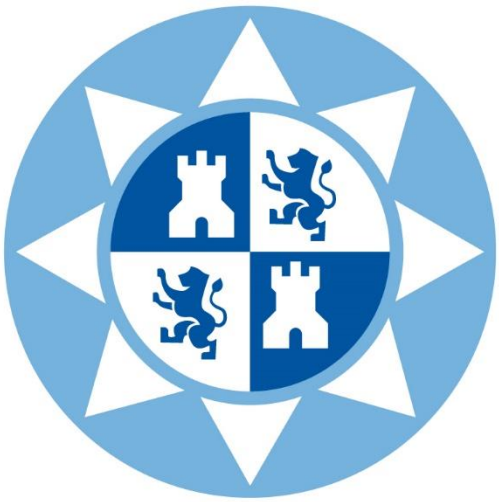
**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN**

**UNIVERSIDAD POLITÉCNICA DE CARTAGENA**

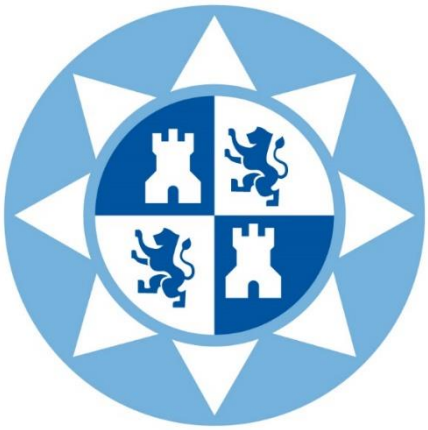


**Trabajo Fin de Grado**

# Estudio e implementación de casos de uso de IPv6/IPv4 en redes de empresa.


AUTOR: Jose Carlos Sáez Agüera

TUTOR: Juan Carlos Sánchez Aarnoutse



| | |
|---|---|
| **Autor** | Jose Carlos Sáez Agüera |
| **E-mail del autor** | jcsaguera@gmail.com |
| **Director(es)** | Juan Carlos Sánchez Aarnouste |
| **Título del TFG** | Estudio e implementación de casos de uso de IPv6/IPv4 en redes de empresa. |


**Resumen:**

**En este proyecto se va a realizar un estudio teórico y práctico sobre el protocolo de nivel de red Ipv6 intentando ahondar en las novedades que trae consigo y en su posible aplicación a una red de empresa. Evidentemente, a fecha de hoy es impensable que IPv6 desplace completamente a IPv4, por lo que entre los casos de uso debe estudiarse los casos en los que deben convivir ambos protocolos. En el desarrollo del proyecto se debe abordará tanto la configuración de los equipos físicos, como la adaptación de las versiones de protocolos ya existentes, intentando ver qué diferencias tiene con su predecesor. Asimismo, se verá qué vías existen para una adaptación de las redes ya existentes a Ipv6 mientras se produce la transición definitiva, y se estudiará cómo se han adaptado (o no) algunos programas y servicios dirigidos al usuario final.**


| | |
|---|---|
| **Titulación** | Grado en Ingeniería Telemática |
| **Departamento** | Tecnologías de la Información y las Comunicaciones |
| **Fecha de Presentación** | 16 de Octubre del 2019 |

# Índice

# Capítulo 1. Introducción (Teórico)

## 1.1. Descripción del trabajo y sus objetivos

Este trabajo consta de un estudio teórico y práctico sobre el protocolo de nivel de red Ipv6 donde se verá qué novedades trae consigo, qué se debe hacer para configurar tanto equipos como enrutadores para adaptarlos a él, cómo cambian las versiones de protocolos ya existentes como RIP u OSPF y qué diferencias tiene con su predecesor.

Se verá qué vías existen para una adaptación de las redes ya existentes a Ipv6 mientras se produce la transición definitiva y se estudiará cómo se han adaptado (o no) algunos programas y servicios dirigidos al usuario final.

Además, se indagará sobre herramientas que permitan controlar el tráfico (como PSVT) o técnicas de denegación de servicios. Igualmente se explorará cómo introducir elementos de IoT tanto de forma teórica como práctica en redes de pequeño tamaño.

A lo largo de este proyecto se mostrarán tres formas de abordar cada caso de estudio.

Normalmente se comenzará con una parte de documentación destinada a tener una base teórica de las funcionalidades, fortalezas, debilidades y funcionamiento del protocolo o servicio a estudiar.

Después se llevará a cabo normalmente una maqueta sobre el simulador de un caso de estudio que nos permitirá ver más a fondo el funcionamiento del mismo.

Para terminar, en caso de que los instrumentos a nuestra disposición nos lo permitan, se realizará un estudio sobre un caso práctico, con la intención de complementar el estudio teórico.

## 1.2 Estructura del proyecto

Una vez queda claro los objetivos del proyecto, toca hablar de cómo se ha estructurado.

En el capítulo dos se especifica qué programas y herramientas se han usado en la realización de las maquetas teóricas, así como qué equipos de red se ha trabajado y porqué, además de un resumen de los trabajos teóricos sobre simulador y prácticos en laboratorio que se han llevado a cabo.

El capítulo tres reúne la documentación necesaria entender el funcionamiento básico de Ipv6.

Durante los capítulos cuatro y cinco se expone el desarrollo de protocolos propios de Ipv6, redes funcionales en Ipv6 con elementos de red añadidos no vistos en los contenidos de las asignaturas del Grado cursado, y una serie de escenarios adaptando los dos protocolos de red.

En los capítulos seis y siete se exponen ejemplos teóricos que amplían un poco más el manejo de las herramientas usadas en clase (IoT en simulador) así como de complejidad en la red (PSVT para el control de tráfico) y una serie de estudios sobre cómo se adaptan (o no) servicios dedicados al usuario final como VOIP o escritorio remoto a IPV6.

Finalmente se presentará una breve conclusión sobre el estado de Ipv6 y posibles formas de ampliar el estudio sobre este protocolo que se ha realizado en este proyecto.

# Capítulo 2. Entorno de trabajo

## 2.1. Herramientas usadas en el proyecto

**Cisco Packet Tracer** [1]de Cisco es un programa de simulación de redes que permite a los estudiantes experimentar con el comportamiento de la red.

Aunque utiliza una versión actual del Cisco IOS, sólo hace uso de un pequeño número de características encontradas en el hardware, por lo que no es adecuado para redes en producción.

Con este programa se crea la topología física de la red simplemente arrastrando los dispositivos a la pantalla. Luego, haciendo clic sobre ellos se puede ingresar a sus consolas de configuración. Una vez completada la configuración física y lógica de la red, también se puede hacer simulaciones de conectividad (pings, traceroutes) todo ello desde las mismas consolas incluidas.

Una de las grandes ventajas de utilizar este programa es que permite "ver" (opción "*Simulation*") cómo deambulan los paquetes por los diferentes equipos (switchs, routers, PCs), además de poder analizar de forma rápida el contenido de cada uno de ellos en las diferentes "capas" y "datos".

**RouterOS**[2] es un sistema operativo basado en el núcleo Linux, el cual implementa funcionalidades que los NSP (*Network Service Provider*) e ISP tienden a implementar, como por ejemplo BGP, Ipv6, OSPF o MPLS.

RouterOS es un sistema versátil, con un gran soporte por parte de MikroTik[3], tanto a través de un foro como de su Sitio Wiki, proporcionando una amplia variedad de ejemplos de configuración.

La venta de RouterOS, combinado con su línea de productos de hardware conocida como MikroTik RouterBOARD, está enfocada a los pequeños y medianos proveedores de acceso a Internet, que normalmente proporcionan acceso de banda ancha en áreas remotas.

[1] Cisco Packet Tracer - https://www.netacad.com/es/courses/packet-tracer
[2] RouterOS - https://mikrotik.com/download
[3] Mikrotik - https://mikrotik.com/software

En este trabajo también se hará uso de los servicios de **Hurricane Electric,**.concretamente se hará uso de su servicio de tunnel brokers. Este servicio ofrece túneles Ipv6 a webs o usuarios mediante IPv4. En general, los tunnel brokers Ipv6 ofrecen lo que se llaman túneles 'protocolo 41' o proto-41. Estos son túneles donde Ipv6 se tunela directamente en paquetes IPv4, fijando el campo de protocolo en '41' (Ipv6) en el paquete IPv4.

Se empleará **Wireshark**[4] en determinadas ocasiones para comprobar qué falla cuando se desarrolla alguna maqueta en el laboratorio.

Wireshark, antes conocido como Ethereal, es un analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de comunicaciones, para desarrollo de software y protocolos, y como una herramienta didáctica. Permite examinar datos o de un archivo de captura salvado en disco.

En él se puede analizar la información capturada, a través de los detalles y sumarios por cada paquete. Wireshark incluye un completo lenguaje que permite filtrar lo que se desee ver. Además dispone de una funcionalidad que permite mostrar el flujo reconstruido de una sesión de TCP.

De la misma forma, se realizará la configuración de programas conocidos para acceso remoto a escritorio, VOIP o control de dispositivos IP. En el caso de que sea posible, se hará una adaptación a una red Ipv6 Only.

[4] Wireshark - https://www.wireshark.org

## 2.2. Equipos

Para las pruebas se han usado los siguientes switches, routers y equipos para monitorizar la red, habiendo comprobado el funcionamiento de los protocolos que implementan.

A continuación, se listan sus principales características y su uso en el proyecto:

### Switch Netgear GS716t

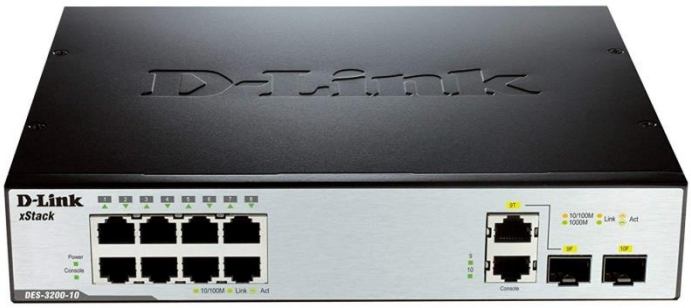
Es el switch usado en la primera instancia, donde se han probado tanto servicios como configuraciones en Ipv4. En las maquetas recreadas en el laboratorio se emplearán como switches de acceso en redes pequeñas. Igualmente se utilizarán cuando no sea necesario usar los los servicios extra que aportan los switches D-Link.

Éstas son las características más importantes:

- 16 puertos 10/100/1000
- 2 Puertos SPF 10/100/1000
- Standards de red: IEEE 802.3, IEEE 802.3ab, IEEE8
- 02.3u, IEEE 802.3x, IEEE 802.3z
- Incorpora opciones de DHCPv4, web management, Spanning tree protocol, Jumbo frames support, Port mirroring
- Switch de capas 2 y 3 gestionable con capacidad para agregación de enlaces
- Capaz de almacenar hasta 8000 direcciones MAC
- Capacidad de 32 Gb/s
- Soporta Vlan con aplicaciones externa

## Switch D-Link DES-3200-10

Switch utilizado para configuración avanzada de Ipv6, Vlan en Ipv6, ACL, redes con protocolo RADIUS en ella.




Éstas son las características más importantes:

- 8 puertos 10/100 BASE-TX
- 1 puerto 100/1000 SPF
- 1 puerto 10/100/1000 BASE-T/100/1000 SPF
- Tabla de direcciones MAC de 16 Kb
- Buffer de paquetes de 1.5MB
- Capacidad de 5.6 Gbps
- Switch de capas 2 y 3
- Ratio de reenvío de paquetes de 64Bytes de 4.2Mpps
- Soporta Ipv6

## Switch D-Link DGS-3200-10


Éstas son las características más importantes:

- 8 puertos 10/100 BASE-TX
- 2 puerto 1000 BASE-T/SPF
- 1 puerto RS-232 para consola
- Buffer de paquetes de 128Kbytes
- Capacidad de 20 Gbps
- Tabla de direcciones MAC de 8 Kb
- Switch de capas 2 y 3
- Ratio de reenvío de paquetes de 64Bytes de 14.88 Mbps
- Soporta Ipv6

Los Pcs que se han usado en el laboratorio a modo de usuario donde probar los servicios tienen las siguientes propiedades:

- Intel Core i5-4460 a 2x3.2Ghz
- RAM de 8 Gb
- Sistema Operativo Windows 10 pro
- Arquitectura de 64 bits
- Tarjeta de red con controlador Realtek PCIe GBE Family
- ¿Sistema OPERATIVO?

## Router/AP mikrotik RB941-2ND-TC

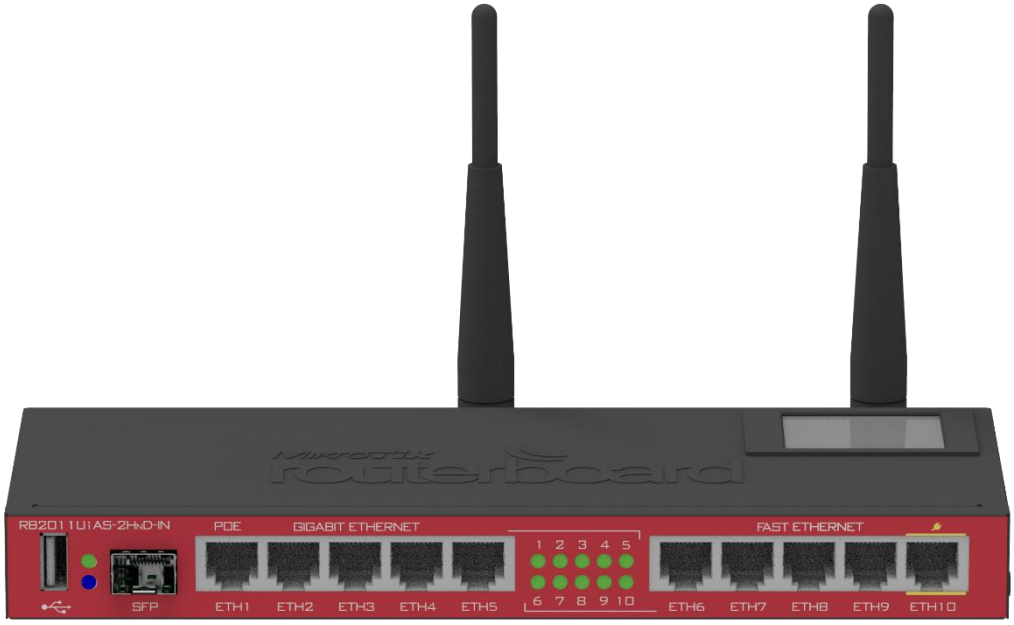
Router mikrotik adquirido con la intención de que soporte todos los requerimientos de routing Ipv6, además de darnos algunos servicios seguridad añadidos y con intención de probar esta marca que se está haciendo un gran hueco en el mercado.


Éstas son las características más importantes:

- 3 puertos 10/100
- Velocidad máxima de transferencia 300 Mbit/s
- Estándares de red IE802.11b, IEE802.11g, IEEE 802.11n
- Puerto Micro-USB
- Seguridad por pulsador WPS
- 2 Antenas integradas
- Ganancia de la antena de 1.5dBi
- Sistema operativo RouterOS
- Paquete para soporte Ipv6

## MikroTik RouterBOARD RB2011UiAS-IN

Router mikrotik disponible en el laboratorio para investigación. Será el router que se usará normalmente para redes con Ipv6 en el laboratorio.




- 5 puertos 10/100
- 5 puertos 10/100/1000
- Velocidad máxima de transferencia 300 Mbit/s
- Estándares de red IE802.11b, IEE802.11g, IEEE 802.11n
- Puerto Micro-USB
- Memoria Ram 128MB
- 2 Antenas integradas
- Puerto serie RJ45
- Sistema operativo RouterOS
- Paquete para soporte Ipv6
- Consumo máximo 15W

## Raspberry Pi 2 model B

Usada para la configuración de un servidor de Kerberos.io para el control de cámaras ip conectadas a ella.


- A 900MHz quad-core ARM Cortex-A7 CPU1GB RAM10
- 0 Base Ethernet
- 4 USB ports
- 40 GPIO pins
- Full HDMI port
- Combined 3.5mm audio jack and composite video
- Camera interface (CSI)
- Display interface (DSI)
- Micro SD card slot
- VideoCore IV 3D graphics core

## 2.3. Maquetas teóricas y prácticas desarrolladas

Para el desarrollo de este proyecto se han realizado una serie de maquetas en el simulador Cisco Packet Tracer con el fin de contrastar el trabajo de investigación realizado sobre determinados protocolos, interfaces y servicios de red.

Una vez realizado el trabajo de investigación y configurada la maqueta en el simulador, se intentaba realizar un montaje en el laboratorio con los equipos disponibles.

Hay que recalcar que existen protocolos y/o servicios que debido a los equipos de los que se disponía, o debido a las limitaciones de los sistemas operativos de los router y/o switches no se han podido llevar a cabo en el laboratorio.

Las maquetas que se han realizado sobre el simulador son las siguientes:

- Maqueta de red pequeña en Ipv6 (varias redes Ipv6 con comunicación entre sí)
- Maqueta sobre DHCP sin estado
- Maqueta sobre DHCP con estado
- Maqueta sobre ACL y filtrado de tráfico en una red Ipv6
- Maqueta con una red DMZ en Ipv6
- Maqueta sobre el protocolo dinámico RIPng (versión para Ipv6)
- Maqueta sobre el protocolo dinámico OSPFv3(versión para Ipv6)
- Maqueta sobre el mecanismo de adaptación a ipv4-Ipv6 túnel 6to4
- Maqueta sobre el mecanismo de adaptación a ip4-Ipv6 NATPT
- Maqueta sobre IoT
- Maqueta sobre el protocolo PSVT (Per VLAN Spanning Tree)

Como se indicó anteriormente, se ha respaldado este trabajo en el simulador *Packet Tracer* con algunos desarrollos en el laboratorio sobre

- DHCP sin estado
- DHCP con estado
- SLAAC
- ACL y filtrado de tráfico
- Configuración de varias redes Ipv6 con conexión entre ellas
- Dual-Stack
- Túnel 6to4 ayudado de servicios de Tunnel Brocker
- Configuración de un servidor RADIUS para nuestra red WAN
- Configuración de un servicio Kerberos para el control de cámaras IP

# Capítulo 3. Conceptos básicos de Ipv6

Ipv6 es un protocolo que se encuadra en la capa 3 del modelo OSI. Ya en los años noventa, incluso habiéndose desarrollado el rediseño a redes sin clase, resultaba palpable que en un futuro se agotarían las ips disponibles para Ipv4.

Ipv6 es más que solo direcciones más extensas. Cuando el IETF[5] (Internet Engineering Task Force) comenzó el desarrollo de un sucesor de IPv4, utilizó esta oportunidad para corregir las limitaciones de IPv4 e incluir mejoras adicionales como el protocolo de mensajes de control de Internet (ICMPv6) o la configuración automática sin estado (SLAAC) entro otros muchos.

## 3.1. ¿Por qué usar Ipv6?

A finales de 1992, la IETF anuncio la creación del primer grupo de trabajo para la nueva generación del protocolo IP.

De las cinco RIR (*Regional Internet Registry*) que gestionan la administración de direcciones ip, solo la de Asia [6](cuya fecha prevista de agotamiento es en 2019) posee ips libres. Europa, América, África y Australia agotaron sus direcciones entre 2011 y 2015.

Ipv6 se considera una extensión conservadora de su antecesor ya que los protocolos de capas OSI superiores como los de transporte y aplicación necesitan pocos o ningún cambio para operar sobre Ipv6, a excepción de protocolos de aplicación que integran direcciones de capa de red como FTP o NTP.

Se podría resumir

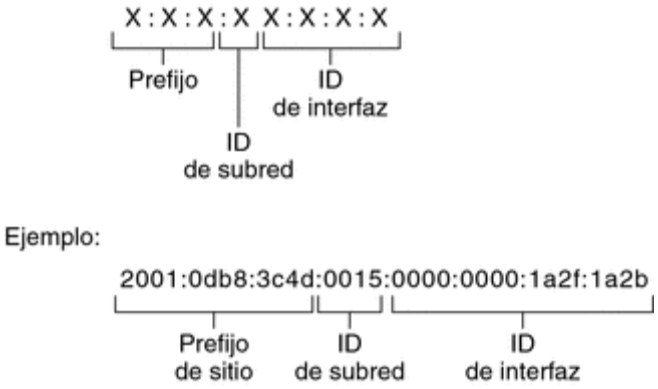
- Porque se han agotado las direcciones IPv4
- Porque Ipv6 ha sido diseñado para ser fácil
- Recupera la conectividad extremo-a-extremo (no existe NAT)

[5] IETF- https://www.ietf.org/
[6] Ultimos bloques de Ips entregados por la IANA - https://twitter.com/theiana

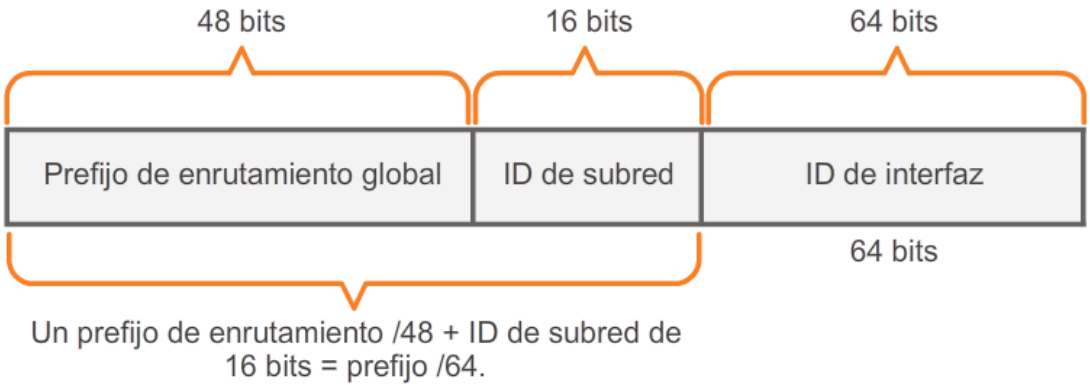
- Porque disponemos de un número casi ilimitado de direcciones
- Porque las direcciones no son dinámicas
- Incrementa la seguridad en Internet, tanto por el uso de Ipsec extremo-a-extremo como por el abundante número de direcciones que dificultan los ataques
- Es más apropiado para sistemas de multidifusión pues aprovecha mejor la capacidad de las redes para servicios de valor añadido de audio y video sobre banda ancha

## 3.2. Funcionamiento básico Ipv6

Las direcciones Ipv6 tienen una longitud de 128 bits y se escriben como una cadena de valores hexadecimales. Cada 4 bits se representan con un único dígito hexadecimal para llegar a un total de 32 valores hexadecimales.




Donde cada “x” consta de cuatro valores hexadecimales, que son a su vez 16 bits. Esto es llamado de forma no oficial “hexteto”.




*Nota- Normalmente, el identificador de interfaz se genera automáticamente a partir de la dirección MAC de la interfaz a la que está asignada la dirección.*

Aunque éste es el formato normal de una dirección Ipv6, existen dos reglas que permiten reducir el número de dígitos necesarios para representar una dirección:

### 1º Regla: Omisión de ceros iniciales

Dentro de cada hexteto, se puede omitir todos los ceros iniciales.

*2041: 0000:134F:0000:0010:09FF:7834:BB00*

Se podría convertir en:

*2041:0:134F:0:10:9FF:7834:BB00*

Esta regla solo es válida para los ceros iniciales, y NO para los ceros finales; de lo contrario, la dirección sería ambigua, pues:

**3B4** podría ser tanto 03B4 como 3B40

### 2º Regla: omitir los segmentos de 0

Esta regla permite reducir, mediante los dos puntos dobles ( :: ) una cadena de uno o más hextetos de compuestos solo por ceros.

De tal forma que la siguiente dirección Ipv6

2001:DB8:0000:1234: **0000:0000:0000**:0401

usando esta regla de omitir los segmentos de 0

2001:DB8:**0000**:1234::**0**401

Que juntándola con la primera regla de ceros iniciales quedaría como:

2001:DB8:0:1234::401

Los dos puntos dobles (::) se pueden utilizar solamente una vez dentro de una dirección; de lo contrario, habría más de una dirección resultante posible.

Cuando una dirección usa ambas decimos que la dirección está en *formato comprimido.*

## 3.3. Tipos de direccionamiento Ipv6

### Unidifusión

Una dirección Ipv6 de unidifusión identifica de manera única una interfaz de un dispositivo habilitado para Ipv6. Existen varios tipos de direcciones de unidifusión, entre ellas:

- **Unidifusión global:** Las direcciones de unidifusión globales son similares a las direcciones IPv4 públicas. Estas son direcciones enrutables de Internet globalmente exclusivas. Las direcciones de unidifusión globales pueden configurarse estáticamente o asignarse de forma dinámica.

- **Link-local:** Las direcciones link-local se utilizan para comunicarse con otros dispositivos en el mismo enlace local. Con Ipv6, el término “enlace” hace referencia a una subred. Las direcciones link-local se limitan a un único enlace. Su exclusividad se debe confirmar solo para ese enlace, ya que no se pueden enrutar más allá del enlace. En otras palabras, los routers no reenvían paquetes con una dirección de origen o de destino link-local. Las direcciones Ipv6 link-local están en el rango de FE80::/10.

- **Local única:** Otro tipo de dirección de unidifusión es la dirección de unidifusión local única. Las direcciones Ipv6 locales únicas tienen ciertas similitudes con las direcciones privadas RFC 1918 para IPv4, pero existen grandes diferencias. Las direcciones locales únicas se utilizan para el direccionamiento local dentro de un sitio o entre una cantidad limitada de sitios. Estas direcciones no deberían poder enrutarse en la Ipv6 global, y

no deberían traducirse hacia direcciones Ipv6 globales. Las direcciones locales únicas están en el rango de FC00::/7 a FDFF::/7.

## Multidifusión

Las direcciones Ipv6 de multidifusión se usan para enviar un único paquete Ipv6 a varios destinos. Las direcciones Ipv6 de multidifusión tienen el prefijo FF00::/8.

Existen dos tipos de direcciones Ipv6 de multidifusión:

**Dirección de multidifusión asignada:** una dirección de multidifusión asignada es una única dirección que se utiliza para llegar a un grupo de dispositivos que ejecutan un protocolo o servicio común, como por ejemplo DHCPv6.

- **FF02::1:** Es un grupo de multidifusión al que se unen *todos los dispositivos* con Ipv6 habilitado. Los paquetes que se envían a esta dirección son procesados por todas las interfaces en el enlace o en la red. *Su efecto es el mismo que el de una dirección de difusión de Ipv4.* Esta dirección es usada, por ejemplo, en protocolos como ICMPv6, que proporcionando a todos los dispositivos de la red información de direccionamiento, longitud de prefijo y Gateway.

- **FF02::2:** Es un grupo de multidifusión al que se unen todos los routers Ipv6 y al que se unen cuando ejecutan el comando *Ipv6 unicast-routing.* Los paquetes que se envían a este grupo son recibidos por todos los routers Ipv6 en el enlace o red.

| Multicast Address | Multicast Group |
|---|---|
| FF01::1 | All IPv6 nodes within the node-local scope |
| FF01::2 | All IPv6 routers within the node-local scope |
| FF02::1 | All IPv6 nodes within the link-local scope |
| FF02::2 | All IPv6 routers within the link-local scope |
| FF02::5 | All OSPFv3 routers within the link-local scope |
| FF02::6 | All OSPFv3 designated routers within the link-local scope |
| FF02::9 | All RIPng routers within the link-local scope |
| FF02::A | All EIGRP routers within the link-local scope |
| FF02::D | All PIM routers within the link-local scope |
| FF02::1:2 | All DHCPv6 agents (servers and relays) within the link-local scope |
| FF05::2 | All IPv6 routers within the site-local scope |
| FF02::1:FF00:0/104 | IPv6 solicited-node multicast address within the link-local scope |

La dirección más alta de la red (la dirección de broadcast en una red IPv4) es considerada una dirección normal en Ipv6´, al contrario que en Ipv4.

### Difusión por proximidad

Una dirección Ipv6 de difusión por proximidad es cualquier dirección Ipv6 de unidifusión que puede asignarse a varios dispositivos. Un paquete enviado a una dirección de difusión por proximidad se entrega únicamente a una de las interfaces identificadas con esa dirección. Esta interfaz coincide con la de menor coste según la definición de métrica de encaminamiento. Este hecho permite equilibrar la carga entre distintos nodos. Las direcciones de difusión por proximidad sólo se utilizan como direcciones de destino y sólo se asignan a encaminadores.

## Direcciones especiales:

- **Dirección no especificada:** Equivale a la dirección 0.0.0.0 de IPv4. La dirección no especificada (0:0:0:0:0:0:0:0 ó ::) sólo se utiliza para indicar la ausencia de dirección. La dirección no especificada nunca se asigna a una interfaz ni se utiliza como dirección de destino. Esta dirección se suele utilizar durante el proceso de asignación automática de direcciones como dirección de origen en paquetes que intentan comprobar si una determinada dirección está ya asignada a un equipo de la red.

- **Dirección de loopback**: fe80::/10. Equivale a la dirección 127.0.0.1 de IPv4. La dirección de loopback (0:0:0:0:0:0:0:1 ó ::1) permite que un nodo se envíe paquetes a sí mismo. Los paquetes dirigidos a la dirección de loopback no salen fuera de la máquina.

- **Direcciones de compatibilidad:** Para facilitar la migración de IPv4 a Ipv6 y la coexistencia de nodos de ambos tipos en Internet, se han definido las siguientes direcciones:
    - Direcciones compatibles con IPv4 (obsoleta según la especificación RFC 4291). Estas direcciones son utilizadas por los nodos con protocolos IPv4 e Ipv6 que se comunican con Ipv6 a través de IPv4. Cuando la dirección compatible con IPv4 se utiliza como destino Ipv6, el tráfico Ipv6 se encapsula de forma automática con un encabezado IPv4 y se envía al destino utilizando el protocolo IPv4.
    - Direcciones asignadas o mapeadas a IPv4. Estas direcciones se usan para representar ante un nodo Ipv6 un nodo que utiliza sólo protocolo IPv4. Sólo sirve para la representación interna. La dirección asignada a IPv4 nunca se utiliza como dirección de origen o destino de un paquete Ipv6. El protocolo Ipv6 no admite el uso de direcciones asignadas a IPv4.
    - Direcciones 6to4 Las direcciones 6to4 se utilizan para la comunicación entre dos nodos que ejecutan IPv4 e Ipv6.

- **Direcciones reservadas:** La IANA[7] (*Internet Assigned Numbers Authority*) ha reservado un bloque de direcciones llamado Sub-TLA ID que son 64 prefijos de red desde 2001:0000::/29 hasta 2001:01f8::/29, asignando los bloques en tres partes:
  - 2001::/32. Utilizado por el protocolo de túneles Teredo (también usado como mecanismo de transición Ipv6).
  - 2001:2::/48. Asignado a *Benchmarking Methodology Working Group* (BMWG) para realizar comparativas y test en Ipv6 (similar a la red 198.18.0.0/15 de IPv4).
  - 2001:10::/28 ORCHID (*Overlay Routable Cryptographic Hash Identifiers*) son direcciones Ipv6 no enrutables que se utilizan para identificadores criptográficos de hash.
- **Documentación:** 2001:db8::/32. Es un prefijo reservado para documentación. Estas direcciones se deben usar siempre que se quiera escribir un ejemplo de dirección Ipv6 o se creen modelos de red (similar a las redes 192.0.2.0/24, 198.51.100.0/24 y 203.0.113.0/24 en Ipv6

## 3.4. Cabecera Ipv6

Un paquete en Ipv6 está compuesto principalmente de dos partes: la cabecera (que tiene una parte fija y otra con las opciones) y la carga útil (los datos) y a su vez el tamaño de la cabecera está formado por una parte fija de 40 bytes (320 bits) y una variable.

### ➢ Cabecera fija

Dentro de la cabecera, en su formato fijo tenemos los siguientes campos, los cuales aparecen siempre en un mensaje Ipv6:

- Direcciones de origen (128 bits)
- Direcciones de destino (128 bits)
- Versión del protocolo IP (4 bits)
- Clase de tráfico (8 bits, Prioridad del Paquete)
- Etiqueta de flujo (20 bits, manejo de la Calidad de Servicio)

[7] IANA - https://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.xhtml

- Longitud del campo de datos (16 bits)
- Cabecera siguiente (8 bits)
- Límite de saltos (8 bit). También llamado “tiempo de vida”(TTL)


*En Ipv6 la fragmentación se realiza solamente en el nodo origen del paquete*, al contrario que en IPv4 en donde los routers pueden fragmentar un paquete. En Ipv6, las opciones también desaparecen de la cabecera estándar y son especificadas por el campo "Cabecera Siguiente" (*Next Header*), similar en funcionalidad en IPv4 al campo Protocolo.

### ➢ Cabeceras de extensión

El uso de un formato flexible de cabeceras de extensión opcionales es una idea innovadora que permite ir añadiendo funcionalidades de forma paulatina. Este diseño aporta gran eficacia y flexibilidad ya que se pueden definir en cualquier momento a medida que se vayan necesitando entre la cabecera fija y la carga útil.

La cabecera fija y las de extensión opcional incluyen el *campo de cabecera siguiente* que identifica el tipo de cabeceras de extensión que viene a continuación o el identificador del protocolo de nivel superior. Las cabeceras de extensión se van encadenando utilizando el campo de cabecera siguiente que aparece tanto en la cabecera fija como en cada una de las citadas cabeceras de extensión. Como resultado de la secuencia anterior, *dichas*

*cabeceras de extensión se tienen que procesar en el mismo orden* en el que aparecen en la siguiente figura.

| Cabecera de Extensión | Tipo | Tamaño | Descripción |
|---|---|---|---|
| Opciones salto a salto (*Hop-By-Hop Options*) | 0 | variable | Contiene datos que deben ser examinados por cada nodo a través de la ruta de envío de un paquete. |
| Enrutamiento (*Routing*) | 43 | variable | Métodos para especificar la forma de rutear un datagrama. (Usado con Ipv6 móvil). |
| Cabecera de fragmentación (*Fragment*) | 44 | 64 bits | Contiene parámetros para la fragmentación de los datagramas. |
| Cabecera de autenticación (*Authentication Header* (AH)) | 51 | variable | Contiene información para verificar la autenticación de la mayor parte de los datos del paquete. |
| Encapsulado de seguridad de la carga útil (*Encapsulating Security Payload* (ESP)) | 50 | variable | Lleva la información cifrada para comunicación segura |
| Opciones para el destino (*Destination Options*) | 60 | variable | Información que necesita ser examinada solamente por los nodos de destino del paquete. |
| *No Next Header* | 59 | vacío | Indica que no hay más cabeceras. |

Por último y para dejar de explicar los campos de la cabecera, hay que decir que la carga útil dispone de 64 KB en modo estándar, pudiendo aumentar con la opción de carga jumbo (jumbo payload) hasta los 4Gb

# Capítulo 4. Maquetas

## 4.1. Servicio DHCPv6 con estado, sin estado y funcionamiento de SLAAC (Simulador y laboratorio)

### SLAAC.

SLAAC (*Stateless Address AutoConfiguration)* es un método en el cual un dispositivo puede obtener una dirección Ipv6 de unidifusión global sin los servicios de un servidor de DHCPv6. SLAAC utiliza mensajes de solicitud y de anuncio de router ICMPv6 para proporcionar direccionamiento y otra información de configuración que normalmente proporcionaría un servidor de DHCP.

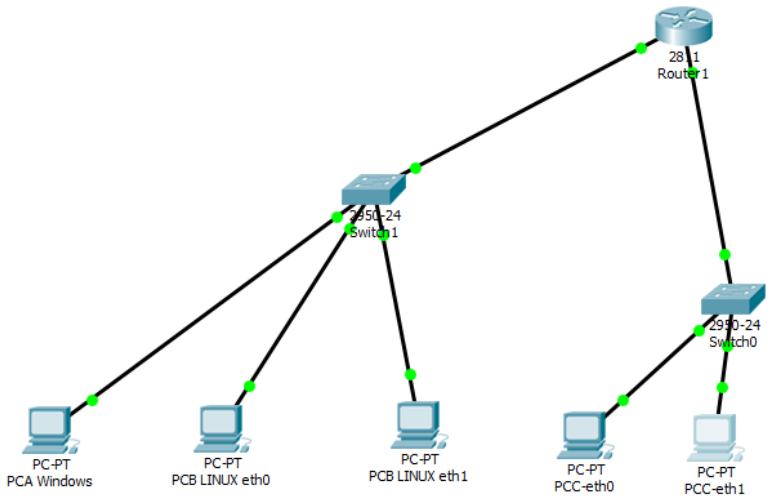



**Figura 4.1- Maqueta "SLAAC Ipv6" donde se configura el router y los Pcs para usar ese mecanismo**

**Mensaje de solicitud de router (RS):** cuando un cliente está configurado para obtener la información de direccionamiento de forma automática mediante SLAAC, el cliente envía un mensaje RS al router. El mensaje RS se envía a la dirección Ipv6 de multidifusión de todos los routers, FF02::2.

**Mensaje de anuncio de router (RA):** los routers envían mensajes RA para proporcionar información de direccionamiento a los clientes configurados para obtener sus direcciones Ipv6 de forma automática. El mensaje RA incluye el prefijo y la longitud de prefijo del segmento local. Un cliente utiliza esta información para crear su propia dirección Ipv6 de unidifusión global. Los routers envían mensajes RA de forma periódica o en respuesta a un mensaje RS. Los mensajes RA siempre se envían a la dirección Ipv6 de multidifusión de todos los nodos, FF02::1.

