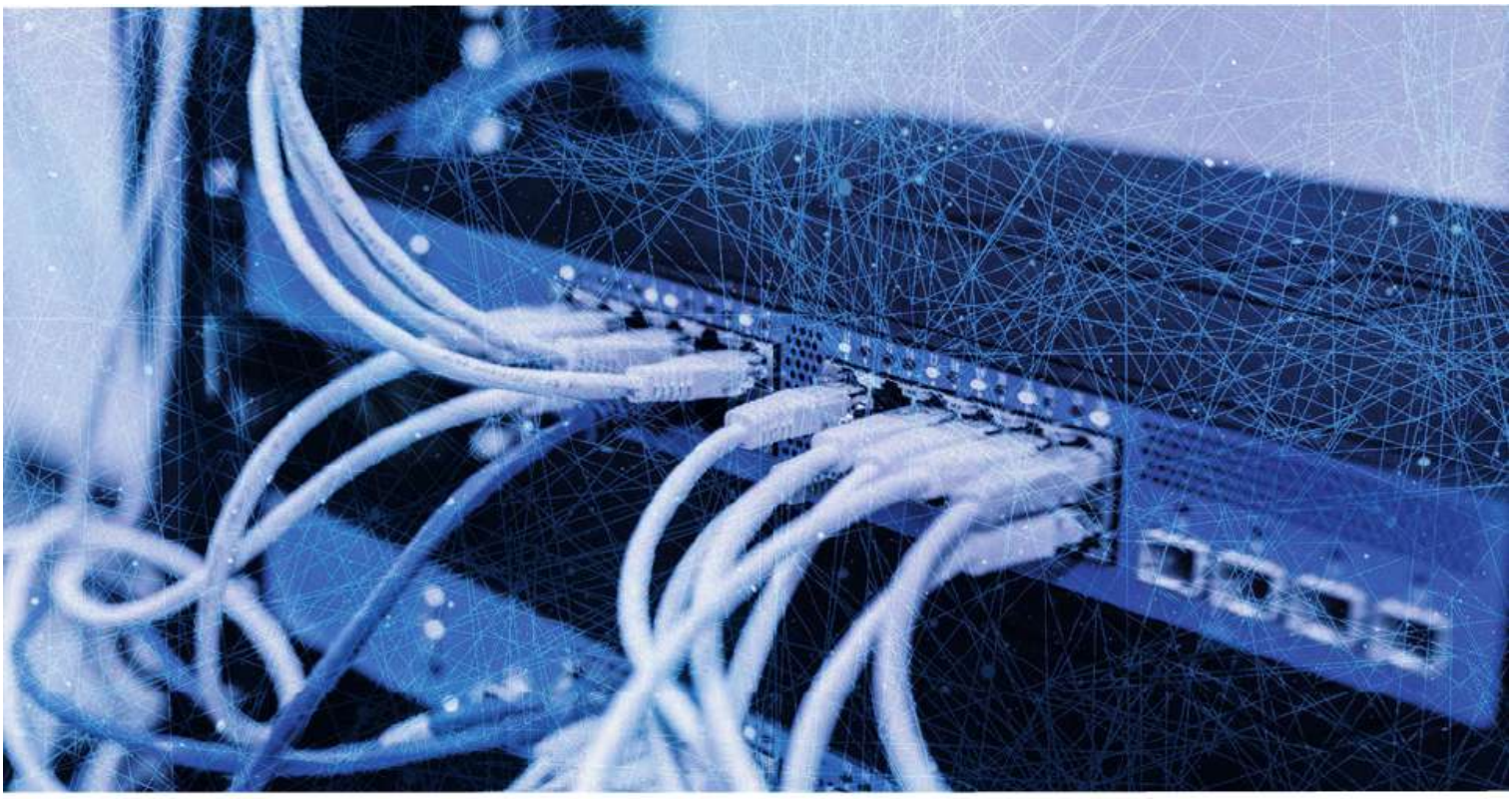
Redes CISCO

Curso práctico de formación para la certificación CCNA



Daniel Pérez Torres

Alfaomega

libros R

Redes CISCO

Curso práctico de formación para la certificación CCNA

Daniel Pérez Torres

Alfaomega

Libros R

Diseño de colección y preimpresión:
Grupo RC
Diseño cubierta: Cuadratín

Datos catalográficos

Pérez, Daniel
Redes CISCO. Curso práctico de formación para la certificación CCNA
Primera Edición
Alfaomega Grupo Editor, S.A. de C.V., México

ISBN: 978-607-538-214-2

Formato: 17 x 23 cm Páginas: 612

Redes CISCO. Curso práctico de formación para la certificación CCNA
Daniel Pérez Torres
ISBN: 978-84-947170-3-1 edición original publicada por RC Libros, Madrid, España.

Primera edición: Alfaomega Grupo Editor, México, abril 2018

Dr. Isidoro Olvera (Eje 2 sur) No. 74, Col. Doctores, 06720, Ciudad de México.

Miembro de la Cámara Nacional de la Industria Editorial Mexicana
Registro No. 2317

Pág. Web: **http://www.alfaomega.com.mx**
E-mail: **atencionalcliente@alfaomega.com.mx**

ISBN: 978-607-538-214-2

Nota importante:
La información contenida en esta obra tiene un fin exclusivamente didáctico y, por lo tanto, no está previsto su aprovechamiento a nivel profesional o industrial. Las indicaciones técnicas y programas incluidos, han sido elaborados con gran cuidado por el autor y reproducidos bajo estrictas normas de control. ALFAOMEGA GRUPO EDITOR, S.A. de C.V. no será jurídicamente responsable por: errores u omisiones; daños y perjuicios que se pudieran atribuir al uso de la información comprendida en este libro, ni por la utilización indebida que pudiera dársele. d e s c a r g a do en: e y b o oks. c o m

Edición autorizada para venta en México y todo el continente americano.

Impreso en México. Printed in Mexico.

Empresas del grupo:

México: Alfaomega Grupo Editor, S.A. de C.V. – Dr. Isidoro Olvera (Eje 2 sur) No. 74, Col. Doctores, C.P. 06720, Del. Cuauhtémoc, Ciudad de México – Tel.: (52-55) 5575-5022 – Fax: (52-55) 5575-2420 / 2490. Sin costo: 01-800-020-4396 – E-mail: atencionalcliente@alfaomega.com.mx

Colombia: Alfaomega Colombiana S.A. – Calle 62 No. 20-46, Barrio San Luis, Bogotá, Colombia, Tels.: (57-1) 746 0102 / 210 0415 – E-mail: cliente@alfaomega.com.co

Chile: Alfaomega Grupo Editor, S.A. – Av. Providencia 1443. Oficina 24, Santiago, Chile Tel.: (56-2) 2235-4248 – Fax: (56-2) 2235-5786 – E-mail: agechile@alfaomega.cl

Argentina: Alfaomega Grupo Editor Argentino, S.A. – Av. Córdoba 1215, piso 10, CP: 1055, Buenos Aires, Argentina, Tel./Fax: (54-11) 4811-0887 y 4811 7183 – E-mail: ventas@alfaomegaeditor.com.ar

ÍNDICE

PREFACIO

Dentro del ámbito informático, las certificaciones constituyen uno de los títulos más importantes y reconocidos a nivel mundial. Gracias a ellas, empresas líderes en el sector acreditan que sus poseedores disponen de los conocimientos y habilidades necesarias para ejercer laboralmente las funciones de una determinada rama profesional. Microsoft, Cisco, HP, VMWare, Juniper, Fortinet, Oracle, IBM, CheckPoint o Citrix son solo algunos ejemplos de compañías que basan su formación en torno a certificaciones.

En cuanto a redes y seguridad se refiere, el CCNA es una de las más valoradas, primero, porque abarca desde los conceptos más básicos de *routing* y *switching* hasta protocolos realmente avanzados, y segundo, porque su título es acreditado por Cisco, compañía líder en el sector de redes y comunicaciones.

El objetivo principal de este libro consiste en dotar a sus lectores de los conocimientos necesarios para afrontar con éxito el examen de certificación del CCNA. Su contenido, dividido en 12 capítulos, incluye la totalidad del temario oficial, destacando las siguientes características como las más significativas.

- Contenido estructurado: el contenido se desarrolla de menor a mayor dificultad, no requiriendo ningún conocimiento previo sobre los conceptos tratados.
- Facilidad de aprendizaje: el lenguaje utilizado para desarrollar cada capítulo resulta de comprensión sencilla, lo que, junto a los numerosos ejemplos incluidos en cada apartado, facilita el aprendizaje de cada uno de los fundamentos tratados en el libro.
- Enfoque práctico: toda teoría es acompañada de ejemplos y supuestos prácticos de configuración en aquellas materias que lo requieran. En este aspecto, se recomienda hacer uso de la aplicación "*Packet tracer",* desarrollada por Cisco.

- Preguntas tipo test: al finalizar cada capítulo, el estudiante podrá poner a prueba los conocimientos adquiridos gracias al test incluido en cada uno de ellos. Estos también sirven como preparación para el examen real de certificación, ya que tiene el mismo formato de cuestiones.

Gracias a todo ello, y tras finalizar el estudio y las prácticas incluidas, el lector adquiere los conocimientos necesarios para administrar y asegurar una red corporativa de tamaño medio, aplicando sobre la misma los protocolos y configuraciones más adecuadas en relación con la topología y el propósito final.

El autor

Daniel Pérez Torres nació en Santa Cruz de Tenerife en 1983. Basó sus estudios en la administración de sistemas informáticos, especializándose a posteriori en la rama de redes y seguridad, en cuyo campo posee las certificaciones Cisco CCNP, CCNA, CCNA Security, Juniper JNCIA y CompTIA Security+, entre otros muchos títulos. Propietario del blog http://desdelacli.blogspot.com y cooperador en diferentes portales web, así como instructor de CCNP, CCNA y CCNA Security desde el año 2010.

Su trayectoria profesional ha estado vinculada desde el año 2006 al servicio de la administración pública, donde actualmente pertenece al área de redes y comunicaciones, trabajando a diario con las tecnologías más avanzadas del sector como Cisco, Extreme Networks, FortiNet, ForcePoint o F5.

1 REDES INFORMÁTICAS. CONCEPTOS BÁSICOS

INTRODUCCIÓN A LAS REDES INFORMÁTICAS

El objetivo principal del CCNA consiste en que sus aspirantes obtengan los conocimientos necesarios para crear y administrar una red de tamaño medio de manera segura y eficiente. Para lograrlo, Cisco basa su estudio en análisis detallados de cada uno de los elementos que la conforman, abarcando desde las nociones más básicas hasta los protocolos más avanzados, comenzando por el concepto más esencial. ¿Qué es una red?

Una red puede ser definida como la comunicación entre un conjunto de miembros que hacen uso del mismo medio compartido con el fin de intercambiar información y recursos entre sí. Este concepto, aplicado al ámbito informático, se lleva a cabo mediante la interconexión de dispositivos, donde cada uno de ellos tomará un rol y la totalidad de los mismos definirá el tamaño y el propósito final. Dicha comunicación resulta posible gracias a la aplicación de diferentes medios, tanto físicos como lógicos. Los primeros hacen referencia a elementos de hardware, como cableado y tarjetas de red, mientras, los segundos, al software y protocolos necesarios para poder llevar a cabo la comunicación.

En cuanto al tamaño, la red más básica se compone de dos equipos, físicamente en el mismo lugar y conectados entre sí mediante un simple cable. Mientras, la más compleja puede albergar millones de host ubicados a lo largo del planeta, comunicándose gracias a multitud de dispositivos intermediarios como routers,

switchs, o firewalls, entre muchos otros. Un ejemplo bastante claro de ello es Internet.

Evidentemente, este nivel de complejidad nace como fruto de la evolución llevada a cabo a lo largo del tiempo. Así mismo, una de las primeras redes de computadoras creadas y que sin duda establece el origen de las actuales fue ARPANET, desarrollada en 1968 por el departamento de defensa de EE.UU. y utilizada para la comunicación privada entre diferentes instituciones del país. A raíz de ella, el estudio y avance de esta tecnología ha sufrido un crecimiento exponencial, hasta la actualidad, donde cualquier dispositivo puede acceder a información ubicada en cualquier parte del planeta.

Por último, una red puede ser apreciada de diferentes maneras. Para un usuario simplemente significa obtener acceso a determinados recursos o servicios, como aplicaciones corporativas o Internet. Sin embargo, desde el punto de vista de un administrador resulta más complejo, incluyendo aquellos dispositivos encargados de la comunicación, configuraciones, seguridad, diseño, protocolos, servidores, etc.

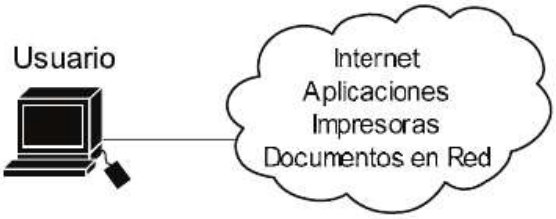


Fig. 1-1 Concepto de red para un usuario.

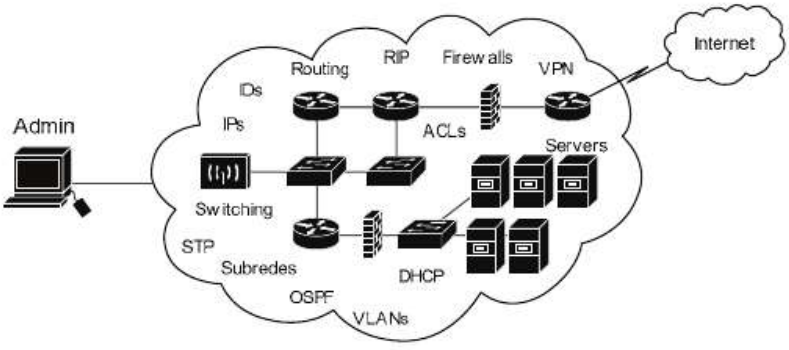


Fig. 1-2 Concepto de red para un administrador.

Como aspirante al CCNA el objetivo consiste en tomar el rol de administrador, para lo cual resulta imprescindible conocer los modelos de comunicación TCP/IP y OSI, en relación con los cuales operan la red y los diferentes protocolos aplicados en la misma.

ESTÁNDARES DE COMUNICACIÓN: TCP/IP Y OSI

La finalidad de una red informática consiste en habilitar la comunicación entre todos los dispositivos que la componen, pero ¿cómo es posible llevarla a cabo? Para lograrlo, resulta imprescindible cumplir una serie de "reglas", gracias a las cuales los datos generados por cualquier host puedan ser interpretados por el receptor de los mismos. Con dicho objetivo nacen los modelos de comunicación TCP/IP y OSI, los cuales definen los estándares, procedimientos y protocolos a aplicar para que la creación, el transporte y la entrega de datos se lleven a cabo de igual manera en cada dispositivo, sin importar ni el fabricante ni los elementos de hardware presentes en el mismo. OSI fue desarrollado por la agencia ISO *(International Organization for Standarization)* mientras que TCP/IP por voluntarios de varias universidades, siendo ambos modelos abiertos, es decir, sin coste económico ni limitaciones sobre su implementación.

Hoy día resulta prácticamente imposible encontrar dispositivos que no los soporten. Todos los sistemas operativos, incluyendo aquellos presentes en smartphones o tablets, lo implementan. Entonces, ¿cuál utilizar? Normalmente dependerá de la aplicación o sistema, pero, de ambos, el más común resulta TCP/IP, primero, porque se estandarizó con mayor rapidez, y segundo, porque la productividad de los datos es considerada más eficiente que en OSI.

A lo largo de la historia se han desarrollado diversos estándares con el mismo propósito, como SNA *(System Network Architecture)*, creado por IBM en el año 1974. Sin embargo, no han tenido éxito ni continuidad por tratarse de modelos propietarios de dichas compañías, debido a lo cual su utilización supone un coste económico, y lo que es peor, las modificaciones y actualizaciones del mismo solo pueden ser llevadas a cabo por la compañía en cuestión.

Modelo TCP/IP

TCP/IP es considerado el estándar por excelencia para llevar a cabo la comunicación en redes informáticas. Su función consiste en definir el procedimiento necesario para que los datos generados en el origen sean entregados y legibles en el destino. Para lograrlo hace uso de diferentes protocolos, cada uno de ellos con una función específica, las cuales serán analizadas a lo largo del capítulo.

Una manera de comprenderlo mejor es comparándolo con la telefonía. Si en nuestro hogar disponemos de un teléfono antiguo y lo sustituimos por otro de última generación, al conectarlo a la línea telefónica permitirá realizar y recibir llamadas de

la misma manera que el anterior, no serían necesarias ni configuraciones especiales ni la sustitución del cableado. Ello es posible gracias a que ambos hacen uso de los mismos protocolos de comunicación, los cuales han sido definidos y aprobados para su aplicación a nivel mundial. Lo mismo ocurre con TCP/IP, cualquier dispositivo que haga uso de él podrá comunicarse con otros que también lo hagan sin importar el fabricante, el modelo o el lugar donde se encuentren.

Como otros estándares de red, TCP/IP basa su modo de operar en capas, cada una de ellas con una función específica e incluyendo los protocolos necesarios para poder llevar a cabo diferentes tipos de comunicación. Estas son:

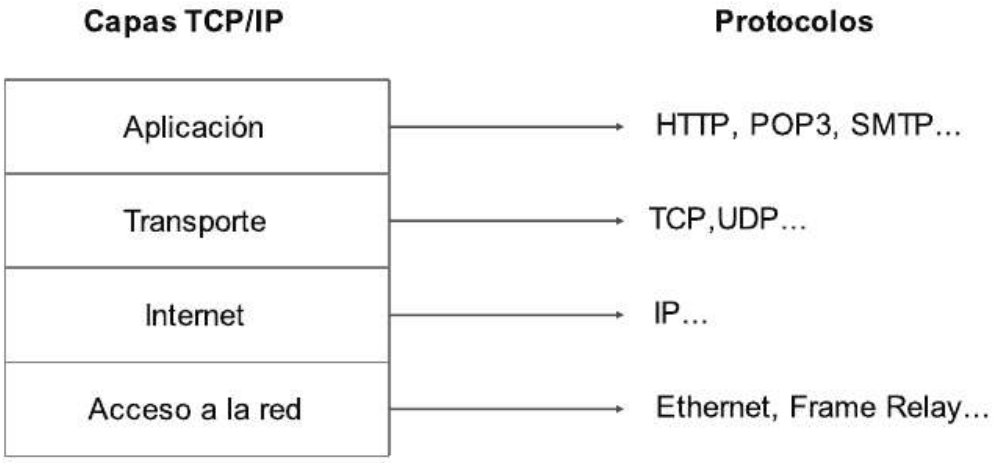


Fig. 1-3 Asociación de capas TCP/IP y sus protocolos.

En relación con las mismas queda definida la comunicación entre dos sistemas, llevando a cabo siempre el mismo procedimiento, donde los datos son generados en la capa de aplicación y enviados sucesivamente hacia las capas inferiores, aplicando cada una de ellas el protocolo correspondiente. Una vez finalizado el proceso, dichos datos son enviados al medio y recibidos por el destinatario.

Una de las grandes ventajas de TCP/IP es que es un estándar abierto, de tal manera que, si fuera necesaria la inclusión de algún nuevo protocolo, podría llevarse a cabo sin problema. Un claro ejemplo de ello fue la aparición de Word Wide Web (www), hecho que conllevó agregar HTTP en la capa de aplicación, cuyo propósito consiste en enviar solicitudes a servidores web para que estos respondan con el contenido requerido.

El proceso y las funciones llevadas a cabo en cada una de las capas son los siguientes.

CAPA DE APLICACIÓN

Es la encargada de brindar los protocolos necesarios a servicios o aplicaciones para que estos puedan iniciar el proceso de comunicación en red. Para una mejor comprensión, tomaremos como ejemplo el intercambio de mensajes entre un cliente y un servidor web, con el fin de analizar cómo son manipulados los datos en cada una de las capas para luego ser enviados al medio.

En este caso el proceso lo inicia el cliente a través de un navegador, por ejemplo, Firefox, haciendo uso del protocolo HTTP en la capa de aplicación. ¿Qué sucede cuando un dispositivo desea enviar una solicitud a un servidor web? Realmente lo que se generan son una serie de mensajes definidos por el propio protocolo, con el fin de que ambos sistemas se "entiendan", logrando con ello que la comunicación concluya con éxito. En el lado del cliente se generan mensajes GET, mientras que el servidor responde a estos mediante algún código (como el 200, con significado OK), además entra en juego otro protocolo, HTML, que define el formato de la página que se enviada.

La comunicación a nivel de capa de aplicación sería la siguiente…

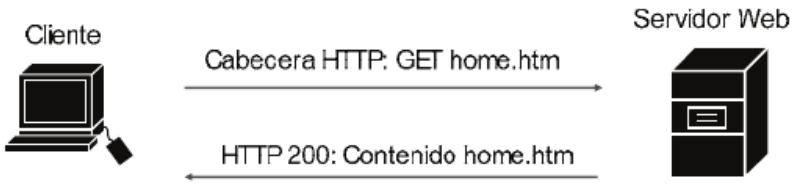


Fig. 1-4 Proceso inicial de comunicación HTTP, capa de aplicación.

Donde el navegador ha solicitado el documento "home.htm" y ha obtenido como respuesta el código 200. Ello significa que efectivamente dicho documento se encuentra almacenado en el servidor, que será enviado posteriormente. Cualquier otra circunstancia daría como resultado la generación de otro código, siendo el más común el 404, utilizado para indicar que el contenido solicitado no se encuentra disponible *(Page not Found)*.

En HTTP, el cliente genera una cabecera, que incluye información y datos propios de la capa de aplicación. Esta será recibida, analizada y respondida por su homóloga en el destino. Este modo de operar también se aplica a las diferentes capas, es decir, los datos agregados por cada una de ellas solo serán analizados y comprendidos por la misma en ambos sistemas (cliente y servidor).

La capa de aplicación no identifica al software en sí, sino los protocolos que se ejecutan en él.

CAPA DE TRANSPORTE

Una vez la capa de aplicación ha generado sus datos estos son enviados a la capa de transporte, la cual provee diferentes funciones, entre las que se encuentra identificar la aplicación a la que va dirigida la comunicación. Para ello hace uso de dos protocolos, TCP *(Transmission Control Protocol)* y UDP *(User datagram Protocol)*, ambos analizados en profundidad en este mismo capítulo.

Continuando con el ejemplo web. ¿Qué ocurriría si la solicitud enviada por el cliente no es recibida por el servidor, o viceversa? ¿Cómo sabe un dispositivo que sus datos han sido recibidos por el destinatario? TCP/IP necesita un mecanismo que garantice la entrega de datos de manera fiable de extremo a extremo. Este servicio es requerido por gran parte de las aplicaciones de red y de ello también se encarga la capa de transporte, más concretamente el protocolo TCP, que provee recuperación de errores mediante el uso de paquetes ACK (*acknowledgments*), basándose en una lógica bastante sencilla para lograrlo:

- Cuando el origen hace uso de TCP, para cada paquete enviado se espera una respuesta de confirmación de recepción por parte del destinatario, la cual se lleva a cabo mediante un mensaje ACK.
- Si transcurrido un tiempo no es recibido dicho ACK, el origen reenvía los datos.

Aplicado al ejemplo:

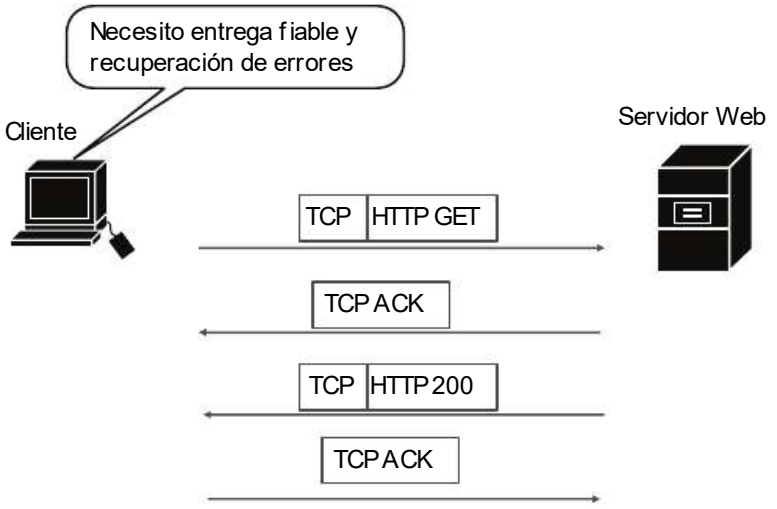


Fig. 1-5 Comunicación TCP, capa de transporte.

Si tanto cliente como servidor no hubieran recibido alguno de los ACK de confirmación, TCP reenviaría los datos nuevamente.

En este proceso se demuestra cómo un protocolo de la capa de aplicación como HTTP puede servirse de otro de la capa de transporte para agregar fiabilidad y control sobre la comunicación. Ello establece una interacción entre capas adyacentes, de tal manera que todas ellas se complementan.

Hasta ahora han sido mencionados dos conceptos que no pueden confundirse: interacción entre la misma capa en diferentes dispositivos e interacción entre capas adyacentes. La primera hace referencia a que los protocolos e información generada en una capa en el origen tan solo será analizada y comprendida por su homóloga en el destino. Mientras, la segunda se refiere a que las distintas capas en un mismo dispositivo se complementan, agregando entre todas ellas las cabeceras necesarias para que la comunicación pueda llevarse a cabo.

CAPA DE INTERNET

La capa de Internet, que se basa mayormente en el protocolo IP, es la encargada de agregar la información necesaria a los datos para que estos puedan ser enviados al destino correcto. Esta tarea se lleva a cabo gracias a las direcciones IP, las cuales identifican a cada uno de los miembros ubicados en la red.

Imagina que deseas establecer una llamada telefónica, pero desconoces el número de destino. Sin él sería imposible realizarla. Lo mismo ocurre con los datos, requieren una dirección para que la comunicación concluya con éxito.

Continuando con el ejemplo anterior, supongamos que el cliente dispone la IP 10.10.10.10 y el servidor la 20.20.20.20.

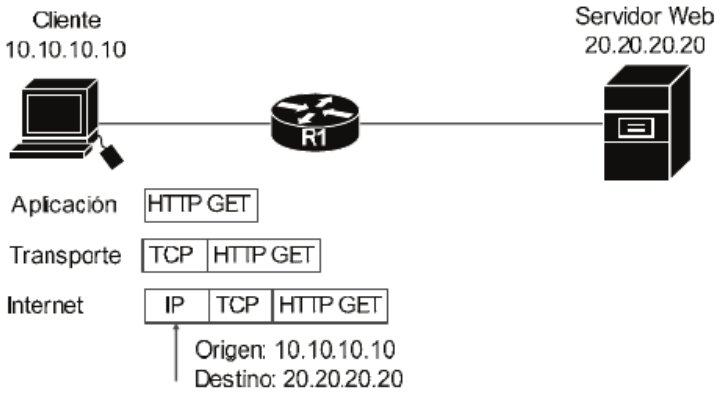


Fig. 1-6 Direccionamiento IP, capa de Internet.

En relación con la información incluida en esta capa, el router (en este caso R1) llevará a cabo el proceso de enrutamiento, mediante el cual toma la decisión de reenvío más adecuada para que los datos sean recibidos por el destinatario de la comunicación.

CAPA DE ACCESO A LA RED

Por último, el acceso a la red define el procedimiento y hardware necesario para que la entrega de datos de un extremo a otro pueda llevarse a cabo a través del medio físico disponible. Esta capa incluye una gran variedad de protocolos, que dependerán del tipo de red y conexiones, por ejemplo, para entornos LAN lo más común es aplicar Ethernet, sin embargo, en WAN resulta necesario PPP o HDLC, entre otros.

Es la última capa que atraviesan los datos antes de ser enviados al medio, por lo que debe definir el formato final de estos. Para ello, además de agregar una nueva cabecera al inicio, también incluye un tráiler al final.

Aplicado al ejemplo, y haciendo uso de una red LAN Ethernet (ETH):

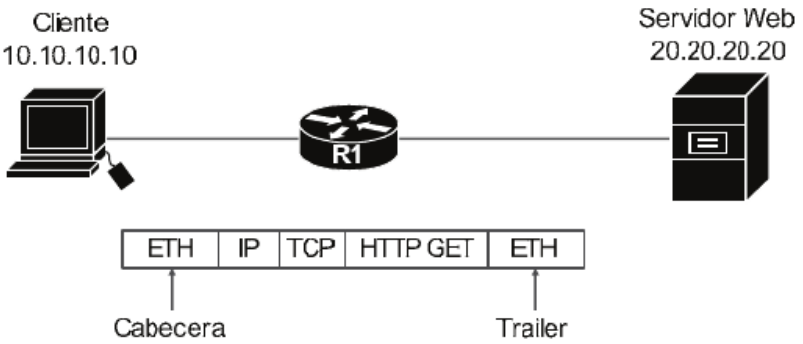


Fig. 1-7 Envío de datos, capa de acceso a la red.

Diferentes libros y webs de documentación dividen la capa de acceso a la red de TCP/IP en dos subcapas, enlace de datos (LLC) y física. Ello es debido a la comparación con el modelo OSI, el cual será objeto de estudio a continuación.

PROCESO DE ENCAPSULACIÓN Y ENVÍO DE DATOS

Como se ha analizado, cada capa agrega una cabecera con información específica a los datos. Este proceso es conocido como encapsulación, y puede ser resumido de la siguiente manera:

- *Paso 1:* Los datos generados por el software son recibidos por la capa de aplicación, que ejecutará el protocolo necesario sobre los mismos. En el ejemplo de comunicación web, HTTP.
- *Paso 2:* Una vez concluido son enviados a la capa de transporte, que agrega una nueva cabecera con información propia del protocolo aplicado. TCP, en el caso del ejemplo anterior.
- *Paso 3:* En la capa de Internet se identifican las direcciones de origen y destino, incluidas en una nueva cabecera IP.
- *Paso 4:* Por último, la capa de acceso a la red establece el formato final de los datos gracias a la cabecera y tráiler correspondientes. Comúnmente Ethernet (ETH) en redes LAN.
- *Paso 5:* Tras todo ello, son generadas las señales necesarias para su posterior transmisión a través del medio físico correspondiente (cobre, fibra, wireless…).

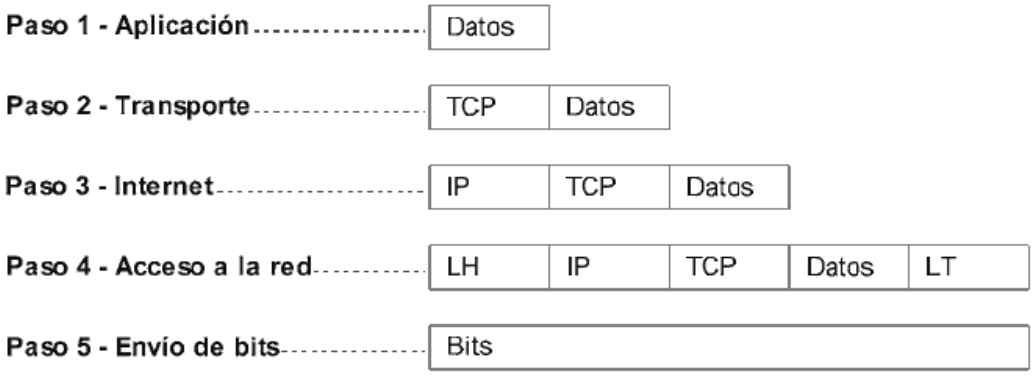


Fig. 1-8 Proceso de encapsulación en TCP/IP.

LH *(Link Header)* y LT *(Link Trailer)* corresponden a la cabecera y al tráiler.

Además, los datos, a medida que atraviesan las diferentes capas, reciben un nombre específico, siendo los siguientes:

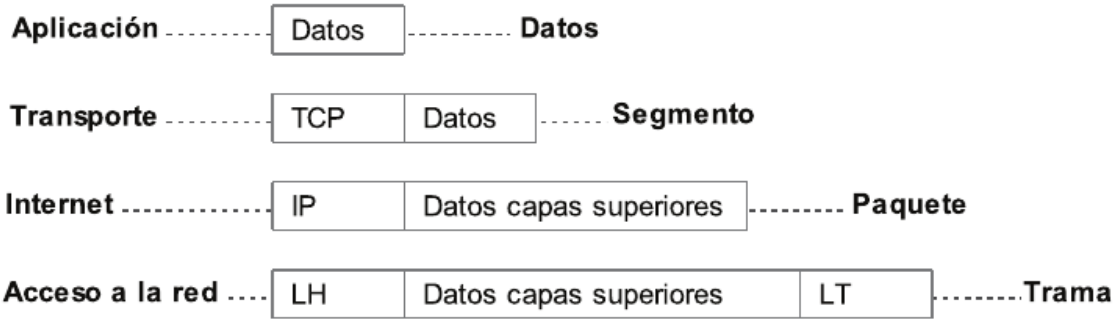


Fig. 1-9 Asociación entre capas y nombre de PDU en TCP/IP.

De ahora en adelante, cuando se haga mención a un segmento automáticamente debe ser asociado con la capa de transporte, paquete con la capa de Internet y trama (o *frame*) con la capa de acceso a la red.

Modelo OSI

OSI *(Open System Interconnection)*, creado en 1984 por ISO *(Organización Internacional para la Estandarización)*, es otro de los estándares definidos para llevar a cabo la comunicación a nivel de red. Este coincide en su finalidad con TCP/IP, es decir, definir el proceso necesario para que los datos generados en un origen sean transportados, recibidos y legibles por el destinatario de los mismos.

Una de las principales diferencias entre ambos modelos consiste en el número de capas utilizadas para lograr su objetivo, mientras que TCP/IP hace uso de 4, OSI implementa 7, siendo las siguientes:

Fig. 1-10 Capas presentes en el modelo OSI.

El emisor genera los datos en la capa de aplicación y son enviados de manera sucesiva hacia las capas inferiores, en las cuales se aplicará el encapsulamiento necesario, agregando la cabecera correspondiente en cada una de ellas para posteriormente ser enviados al medio.

En el destino, el receptor analiza la información de manera ascendente, des-encapsulando la información previamente agregada por el origen. Este proceso concluye en la capa 7 obteniendo los datos originales generados por el emisor.

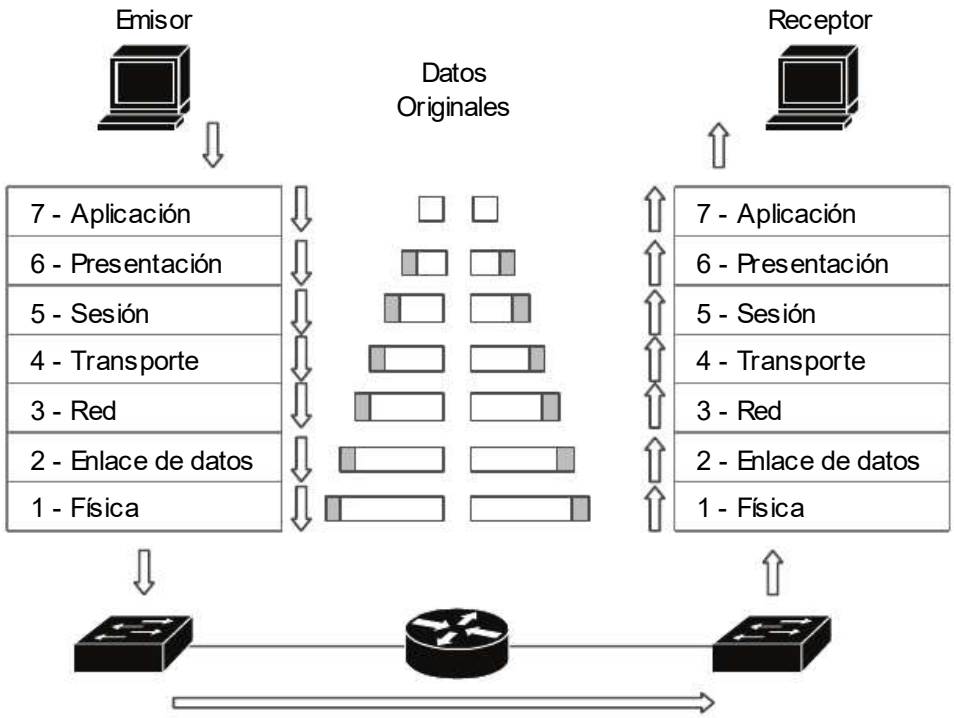


Fig. 1-11 Proceso de comunicación en el modelo OSI.

Además, y al igual que ocurre en TCP/IP, a medida que los datos atraviesan las diferentes capas son reconocidos mediante su propia PDU (*Protocol Data Unit*), siendo, en el modelo OSI, las siguientes:

Capa	**PDU**
7 Aplicación	Datos
6 Presentación	Datos
5 Sesión	Datos
4 Transporte	Segmento
3 Red	Paquete
2 Enlace de datos	Trama (o *Frame*)
1 Física	Bits

Una PDU simplemente es la nomenclatura utilizada para identificar la capa en la que se están procesando los datos, y con ello, la información manipulada.

En OSI, las capas de transporte, red, enlace de datos y física son consideradas "capas de red", mientras que aplicación, presentación y sesión, "capas de host". Cada una de ellas desarrolla una finalidad única, complementándose entre sí y realizando prácticamente las mismas funciones que en TCP/IP.

CAPA 7 - APLICACIÓN

Es la más cercana al usuario y proporciona la interactividad de este con la red. Se encarga de proveer servicio al software instalado en el dispositivo, brindándole los protocolos necesarios para llevar a cabo la comunicación. Para ello, hace uso del modelo cliente-servidor, donde el dispositivo que inicia la solicitud es el cliente y el que la recibe el servidor.

Algunos de los protocolos más conocidos presentes en esta capa son:

Protocolo	Función
DNS	Resolución de nombres de hosts a direcciones IP.
HTTP	Transferencia de páginas web.
SMTP	Envío de emails.
POP	Recepción de emails.
FTP	Transferencia de ficheros.
DHCP	Proporciona a los hosts configuración automática de red.
TELNET	Conexiones virtuales para acceso remoto.

CAPA 6 - PRESENTACIÓN

Se encarga de aplicar la conversión y codificación necesaria a los datos para que estos puedan ser legibles por el destino. Por ejemplo, definir el formato necesario a una determinada imagen (JPG, BMP, etc.).

CAPA 5 - SESIÓN

La capa de sesión establece, administra y finaliza las sesiones entre un origen y un destino.

CAPA 4 - TRANSPORTE

La capa 4 comienza a aplicar y definir funciones a nivel de red. Es la encargada de diferentes procesos, entre los que se encuentran el control de flujo, la identificación de aplicaciones, segmentación y re-ensamblaje.

El control de flujo consiste en el seguimiento de la comunicación entre el origen y el destino.

La identificación de aplicaciones es el proceso mediante el cual el dispositivo que recibe los datos conoce el software al cual va dirigida la comunicación. Esta función

es posible gracias a la utilización de puertos, los cuales hacen referencia a un valor numérico comprendido entre 0 y 65535 destinado a identificar de manera única a cada una de las aplicaciones ejecutadas en el sistema. Por ejemplo, HTTP utiliza por defecto el puerto 80… cuando un cliente solicita una web a un servidor, incluirá este como destino, de tal manera que el servidor, al recibirlo, lo leerá y reenviará al software oportuno para que responda con el documento solicitado.

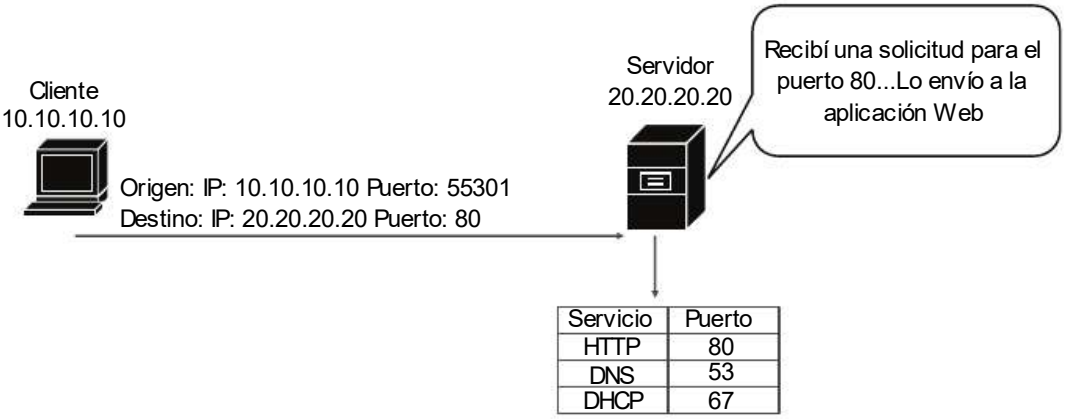


Fig. 1-12 Identificación de aplicaciones mediante número de puerto.

Esta función resulta de extrema importancia ya que sin ella la transferencia de datos no podría llevarse a cabo. En el ejemplo, el servidor ejecuta tres servicios. Cuando recibe una solicitud, ¿a cuál de ellos debe reenviarla? Gracias a que el origen ha indicado que el puerto de destino es el 80, el servidor la redirigirá al servicio web.

Los puertos son clasificados en dos tipos, bien conocidos, que son aquellos reservados para servicios y protocolos ampliamente utilizados a nivel mundial y a su vez registrados exclusivamente para ello, y dinámicos, los cuales identifican números de puerto aleatorios utilizados mayormente por los clientes para recibir la respuesta del servicio solicitado. En el ejemplo, cuando el servidor responda a la solicitud HTTP, ¿qué puerto de destino incluirá? En este caso el 55301, ya que el mismo ha sido seleccionado aleatoriamente por la aplicación del cliente (navegador web) para dicha comunicación.

Algunos ejemplos de puertos bien conocidos son:

Protocolo	**Número de puerto**
DNS	Puertos TCP y UDP 53
HTTP	Puerto TCP 80
SMTP	Puerto TCP 25
POP	Puerto TCP 110

FTP	Puertos 20 y 21 TCP
DHCP	Puertos UDP 67 y 68
TELNET	Puerto TCP 23

Otra de las funciones de la capa de transporte consiste en la segmentación y reensamblaje. Esta tarea es necesaria ya que la mayoría de redes poseen una limitación en cuanto al número total de bytes que puede contener cada PDU. En el origen, la capa 4 segmenta los datos en bloques de un tamaño adecuado para que la transmisión pueda llevarse a cabo. En el destino, la misma los reensambla y ordena antes de enviarlos a la aplicación o servicio de destino.

Por último, los protocolos que pueden ser aplicados en esta capa son TCP o UDP, ambos con características propias que serán analizadas en detalle a lo largo de este mismo capítulo.

CAPA 3 - RED

La capa de red del modelo OSI equivale a la capa de Internet de TCP/IP, desarrollando ambas las mismas funciones, basadas principalmente en ejecutar el direccionamiento lógico desde el origen hasta el destino. Para ello, y dependiendo del tipo de red, se hace uso de diferentes protocolos, como IPv4, IPv6, IPX o AppleTalk, siendo el más común IPv4 y en el futuro IPv6. Independientemente de este, la función a realizar de todos ellos coincide, y consiste en agregar a los segmentos provenientes de la capa 4 la cabecera necesaria para que los datos puedan ser enrutados hacia su destino, la cual incluirá las direcciones de origen y destino, ambas lógicas y únicas tanto a nivel local (LAN) como global (Internet).

Además, también se incluye el campo TTL *(Time to Live)*, compuesto por un valor numérico que establece el total de saltos que puede dar el paquete y disminuyendo su valor en 1 cada vez que atraviesa un router. Por ejemplo, si su valor es igual a 3 y el paquete tiene que atravesar 5 routers para llegar a su destino, cuando sea recibido por el cuarto será descartado porque su valor TTL es igual a 0.

Por último, el dispositivo por excelencia en esta capa es el router.

CAPA 2 - ENLACE DE DATOS

Una vez concluido el proceso de encapsulación en capa 3 el paquete es enviado a capa 2, que desarrolla dos funciones principales: primero, aplica el protocolo necesario en relación con el medio físico disponible, y segundo, ejecuta las técnicas

necesarias de control de acceso al medio. Para ello divide su modo de operar en dos subcapas, LLC y MAC.

- LLC *(Logical Link Control)*: su misión consiste en identificar el protocolo aplicado en capa 3 y convertir el paquete en trama.
- MAC *(Media Access Control):* agrega las direcciones físicas del origen y destino de la comunicación (direcciones MAC), controla el acceso al medio mediante diferentes técnicas y dispone funciones de control de flujo y detección de errores.

El control de acceso al medio se encarga de examinar el medio físico antes de proceder al envío de datos, con el objetivo de que no se produzcan colisiones y la transmisión resulte fiable. Para lograrlo se puede hacer uso de dos técnicas, CSMA/CD o CSMA/CA.

- En CSMA/CD el dispositivo monitoriza el medio físico en busca de una señal de datos. Si no la detecta significa que está libre, por lo tanto comienza a transmitir. Sin embargo, aun así es posible que se produzcan colisiones. En estos casos todos los dispositivos detienen el envío para volverlo a intentar pasado un tiempo aleatorio definido por cada uno de ellos. Esta técnica es la aplicada mayormente en redes Ethernet.
- CSMA/CA resulta bastante similar en cuanto a modo de operar pero agrega una pequeña característica, que consiste en el envío de una notificación antes de transmitir datos. Es decir, primero se examina el medio en busca de alguna señal, y si está libre, envía una notificación informando al resto de dispositivos su intención de utilizarlo. Esta técnica es la aplicada generalmente en tecnologías inalámbricas 802.11.

Tanto CSMA/CD como CSMA/CA se aplican en medios compartidos como Ethernet o inalámbricos. En conexiones punto a punto la utilización de estas técnicas no es necesaria ya que ambos extremos del enlace negocian el modo de transferencia antes de llevarla a cabo, por lo que resulta casi imposible que se produzcan colisiones. Estas conexiones pueden ser de dos tipos, *half-duplex* o *full-duplex*. En la primera solo un dispositivo puede transmitir; si uno envía, el otro recibe, y viceversa. Mientras, en la segunda ambos pueden realizar las dos funciones de manera simultánea a través del mismo medio.

Por último, la trama creada incluirá una nueva cabecera y tráiler, y con ella queda definido el formato final de los datos que serán transmitidos, el cual varía en función del protocolo aplicado, que a su vez depende del medio físico. Los más comunes son:

- IEEE 802.3 (Ethernet)
- IEEE 802.5 (Token Ring)
- IEEE 802.11 (Wireless)
- ITU Q.922 (Frame Relay)
- ITU Q.921 (ISDN)
- ITU HDCL (Control de enlace de datos de alto nivel)

El dispositivo de red por excelencia en capa 2 es el switch.

La gran mayoría de protocolos de enlace de datos, incluido Ethernet, permiten un máximo de 1500 bytes recibidos desde capa 3. Este tamaño es denominado MTU *(Maximum transmission unit).*

CAPA 1 - FÍSICA

La capa 1 es aquella que conecta directamente con los medios para realizar el envío de datos, desarrollando principalmente tres funciones: identificación de componentes físicos, codificación y señalización.

La identificación de componentes hace referencia al tipo de cableado, conectores, circuitos, señales inalámbricas, etc. En definitiva, el medio disponible para transportar los bits que conforman la trama desde el origen hasta el destino. Los más comunes son el cobre, fibra o inalámbricos.

La codificación es la técnica aplicada para transformar los datos en bits. Este hecho resulta importante, ya que la capa física no transporta tramas, ni paquetes, simplemente transfiere bits. Además, también se encarga de agrupar los mismos mediante algún tipo de patrón predecible que sea reconocido tanto por el emisor como por el receptor.

Una vez codificados los datos, deben ser señalizados. Esta tarea consiste en representar los bits “0” y “1” en el medio físico, aplicando para ello diferentes estándares como NRZ o Manchester. De tal manera que:

Fig. 1-13 Transmisión a través del medio físico.

Dependiendo del medio físico disponible la transferencia podrá llevarse a cabo a diferentes velocidades. Esta puede ser medida con relación a tres conceptos: ancho de banda, rendimiento y capacidad de transferencia útil.

- El ancho de banda se refiere a la capacidad total que posee un medio para transportar datos.
- El rendimiento es la velocidad real de transferencia. Generalmente no coincide con el ancho de banda debido a diferentes factores entre los que se encuentran el volumen, el tipo de tráfico que atraviesa la red o la cantidad de dispositivos conectados a ella.
- La capacidad de transferencia útil puede ser entendida como la medida y velocidad de transferencia de los datos generados en la capa de aplicación (eliminando la sobrecarga del tráfico generado por las encapsulaciones, acuses de recibo, establecimiento de sesiones, etc.) durante un periodo de tiempo determinado.

Comparación entre el modelo OSI y TCP/IP

La mayor diferencia entre ambos modelos simplemente radica en el número de capas, OSI hace uso de 7, mientras que TCP/IP de 4. Sin embargo, el procedimiento llevado a cabo para establecer, mantener y transportar la comunicación entre dispositivos resulta prácticamente el mismo. Ello es debido a que tanto OSI como TCP/IP hacen uso de protocolos ya existentes, como HTTP en aplicación, TCP o UDP para transporte, IP en capa de red, etc. Estos son los que realmente manipulan los datos, por lo tanto, los procesos llevados a cabo en ambos modelos coinciden. OSI además tiene la peculiaridad que al ser dividido en más capas, cada una de ellas está muy bien definida, mientras que TCP/IP al englobar funcionalidades, en ocasiones resulta más tedioso. La comparación entre ambos estándares es la siguiente:

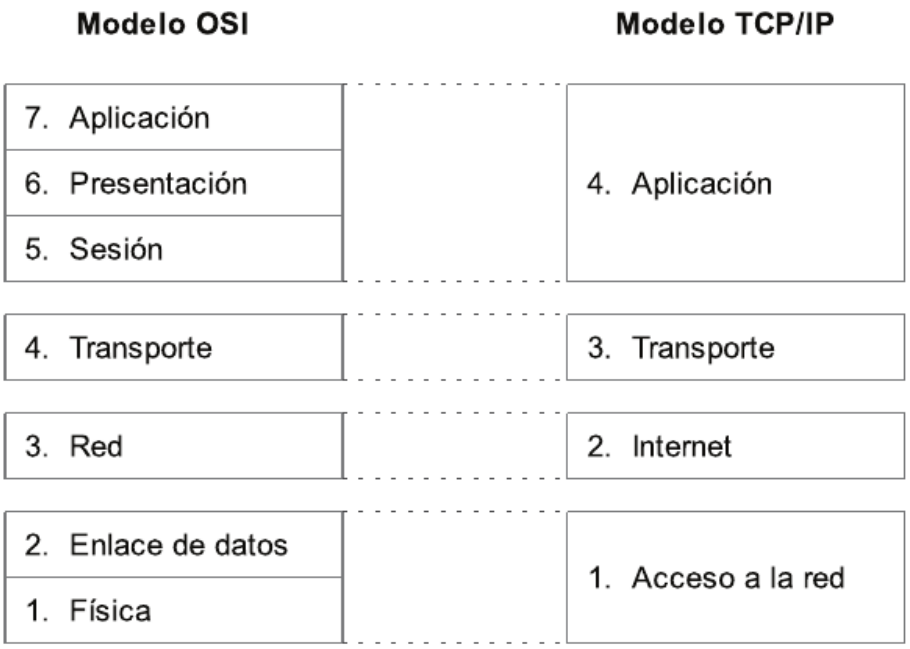


Fig. 1-14 Comparación entre los modelos OSI y TCP/IP.

El modelo de referencia utilizado tanto en este libro como en el examen de CCNA es OSI. De ahora en adelante cuando se haga mención a capa 1 deberá ser asociada a la física; capa 2, enlace de datos; capa 3, red; capa 4, transporte, etc.

REDES LAN ETHERNET

Una LAN (red de área local), en su concepto más básico, puede ser definida como un conjunto de dispositivos ubicados físicamente en el mismo lugar y comunicados entre sí a través de algún medio físico. El modo de operar de estas se basa en Ethernet, definido en el estándar IEEE 802.3, el cual establece los protocolos y tecnologías necesarios a aplicar tanto en capa física como en enlace de datos para que dicha comunicación pueda ser llevada a cabo entre todos los miembros de la red. A su vez dispone de diferentes variantes, adaptadas a cada medio y velocidad, de tal manera que el estándar necesario para una LAN de 10 Mbps no coincide con el aplicado sobre otra de 1000 Mbps.

El primero de ellos en aparecer fue 802.3i, creado en 1983, el cual define una red de 10 Mbps utilizando como medio de transporte cableado de cobre. A raíz del mismo, y en relación con la aparición de nuevas tecnologías resultó necesaria la adaptación del estándar a estas, dando lugar al desarrollo de diferentes versiones, siendo las más comunes las siguientes:

Nombre común	Estándar IEEE	Nombre alternativo	Velocidad	Medio físico, longitud máxima
Ethernet	802.3i	10Base-T	10 Mbps	Cobre, 100 m
Fast Ethernet	802.3u	100Base-TX	100 Mbps	Cobre, 100 m
Gigabit Ethernet	802.3z	1000Base-LX 1000Base-SX	1000 Mbps	Fibra, 550 m (SX), 5 km (LX)
Gigabit Ethernet	802.3ab	1000Base-T	1000 Mbps	Cobre, 100 m

Donde todas ellas comparten la misma finalidad, basada en ejecutar las funciones necesarias en capas 1 y 2 para que la transmisión de datos concluya con éxito.

En capa 1 se definen las señales, cadenas de bits, componentes físicos y distintas topologías de red. Mientras, en enlace de datos, cada una de las subcapas ejecutará funciones específicas. En este aspecto, LLC aplica el estándar 802.2, encargándose de:

- Establecer la conexión con capas superiores.
- Crear la trama en capa 2.
- Identificar el protocolo aplicado en capa 3.

Mientras, MAC hace uso de 802.3, siendo sus funciones:

- Encapsulado de datos, lo que incluye delimitación de tramas, direccionamiento físico y detección de errores.
- Control de acceso al medio.

De tal manera que:

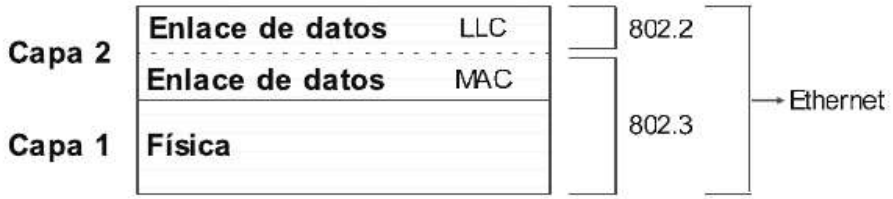


Fig. 1-15 Gestión de capas en Ethernet.

Pero si Ethernet está definido en el estándar 802.3, ¿por qué LLC opera en 802.2? La respuesta es sencilla, Ethernet realmente ejecuta ambos, pero el primero (LLC) nunca varía, es decir, realiza siempre las mismas funciones de la misma manera sin importar el medio físico al que esté conectado. Sin embargo, la subcapa MAC y la capa 1 sí que necesitan variar las técnicas aplicadas dependiendo de los componentes físicos. La función principal de Ethernet consiste en transformar los

datos para adaptarlos al medio, es por ello que se asocia directamente con el estándar 802.3.

Actualmente, la creación una LAN doméstica consta de un procedimiento bastante sencillo, donde tan solo bastarían 3 elementos:

- Una tarjeta de red (NIC) en cada uno de los PCs.
- Un hub o switch Ethernet.
- Cableado UTP para establecer la conexión entre los PCs y el hub o switch.

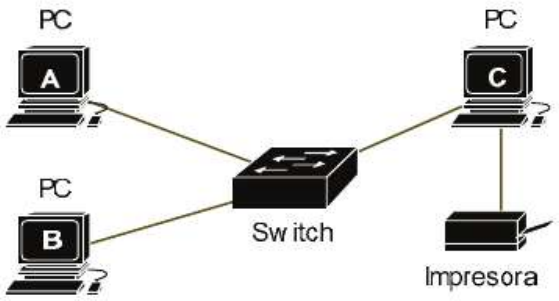


Fig. 1-16 LAN Ethernet básica.

Gracias al switch, la comunicación entre los diferentes PCs podrá llevarse a cabo, logrando con ello numerosos beneficios como la transferencia de ficheros, compartir la impresora o aplicaciones en red, entre muchos otros. Realmente, podría ser implementado cualquier servicio basado en el modelo cliente-servidor.

Evolución de las redes LAN

A día de hoy, y gracias al avance de las nuevas tecnologías, crear una LAN Ethernet no implica ninguna dificultad; sin embargo, en sus comienzos resultaba una tarea más tediosa y bastante diferente a las topologías actuales. Antes de la aparición de 10BASE-T, Ethernet operaba en los estándares 10BASE5 y 10BASE2. Estos, aunque con diferentes características, se basaban en la utilización de cableado coaxial como medio físico, implementando una topología basada en un bus de comunicación al cual conectaba cada uno de los PC, logrando gracias a ello la creación de la red sin necesidad de switchs o hubs.

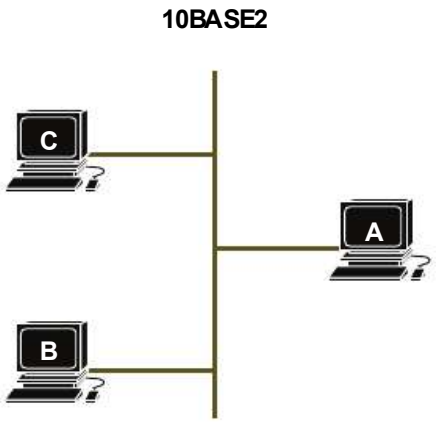


Fig. 1-17 Topología 10BASE2.

Cualquier comunicación entre sus miembros se lleva a cabo a través del bus mediante el envío de señales eléctricas. El problema reside en que estas son recibidas por todos los dispositivos, de tal manera que si A envía datos a B, también serán recibidos por C.

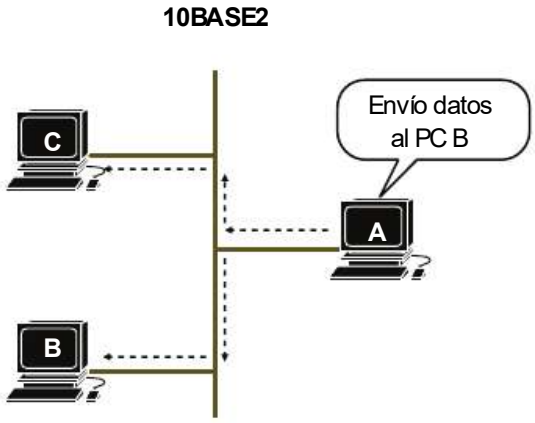


Fig. 1-18 Envío de datos en 10BASE2.

Este hecho evidentemente supone un problema de seguridad importante, que en aquellos años probablemente no tendría demasiada importancia pero que actualmente resultaría totalmente inviable. A ello hay que sumarle las constantes colisiones que se producían, debido a que los cables coaxiales están compuestos por un núcleo que tan solo permite el envío de una señal, por lo que si dos dispositivos utilizan el medio de manera simultánea, se generarían dos señales, sobreposicionándose una sobre la otra y haciendo ilegible los datos.

Si A transmite, y a su vez lo hace B…

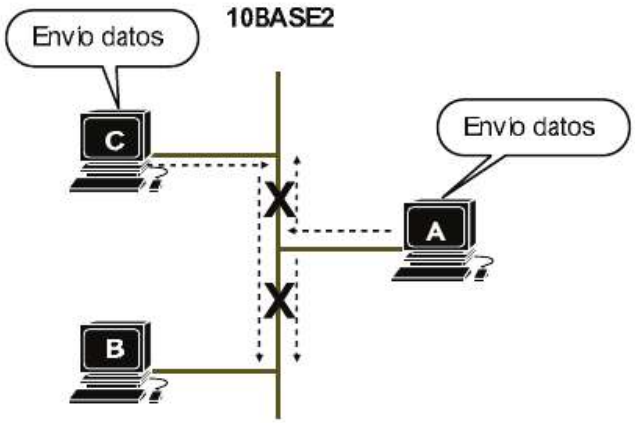


Fig. 1-19 Colisión en 10BASE2.

Para solucionar tal inconveniente fue necesario desarrollar diferentes técnicas de control de acceso al medio, compuestas por algoritmos capaces de asegurar que tan solo un dispositivo envía tráfico a la red. Uno de ellos es CSMA/CD *(Carrier Sense Multiple Access with Collision Detection),* siendo sus funciones las siguientes:

- Cuando un dispositivo desea enviar tráfico, primero comprueba que el medio esté libre. Si es así, hace uso de él, de lo contrario volverá a comprobarlo transcurrido un tiempo.
- En caso de colisión, los dispositivos involucrados deben detener la transmisión y esperar un tiempo aleatorio calculado por Ethernet mediante la ecuación matemática. Dicho período resulta prácticamente imposible que coincida en ambos. Una vez transcurrido el mismo, pueden volver a comprobar el medio, y si está libre, hacer uso de él.

