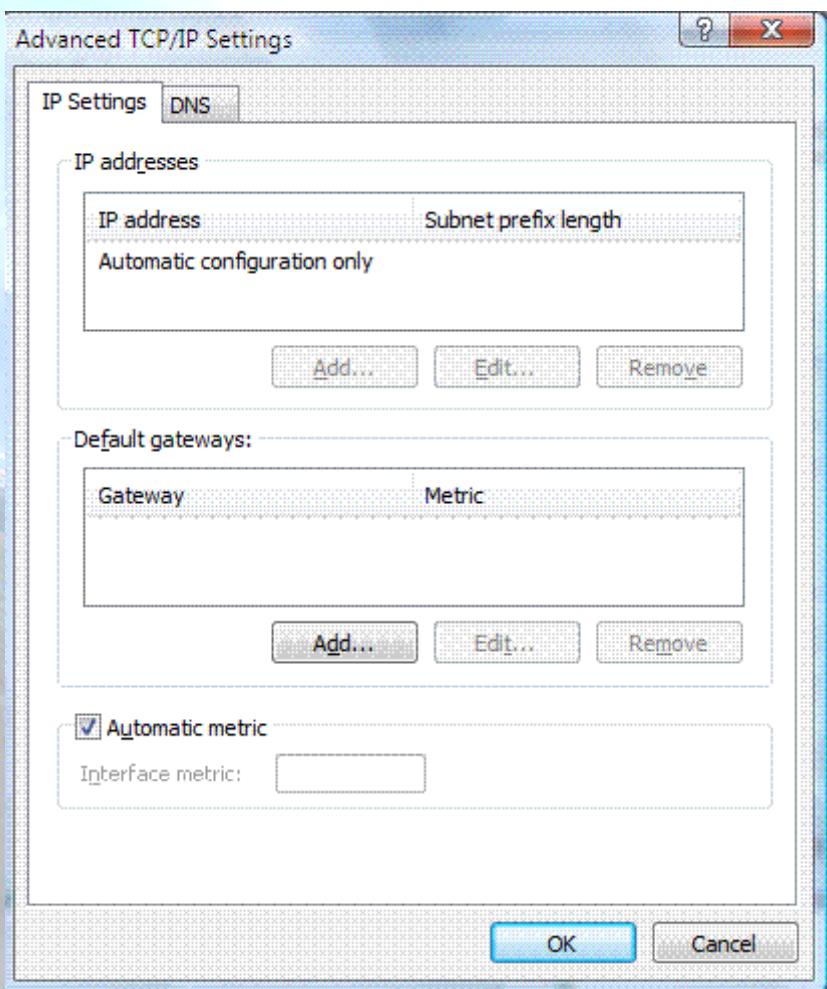


# Basic Configuration: Vista (2)

- Basic Configuration based on GUI
  - Network Connections → Connection → Properties → TCP/IPv6
  - Address configuration either automatic or manual
  - Configuration of DNS server



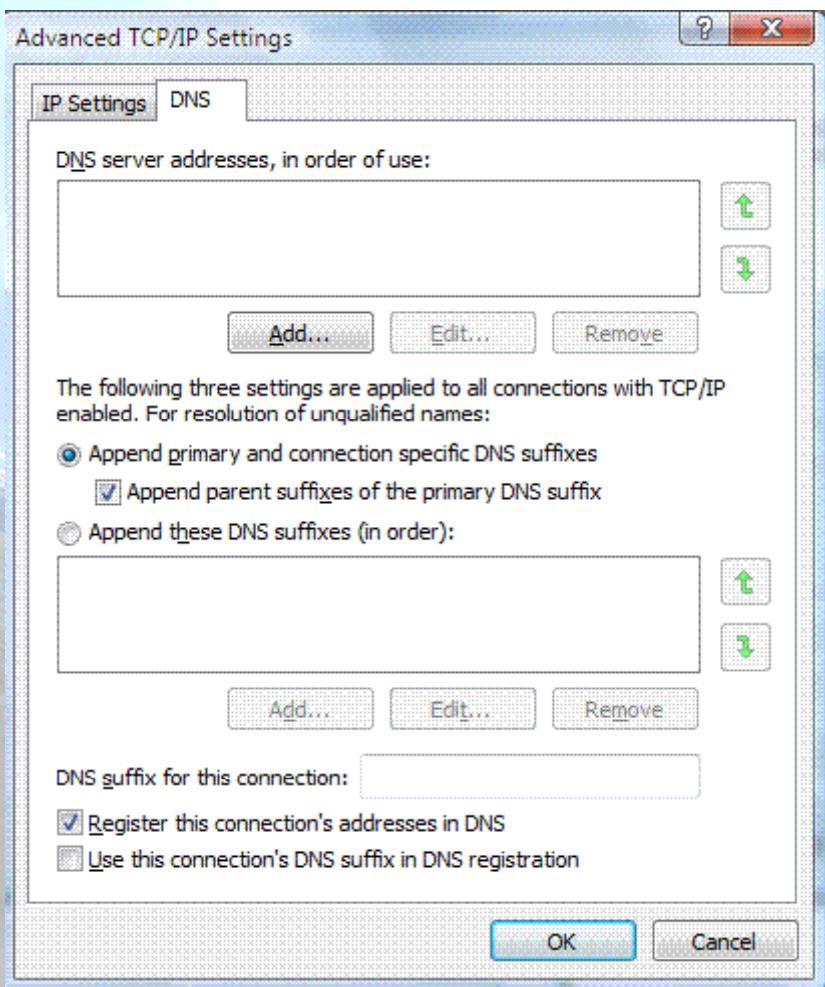
# Basic Configuration: Vista (3)



- Advanced Configuration based on GUI
  - **Network Connections** → **Connection** → **Properties** → **TCP/IPv6** → **Advanced**
  - Manual Configuration of multiple IPv6 addresses
  - Default IPv6 gateway configuration
  - Metric Configuration for routes



# Basic Configuration: Vista (4)



- Advanced Configuration based on GUI
  - Network Connections → Connection → Properties → TCP/IPv6 → DNS
  - DNS IPv6 servers manual configuration



# Basic Configuration: Linux (1)

- **Basic Commands (1)**
  - ifconfig
  - ping6 <hostcondirIPv6>|<dirIPv6>|[-I <interface>]  
<link-local-ipv6address>
  - traceroute6 <hostcondirIPv6>|<dirIPv6>
  - tracepath6 <hostcondirIPv6>|<dirIPv6>
  - tcpdump



# Basic Configuration: 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
```

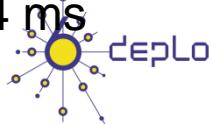
```
64 bytes from fe80::2e0:81ff:fe05:4657: icmp_seq=3 ttl=64 time=0.048 ms
```

```
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
```



# Basic Configuration: Linux (3)

- **Basic Commands (2)**
- Adding an Address:

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

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

- Deleting an Address:

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

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



# Basic Configuration: Linux (4)

- **Static Routes**
- **Showing Routes:**

```
# /sbin/ip -6 route show [dev <device>]  
# /sbin/route -A inet6
```

- **Adding a Default Route via a Gateway:**

```
# /sbin/ip -6 route add <ipv6network>/<prefixlength> via  
    <ipv6address> [dev <device>]  
#/sbin/route -A inet6 add <ipv6network>/<prefixlength> gw  
    <ipv6address> [dev <device>]
```



# Basic Configuration: Linux (5)

- Deleting a Default Route via a Gateway:

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

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

- Adding a Route via an interface:

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

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

- Deleting a Route via an interface:

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

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



# Basic Configuration: Linux (6)

- Showing Neighbors Table

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

- Adding a Neighbor

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

- Deleting a Neighbor

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



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 53

# Basic Configuration: BSD (1)

- **Basic Commands**

- Adding an IPv6 Address

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

- Deleting an IPv6 Address

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

- Adding a Default Route:

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

- Deleting a Default Route:

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



# Basic Configuration: BSD (2)

- **Persistent Configuration:**

Edit file /etc/rc.conf:

```
ipv6_enable="YES"
```

```
ipv6_ifconfig_rl0="2001:db8:10:4::4 prefixlen 64"
```

In /etc.defaults/rc.conf you can find the different parameters to configure and the defaults values

- To make apply changes in rc.conf you must reboot



# Basic Configuration: Exercise 1

- **ping6** to link-local Address of a Neighbor
- At the same time, capture packets using **tcpdump**:  
`# tcpdump -t -n -i eth0 -s 512 -vv ip6 or proto ipv6`
- Another way to show addresses:  
`# /sbin/ip -6 addr show dev eth2`  
`# ifconfig eth0`
- Add and delete the address:

2001:db8:40:2a09:1:2:3:4 in the eth0 interface



# Basic Configuration: Exercise 2

## Linux

- Add and delete a route through a gateway
- Add and delete a route through an interface
- Show neighbors table
- Add and delete a neighbor

## BSD

- Add and delete a route through a gateway



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 57

# Stateless Autoconfiguration (1)

- RFC4862: IPv6 Stateless Address Autoconfiguration
- [STATELESS] Provides information about:
  - Network Prefix
  - Routing
- Global Addresses are built by two elements
  - Interface Identifier (64 bits based on EUI-64, and usually obtained from IEEE 48 bit MAC Address)
  - Prefix obtained from de Prefix Information Options contain in the Router Advertisements
- Easing the Configuration
  - The user does not need to configure any network parameter in order to obtain native IPv6 connectivity



# Stateless Autoconfiguration (2)

- In Windows XP/2003 hosts, it is enabled by default
- **ipconfig** o **ipv6 if** to check which is the autoconfigured address
- Example: **2001:db8:10:10:201:4aff:fe18:26c7**
  - Interface Identifier EUI-64 obtained from this MAC address: 4aff:fe18:26c7
  - Prefix provided by the router: **2001:db8:10:10**



# Stateless: Exercise 1 (1)

- Configure a Linux router to send RA packets to the network
- Get a ‘radvd’ daemon for the used Linux distribution
  - <http://www.rpmfind.net/linux/rpm2html/search.php?query=radvd&submit=Search+...>
- Install it
- Enable routing capabilities
  - echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
- Edit /etc/radvd.conf file with the following content:



# Stateless: Exercise 1 (2)

```
interface eth00
{
    AdvSendAdvert on;

    MinRtrAdvInterval 3;
    MaxRtrAdvInterval 5;

    AdvHomeAgentFlag off;

    prefix 2001:db8:40:2a30::/64
    {
        AdvOnLink off;
        AdvAutonomous on;
        AdvRouterAddr off;
    };
};

};
```



# Stateless: Exercise 1 (3)

- Launch radvd daemon
  - radvd
- Check that other computers in the network are autoconfigured thanks to our radvd daemon



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 62

# Stateful Autoconfiguration (1)

- [STATEFUL] Similar to DHCP in IPv4
- An IPv6 address is provided. This address can be different each time a node gets connected
- Provides information complementary to the stateless one
  - DNS Server (could be IPv6)
  - domain name
  - NTP server (could be IPv6)
  - SIP server (could be IPv6)
  - SIP domain name
  - Prefix delegation
  - Etc.
- DHCPv6 implementations are still not available in the most common OSs
  - An specific installation of a DHCPv6 application is needed (server and/or client)
    - <http://klub.com.pl/dhcpv6/>
    - <http://sourceforge.net/projects/dhcpv6-linux/>



# Stateful: Exercise 1 (1)

- Configuring a DHCPv6 server on Linux
  - Obtain the DHCPv6 implementation for Linux from:  
<http://klub.com.pl/dhcpv6/dibbler/dibbler-0.4.0-linux.tar.gz>
  - Untar the file
    - tar -xvzf dibbler-0.4.0-linux.tar.gz
  - Make these directories
    - /var/lib/dibbler
    - /etc/dibbler



# Stateful: Exercise 1 (2)

- Edit the content of file server.conf

```
log-level 7  
log-mode short
```

```
iface eth0 {  
    T1 1000  
    T2 2000  
    class {  
        pool 2001:db8:40:2a03::10-2001:db8:40:2a03:ffff:ffff:ffff:ffff  
    }  
    option dns-server 2001:db8:40:2a03::2, 2001:db8:40:2a04::2  
    option domain example.com, test1.example.com  
}
```

- The given addresses will be in the prefix 2001:db8:40:2a03::/64 starting from 2001:db8:40:2a03::10
- Copy the file server.conf in the directory /etc/dibbler
- Launch dhcpcv6 server
  - dhcpcv6-server run



# Stateful: Exercise 2 (1)

- Configure DHCPv6 client in Linux
  - Get a DHCPv6 implementation for Linux from: <http://klub.com.pl/dhcpv6/dibbler/dibbler-0.4.0-linux.tar.gz>
  - Untar the file
    - tar -xvzf dibbler-0.4.0-linux.tar.gz
  - Create the directories
    - /var/lib/dibbler
    - /etc/dibbler



# Stateful: Exercise 2 (2)

- Edit the content of file server.conf