Como lo indica el nombre, SLAAC quiere decir “sin estado”. Un servicio sin estado significa que no hay ningún servidor que mantenga la información de la dirección de red. A diferencia de DHCP, no hay servidor de SLAAC que tenga información acerca de cuáles son las direcciones Ipv6 que están en uso y cuáles son las que se encuentran disponibles.

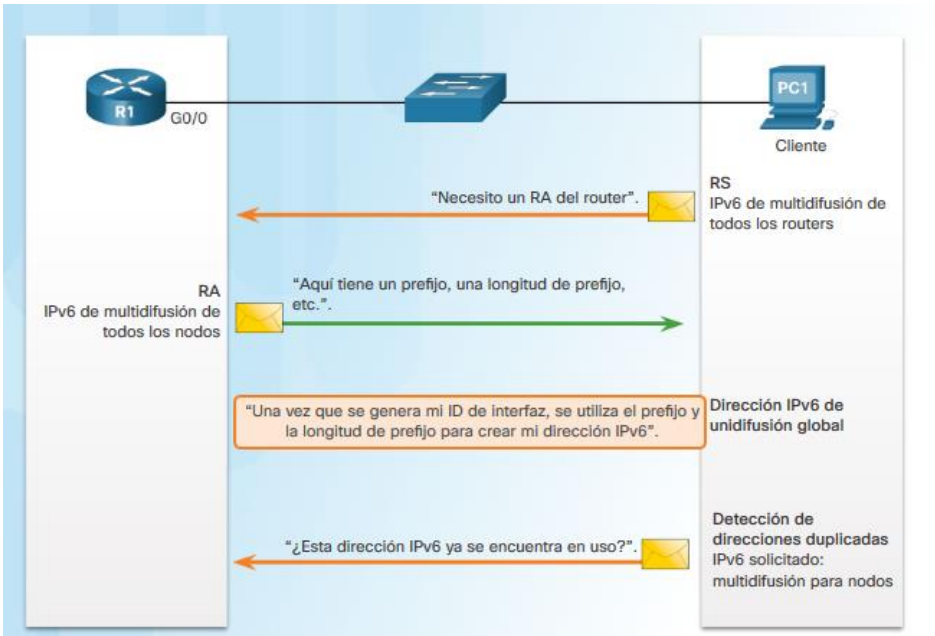



**Figura4.2- Envio de mensajes entre PCs y Routers cuando se usa SLAAC.**

En este ejemplo de la figura 4.1, en la red de la izquierda, se tiene la interfaz de router donde se dirigirán los mensajes RS y desde donde se envían los mensajes RA con dirección 2001:db8:a000:0::50 /64. Eso servirá para que en el mensaje RA se especifique que la red es la 2001:db8: a000:0 y la red es de tamaño de 64 bits.

```
Router>enable
Router#show ipv6 interface brief
GigabitEthernet0/0         [up/up]
    FE80::202:16FF:FE0C:A401
    2001:DB8:A000::50
GigabitEthernet0/1         [up/up]
    FE80::202:16FF:FE0C:A402
    2001:DB8:B000::50
Vlan1                      [administratively down/down]
```

**Figura 4.3- Resultado del comando "show ipv6 interfaces Brief" en el router.**

Como se observa en la figura 4.4, la Ip que ha recibido el PCB, en el que se ha usado la opción de autoconfiguración, se divide en:

1) La porción de red, recibida del mensaje RS al router, coincide con la porción de red de la interfaz de red del router en su red.

2) La porción de ID, en este caso, se consigue mediante el proceso EUI-64, utilizando su dirección MAC de 48 bits. Esta parte podría haber sido configurada para que se generara a través de un generador de números aleatorios.

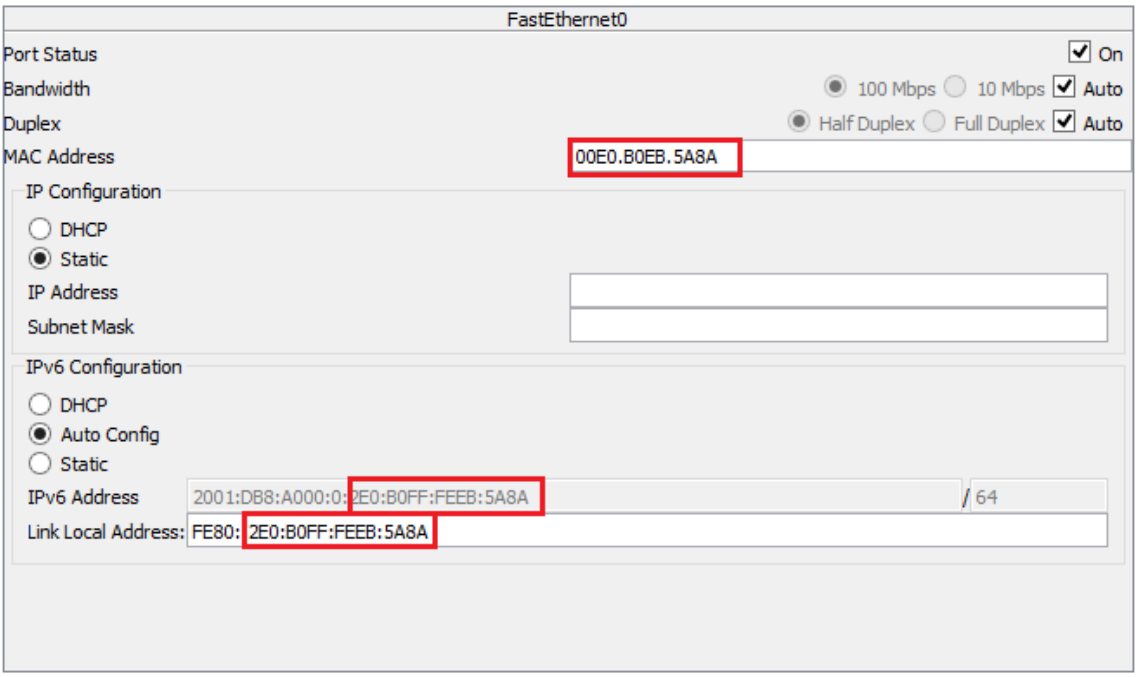



**Figura 4.4- Resultado de usar la auto configuración (SLAAC) después de configurar el router.**

## DHCPv6 con estado

En este caso, el mensaje RA le informa al cliente que no utilice la información contenida en el mensaje RA. Toda la información de direccionamiento y de configuración debe obtenerse de un servidor de DHCPv6 con estado. Esto se conoce como DHCPv6 con estado, debido a que el servidor de DHCPv6 mantiene información de estado de Ipv6.

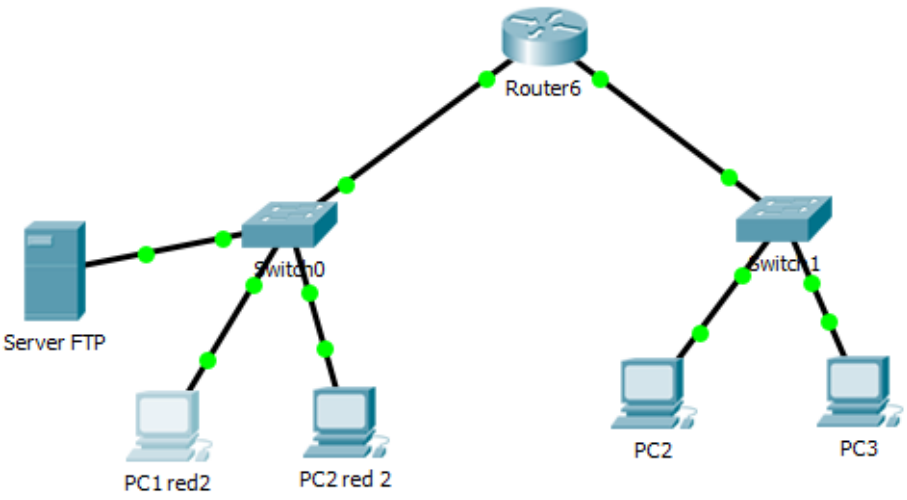



**Figura 4.5- Maqueta "DHCPv6 con estado" donde se configura el router para hacer de DHCPv6.**

El indicador M debe cambiarse de 0 a 1 mediante el comando de interfaz **Ipv6 nd managed-config-flag.** Esto le informa al dispositivo que no utilice SLAAC, sino que obtenga el direccionamiento Ipv6 y todos los parámetros de configuración de un servidor de DHCPv6 con estado.

Con la introducción de las versiones 7.x de Cisco Packet Tracer, hay que tener cierto cuidado a la hora de configurar un servidor DHCPv6 Statefull:

- Se eliminó el comando address prefix por lo que las configuraciones de versiones más antiguas no funcionaran a partir de la versión 7.1.
- En esta versión se usa el comando prefix-delegation para nombrar una pool a la que luego se le dice el rango de direcciones que la suministra.
- Hay que configurar la dirección Ipv6 antes de asignarle un rango de direcciones a la pool. Si no se hace así puede dar error.
- No se debe olvidar los comandos Ipv6 unicast-routing ni Ipv6 nd managed-config-flag

Ejemplo de código para una de las dos interfaces:

```
Ipv6 unicast-routing
interface g0/1
Ipv6 address 2001:A:A:A::1/64
!
Ipv6 dhcp pool ITELAR1
prefix-delegation pool prefijoRed1
dns-serverAAAA:BBBB:CCCC:DDDD::FFFF
domain-name cisco.com
!
Ipv6 local pool prefijoRed1 2001:A:A:A::/48 64
```

!

Int g0/1

duplex auto

speed auto

Ipv6 address FE80::1 link-local

Ipv6 nd managed-config-flag

Ipv6 dhcp server ITELAR1

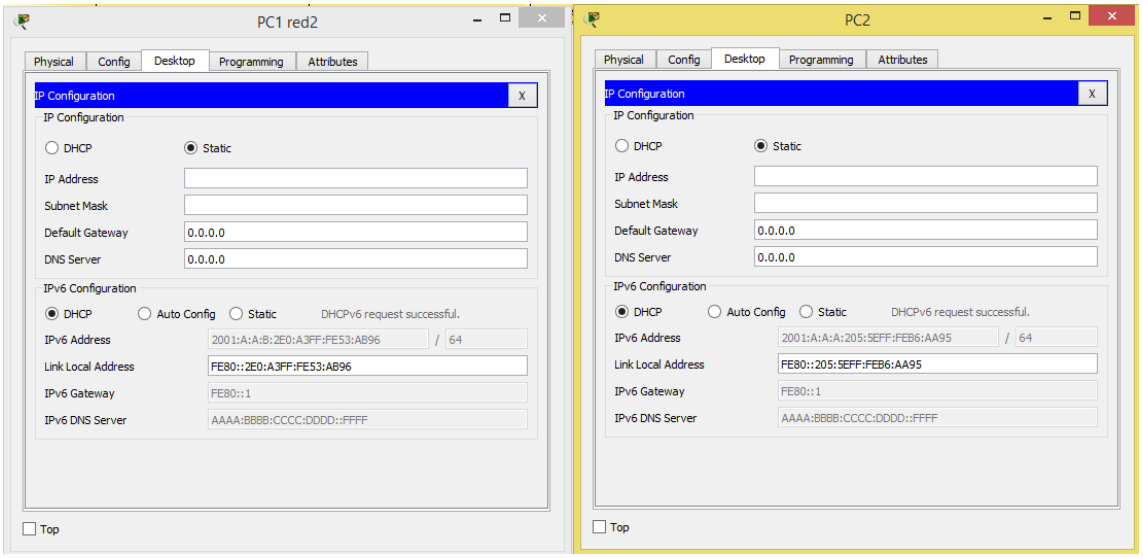



**Figura 4.6- Resultado en los PCs de diferentes redes al usar la opción DHCP una vez configurado el router.**

## DHCPv6 sin estado

La opción de DHCPv6 sin estado informa al cliente que utilice la información del mensaje RA para el direccionamiento, pero que hay más parámetros de configuración disponibles de un servidor de DHCPv6.

Mediante el prefijo y la longitud de prefijo en el mensaje RA, junto con EUI-64 o una ID generada aleatoriamente, el cliente crea la dirección Ipv6 de unidifusión global.

A continuación, el cliente se comunica con un servidor de DHCPv6 sin estado para obtener información adicional que no se proporciona en el mensaje RA. Puede tratarse de una lista de direcciones Ipv6 del servidor DNS, por ejemplo. Este proceso se conoce como DHCPv6 sin estado, debido a que el servidor no

mantiene información de estado del cliente (es decir, una lista de direcciones Ipv6 asignadas y disponibles). El servidor de DHCPv6 sin estado solo proporciona parámetros de configuración para los clientes, no direcciones Ipv6.

Para DHCPv6 sin estado, el indicador O se configura en 1 y el indicador M se deja en la configuración predeterminada de 0. El valor 1 del indicador O se utiliza para informarle al cliente que hay información de configuración adicional disponible de un servidor de DHCPv6 sin estado.

Router(config-if)# **Ipv6 nd other-config-flag**

*LA CONFIGURACIÓN DE UN DHCPv6 STATELESS ES IGUAL QUE UNA CONFIGUARIÓN STATEFULL SALVO QUE NO SE APORTA UN RANGO ESPECIFICO DE IPS Y QUE SE CAMBIA EL COMANDO "nd managed-config-flag" por "Ipv6 nd other-config-flag".*

**Una vez realizadas las diversas maquetas en el simulador, se pasará a ver como se realizan estas configuraciones en el laboratorio.**

Lo primero que se hará para crear un servidor DHCPv6 será crear un pool de direcciones.

1) Acceder al menú Ipv6
2) Añadir el nombre del pool, el prefijo de red que tendrán las direcciones seguidas del tamaño de la máscara de las mismas, y el tamaño del prefijo del pool de direcciones.

NOTA- Estos dos últimos puedes ser distintos, pues puede haber recibido un rango de direcciones a repartir de /48 pero se desea que el server reparta direcciones de un tamaño /64.

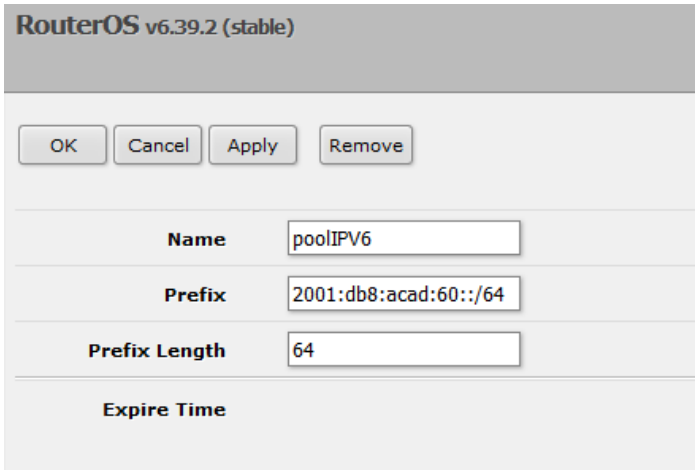



**Figura 4.7- Configuración del sub-menú DHCPv6 Pool.**

**3)** Dentro del menú de Ipv6 se accede al sub-menú DHCPserver.

**4)** Allí se marcará la casilla “enable”, asignando un nombre al servidor, eligiendo la interfaz destino para la que se monta el servidor, en este caso la Vlan 60, y dentro del despegable se especificará que las direcciones Ipv6 se obtendrán de la pool previamente creada, en este caso, poolIpv6.

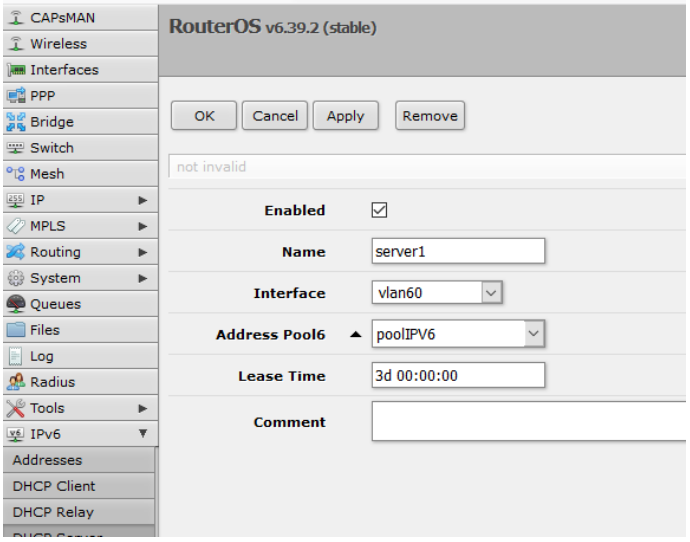


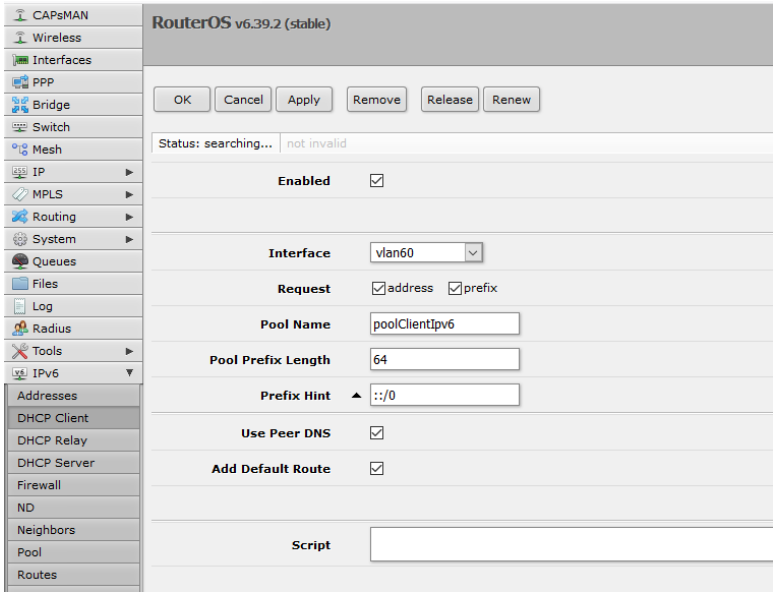

**Figura 4.8- Configuración del sub-menú DHCPv6 Server.**

**5)** Dentro del menú Ipv6 accederemos al submenú DHCP client

**6)** ***En este punto es donde según la configuración por la que se opte, se configurará un servidor sin estado, con estado o SLAAC.***

Se le asignará un nombre al Pool de nombres, se le indicará a qué interfaz va dirigida, en este caso a la Vlan 60 y ya según se exija en la configuración:

- Si se marcan las casillas address y prefix, como en el caso ejemplo, se tendrá un servidor DHCPv6 **con** estado.
- Si sólo se marca prefix se tendrá un servidor **sin** esta**Figura**




**4.8- Configuración del sub-menú DHCPv6 Client.**

Esta configuración se vería en los usuarios finales de la siguiente manera:

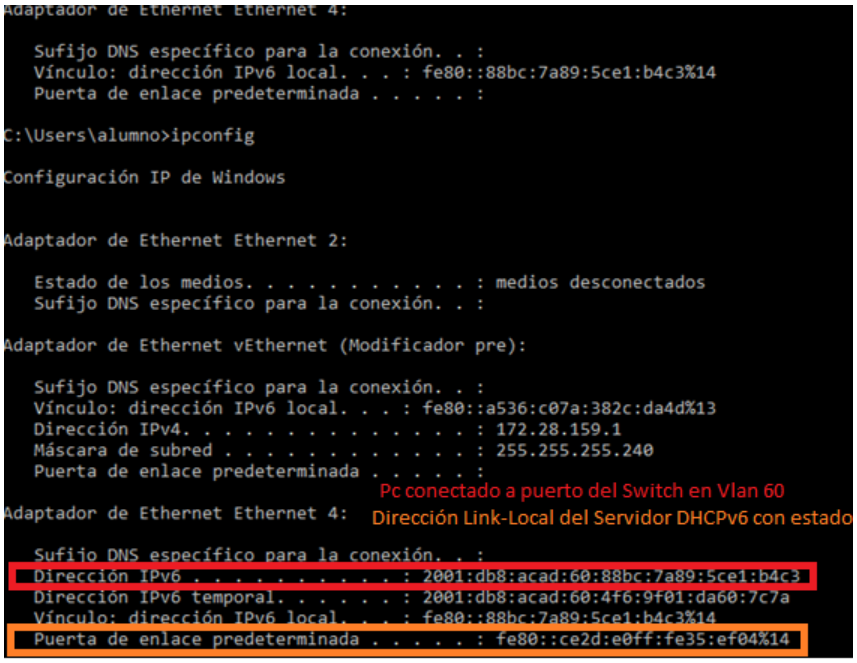



**Figura 4.8- Resultado de la configuración en el laboratorio de un servidor DHCPv6 con estado**

Se observa que ha obtenido el prefijo de red: 2001:db8:acad:60::/64 y la porción de usuario la ha auto rellenado mediante **EUI64.**

## 4.2. Lista de acceso para Ipv6 y denegación de servicios

En esta maqueta se simula un escenario donde el servidor HTTP está recibiendo un ataque por denegación de servicios tanto con tráfico HTTP como con tráfico IP y como configurar las ACLs para evitarlo.

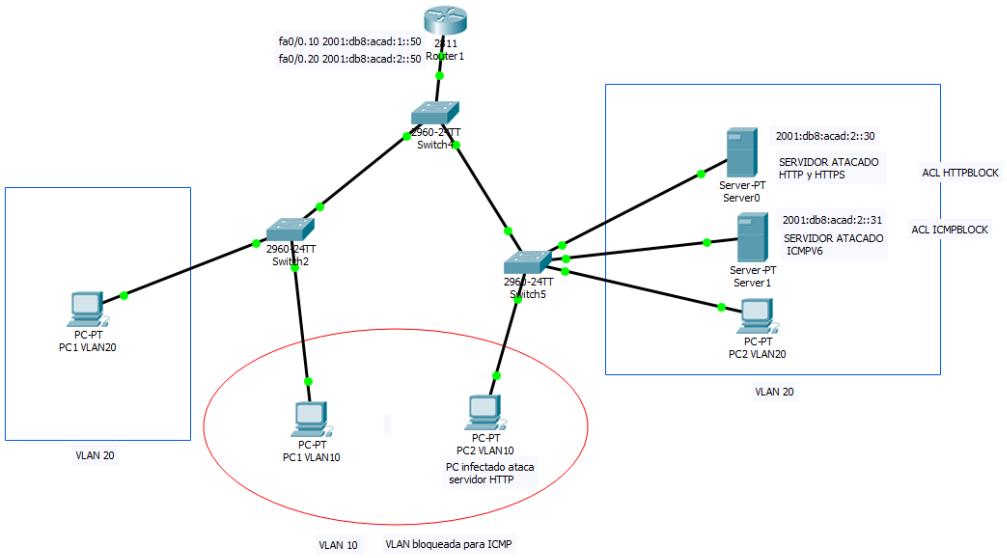



**Figura 4.9- Maqueta "Ipv6VLAN ACL" donde se configurarán varias ACL para denegar accesos o servicios VLANs.**

A continuación, en la figura 4.10 mostramos el estado del PC atacante antes de bloquear su acceso con una ACL que restrinja el tráfico HTTP y HTTPS.

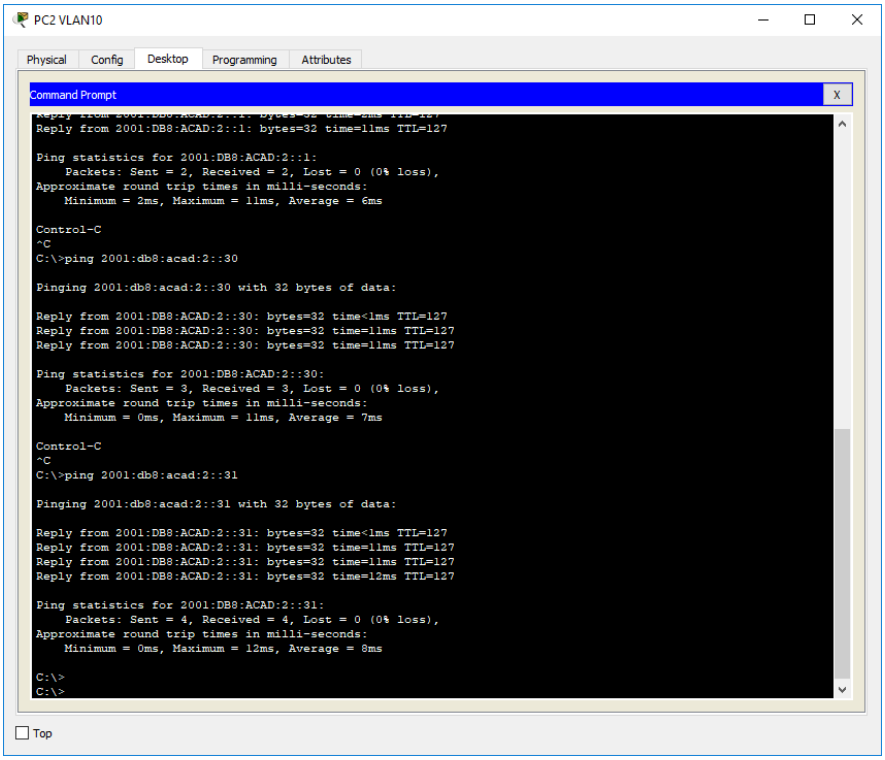



**Figura 4.10- Estado inicial de un PC de la VLAN a estudio (conexión total a todas las redes).**

Llegado el momento se ve que un Pc de la vlan10 realiza continuamente peticiones HTTP al servidor, por lo que se procede a bloquear el tráfico HTTP para la red atacante.

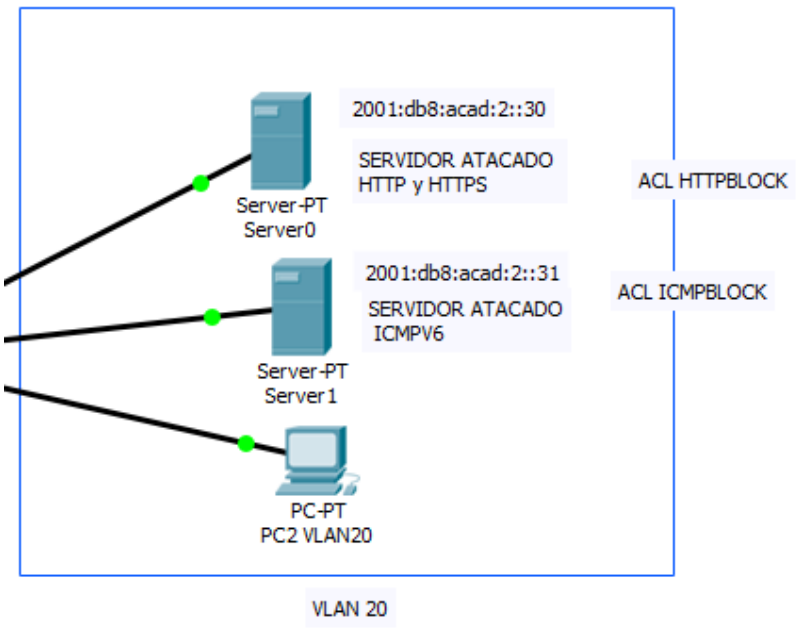



**Figura 4.11- Direcciones Ipv6 de nuestros servidores, los protegeremos usando ACLs y denegando sus servicios a los PCs atacantes**

**Código:**

```
Router(config)#Ipv6 access-list BLOCKHTTP
Router(config-Ipv6-acl)#deny tcp 2001:db8:acad:1:: /64 2001:db8:acad:2::30/64
eq www
Router(config-Ipv6-acl)#deny tcp 2001:db8:acad:1:: /64 2001:db8:acad:2::30/64
eq 443
Permit Ipv6 any any
Router(config-Ipv6-acl)#exit
Router(config)#int f0/0.10
Router(config-if)#Ipv6 traffic-filter BLOCKHTTP in
Router(config-if)#
```

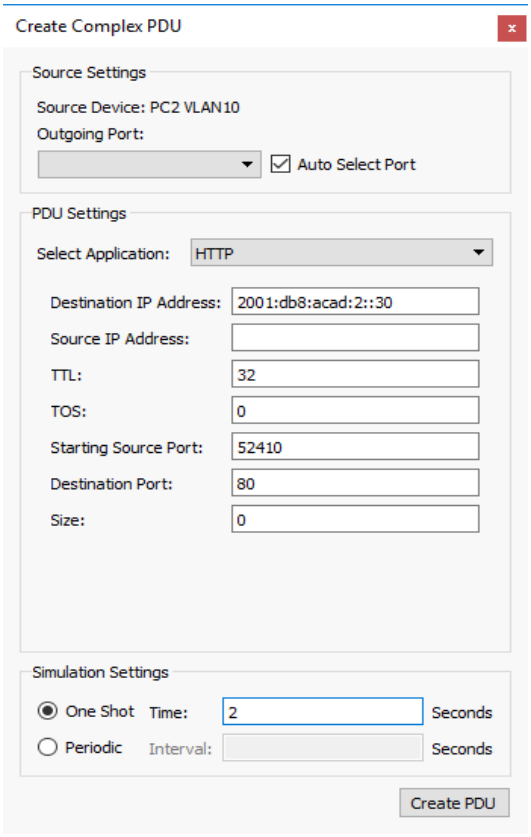



**Figura 4.12- Formato de la Complex PDU usada para comprobar el tráfico http.**

Es importantísimo que la interfaz donde se use la ACL sea en este caso la subinterfaz usada de Gateway para la vlan a la que pertenece el PC infectado.

En el caso de poner la ACL en la interfaz física no se producirá el bloqueo de tráfico.

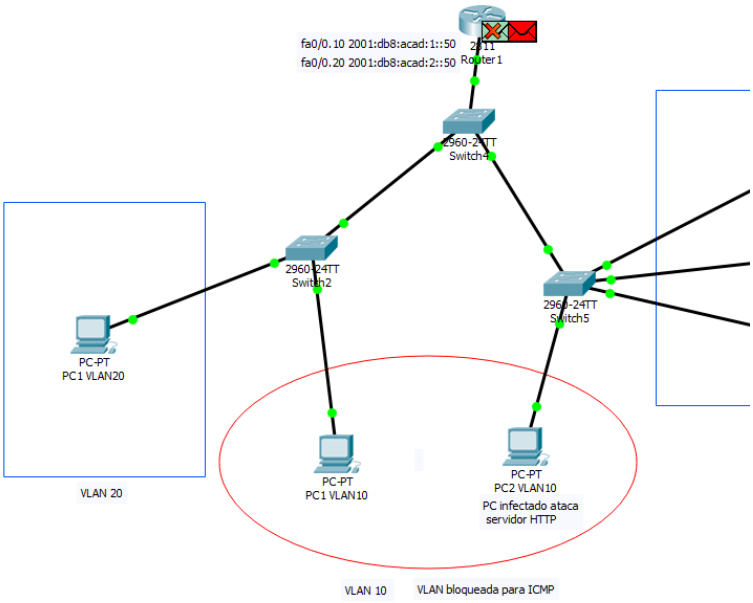



**Figura 4.13- Error al hacer una petición de un servicio HTTP al servidor desde la VLAN 10.**

En la figura 4.13 y y 4.14 se observa como el router bloquea el tráfico http procedente de la vlan del PC atacante (2001:db8:acad:1::2) cuando éste llega a la subinterfaz que hace de Gateway para la Vlan10.

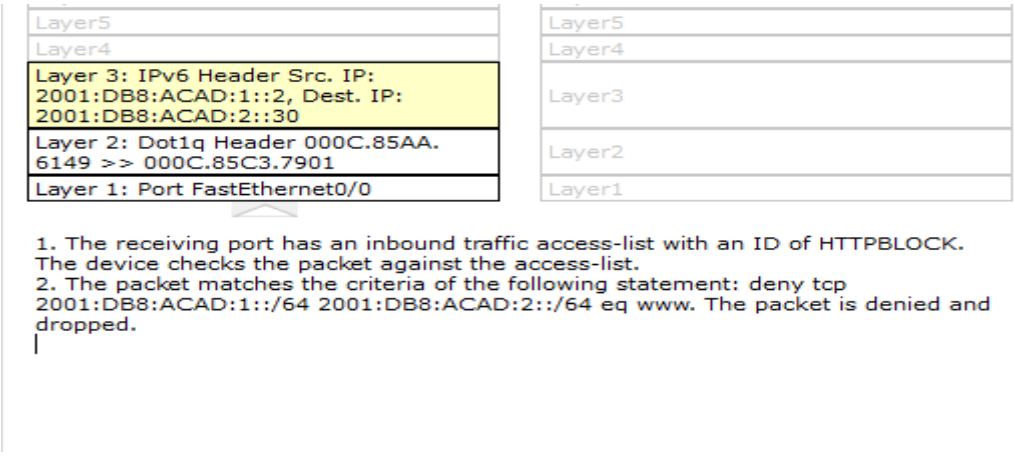



**Figura 4.14- Mensaje de error cuando la desencapsulación del mensaje de petición llega a la capa 3.**

En cambio, como solo se bloquea el tráfico HTTP y HTTPS, sigue habiendo conectividad entre ambos terminales.

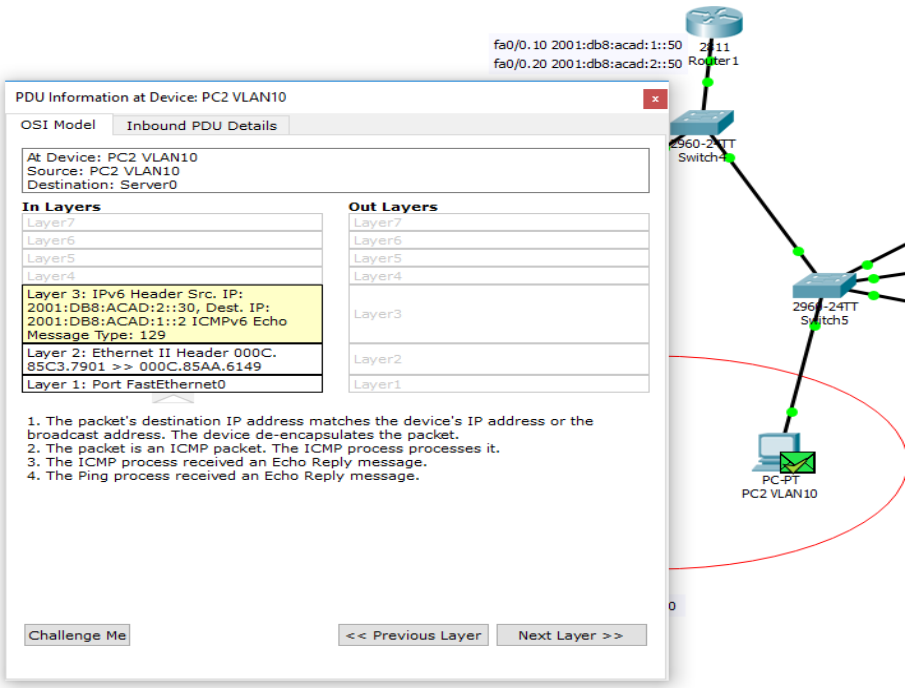



**Figura 4.15-Comprobación de que la Vlan10 sólo tiene bloqueado el tráfico HTTP y HTTPS.**

Ahora simularemos que el servidor localizado en la 2001:db8:acad:2::31/64 es atacado mediante el envió de multiples Echo Request. Se procederá a bloquear el tráfico ICMP de esa Vlan.

**Código:**

Router(config-Ipv6-acl)#deny **icmp** 2001:db8:acad:1::/64
2001:db8:acad:2::31/64
Router(config-Ipv6-acl)#permit Ipv6 any any
Router(config-Ipv6-acl)#exit
Router(config)#int f0/0**.10**
Router(config-subif)#Ipv6 traffic-filter ICMPBLOCK **out**
Router(config-subif)#exit

Como se quería demostrar, la Vlan20 no tiene problemas para realizar ping al servidor. En cambio, la Vlan10 es bloquea en la subinterfaz del router.