¿Cómo es posible que se produzcan colisiones si solo se envían datos cuando el medio está libre? En el ejemplo analizado el bus es corto y está compuesto por pocos dispositivos, pero en una red mucho más amplia es posible que un PC ubicado en un extremo aún no haya recibido la señal de otro, procediendo también a enviar datos. En algún punto del cable se producirá la colisión.

El ejemplo se ha basado en 10BASE2, pero resulta exactamente igual sobre 10BASE5.

Como en todos los estándares, estos también disponen de limitaciones en cuanto a distancia del cableado, siendo la longitud máxima de 500 m para 10BASE5, y de 185 m para 10BASE2. La velocidad en ambos casos es de 10M bps. En ocasiones, sobre todo en 10BASE2, esta distancia puede resultar insuficiente, hecho que se

soluciona gracias a la instalación de repetidores a lo largo del bus, los cuales desarrollan la función de limpiar la señal y repetirla, sin ni siquiera interpretarla.

Actualmente este tipo de topologías han caído en desuso pero sin duda marcan el inicio de las redes modernas.

LAN ETHERNET 10BASE-T

Con el paso del tiempo, y con el objeto de mejorar el rendimiento y las capacidades de las LAN, comenzaron a desarrollarse nuevos componentes físicos y con ellos un nuevo estándar Ethernet para soportarlos. Más concretamente el 802.3i, también conocido como 10BASE-T.

Este incluye multitud de avances respecto a sus antecesores, gracias en gran parte a la utilización de cableado UTP y de un dispositivo intermediario, agregando numerosos beneficios a la red como su facilidad de instalación, mayor disponibilidad, escalabilidad y reducción de costes.

En este caso, para la puesta en marcha de la LAN bastará con conectar los PC al hub, siendo este el encargado de establecer la comunicación entre todos ellos. A este diseño se le conoce como una topología estrella, que puede ser definida como aquella que crea una red con relación a un elemento central de conexión, que gestionará la comunicación de los dispositivos. En 10BASE-T es el hub, pero en redes actuales suele ser el switch.

La misma topología que en 10BASE2 pero esta vez haciendo uso de 10BASE-T quedaría definida de la siguiente manera.

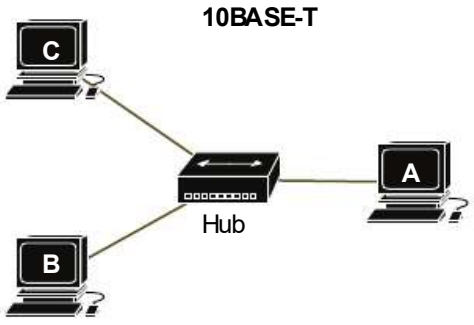


Fig. 1-20 Topología 10BASE-T.

Donde el bus es sustituido por un hub, al igual que el cableado utilizado para establecer la conexión (antes coaxial y ahora UTP). Sin embargo, este cambio de diseño no supone ninguna mejora en cuanto al modo de operar, siendo el

funcionamiento en ambos estándares prácticamente el mismo. Cuando un hub recibe señales por alguno de sus puertos serán reenviadas a través de todos los restantes, causando el mismo problema de seguridad que en 10BASE2. Además, tampoco se resuelve el problema de colisiones, siendo necesario aplicar técnicas de control de acceso al medio como CSMA/CD.

En cuanto a diferencias, la más destacada entre ambos modelos, además de los ya mencionados elementos físicos, radica en que el hub también limpia y regenera la señal, tarea de la cual se encargaban los repetidores.

Continuando con el ejemplo anterior. Si A envía datos a B, también serán recibidos por C.

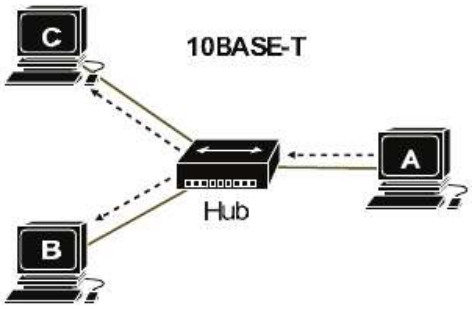


Fig. 1-21 Envío de datos en 10BASE-T.

El hub es considerado un dispositivo de capa 1, ya que no interpreta los datos, tan solo regenera la señal y la reenvía.

MEJORAS DE RENDIMIENTO GRACIAS AL SWITCH

Los hubs actúan como bus y repetidor en un solo dispositivo, aplicando a su vez técnicas de control de acceso al medio como CSMA/CD. Gracias a ello se logró un buen funcionamiento en redes Ethernet pero aun así mantenía las siguientes limitaciones:

- Tanto en 10BASE2, 10BASE5 como 10BASE-T, el ancho de banda disponible era compartido por todos los equipos que formaban parte de la red.
- Aun aplicando CSMA/CD es posible que se produzcan colisiones, lo que conlleva gasto de ancho de banda y retraso en las comunicaciones.
- El modo de transmisión utilizado era *half-duplex*, con lo cual los dispositivos no podían enviar y recibir datos de manera simultánea.

Con el objetivo de poner fin a todo ello nace un nuevo dispositivo, el switch, un elemento imprescindible en redes actuales y que soluciona las limitaciones anteriores. Estos hacen uso del mismo cableado que los hubs (UTP) y la misma topología estrella, pero agregando los beneficios enumerados a continuación.

Primero, tienen la capacidad de reenviar los datos solo al destinatario real de estos, lo que se traduce en una capa de seguridad y en un mejor aprovechamiento del ancho de banda. Si en la topología anterior sustituimos el hub por un switch y el PC A envía datos a B…

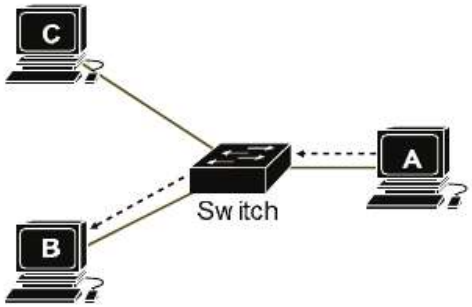


Fig. 1-22 Reenvío de datos ejecutado por el switch.

¿Cómo lo logra? Uno de los campos que forman parte de la cabecera en capa 2 es la dirección física del destino, la cual identifica a un único dispositivo. Cuando el PC A envía datos a B incluye en dicho campo la MAC utilizada por este, de tal manera que cuando el switch recibe la trama, la analiza y lee el destino, reenviando los datos solo a través del puerto que conecta con B. Ello es posible gracias a una tabla almacenada en su memoria, la cual asocia cada número de puerto con la MAC del dispositivo vinculado a él. Esta función será analizada en profundidad en capítulos posteriores.

Como los switchs tienen que analizar la trama y en relación con ella tomar una decisión de reenvío son considerados dispositivos de capa 2.

Segundo, permiten enviar y recibir tráfico de manera simultánea a través del mismo enlace, es decir, hacen uso de transmisión *full-duplex*, por lo que el PC A, que actualmente está enviando datos a B, también podría recibirlos al mismo tiempo, por ejemplo, del PC C…

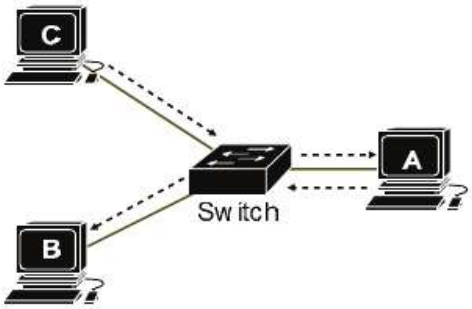


Fig. 1-23 Transmisión full-duplex.

Tercero, establecen un ancho de banda dedicado para cada puerto y por lo tanto para cada PC. En estándares anteriores este era compartido por todos los dispositivos que se conectaban al bus o hub, disminuyendo en gran medida el rendimiento de la red.

Por último, se soluciona el problema de colisiones, el cual se genera cuando el mismo medio es utilizado de manera simultánea por diferentes dispositivos. Para solventarlo, los switchs establecen una conexión punto a punto en cada una de sus interfaces, de tal manera que cada una de ellas dispone de un medio físico dedicado (no compartido), logrando así que las colisiones resulten prácticamente inexistentes y siendo innecesario aplicar técnicas de control de acceso al medio como CSMA/CD.

Dicho concepto es denominado como dominios de colisión. Los hub crean un solo dominio de colisión para todos los dispositivos conectados, mientras que los switchs uno por cada interfaz.

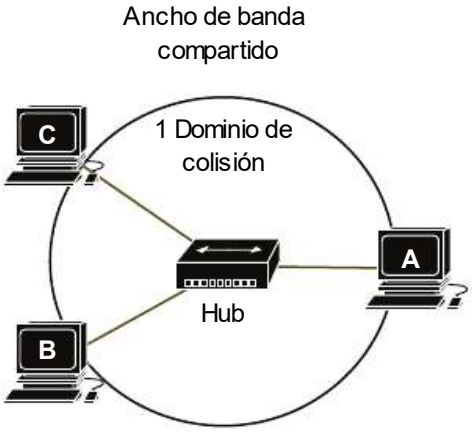


Fig. 1-24 Dominios de colisión en hub.

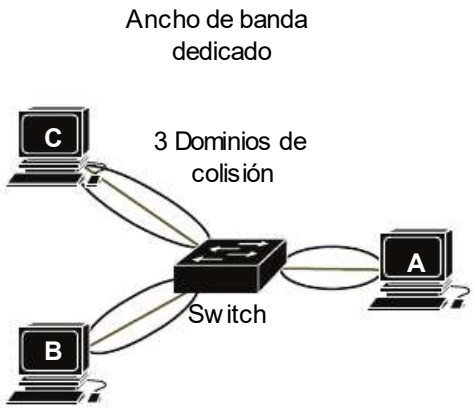


Fig. 1-25 Dominios de colisión en switch.

En cuanto al estándar IEEE utilizado por los Switchs, varía dependiendo del medio y ancho de banda aplicado, los hay desde 10 Mbps hasta 1000 Mpbs, operando en fibra o cobre, por lo tanto, se aplicará el que se considere más oportuno en cada caso.

Elementos en el diseño de LANs Ethernet

A la hora de diseñar una LAN Ethernet resulta imprescindible evaluar cada uno de los elementos que pueden afectar o mejorar el rendimiento de esta. Una mala implementación se traduce en el desaprovechamiento de ancho de banda, fallos de seguridad, bucles, etc. Problemas que se resuelven aplicando un diseño bien estructurado. La mejor manera de lograrlo es llevando a cabo una segmentación de la red, utilizando para ello dominios de colisión, de broadcast y VLANs.

DOMINIOS DE COLISIÓN

Los dominios de colisión, ya analizados en párrafos anteriores, pueden ser definidos como un único medio físico compartido entre varios dispositivos, cuantos más accedan, mayor probabilidad de colisión y menor ancho de banda disponible. Por lo tanto, un buen diseño de red tratará de limitarlos lo máximo posible, tarea de la cual se encargan los switchs y routers.

Cada dispositivo genera los siguientes dominios de colisión:

- Hub: Un único dominio de colisión para todos los dispositivos conectados a él.
- Bridge: Un dominio de colisión por cada interfaz.

- Switch: Un dominio de colisión por cada interfaz, creando una conexión punto a punto en cada una de ellas.
- Router: Un dominio de colisión por cada interfaz.

Siendo, entre todos ellos, el switch el más recomendable para llevar a cabo esta función.

¿Cuántos dominios de colisión se han generado en la siguiente topología?

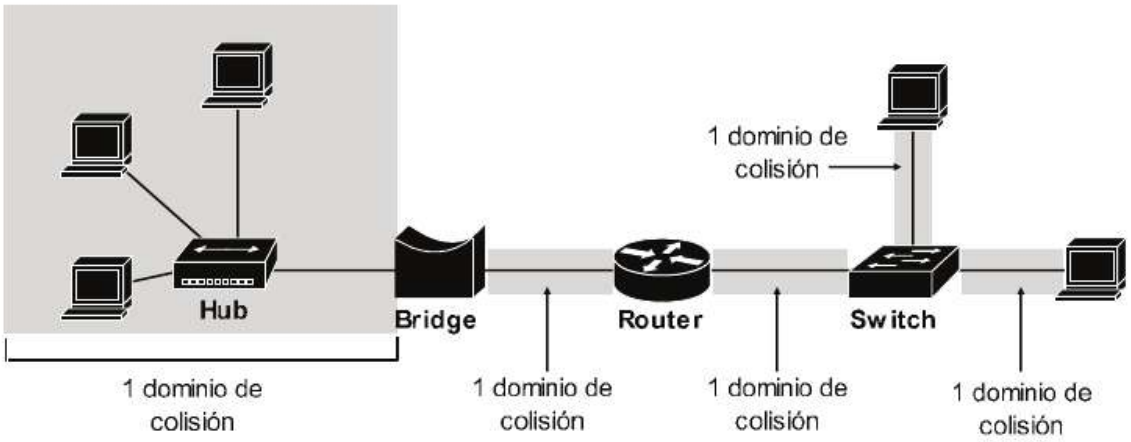


Fig. 1-26 Dominios de colisión entre diferentes dispositivos.

DOMINIOS DE BROADCAST

Hacen referencia al alcance que tendrá un paquete broadcast enviado desde algún segmento de la red, es decir, qué dispositivos lo procesarán y dónde se detiene. El dispositivo por excelencia para limitarlos es el router, ya que por defecto no reenvían este tipo de paquetes a través de sus interfaces.

Cada dispositivo genera los siguientes dominios de broadcast:

- Hub: Un único dominio de broadcast para todos los dispositivos conectados a él.
- Bridge: Un único dominio de broadcast para todos los dispositivos conectados a él.
- Switch: Un único dominio de broadcast para todos los dispositivos conectados a él, salvo que se definan VLANs, concepto que será analizado en párrafos posteriores.
- Router: Un dominio broadcast por cada interfaz.

En el siguiente ejemplo, el router crea dos dominios de broadcast, uno por cada interfaz, lo cual significa que si algún PC del segmento de red A genera un paquete cuya dirección de destino sea broadcast, tan solo será recibido por los dispositivos pertenecientes a dicho segmento, el router no lo reenviará a través de su interfaz Fa0/1. Exactamente lo mismo sucedería si el paquete es iniciado en el segmento B, no sería reenviado hacia el A.

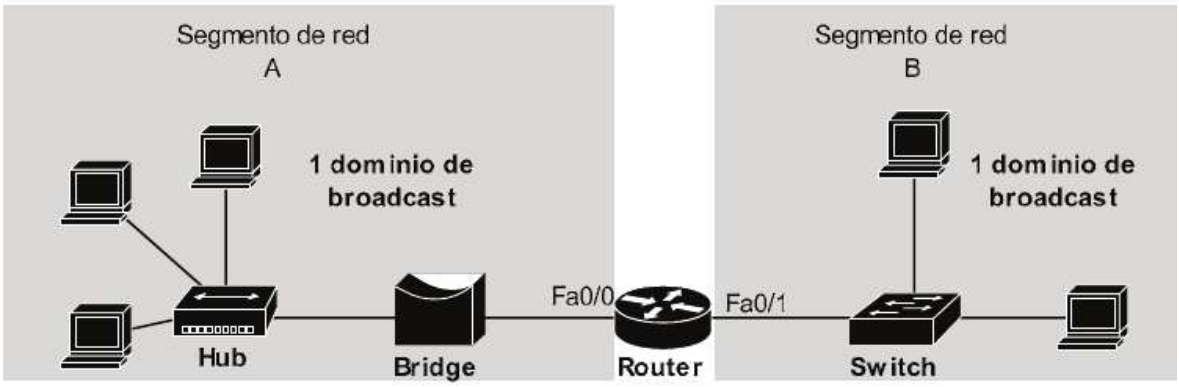


Fig. 1-27 Dominios de broadcast.

Si el router fuera eliminado, y el switch y el bridge conectaran directamente, tan solo existiría un dominio de broadcast.

IMPORTANCIA DE LOS DOMINIOS DE COLISIÓN Y BROADCAST

Definir bien ambos aspectos a la hora de diseñar una LAN resulta una tarea de vital importancia si se desea obtener un buen rendimiento de la misma.

En cuanto a los dominios de colisión, todos los dispositivos ubicados en el mismo comparten medio físico, y, por lo tanto, ancho de banda. Este hecho evidentemente disminuye el rendimiento de la red, no solo por compartir ancho de banda, si no también porque se producirán colisiones y con ello retrasos en las comunicaciones. Además, el nivel de seguridad resulta bastante bajo. Siempre que sea posible debe evitarse el uso de hubs y bridges, sustituyéndolos por switchs.

En cuanto a los broadcasts, un dominio de broadcast demasiado amplio genera una bajada de rendimiento tanto a nivel de red como de host. Ello es debido a que el mismo paquete recorre múltiples dispositivos y medios físicos, afectando al ancho de banda y velocidad de procesamiento tanto de dispositivos intermediarios como de hosts finales.

Una comparación entre ambos podría ser la siguiente:

Característica	Hub	Switch	Router
Crea múltiples dominios de colisión	No	S í	Sí
Aumenta y aprovecha mejor el ancho de banda	No	S í	Sí
Crea múltiples dominios de broadcast	No	No	Sí

VLANS (VIRTUAL LANS)

Sin duda, el elemento más importante en cuanto a segmentación se refiere son las VLANs. Estas pueden ser definidas como una tecnología en capa 2 utilizada para dividir la red en segmentos con el fin de obtener beneficios como mayor seguridad, facilidad de administración y creación de dominios de broadcast, siendo generadas y gestionadas desde los switchs.

El concepto resulta sencillo, cada interfaz es configurada para formar parte de una VLAN, y solo podrán comunicarse entre sí aquellas que pertenezcan a la misma. La comunicación entre diferentes VLANs no podrá llevarse a cabo sin la existencia de un router que lo permita, por lo que a su vez se crea un dominio de broadcast por cada una de ellas.

Por ejemplo, el siguiente switch ha sido configurado de tal manera que:

- Las interfaces Fa0/1 y Fa0/2 formen parte de la VLAN 1.
- Las interfaces Fa0/3 y Fa0/4 formen parte de la VLAN 2.
- Las interfaces Fa0/5, Fa0/6 y Fa0/7 formen parte de la VLAN 3.

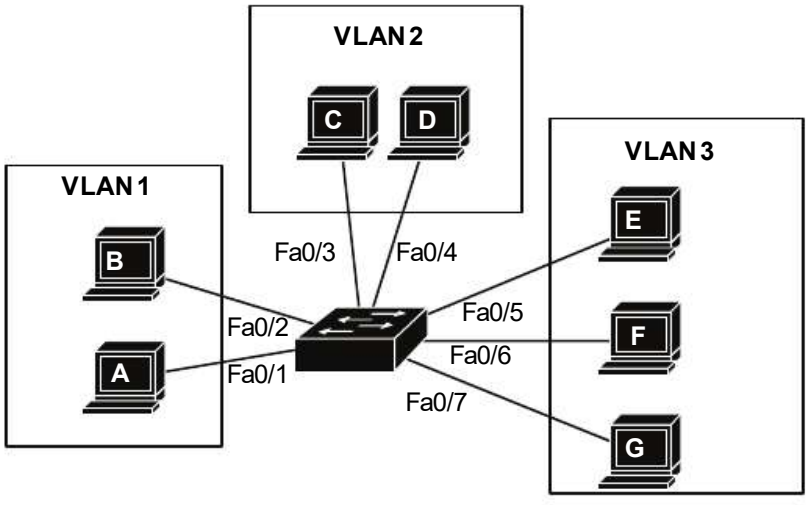


Fig. 1-28 Diseño basado en VLANs.

Las características y la configuración de las VLANs serán objeto de estudio en el capítulo 2 *"Configuración de switchs Cisco"*.

REDUNDANCIA

La redundancia hace referencia a la capacidad que tiene una red de mantenerse operativa aun cuando se produzcan caídas en alguno de sus servicios o dispositivos, es decir, la tolerancia a fallos y disponibilidad de esta. Un ejemplo puede ser la conexión entre dos switchs, si se utilizan dos enlaces y uno falla, la comunicación no se interrumpe gracias al restante.

Una buena práctica para optimizar el rendimiento y mejorar la redundancia consiste en distribuir la red en capas (acceso, distribución y núcleo). Cada una de ellas con una función y todas conectadas entre sí.

La capa de acceso está compuesta por switchs que interactúan directamente con los dispositivos finales, como PCs de usuarios, impresoras, etc. Estos normalmente ofrecen acceso a la red mediante tecnologías 100BASE-T o 1000BASE-T.

La capa de distribución está compuesta por switchs, normalmente de mayor capacidad que los anteriores y en ella se aplica la redundancia, utilizando para ello múltiples enlaces hacia las diferentes capas, logrando así un ancho de banda superior y aplicando normalmente tecnologías 1000BASE-T o variantes de fibra (1000BASE-LX o 1000BASE-SX).

La capa núcleo alberga dispositivos críticos de red y ejecuta funciones como el enrutamiento en capa 3 o la conexión con redes externas. Los dispositivos ubicados en ella son routers o switchs de alto rendimiento que comunican con la capa de distribución a través de múltiples enlaces con el objetivo de aumentar el ancho de banda y proveer redundancia.

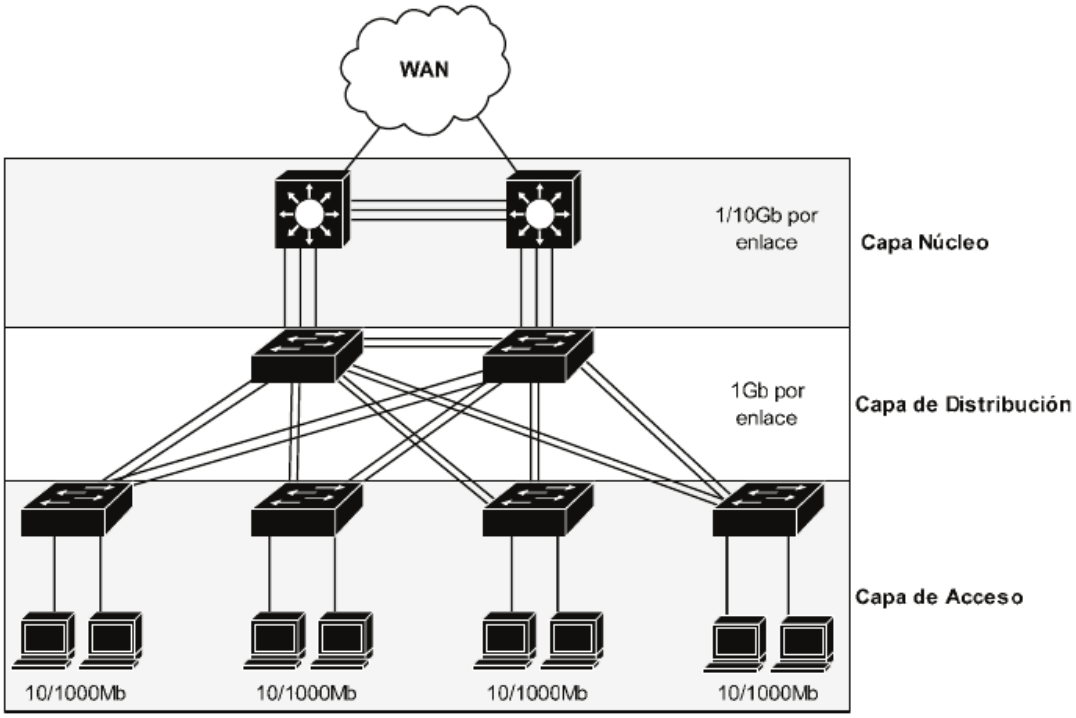


Fig. 1-29 Diseño de red distribuido en capas.

La tecnología Ethernet aplicada en cada capa suele variar, lo que equivale a diferentes velocidades y longitudes de cableado. Estas hay que tenerlas muy en cuenta para que el rendimiento de la red sea óptimo. Un cable que supere su longitud máxima tendrá como consecuencias pérdidas de señal, errores de transmisión y por lo general fallos de comunicación entre los dispositivos de ambos extremos.

Las características de los medios físicos más comunes y su tecnología Ethernet son:

Ethernet	Velocidad	Medio físico	Longitud máxima
10BASE-T	10Mb	CAT3 o superior, dos pares de cobre	100 metros
100BASE-T	100Mb	UTP CAT5 o superior, dos pares de cobre	100 metros
1000BASE-T	1Gb	UTP CAT5 o superior, 4 pares de cobre	550 metros
1000BASE-SX	1Gb	Fibra multimodo, luz	550 metros
1000BASE-LX	1Gb	Fibra multimodo, luz	550 metros
1000BASE-LX	1Gb	Fibra monomodo, luz	5 Km

AUTONEGOCIACIÓN

La autonegociación no consta como un elemento propio del diseño de una red, sin embargo, sí que influye en el establecimiento, o no, del enlace. Un cable Ethernet conecta dos extremos, con un dispositivo en cada uno de ellos, siendo a su vez necesario que ambos operen con el mismo protocolo para que la comunicación pueda realizarse, de lo contrario no sería posible. Por ejemplo, un puerto de un switch en 1000BASE-T no podría comunicarse con un PC que aplique 10BASE-T.

Con el fin de evitar dicho problema nace la autonegociación, gracias a la cual los dispositivos intervinientes en un enlace deciden de manera conjunta qué tecnología aplicar y qué modo de transferencia utilizar *(duplex* o *full duplex)* antes de enviar cualquier tipo de tráfico. Esta característica, aunque es ejecutada por defecto en todos los dispositivos Cisco, puede ser deshabilitada, acción que no es recomendable llevar a cabo en interfaces que conecten con dispositivos finales. Sin embargo, sí puede ser una buena opción para aquellos enlaces entre dos switchs, siendo necesario configurar manualmente la velocidad en ambas interfaces (evidentemente la misma).

Por último, también existe la posibilidad de que un extremo aplique autonegociación y el otro no. En este caso, el dispositivo que no autonegocia debe ser configurado con las mismas características del que negocia, tanto en velocidad como en modo de transferencia, de lo contrario la comunicación entre ambos no se llevará a cabo.

Cableado UTP

UTP *(Unshielded twisted pair)* es el medio físico utilizado por excelencia para los estándares 10BASE-T, 100BASE-TX y 1000BASE-T. El cable en sí consiste en 4 pares de cobre trenzados e identificados por diferentes colores, que a su vez son protegidos por un revestimiento exterior. Ambos extremos finalizan en un conector RJ-45, gracias al cual se establece la conexión física con los dispositivos.

Fig. 1-30 Cableado UTP.

Fig. 1-31 Conectores RJ-45.

Fig. 1-32 Puertos RJ-45.

La transmisión de datos fluye a través de los pares de cobre, los cuales establecen la codificación necesaria en relación con la distribución de los colores. Dependiendo de la misma, el cable podrá ser directo o cruzado, siendo necesario cada uno de ellos para establecer las siguientes conexiones entre dispositivos:

Tipo de cable	**Conexión**
Directo	- Switch a Router - PC a Switch - PC a Hub
Cruzado	- Switch a Switch - Switch a Hub - Hub a Hub - Router a Router - PC a PC - PC a Router

La creación de ambos se basa en dos estándares, el T568A y T568B, los cuales simplemente identifican el color de cable que debe ubicarse en cada PIN del conector RJ-45 (8 pines en total). Un cable directo aplicará el mismo estándar en ambos extremos, normalmente el T568B, mientras que el cruzado aplicará uno diferente en cada uno de ellos. Las codificaciones para ambos casos son las siguientes.

Codificación para cable directo:

Extremo 1 (T568B)	**Extremo 2 (T568B)**
Pin 1: Blanco Naranja	Pin 1: Blanco Naranja
Pin 2: Naranja	Pin 2: Naranja
Pin 3: Blanco Verde	Pin 3: Blanco Verde

Pin 4: Azul	Pin 4: Azul
Pin 5: Blanco Azul	Pin 5: Blanco Azul
Pin 6: Verde	Pin 6: Verde
Pin 7: Blanco Marrón	Pin 7: Blanco Marrón
Pin 8: Marrón	Pin 8: Marrón

Codificación para cable cruzado:

Extremo 1 (T568B)	**Extremo 2 (T568A)**
Pin 1: Blanco Naranja	Pin 1: Blanco Verde
Pin 2: Naranja	Pin 2: Verde
Pin 3: Blanco Verde	Pin 3: Blanco Naranja
Pin 4: Azul	Pin 4: Azul
Pin 5: Blanco Azul	Pin 5: Blanco Azul
Pin 6: Verde	Pin 6: Naranja
Pin 7: Blanco Marrón	Pin 7: Blanco Marrón
Pin 8: Marrón	Pin 8: Marrón

De todos ellos, los pines 1 y 2 son utilizados para transmitir datos, mientras que el 3 y el 6 para recibirlos. Los pines 4, 5, 7 y 8 no son útiles en 10BASE-T y 100BASE-TX, sin embargo, sí que lo son para 1000BASE-T, ya que gracias a ellos se logra la velocidad de 1G bps sobre UTP.

En relación con las diferentes aplicaciones para dichos cables se puede concluir que en una misma LAN será necesaria la instalación de ambos tipos, por ejemplo:

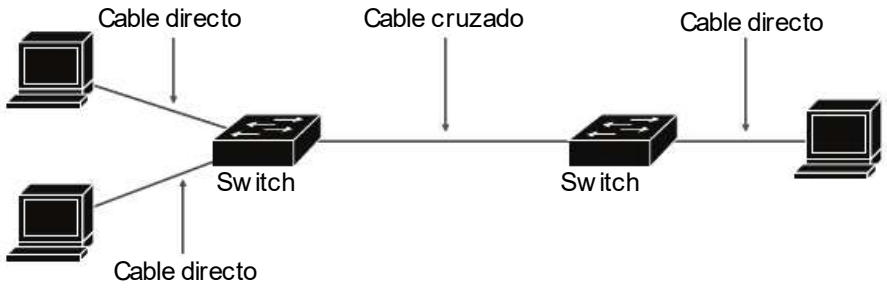


Fig. 1-33 Tipo de cableado entre dispositivos.

En cuanto a velocidad de transmisión se refiere, dependerá del tipo de cable UTP utilizado. Estos son divididos en categorías, por ejemplo, el Cat5 se utiliza

comúnmente en instalaciones Fast Ethernet 100BASE-TX y soporta hasta 100 Mbps. Para velocidades mayores se puede optar por el Cat5e (categoría 5 mejorado) o Cat6, ambos soportan hasta 1 Gbps.

Protocolos de enlace de datos

A excepción de los switchs, todas las funciones analizadas en párrafos anteriores son ejecutadas en capa 1. Sin embargo, Ethernet también opera en enlace de datos, encargándose principalmente de crear la trama, llevar a cabo el direccionamiento físico y proveer detección de errores.

DIRECCIONAMIENTO

El direccionamiento en Ethernet consiste en la identificación, en capa 2, de los dispositivos intervinientes en una comunicación, teniendo como objetivo que la trama generada por el origen tan solo sea recibida por los destinatarios correctos. Para lograrlo se agregan dos campos en la cabecera de la trama, utilizados para identificar las direcciones físicas tanto del origen como del destino, las cuales, como se ha nombrado con anterioridad, corresponden a las direcciones MAC. Estas utilizan un formato de 12 dígitos hexadecimales (6 bytes), pudiendo ser representados de diferentes maneras. Por ejemplo, Cisco la divide en 3 bloques de 4 dígitos cada uno, sin embargo, Microsoft lo hace en 6 bloques de 2 dígitos. Por lo tanto, para una misma dirección, diferentes representaciones.

Dispositivos Cisco: 000A.4D5F.00AA
Sistemas operativos: 00-0A-4D-5F-00-AA

La MAC es almacenada en un chip ROM ubicado en cada tarjeta de red y su valor es asignado por el fabricante, logrando que cada una de ellas obtenga un valor único a nivel mundial. Para dicho objetivo, IEEE creó un estándar, el cual consiste en dividir la dirección en dos partes, la primera de 3 bytes (24 bits) que identifica al fabricante (*organizationally unique identifier – OUI*), y la segunda, de otros 3 bytes, que el fabricante en cuestión irá utilizando para enumerar cada tarjeta sin repetir nunca el mismo valor (*vendor assigned*).

Imaginemos que IEEE otorga a Cisco el OUI 00.AA.00. Esto quiere decir que todas las direcciones MAC de los dispositivos de red que Cisco fabrique deben comenzar con dicha numeración hexadecimal. Los siguientes 24 bits se asignarán de tal manera que no se repita nunca el mismo valor.

Por ejemplo, la MAC 00.AA.00.00.00.01

	Organizationally Unique Identifier - OUI	**Vendor Assigned**
Valor	00.AA.00	00.00.01
Tamaño en hexadecimal	6 dígitos hexadecimales	6 dígitos hexadecimales
Tamaño en bits	24 bits (3 bytes)	24 bits (3 bytes)

Dependiendo del propósito y destinatarios de la comunicación, existen 3 tipos de direcciones en capa 2:

- u*nicast*: Son aquellas que identifican la MAC de un único dispositivo. La comunicación será recibida por un solo destinatario.
- *multicast*: Son direcciones especiales utilizadas por protocolos o aplicaciones y cuyo destino define un grupo de dispositivos de la LAN. Estas siempre comienzan con los dígitos 01.00.5E, mientras que el resto de la numeración es aplicada por el protocolo o aplicación en cuestión.
- *broadcast*: La dirección broadcast en capa 2 es la FF.FF.FF.FF.FF.FF e indica que la trama tiene como destino a todos los dispositivos pertenecientes a la LAN.

Las MAC también pueden ser nombradas como: direcciones LAN, direcciones Ethernet, direcciones de hardware, direcciones físicas, direcciones de capa 2, direcciones BIA *(burned-in addresses)* o direcciones UAA *(universally administered addresses)*. A lo largo del libro se hará uso de los términos MAC, direcciones físicas o direcciones de capa 2, ya que son las nomenclaturas más comunes.

ETHERNET FRAMING

El término *framing* se refiere a la manera en que Ethernet crea la trama en capa 2 (en el origen) e interpreta los bits recibidos desde otro dispositivo (en el destino). Los datos son transmitidos en el medio a través de señales, las cuales son transformadas en cadenas de bits en capa 1, que a su vez son enviados a capa 2. ¿Cómo interpreta el dispositivo dichas cadenas? De ello se encarga el *framing*. Ethernet define una serie de campos en sus tramas, cada uno de ellos con una longitud determinada y manteniendo siempre el mismo formato. Gracias a ello, al ser recibidos los bits, se contabilizan y analizan los bytes, determinando cuáles pertenecen a un campo y cuáles a otro.

Una trama 802.3 está compuesta por una cabecera y un tráiler. La primera incluye todos los campos agregados antes de los datos provenientes de la capa 3, mientras que el tráiler son aquellos incluidos al final de la trama. El formato es el siguiente:

Preámbulo	SFD	Destino	Origen	Longitud/Tipo	Datos	FCS
7	1	6	6	2	46 – 1500	4

De tal manera que la cabecera está compuesta por los campos preámbulo, SFD, destino, origen y longitud/tipo, mientras que el tráiler lo forma el campo FCS.

DETECCIÓN DE ERRORES

Otra de las funciones desarrolladas en capa 2 consiste en la detección de errores, la cual se encarga de identificar si los bits de una trama han sufrido alguna modificación durante el proceso de envío desde el origen hasta el destino de la comunicación. Este hecho puede suceder por varias razones, pero fundamentalmente debido a interferencias eléctricas.

El proceso consta de los siguientes pasos:

- En el origen, el dispositivo que crea la trama comprueba todos los bits que la forman y con relación a ellos ejecuta una ecuación matemática. El resultado obtenido es incluido en el campo FCS.
- El origen envía la trama al destinatario.
- En el destino, el dispositivo ejecuta la misma acción, es decir, comprueba todos los bits recibidos y aplica sobre ellos la misma ecuación matemática. Si el resultado coincide con el valor recibido en el campo FCS significa que no ha habido errores durante la transmisión, sin embargo, si es diferente se da por hecho que al menos un bit ha sido alterado y por lo tanto se descarta la trama.

Ethernet tan solo provee detección de errores, que no es lo mismo que recuperación de los mismos. Si una trama es descartada, no es reenviada. De esta función se encargan protocolos de capas superiores como TCP.

Wireless LAN

Cada vez resulta más común y necesaria la aplicación de soluciones inalámbricas con el fin de facilitar la movilidad de trabajadores dentro de la compañía y a su vez permitir la conexión de dispositivos carentes de puertos Rj-45, como smartphones o tablets. Su puesta en marcha puede resultar tan sencilla como la instalación de

puntos de accesos (APs) en aquellos lugares donde se considere oportuno, los cuales, por un extremo conectarán con la red cableada de la compañía, mientras que por el otro ofrecerán cobertura inalámbrica a aquellos clientes ubicados en el espacio físico que abarca la misma. Un ejemplo de ello podría ser el siguiente:

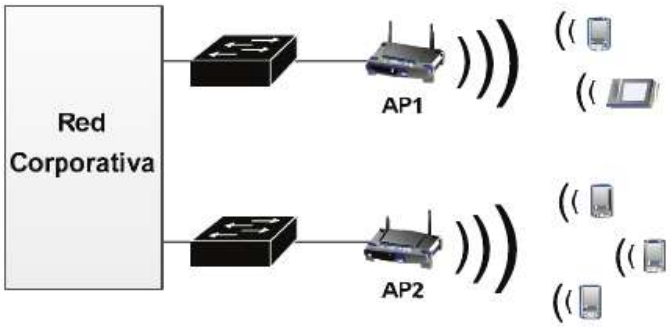


Fig. 1-34 Solución básica Wireless.

Donde AP1 y AP2 permiten la conexión a la red mediante tecnología Wi-Fi. Sin embargo, este modelo sobre entornos corporativos presenta los siguientes inconvenientes:

- El roaming resulta limitado o interrumpido, ya que, si el cliente se desplazara entre dos zonas de cobertura ofrecidas por diferentes APs, la transición entre una y otra requiere su desconexión y conexión nuevamente.
- La administración de los puntos de acceso no se lleva a cabo de manera centralizada, por lo que, ante cualquier cambio o gestión requerida, se deberá proceder en cada uno de ellos de manera independiente.
- La comunicación iniciada por los clientes fluye directamente desde el origen hacia el destino a través de la red corporativa, hecho que puede suponer una brecha de seguridad.

Debido a ello, lo ideal consiste en un tipo de infraestructura que solvente dichas trabas, la cual se hace posible gracias a la aplicación de los siguientes dispositivos:

- *Wireless LAN Controller (WLC)*: Gestiona de manera centralizada todos los APs corporativos y la autenticación de sus clientes. Cualquier configuración o modificación necesaria tan solo debe ser aplicada sobre el mismo y automáticamente será distribuida a lo largo de la infraestructura.
- *Lightweight AP (LWAPP)*: Son los diferentes puntos de acceso, denominados "ligeros" ya que sus características y su modo de operar son definidos y gestionados por el WLC.

De tal manera que la infraestructura WLAN estará compuesta por diferentes APs y un controlador de estos, el cual actuará como intermediario entre los clientes y la red corporativa. En este caso, cualquier comunicación inalámbrica es enviada desde los LWAPP hacia el WLC, y este, a su vez, lo reenvía a la red corporativa. Gracias a ello se hace posible establecer filtros o políticas de seguridad que tan solo afectarán a los clientes Wi-Fi.

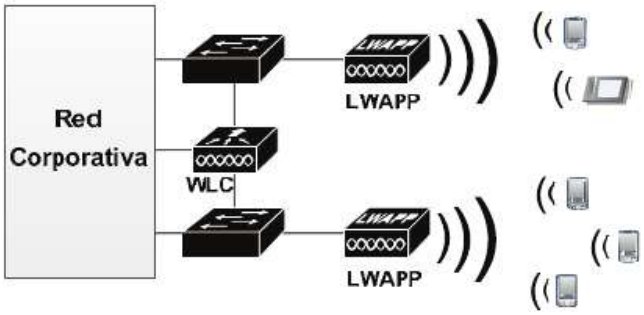


Fig. 1-35 Solución Wireless con gestión avanzada.

El estándar IEEE 802.11 define las especificaciones necesarias de comunicación inalámbrica (Wi-Fi). A lo largo del tiempo se han desarrollado diferentes variantes del mismo (802.11a, 802.11b, 802.11g, etc.) las cuales son contenido de certificaciones dedicadas a tal propósito, como el "CCNA Wireless".

REDES WAN

Analizados los aspectos más básicos de una LAN, dedicaremos la siguiente sección a abordar las características principales de otro tipo de redes, aquellas de área extensa, también denominadas WAN (*Wide Area Network*). Ambas se diferencian principalmente en el alcance geográfico que abarcan y en los protocolos y medios físicos necesarios para la comunicación. Mientras que una LAN conecta PCs, dispositivos intermediarios y diferentes elementos de red en un solo edificio o área geográfica limitada, una WAN permite la comunicación a distancias mucho mayores, siendo necesaria la subscripción a un proveedor de servicios (ISP) para acceder a ellas, el cual brindará al cliente un determinado ancho de banda previamente contratado.

Su modo de operar se basa principalmente en las capas 1 y 2 del modelo OSI. En capa 1 se establecen las tecnologías necesarias para proporcionar conexiones eléctricas, mecánicas, operativas y funcionales a los servicios brindados por el ISP.

Mientras, en capa 2 se encapsulan los datos para su transmisión, haciendo uso de diferentes protocolos como PPP, Frame Relay y HDLC.

Capa 1 en redes WAN punto a punto

Al igual que ocurre en las redes LAN, la capa 1 define el hardware, protocolos y estándares necesarios para transmitir bits desde un origen hacia un destino a través de los medios físicos disponibles. Existen dos tipos de redes WAN, punto a punto y multiacceso, centrándonos en esta sección en las primeras.

Un enlace punto a punto es un circuito dedicado entre dos dispositivos, implementado normalmente para unir varias LAN a través de una WAN. Como se ha mencionado en párrafos anteriores, para establecer dicho enlace es necesario contratar los servicios de un ISP, por lo que prácticamente todos los elementos físicos en capa 1 como cableado o dispositivos intermediarios dependen por completo del proveedor de servicios en cuestión. Desde el punto de vista del administrador de la LAN, contratar este tipo de circuito se conoce como "línea arrendada".

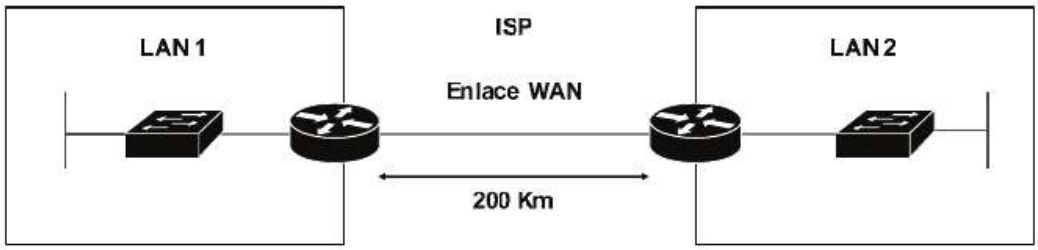


Fig. 1-36 Enlace WAN.

Existe otro tipo de red denominadas MAN *(metropolitan area-network).* Estas son de mayor tamaño que una LAN, pero menor que una WAN, soliendo abarcar como máximo distancias que no superan el área metropolitana.

ELEMENTOS FÍSICOS

Evidentemente, en la conexión punto a punto recién analizada entran en juego diferentes componentes físicos para que la comunicación pueda llevarse a cabo. Normalmente, una red WAN está compuesta por los siguientes elementos.

- **Oficina central (CO):** Es el edificio físico del ISP donde se ubican sus dispositivos de red. En otras palabras, el lugar donde van a parar las comunicaciones de los clientes para luego enrutarlas hasta su destino.
- **Switch WAN:** Los switchs WAN también forman parte del ISP, pero no se encuentran ubicados en la oficina central. Estos son distribuidos a lo largo de

la ciudad, provincia o país. La red LAN conecta con el switch WAN más cercano y este enruta la comunicación hasta la CO.

- **Punto de demarcación:** Es el punto de la LAN que conecta con el cableado de la WAN. Se refiere, por ejemplo, a un cajetín de telefonía o una ONT de fibra. Se considera el límite entre los dispositivos que físicamente forman parte de la WAN y aquellos otros de la LAN.
- **CSU/DSU:** El CSU/DSU es un dispositivo instalado por el ISP dentro de la LAN al cual se conectará el router para acceder a través de él a los servicios de la WAN. Por un extremo conecta con el punto de demarcación y por el otro con el router de la red LAN.
- **Router:** El router de la red de área local que tendrá acceso a la WAN.
- **CPE:** El CPE se refiere a los dispositivos necesarios para conectar con la WAN y que físicamente se encuentran ubicados en la LAN. Por lo descrito hasta ahora, los CPE son el CSU/DSU y el router.

Un ejemplo gráfico de todos ellos:

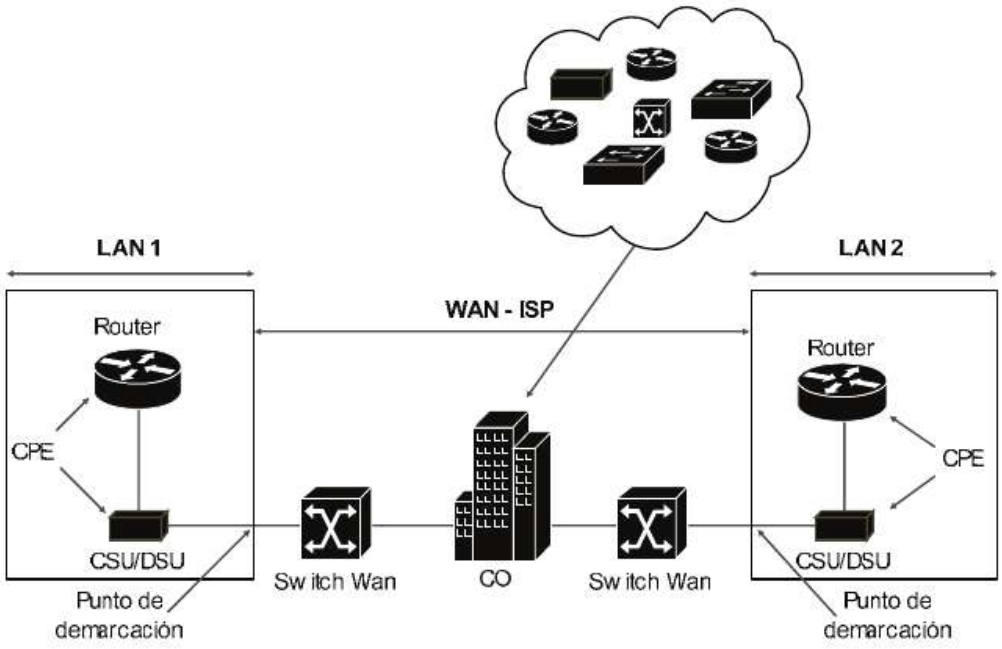


Fig. 1-37 Elementos físicos en una red WAN.

Haciendo uso del siguiente cableado:

- Un cable serial para establecer la conexión entre el router y el CSU/DSU.
- Diferentes tipos de cableado aplicados desde el punto de demarcación hasta el switch WAN. Estos dependen por completo del ISP y su longitud puede ser de kilómetros de distancia.

ESTÁNDARES DE CABLEADO

La conexión entre el router y el CSU/DSU se establece mediante un cable "especial" denominado serial o WAN. Existen diferentes tipos y estándares para este, entre ellos el EIA/TIA-232, EIA/TIA-449, V.35, X.21 y EIA-530. No es necesario profundizar en sus características, pero sí resulta importante saber que todos ellos destinan determinados pines para funciones de control, transferencia de datos y control del reloj.

Los routers Cisco normalmente incluyen interfaces WAN, y si no es así, pueden ser agregadas mediante módulos de expansión. Estas harán uso de un estándar u otro, dependiendo del modelo de dispositivo. Además, muchos también incluyen la funcionalidad de CSU/DSU, en cuyo caso la conexión se realiza directamente con el punto de demarcación y operará internamente como CSU/DSU y router. Por lo tanto, ¿Qué dispositivo de red puede ejercer las funciones de CSU/DSU? ¡El router!

VELOCIDAD DE RELOJ, SINCRONIZACIÓN, DCE Y DTE

La instalación de una línea WAN punto a punto requiere diferentes acciones, donde el cliente contratará el servicio a su ISP, para este proceder a la instalación y configuración del CSU/DSU en ambos extremos del enlace. Estas líneas operan a velocidades predefinidas, lo que se conoce como ancho de banda o velocidad de reloj, la cual debe coincidir en ambos extremos.

Lo mismo ocurre con la conexión entre el CSU/DSU y el router. Ambos deben operar a la misma velocidad, de lo contrario la comunicación no se llevará a cabo. Para lograrlo, cada uno de ellos toma un rol, DCE o DTE, desarrollando la siguiente función.

- El DCE marca la velocidad de reloj a la que operará el enlace, siendo en este caso el CSU/DSU.
- El DTE se limita a aceptar la velocidad marcada por el DCE y aplicarla, siendo en este caso el router el que asume dicho rol.

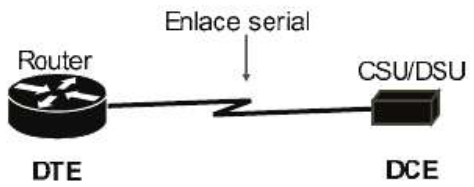


Fig. 1-38 Sincronización de enlace serial.

Entre dos routers Cisco existe la posibilidad de crear un enlace serial punto a punto, actuando uno como DTE y otro como DCE. Esta configuración será aplicada en capítulos posteriores como práctica de laboratorio, pero en entornos reales dicha tarea corresponde al ISP.

Capa 2 en redes WAN punto a punto

Los protocolos normalmente aplicados en capa 2 para conexiones WAN punto a punto son HDLC y PPP, ambos muy similares y desarrollan las mismas funciones.

HDLC (HIGH-LEVEL DATA LINK CONTROL)

Como el modo de operar en los enlaces punto a punto es relativamente sencillo, las funciones de los protocolos de capa 2 para este propósito también lo son. HDLC se encarga principalmente de dos tareas, primero, comprobar que la trama atraviesa el enlace sin errores, y de lo contrario, descartarla, y segundo, identificar el protocolo de capa 3 encapsulado en ella. Para lograrlo se incluyen una serie de campos tanto en la cabecera como en el tráiler, los cuales varían dependiendo de la versión HDLC aplicada, que puede ser la estándar o la propietaria de Cisco. La primera es la versión original del protocolo mientras que la segunda es una modificación creada por Cisco y a su vez propietaria, por lo que solo está disponible en sus routers. La diferencia entre ambas radica en que esta última incluye en la cabecera un campo denominado "Tipo de protocolo", con un tamaño de 2 bytes y utilizado para identificar el protocolo de capa 3 transportado.

Las tramas creadas por ambas versiones constan de los siguientes campos:

HDLC Estándar

Flag 1 Byte	Dirección 1 Byte	Control 1 Byte	Datos Variable	FCS 4 Bytes

HDLC Propietario de Cisco

Flag 1 Byte	Dirección 1 Byte	Control 1 Byte	Tipo Protocolo 2 Bytes	Datos Variable	FCS 4 Bytes

La detección de errores se lleva a cabo de la misma manera que en Ethernet, gracias al campo FCS incluido en el tráiler. Si su valor no coincide tanto en origen como destino, la trama es descartada. HDLC tampoco provee recuperación de errores, de esta tarea se encargan protocolos de capas superiores.

Por último, el campo dirección a menudo no es utilizado, ello se debe a que en enlaces punto a punto el destino siempre va a ser el mismo (el router del otro extremo).

PPP (POINT-TO-POINT PROTOCOL)

PPP es otro protocolo disponible en capa 2 para enlaces punto a punto en redes WAN. Fue creado después de HDLC aunque su modo de operar coincide, es más, la trama creada es idéntica a la propietaria de Cisco de HDLC, al igual que las funciones de cada uno de sus campos.

Entonces, ¿Para qué crear dos protocolos iguales? Simplemente porque la versión HDLC de Cisco solo está disponible en dispositivos de esta compañía, siendo necesario un protocolo estándar con las mismas características pero aplicable en cualquier router. Por ello se desarrolló PPP.

Hoy en día es el protocolo más utilizado para este propósito y será analizado en profundidad en capítulos posteriores.

Servicios de conmutación por paquetes: Frame Relay

Los enlaces punto a punto establecen un circuito físico y permanente entre emisor y receptor, transmitiendo la comunicación entre ambos a través del mismo. Este concepto es denominado conmutación por circuito, siendo la red de telefonía un claro ejemplo de ello. El problema principal en este modelo reside en que un fallo físico a lo largo del enlace supone la pérdida de conexión entre ambos extremos, además, el coste de contratar una línea física y dedicada puede resultar muy elevado.

Otra opción disponible para lograr el mismo fin consiste en la conmutación por paquetes, la cual establece un enlace entre dos extremos, pero ni físico, ni dedicado. Es decir, la comunicación podrá tomar diferentes rutas para llegar a su destino, lo que se conoce como un circuito virtual. Imagina que una compañía dispone de 50 sedes y desea conectarlas mediante enlaces WAN. Si optan por la conmutación por circuito serían necesarias 49 líneas diferentes y 49 CSU/DSU en cada sede, convirtiendo a esta opción en poco viable y nada escalable. Sin embargo, gracias a la conmutación por paquetes conectarlas sería posible mediante una única línea y un único CSU/DSU en cada sede, ya que puede albergar multitud de enlaces virtuales.

Una de las tecnologías disponibles para implementar este modelo es Frame Relay.

CONCEPTOS BÁSICOS DE FRAME RELAY

Frame Relay puede ser definido como una tecnología WAN de conmutación por paquetes que opera en capa 2 y que basa su modelo de comunicación en una red multiacceso gracias a la cual los dispositivos pertenecientes a ella dispondrán de diferentes circuitos virtuales hacia cada uno de los destinos configurados.

Este modo ofrece mejores características y beneficios que los enlaces punto a punto, siendo uno de ellos que la caída de un circuito no supone la pérdida de conectividad entre ambos extremos. Además, no son dedicados, por lo que el mismo medio físico puede ser utilizado por diferentes dispositivos.

Una red Frame Relay consta de los siguientes elementos físicos:

- **Routers:** Establecen la conexión con los switchs FR. Al igual que en los enlaces punto a punto, actúan como DTE.
- **Links de acceso:** Se refiere al enlace serial entre el router y el switch FR. La diferencia con las líneas dedicadas es que un solo link de acceso puede albergar múltiples circuitos virtuales.
- **Switch Frame Relay:** Son switchs ubicados en la WAN, administrados por el ISP y cuya función consiste en recibir las tramas generadas por los routers para reenviarlas hacia su destino, pudiendo hacer uso para ello de diferentes rutas. Los switch FR toman el rol de DCE.

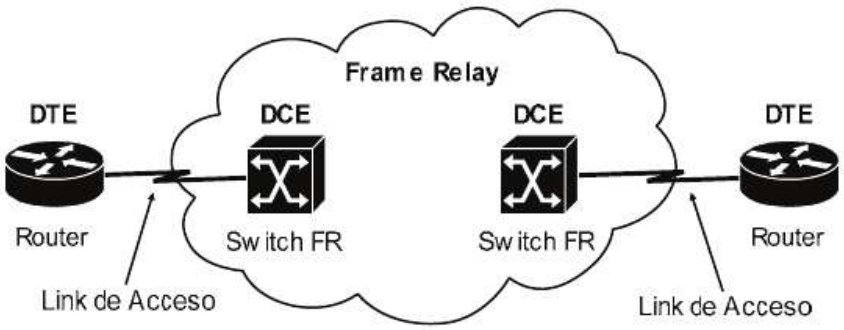


Fig. 1-39 Elementos físicos en una red Frame Relay.

El link de acceso puede comunicar múltiples enlaces virtuales, es por ello que en este caso sí resulta necesario identificar el destino en el campo “Dirección”, ubicado en la cabecera de la trama, de tal manera que cuando esta es recibida por el switch FR, la lee y reenvía a través del enlace virtual adecuado. Sin embargo, esta dirección no corresponde a la MAC como en redes Ethernet, ya que estas hacen referencia a dispositivos físicos, mientras que en Frame Relay lo que se requiere identificar son circuitos. En este caso, es necesario especificar el DLCI (*Data Link Connection*

Identifier), que es un valor decimal asignado a cada circuito virtual. Tanto las MAC como los DLCI son direcciones de capa de enlace de datos, pero utilizadas en diferentes tecnologías.

Un ejemplo de una red Frame Relay podría ser:

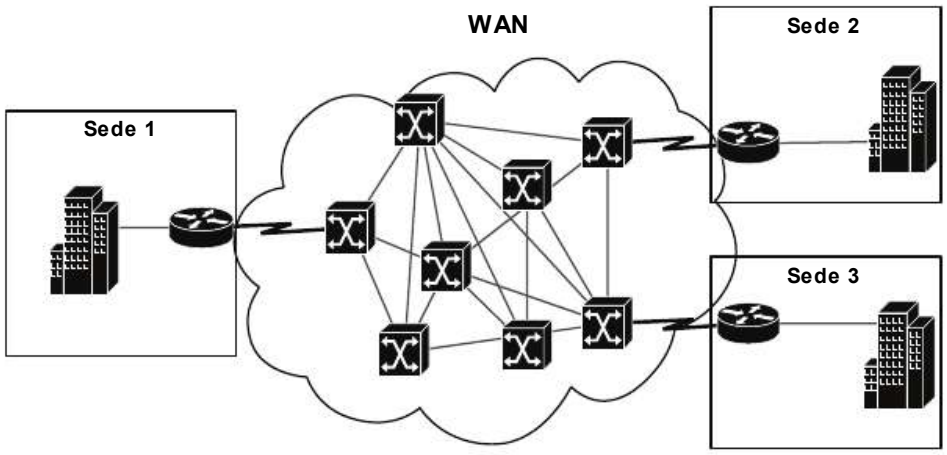


Fig. 1-40 Topología de red Frame Relay.

La comunicación entre las diferentes sedes podrá llevarse a cabo tomando diferentes rutas, es decir, circuitos virtuales. Como los switchs Frame Relay toman la decisión de reenvío de tramas con relación al DLCI son considerados dispositivos de capa 2.

Las características técnicas y su configuración serán objeto de estudio en capítulos posteriores.

ENRUTAMIENTO Y DIRECCIONAMIENTO IP

Las capas 1 y 2 desarrollan las funciones de transmisión de bits e identificación de dispositivos físicos, resultando esta última tarea esencial para que la comunicación entre hosts pertenecientes a la misma red pueda llevarse a cabo. Sin embargo, ¿qué ocurre si el origen y destino se encuentran ubicados en diferentes redes? Tanto OSI como TCP/IP necesitan mecanismos capaces de realizar dicha comunicación, siendo esta la función a desarrollar en capa 3. Para lograr su objetivo, entran en juego los conceptos definidos a continuación:

- **Enrutamiento:** Es el proceso basado en el reenvío y entrega de paquetes desde el origen hasta el destino. El dispositivo encargado de ello es el router.

- **Protocolos de enrutamiento:** son protocolos aplicados en dispositivos de capa 3 con la finalidad de aprender de manera automática direcciones de redes remotas y la ruta necesaria para acceder a ellas.
- **Direccionamiento IP:** hace referencia a la dirección de capa 3 asignada a cada host, la cual resulta imprescindible para que el enrutamiento pueda llevarse a cabo. Estas direcciones son lógicas, ya que no identifican de manera única y global a cada dispositivo, a diferencia de las MAC en capa 2.
- **Utilidades de capa 3:** protocolos pertenecientes a esta capa, como DNS, DHCP, ping y ARP.

Enrutamiento

El enrutamiento es el proceso mediante el cual un paquete es reenviado entre dispositivos de red desde un origen hasta un destino, siendo los routers y el protocolo IP los máximos responsables de llevarlo a cabo. Dicho proceso se basa por completo en las direcciones de capa 3 tanto del emisor como del receptor, gracias a las cuales se tomará la decisión de reenvío más adecuada para que la comunicación concluya con éxito. Un detalle a tener en cuenta es que el protocolo IP no es orientado a conexión, hecho por el cual no ofrece recuperación de errores, si un paquete no llega a su destino, simplemente es descartado.

Un ejemplo de comunicación entre diferentes redes y cómo procede cada dispositivo podría ser el siguiente:

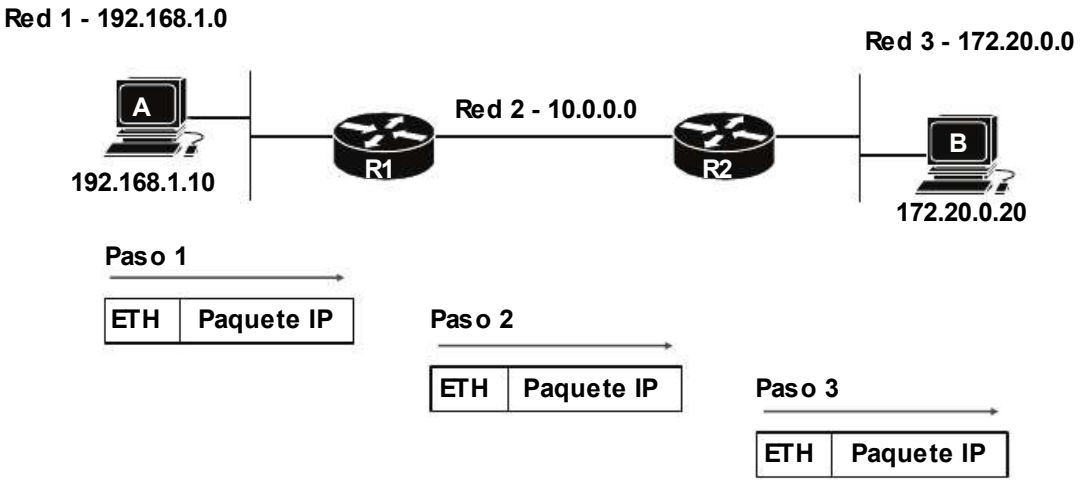


Fig. 1-41 Proceso de comunicación en capa 3.