```
log-mode short

iface eth0
{
    IA
    option dns-server
    option domain
}
```

- With this configuration you get

- An IPv6 address
  - DNS servers
  - Domain name

- Copy client.conf file in the directory /etc/dibbler

- Launch dhcpcv6 client

- dhcpcv6-client run

- With 'ifconfig eth0' you can check if you have got an IPv6 address

- In /etc/resolv file you can check the DNS servers obtained

- Note that you don't get routing information, so you can't make ping6

- The routing information is obtained by means of stateless autoconfiguration (RA)



# Privacy (1)

- RFC 4941: Privacy Extensions for Stateless Address Autoconfiguration in IPv6
- Extension of Stateless Autoconfiguration
- It generates a global address that changes over time
- It makes more difficult to identify when different addresses used in different transactions actually correspond to the same node



# Privacy (2)

- In Windows XP/2003 hosts, it is enabled by default
- **ipconfig o ipv6 if** to check which is the autoconfigured address
- There are two ways to disable it:
  1. **netsh interface ipv6 set privacy state=disabled store=persistent**
  2. **ipv6 [-p] gpu UseTemporaryAddresses no**
- To check the change: “disable” and “enable” the physical interface on Windows Network Connection, then **ipconfig o ipv6 if**



# Privacy (3)

- **Additional options with netsh command:**
- netsh interface ipv6 set privacy  
[[state=]enabled|disabled]  
[[maxdadattempts=<integer>]  
[[maxvalidlifetime=<integer>]  
[[maxpreferredlifetime=<integer>]  
[[regeneratetime=<integer>]  
[[maxrandomtime=<integer>]  
[[randomtime=<integer>]  
[[store=]active|persistent]



## Part 3

# Transition Mechanisms Configuration



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 71

# Configuration of Transition Mechanisms: Exercises

- E1: Setup a 6in4 tunnel between two alumni's hosts
- E2: Delete the 6in4 tunnel
- E3: Get IPv6 connectivity by means of a 6in4 tunnel by using a TB
  - See the path to different IPv6 web sites
  - See the path to the provided IPv6 address from a looking glass
- E4: Get IPv6 connectivity by means of a 6to4 tunnel
  - See the path to different IPv6 web sites
  - See the path to the provided IPv6 address from a looking glass
- E5: Setup a 6to4 relay (Windows 2003)
- E6: Setup a Teredo Client (Windows XP/2003)
- E7: Usage of IPv4/IPv6 proxies
  - 46Bouncer
  - Windows XP/2003



# E1: 6in4 Tunnel Setup (1)

1. Exercise to be made with partners (\*)
    - Alumni A ==> ADD\_IPv4\_A
    - Alumni B ==> ADD\_IPv4\_B
  2. Alumni A sets up the tunnel in his side by using the following data:
    - Local IPv6 address ==> ADD\_IPv4\_A
    - Remote IPv4 address ==> ADD\_IPv4\_B
    - IPv6 address ==> 2001:db8:20:30::12/126
    - IPv6 gateway address ==> 2001:db8:20:30::11/126
  3. Alumni B sets up the tunnel in his side by using the following data:
    - Local IPv4 address ==> ADD\_IPv4\_B
    - Remote IPv4 address ==> ADD\_IPv4\_A
    - IPv6 address ==> 2001:db8:20:30::11/126
    - IPv6 gateway address ==> 2001:db8:20:30::12/126
  4. Check IPv6 connectivity between both alumni
    - Alumni A ==> ping6 IPv6\_Address\_Alumna\_B
    - Alumni B ==> ping6 IPv6\_Address\_Alumna\_A
  5. Enable forwarding
    - Alumni A ==> enable forwarding in both tunnel and LAN interfaces
    - Alumni B ==> enable forwarding in both tunnel and LAN interfaces
- (\*) This exercise does not provide global IPv6 connectivity, just IPv6 connectivity between alumni A and alumni B



# E1: 6in4 Tunnel Setup (2)

- Scripts for setting up 6in4 tunnels
  - Windows XP/2003 (from the command line window)
    - netsh interface ipv6 add v6v4tunnel "Tunnel01" Address\_IPv4\_local Address\_IPv4\_remote
    - netsh interface ipv6 add address "Tunnel01" Address\_IPv6
    - netsh interface ipv6 add route ::/0 "Tunnel01" Address\_gateway\_IPv6 publish=yes
    - netsh interface ipv6 set interface "Tunnel01" forwarding=enable
    - netsh interface ipv6 set interface "LAN" forwarding=enable
  - Linux/UNIX (from the shell)
    - modprobe ipv6
    - ip tunnel add Tunnel01 mode sit remote Address\_IPv4\_remote local Address\_IPv4\_local ttl 255
    - ip link set Tunnel01 up
    - ip addr add Address\_IPv6/126 dev Tunnel01
    - ip route add 2000::/3 dev Tunnel01
  - FreeBSD
    - gifconfig gif0 Address\_IPv4\_local Address\_IPv4\_remote
    - ifconfig gif0 inet6 Address\_IPv6 Address\_gateway\_IPv6 prefixlen 128
    - route -n add -inet6 default Address\_gateway\_IPv6



# E1: 6in4 Tunnel Setup (3)

- Scripts for setting up 6in4 tunnels
  - FreeBSD >= 4.4
    - ifconfig gif0 create
    - ifconfig gif0 tunnel Address\_IPv4\_local Address\_IPv4\_remote
    - ifconfig gif0 inet6 Address\_IPv6 Address\_gateway\_IPv6 prefixlen 128
    - route add -inet6 default Address\_gateway\_IPv6
  - NetBSD
    - ifconfig gif0 Address\_IPv4\_local Address\_IPv4\_remote
    - ifconfig gif0 inet6 Address\_IPv6 Address\_gateway\_IPv6 prefixlen 128
    - route -n add -inet6 default Address\_gateway\_IPv6
  - OpenBSD
    - ifconfig gif0 giftunnel Address\_IPv4\_local Address\_IPv4\_remote
    - ifconfig gif0 inet6 Address\_IPv6 Address\_gateway\_IPv6 prefixlen 128
    - route -n add -inet6 default Address\_gateway\_IPv6



# E2: Deleting 6in4 tunnels (1)

- Exercise to be done by each alumni (individually)
- The alumni deletes the tunnel configured previously according to the configuration script of its Operating System
- The alumni has to check that the tunnel has been deleted by using:
  - ipconfig on Windows XP/2003
  - ifconfig on Unix/Linux/\*BSD



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 76

# E2: Deleting 6in4 Tunnels (2)

- Scripts for deleting 6in4 tunnels
  - Windows XP/2003 (from the command line window)
    - netsh interface ipv6 del route ::/0 "Tunnel01" Address\_gateway\_IPv6
    - netsh interface ipv6 del address "Tunnel01" Address\_IPv6
    - netsh interface ipv6 del int "Tunnel01"
  - Linux/UNIX (from the shell)
    - ip route del 2000::/3 dev Tunnel01
    - ip addr del Address\_IPv6/126 dev Tunnel01
    - ip link set Tunnel01 down
    - ip tunnel del Tunnel01 mode sit remote Address\_IPv4\_remote local Address\_IPv4\_local ttl 255
  - FreeBSD
    - route delete -inet6 default
    - ifconfig gif0 inet6 delete Address\_IPv6
    - ifconfig gif0 down



# E2: Deleting 6in4 Tunnels (3)

- Scripts for deleting 6in4 tunnels
  - FreeBSD >= 4.4
    - route delete -inet6 default Address\_gateway\_IPv6
    - ifconfig gif0 inet6 Address\_IPv6 prefixlen 128 delete
    - ifconfig gif0 delete
  - NetBSD
    - route delete -inet6 default
    - ifconfig gif0 inet6 delete Address\_IPv6
    - ifconfig gif0 down
  - OpenBSD
    - ifconfig gif0 inet6 delete Address\_IPv6
    - ifconfig gif0 deletetunnel
    - ifconfig gif0 down
    - route delete -inet6 default



# E3: IPv6 Connectivity via a TB

1. Choose a TB from  
<http://www.ipv6tf.org/using/connectivity/test.php>
2. Follow the steps provided by the TB
3. Check that the IPv6 connectivity is available
  - ping6, traceroute6 (ping & tracert on windows)
    - [www.kame.net](http://www.kame.net), [www.6power.org](http://www.6power.org), [www.ipv6.org](http://www.ipv6.org)
  - Browsing to the same web sites
4. Check the path to the assigned IPv6 address from an external looking glass
  - [http://www.ipv6tf.org/using/connectivity/looking\\_glass.php](http://www.ipv6tf.org/using/connectivity/looking_glass.php)
  - <http://www.ipv6.udg.mx/lg.php>
  - <http://www.v6.dren.net/lg/>



# E4: IPv6 Connectivity with 6to4 (1)

1. Choose a 6to4 relay from  
<http://www.ipv6tf.org/using/connectivity/6to4.php>
2. Follow the configuration script according to the proper Operating System
3. Check that the IPv6 connectivity is available
  - ping6, traceroute6 (ping & tracert en windows)
    - [www.kame.net](http://www.kame.net), [www.6power.org](http://www.6power.org),  
[www.ipv6.org](http://www.ipv6.org)
    - Browsing to the same web sites
4. Check the path to the assigned IPv6 address from an external looking glass
  - [http://www.ipv6tf.org/using/connectivity/looking\\_glass.php](http://www.ipv6tf.org/using/connectivity/looking_glass.php)
  - <http://www.ipv6.udg.mx/lg.php>
  - <http://www.v6.dren.net/lg/>

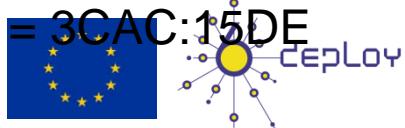


# E4: IPv6 Connectivity with 6to4 (2)

- Scripts for deleting the 6to4 tunnels