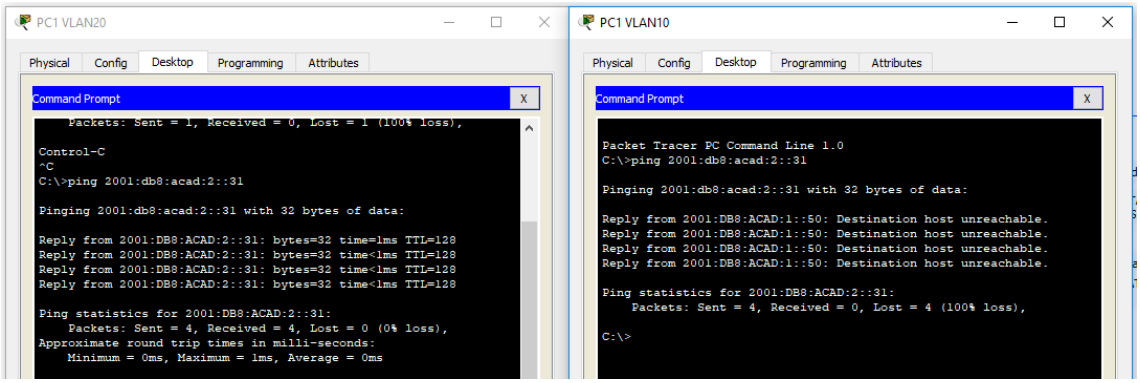



**Figura 4.16- A la derecha vemos que la Vlan10 tiene bloqueado el tráfico ICMP mientras a la izquierda vemos que la Vlan 20 no.**

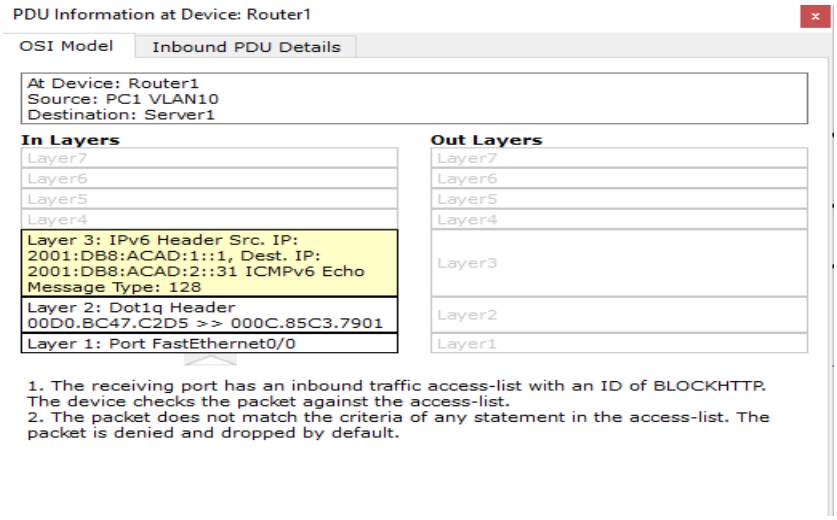



**Figura4.17- Al llegar a la capa 3, el router desecha la petición ICMP al tener una entrada en la ACL para esa Vlan.**

## Montaje en el laboratorio

Por otro lado, en el laboratorio se ha intentado reproducir los mecanismos de filtrado de tráfico y de control de acceso.

### Crear una regla

Para crear una regla es importante tener en cuenta que el orden sí importa, es decir, si se permite por ejemplo el tráfico TCP sobre un vlan determinada y en otra regla posterior se deniega todo el tráfico TCP, la primera regla prevalecerá sobre la segunda, pudiendo esa vlan aceptar tráfico TCP.

Los pasos son los siguientes:

1. Pinchar en el icono de crear nueva regla ( + )
    i. Establecer los criterios de la regla:
2. Decir si es una regla que afectará a la entrada o a la salida
3. Seleccionar las direcciones de origen y destino.
4. Elegir el protocolo objetivo (pasos del1 al 4 en la figura 4.18).
5. Seleccionar o no determinadas interfaces y opciones adicionales.
    i. Dentro de éste hay una opción la cual ayudará a una configuración más rápida. La opción ***copy***: Crea en otra pestaña una regla configurada exactamente igual que la actual. Esto es muy útil si, por ejemplo, se van a crear varias reglar para que varias vlanes tengan acceso a X red. Se configura una regla y antes de aplicarla se crean dos o varias copias con el comando copy. En las copias creadas sólo tendrías que cambiar la red origen.
    ii. En la pestaña **Action** (figura 4.19) se elige si la nueva regla será de aceptación o de negación.

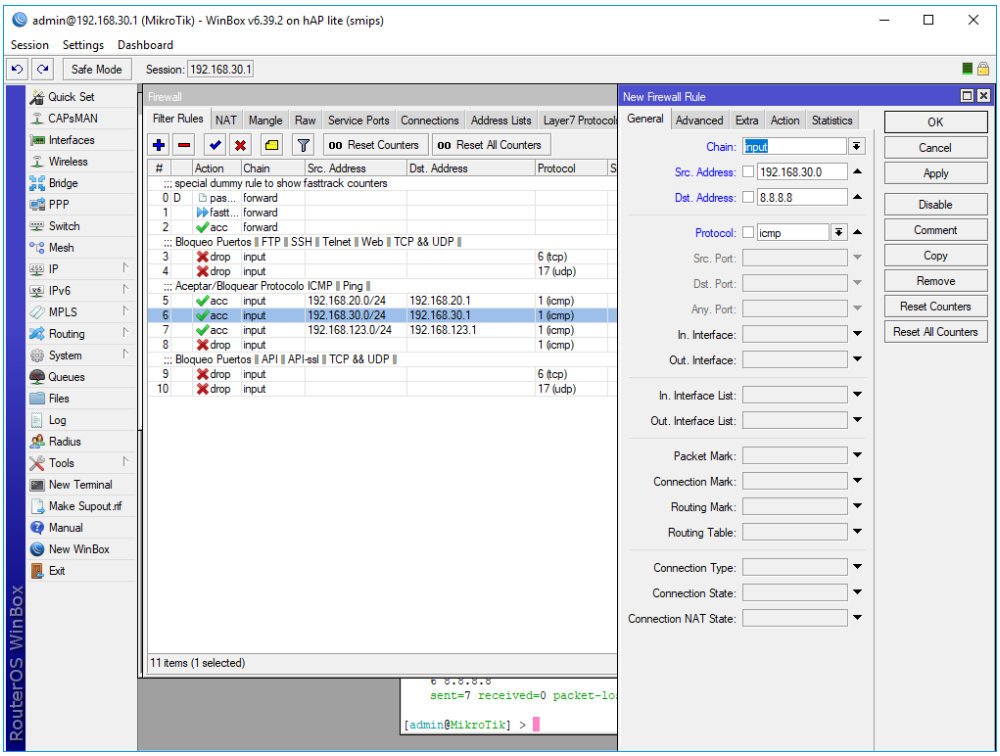



**Figura 4.18- Creación de una regla es Router Mikrotik, bloque de tráfico ICMP (Entrada 8).**

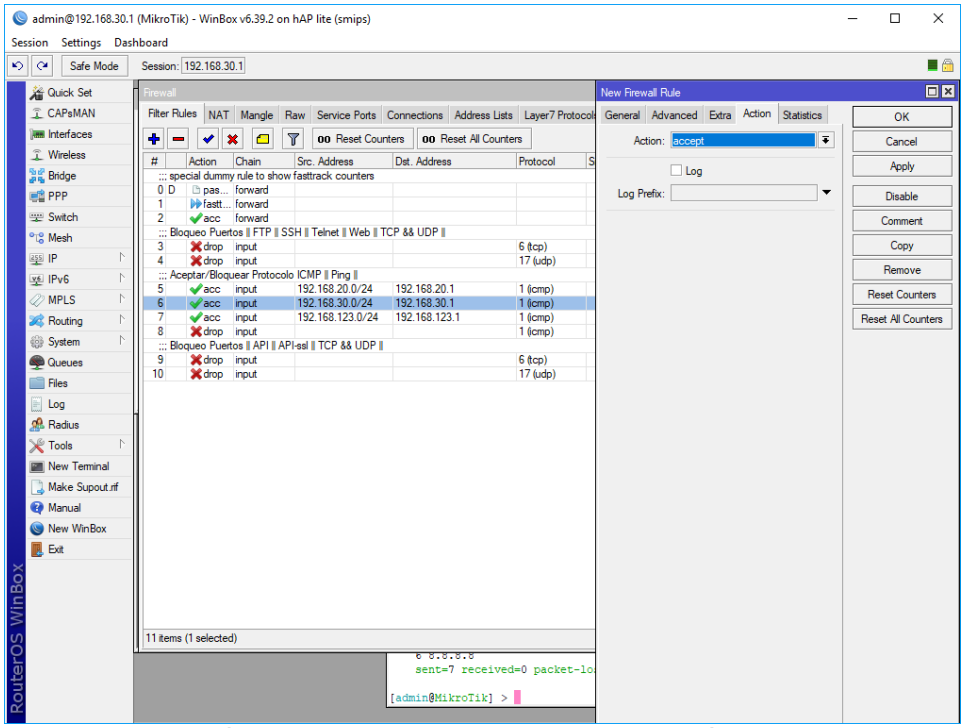



**Figura 4.19- Creación de una regla es Router Mikrotik, bloque de tráfico ICMP (Entrada 8).**

En algunas ocasiones interesará bloquear un tipo de tráfico concreto, con algunas excepciones. Como ejemplo se usará la regla genérica “drop input ICMP”, para bloquear el tráfico ICMP. Se selecciona en **rules** el protocolo ICMP, no seleccionando ninguna red de entrada y en la pestaña **Action** se opta por la opción **Deny.**

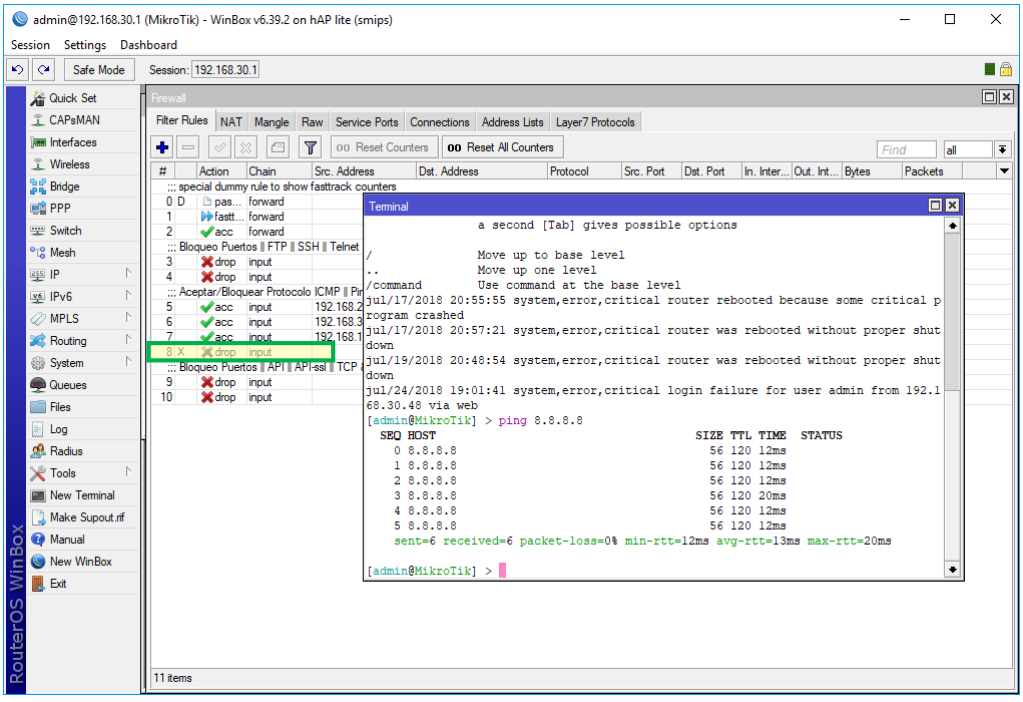



**Figura 4.20- Deshabilitar una regla es Router Mikrotik ”drop input ICMP” (Entrada 8).**

En la figura 4.20(en verde) se aprecia que la entrada 8 del menú firewall, la regla “drop input“, está en gris, significando esto que la regla se ha deshabilitado. Esto se consigue simplemente clickando en la regla y haciendo uso de la opción deshabilitar y permitiendo hacer pings tanto a las redes internas como hacia fuera, como en el ejemplo, donde se observa un ping exitoso a Google.

### Bloqueo de tráfico a redes específicas (ICMP)

Ahora se probará a quitar de la lista de excepciones a la Vlan30. Se desactivará la norma antes escrita que le permitía enviar y recibir tráfico ICMP (al hacerlo esta cambiará su color a gris). Al no tener una regla que se lo permita, se le aplica la regla por defecto, que inhabilita cualquier tipo de tráfico ICMP.

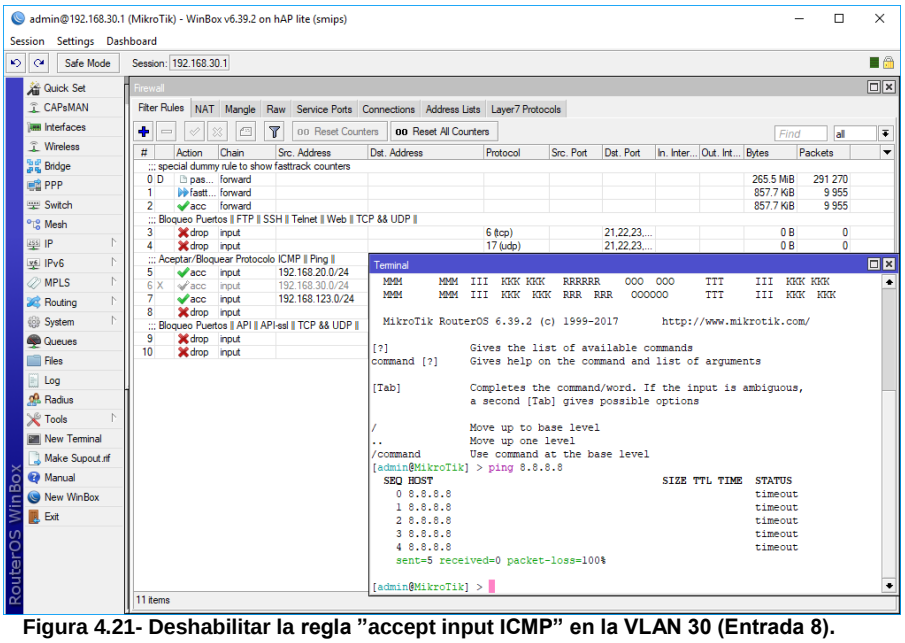



**Figura 4.21- Deshabilitar la regla ”accept input ICMP” en la VLAN 30 (Entrada 8).**

Como se observa en la figura 4.22, al deshabilitar el tráfico ICMP en la Vlan30 no se obtiene respuesta al intentar hacer ping a Google (8.8.8.8).

Para terminar, se dejará la configuración de manera que pueda haber comunicación entre las redes privadas pero evitando que tengan comunicación con el exterior.

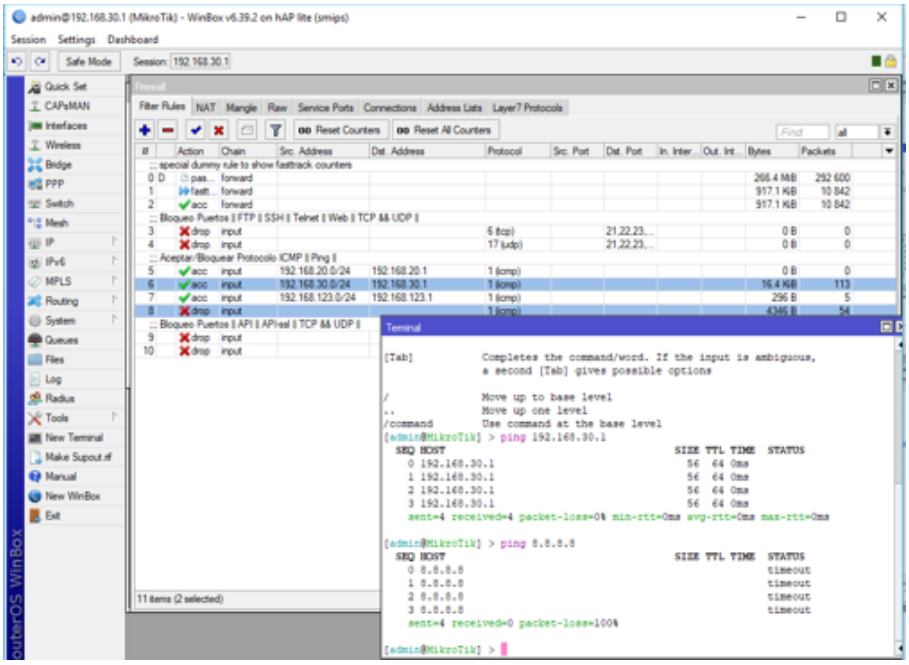



**Figura 4.22- AL deshabilitar la regla ”accept input ICMP” en la VLAN 30 fallan los pines tanto a la red interna como hacia el exterior.**

Estos ejemplos se han realizado sobre vlanes con direcciones y redes para ipv4, pero funcionaría igual si la vlan con la que se trabaja posee redes con Ipv6. Vlan es un protocolo de capa 2 y es independiente del protocolo de capa de red usado.

## 4.3. Creación de una zona DMZ (Simulador)

Desde un punto de vista técnico, se puede definir DMZ como una zona desmilitarizada (demilitarized zone) que se sitúa entre la red interna y la red externa (normalmente Internet). La función de una DMZ es permitir las conexiones tanto desde la red interna como de la externa, mientras que las conexiones que parten de la DMZ generalmente irán dirigidas a la red externa así, los equipos externos jamás podrían conectarse a la red interna. El siguiente esquema explica de manera gráfica cómo funciona:

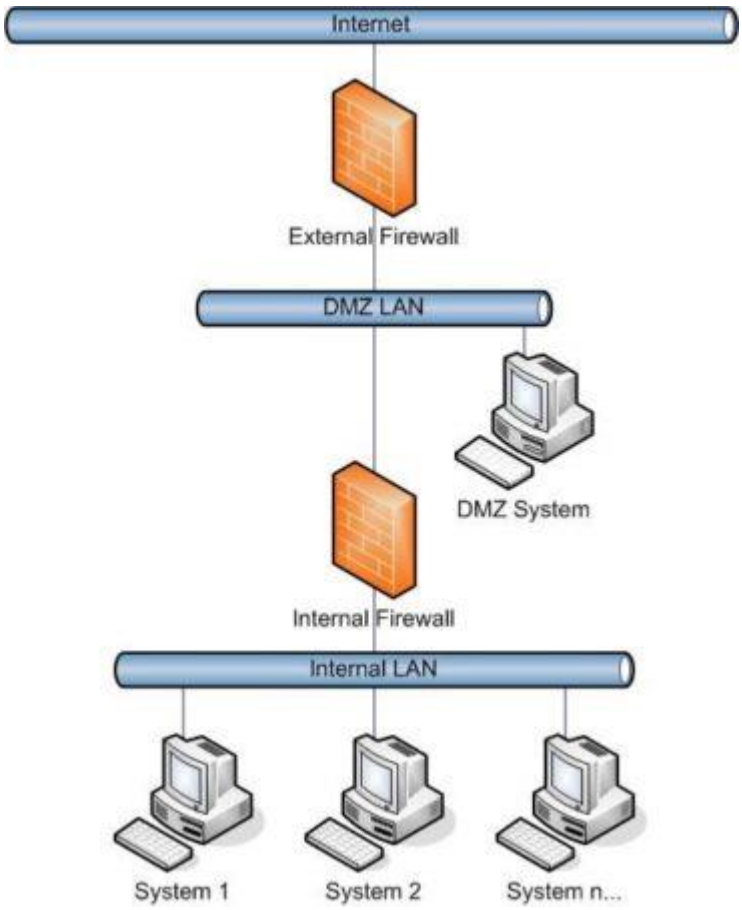



**Figura 4.22- Esquema de red de una red con DMZ.**

En el caso particular de los routers domésticos o para pymes, el concepto DMZ host se refiere a la posibilidad *de establecer una dirección IP que tendrá todos los puertos abiertos en el router*, con la excepción de aquellos que ya estuvieran definidos en la sección NAT.

Para acceder a la configuración DMZ en un router se debe acceder al panel de administración escribiendo su IP en cualquier navegador (por defecto 192.168.0.1) e introducir nombre de usuario y contraseña. Después, acceder a la sección Advanced y en Firewall Settings se verá la opción DMZ Host; simplemente hay que introducir la IP y marcar Enable DMZ para activarlo (en algunos modelos de routers puede ser necesario un reinicio).

Establecer un DMZ Host para un dispositivo puede ser muy útil cuando se tienen problemas con la configuración y se quiere descartar que el origen del error sean los puertos. Así, temporalmente se puede dejar libre acceso a la IP que se haya asignado al dispositivo para, una vez detectado el problema, volver configurar únicamente los puertos que vayan en la sección Port Forwarding.

Es muy importante entender los importantes riesgos de seguridad que conlleva DMZ Host y utilizarlo únicamente como herramienta de diagnóstico. En escenarios domésticos y sin los conocimientos ni herramientas adecuadas, desproteger completamente un equipo ante posibles intrusiones es una fuente casi segura de problemas.

Para el entorno considerado, el diseño de una red corporativa, este elemento permitirá tener una DMZ para el acceso desde el exterior de la red a servidores en dicha zona.

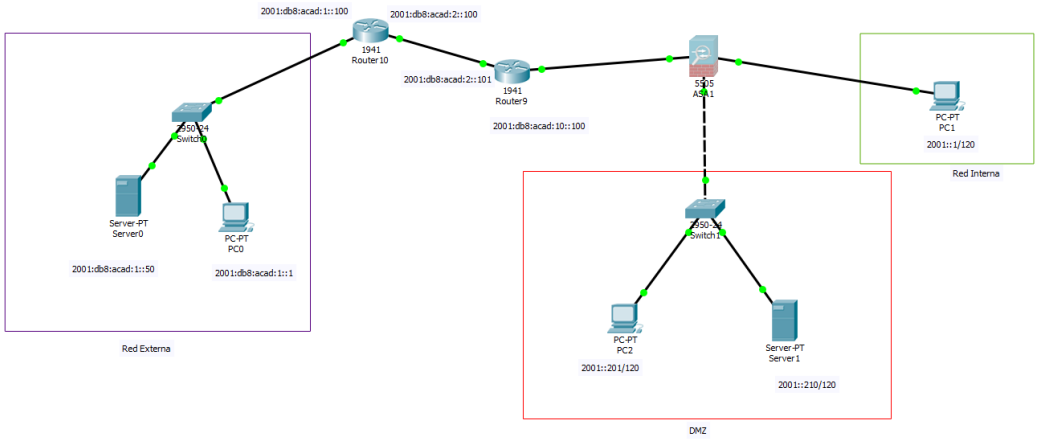



**Figura 4.23- Maqueta "DMZ" donde se configura un ASA para que diferencie entre redes externa, interna y DMZ.**

Con este ejemplo se intenta crear una zona que sea accesible desde la red externa e interna, pero que no comprometa la red interna desde la red externa o la misma DMZ.
Para ello se crean 3 vlans según siguen:

```
ciscoasa(config)#int et0/0
ciscoasa(config-if)#no shutdown
ciscoasa(config-if)#exit

ciscoasa(config)#interface Vlan1
ciscoasa(config-if)#security-level 100
ciscoasa(config-if)#ip address 192.168.1.1 255.255.255.0
ciscoasa(config-if)#Ipv6 address 2001::ff/120
ciscoasa(config-if)#nameif inside
ciscoasa(config-if)#!

ciscoasa(config-if)#interface Vlan2
ciscoasa(config-if)#nameif outside
ciscoasa(config-if)#security-level 0
ciscoasa(config-if)#ip address 192.168.3.1 255.255.255.0
ciscoasa(config-if)#Ipv6 address 2001:db8:acad:10::101/64
ciscoasa(config-if)#!

ciscoasa(config-if)#interface Vlan3
ciscoasa(config-if)#no forward interface Vlan1
ciscoasa(config-if)#nameif dmz
INFO: Security level for "dmz" set to 0 by default.
ciscoasa(config-if)#security-level 0
ciscoasa(config-if)#ip address 192.168..2.1 255.255.255.0
ciscoasa(config-if)#Ipv6 address 2001::250/120
ciscoasa(config)#exit

ciscoasa(config)#Ipv6 route outside ::/0 2001:db8:acad:10::100
```

```
ciscoasa(config)#class-map INSPECCION
ciscoasa(config-cmap)#match any
ciscoasa(config-cmap)#exit

ciscoasa(config)#policy-map POLITICA
ciscoasa(config-pmap)#class INSPECCION
ciscoasa(config-pmap-c)#inspect icmp
ciscoasa(config-pmap-c)#exit
ciscoasa(config)#service-policy POLITICA global
```

Después habrá que asegurarse que en las interfaces del ASA la VLAN a la que pertenece coincide con la que se ha configurado (a veces no la cambia).

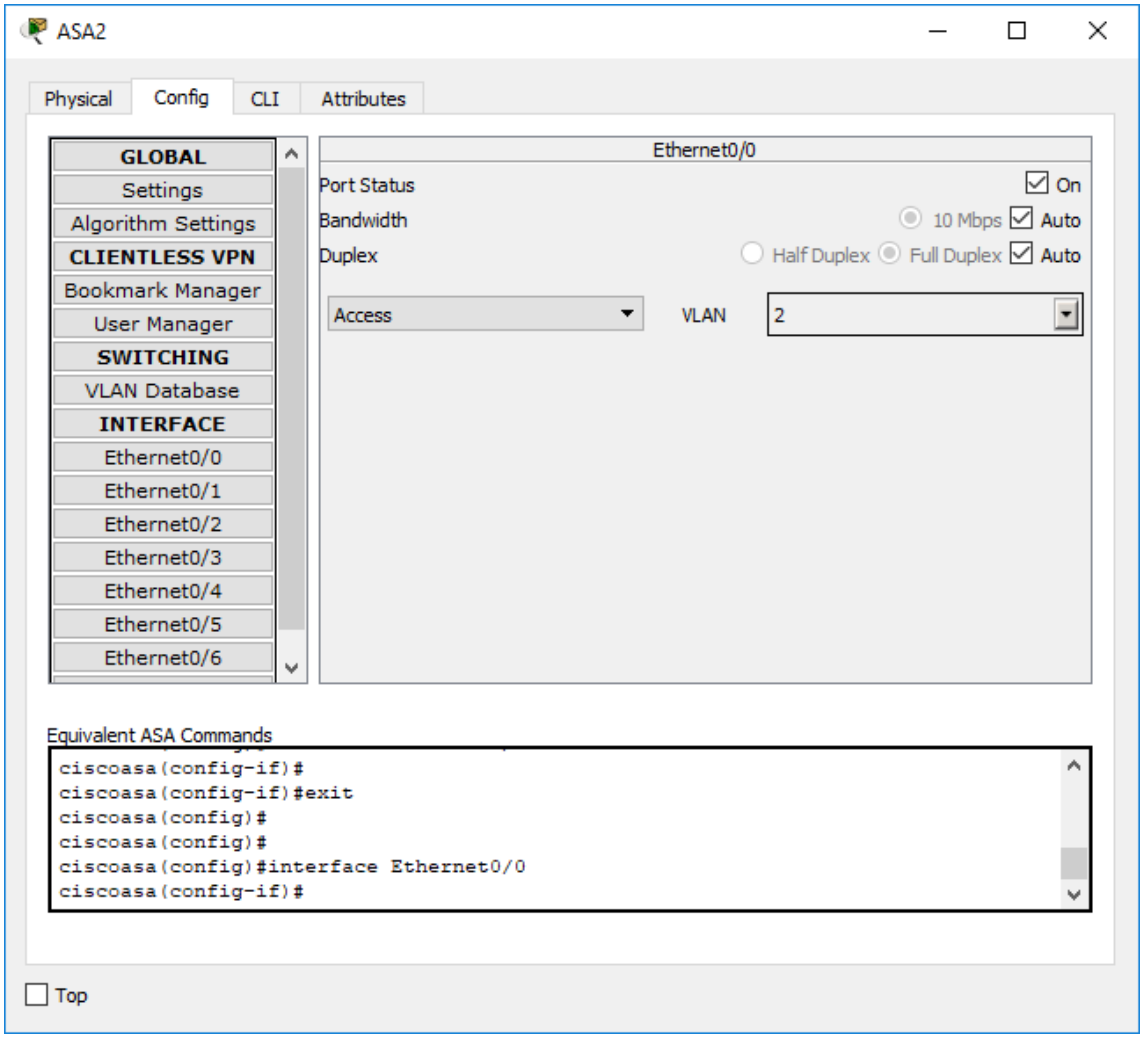



**Figura 4.24- Dentro del ASA se asociarán las interfaces a las VLAN que les correspondan.**

Además, se deberá configurar el enrutamiento estático de todas las redes en los respectivos routers.

En la figura 4.25 y 4.26 se muestran el mensaje de error que da el intento de hacer ping desde la red externa y desde la red DMZ a la red interna.

*Desde la red externa*

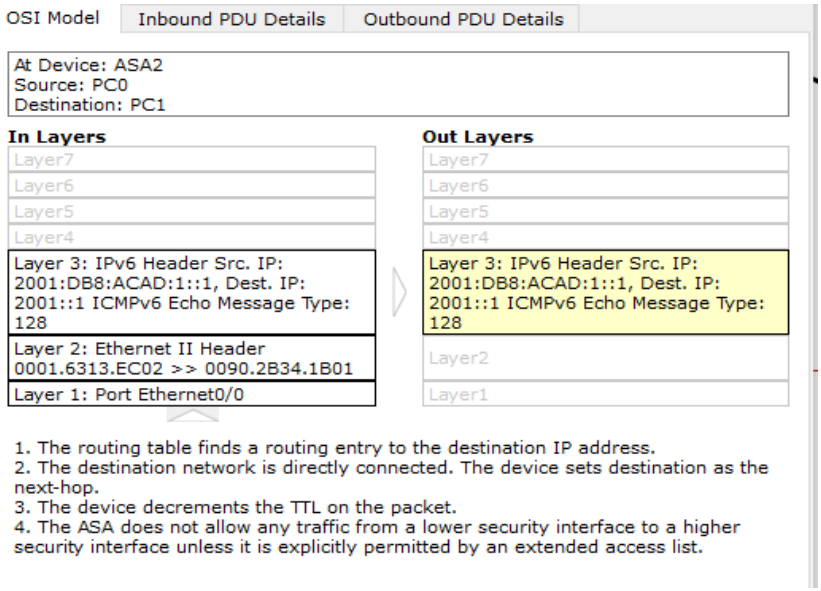



**Figura 4.25- Mensaje de error al intentar hacer ping a la red interna DESDE la externa.**

*Desde la red DMZ*

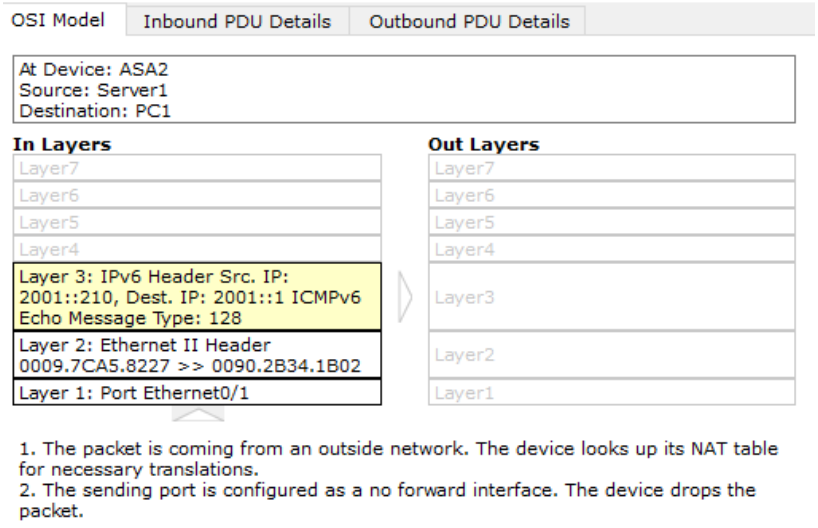



**Figura 4.26- Mensaje de error al intentar hacer ping a la red interna DESDE la zona DMZ.**

Y el segundo objetivo también se cumple, pues se tiene una zona que es accesible tanto desde la red interna y segura como desde el exterior.

## 4.4. Direccionamiento dinámico en Ipv6

### 4.4.1 RIPng

RIPng es un protocolo nuevo que está diseñado para que funcione como sus predecesores de Ipv4. Sus operaciones básicas son las mismas, y utiliza el algoritmo general de Bellman-Ford y todas sus operaciones. Las principales características nuevas con respecto a versiones están enfocadas principalmente a adaptarse a Ipv6.

A continuación, se citarán las más importantes:

- Deja de usar definitivamente el protocolo UDP.
- Usa el número de puerto estándar 521. Los routers que utilizan el protocolo RIPng escuchan a la dirección de multicast FF02::9 y mandan sus mensajes de actualización a esa dirección.
- Usa el algoritmo vector distancia para determinar una ruta óptima hacia el destino, usando la cuenta de saltos (hop count, número de routers entre un nodo de origen y uno de destino) como métrica. Selecciona a la ruta con la métrica más baja como la preferida para enviar paquetes.
- Los routers configurados con RIPng intercambian información acerca de la disponibilidad de la red mediante mensajes de actualización de ruta.
- Opera dentro de un Sistema Autónomo (AS), que es un conjunto de routers y redes controladas por un único administrador.
- Usa actualizaciones de envenenamiento en reversa y horizonte dividido para evita routing loops.

El formato de mensaje en un paquete RIP se muestra en la figura 4.28:

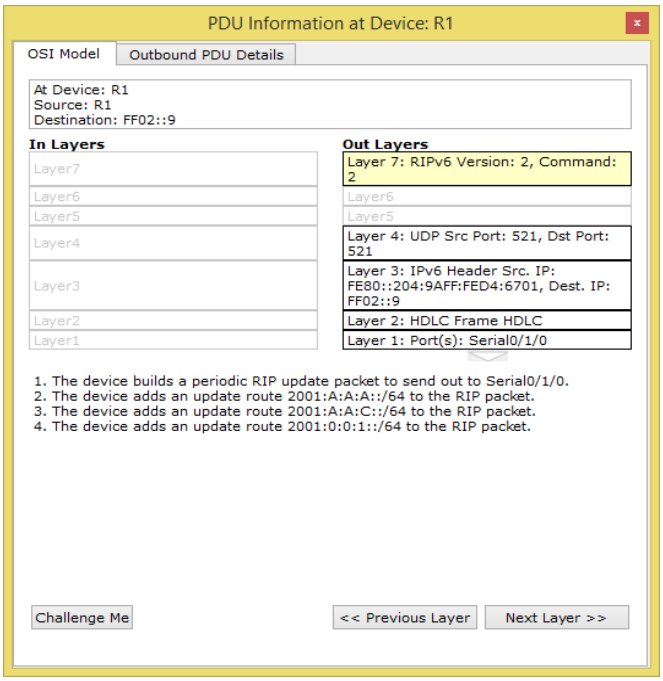



**Figura 4.28-Formato de mensajes de actualización/anuncio de redes en RIP.**

A continuación, la figura 4.29 mostrará un ejemplo simple de una red simulada en Cisco Packet Tracer.

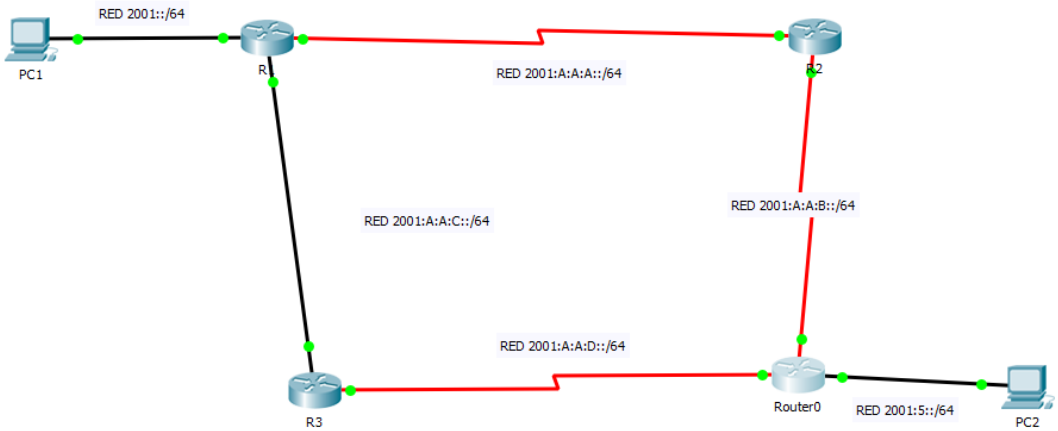



**Figura 4.29- Maqueta "RIPng" sobre Cisco Packet Tracer donde se configura la versión del protocolo para Ipv6.**

La configuración de la red se llevó a cabo de la siguiente forma:

1) Configurar las direcciones Ipv6 de cada interfaz.
2) Activar el enrutamiento Ipv6 (Ipv6 unicast-routing) para que los router se puedan comunicar por Ipv6.
3) Habilitar una interfaz para que envía y reciba mensajes RIP de un determinado dominio (RIP).
4) Asegurarse que en todos los router del dominio RIP el nombre de dominio sea el mismo, en este ejemplo será id1.
5) Recordar que hay que habilitar rip en todas las interfaces, incluidas las interfaces que están conectadas directamente a redes internas, pues si no hace, éstas no se enviarán en los mensajes RIP y no se podrá llegar hasta ellas desde redes a más de un salto.
6) No es necesario configurar nada en los pcs, pues sólo marcando la opción autoconfig recibirán una dirección Ipv6 (SLAAC).