Paso 1: PC A, para comunicarse con B, crea un paquete IP con dirección de origen 192.168.1.10 y destino 172.20.0.20. Como ambas pertenecen a diferentes redes envía la comunicación al router de su propia red, el cual actúa como puerta de

enlace, en este caso R1. Para ello, hace uso de su dirección física (capa 2), incluyéndola en el campo "destino" de la trama ETH. Por lo tanto, PC A hace uso de dos direcciones de destino, una en capa 2 que corresponde a la MAC de R1 y otra en capa 3 que identifica la IP de B. El paquete es enviado a la red y recibido por R1.

Paso 2: R1 desencapsula el paquete recibido, lee la dirección IP de destino y comprueba que no está conectado directamente a dicha red, por lo tanto busca en su tabla de rutas la interfaz por la cual debe ser reenviado, siendo en este caso aquella que conecta con R2. Tras ello, ejecuta el mismo procedimiento que PC1, definiendo dos direcciones de destino, una en capa 2 que corresponde a la MAC de R2 y otra en capa 3 que identifica la IP de PC B.

Paso 3: R2 recibe el paquete, lo desencapsula, lee la dirección IP de destino y comprueba que forma parte de su propia LAN, por lo tanto simplemente lo reenvía por la interfaz adecuada para que sea recibido por PC B.

La interacción entre capa 2 y capa 3 resulta fundamental para que la comunicación concluya con éxito. En capa 3, la dirección de destino no varía a lo largo del enrutamiento, sin embargo, en capa 2 sí que es modificada con el fin de que los datos sean recibidos por el dispositivo correcto en cada momento. Por ejemplo, el enlace entre R1 y R2 pertenece a la red 10.0.0.0; sin embargo, durante el proceso de enrutamiento nunca se ha utilizado una IP perteneciente a dicho rango. ¿Por qué? Porque la comunicación entre dispositivos intermediarios se lleva a cabo en capa 2, en este caso a través de la dirección MAC.

¿Cómo conoce R1 la MAC de R2? Para averiguarla ejecuta el protocolo ARP, el cual será objeto de estudio en este mismo capítulo y cuya misión consiste en, dada una dirección IP, averiguar la MAC del dispositivo que hace uso de ella.

LÓGICA DE ENRUTAMIENTO

Analizando el proceso de reenvío desde PC A hasta B se podría concluir que la lógica de enrutamiento llevada a cabo tanto por hosts como por routers varía significativamente. El método aplicado por cada uno de ellos consiste en:

Hosts

Cuando un host desea enviar un paquete, comprueba la IP de destino y aplica la siguiente lógica:

- Si forma parte de su misma red, lo envía directamente al destinatario sin necesidad de ser enrutado.

- Si no forma parte de su misma red, envía el paquete a su puerta de enlace.

Routers

Cuando un router recibe un paquete que debe ser reenviado ejecuta las siguientes acciones:

- Comprueba el campo FCS en capa 2 para verificar que no existieron errores durante la transmisión. Si los hay, descarta la trama, de lo contrario, continúa con el enrutamiento.
- Elimina la trama.
- Lee la dirección IP de destino y comprueba si dispone de alguna ruta para acceder a su red. Si existe, continúa con el proceso de enrutamiento, de lo contrario, descarta el paquete.
- Por último, crea una nueva trama en capa 2, incluyendo las direcciones MAC de origen y destino correctas. Tras ello, reenvía la comunicación a través de la interfaz adecuada.

PAQUETES Y CABECERA IP

El proceso de encapsulación en capa 3 consiste en agregar una cabecera al segmento recibido desde la capa de transporte, con el objeto de incluir toda la información de red necesaria para que los datos puedan ser transportados desde el origen hasta el destino. Su tamaño es de 20 bytes e incluye los siguientes campos:

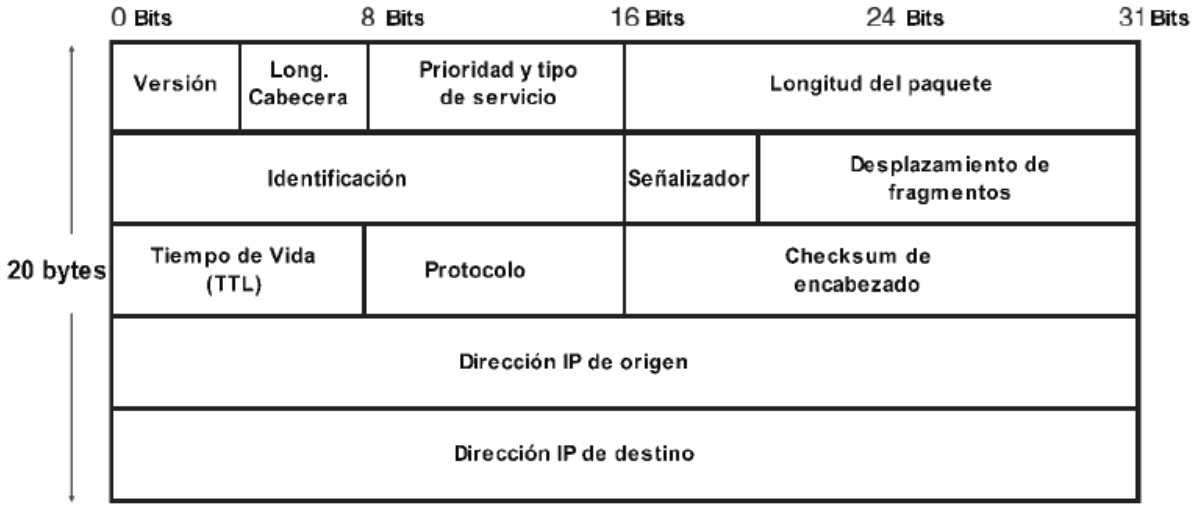


Fig. 1-42 Formato de cabecera de un paquete IP.

De los cuales, dirección de origen, destino y tiempo de vida (TTL) son objeto de estudio en CCNA, desarrollando las siguientes funciones:

- Dirección IP de origen: Como su nombre indica, identifica la IP del dispositivo que inicia la comunicación.
- Dirección IP de destino: Hace referencia a la IP del dispositivo de destino.
- Tiempo de vida (TTL): Es un valor decimal que limita el número de saltos que puede dar un paquete desde el origen hasta el destino, restándole 1 cada vez que este atraviesa un router, de tal manera que cuando su valor es igual a 0, el paquete es descartado. La función principal del tiempo de vida consiste en prevenir loops de enrutamiento.

Protocolos de enrutamiento

El enrutamiento depende por completo de la tabla de rutas almacenada en cada router. Esta puede entenderse como una base de datos que contiene toda la información referente a las redes remotas aprendidas por el dispositivo, asociándolas con la interfaz necesaria para acceder a cada una de ellas.

En el ejemplo analizado en párrafos anteriores, R1 reenvía el paquete a R2 para que este llegue a su destino, pero ¿cómo sabe que a la red 172.20.0.0 se accede a través de dicho router? Esta información puede ser agregada de dos maneras, manual y automática, siendo esta última opción la más aplicada y recomendable. Con este propósito nacen los protocolos de enrutamiento, que son aquellos cuya misión consiste en incluir información en la tabla de rutas automáticamente gracias al intercambio de mensajes entre los diferentes routers ubicados en la red.

Aplicado a la topología anterior, R1 notifica a R2 que dispone de la red 192.168.1.0. Este último la agrega a su tabla de rutas asociándola con la interfaz que conecta con R1. De esta manera, todos los paquetes cuya red de destino sea la 192.168.1.0 serán reenviados a R1. R2 ejecuta el mismo procedimiento, informa a R1 que dispone de la red 172.20.0.0 para que este la agregue a su tabla.

Realmente, en una topología tan pequeña las rutas pueden ser agregadas manualmente, pero imaginemos una red mayor, con 50 routers y 100 subredes, resultaría prácticamente imposible establecer la configuración de manera manual. Es más sencillo, eficaz y escalable aplicar un protocolo de enrutamiento dinámico que ejecute esta tarea de manera automática.

El proceso, en detalle, consta de:

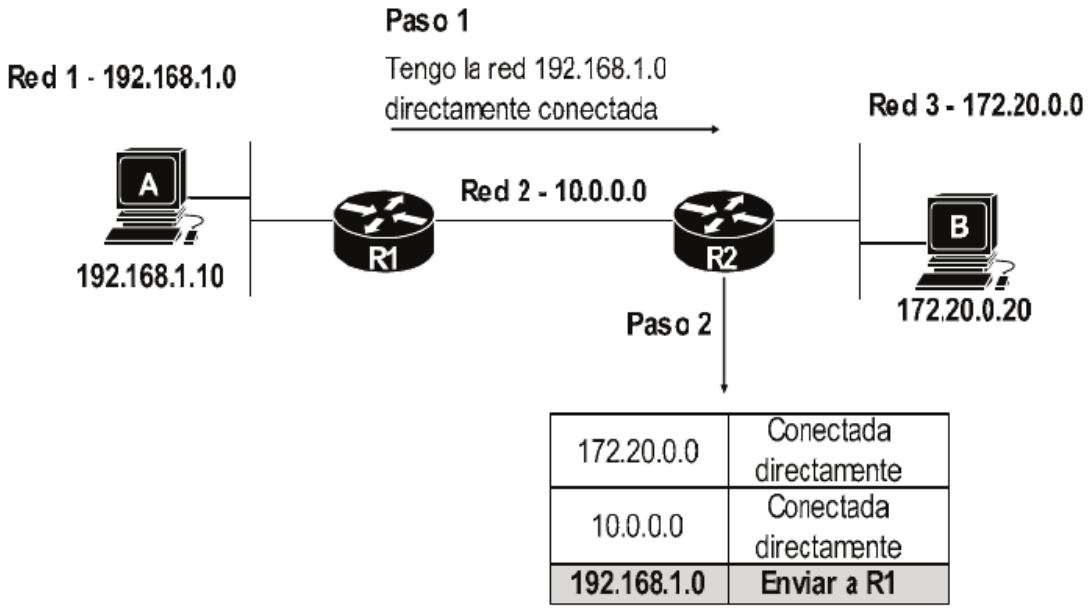


Fig. 1-43.1 Intercambio de rutas en protocolos de enrutamiento.

Paso 1: R1 notifica a R2 sus redes directamente conectadas. En este caso la 192.168.1.0.

Paso 2: R2 recibe la notificación, comprueba que no dispone de dicha red almacenada en su tabla de rutas y la agrega, marcando a R1 como siguiente salto hacia ella. A partir de ahora, cualquier paquete recibido por R2 con destino 192.168.1.0 será reenviado a R1.

Realmente, R1 también notifica la red 10.0.0.0, pero como R2 ya la tiene almacenada en su tabla de rutas como directamente conectada no es agregada nuevamente. El ejemplo tan solo se centra en la red 192.168.1.0 con el fin de facilitar la comprensión.

R2 ejecuta el mismo procedimiento, notificar a R1 de sus redes directamente conectadas.

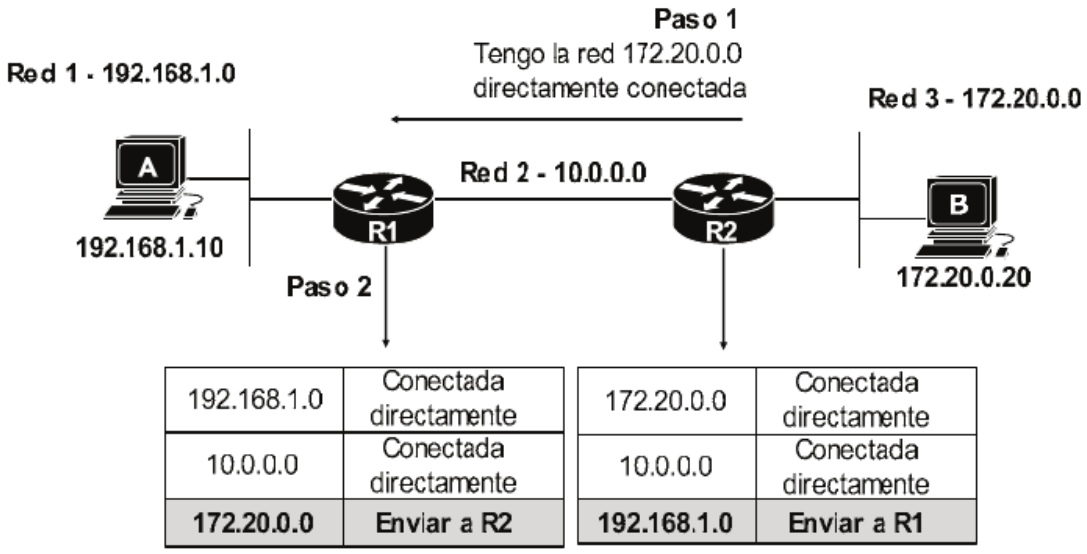


Fig. 1-43.2 Intercambio de rutas en protocolos de enrutamiento.

¿Qué ocurre si PC A envía un paquete a R1 con dirección de destino 172.60.0.10? Este comprueba en su tabla de rutas que no dispone de ninguna hacia dicha red, por lo que simplemente descarta el paquete.

Los protocolos de enrutamiento tienen como finalidad:

- Aprender y completar de manera dinámica y automática la tabla de rutas.
- Si existe más de una ruta hacia la misma red, agregar la mejor de ellas.
- Cuando una ruta deja de ser válida, ya sea por fallo de algún router, enlace, o porque simplemente es eliminada, notificar a los demás para que no hagan uso de la misma.
- Si existen varias rutas hacia la misma red y una de ellas cae, eliminarla e incluir otra operativa.
- Prevenir loops de enrutamiento.

Para lograrlo, basan su modo de operar en la siguiente secuencia:

- *Paso 1:* Cada router agrega a su tabla las redes directamente conectadas.
- *Paso 2:* Cada router envía a sus vecinos todas las rutas existentes en su tabla, tanto las directamente conectadas como aquellas aprendidas desde otros vecinos.
- *Paso 3:* Cuando se instala una nueva ruta, se establece como siguiente salto el router desde el cual fue recibida.

Direccionamiento IP

El direccionamiento en capa 3 consiste en la identificación de cada dispositivo y red a la que pertenecen para que la comunicación entre dos extremos pueda llevarse a cabo con éxito. Ello es posible gracias a las direcciones IP, las cuales son consideradas lógicas, en contraste con las MAC, que son físicas. Ambas se diferencian en los siguientes aspectos:

- Una dirección física, como su nombre indica, identifica de manera única y global a un solo dispositivo, gracias a su MAC, almacenada en la tarjeta de red. Por ejemplo, un PC ubicado en una red de España es movido a otra físicamente en Francia. Bien, este mantendrá la misma MAC en ambos países. Además, si es desconectado, su dirección no podrá volver a ser utilizada. Sin embargo, una dirección lógica puede variar en un mismo dispositivo físico. Por ejemplo el mismo PC, cuando es instalado en la nueva red de Francia es altamente probable que se le asigne una IP diferente a la utilizada en España. Además, cuando sea desconectado, su dirección queda disponible y puede volver a ser asignada.
- Además, el direccionamiento lógico permite agrupar dispositivos en relación con la IP utilizada, lo cual permite la creación de subredes. Esta característica en las direcciones físicas no existe.

IP es el protocolo mayormente utilizado en capa 3, y entre sus variantes, IPv4. El formato de sus direcciones consta de 32 bits, separados en 4 partes de 8 bits cada una y representadas en formato decimal.

Por ejemplo, la IP 11000000.10101000.00001010.00000001 es expresada en números decimales como 192.168.10.1.

Este cálculo se realiza de la siguiente manera. A cada bit, en cada octeto de la dirección le corresponde un valor decimal, siendo los siguientes:

128	64	32	16	8	4	2	1
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**

De tal manera que la conversión se lleva a cabo sumando los valores equivalentes de aquellos bits establecidos a 1. Ello, aplicado a la dirección anterior:

Primer octeto: **11**000000 = 128 + 64 = **192**

Segundo octeto: **1**0**1**0**1**000 = 128+32+8 = **168**
Tercer octeto: 0000**1**0**1**0 = 2+8 = **10**
Cuarto octeto: 0000000**1** = **1**

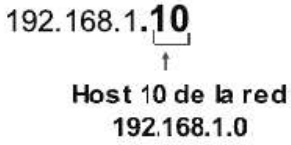
CÓMO AGRUPAR HOSTS EN RELACIÓN CON LA DIRECCIÓN IP

En ámbito informático, el término "red" hace referencia a un grupo de dispositivos conectados y agrupados entre sí gracias al protocolo IP, el cual, para lograr dicha agrupación, divide la dirección en dos, parte de red y parte de host. La primera de ellas, como su nombre indica, identifica la red de la cual forma parte el dispositivo, mientras que la segunda hace referencia a la dirección utilizada por el host en dicha red, la cual debe ser exclusiva.

Para analizarlo de modo práctico, volvamos a la topología de la *"Fig. 1-43.2 Intercambio de rutas en protocolos de enrutamiento"*, donde PC A forma parte de la red 192.168.1.0 y hace uso de la IP 192.168.1.10. En relación con dichos datos, se puede deducir la siguiente información.

- La dirección 192.168.1.0 identifica a la red, lo cual significa que todos los dispositivos cuya IP comience por 192.168.1.x formarán parte de la misma, haciendo uso del último octeto para identificar a cada host.

- La IP utilizada por PC A no debe ser asignada a ningún otro dispositivo de la misma red, ya que lo identifica de manera exclusiva en la misma.



Conociendo estos datos, ¿cuántos dispositivos pueden formar parte de la misma red? Para calcularlo se debe aplicar la fórmula 2^n-2, donde *n* es el número de bits destinados para hosts. Al resultado se le resta dos porque una dirección corresponde a la red y otra al broadcast, siendo ambas no asignables sobre dispositivos.

Por ejemplo, en la red 192.168.1.0, la parte de hosts corresponde al último octeto, es decir, los últimos 8 bits. Aplicando la fórmula…

2^8- 2 = 256 – 2 = 254. La red 192.168.1.0 permitirá como máximo 254 dispositivos.

Las dos direcciones que deben ser restadas son aquellas cuyos valores en binario de la parte de hosts equivalen todo a 0 o todo a 1. Para la red 192.168.1.0:

- Convertir la parte de host a binario con todos sus valores a 0, por lo tanto, 192.168.1.**00000000.** Todos los bits a 0 identifican a la red, en este caso la 192.168.1.0. Esta dirección no puede ser asignada a ningún dispositivo, por ello debe ser restada.
- Convertir la parte de host a binario con todos sus valores a 1, por lo tanto, 192.168.1.**11111111**. Identifica la dirección broadcast de la red. Esta es utilizada cuando la comunicación debe ser recibida por todos los miembros de la misma. En este caso corresponde a la 192.168.1.255 y tampoco puede ser asignada a ningún dispositivo, es por ello que también debe ser restada.

Pero ¿Qué sucedería si la red estuviera compuesta por 600 dispositivos? Evidentemente, en la 192.168.1.0 no se podría albergar a todos ellos.

Para solucionarlo se crearon diferentes rangos divididos en cinco clases, A, B, C, D y E, donde cada uno de ellos proporciona un tamaño definido por el número de bits destinados a la parte de red y de hosts. Las clases D y E quedan reservadas para propósitos especiales y no pueden ser utilizadas para la creación de redes privadas, mientras que las A, B y C sí que lo permiten. El tamaño de cada una de ellas es el siguiente:

Red Clase	**Bytes (bits) para la porción de red**	**Bytes (bits) para la porción de hosts**	**Número de hosts permitido por red**
A	1 (8)	3 (24)	2^{24}-2
B	2 (16)	2 (16)	2^{16}-2
C	3 (24)	1 (8)	2^8-2

De las cuales, la clase A es aquella que permite un mayor número de hosts por red, mientras que la C la que menos. Durante el diseño de una topología se debe calcular primero el total de dispositivos que la formarán para luego decidir qué clase aplicar, por ejemplo, si la red está compuesta por 10 PCs lo más lógico sería hacer uso de una clase C.

Además, para cada una de ellas se define un rango de direcciones, establecido por ICANN *(Internet Corporation for Assigned Network Numbers)*, siendo los siguientes:

Clase	Rango de Red	Bytes para red	Bytes para hosts	Total de redes disponibles	Total de Hosts por cada red
A	1.0.0.0 - 126.0.0.0	1	3	27-2 = 126	224-2 = 16.777.214
B	128.0.0.0 - 191.255.0.0	2	2	214 = 16384	216-2 = 65534
C	192.0.0.0 - 223.255.255. 0	3	1	221 =2.097.152	28-2 = 254

La tabla también muestra el total de redes disponibles para cada clase, el cual es calculado aplicando la fórmula $\mathbf{2^n}$, donde *n* corresponde al número de bits de la parte de red (7 porque en cada clase se reserva un bit para definir la misma). Además, en la clase A se restan 2 debido a que existen direcciones reservadas, la 0.0.0.0 y la 127.0.0.0.

SUBREDES

Crear subredes consiste en dividir una red de clase definida (A, B o C) en segmentos más pequeños con el fin de aprovechar el espacio de direcciones lo máximo posible, logrando, además, mayor facilidad de administración, seguridad y mejoras de enrutamiento.

Durante el diseño de una topología se puede optar por diferentes modelos, por ejemplo, imagina que una compañía dispone de 3 departamentos con 20 dispositivos en cada uno de ellos y deben pertenecer a redes diferentes. Podría plantearse de las siguientes maneras:

- Opción A. Definir 3 redes de clase C, que permiten 254 equipos en cada una de ellas.

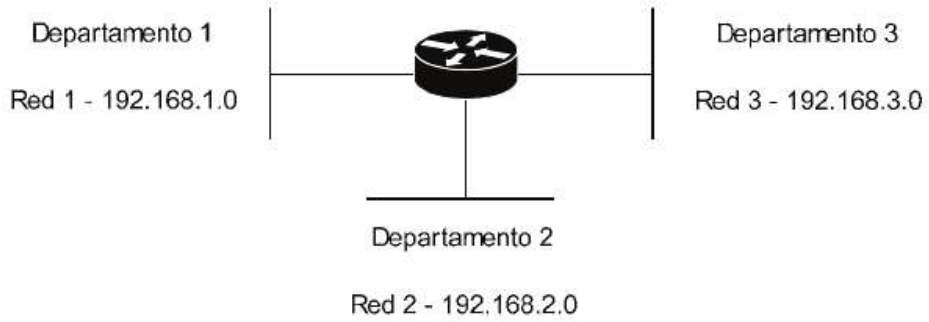


Fig. 1-44 Diseño de red sin aplicar subredes.

- Opción B. Seleccionar una red de clase C y crear 3 subredes. Por ejemplo, a partir de la 192.168.1.0

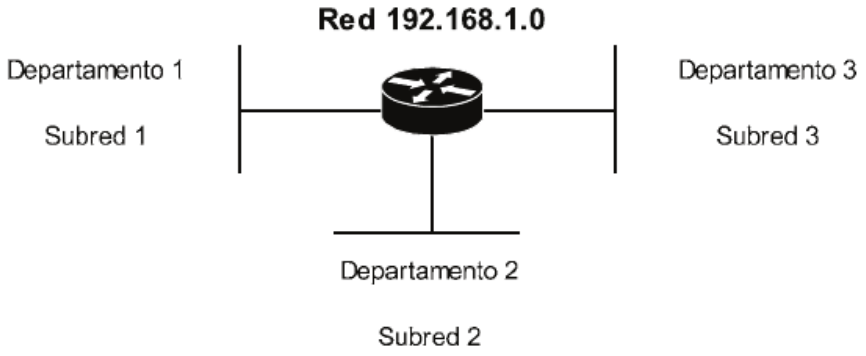
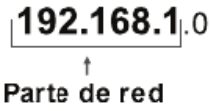


Fig. 1-45 Diseño de red basado en subredes.

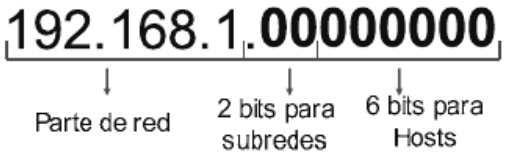
¿Qué diferencias existen entre los dos modelos? Con ambos modelos se logra el objetivo de separar los departamentos en diferentes redes, sin embargo, en cuanto a administración, enrutamiento y sobre todo aprovechamiento de direcciones la mejor opción corresponde a la B.

El proceso para crearlas se basa en, dada una dirección de red, tomar bits de la parte de hosts y utilizarlos como parte de red. De tal manera que:

- 192.168.1.0 es una dirección de clase C, por lo tanto, 3 bytes (24 bits) forman la parte de red y 1 byte (8 bits) la parte de host…



- De los 8 bits destinados para hosts, tomar los necesarios para la creación de las diferentes subredes. En este caso se hará uso de 2…



- Con dos bits destinados para subredes se podrán crear un total de 4 (2^2), donde cada una de ellas podrá albergar 62 hosts (2^6 - 2). Los 3 primeros bytes (192.168.1) nunca se modifican.

El cálculo de subredes será objeto de estudio en el capítulo 4, por lo pronto, tan solo basta conocer qué es una subred y su finalidad principal (aprovechamiento de direcciones, administración y seguridad).

DIRECCIONES IP UNICAST RESERVADAS

Existen multitud de direcciones IP reservadas para determinados propósitos, las cuales no pueden ser asignadas sobre dispositivos ya que con ellas no se podrá enrutar tráfico. Las más comunes son:

- 127.0.0.1: También denominada *loopback address.* Es una dirección de testeo local del protocolo TCP/IP. Hacer ping a ella y obtener respuesta significa que TCP/IP está instalado correctamente en el dispositivo en el cual se ejecuta. Si por el contrario no responde se deberá reinstalar el protocolo o sustituir la tarjeta de red. También recibe el nombre de *localhost.*
- 169.254.x.x: Estas direcciones corresponden a un método de autoconfiguración utilizado en sistemas Windows y conocido como APIPA *(Automatic Private Internet Protocol Addressing)*. Realmente es un rango que abarca desde la IP 169.254.0.1 hasta la 169.254.255.254, siendo asignadas cuando el sistema operativo detecta la conexión a alguna red (ya sea cableada o wifi) pero no dispone datos válidos para acceder a ella, ya que no han sido configurados manualmente ni obtenidos mediante DHCP. Aun así, el dispositivo no tendrá acceso a la red real.
- 0.0.0.0: Es una dirección reservada para ciertos broadcast, como el utilizado durante el proceso de obtención de IP mediante DHCP.

Las direcciones multicast y broadcast de capa 3 serán analizadas a lo largo de los próximos capítulos.

Utilidades de capa 3

Los servicios más comunes y útiles basados en capa 3 son:

- ARP (Address Resolution Protocol).
- DNS (Domain Name System).
- DHCP (Dynamic Host Configuration Protocol).
- PING.

ARP Y DNS

A lo largo de párrafos anteriores se ha analizado el proceso llevado a cabo para que la comunicación entre un origen y un destino concluya con éxito, resultando imprescindible para ello las direcciones IP y MAC. Estos datos, en una red de tamaño medio o grande, serían imposibles de memorizar por los seres humanos, siendo mucho más sencillo conocer tan solo el nombre del dispositivo. Un claro ejemplo de ello es Internet, donde resulta más fácil memorizar una dirección web que la IP del servidor que la almacena.

Con el objeto de poder iniciar una comunicación basándose en un nombre de host de destino nacen los protocolos DNS *(Domain Name System)* y ARP *(Address Resolution Protocol)*, ambos muy sencillos y a la vez imprescindibles en todo tipo de redes.

DNS es el encargado de resolver automáticamente la dirección IP de un dispositivo en relación con su nombre. Para ello es necesaria la instalación de un servidor dedicado a ello en la red, que responderá a las solicitudes de los dispositivos. El proceso es el siguiente, imaginemos que PC A desea comunicarse con C pero no dispone ni de su IP ni de su MAC, tan solo su nombre…

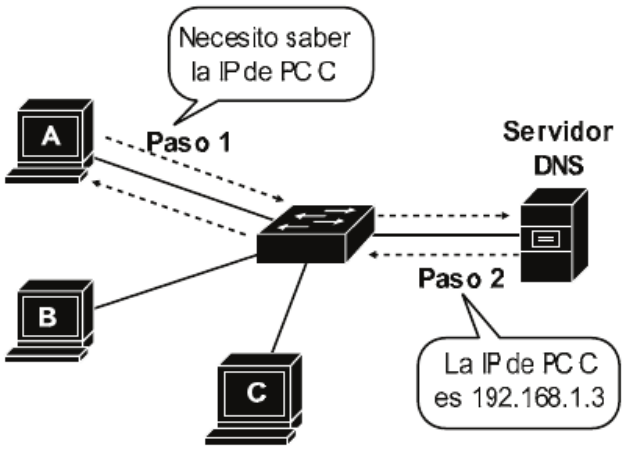


Fig. 1-46 Solicitud DNS.

Paso 1: PC A envía directamente una petición al servidor DNS solicitando la IP del host cuyo nombre es “PC C”.

Paso 2: Este responde facilitando dicho dato, en este caso, 192.168.1.3.

Sin embargo, PC A aún no dispone de la dirección física de C para poder crear la trama en capa 2. Para este propósito ejecuta otro protocolo, ARP, cuya función consiste en resolver la MAC de un dispositivo con relación a su IP.

Para lograrlo, envía un paquete *ARP broadcast* a la red, el cual será recibido por todos sus componentes pero al que tan solo contestará aquel cuya IP sea la indicada en dicho paquete.

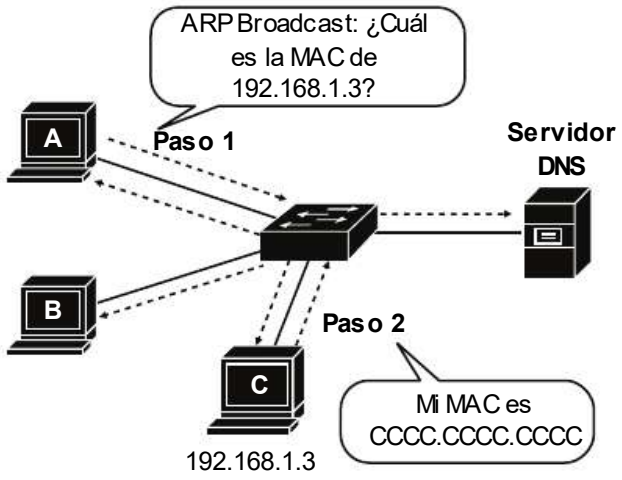


Fig. 1-47 Solicitud ARP.

Paso 1: PC A envía un paquete *ARP Broadcast* a la red solicitando la MAC del host con IP 192.168.1.3.

Paso 2: Al mismo tan solo responde C, ya que es el propietario de dicha dirección, incluyendo en la respuesta su MAC. El resto de dispositivos simplemente ignoran el paquete.

Con ambos datos, PC A ya puede crear el paquete en capa 3 y la trama en capa 2.

Las peticiones ARP de las cuales se ha obtenido respuesta son almacenadas en una memoria denominada *ARP caché*. Esta vincula una relación de IPs con sus correspondientes MACs y tiene como objetivo que el dispositivo no tenga que realizar consultas que ya ha ejecutado con anterioridad.

En sistemas Microsoft Windows, esta memoria puede ser consultada desde *cmd* con el comando *"arp -a".*

DHCP (DYNAMIC HOST CONFIGURATION PROTOCOL)

Cada dispositivo que forme parte de una red requiere una dirección IP para poder comunicarse en ella, la cual puede ser obtenida de manera estática o dinámica. Las primeras son configuradas manualmente en el host y suelen ser aplicadas en elementos críticos como routers, puntos de acceso, switchs y servidores, así como en elementos menos prioritarios como impresoras de red.

Por el contrario, en la configuración dinámica el host recibe automáticamente sus datos de conexión a través de un servidor, lo que incluye dirección IP, máscara, puerta de enlace y servidores DNS. El protocolo encargado de ello es DHCP.

El proceso, en detalle, y su configuración en routers Cisco serán objeto de estudio en el capítulo 5 "*Instalación y configuración inicial de routers Cisco".*

PING

Ping es una utilidad de capa 3 desarrollada con la finalidad de realizar testeos de conectividad de extremo a extremo. Su modo de operar se basa en el protocolo *ICMP (Internet Control Message Protocol)*, donde el origen envía un paquete *ICMP Echo Request* al destino, y este, si está operativo, responderá con un *ICMP Echo Reply*.

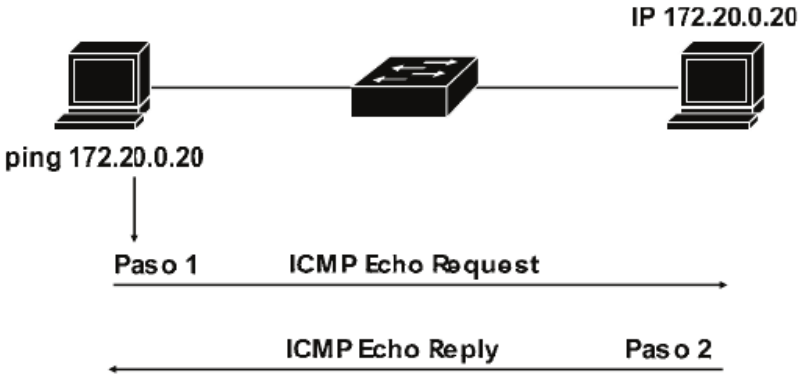


Fig. 1-48 Operación del comando PING.

El comando para ejecutarlo en sistemas Microsoft Windows es *"ping [IP destino]".*

Es posible que algunos dispositivos de red críticos, como servidores o routers, aun estando operativos no respondan a solicitudes de ping. Son configurados así como medida de seguridad para evitar determinados ataques de red.

PROTOCOLOS TCP Y UDP

Las funciones a desarrollar en las capas 2 y 3 consisten principalmente en la identificación de dispositivos, físicos en enlace de datos, y lógicos en red. Sin embargo, la comunicación también debe llevarse a cabo entre las aplicaciones intervinientes en ambos extremos. De ello se encarga la capa 4 gracias a los protocolos TCP y UDP, siendo ambos objeto de estudio en la presente sección.

TCP (Transmission Control Protocol)

El objetivo principal de TCP consiste en la entrega fiable de datos de extremo a extremo, proporcionando para ello detección y recuperación de errores, de tal manera que si algún segmento no es recibido por el destinatario, será reenviado por el origen. Para lograrlo, define su modo de operar en relación con:

- Utilización de puertos.
- Multiplexación.
- Recuperación de errores.
- Control de flujo.
- Reensamblaje.

UTILIZACIÓN DE PUERTOS

La comunicación entre aplicaciones en capa 4 se basa por completo en el uso de puertos, los cuales hacen referencia a valores numéricos utilizados para identificar cada una de las aplicaciones ejecutadas en un sistema, siendo imprescindibles para que la transmisión de datos entre emisor y receptor concluya con éxito. Imagina que un servidor ofrece servicios web y FTP de manera simultánea y recibe 3 solicitudes de clientes. ¿Cómo sabe el sistema a que aplicación va dirigida cada una de ellas? Gracias a la capa 4 y más concretamente al número de puerto incluido en el segmento. Por ejemplo:

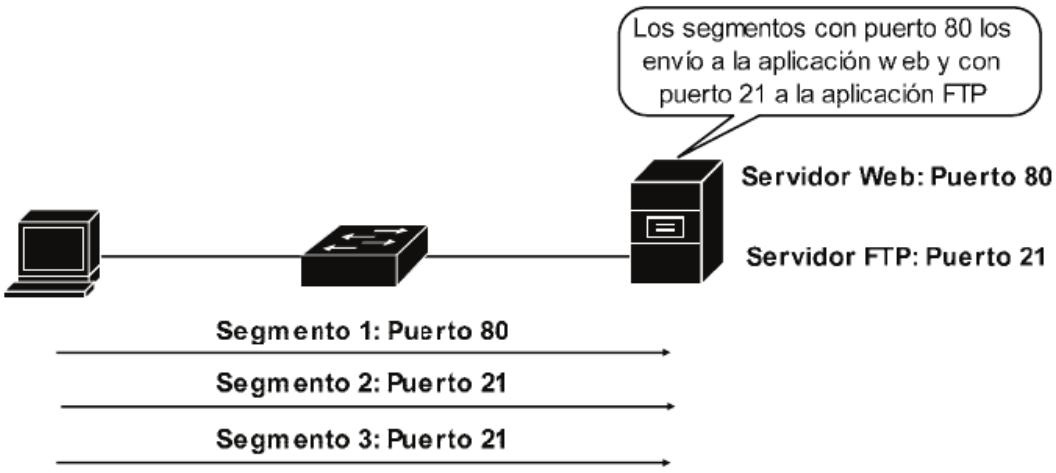


Fig. 1-49 Comunicación mediante puertos.

Los puertos pueden ser comparados con las direcciones en capa 2 y 3. Estas comunican dispositivos, identificándolos mediante direcciones MAC e IP, mientras que la capa 4 comunica aplicaciones, identificándolas mediante un número de puerto. Para lograrlo, durante la encapsulación se incluyen los campos "puerto de origen" y "puerto de destino" (entre otros) en el segmento.

Algunos protocolos que hacen uso de TCP son:

Aplicación	**Puerto**	**Protocolo**
FTP Data	20	TCP
FTP Control	21	TCP
SSH	22	TCP
Telnet	23	TCP
SMTP	25	TCP
DNS	53	TCP (también UDP)
HTTP	80	TCP
POP3	110	TCP
SSL	443	TCP

Los puertos mantienen una estrecha relación con la dirección IP. Al conjunto de ambas se conoce como ***socket*** y es representado mediante el formato "*IP:Puerto*"**.** Un ejemplo podría ser 192.168.1.20:80, donde los datos irán dirigidos al dispositivo 192.168.1.20 y su aplicación que maneja el puerto 80.

MULTIPLEXACIÓN

La multiplexación consiste en la utilización del mismo canal de información (medio físico) de manera simultánea por diferentes dispositivos. Ello es posible gracias a la segmentación llevada a cabo en la capa de transporte, la cual divide los datos provenientes de capas superiores en porciones de menor tamaño. Imagina el envío de un fichero de 5Gb a través de la red. Si fuera transferido sin segmentar, tanto el ancho de banda como la disponibilidad de los enlaces intervinientes quedaría totalmente reservada para el dispositivo que realiza el envío. A ello hay que sumarle que si se produjera cualquier error se tendría que reiniciar la transferencia. Una red de estas características resultaría poco o nada productiva.

Para evitar dicho problema la capa de transporte segmenta los datos, lo cual, a su vez, permite llevar a cabo la técnica de multiplexación, obteniendo beneficios como:

- Mejor aprovechamiento del ancho de banda.
- Transmisión de datos de manera simultánea por diferentes aplicaciones.
- En caso de error, tan solo se tendría que reenviar el segmento corrupto.

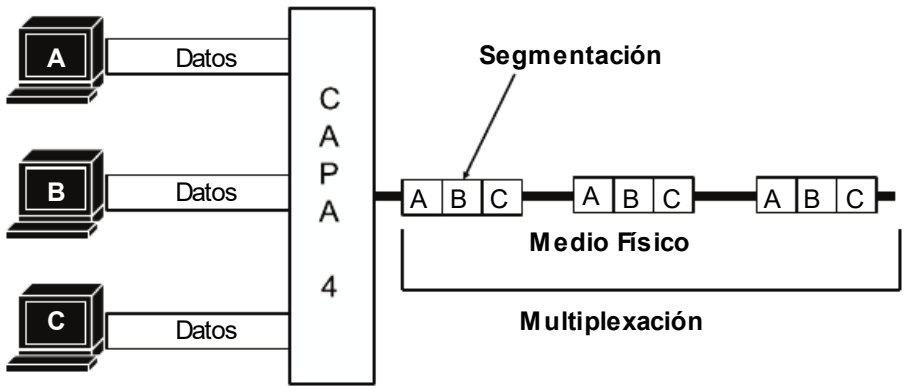


Fig. 1-50 Multiplexación.

RECUPERACIÓN DE ERRORES

La recuperación de errores, también denominada fiabilidad, se refiere al mecanismo mediante el cual los segmentos que no han sido recibidos por el destino son reenviados por el origen. Para lograrlo, TCP aplica la siguiente técnica:

- Emisor y receptor pactan un tamaño de segmento y un número de secuencia inicial, el cual permite establecer un seguimiento de los datos transmitidos. Además, serán enviados en conjuntos formados por varios de ellos, también pactado entre ambos extremos.

- Para cada conjunto el receptor envía un mensaje ACK informando de que han sido recibidos correctamente e indicando el siguiente número de secuencia a transmitir. En caso de error, el ACK incluirá las secuencias de los segmentos que no han sido recibidos.
- El emisor reenviará aquellos que han fallado o el nuevo conjunto. Además, si no recibe ningún ACK en un tiempo determinado dará por hecho que ha sucedido algún error y por consiguiente también serán reenviados.

Por ejemplo, emisor y receptor establecen como tamaño máximo de segmento 1000 bytes, teniendo que ser enviados en conjuntos de 3.

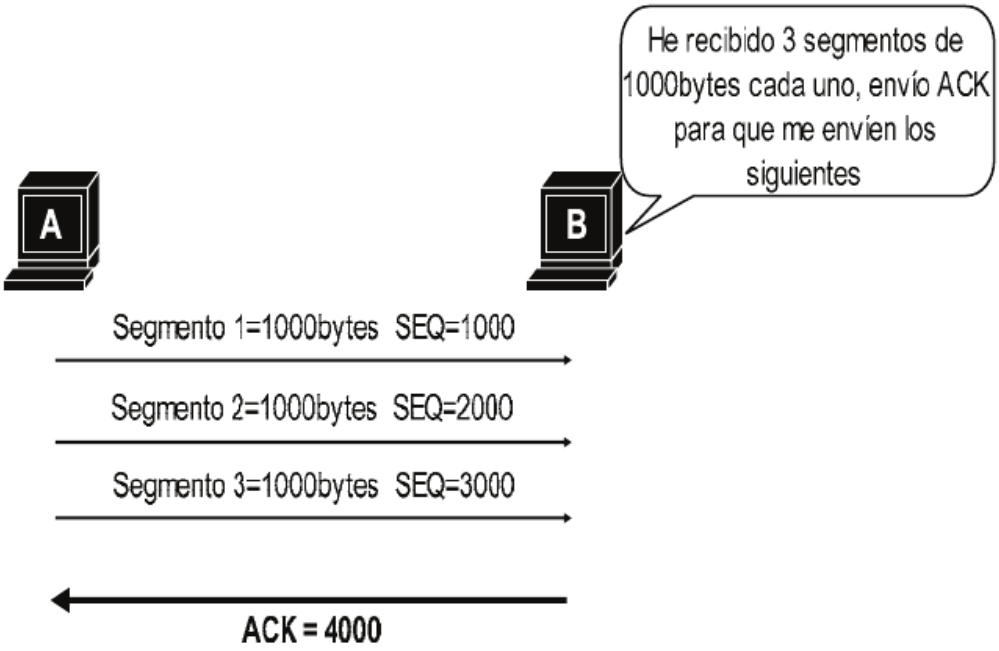


Fig. 1-51.1 Recuperación de errores.

La transmisión de datos entre ambos dispositivos se ha realizado según lo pactado y sin errores, por lo tanto, B envía un ACK informando de que los segmentos han sido recibidos correctamente y solicitando continuar con la secuencia 4000.

En caso de error…

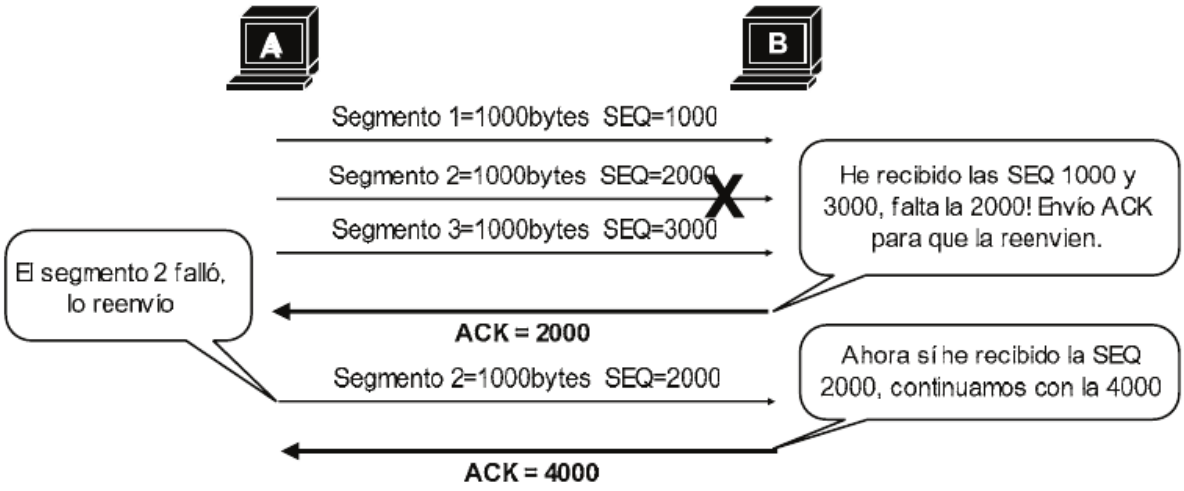


Fig. 1-51.2 Recuperación de errores.

En esta ocasión el segmento 2 no ha sido recibido por B, por lo que solicita su reenvío.

La recuperación de errores representa la mayor diferencia de TCP respecto a UDP.

CONTROL DE FLUJO - VENTANA DESLIZANTE

El control de flujo es la función de TCP mediante la cual se limita la cantidad de datos que pueden ser recibidos de manera simultánea por el destino, logrando beneficios como un mayor aprovechamiento del ancho de banda y mejoras en la recuperación de errores. Se lleva a cabo mediante una técnica denominada ventana deslizante (también conocida simplemente como ventana).

En el ejemplo recién analizado, PC A envía segmentos de 3 en 3 y detiene la transferencia hasta que reciba algún ACK. Precisamente en ello consiste el control de flujo, entre A y B se estableció un tamaño de ventana de 3000 bytes, que indica la cantidad máxima de datos que pueden ser enviados de manera simultánea (3 segmentos de 1000 bytes cada uno).

Una característica a destacar es que su valor puede cambiar durante una misma transferencia, dependiendo de diferentes factores como el ancho de banda disponible o la cantidad de errores que se produzcan. A mayor número de errores, menor tamaño de ventana. El cambio lo establece el destinatario, incluyendo el nuevo tamaño en el ACK que envía al origen.

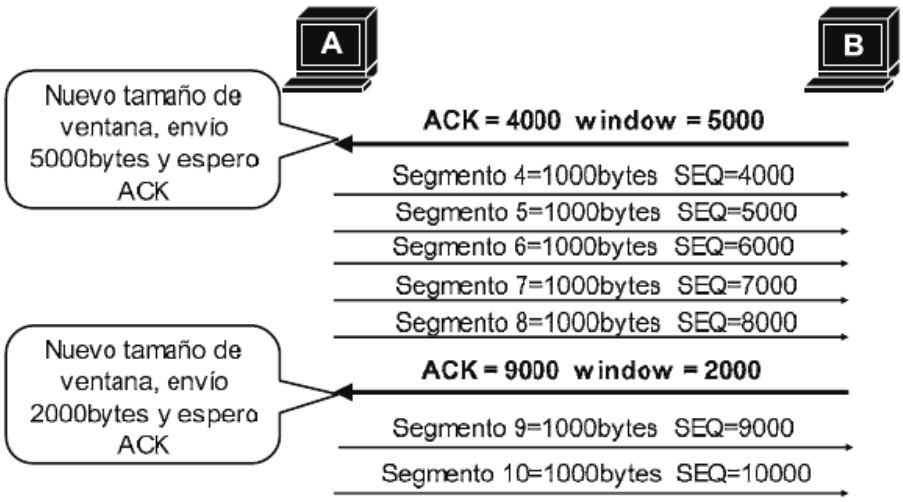


Fig. 1-52 Control de flujo, aplicación de ventana deslizante.

ESTABLECIMIENTO Y FINALIZACIÓN DE LA CONEXIÓN

TCP se define como un protocolo orientado a conexión porque entre emisor y receptor se establece una sesión que permanece activa durante toda la comunicación. Es el primer paso que se lleva a cabo antes de comenzar con el envío de datos y para ello se ejecuta un proceso denominado enlace a 3 vías, donde ambos intercambian una serie de mensajes, definidos en los siguientes pasos:

Paso 1: El origen envía un paquete SYN con un valor aleatorio de secuencia inicial, el cual es utilizado para rastrear el flujo de datos durante la sesión.

Paso 2: El destino recibe el paquete SYN y responde con un ACK para informar que no se produjeron errores durante la transferencia. Además, este mismo ACK contiene el valor del campo enviado por el origen, sumándole 1, junto con un nuevo número de secuencia aleatorio.

Paso 3: Por último, el origen lo recibe y responde con otro ACK.

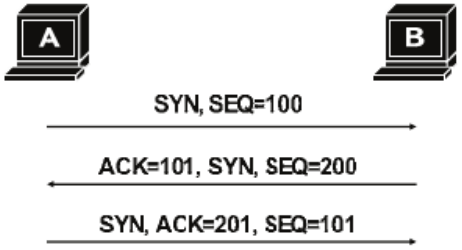


Fig. 1-53 Enlace a 3 vías. Establecimiento de conexión.

Finalizado el intercambio de mensajes, la conexión se da por establecida y comienza la transferencia de datos entre ambos, aplicando la ventana deslizante y recuperación de errores.

Del enlace a 3 vías tal vez produzcan confusión los números de secuencia y ACK. Los primeros son seleccionados aleatoriamente por cada dispositivo, en este caso, PC A tomó el valor 100 y B el 200. Mientras, el ACK indica que el paquete fue recibido correctamente. Para ello, suma uno al número de secuencia recibido. Por ejemplo, B recibió un SYN con secuencia 100, y responde con un ACK 101, mientras que A recibió un SYN con secuencia 200 y responde a este con un ACK 201.

Una vez concluida la transferencia de datos, la sesión entre ambos debe cerrarse. Para ello, TCP ejecuta otro proceso de intercambio de mensajes, el cual es iniciado por el origen enviando un segmento con el mensaje FIN para terminar la sesión. El otro extremo responde primero con un ACK y luego con un FIN. Tras una última respuesta incluyendo el ACK, la sesión entre ambos se da por finalizada.

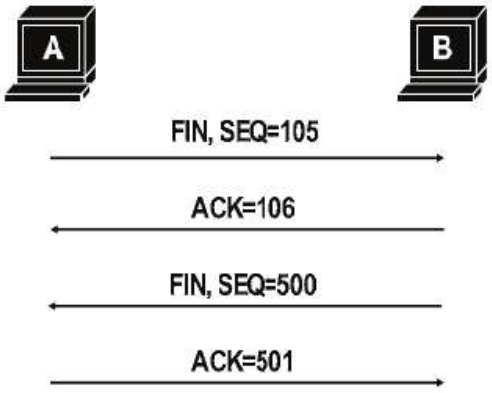


Fig. 1-54 Finalización de conexión en TCP.

REENSAMBLAJE DE DATOS EN EL DESTINO

Debido al tamaño de la ventana deslizante, a los errores durante la transmisión y a los medios físicos disponibles, los diferentes segmentos TCP pueden llegar desordenados al destino, por lo que es necesario reensamblarlos. Esta tarea consiste en aplicar el orden correcto a los datos para que sean reconstruidos exactamente igual a como fueron enviados desde el origen. Para lograrlo se hace uso de los números de secuencia, gracias a los cuales TCP puede llevar a cabo dicho reensamblaje.

UDP (User Datagram Protocol)

Al igual que TCP, UDP facilita la comunicación entre aplicaciones de extremo a extremo en capa 4, sin embargo, es un protocolo muchísimo más sencillo ya que no ofrece ni recuperación de errores (fiabilidad), ni ventana deslizante, ni reensamblaje de datos. Ni siquiera establece una conexión inicial y finalización de esta. Las únicas funciones que incluye son la segmentación, multiplexación y utilización de puertos.

Su modo de operar consiste en enviar datos al destino sin importar el orden en que sean recibidos ni los errores que puedan existir durante la transferencia. Evidentemente todo ello supone algunas desventajas pero también beneficios ya que la cabecera del segmento es mucho más ligera que la creada por TCP, traduciéndose en una mayor velocidad de procesamiento y transferencia, debido en gran medida a que el origen no tiene que esperar ACKs ni reenviar segmentos.

Las aplicaciones que hacen uso de UDP como protocolo en capa de transporte deben ser tolerantes a fallos, dos ejemplos de ello son VoIP o streaming de vídeo, donde la pérdida de algún segmento no produce prácticamente ningún impacto sobre la comunicación. Lo que sí podría causar problemas es que estos lleguen desordenados. Como se ha nombrado con anterioridad, UDP no incluye reensamblaje, por lo que deben ser las propias aplicaciones las que apliquen técnicas para mostrar los datos de manera ordenada.

Algunos servicios que hacen uso de UDP son:

Aplicación	Puerto	Protocolo
DNS	53	UDP (también hace uso de TCP)
DHCP	67, 68	UDP
SNMP	161	UDP
RTP (VoIP) y vídeo	16, 384-32.767	UDP

DIFERENCIAS ENTRE TCP Y UDP

	TCP	UDP
Utilización de puertos	SI	SI
Multiplexación	SI	SI
Orientado a conexión	SI	NO
Control de flujo	SI	NO
Recuperación de errores	SI	NO
Reensamblaje de segmentos	SI	NO

TEST CAPÍTULO 1: REDES INFORMÁTICAS. CONCEPTOS BÁSICOS

1.- ¿Cuál de los siguientes protocolos es utilizado para transferencia de páginas web?

A. FTP
B. SMTP
C. POP3
D. HTTP

2.- En OSI, ¿qué nombre recibe la PDU en capa 2?

A. Trama
B. Segmento
C. Paquete
D. Bits

3.- ¿En cuántas capas se divide el modelo TCP/IP?

A. Siete
B. Cuatro
C. Dos
D. Cinco

4.- ¿Cuáles de los siguientes protocolos operan en capa 4? (Seleccionar dos respuestas)

A. IP
B. UDP
C. Frame Relay
D. POP3
E. TCP

5.- La encapsulación de datos consiste en…

A. Comprimir los datos para que su tamaño sea menor y por lo tanto se aproveche mejor el ancho de banda disponible en la red.
B. El proceso ejecutado en capa 1 que transforma los datos en bits con el fin de poder ser enviados a través del medio físico.
C. El proceso por el cual se agregan cabeceras y tráiler a los datos mientras atraviesan las diferentes capas, creando en cada una de ellas la PDU correspondiente.
D. Utilizar las direcciones de origen y destino para que la comunicación pueda establecerse entre emisor y receptor.

6.- El _____ es considerado el dispositivo por excelencia en capa 3.

A. Switch
B. Hub
C. Router
D. Punto de acceso

7.- ¿Qué capa del modelo OSI es la encargada de proporcionar una entrega fiable y libre de errores?

A. Red
B. Enlace de datos
C. Transporte
D. Física

8.- En la cabecera creada en capa 3, ¿qué representa el campo TTL?

A. El número máximo de saltos que puede dar el paquete antes de ser descartado.
B. El tiempo de vida (*time to live*) en minutos del paquete antes de ser descartado.
C. El tiempo, en segundos, que tarda un paquete desde un origen hasta un destino.
D. Es el campo encargado de identificar al origen de la conexión.

9.- Ethernet está definido en el estándar IEEE...

A. 802.5
B. 802.3
C. 802.11
D. Q.921

10.- Token Ring está definido en el estándar IEEE...

A. 802.5
B. 802.3
C. 802.11
D. Q.921

11.- ¿Qué capas del modelo OSI se engloban en la capa de acceso a la red del modelo TCP/IP? (Seleccionar dos respuestas)

A. Sesión
B. Enlace de datos
C. Red
D. Física
E. Transporte

12.- FastEthernet opera a una velocidad de…

A. 10M bps
B. 100 Mbps
C. 1000 Mbps
D. 10000 Mbps

13.- ¿Qué tipo de medio físico es necesario en una red 1000BASE-SX?

A. Cableado STP
B. Cableado UTP
C. Fibra óptica
D. Cable coaxial

14.- ¿Cuál de las siguientes afirmaciones corresponde a una diferencia entre un hub y un switch?

A. El switch es capaz de enrutar paquetes, el hub no.
B. El switch establece un dominio de colisión en cada uno de sus puertos, el hub no.
C. El switch establece un dominio de broadcast en cada uno de sus puertos, el hub no.
D. El switch agrega una capa de seguridad a la red porque en él se pueden configurar funciones de firewall, el hub no.

15.- ¿Por qué los switchs son considerados dispositivos de capa 2?

A. Porque aplican los protocolos TCP y UDP.
B. Porque el reenvío de tramas se lleva a cabo gracias al uso direcciones IP.
C. Porque el reenvío de tramas se lleva a cabo transfiriendo bits.
D. Porque el reenvío de tramas se lleva a cabo a través de direcciones MAC.

16.- Se ha agregado un nuevo switch a la red, el cual deber ser conectado a otro ya existente. ¿Qué tipo de cable es necesario para establecer dicha conexión?

A. Directo
B. Cruzado
C. Serial
D. Consola

17.- La dirección FFFF.FFFF.FFFF representa…

A. Un broadcast de capa 2.
B. Un broadcast de capa 3.
C. Un unicast de capa 2.

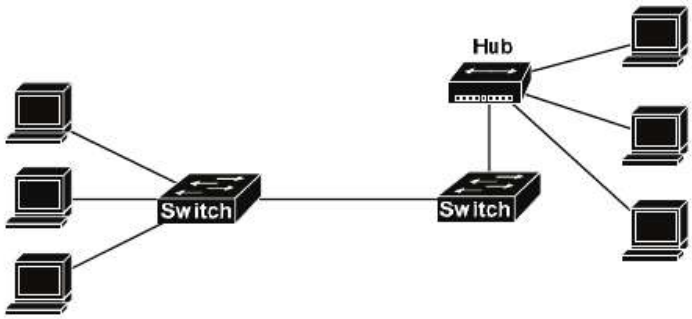
D. Un unicast de capa 3.
E. Un multicast de capa 2.
F. Un multicast de capa 3.

18.- En un enlace serial, ¿qué router establece la velocidad de reloj?
A. En un enlace serial no es necesaria la velocidad de reloj.
B. Ambos, pero para que el enlace opere correctamente deben aplicar el mismo valor.
C. El router que actúa como DCE.
D. El router que actúa como DTE.

19.- ¿En qué capa opera el protocolo PPP?
A. Capa 1.
B. Capa 2.
C. Capa 3.
D. Capa 4.

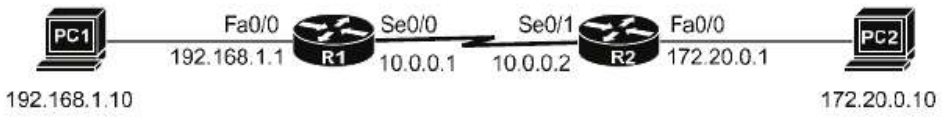
20.- En la conmutación por paquetes….
A. Se establece siempre el mismo circuito físico de comunicación.
B. Los paquetes pueden tomar diferentes circuitos virtuales para llegar a su destino.
C. Se requiere usuario y contraseña para acceder al circuito.
D. El protocolo mayormente usado es Ethernet.

21.- En la siguiente topología…



¿Cuántos dominios de colisión existen?
A. Ocho
B. Tres
C. Cinco
D. Dos

22.- En la siguiente comunicación…



PC1 envía un paquete IP a PC2. ¿Qué dirección IP de origen y destino tendrá dicho paquete cuando sea reenviado a través de la interfaz Fa0/0 de R2?

A. Origen: 10.0.0.2 Destino: 172.20.0.10
B. Origen: 172.20.0.1 Destino: 172.20.0.10
C. Origen: 192.168.1.1 Destino: 172.20.0.10
D. Origen: 192.168.1.10 Destino: 172.20.0.10

23.- Haciendo uso de la misma topología de la cuestión número 22… PC1 envía un paquete IP a PC2, sin embargo, R1 no dispone de ninguna ruta hacia la red 172.20.0.0. ¿Cómo procede R1?

A. Lo reenvía a través de todas sus interfaces.
B. Lo reenvía a través de todas sus interfaces excepto por la cual fue recibido el paquete IP.
C. Envía un mensaje broadcast a la red solicitando a otros dispositivos una ruta válida hacia la red 172.20.0.0.
D. Reenvía el paquete a través de la interfaz Se0/0.
E. Descarta el paquete.