    - Windows XP/2003 (from the command line window)
      - netsh int ipv6 6to4 set relay <Address\_6TO4\_RELAY> enabled 1440
    - Linux/UNIX (from the shell)
      - ip tunnel add tun6to4 mode sit ttl 80 remote any local <Address\_public\_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 that XXYY:ZZUU** is the hexadecimal notation for Address\_public\_IPv4\_local (the public IPv4 address) according to the following:

Address\_public\_IPv4\_local = 60.172.21.22 -> 60 -> 3C  
172 -> AC  
21 -> 15  
22 -> DF

60.172.21.22 -> XXYY:ZZUU = 3CAC:15DE



# E4: IPv6 Connectivity with 6to4 (3)

- Scripts for deleting 6to4 tunnels
  - \*BSD
    - Be sure that there is at least one stf(4) interface configured in the kernel
      - In <http://www.netbsd.org/Documentation/kernel/> information about that can be found
    - ifconfig stf0 inet6 2002:XXYY:ZZUU::1 prefixlen 16 alias
    - route add -inet6 default 2002:c058:6301::1
  - **Note that XXYY:ZZUU is the hexadecimal notation for Address\_public\_IPv4\_local (the public IPv4 address) according to the following:**

Address\_public\_IPv4\_local = 60.172.21.22 -> 60 -> 3C  
172 -> AC  
21 -> 15  
22 -> DE

60.172.21.22 -> XXYY:ZZUU = 3CAC:15DE



# E5: Setting-Up a 6to4 Relay (Windows 2003)

- The 6to4 Relay configuration is very easy in case of Windows 2003
  - netsh interface ipv6 set interface interface="Local area connection" 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
  - netsh interface ipv6 set routing routing=enabled sitelocals=enabled
- Every 6to4 packet received by the "Local area connection" interface will be forwarded to the proper IPv6 destination
- In order to check the 6to4 relay configuration, a 6to4 tunnel can be configured in other host (following the instructions of previous slides) and the 6to4 server in such a new host will be the 6to4 relay just configured
  - Doing ping6 and traceroute6 (ping and tracert on Windows XP/2003) to check IPv6 connectivity



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 83

# E6: Setting-Up a Teredo Client (Windows XP/2003 w/o SP1)

- There are other Teredo implementations for other Operating Systems such as:
  - Linux: <http://www.simpahalempin.com/dev/miredo/>
  - FreeBSD: <http://www-rp.lip6.fr/teredo/>
- Windows XP/2003 presents an implementation of Teredo Client
- From a DOS window type the following:
  - set teredo client teredo.ipv6.microsoft.com. 60 34567
  - a public Teredo Server by Microsoft is used: teredo.ipv6.microsoft.com
- There exist other experimental Teredo Server/Relays (without guaranteed service)
  - teredo.ipv6.vol.cz
  - teredo.ipv6.wind.com
  - teredo.via.ecp.fr
- Check the provided IPv6 address
  - ipconfig
- Check the data of the Teredo interface
  - netsh int ipv6 show teredo
  - netsh int ipv6 show int teredo
- Global IPv6 connectivity is not provided because Microsoft does not provide any Teredo Relay
- IPv6 connectivity with other Teredo clients is available
  - Check by pinging to the IPv6 address of other alumni's Teredo Client



# E7: Use of IPv4/IPv6 Proxies (1)

- An IPv4/IPv6 proxy is not the same than a transition mechanism based on translation (NAT-PT)
- The proxy is an intermediate host working on the application level
  - It receives TCP connections over a protocol (IPv4 or IPv6) and it extracts all the data from the application level
  - Then it establishes TCP connection (IPv6 or IPv4) with the destination host and it put in the new connection the application data extracted in the previous step
- So, it allows connections between:
  - Client IPv4 ==> Proxy IPv4/IPv6 ==> Server IPv6
  - Client IPv6 ==> Proxy IPv6/IPv4 ==> Server IPv4
- There are two well-known proxies:
  - 46Bouncer (Windows y Linux)
  - Windows XP/2003



# E7: Use of IPv4/IPv6 Proxies (2)

- Implement a IPv4/IPv6 Proxy on Windows XP/2003
  - Forward the TCP/ IPv4 8220 port to the TCP/IPV6 80 port of www.kame.net (2001:200:0:8002:203:47ff:fea5:3085)
  - netsh int port set v4tov6 Port\_v4\_TCP\_local Address\_IPv6\_remote Port\_v6\_TCP\_remote Address\_IPv4\_local
  - netsh int port set v4tov6 8220 2001:200:0:8002:203:47ff:fea5:3085 80 Address\_IPv4\_local
  - Check with http://address\_IPv4\_local
- Implement a IPv6/IPv4 Proxy on Windows XP/2003
  - Forward the TCP/IPV6 8330 port to the TCP/IPV4 80 port of www.kame.net (203.178.141.194)
  - netsh int port set v6tov4 8330 203.178.141.194 80 Address\_IPv6\_local



## Part 4

# Examples of Applications



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 87

# IPv6 Applications (1)

- Client-Server model implies that it is possible to have Client/Server applications working:
  - IPv4 Only
  - IPv6 Only
  - IPv4 + IPv6
- Thus provides a set of combinations that is needed to consider jointly with the availability or unavailability of IPv4/IPv6 connectivity



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

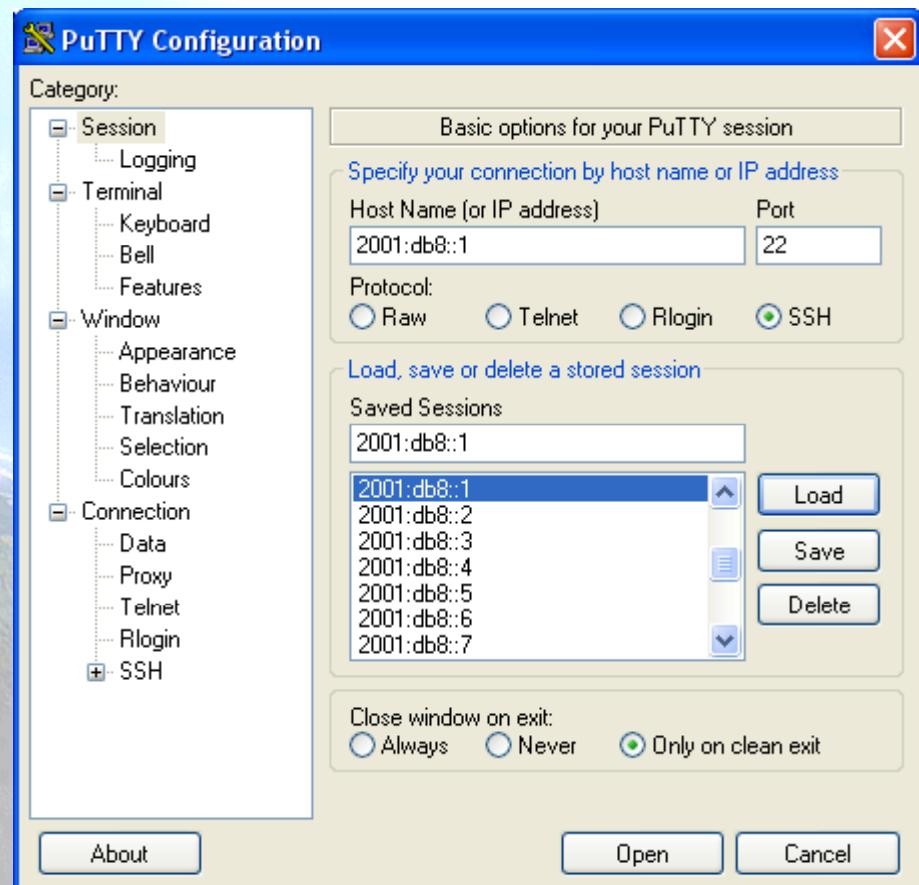
- 88

# IPv6 Applications (2)

- **DNS lookups** are used to make or differentiate an available service through IPv4 and/or IPv6
- If a client wants to connect to service.example.com, when resolving the domain name he/she can get an IPv4, IPv6 or both addresses
- In the case of getting both (v4 and v6) it is up to the client which protocol (v4/v6) to choose. The common practice is to choose v6 as the first option by default



# IPv6 Applications (3)

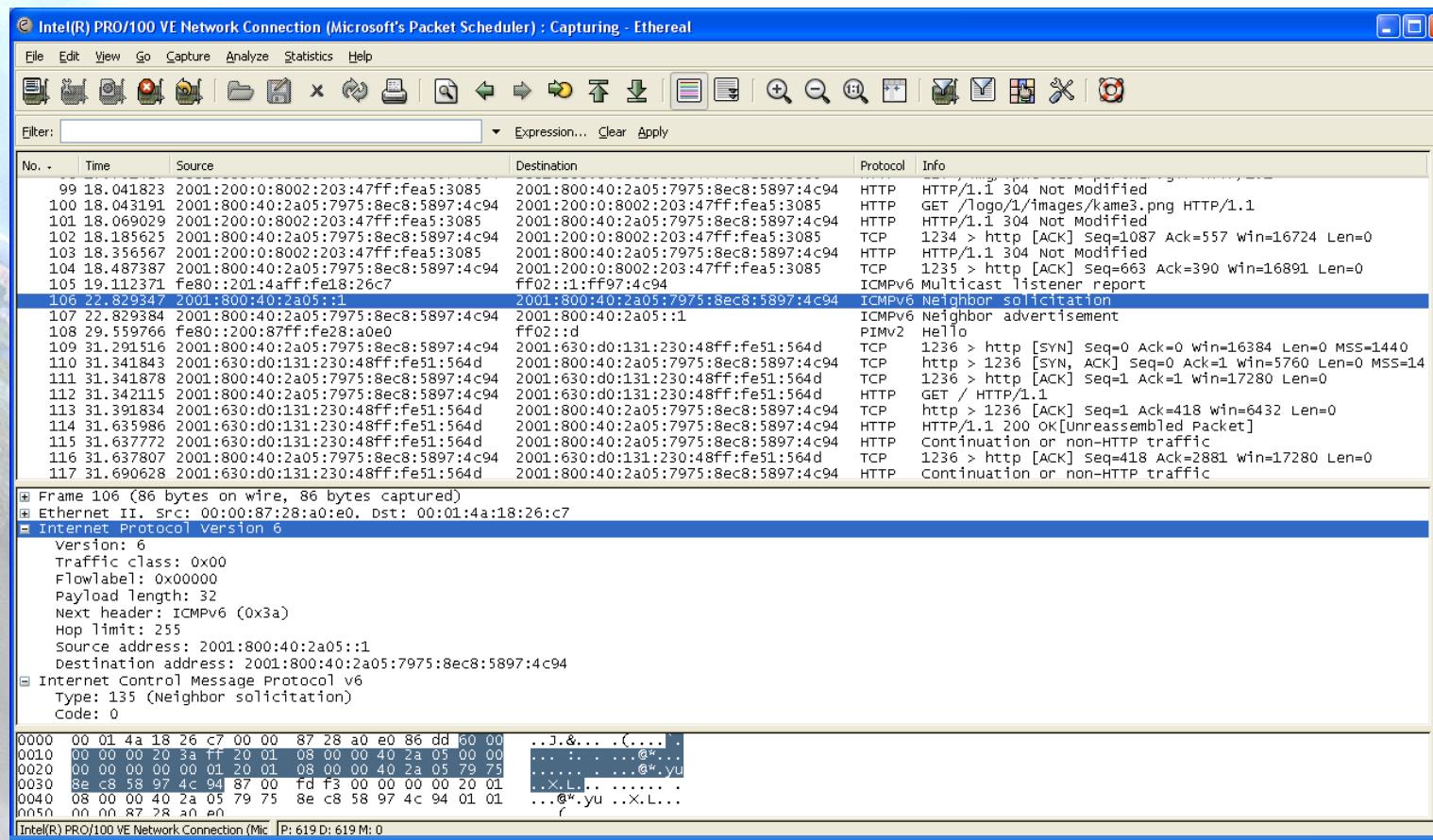


- Putty
- IPv4/IPv6 Client for Telnet and SSH
- Very useful for Administration and Management of devices
- Available at <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>



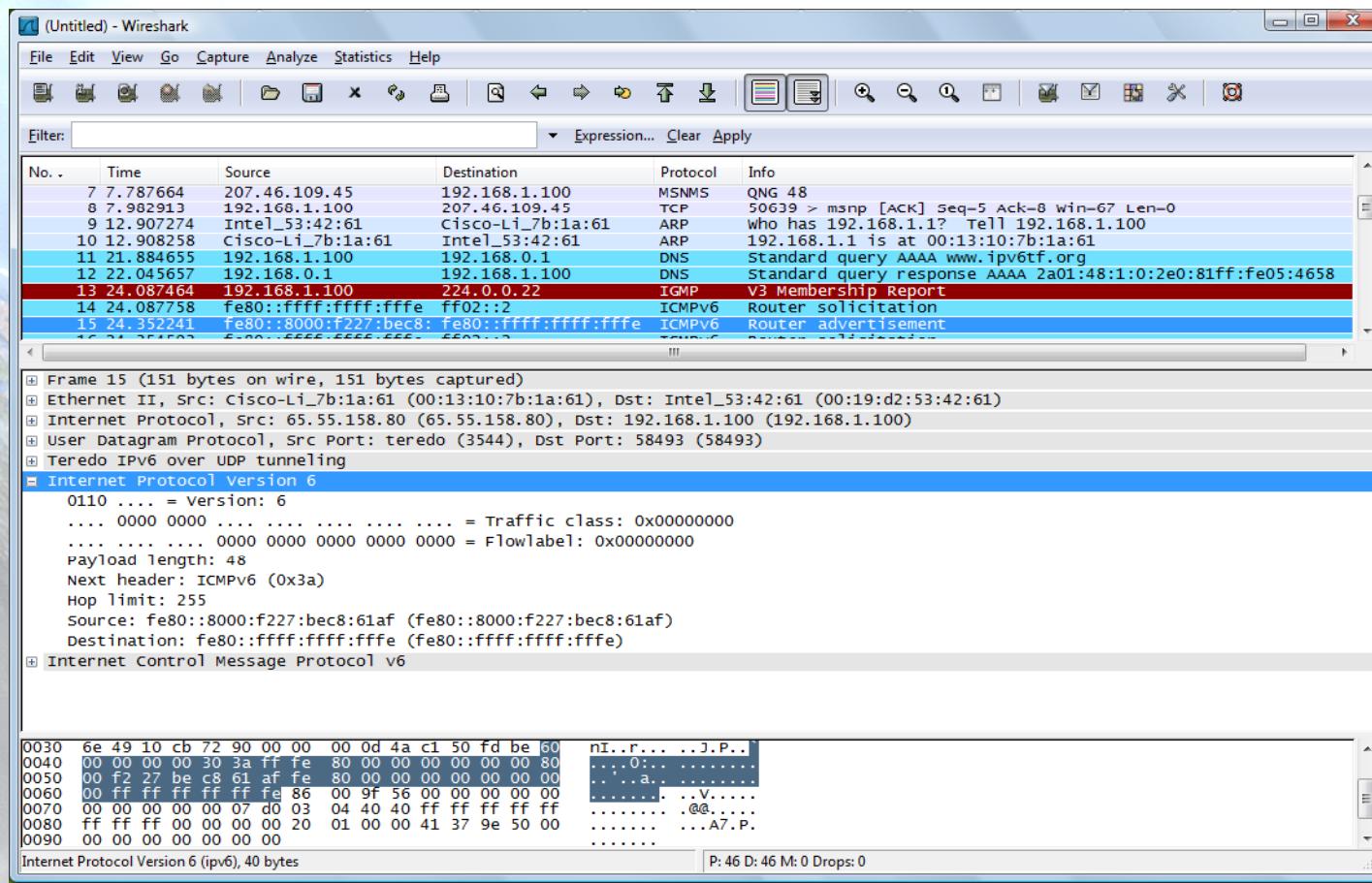
# IPv6 Applications (4)

- **Ethereal** (last version 0.99.0 – 4/24/2006)
- Captures y Decodes IPv4/IPv6 Traffic
- Very useful for connectivity validation and troubleshooting
- Available at <http://www.ethereal.com/download.html>



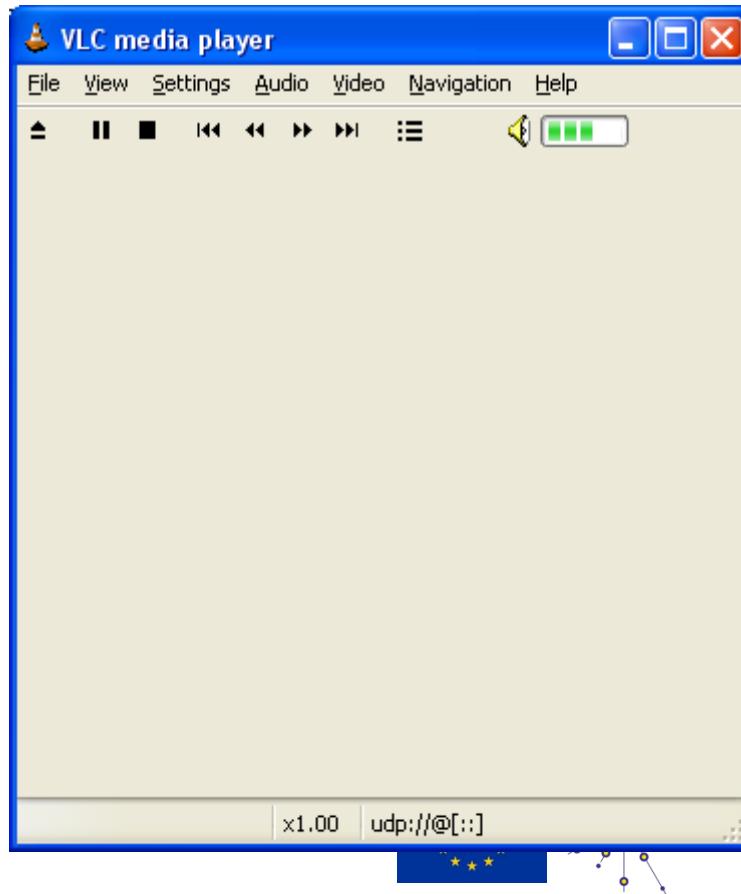
# IPv6 Applications (5)

- Wireshark (last version 1.0.0 - 3/31/2008)
- Captures and Decodes IPv4/IPv6 Traffic
- Very useful for connectivity validation and troubleshooting
- Available at <http://www.wireshark.org/download.html>



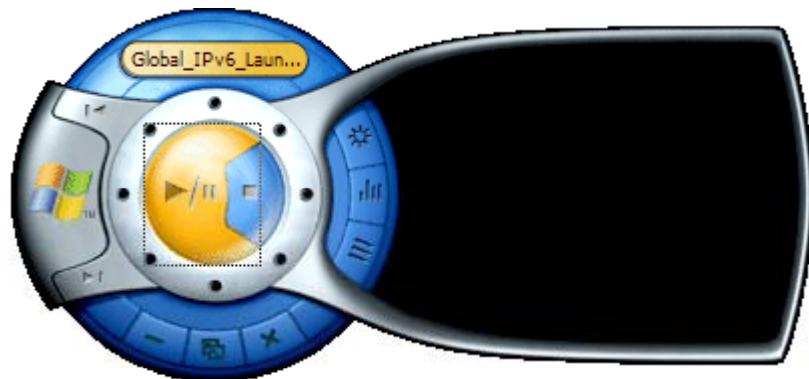
# IPv6 Applications (6)

- VLC
- Multimedia Client and Server
- Unicast y Multicast Support
- Available at <http://www.videolan.org/vlc/>



# IPv6 Applications (7)

- Microsoft Windows Media Player and Server
- Multimedia Client and Server
- It supports both IPv4/IPv6 Unicast/Multicast
- <http://www.microsoft.com/windows/windowsmedia/default.aspx>



WMP Client



# IPv6 Applications (8)

- ISABEL
- IPv4/IPv6 Unicast/Multicast
- <http://isabel.dit.upm.es/>



# IPv6 Applications (9)

- BitTorrent
  - File Sharing



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 96

# IPv6 Applications (10)

- **VNC**
  - Remote Access to a PC using IPv6
  - Graphic Environment
- Client/server Model
  - Server installed in remote PC which is the target
  - Client installed in local PC for remote access
- Supported for
  - Windows XP
  - Linux
- Available at
  - <http://jungla.dit.upm.es/~acosta/paginas/vncIPv6.html>



# IPv6 Applications (11)

- Web
- The most common Clients: Firefox, IE, Konqueror, Opera, Safari support IPv6
- Servers: Apache 2, IIS 6+ support IPv6

The screenshot shows a Mozilla Firefox window displaying 'The IPv6 Portal' website. The URL in the address bar is <http://www.ipv6tf.org/news/newsroom.php>. The page features a large yellow '2' indicating it's the second news item. The main headline is 'OMB details milestones to move to IPv6'. Other news items include '3G Americas Publishes White Paper on Convergence and Its Cross-Industry Impact' and 'Korea Moves into 'Ubiquitous' Mode'. A sidebar on the right contains links for 'SEARCH', 'POLICY MAKER', 'JOURNALIST', 'ISP', 'MANAGER', 'ENGINEER', 'END USER', 'NEWSROOM', 'POLL', 'FAQS', 'PROJECTS', and 'WHO'S ONLINE'. A login form is also present in the sidebar.

# IPv6 Applications (12)

- FreeBSD
- You can use FreeBSD ports:

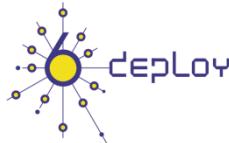
```
#>cd /usr/ports  
#>make search key="ipv6"
```
- A list of available IPv6 applications with IPv6 support will appear. Among the information of each application you can find the *path*, which is the folder where we will go and from where we can install the application:

```
#>cd path  
#>make install
```
- This starts a search over different source code servers, from where the application will be downloaded, compiled and installed
- You can also download just the source code, that will be in /usr/ports/distfiles, using instead of make install, make fetch



# IPv6 Applications: Exercise 1

- To install (in case those are not already installed):
  - SSH Client with IPv6 support (Putty)
  - FTP Client (Command line on BSD, Linux, Windows)
  - Web Browser (Firefox, IE)
  - Ethereal/Wireshark
  - VLC
  - VNC



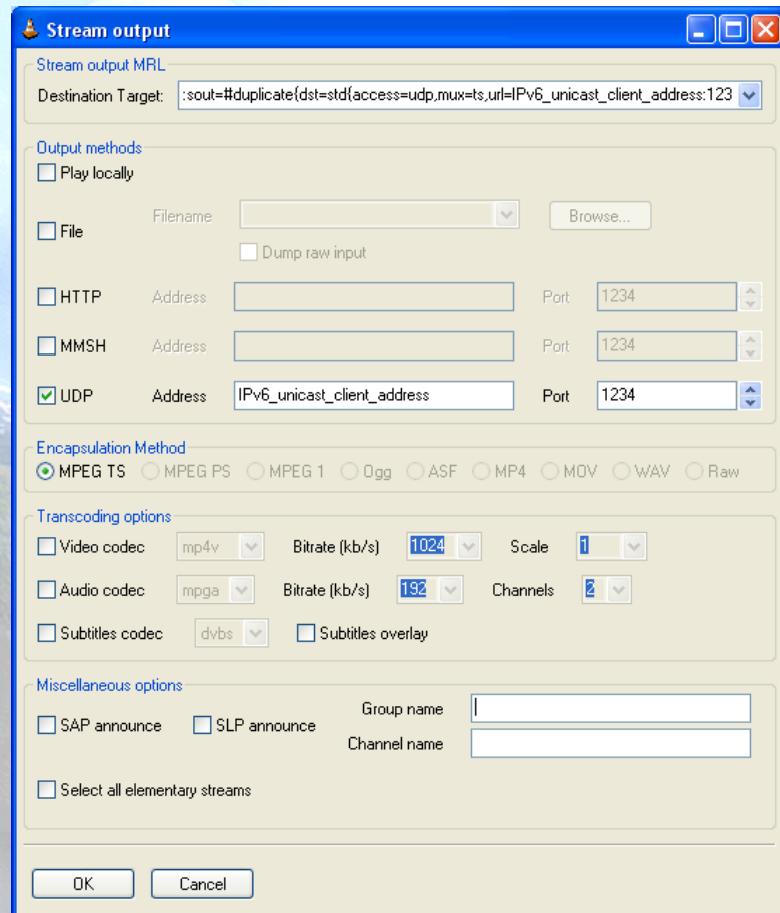
# IPv6 Applications: Exercise 2

- To use the different services while Ethereal/Wireshark (or tcpdump) is used, in order to capture packets
- To use the SSH client to access by v4 or v6 choosing by means of DNS resolution
- To use the SSH client to access by v4 or v6 choosing by means of an application parameter (linux: #ssh -6|-4)(XP: ping -6|-4)

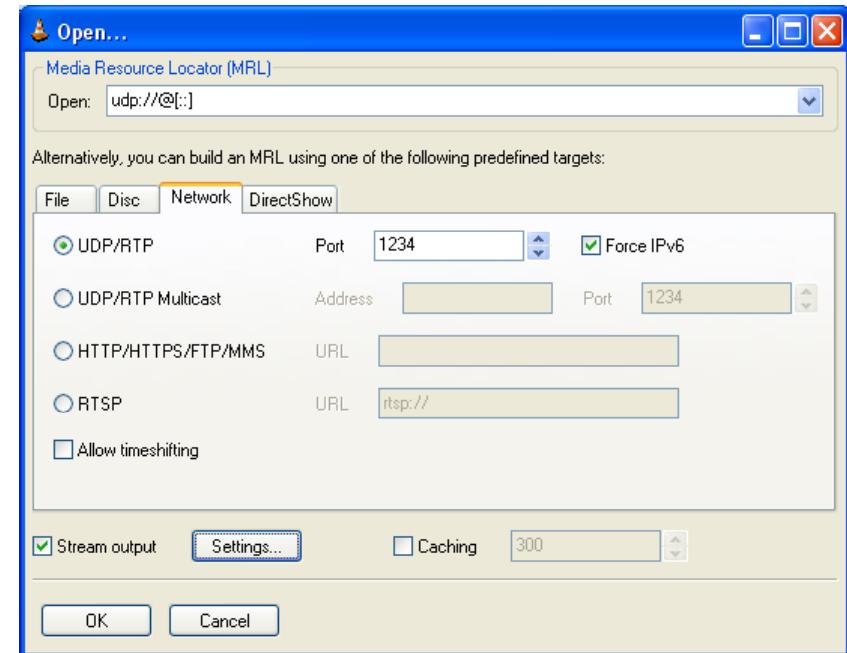


# IPv6 Applications: Exercise 3 (1)

- VLC with Unicast



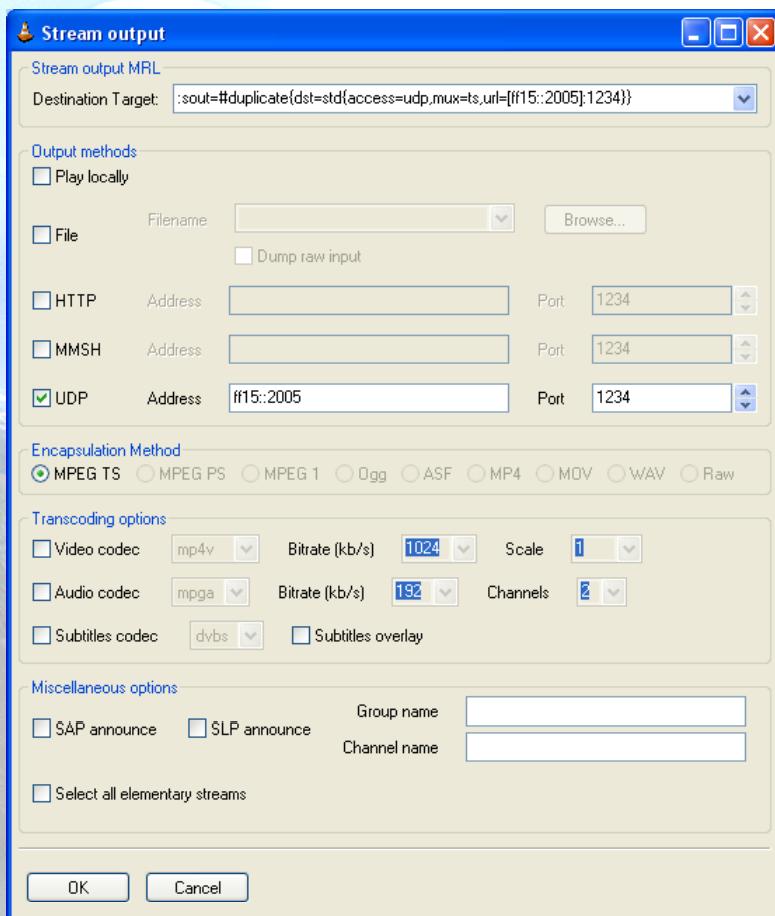
Server



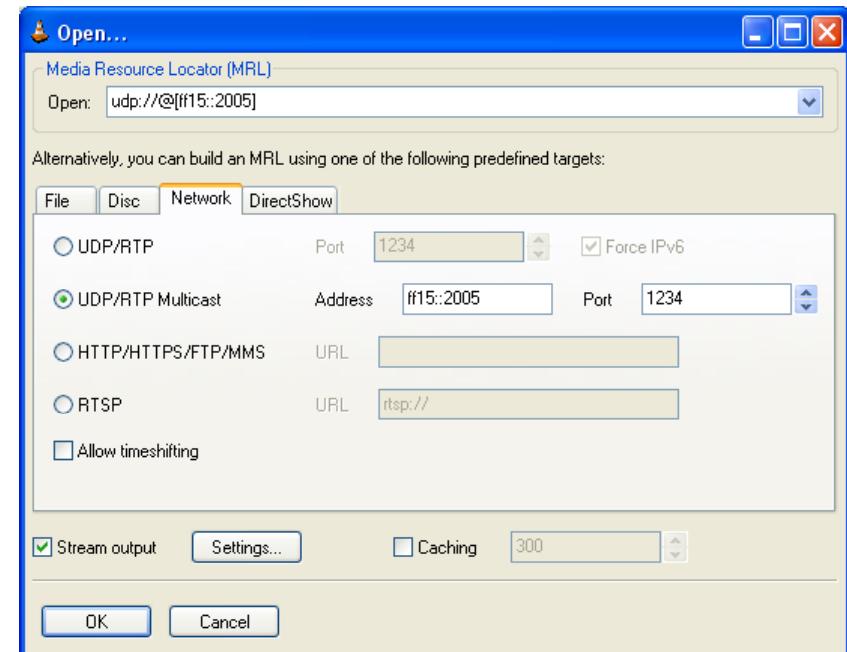
Client

# IPv6 Applications: Exercise 3 (2)

- VLC with Multicast



Server



Client

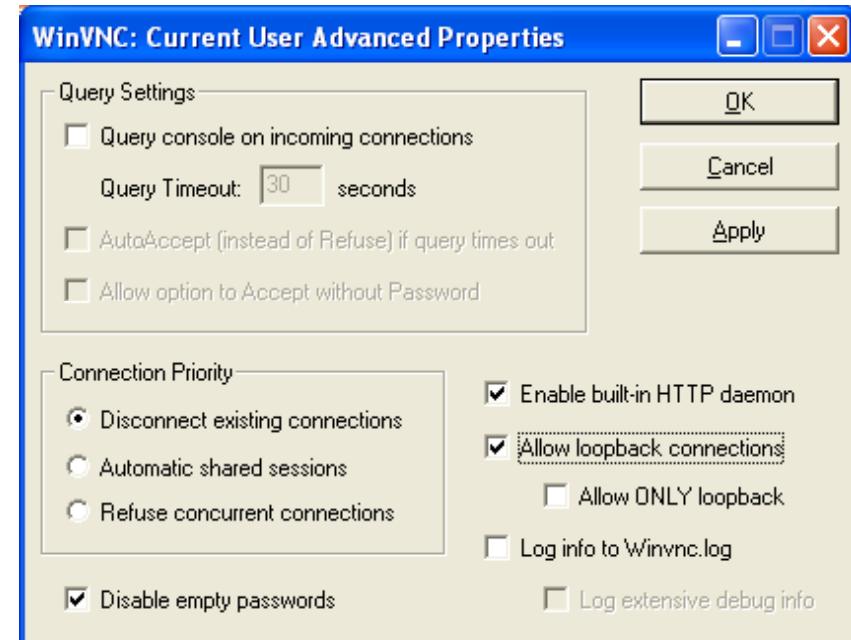
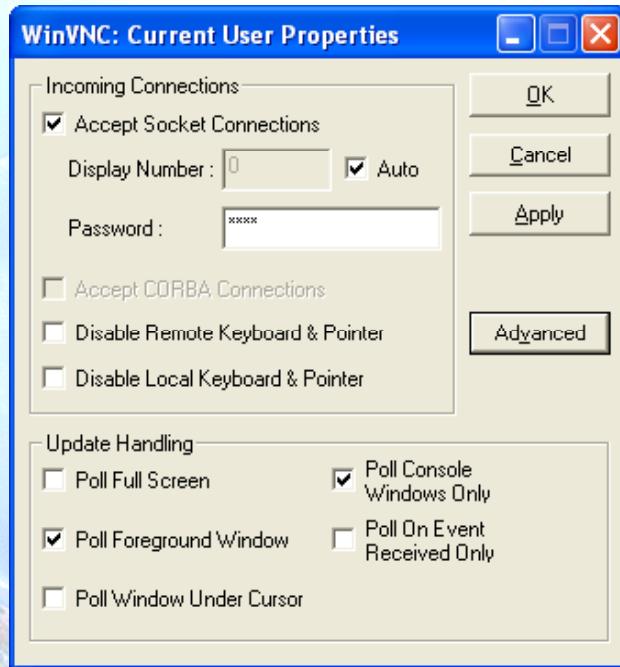


**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 103

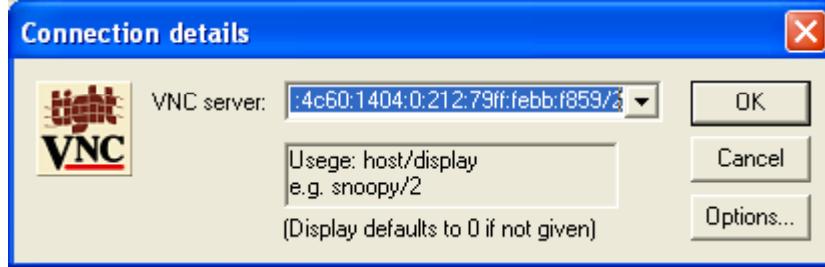
# IPv6 Applications: Exercise 4 (1)



- **VNC Server Properties**
  - It is needed to configure the “Display Number” so as to receive the connections
    - Default value is 0
  - It is needed to define a password
- VNC Server Properties ==> Advanced
  - Also enable “allow loopback connections”



# IPv6 Applications: Exercise 4 (2)

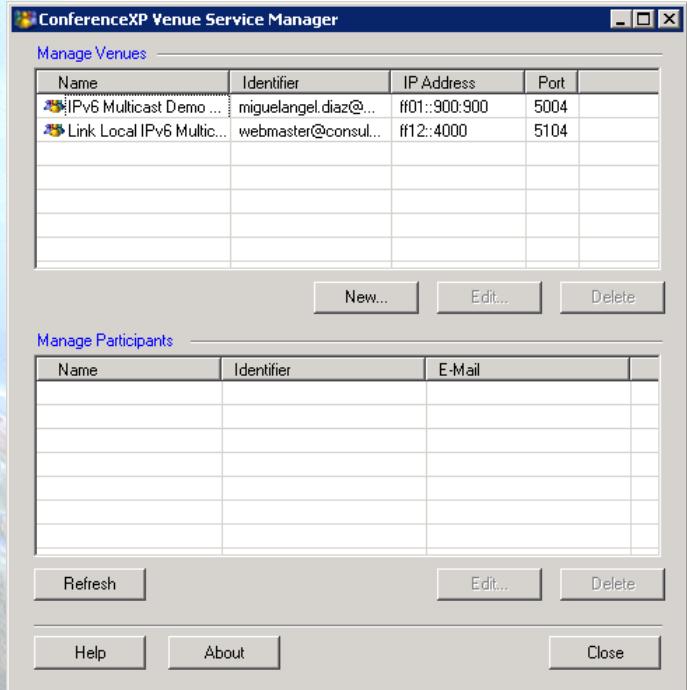


- **VNC client**
  - VNC server is specified through
    - An IPv6 address
    - Or a DNS name
  - Then, the “Display” is added after the VNC server
    - It is specified by a number separate from VNC server with a ‘/’



# IPv6 Applications: Exercise 5

- ConferenceXP with Multicast



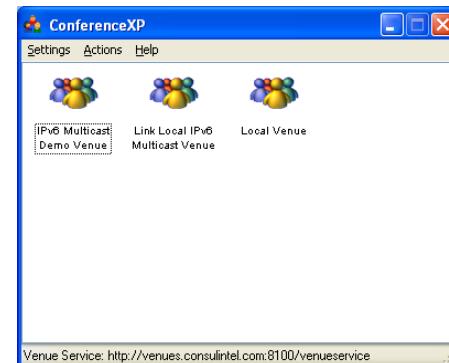
Venue Service Manager

Windows Server 2003, Enterprise edition



Reflector Service

Windows Server 2003, Enterprise edition



Client

Windows XP

# Part 5

## IPv6 DNS



# IPv6 DNS (1)

- Exercise: BIND ([www.isc.org](http://www.isc.org)) in Linux
  1. Installation BIND 9.x (Download apt or red-had package)
  2. Configuration
  3. Tests



# IPv6 DNS (2)

- BIND Configuration:

**/etc/named.conf**: is the main configuration file. There are the following options:

```
options {  
    directory "/var/named/";  
    listen-on-v6 { any; };  
};
```

Which inform about the directory containing the rest of configuration files and also enables IPv6 support



# IPv6 DNS (3)

- BIND Configuration:

**/etc/named.conf**: includes the declaration of forward and reverse zones that the server will manage, not only as master but also as slave:

```
zone  "."  {
    type hint;
    file  "named.ca";
};

zone  "localhost"  {
    type master;
    file  "localhost.zone";
};

zone  "learn.example.com"  {
    type master;
    file  "learn.example.zone";
};
```



# IPv6 DNS (4)

- **/etc/named.conf:**



# IPv6 DNS (5)

- **/var/named/learn.example.com:**

```
$TTL 86400
@ IN SOA ns1.example.com. dnsadmin.example.com (
    2002071901 ; serial
    28800 ; refresh
    7200 ; retry
    604800 ; expire
    86400 ; ttl
)
IN NS ns1.example.com.

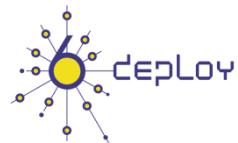
www.learn.example.com. IN AAAA 2001:db8:40:2a03::11
ftp.learn.example.com. IN AAAA 2001:db8:40:2a03::11

db.learn.example.com. IN AAAA 2001:db8:40:2a03::12
```



# IPv6 DNS (6)

- **/var/named/2001:0db8:0006:2a.zone:**



# IPv6 DNS (7)

- Tests:

- #>dig aaaa www.learn.example.com

;; QUESTION SECTION:

; www.learn.example.com. IN AAAA

;; ANSWER SECTION:

www.learn.example.com. 86400 IN AAAA 2001:db8:1000:1::103



## **QUESTION SECTION:**

; 1.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.3.0.a.2.0.4.0.0.f.f.f.f.e.f.f.3.ip6.arpa.  
IN PTR

**ANSWER SECTION:**

```
1.1.0.0.0.0.0.0.0.0.0.0.0.0.0.3.0.a.2.0.4.0.0.f.f.f.f.e.f.f.3.ip6.arpa.  
86400 IN PTR www.learn.example.com.
```

;; AUTHORITY SECTION:

a.2.0.4.0.0.f.f.f.e.f.f.3.ip6.arpa. 172800 IN NS ns1.example.com.



# DNS IPv6: Windows 2003 dnscmd (1)

Usage: DnsCmd <ServerName> <Command> [<Command Parameters>]

<ServerName>:  
IP address or host name -- remote or local DNS server  
. -- DNS server on local machine

<Command>:  
/Info -- Get server information  
/Config -- Reset server or zone configuration  
/EnumZones -- Enumerate zones  
/Statistics -- Query/clear server statistics data  
/ClearCache -- Clear DNS server cache  
/WriteBackFiles -- Write back all zone or root-hint datafile(s)  
/StartScavenging -- Initiates server scavenging  
/ResetListenAddresses -- Set server IP address(es) to serve DNS requests  
/ResetForwarders -- Set DNS servers to forward recursive queries to  
/ZoneInfo -- View zone information  
/ZoneAdd -- Create a new zone on the DNS server  
/ZoneDelete -- Delete a zone from DNS server or DS  
/ZonePause -- Pause a zone  
/ZoneResume -- Resume a zone  
/ZoneReload -- Reload zone from its database (file or DS)  
/ZoneWriteBack -- Write back zone to file  
/ZoneRefresh -- Force refresh of secondary zone from master  
/ZoneUpdateFromDs -- Update a DS integrated zone by data from DS  
/ZonePrint -- Display all records in the zone  
/ZoneResetType -- Change zone type  
/ZoneResetSecondaries -- Reset secondary\notify information for a zone  
/ZoneResetScavengeServers -- Reset scavenging servers for a zone  
/ZoneResetMasters -- Reset secondary zone's master servers  
/ZoneExport -- Export a zone to file  
/ZoneChangeDirectoryPartition -- Move a zone to another directory partition  
/EnumRecords -- Enumerate records at a name  
/RecordAdd -- Create a record in zone or RootHints  
/RecordDelete -- Delete a record from zone, RootHints or cache  
/NodeDelete -- Delete all records at a name  
/AgeAllRecords -- Force aging on node(s) in zone  
/EnumDirectoryPartitions -- Enumerate directory partitions  
/DirectoryPartitionInfo -- Get info on a directory partition  
/CreateDirectoryPartition -- Create a directory partition  
/DeleteDirectoryPartition -- Delete a directory partition  
/EnlistDirectoryPartition -- Add DNS server to partition replication scope  
/UnenlistDirectoryPartition -- Remove DNS server from replication scope  
/CreateBuiltInDirectoryPartitions -- Create built-in partitions

<Command Parameters>:  
DnsCmd <CommandName> /? -- For help info on specific Command



# DNS IPv6: Windows 2003 dnscmd (2)

C:\>dnscmd ::1 /Info

Query result:

Server info

server name	=
	dns1.novagnet.com
version	= 0ECE0205 (5.2
build 3790)	
DS container	= N/A
forest name	= N/A
domain name	= N/A
builtin domain partition	= N/A
builtin forest partition	= N/A
last scavenge cycle	= not since
restart (0)	

Configuration:

dwLogLevel	= 00000000
dwDebugLevel	= 00000000
dwRpcProtocol	= FFFFFFFF
dwNameCheckFlag	= 00000002
cAddressAnswerLimit	= 0
dwRecursionRetry	= 3
dwRecursionTimeout	= 15
dwDsPollingInterval	= 180

Configuration Flags:

fBootMethod	= 1
fAdminConfigured	= 1
fAllowUpdate	= 1
fDsAvailable	= 0
fAutoReverseZones	= 1
fAutoCacheUpdate	= 0
fSlave	= 0
fNoRecursion	= 0
fRoundRobin	= 1
fStrictFileParsing	= 0
fLooseWildcarding	= 0
fBindSecondaries	= 1
fWriteAuthorityNs	= 0
fLocalNetPriority	= 1

ServerAddresses:

Addr Count = 1

Addr[0] => 213.172.48.139

ListenAddresses:

NULL IP Array.

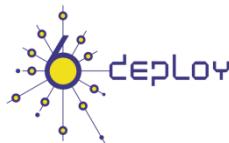
Forwarders:

NULL IP Array.

forward timeout = 5

# DNS IPv6: Windows 2003 dnscmd (3)

- Enabling IPv6 in the DNS server
  - dnscmd /config /EnableIPv6 1
  - Dnscmd.exe is part of Windows Server 2003 Support Tools. These tools can be found at the Support\Tools folder of the Windows Server 2003 CD and they are installed by running suptools.msi in such a folder
  - Restart the DNS server
- Adding a zone
  - dnscmd serverName /ZoneAdd zoneName zoneType [options]
- Deleting a zone
  - dnsdmd serverName /ZoneDelete zoneName [/DsDel] [/f]
- Adding a record
  - dnscmd serverName /RecordAdd zoneName nodeName [/Aging] [/OpenAcl] [Ttl] typeRR dataRR
- Deleting a record
  - dnscmd serverName /RecordDelete zoneName nodeName typeRR dataRR [/f]



# DNS IPv6: Exercise 1 (1)

- 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 =

2a01:48:1:0:2e0:81ff:fe05:4658



# DNS IPv6: Exercise 1 (2)

- 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
```

```
2a01:48:1:0:2e0:81ff:fe05:4658
```



# DNS IPv6: Exercise 1 (3)

- Linux:

```
#dig aaaa www.kame.net @  
2a01:48:1:0:2e0:81ff:fe05:4658
```

```
;; 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: 2a01:48:1:0:2e0:81ff:fe05:4658#53
```

```
(2a01:48:1:0:2e0:81ff:fe05:4658)
```

```
;; WHEN: Fri Jun 24 13:49:41 2005
```

```
;; MSG SIZE rcvd: 107
```



- 120

## Part 6

# IPv6 and PPP



# IPv6 and PPP

- PPP (point-to-point protocol) provides an standard method for transporting datagrams of several protocols over point-to-point links. [RFC1661]
- PPP has been updated to support the transport of IPv6 datagrams [RFC2472]
- It consists of three parts:
  - Definition of the encapsulation method of IPv6 datagrams over a point-to-point link
  - LCP (Link Control Protocol) to establish, configure and check the connection on the link level
  - NCP (Network Control Protocol) to establish and configure the network level
    - It only provides one link local address (fe80::/64)



# Implementations PPP

- Windows
  - Not IPv6 support in Windows 2000/XP/2003
  - Support in Windows Vista
- Linux/\*BSD/Solaris
  - <http://sourceforge.net/projects/ppcbc>
  - <http://freshmeat.net/projects/pppd>
- Unix implementations include:
  - /etc/ppp/ipv6-up
  - /etc/ppp/ipv6-down
  - /etc/ppp/ip-up.ipv6to4
  - /etc/ppp/ip-down.ipv6to4