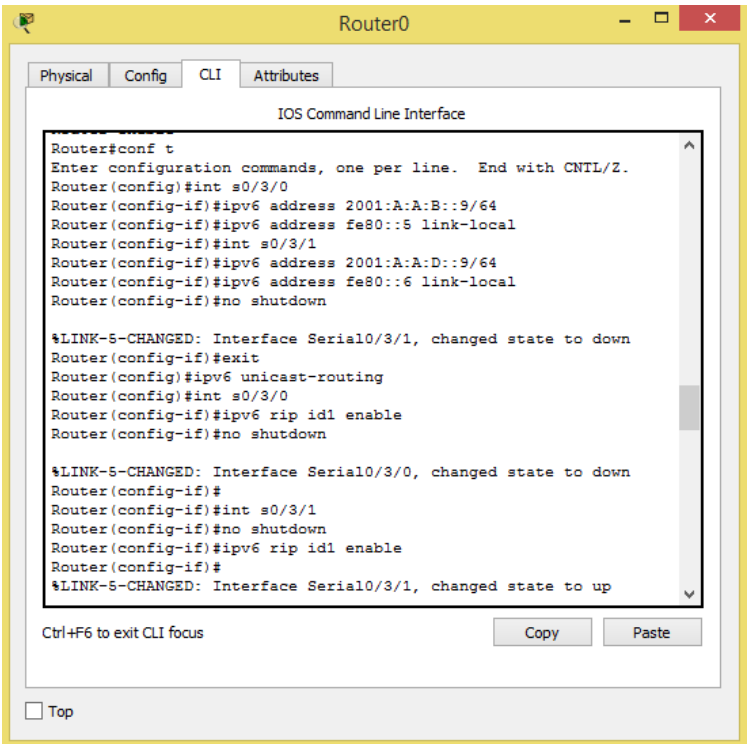



**Figura 4.30- Para el uso del protocolo hay que habilitar en cada interfaz RIp.**

7) Con el comando ***show Ipv6 route*** se comprobará que el router 1 ha aprendido la red 2001:A:A:C::/64 y 2001:A:A:A::/64 por estar directamente conectadas (C) pero las redes 2001:A:A:D::/64 y 2001:A:A:B::/64 las ha aprendido de varias interfaces distintas y por medio de RIPng(R).

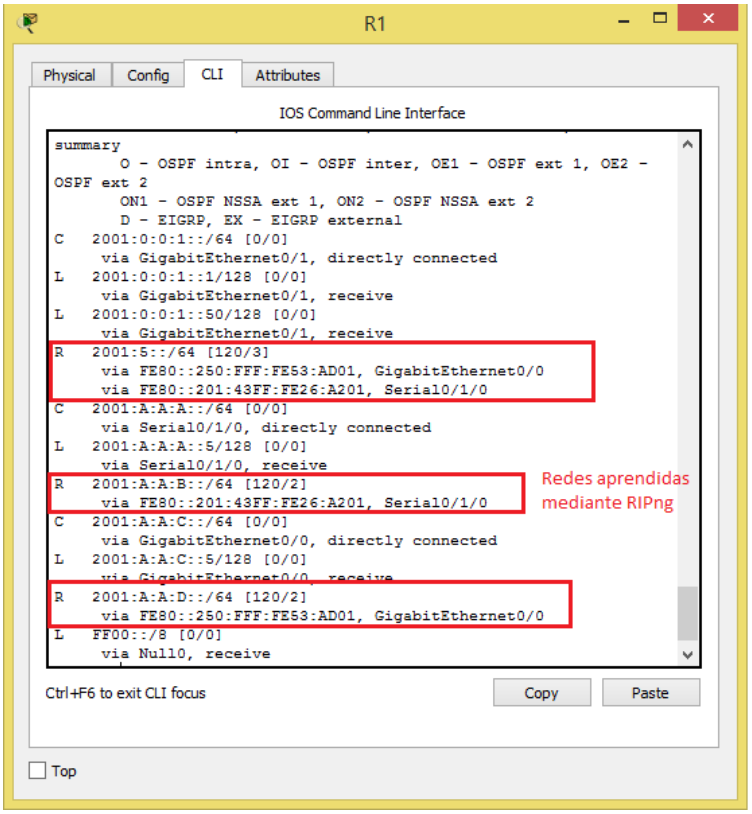



**Figura 4.31- Resultado del comando "Show Ipv6 route".**

Se puede observar que una red se puede obtener mediante varias interfaces. Cuando un router manda un mensaje RIP incluye en el todas las redes que conoce, no solo las que tiene directamente conectadas,. El router objetivo se quedará como fuente origen (via) la que este directamente conectada. Si se recibe información de una red pero ninguno de los emisores está directamente conectado a ella, se dejan ambas fuentes , como es el caso en el router 1 con la red 2001:5::/64 recibida desde la interfaz gigabit y desde la intenrfaz serial.

Por último se observa que la maqueta después de seguir esta configuración tiene comunicación total.

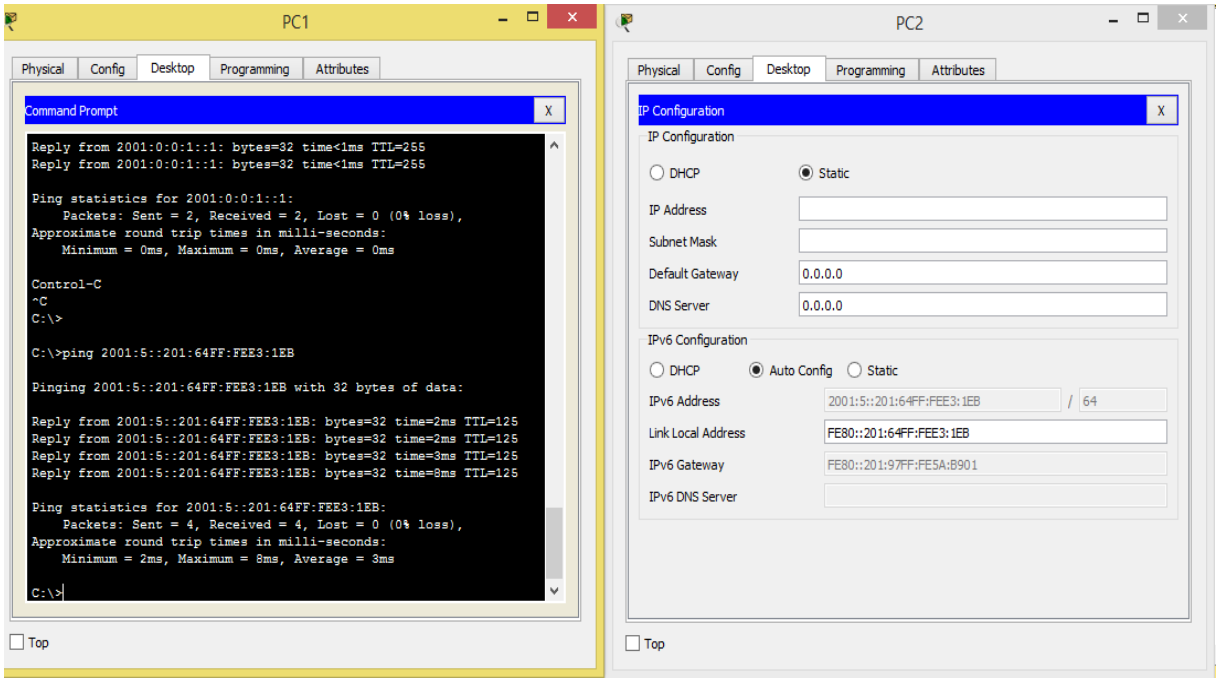



Figura 4.32- Comunicación total en toda la red usando únicamente RIPng.

### 4.4.2 OSPFv3

OSPF (Open Shortest Path First) es un protocolo de ruteo para IP. Un Link-State Protocol el cual toma las decisiones de ruteo basándose en los estados de los enlaces que conectan la fuente y las máquinas de destino. El estado de un enlace es una descripción de esa interfaz y de la relación a sus dispositivos de interconexión de redes vecinos. La información que se manda incluye el prefijo del Ipv6 de la interfaz, la máscara de la red, el tipo de red que está conectada, los routers conectados con esa red. Esta información se propaga en los anuncios de estado del vínculo (LSA).

La versión 3 OSPF, que se describe en el RFC 2740, soporta el Ipv6.

**Tipo 1 (Router LSA)**

Cada router dentro de un área X envía LSA de tipo 1 a sus vecinos. Este LSA nunca sale del área a la cual pertenece y contiene el Router-ID del remitente, y todos los enlaces que lo conectan.

**Tipo 2 (Network LSA)**

Es enviado por el DR (Designated Router) dentro de la red. Él informa a los demás de las redes y su máscara que tiene conectados. Este LSA nunca sale del área a la cual corresponde. Es decir, un ABR no lo reenvía a otra área.

**Tipo 3 (Summary LSA)**

Las envía un ABR para traspasar la información de un área a otra. OSPF las denomina “summary”.

**Tipo 4 (ASBR-Summary LSA)**

Representa a un ASBR (Autonomous System Border Router)

**Tipo 5 (External LSA)**

Representa a una ruta externa redistribuida dentro de OSPF desde otro protocolo (Ej: EIGRP). El ASBR toma las rutas provenientes del protocolo externo y las reenvía como tipo 5 a todas las áreas internas, excepto a las de tipo Stub.

**Tipo 7** *(Usados en router con varios protocolos de encaminamiento distintos)*

Las normas de OSPF dicen que solamente en un área Backbone (Area 0) debería haber redistribución. En un área NSSA se puede conectar un router que tenga conexión con otro protocolo de enrutamiento externo (ej: RIP) y el ASBR enviaría esas redes en formato de tipo 7, de tal manera que el ABR las tome y las redistribuya como tipo 5.

Para que se entienda un poco mejor cuando se usa cada tipo de LSA se empleará el siguiente ejemplo de red usado en la web de Cisco.

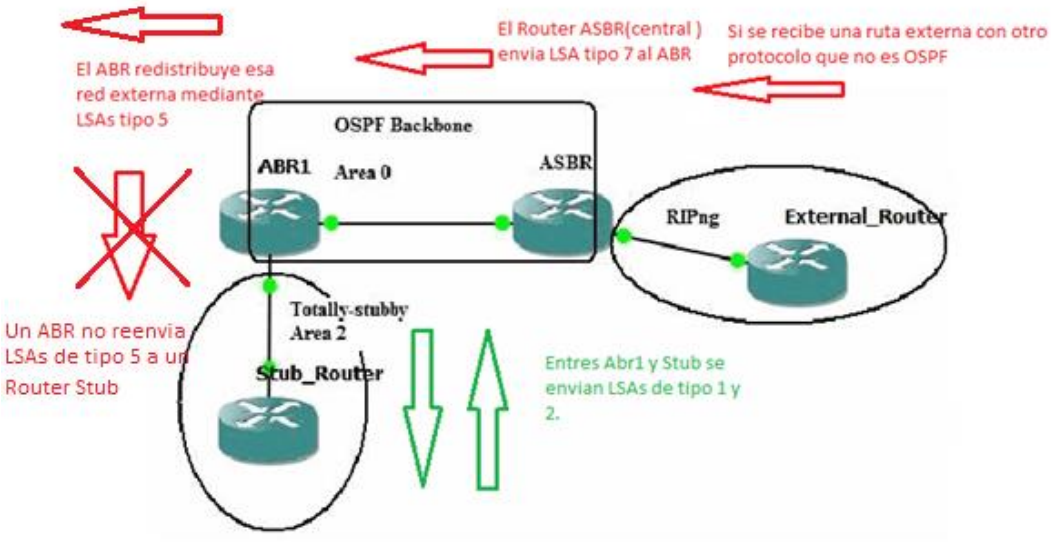



**Figura 4.33- Figura que muestra la relación entre los tipos de LSA y el rol del router que lo manda.**

Los LSAs tipo 1 y 2 son mensajes internos al área donde se envían información del estado de los enlaces y routers.
Los LSAs tipo 3 y 4 son enviados entre los router internos al dominio OSPF que no son ABR ni ASBR. Son los mensajes que transportan las informaciones sobre las redes conocidas.
Los LSA de tipo 5 también son conocidos como external links y son los mensajes que circulan por las áreas estándar y Backbone.
Los LSA tipo 7 son mensajes sobre redes externas conocidas a través de otros protocolos también, pero creados por el ABSR. Estos mensajes son los únicos que pueden entrar si un área se declara NSSA (no acepta LSAs de tipo 5).

Después de esta explicación se abordará la red de estudio de la figura 4.34, la cual posee cinco áreas, cuatro de ellas serán estándar y una será del tipo NSSA. Más adelante se explicará los tipos de áreas.

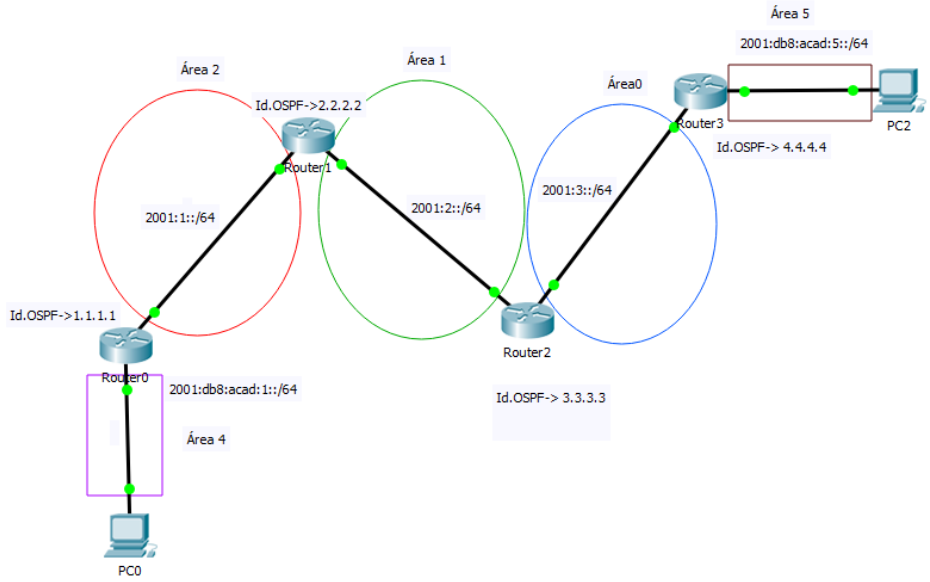



**Figura 4.34- Maqueta en Cisco Packet Tracer sobre el protocolo OSPFv3.**

Para configurar OSPF en un router se seguirán los siguientes pasos

1) Habilitar el unicast-routing
2) Habilitar las interfaces y poner la respectiva dirección.

3) Dentro de cada interfaz añadir una sentencia donde se indique a qué dominio OSPF pertenece la interfaz y dentro de qué área se integra.
4) Dentro de cada router nombrar un router-id e indicar dentro de qué tipo de área estará el router (por defecto aceptará todos los tipos de LSAs y retransmitirá periódicamente las redes que conoce).
5) Todos los routers de un mismo dominio deben de tener el mismo identificador y todos los router dentro de una misma área también deben de reflejar el mismo id de área.

El código para configurar un router 1 es éste:

```
Router(config-if)#ipv6 address 2001:1::2/64
Router(config-if)#ipv6 ospf 1 area 2
%OSPFv3-4-NORTRID: OSPFv3 process 1 could not pick a router-
id,please configure manually
Router(config-if)#int f0/1
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up

Router(config-if)#ipv6 address 2001:2::1/64
Router(config-if)#ipv6 ospf 1 area 1
Router(config-if)#exit
Router(config)#ipc6 router ospf 1
               ^
% Invalid input detected at '^' marker.

Router(config)#ipv6 router ospf 1
Router(config-rtr)#router-id 2.2.2.2
Router(config-rtr)#
```

**Figura 4.35- Código para la configuración de OSPFv3 en el router 1.**

Y a continuación se mostrará cómo queda la tabla de enrutamiento dentro de uno de los router. Se observa que las rutas aprendidas por OSPF vienen indicadas con el prefijo OI.

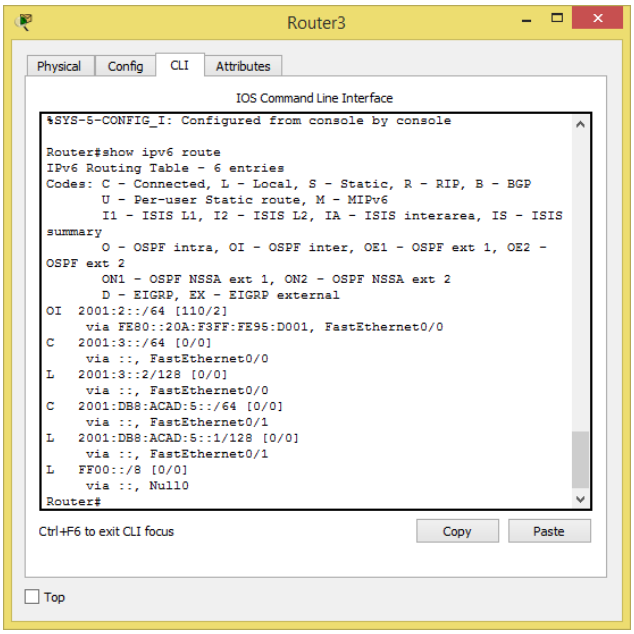



**Figura 4.36- Resultado del comando "Show Ipv6 route".**

Como se puede observar en la figura 4.36, la red 2001:1::/64 no aparece en la tabla de enrutamiento. Esto es porque al configurar el router1, dentro de la configuración del OSPF, se añade un comando:

area 4 stub

Ésta es una de las muchas opciones que da OSPFv3 y que permite, deniega o restringe el envío y recepción de mensajes LSA.

A continuación se nombran los tipos de áreas más importantes:

| **Área** | **Restricción** |
|---|---|
| Normal | Ninguna |
| Stub | No se permite ningún LSA externo de tipo 5 |
| Totally-stub | No se permite ningún tipo 3,4 ó 5 a excepción de la ruta de resumen predeterminada |
| NSSA | No se permite ningún LSA externo de tipo 5 salvo los que han sido previamente transformados de LSA tipo 7 en el ABR |
| NSSA totally stub | No permite LSAs de tipos 3,4, ruta predeterminada y tipo 5 que no hayan sido transformados previamente de tipo 7 en el ABR. |

**Figura 4.37- Figura que relaciona los tipos de áreas en OSPF y su nivel de restricción con respecto a la recepción de mensajes LSA.**

Se puede comprobar en la figura 4.38 que salvo el área 4, la cual se ha configurado para no responder ni enviar LSAs (salvo los de tipo 7), toda la red tiene comunicación total.

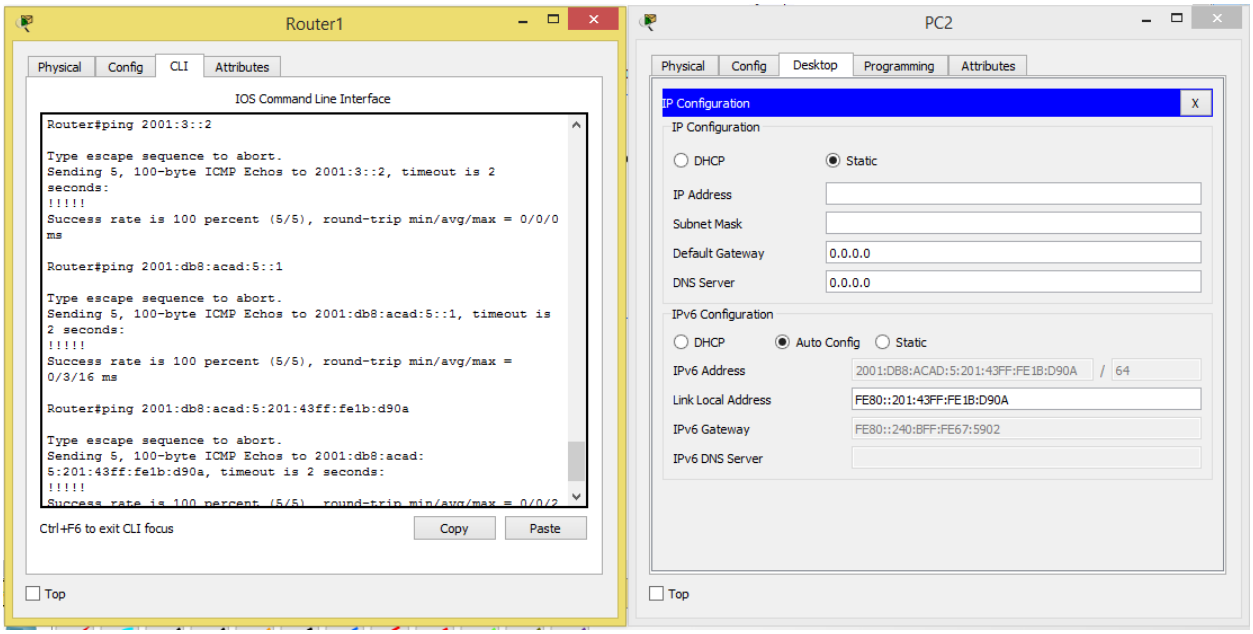



**Figura 4.38- Resultado de realizar pings a todas las rutas aprendidas mediante OSPFv3.**

# Capítulo 5. Convivencia y/o adaptación de ipv4 a Ipv6

En la actualidad existen miles de millones de dispositivos que utilizan IPv4. Así que pensar en una migración simultánea de IPv4 a Ipv6 de todos estos dispositivos es inviable. En algunos casos, aunque se quiera migrar a Ipv6, los dispositivos o el software pueden no admitir Ipv6 o no tener soporte adecuado para Ipv6. Por lo que la migración de IPv4 a Ipv6 requerirá años incluso décadas. Afortunadamente, se han desarrollado diferentes mecanismos de transición que permiten una integración fluida de IPv4 e Ipv6 y no requieren que la actualización de todos los nodos sea simultánea. Entre ellos cabe destacar:

- Pilas duales IPv4/IPv6
- Túneles
- Conversión entre IPv4 e Ipv6 por medio de NAT-PT

Ninguna de estas soluciones basta para resolver todos los problemas. Diferentes redes requieren distintas estrategias. En la mayoría de las redes, se necesita una combinación de estas herramientas. A continuación, se presentan las bases de cada uno de estos mecanismos.

## 5.1. Dual Stack

El método de integración mediante pilas duales o dual stack se basa en la utilización de uno o varios nodos que tienen instaladas la pila de protocolos IPv4 y la pila Ipv6 a la vez. Los dispositivos con ambas pilas, también denominados nodos IPv4/Ipv6, pueden recibir y enviar tráfico a nodos que sólo soportan uno de los dos protocolos (nodos sólo IPv4 o sólo Ipv6).

Estos nodos de pila dual pueden ser tanto encaminadores como equipos de trabajo. Si se trata de un equipo tendrá asociada a cada tarjeta de red tanto una dirección IPv4 como una dirección Ipv6, y podrá enviar paquetes IPv4 a otros equipos IPv4 y paquetes Ipv6 a otros equipos Ipv6. Si se tratara de un encaminador, además de las direcciones IPv4 y los protocolos de encaminamiento habituales, tendrá que tener direcciones Ipv6 y protocolos de

encaminamiento Ipv6 para admitir tanto tráfico de equipos IPv4 como los Ipv6. El encaminador podrá recibir y enviar tanto paquetes IPv4 como paquetes Ipv6. Un nodo de pila dual (implementa los dos protocolos IPv4 e Ipv6) elige qué pila (protocolo) utilizar en función de la dirección de destino del paquete. De esta forma, cuando se establece una conexión hacia un destino sólo IPv4, se utilizará la conectividad IPv4. En cambio, si la comunicación es hacia una dirección Ipv6, se utilizará la red Ipv6. En caso que el destino tenga ambos protocolos, normalmente se intentará conectar primero por Ipv6 y como segunda opción por IPv4.

### 5.1.1 Ejemplos de convivencia

Una configuración Dual Stack, como hemos dicho en el apartado anterior se conseguiría replicando las configuraciones para poder dar servicio tanto en Ipv4 como en Ipv6.
Los equipos deberían tener asociado Ips para ambos protocolos, los cuales deberían estar activos (como se muestra en la figura 5.1):

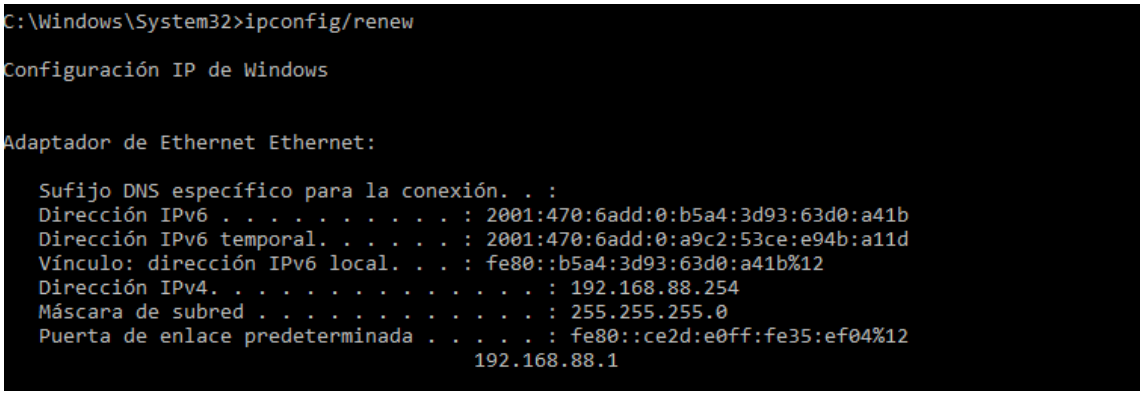



**Figura 5.1- Configuración dual-stack de un PC. Tiene dirección Ipv4 e Ipv6.**

Los enrutadores deberán tener una tabla de enrutamiento y direcciones por defecto para ambos protocolos, así como tener el direccionamiento Ipv6 activado (en algunos routers viene desactivado por defecto).

Ipv4

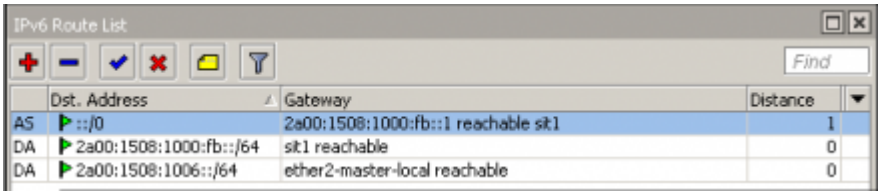



**Figura 5.2- Tabla enrutamiento para Ipv4.**

Ipv6

| | Dst. Address | Gateway | Distance | Routing Mark | Pref. Source |
|---|---|---|---|---|---|
| AS | 0.0.0.0/0 | 109.60.164.1 reachable gateway1 | 52 | | |
| AS | 0.0.0.0/0 | 109.60.164.1 reachable gateway1 | 51 | gate2 | 109.60.164.128 |
| AS | 0.0.0.0/0 | 109.60.164.1 reachable gateway1 | 52 | gate1 | 109.60.164.72 |
| AS | 10.0.0.0/8 | center reachable | 1 | | |
| AS | 10.0.0.0/8 | center reachable | 1 | gate2 | |
| AS | 10.0.0.0/8 | center reachable | 1 | gate1 | |
| DAC | 10.34.4.0/24 | local4 reachable | 0 | | 10.34.4.1 |
| DAC | 10.46.3.0/24 | local3 reachable | 0 | | 10.46.3.1 |
| DAC | 10.46.4.0/24 | local4 reachable | 0 | | 10.46.4.1 |
| DC | 10.46.5.0/24 | local5 unreachable | 255 | | 10.46.5.1 |
| AS | 109.60.164.0/24 | gateway2 reachable | 1 | gate2 | 109.60.164.128 |
| AS | 109.60.164.0/24 | gateway1 reachable | 1 | gate1 | 109.60.164.72 |
| DAC | 109.60.164.0/24 | gateway2 reachable, gateway1 reachable | 0 | | 109.60.164.128 |
| DAC | 192.168.46.0/24 | local3 reachable | 0 | | 192.168.46.1 |

**Figura 5.3-Tabla enrutamiento para Ipv6.**

Así como las interfaces configuradas para ambos protocolos.

Ipv4

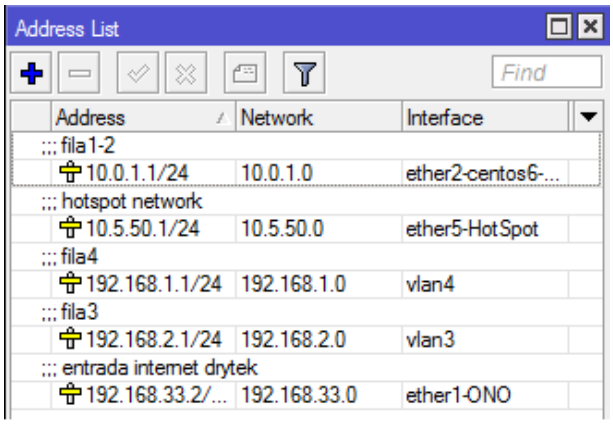



**Figura 5.4- Tabla de interfaces Ipv4.**

Ipv6

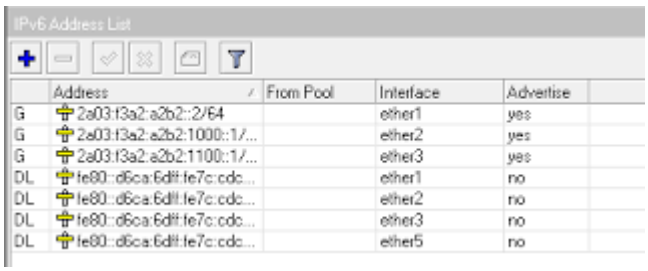



**Figura 5.5- Tabla de interfaces Ipv6.**

Si la adquisición de Ips se realiza mediante DHCP también se deberían tener uno por cada protocolo.

### 5.1.2- Ventajas y desventajas

Las aplicaciones antiguas que sólo admiten IPv4 siguen funcionando igual que antes. Las nuevas aplicaciones aprovechan las dos capas IP. Este es uno de los métodos más comúnmente utilizados para migrar de las comunicaciones dentro de una empresa a Ipv6. Este método facilita una transición lenta que permite al personal técnico de la empresa adaptarse paulatinamente a la tecnología Ipv6.

En algunos casos, la actualización podría exigir nuevo software o hardware, pero los encaminadores se podrían migrar fácilmente al uso de pilas duales, y la mayoría de los sistemas operativos admiten Ipv6 en la actualidad.

La principal desventaja de este método es una disminución del rendimiento de los equipos de red, ya que tienen que mantener tablas de encaminamiento y rutas independientes para cada uno de los protocolos.

## 5.2 Túneles

Los túneles proporcionan un mecanismo que permite establecer conexiones Ipv6 sobre una red IPv4 (y viceversa).

Los túneles se utilizan cuando un equipo desea acceso a la red Ipv6 existente. Para ello el equipo deberá crear un túnel a través de IPv4 con un router que tenga tanto acceso a Ipv6 como IPv4. Este método se está utilizando en la actualidad por parte de algunos ISPs que sólo dan conexión IPv4 para que cualquiera pueda tener acceso a la red Ipv6.

También permiten unir redes Ipv6 utilizando la infraestructura IPv4 existente. Este mecanismo consiste en enviar datagramas Ipv6 encapsulados en paquetes IPv4 (y viceversa). Los extremos finales del túnel siempre son los responsables de realizar la operación de encapsulado y desencapsulado del paquete Ipv6 en IPv4.

Existen varias técnicas de tunneling, entre ellas:

**6over4 -** Esta técnica se utiliza para comunicar nodos Ipv6 aislados dentro de un sitio (nodos sin conexión directa a routers Ipv6) con el resto de nodos IPv4. También se emplea cuando el router Ipv6 no tiene acceso o permiso para transmitir paquetes Ipv6 sobre el enlace.

*Para ello se emplean redes IPv4 que soportan multidifusión.* Esta técnica crea un enlace virtual utilizando un grupo de multidifusión IPv4, *mapeando las direcciones Ipv6 dentro de este grupo de multidifusión.* De esta forma, estos equipos Ipv6 no requieren direcciones IPv4 compatibles, ni túneles configurados.

Cualquier host que quiera participar en 6over4 sobre una red IPv4 puede establecer una interfaz de red virtual Ipv6. La dirección local se determina:

- Empieza con fe80:0000:0000:0000:0000:0000:,
- los 32 bits más bajos tienen que ser los de la dirección IPv4 del host.

Por ejemplo, el host 192.0.2.142 sería:
*fe80:0000:0000:0000:0000:0000:c000:028e.*

6over4 confía en la disponibilidad de multicast de IPv4, que no suele ser implementado en la infraestructura IPv4 (multicast es casi tan reciente como IPv6). 6over4 tiene poco uso práctico y no está soportado por los sistemas operativos más comunes.

**6 to 4-** Este mecanismo se utiliza para comunicar redes Ipv6 aisladas por medio de la red IPv4 normalmente Internet. Es la forma más usual de conectar redes al mundo Ipv6 sin tener asignadas direcciones Ipv6 ya que permite construir una red completa Ipv6 a partir de una única dirección pública IPv4. De esta forma, los sitios pueden empezar a utilizar Ipv6 sin solicitar espacio de direccionamiento a los organismos reguladores.

El router extremo de la red Ipv6 crea un túnel sobre IPv4 para alcanzar la otra red Ipv6. Los extremos del túnel son identificados por el prefijo del sitio Ipv6. Esta técnica deriva automáticamente a partir de una dirección IPv4 un prefijo Ipv6 válido y único para cada isla de Ipv6. Este prefijo /48 está formada por 16 bits fijos que indican que se está utilizando la técnica 6to4(2002::/16) más 32 bits que identifican al router externo del sitio. *Este mecanismo funciona incluso cuando la dirección IPv4 pública es única y se accede a la red global utilizando el protocolo NAT, que la forma de acceso a Internet a través de ISPs.*

**Direccionamiento automático de túnel dentro de un sitio** (*Intra-Site Automatic TunnelAddressing Protocol* - ISATAP) Este método está pensado para la comunicación entre nodos de un mismo sitio. La técnica funciona empotrando la dirección IPv4 del nodo en el identificador EUI-64 del interfaz. Puesto que este módulo viene a solucionar los problemas de comunicación dentro de un sitio, *las direcciones IPv4 no tienen por qué ser globales*. Esto significa que, aunque exista NAT, el mecanismo seguirá funcionando correctamente.

### 5.2.1 Ejemplo de Túnel 6to4

En la figura 5.6 se muestra la maqueta que se ha construido en cisco Packet Tracer para simular dos redes Ipv6 que necesitan tener comunicación, pero cuyo tráfico ha de pasar por una red Ipv4.

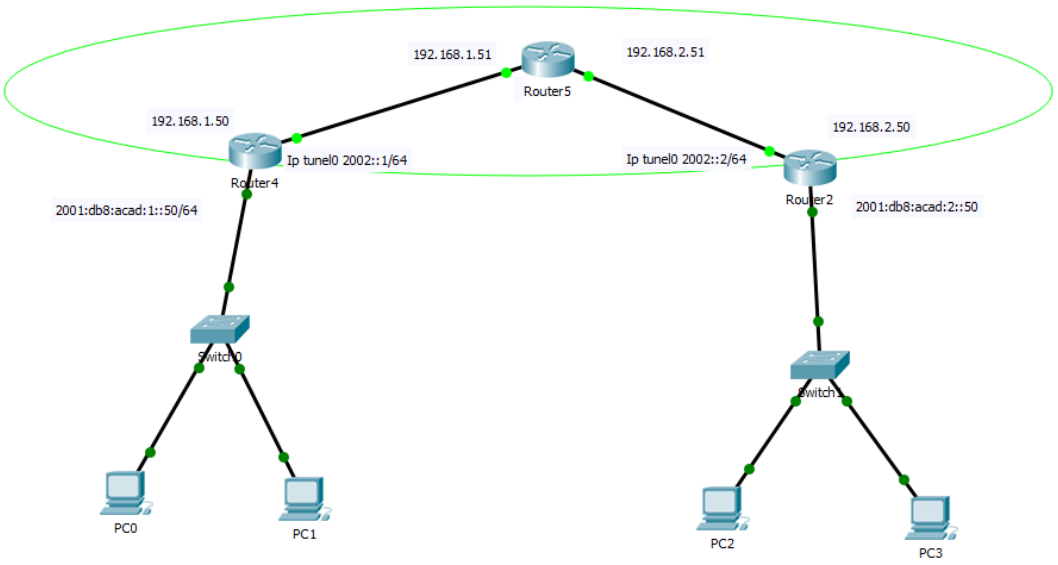



**Figura 5.6- Maqueta sobre Cisco Packet Tracer de mecanismo de adaptación túnel 6to4.**

Los pasos a seguir a la hora de configurar un túnel 6to4 son:

- Configurar las ips de la interfaz de la red interna (Ipv6) como de la interfaz de red externa (Ipv4).
- Activar el direccionamiento unicast de Ipv6
- Configurar en un router:
    - Darle un nombre a la interfaz túnel el cual tendrá que ser igual en ambos router.
    - La interfaz túnel, la cual estará vinculada la interfaz interna Ipv4.
    - Introducir la Ipv4 del otro extremo del túnel.
    - Añadir el comando que que indica que este será un túnel 6to4
- Configurar de manera similar el otro router extremo

- Añadir en ambos routers los mecanismos de direccionamiento (estático o dinámico) para que se pueda llevar a cabo la comunicación.