24.- La dirección IP 192.168.50.33 es representada en binario como…

A. 11000000.10101000.00110010.00100001
B. 11000000.10101000.00111010.10100001
C. 11000000.10101000.10110000.00101000
D. 11000000.10101000.00111110.00111001

25.- La dirección IP 172.30.0.156 es representada en binario como…

A. 10101100.00011110.00000000.10011100
B. 10101100.00001100.00000000.10011111
C. 10101110.01011110.00000000.11111100
D. 10101100.00011111.00000000.10111100

26.- La dirección IP 00001010.10010110.00001110.00010101 es representada en formato decimal como…

A. 10.65.14.99
B. 10.22.128.64
C. 10.190.15.251

D. 10.150.14.21

27.- La dirección IP 10101100.10110100.00101111.00000010 es representada en formato decimal como…

A. 172.19.86.32
B. 172.180.47.2
C. 172.180.58.2
D. 172.192.31.2

28.- La dirección IP 125.0.0.1 pertenece a la clase…

A. A
B. B
C. C
D. D

29.- Se ha creado una nueva subred en la topología y su direccionamiento debe ser de clase B. De las redes mostradas en la lista, ¿cuáles cumplen los requisitos? (Seleccionar dos respuestas)

A. 125.0.0.0
B. 192.168.0.0
C. 129.0.0.0
D. 191.0.0.0
E. 10.0.0.0

30.- ¿Qué utilidad de red envía mensajes ICMP?

A. DHCP
B. DNS
C. PING
D. ARP

31.- En la capa de transporte, ¿cómo se logra que los segmentos sean recibidos por la aplicación de destino correcta?

A. Gracias a la dirección IP de destino.
B. Gracias a la utilización de puertos.
C. Gracias al número de secuencia.
D. Gracias al enrutamiento en capa 3.
E. Gracias al uso de ventana deslizante.
F. Gracias al número de ACK.

32.- ¿Qué protocolo es más fiable en cuanto a entrega de extremo a extremo?

A. UDP
B. IP

C. SSH
D. TCP

33.- ¿En qué consiste la multiplexación?
A. En la utilización de diferentes medios físicos por distintos dispositivos de manera simultánea.
B. En segmentar los datos para que diferentes dispositivos puedan hacer uso del mismo medio físico de manera simultánea.
C. En utilizar números de puerto para la correcta comunicación entre aplicaciones.
D. En proveer de recuperación de errores al protocolo TCP.

34.- ¿Cuáles de los siguientes protocolos no son orientados a conexión? (Seleccionar dos respuestas)
A. UDP
B. IP
C. TCP
D. HTTP

35.- ¿Para qué son utilizados los números de secuencia (SEQ) y los ACK en TCP?
A. Para el proceso de multiplexación.
B. Para definir el tamaño de segmento.
C. Para la recuperación de errores.
D. Para identificar a la aplicación correcta durante la comunicación.

36.- ¿Cuáles de las siguientes características corresponden al uso de ventana deslizante en TCP? (Seleccionar dos respuestas)
A. Control de flujo.
B. Mayor velocidad de enlace.
C. Mejora la capacidad de envío de datos.
D. Mejor aprovechamiento del ancho de banda.
E. Es una función de los protocolos TCP y UDP.

37.- ¿Cuáles de los siguientes protocolos hacen uso de UDP? (Seleccionar tres respuestas)
A. HTTP
B. FTP
C. POP3
D. HTTPS
E. SMTP
F. SNMP
G. DHCP

H. TFTP

38.- De los siguientes protocolos, ¿cuáles forman parte de la capa de enlace de datos de TCP/IP? (Seleccionar dos respuestas)

A. HTTP
B. FTP
C. PPP
D. UDP
E. IP
F. Ethernet
G. SMTP
H. POP3

39.- ¿Cómo se denomina la dirección de capa 2 utilizada en redes Frame Relay?

A. MAC
B. IP
C. DLCI
D. PPP
E. TCP

40.- ¿Qué capa del modelo OSI define los estándares de cables y conectores?

A. Capa 1
B. Capa 2
C. Capa 3
D. Capa 4
E. Capa 5
F. Capa 6
G. Capa 7

41.- De las siguientes características, ¿cuál corresponde a una función de los protocolos de enrutamiento?

A. Agregar una capa de seguridad a la red.
B. Comprobar automáticamente las direcciones IP de los diferentes routers de la red.
C. Prevenir bucles de capa 2.
D. Prevenir bucles de capa 3.

42.- De las siguientes afirmaciones, ¿cuál define mejor al algoritmo CSMA/CD?

A. Es más eficiente cuando en el mismo segmento de red tan solo existen dos dispositivos.
B. Nunca se producen colisiones cuando es aplicado.

C. Pueden producirse colisiones, pero el algoritmo define cómo debe actuar cada dispositivo en estos casos.
D. Ninguna de las respuestas es correcta.

43.- ¿Cuáles de las siguientes direcciones deben ser únicas a nivel global? (Seleccionar dos respuestas)
A. Dirección MAC.
B. Dirección IP privada.
C. Dirección IP pública.
D. Dirección de destino en capa 3.

44.- ¿Cuál de los siguientes campos es agregado en la versión HDLC de Cisco con respecto a la versión original del protocolo creada por ISO?
A. Dirección
B. FCS
C. ACK
D. Flag
E. Type

45.- De las siguientes funciones, ¿cuáles corresponden a la capa 3 del modelo OSI? (Seleccionar dos respuestas)
A. Direccionamiento físico.
B. Recuperación de errores.
C. Comunicación entre aplicaciones.
D. Direccionamiento lógico.
E. Selección de rutas.

46.- PC1 y PC2 están situados en diferentes segmentos separados por un router Cisco, en cuya topología no se ha aplicado subnetting. Si la IP de PC1 es la 10.10.10.1, ¿cuáles de las siguientes direcciones pueden corresponder a PC2? (Seleccionar dos respuestas)
A. 10.1.1.1
B. 10.2.2.2
C. 121.100.20.1
D. 172.30.30.30
E. 192.168.15.2
F. 127.0.01

47.- ¿Cuál de las siguientes afirmaciones es correcta respecto al envío de paquetes de un host a su puerta de enlace?
A. Por defecto, los hosts envían todos los paquetes a su puerta de enlace.

B. Los hosts envían los paquetes a su puerta de enlace si la dirección de destino pertenece a la misma subred que el host.
C. Los hosts envían los paquetes a su puerta de enlace si la dirección de destino pertenece a una subred diferente que el host.
D. Los hosts solo envían paquetes a su puerta de enlace cuando la red de destino se encuentra en su tabla de rutas.

48.- PC1 recibe un paquete "ICMP Echo Reply" desde PC2. ¿Qué utilidad, protocolo o paquete utilizó PC1 para recibir dicha respuesta? (Seleccionar dos respuestas)

A. FTP
B. HTTP
C. HTTP GET
D. Ping
E. DNS
F. DHCP
G. DHCP REQUEST
H. ICMP Echo Request

49.- ¿Cuál de las siguientes funciones es ejecutada por los protocolos TCP y UDP?

A. Enrutamiento.
B. Direccionamiento.
C. Recuperación de errores.
D. Ventana deslizante.
E. Cifrado.
F. Entrega fiable.
G. Multiplexación y utilización de puertos.
H. Ninguna de las anteriores.
I. Todas las anteriores.

CONFIGURACIÓN DE SWITCHS CISCO 2

MODO DE OPERAR DE SWITCHS

A día de hoy, las características y prestaciones ofrecidas por los switchs lo convierten en un elemento imprescindible en todo tipo de redes. En el capítulo 1 han sido analizados los conceptos más básicos de este dispositivo, pero ¿cómo operan internamente? ¿Qué funciones ofrecen y cómo configurarlas? Todo ello será objeto de estudio a lo largo de las siguientes secciones, con la finalidad de lograr crear una LAN Ethernet segura y eficiente.

Para comprender tanto la lógica aplicada por los switchs como su modo de operar resulta necesario conocer a sus antecesores, los hubs y bridges (puentes).

Los hubs son dispositivos intermediarios que crean una LAN 10BASE-T entre los diferentes elementos conectados a él, posibilitando la comunicación entre todos ellos y utilizando cableado UTP para establecer la conexión física. Nacen con el objetivo de solucionar los problemas de la topología bus, pero aún mantienen los siguientes inconvenientes:

- Cuando recibe señales (datos) por un puerto, las reenvía a través de todos excepto por el cual fueron recibidas, lo que significa que cualquier tipo de tramas, ya sean broadcast, multicast o unicast serán procesadas por todos los miembros de la LAN, suponiendo un problema de seguridad.
- Todos los dispositivos conectados al hub hacen uso del mismo medio compartido, por lo tanto, se establece un solo dominio de colisión para todos

ellos, de tal manera que cuando dos o más intenten transmitir de manera simultánea se producirán colisiones, en cuyo caso aplicarán la técnica de control de acceso al medio CSMA/CD para solucionar el problema.

Una de las mayores desventajas de los hubs es la utilización de un medio compartido y la creación de un solo dominio de colisión, lo que se traduce en una mala gestión del ancho de banda y retraso en las comunicaciones debido a colisiones. Cuanto mayor sea la red, más se agrava el problema. Es por ello que comenzaron a desarrollarse nuevos dispositivos con el fin de solucionar o al menos aliviar dicho inconveniente. Uno de ellos fue el *bridge*, cuya función principal consiste en dividir los dominios de colisión creados por los hubs, logrando los siguientes beneficios en cuanto a rendimiento se refiere:

- Mejoran el aprovechamiento del ancho de banda gracias a la creación de diferentes medios compartidos.
- Gracias a ello también se reduce el número de colisiones existentes, ya que las tramas no son reenviadas entre diferentes dominios de colisión.

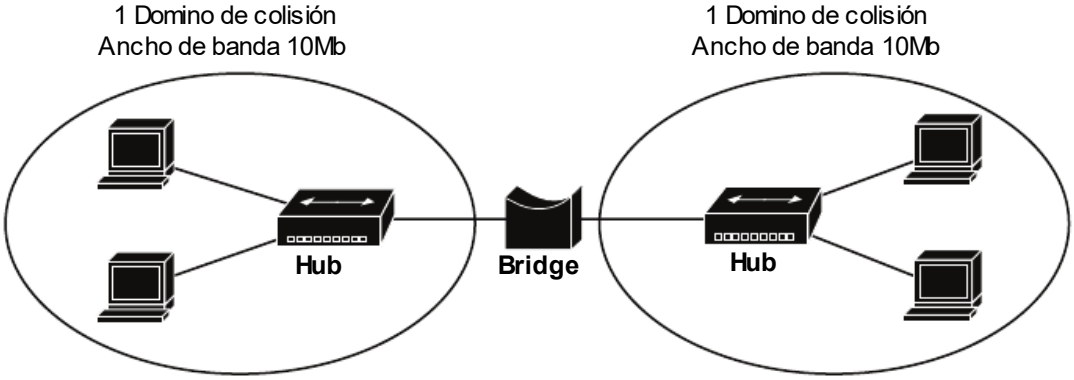


Fig. 2-1 Dominios de colisión, bridge y hub.

En la topología, el bridge ha generado dos dominios de colisión con 10Mb de ancho de banda dedicado en cada uno de ellos. Si fuera eliminado y los hub conectaran directamente, tan solo existiría uno, con 10 Mb de ancho de banda para todos los dispositivos.

Su utilización supuso avances en la creación de redes, sin embargo, aún continuaba siendo necesario desarrollar un único dispositivo que incorporara todas las funciones que tanto bridges como hubs ofrecen por separado, es decir, crear LANs, evitar colisiones, separar dominios de colisión y un mejor aprovechamiento del ancho de banda.

Con ello nació el switch, que a día de hoy continúa siendo el dispositivo por excelencia en toda red.

Switchs

Como se ha mencionado, los switchs son dispositivos intermediarios de red que incluyen todas las funciones llevadas a cabo tanto por hubs como por bridges, agregando, además, numerosos avances con el fin de optimizar el rendimiento y capacidades de la red. Sus características principales son:

- Crean una red LAN Ethernet entre todos los dispositivos conectados a él.
- Soporta velocidades mayores que sus predecesores, pudiendo alcanzar los 1000 Mbps.
- Establecen un dominio de colisión por cada interfaz, lo que se traduce en un ancho de banda dedicado en cada una de ellas y además una conexión punto a punto, por lo que no es necesario aplicar ninguna técnica de control de acceso al medio ya que las colisiones resultan prácticamente inexistentes.
- Cuando recibe una trama por alguna de sus interfaces, lee la MAC de destino y la reenvía solo a su destinatario, hecho que genera una capa de seguridad y un mejor aprovechamiento del ancho de banda. Como el Switch basa su decisión de reenvío en la dirección MAC, es considerado un dispositivo de capa 2.
- Cada uno de sus enlaces soporta el modo de transferencia *full duplex*, el cual permite enviar y recibir de manera simultánea a través del mismo medio físico.

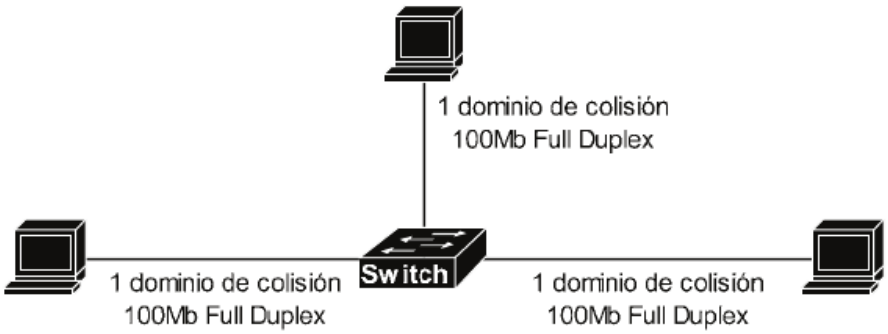


Fig. 2-2 Dominios de colisión generados por el switch.

Este switch crea una LAN 100BASE-T compuesta por 3 PCs donde cada uno de ellos dispondrá de un ancho de banda dedicado de 100 Mb y una conexión *full duplex* en la cual no se producirán colisiones durante la comunicación. Para lograrlo, se

generan 3 dominios de colisión diferentes, uno por cada enlace. Además, los switchs solucionan el problema principal de los hubs, ya que reenvían las tramas solo al destinatario correcto, basándose para ello en la dirección MAC de destino. Estas pueden ser de tres tipos: unicast, multicast y broadcast.

Una dirección unicast es aquella que identifica a un único dispositivo. Por ejemplo, si A desea enviar datos a B, incluirá la MAC de este como destino.

Una multicast identifica a un grupo de dispositivos, por ejemplo, una aplicación envía una actualización a la red utilizando una dirección multicast como destino. Las tramas solo serán procesadas por aquellos dispositivos que ejecuten dicha aplicación.

Por último, una dirección broadcast es aquella que identifica a todos los dispositivos, por lo tanto, cuando un switch reciba una trama de este tipo la reenviará a través de todas sus interfaces. La dirección broadcast en capa 2 es FFFF.FFFF.FFFF.

Conociendo las características de cada una de ellas se puede proceder a analizar las 3 funciones principales desempeñadas por los switchs:

- Aprender direcciones MAC de dispositivos conectados.
- Decidir por cuál de sus interfaces reenviar una trama, basándose para ello en la MAC de destino.
- Cuando existen varios switchs conectados entre sí, evitar bucles de capa 2 mediante la aplicación del protocolo STP *(Spanning Tree Protocol.)*

APRENDER DIRECCIONES MAC DE DISPOSITIVOS CONECTADOS

Para lograr que el reenvío de una trama se dirija solo al destinatario correcto, los switchs almacenan en su memoria una tabla que asocia las direcciones MAC con la interfaz a la que están conectadas. Esta recibe el nombre de *MAC address table,* tabla *CAM (Content Addressable Memory)* o simplemente *Switching Table*.

El proceso para crearla y actualizarla resulta bastante sencillo. Cada vez que una trama atraviesa el switch, este lee su MAC de origen y la agrega a la tabla, vinculándola con la interfaz por la cual fue recibida.

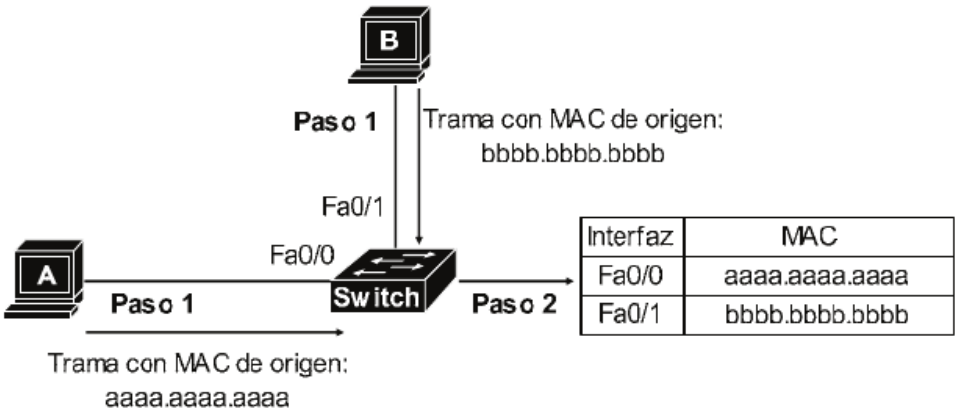


Fig. 2-3 Tabla de MACs.

Paso 1: el switch recibe tramas desde A y B.

Paso 2: lee las direcciones de capa 2 de origen y las agrega a la tabla de MACs, asociando cada una de ellas a la interfaz por la cual fue recibida.

Pero ¿qué ocurre si un switch recibe una trama por una interfaz, lee la MAC de origen y resulta que ya está asociada a otra interfaz? En este caso se borraría la entrada antigua y se crearía una nueva, vinculando la MAC con la interfaz por la cual fue recibida.

¿Una vez aprendida y asociada una MAC con una interfaz, permanece siempre grabada en la memoria del switch? No, cada entrada en la tabla contiene un temporizador denominado *"inactivity timer",* el cual tiene por objeto eliminar aquellas que permanezcan inactivas durante un tiempo determinado (varía dependiendo del modelo de switch). Si durante dicho tiempo no se ha recibido ninguna trama de una determinada MAC, la entrada es eliminada. Si por el contrario se reciben tramas, el temporizador es reseteado a 0.

¿Cuántos registros puede almacenar una tabla de MACs? Depende del modelo de switch, versión de IOS y hardware. Sea como fuere, el tamaño siempre es una memoria limitada, normalmente más que suficiente. Si fuera sobrepasada, se sobrescriben las entradas más antiguas por las nuevas. Existe un ataque de red basado precisamente en ello, completar la tabla de MACs con solicitudes falsas para que el switch no sepa por dónde reenviar las tramas.

REENVÍO DE TRAMAS EN RELACIÓN CON LA MAC

La decisión de reenvío de los switchs se basa por completo en la tabla de MACs, aplicando la siguiente lógica:

- *Paso 1:* el switch recibe una trama y lee la dirección de destino.
- *Paso 2:* acto seguido la busca en su tabla de MACs. Si existe, la envía por la interfaz asociada a ella. Si no existe, la reenvía a través de todas sus interfaces esperando la respuesta del dispositivo que hace uso de la misma. Si este responde, se asocia a la interfaz por la cual recibió la respuesta.

Por ejemplo, PC A envía una trama unicast a B…

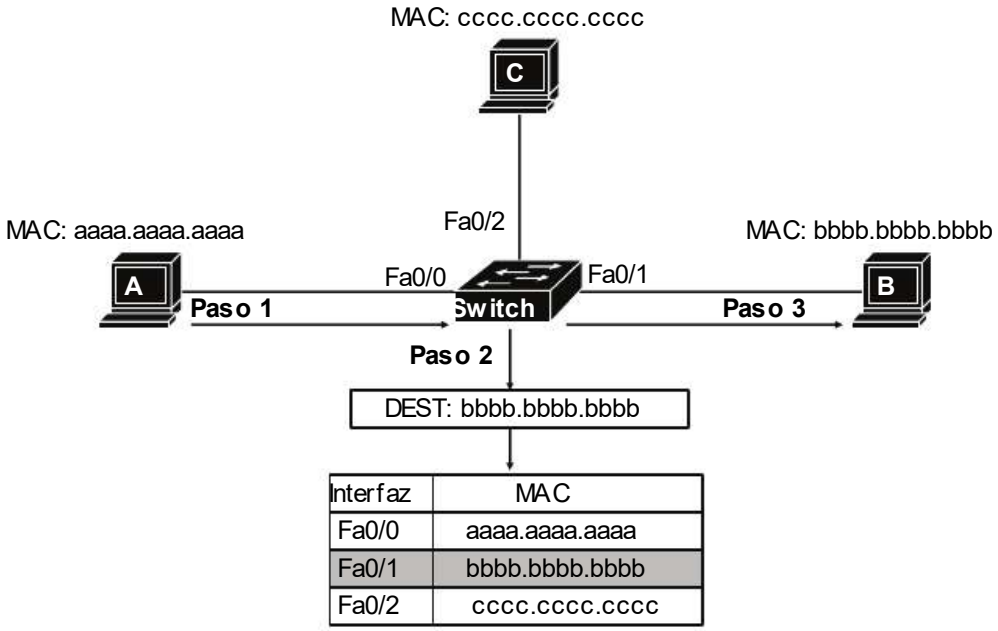


Fig. 2-4 Reenvío de tramas en relación con la MAC de destino.

Paso 1: la trama es recibida por el switch.

Paso 2: lee la dirección de destino en capa 2, que en este caso es *bbbb.bbbb.bbbb.* Acto seguido la busca en su tabla y comprueba que debe ser reenviada por la interfaz Fa0/1 para que llegue a su destino.

Paso 3: la trama es recibida solo por el PC B.

La tabla de MACs puede contener más de una entrada por interfaz, esto ocurre cuando existen dos o más switchs conectados entre sí. En estos casos, la interfaz que

los une tendrá asociada las MACs de todos los dispositivos pertenecientes al switch vecino.

PROCESAMIENTO INTERNO EN SWITCHS CISCO

En capa 2, el campo "dirección de destino" se encuentra ubicado en la cabecera, por lo tanto, no resulta necesario analizar la trama al completo para obtener dicho dato. Esto da lugar a que se puedan aplicar diferentes métodos de procesamiento para su reenvío, siendo los siguientes:

- *Store-and-Forward:* Es el método más común en switchs Cisco y consiste en recibir la trama al completo, almacenarla en un buffer y acto seguido reenviarla por la interfaz oportuna. A su vez es el más recomendado porque permite comprobar el campo FCS (errores).
- *Cut-Through:* En este método el switch lee la MAC de destino, selecciona la interfaz de salida y reenvía cada bit al mismo tiempo que es recibido. Es decir, la trama no se almacena en un buffer. Gracias a ello se logra reducir la latencia pero en contra no se puede comprobar el campo FCS.
- *Fragment-free*: El switch recibe los primeros 64 bytes de la trama para acto seguido reenviarla aplicando el método *cut-through*.

El procesamiento interno puede variar dependiendo del modelo de switch y fabricante. En Cisco se aplica el método *Store-and-forward* por defecto.

EVITAR BUCLES DE CAPA 2 MEDIANTE STP

Lo más común en una red corporativa es que los switchs que la componen conecten entre sí a través de varios enlaces, lo cual agrega beneficios como mayor ancho de banda y disponibilidad gracias a la redundancia. Sin embargo, esta práctica también puede suponer un grave problema, ya que al existir diferentes rutas y multitud de links pueden formarse bucles de capa 2.

Para evitarlo, los switchs ejecutan el protocolo STP *(Spanning Tree Protocol)* el cual se encarga de crear una topología libre de bucles. STP es un protocolo tan amplio e importante que será analizado en profundidad en el capítulo 3 *"Spanning Tree Protocol"*.

Switch Stacking

En ocasiones, el número total de dispositivos pertenecientes a una sede o lugar físico en concreto requiere la instalación de varios switchs. En estos casos, una de las

prácticas más habituales consiste en la conexión física de los mismos mediante fibra o etherchannels (analizados en el capítulo 3), sin embargo, cada cual operará de manera independiente, lo que implica que:

- En cada switch se debe configurar una IP de gestión con el fin de poder acceder al mismo mediante el modo de acceso remoto habilitado (SSH, Telnet, GUI).
- Cada uno de ellos participará en STP y/o VTP de manera independiente.
- Cada switch gestionará su propia tabla de MACs, así como las interfaces físicas pertenecientes al dispositivo en cuestión.
- Cada uno de ellos dispone su propio fichero de configuración y basa su modo de operar en torno al mismo.

En definitiva, cada dispositivo opera de manera totalmente independiente al resto. Ello no significa que suponga un mal diseño o que la red no opere con normalidad, ya que, correctamente implementado y sin importar el Switch al que conecten, cada host obtendrá acceso a la red y a los recursos ubicados en la misma.

Sin embargo, este sistema puede ser optimizado gracias al "*stacking*", método que consiste en agrupar varios switchs físicos en tan solo uno lógico, logrando con ello los siguientes beneficios:

- Tan solo se debe definir una IP de gestión, a través de la cual se accede a la configuración de todos los dispositivos físicos de manera conjunta, facilitando en gran medida la administración.
- En STP y/o VTP tan solo participa el switch virtual.
- Tanto la tabla de MACs como las interfaces de todos los switchs físicos son gestionadas por el lógico.
- Tan solo se requiere un fichero de configuración.

Para agrupar switchs Cisco en stack bastará con apilarlos y establecer la conexión entre los mismos a través de la interfaz destinada a ello. Esta se encuentra ubicada en la parte posterior del dispositivo y requiere un cable VHDCI dedicado a tal propósito, el cual, dependiendo del modelo, podrá alcanzar velocidades de 10 Gbps (*FlexStack*, en Cisco 2960-S y 2960-X) o 20 Gbps (*FlexStack-Plus*, en modelos 2960-XR). Un ejemplo de stacking podría ser el siguiente:

Fig. 2-5 Switchs en stack.

Donde 4 switchs crearán uno lógico que gestionará todos ellos de manera conjunta. Para que sea posible, uno de los dispositivos físicos toma el rol de máster *(stack master),* sobre el cual recae tanto la configuración como la administración y operación del resto de swicths. Además, contiene la tabla de MACs, por lo que la decisión de reenvío ante cualquier comunicación también debe ser gestionada por el máster. Por ejemplo, imagina que el switch 1 ha tomado dicho rol, y el switch 4 ha recibido una trama cuyo destino está conectado en la interfaz Gi3/20 del switch 3. Bien, el modo de proceder consistirá en que el switch 4 reenvíe la trama al 1, este compruebe su tabla de MACs y concluya que el destino está conectado en la interfaz Gi3/20 del switch 3, reenviando la trama hacia dicho puerto.

Esta técnica suele ser aplicada sobre Switchs ubicados en la capa de acceso, diseñados para dar soporte a dispositivos finales y equipados con hardware específico que permita implementar dicha tecnología.

En las capas de distribución o núcleo, donde los switchs requieren mayor capacidad, puede ser implementada otra tecnología denominada *chassis aggregation*, cuya finalidad coincide con la recién analizada, es decir, agrupar una serie de switchs físicos en tan solo uno lógico. Sin embargo, en este caso, la conexión entre dispositivos se lleva a cabo mediante uno o varios etherchannels (LACP).

ACCESO Y CONFIGURACIÓN BÁSICA

Cuando se instala un switch Cisco por primera vez tan solo bastará con conectar los dispositivos a sus interfaces para que formen parte de la misma LAN. Ello es debido a que incluyen una configuración por defecto para que resulten operativos desde el primer momento, sin embargo, esta es muy básica y carece de seguridad, por lo que en redes de tamaño medio o grande se hace necesario aplicarla manualmente con el fin de adecuarla a las necesidades de la compañía.

En la presente sección serán analizados los diferentes modos de configuración del switch, así como los aspectos más básicos de acceso y seguridad. Todo ello a través de la CLI, que es la consola incorporada en el sistema operativo y a través de la cual se administra el dispositivo mediante línea de comandos.

Acceso a la configuración a través de la CLI

Tanto switchs como routers Cisco incorporan IOS *(Internetwork Operating System)* como sistema operativo, el cual desarrolla la tarea de implementar controles y funciones lógicas sobre el dispositivo. Este a su vez incluye una consola de administración denominada CLI, desde la cual se aplican las configuraciones necesarias. Pero, ¿cómo acceder a ella?

Se podrá llevar a cabo de dos maneras, remota y físicamente, sin embargo, el primer acceso debe ser físico, ya que la configuración por defecto no habilita el acceso remoto. Para ello resulta necesario un cable de consola, también denominado "*rollover cable",* cuya función consiste en comunicar un PC con el sistema de administración del dispositivo. En un extremo dispone de una conexión RJ-45, que conecta con el puerto de consola del switch, mientras que el otro extremo está compuesto por un puerto serie (o DB-9), que conecta con el PC.

Establecida la conexión física se podrá acceder a la CLI a través de cualquier software emulador de terminal ejecutado en el PC, como "*Putty*", configurado con los siguientes parámetros:

- Tipo de conexión: Serial
- Speed (baud): 9600 bits/ segundo
- Data bits: 8
- Stop Bits: 1
- Parity: None
- Flow Control: XON/XOFF

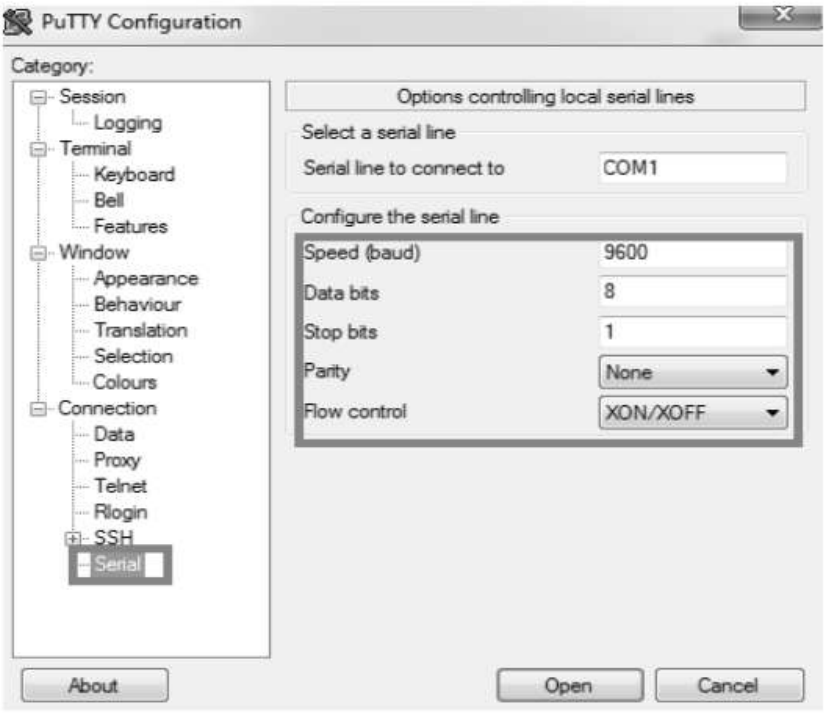


Fig. 2-6 Acceso a la CLI mediante Putty.

Tras pulsar sobre "*Open*" se abrirá una nueva ventana en línea de comando, la CLI.

Los modelos actuales también incluyen una conexión USB para poder acceder a la consola.

MODOS DE OPERAR

La CLI dispone de dos modos, usuario y privilegiado. Cuando se accede a ella, ya sea de manera local o remota, siempre se inicia en modo usuario, también denominado *EXEC* y representado mediante el *prompt* "[*Nombre*]**>**". Este tan solo permite ejecutar comandos de visualización de configuraciones y de testeo, pero nunca modificaciones de configuración.

Para poder aplicar cualquier tipo de cambio se debe acceder al modo privilegiado, también denominado *enable* y representado mediante el *prompt* "[*Nombre*]**#**". Este habilita la ejecución de todos los comandos disponibles en el switch, es decir, se obtiene control total sobre el mismo. Para acceder, se debe, desde el modo usuario, introducir el comando "enable", mientras que para finalizar la sesión, "disable".

```
Switch>
Switch> enable
Switch# disable
Switch>
```

MODOS DE CONFIGURACIÓN

En IOS, al modo configuración global, tan solo disponible desde el modo privilegiado, se accede mediante el comando **configure terminal.** Una vez ejecutado, el *prompt* es representado como "[*Nombre*]**(config)#".**

```
Switch>
Switch> enable
Switch# configure terminal
Switch(config)#
```

Desde el mismo se obtiene acceso a todos los comandos de configuración disponibles en el switch. Sin embargo, para hacer este proceso más sencillo e intuitivo, IOS divide su administración en diferentes submodos, los cuales pueden ser entendidos como un apartado específico para una determinada función o característica y que incluye únicamente los comandos disponibles para ella. Por ejemplo, para configurar una interfaz primero se debe acceder al submodo apropiado, para a posteriori aplicar los comandos de configuración necesarios.

A los diferentes submodos se accede siempre desde el modo de configuración global y para salir se debe ejecutar el comando **exit.** Lo más comunes son:

Nombre de submodo	Comando para acceder	Prompt
Line	line console 0 line vty 0 15	Hostname(config-line)#
Interface	Interface [interfaz]	Hostname(config-if)#
VLAN	vlan [num de vlan]	Hostname(config-vlan)#

```
Switch# configure terminal
Switch(config)#line console 0
Switch(config-line)#exit
Switch(config)#line vty 0 15
Switch(config-line)#exit
Switch(config)#interface Fa0/1
Switch(config-if)#exit
Switch(config)#vlan 10
Switch(config-vlan)#exit
Switch(config)#
```

SEGURIDAD BÁSICA DE ACCESO A LA CLI

Una de las primeras acciones a realizar sobre cualquier dispositivo consiste en asegurar su acceso, de tal manera que tan solo el personal autorizado pueda ingresar en el mismo. En este aspecto, IOS ofrece diferentes métodos de autenticación,

resultando el más básico de ellos mediante password local, el cual consiste en generar contraseñas que serán almacenadas en el propio dispositivo y requeridas para acceder a la CLI o al modo privilegiado.

Como se ha mencionado en párrafos anteriores, dicho acceso puede llevarse a cabo de manera local a través del puerto de consola, o remotamente, para lo cual resultan necesarias las líneas vty. Por lo tanto, se deberá configurar una contraseña para cada caso, la cual será solicitada cada vez que se establezca una conexión.

En cuanto al modo privilegiado, el password configurado será requerido tras la ejecución del comando *enable*.

La configuración de este tipo de autenticación consta de las siguientes acciones:

Tipo de acceso	**Configuración**

Consola (Local)	`Switch(config)#line console 0`
	`Switch(config-line)#password "contraseña"`
	`Switch(config-line)#login`

Líneas VTY (Remoto)	`Switch(config)#line vty 0 15`
	`Switch(config-line)#password "contraseña"`
	`Switch(config-line)#login`

Modo privilegiado	`Switch(config)#enable password "contraseña"`

Ejemplo: Configurar el switch de tal manera que:

- La autenticación para el acceso por consola sea "cisco-console".
- La autenticación para el acceso remoto sea: "cisco-vty".
- La autenticación para acceder al modo privilegiado sea: "cisco-enable".

```
Switch>enable
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.

Switch(config)#line console 0
Switch(config-line)#password cisco-console
Switch(config-line)#login
Switch(config-line)#exit

Switch(config)#line vty 0 5
Switch(config-line)#password cisco-vty
Switch(config-line)#login
Switch(config-line)#exit

Switch(config)#enable password cisco-enable
```

En el ejemplo, "*line vty 0 5*" tiene como significado que se han habilitado 6 líneas de acceso, hecho que permitirá 6 conexiones remotas simultáneas, desde la 0 hasta la 5, las cuales deben efectuarse mediante Telnet o SSH, protocolos que serán analizados en próximos capítulos. El máximo configurable son 16 líneas (0-15).

Con los cambios aplicados, cualquier conexión a la CLI o intento de acceso al modo de privilegiado mostrará la siguiente pantalla:

```
User Access Verification

Password:

Switch>enable
Password:
Switch#
```

MODIFICAR EL NOMBRE DEL DISPOSITIVO

Aunque su cambio no influya para nada sobre el modo de operar del dispositivo, sí que resulta conveniente modificarlo con el fin de que pueda ser identificado con facilidad. Los switchs Cisco incluyen por defecto el nombre "*Switch*" o el modelo de este, siendo recomendable aplicar cualquier otro criterio, como el departamento donde está ubicado o la función que desarrolla en la red.

Para ello se debe ejecutar la sentencia **hostname "*nombre*"**, desde el modo de configuración global. Por ejemplo, cambiar el nombre del Switch ubicado en el departamento de ventas...

```
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname VENTAS
VENTAS(config)#
```

COMANDOS SHOW Y DEBUG

IOS, además de incorporar infinidad de comandos de configuración, también ofrece otros con la función de verificación y análisis. Las dos opciones por excelencia para ello son *show* y *debug*.

Show tiene por objeto mostrar en pantalla configuraciones, características y estado de cualquier protocolo o función disponible en el dispositivo. Permite una gran variedad de parámetros que irán siendo analizados a lo largo de los diferentes capítulos, ya que para cada configuración existe un comando *show* de verificación.

Mientras, *debug* permite analizar en tiempo real el tráfico que atraviesa el switch, es decir, lo monitoriza. Resulta de gran ayuda ante la resolución de cualquier tipo de

incidencia, sin embargo, se recomienda ejecutarlo con precaución y tan solo de forma puntual, ya que consume bastantes recursos de hardware y puede ocasionar una notable bajada de rendimiento. Una vez ejecutado, la manera de detenerlo es mediante el comando **undebug all**.

Ficheros de configuración en IOS

La secuencia de arranque de un switch Cisco concluye con la carga del fichero *startup-config,* almacenado en la memoria NVRAM del dispositivo y el cual aplica la configuración necesaria al sistema. Sin embargo, IOS también opera con otros tipos de memoria, cada una de ellas con diferentes funciones, siendo las siguientes:

- RAM: también llamada DRAM *(Dynamic Random Access Memory.)* Es la memoria que utiliza el sistema operativo para almacenar cargas de trabajo durante su ejecución. Es volátil, es decir, cuando el switch se apaga o reinicia, su contenido es eliminado.
- ROM: *(Read Only Memory.)* Como su nombre indica, es una memoria de solo lectura utilizada para almacenar el bootstrap, cuya función consiste en buscar la imagen IOS en el dispositivo, copiarla en la memoria RAM y ejecutarla posteriormente. A la ROM solo se accede cuando se inicia el switch.
- Flash Memory: es utilizada para almacenar la IOS y cualquier otro tipo de ficheros de manera permanente. Puede presentarse en dos formatos, mediante un chip integrado en el hardware o bien mediante tarjeta de memoria externa.
- NVRAM: *(Nonvolatile Ram.)* Es una memoria permanente que almacena el fichero de configuración y a la que solo se accede cuando el dispositivo ejecuta la secuencia de arranque.

Donde cada una de ellas almacena la siguiente información:

RAM	FLASH	ROM	NVRAM
Configuración en ejecución (running-config) y procesos.	Cisco IOS Software	Programa Bootstrap	Fichero de configuración inicial (startup-config)

Como se puede observar, Cisco hace uso de dos ficheros de configuración, denominados *startup-config* y *running-config*. El *startup-config* es almacenado en la memoria NVRAM y ejecutado durante el proceso de arranque del dispositivo,

aplicando al sistema operativo la configuración incluida en él. Tras ello, no vuelve a ser utilizado hasta el próximo reinicio. Mientras, el *running-config* es el fichero que se utiliza mientras IOS está en ejecución, es decir, todos los cambios que se apliquen a la configuración serán almacenados en él. Está ubicado en la memoria RAM.

Entonces, si los cambios de configuración son guardados en el *running-config* y este a su vez es almacenado en la memoria RAM, ¿qué ocurre si el switch se reinicia? Una de las características de la RAM es su volatilidad, por lo tanto, su contenido sería eliminado. Además, ante tal reinicio, el fichero cargado es el *startup-config*, por lo que se puede concluir que las modificaciones efectuadas con anterioridad se perderían. Evidentemente, este hecho resulta inaceptable en cualquier tipo de red. La solución a ello consiste en realizar una copia del fichero *running-config* al *startup-config* cada vez que se aplique algún cambio. Para ello se debe ejecutar la sentencia **copy running-config startup-config** desde el modo privilegiado:

```
VENTAS# copy running-config startup-config
```

Una vez hecho, los cambios realizados se almacenan en el *startup* y con ello en la NVRAM, de tal manera que si el switch se reinicia volvería a aplicar las modificaciones que se hubieran aplicado.

Un proceso más detallado de cómo el switch maneja estos dos ficheros podría ser el siguiente:

- *Paso 1: e*l switch inicia y carga el *startup-config* desde la memoria NVRAM.
- *Paso 2:* su contenido es copiado al *running-config,* en la RAM, con el cual trabajará IOS durante su ejecución.
- *Paso 3:* cualquier cambio de configuración efectuado será almacenado en el *running-config*.
- *Paso 4:* para que dichas modificaciones no se pierdan al reiniciar el switch es necesario ejecutar el comando "*copy running-config startup-config"*.

Los ficheros de configuración son considerados elementos críticos, por lo que resulta altamente recomendable crear backups de los mismos. Una manera bastante sencilla de proceder a ello consiste en copiarlos en algún medio externo, como un servidor TFTP, para lo cual bastará con acceder al modo privilegiado y ejecutar la sentencia **copy *[fichero]* tftp.** Tras ello, IOS solicitará la IP del servidor en cuestión y realizará la transferencia.

Si por el contrario fuera necesario importar alguna copia de seguridad desde el TFTP hacia el Switch, se puede llevar a cabo ejecutando el comando **copy tftp *[fichero],*** también desde el modo privilegiado e indicando la IP del servidor.

```
VENTAS# copy startup-config tftp
Address or name of remote host []?
VENTAS# copy tftp startup-config
Address or name of remote host []?
```

Por último, si el *startup-config* fuera eliminado por cualquier motivo o el switch no fuera capaz de ubicarlo durante la carga inicial, IOS creará uno nuevo con la configuración por defecto de fábrica (muy básica).

Para eliminar ficheros manualmente se puede ejecutar el comando **erase *[tipo de memoria]:[fichero]***, desde el modo enable, por ejemplo:

```
VENTAS#erase nvram:startup-config
```

Dependiendo del modelo de switch o versión de IOS es posible que no permita indicar el tipo de memoria, en estos casos bastará con aplicar la sentencia *erase [fichero].*

CONTENIDO DE LOS FICHEROS DE CONFIGURACIÓN

En Cisco, un fichero de configuración simplemente está compuesto por comandos. De esta manera, cuando IOS lo carga, tan solo se limita a ejecutar las sentencias incluidas en él una a una y de manera secuencial hasta que todas han sido aplicadas.

Para visualizar su contenido se debe ejecutar un **show running-config** o **show startup-config**, dependiendo del fichero deseado.

Con las modificaciones realizadas a lo largo del capítulo, el fichero *running-config* incluiría todos los comandos ejecutados más la configuración por defecto incluida en el dispositivo, de tal forma que:

```
VENTAS#show running-config
Building configuration...

Current configuration : 1091 bytes
!
version 12.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
```

```
hostname VENTAS
!
enable password cisco-enable
!
!
spanning-tree mode pvst
!
interface FastEthernet0/1
!
interface FastEthernet0/2
!
interface FastEthernet0/3
!
!
...Resultado omitido por brevedad...
!
interface Vlan1
 no ip address
 shutdown
!
!
line con 0
 password cisco-console
 login
!
line vty 0 5
 password cisco-vty
 login
line vty 6 15
 login
!
!
end
```

VERSIÓN DE IOS

La versión de IOS varía dependiendo del modelo de Switch y sus características. Para comprobar con cuál de ellas opera el dispositivo se debe hacer uso del comando **show version**, el cual también muestra información de hardware. Los datos más relevantes que proporciona son:

- Versión de IOS.
- Tiempo que el switch lleva operativo.
- Número de interfaces Ethernet.
- Modelo del dispositivo.

CDP (Cisco Discovery Protocol)

Un protocolo que puede resultar de bastante utilidad en determinadas ocasiones es CDP, propietario de Cisco y encargado de facilitar información básica sobre dispositivos vecinos sin necesidad de conocer el password de estos. La manera de

lograrlo es sencilla, tras ejecutar el comando **show cdp neighbors** se inicia un envío de mensajes propios del protocolo a través de todas las interfaces, los cuales serán respondidos por aquellos vecinos que también hagan uso de CDP, proporcionando los siguientes datos:

- Identificador: Normalmente el nombre de host.
- Tipo de dispositivo.
- Port ID: La interfaz local que conecta con el dispositivo en cuestión.
- Plataforma: El modelo y sistema operativo que ejecuta actualmente.
- Tipo de enlace: Duplex o Full duplex.

CDP es habilitado por defecto en IOS, siendo los comandos necesarios para obtener información los siguientes, todos ellos ejecutados desde el modo usuario o privilegiado:

- show cdp neighbors: Muestra un resumen de todos los dispositivos Cisco vecinos.
- show cdp neighbors detail: Muestra información más detallada sobre cada uno de ellos.
- show cdp entry *[nombre]*: Muestra la misma información que *show cdp neighbors detail,* pero tan solo para un vecino, definido en el parámetro *[nombre]*.

Fig. 2-7 CDP. Información entre vecinos.

```
SW1# show cdp neighbors
Device ID    Local Intrfce    Capability   Platform    Port ID
SW2          Fas 0/1               S       2950        Fas 0/10
```

SW1 ejecuta un *show cdp neighbor,* y SW2, que es un vecino directamente conectado, responde a los mensajes facilitando datos sobre sí mismo, como su nombre (SW2), interfaz local por la cual recibió la solicitud (Fas 0/10), plataforma (2950) y dispositivo (switch, representado con el código "S"). Si fuera necesario información más detallada, como la versión de IOS, se debe optar por ejecutar un *show cdp neighbors detail.*

CDP proporciona información bastante útil sobre dispositivos de red sin necesidad de ningún tipo de autenticación, hecho que puede suponer un riesgo de seguridad. Es por ello que Cisco recomienda deshabilitarlo con el fin de evitar posibles ataques. Para ello existen dos opciones, a nivel de interfaz con el comando **no cdp enable**, o a nivel global con el comando **no cdp run.** Una buena práctica es hacer uso de esta última opción y en el caso de requerirlo, habilitarlo de manera puntual con el comando **cdp run.**

CDP es propietario de Cisco, por lo tanto, su aplicación se limita a dispositivos de este fabricante.

LLDP (Link Layer Discovery Protocol)

Otro de los protocolos disponibles en capa 2 para obtener información de vecinos directamente conectados es LLDP *(Link Layer Discovery Protocol),* cuyas funciones resultan idénticas a las ofrecidas por CDP, tanto es así, que la mayor diferencia entre ambos radica en que este último es propietario de Cisco, mientras que LLDP hace referencia al estándar abierto IEEE 802.11ab.

En IOS, su configuración se lleva a cabo a través del comando **lldp run,** el cual habilita el protocolo de manera global, o mediante **lldp transmit** y/o **lldp receive** para habilitarlo tan solo en aquellas interfaces deseadas (desde el modo de configuración de las mismas), en cuyo caso se dispone la posibilidad de transmitir información, recibirla o ambos modos.

En cuanto a verificación y obtención de datos de vecinos, los comandos disponibles en dispositivos Cisco son:

- show lldp interface *[interfaz]*: Muestra el estado del protocolo sobre una determinada interfaz y estadísticas del mismo en caso de encontrarse habilitado.
- show lldp neighbors: Genera un listado de los vecinos directamente conectados de los cuales se ha obtenido información mediante LLDP, facilitando datos como su nombre, tipo de dispositivo o interfaz local y remota que establecen el enlace.
- show lldp entry *[nombre]*: Muestra información específica y detallada sobre un determinado vecino, definido mediante el parámetro *nombre*, el cual es obtenido previamente mediante un *"show lldp neighbors".*

CONFIGURACIÓN DE SWITCHS

En párrafos anteriores han sido analizados y configurados los aspectos más básicos del switch, dedicando especial atención al acceso a la CLI y sus ficheros, pero sin profundizar en el resto de servicios. IOS incluye multitud de comandos y opciones para adecuar cada switch a las necesidades de cada red, siendo nosotros, los administradores, los encargados de dicha tarea.

La siguiente sección se dedicará al estudio y aplicación de las configuraciones más comunes en estos dispositivos, dividiéndola en tres apartados con el fin de facilitar la comprensión: Asegurar el acceso a la CLI, Configuración de Interfaces y VLANs.

Asegurar el acceso a la CLI

El comando "password", ya analizado anteriormente, habilita la autenticación de acceso al dispositivo en relación con contraseñas almacenadas localmente. Sin embargo, esta opción por sí sola representa un nivel de seguridad bastante bajo y por lo tanto poco recomendable sobre entornos corporativos.

IOS ofrece métodos más seguros tanto para el acceso local como remoto, siendo los siguientes.

AUTENTICACIÓN MEDIANTE CONTRASEÑA

Uno de los mayores inconvenientes de **password** consiste en que la contraseña es almacenada en texto-plano en los ficheros de configuración. Para comprobarlo, simplemente bastará con ejecutar un *show running-config*, donde serán mostradas totalmente legibles. Imagina que dichos ficheros son guardados en un servidor TFTP o cualquier otro medio externo como copia de seguridad, cualquiera que tenga acceso a ellos podrá obtener acceso al dispositivo.

Una manera de solucionarlo es aplicando cifrado sobre la contraseña, ejecutando para ello el comando **enable secret *"contraseña"*** desde el modo de configuración global. La función es exactamente la misma que con *enable password*, es decir, habilita la autenticación para acceder al modo privilegiado, sin embargo, la gran diferencia entre ambos radica en que ahora la contraseña no será legible en los ficheros de configuración.

Para llevar a cabo el cambio, primero se debe eliminar el password anteriormente creado, ejecutando un **no enable password,** para luego introducir la nueva configuración:

```
VENTAS#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
VENTAS(config)#no enable password
VENTAS(config)#enable secret cisco-enable
VENTAS(config)#exit
VENTAS#
%SYS-5-CONFIG_I: Configured from console by console

VENTAS# show running-config
Building configuration...

Current configuration : 1109 bytes
!
version 12.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname VENTAS
!
enable secret 5 $1$mERr$fDuZEPfRLI/aWvzEgceCS1
!
...Resultado omitido por brevedad...
!
interface Vlan1
 no ip address
 shutdown
!
!
line con 0
 password cisco-console
 login
!
line vty 0 5
 password cisco-vty
 login
line vty 6 15
 login
!
!
end
```

Ahora, la contraseña para acceder al modo privilegiado ha sido cifrada. El 5 que aparece a continuación del comando *enable secret* indica que el método utilizado para ello es MD5.

Sin embargo, aquellas contraseñas necesarias para el acceso por consola y líneas vty aún permanecen visibles y por lo tanto con un nivel de seguridad bajo. Para cifrar tanto estas como cualquier otra que fuera creada resulta necesario ejecutar un servicio, el cual se encargará de ello automáticamente y que por defecto está deshabilitado. Su activación se lleva a cabo mediante el comando **service password-encryption** desde el modo de configuración global, siendo sus características las siguientes:

- Cuando es ejecutado, IOS cifra de manera inmediata todas las contraseñas existentes en el fichero *running-config*.
- Mientras el servicio se mantenga activo, todas las contraseñas creadas serán almacenadas con cifrado.
- Para deshabilitarlo resulta necesario introducir la sentencia **no service password-encryption.** En este caso, las contraseñas que fueron cifradas permanecerán igual, pero las nuevas serán almacenadas en texto-plano.

El objetivo consiste en dotar al switch de la máxima seguridad posible, por lo tanto, habilitar dicho servicio es considerada una buena práctica…

```
VENTAS(config)#service password-encryption
VENTAS(config)#exit
VENTAS#show running-config
Building configuration...

...Resultado omitido por brevedad...

service password-encryption
!
hostname VENTAS
!
enable secret 5 $1$mERr$fDuZEPfRLI/aWvzEgceCS1
!
line con 0
 password 7 0822455D0A1648141D051F0B262E
 login
!
line vty 0 4
 password 7 0822455D0A1648010612
 login
line vty 5
 password 7 0822455D0A1648010612
 login
```

AUTENTICACIÓN MEDIANTE USUARIO Y CONTRASEÑA

El acceso por contraseña analizado hasta ahora tiene un punto desfavorable, y es que esta debe ser compartida y por lo tanto conocida por todos los miembros que administren el dispositivo. Este hecho, sobre entornos pequeños, donde tan solo deben acceder uno o dos administradores puede resultar una buena opción. Sin embargo, no lo es tanto en redes de mayor tamaño, donde la administración es llevada a cabo por numerosos miembros y consultoras externas.

IOS ofrece otra modalidad basada en autenticación por usuario, donde cada uno de ellos dispondrá de sus propias credenciales, eliminando así el problema de la contraseña compartida. Este método puede ser configurado de dos maneras, localmente o a través de servidores AAA (*Authentication, Authorization and Accounting).*

Cuando se opta por la primera de ellas, el Switch genera una base de datos en el propio dispositivo que almacena tanto los usuarios como las contraseñas definidas. Su configuración consta de los siguientes pasos:

- *Paso 1:* Crear los usuarios necesarios con el comando **username *[nombre]* password *[contraseña]*** o **username *[nombre]* secret *[contraseña]*** desde el modo de configuración global. La diferencia entre ambos radica en que *secret* la almacena con cifrado SHA-256 mientras que *password* lo hace en texto-plano.
- *Paso 2*: Habilitar la autenticación sobre la conexión de consola y líneas vty, aplicando el comando **login local** desde el modo de configuración de cada una de ellas.

```
VENTAS(config)#username prueba secret Tenerife
VENTAS(config)#username prueba1 secret LasPalmas
VENTAS(config)#line console 0
VENTAS(config-line)#login local
VENTAS(config-line)#exit
VENTAS(config)#line vty 0 15
VENTAS(config-line)#login local
VENTAS(config-line)#exit
```

Se han creado los usuarios “prueba” con contraseña “Tenerife” y “prueba1” con contraseña “LasPalmas” y a su vez se han configurado los accesos por consola y remoto con autenticación local. La próxima vez que se acceda al dispositivo se mostrará la siguiente pantalla:

```
User Access Verification

Username: prueba
Password:

VENTAS>enable
Password:
VENTAS#
```

La contraseña para acceder al modo privilegiado continúa siendo la configurada anteriormente con el comando *enable secret [password]*.

El otro sistema disponible de autenticación basado en usuario y contraseña consiste en la implementación de un servidor AAA, cuya misión, entre otras, consiste en crear una base de datos centralizada que será utilizada por los dispositivos de red para validar a sus usuarios. El modo de operar se basa en dos protocolos, RADIUS o TACACS+, ambos con características propias, pero con la misma finalidad. El proceso llevado a cabo es el siguiente:

- *Paso 1:* El usuario intenta acceder al switch con usuario y contraseña.
- *Paso 2:* Las credenciales introducidas son enviadas al servidor AAA para validarlas.
- *Paso 3:* Este comprueba su base de datos y responde al switch, permitiendo o denegando el acceso de dicho usuario.
- *Paso 4:* Si las credenciales son válidas, se autoriza automáticamente el acceso a la CLI, si por el contrario no lo son, se mostrará un mensaje en pantalla denegando el acceso.

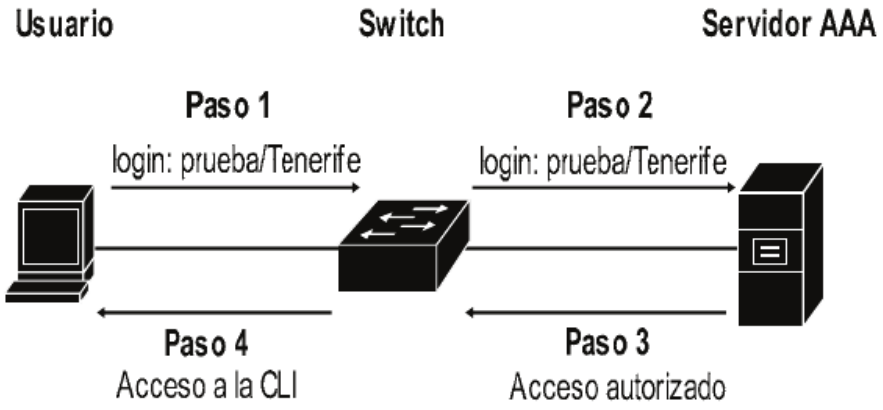


Fig. 2-8 Autenticación en servidores AAA.

Ambos métodos, tanto el local como el basado en AAA, consisten en el mismo tipo de autenticación; sin embargo, la gran ventaja de los servidores es que crean una base de datos centralizada, lo que lo convierte en la opción más escalable y sencilla de aplicar. Imagina una topología con 100 dispositivos de red y 20 usuarios a configurar. Si se implementara el método local se tendrían que agregar los 20 usuarios en cada uno de ellos, sin embargo, configurando AAA bastaría con crearlos una sola vez en el servidor. Además, también incorpora opciones adicionales como la autorización, la cual permite definir diferentes privilegios para cada usuario, y la auditoría, utilizada para realizar un seguimiento de cada sesión y registrar eventos. Sin duda, el método más recomendable es el basado en AAA.

Configuración de RADIUS

El procedimiento para configurar la autenticación mediante RADIUS en un dispositivo Cisco consta de los siguientes pasos, todos ellos ejecutados desde el modo de configuración global (a excepción del 5):

- *Paso 1:* Habilitar aaa con el comando **aaa new-model**.

- *Paso 2:* Definir RADIUS como primer método de autenticación, ejecutando **aaa authentication login default group radius**.
- *Paso 3:* Identificar el servidor RADIUS a través de la sentencia **radius-server host *[dir.IP.servidor]***.
- *Paso 4* (Opcional): Si el servidor requiere validación para acceder a su base de datos de usuarios, configurarla con el comando **radius-server key *[clave]***.
- *Paso 5:* Habilitar la autenticación tanto en la consola como en las líneas vty, utilizando para ello **login authentication default**, desde el modo de configuración de la línea en cuestión.

Ejemplo: Aplicar RADIUS para los accesos por consola y remoto al switch VENTAS. El servidor hace uso de la IP 172.18.10.10 y requiere la clave "AccesoServerAuth" como validación para acceder a su base de datos.

```
VENTAS(config)#aaa new-model
VENTAS(config)#aaa authentication login default group radius
VENTAS(config)#radius-server host 172.18.10.10
VENTAS(config)#radius-server key AccesoServerAuth
VENTAS(config)#line vty 0 15
VENTAS(config-line)#login authentication default
VENTAS(config-line)#exit
VENTAS(config)#line console 0
VENTAS(config-line)#login authentication default
VENTAS(config-line)#exit
```

En el paso 2, el comando "*aaa authentication login default group radius*" realmente crea un listado de opciones de autenticación, en este caso denominado "default" y el cual tan solo incluye la opción por RADIUS (*group radius*). Sin embargo, pueden habilitarse diferentes métodos, ejecutados en orden secuencial, siempre y cuando los anteriores fallen. Por ejemplo, la siguiente sentencia:

```
VENTAS(config)#aaa authentication login default group radius local
```

Genera un listado de autenticación con nombre "default", el cual incluye dos opciones de acceso, RADIUS (*group radius*) y local (*local*). Ello significa que cuando algún usuario intente acceder al dispositivo, primero se aplicará la autenticación por RADIUS, y si esta falla, la local. La segunda opción solo es utilizada como respaldo en caso de que la primera falle. Es decir, si el servidor RADIUS se ha caído o no es posible la comunicación con él, el dispositivo intentará comprobar las credenciales del usuario basándose en autenticación local. Sin embargo, si la primera opción está operativa pero las credenciales del usuario son incorrectas y por lo tanto se deniega su acceso, el dispositivo no hará uso de la segunda opción.

Configuración de TACACS+

Para habilitar tacacs+ como método de autenticación bastará con llevar a cabo el siguiente procedimiento, ejecutado desde el modo de configuración global (a excepción del paso 4):

- *Paso 1:* Habilitar aaa con el comando **aaa new-model**.
- *Paso 2:* Definir TACACS+ como primer método de autenticación, ejecutando **aaa authentication login default group tacacs+**.
- *Paso 3:* Identificar el servidor TACACS+ a través de la sentencia **tacacs-server host *[dir.IP.servidor]* key *[clave]***.
- *Paso 4:* Habilitar la autenticación tanto en la consola como en las líneas vty, utilizando para ello **login authentication default**, desde el modo de configuración de la línea en cuestión.