# Configuration example pppd (1)

- /etc/ppp/options.server

```
[root@ns1 root]# more /etc/ppp/options.server
debug
lock
modem
crtsccts
/dev/ttys0
115200
escape FF
asyncmap 0
auth
192.168.11.201:192.168.11.220
+ipv6
ipv6 ::201,::220
proxyarp
nodetach
require-chap
refuse-pap
show-password
```

- Client

- pppd connect 'chat -v "' AT OK ATM1L2 OK ATDT911518197 CONNECT' /dev/ttys0  
115200 lock crtsccts debug asyncmap 0 escape FF +ipv6 user USUARIO password  
MI\_PASSWORD



- 124

# Configuration example pppd (2)

```
ppp0 Link encap:Point-to-Point Protocol
      inet addr:192.168.11.201 P-t-P:192.168.11.220
              Mask:255.255.255.255
      inet6 addr: fe80::201/10 Scope:Link
              UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
              RX packets:11 errors:0 dropped:0 overruns:0 frame:0
              TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:3
              RX bytes:258 (258.0 b) TX bytes:252 (252.0 b)
```

```
ppp0 Link encap:Point-to-Point Protocol
      inet addr:192.168.11.220 P-t-P:192.168.11.201
              Mask:255.255.255.255
      inet6 addr: fe80::220/10 Scope:Link
              UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
              RX packets:9 errors:2 dropped:0 overruns:0 frame:0
              TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:3
              RX bytes:156 (156.0 b) TX bytes:162 (162.0 b)
```



# Use of pppd

- RAS (Remote Access Server)
  - Loggin into the host/network by using a modem
  - It is required:
    - ppp with IPv6 support
    - Configure
      - /etc/inittab
      - /etc/mgetty+sendfax/login.config
      - /etc/mgetty+sendfax/mgetty.config
      - /etc/ppp/options.server
    - One autoconfiguration method (radvd or dhcpcv6)
      - /etc/sysconfig/network
      - /etc/ppp/ipv6-up.local
      - /etc/ppp/ipv6-down.local
      - /etc/radvd.conf
- VPNs (Server: softwires, pptpd, client: pptp)
  - They use PPP as link protocol, so IPv6 is used as network protocol over the VPN



## Part 7

# Firewall IPv6



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

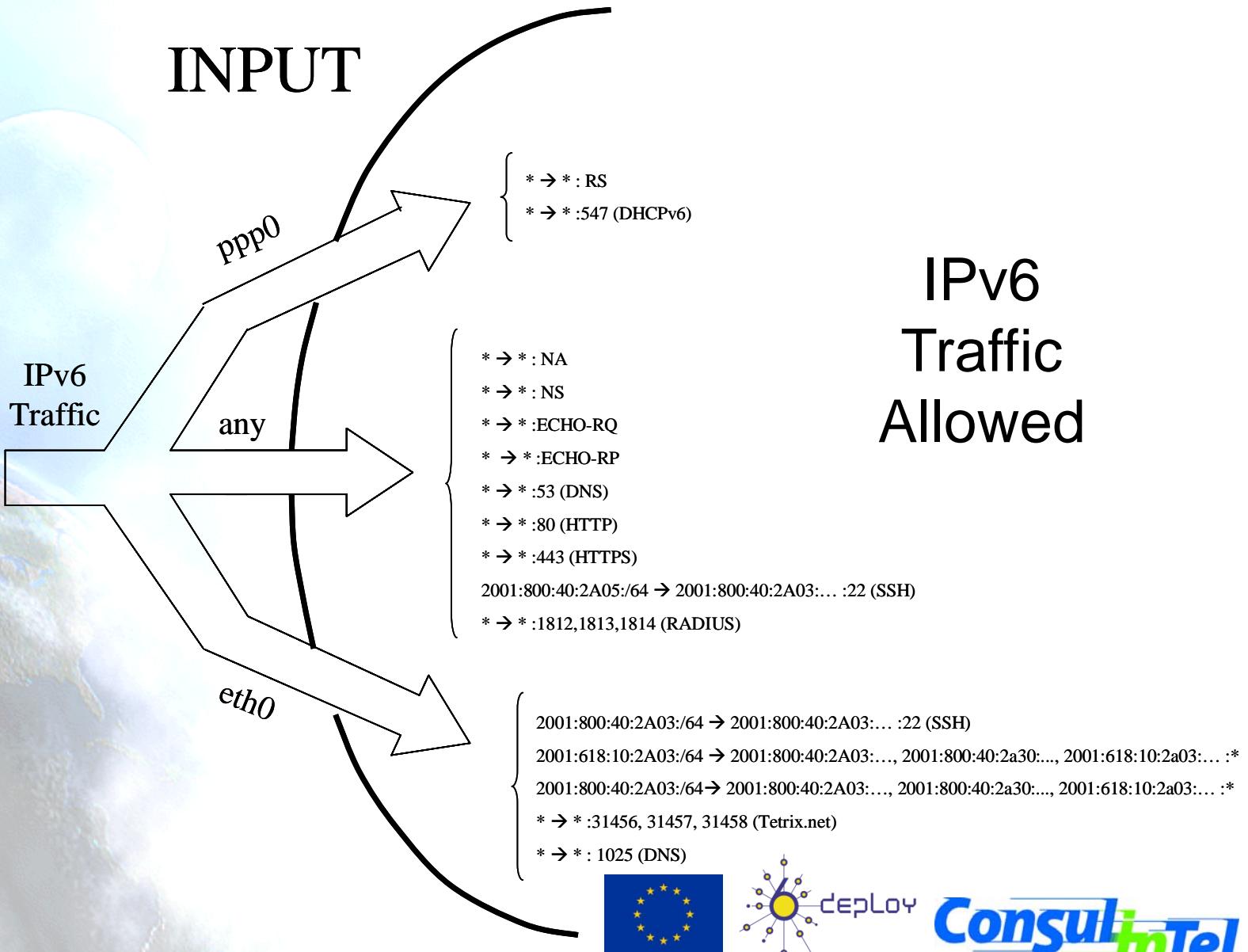
- 127

# Firewall IPv6

- Windows XP/2003
  - Common security GUI for IPv4 and IPv6
  - Specific configurations with “netsh firewall”
    - add - add the security server configuration.
    - delete - delete the security server configuration.
    - dump - show the configuration command sequence.
    - help - show the command list.
    - reset - reset the security server configuration.
    - set - set the security server configuration.
    - show - show the security server configuration.
- Unix systems
  - ip6tables. Tool that configures and shows the kernel built-in filter tables.
  - Functionality similar to the IPv4 iptables



# Firewall Example IPv6 Linux (1)

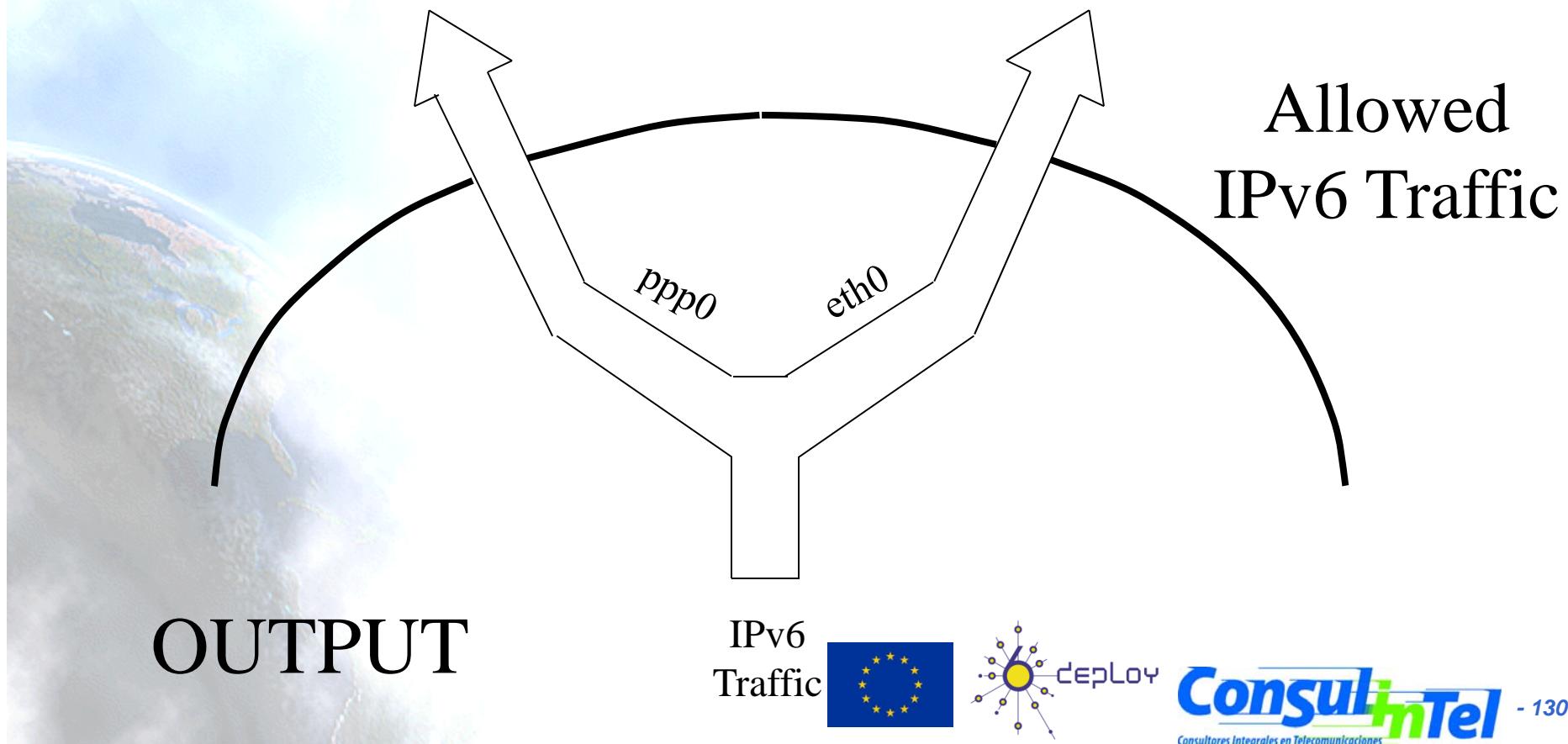


# Firewall Example IPv6 Linux (2)

$\left\{ \begin{array}{l} * \rightarrow * : RA \\ * \rightarrow * : NA \\ * \rightarrow * : NS \\ * \rightarrow * : 546 (\text{DHCPv6}) \\ 2001:618:10:2A03:\dots \\ 2001:800:40:2A03:\dots \\ 2001:618:10:2A03:\dots \end{array} \right\} \rightarrow 2001:800:40:2A30:/64 :*$

$\left\{ \begin{array}{l} 2001:618:10:2A03:\dots \\ 2001:800:40:2A03:\dots \\ 2001:618:10:2A03:\dots \end{array} \right\} \rightarrow 2001:800:40:2A03:/64, 2001:618:10:2A03:/64 :*$

$2001:800:40:2A03:\dots, 2001:618:10:2A03:\dots \rightarrow * :*$



IPv6  
Traffic



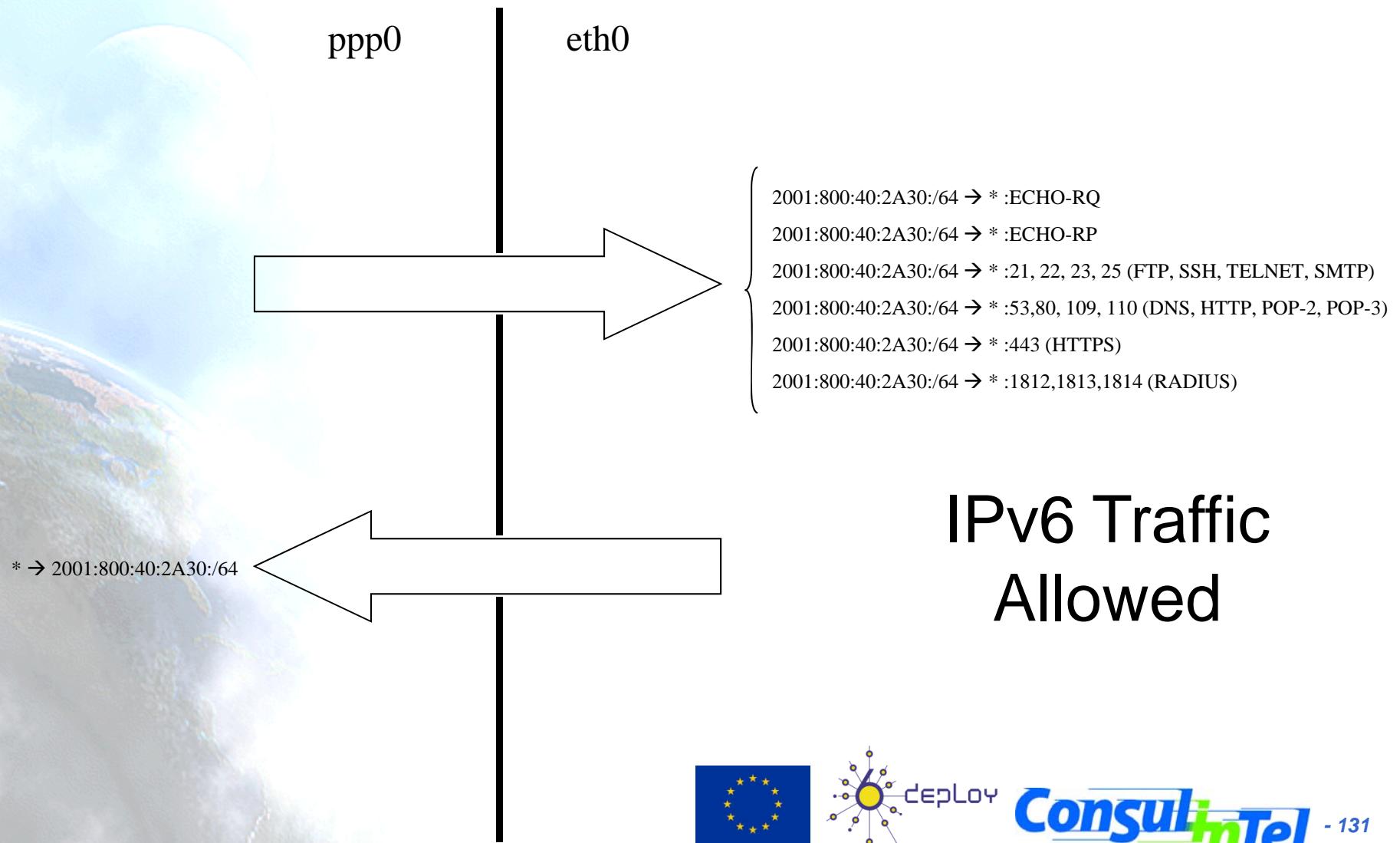
**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 130