Código:

*Router(config)#int f0/0*
*Router(config-if)#int f0/1*
*Router(config-if)#Ipv6 address 2001:db8:acad:1::50/64*
*Router(config-if)#no shutdown*

*Router(config-if)#*
*%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up*

*%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up*

*Router(config-if)#int f0/0*
*Router(config-if)#ip address 192.168.1.50 255.255.255.0*
*Router(config-if)#no shutdown*

*Router(config-if)#*
*%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up*

*Router(config-if)#int tunnel 0*

*Router(config-if)#*
*%LINK-5-CHANGED: Interface Tunnel0, changed state to up*

*Router(config-if)#tunnel source fa0/0*
*Router(config-if)#tunnel destination 192.168.2.50*
*Router(config-if)#tunnel mode Ipv6ip*
Router(config-if)#Ipv6 address 2002::1/64

Router(config-if)#exit

Router(config)#Ipv6 route 2001:db8:acad:2::/64 2002::2

ip route 192.168.2.0 255.255.255.0 192.168.1.51

En la figura 5.7 se ve como se trata un paquete dirigido a una red Ipv6 lejana que tiene que ser enviado a través de una red Ipv4:

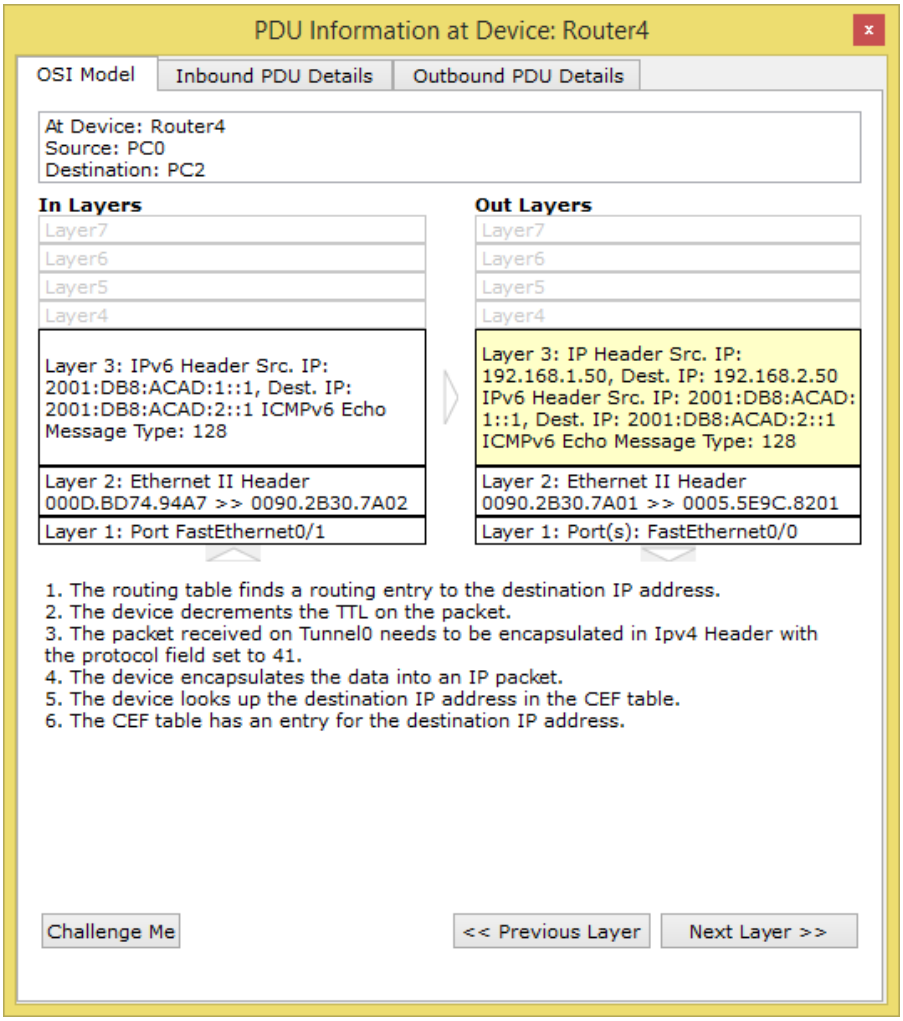



**Figura 5.7- Router extremo del tunner que recibe mensaje con destino a una red lejana Ipv6.**

El paquete Ipv6 se encapsula con una cabecera ipv4 con dirección el otro extremo del túnel para que así, un paquete (Ipv6) que en teoría no debería poder usar una red Ipv4 lo utilice y así poder llegar a su destino.

Usando el comando *show Ipv6 route* se puede ver la tabla de enrutamiento del router 4 y que al haberle puesto la ruta: Ipv6 route 2001:db8:acad:2::/64 2002::2 el mismo ya ha reconocido la dirección 2002::2 como el otro extremo de la interfaz túnel..

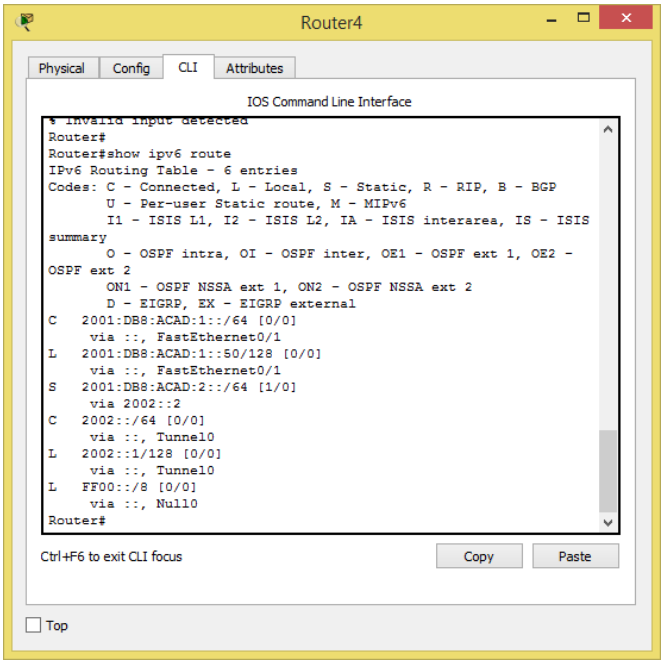



**Figura 5.8- Comando "Show ipv6 route" donde se muestra la interfaz tunnel0.**

Y por último unas muestras de que hay conexión total entre ambas redes

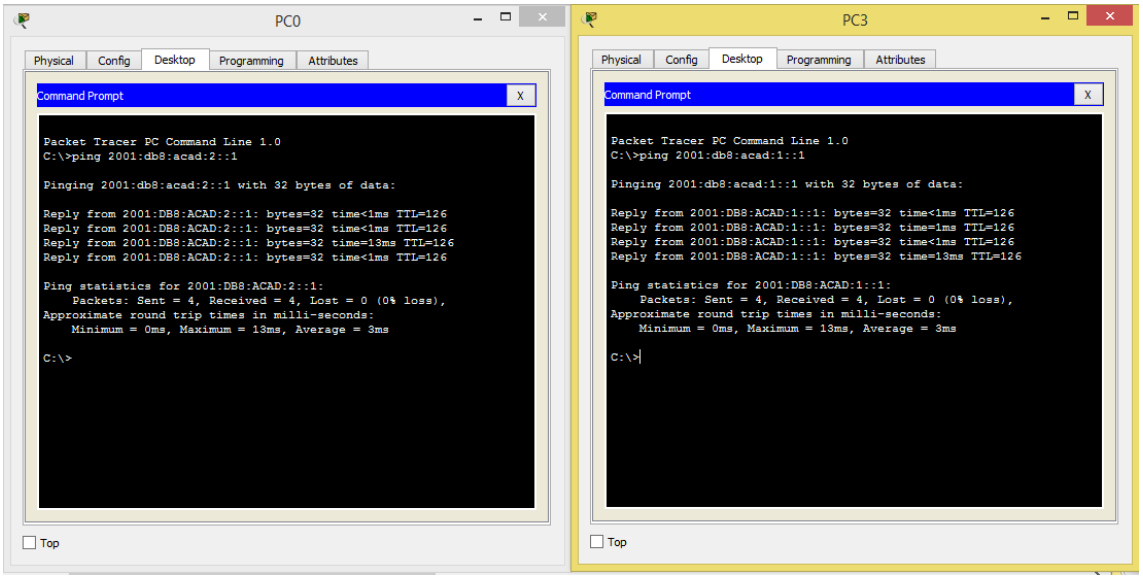



**Figura 5.9- Éxito al realizar ping a redes lejanas pasando por un túnel 6to4.**

### 5.2.2 Ejemplo de túnel 6to4 en el laboratorio

Ahora se va a llevar a cabo una demostración de cómo hacer uso de un servicio de Tunnel Broker para crear un túnel 6to4 que permita **a tu red doméstica** tener acceso a internet haciendo uso del protocolo Ipv6.

Este apartado no ha sido realizado en el laboratorio sino en mi propia casa. Actualmente mi servicio de internet está contratado con Telecartagena[8], la cual no da soporte en Ipv6.

Después de darse de alta y crear una cuenta en Hurricane Electric (anexo), se reciben los datos que tendrá nuestro túnel (Figura 5.10).

- La dirección Ip 84.236.143.102 es mi Ip pública.
- La dirección Ip 216.66.80.26 es la Ip del otro extremo del túnel, en este caso situado en Londres.
- La dirección 2001:470:1f08:656::1/64 es la Ip versión 6 de nuestro router.
- La dirección 2001:470:1f08:656::2/64 es la dirección Ip versión 6 del otro extremo del router.
- Por último, 2001:470:6add::/48 será nuestra pool que posteriormente utilizaremos para crear un DHCPv6 con estado que repartirá Ips a los ordenadores de casa.

[8] Operador Telecartagena- https://www.telecartagena.es/internet.php

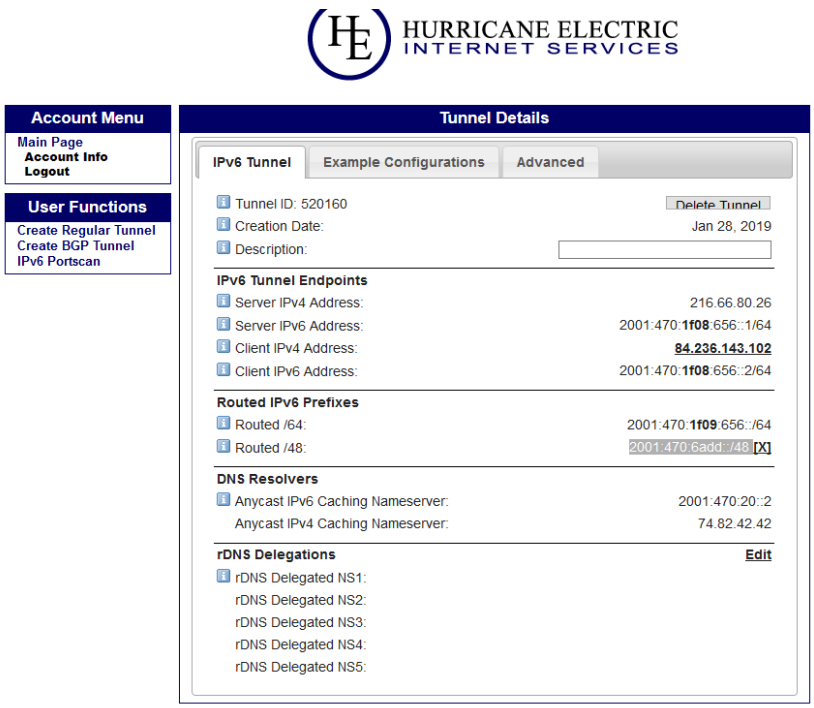



**Figura 5.10- Datos proporcionados por Hurricane Electric para establecer nuestro túnel 6to4.**

El primer paso será crear en el router nuestro extremo del túnel. Para ello vamos al menú Interfaces → Interface → new → 6to4 tunnel.

Y allí se introduce la Ip publica (Local Address) y la Ip del servidor de Tunnel Broker (Remote Address).

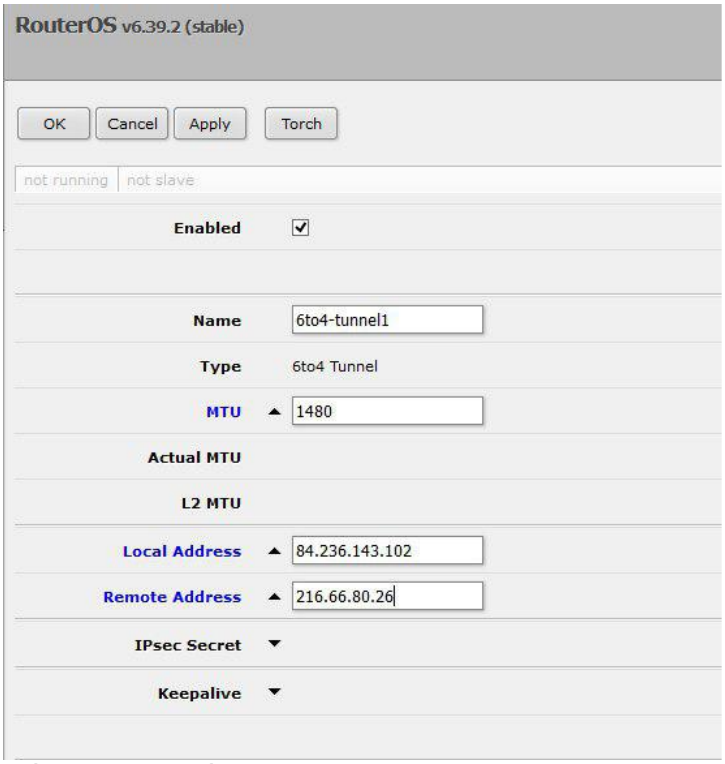



**Figura 5.11- Creación de la interfaz túnel en mikrotik a partir de nuestra Ip publica y la del servidor.**

El siguiente paso es usar la dirección Ipv6 que nos han asignado y añadirlo en nuestra lista de direcciones y asociarla a nuestro túnel.

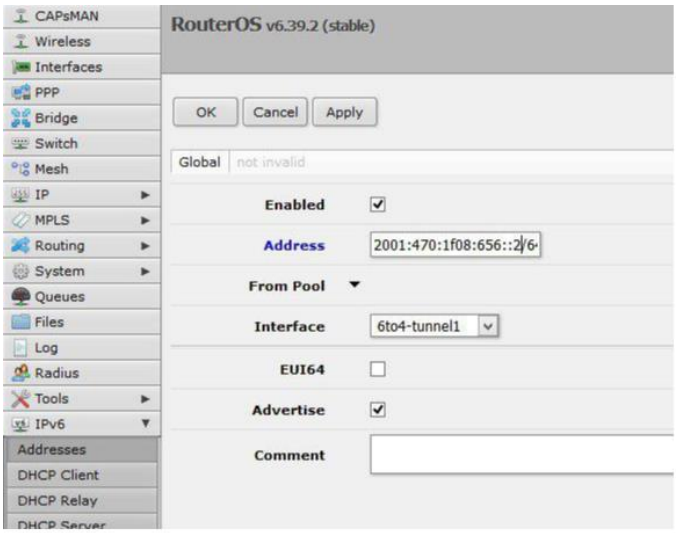



**Figura 5.12- Asignación de dirección Ipv6 en la interfaz de nuestro router, proporcionada por el servidor.**

Una vez hecho esto, ya se podría comprobar si ambos extremos tienen conectividad, para ello se usa la herramienta ping del router:

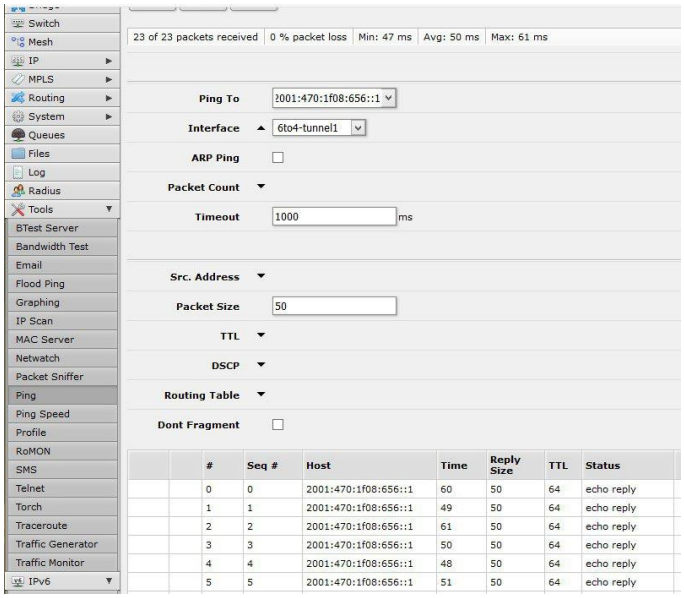



**Figura 5.13- Realización de pings con exite entre extremos del túnel.**

El paso siguiente será redirigir todo nuestro tráfico Ipv6 a la interfaz túnel, para ello hay que ir al menú Ipv6→ Routes→ Add Route y configurar la ruta de enlace predeterminada, que será el otro extremo del túnel.

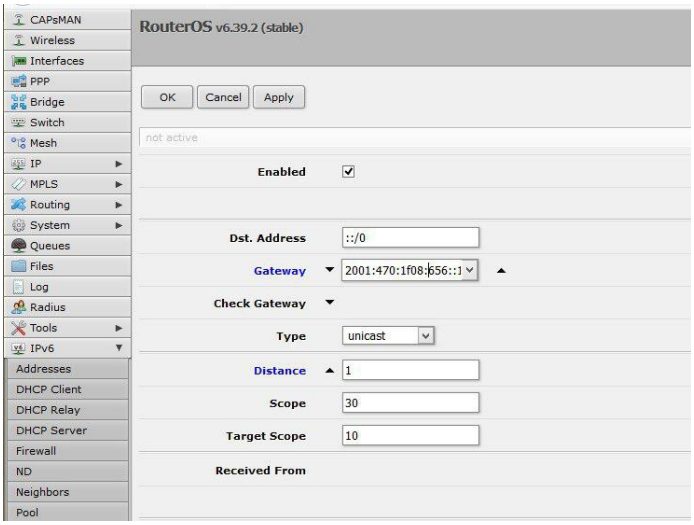



**Figura 5.14- Ruta default para redirigir nuestro tráfico ipv6 al túnel.**

Con este último paso se tendría configurado el túnel. Ahora, haciendo uso del rango de Ip para la delegación que nos ha dado el servidor, se crea el DHCPv6 como ya se hizo en el capítulo 4 de este proyecto. En resumen:

- Se crea una pool del rango dado 2001:470:6add::/48 del que sacaremos direcciones /64.
- Se crea el DHCP server donde se asigna a la interfaz Tunnel6to4 direcciones de la pool anterior.
- Se crea el DHCP Client donde se indica que comunique a los clientes su prefijo y su dirección

En la figura 5.15 mostrada abajo se aprecia cómo queda la configuración en el PC empleado una vez hechas estas configuraciones en el router:

```
C:\Windows\System32>ipconfig/renew

Configuración IP de Windows


Adaptador de Ethernet Ethernet:

   Sufijo DNS específico para la conexión. . :
   Dirección IPv6 . . . . . . . . . . . . : 2001:470:6add:0:b5a4:3d93:63d0:a41b
   Dirección IPv6 temporal. . . . . . . . : 2001:470:6add:0:a9c2:53ce:e94b:a11d
   Vínculo: dirección IPv6 local. . . . . : fe80::b5a4:3d93:63d0:a41b%12
   Dirección IPv4. . . . . . . . . . . . . . . . : 192.168.88.254
   Máscara de subred . . . . . . . . . . . . . . : 255.255.255.0
   Puerta de enlace predeterminada . . . . . . . : fe80::ce2d:e0ff:fe35:ef04%12
                                       192.168.88.1
```

**Figura 5.15- Obtención de dirección IPv6 global, obtenida del rango de direcciones dado por H.E y surtida por nuestro servidor DHCP.**

Una vez ya en el PC donde se desea que haya comunicación total con el exterior usando Ipv6, se harán diferentes pruebas para comprobar que todo funciona correctamente:

Ping a los dos extremos del túnel 6to4:

```
C:\Windows\System32>ping 2001:470:1f08:656::2

Haciendo ping a 2001:470:1f08:656::2 con 32 bytes de datos:
Respuesta desde 2001:470:1f08:656::2: tiempo=1ms
Respuesta desde 2001:470:1f08:656::2: tiempo<1m
Respuesta desde 2001:470:1f08:656::2: tiempo<1m
Respuesta desde 2001:470:1f08:656::2: tiempo<1m

Estadísticas de ping para 2001:470:1f08:656::2:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 1ms, Media = 0ms

C:\Windows\System32>ping 2001:470:1f08:656::1

Haciendo ping a 2001:470:1f08:656::1 con 32 bytes de datos:
Respuesta desde 2001:470:1f08:656::1: tiempo=64ms
Respuesta desde 2001:470:1f08:656::1: tiempo=49ms
Respuesta desde 2001:470:1f08:656::1: tiempo=47ms
Respuesta desde 2001:470:1f08:656::1: tiempo=48ms

Estadísticas de ping para 2001:470:1f08:656::1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 47ms, Máximo = 64ms, Media = 52ms
```

**Figura 5.16- Realización de pings con éxito a ambos extremos del túnel.**

Ping a Google en su página versión Ipv6 por dirección.

```
C:\Windows\System32>ping 2001:4860:4860::8888

Haciendo ping a 2001:4860:4860::8888 con 32 bytes de datos:
Respuesta desde 2001:4860:4860::8888: tiempo=49ms
Respuesta desde 2001:4860:4860::8888: tiempo=50ms
Respuesta desde 2001:4860:4860::8888: tiempo=51ms
Respuesta desde 2001:4860:4860::8888: tiempo=50ms

Estadísticas de ping para 2001:4860:4860::8888:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 49ms, Máximo = 51ms, Media = 50ms
```

**Figura 5.17- Realización de pings con éxito a google.**

Ping a Google en su página versión Ipv6 por nombre.

```
C:\Windows\System32>ping ipv6.google.com

Haciendo ping a ipv6.l.google.com [2a00:1450:4003:802::200e] con 32 bytes de datos:
Respuesta desde 2a00:1450:4003:802::200e: tiempo=75ms
Respuesta desde 2a00:1450:4003:802::200e: tiempo=76ms
Respuesta desde 2a00:1450:4003:802::200e: tiempo=78ms
Respuesta desde 2a00:1450:4003:802::200e: tiempo=73ms

Estadísticas de ping para 2a00:1450:4003:802::200e:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 73ms, Máximo = 78ms, Media = 75ms
```

**Figura 5.18- Realización de pings con éxito a Google en su versión Ipv6.**

Por último, si se emplean páginas especializadas en monitorizar nuestro tráfico y ver si la configuración elegida para usar Ipv6 es correcta se obtienen los siguientes resultados (Figuras 5.18 y 5.20):

Pruebas de Ipv6

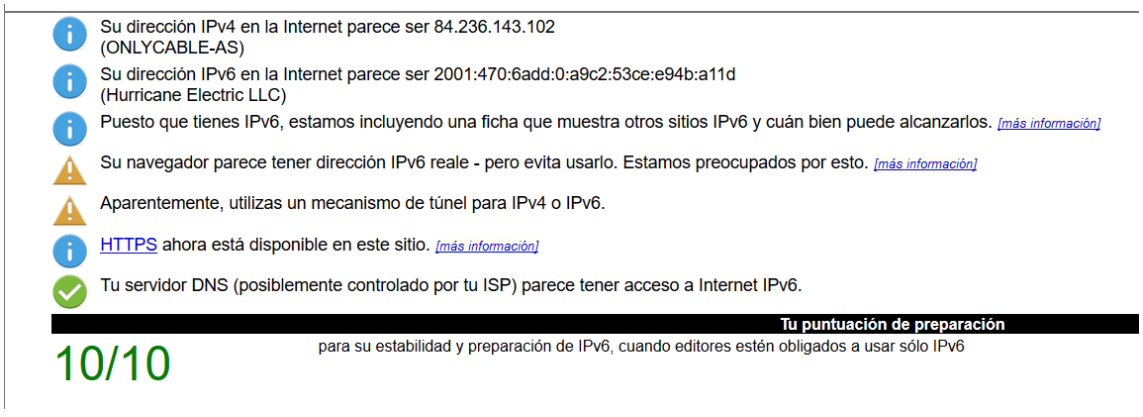



**Figura 5.19- Comprobamos que nuestra configuración de ipv6 se considera completa.**

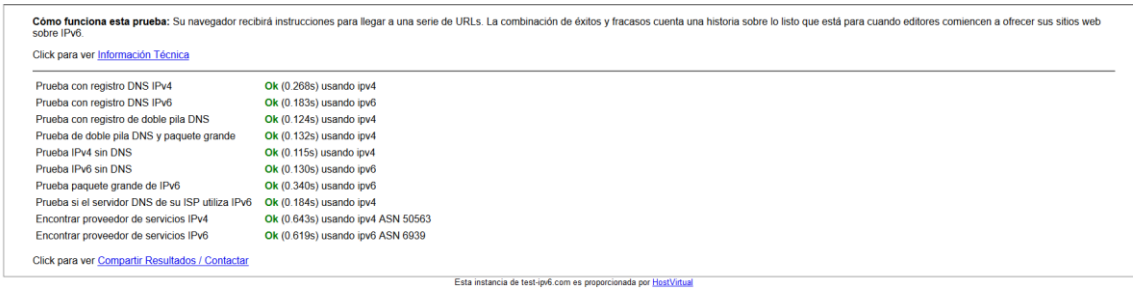



**Figura 5.20- Comprobamos que nuestra configuración de ipv6 se considera completa.**

Como se puede ver, la página dice que se está usando un mecanismo de túnel en la red.

### 5.3.1 Ventajas y desventajas

#### 6over4

Ventajas: 6to4 es un sistema que permite enviar paquetes Ipv6 sobre redes IPv4 obviando la necesidad de configurar túneles manualmente. Fue diseñado para permitir conectividad Ipv6 sin la cooperación de los proveedores de Internet.

Este sistema puede funcionar en un router, proveyendo conectividad a toda una red, o en una máquina en particular.

Desventajas: 6over4 confía en la disponibilidad de multicast de IPv4, que no suele ser implementado en la infraestructura IPv4 (multicast es casi tan reciente como Ipv6). 6over4 tiene poco uso práctico y no está soportado por los sistemas operativos más comunes.

ISATAP es una alternativa más completa que 6over4 y no confía en el multicast IPv4.

#### 6to4

Ventajas: Puede ser usado en la fase inicial del despliegue de Ipv6, como mecanismo temporal mientras el ISP despliega Ipv6 nativo. 6to4 es un sistema que permite enviar paquetes Ipv6 sobre redes IPv4 obviando la necesidad de configurar túneles manualmente. Fue diseñado para permitir conectividad Ipv6 sin la cooperación de los proveedores de Internet.

A través de una sola dirección IPv4 pública se puede configurar un túnel para varios hosts del sitio 6to4.

Desventajas: El relay 6to4 se encuentra bajo el control de un tercero (anuncia el prefijo 2002::/16). Es muy difícil controlar el tráfico que circula a través él. Vulnerable a ataques de DoS y Spoofing.

### ISATAP

Ventajas- Tiene algunas ventajas respecto a 6over4, ya que no necesita multicast IPv4 y que soluciona los problemas que se dan cuando una misma organización no tiene toda su red en el mismo lugar.

ISATAP está implementado en Microsoft Windows Vista, Windows XP, Windows Mobile y en algunas versiones de Cisco IOS.

Desventajas- ISATAP también conlleva los mismos riesgos de seguridad que 6to4: el enlace virtual IPv4 debe definirse con cuidado en el perímetro de la red, para que los hosts IPv4 externos no intenten ser parte del enlace ISATAP. Normalmente se puede evitar asegurando que el protocolo 41 no pueda atravesar el cortafuegos.

## 5.3. NAT64

NAT64 es un mecanismo que permite a hosts Ipv6 comunicarse con servidores IPv4. El servidor NAT64 dispone de al menos una dirección IPv4 y un segmento de red Ipv6 de 32-bits. El cliente Ipv6 construye la dirección Ipv6 destino utilizando el rango anterior de 96 bits más los 32 bits de la dirección IPv4 con la que desea comunicarse, enviando los paquetes a la dirección resultante. El servidor NAT64 crea entonces un mapeo de NAT entre la dirección Ipv6 y la dirección IPv4, permitiendo la comunicación.

Un entorno de NAT64 simplista puede verse como un dispositivo de red (un router, por ejemplo) con al menos dos interfaces. Uno de los interfaces está conectado a la red IPv4, y el otro a la red Ipv6. La red estará configurada de modo que los paquetes de la red Ipv6 a la red IPv4 son encaminados a través de este router. El router realizará todas las traducciones necesarias para transferir paquetes de la red Ipv6 a la red IPv4, y viceversa.

La traducción no es simétrica, dado que el espacio de direcciones Ipv6 es mucho mayor que el de direcciones IPv4, por lo que no es posible una traducción una-una. Para poder llevar a cabo la traducción, el equipo NAT64 debe mantener un mapeo de direcciones Ipv6 a IPv4 (es decir, mantiene estado). Este tipo de mapeo de direcciones se configura estáticamente por los administradores del sistema o, habitualmente, se crea automáticamente cuando llega el primer paquete Ipv6 al servidor NAT64. Después de que se haya creado este flujo, los paquetes pueden pasar en ambas direcciones.

### 5.3.1 Ejemplo NATPT

La maqueta se muestra en la figura 5.21

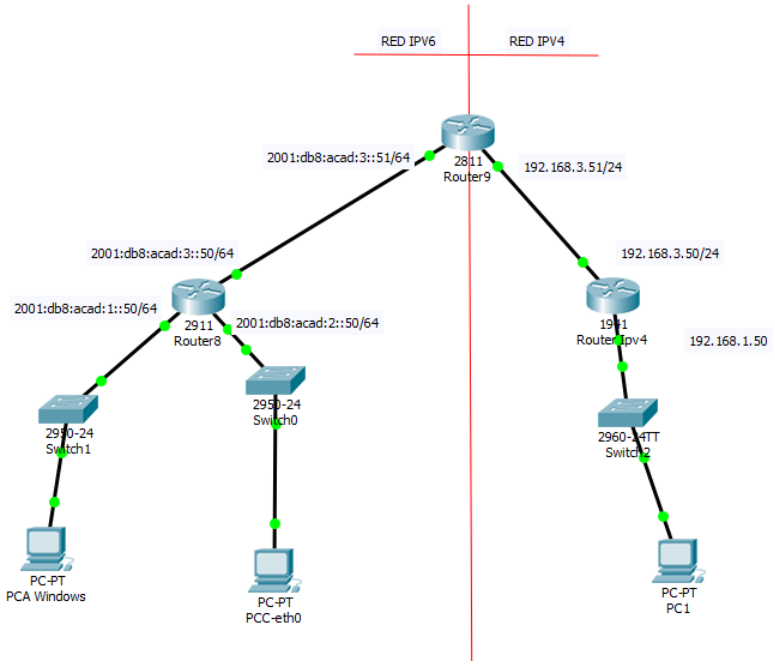



**Figura 5.21- Maqueta sobre Cisco Packet Tracer de una red con mecanismo de adaptación NATPT.**

| | Interfaz | Ipv4 | Mascara v4 | Ipv6 | Mascara v6 |
|---|---|---|---|---|---|
| **R.Ipv6(R8)** | G0/0 | NA | NA | 2001:db8:acad:1::50 | /64 |
| | G0/1 | NA | NA | 2001:db8:acad:2::50 | /64 |
| | G0/2 | NA | NA | 2001:db8:acad:3::50 | /64 |
| **R.Ipv4(RouterIpv4)** | G0/0 | 192.168.3.50 | /24 | NA | NA |
| | G0/1 | 192.168.1.50 | /24 | NA | NA |

| | | | | | |
|---|---|---|---|---|---|
| **R.NatPT(R9)** | Fa0/0 | NA | NA | 2001:db8:acad:3::51 | /64 |
| | Fa0/1 | 192.168.3.51 | /24 | NA | NA |

**Figura 5.22- Datos de nuestra red ejemplo.**

### Configuración Router Ipv6

int g0/0

Ipv6 address 2001:db8:acad:1::50/64

Ipv6 enable

no shutdown

!

int g0/1

Ipv6 address 2001:db8:acad:2::50/64

Ipv6 enable

no shutdown

!

int g0/2

Ipv6 address 2001:db8:acad:3::50/64

Ipv6 enable

no shutdown

!

### Ipv6 unicast-routing

Ipv6 route ::/0 2001:db8:acad:3::51

**Configuración Router NAT-PT**

**Ipv6 unicast-routing**

int f0/0

Ipv6 address 2001:db8:acad:3::51/64

Ipv6 enable

**Ipv6 nat**

no shutdown

!

int f0/1

ip address 192.168.3.51 255.255.255.0

**Ipv6 nat**

no shutdown

!

ip route 192.168.1.0 255.255.255.0 192.168.3.50

Ipv6 route 2001:db8:acad:1::/64 2001:db8:acad:3::50

Ipv6 route 2001:db8:acad:2::/64 2001:db8:acad:3::50

Ipv6 route ::/0 2001:db8:acad:3::50

Ipv6 nat prefix 2000:12::/96

Ipv6 nat v4v6 source 192.168.1.1 2000:12::2

Ipv6 nat v4v6 source 192.168.1.50 2000:12::4

Ipv6 nat v4v6 source 192.168.3.50 2000:12::3

Ipv6 nat v6v4 source 2001:db8:acad:1::1 20.0.0.2

Ipv6 nat v6v4 source 2001:db8:acad:2::1 20.0.0.3

Ipv6 nat v6v4 source 2001:db8:acad:3::50 20.0.0.4

Ipv6 nat v6v4 source 2001:db8:acad:1::50 20.0.0.5

Ipv6 nat v6v4 source 2001:db8:acad:2::50 20.0.0.6

Con estos commandos se ha conseguido:

- Activar las interfaces y colocado sus respectivas direcciones
- Activar la Nat en ambas interfaces (en la interfaz v6 hay que especificar que es una Nat Ipv6)
- Crear las rutas de direccionamiento para Ipv6 e Ipv4
- Ipv6 nat prefix 2000:12::/96 es una forma de indicar que cuando veamos una dirección que empieza por 2000:12 sepamos que es una dirección originalmente Ipv4
- A cada dirección origen (v4 y v6) se le ha asociado una otra dirección del protocolo contrario

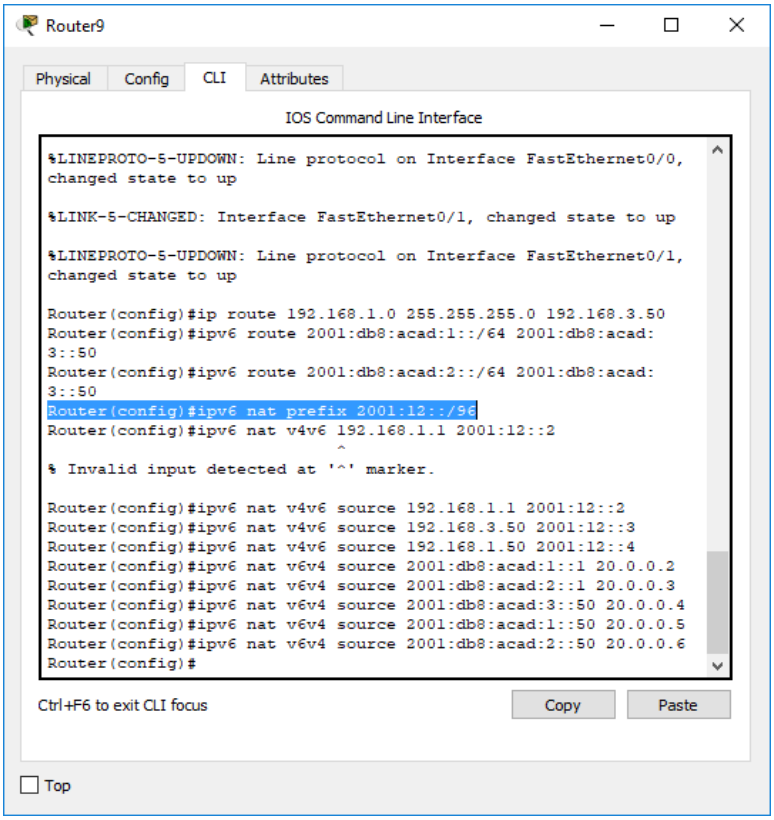



**Figura 5.23- Comandos para la traducción de las direcciones de ipv4 a Ipv6 y viceversa.**

Ahora se comprueba si la configuración ha sido correcta y como se refleja el proceso de traducción.