Ejemplo: Aplicar TACACS+ como método de autenticación sobre las conexiones vía consola y remotas hacia el switch VENTAS. El servidor hace uso de la IP 172.18.10.250 y requiere la clave "AccesoAuth" como validación para acceder a su base de datos. Además, si dicho servidor no estuviera disponible, la autenticación se deberá llevar a cabo de manera local.

```
VENTAS(config)#aaa new-model
VENTAS(config)#aaa authentication login default group tacacs+ local
VENTAS(config)#tacacs-server host 172.18.10.250 key AccesoAuth
VENTAS(config)#line vty 0 15
VENTAS(config-line)#login authentication default
VENTAS(config-line)#exit
VENTAS(config)#line console 0
VENTAS(config-line)#login authentication default
VENTAS(config-line)#exit
```

APLICACIÓN DE SSH EN LUGAR DE TELNET

Por defecto, Cisco habilita Telnet como protocolo para la conexión remota a través de las líneas vty. Este método es totalmente operativo pero considerado inseguro, ya que la comunicación entre ambos extremos es transmitida en texto plano. Actualmente existen opciones más avanzadas y por lo tanto más recomendables, las cuales cifran el tráfico generado entre emisor y receptor. Una de ellas, SSH (*Secure Shell*), es considerado el protocolo más seguro para este propósito. Para habilitarlo se deben llevar a cabo las siguientes acciones:

- *Paso 1:* Definir el nombre del dispositivo con el comando **hostname *[nombre]***.
- *Paso 2:* Configurar un nombre de dominio con el comando **ip domain-name *[nombre dominio]*** desde el modo de configuración global.

- *Paso 3:* Crear las claves asimétricas, las cuales serán utilizadas entre emisor y receptor para cifrar y descifrar la comunicación, son generadas con el comando **crypto key generate rsa** desde el modo de configuración global. El proceso solicita un tamaño de clave (en bits), que cuanto mayor sea más fuerte será el cifrado, pero también más lenta la comunicación. El valor más común es 1024 bits.
- *Paso 4:* Generada la clave, aplicar SSH a las líneas vty, con el comando **transport input ssh** desde el modo de configuración de la propia línea.
- *Paso 5* (Opcional): Habilitar la versión 2 del protocolo con el comando **ip ssh version 2** desde el modo de configuración global. Esta ofrece mejores características de seguridad que su antecesora. Si no se lleva a cabo, el switch hará uso de la versión 1.

```
Switch(config)#hostname VENTAS
VENTAS(config)#ip domain-name cert-ccna
VENTAS(config)#crypto key generate rsa

The name for the keys will be: VENTAS.cert-ccna
Choose the size of the key modulus in the range of 360 to 2048 for your
  General Purpose Keys. Choosing a key modulus greater than 512 may take
  a few minutes.

How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

VENTAS(config)#line vty 0 15
VENTAS(config-line)#transport input ssh
VENTAS(config-line)#exit
VENTAS(config)#ip ssh version 2
```

TIEMPO DE INACTIVIDAD

La inactividad se refiere al tiempo que un usuario permanece conectado a la CLI sin enviar ninguna señal de teclado. Por defecto IOS realiza una desconexión automática pasados 5 minutos de inactividad, tiempo que puede resultar excesivo en algunos entornos. Para modificar su valor bastará con aplicar el comando **exec-timeout *[minutos] [segundos],*** desde el modo de configuración de la línea.

Ejemplo: Desconectar automáticamente a los usuarios remotos pasados 1 minuto y 30 segundos de inactividad.

```
VENTAS(config)#line vty 0 15
VENTAS(config-line)# exec-timeout 1 30
```

CONFIGURACIÓN DE BANNERS

Los banners son mensajes que serán mostrados al iniciar sesión o al conectar a la CLI. No representan ninguna función de seguridad, pero suelen ser configurados para advertir sobre el uso y acceso al dispositivo o simplemente para informar de determinados eventos. Realmente el contenido depende por completo del administrador.

Existen 3 tipos: Mensaje del día (MOTD), mensaje de login y mensaje EXEC.

- El mensaje del día (MOTD) es mostrado cuando el usuario conecta a la CLI. Su función suele ser para publicar mensajes temporales que pueden cambiar con el tiempo, por ejemplo "Switch actualmente en mantenimiento". Se aplica con el comando **banner motd *[delimitador][mensaje][delimitador]***, desde el modo de configuración global.
- El mensaje de login es mostrado después del MOTD y justo antes de la pantalla de autenticación. Su función más común suele ser para publicar mensajes permanentes como "Acceso autorizado solo para los administradores de esta compañía" o similares. Es definido haciendo uso de la sentencia **banner login *[delimitador][mensaje][delimitador]***, desde el modo de configuración global.
- Por último, el mensaje EXEC es mostrado después de que un usuario se autentique con éxito en la CLI y suele informar de determinados eventos, por ejemplo "Nuevo link creado entre SW1 y SW2 en la VLAN x. No modificar". Se aplica con el comando **banner exec *[delimitador][mensaje][delimitador]***, desde el modo de configuración global.

El delimitador es un carácter cualquiera introducido para marcar el inicio y fin del mensaje. En el ejemplo se hará uso de "#".

```
VENTAS(config)#banner motd #
Enter TEXT message.  End with the character '#'.
Mantenimiento de Switch previsto para las 11:00AM#

VENTAS(config)#banner login #
Enter TEXT message.  End with the character '#'.
Acceso solo personal autorizado#

VENTAS(config)#banner exec #
Enter TEXT message.  End with the character '#'.
Cambios en el link con SW2. VLAN 5 creada. No modificar#
```

Lo que dará como resultado…

```
Mantenimiento de Switch previsto para las 11:00AM
```

```
Acceso solo personal autorizado

User Access Verification

Username: prueba
Password:

Cambios en el link con SW2. VLAN 5 creada. No modificar

VENTAS>
```

Todas las configuraciones llevadas a cabo hasta ahora pueden ser aplicadas tanto en switchs como routers Cisco. Las analizadas en las próximas secciones de este mismo capítulo son exclusivas de capa 2, por lo tanto, solo disponibles en switchs.

Configuración de interfaces

Las interfaces del switch representan el elemento físico más importante de este, gracias a las cuales los diferentes dispositivos conectados a ellas formarán parte de la misma LAN. Su funcionamiento y su modo de operar dependerán tanto de la configuración establecida como del propio hardware del dispositivo. En cuanto a la configuración, existen multitud de opciones que definirán el rendimiento y funciones de cada una de ellas, mientras que el hardware establece la capacidad máxima de transmisión real permitida y tipo de medio físico. Este último mayormente será cableado UTP, aunque también se suelen incluir interfaces de fibra.

A lo largo de los siguientes párrafos serán analizados los aspectos más importantes de su configuración, seguridad y segmentación la red gracias a las VLANs.

CONFIGURACIÓN DE IP PARA ACCESO REMOTO

Del acceso remoto a la CLI han sido analizados diferentes aspectos a lo largo del capítulo, como que se realiza a través de las líneas vty, que se pueden configurar hasta un máximo de 16 conexiones simultáneas, los diferentes tipos de autenticación y los protocolos disponibles para llevarla a cabo. Sin embargo, para poder establecer esta conexión resulta imprescindible que el switch sea configurado con una dirección IP, también denominada IP de gestión y utilizada por el dispositivo solo para fines administrativos, nunca para enrutamiento ni tareas de capa 3.

Para ello es necesario la creación de una interfaz virtual, denominada *Switched Virtual Interface (SVI)* o *VLAN interface*, la cual será configurada con una dirección IP, máscara de red y puerta de enlace, siendo sus funciones las mismas que las llevadas a cabo por una tarjeta de red (NIC) en cualquier PC.

¿Por qué es necesario crear una interfaz virtual y no es configurado directamente en una física? Porque las interfaces físicas de un switch solo operan en capa 2.

Los pasos para definir una IP de gestión son:

- *Paso 1:* Crear la interfaz virtual con el comando **interface vlan *[num de vlan]*** desde el modo de configuración global.
- *Paso 2:* Asignarle una dirección IP con el comando **ip address *[dir ip]* *[máscara]*** desde el modo de configuración de la interfaz virtual.
- *Paso 3:* Habilitarla ejecutando un **no shutdown**.
- *Paso 4:* Definir la puerta de enlace a través de la sentencia **ip default-gateway *[dir ip]*** desde el modo de configuración global. Esta IP debe ser la del dispositivo encargado del enrutamiento de la LAN (normalmente un router).
- *Paso 5* (Opcional): Definir los DNS con el comando **ip name-server *[dir ip1] [dir ip2]…***, desde el modo de configuración global.

```
VENTAS(config)#interface vlan 5
VENTAS(config-if)#ip address 192.168.5.100 255.255.255.0
VENTAS(config-if)#no shutdown
VENTAS(config-if)#exit
VENTAS(config)#ip default-gateway 192.168.5.1
```

Se ha creado una interfaz virtual con IP 192.168.5.100. Las conexiones remotas mediante Telnet o SSH a este switch deberán efectuarse a dicha dirección.

CONFIGURACIÓN BÁSICA DE INTERFACES

Las interfaces de un switch identifican a cada una de las conexiones físicas de este. IOS permite modificar sus características para adecuarlas a las necesidades de la red, permitiendo habilitarlas o deshabilitarlas, modificar su modo de transferencia, velocidad, descripción y por supuesto asegurarlas. Para todo ello primero hay que acceder a su modo de configuración, a través del comando **interface *[número interfaz]***. El número de interfaz es visible físicamente en cada puerto del Switch.

Aquellas deshabilitadas no generan ningún tipo de tráfico ni permiten establecer comunicación con el switch, aunque el cable esté físicamente conectado. Por contra, las habilitadas se encuentran totalmente operativas y pueden ser utilizadas para la conexión de cualquier dispositivo a la LAN. Por defecto, todas se encuentran en este estado. Para deshabilitarlas es necesario ejecutar un **shutdown,** y para habilitarlas, **no shutdown**.

El modo de transferencia se refiere al tipo de transmisión que se aplicará en el enlace, pudiendo ser *full-duplex*, *half-duplex* o *auto* (negociada). De todas ellas, la

opción por defecto es la negociación, pero puede ser definido de manera manual con el comando **duplex *[full | half | auto]***.

La velocidad se refiere a la capacidad máxima de transmisión definida en la interfaz. Al igual que en el modo de transferencia, la opción por defecto es la negociación automática con el otro extremo del enlace, pero también es posible ajustarla manualmente con el comando **speed *[10 | 100 | 1000 | auto]***. Las opciones varían dependiendo del switch, por ejemplo, si la capacidad máxima del hardware es de 100 Mbps, las opciones disponibles serán *10*|*100*|*auto.*

Por último, también es posible configurar una descripción para cada interfaz. Esta no causa ningún impacto sobre el modo de operar, simplemente sirve como ayuda administrativa para identificar enlaces. Por ejemplo, aquellas que conectan con dispositivos finales podrían ser descritas como "Conexión para PCs de usuarios". Su configuración se lleva a cabo mediante la sentencia **description *[descripción]***.

Ejemplo: Configurar el switch VENTAS de tal manera que:

- La interfaz Fa0/24, que conecta con el router central de la compañía opere a una velocidad de 100 mb, *full-duplex* e identificarla.
- El rango de interfaces 0/1 - 0/10, que conectan con dispositivos finales, deben ser configurados con negociación automática tanto en velocidad como modo de transmisión. También resulta necesaria su identificación.
- El rango de interfaces 0/11 - 0/23 no se encuentran en uso, por lo tanto, deben ser deshabilitadas.

```
VENTAS(config)#interface Fa0/24
VENTAS(config-if)#speed 100
VENTAS(config-if)#duplex full
VENTAS(config-if)#description Enlace con Router Central
VENTAS(config-if)#exit
VENTAS(config)#interface range fa0/1 - 10
VENTAS(config-if-range)#description Puertos destinados a dispositivos
finales
VENTAS(config-if-range)#exit
VENTAS(config)#interface range fa0/11 - 23
VENTAS(config-if-range)#shutdown
```

Las características de la interfaz 24 son modificadas manualmente para que cumpla con los requisitos, sin embargo, el rango de la 1 a la 10 no requiere configuración ya que por defecto se aplicará la negociación automática, tan solo es necesaria la descripción. El resto de interfaces son deshabilitadas.

El comando **interface range *[Interfaz inicial - Interfaz final]*** identifica un conjunto de ellas, de tal manera que la configuración llevada a cabo será aplicada de manera individual en cada una, ahorrando con ello bastante tiempo administrativo.

Estado de las interfaces

Una vez finalizada la configuración y establecido el enlace físico entre ambos extremos se deberá verificar que este opera con normalidad. Esta tarea es llevada a cabo comprobando el estado de las interfaces a través del comando **show ip interface brief**, desde el modo privilegiado. Gracias a la información obtenida se determinará su correcto funcionamiento, o en su defecto, servirá de ayuda para averiguar el motivo de falla y tomar las medidas oportunas para solucionar el problema.

El resultado de dicha sentencia mostrará un listado de todas las interfaces del switch y su estado actual, el cual será definido gracias a la información incluida en las columnas *Status* y *Protocol.*

- *Status* se refiere al estado de la línea (*line status*) y en ella pueden identificarse tres modos: *Administratively down, down* o *up*.
- A su vez, *Protocol* hace referencia al estado de los protocolos utilizados en la interfaz y en ella también se identifican tres modos: *Down, Down (err-disabled)* o *up.*

En relación con la combinación de ambas se obtiene el estado final del enlace, el cual solo será operativo si tanto *Status* como *Protocol* operan en **up/up.** Cualquier otro resultado significa que la interfaz no trabaja con normalidad por alguna razón, ya sea debido a fallos físicos o errores de configuración. En estos casos se debe buscar la solución más eficiente y aplicarla para solventar el problema. Las posibles combinaciones de estado y sus causas son las siguientes:

Status	Protocol	Estado de la interfaz	Causa más común
Administratively down	Down	DESHABILITADA	La interfaz ha sido deshabilitada administrativamente con el comando *shutdown*.
Down	Down	NO CONECTADA	No hay cable conectado en la interfaz o este está dañado. La velocidad de la interfaz no coincide en ambos extremos del enlace. El cable en el otro extremo no está conectado.

Up	Down	NO CONECTADA	Error en capa 2. Ambos extremos con configuraciones diferentes.
Down	Down (err-disabled)	ERR-DISABLED	Interfaz deshabilitada por *port-security* debido a una violación de seguridad.
Up	Up	CONECTADA	Interfaz operativa.

Un ejemplo podría ser el siguiente:

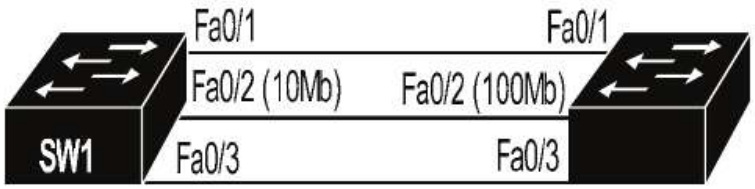


Fig. 2-9 Estado de interfaces.

```
SW1# show ip interface brief
Interface                  IP-Address      OK? Method  Status
Protocol

FastEthernet0/1            unassigned      YES manual  up                    up

FastEthernet0/2            unassigned      YES manual  down                  down

FastEthernet0/3            unassigned      YES manual  administratively down down

FastEthernet0/4            unassigned      YES manual  down                  down

FastEthernet0/5            unassigned      YES manual  down                  down

FastEthernet0/6            unassigned      YES manual  down                  down

FastEthernet0/7            unassigned      YES manual  down                  down

FastEthernet0/8            unassigned      YES manual  down                  down

…Resto de interfaces omitidas por brevedad…
```

- Fa0/1 (*up/up*), la interfaz está habilitada, existe enlace con el otro extremo y opera con normalidad.
- Fa0/2 (*down/down*), el cable está conectado en ambos extremos pero no se establece el enlace. Ello es debido a que una interfaz opera a 10 mb/s y la otra a 100. Se soluciona igualando la velocidad con el comando **speed *[velocidad].***

- Fa0/3 (*administratively down/down*), el cable está conectado en ambos extremos, pero no se establece el enlace debido a que la interfaz ha sido deshabilitada administrativamente con el comando **shutdown**. Para solucionarlo basta con ejecutar un **no shutdown** en ella.
- El resto de interfaces no están cableadas en el Switch, por eso su estado es "*down/down"*.

Otro comando útil para realizar esta misma tarea es **show interfaces status,** sin embargo, su resultado tan solo muestra el estado final del enlace (*not-connected, connected* o *disabled*) omitiendo los detalles de línea y protocolo. A la hora de resolver incidencias se recomienda el uso de *show ip interfaces brief*, ya que proporciona información más detallada.

ASEGURAR LAS INTERFACES

A nivel de red, la seguridad puede y debe implementarse desde la capa 1 hasta la 7, siendo el switch el dispositivo ideal para llevar a cabo esta tarea en capa 2. Estos permiten controlar a qué dispositivos se concederá, o no, acceso a la LAN, basándose para ello en la dirección MAC.

El método llevado a cabo consiste en definir una lista de MACs permitidas, la cual será consultada por el switch cada vez que un dispositivo conecte a través de la interfaz, permitiendo su acceso o ejecutando alguna medida de seguridad sobre ella. Por defecto, una MAC no permitida generará la acción de deshabilitar la interfaz, opción que puede ser modificada manualmente.

La lista puede crearse en relación con tres métodos:

De manera dinámica, donde se establece un número máximo de conexiones permitidas. En este caso la interfaz aprende automáticamente las direcciones MAC de los dispositivos en el mismo orden en que conectan a través de ella. Cuando se supere el máximo configurado se genera una violación de seguridad.

De manera estática, donde se definen manualmente las direcciones que deben ser permitidas en la interfaz. Cualquier conexión de algún dispositivo cuya MAC que no pertenezca a la lista generará una violación de seguridad.

Por último, se puede optar por una combinación de ambos métodos. En este caso se deben configurar algunas direcciones estáticas, mientras que el resto serán aprendidas automáticamente. Por ejemplo, una interfaz con un máximo de 4 MACs permitidas donde tan solo se asigna una manualmente.

Una diferencia importante entre el modelo dinámico y estático radica en la manera de permitir conexiones en ambos casos. El primero establece un límite máximo, pero no tienen por qué ser siempre los mismos dispositivos. Por ejemplo, una interfaz configurada de manera dinámica para un total de dos MACs, una vez permitidas, una tercera conexión generaría una violación de seguridad. Sin embargo, si se desconecta uno y se vuelve a conectar el tercero obtendría acceso ya que el límite continúa en dos conexiones y por lo tanto no se ha violado la seguridad.

Este hecho en el modelo estático es totalmente diferente, ya que las direcciones permitidas son configuradas manualmente y cualquier dispositivo que no coincida con ellas será denegado, aplicando la acción de seguridad correspondiente. Continuando con el ejemplo anterior, una interfaz configurada de manera estática para dos MACs, donde se desconecta uno y en su lugar se conecta otro cuya dirección no coincide con las asignadas… en este caso se genera una violación de seguridad.

Otra de las diferencias a destacar entre ambos modelos es que las MACs estáticas son almacenadas en el fichero de configuración *running-config* mientras que las dinámicas no.

Los pasos para llevar a cabo su configuración son los siguientes, todos ellos ejecutados desde el modo de configuración de la interfaz:

- *Paso 1:* Definir la interfaz en modo acceso o modo *trunk*. El primero indica que el enlace conectará con un dispositivo final como un PC o impresora, mientras que el *trunk* es aplicado en interfaces que conectan directamente con otro switch o router y por el cual es necesario transmitir tráfico de diferentes VLANs (será analizado en párrafos posteriores). Para configurar dichos modos se debe aplicar el comando **switchport mode access** o **switchport mode trunk.**
- *Paso 2:* Habilitar la seguridad de puerto con el comando **switchport port-security**. Al ejecutarlo se aplica por defecto el modo dinámico, con un máximo de una MAC permitida y en caso de violación de seguridad la interfaz será deshabilitada automáticamente.
- *Paso 3* (Opcional): El número máximo de dispositivos permitidos puede ser modificado manualmente a través de la sentencia **switchport port-security maximum *[num máximo].***
- *Paso 4* (Opcional): Configurar la acción a realizar en caso de violación de seguridad con el comando **switchport port-security violation *[protect | restrict | shutdown]***. Donde, ***protect*** simplemente descarta el tráfico del

dispositivo, no genera mensajes de log ni SNMP y tampoco deshabilita la interfaz. ***Restrict,*** además de descartar el tráfico también genera una alerta de log y SNMP al administrador, sin embargo, la interfaz tampoco es deshabilitada. Por último, con ***shutdown*** (opción por defecto) se descarta el tráfico, se genera un log y SNMP y además se deshabilita la interfaz. Cuando esto ocurre pasa a un estado de ***error disabled*** donde es necesaria la intervención del administrador para volver a habilitarla, teniendo primero que apagarla con un **shutdown** para luego volver a activarla con un **no shutdown.**

- *Paso 5* (Opcional): Configurar las direcciones estáticas deseadas gracias al comando **switchport port-security mac-address *[dir mac]***, estas serán almacenadas en el fichero *running-config* y asociadas a la interfaz en cuestión.

En entornos corporativos, con multitud de switchs y dispositivos finales, optar por la configuración estática puede resultar una tarea bastante tediosa y prácticamente interminable. Para poder llevarlo a cabo de una manera más sencilla, IOS incorpora la opción **switchport port-security mac-address sticky**, que ejecutada desde el modo de configuración de la interfaz aprende las MACs de los dispositivos conectados y las define de manera estática.

Ejemplo: Configurar la seguridad de puerto del switch de la siguiente topología para que cumpla los siguientes requisitos:

- La interfaz Fa/01 solo debe permitir la MAC del servidor DHCP. En caso de violación debe ser deshabilitada.
- La interfaz Fa0/2 puede permitir un máximo de 20 dispositivos de forma dinámica. En caso de violación se debe enviar una alerta al administrador, pero no deshabilitarla.
- La interfaz Fa0/3 puede permitir un máximo de 5 MACs, que serán aprendidas automáticamente, pero almacenadas de manera estática. En caso de violación de seguridad el puerto debe ser apagado.

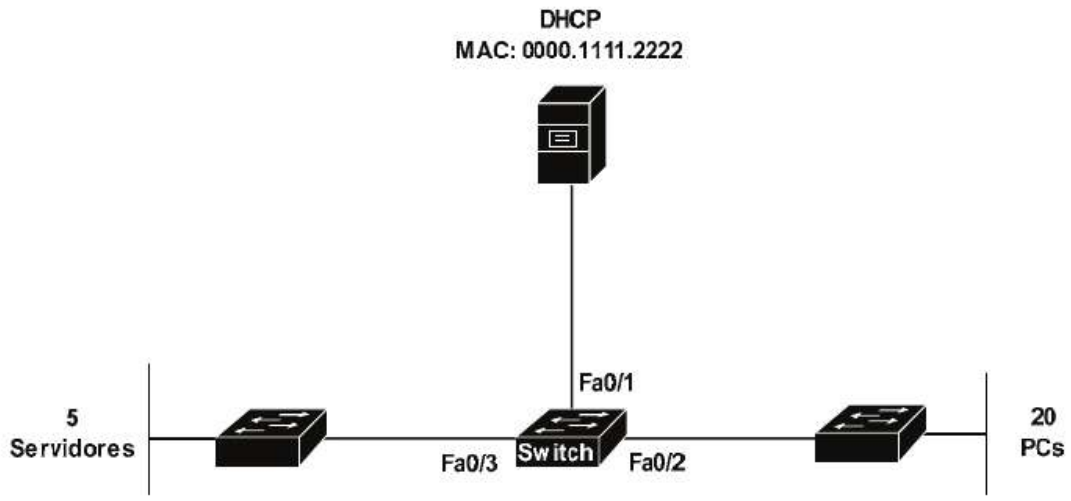


Fig. 2-10 Ejemplo de configuración. Seguridad de puerto.

```
Switch(config)#interface Fa0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security mac-address 0000.1111.2222
Switch(config-if)#exit
Switch(config)#interface Fa0/2
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security maximum 20
Switch(config-if)#switchport port-security violation restrict
Switch(config-if)#exit
Switch(config)#interface Fa0/3
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security maximum 5
Switch(config-if)#switchport port-security mac-address sticky
```

Con la configuración aplicada, Fa0/1 solo permitirá la conexión del servidor DHCP, cualquier otro intento de acceso supondrá una violación de seguridad y se deshabilitará la interfaz, ya que es la acción por defecto. En Fa0/2 podrán conectarse un máximo de 20 PCs, que no tienen por qué ser siempre los mismos. Si existe una violación de seguridad será almacenada en el log del dispositivo y a su vez se enviará una alerta por SNMP al administrador, pero no se deshabilitará la interfaz. En este caso el enlace es trunk porque conecta con otro switch. Mientras, en la interfaz Fa0/3 se permitirán 5 dispositivos cuyas MACs serán aprendidas automáticamente, pero almacenadas de manera estática.

La seguridad también debe ser aplicada en aquellas interfaces que no estén en uso con el fin de evitar accesos no autorizados. Los métodos más comunes para estos casos son:

- Deshabilitarlas administrativamente con el comando **shutdown.**

- Para aquellas en modo acceso, configurarlas para que formen parte de alguna VLAN que no esté en uso, con el comando **switchport access vlan *[num de vlan]*.**
- Para las configuradas en modo trunk, cambiar la VLAN nativa (VLAN 1 por defecto), con el comando **switchport trunk native vlan *[num de vlan]***.

Como se puede observar, algunas de estas medidas conllevan el uso de VLANs. Estas influyen en gran medida tanto en la seguridad como en el rendimiento de la red. Serán analizadas en este mismo capítulo.

Cualquier configuración de seguridad realizada en el switch puede ser fácilmente comprobada gracias al comando **show port-security interface *[interfaz],*** que ejecutado desde el modo privilegiado mostrará un listado que incluye las características aplicadas en la interfaz indicada.

COMPROBACIÓN DE LA TABLA DE MACS

Aunque no corresponda a una configuración propia de las interfaces, la tabla de MACs influye sobre ellas y su correcto funcionamiento. A lo largo del capítulo han sido analizados diferentes aspectos sobre su función y creación, sin embargo, aún no se ha accedido a su contenido para verificar que las entradas almacenadas son las correctas, o en su defecto, poder identificar y solucionar cualquier posible incidencia.

Para ello resulta necesario ejecutar el comando **show mac-address-table,** gracias al cual se obtiene un listado con las diferentes asociaciones aprendidas. Por ejemplo, para la siguiente topología:

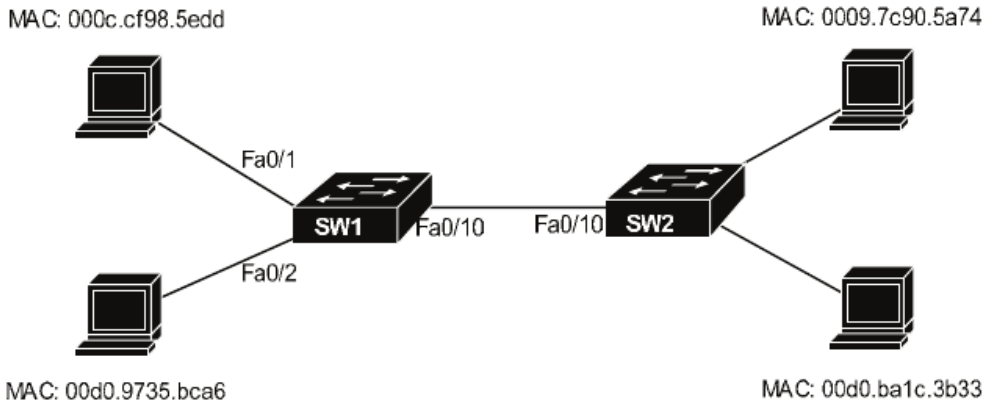


Fig. 2-11.1 Ejemplo. Acceso al contenido de la tabla de MACs.

```
SW1# show mac-address-table

          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----

   1    0009.7c90.5a74    DYNAMIC     Fa0/10
   1    000c.cf98.5edd    DYNAMIC     Fa0/1
   1    00d0.58e6.4918    DYNAMIC     Fa0/10
   1    00d0.9735.bca6    DYNAMIC     Fa0/2
   1    00d0.ba1c.3b33    DYNAMIC     Fa0/10
```

La asociación MAC/Interface creada por SW1 es correcta. En Fa0/10 aparecen registradas 3 entradas, estas corresponden a los dos 2 PCs conectados a SW2 y al propio Switch SW2.

Si existiera alguna incoherencia, como una interfaz asociada a una MAC errónea, la manera más sencilla de solucionarlo es borrar todas las entradas y obligar al Switch a aprenderlas de nuevo. La sentencia utilizada para ello es **clear mac-address-table,** desde el modo privilegiado.

```
SW1# clear mac-address-table
SW1# show mac-address-table

          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
```

Otros comandos útiles son:

- show mac-address-table dynamic: Muestra tan solo las entradas cuyas MACs han sido aprendidas de forma dinámica.
- show mac-address-table static: El resultado muestra las entradas configuradas de manera estática.
- show mac-address-table interfaces *[interfaz]*: Muestra un listado de las MACs asociadas a una determinada interfaz.

VLANS (Virtual LANs)

Hasta el momento, el concepto analizado de los switchs se basa principalmente en crear una LAN entre todos los miembros conectados al mismo, teniendo que hacer uso de un dispositivo de capa 3, como un router, cuando fuera necesario crear y conectar diferentes redes. Este hecho limita en gran medida la escalabilidad, ya que

a la hora de hacerlo sería necesario un switch por cada segmento y un router que conecte con cada uno de ellos. Imagina que una compañía dispone de dos departamentos, Ventas y RRHH, los cuales por seguridad y rendimiento deben permanecer en redes diferentes. Aplicando el modelo analizado hasta ahora la solución sería la siguiente:

Fig. 2-11.2 Diseño de red sin aplicar segmentación.

Esta práctica, en un entorno con multitud de subredes, supondría un coste económico elevado, así como la instalación y administración de demasiados dispositivos de red. Otra manera de lograr el mismo resultado, pero esta vez ahorrando en recursos sería mediante la implementación de VLANs.

Una VLAN puede ser definida como una tecnología de capa 2 que permite la segmentación de la red de manera lógica, logrando que dispositivos conectados al mismo o diferentes switchs puedan pertenecer a distintos segmentos de red sin la necesidad de un router para ello.

Con lo cual, la misma topología anterior puede ser simplificada a un solo switch con dos 2 VLANs.

Fig. 2-12.1 Diseño de red segmentada en VLANs.

Donde los dispositivos pertenecientes a una VLAN no podrán comunicarse con aquellos que formen parte de cualquier otra. Gracias a ello se ha logrado crear dos segmentos de red en un mismo Switch, generando a su vez dos dominios de broadcast, uno por subred, de tal manera que, si un dispositivo de "Ventas" envía un paquete broadcast, solo será recibido por aquellos de su misma VLAN.

Dicha segmentación agrega los siguientes beneficios sobre la red:

- Seguridad: La creación de VLANs implica que los miembros de una no puedan comunicarse con otras, salvo que sea configurado manualmente.

- Reducción de coste: Con un solo switch es posible configurar diferentes subredes sin necesidad de routers, lo que supone un ahorro de costes a la compañía.
- Rendimiento: Con las VLANs se reduce y limita el tráfico innecesario en la red.
- Mitigación de tormentas de broadcast: la creación de diferentes dominios de broadcast genera un mejor aprovechamiento del ancho de banda.
- Segregación lógica de la red: gracias a las VLANs, dispositivos que no estén ubicados físicamente en el mismo lugar ni conectados al mismo switch pueden formar parte de la misma subred.

CONFIGURACIÓN Y VERIFICACIÓN DE VLANS

Crear una VLAN en un switch Cisco y asignar las interfaces que formarán parte de ella es una tarea bastante sencilla. Bastará con aplicar la siguiente configuración:

- *Paso 1:* Crear la VLAN con el comando **vlan *[id de vlan]***, desde el modo de configuración global.
- *Paso 2* (Opcional): Definirle un nombre con el comando **name *[nombre]***, desde el modo de configuración de la VLAN. Este simplemente tiene fines administrativos, sobre todo de identificación. Si no fuera configurado, por defecto se llamará *VLANxxxx*, donde *xxxx* hace referencia al id numérico creado en el paso 1.
- *Paso 3:* Configurar en modo acceso las interfaces que formarán parte de la VLAN, ejecutando la sentencia **switchport mode access** desde el modo de configuración de cada una de ellas.
- *Paso 4:* Asociar la interfaz a la VLAN deseada con el comando **switchport access vlan *[vlan id]***, desde el modo de configuración de la interfaz y donde *vlan id* corresponde al valor numérico definido en el paso 1.

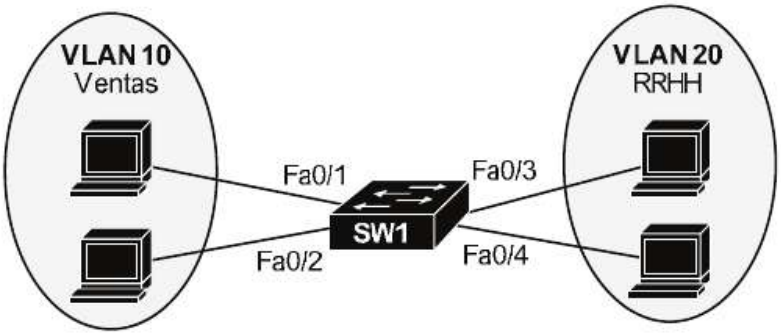


Fig. 2-12.2 Diseño de red segmentada en VLANs.

```
SW1(config)#vlan 10
SW1(config-vlan)#name VENTAS
SW1(config-vlan)#exit
SW1(config)#interface Fa0/1
SW1(config-if)#switchport mode access
SW1(config-if)#switchport access vlan 10
SW1(config-if)#exit
SW1(config)#interface Fa0/2
SW1(config-if)#switchport mode access
SW1(config-if)#switchport access vlan 10
SW1(config-if)#exit
SW1(config)#vlan 20
SW1(config-vlan)#name RRHH
SW1(config-vlan)#exit
SW1(config)#interface range Fa0/3 - 4
SW1(config-if-range)#switchport mode access
SW1(config-if-range)#switchport access vlan 20
```

En el supuesto caso de que a una interfaz se le asigne una VLAN inexistente, esta sería creada de manera automática. Por ejemplo, si Fa0/24 fuera configurada para formar parte de la VLAN30…

```
SW1(config)#interface Fa0/24
SW1(config-if)#switchport mode access
SW1(config-if)#switchport access vlan 30
% Access VLAN does not exist. Creating vlan 30
```

Por último, la verificación de configuración en este caso se realiza mediante el comando **show vlan brief**. IOS mostrará un resumen de las VLANs creadas y las interfaces asociadas a cada una de ellas.

```
SW1# show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22, Fa0/23
10   VENTAS                           active    Fa0/1, Fa0/2
20   RRHH                             active    Fa0/3, Fa0/4
30   VLAN0030                         active    Fa0/24
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
```

La VLAN por defecto en switchs Cisco es la 1 y contiene todas las interfaces, excepto aquellas que han sido configuradas manualmente para que formen parte de otra. Mientras, las VLANs 1002 hasta la 1005 son reservadas, incluidas en la configuración por defecto y no pueden ser eliminadas.

ENLACES TRONCALES

Una práctica muy habitual en redes corporativas consiste en la creación de las mismas VLANs en diferentes switchs y por supuesto la comunicación entre ellas, existiendo dos métodos posibles para llevarla a cabo. El primero consiste en establecer tantos enlaces como VLANs existentes, donde cada uno de ellos se encargará de transportar únicamente el tráfico de aquella en la que ha sido configurado.

De tal manera que:

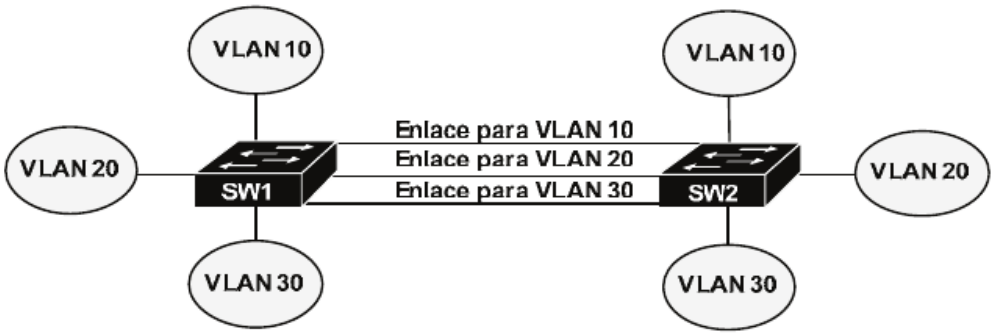


Fig. 2-13 Comunicación de VLANs en diferentes switchs sin enlace Trunk.

Este modelo es perfectamente operativo, pero a su vez muy poco práctico, si en lugar de 3, son configuradas 20 VLANs, serían necesarios 20 links.

La segunda opción, mucho más lógica y recomendable, consiste en la creación de un único enlace, denominado *Trunk* (o troncal) el cual será capaz de transportar el tráfico de todas las VLANs configuradas.

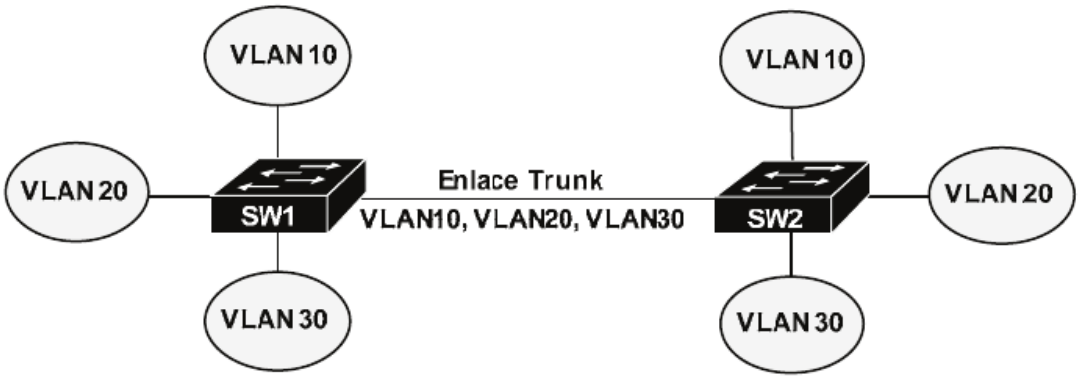


Fig. 2-14 Comunicación de VLANs en diferentes switchs con enlace Trunk.

Ahora bien, si SW1 envía todas las tramas a través de un mismo link, ¿cómo sabe SW2 a qué VLAN pertenece cada una de ellas? En este modelo se hace uso del etiquetado, que consiste en agregar un nuevo campo denominado *vlan identifier (vlan id)* a la trama creada en capa 2. El proceso llevado a cabo por SW2 consta de:

- *Paso 1:* Para cada trama recibida, el switch examina los campos MAC de destino y *vlan id*. Una vez hecho elimina este último.
- *Paso 2:* Comprueba que la MAC de destino pertenece a una interfaz configurada en la misma VLAN que la indicada en el campo *vlan id*.
- *Paso 3:* Si es así, reenvía la trama a su destino. De lo contrario es descartada.

Para llevar a cabo esta operación se puede hacer uso de los protocolos ISL (*Inter-Switch Link*) o IEEE 802.1Q. De ellos, el primero está en desuso debido a su antigüedad, por lo que se recomienda 802.1Q, es más, hace un tiempo que muchos dispositivos Cisco solo aceptan este. Su función consiste en identificar cada trama con la VLAN a la que pertenece. Para ello agrega una nueva etiqueta en capa 2, con un tamaño de 4 bytes y compuesta por cuatro campos: tipo, prioridad, flag y vlan id, con el siguiente formato:

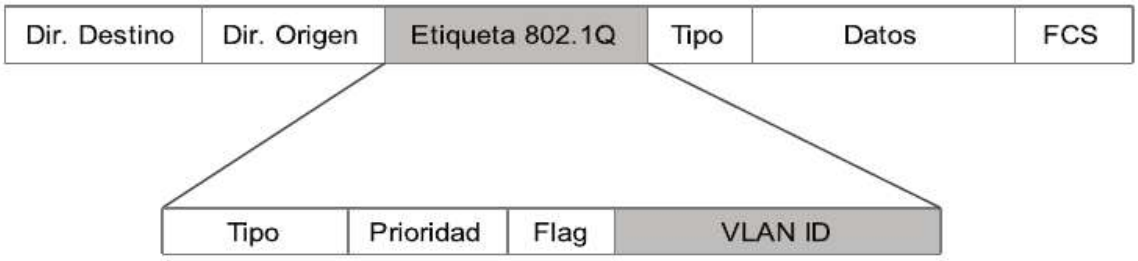


Fig. 2-15 Etiquetado VLAN 802.1Q.

De todos ellos, el único que es objeto de estudio en CCNA es el *vlan id*. Su longitud es de 12 bits, debido al número máximo de VLANs permitidas ($2^{12\ bits}$=4096) de las cuales solo pueden ser utilizadas 4094, y que a su vez Cisco las divide en dos rangos, normal, que abarca desde la 1 hasta la 1005, y extendido, desde la 1006 hasta la 4094. Solo algunos switchs permiten el rango extendido por lo que lo más común es hacer uso del normal.

Otra característica de los enlaces troncales es el uso de una VLAN nativa, la cual es necesaria para etiquetar el tráfico que debe ser enviado de aquellas interfaces que no han sido configuradas manualmente en ninguna VLAN, así como el tráfico "especial" generado por el propio switch, como el envío de logs, SNMP o conexiones vía telnet. Por defecto, la VLAN nativa es la 1. En el caso de modificar esta configuración, debe realizarse en los dos switchs que conforman el enlace, de lo contrario existirán problemas de comunicación.

Otra manera de enviar este tipo de tráfico es no agregar ninguna etiqueta a dichas tramas. Este es el método llevado a cabo por 802.1Q y así se asegura la compatibilidad con dispositivos que ejecuten otro protocolo. Cuando un Switch recibe una trama sin etiquetado, automáticamente la asocia con su VLAN Nativa.

Un ejemplo de comunicación a través de un enlace troncal podría ser:

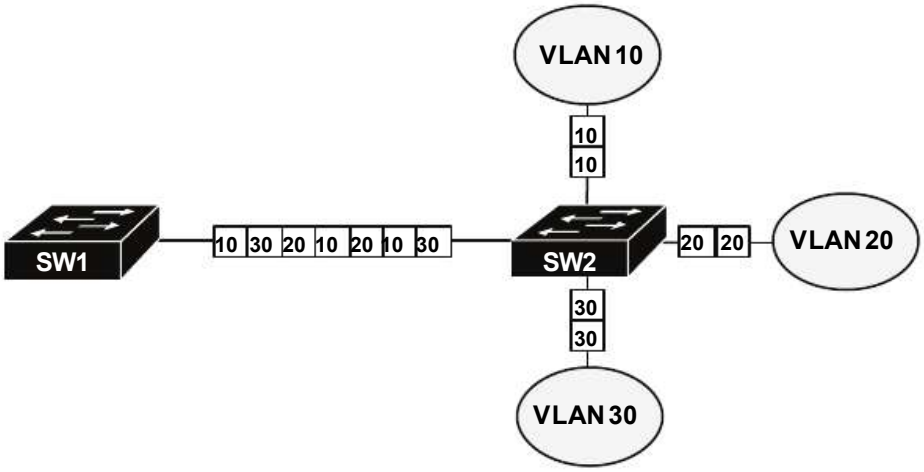


Fig. 2-16 Comunicación con etiquetado VLAN.

Configuración y verificación de enlaces troncales

La creación de este tipo de enlaces en un switch Cisco consta de las siguientes acciones, todas ellas ejecutadas desde el modo de configuración de la interfaz:

- *Paso 1:* Definir la interfaz en modo troncal con el comando **switchport mode trunk**.
- *Paso 2* (Opcional): Seleccionar el protocolo de encapsulación a utilizar a través de la sentencia **switchport trunk encapsulation *[dot1q | isl | negotiate]***. Si no es configurado se aplica por defecto dot1q (802.1Q). Algunos Switchs Cisco ya no incorporan esta opción porque solo permiten este último.
- *Paso 3* (Opcional): Identificar qué VLANs podrán ser transportadas a través del enlace, con el comando **switchport trunk allowed vlan *[add | all | except | remove] [vlan ids]*** donde cada parámetro desarrolla una función concreta como se verá a continuación.
- *Paso 4* (Opcional): Definir una VLAN nativa con el comando **switchport trunk native vlan *[id vlan]***. Si no es configurada automáticamente se aplica la 1.

Si se omitiera el paso número 3, el enlace troncal transportará por defecto el tráfico de todas las VLANs existentes (desde la 1 hasta la 4094). Sin embargo, por

razones de seguridad es recomendable configurarlo y tan solo permitir el paso de aquellas necesarias. Las funciones de los diferentes parámetros disponibles para ello son:

add: agrega VLANs a la lista de permitidas.
all: permite el tráfico de todas las existentes, desde la 1 hasta la 4094. Es la configuración por defecto.
except: no permite el paso de aquellas seleccionadas.
remove: elimina las VLANs indicadas de la lista de permitidas.

Ejemplo: Configurar un enlace troncal en la interfaz Fa0/10 de SW1, aplicando el protocolo 802.1Q y permitiendo tan solo el tráfico de las VLANs 10,20 y 30. (SW2 ya ha sido correctamente configurado).

```
SW1(config)#interface fa0/10
SW1(config-if)#switchport mode trunk
SW1(config-if)#switchport trunk allowed vlan 10,20
```

Con las sentencias aplicadas, el enlace opera haciendo uso del protocolo 802.1Q, tan solo transportará tráfico de las VLANs 10 y 20 y utiliza como nativa la 1. Durante su configuración el administrador ha olvidado incluir la 30. Para solucionarlo bastará con ejecutar el mismo comando haciendo uso del parámetro *add*…

```
SW1(config-if)#switchport trunk allowed vlan add 30
```

Imaginemos ahora que por motivos de seguridad los datos de la 10 no deben atravesar el enlace. Ello se soluciona gracias al parámetro *remove*…

```
SW1(config-if)#switchport trunk allowed vlan remove 10
```

Aplicar la misma configuración en ambos extremos resulta imprescindible con el fin de evitar problemas o bloqueo de determinado tráfico. En este aspecto se debe prestar especial atención al listado de VLANs permitidas y a la definida como nativa.

Una vez finalizado podrá ser verificado gracias al comando **show interfaces trunk.** El resultado mostrará información sobre las interfaces activas que a su vez operen en modo troncal, incluyendo el tipo de encapsulación y VLANs permitidas y nativa. Si una vez ejecutado no se muestra la interfaz deseada significa que existe algún problema que evita que el enlace se establezca. Estos suelen ser debidos a configuraciones diferentes en ambos extremos o errores físicos.

```
SW1# show interfaces trunk
Port        Mode         Encapsulation  Status        Native vlan
Fa0/10      on           802.1q         trunking      1
```

```
Port        Vlans allowed on trunk
Fa0/10      20,30

Port        Vlans allowed and active in management domain
Fa0/10      20,30

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/10      none
```

Otra manera de comprobarlo es mediante el comando **show interfaces *[interfaz]* switchport**, el cual mostrará un listado con las características del puerto en cuestión, incluyendo datos como el modo aplicado, tipo de encapsulación, etc.

Por último, en párrafos anteriores se ejecutó un **show vlan brief** con el fin de comprobar el listado de VLANs creadas y las interfaces asociadas a cada una de ellas. Si ahora se volviera a ejecutar, el resultado sería el siguiente:

```
SW1# show vlan brief

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active    Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                 Fa0/9, Fa0/11, Fa0/12, Fa0/13
                                                 Fa0/14, Fa0/15, Fa0/16, Fa0/17
                                                 Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                                 Fa0/22, Fa0/23
10   VENTAS                            active    Fa0/1, Fa0/2
20   RRHH                              active    Fa0/3, Fa0/4
30   VLAN0030                          active    Fa0/24
1002 fddi-default                      active
1003 token-ring-default                active
1004 fddinet-default                   active
1005 trnet-default                     active
```

Como se puede observar, la interfaz Fa0/10 no aparece asociada a ninguna VLAN. Ello es debido a que ha sido configurada como troncal.

Lista de VLANs permitidas en enlaces troncales

Uno de los problemas más comunes en los enlaces troncales es la falta de concordancia de VLANs permitidas en ambos extremos, teniendo como consecuencia el bloqueo de tráfico de aquellas que no coincidan. Por ejemplo:

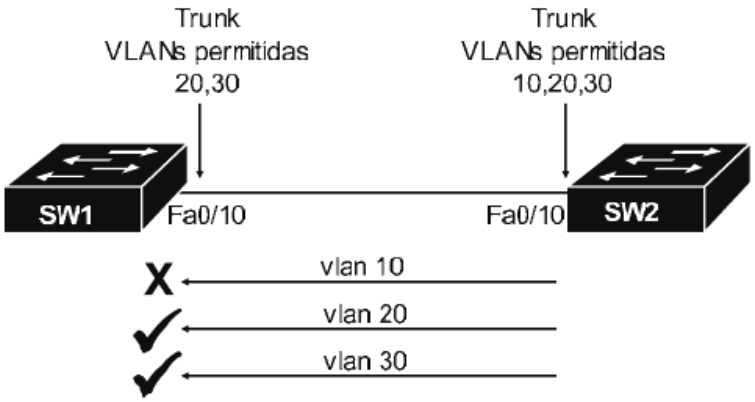


Fig. 2-17 Falta de concordancia de VLANs permitidas en enlaces troncales.

SW1 no aceptará ni enviará tráfico de la VLAN 10 porque no ha sido configurado para ello.

La manera más sencilla de identificar este tipo de errores es ejecutando un **show interfaces trunk** en ambos switchs y comparar sus resultados, prestando especial atención al listado de VLANs permitidas en cada uno de ellos.

```
SW1# show interfaces trunk

Port        Mode         Encapsulation  Status        Native vlan
Fa0/10      on           802.1q         trunking      1

Port        Vlans allowed on trunk
Fa0/10      20, 30


SW2# show interfaces trunk

Port        Mode         Encapsulation  Status        Native vlan
Fa0/10      on           802.1q         trunking      1

Port        Vlans allowed on trunk
Fa0/10      10, 20, 30
```

Gracias a la información obtenida se puede concluir que el problema reside en que el enlace troncal de SW1 no ha sido configurado para transportar tráfico de la VLAN 10. Para solucionarlo bastará con hacer uso de los comandos de configuración ya analizados anteriormente.

```
SW1(config)# int fa0/10
SW1(config-if)# switchport trunk allowed vlan add 10
```

Con los cambios realizados, el tráfico que antes era descartado ahora será permitido, solucionando así el problema de comunicación que existía entre ambos switchs.

ENRUTAMIENTO ENTRE VLANs

Como se ha mencionado en párrafos anteriores, cada VLAN debe pertenecer a un rango de red diferente, logrando así la segmentación. Este hecho tiene como consecuencia que los switchs no puedan llevar a cabo la comunicación entre ellas, ya que al operar en capa 2 no son capaces de realizar el enrutamiento, siendo un Router o un switch de capa 3 los dispositivos necesarios para tal propósito.

El modelo a implementar resulta sencillo, bastará con establecer un enlace entre el switch y el router y configurar este último para que ejecute el enrutamiento entre los diferentes segmentos. Para ello se puede optar por dos opciones.

La primera es hacer uso de una interfaz para cada VLAN, en cuyo caso deberán cumplirse las siguientes condiciones:

- Cada interfaz del switch que conecta con el router debe operar en modo acceso y formar parte de la misma VLAN que se pretende enrutar.
- La interfaz del router debe ser configurada con una dirección IP perteneciente al rango de la VLAN que va a enrutar.
- Los dispositivos pertenecientes a la misma VLAN deberán utilizar como puerta de enlace la IP de la interfaz del router que se encargará de su enrutamiento.

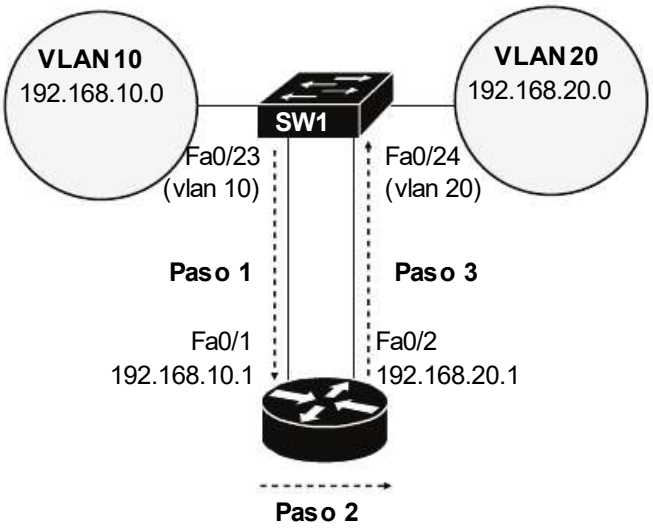


Fig. 2-18 Enrutamiento Inter-VLAN sin enlace troncal.

Paso 1: Un PC de la VLAN10 desea comunicarse con otro de la VLAN20. Como son subredes diferentes envía el paquete a su puerta de enlace, que es la interfaz Fa0/1 del router.

Paso 2: El router recibe el paquete, elimina la etiqueta de la VLAN10, lee la dirección IP de destino, comprueba su tabla de enrutamiento y concluye que la red de destino está directamente conectada a través de su interfaz Fa0/2, por lo tanto, reenvía el paquete a través de ella.

Paso 3: La trama es recibida por la interfaz Fa0/24 del switch, que a su vez la reenvía al PC de destino.

Al igual que sucede con la comunicación entre mismas VLANs en diferentes switchs, el método de necesitar un enlace por cada una de ellas resulta poco práctico. Lo ideal en estos casos consiste en aplicar el modelo *router-on-a-stick*, el cual basa el enrutamiento en la utilización de un solo link que comunicará todas las VLANs. Para llevarlo a cabo se deben cumplir los siguientes requisitos:

- La interfaz del switch que conecta con el router debe estar configurada en modo troncal, permitiendo el tráfico de todas las VLANs que se desean comunicar.
- La interfaz del router será configurada para que opere con el protocolo 802.1Q. Además, en la misma se crearán subinterfaces, una por cada VLAN, que a su vez serán configuradas con una IP dentro del rango de cada una de ellas.
- Los dispositivos de las VLANs utilizarán como puerta de enlace la IP de la subinterfaz encargada de su enrutamiento.

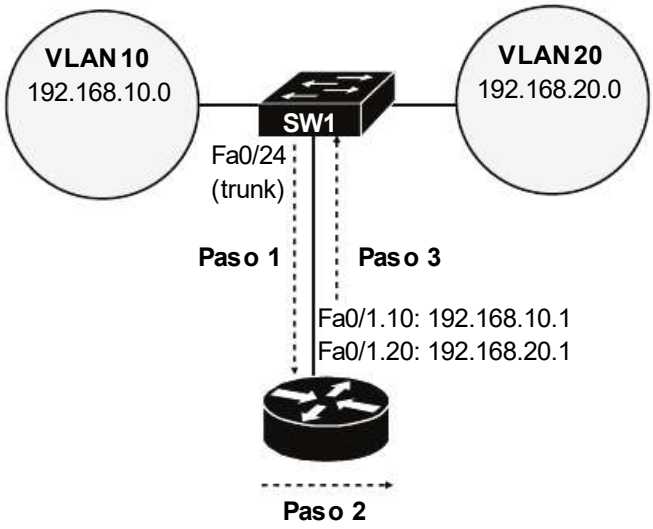


Fig. 2-19 Enrutamiento Inter-VLAN. Modelo router-on-a-stick.

Paso 1: Un PC de la VLAN10 desea comunicarse con otro de la VLAN20. Como son subredes diferentes, envía el paquete a su puerta de enlace, cuya IP es la 192.168.10.1.

Paso 2: El router recibe el paquete, elimina la etiqueta de la VLAN10, lee la dirección IP de destino, comprueba su tabla de enrutamiento y concluye que la red de destino está directamente conectada a través de su subinterfaz Fa0/1.20. El router reenvía el paquete a través de la interfaz física Fa0/1 agregando a la trama una etiqueta de la VLAN20.

Paso 3: La trama es recibida por el switch, que a su vez la reenvía al PC de destino.

Las subinterfaces simplemente crean diferentes interfaces lógicas sobre una física. Ello significa que el tráfico tan solo atravesará un link físico, pero a nivel de procesamiento interno cada una de ellas es tratada de manera individual, con su propia configuración y características.

La configuración de ambos modelos se realiza desde el router y serán analizados en el capítulo 5 "*Instalación y configuración inicial de routers Cisco*".

MODO DE OPERAR DE LAS INTERFACES

El comando **switchport mode *[opciones]*** es utilizado para definir el modo de operar de cada interfaz, el cual puede ser troncal o de acceso. Sin embargo, también permite la negociación automática entre ambos extremos para aplicar uno u otro. Las 4 opciones permitidas son: *access*, *trunk*, *dynamic desirable* y *dynamic auto*. Las dos primeras ya han sido analizadas, siendo *dynamic desirable* y *dynamic auto* las encargadas de llevar a cabo dicha negociación. Las características de cada una de ellas son:

Comando	Descripción
`switchport mode access`	La interfaz solo opera en modo acceso.
`switchport mode trunk`	La interfaz solo opera en modo troncal.
`switchport mode dynamic desirable`	La interfaz inicia una negociación con el otro extremo para convertir el enlace en troncal. También responde a los mensajes cuando dicha negociación es iniciada desde el otro extremo.
`switchport mode dynamic auto`	La interfaz no inicia ninguna negociación, pero espera a que el otro extremo sí lo haga. Si es así, responde a los mensajes para negociar el trunk del enlace. Es el modo por defecto utilizado en switchs Cisco.

Siendo las posibles combinaciones y el modo aplicado en cada caso los siguientes:

Modo configurado en la interfaz	Modo configurado en la interfaz del otro extremo			
	Access	Dynamic auto	Trunk	Dynamic desirable
Access	Enlace en modo acceso	Enlace en modo acceso	No se forma enlace	Enlace en modo acceso
Dynamic auto	Enlace en modo acceso	Enlace en modo acceso	Enlace en modo trunk	Enlace en modo trunk
Trunk	No se forma enlace	Enlace en modo trunk	Enlace en modo trunk	Enlace en modo trunk
Dynamic desirable	Enlace en modo acceso	Enlace en modo trunk	Enlace en modo trunk	Enlace en modo trunk

VTP (VLAN Trunking Protocol)

En entornos corporativos de gran tamaño compuestos por multitud de sedes y dispositivos se hace imprescindible un buen diseño de red, siendo necesaria la segmentación de esta y con ello la creación de numerosas VLANs. Su configuración debe llevarse a cabo en cada uno de los switchs, y a su vez, cualquier modificación sobre las mismas también deberá ser gestionada de manera independiente en cada dispositivo.

Con el fin de automatizar dicho proceso nace VTP, un protocolo en capa 2 desarrollado por Cisco con la finalidad de habilitar la propagación de VLANs de manera automática entre los diferentes switchs ubicados en la red, de tal manera que cada uno de ellos aprenda y agregue en su configuración aquellas creadas en otros dispositivos.

Para que ello sea posible, el protocolo establece 3 roles que definirán el modo de operar de cada uno de los miembros, siendo los siguientes:

Modo servidor: los servidores son los switchs que propagan sus VLANs a todos los dispositivos que formen parte del mismo dominio VTP. Permiten crearlas, eliminarlas o renombrarlas y es el modo aplicado por defecto en switchs Cisco.

Modo cliente: los clientes no pueden crear, cambiar o eliminar VLANs, tan solo se limitan a aprender la información desde el servidor y agregarla en su configuración.

En este caso, ante un reinicio del dispositivo toda ella es eliminada, debiendo aprenderla de nuevo.

Modo transparente: los switchs en modo transparente reciben las actualizaciones de los servidores VTP, pero no las aplican sobre sí mismos, ni siquiera las leen ni mucho menos las almacenan, tan solo las reenvían. Este modo permite crear, eliminar y renombrar VLANs, pero tan solo serán aplicadas en el switch local.

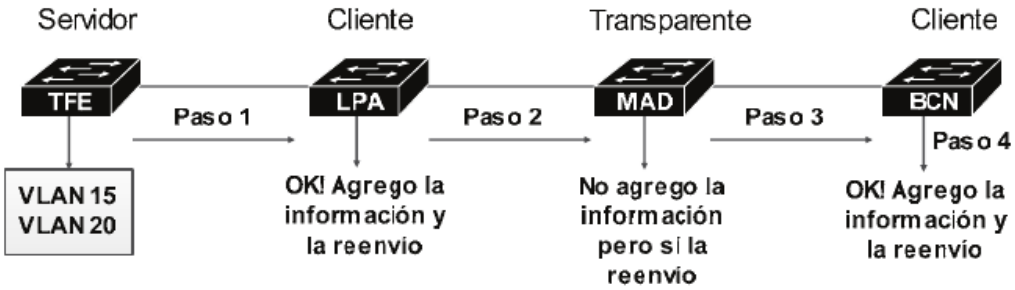


Fig. 2-20 Roles en VTP.

Paso 1: TFE, que actúa como servidor VTP, propaga sus VLANs al resto de miembros.

Paso 2: LPA, que ha sido configurado en modo cliente, las acepta y agrega en su propia configuración, para acto seguido reenviarlas.

Paso 3: MAD opera en modo transparente, por lo que simplemente se limita a reenviar la información recibida.

Paso 4: Por último, BCN actúa como cliente, por lo tanto, agrega las VLANs recibidas en su propia configuración.