# Firewall Example IPv6 Linux (3)

## FORWARDING



# Firewall Example IPv6 Linux (4)

```
#!/bin/sh
#
# rc.firewall-2.4-stronger for zafiro
#
FWVER=0.77s

echo -e "\nLoading rc.firewall - version $FWVER..\n"

IP6TABLES=/sbin/ip6tables
LSMOD=/sbin/lsmod
DEPMOD=/sbin/depmod
INSMOD=/sbin/insmod
GREP=/bin/grep
AWK=/bin/awk
SED=/bin/sed
IFCONFIG=/sbin/ifconfig

#Setting the EXTERNAL and INTERNAL interfaces for the network
EXTIF="ppp0"
INTIF="eth0"
echo " External Interface: $EXTIF"
echo " Internal Interface: $INTIF"
echo " ---"
```



# Firewall Example IPv6 Linux (5)

```
# Specify your IP address here or let the script take care of it for you.
```

```
#  
INTIPV6_1="2001:800:40:2a03:204:acff:fe77:b83d"  
INTIPV6_2="2001:618:10:2a03:204:acff:fe77:b83d"  
INTNETV6_1="2001:800:40:2a03::/64"  
INTNETV6_2="2001:618:10:2a03::/64"  
echo " Internal IPv6 1: $INTIPV6_1"  
echo " Internal IPv6 2: $INTIPV6_2"  
echo " Internal IPv6 Network 1: $INTNETV6_1"  
echo " Internal IPv6 Network 2: $INTNETV6_2"  
echo " ---"
```

```
# Assign the external TCP/IP network and IP address
```

```
EXTNETV6="2001:800:40:2a30::/64"  
EXTIPV6="2001:800:40:2a30::201"  
echo " External Network: $EXTNETV6"  
echo " External IPv6: $EXTIPV6"  
echo " ---"
```

```
# Setting a few other local variables
```

```
#  
UNIVERSE="::/0"
```



# Firewall Example IPv6 Linux (6)

```
# Need to verify that all modules have all required dependencies
echo " Verifying that all kernel modules are ok"
$DEPMOD -a

echo -en " Loading kernel modules: "
echo -en "ip6_tables, "
#
#Verify the module isn't loaded. If it is, skip it
#
if [ -z `\$LSMOD | \$GREP ip6_tables | \$AWK {'print $1'}` `` ]; then
    $INSMOD ip6_tables
fi

echo " ---"

#####
#
#Clearing any previous configuration
# Unless specified, the defaults for INPUT, OUTPUT, and FORWARD to DROP
#
echo " Clearing any existing rules and setting default policy to DROP.."
$IP6TABLES -P INPUT DROP
$IP6TABLES -F INPUT
$IP6TABLES -P OUTPUT DROP
$IP6TABLES -F OUTPUT
```



- 134

# Firewall Example IPv6 Linux (7)

```
$IP6TABLES -P FORWARD DROP  
$IP6TABLES -F FORWARD  
#  
#Flush the user chain.. if it exists  
if [ -n "$IP6TABLES -L | $GREP drop-and-log-it" ]; then  
    $IP6TABLES -F drop-and-log-it  
fi  
#  
# Delete all User-specified chains  
$IP6TABLES -X  
#  
# Reset all IP6TABLES counters  
$IP6TABLES -Z
```

```
#Configuring specific CHAINS for later use in the ruleset  
#  
echo " Creating a DROP chain: 'drop-and-log-it'."  
echo " ---"  
$IP6TABLES -N drop-and-log-it  
$IP6TABLES -A drop-and-log-it -j LOG --log-level info  
$IP6TABLES -A drop-and-log-it -j DROP  
  
echo -e " Loading INPUT rulesets"
```



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 135

# Firewall Example IPv6 Linux (8)

```
#####
# INPUT: Incoming traffic from various interfaces. All rulesets are
#       already flushed and set to a default policy of DROP.
#
#
# loopback interfaces are valid.
#
$IP6TABLES -A INPUT -i lo -s $UNIVERSE -d $UNIVERSE -j ACCEPT

# ***** Internal specific interface rules *****
#
# all traffic from local network is valid
#
$IP6TABLES -A INPUT -i $INTIF -s $INTNETV6_1 -d $INTIPV6_1 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -s $INTNETV6_1 -d $INTIPV6_2 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -s $INTNETV6_1 -d $EXTIPV6 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -s $INTNETV6_2 -d $INTIPV6_1 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -s $INTNETV6_2 -d $INTIPV6_2 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -s $INTNETV6_2 -d $EXTIPV6 -j ACCEPT

# SSH connections from internal interface are permitted
#
$IP6TABLES -A INPUT -i $INTIF -p tcp -s $INTNETV6_1 -d $INTIPV6_1 --destination-port 22 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -p tcp -s 2001:800:40:2a05::/64 -d $INTIPV6_1 --destination-port 22 -j ACCEPT
```



# Firewall Example IPv6 Linux (9)

```
# OPEN PORTS on 'esmeralda' before start ppp link
#
$IP6TABLES -A INPUT -i $INTIF -p tcp --destination-port 31456 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -p tcp --destination-port 31457 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -p tcp --destination-port 31458 -j ACCEPT
$IP6TABLES -A INPUT -i $INTIF -p udp --destination-port 1025 -j ACCEPT

# ***** External specific interface rules *****
# remote interface, claiming to be local machines, IP spoofing, get lost
#
$IP6TABLES -A INPUT -i $EXTIF -s $INTNETV6_1 -d $UNIVERSE -j drop-and-log-it
$IP6TABLES -A INPUT -i $EXTIF -s $INTNETV6_2 -d $UNIVERSE -j drop-and-log-it

# external interface, for stateless autoconfiguration traffic
#
$IP6TABLES -A INPUT -i $EXTIF -p ipv6-icmp --icmpv6-type router-solicitation -j ACCEPT

# enable internal dhcp6 server for external interface
#
$IP6TABLES -A INPUT -i $EXTIF -p tcp --destination-port 547 -j ACCEPT
$IP6TABLES -A INPUT -i $EXTIF -p udp --destination-port 547 -j ACCEPT
```



# Firewall Example IPv6 Linux (10)

```
# ***** General rules for any interface *****

# ping traffic is valid
#
$IP6TABLES -A INPUT -p ipv6-icmp --icmpv6-type echo-request -s $UNIVERSE -d $EXTIPV6 -j ACCEPT
$IP6TABLES -A INPUT -p ipv6-icmp --icmpv6-type echo-request -s $UNIVERSE -d $INTIPV6_1 -j ACCEPT
$IP6TABLES -A INPUT -p ipv6-icmp --icmpv6-type echo-request -s $UNIVERSE -d $INTIPV6_2 -j ACCEPT
$IP6TABLES -A INPUT -p ipv6-icmp --icmpv6-type echo-reply -s $UNIVERSE -d $EXTIPV6 -j ACCEPT
$IP6TABLES -A INPUT -p ipv6-icmp --icmpv6-type echo-reply -s $UNIVERSE -d $INTIPV6_1 -j ACCEPT
$IP6TABLES -A INPUT -p ipv6-icmp --icmpv6-type echo-reply -s $UNIVERSE -d $INTIPV6_2 -j ACCEPT

# external interface, for stateless autoconfiguration traffic
#
$IP6TABLES -A INPUT -p ipv6-icmp --icmpv6-type neighbour-solicitation -j ACCEPT
$IP6TABLES -A INPUT -p ipv6-icmp --icmpv6-type neighbour-advertisement -j ACCEPT

# HTTPd - Enable the following lines if you run an EXTERNAL WWW server
#
$IP6TABLES -A INPUT -p tcp -s $UNIVERSE --destination-port 80 -j ACCEPT
$IP6TABLES -A INPUT -p tcp -s $UNIVERSE --destination-port 443 -j ACCEPT

# DNS request are attendant for any interface
#
$IP6TABLES -A INPUT -p tcp --destination-port 53 -j ACCEPT
$IP6TABLES -A INPUT -p udp --destination-port 53 -j ACCEPT
```



# Firewall Example IPv6 Linux (11)

```
# RADIUS traffic is valid
#
$IP6TABLES -A INPUT -p tcp --destination-port 1812 -j ACCEPT
$IP6TABLES -A INPUT -p udp --destination-port 1812 -j ACCEPT
$IP6TABLES -A INPUT -p tcp --destination-port 1813 -j ACCEPT
$IP6TABLES -A INPUT -p udp --destination-port 1813 -j ACCEPT
$IP6TABLES -A INPUT -p tcp --destination-port 1814 -j ACCEPT
$IP6TABLES -A INPUT -p udp --destination-port 1814 -j ACCEPT

# Catch all rule, all other incoming is denied and logged.
#
$IP6TABLES -A INPUT -s $UNIVERSE -d $UNIVERSE -j drop-and-log-it

echo -e " Loading OUTPUT rulesets"

#####
# OUTPUT: Outgoing traffic from various interfaces. All rulesets are
# already flushed and set to a default policy of DROP.
#
# loopback interface is valid.
#
$IP6TABLES -A OUTPUT -o lo -s $UNIVERSE -d $UNIVERSE -j ACCEPT
```



# Firewall Example IPv6 Linux (12)

```
# ***** Internal specific interface rules *****

# local interfaces, any source going to local net is valid
#
$IP6TABLES -A OUTPUT -o $INTIF -s $EXTIPV6 -d $INTNETV6_1 -j ACCEPT
$IP6TABLES -A OUTPUT -o $INTIF -s $EXTIPV6 -d $INTNETV6_2 -j ACCEPT

# local interface, any source going to local net is valid
#
$IP6TABLES -A OUTPUT -o $INTIF -s $INTIPV6_1 -d $INTNETV6_1 -j ACCEPT
$IP6TABLES -A OUTPUT -o $INTIF -s $INTIPV6_1 -d $INTNETV6_2 -j ACCEPT
$IP6TABLES -A OUTPUT -o $INTIF -s $INTIPV6_2 -d $INTNETV6_1 -j ACCEPT
$IP6TABLES -A OUTPUT -o $INTIF -s $INTIPV6_2 -d $INTNETV6_2 -j ACCEPT

# anything else outgoing on internal interface is valid
#
$IP6TABLES -A OUTPUT -o $INTIF -s $INTIPV6_1 -d $UNIVERSE -j ACCEPT
$IP6TABLES -A OUTPUT -o $INTIF -s $INTIPV6_2 -d $UNIVERSE -j ACCEPT

# ***** External specific interface rules *****

# outgoing to local net on remote interface, stuffed routing, deny
#
$IP6TABLES -A OUTPUT -o $EXTIF -s $UNIVERSE -d $INTNETV6_1 -j drop-and-log-it
$IP6TABLES -A OUTPUT -o $EXTIF -s $UNIVERSE -d $INTNETV6_2 -j drop-and-log-it
```



# Firewall Example IPv6 Linux (13)

```
# enable stateless autoconfiguration
$IP6TABLES -A OUTPUT -o $EXTIF -p ipv6-icmp --icmpv6-type router-advertisement -j ACCEPT

# anything else outgoing on remote interface is valid
#
$IP6TABLES -A OUTPUT -o $EXTIF -s $EXTIPV6 -d $EXTNETV6 -j ACCEPT
$IP6TABLES -A OUTPUT -o $EXTIF -s $INTIPV6_1 -d $EXTNETV6 -j ACCEPT
$IP6TABLES -A OUTPUT -o $EXTIF -s $INTIPV6_2 -d $EXTNETV6 -j ACCEPT

# enable stateful autoconfiguration for the remote host (DHCPv6)
#
$IP6TABLES -A OUTPUT -o $EXTIF -p tcp --destination-port 546 -j ACCEPT
$IP6TABLES -A OUTPUT -o $EXTIF -p udp --destination-port 546 -j ACCEPT

# ***** General rules for any interface *****
# enable stateless autoconfiguration
#
$IP6TABLES -A OUTPUT -p ipv6-icmp --icmpv6-type neighbour-advertisement -j ACCEPT
$IP6TABLES -A OUTPUT -p ipv6-icmp --icmpv6-type neighbour-solicitation -j ACCEPT

# Catch all rule, all other outgoing is denied and logged.
#
$IP6TABLES -A OUTPUT -s $UNIVERSE -d $UNIVERSE -j drop-and-log-it
```



# Firewall Example IPv6 Linux (14)

```
echo -e " Loading FORWARD rulesets"

#####
# FORWARD: Enable Forwarding and thus IPMASQ
#
# ***** Forwarding from ppp0 to eth0 *****
#
# HTTP traffic is valid
#
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 80 -j ACCEPT
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 443 -j ACCEPT

# FTP traffic is valid
#
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 21 -j ACCEPT
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 21 -j ACCEPT

# DNS request are valid
#
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 53 -j ACCEPT
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 53 -j ACCEPT

# TELNET traffic is valid
#
```