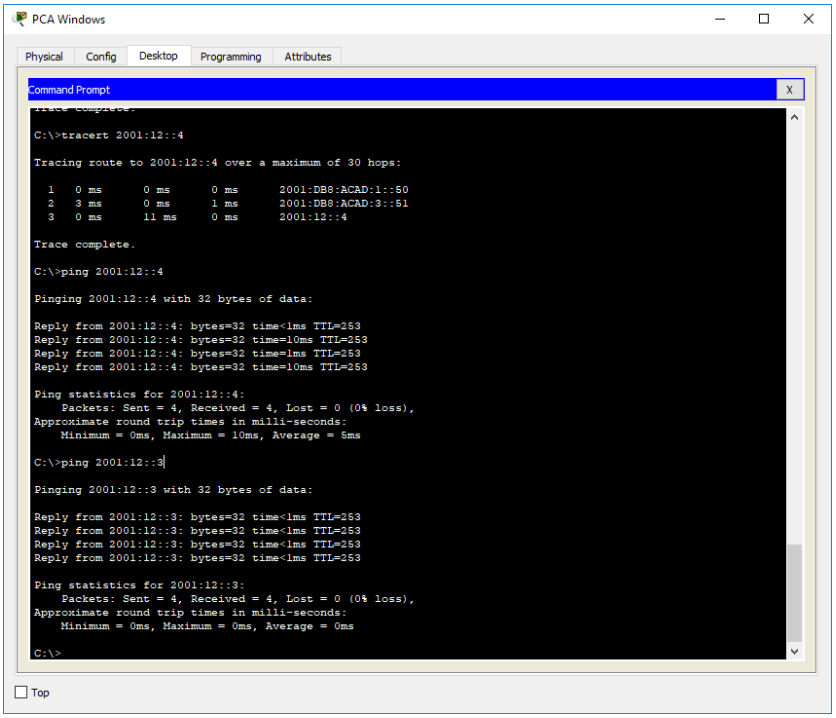



**Figura 5.24 Resultado exitoso de un ping de la red 2001:db8:acad:1::/64 a 192.168.1.1, 192.168.1.50 y a 192.168.3.**

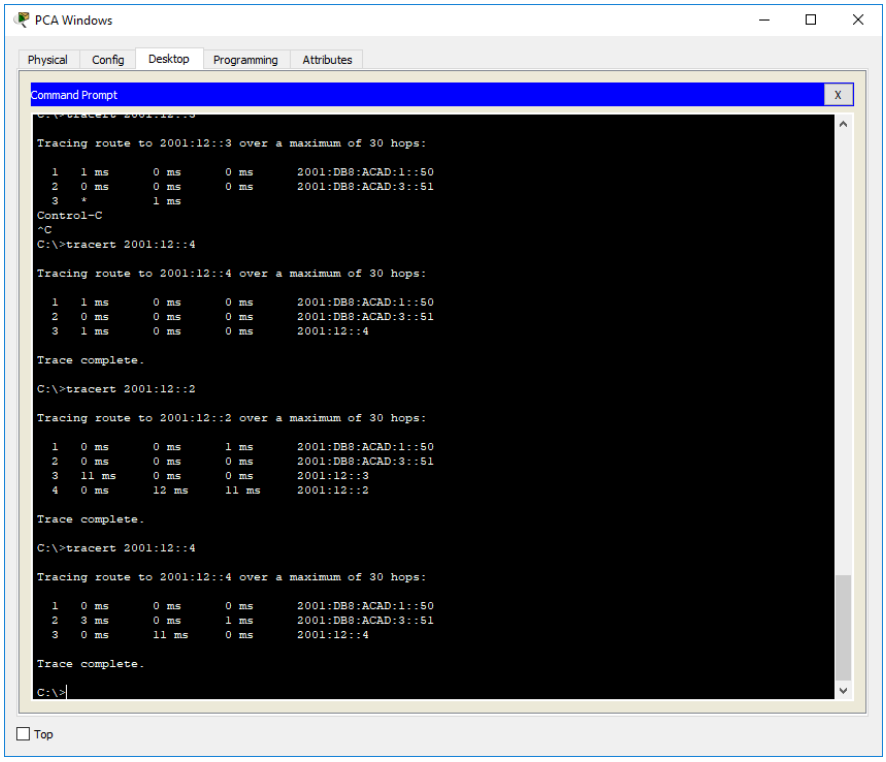



**Figura 5.26- Ping exitoso desde la red 192.168.1.0 a 2001:db8:acad:1::1, 2001:db8:acad:2::1,2001:db8:acad:1::50, 2001:db8:acad:2::50 y 2001:db8:acad:3::50 (20.0.0..2, 20.0.0.3, 20.0.0.4, 20.0.0.5 y 20.0.0.6).**

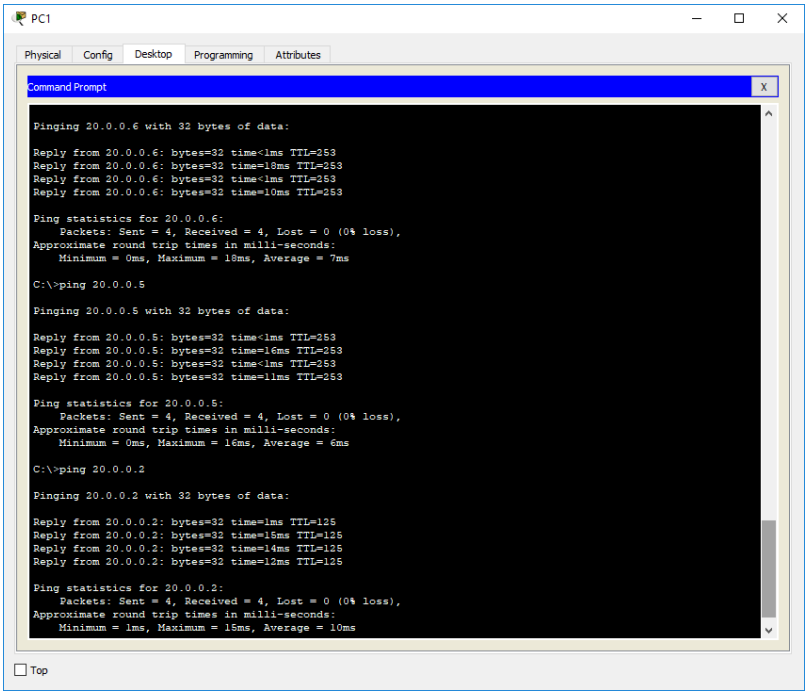



**Figura 5.27- Ping exitoso a 20.0.0.3, 20.0.0.5 y 20.0.0.6.**

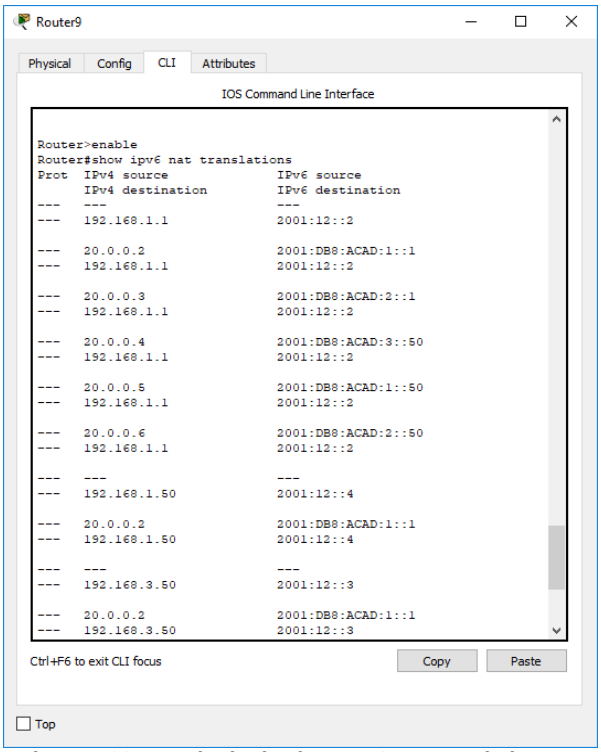



**Figura 5.28- Resultado de show Ipv6 nat translations**

Este comando (figura 4.26) permite ver la tabla de traducciones. Qué direcciones Ipv6 se le ha asignado a las direcciones Ipv4 y viceversa.

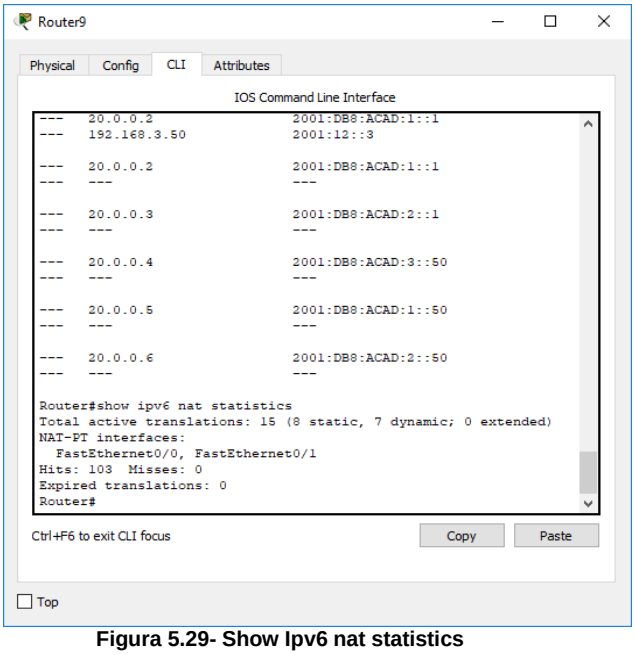



**Figura 5.29- Show Ipv6 nat statistics**

Este comando (figura 4.27) permite llevar un registro del número de traducciones que se han hecho en el router. En el caso de estudio, al haber hecho un gran número de pings y tracert muestra 15 traducciones.

### 5.3.2 Ventajas y desventajas

Ventajas: NAT64 permite a múltiples nodos sólo-Ipv6 para compartir una dirección IPv4 para acceder a Internet IPv4. Buena solución si no se requiere IPv4 en el cliente.

Desventajas: Es la peor solución puesto que la traducción no es perfecta y requiere soporte de ALGs (Application Layer Gateway), como en el caso de los NATs IPv4. Algunas aplicaciones no funcionan (como Skype). Si la página o aplicación usa direcciones literales no funciona al igual que si se usan socket APIs. Sólo traduce paquetes unicast con tráfico TCP, UDP e ICMP.

La siguiente figura[9] es un ejemplo de aplicaciones y servicios que no funcionaban con nat64 a finales de 2016. La mayoría de ellos son arreglados con 464XLAT (una técnica que trata de dar acceso ipv4 a redes con uso puramente Ipv6).

| App Name | Functionality | Version | **464XLAT Fixed** |
|---|---|---|---|
| connection tracker | Broken | NA | NA |
| DoubleTwist | Broken | 1.6.3 | YES |
| Go SMS Pro | Broken | NA | YES |
| Google Talk | Broken | 4.1.2 | YES |
| Google+ | Broken | 3.3.1 | YES |
| IP Track | Broken | NA | NA |
| Last.fm | Broken | NA | YES |
| Netflix | Broken | NA | YES |
| ooVoo | Broken | NA | YES |
| Pirates of the Caribean | Broken | NA | YES |
| Scrabble Free | Broken | 1.12.57 | YES |
| Skype | Broken | 3.2.0.6673 | YES |
| Spotify | Broken | NA | YES |
| Tango | Broken | NA | YES |
| Texas Poker | Broken | NA | YES |
| TiKL | Broken | 2.7 | YES |
| Tiny Towers | Broken | NA | YES |
| Trillian | Broken | NA | YES |
| TurboxTax Taxcaster | Broken | NA | |
| Voxer Walkie Talkie | Broken | NA | YES |
| Watch ESPN | Broken | 1.3.1 | |
| Zynga Poker | Broken | NA | YES |

**Figura 5.29- Lista servicios que no funcionaban en redes Ipv6-only sin 464XLAT a finales de 2016.**

[9] Tested application worked or didn´t work in the Ipv6-only networks with NAT64/DNS64 https://www.slideshare.net/apnic/464xlat-tutorial

# Capítulo 6. Estudio de otros servicios sobre PT 7.1

## 6.1. IoT

Aprovechando que las últimas versiones de Cisco Packet Tracer comienzan a introducir poco a poco elementos para simular el comportamiento de varios de dispositivos relacionados con IoT[10], en este apartado se muestra un ejemplo sencillo de una red doméstica donde se han incluido varios elementos programables y con posibilidad de una conexión total.

Los elementos a destacar en esta maqueta serán:

- Ventana programable
- Puerta programable
- Lámpara programable
- Servidor IoT
- MCU y SBC boards. Funcionalmente equiparables a un arduino/raspberry.

Dentro de la maqueta varios de los elementos necesitarán ser programados para que cumplan la funcionalidad que se desea reflejar en el ejemplo. Para ello se usarán dos lenguajes: Python y Javascript.

Debido a la naturaleza del proyecto, realizar programas muy complejos no es el objetivo de la maqueta, sino esbozar las múltiples posibilidades que ofrece la IoT y como lo refleja se refleja en el simulador.

En la maqueta de ejemplo se configurará un servidor que tendrá registrados todos los elementos IoT de la casa, a la vez que hace de DHCP para dotarles de conexión con la red interna y a internet. Después se programará

[10] Internet of Things - https://www.cisco.com/c/en/us/solutions/internet-of-things/overview.html


**Figura 6.1-Maqueta "Proyecto maqueta IoT" donde configuraremos elementos básicos de IoT.**

Los pasos:

**Paso 1-** Seleccionar todos los elementos de la maqueta y configurar la red.

Server- getaway 192.168.1.1

Ip eth0 192.168.1.2

Servidor Iot- dirección 192.168.1.2

Usuario "Iot "

pass "cisco"

Servicio DHCP- start en 192.168.1.10

Max ips- 10

Dirección geta 192.168.1.1

En cada elemento de la maqueta se obtiene ip por dhcp y se conecta remotamente con el servidor Iot introduciendo la ip del servidor usuario y pass.

(Si el botón connect cambia a refresh es que se ha conectado correctamente)

En este punto se puede acceder a “IOT Monitor” en el server y controlar ventana y puerta. Se puede acceder desde el pc al servidor desde el pc.

**Paso 2-** Introducir las SBC Board y programar botón puerta.

Pinchar en el botón, seleccionar “advanced” y programming.

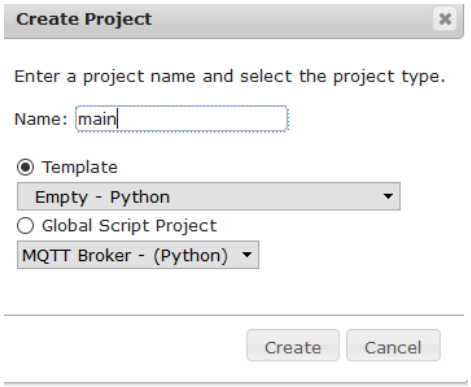



**Figura 6.2-Creación de un proyecto dentro de una SBC Board.**

El programa estará infinitamente en el bucle while e irá leyendo el puerto uno de la SBC y guardando el valor en la variable value.

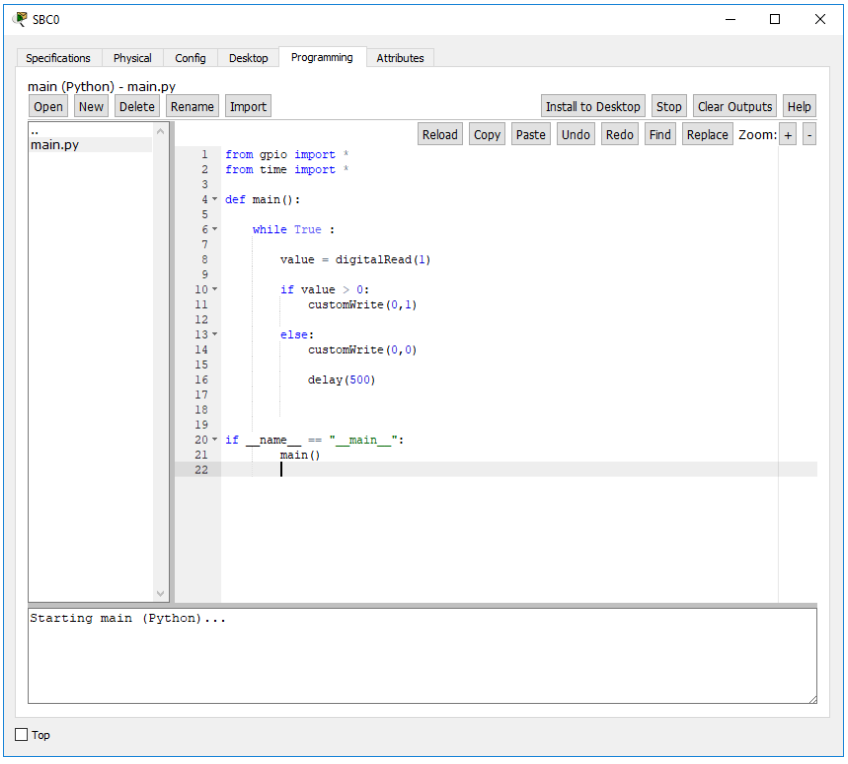



**Figura 6.3- Código programado dentro del proyecto main (en Pyton) para el control de la puerta.**

Si el valor es mayor que 0 (si viene 1 del botón) se escribirá en el puerto 0 (puerta) un 1 (abrir). Si lo que recibe del botón es un 0 le enviará un 0 a la puerta (cerrar).
Después de hacer un ciclo de while esperara 500ms para volver a realizar otro.
Al terminar de realizar el programa darle al botón “run”.
Según vayamos a conectar más de un dispositivo IoT a la SBC Board necesitaremos ampliar el número de Slots digitales.

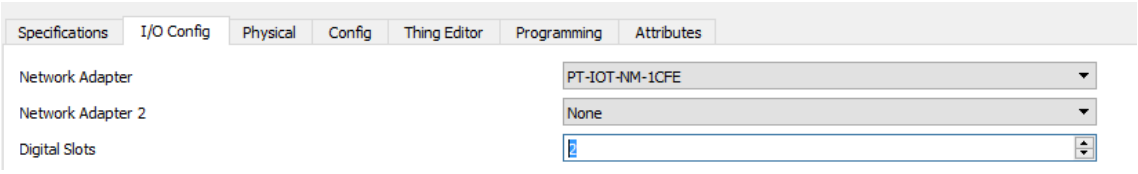



**Figura 6.4- Si necesitamos conectar varios dispositivos IoT hay que aumentar los slots.**

**Pasos 3 y 4-** Con ventana y lámpara usaremos códigos parecidos en sus respectivas SBC para estipular como se abren/cierran o se encienden/ apagan/.

### Automatización del conjunto ventana-puerta-lámpara

Para la automatización se realizará un programa en JavaScript que controle los tres dispositivos y vaya enviando unos o ceros según convenga por los slots conectados a los dispositivos.

El código comienza con variables que almacenan horas, minutos, segundos actuales y el estado de los puertos conectados a nuestros aparatos IoT.

Básicamente es un bucle que siempre se repite donde se actualizan las variables y en función de ellas, según la hora que sea las puertas y ventanas se abrirán/cerrarán y las lámparas se encenderán/ apagarán/ funcionarán a intensidad media automáticamente.

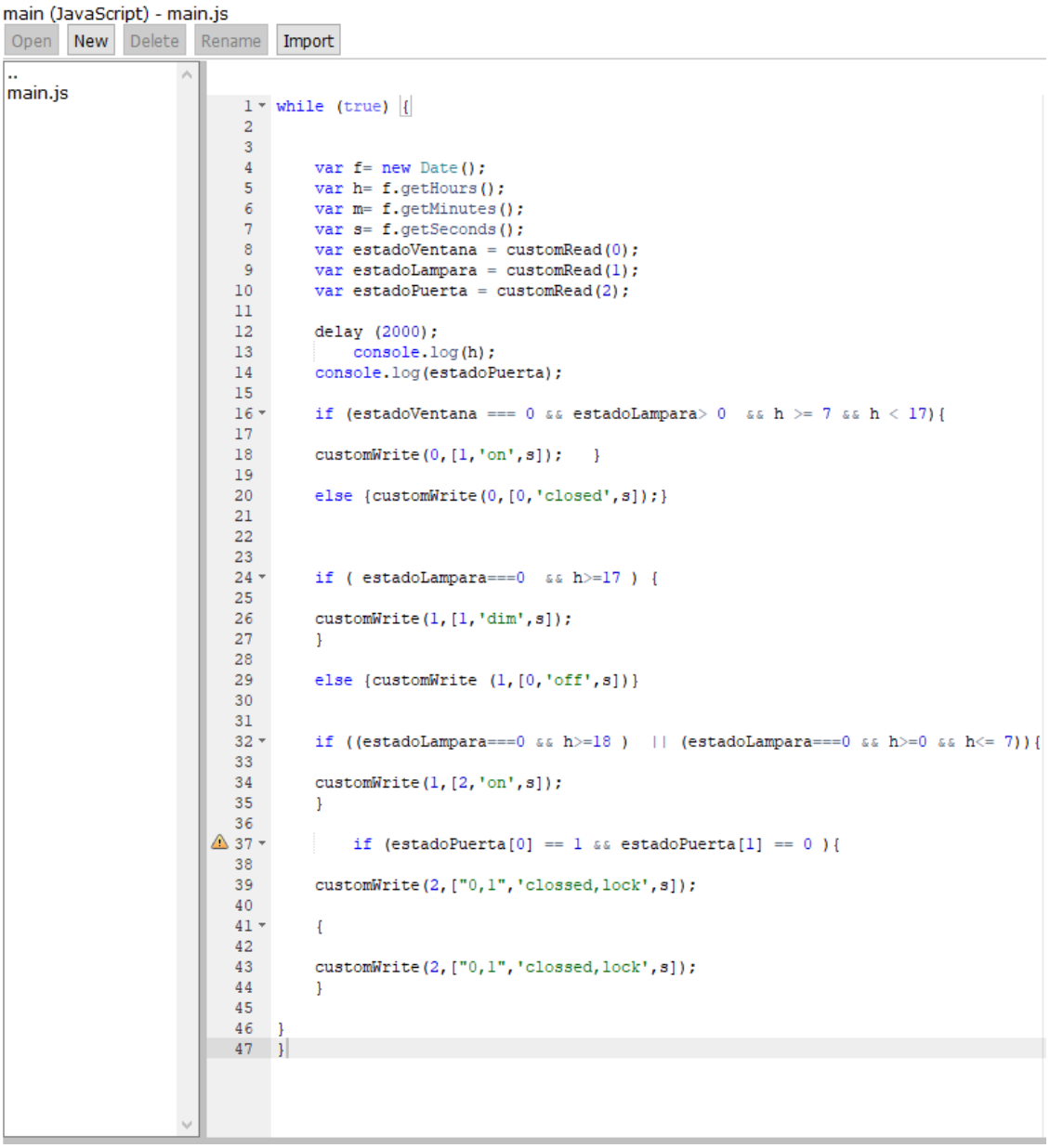



**Figura 6.5- Código JavaScript de nuestro programa de automatización.**

```
if (estadoVentana==0&& h>7 && h<=17){
                customWrite(0,[1,'open',s]);   }
        else {customWrite(0,[0,'closed',s]);}
```

Si el reloj marca entre las 7 y las 17 y la ventana está cerrada, se abre la ventana. Si no cerrará la ventana.

```
if ( estadoLampara===0  && h>=17 ) {
        customWrite(1,[1,'dim',s]);
        }
        else {customWrite (1,[0,'off',s])}
```

Si la lámpara está apagada y son más de las 17h se pone a potencia media, si no se apaga:

```
        if ((estadoLampara===0 && h>=18 )  || (estadoLampara===0 && h>=0 &&
h<= 7)){

        customWrite(1,[2,'on',s]);
        }
```

Si la lámpara está apagada y son más de las 18 **o** la lámpara está apagada y estamos entre las 00 y las 7 la lámpara se enciende.

La puerta es un caso especial pues guarda 2 datos y hay que tratarlo como un array (estado puerta, estado cerradura) así que las comprobaciones se realizarán como si fuera un array

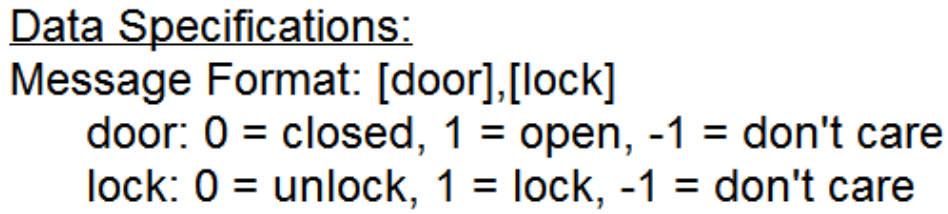



**Figura 6.6- Formato de mensaje para comunicarse con la puerta.**

```
if (estadoPuerta[0] == 1 && estadoPuerta[1] == 0 && h>17 ){
customWrite(2,["0,1",'clossed,lock',s]); }
```

Si la puerta está abierta y la cerradura desbloqueada más allá de las 17. La puerta se cierra.

## 6.2 Per-VLAN Spanning Tree (PVST)

En este ejemplo se partirá de una red Ipv4 ya configurada y una estructura de Spanning-tree que se debe cambiar para satisfacer las necesidades del diseño.

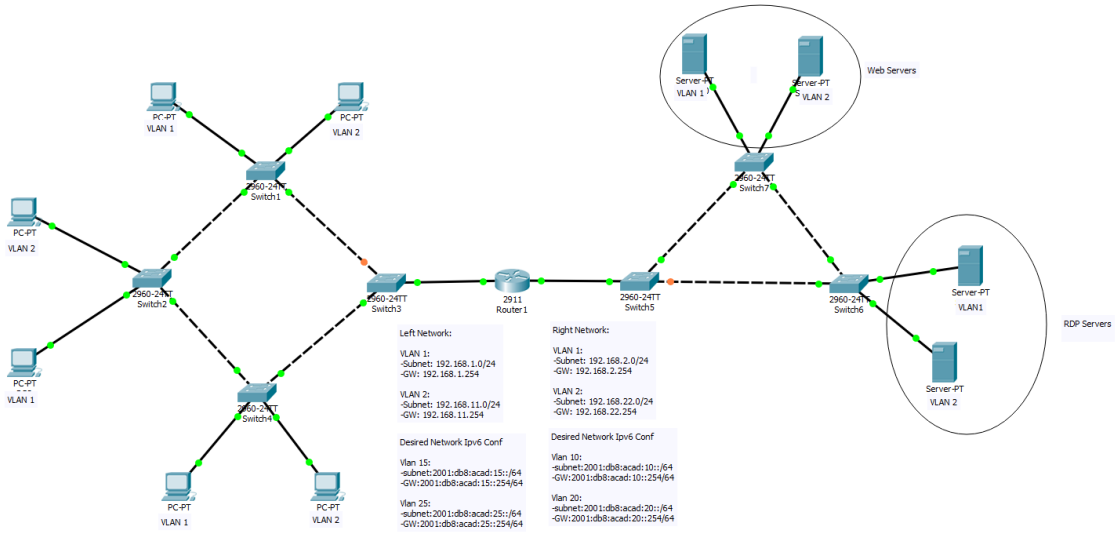



**Figura 6.7- Maqueta "STP-Ipv4 2411" donde buscaremos realizar un balance de tráfico por Vlan.**

En este ejemplo se busca que el PC perteneciente a la Vlan1 en el Switch1 tenga designada la ruta más corta a los servidores web.

En el punto de partida de la maqueta, como se ve en la figura, para llegar hasta allí, pasa por el switch2, switch3 y switch4 y de ahí llegar al servidor. Se buscará cambiar la configuración del protocolo en los switches para que se llegue por el camino más corto.

Para ellos se realizarán los siguientes pasos:

1) Comprobar el estado inicial del protocolo PSVT y la configuración en los switches.

2) Realizar los cambios oportunos en los switches para que la vlan siga el camino que se ha elegido.

3) Comprobar que existe conectividad entre los elementos de la red con Ipv4.

Esta es la configuración inicial de la red izquierda (Figuras 6.8, 6.9 y 6.10):

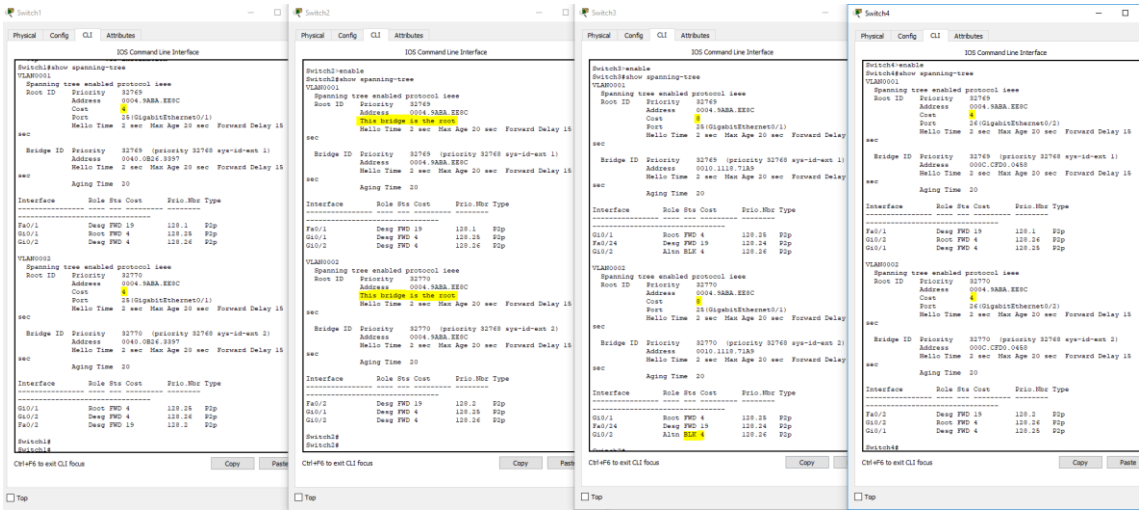
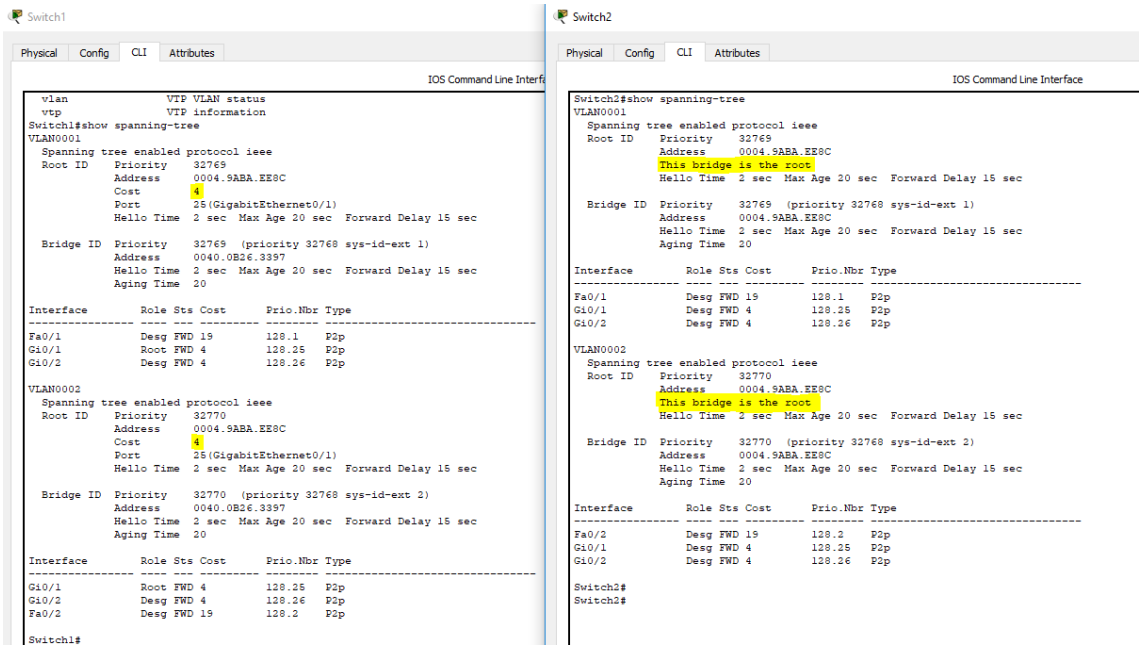


**Figura 6.8- Estado inicial de la configuración PSVT en los switches.**



```
Switch1
Physical  Config  CLI  Attributes

IOS Command Line Interface

  vlan              VTP VLAN status
  vtp               VTP information
Switch1#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID     Priority    32769
              Address     0004.9ABA.EE8C
              Cost        4
              Port        25(GigabitEthernet0/1)
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    32769  (priority 32768 sys-id-ext 1)
              Address     0040.0B26.3397
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/1            Desg FWD 19        128.1    P2p
Gi0/1            Root FWD 4         128.25   P2p
Gi0/2            Desg FWD 4         128.26   P2p

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID     Priority    32770
              Address     0004.9ABA.EE8C
              Cost        4
              Port        25(GigabitEthernet0/1)
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    32770  (priority 32768 sys-id-ext 2)
              Address     0040.0B26.3397
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi0/1            Root FWD 4         128.25   P2p
Gi0/2            Desg FWD 4         128.26   P2p
Fa0/2            Desg FWD 19        128.2    P2p

Switch1#
```

```
Switch2
Physical  Config  CLI  Attributes

IOS Command Line Interface

Switch2#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID     Priority    32769
              Address     0004.9ABA.EE8C
              This bridge is the root
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    32769  (priority 32768 sys-id-ext 1)
              Address     0004.9ABA.EE8C
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/1            Desg FWD 19        128.1    P2p
Gi0/1            Desg FWD 4         128.25   P2p
Gi0/2            Desg FWD 4         128.26   P2p

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID     Priority    32770
              Address     0004.9ABA.EE8C
              This bridge is the root
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    32770  (priority 32768 sys-id-ext 2)
              Address     0004.9ABA.EE8C
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/2            Desg FWD 19        128.2    P2p
Gi0/1            Desg FWD 4         128.25   P2p
Gi0/2            Desg FWD 4         128.26   P2p

Switch2#
Switch2#
```

**Figura 6.9- Estado inicial de la configuración PSVT en los switches 1 y 2**

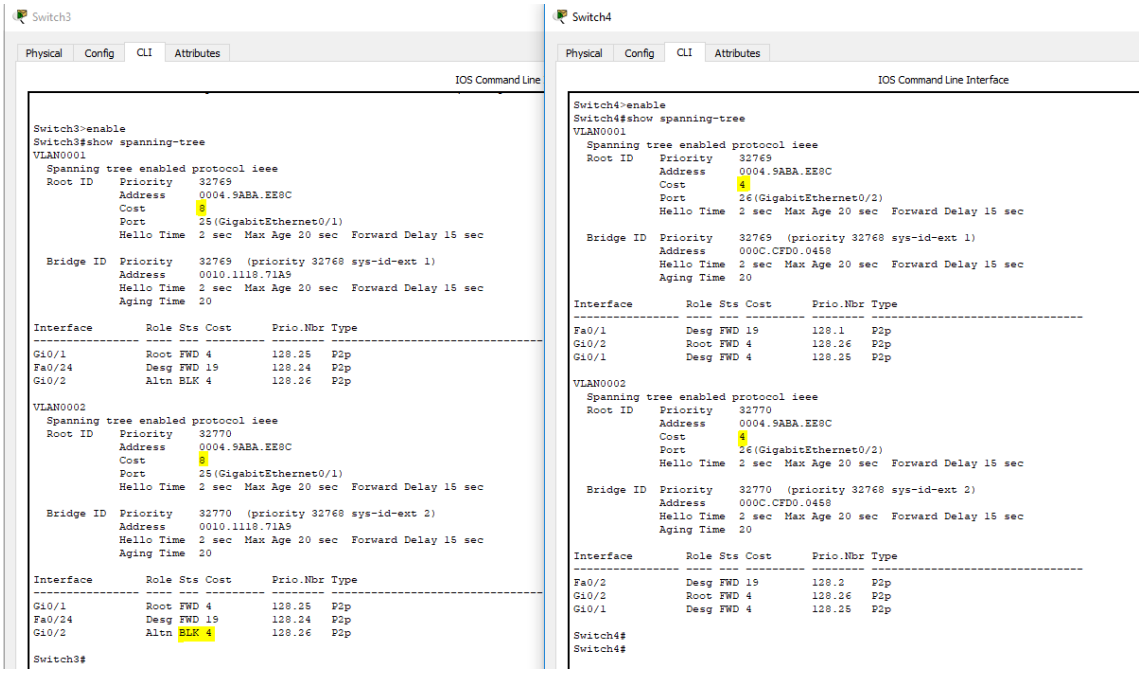



**Figura 6.10- Estado inicial de la configuración PSVT en los switches 3 y 4.**

Como se puede ver, el switch2 está configurado como root de ambas vlanes, todos los demás switches tienen designado un puerto root (Sw1& Sw3→ g0/1, Sw4→g0/1) mientras todos los demás puertos están en estado fordwarding salvo el puerto g0/2 del Switch3 que está en estado bloqueado (evitando así el bucle).

El estado de la red derecha es casi simétrico.

Para realizar los cambios requeridos se cambiarán los puertos bloqueados en la red izquierda. Esta vez el elegido será g0/1 del switch3 permitiendo acortar el número de saltos hasta la red de servidores y cumpliendo el objetivo para la Vlan1.

Se usaran los siguientes comandos en esta maqueta :

```
spanning-tree mode pvst
spanning-tree vlan X priority Y
spanning-tree vlan vlan_ID root primary
spanning-tree vlan vlan_ID root secondary
```

Donde Y debe ser un incremento de 4096 entre 0 y 61440.

Como se vió anteriormente para llegar a los servidores el tráfico sigue el camino Sw1 → Sw2 → Sw4 → Sw3→ Servidores (Figura 6.8).

SPT bloquea por defecto algunos enlaces entre switches para evitar que el tráfico de multidifusión se propague indefinidamente por la red hasta colapsar los servidores. En la configuración de inicio el enlace bloqueado es el Gi0/2 del Sw 4(Figura 6.10).

Lo que se hará será cambiar los valores de la prioridad para la Vlan 1, forzando a que el sw4 tenga el valor más alto y el sw1 el más bajo, convirtiéndolo así en root para esa vlan.