VTP permite definir diferentes dominios con opción de autenticación sobre cada uno de ellos, de tal manera que el intercambio de información tan solo se llevará a cabo entre miembros pertenecientes al mismo y autenticados correctamente. Además, el enlace entre ambos switchs debe ser configurado en modo troncal. Cumpliendo dichas condiciones, la propagación de VLANs entre los diferentes dispositivos se hace posible, pero ¿cómo gestiona VTP sus actualizaciones? Para ello, y con el fin de ejecutar la sincronización de manera correcta, cada actualización tan solo incluye información que no ha sido enviada previamente, siendo a su vez identificada mediante un número de revisión, el cual, cuando es recibido por el cliente, lo compara con el último aplicado sobre su propia configuración y conforme al resultado actúa de una manera u otra. Si coinciden o es menor que el local,

significa que dispone de dicha actualización, por lo tanto, no agrega la información incluida, simplemente se limita a reenviarla. Si por el contrario el número de revisión recibido resulta mayor que el propio, significa que la actualización aún no ha sido aplicada sobre el dispositivo, en cuyo caso se agrega y reenvía.

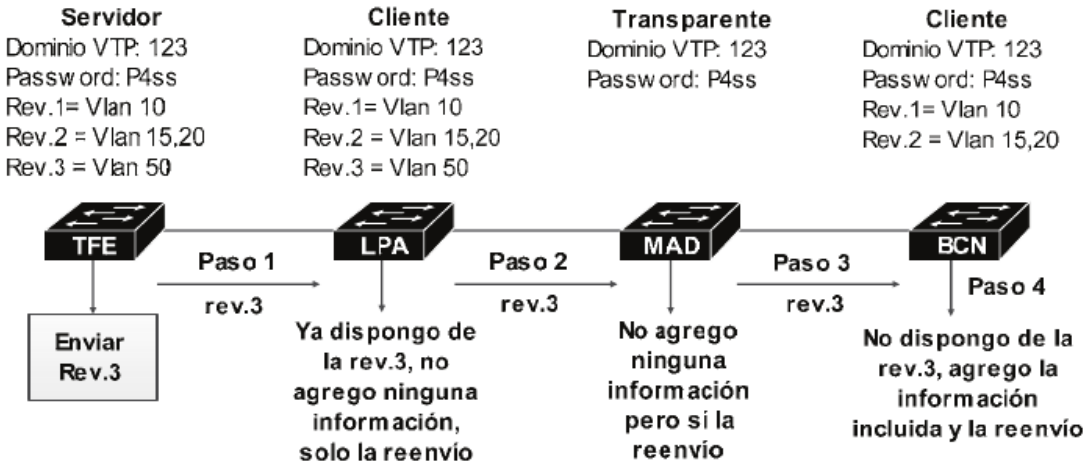


Fig. 2-21 Modo de operar de VTP.

Primero, TFE envía una actualización con número de revisión 3, la cual incluye la VLAN 50. Esta es recibida por LPA, que en relación con su propia configuración verifica que ya ha sido aplicada con anterioridad, por lo que simplemente la reenvía. MAD, que opera en modo transparente, se limita a reenviarla sin siquiera analizarla. Por último, BCN comprueba que no dispone de dicha actualización, por lo tanto, la agrega en su configuración.

Configuración y verificación de VTP

El proceso, en switchs Cisco, consta de las siguientes acciones, todas ellas ejecutadas desde el modo de configuración global:

- *Paso 1:* Definir el modo de operación del dispositivo, con el comando **vtp mode [*server* / *client* / *transparent*].**
- *Paso 2:* Asignar un dominio VTP mediante la sentencia **vtp domain [*nombre*].**
- *Paso 3* (Opcional): Establecer la autenticación necesaria para el dominio con el comando **vtp password [*contraseña*].**
- *Paso 4* (Opcional): Seleccionar la versión en la cual operará el dispositivo, ejecutando para ello el comando **vtp version [*1* / *2* / *3*],** de las cuales la 1 y 2 son compatibles. Sin embargo, la versión 3 no soporta ninguna de las anteriores.

Ejemplo: Configurar VTP en los switchs TFE y LPA de tal manera que:

- TFE actúe como servidor.
- LPA opere en modo cliente.
- Ambos pertenezcan al dominio VTP "CorpSA" con contraseña "P4ss", y ejecutando la versión 2 del protocolo.

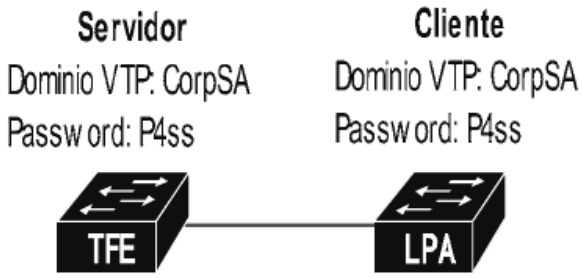


Fig. 2-22 Ejemplo de VTP.

```
---Configuración en TFE---
TFE(config)#vtp mode server
TFE(config)#vtp domain CorpSA
TFE(config)#vtp password P4ss
TFE(config)#vtp version 2

---Configuración en LPA---
LPA(config)#vtp mode client
LPA(config)#vtp domain CorpSA
LPA(config)#vtp password P4ss
LPA(config)#vtp version 2
```

Una vez aplicados los cambios comienza el intercambio de información entre ambos, donde aquellas VLANs creadas en TFE serán propagadas de manera automática hacia LPA.

Tras ello, y con el fin de verificar su correcto funcionamiento, IOS dispone del comando **show vtp status**, que facilita información como la versión ejecutada, número de revisión, dominio o número de VLANs totales, entre otros.

```
LPA>show vtp status
VTP Version                     : 2
Configuration Revision          : 3
Maximum VLANs supported locally : 255
Number of existing VLANs        : 7
VTP Operating Mode              : Client
VTP Domain Name                 : CorpSA
VTP Pruning Mode                : Disabled
VTP V2 Mode                     : Disabled
VTP Traps Generation            : Disabled
MD5 digest                      : 0x1B 0x07 0x51 0x1D 0xB3 0x26 0x28
```

Además, ejecutando un **show vlan brief** en ambos switchs se deberá obtener exactamente el mismo resultado (a excepción de las interfaces asociadas a cada VLAN).

```
TFE#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/2, Fa0/3, Fa0/4, Fa0/5
                                                Fa0/10, Fa0/11, Fa0/12, Fa0/13
                                                Fa0/14, Fa0/15, Fa0/16, Fa0/17
                                                Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                                Fa0/22, Fa0/23, Fa0/24, Gig1/1
                                                Gig1/2
15   VLAN0015                         active    Fa0/6, Fa0/7, Fa0/8, Fa0/9
20   VLAN0020                         active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active


LPA>show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/2, Fa0/4, Fa0/5, Fa0/6
                                                Fa0/7, Fa0/8, Fa0/9, Fa0/10
                                                Fa0/11, Fa0/12, Fa0/13, Fa0/14
                                                Fa0/19, Fa0/20, Fa0/21, Fa0/22
                                                Fa0/23, Fa0/24, Gig1/1, Gig1/2
15   VLAN0015                         active
20   VLAN0020                         active    Fa0/15, Fa0/16, Fa0/17, Fa0/18
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
```

TEST CAPÍTULO 2: CONFIGURACIÓN DE SWITCHS CISCO

1.- ¿Qué dispositivo opera en 10BASE-T y reenvía las tramas a través de todos sus puertos?

A. Switch.
B. Puente.
C. Hub.
D. Router.
E. Switch de capa 3.

2.- ¿Qué dispositivo crea una tabla de MACs y basa sus decisiones de reenvío en torno a ella?

A. Switch.
B. Puente.
C. Hub.
D. Router.

3.- Cuando un switch no conoce la MAC de destino de una trama, ¿cómo procede?

A. Descarta la trama.
B. Ejecuta un ARP para averiguar su IP.
C. Reenvía la trama a través de todas sus interfaces excepto por la cual fue recibida.
D. Descarta la trama e informa al host con un mensaje de "*Destino inalcanzable*".

4.- Un administrador de red visualiza la tabla de MACs de un switch y comprueba que una de sus interfaces tiene asociados más de 10 dispositivos diferentes. ¿A qué puede ser debido?

A. A que a ella se han conectado dispositivos con diferentes MACs durante los últimos 60 días.
B. A que a ella se conecta un solo dispositivo en el que se ha cambiado su tarjeta de red en más de 10 ocasiones o se está falsificando su dirección MAC.
C. Que la interfaz conecta con algún otro dispositivo de red como un switch o un punto de acceso.
D. Que la interfaz tiene más de 10 tramas en cola para ser reenviadas.

5.- ¿Qué protocolo es utilizado por los switchs para evitar bucles de capa 2?

A. FTP.

B. STP.
C. SSP.
D. SSC.

6.- Los cambios de configuración en un switch Cisco se realizan desde el modo:
A. Usuario.
B. Administrador.
C. Super-usuario.
D. Privilegiado.

7.- ¿Qué tipo de procesamiento es aplicado normalmente por los switchs Cisco?
A. Store-and-Forward.
B. Cut-Through.
C. Fragment-Free.
D. Store-and-Reload.

8.- Un Switch está compuesto por 24 interfaces, todas en uso. Una de ellas conecta directamente con un router y otra con un hub de 3 puertos. ¿Cuántos dominios de colisión se crean en el switch?
A. 24
B. 22
C. 27
D. 23

9.- Un switch ha sido configurado con 4 VLANs. La VLAN1 está compuesta por 10 interfaces, la 2 por 5 interfaces, la 3 por 15 y la 4 por ninguna. ¿Cuántos dominios de broadcast existen en el switch?
A. 1
B. 4
C. 30
D. 3
E. 0

10.- ¿Qué tipo de asociación se crea en la tabla de MACs de un switch?
A. MAC/IP.
B. MAC/interfaz.
C. Interfaz/IP.
D. MAC/Protocolo.

11.- Un administrador de red ha configurado el acceso remoto a un switch utilizando para ello el comando "*line vty 0 10*". ¿Cuántas conexiones simultáneas se podrán realizar al dispositivo?

A. Ninguna. El comando correcto es “interface vty 0 10”.
B. Ninguna. El comando correcto es “line vty 0 15”.
C. 0.
D. 10.
E. 11.

12.- ¿En qué tipo de memoria es almacenado el fichero *startup-config*?
A. NVRAM.
B. RAM.
C. ROM.
D. Flash.

13.- ¿Qué tipo de cable es necesario para acceder localmente a la CLI de un dispositivo Cisco?
A. STP directo.
B. STP cruzado.
C. Serial.
D. Rollover.

14.- Un administrador de red conecta a la CLI de un switch Cisco y aparece el prompt “*Switch>”,* a través del cual no puede realizar ninguna configuración. ¿Qué comando deberá ejecutar en primer lugar para poder aplicar cambios en el dispositivo?
A. Configure terminal.
B. Enable.
C. Access mode.
D. Show Running-config.

15.- ¿En qué memoria de un switch Cisco es almacenado el fichero *running-config*?
A. RAM.
B. Flash.
C. ROM.
D. NVRAM.

16.- ¿En qué memoria de un switch Cisco es almacenada la imagen IOS?
A. RAM.
B. Flash.
C. ROM.
D. NVRAM.

17.- Un administrador de red ha cambiado el nombre de un switch ejecutando el comando "*hostname Central*". Se han aplicado los cambios con éxito, sin embargo, tras una caída de electricidad se produce un reinicio y el administrador observa que el nombre vuelve a ser el inicial. ¿A qué puede ser debido?

A. El administrador ha realizado los cambios desde el modo usuario y por lo tanto no se han guardado.
B. Debido a la caída de electricidad, el switch no ha guardado bien los cambios en sus ficheros de configuración, cargando la configuración de fábrica.
C. El administrador de red no ha ejecutado el comando "*save config*" al finalizar los cambios de configuración.
D. El switch tiene algún fallo de memoria y debe ser reemplazado.
E. El administrador de red no ha ejecutado el comando "*copy running-config startup-config*" al finalizar la configuración.

18.- Un administrador de red observa en el fichero *running-config* que la contraseña "enable" aparece cifrada, pero el resto, como la de las líneas VTY continúan en texto plano. ¿A qué se debe?

A. La contraseña enable aparece cifrada porque se ha ejecutado el comando "*service password-encryption*".
B. Es normal, IOS cifra automáticamente la contraseña enable por seguridad.
C. El administrador de red ha ejecutado el comando "*enable secret [contraseña]*".
D. Ninguna de las anteriores.

19.- El switch central de la compañía ha sido configurado para que el acceso remoto se lleve a cabo mediante usuario y contraseña, sin embargo, cuando el administrador de red conecta a través de las líneas vty, el switch no aplica este tipo de autenticación. Tras ejecutar un "*show running-config*" se obtiene el siguiente resultado…

```
no service timestamps debug datetime msec
service password-encryption
!
hostname VENTAS
!
enable secret 5 $1$mERr$fDuZEPfRLI/aWvzEgceCS1
username acceso secret cisco
username acceso1 secret cisco
!
...Resultado omitido por brevedad...
!
line con 0
 password 7 0822455D0A1648141D051F0B262E
 login
!
line vty 0 15
 password 7 0822455D0A1648010612
 login
```

¿A qué es debido?

A. "*enable secret*" no es compatible con autenticación de usuario y contraseña.
B. En los comandos de creación de usuarios se debe utilizar el parámetro "*password*" en lugar de "*secret*".
C. Falta el comando "*no shutdown*" en las líneas vty.
D. Falta el comando "*login local*" en las líneas vty.

20.- ¿Cuál es la función principal de los servidores AAA?

A. Autenticación centralizada.
B. Asignación automática de direcciones de red.
C. Control de acceso a la red.
D. Resolver nombres a direcciones IP.

21.- ¿Cuáles de los siguientes pasos son necesarios para implementar SSH en lugar de telnet en IOS? (Seleccionar tres respuestas)

A. Configurar autenticación con usuario y contraseña.
B. Configurar un nombre de dominio en el dispositivo.
C. Configurar una dirección IP para acceso remoto.
D. Crear una clave asimétrica.
E. Cifrar el acceso con el comando "*service password-encryption*".
F. Aplicar el comando "*transport input ssh*".
G. Configurar el dispositivo para que conecte con un servidor AAA.

22.- ¿Qué utilidad tiene el comando "*interface vlan [num vlan]*" en un switch Cisco?

A. Crea una interfaz virtual para el acceso remoto.
B. Crea una VLAN que podrá ser asignada a las diferentes interfaces.
C. Accede al modo de configuración de una determinada VLAN.
D. Muestra información detallada de la VLAN indicada.

23.- La tarjeta de red de un PC opera a 10 Mbps y la interfaz del switch con el que conecta ha sido configurada con el comando "speed 100". ¿Qué velocidad se aplicará al enlace?

A. 10 Mbps full-duplex.
B. 10 Mbps half-duplex.
C. 100 Mbps.
D. No se establecerá el enlace.

24.- En relación con la siguiente configuración, ¿Cuál de las siguientes afirmaciones es correcta?

```
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security mac-address aaaa.bbbb.cccc
```

A. Aplica seguridad de puerto, de tal manera que si un dispositivo con MAC *aaaa.bbbb.cccc* conecta a la interfaz genera una violación de seguridad y esta es deshabilitada.
B. Aplica seguridad de puerto, de tal manera que si un dispositivo con MAC *aaaa.bbbb.cccc* conecta a la interfaz genera una violación de seguridad y un log que se enviará por SNMP.
C. Aplica seguridad de puerto, de tal manera que si un dispositivo con MAC *aaaa.bbbb.cccc* conecta a la interfaz genera una violación de seguridad que apaga el switch.
D. Ninguna de las anteriores.

25.- ¿Qué protocolos pueden ser utilizados en los enlaces troncales para el etiquetado de VLANs? (Seleccionar dos respuestas)

A. 802.11
B. 802.3
C. 802.1Q
D. ISL
E. HDLC

26.- La siguiente información no muestra la interfaz Fa0/1. ¿A qué puede ser debido?

```
Switch# show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22, Fa0/23
10   VLAN0050                         active    Fa0/10, Fa0/2
20   VLAN0010                         active    Fa0/3, Fa0/4
30   VLAN0090                         active    Fa0/24
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
```

A. La interfaz está apagada con el comando "*shutdown*".
B. La interfaz conecta directamente con un router.

C. La interfaz tiene activada la seguridad de puerto.
D. La interfaz está configurada como trunk.

27.- En un enlace entre dos switchs, un extremo está configurado como "*dynamic auto*" y el otro como "*trunk*". ¿En qué modo operará el enlace?

A. Modo Trunk.
B. Modo Acceso.
C. No se establecerá el enlace.
D. Dynamic desirable.

28.- Un administrador recibe una alerta de un usuario que no tiene conexión con la red. Examinando el switch de acceso comprueba que la interfaz que conecta con dicho usuario se encuentra en estado "ERR-DISABLED". ¿Cuál es la causa más probable de la incidencia?

A. Fallo en el cable.
B. Fallo de negociación en la velocidad del enlace.
C. Fallo en la tarjeta de red del PC del usuario.
D. Interfaz apagada con el comando "*shutdown"*.
E. Violación de seguridad en la interfaz.
F. La interfaz pertenece a una VLAN que no existe en el switch.

29.- ¿Qué acciones debe ejecutar un administrador de red para que una interfaz en modo ERR-DISABLED vuelva a ser operativa?

A. Desconectar el cable de red y volver a conectarlo.
B. Cambiar el cable de red.
C. Configurar la misma velocidad de enlace en ambos extremos.
D. Aplicar, primero el comando "*shutdown*" y después "*no shutdown*" en la interfaz.
E. Configurar la interfaz para que forme parte de una VLAN existente.

30.- ¿A qué se debe que el estado de una interfaz sea "*Up/Down*"?

A. A un fallo de protocolo en ambos extremos.
B. Falta de concordancia de velocidad en ambos extremos.
C. Un extremo está operativo pero el otro no.
D. Un extremo está configurado con el comando "*no shutdown*", mientras que el otro extremo con el comando "*shutdown*".
E. Ninguna de las anteriores.

31.- En el switch central de la compañía se ha ejecutado el comando "*clear mac-address-table*". ¿Qué hará el switch con la primera trama que reciba tras ejecutar el comando?

A. Descartarla.
B. Reenviarla a través de todas sus interfaces excepto por la cual fue recibida.
C. Reenviarla al destinatario mediante comunicación unicast.
D. No reenviará ninguna trama hasta que vuelva a completar su tabla de MACs.

32.- Un administrador de red aplica el comando "*switchport access vlan 10*" en una interfaz. Sin embargo, dicha VLAN no existe en el dispositivo. ¿Cómo procederá el switch?

A. Mostrará un mensaje de error en la CLI.
B. Cuando la VLAN10 sea creada manualmente, se asignará automáticamente a la interfaz.
C. Creará la VLAN10 de manera automática.
D. Aunque no exista, el tráfico que genere la interfaz será etiquetado con el id de vlan 10.

33.- ¿Cuál de los siguientes dispositivos se encuentra en el mismo dominio de colisión que PC1?

A. PC2, que se encuentra separado de PC1 por un router.
B. PC3, que se encuentra separado de PC1 por un switch.
C. PC4, que se encuentra separado de PC1 por un bridge.
D. Ninguno.

34.- En un switch se han configurado los comandos "*enable secret AccessPass*" y "*enable password ContraseñaSecreta*". Cuando un usuario acceda al modo privilegiado, ¿qué contraseña será aceptada por IOS?

A. AccessPass.
B. ContraseñaSecreta.
C. Cualquiera de las dos contraseñas será aceptada.
D. No es posible configurar ambos comandos en el switch de manera simultánea.

35.- En un switch han sido configuradas 10 VLANs, estando todas ellas en uso. ¿Cuántas subredes son necesarias?

A. 0
B. 1
C. 10

D. Las que el administrador de red considere oportunas.
E. Dependiendo del modelo de switch se necesitarán 5 o 10.

36.- Cuando una trama es enviada a través de un enlace troncal, ¿cómo determina el receptor a qué VLAN pertenece?

A. Gracias a la dirección IP de origen.
B. Gracias a la dirección IP de destino.
C. Gracias a la dirección MAC de origen.
D. Gracias a la dirección MAC de destino.
E. Gracias al campo ID VLAN del frame.
F. Gracias al campo FCS del frame.

37.- Desde un switch se necesita disponer de información detallada sobre dispositivos de red directamente conectados. ¿Qué comando es necesario para ello?

A. Show cdp neighbors.
B. Show cdp neighbors detail.
C. Show cdp neighbors brief.
D. Show cdp neighbors table.

SPANNING TREE PROTOCOL 3

CONCEPTOS BÁSICOS DE STP

Una práctica muy habitual en redes corporativas consiste en la aplicación de enlaces redundantes con el fin de mejorar el servicio y rendimiento de estas. Sin embargo, dicha redundancia puede generar problemas, traducidos normalmente en bucles de capa 2. Este hecho, también denominado *tormenta de broadcast*, se produce cuando la misma trama recorre los mismos enlaces de manera infinita, pero ¿cuáles son las consecuencias reales ante tal situación? Se podrían identificar las siguientes:

- Las tramas que hayan entrado en bucle lo recorrerán de manera infinita, sin ser descartadas nunca por los switchs. Ello es debido a que no disponen del campo TTL, el cual es utilizado en capa 3 para eliminar paquetes que superen un determinado número de saltos. Su ausencia en capa 2 conlleva que no exista un control sobre la vida o recorrido de las tramas, por lo tanto, serán siempre procesadas y reenviadas. A consecuencia de ello, el ancho de banda de los enlaces se verá afectado considerablemente, hasta tal punto que la red quedará inutilizable. La única manera de detenerlo es desconectando físicamente alguno de los enlaces intervinientes o apagando y encendiendo administrativamente la interfaz implicada con los comandos *shutdown* y *no shutdown*.
- La tabla de MACs de los switchs que forman parte del bucle estarán continuamente actualizándose, dando como resultado registros incorrectos y con ello reenvíos de tramas a través de interfaces erróneas.

- Los switchs que intervienen en el bucle tienen que procesar todas las tramas que lo atraviesan de manera continua e infinita, lo que genera una bajada de rendimiento tanto a nivel de hardware como de software (IOS).
- Las tramas que atraviesan el bucle también son recibidas por dispositivos finales. Estos las aceptan y procesan siempre, hecho que genera una disminución de su ancho de banda y rendimiento.

Un ejemplo de tormenta de broadcast podría ser el siguiente:

Paso 1: Un PC de la Red A envía una trama al SwitchA. Este, como se trata de un broadcast, la reenvía a través de todas sus interfaces excepto por la que fue recibida, lo que incluye los enlaces hacia el SwitchB y SwitchC…

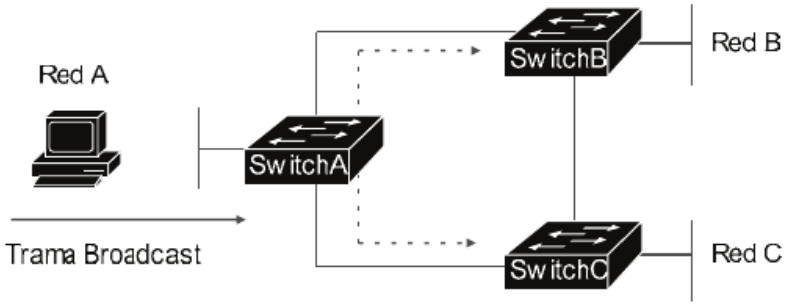


Fig. 3-1.1 Tormenta de broadcast.

Paso 2: La trama es recibida por los SwitchsB y C, la procesan, y al ser un broadcast actúan exactamente igual, reenviándola por todas las interfaces excepto por la cual fue recibida, lo que incluye el enlace entre ambos y sus respectivas redes…

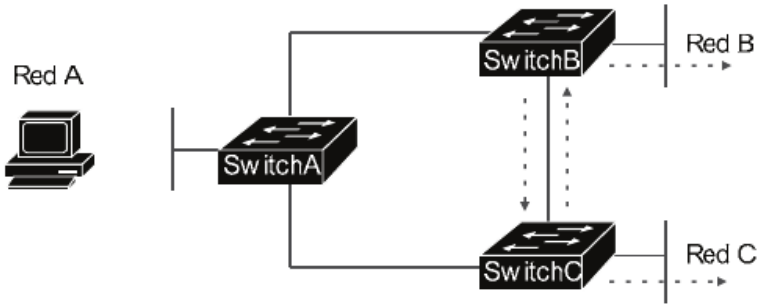


Fig. 3-1.2 Tormenta de broadcast.

Paso 3: Tanto B como C han vuelto a recibir un broadcast, procediendo ambos de la misma manera que anteriormente…

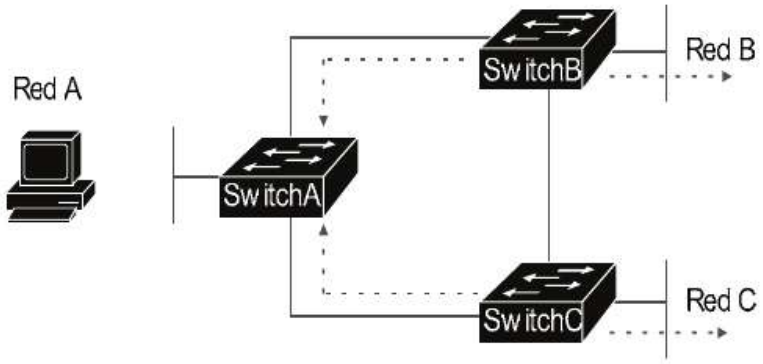


Fig. 3-1.3 Tormenta de broadcast.

Llegados a este punto, los pasos 1, 2 y 3 se repetirán de manera infinita. Si no se soluciona el problema, el ancho de banda disponible en la red, la capacidad de procesamiento de los switchs y el rendimiento de los hosts disminuirán considerablemente.

Se ha generado un bucle de capa 2 en una topología tan sencilla como la recién mostrada, compuesta tan solo por 3 dispositivos que hacen uso de enlaces redundantes. Imagina un entorno corporativo, con decenas o cientos de Switchs conectados entre sí, el problema resultaría incontrolable. Un detalle a tener en cuenta es que las tormentas de broadcast suelen originarse con los siguientes tipos de tramas:

- *Broadcast*: Cuando un switch recibe una trama broadcast la reenvía a través de todas sus interfaces excepto por la cual fue recibida, lo que incluye los enlaces redundantes hacia otros switchs, pudiendo crear un bucle, tal y como se analizó en el ejemplo.
- *Unicast con dirección de destino desconocida:* Una trama unicast es aquella cuyo destinatario es un solo dispositivo. En este caso, los switchs leen su dirección de destino, la buscan en la tabla de MACs y la reenvían únicamente por la interfaz asociada. Sin embargo, si la MAC no se encuentra registrada, la trama será reenviada a través de todas las interfaces, al igual que sucede con los broadcasts, generando también bucles de capa 2.

El término *tormenta de broadcast* se aplica por igual a los bucles generados por ambos tipos de tramas.

Continuando con el ejemplo anterior, para solucionar el problema se puede proceder de dos maneras, bien desconectando físicamente algún enlace que intervenga en el bucle, o bien apagando y encendiendo la interfaz

administrativamente con los comandos *shutdown* y *no shutdown.* Por ejemplo, se podría aplicar cualquiera de las dos acciones sobre el enlace entre el SwitchA y B y el bucle se detendría. Aun así, esta solución sería temporal, ya que una vez conectado el cable o habilitada la interfaz se volverán a formar tormentas de broadcast ante cualquier trama de las ya mencionadas.

Entonces, ¿cada vez que se genere un bucle se debe proceder de esta manera? Evidentemente no, resultaría imposible administrar una red de estas características. Para poner fin a este problema nace el protocolo STP (*Spanning Tree Protocol - IEEE 802.1D*) cuya función consiste en evitar bucles de capa 2 de manera automática mediante el bloqueo de enlaces redundantes.

La misma topología, aplicando STP, podría quedar definida como:

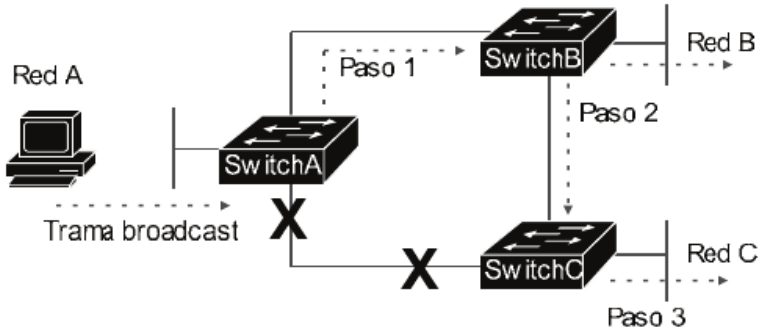


Fig. 3-2 Bloqueo de enlaces por STP.

En este caso se ha aplicado STP y el protocolo ha bloqueado automáticamente el enlace entre el SwitchA y C para evitar tormentas de broadcast. Con ello, la comunicación se llevaría a cabo de la siguiente manera:

Paso 1: SwitchA recibe la trama broadcast y la reenvía a través de todas sus interfaces excepto por la cual fue recibida. Como el enlace entre A y C está bloqueado por STP, solo será reenviada hacia B.

Paso 2: SwitchB recibe la trama, la procesa y reenvía por todas las interfaces excepto por la recibida, lo que incluye la red B y el enlace con C.

Paso 3: Por último, el SwitchC recibe la trama, la procesa y tan solo la reenvía hacia la red C. Ello es debido a que el enlace entre C y A está bloqueado por STP y el de C y B recibió la trama, por lo tanto, no se reenvía a través del mismo.

Gracias a ello se evita la formación de bucles de capa 2 pero por contra se inutiliza un enlace. En el caso de que el link entre A y B caiga, el protocolo activa automáticamente el de A y C, por lo que la comunicación entre las diferentes redes no se verá afectada.

En resumen, STP es un protocolo de capa 2 utilizado para evitar tormentas de broadcast mediante el bloqueo de enlaces redundantes.

Este protocolo no solo es utilizado por switchs, también los bridges lo aplican. ¿Por qué? Porque ambos son dispositivos de capa 2.

MODO DE OPERAR DE STP

En STP, cada switch asume un rol y cada interfaz es definida mediante un tipo y estado, siendo determinados cada uno de ellos en relación con el cálculo realizado por el algoritmo STA (*Spanning-Tree Algorithm*), el cual analiza una serie de factores con el objetivo final de seleccionar rutas libres de bucles hacia cada destino, bloqueando enlaces redundantes si fuera necesario.

La operación llevada a cabo por el protocolo se basa en definir qué rol tomará cada switch y qué estado asumirá cada interfaz. Así pues, la siguiente sección abordará cada escenario por separado, para acto seguido finalizar con un supuesto práctico.

Roles del switch

El proceso comienza con la elección de un dispositivo que será utilizado como núcleo de la topología STP. Este recibe el nombre de puente raíz (*root bridge*) y con relación al mismo serán calculadas las rutas hacia los diferentes destinos. El puente raíz es seleccionado a través de una negociación llevada a cabo entre todos los switchs de la topología, dando como resultado aquel que tomará dicho rol. Este proceso es bastante sencillo y se basa en dos valores, la prioridad y la dirección MAC. El primero es un número decimal que, como su nombre indica, otorga una prioridad en STP al dispositivo. El valor por defecto en switchs Cisco es 32768, pudiendo ser modificado manualmente. La dirección MAC, ya analizada, es un valor hexadecimal no configurable y único a nivel global. El conjunto de ambos recibe el nombre de Bridge ID (BID), con formato "Prioridad:MAC" y es el campo utilizado durante la negociación entre los switchs. Un ejemplo de BID podría ser 32768:0000.1111.2222, donde 32768 es la prioridad y 0000.1111.2222 hace referencia a la MAC.

Con ello, la selección se llevará a cabo conforme al siguiente criterio:

- 1.- El puente raíz será aquel cuyo valor de prioridad incluida en el BID sea la menor de todos los switchs que componen la topología STP.
- 2.- Si varios de ellos coinciden en el menor valor de prioridad, la selección se llevará a cabo en relación con la MAC, donde aquel con valor más bajo será considerado puente raíz.

Resulta importante tener en cuenta dicho criterio de elección, ya que a través del Switch resultante circulará gran cantidad de tráfico, por lo que se recomienda que su rendimiento sea superior al resto de dispositivos, dotándolo para ello de mayor capacidad de memoria y procesador. ¿Cómo lograr que un dispositivo en concreto sea seleccionado como puente raíz? Sencillo, configurando manualmente su prioridad de tal manera que sea menor que cualquier otra.

Para que el proceso se lleve a cabo de forma correcta todos los switchs deben conocer el BID de los restantes. Para ello, intercambian una serie de mensajes, los cuales simplemente son tramas con un formato definido y denominadas BPDU (*bridge protocol data units).* Estas incluyen diferentes campos, entre los que se encuentran:

- Root Bridge BID: Hace referencia al valor BID del switch que ha sido seleccionado como puente raíz.
- Bridge BID: En este campo el switch incluye su propio BID.
- Root Cost: Es un valor decimal que indica el coste total desde el propio Switch hasta el puente raíz. Este varía dependiendo de la velocidad y número de enlaces necesarios para llegar al él. Será analizado en párrafos posteriores.

Con las BPDU creadas, el intercambio de mensajes y elección del puente raíz consta de los siguientes pasos:

Paso 1: Lo primero que hace cada switch es crear su propia BPDU. En este paso aún no han recibido ningún mensaje de switchs vecinos, por lo tanto, cada uno se nombra a sí mismo como puente raíz. En el ejemplo todos aplicarán la prioridad por defecto, 32768, ya que en ninguno de ellos ha sido modificada manualmente.

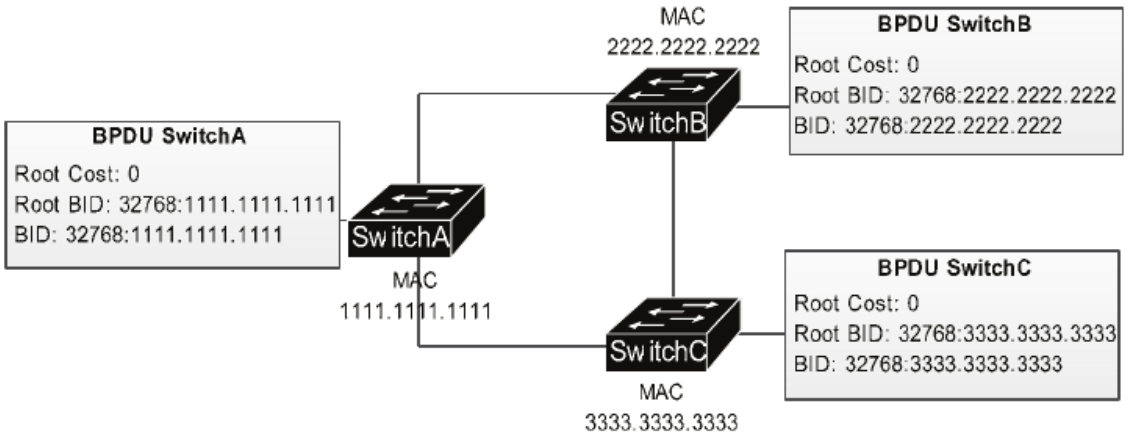


Fig. 3-3.1 Proceso de elección del puente raíz.

Centremos especial atención a los campos de algún dispositivo en concreto, por ejemplo, B.

- *Root Cost*: El valor es 0 porque en este paso el switch se autoproclama puente raíz por lo que el coste para llegar a sí mismo siempre será 0. La misma teoría aplican los switchs A y C.
- *Root BID:* Al autodefinirse como puente raíz, el valor de este campo estará compuesto por su propio BID. En este caso 32768:2222.2222.2222. La misma teoría aplican A y C.
- *BID*: Su propio BID.

Paso 2: Cada switch envía su propia información al medio a través de todas sus interfaces. Existen diferentes tipos de tramas BPDU utilizadas por STP, pero estas en concreto reciben el nombre de "*hello BPDU*".

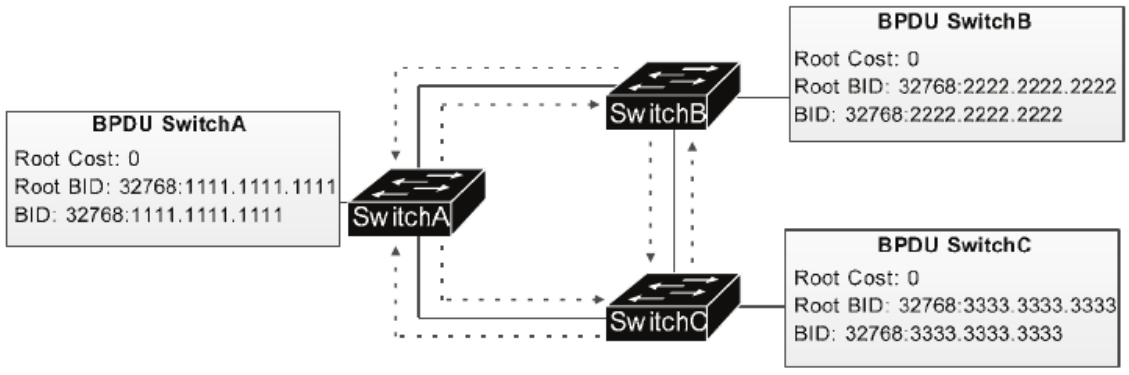


Fig. 3-3.2 Proceso de elección del puente raíz.

Paso 3: Cada switch recibe las tramas de los restantes y las compara con la suya propia en busca de un BID menor. En caso de que exista, automáticamente es considerado como puente raíz el switch desde el cual fue recibido. Continuando con el ejemplo, el SwitchB recibe dos mensajes con BIDs 32768:3333.3333.3333 (SwitchC) y 32768:1111.1111.1111 (SwitchA). Estos los compara con su propia BPDU creada en el paso 1, aplicando el siguiente criterio.

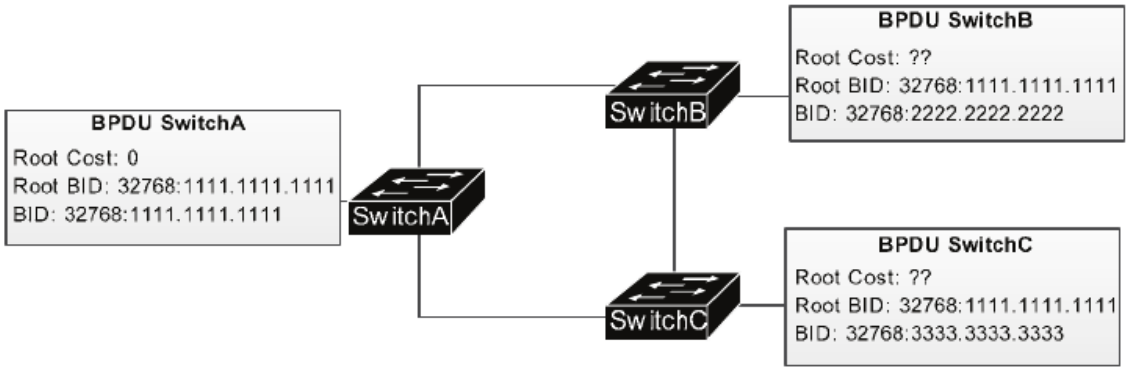


Fig. 3-3.3 Proceso de elección del puente raíz.

Paso 4: Tanto B como C incluyen el BID de A en el campo *"Root BID"* de sus tramas BPDU. El siguiente paso consiste en que ambos calculen el coste total hasta el nuevo puente raíz y seleccionen la mejor ruta hacia el mismo. Como ya se ha nombrado en párrafos anteriores, el coste es un valor decimal que se basa en la velocidad y cantidad de enlaces desde el switch local hasta el destino. Por ejemplo, B dispone de dos rutas para llegar hasta A, una mediante un enlace directo y otra a través de C. Calculados los costes de ambas aplicará la mejor de ellas.

Los valores en cada caso dependerán de la velocidad de cada enlace, definidos por IEEE y siendo los siguientes.

Velocidad de enlace	**Coste**
10 Mbps	100
100 Mbps	19
1 Gbps	4
10 Gbps	2

Supongamos que la red del ejemplo opera a 100 Mbps. ¿Qué costes calcularían y cuál de ellos aplicarían los SwitchsB y C?

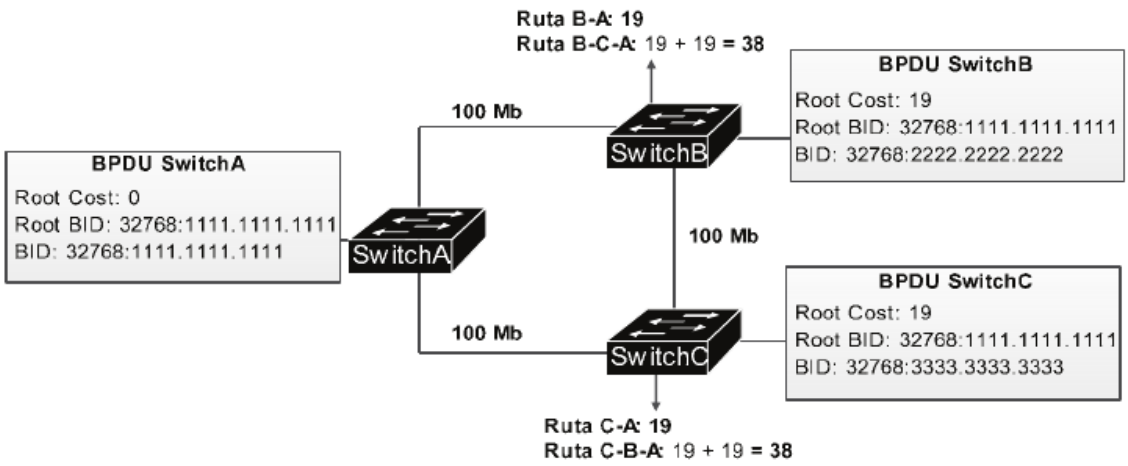


Fig. 3-4 Costes de enlace en STP.

En este caso, la mejor ruta para ambos es el enlace que conecta directamente con A, con coste 19, por lo tanto, es la que agregan a sus BPDUs. Llegados a este punto se ha seleccionado un puente raíz y los demás miembros de la topología han calculado la mejor ruta hacia él, por lo tanto, la convergencia STP ha concluido satisfactoriamente. De ahora en adelante el único dispositivo que envía mensajes *"hello BPDU"* será el puente raíz, los demás simplemente se limitan a reenviarlos a través de todas las interfaces excepto por la cual fue recibido.

Si se produjera cualquier cambio en la topología, como eliminar o agregar un nuevo switch, modificar la prioridad de alguno de ellos o cambiar la velocidad de los enlaces, se ejecutaría un nuevo proceso de convergencia y cálculo de rutas. Debe prestarse especial atención a estas situaciones, en una topología bien diseñada el switch que actúa como puente raíz tendrá mayor capacidad que los restantes, forzando su elección mediante la modificación de su prioridad. La inclusión de un nuevo dispositivo debe llevarse a cabo asegurando que su BID sea mayor, de lo contrario se corre el riesgo de que el puente raíz cambie, hecho que a su vez puede acarrear problemas como mayor lentitud en las comunicaciones, procesamiento más lento o uso de rutas más largas.

Tipos y estado de interfaz

Una vez seleccionado el puente raíz, el segundo paso llevado a cabo por STP consiste en definir el rol de cada interfaz, el cual determinará el modo de operar de estas. Existen 3 tipos.

Puerto raíz (*Root port*): Hace referencia a la interfaz utilizada para llegar al puente raíz. En cada switch tan solo puede existir un puerto de este tipo, exceptuando el

seleccionado como *root bridge*, donde ninguno adoptará este rol porque ninguna de sus interfaces conecta con el propio switch. Los puertos raíz transmiten tráfico y operan con normalidad.

Puerto designado (*Designated port*): son las interfaces activas, que transmiten tráfico y operan con normalidad pero que no son puertos raíz. En cada enlace debe existir un puerto designado.

Puerto bloqueado o no designado (*non-designated port*): son las interfaces bloqueadas por STP para evitar bucles de capa 2. Estas no transmiten tráfico.

Conforme a ello, ¿qué rol tomará cada interfaz en cada switch de la topología utilizada como ejemplo?

Recapitulando la información ya obtenida:
- El SwitchA es el puente raíz porque su BID es menor que el de B y C.
- Tanto B como C seleccionaron como mejor ruta hacia el puente raíz el enlace que les une directamente (B-A y C-A).

SwitchA: En el switch que actúa como puente raíz todas las interfaces asumen obligatoriamente el rol de puerto designado. Esto es debido a que deben estar activas y transmitiendo con normalidad. Por lo tanto:

- Fa0/0: Puerto designado.
- Fa0/1: Puerto designado.

SwitchB: Dispone de dos enlaces, uno que conecta directamente con el puente raíz (B-A) y otro cuya ruta ha sido descartada por tener un coste mayor que la anterior (B-C-A). La interfaz utilizada para llegar al puente raíz tomará el rol de puerto raíz, mientras que la restante iniciará una negociación con el fin de determinar qué rol adoptará.

- Fa0/11: Puerto raíz.
- Fa0/12: Negociación (analizada a continuación).

SwitchC: Este también dispone de dos enlaces, uno directo hacia el puente raíz y otro a través de B. El cálculo de costes determinó que el enlace directo entre A y C es la mejor ruta hacia el puente raíz, por lo tanto, la interfaz que conecta directamente con A tomará el rol de puerto raíz, mientras que la otra negociará qué rol tomar.

- Fa0/20: Puerto raíz.
- Fa0/21: Negociación (analizada a continuación).

Por lo pronto se han definido los siguiente roles:

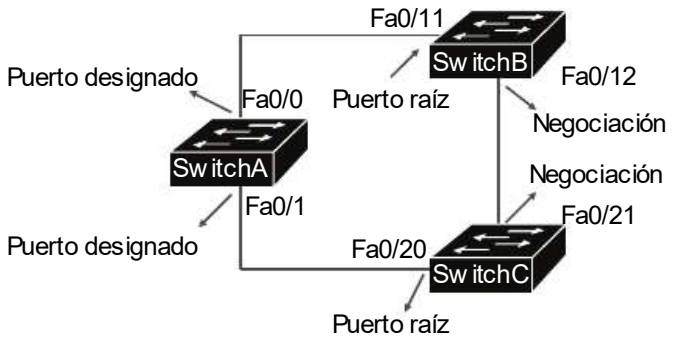


Fig. 3-5.1 Roles de interfaz en STP.

Tan solo faltaría por calcular el enlace entre B y C, el cual será bloqueado para evitar bucles de capa 2. La definición de "puerto designado" señala que en cada enlace debe existir una interfaz con dicho rol, por lo tanto, entre ambos debe establecerse un extremo como "designado" y otro como "no-designado (bloqueado)", pero ¿qué rol tomará cada interfaz? Para ello se lleva a cabo una negociación bastante sencilla basada en los siguientes criterios:

1.- Se comparan los costes de cada uno de los switchs hacia el puente raíz a través de la interfaz que se está negociando, aquella con mayor valor será bloqueada. Siendo la restante la que asume el rol de puerto designado.

2.- Si ambos costes resultaran ser iguales, se comparan los BID de los switchs, donde aquel con un valor mayor bloqueará su interfaz, mientras que el restante aplicará el rol de puerto designado.

Que aplicados sobre el ejemplo:

1.- Se comparan los costes de cada switch hacia el puente raíz a través de la interfaz que se está negociando. En ambos casos, ya calculados en párrafos anteriores, el valor es igual a 38, por lo tanto, la negociación continúa, aplicando el segundo criterio.

2.- Se comparan los BID de ambos switchs, y el que tenga un valor mayor bloqueará su interfaz. El de B es 32768:2222.2222.2222, mientras que el de C 32768:3333.3333.3333, por lo tanto, este último bloqueará su interfaz Fa0/21 asignándole el rol de puerto no-designado, mientras que Fa0/12 del SwitchB quedará definida como designado.

Concluida la negociación la topología quedaría definida de la siguiente manera:

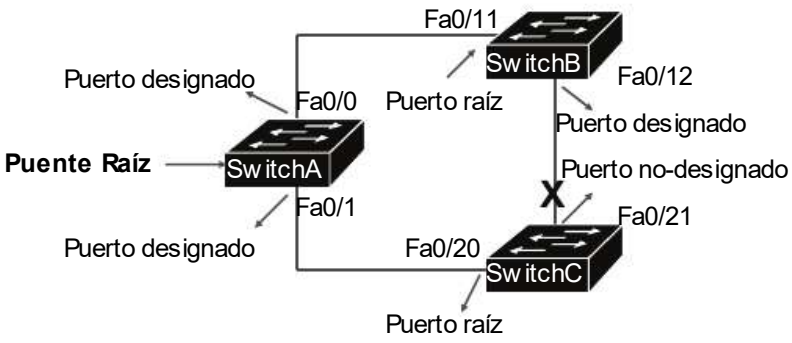


Fig. 3-5.2 Roles de interfaz en STP.

Por último, ¿qué sucedería si se agregara un nuevo enlace redundante entre, por ejemplo, A y C? ¿Qué interfaz tomará el rol de no designada? En estos casos la solución es mucho más sencilla, STP bloqueará uno de los dos enlaces para evitar bucles. ¿Cuál de ellos? En el puente raíz todos las interfaces toman el rol de puerto designado obligatoriamente, por lo que el bloqueo se producirá en el SwitchC. En estos casos, la decisión se toma simplemente basándose en el número de puerto, donde la interfaz con la numeración más alta será bloqueada, en este caso la Fa0/25.

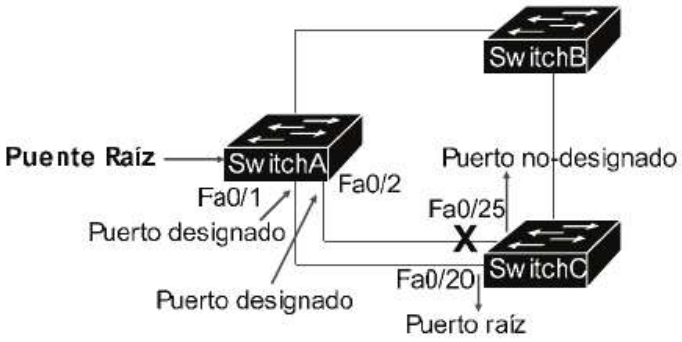


Fig. 3-5.3 Roles de interfaz en STP.

Una vez establecidos los roles, el tercer y último paso ejecutado por STP consiste en definir el estado de las interfaces, el cual estará estrechamente vinculado al rol adoptado con anterioridad. Existen 4 modos de operación: *Forwarding* (enviando), *blocking* (bloqueado), *listening* (escucha) y *learning* (aprendiendo). Evidentemente, los puertos designados y puertos raíz harán uso del estado *forwarding* ya que deben ser activos y enviar tráfico con normalidad, mientras que los puertos no-designados aplicarán el estado *blocking*, ya que por ellas no se debe enviar ni recibir tráfico.

Sin embargo, durante la convergencia de STP o ante cualquier cambio en la topología las interfaces pueden cambiar de un rol a otro con facilidad, y con ello de estado. Una en *forwarding* puede cambiar a *blocking* inmediatamente, no así a la inversa, ya que de *blocking* a *forwarding* se debe ejecutar un proceso de transición que implica el paso por todos los estados: primero a *listening*, luego a *learning* y por último a *forwarding*…

Sus funciones son:

Listening: El switch no envía ninguna trama a través de la interfaz, simplemente borra la tabla de MACs anterior con el fin de crear una nueva. Si no lo hiciera se correría el riesgo de formar bucles de capa 2.

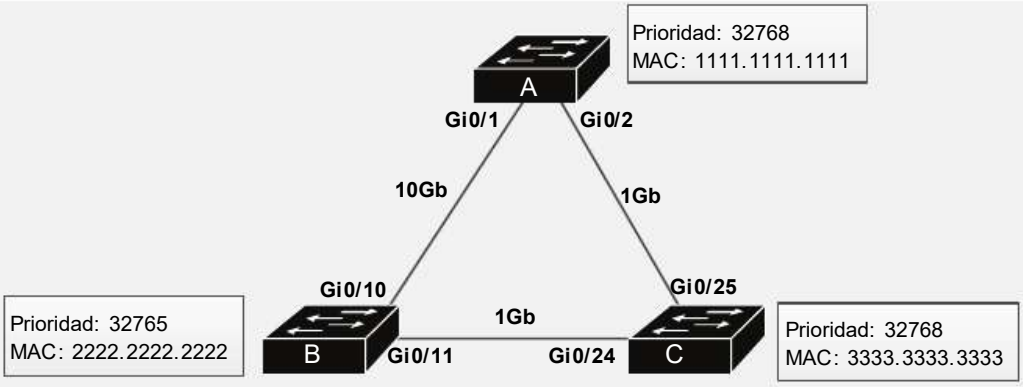
Learning: El switch aún no envía ninguna trama a través de la interfaz, pero examina las que recibe con el objetivo de ir completando la nueva tabla de MACs.

Una vez concluido este proceso, la interfaz cambia a *forwarding* y envía y recibe tramas con normalidad. El tiempo que permanece en cada uno de estos dos estados es de 15 segundos por defecto, por lo que el tiempo que tarda una interfaz en pasar de bloqueada a activa es de 30 segundos.

Las operaciones llevadas a cabo en cada caso son:

Estado	**Función**
Blocking	Interfaz bloqueada. No envía ni recibe tramas. Es un estado estable.
Listening	No se reciben ni envían tramas, su función es borrar la tabla de MACs almacenada en el switch. Es un estado transitorio.
Learning	No se envían tramas pero sí se reciben con el fin de actualizar la tabla de MACs. Es un estado transitorio.
Forwarding	La interfaz envía y recibe tráfico con normalidad. Es un estado estable.

Reto 3.1 – En la siguiente topología y una vez finalizada la convergencia en STP, ¿qué switch tomará el rol de puente raíz? ¿Qué rol tomarán las interfaces de cada uno de los Switchs?



Solución al final del capítulo.

RSTP (Rapid-STP)

RSTP (protocolo IEEE 802.1w) nace como mejora del recién analizado STP. El funcionamiento en ambos resulta bastante similar, es decir, la elección del puente raíz, roles y estados de las interfaces se basan en los mismos criterios, sin embargo, la convergencia se realiza de forma mucho más rápida en el nuevo protocolo, tardando tan solo 10 segundos en completarla, siendo en STP, y con la configuración por defecto, de 50 segundos (una convergencia tan lenta puede provocar bucles de capa 2 temporales).

Otra de sus características es que es compatible con su antecesor. En una topología se puede hacer uso de ambos de manera simultánea, siendo un detalle importante ya que en algunos modelos de switchs no es posible implementar RSTP debido a su antigüedad o versión de IOS.

Su modo de operar, configuración y detalles técnicos no forman parte del contenido de CCNA, siendo analizado en certificaciones más avanzadas como CCNP.

CONFIGURACIÓN Y ASPECTOS DE SEGURIDAD

El procedimiento de configuración de STP puede ser dividido en 5 pasos:

- *Paso 1:* Diseño de la topología STP.
- *Paso 2:* Modo de STP.
- *Paso 3:* Configuración de prioridad en los switchs.
- *Paso 4:* Configuración de costes de los enlaces.
- *Paso 5:* Configuración de Portfast y BPDUGuard.

Paso 1: Diseño de la topología STP

El diseño de la topología STP hace referencia a la ubicación de los Switchs y enlaces entre los mismos. Este paso no representa ningún elemento propio de configuración, pero sí que resulta imprescindible para implementar esta de manera correcta y eficiente.

En párrafos anteriores se ha hecho mención a la necesidad de que el puente raíz disponga de características de hardware superiores ya que deberá procesar gran cantidad de tráfico. Además, el resto de switchs de la topología calcularán sus rutas en relación con él, las cuales, cuantos menos enlaces dispongan, mejor. Para lograr dicho objetivo es necesario que su ubicación resulte lo más "céntrica" posible, logrando además un mejor aprovechamiento del ancho de banda disponible.

Conforme a ello se puede concluir que el lugar idóneo para situar el puente raíz es la capa de distribución, primero, porque actúa como intermediaria entre las capas de acceso y núcleo, y segundo, porque los switchs ubicados en la misma son considerados elementos críticos de red, por lo que deben disponer de capacidades superiores al resto.

Sin embargo, dicha capa suele estar compuesta por numerosos switchs. ¿Cuál seleccionar? Aquel que el administrador de red considere más oportuno. Lo importante es forzar a STP a su elección, y en caso de que este caiga, que otro con las mismas o similares prestaciones tome su rol, tarea que se lleva a cabo modificando la prioridad, analizada en el paso 3.

Paso 2: Modo de STP

STP dispone de diferentes modos de operar, los cuales han sido desarrollados gracias al avance de nuevas tecnologías y sobre todo debido a la aparición de las VLANs. Uno de ellos es PVST+ (*Per VLAN STP plus, también denominado PVSTP)* cuya metodología se basa en implementar una topología STP por cada VLAN existente en la red. De la misma manera, para RSTP se desarrolla un modo mejorado denominado RPVST+ *(Rapid Per VLAN STP plus, o RPVSTP),* también definiendo una topología por VLAN. Por último, MST *(Multiple Spanning Tree)* se convierte en la tercera variante del protocolo.

De todos ellos, PVST+ es el modo aplicado por defecto en la mayoría de switchs Cisco, pudiendo ser modificado mediante los siguientes comandos, ejecutados desde el modo de configuración global:

- Modo PVST+: spanning-tree mode pvst
- Modo RPVST+: spanning-tree mode rapid-pvst
- Modo MST: spanning-tree mode mst

PVST+ y RPVST+ son propietarios de Cisco, por lo tanto, solo estarán disponibles y podrán ser utilizados en switchs de este fabricante.

RPVST+ y MST no son contenido de CCNA.

PVST+ se basa en crear una topología STP por cada VLAN existente en la red. Este hecho en la práctica se traduce en una serie de mejoras, entre las que cabe destacar un mejor aprovechamiento del ancho de banda, lo cual también puede ser considerado como un balanceo de carga. Por ejemplo, un enlace bloqueado por STP para una determinada VLAN puede ser utilizado en la topología de otra. Además, es posible reducir la carga de trabajo del puente raíz al poder seleccionar uno diferente para cada VLAN.

Es decir, PVST+ crea diferentes topologías STP, siendo cada una de ellas totalmente independiente, con su propio puente raíz, costes y rutas. Es por ello que cada switch dispondrá de un BID por cada VLAN, lo que permite modificar sus prioridades para lograr la elección de diferentes puentes raíz.

Ver supuesto práctico en el apartado “Ejemplo de configuración y verificación de STP”.

Paso 3: Configuración de prioridad en los switchs

Una de las acciones más importantes a la hora de configurar STP consiste en influir de manera manual en la elección del puente raíz. Conociendo que por defecto todos los switchs Cisco aplican la prioridad 32768 bastará con asignar valores menores al indicado. Cabe recordar que en PVST+ se debe configurar una prioridad por cada VLAN existente. El comando necesario para ello es **spanning-tree vlan *[vlan id]* priority *[prioridad]***, desde el modo de configuración global, donde *[vlan id]* indica el número de VLAN a la cual se aplicará la prioridad STP, mientras que *[prioridad]* hace referencia al valor decimal que se desea asignar, el cual debe estar comprendido entre 0 y 65535.

Gracias a ello se asegura que aquel configurado con una prioridad más baja que los demás para una determinada VLAN será seleccionado como puente raíz.

Sin embargo, Cisco ofrece otro método de configuración aún más sencillo, basado en configurar un switch para que actúe como puente raíz y otro como secundario sin necesidad de modificar la prioridad manualmente. Para ello son necesarios los siguientes comandos:

1.- Para el puente raíz se debe aplicar la sentencia **spanning-tree vlan *[vlan id]* root primary,** desde el modo de configuración global. ¿Qué acción conlleva? La elección del puente raíz se lleva a cabo comparando todos los BID recibidos. Bien, el switch lo que hará será analizar todas las prioridades y asignarse automáticamente una menor que todas ellas, así se convertirá en puente raíz para la topología de la VLAN seleccionada.

2.- Para el switch que se desee como secundario se debe aplicar la sentencia **spanning-tree vlan *[vlan id]* root secondary,** desde el modo de configuración global. El funcionamiento es prácticamente igual que el explicado anteriormente, con la única diferencia de que en este caso el switch se asignará un valor de prioridad mayor que el del puente raíz y a su vez menor que el del BID más bajo del resto de switchs de la topología.

De esta manera, aunque se introduzcan nuevos switchs o cambios en la topología STP, aquel configurado como *root primary* siempre modificará su prioridad para que durante la convergencia vuelva a ser seleccionado como puente raíz.

Paso 4: Configuración de costes de enlace

Otra de las configuraciones llevadas a cabo en STP consiste en asignar manualmente el coste de los enlaces con el fin de influir en la selección de rutas y con ello en los roles adoptados por cada una de las interfaces intervinientes. Para cada velocidad existe un valor de coste predefinido, siendo las rutas de cada uno de los Switchs hacia el puente raíz calculadas en relación con la suma total de dichos valores, ya analizados en párrafos anteriores.