# Firewall Example IPv6 Linux (15)

```
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 23 -j ACCEPT  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 23 -j ACCEPT
```

```
# SSH traffic is valid  
#  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 22 -j ACCEPT  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 22 -j ACCEPT
```

```
# POP-2 traffic is valid  
#  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 109 -j ACCEPT  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 109 -j ACCEPT
```

```
# POP-3 traffic is valid  
#  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 110 -j ACCEPT  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 110 -j ACCEPT
```

```
# SMTP traffic is valid  
#  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 25 -j ACCEPT  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 25 -j ACCEPT
```

```
# RADIUS traffic is valid  
#
```



# Firewall Example IPv6 Linux (16)

```
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 1812 -j ACCEPT  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 1812 -j ACCEPT  
  
# RADIUS ACCOUNTING traffic is valid  
#  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 1813 -j ACCEPT  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 1813 -j ACCEPT  
  
# RADIUS  
#  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p tcp --destination-port 1814 -j ACCEPT  
$IP6TABLES -A FORWARD -i $EXTIF -o $INTIF -s $EXTNETV6 -p udp --destination-port 1814 -j ACCEPT  
  
# ***** Forwarding from eth0 to ppp0 *****  
  
# FORWARDING traffic from anywhere to internal net is valid  
#  
$IP6TABLES -A FORWARD -i $INTIF -o $EXTIF -d $EXTNETV6 -j ACCEPT
```



Consultores Integrales en Telecomunicaciones

- 144

# Firewall Example IPv6 Linux (17)

```
# ***** Forwarding from any to any *****

# Ping traffic is valid
#
$IP6TABLES -A FORWARD -p ipv6-icmp --icmpv6-type echo-request -j ACCEPT
$IP6TABLES -A FORWARD -p ipv6-icmp --icmpv6-type echo-reply -j ACCEPT

# Catch all rule, all other forwarding is denied and logged.
#
$IP6TABLES -A FORWARD -j drop-and-log-it

#####
echo -e "\nStronger rc.firewall-2.4 $FWVER done.\n"
```



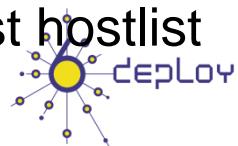
## Part 8

# Enable IPv6 and IPv6 ACLs on Cisco Routers



# Enable Telnet over IPv6 transport

- Router> enable
- Router# configure terminal
- Router(config)# ipv6 host name [port] ipv6-address1 [ipv6-address2...ipv6-address4]
  - Example: Router(config)# ipv6 host cisco-sj 2001:DB8:10:20::1
- Router(config)# line [aux | console | tty | vty] line-number [ending-line-number]
  - Example: Router(config)# line vty 0 4
- Router(config)# password password
  - Example: Router(config)# password hostword
- Router(config)# login [local | tacacs]
  - Example: Router(config)# login local
- Router(config)# ipv6 access-class acl-name (Optional: Add a host list which can access)
  - Example: Router(config)# ipv6 access-list hostlist



# Enable SSH over IPv6 transport

- Router> enable
- Router# configure terminal
- Router(config)# ip ssh {[timeout seconds] | [authentication-retries integer]}
  - Example1: Router(config)# ip ssh
  - Example2: Router(config)# ip ssh timeout 100 authentication-retries 2



**ConsultInTel**

- 148

Consultores Integrales en Telecomunicaciones

# Enable IPv6 on interfaces (1)

- Router> enable
- Router# configure terminal
- Router(config)# interface type number
- Router(config-if)# ipv6 enable
  - Example: Router(config)# ipv6 enable
- Router(config-if)# ipv6 address
  - Example: Router(config)# ipv6 address 2001:DB8:10:20::1/64 (Configure one address and sends correspondend RA messages)
- Router(config-if)# ipv6 address autoconfig (Configure one address by autoconfiguration)



# Enable IPv6 on interfaces (2)

- Is possible to configure different ND parameters
  - Router(config-if)#ipv6 nd ?
    - dad Duplicate Address Detection
    - managed-config-flag Hosts should use DHCP for address config
    - ns-interval Set advertised NS retransmission interval
    - other-config-flag Hosts should use DHCP for non-address config
    - prefix Configure IPv6 Routing Prefix Advertisement
    - ra-interval Set IPv6 Router Advertisement Interval
    - ra-lifetime Set IPv6 Router Advertisement Lifetime
    - reachable-time Set advertised reachability time
    - suppress-ra Suppress IPv6 Router Advertisements
- Is possible to configure more prefixes in the RA
  - Example: Router(config)# ipv6 nd prefix 2001:DB8:10:20::/64
- Is possible to stop the RA of certain prefix
  - Example: Router(config-if)#ipv6 nd prefix 2001:DB8:10:20::/64 no-advertise
- Is possible to suppress the RA
  - Example: Router(config)# ipv6 nd suppress-ra



# Enable IPv6 on interfaces (3)

- Is possible to configure different RA parameters
  - Router(config-if)#ipv6 nd prefix 2001:DB8:10:20::/64 ?  
<0-4294967295> Valid Lifetime (secs)  
at              Expire prefix at a specific time/date  
infinite        Infinite Valid Lifetime  
no-advertise    Do not advertise prefix  
no-autoconfig   Do not use prefix for autoconfiguration  
off-link        Do not use prefix for on link determination  
<cr>
- Router(config-if)#ipv6 nd ra-inteval
  - Configure the interval between RAs
- Router(config-if)#ipv6 nd ra-lifetime
  - Configure lifetime of the RA



# Show IPv6 information

- Interfaces
  - Router#show ipv6 interface
- Routing table
  - Router#show ipv6 route
- Routing protocols
  - Router#show ipv6 protocols
  - Router#show ipv6 ospf



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 152

# Configure 6in4 tunnel

- Router#configure terminal

```
Router(config)#interface Tunnel 0
```

```
Router(config-if)#ipv6 address <IPv6 address>/<prefix length>
```

```
Router(config-if)#tunnel source <IPv4 address>
```

```
Router(config-if)#tunnel source <Interface type> <Interface  
number>
```

```
Router(config-if)#tunnel destination <remote IPv4 address >
```

```
Router(config-if)#tunnel mode ipv6ip
```



# Access Control List to filter IPv6 traffic

- General steps
  - Create an IPv6 Access Control List (ACL)
  - Configure the IPv6 ACL to permit or deny the IPv6 traffic
  - Apply the IPv6 ACL in the interface



Consultores Integrales en Telecomunicaciones

# Create and configure the IPv6 ACL (1)

Cisco IOS Release 12.2(11)T, 12.0(22)S, 12.0(21)ST

- Router> enable
- Router# configure terminal
- Router(config)# ipv6 access-list access-list-name {permit | deny} { source-ipv6-prefix/ prefix-length | any} { destination-ipv6-prefix/ prefix-length | any} [priority value]
  - Example: Router(config)# ipv6 access-list list2 deny 2001:DB8:0:0:2::/64 any



- 155

# Create and configure the IPv6 ACL (2)

Cisco IOS Release 12.2(13)T, 12.0(23)S and later

- Router> enable
- Router# configure terminal
- Router(config)# ipv6 access-list access-list-name (Define the IPv6 ACL)
  - Example: Router(config)# ipv6 access-list outbound
- Router(config-ipv6-acl)# **permit** {protocol} {source-ipv6-prefix/ prefix-length | any | host source-ipv6-address} [operator [ port-number]] {destination-ipv6-prefix/ prefix-length | any | host destination-ipv6-address} [operator [ port-number]] [dscp value] [flow-label value] [fragments] [log] [log-input] [reflect name [timeout value]] [routing] [time-range name] [sequence value] or **deny** {protocol} {source-ipv6-prefix/ prefix-length | any | host source-ipv6-address} [ operator [port-number]] {destination-ipv6-prefix/ prefix-length | any | host destination-ipv6-address} [operator [port-number]] [dscp value] [flow-label value] [fragments] [log] [log-input] [routing] [time-range name] [undetermined transport] [sequence value] (Define the behavior of the IPv6 ACL)
  - Example1: Router(config-ipv6-acl)# permit tcp 2001:DB8:300:200::/32 eq telnet any reflect reflectout
  - Example2: Router(config-ipv6-acl)# deny tcp host 1::1 any log-input



- 156

# Apply the IPv6 ACL to the interface

Cisco IOS Release 12.2(11)T, 12.0(22)S, 12.0(21)ST

Cisco IOS Release 12.2(13)T, 12.0(23)S and later

- Router> enable
- Router# configure terminal
- Router(config)# interface type number
  - Example: Router(config)# interface ethernet 0
- Router(config-if)# ipv6 traffic-filter access-list-name {in | out}
  - Example: Router(config-if)# ipv6 traffic-filter outbound out



- 157

# Example

- IPv6 network
- Create and apply four ACL IPv6:
  - Router(config)# ipv6 access-list inboundN
  - Router(config-ipv6-acl)#
    - deny IPv6 any {host1 destination-ipv6-address}
    - deny IPv6 any {host2 destination-ipv6-address}
    - deny udp any {host destination-ipv6-address} eq 80 log-input (HTTP)
    - deny tcp any {host destination-ipv6-address} eq 20 log-input (FTP)
  - Router(config)# interface ethernet X (Interface toward Router 1)
  - Router(config-if)# ipv6 traffic-filter inboundN in



## Part 9

# SNMP over IPv6



**ConsultInTel**

Consultores Integrales en Telecomunicaciones

- 159

# Configure SNMP over IPv6

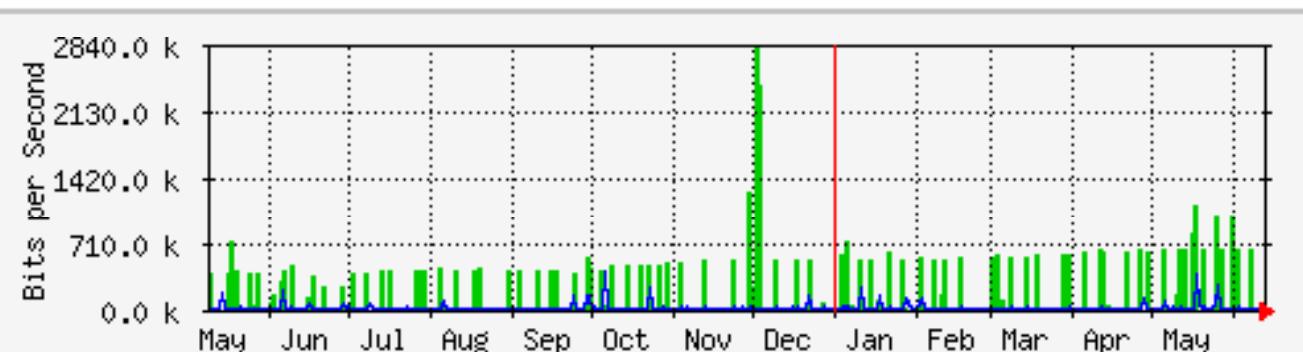
- Router> enable
- Router# configure terminal
- Router(config)# snmp-server community string [view view-name] [ro | rw] [ipv6 nacl] [access-list-number]
  - Example: Router(config)# snmp-server community mgr view restricted rw ipv6
- Router(config)# snmp-server engineID remote { ipv4-ip-address | ipv6-address } [udp-port udp-port-number] [vrf vrf-name] engineid-string (Opcional: Especifica el nombre de una SNMP engine remota)
  - Example: Router(config)# snmp-server engineID remote 2001:DB8:C18:1::3/126 remotev6
- Router(config)# snmp-server group group-name {v1 | v2c | v3 { auth | noauth | priv } } [read read-view] [write write-view] [notify notify-view] [context context-name] [access [ipv6 named-access-list]{ acl-number | acl-name }] (Opcional: Configura un nuevo grupo SNMP, o una tabla que mapea usuarios SNMP a vistas SNMP)
  - Example: Router(config)# snmp-server group public v2c access ipv6 public2
- Router(config)# snmp-server host { hostname | ip-address } [traps | informs] [version {1 | 2c | 3 [auth | noauth | priv]}] community-string [udp-port port] [ notification-type ] [vrf vrf-name]
  - Example: Router(config)# snmp-server host host1.com 2c vrf trap-vrf
- Router(config)# snmp-server user username group-name [remote host [udp-port port]]
- {v1 | v2c | v3 [encrypted] [auth {md5 | sha} auth-password]} [access [ipv6 nacl] { acl-number | acl-name }] (Opcional: Configura un usuario nuevo para un grupo SNMP existente)
  - Example: Router(config)# snmp-server user user1 bldg1 remote 2001:DB8:C18:1::3/126 v2c access ipv6 public2
- Router(config)# snmp-server enable traps [ notification-type [ notification-options]]
  - Example: Router(config)# snmp-server enable traps



# Exercise

- IPv6 configured on the network
- Configure the router to accept SNMP requests over IPv6 from host1
- Configure in host1 both an SNMP agent and a graphic tool (MRTG or another)
- Check the IPv6 traffic on interfaces

'Yearly' Graph (1 Day Average)



Max In:2825.5 kb/s (2.8%)    Average In:135.7 kb/s (0.1%)    Current In:9184.0 b/s (0.0%)

Max Out:402.9 kb/s (0.4%)    Average Out:21.8 kb/s (0.0%)    Current Out:8576.0 b/s (0.0%)



# References (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] RFC4380
- [TEREDOC]  
<http://www.microsoft.com/technet/prodtechnol/winxppro/maintain/teredo.mspx>
- [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



# References (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] RFC4862
- [STATEFULL] RFC3315
- [PRIVACY] RFC4941
- 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/cg\\_0902.mspx](http://www.microsoft.com/technet/community/columns/cableguy/cg_0902.mspx).



# Thanks !

## Contact:

- Jordi Palet Martínez (Consulintel): [jordi.palet@consulintel.es](mailto:jordi.palet@consulintel.es)

## The IPv6 Portal:

- <http://www.ipv6tf.org>



**Consul*in*Tel**

Consultores Integrales en Telecomunicaciones

- 164