Los valores de prioridad antes de realizar los cambios en los valores.

| Switch | Switch 1 | Switch 2 | Switch 3 | Switch 4 |
|---|---|---|---|---|
| Prioridad | 32769 | 32769 | 32769 | 32769 |

Los valores de prioridad después de realizar los cambios en los valores.

| Switch | Switch 1 | Switch 2 | Switch 3 | Switch 4 |
|---|---|---|---|---|
| Prioridad | 24577 | 32769 | 61441 | 32769 |

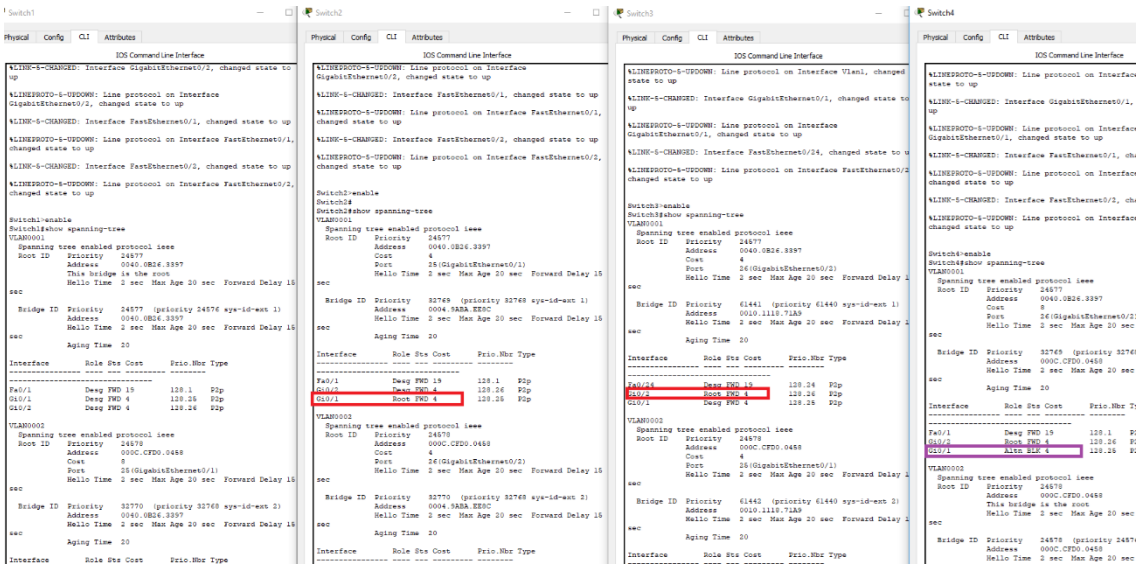


**Figura 6.11- Resultado del camndo show spanning-tree en todos los Switches tras el cambio de valores.**

Para la vlan 2 se usó el comando `spanning-tree vlan vlan_ID root primary` en lugar de tocar los valores, este comando ajusta el valor de prioridad,

decreciéndolo en 8192 y haciendo que sea el más bajo (Sw4) y así se proclame root.

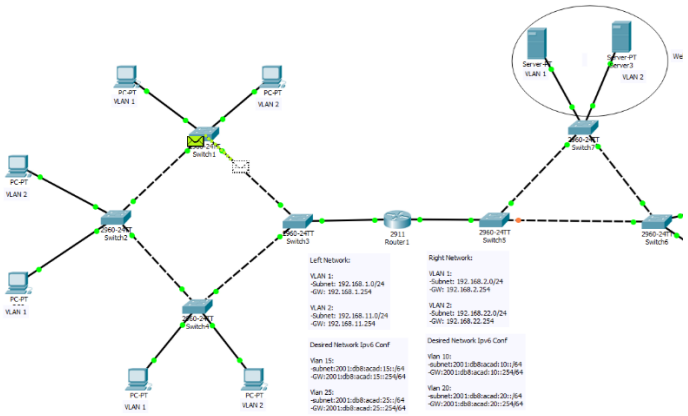



**Figura 6.12- Se muestra como el curso de tráfico ha cambiado como se esperaba, usando el enlace G0/2 del Sw1.**

# Capítulo 7 Configuración de servicios para el usuario final en Ipv4 e Ipv6

## 7.1 Servidor RADIUS en una red inalámbrica (Ipv4).

Es un protocolo de autenticación y autorización para aplicaciones de acceso a la red o movilidad IP.

Cuando se realiza la conexión con un ISP mediante módem, DSL, cablemódem, Ethernet o Wi-Fi, se envía una información que generalmente es un nombre de usuario y una contraseña. Esta información se transfiere a un dispositivo Network Access Server (NAS) sobre el protocolo PPP, quien redirige la petición a un servidor RADIUS[11] sobre el protocolo RADIUS. El servidor RADIUS comprueba que la información es correcta utilizando esquemas de autenticación como PAP, CHAP o EAP. Si es aceptado, el servidor autorizará el acceso al sistema del ISP y le asigna los recursos de red como una dirección IP, y otros parámetros como L2TP, etc.

Los datos a tener en cuenta para la configuración de este servidor son los siguientes:

La dirección Ip del servidor radius: 192.168.3.100

La dirección Ip del servicio (Cliente): 192.168.3.254

Clave para la encriptación de mensajes entre servidor RADIUS y el NAS: “Itelar09”

Usuarios admitidos: testuser(password:”testpw”) e it5 (password:”Itelar09”). Y posteriormente: “josec”(password:”Itelar09”).

Los pasos para la creación del servidor en nuestra red WAN serán los siguientes:

1) Descargar el ejecutable del servidor RADIUS

[11] FreeRADIUS - https://freeradius.org/documentation/

2) Configurar la interfaz de red del servidor con la dirección 192.168.3.100

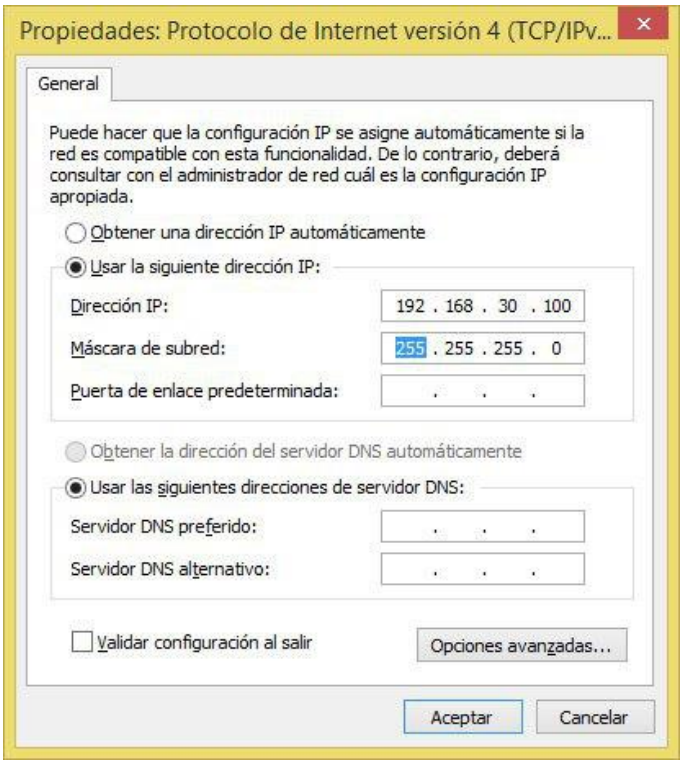



**Figura 7.1- Configuración de la interfaz de red del servidor RADIUS.**

3) Modificar el archivo client.conf con la ip del servicio cliente y la clave secreta.

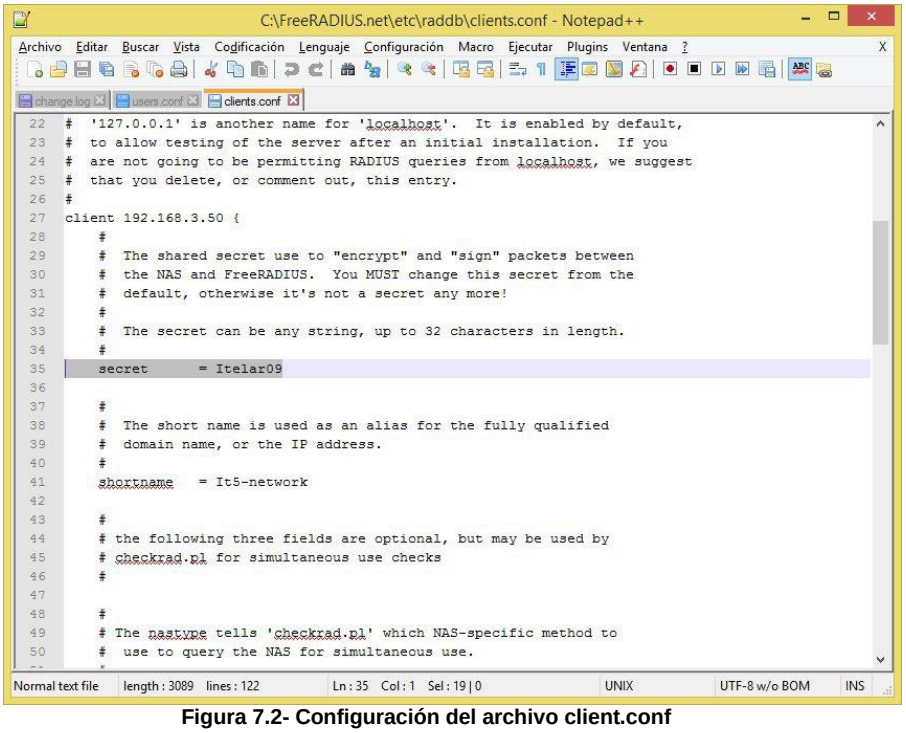



**Figura 7.2- Configuración del archivo client.conf**

4) Modificar el archivo user.conf donde se crean los usuarios a los que se conceden los permisos.

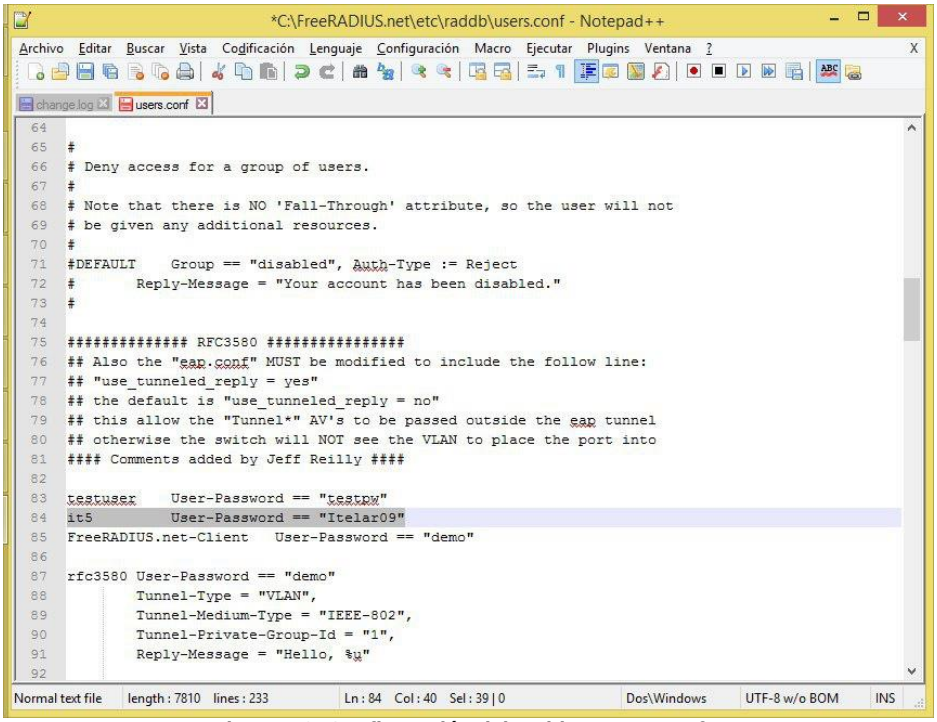



**Figura 7.3- Configuración del archivo users.conf**

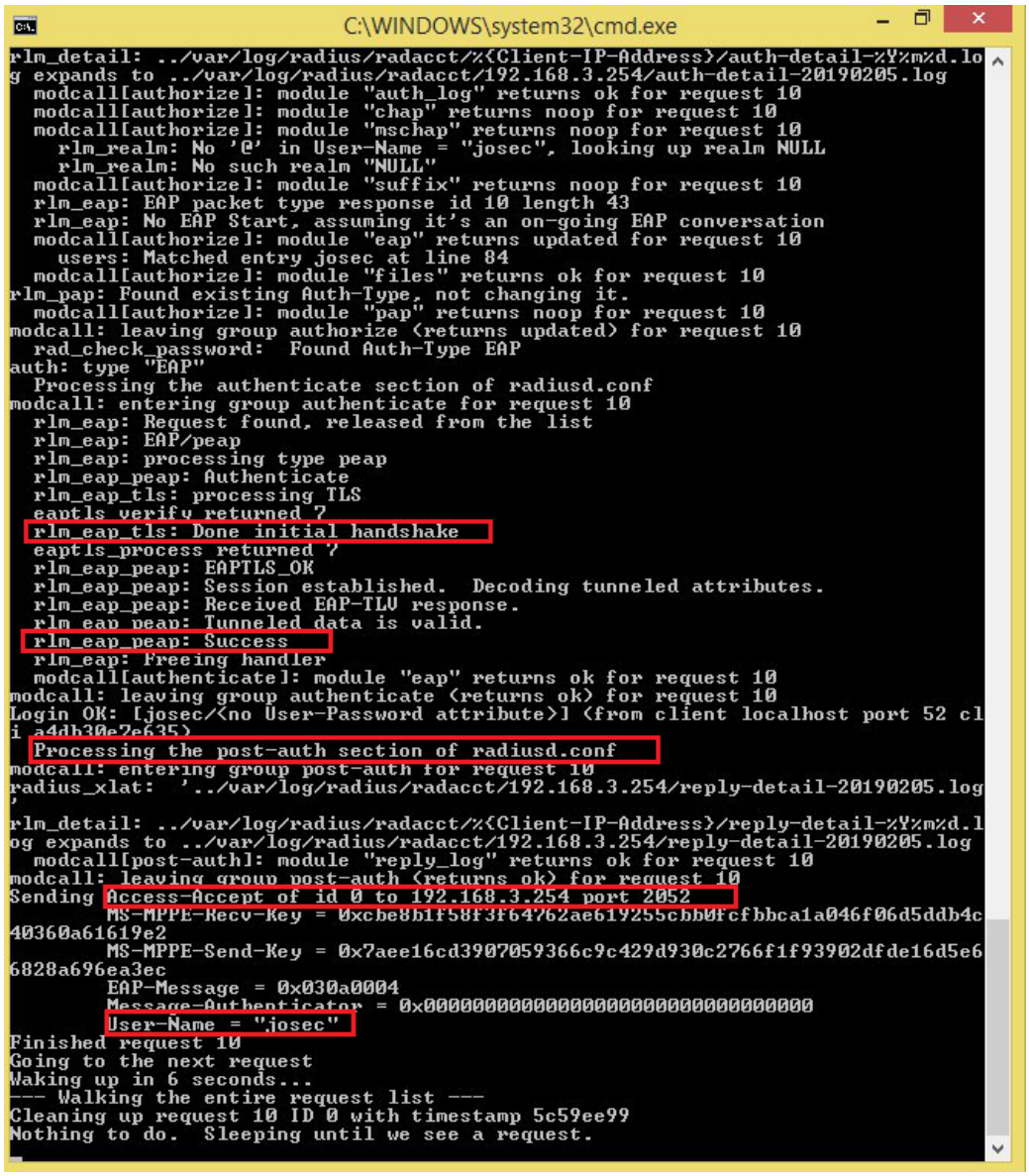
*Nota- También se ha configurado el usuario “josec” con password “Itelar09”

5) En el router, como método de autentificación usar “WPA2 ENTERPRISE” y como método de autentificación “servidor RADIUS” y clave compartida “Itelar09”.

El servidor estaría ya configurado y ahora se pasaría a intentar logearse:


**Figura 7.4- Red wiffi con el servicio de autenticación RADIUS.**

Usando el usuario “josec” con clave “Itelar09” se obtiene el resultado esperado mostrado a través de la consola del servidor RADIUS:



Y por último se obtiene el acceso a la red, que al tener conexión a internet también da acceso a él:


**Figura 7.5- Obtenemos acceso a la red.**

## 7.2 Acceso a escritorio remoto sobre Ipv6 con TeamViewer (Ipv6) en local.

Otro de los servicios para usuario final en el que se ha probado su aclimatación a Ipv6 es TeamViewer[12] y se desea que para ello los equipos **usen exclusivamente Ipv6**.

Para el caso de estudio actual es imprescindible habilitar la opción de llamadas de Lan entrantes.

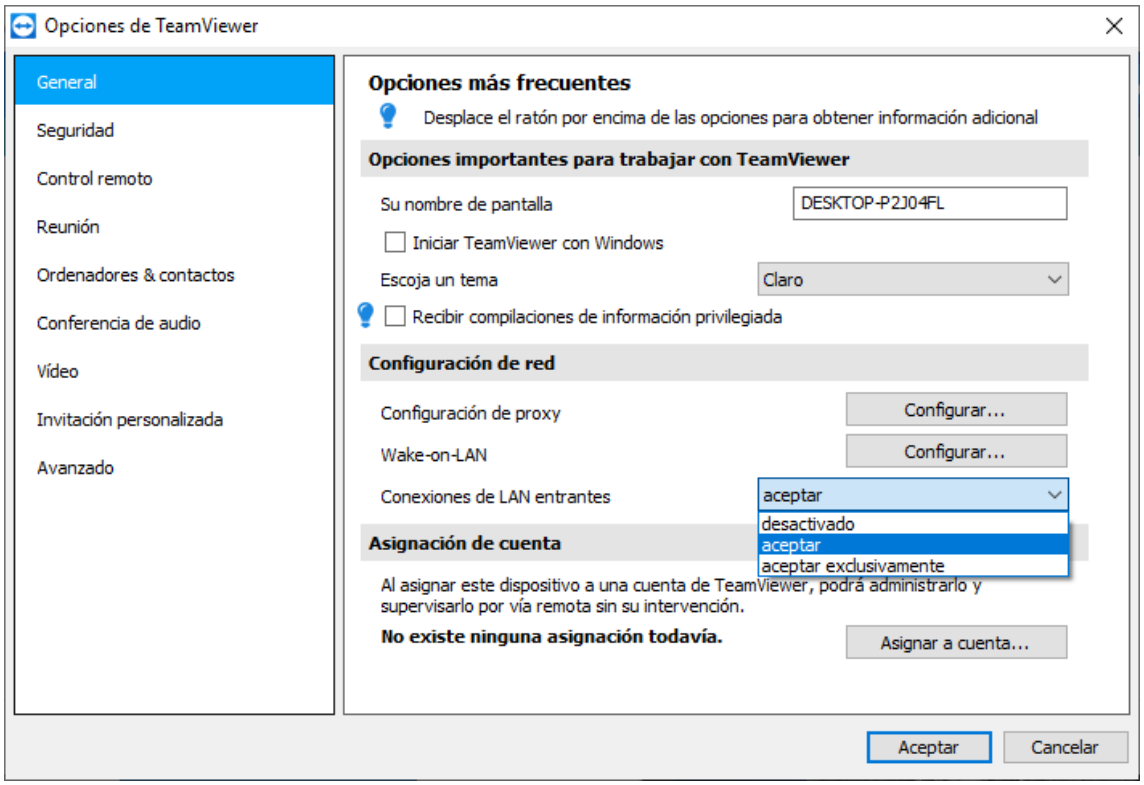



**Figura 7.6- El primer paso para poder usar el programa en local es activar la opción de conexiones de Lan.**

Una vez hecho esto se observa que el programa se comporta de una forma peculiar.

El programa antepone Ipv4 a Ipv6 sugiriendo en principio *identificadores Ipv4 basados en las direcciones Ip de todas las tarjetas, estén o no activas.*

[12] Página web oficial de TeamViewer - https://www.teamviewer.com/es/

Como se podrá ver, solo con eso no se conseguirá una conexión exitosa Ipv6-Only ya que existe una tarjeta deshabilitada que provoca *fallo a la hora de conectar*.

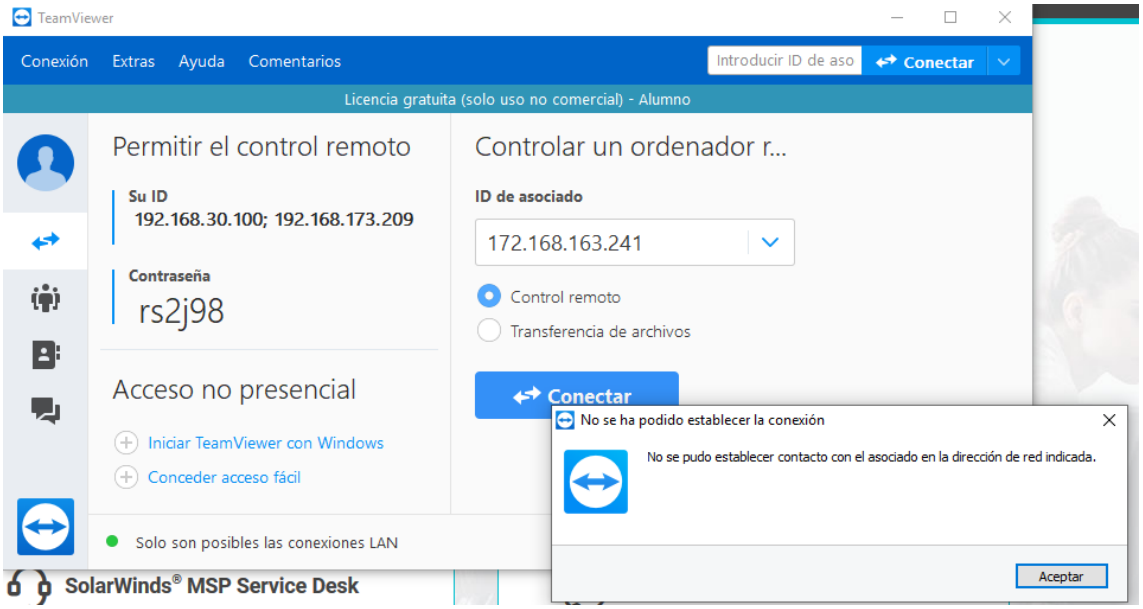



**Figura 7.7- Por defecto intenta usar Ipv4, aunque le estemos dando preferencia a Ipv6.**

Como el objetivo es ver si funciona en Ipv6, se decidió quitar las direcciones Ipv4, lo cual hace que aparezca una dirección que está fuera de rango del tipo 169.254.x.x y las direcciones Ipv6. Aquí el programa pasa a dar *como identificadores la Ip fuera de rango o Ips de controladores virtuales* (Figuras 7.8 y 7.9).

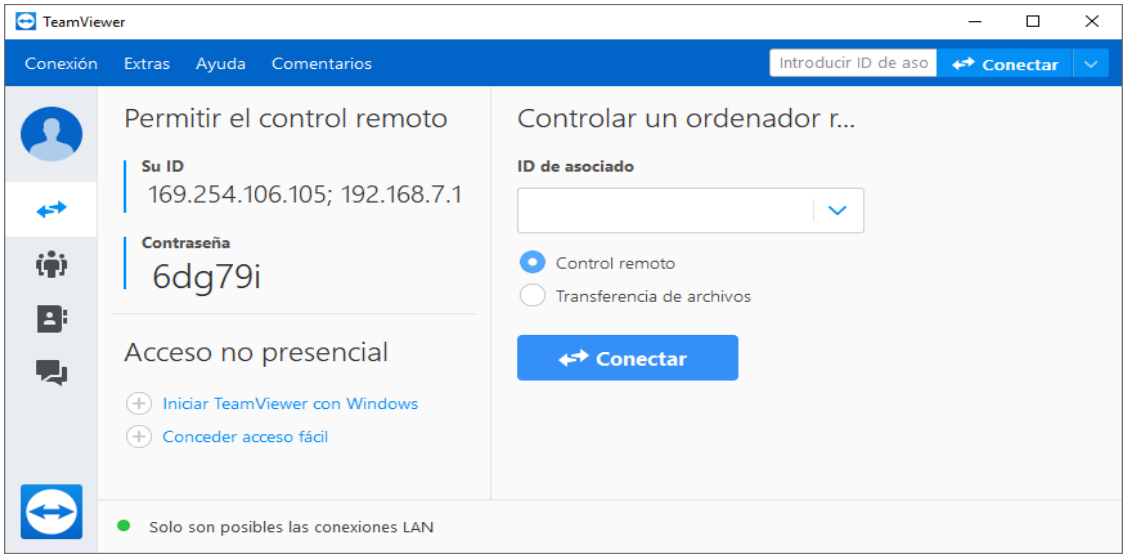



**Figura 7.8- Por defecto intenta seguir usando Ipv4 aunque las tarjetas estén deshabilitadas.**

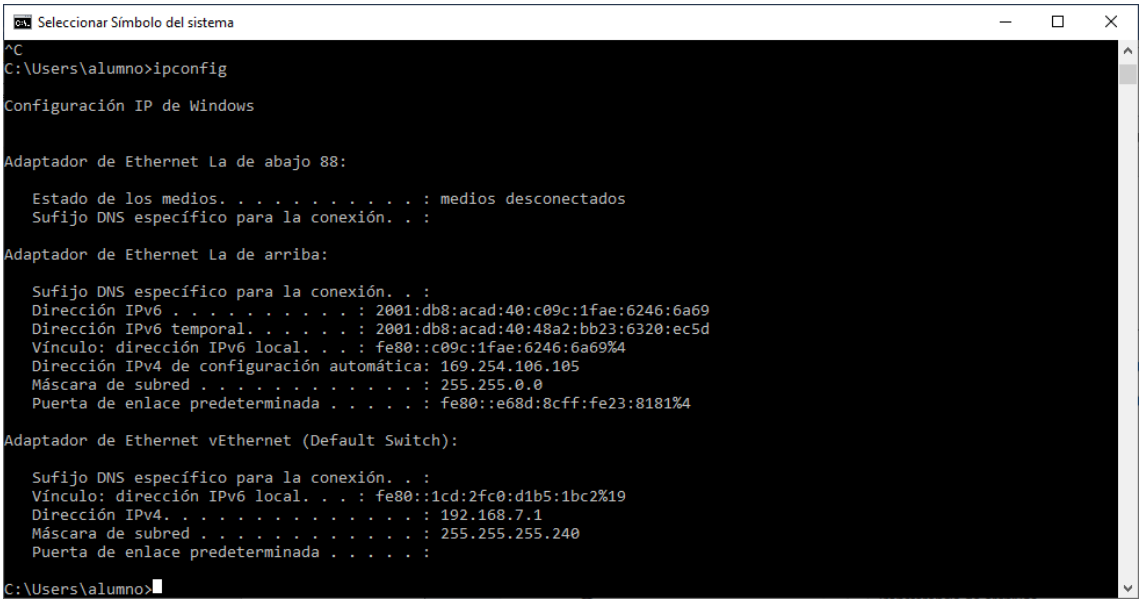



**Figura 7.9- La figura muestra que las direcciones a las que ahora redirige son las virtuales y la Ip fuera de rango.**

Pero el objetivo sigue siendo usar exclusivamente en Ipv6.

Para conseguir identificadores Ipv6 se deben quitar todas las direcciones Ip del Pc y **deshabilitar el protocolo Ipv4 en todas las redes tanto físicas como virtuales.**

Haciendo esto ya se consigue una comunicación en Lan enteramente sobre el protocolo Ipv6

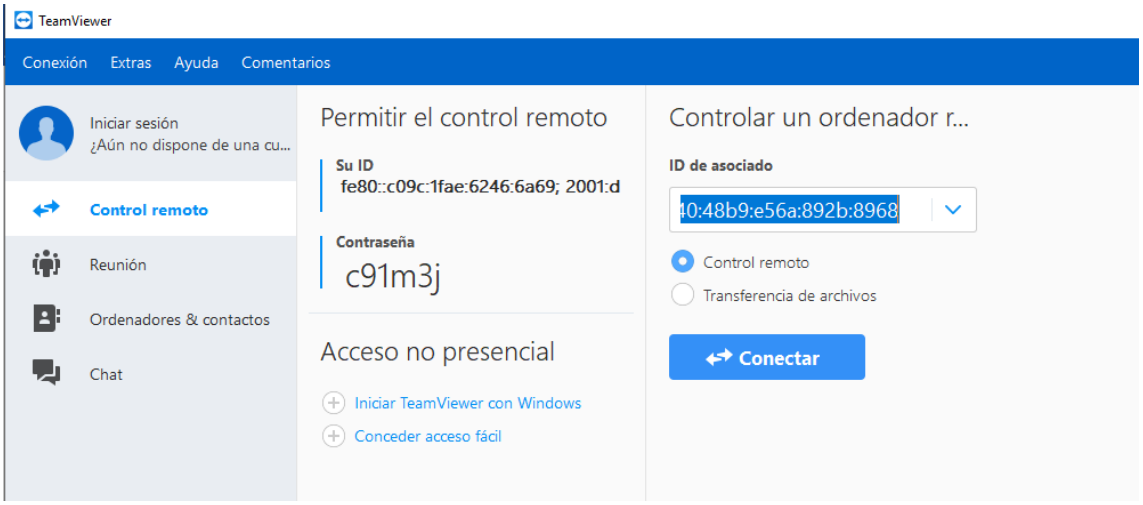



**Figura 7.10- Conexión en Lan con TeamViewer usando Ipv6-Only.**

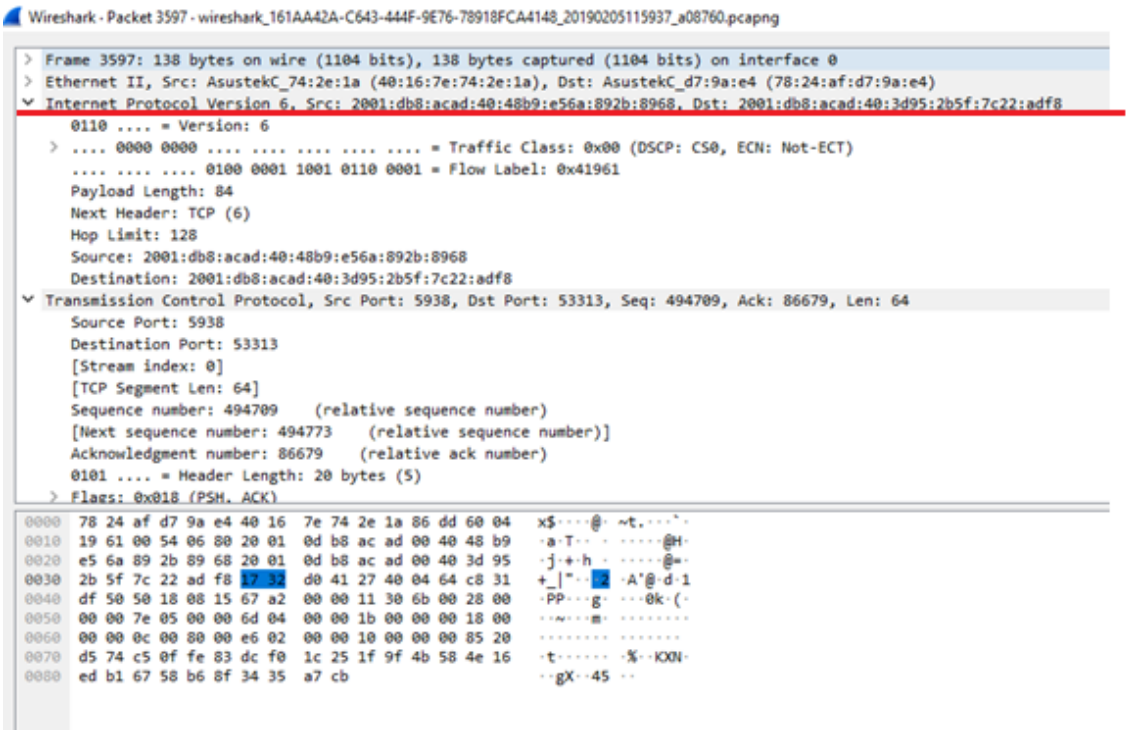



**Figura 7.11- Comprobamos con WireShark que la conexión se realiza efectivamente con Ipv6.**

## 7.3 VOIP entre ordenadores sobre Ipv6 con Linphone (Ipv6)

En el caso de Linphone[13], cómo se adapta a Ipv6 difiere del caso de estudio anterior.

En Linphone se tiene una opción "*Use Ipv6 instead of Ipv4".* Lo cual facilita mucho las cosas, pues no hay que tocar nada de la posible configuración Ipv4.

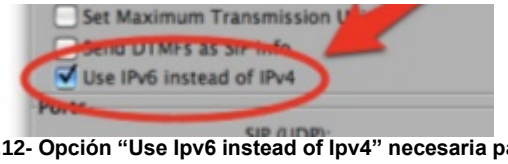



**Figura 7.12- Opción "Use Ipv6 instead of Ipv4" necesaria para nuestro objetivo.**

Una vez se marca esta opción el programa reconoce inmediatamente la dirección Ipv6, usándola de identificador en el caso de no haber creado cuenta y siendo la dirección referencia a la hora de hacer y recibir llamadas.

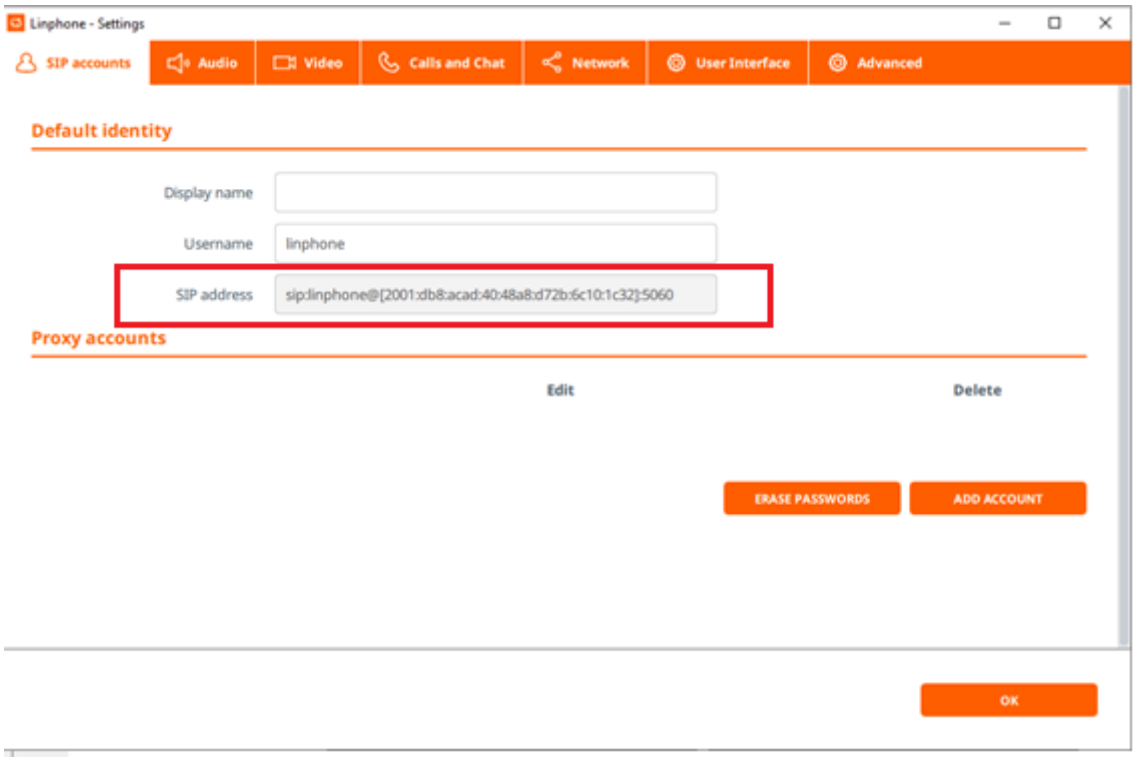



**Figura 7.13- Automáticamente después de marcar la opción nos reconoce la dirección Ipv6.**

[13] Página web oficial de Linphone - http://www.linphone.org/

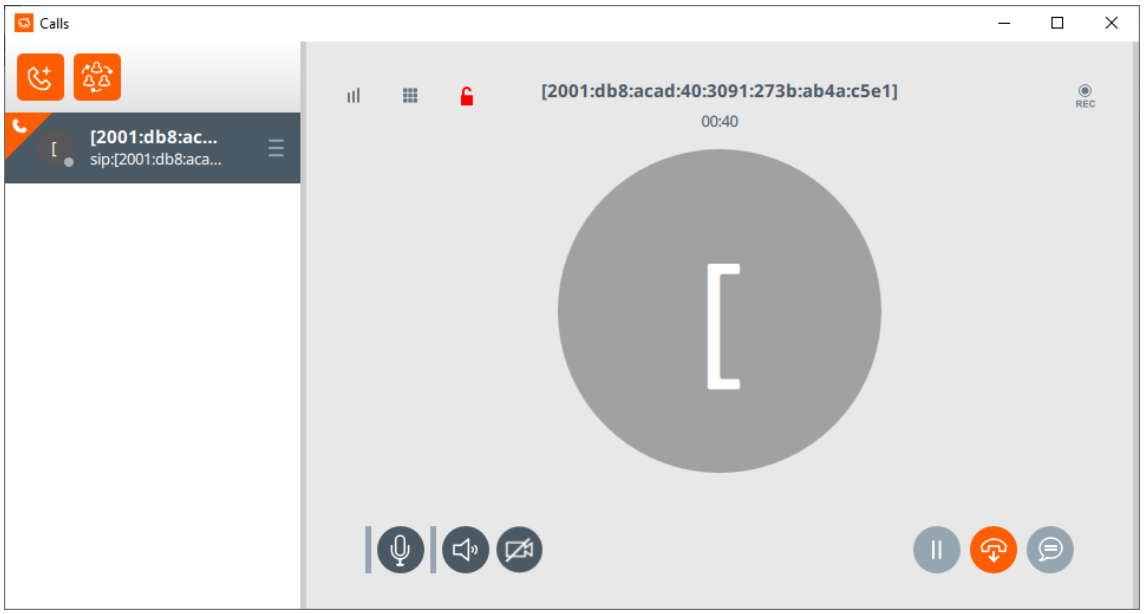



**Figura 7.14- Llamada de VoIP entre dos Pcs del laboratorio usando linphone sobre Ipv6.**

Como se puede apreciar, éste es un ejemplo de buena adaptación y no requiere casi configuración por parte del usuario, al contrario que TeamViewer.