Si se deseara modificar el coste de cualquier enlace, se debe aplicar el comando **spanning-tree vlan *[vlan id]* cost *[costo]***, desde el modo de configuración de la interfaz que establece el enlace en cuestión. *[vlan id]* indica el número de VLAN y por lo tanto la topología STP en la cual se aplicará, mientras que *[coste]* hace referencia al nuevo valor asignado.

Esta modificación puede resultar de gran utilidad cuando el enlace activo hacia el puente raíz alberga gran cantidad de tráfico y está siendo sobrecargado. En estos casos se puede optar por modificar el coste del enlace redundante que ha sido

bloqueado para que sea utilizado como activo para una o varias VLANs, logrando con ello que el tráfico de estas utilice otra ruta para llegar al puente raíz. Un ejemplo podría ser el siguiente:

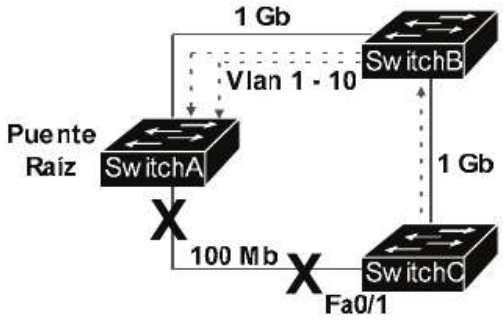


Fig. 3-6.1 Modificación de costes de enlace.

- Existen 10 VLANs, por lo tanto, se han creado 10 topologías STP.
- El SwitchA ha sido seleccionado como puente raíz de todas ellas, por lo que los demás miembros deberán calcular la mejor ruta hacia él, basándose para ello en el coste de los enlaces.
- El SwitchB selecciona como ruta principal el enlace directo con A (ruta B-A) porque su coste (4) es menor que el de la ruta B-C-A (4+19=23).
- El SwitchC selecciona como principal la ruta C-B-A porque su coste (4+4=8) es menor que el de la ruta C-A (coste 19).

Supongamos que en la red se genera gran cantidad de tráfico. Actualmente las 10 VLANs atraviesan el enlace B-A, por lo que puede presentar sobrecarga. Para aliviar el problema se puede configurar el SwitchC para que utilice como ruta principal hacia A, y para determinadas VLANs, el enlace directo. ¿Cómo proceder a ello? El enlace C-A es de 100 mb, con un coste de valor 19 por defecto, mientras que la ruta principal (C-B-A) tiene un valor de 8 (4+4), por lo tanto, se debe lograr que C-A utilice un coste menor de 8 para que sea seleccionada como principal. En este caso, este cambio será aplicado tan solo a las topologías STP de las VLANs 1 hasta la 5 para así llevar a cabo un balanceo de carga con la ruta actualmente en uso.

Aplicando la configuración:

```
SwitchC# configure terminal
SwitchC(config)# interface Fa0/1
SwitchC(config-if)# spanning-tree vlan 1 cost 4
SwitchC(config-if)# spanning-tree vlan 2 cost 4
SwitchC(config-if)# spanning-tree vlan 3 cost 4
SwitchC(config-if)# spanning-tree vlan 4 cost 4
SwitchC(config-if)# spanning-tree vlan 5 cost 4
```

Cuando se modifica el coste de cualquier interfaz inmediatamente se recalculan las rutas hacia el puente raíz y se aplican los cambios oportunos. Todo ello puede ser monitorizado en tiempo real haciendo uso del comando **debug spanning-tree events**, ejecutado desde el modo privilegiado.

```
SwitchC#
*Jan 10 09:16:03.120: STP: VLAN0005 new root port Fa0/1, cost 4
*Jan 10 09:16:03.120: STP: VLAN0005 Fa0/1 -> listening
*Jan 10 09:16:03.120: STP: VLAN0005 sent Topology Change Notice on Fa0/1
*Jan 10 09:16:03.120: STP[05]: Generating TC trap for port GigabitEthernet0/1
*Jan 10 09:16:03.120: STP: VLAN0005 Gi0/1 -> blocking
*Jan 10 09:16:18.434: STP: VLAN0005 Fa0/1 -> learning
*Jan 10 09:16:33.468: STP[05]: Generating TC trap for port FastEthernet0/1
*Jan 10 09:16:33.904: STP: VLAN0005 sent Topology Change Notice on Fa0/1
*Jan 10 09:16:33.907: STP: VLAN0005 Fa0/1 -> forwarding
```

En el ejemplo se puede observar cómo la interfaz Fa0/1 es seleccionada como puerto raíz con un coste de 4, tras lo cual atraviesa los diferentes estados de puerto (listening, learning y forwarding). También como la interfaz Gi0/1 (que antes presentaba el rol de forwarding) es bloqueada de manera inmediata. Esta sucesión de cambios corresponde a la topología STP de la VLAN 5, pero en este caso se aplicaría exactamente lo mismo para el rango 1 - 4.

Con los cambios realizados, el SwitchC redirigiría el tráfico de la siguiente manera:

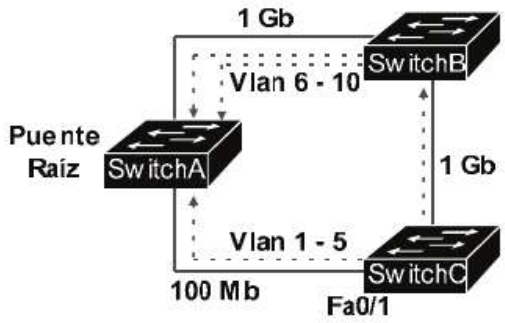


Fig. 3-6.2 Modificación de costes de enlace.

Paso 5: Configuración de Portfast y BPDUguard

Por último, Portfast y BPDUguard son dos características opcionales de STP orientadas a interfaces que conectan con dispositivos finales, cada una de ellas con distintos fines, pero ambas muy recomendables de configurar.

PORTFAST

En párrafos anteriores se ha hecho mención a la duración total de una interfaz en cambiar de estado desde "*Blocking*" hasta "*Forwarding*", siendo de 30 segundos. Este proceso, y el tiempo transcurrido para ello, resulta vital mantenerlo con el fin de evitar bucles de capa 2, sin embargo, dicha recomendación solo es aplicable a enlaces entre switchs. ¿Qué ocurre en una interfaz que conecta con un dispositivo final, por ejemplo, un PC? El PC tardaría 30 segundos en obtener acceso a la red, lo cual, en entornos corporativos puede resultar un tiempo excesivo. Una solución a ello consiste en hacer uso de la característica Portfast de STP.

Su función consiste en que la interfaz en la cual ha sido configurado cambie de bloqueada a activa de manera inmediata, sin necesidad de atravesar los estados transitorios "*Listening"* y "*Learning*", logrando con ello que el enlace opere de forma instantánea. ¿En qué interfaces aplicarlo? En aquellas que solo conecten con dispositivos finales como PCs, impresoras, etc. El motivo es que estos no participan en el proceso de STP ni generan tramas BPDU, por lo tanto, no existe riesgo de que se generen bucles de capa 2.

Portfast no es habilitado por defecto en switchs Cisco. Para llevarlo a cabo se debe aplicar la sentencia **spanning-tree portfast**, ejecutada desde el modo de configuración de la interfaz en cuestión.

También resulta posible habilitarlo en todas las interfaces de manera simultánea mediante el comando **spanning-tree portfast default,** desde el modo de configuración global. Tras ello habría que deshabilitarlo de manera individual en aquellas con enlaces hacia otros switchs, con el comando **spanning-tree portfast disable,** desde el modo de configuración de la interfaz en cuestión.

De ambos métodos, **spanning-tree portfast** es el más común y recomendable.

BPDUGUARD

Uno de los ataques más comunes a *Spanning Tree* consiste en configurar un switch con una prioridad muy baja y conectarlo a la red en alguna interfaz dedicada para dispositivos finales, por ejemplo, desconectar un PC y conectar el nuevo switch. Este proceso dará como resultado un nuevo cálculo y convergencia de STP, siendo muy probable que sea seleccionado como puente raíz. Si esto ocurre, gran parte del tráfico de la red fluirá a través de él, dando como resultado que el atacante pueda capturarlo y con ello obtener información confidencial de usuarios, de la propia compañía y de otros dispositivos de red.

Con el fin de poder evitarlo nace BPDUguard, cuya misión consiste en detener las tramas BPDU recibidas a través de las interfaces donde ha sido configurado, generando una violación de seguridad y bloqueando el acceso cuando ello ocurra. El objetivo consiste en que switchs maliciosos, conectados a interfaces dedicadas a dispositivos finales, no puedan formar parte de STP y mucho menos iniciar una nueva convergencia.

Al igual que Portfast, BPDUguard no es habilitado por defecto en Switchs Cisco, debiendo ser configurado manualmente a través del comando **spanning-tree bpduguard enable**, desde el modo de configuración de la interfaz donde se desee aplicar, o bien habilitarlo de manera simultánea en todas las interfaces con el comando **spanning-tree bpduguard default,** desde el modo de configuración global, para luego deshabilitarlo de manera individual en aquellas con enlaces hacia otros switchs, mediante la sentencia **spanning-tree bpduguard disable,** desde el modo de configuración de la interfaz en cuestión. De ambos métodos, la primera de las opciones resulta la más idónea.

Ejemplo de configuración y verificación de STP

Dada la siguiente topología, se debe configurar STP de tal manera que:

- Todos los switchs operen con PVST+.
- S1 actúe como puente raíz para el rango de VLANs 1-5 y como secundario para el 6-10.
- S2 actúe como puente raíz para el rango de VLANS 6-10 y como secundario para el 1-5.
- Las interfaces Gi0/1 y Gi0/2 de S1 deben tener un coste con valor 2 para las VLANs 1-5.
- Las interfaces Gi0/1 y Gi0/2 de S2 deben tener un coste con valor 2 para las VLANs 6-10.
- El rango de interfaces Fa0/1 – Fa0/20 de todos los switchs de la capa de acceso deben ser configurados con portfast y bpduguard, ya que a ellas conectarán únicamente dispositivos finales.

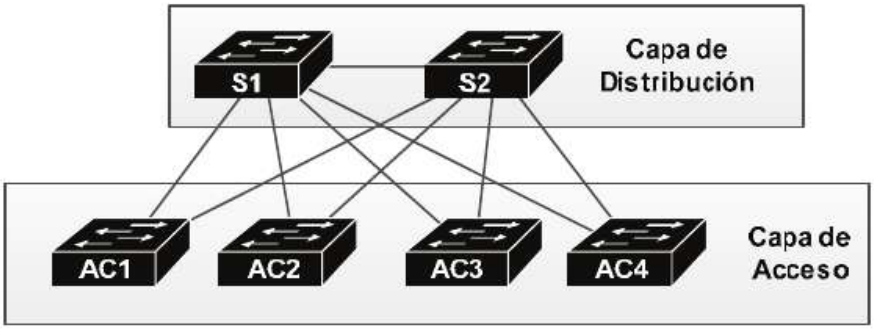


Fig. 3-6.2 Enlaces redundantes entre capas. Aplicación de STP.

1.- Todos los switchs deben operar con PVST+.

El modo PVST+ es habilitado por defecto en switchs Cisco, por lo tanto, no haría falta aplicar ninguna configuración. Aun así, si fuera necesario se debería ejecutar el siguiente comando en todos los switchs.

```
S1# configure terminal
S1(config)# spanning-tree mode pvst

S2# configure terminal
S2(config)# spanning-tree mode pvst

AC1# configure terminal
AC1(config)# spanning-tree mode pvst

AC2# configure terminal
AC2(config)# spanning-tree mode pvst

AC3# configure terminal
AC3(config)# spanning-tree mode pvst

AC4# configure terminal
AC4(config)# spanning-tree mode pvst
```

2.- S1 actúe como puente raíz para el rango de VLANs 1-5 y como secundario para el 6-10.

```
S1# configure terminal
S1(config)# spanning-tree vlan 1-5 root primary
S1(config)# spanning-tree vlan 6-10 root secondary
```

3.- S2 actúe como puente raíz para el rango de VLANs 6-10 y como secundario para el 1-5.

```
S2# configure terminal
S2(config)# spanning-tree vlan 6-10 root primary
S2(config)# spanning-tree vlan 1-5 root secondary
```

4.- Las interfaces Gi0/1 y Gi0/2 de S1 deben tener un coste con valor 2 para las VLANs 1-5.

```
S1# configure terminal
S1(config)# interface range Gi0/1-2
S1(config-if-range)# spanning-tree vlan 1-5 cost 2
```

5.- Las interfaces Gi0/1 y Gi0/2 de S2 deben tener un coste con valor 2 para las VLANs 6-10.

```
S2# configure terminal
S2(config)# interface range Gi0/1-2
S2(config-if-range)# spanning-tree vlan 6-10 cost 2
```

6.- El rango de interfaces Fa0/1 – Fa0/20 de todos los switchs de la capa de acceso deben ser configurados con portfast y bpduguard, ya que a ellas conectarán únicamente dispositivos finales.

```
AC1(config)# interface range Fa0/1-20
AC1(config-if-range)# spanning-tree portfast
AC1(config-if-range)# spanning-tree bpduguard enable

AC2(config)# interface range Fa0/1-20
AC2(config-if-range)# spanning-tree portfast
AC2(config-if-range)# spanning-tree bpduguard enable

AC3(config)# interface range Fa0/1-20
AC3(config-if-range)# spanning-tree portfast
AC3(config-if-range)# spanning-tree bpduguard enable

AC4(config)# interface range Fa0/1-20
AC4(config-if-range)# spanning-tree portfast
AC4(config-if-range)# spanning-tree bpduguard enable
```

Conforme a la configuración aplicada, S1 debería ser el puente raíz de las topologías STP para las VLANs 1-5, mientras que S2 debería serlo para el rango 6-10. Para comprobarlo bastará con ejecutar un **show spanning-tree vlan *[id vlan]***, desde el modo privilegiado.

Por ejemplo, en S1 y para la vlan 2:

```
S1# show spanning-tree vlan 2
VLAN0002
  Spanning tree enabled protocol ieee
  Root ID     Priority    24576
              Address     0001.4326.99B4
              This bridge is the root
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    24576
              Address     0001.4326.99B4
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
```

```
              Aging Time  20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi0/1            Desg FWD 19        128.2    P2p
Gi0/2            Desg FWD 19        128.2    P2p
Fa0/1            Desg FWD 19        128.2    P2p
Fa0/2            Desg FWD 19        128.2    P2p
Fa0/3            Desg FWD 19        128.2    P2p
```

Del resultado obtenido se puede concluir que:

- S1 es actualmente el puente raíz para la topología STP de la VLAN 2.
- La prioridad que se ha aplicado automáticamente tras configurar el comando "*spanning-tree vlan 2 root primary*" tiene un valor de 24576.
- Su dirección MAC es 0001.4326.99B4.
- Las interfaces que forman parte de esta topología en el switch son la "Gi0/1, Gi0/2, Fa0/1, Fa0/2 y Fa0/3", todas ellas con rol de puerto designado y estado "*Forwarding*". (En el puente raíz todas las interfaces activas deben tener el rol de puerto designado).
- El valor de los diferentes contadores como "Hello time", "Max Age", etc.

El mismo resultado es aplicable al rango de VLANs 1-5 en el switch S1, ya que actúa como puente raíz para dichas topologías. Lo mismo sucedería en S2 si el comando es aplicado dentro del rango 6-10.

Sin embargo, ¿qué resultado se obtiene en un switch cuyo rol no sea puente raíz? Por ejemplo, en AC1:

```
AC1#show spanning-tree vlan 2
VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    24576
             Address     0001.4326.99B4
             Cost        19
             Port        24(FastEthernet0/24)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768
             Address     000A.F330.5632
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/24            Root FWD 19        128.1    P2p
Fa0/23            Desg FWD 19        128.1    P2p
Fa0/22            Altn BLK 19        128.1    P2p
Fa0/21            Altn BLK 19        128.1    P2p
```

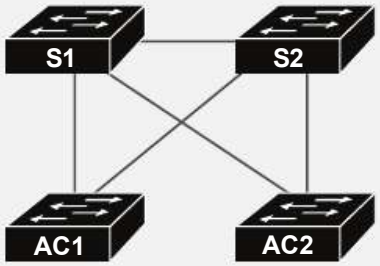
Del cual también se obtienen datos interesantes como:

- La prioridad del switch que actúa como puente raíz para esta topología STP (24576), su dirección MAC (0001.4326.99B4) y el coste total hacia el mismo (19).
- La interfaz local necesaria para llegar hasta el puente raíz (FastEthernet0/24).
- La prioridad del switch local (32768).
- La MAC local (000A.F330.5632).
- El rol y el estado de las diferentes interfaces que forman parte de STP, donde Fa0/22 y Fa0/21 han sido bloqueadas para evitar bucles de capa 2.

Por último, la manera más sencilla de comprobar que portfast y bpduguard se han aplicado según lo previsto consiste en ejecutar un **show running-config**. Del resultado obtenido bastará con revisar la configuración definida en las interfaces.

Reto 3.2 – La siguiente red consta de 6 VLANs, de las cuales se requiere crear una topología STP para cada una de ellas. Los switchs deben ser configurados de tal manera que:

- S1 asigne una prioridad de 12288 para las topologías de las VLANs 1, 2 y 3, y una prioridad de 49152 para las VLANs 4, 5 y 6.
- S2 actúe como puente raíz para las VLANs 4, 5 y 6 y como secundario para el rango 1-3.
- El rango de interfaces Fa0/1-15 de los switchs AC1 y AC2 deben ser configurado con portfast y bpduguard.
- El coste de la interfaz Gi0/1 en los Switchs S1 y S2 debe tener un valor de 3.



Solución al final del capítulo.

ETHERCHANNELS

Una de las mayores desventajas de Spanning Tree consiste en el impedimento de hacer uso de enlaces redundantes de manera simultánea porque generarían bucles de capa 2. STP soluciona este problema bloqueando interfaces, hecho que a su vez acarrea dos consecuencias, primero, el desaprovechamiento de ancho de banda, y segundo, el recálculo de rutas cada vez que se produzca algún cambio en la topología, como la caída del enlace activo.

Con el fin de evitar ambas circunstancias resulta recomendable la configuración de etherchannels, los cuales pueden ser definidos como una tecnología de capa 2 mediante la cual se establece un enlace lógico compuesto por varios enlaces físicos, los cuales operarán como una sola interfaz. Estos también pueden ser denominados como *Portchannel* o *Channel-group.* Por ejemplo…

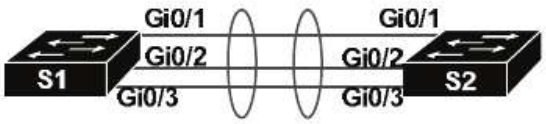


Fig. 3-7 Etherchannel.

Entre S1 y S2 se ha definido un etherchannel a través de las interfaces Gi0/1, Gi0/2 y Gi0/3, de tal manera que a nivel interno se crea una sola interfaz lógica que operará haciendo uso de todas ellas conjuntamente. Algunas de las ventajas obtenidas son:

- Aumento del ancho de banda. En el ejemplo se ha logrado crear un enlace con un ancho de banda de 3 Gb.
- Si un enlace físico cae, el etherchannel continúa operativo haciendo uso de los restantes. Por ejemplo, si la interfaz Gi0/1 fuera desconectada, la comunicación no se vería afectada ni se produciría un re-cálculo de rutas en STP ya que el enlace continúa operativo a través de Gi0/2-3.

Un etherchannel puede ser configurado con un mínimo de 2 y un máximo de 8 enlaces físicos, siempre y cuando estos operen a la misma velocidad. Además, durante el envío de datos, el switch realizará balanceo de carga entre todos ellos. Esta configuración puede ser llevada a cabo de manera manual o automática mediante protocolos de autonegociación, sin embargo, antes de proceder a la misma, y con el objetivo de que el enlace sea creado correctamente, todas las interfaces físicas que intervengan tienen que cumplir los siguientes requisitos:

- La velocidad debe coincidir, por lo que si son de diferente tipo (como FastEthernet y GigabitEthernet) se tendrá que hacer uso del comando **speed** para igualarla.
- Todas deben ser configuradas en modo acceso o en modo trunk. Si están en modo acceso tienen que formar parte de la misma VLAN, mientras que, si están en modo troncal, aplicar la misma VLAN nativa y lista de VLANs permitidas.
- El modo de envío debe coincidir, por defecto full-duplex.
- En todas ellas se deberán aplicar las mismas características de STP.

CONFIGURACIÓN MANUAL DE UN ETHERCHANNEL

Es el método más sencillo de configuración, basado en definir qué interfaces físicas serán agrupadas sobre una lógica, evitando con ello cualquier tipo de autonegociación. Para llevarlo a cabo bastará con aplicar el comando **channel-group *[num]* mode on** sobre cada una de las interfaces que lo conforman, donde *[num]* hace referencia a la interfaz lógica que se creará. Todas aquellas que formen parte del mismo *etherchannel* deberán ser configuradas con el mismo número de *channel-group*. Este solo tiene importancia a nivel local, es decir, el switch del otro extremo podrá hacer uso de un valor diferente y el enlace se establecerá sin problemas.

Aplicado en S1 y S2 *(Fig 3-7)*.

```
S1(config)#interface Gi0/1
S1(config-if)#channel-group 1 mode on
S1(config-if)#
Creating a port-channel interface Port-channel 1

%LINK-5-CHANGED: Interface Port-channel 1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Port-channel 1, changed
state to up

S1(config-if)#exit
S1(config)#interface Gi0/2
S1(config-if)#channel-group 1 mode on
S1(config-if)#exit
S1(config)#interface Gi0/3
S1(config-if)#channel-group 1 mode on



S2(config)#interface Gi0/1
S2(config-if)#channel-group 2 mode on
S2(config-if)#
Creating a port-channel interface Port-channel 2

%LINK-5-CHANGED: Interface Port-channel 2, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Port-channel 2, changed
state to up
```

```
S2(config-if)#exit
S2(config)#interface Gi0/2
S2(config-if)#channel-group 2 mode on
S2(config-if)#exit
S2(config)#interface Gi0/3
S2(config-if)#channel-group 2 mode on
```

Gracias a los cambios realizados se ha establecido un etherchannel entre S1 y S2, haciendo uso de las interfaces Gi0/1, Gi0/2 y Gi0/3. Cualquier modificación sobre su configuración puede ser llevada a cabo directamente a través del comando **interface port-channel *[num]***, en cuyo caso también serán ejecutados automáticamente en cada una de las interfaces físicas que lo conforman.

CONFIGURACIÓN DE UN ETHERCHANNEL MEDIANTE AUTONEGOCIACIÓN

El segundo método de configuración consiste en la autonegociación del enlace. En este caso, ambos extremos harán uso de protocolos encargados de intercambiar una serie de mensajes que culminarán con la creación, o no, del etherchannel. Existen dos opciones para ello, PAgP *(Port agregation protocol)*, el cual es propietario de Cisco y por lo tanto tan solo disponible en sus switchs, y LACP *(Link Agregation Control Protocol)*, desarrollado por IEEE (802.3ad) y aplicable en dispositivos de cualquier fabricante. Las diferencias entre ambos no forman parte del contenido de CCNA, pero sí su configuración, y, sobre todo, cuando se establecerá, o no, el enlace.

La configuración de PAgP se basa en:

Paso 1 (opcional): Habilitar el protocolo mediante el comando **channel-protocol pagp**, desde el modo de configuración de la interfaz.

Paso 2: Agregar la interfaz al channel-group con el comando **channel-group *[num]* mode *[desirable | auto]***. El modo "auto" nunca iniciará una negociación, simplemente se limita a recibir mensajes y aceptar la formación del enlace iniciada por el otro extremo. Por su contra, aquellas en modo "desirable" envían mensajes pagp con el fin de establecer el etherchannel.

Siendo las diferentes combinaciones y sus resultados los siguientes:

PAgP		
Switch 1	**Switch 2**	**Resultado**
auto	auto	No se establece el etherchannel porque en este modo ninguna interfaz iniciaría la negociación.
desirable	auto	Se negociará y creará el etherchannel entre las dos interfaces.
desirable	desirable	Se negociará y creará el etherchannel entre las dos interfaces.

Durante la configuración, el paso 1 es opcional, ya que al aplicar el modo *desirable* o *auto*, el switch automáticamente activa el protocolo PAgP.

Ejemplo de configuración entre S1 y S2 mediante PAgP.

```
S1(config)#interface range Gi0/1-3
S1(config-if-range)# channel-protocol pagp
S1(config-if-range)# channel-group 1 mode desirable


S2(config)#interface range Gi0/1-3
S2(config-if-range)# channel-protocol pagp
S2(config-if-range)# channel-group 2 mode auto
```

La segunda opción disponible, LACP, resulta muy similar a PAgP, también en su modo de configuración:

Paso 1: Habilitar el protocolo mediante el comando **channel-protocol lacp**, desde el modo de configuración de la interfaz.

Paso 2: Agregar la interfaz al channel-group con el comando **channel-group *[num]* mode *[active | passive]***. El modo pasivo nunca iniciará una negociación, pero sí aceptará las iniciadas por el otro extremo. Mientras, el modo activo envía mensajes LACP con el fin de establecer el etherchannel.

Las diferentes combinaciones culminarán con los siguientes resultados.

LACP		
Switch 1	**Switch 2**	**Resultado**
passive	passive	No se establece el etherchannel porque en este modo ninguna interfaz iniciaría la negociación LACP.
active	passive	Se negociará y creará el etherchannel entre las dos interfaces.
active	active	Se negociará y creará el etherchannel entre las dos interfaces.

El mismo ejemplo anterior, esta vez haciendo uso de LACP.

```
S1(config)#interface range Gi0/1-3
S1(config-if-range)# channel-protocol lacp
S1(config-if-range)# channel-group 1 mode active
```

```
S2(config)#interface range Gi0/1-3
S2(config-if-range)# channel-protocol lacp
S2(config-if-range)# channel-group 2 mode passive
```

Reto 3.3 – Se ha configurado un etherchannel entre las interfaces Gi0/1 y Gi0/2 de los switchs S1 y S2 de la siguiente manera:

```
S1(config)#interface range Gi0/1-2
S1(config-if-range)#channel-group 1 mode desirable

S2(config)#interface range Gi0/1-2
S2(config-if-range)#channel-group 1 mode passive
```

Sin embargo, el enlace no se ha establecido. ¿Cuál puede ser la causa del problema y cómo deber ser solucionado?

Solución al final del capítulo.

Reto 3.4 – Se ha configurado un etherchannel entre las interfaces Gi0/1 y Gi0/2 de los Switchs TFE y LPA aplicando los siguientes comandos:

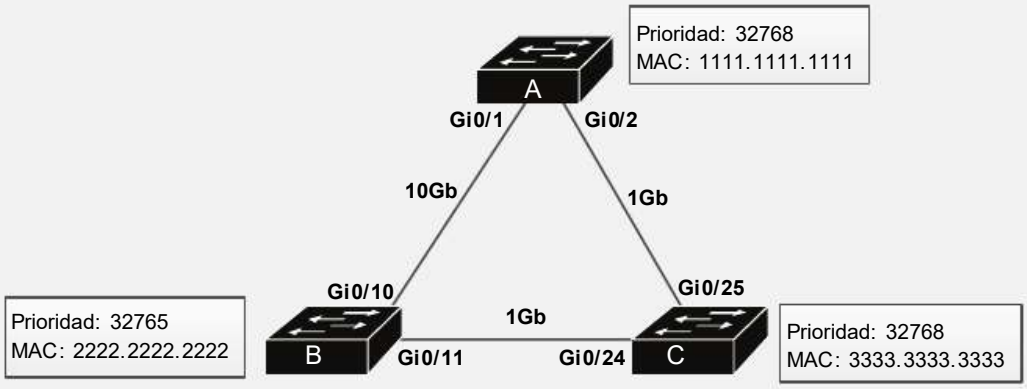
```
TFE(config)#interface range Gi0/1-2
TFE(config-if-range)#channel-group 1 mode on
TFE(config-if-range)#switchport mode trunk
TFE(config-if-range)#switchport trunk allowed vlan 1-20

LPA(config)#interface range Gi0/1-2
LPA(config-if-range)#channel-group 1 mode on
LPA(config-if-range)#switchport mode access
LPA(config-if-range)#switchport access vlan 20
```

Sin embargo, el enlace no se ha establecido. ¿Cuál puede ser la causa del problema y cómo debe ser solucionado?

Solución al final del capítulo.

SOLUCIÓN DE RETOS: STP

Reto 3.1 – En la siguiente topología y una vez finalizada la convergencia en STP, ¿qué switch tomará el rol de puente raíz? ¿Qué rol tomarán las interfaces de cada uno de los switchs?



El primer valor a examinar para seleccionar el puente raíz es la prioridad, siendo el Switch con aquella más baja la que tome dicho rol. En este caso, A y C aplican 32768, mientras que B, 32765, por lo tanto, este último es seleccionado como puente raíz.

Para determinar el rol de cada una de las interfaces se deben calcular los costes de las diferentes rutas disponibles hacia el puente raíz.

- El Switch A dispone de dos rutas. La A-B y la A-C-B. La primera tiene un coste con valor 2 porque es un enlace directo de 10 Gb, mientras que A-C-B tiene un costo de 8 porque atraviesa dos enlaces de 1 Gb cada uno (4+4). La ruta más corta es la primera, por lo tanto, el Switch A seleccionará como puerto raíz la interfaz que conecta directamente con B, que en este caso es la Gi0/1.
- El Switch C también dispone de dos rutas hacia el puente raíz, la C-B y la C-A-B. La primera tiene un coste con valor 4 porque está compuesta por un solo enlace de 1 Gb, mientras que C-A-B tiene un coste de 6 porque atraviesa dos enlaces, uno de 1 Gb y otro de 10 Gb (4+2). El coste de C-B es menor, por lo tanto, la interfaz utilizada para esta ruta será seleccionada como puerto raíz, en este caso, la Gi0/24.
- En el Switch B todas las interfaces toman el rol de puerto designado.

Tanto A como C harán uso del enlace directo como ruta activa hacia el puente raíz, por lo que el link A-C será bloqueado por STP para evitar bucles de capa 2, inhabilitando una de las interfaces intervinientes. Para determinar cuál, el primer

factor a tener en cuenta es el coste total desde cada una de ellas hacia el puente raíz. Aquella con un valor mayor será bloqueada. La ruta A-C-B tiene un coste de 8, mientras que el de C-A-B es de 6. La primera obtiene un valor mayor, por lo tanto, la interfaz Gi0/2 del Switch A será bloqueada aplicando sobre ella el rol de puerto no-designado, mientras que Gi0/25 del Switch C obtendrá el rol de puerto designado.

Con ello, la topología STP queda definida de la siguiente manera:

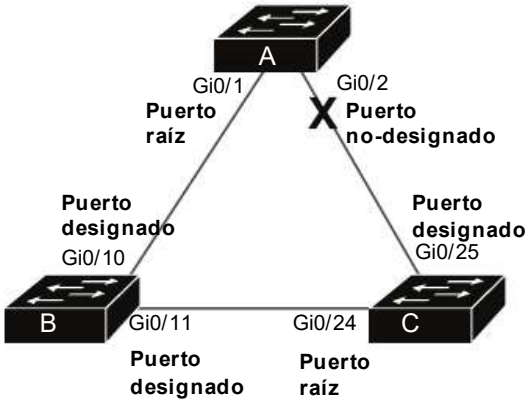
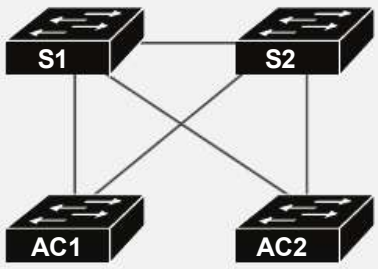


Fig. 3-8 Solución Reto 3.1.

Reto 3.2 – La siguiente red consta de 6 VLANs, de las cuales se requiere crear una topología STP para cada una de ellas. Los switchs deben ser configurados de tal manera que:

- S1 asigne una prioridad de 12288 para las topologías de las VLANs 1, 2 y 3, y una prioridad de 49152 para las VLANs 4, 5 y 6.
- S2 actúe como puente raíz para las VLANs 4, 5 y 6 y como secundario para el rango 1-3.
- El rango de interfaces Fa0/1-15 de los switchs AC1 y AC2 debe ser configurado con portfast y bpduguard.
- El coste de la interfaz Gi0/1 en los Switchs S1 y S2 debe tener un valor de 3.



```
S1(config)#spanning-tree vlan 1-3 priority 12288
S1(config)#spanning-tree vlan 4-6 priority 49152
S1(config)# interface Gi0/1
S1(config-if)#spanning-tree vlan 1-6 cost 3


S2(config)#spanning-tree vlan 1-3 root secondary
S2(config)#spanning-tree vlan 4-6 root primary
S2(config)# interface Gi0/1
S2(config-if)#spanning-tree vlan 1-6 cost 3


AC1(config)#interface range Fa0/1-15
AC1(config-if)# spanning-tree portfast
AC1(config-if)# spanning-tree bpduguard enable


AC2(config)#interface range Fa0/1-15
AC2(config-if)# spanning-tree portfast
AC2(config-if)# spanning-tree bpduguard enable
```

Reto 3.3 – Se ha configurado un etherchannel entre las interfaces Gi0/1 y Gi0/2 de los Switchs S1 y S2 de la siguiente manera:

```
S1(config)#interface range Gi0/1-2
S1(config-if-range)#channel-group 1 mode desirable

S2(config)#interface range Gi0/1-2
S2(config-if-range)#channel-group 1 mode passive
```

Sin embargo, el enlace no se ha establecido. ¿Cuál puede ser la causa del problema y cómo deber ser solucionado?

El problema se encuentra en el protocolo aplicado en ambos switchs para crear el etherchannel. S1 hace uso de PAgP, mientras que S2 de LACP. Es por ello que la negociación no se lleva a cabo y como consecuencia no se establece el etherchannel. Para solucionarlo bastará con definir el mismo protocolo en ambos extremos, por ejemplo, PAgP en modo *desirable* en S1 y *auto* en S2.

```
S1(config)#interface range Gi0/1-2
S1(config-if-range)#channel-group 1 mode desirable

S2(config)#interface range Gi0/1-2
S2(config-if-range)#channel-group 1 mode auto
```

Reto 3.4 – Se ha configurado un etherchannel entre las interfaces Gi0/1 y Gi0/2 de los Switchs TFE y LPA aplicando los siguientes comandos:

```
TFE(config)#interface range Gi0/1-2
TFE(config-if-range)#channel-group 1 mode on
TFE(config-if-range)#switchport mode trunk
TFE(config-if-range)#switchport trunk allowed vlan 1-20

LPA(config)#interface range Gi0/1-2
LPA(config-if-range)#channel-group 1 mode on
LPA(config-if-range)#switchport mode access
LPA(config-if-range)#switchport access vlan 20
```

Sin embargo, el enlace no se ha establecido. ¿Cuál puede ser la causa del problema y cómo debe ser solucionado?

El problema en este caso reside en la configuración de las interfaces que forman parte del etherchannel. En TFE han sido configuradas en modo trunk, mientras que en LPA en modo acceso. Uno de los requisitos para que se establezca el enlace es que coincidan en este parámetro y que además pertenezcan a la misma VLAN (si están en modo acceso) o que incluyan la misma lista de VLANs permitidas (en modo trunk). Para solucionarlo se debe aplicar la misma configuración en ambos extremos. Para el ejemplo se optará por el modo troncal ya que consiste en la operación más común sobre etherchannels.

```
TFE(config)#interface range Gi0/1-2
TFE(config-if-range)#channel-group 1 mode on
TFE(config-if-range)#switchport mode trunk
TFE(config-if-range)#switchport trunk allowed vlan 1-20

LPA(config)#interface range Gi0/1-2
LPA(config-if-range)#channel-group 1 mode on
LPA(config-if-range)#switchport mode trunk
LPA(config-if-range)#switchport trunk allowed vlan 1-20
```

TEST CAPÍTULO 3: SPANNING TREE PROTOCOL

1.- ¿En qué capa del modelo OSI opera Spanning Tree?

A. Capa 1
B. Capa 2
C. Capa 3
D. Capa 4
E. Capa 7

2.- Spanning Tree es definido por IEEE en el estándar…

A. 802.1A
B. 802.1B
C. 802.1C
D. 802.1D

3.- STP se puede definir como…

A. Un protocolo de capa 2 utilizado para prevenir loops de enrutamiento.
B. Un protocolo de capa 3 utilizado con el fin de evitar tormentas de broadcast.
C. Un protocolo de capa 2 cuyo objetivo consiste en aprovechar enlaces redundantes utilizando para ello métodos de balanceo de carga.
D. Ninguno de los anteriores.

4.- ¿Cuáles de los siguientes problemas pueden ocurrir en una red donde no se ha implementado STP? (Seleccionar dos respuestas).

A. Mayor lentitud durante el enrutamiento de paquetes.
B. Redes inaccesibles.
C. Degradación del ancho de banda disponible en la red.
D. Tormentas de broadcast.
E. Mayor dificultad de administración de dispositivos IOS.

5.- STP puede ser configurado en… (Seleccionar dos respuestas).

A. Switchs.
B. Routers.
C. Firewalls.
D. Puntos de acceso.
E. Bridges.
F. Dispositivos finales.

6.- Durante el proceso de convergencia de STP, ¿cuál de los siguientes BID sería seleccionado como puente raíz?

A. 32768:0000.1111.1111

B. 32767:0000.2222.2222
C. 32768:0000.0000.0002
D. 32767:0000.1111.2222

7.- Un BID está compuesto por…

A. MAC más baja
B. Prioridad:MAC
C. MAC:Prioridad
D. Prioridad:IP
E. IP:Prioridad

8.- ¿Qué rol tomarán todas las interfaces del switch seleccionado como puente raíz?

A. Puerto designado.
B. Puerto no-designado.
C. Puerto raíz.
D. Puerto designado o puerto no-designado, pero nunca puerto raíz.

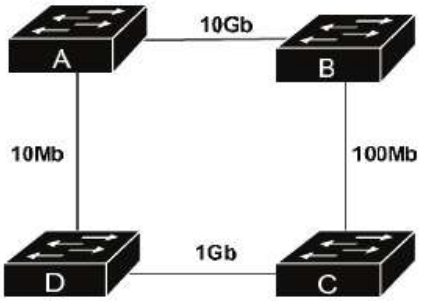
9.- ¿De qué manera se puede influir sobre la elección de un switch como puente raíz? (Seleccionar dos respuestas).

A. Modificando su MAC.
B. Modificando su prioridad.
C. Modificando su IP.
D. Conectando el dispositivo en un lugar estratégico para que los demás switchs de la topología utilicen al menos un enlace hacia él.
E. Aplicando el comando *"spanning-tree vlan [num vlan] root primary".*

10.- ¿Qué nombre reciben las tramas generadas por STP para la comunicación entre los diferentes Switchs de la topología?

A. Multicast.
B. Unicast.
C. Broadcast.
D. BPDU.

11.- Dada la siguiente topología...



¿Qué coste tendrá la ruta A-B-C-D-A?

A. 25
B. 121
C. 104
D. 119
E. 125

12.- En un enlace se debe bloquear una de las interfaces para evitar bucles de capa 2. ¿En qué factores se basa la negociación llevada a cabo entre ambos extremos para determinar qué Switch debe llevar a cabo la acción?

A. Se bloqueará la interfaz cuyo coste hacia el puente raíz sea mayor, y si coincide, se bloqueará aquella cuyo BID sea menor.
B. Se bloqueará la interfaz cuyo coste hacia el puente raíz sea menor, y si coincide, se bloqueará aquella cuyo BID sea menor.
C. Se bloqueará la interfaz cuyo coste hacia el puente raíz sea mayor, y si coincide, se bloqueará aquella cuyo BID sea mayor.
D. Se bloqueará la interfaz cuyo coste hacia el puente raíz sea menor, y si coincide, se bloqueará aquella cuyo BID sea mayor.

13.- Por defecto, ¿cuánto tiempo tarda una interfaz en cambiar de estado de Blocking a Forwarding?

A. La transición se realiza de manera inmediata.
B. 10 segundos.
C. 15 segundos.
D. 30 segundos.
E. 45 segundos.

14.- Una interfaz recibe tramas, analiza sus MACs y las agrega a la tabla, sin embargo, aún no envía tráfico a la red. ¿En qué estado de STP se encuentra?

A. Blocking.

B. Listening.
C. Learning.
D. Forwarding.

15.- Un conjunto de interfaces han sido configuradas con el comando "spanning-tree bpduguard enable". ¿Cuál es el objetivo del administrador de red al aplicar dicha configuración? (Seleccionar dos respuestas).

A. Que las interfaces no intervengan en el proceso STP, aplicando para ello la transición del estado blocking a forwarding de manera inmediata.
B. Que las interfaces no intervengan en el proceso STP, bloqueando la interfaz si recibe alguna BPDU a través de ella.
C. Que los dispositivos conectados a dichas interfaces no puedan participar en el cálculo y convergencia de STP.
D. Que los dispositivos conectados a dichas interfaces no obtengan acceso a la red como medida de seguridad, bloqueando la interfaz si se recibe cualquier tráfico a través de ella.

16.- ¿Qué modo de STP es aplicado por defecto en switchs Cisco?

A. STP
B. RSTP
C. PVST+
D. RPVST+
E. MST

17.- ¿En qué se diferencia RPVST+ y PVST+?

A. RPVST+ crea una topología STP por cada VLAN, mientras que PVST+ no.
B. RPVST+ permite balanceo de carga, PVST+ no.
C. La convergencia en RPVST+ es más rápida que en PVST+.
D. RPVST+ es un protocolo abierto que puede ser utilizado en dispositivos de cualquier fabricante, mientras que PVST+ es propietario de Cisco.

18.- Una red está compuesta por 40 VLANs y 10 switchs Cisco, los cuales aplican el modo PVST+. ¿Cuántas topologías diferentes de STP se crearán?

A. 40.
B. 10.
C. 1.
D. Las que el administrador considere oportunas.

19.- Cada vez que se conecta un dispositivo final a una interfaz tarda 30 segundos en poder acceder a la red. ¿Qué comando se debe aplicar para que la conexión sea inmediata?

A. spanning-tree portfast.
B. spanning-tree bpduguard enable.
C. switchport mode access.
D. switchport mode access vlan [num vlan].

20.- Como máximo, ¿cuántas interfaces físicas pueden formar parte del mismo etherchannel?

A. 10.
B. 9.
C. 8.
D. Ilimitadas.

21.- ¿Cuáles de los siguientes constan como requisitos para que se pueda establecer un etherchannel? (Seleccionar tres respuestas).

A. El modo duplex debe coincidir en todos los enlaces.
B. Los switchs deben operar en la misma versión de IOS.
C. La velocidad debe coincidir en todas las interfaces que formen parte del etherchannel.
D. El número de channel-group configurado en ambos switchs debe ser el mismo.
E. Si las interfaces están configuradas en modo acceso, todas deben pertenecer a la misma VLAN.
F. El cableado de los diferentes enlaces debe utilizar la misma categoría UTP.

22.- Tras ejecutar un "show spanning-tree vlan 5" se obtiene el siguiente resultado…

```
Switch#show spanning-tree vlan 5
VLAN0005
  Spanning tree enabled protocol ieee
  Root ID    Priority    24576
             Address     0001.7626.AAB4
             Cost        100
             Port        24(FastEthernet0/24)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768
             Address     0004.F300.56BB
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20
```

```
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/24           Root FWD 100       128.1    P2p
Fa0/23           Desg FWD 100       128.1    P2p
Fa0/22           Altn BLK 100       128.1    P2p
Fa0/21           Altn BLK 100       128.1    P2p
Fa0/20           Altn BLK 100       128.1    P2p
```

¿Cuántas de sus interfaces participan de forma activa en STP?

A. 5.
B. 3.
C. 2.
D. Ninguna.

23.- A un administrador de red se le ha encomendado la tarea de crear un etherchannel entre dos switchs de diferentes fabricantes, utilizando para ello algún protocolo de autonegociación. De las siguientes, ¿cuál sería la opción correcta a implementar?

A. PAGP.
B. LACP.
C. Configurar todos los enlaces en modo ON.
D. Ninguna de las anteriores.

SUBNETTING EN IPV4 4

INTRODUCCIÓN

En el capítulo 1, "*Redes informáticas. Conceptos básicos",* fue analizada la definición de red como un conjunto de dispositivos conectados y comunicados entre sí con el fin de acceder a los mismos servicios o recursos compartidos, siendo necesario para lograrlo un rango de direcciones común para todos ellos. Dichos rangos suelen ser bastante amplios, hecho que en entornos corporativos supone el desaprovechamiento de direcciones IP, y lo que es peor, una bajada del rendimiento y ancho de banda disponible.

Es por ello que en estos casos la práctica más habitual consiste en la división de la red en segmentos más pequeños, solventando con ello los problemas recién mencionados. Dichos segmentos son las denominadas subredes, las cuales serán objeto de estudio a lo largo del presente capítulo, desde su propósito y conceptos generales hasta su cálculo e implementación.

Las subredes suelen ser definidas conforme a determinados factores, siendo los más comunes:

- Situación geográfica: Segmentar la red dependiendo de la ubicación física. Por ejemplo, una compañía con varias sucursales en diferentes puntos geográficos, crea una subred para cada una de ellas.

- Propósito: Segmentar la red en relación con la función a desarrollar. Por ejemplo, establecer subredes para cada departamento de la compañía, independientemente del lugar físico donde se encuentren ubicados.

- Propiedad: Dividir la red conforme al acceso, permisos o nivel de seguridad. Por ejemplo, una compañía crea una subred para el acceso público a sus servidores web y otra a la que solo puedan acceder sus trabajadores (red privada).

Uno de los principales objetivos de la segmentación consiste en el aprovechamiento de direcciones IP, el cual se basa en ajustar al máximo posible el número total de hosts permitidos por el rango utilizado con el número total de dispositivos que formarán parte de la subred. Un supuesto práctico de ello podría ser el siguiente… Sabiendo que una dirección de clase C permite un máximo de 254 dispositivos, imagina una compañía compuesta por 7 departamentos, con 10 hosts en cada uno de ellos y que por razones de seguridad necesita que formen parte de subredes diferentes. La compañía puede optar por dos métodos. Hacer uso de una dirección de clase C para cada departamento, con lo cual se desaprovecharían 244 direcciones por cada uno de ellos (254 -10) y 1708 en total, o bien utilizar solo una dirección de clase C y dividirla en 7 subredes que permitan un máximo de 10 hosts en cada una de ellas, con lo cual solo se desaprovecharían 184 direcciones en total (254-70).

Segmentar la red requiere tiempo y un estudio detallado sobre las necesidades reales de la compañía y su posterior puesta en práctica. Se deben tener en cuenta diferentes factores entre los que se incluyen: número de subredes necesarias, selección del rango de direcciones e implementación.

Antes de continuar se recomienda volver a leer y practicar los conceptos de "Direccionamiento IP", incluidos en el capítulo 1 "*Redes Informáticas. Conceptos básicos"*.

Número de subredes necesarias

El primer paso a llevar a cabo consiste en realizar un estudio que concluya con el número total de subredes necesarias y hosts que formarán parte de cada una de ellas. Una buena manera de comenzar sería plantear las siguientes cuestiones:

- ¿Qué dispositivos deben ser agrupados en la misma subred?
- ¿Cuántas subredes son necesarias en total?
- ¿Cuántas direcciones IP se requieren en cada una de ellas?

¿Qué dispositivos deben ser agrupados en la misma subred?

Esta distribución depende por completo del administrador o de la política de seguridad de la compañía. Puede definirse una subred por cada sucursal o planta de edificio, aunque lo más lógico y común por aspectos de seguridad consiste en crear una por cada departamento y/o propiedad (servidores que deben ser accesibles desde redes públicas, clientes externos, etc.).

Se debe tener en cuenta que los dispositivos que la conforman no solo hacen referencia a PCs de usuarios, también impresoras, servidores, puntos de acceso, routers, switchs, etc. En definitiva, cualquier elemento que requiera conexión de red.

¿Cuántas subredes son necesarias en total?

Dependerá de la topología implementada. Hay que tener en cuenta que se requiere una por cada VLAN, enlace WAN EoMPLS *(Ethernet over MPLS)*, enlace punto a punto y circuito virtual *Frame Relay* (PVC).

Por ejemplo…

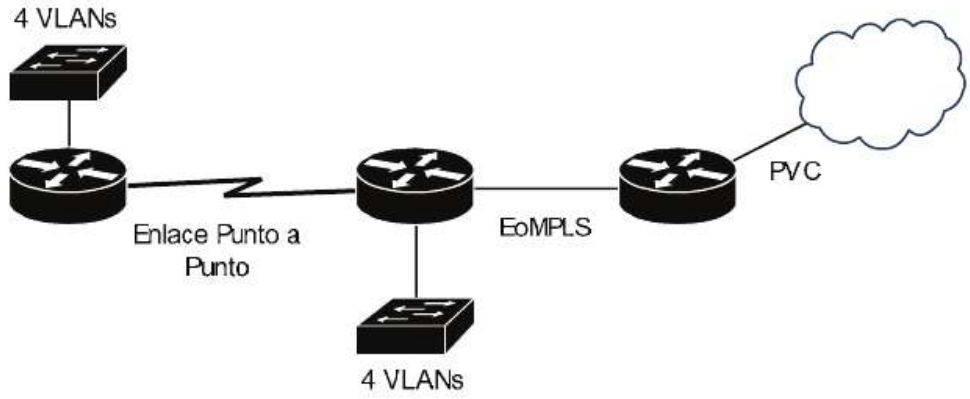


Fig. 4-1 Cálculo del número total de subredes necesarias.

La topología mostrada está compuesta por un total de 8 VLANs, 1 enlace punto a punto, otro EoMPLS y un circuito virtual. Por lo tanto, serían necesarias un total de 11 subredes. Además, en un entorno real se deberá tener en cuenta el crecimiento de la compañía, reservando un número adecuado de subredes para su futura implementación.

¿Cuántas direcciones IP se requieren por subred?

Cada host conectado requiere una IP. Estos, como se ha mencionado en párrafos anteriores, no hacen referencia solo a PCs de usuarios, también a routers, switchs,

impresoras, etc. Tenerlo en cuenta resulta imprescindible ya que un cálculo erróneo puede dar como resultado la aplicación de un rango cuyo máximo de dispositivos permitidos sea inferior al necesario.

Para calcular el total de dispositivos que puede albergar una subred se debe aplicar la fórmula **2^n-2,** donde *n* hace referencia al número de bits de la máscara destinados a la parte de hosts. Al resultado se le resta dos porque una dirección corresponde al ID de red y otra al broadcast. Por ejemplo, la máscara por defecto de una clase C es 255.255.255.0, la cual hace uso de 24 bits para la parte de red y 8 para hosts. ¿Cuál es el número máximo de dispositivos permitidos en ella?

2^8-2 = 254 hosts.

¿Y para una clase B? Estas por defecto aplican la máscara 255.255.0.0, con 16 bits para la parte de red y otros 16 para hosts, de tal manera que:

2^{16}-2 = 65534 hosts.

En párrafos posteriores se analizarán en profundidad la máscara y el cálculo de subredes. Por el momento solo interesan los conceptos más básicos.

Selección del rango de direcciones

Una vez obtenido el número de subredes necesarias y el total de hosts para cada una de ellas se debe seleccionar qué clase de direcciones (A, B o C) aplicar para cumplir con los requisitos calculados. Para ello, una buena práctica consiste en llevar a cabo las siguientes acciones:

- Seleccionar una red con clase que se adapte a las necesidades.
- Estudio de la máscara de red.
- Cálculo de subredes.
- Crear un listado de subredes disponibles conforme al cálculo realizado.

Seleccionar una red con clase que se adapte a las necesidades

En el capítulo 1 fueron analizadas las direcciones IPv4, dividiéndolas en diferentes clases, A, B o C, donde cada una de ellas dispone de un número de bits predefinidos para la parte de hosts y de red. Bien, una red con clase *(classful)* es aquella en la que estos bits por defecto no se modifican, cumpliendo con los siguientes valores:

Red Clase	Bytes (bits) para la porción de red	Bytes (bits) para la porción de hosts	Número de hosts permitido por red
A	1 (8)	3 (24)	2^{24}-2
B	2 (16)	2 (16)	2^{16}-2
C	3 (24)	1 (8)	2^{8}-2

Como ejemplo práctico tomaremos una red compuesta por 500 hosts. ¿Qué clase se debe seleccionar para albergar tal cantidad de dispositivos? La clase C permite como máximo 2^{8}-2, que suman un total de 254. Esta no cumple con los requisitos necesarios, por lo tanto, se repite el cálculo para una clase B, que permite un total de 2^{16}-2, con lo que suman 65534. En este caso sí se cumplen las condiciones, por lo tanto, resulta necesaria una clase B para albergar a todos ellos.

El siguiente paso consiste en determinar una dirección de red privada perteneciente a la clase seleccionada, en este caso, la B. Para ello existen una serie de rangos disponibles, siendo los siguientes:

Clase	Rango de direcciones de red privadas	Número de redes
A	10.0.0.0	1
B	172.16.0.0 hasta 172.31.0.0	16
C	192.168.0.0 hasta 192.168.255.0	256

La dirección seleccionada dependerá por completo del administrador. Para el ejemplo se tomará la 172.20.0.0.

Resumiendo, con el cálculo realizado se ha concluido que para albergar 500 dispositivos en una misma red resulta necesaria, como mínimo, una dirección de clase B, la cual permite un máximo de 65534 hosts. Como contra cabe mencionar la enorme cantidad de direcciones disponibles que quedarían sin asignar. Sin embargo, gracias a ello, si la red creciera se podrían crear subredes sobre la misma dirección ya seleccionada, como se verá en párrafos posteriores.

Estudio de la máscara de red

Toda dirección IPv4 debe estar acompañada de una máscara de red, la cual también es representada en formato decimal, dividida en 4 octetos de 8 bits cada uno. Su función consiste en determinar qué bits de la dirección IP identifican a la red y cuáles al host.

Las máscaras de redes con clase aplican valores por defecto, donde todos los bits a 1 identifican a la red, mientras que aquellos establecidos a 0 al host. Estos valores son los siguientes:

Red Clase	Bytes (bits) destinados para identificar la red	Máscara de red por defecto
A	1 (8)	255.0.0.0
B	2 (16)	255.255.0.0
C	3 (24)	255.255.255.0

De tal manera que…

Clase A:

Parte de red: 8 bits	Parte de hosts: 24 bits	
11111111	. 00000000.00000000.00000000	= 255.0.0.0

Clase B:

Parte de red: 16 bits	Parte de hosts: 16 bits	
11111111.11111111	. 00000000.00000000	= 255.255.0.0

Clase C:

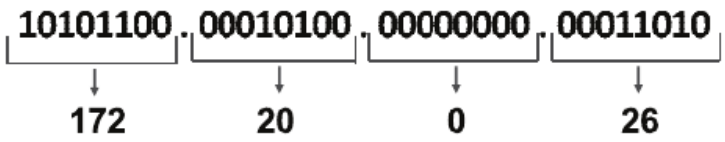
Parte de red: 24 bits	Parte de hosts: 8 bits	
11111111.11111111.11111111	. 00000000	= 255.255.255.0

Gracias a ello, y aplicando el siguiente procedimiento, resulta posible identificar la red a la que pertenece un determinado host. Por ejemplo, ¿de qué red forma parte la siguiente dirección IP?
IP: 172.20.0.26
Máscara: 255.255.0.0

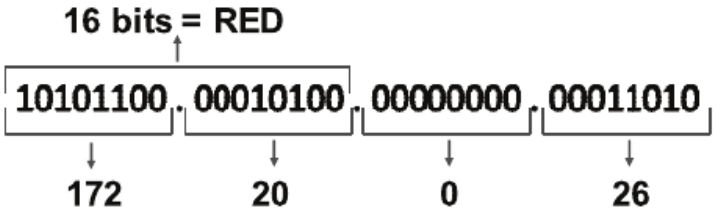
Paso 1: Para averiguarlo se debe examinar la máscara.

Paso 2: Su valor en decimal es igual a 255.255.0.0, que en formato binario corresponde a 11111111.11111111.00000000.00000000. Como es la máscara predeterminada para una clase B, todos los bits con valor 1 (un total de 16) identifican a la red, mientras que el resto al host.

Paso 3: Convertir la IP a binario.



Paso 4: Del resultado, los 16 primeros bits hacen referencia a la red a la que pertenece el host.



Paso 5: Estos corresponden a 172.20. El resto de bits toman el valor 0, por lo que la dirección de red en este caso es la **172.20.0.0**. El 26 indica que el dispositivo con dicha IP es el host número 26. Concluyendo, la IP 172.20.0.26 pertenece a la red 172.20.0.0/16.

La asociación "IP - máscara" también puede ser representada en el formato "172.20.0.26/16". Ello indica que la máscara de red está compuesta por 16 bits, conocido como prefijo de red.

Cálculo de subredes

Crear subredes consiste en dividir una red de clase definida en segmentos más pequeños con el fin de aprovechar el espacio de direcciones lo máximo posible. ¿Cómo llevarlo a cabo? El proceso consiste en tomar bits de la parte de hosts y destinarlos a la parte red. Estos crearán e identificarán las diferentes subredes, de tal manera que…

Clase A:

Clase B:

Clase C:

Para facilitar la compresión, el estudio se basará en el siguiente supuesto práctico, donde la red mostrada a continuación debe ser segmentada mediante la implementación de subredes:

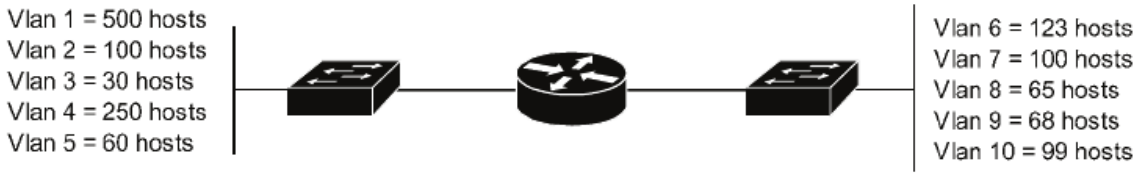


Fig. 4-2 Ejemplo de segmentación en subredes.

¿Cuántas son necesarias?

10 VLANs = 10 subredes

¿Número de hosts en cada una de ellas?

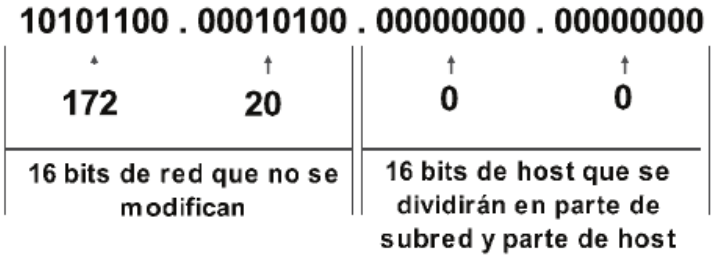
Indicado en la topología, la mayor está compuesta por 500 dispositivos.

¿Qué clase resulta la más adecuada para crearlas?

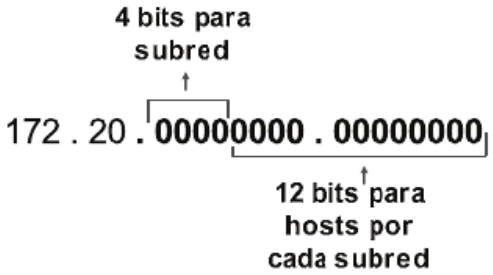
Las redes de clase C permiten un máximo de 254 hosts, por lo que no cumple los requisitos mínimos, sin embargo, la clase B alberga un total de 65534 dispositivos, convirtiéndose en la elección idónea para esta topología. Continuando con el ejemplo anterior se hará uso de la dirección de red 172.20.0.0/16.

El procedimiento se basa en definir cuántos bits son necesarios para crear las 10 subredes. Una vez hecho también se podrá calcular el máximo de hosts permitidos en cada una de ellas. Para ello se deberán llevar a cabo las siguientes acciones:

Paso 1: Localizar qué bits pueden ser divididos en parte de subred y de hosts. La máscara por defecto para la clase B es /16, por lo tanto, los 16 primeros bits siempre permanecerán igual, el cálculo de subredes se llevará a cabo con los restantes.



Paso 2: Una vez identificados se puede proceder a calcular cuántos de ellos permiten la creación del total de subredes necesarias. Para ello se aplica la fórmula 2^n, donde *n* hace referencia al número de bits a utilizar.
$2^2 = 4$ → con 2 bits no resulta posible crear 10 subredes.
$2^3 = 8$ → con 3 bits no resulta posible crear 10 subredes.
$2^4 = 16$ → con 4 bits sí se logra el objetivo.



Paso 3: Calcular el máximo de dispositivos permitidos en cada subred mediante la aplicación de la fórmula ya analizada 2^n-2.
$2^{12} - 2 = 4094$ dispositivos permitidos en cada subred.

En el supuesto práctico, la mayor subred consta de 500 hosts, por lo que se cumplen los requisitos necesarios.

Para lo que resta de capítulo resulta importante controlar la conversión de formato binario a decimal y viceversa, analizada en el capítulo 1.