## 7.4 Servidor de Kerberos.io con control y visualización imágenes sobre Raspberry con cámaras Ip (Ipv4).

Kerberos.io [14] es una solución gratuita de videovigilancia, que funciona con cualquier cámara IP y en todas las máquinas basadas en Linux. Puede implementar un sistema de videovigilancia en entornos como Raspberry Pi, Orange Pi[15], Docker, Ubuntu etc.

Los datos de la maqueta para este ejemplo son los siguientes:

Ip de la cámara: 192.168.3.50

Ip default Gateway: 192.168.3.108

Ip interfaz de las Raspberry con Kerberos.io: 192.168.3.223

En este caso de estudio se va a instalar el servidor en una Raspeberry Pi 2 y configurarlo para poder controlar las imágenes dadas por una cámara Ip.

El primer paso es descargar el ejecutable que mejor se adecue al SO del equipo, en este caso Ubuntu 14.04 (64bits):

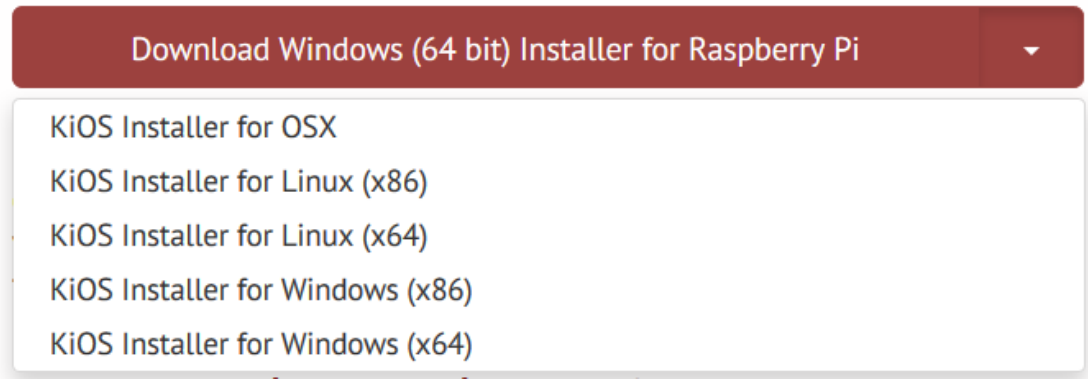



**Figura 7.15- Elección del instalador adecuada para el SO, en nuestro caso Windows x64.**

[14] Página web oficial de Kerberos.io- https://kerberos.io/
[15] Página web oficial de Orange Pi – http://www.orangepi.org/index.html

Se selecciona la última versión, la 2.7.2 y puesto que se está configurando la imagen del servidor en un PC de mesa y no se dispone de información de las redes WIFI, se elige una configuración ethernet:

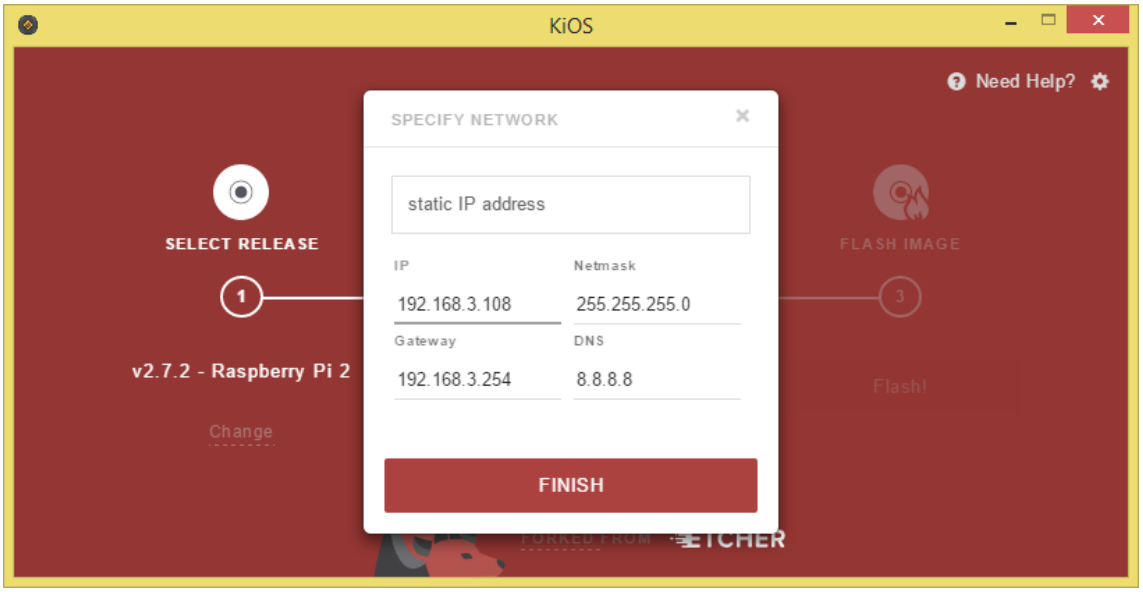



**Figura 7.16- Configuración de la Ip de la interfaz de red de la Raspberry y datos de red.**

Se monta en una tarjeta SD. Al terminar el proceso se conecta la tarjeta a la Raspberry y se espera 5 minutos a que se monte el SO.

Se conecta un cable ethernet la Raspberry y se introduce la Ip que se le ha configurado, 192.168.3.108:

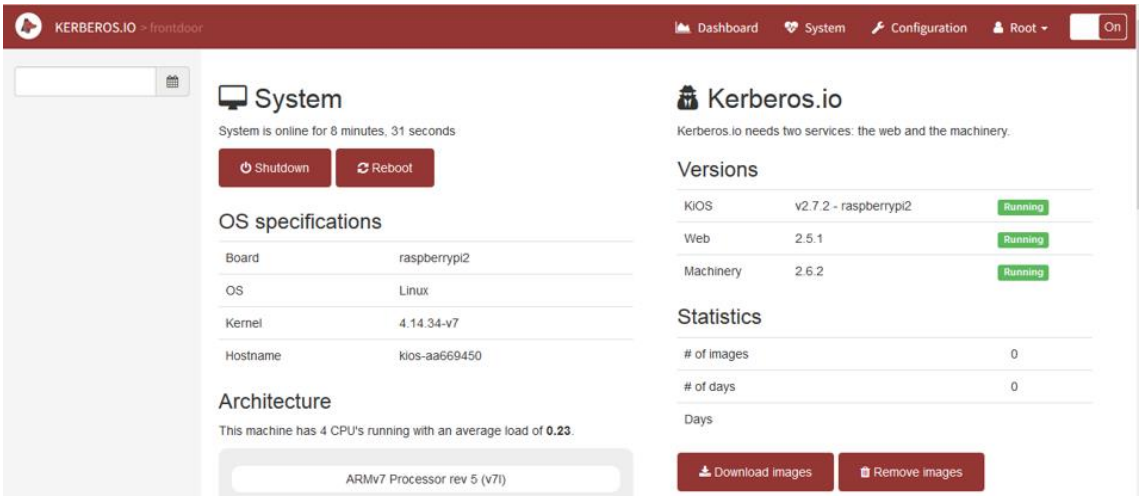



**Figura 7.17- Una vez conseguimos la conexión, entramos a la página principal.**

Ahora sólo queda configurar el acceso a la cámara Ip de la red. Para ello hay que ir al menú de configuración, buscar el apartado de cámaras y seleccionar cámara Ip:

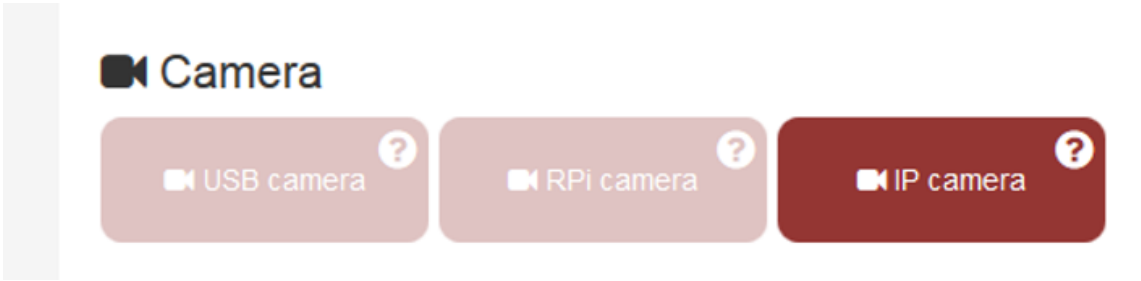



**Figura 7.18- En nuestro caso buscamos controlar la salida de la cámara Ip conectada.**

Una vez allí, en la zona seleccionada se cambia la Ip que viene por defecto por la dirección Ip de la cámara, en este caso 192.168.3.50 y se ajusta la configuración de la cámara a los valores que se desee.

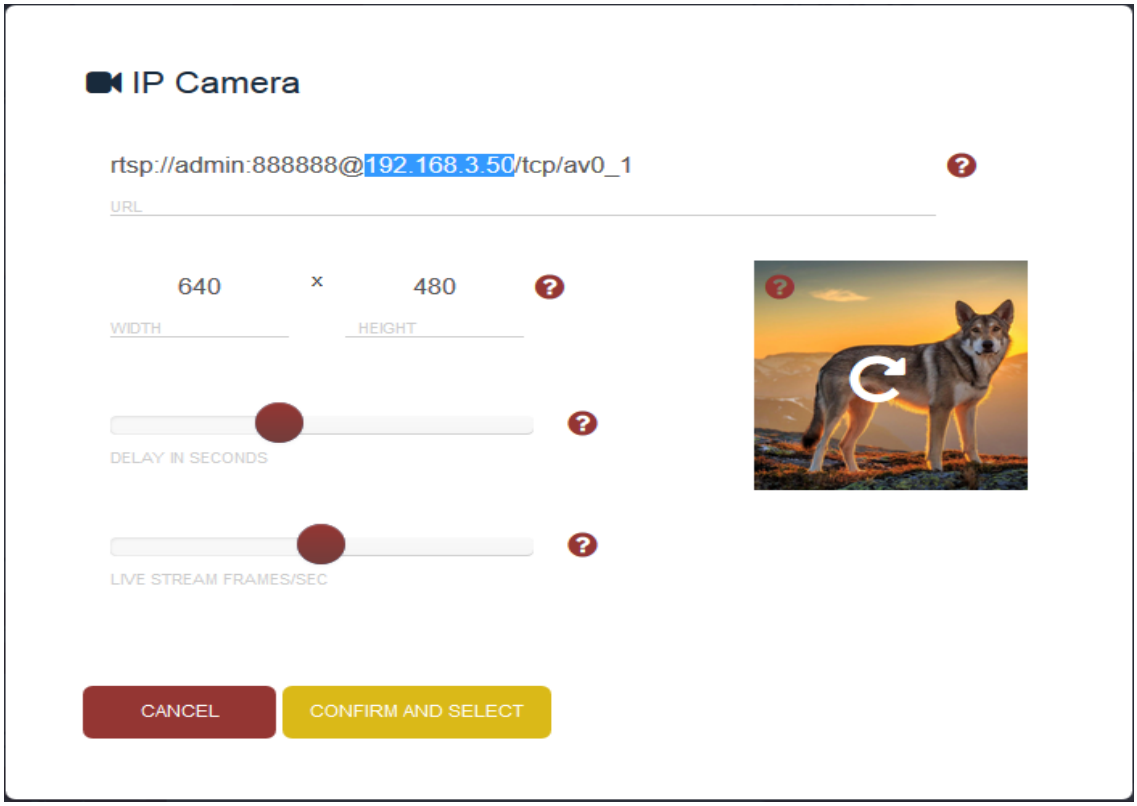



**Figura 7.19- Opciones dentro del apartado Cámara Ip.**

Una vez hecho esto se regresa a la página general donde ahora se recibirá la imagen de la cámara.

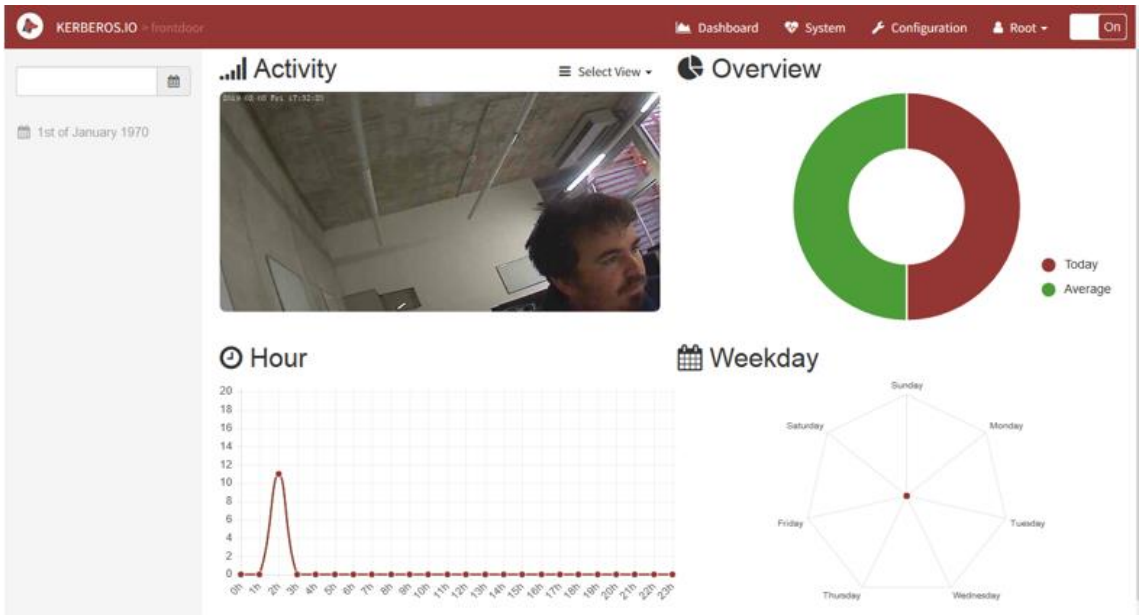



**Figura 7.20- Configuración exitosa de Kerberos usando Raspberry para controlar cámaras Ips.**

Si se quisiera añadir más de una cámara se requeriría de una configuración por dockers. La instalación sobre SO Windows, a día de hoy puede producir varios errores, al igual que en algunos sistemas Linux, cosa que puede ser razonable debido a que este servicio reciente todavía está en Beta.

# Capítulo 8 Vías de ampliación y conclusiones

## 8.1 Posibles vías de ampliación

El protocolo **6loWPAN**[16] (Ipv6 Over Low power Wireless Personal Area Networks) surgió como una idea del IETF en 2004 para aprovechar la potencia de direccionamiento de Ipv6 junto con la flexibilidad de los dispositivos de bajo procesamiento y consumo.

Tras 10 años de desarrollo se puede decir que se ha creado un protocolo que puede ser aplicado a dispositivos simples, con limitado procesamiento y baja potencia para ser integrado en el Internet de las Cosas.

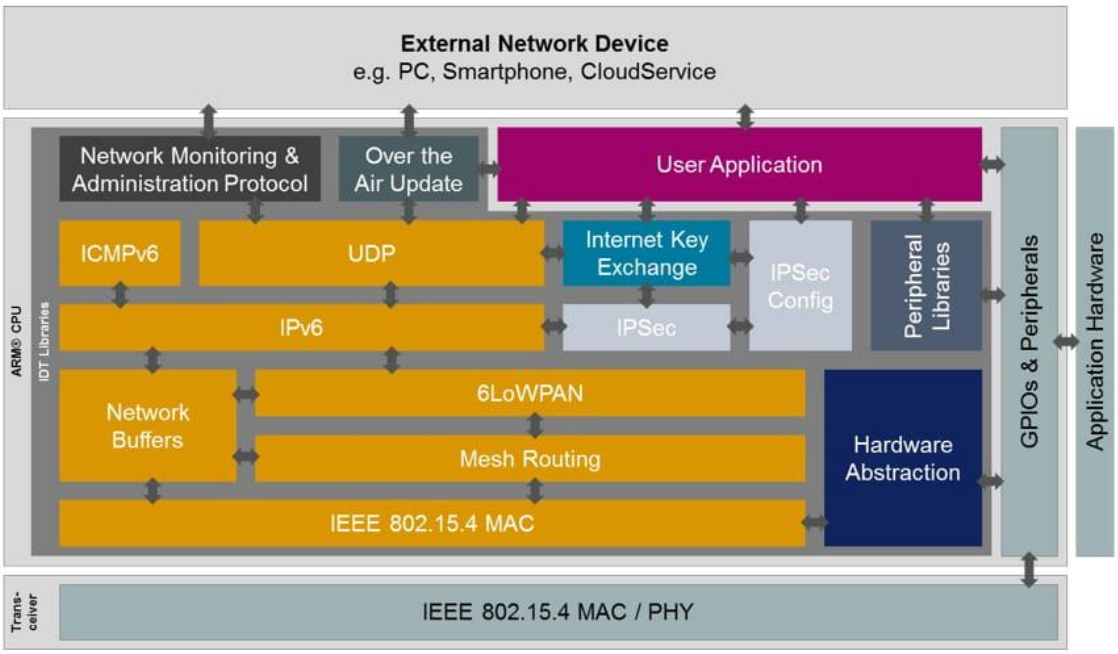



**Figura 8.1- La figura muestra que 6LowPan trabaja entre los protocolos de enlace y red.**

6loWPAN se encuentra entre las capas de enlace y red, pudiendo llegar incluso a la capa de transporte. Compite directamente con protocolos PAN como Zigbee, Bluethooth y RFID.

[16] 6loWPAN - https://datatracker.ietf.org/group/6lowpan/about/

En este proyecto se dedica un espacio considerable a estudiar algunos de los protocolos de direccionamiento dinámico (RIP y OSPF) y su versión compatible con Ipv6 (RIPng y OSPFv3) pero existen algunos más como **BGP o EIGPR**.

**BGP** (del inglés Border Gateway Protocol) es un protocolo mediante el cual se intercambia información de encaminamiento entre sistemas autónomos. Por ejemplo, los proveedores de servicio registrados en Internet suelen componerse de varios sistemas autónomos y para este caso es necesario un protocolo como BGP.

**EIGRP** (Protocolo de Enrutamiento de Puerta de enlace Interior Mejorado en español) es un protocolo de encaminamiento de estado de enlace, propiedad de Cisco Systems, que ofrece lo mejor de los algoritmos de vector de distancias y del estado de enlace. Supone una mejora del protocolo **IGPR**. Se considera un protocolo avanzado que se basa en las características normalmente asociadas con los protocolos del estado de enlace. Aunque no garantiza el uso de la mejor ruta, es bastante usado porque EIGRP es algo más fácil de configurar que OSPF. EIGRP mejora las propiedades de convergencia y opera con mayor eficiencia que IGRP.

Por otro lado, otra de las vías relacionadas con el proyecto que se podría desarrollar es el estado actual de Ipv6. A modo de resumen rápido podríamos decir que en los últimos 6 años el desarrollo de Ipv6 en redes y proveedores de servicios ha sufrido un aumento bastante importante.

- Más del 25% de todas las redes conectadas a Internet tienen algún tipo de conectividad IPv6[17].

- Según aportes de Google en 49 países**, más del 5% del tráfico de internet es sobre Ipv6**, número que va aumentando cada año.

- De esos 49 países, **24** actualmente aumentan ese ratio de tráfico en Ipv6 al **15%**.

[17]Porcentaje de usuarios que acceden a Google mediante IPV6
https://www.google.com/intl/es/ipv6/statistics.html#tab=ipv6-adoption&tab=per-country-ipv6-adoption

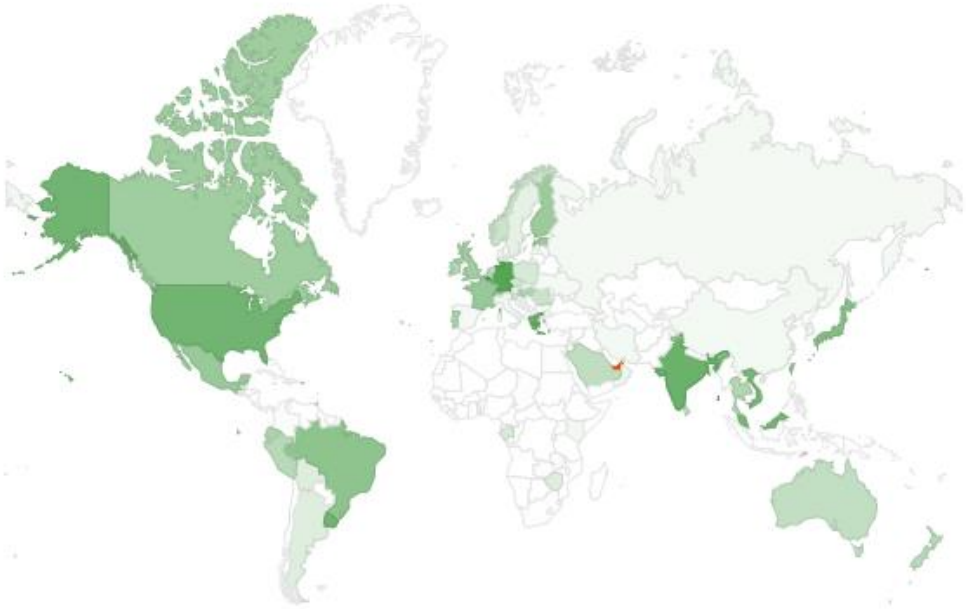


**Figura 8.2- En la figura[18] se resaltan los países con más de un 15% de tráfico en Ipv6 según Google statistics.**

En algunos países, la mayor parte de las redes móviles se están adaptando en gran medida a Ipv6. En Japan (NTT – 7%, KDDI – 42% y Softbank – 34%), India (Reliance JIO – 87%) o en EEUU con (Verizon Wireless – 84%, Sprint – 70%, T-Mobile USA – 93%, y AT&T Wireless – 57%) tienen una buena adaptación a Ipv6 a día de hoy. Algunas de estas redes móviles están yendo más allá y están dando pasos para que sólo se use Ipv6-only, lo que significaría una simplificación de la red y una reducción de costes.

Actualmente se puede decir que Ipv6 ha dejado de estar en fase de desarrollo para decir sin duda alguna que ahora se encuentra en fase de crecimiento y expansión.

[18] Estadísticas de Google adaptación a Ipv6 por países - https://www.google.com/intl/es/ipv6/statistics.html#tab=per-country-ipv6-adoption&tab=per-country-ipv6-adoption

## 8.2 Conclusiones

Debido a que desde 2011 los últimos **bloques de direcciones Ips están agotados en algunos continentes**, Ipv6 se ha consolidado como el protocolo destinado a ser su sucesor.

Es un protocolo que debido a su aumento en el tamaño del paquete Ip a 128 bits **permite una cantidad de Ips que a día de hoy es impensable agotar ni en cientos de años.**

En cuanto su funcionamiento Ipv6, aunque mejora en algunos aspectos a su predecesor, a la hora de **funcionamiento y configuración no es muy diferente a Ipv4**. Los desarrolladores de Ipv6 trataron que los protocolos destinados a suplantar los protocolos de Ipv4 fueran lo más parecido posible en su funcionamiento. De hecho, en algunos casos **aunque su funcionamiento sea algo diferente, su configuración es más sencilla.**

Ipv6 recupera la conexión punto a punto debido a la ausencia de NAT, mecanismo usado en Ipv4 para un mayor aprovechamiento de las IPs que Ipv6 no necesita. Esto hace que la seguridad en Ipv6 esté en los extremos. Debido a que la seguridad en Ipv6 se centra en los cortafuegos y otros dispositivos especializados, junto con IPsec (Incorporado obligatoriamente) y el gran número de direcciones a atacar en una red /64 por ejemplo, hace que para muchos Ipv6 sea **más seguro** que Ipv4.

Por otro lado, hay que resaltar que actualmente a nivel usuario la adaptación es más difícil.

Los routers y switches usados normalmente en el hogar no poseen opciones para desarrollar por completo Ipv6 en una red de hogar. Desde la Ip para acceder al router o switch de turno, que sólo tienen la opción de Ipv4, hasta la parte WAN que en dispositivos de coste medio o bajo es inexistente, pasando por la no prestación de servicios Ipv6 en muchos proveedores de servicios hace que sea difícil su implantación.

También debemos resaltar que muchas aplicaciones conocidas como Skype[19], Netflix[20] o Spotify[21] o gran cantidad de páginas webs que por su diseño y/o antigüedad o no funcionan como deberían o directamente no funcionan con Ipv6.

Esto hace que, aunque Ipv6 se haya ido convirtiendo en un protocolo más de presente que de futuro, que a día de hoy todavía tiene muchos campos en los que debe seguir creciendo y expandiéndose, para que en unas décadas sea el protocolo de red predominante.

[19] Página oficial Skype - https://www.skype.com/es/
[20] Pagina oficial Netflix- https://www.netflix.com/es/
[21] Página oficial Spotify - https://www.spotify.com/es/

# Bibliografía


- http://www.Ipv6.es/es/ES/transicion/usuarios/Paginas/Ipv6_usuarios.asp
- https://sites.google.com/site/redeslocalesyglobales/6-arquitecturas-de-redes/6-arquitectura-tcp-ip/7-nivel-de-red/8-direccionamiento-Ipv6/6-transicion-de-ipv4-a-Ipv6
- https://es.slideshare.net/dan2ysgl/mecanismos-de-transicin-Ipv6-teredo-6to4-y-6rd
- http://www.redescisco.net/sitio/2011/07/13/direccionamiento-basico-con-Ipv6-ripng/
- https://www.muycomputer.com/2014/01/13/que-es-dmz-dlink/
- https://wikipedia.org
- https://mikrotik.com/support
- https://www.netacad.com/es
- http://blackhold.nusepas.com/2012/10/26/tunel-6to4-mikrotik/
- https://networklessons.com/cisco/ccie-routing-switching-written/Ipv6-dhcpv6-prefix-delegation/
- https://www.eduangi.org/node131.html
- http://www.redescisco.net/sitio/2011/08/11/tipos-de-areas-en-ospf/
- http://he.net/
- http://slides.lacnic.net/wpcontent/uploads/2017/05/mecanisimo_transicion.pdf
- https://doc.kerberos.io/
- https://www.google.com/intl/es/ipv6/statistics.html#tab=ipv6-adoption&tab=per-country-ipv6-adoption

- https://www.google.com/intl/es/ipv6/statistics.html#tab=per-country-ipv6-adoption&tab=per-country-ipv6-adoption

- https://datatracker.ietf.org/group/6lowpan/about/

- http://ipv6.ift.org.mx/

# Anexos

## Manual de configuración de Ipv6 en Windows 10

- Selecciona Inicio y, a continuación, Configuración > Ethernet > Cambiar opciones del adaptador.
- En conexiones, elige la red a la que quieras cambiar la configuración y luego selecciona Propiedades.
- En la asignación de IP, selecciona Editar.
- En Editar la configuración IP, elige Manual y, a continuación, activa Ipv6.

    - Para especificar una dirección IP, en los cuadros dirección IP, Longitud de prefijo de subred, y Puerta de enlace, escriba la configuración de la dirección IP.

    - Para especificar una dirección del servidor DNS, en los cuadros DNS preferido y DNS alternativo, escribe las direcciones de los servidores DNS principales y secundarios.

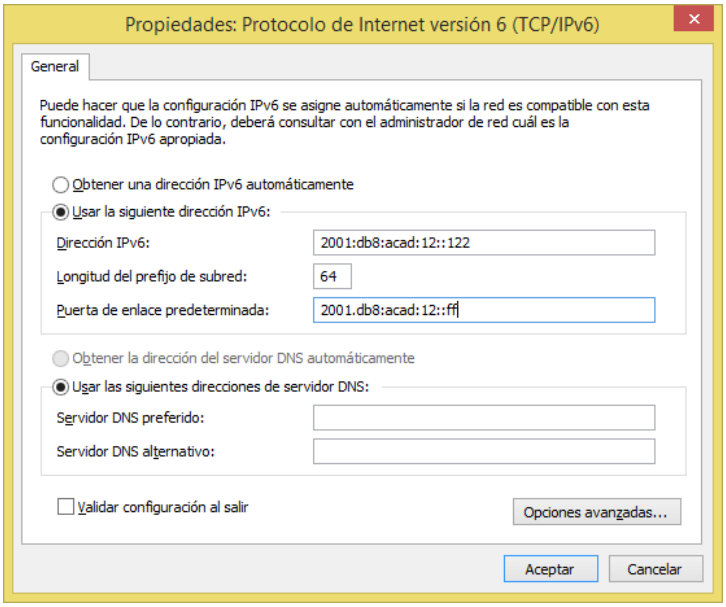



**Figura 9.1- La figura muestra cómo y dónde se configura una dirección Ipv6.**

## Manual de configuración Ipv6 en Ubuntu

En ubuntu, el archivo de configuración principal de los parámetros de red es: interfaces, ubicado en /etc/network/

- Editar el archivo de configuración con:

  nano /etc/network/interfaces

- Todas las líneas que inician con el signo # son comentarios, el archivo debe quedar así:

  # Adaptador local (loopback)
  auto lo
  iface lo inet loopback

➢ Si queremos que la ip nos la surta un servidor DHCP:
  iface eth0 inet6 dhcp

➢ Si queremos asignarle la ip de forma estática:

  iface eth1 inet6 static
  pre-up modprobe Ipv6
  address fec0::10 ← Link-local
  netmask 64

- Si es necesario se puede especificar el gateway
  gateway fec0::1
- Re-iniciamos el equipo para aplicar los cambios:
  reboot

## Manual de instalación de paquete Ipv6 en router Mikrotik

- Nos dirigimos a www. Mikrotik.com y luego hacemos click en Software.
- Elegimos el paquete el paquete Ipv6 asegurándonos que nos descargamos la versión para Hap Lite Mikrotik.
- Lo guardamos y lo extraemos.
- Para instalar los paquetes en el router necesitaremos abrir el Winbox y conectar con el router.
- Hacemos click en system y luego en package

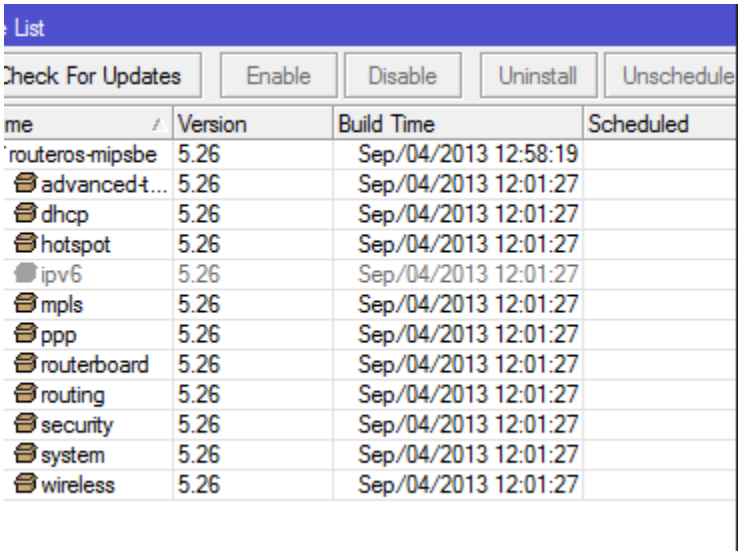



**Figura 9.2- Listado de paquetes dentro del router Mikrotik.**

(En nuestro caso aparece en gris porque ya está instalado)

- Vamos al escritorio y depositamos el paquete de Ipv6 en la lista de paquetes, creándose allí una copia del paquete.
- El paquete ya está instalado, para terminar, hacemos reboot para que se aplique.

**Adquirir servicio gratuito de Tunnel Broker para hacer un túnel 6to4.**

Los pasos que hemos seguido han sido los siguientes:

1) Ir a la página https://he.net

2) A la derecha, hacer click en “Free Ipv6 Tunnel Broker”.

3) Introducir los datos necesarios para la creación de la cuenta .

4) A la izquierda, hacer click en “Create Regular Tunnel”.

5) Elegir un servidor de túnel, cuanto más cercano a España, mejor.


**Figura 9.3- Listado de localización de servidores de Hurricane Electric.**

6) Introducir tu Ip local (pública).

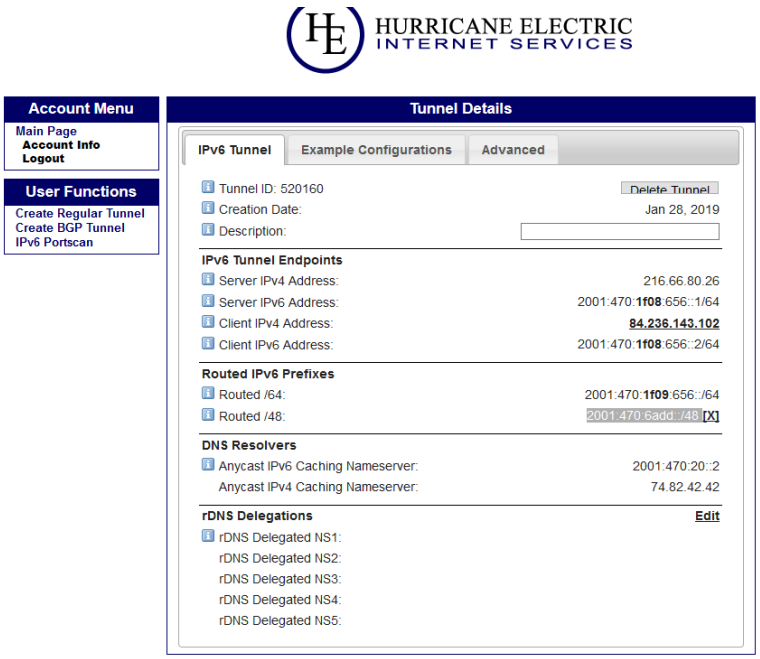



**Figura 9.4- Estado final tras la connexion inicial con el servidor de Hurricane Electric.**

El túnel ya está creado y ya se tienen los mecanismos necesarios para realizar el túnel.

# Glosario de términos

DNS → Domain Name System (Sistema de Nombres de Dominio).

DHCP→ Dynamic Host Configuration Protocol(Protocolo de Configuración Dinámica de Host).

LDAP→Lightweight Directory Access Protocol (Protocolo Ligero de Acceso a Directorios).

RADIUS→Remote Authentication Dial-In User Service (Servicio de Autenticaciónde Usuario).

STP→ (Spanning Tree Protocol.

VLAN→Virtual Local Area Network (Red de Área Local Virtual).

QoS→Quality of Service (Calidad de Servicio).

IPv4→Internet Protocol versión 4.

Ipv6→Internet Protocol versión 6.

PyME→Pequeña y Mediana Empresa.

ACL→Access Control List (Lista de Control de Acceso).

ISP→Internet Service Provider (Proveedor de Servicios de Internet).

TTL→Time to Live (Tiempo de Vida).

ARP→Address Resolution Protocol (Protocolo de Resolución de Direcciones).

VPN→Virtual Private Network (Red Privada Virtual).

MPLS→Multiprotocol Label Switching.

DMZ→Demilitarized Zone (Zona Desmilitarizada).

HTTP→Hypertext Transfer Protocol (Protocolo de Transferencia de Hipertexto).

SMTP→Simple Mail Transfer Protocol (Protocolo para la Transferencia Simple de Correo electrónico).

MAC→Media Access Control (Control de Acceso al Medio).

SSID→Service Set IDentifier.

AP→Access Point (Punto de Acceso).

IOS→Internetwork Operating System.

NAT→Network Address Translation (Traduccion de Dirección de Red).

VoIP→Voice over Internet Protocol (Voz sobre Protocolo de Internet).

SNMP→Simple Network Management Protocol (Protocolo Simple de Administración de Red).

ERPS→Ethernet Ring Protection Switching.(Protección de Anillos en Conmutación Ethernet).

LLDP→Link Layer Discovery Protocol (Protocolo de Descubrimiento a Nivel de Enlace).

BW→Bandwidth (Ancho de Banda).

ICMP→Internet Control Message Protocol (protocolo de Mensajes de Control de Internet).

RTP→Real-time Transport Protocol (Protocolo de Transporte en Tiempo Real).

RTSP→Real Time Streaming Protocol (Protocolo de Streaming en Tiempo Real).

RTCP→Real Time Control Protocol (Protocolo de Control en Tiempo Real).

LAN→Local Area Network (Red de Área Local).

WAN→Wide Area Network (Red de Área Amplia).

CPU→Central Processing Unit (Unicdad Central de Procesado).

NAT-PT→Network Address Translation/Protocol Translation

TUNNEL6TO4→Los túneles 6to4 permiten que ubicaciones Ipv6 aisladas se comuniquen mediante un túnel automático a través de una red IPv4 que no admite Ipv6.

OSPF→protocolo de red para encaminamiento dinámico que usa el algoritmo Dijkstra, para calcular la ruta más corta entre dos nodos.

OSPFv3→Actualización del protocolo OSPF que provee de soporte para Ipv6 a través de la retransmisión de cabeceras y prefijos de red Ipv6 en los LSAs.

RIP→protocolo de puerta de enlace interior (*Interior Gateway Protocol*, **IGP) Su** algoritmo de encaminamiento está basado en el vector de distancia.

RIPng→Versión del protocolo Rip que habilita la transmisión a través de Ipv6.

RSTP→ (Rapid Spanning Tree Protocol).