Crear un listado de subredes disponibles conforme al cálculo realizado

El último paso consiste en definir el rango de cada una de las subredes, el cual estará compuesto por los siguientes valores:

- ID: el ID identifica a la subred y es definido asignando el valor 0 a todos los bits de la parte de hosts.
- Dirección broadcast: es la última dirección IP disponible dentro de la subred, se calcula asignando el valor 1 a todos los bits de la parte de host.
- IPs disponibles: el rango de IPs que pueden ser utilizadas por dispositivos.

¿Qué máscara aplicarán todas las subredes?

La máscara de subred estará compuesta por todos los bits de la máscara por defecto de la clase aplicada, más aquellos necesarios para la creación de las subredes, en este caso, 16 de la clase B más 4 de subred hacen un total

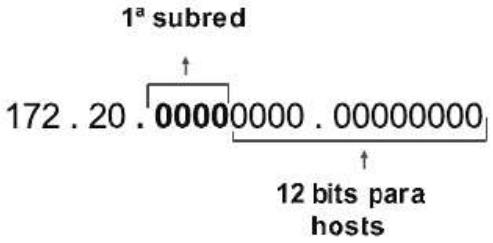
de 20 (/20). Para hallar su valor bastará con asignar el binario 1 a los 20 primeros bits de la máscara, para luego convertirla a formato decimal.

11111111.11111111.11110000.00000000, que en decimal equivale a 255.255.240.0.

Cálculo de la primera subred

El cálculo de la primera subred se lleva a cabo aplicando el valor más bajo posible a los 4 bits destinados para ellas, es decir, 0000. Resulta importante recordar que los 16 primeros bits no se modificarán, ya que al ser una clase B son necesarios para identificar la red segmentada. Su valor en decimal equivale a 172.20, por lo tanto, el cálculo se centrará en los 16 restantes, que son los que realmente interesan.

La primera subred es aquella con todos los bits de subred a 0, de tal manera que:



- ID de subred: el ID se obtiene asignando el valor 0 a la parte de host. Es decir, 172.20.00000000.00000000, tal y como se muestra en la imagen, que en formato decimal equivale a 172.20.0.0.

- Broadcast: el broadcast identifica la última IP disponible dentro de la subred y se calcula asignando el valor 1 a todos los bits de la parte de host, de tal manera que, 172.20.00001111.11111111, que en decimal equivale a 172.20.15.255.

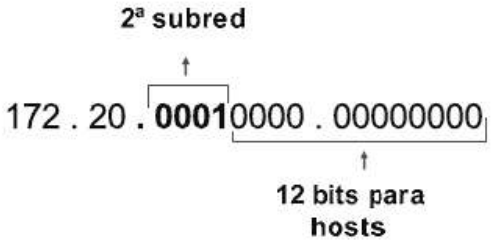
- Por último, calcular el rango de direcciones disponibles para dispositivos, que alberga desde la primera IP después del ID hasta la última antes del broadcast. En este caso el rango en binario es el siguiente: 172.20.00000000.00000001 - 172.20.00001111.11111110 que en formato decimal equivale a 172.20.0.1 - 172.20.15.254.

Con ello, los datos de la primera subred son los siguientes:

ID: 172.20.0.0/20 (Máscara de 20 bits).
Dirección de broadcast: 172.20.15.255.
Rango de IPs disponibles para hosts: 172.20.0.1 - 172.20.15.254.

Cálculo de la segunda subred

Para calcularla bastará con sumar un valor (en binario) a los 4 bits destinados para ello, de tal manera que:



- ID: Se asigna el valor 0 a la parte de hosts, obteniendo el siguiente resultado, 172.20.00010000.00000000, que en formato decimal equivale a 172.20.16.0.

- Broadcast: Se asigna el valor 1 a todos los bits de la parte de hosts, con lo cual, 172.20.00011111.11111111, que en decimal es igual a 172.20.31.255.

- Rango de IPs disponibles: 172.20.00010000.00000001-172.20.00011111.11111110, que en formato decimal corresponde a, 172.20.16.1 - 172.20.31.254.

Datos de la segunda subred:

ID: 172.20.16.0/20 (Máscara de 20 bits).
Dirección de broadcast: 172.20.31.255.
Rango de IPs disponibles para hosts: 172.20.16.1 - 172.20.31.254.

Se deberá aplicar el mismo procedimiento sobre cada una de las 16 subredes posibles, obteniendo el siguiente resultado:

Subred	Bits	ID	Broadcast	Máscara
1	0000	172.20.0.0	172.20.15.255	255.255.240.0
2	0001	172.20.16.0	172.20.31.255	255.255.240.0
3	0010	172.20.32.0	172.20.47.255	255.255.240.0
4	0011	172.20.48.0	172.20.63.255	255.255.240.0
5	0100	172.20.64.0	172.20.79.255	255.255.240.0
6	0101	172.20.80.0	172.20.95.255	255.255.240.0
7	0110	172.20.96.0	172.20.111.255	255.255.240.0
8	0111	172.20.112.0	172.20.127.255	255.255.240.0
9	1000	172.20.128.0	172.20.143.255	255.255.240.0
10	1001	172.20.144.0	172.20.159.255	255.255.240.0

11	1010	172.20.160.0	172.20.175.255	255.255.240.0
12	1011	172.20.176.0	172.20.191.255	255.255.240.0
13	1100	172.20.192.0	172.20.207.255	255.255.240.0
14	1101	172.20.208.0	172.20.223.255	255.255.240.0
15	1110	172.20.224.0	172.20.239.255	255.255.240.0
16	1111	172.20.240.0	172.20.255.255	255.255.240.0

Implementación de subredes en la topología real

Concluido el cálculo tan solo bastaría implementarlas en la topología, para lo cual se deben tener en cuenta principalmente dos aspectos:

- Localización de cada subred.
- Asignación de direcciones IP.

Localización de cada subred

Esta tarea depende por completo del administrador y/o política de seguridad de la compañía, siendo lo más común realizar la división por departamentos o propiedad. Sea como fuere, una buena práctica consiste en la asignación de subredes de manera contigua en los diferentes segmentos físicos de la red, logrando gracias a ello que la sumarización sea posible y que las tablas de enrutamiento contengan menos entradas, hecho que a su vez conlleva mayor velocidad y menor consumo de recursos.

Continuando con el supuesto práctico de la *Fig. 4-2*, de las 16 subredes calculadas tan solo resultan necesarias 10. ¿Cuáles aplicar? Realmente cualquiera de ellas. Para el ejemplo se hará uso de las 10 primeras, asignadas de manera contigua.

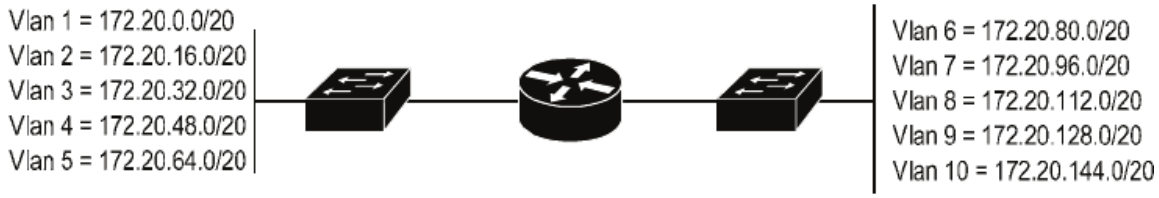


Fig. 4-3 Aplicación de subredes en diseño segmentado.

Una de las tareas más importantes a llevar a cabo consiste en documentar la red y actualizarla siempre que sea necesario. Entre los datos se deben incluir las subredes calculadas, rangos, departamento o lugar físico donde es aplicada cada una de ellas,

número de dispositivos, direcciones IP estáticas y cualquier información de interés al respecto. Todo cálculo y configuración debe ser documentado.

La sumarización será objeto de estudio a lo largo de este mismo capítulo.

Asignación de direcciones IP

La asignación de IPs a los dispositivos de cada subred puede llevarse a cabo de dos maneras, estática y dinámica. Las direcciones estáticas son configuradas manualmente a hosts que requieren siempre la misma IP. Estos suelen ser elementos críticos, como puertas de enlace, puntos de acceso o servidores, siendo también destinadas a recursos compartidos como impresoras en red.

Por el contrario, las direcciones dinámicas son asignadas automáticamente por un servidor DHCP a dispositivos cuya IP puede cambiar sin suponer un problema para la red. Por ejemplo, los PCs de usuarios.

Los routers Cisco pueden actuar como servidores DHCP, su estudio y configuración forman parte del capítulo 5 *"Instalación y configuración inicial de routers Cisco".*

EJERCICIOS PRÁCTICOS DE SUBNETTING

Conversión entre formato binario y decimal

Obtener el valor binario de la dirección IP 192.168.60.10

Paso 1: Una IP está compuesta por 32 bits divididos en 4 octetos, cada uno de ellos separado por un ".". El cálculo de decimal a binario se lleva a cabo teniendo en cuenta los siguientes valores:

128	64	32	16	8	4	2	1
0	0	0	0	0	0	0	0

Fig. 4-4 Equivalencia entre posición de bit binario y su valor en decimal.

Paso 2: Identificar qué bits de cada octeto, sumados, dan como resultado el decimal buscado, y modificar sus valores a 1. De tal manera que:

192 = 128 + 64 = 11000000. Los dos primeros bits del primer octeto se establecen con valor 1 porque su suma da como resultado 192.

168 = 128 + 32 + 8 = 10101000. Los bits con valor 128, 32 y 8 se establecen a 1 ya que su suma da como resultado 168.

60 = 32+16+8+4 = 00111100. En el tercer octeto los bits con valor 32, 16, 8 y 4 hacen la suma de 60.

10 = 8+2 = 00001010, donde los valores 8 y 2 se establecen a 1 ya que su suma da como resultado 10.

Paso 3: Realizada la conversión de formato decimal a binario se puede concluir que:

192.168.60.10 = 110000.10101000.00111100.00001010

Reto 4.1 – Convertir a formato binario las direcciones IP 172.20.0.45 y 192.80.254.255.
(Solución al final del capítulo)

Obtener el valor decimal de la dirección IP 10100000.00101101.01111000.00000001

Paso 1: Al igual que en la práctica anterior, el resultado se obtendrá en relación con su equivalente en decimal de cada bit en cada octeto *(Fig. 4-4)*. En este caso, aquellos con valor 1 deben ser sumados, de tal manera que:

10100000 = 128 + 32 = 160
00101101 = 32 + 8 + 4 + 1 = 45
01111000 = 64 + 32 + 16 + 8 = 120
00000001 = 1 = 1

Paso 2: Realizada la conversión binario - decimal se puede concluir que:

10100000.00101101.01111000.00000001 = 160.45.120.1

Reto 4.2 – Convertir a formato decimal la dirección IP 11000000.01100011.11110010.00000111.
(Solución al final del capítulo)

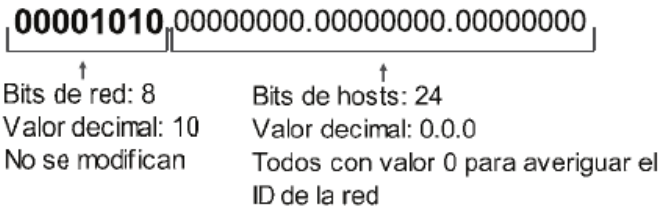
Redes con clase

Conforme a la IP 10.20.20.20, deducir los siguientes datos de la red con clase a la que pertenece: ID, dirección de broadcast y número de bits destinados para red y para hosts.

Paso 1: Al tratarse de una red con clase, los bits destinados tanto para red como para host son los establecidos por defecto, resultando, en este caso, de una dirección de clase A.

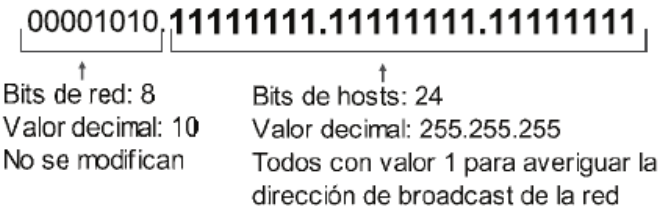
Paso 2: Por lo tanto, se aplican los bits predefinidos para la misma, 8 de red y 24 de hosts.

Paso 3: Tras realizar la conversión a binario de la IP se puede obtener el ID gracias a los 8 bits de red, los cuales forman el primer octeto. El valor de estos no se modifica, mientras que a los 24 restantes se les aplica el binario 0.



Por lo tanto, el ID de red de la IP 10.20.20.20 es 10.0.0.0/8.

Paso 4: La dirección de broadcast es aquella en la que todos los bits de la parte de hosts toman el valor binario 1.



Su resultado en decimal equivale a 10.255.255.255.

Gracias a todo ello, el estudio realizado da como resultado los siguientes datos:

IP	Clase	Bits de red	Bits de Hosts	ID de red	Broadcast
10.20.20.20	A	8	24	10.0.0.0 /8	10.255.255.255

Reto 4.3 – Completar la siguiente tabla con los datos que se solicitan:

IP	Clase	Bits de red	Bits de Hosts	ID de red	Broadcast
80.15.30.254					
172.20.14.55					
1.255.255.10					
200.20.20.20					
192.168.3.90					
191.70.48.163					

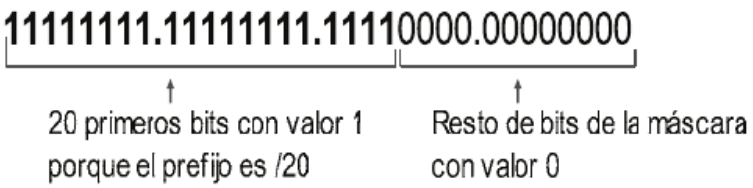
(Solución al final del capítulo)

Cálculo de máscaras de subred

¿Qué máscara de subred está asociada a la IP 172.20.49.1/20?

Paso 1: El prefijo de la dirección es /20, el cual indica el número de bits utilizados por la máscara.

Paso 2: Para calcularla bastará con aplicar el valor 1 a los 20 primeros bits, mientras que al resto, 0, de tal manera que:



Paso 3: Tras realizar la conversión a decimal se obtiene la máscara 255.255.240.0.

¿Qué prefijo es el correcto para la máscara de subred 255.252.0.0?

Paso 1: El prefijo indica el número de bits con valor 1 incluidos en la máscara, por lo tanto, para averiguarlo bastara con convertirla a binario.

255.252.0.0 = 11111111.11111100.00000000.00000000

Paso 2: Hacen un total de 14, por lo tanto el prefijo correcto para la máscara 255.252.0.0 es /14.

Reto 4.4 – Completar la siguiente tabla:

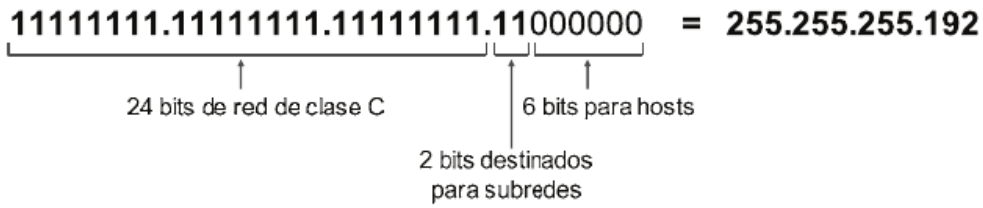
Prefijo	Máscara en binario	Máscara en decimal
/20		
	11111111.11000000.00000000.00000000	
	11111111.11111111.11111111.11100000	
		255.240.0.0
/17		
/12		
		255.255.255.252

(Solución al final del capítulo)

Identificación de subredes

¿A qué subred pertenece el host con IP 192.168.20.150/26?

Paso 1: El primer paso a realizar consiste en calcular cuántos bits son destinados para crear subredes. Para ello se debe identificar el tipo de dirección, que en este caso corresponde a una clase C, la cual utiliza por defecto 24 bits de red y 8 de hosts. Sin embargo, el prefijo de la IP es /26, con lo cual se están creando subredes a raíz de ella. Se debe calcular su máscara estableciendo los 26 primeros bits a 1:



Paso 2: Identificar el número de subredes posibles (2^2= 4) y sus rangos. Los 24 primeros bits (192.168.20) no se modifican, por lo tanto, el cálculo se centrará en los 2 bits de subred:

1ª Subred: 192.168.20.00000000 → 192.168.20.0 (rango 192.168.20.1 – 192.168.20.62)

2ª Subred: 192.168.20.01000000 → 192.168.20.64 (rango 192.168.20.65 – 192.168.20.126)

3ª Subred: 192.168.20.10000000 → 192.168.20.128 (rango 192.168.20.129 – 192.168.20.190)

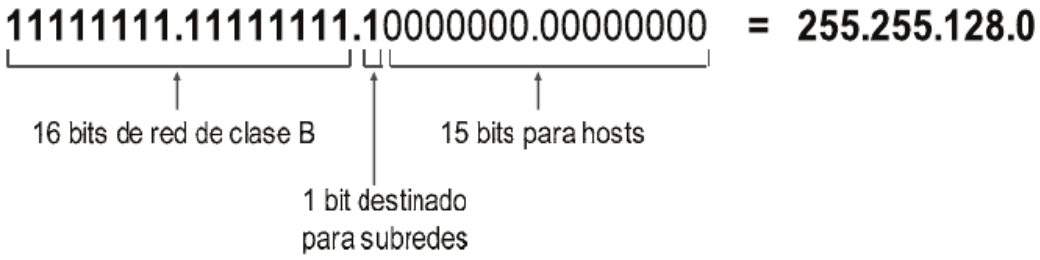
4ª Subred: 192.168.20.11000000 → 192.168.20.192 (rango 192.168.20.193 – 192.168.20.254)

Paso 3: Se puede concluir que la IP 192.168.20.150 pertenece a la subred 192.168.20.128.

Reto 4.5 – ¿A qué subred pertenece el host con IP 172.18.38.0/19?
Reto 4.6 – ¿A qué subred pertenece el host con IP 10.0.0.254/10?
(Solución al final del capítulo)

El host con IP 172.21.130.1/17 desea enviar un paquete broadcast. ¿A qué dirección debe hacerlo? ¿Cuántos hosts como máximo pueden formar parte de su misma subred?

Paso 1: El procedimiento para resolver este ejercicio coincide con el anterior. Se deberá calcular la subred a la que pertenece la IP y su rango, gracias a lo cual se obtiene la dirección de broadcast. La IP corresponde a una clase B, por lo tanto, 16 bits de red y 16 bits para hosts por defecto, y su prefijo es /17, por lo que se está destinando 1 bit para la creación de subredes. Calculando su máscara…



Paso 2: Identificar las subredes posibles y sus rangos (2^1= 2 subredes):

Subred 1: 172.21.000000000.00000000 → 172.21.0.0 (rango 172.21.0.1 – 172.21.127.254)

Subred 2: 172.21.100000000.00000000 → 172.21.128.0 (rango 172.21.128.1 – 172.21.255.254)

Paso 3: La IP 172.21.130.1 pertenece a la subred 172.21.128.0.

Paso 4: Para calcular la dirección de broadcast se establecen todos los bits de la parte de host con valor 1. Por lo tanto…

Broadcast de subred 2: 172.21.11111111.11111111 → 172.21.255.255

Paso 5: Tan solo faltaría hallar cuántos hosts pueden formar parte de la misma subred. Para ello bastará con ejecutar la fórmula $2^n - 2$.

$2^{15} - 2 = 32766$ hosts como máximo.

Reto 4.7 – ¿Cuál es la dirección broadcast de la subred 10.128.0.0/9?
Reto 4.8 – ¿Cuál es la dirección broadcast de la subred a la que pertenece el host con IP 192.168.10.195/27?
(Solución al final del capítulo)

Creación de subredes

Crear las subredes necesarias para la siguiente topología en relación con los siguientes requisitos:

- Las subredes de las diferentes VLANs se crearán a partir de la red 192.168.1.0/24. Todas deben ser del mismo tamaño.

- Las subredes de los enlaces seriales se crearán a partir de la red 192.168.255.0/24. Cada una de ellas debe permitir como máximo dos hosts.

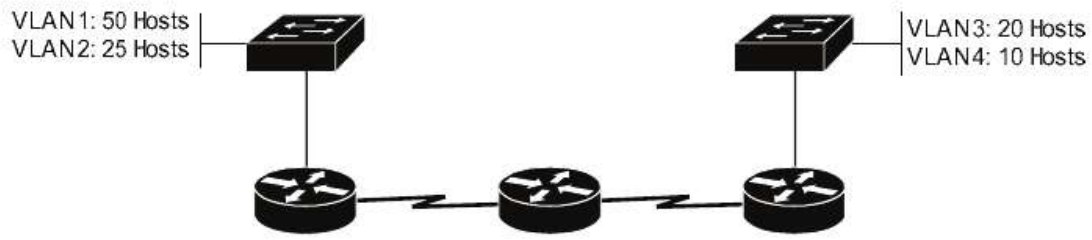


Fig. 4-5.1 Cálculo de subredes e implementación sobre topología real.

Subredes para las VLANs:

Paso 1: Calcular cuántos bits son necesarios para crear 4 subredes, aplicando la fórmula 2^n para averiguarlo.

$2^2 = 4$ → Se requieren 2 bits.

Paso 2: La dirección 192.168.1.0 corresponde a una clase C, por lo tanto, destina 24 bits para la parte de red por defecto, a los cuales han de sumarse los dos necesarios para subredes, haciendo un total de 26. Conociendo dicho dato se debe calcular el máximo de hosts permitidos, para lo cual se toma como referencia la VLAN 1, ya que es aquella que alberga mayor número de dispositivos (50). Por lo tanto, el cálculo debe dar como resultado un valor superior al mismo. En caso contrario se tendría que hacer uso de otra clase que permita un rango más elevado (clase B).

Con 26 bits para red, restan 6 bits para hosts, de tal manera que:

$2^6 - 2 = 62$. Cada subred permite como máximo 62 dispositivos. Se cumplen los requisitos necesarios.

Paso 3: Calcular las 4 subredes posibles y sus rangos:

Subred 1: 192.168.1.00000000 → 192.168.1.0 (rango 192.168.1.1 – 192.168.1.62)

Subred 2: 192.168.1.01000000 → 192.168.1.64 (rango 192.168.1.65 – 192.168.1.126)

Subred 3: 192.168.1.10000000 → 192.168.1.128 (rango 192.168.1.129 – 192.168.1.190)

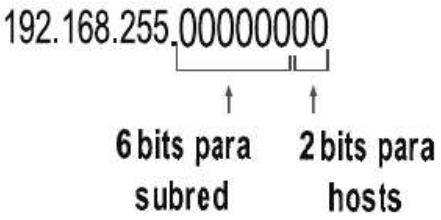
Subred 4: 192.168.1.11000000 → 192.168.1.192 (rango 192.168.1.193 – 192.168.1.254)

Subredes para enlaces seriales:

Paso 1: En este caso se solicita que cada subred permita un máximo de dos dispositivos, por lo tanto, el primer paso a llevar a cabo consiste en calcular cuántos bits de la parte de hosts son necesarios para cumplir dicho requisito, aplicando para ello la fórmula $2^n - 2$:

$2^1 - 2 = 0$ hosts.
$2^2 - 2 = 2$ hosts. Cumple las condiciones.

Paso 2: De la dirección 192.168.255.0, los 24 primeros bits identifican la red por defecto de clase C y no se modifican, de los 8 restantes se ha calculado que 2 son para hosts, por lo tanto, se destinarán 6 para subredes.



Paso 3: Calcular las subredes posibles (2^6=64) y sus rangos. Por brevedad tan solo serán calculadas 2, ya que son las necesarias para la topología.

Subred 1: 192.168.255.00000000 → 192.168.255.0 (rango 192.168.255.1 – 192.168.255.2)

Subred 2: 192.168.255.00000100 → 192.168.255.4 (rango 192.168.255.5 – 192.168.255.6)

…

…..

Paso 4: Implementar los resultados obtenidos en el diseño real.

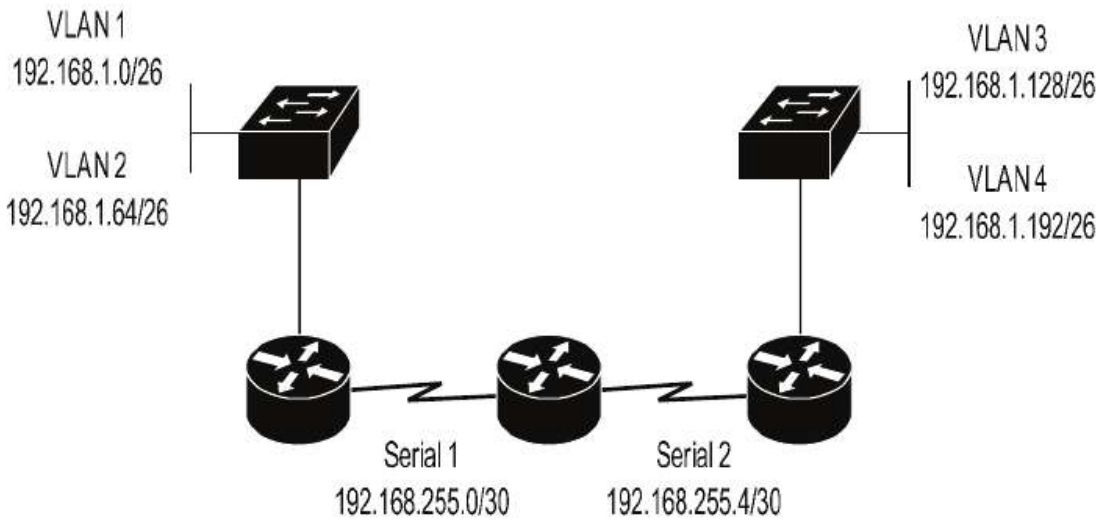
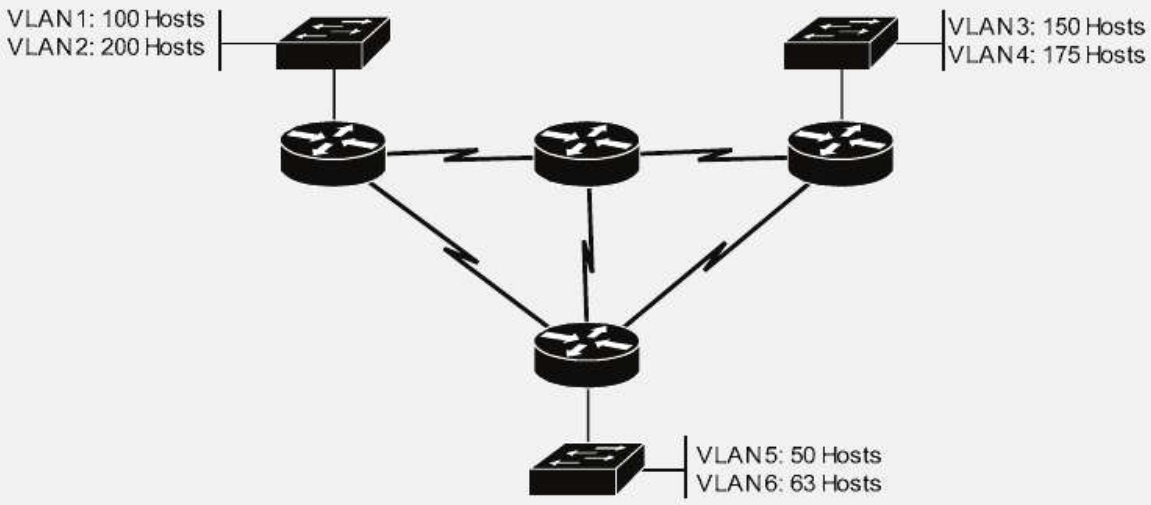


Fig. 4-5.2 Cálculo de subredes e implementación sobre topología real.

Reto 4.9 – Segmentar la siguiente topología de tal manera que:

- Las subredes de las VLANs deben ser calculadas conforme a la red 172.20.0.0/16.

- Las subredes de los enlaces seriales se crearán a partir de la red 192.168.0.0/24. Cada una de ellas debe permitir dos hosts como máximo.



(Solución al final del capítulo)

VLSM (VARIABLE LENGTH SUBNET MASKS)

El cálculo recién analizado permite la división de una red de clase A, B o C en diferentes segmentos, tomando para ello bits de la parte de hosts que serán destinados para la creación de subredes. En este modelo todas ellas permitirán el mismo número máximo de dispositivos y aplicarán la misma máscara, hecho que puede propiciar el desaprovechamiento de direcciones.

Para evitarlo se puede optar por la implementación de VLSM (*Variable Length Subnet Masks)* cuyo método consiste en, dada una red con clase, dividirla en subredes donde cada una de ellas podrá albergar un número máximo de hosts diferente. Este hecho conlleva el uso de máscaras de red variables y gracias a ello se logra un desaprovechamiento mínimo de las direcciones IP disponibles.

Resulta importante distinguir entre un diseño VLSM y otro donde existan subredes pero no sean variables. Por ejemplo...

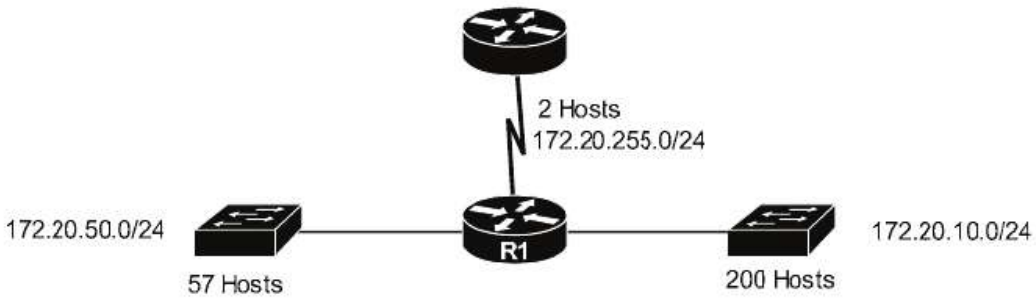


Fig. 4-6 Diseño de red sin VLSM.

La imagen muestra una topología dividida en 3 subredes en relación con la dirección de clase B 172.20.0.0/16. En ella no se aplica VLSM ya que siempre se hace uso de la misma máscara (/24). Se puede observar cómo el segmento 172.20.255.0/24 solo contiene dos hosts, y sin embargo, su rango permite un máximo de 254, lo mismo ocurre con la 172.20.50.0/24, que permite 254 dispositivos y solo necesita 57, produciendo un desaprovechamiento de direcciones de red.

Si se aplicara VLSM en la misma topología, podría quedar definida de la siguiente manera:

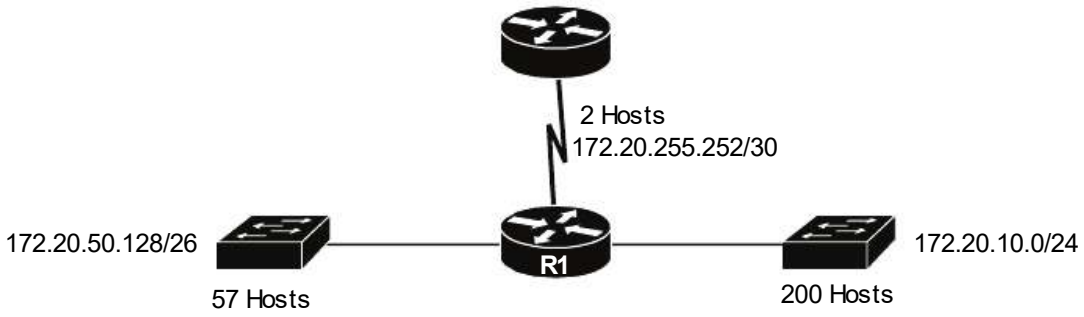


Fig. 4-7 Diseño de red aplicando VLSM.

Donde la red de clase B 172.20.0.0/16 ha sido dividida en 3 subredes haciendo uso de diferentes máscaras, ajustando en cada una de ellas el número máximo de hosts disponibles al número real de dispositivos.

Subred	Rango	Máximo de Hosts
172.20.10.0 /24	172.20.10.1 – 172.20.10.254	254
172.20.50.128/26	172.20.50.129 – 172.20.50.191	62
172.20.255.252 /30	172.20.255.253 – 172.20.255.254	2

Solapamiento de direcciones en VLSM

Uno de los problemas más comunes en VLSM consiste en el solapamiento de direcciones, que se produce cuando un mismo conjunto de IPs forma parte de diferentes subredes. Este hecho puede ser debido a un cálculo erróneo o a configuraciones incorrectas y tiene como consecuencia que a hosts ubicados en diferentes segmentos se les asigne la misma dirección, produciendo errores de comunicación y posibles bucles de capa 3.

¿Cómo determinar si en la red existe dicho problema? Para ello se deben seguir los siguientes pasos:

- *Paso 1:* Calcular el rango de cada una de las subredes, incluyendo su ID y broadcast.

- *Paso 2:* Crear un listado con los resultados obtenidos.

- *Paso 3:* Analizarlo en busca de coincidencias.

En el siguiente diseño se ha aplicado VLSM sobre la red de clase B 172.30.0.0/16. Determinar si existe solapamiento de direcciones entre las diferentes subredes.

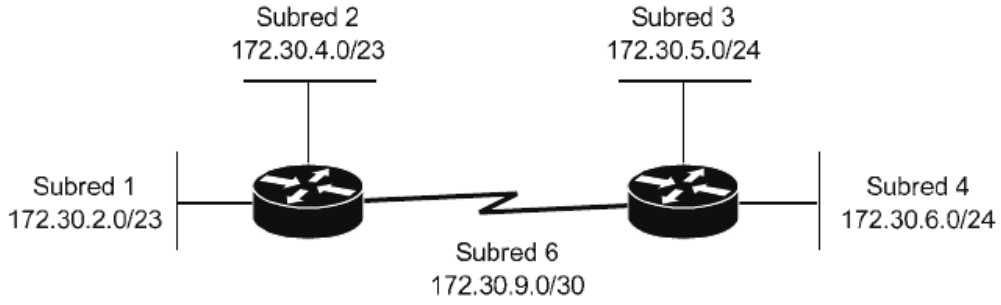


Fig. 4-8 Solapamiento de direcciones en diseño VLSM.

Paso 1: Rangos de red, incluyendo ID y broadcast.

Subred 1: 172.30.2.0/23

ID de red: Los 23 primeros bits no se modifican, el resto con valor 0.
10101100.00011110.0000001 0.00000000 --> 172.30.2.0

Broadcast: Los 23 primeros bits no se modifican, el resto con valor 1.
10101100.00011110.0000001 1.11111111 --> 172.30.3.255

Rango de red de la subred 1: 172.30.2.0 – 172.30.3.255

Subred 2: 172.30.4.0/23

ID de red: 10101100.00011110.0000010 0.00000000 --> 172.30.4.0
Broadcast: 10101100.00011110.0000010 1.11111111 --> 172.30.5.255
Rango de red de la subred 2: 172.30.4.0 – 172.30.5.255

Subred 3: 172.30.5.0/24

ID de red: 10101100.00011110.00000101 .00000000 --> 172.30.5.0
Broadcast: 10101100.00011110.00000101 .11111111 --> 172.30.5.255
Rango de red de la subred 3: 172.30.5.0 – 172.30.5.255

Subred 4: 172.30.6.0/24

ID de red: 10101100.00011110.00000110 .00000000 --> 172.30.6.0
Broadcast: 10101100.00011110.00000110 .11111111 --> 172.30.6.255
Rango de red de la subred 4: 172.30.6.0 – 172.30.6.255

Subred 6: 172.30.9.0/30

ID de red: 10101100.00011110.00001001.000000 00 --> 172.30.9.0
Broadcast: 10101100.00011110.00001001.000000 11 --> 172.30.9.3
Rango de red de la subred 6 : 172.30.9.0 – 172.30.9.3

Paso 2: Listado de todas las subredes y sus rangos.

Subred	**Rango**
Subred 1	172.30.2.0 – 172.30.3.255
Subred 2	172.30.4.0 – 172.30.5.255
Subred 3	172.30.5.0 – 172.30.5.255
Subred 4	172.30.6.0 – 172.30.6.255
Subred 6	172.30.9.0 – 172.30.9.3

Paso 3: Análisis en busca de coincidencias.

Analizando el listado se puede comprobar que entre la subred 2 y 3 existe un solapamiento de direcciones. El rango 172.30.5.0 - 172.30.5.255 forma parte de ambas, lo que tendrá como consecuencia fallos de comunicación y problemas de enrutamiento. La solución consiste en recalcular correctamente una o ambas subredes y aplicar los cambios a la topología.

Reto 4.10 – En una topología VLSM basada en la red de clase A 10.0.0.0/8 se han seleccionado las siguientes direcciones IP, pertenecientes a cada una de las subredes existentes:

Subred A: 10.10.34.10/22
Subred B: 10.10.29.100/23
Subred C: 10.10.23.200/22
Subred D: 10.10.17.16/21
Subred E: 10.10.1.253/20

¿Se produce solapamiento de direcciones entre los diferentes segmentos?

(Solución al final del capítulo)

Reto 4.11 – En una topología se ha aplicado VLSM sobre la red de clase C 192.168.255.0/24. Comprobar si entre las diferentes subredes existentes se produce solapamiento de direcciones.

Subred A: 192.168.255.0 /26
Subred B: 192.168.255.252 /30
Subred C: 192.168.255.128 /28
Subred D: 192.168.255.124 /30

(Solución al final del capítulo)

Agregar una nueva subred a un diseño VLSM

Otro caso de estudio e implementación, tanto en entornos reales como en el examen de CCNA, consiste en, dado un diseño VLSM ya existente, agregar una nueva subred teniendo en cuenta una serie de requisitos y sin que produzca solapamiento de direcciones con las ya existentes.

Para lograrlo se deben ejecutar las siguientes acciones:

- Paso 1: Seleccionar una máscara de red con relación al número total de hosts. Es decir, calcular cuántos bits son necesarios para cubrir todos los dispositivos que formarán parte de la subred, aplicando para ello la fórmula 2n-2.

- Paso 2: Identificar todas las subredes posibles (y sus rangos) aplicando la máscara calculada en el paso 1.

- *Paso 3: Crear un listado con las ya existentes y otro con las calculadas en el paso 2.*

- *Paso 4:* Comparar ambos listados en busca de solapamiento de direcciones y descartar aquellas subredes que lo produzcan.

- *Paso 5:* De las restantes, aplicar la que más se adapte a las necesidades.

En la topología recién analizada *(Fig. 4-8)*, las subredes 2 y 3 producían solapamiento de direcciones. Con el fin de solucionar el problema será eliminada la 3, creando una nueva capaz de albergar 200 dispositivos. De todas las opciones posibles, implementar aquella con menor ID.

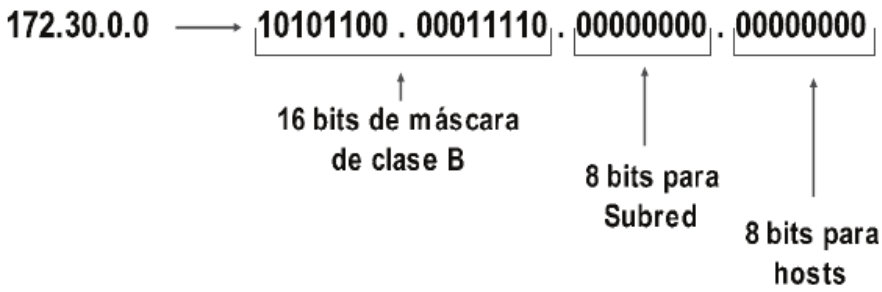
Paso 1: Selección de la máscara necesaria.

El primer paso a llevar a cabo consiste en determinar cuántos bits son necesarios para albergar 200 hosts, utilizando para ello la fórmula $2^{n}- 2$.

$2^{7} – 2 = 126$ hosts.

$2^{8} – 2 = 254$ hosts, cumple con los requisitos.

En la topología se hace uso de una dirección de clase B, la 172.30.0.0/16, por lo tanto, con los 16 bits por defecto más los 8 calculados para hosts, restan 8 para crear subredes, de tal manera que:



Por lo que la máscara de subred estará compuesta por 24 bits (16+8), dando como resultado, en formato decimal, 255.255.255.0.

Paso 2: Cálculo de subredes.

Identificar cada una de las subredes posibles y sus rangos:

Subred Zero (todos los bits de subred a cero):

ID: 172.30.00000000 .00000000 -> 172.30.0.0
Broadcast: 172.30.00000000 .11111111 -> 172.30.0.255

1ª Subred:

ID: 172.30.00000001 .00000000 -> 172.30.1.0
Broadcast: 172.30.00000001 .11111111 -> 172.30.1.255

2ª Subred:

ID: 172.30.00000010 .00000000 -> 172.30.2.0
Broadcast: 172.30.00000010 .11111111 -> 172.30.2.255

3ª Subred:

ID: 172.30.00000011 .00000000 -> 172.30.3.0
Broadcast: 172.30.00000011 .11111111 -> 172.30.3.255

…

….. *(Resultado omitido por brevedad)*

255ª Subred:

ID: 172.30.11111111 .00000000 -> 172.30.255.0
Broadcast: 172.30.11111111 .11111111 -> 172.30.255.255

Paso 3: Listado de subredes en uso y calculadas.

Subredes en uso	
Subred	**Rango**
Subred 1	172.30.2.0 – 172.30.3.255
Subred 2	172.30.4.0 – 172.30.5.255
Subred 3	???
Subred 4	172.30.6.0 – 172.30.6.255
Subred 6	172.30.9.0 – 172.30.9.3

Subredes calculadas	
Subred	**Rango**
0	172.30.0.0 – 172.30.0.255
1	172.30.1.0 – 172.30.1.255
2	172.30.2.0 – 172.30.2.255
3	172.30.3.0 – 172.30.3.255
4	172.30.4.0 – 172.30.4.255
5	172.30.5.0 – 172.30.5.255
6	172.30.6.0 – 172.30.6.255
7	172.30.7.0 – 172.30.7.255
8	172.30.8.0 – 172.30.8.255
9	172.30.9.0 – 172.30.9.255
...	.....
255	172.30.255.0 – 172.30.255.255

Paso 4: Análisis de solapamiento de direcciones.

Comparar ambos listados y detectar qué rangos de las subredes calculadas coindicen con los ya en uso:

- *Subred 1* (172.30.2.0 - 172.30.3.255): de las nuevas calculadas, la 2 y 3 producen solapamiento con ella, por lo tanto, quedan descartadas.

- *Subred 2* (172.30.4.0 - 172.30.5.255): las subredes 4 y 5 coinciden en rango, por lo tanto, también se descarta su uso.

- *Subred 4* (172.30.6.0 – 172.30.6.255): la subred 6 crea solapamiento de direcciones.

- *Subred 6* (172.30.9.0 – 172.30.9.3): de las nuevas calculadas, la 9 crea solapamiento, por lo tanto, también queda descartada.

Con todo ello, las posibles opciones a aplicar son las siguientes:

Subredes calculadas	
Subred	**Rango**
0	172.30.0.0 – 172.30.0.255
1	172.30.1.0 – 172.30.1.255

~~2~~	~~172.30.2.0 – 172.30.2.255~~
~~3~~	~~172.30.3.0 – 172.30.3.255~~
~~4~~	~~172.30.4.0 – 172.30.4.255~~
~~5~~	~~172.30.5.0 – 172.30.5.255~~
~~6~~	~~172.30.6.0 – 172.30.6.255~~
7	172.30.7.0 – 172.30.7.255
8	172.30.8.0 – 172.30.8.255
~~9~~	~~172.30.9.0 – 172.30.9.255~~
...	.....
255	172.30.255.0 – 172.30.255.255

Paso 5: Selección de la subred que más se ajuste a las necesidades.

De las subredes disponibles se ha solicitado que se aplique aquella con menor ID, siendo esta la 172.30.0.0/24, con rango 172.30.0.0 - 172.30.0.255.

Subredes en uso	
Subred	**Rango**
Subred 1	172.30.2.0 – 172.30.3.255
Subred 2	172.30.4.0 – 172.30.5.255
Subred 3	172.30.0.0 – 172.30.0.255
Subred 4	172.30.6.0 – 172.30.6.255
Subred 5	172.30.9.0 – 172.30.9.3

Reto 4.12 – Como administrador se le ha encomendado la tarea de agregar una nueva subred compuesta por 300 dispositivos a una topología VLSM ya existente formada por los siguientes segmentos:

172.16.4.0/22
172.16.8.0/24
172.16.10.128/25
172.16.255.0/30

Aplicar aquella de menor ID posible.
(Solución al final del capítulo)

SUMARIZACIÓN DE RUTAS

La sumarización puede ser definida como el proceso mediante el cual varias rutas hacia diferentes subredes son resumidas en tan solo una. Esta técnica suele llevarse a cabo en entornos corporativos de gran tamaño y gracias a ello se ahorran recursos como memoria RAM y CPU, logrando mayor velocidad de procesamiento.

Existen dos métodos para aplicarla, de manera manual o automática. La presente sección será dedicada a analizar la primera de ellas, ya que el modo automático es ejecutado por los protocolos de enrutamiento y su configuración será objeto de estudio en el capítulo 6 "*Protocolos de enrutamiento*".

El primer factor y sin duda el más importante para poder llevar a cabo la sumarización consiste en definir un diseño pensado para ello, en el cual las diferentes subredes con mismo rango numérico sean agrupadas en el mismo segmento físico. Por ejemplo:

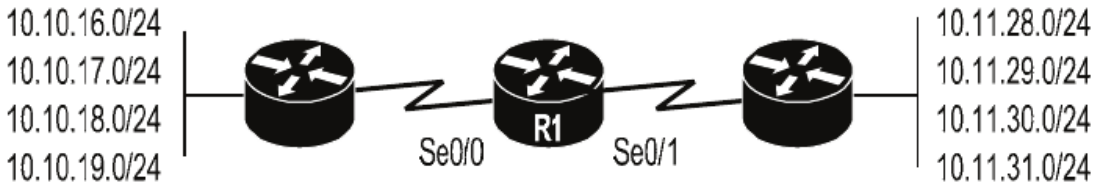


Fig. 4-9 Sumarización de rutas.

En la topología mostrada el administrador ha dividido la red en relación con la dirección de clase A 10.0.0.0/8, aplicando una máscara /24 y a su vez creando dos segmentos, el 10.10.x.x y 10.11.x.x. Además, cada uno de estos se ha implementado físicamente de tal manera que se pueda llevar a cabo una sumarización adecuada tanto en R1 como en el resto de routers.

El estudio se centrará en R1. Actualmente su tabla de rutas consta de las siguientes entradas:

Red	Interfaz salida
10.10.16.0/24	Se0/0
10.10.17.0/24	Se0/0
10.10.18.0/24	Se0/0
10.10.19.0/24	Se0/0
10.11.28.0/24	Se0/1
10.11.29.0/24	Se0/1
10.11.30.0/24	Se0/1
10.11.31.0/24	Se0/1

Fig. 4-10 Tabla de rutas con posibilidad de ser sumarizadas.

El objetivo consiste en definir, para cada interfaz, una sola ruta capaz de agrupar a todas las actuales, de tal manera que con tan solo dos entradas en la tabla se logre llevar a cabo el direccionamiento en capa 3 hacia las 8 subredes existentes.

¿Cómo lograrlo? Se debe calcular una máscara que incluya todas las subredes a sumarizar. Por ejemplo, para el segmento 10.10.x.x se podría aplicar la ruta 10.10.0.0/16, porque una máscara de 16 bits agrupa todas las subredes existentes en dicho segmento (y otras muchas más). Con el rango 10.11.x.x sucede exactamente lo mismo, podría ser definido en la ruta 10.11.0.0/16.

Con ello, la tabla quedaría compuesta por dos entradas, tal que:

Red	Interfaz salida
10.10.0.0/16	Se0/0
10.11.0.0/16	Se0/1

Fig. 4-11.1 Tabla de rutas sumarizada.

Cuando R1 reciba un paquete con destino 10.10.16.251 comprobará su tabla de rutas, determinará que la IP se encuentra dentro del rango 10.10.0.0/16 y lo

reenviará a través de su interfaz Se0/0. Lo mismo sucederá con el resto de direcciones pertenecientes a las diferentes subredes.

Sin embargo, ¿qué ocurre si el destino es, por ejemplo, la IP 10.10.200.1? Aunque dicha subred no exista en la topología, el router también lo reenviará a través de la interfaz Se0/0, ya que la ruta sumarizada 10.10.0.0/16 abarca el rango 10.10.0.0 - 10.10.255.255, el cual incluye demasiados destinos inexistentes. Lo mismo sucede con la 10.11.0.0/16, que alberga desde la dirección 10.11.0.0 hasta la 10.11.255.255.

La solución a dicho problema pasa por realizar un cálculo más detallado de la máscara a aplicar en la ruta sumarizada. Esta debe abarcar un rango que se ajuste lo máximo posible a las subredes incluidas en el segmento de red, para lo cual se deberán llevar a cabo las siguientes acciones:

- *Paso 1:* Crear un listado de todas las subredes a sumarizar, en orden, de menor a mayor ID.

- *Paso 2:* Calcular el rango de todas ellas juntas, tomando el ID de la más baja y la dirección de broadcast de la más alta.

- *Paso 3:* De las máscaras utilizadas, seleccionar la menor y restarle 1 bit.

- *Paso 4:* Asociar la nueva máscara al ID de red más bajo y calcular su rango. El resultado debe incluir el definido en el paso 2. Si es así, se ha obtenido la mejor ruta sumarizada para este conjunto de subredes. De lo contrario, se debe restar otro bit a la máscara y volver a realizar el cálculo. Así sucesivamente hasta lograr el objetivo.

Calcular la mejor ruta sumarizada para el conjunto de subredes del rango 10.10.x.x.

Paso 1: Crear listado de subredes de menor a mayor ID.

10.10.16.0/24
10.10.17.0/24
10.10.18.0/24
10.10.19.0/24

Paso 2: Calcular el rango de todas ellas juntas.

Se toma el ID de la subred más baja (10.10.16.0) y la dirección de broadcast de la más alta (10.10.19.255). Por lo tanto, el rango es el 10.10.16.0 - 10.10.19.255.

Paso 3: Seleccionar una máscara de subred para iniciar el cálculo.

De todas las máscaras, seleccionar aquella con menor número de bits y restarle uno. En este caso todas coinciden, /24, por lo tanto, se comienza sobre una /23.

Paso 4: Calcular el rango de la ruta sumarizada conforme a la nueva máscara.
Hallado el rango necesario (10.10.16.0 – 10.10.19.255) y la primera máscara a aplicar, tan solo basta iniciar el cálculo. Para ello se asocia el prefijo /23 con el ID de subred más bajo, en este caso 10.10.16.0.

10.10.16.0 /23 → 23 bits de red, 9 para hosts.
ID = 00001010.00001010.0001000 0.00000000 = 10.10.16.0
Broadcast = 00001010.00001010.0001000 1.11111111 = 10.10.17.255
Rango: 10.10.16.0 - 10.10.17.255

El resultado obtenido no abarca todas las direcciones necesarias, por lo tanto, esta ruta sumarizada no es válida. Se debe continuar restando otro bit a la máscara de subred. El nuevo prefijo es /22, asociado de nuevo al menor ID.

10.10.16.0/22 → 22 bits de red, 10 bits de hosts
ID = 00001010.00001010.000100 00.00000000 = 10.10.16.0
Broadcast = 00001010.00001010.000100 11.11111111 = 10.10.19.255
Rango: 10.10.16.0 – 10.10.19.255

Haciendo uso de una máscara /22 sí se obtiene el resultado deseado, ya que el rango incluye exactamente todas las direcciones definidas en el paso 2. Por lo tanto, la mejor ruta sumarizada para este conjunto de subredes es 10.10.16.0/22.

Realizar el mismo cálculo para el conjunto de subredes del rango 10.11.x.x.

Paso 1: Crear listado de subredes de menor a mayor ID.
10.11.28.0/24
10.11.29.0/24
10.11.30.0/24
10.11.31.0/24

Paso 2: Calcular el rango de todas ellas juntas.
Menor ID: 10.11.28.0
Broadcast de mayor ID: 10.11.31.255
Rango: 10.11.28.0 – 10.11.31.255

Paso 3: Seleccionar una máscara de subred para iniciar el cálculo.
Todas hacen uso de máscara /24, por lo tanto, se comienza con el prefijo /23, aplicado sobre la subred de menor ID, en este caso la 10.11.28.0.

Paso 4: Hallar el rango de la ruta sumarizada en relación con la nueva máscara.

10.11.28.0 /23 → 23 bits de red, 9 bits de hosts
ID = 00001010.00001011.0001110 0.00000000 = 10.11.28.0
Broadcast = 00001010.00001011.0001110 1. 11111111 = 10.11.29.255
Rango: 10.11.28.0 – 10.11.29.255, no cumple los requisitos calculados en el paso 2.

10.11.28.0 /22 → 22 bits de red, 10 bits de hosts
ID = 00001010.00001011.000111 00.00000000 = 10.11.28.0
Broadcast = 00001010.00001011.000111 11.11111111 = 10.11.31.255
Rango: 10.11.28.0 – 10.11.31.255, ¡Cumple los requisitos!

La ruta 10.11.28.0/22 abarca el rango de direcciones definido en el paso 2, por lo tanto, se convierte en la mejor opción a implementar.

Aplicación de rutas sumarizadas

Una vez concluido el cálculo tan solo bastaría aplicar los resultados obtenidos en el router. La manera más sencilla de llevarlo a cabo es mediante rutas estáticas, haciendo uso del comando **ip route** ***[id de red] [máscara] [interfaz salida]*** desde el modo de configuración global:

```
R1(config)# ip route 10.10.16.0 255.255.252.0 Se0/0
R1(config)# ip route 10.11.28.0 255.255.252.0 Se0/1
```

Con ello, la tabla de rutas de R1 queda definida de la siguiente manera:

Red	Interfaz salida
10.10.16.0/22	Se0/0
10.11.28.0/22	Se0/1

Fig. 4-11.2 Tabla de rutas sumarizada.

La configuración de rutas estáticas en dispositivos Cisco se analizará con mayor detalle en el capítulo 5 "*Instalación y configuración inicial de routers Cisco*".

SOLUCIÓN DE RETOS: SUBNETTING EN IPV4

Reto 4.1 – Convertir a formato binario las direcciones IP 172.20.0.45 y 192.80.254.255.

172.20.0.45 = 10101100.00010100.00000000.00101101
192.80.254.255 = 11000000.01010000.11111110.11111111

Reto 4.2 – Convertir a formato decimal la dirección IP 11000000.01100011.11110010.00000111.

11000000.01100011.11110010.00000111 = 192.99.242.7

Reto 4.3 – Completar la siguiente tabla con los datos que se solicitan:

IP 80.15.30.254	Clase	Bits de red	Bits de Hosts	ID de red	Broadcast
172.20.14.55					
1.255.255.10					
200.20.20.20					
192.168.3.90					
191.70.48.163					

IP	**Clase**	**Bits de red**	**Bits de Hosts**	**ID de red**	**Broadcast**
80.15.30.254	A	8	24	80.0.0.0 /8	80.255.255.255
172.20.14.55	B	16	16	172.20.0.0 /16	172.20.255.255
1.255.255.10	A	8	24	1.0.0.0 /8	1.255.255.255
200.20.20.20	C	24	8	200.20.20.0 /24	200.20.20.255
192.168.3.90	C	24	8	192.168.3.0 /24	192.168.3.255
191.70.48.163	B	16	16	191.70.0.0 /16	191.70.255.255

Reto 4.4 – Completar la siguiente tabla:

Prefijo	Máscara en binario	Máscara en decimal
/20		
	11111111.11000000.00000000.00000000	
	11111111.11111111.11111111.11100000	
		255.240.0.0
/17		
/12		
		255.255.255.252

Prefijo	**Máscara en binario**	**Máscara en decimal**
/20	11111111.11111111.11110000.00000000	255.255.240.0
/10	11111111.11000000.00000000.00000000	255.192.0.0
/27	11111111.11111111.11111111.11100000	255.255.255.224
/12	11111111.11110000.00000000.00000000	255.240.0.0
/17	11111111.11111111.10000000.00000000	255.255.128.0
/12	11111111.11110000.00000000.00000000	255.240.0.0
/30	11111111.11111111.11111111.11111100	255.255.255.252

Reto 4.5 – ¿A qué subred pertenece el host con IP 172.18.38.0/19?

El host con IP 172.18.38.0 pertenece a la subred 172.18.32.0/19.

Reto 4.6 – ¿A qué subred pertenece el host con IP 10.0.0.254/10?

El host con IP 10.0.0.254 pertenece a la subred 10.0.0.0/10.

Reto 4.7 – ¿Cuál es la dirección broadcast de la subred 10.128.0.0/9?

La dirección broadcast de la subred 10.128.0.0/9 es 10.255.255.255.

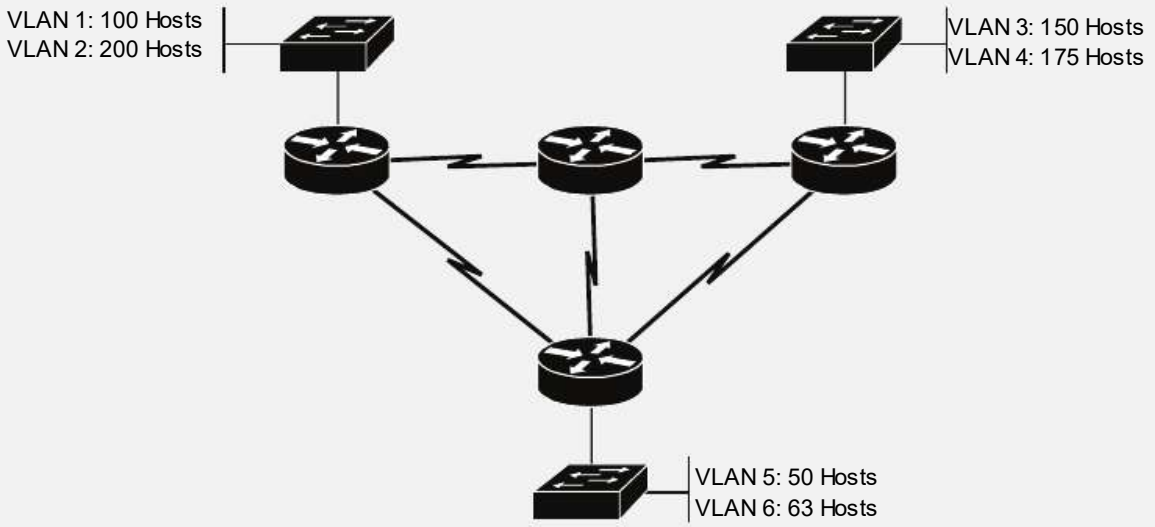
Reto 4.8 – ¿Cuál es la dirección broadcast de la subred a la que pertenece el host con IP 192.168.10.195/27?

El host 192.168.10.195 pertenece a la subred 192.168.10.192/27 y su dirección broadcast es 192.168.10.223.

Reto 4.9 – Segmentar la siguiente topología de tal manera que:

- Las subredes de las VLANs deben ser calculadas conforme a la red 172.20.0.0/16.

- Las subredes de los enlaces seriales se crearán a partir de la red 192.168.0.0/24. Cada una de ellas debe permitir dos hosts como máximo.



Subredes para VLANs

Existen un total de 6 VLANs, por lo tanto, se requieren 3 bits para poder crearlas (2^3=8), que sumados a los 16 de la máscara por defecto establecen un prefijo /19. Los bits restantes serán destinados para hosts (13), por lo que cada subred permitirá un máximo de 8190 dispositivos.

172.20.00000000.00000000 = 172.20.0.0/19
172.20.00100000.00000000 = 172.20.32.0/19
172.20.01000000.00000000 = 172.20.64.0/19
172.20.01100000.00000000 = 172.20.96.0/19
172.20.10000000.00000000 = 172.20.128.0/19
172.20.10100000.00000000 = 172.20. 160.0/19
172.20.11000000.00000000 = 172.20.192.0/19
172.20.11100000.00000000 = 172.20.224.0/19

Subredes para enlaces seriales

En este caso se solicita que cada subred permita un total de dos dispositivos, por lo que tan solo resultan necesarios 2 bits para hosts. Los 6 restantes serán destinados a la creación de subredes, que sumados a los 24 de la máscara por defecto establecen un prefijo /30.

192.168.0.00000000 = 192.168.0.0/30
192.168.0.00000100 = 192.168.0.4/30
192.168.0.00001000 = 192.168.0.8/30
192.168.0.00001100 = 192.168.0.12/30
192.168.0.00010000 = 192.168.0.16/30
…
….Resultado omitido por brevedad

Reto 4.10 – En una topología VLSM basada en la red de clase A 10.0.0.0/8 se han seleccionado las siguientes direcciones IP, pertenecientes a cada una de las subredes existentes:

Subred A: 10.10.34.10/22
Subred B: 10.10.29.100/23
Subred C: 10.10.23.200/22
Subred D: 10.10.17.16/21
Subred E: 10.10.1.253/20

¿Se produce solapamiento de direcciones entre los diferentes segmentos?

Paso 1: Calcular el rango de cada una de las subredes, incluyendo su ID y dirección de broadcast:

10.10.34.10 /22
ID: 00001010.00001010.001000 00.00000000 = 10.10.32.0
Broadcast: 00001010.00001010.001000 11.11111111 = 10.10.35.255
Rango: 10.10.32.0 - 10.10.35.255

10.10.29.100 /23
ID: 00001010.00001010.0001110 0.00000000 = 10.10.28.0
Broadcast: 00001010.00001010.0001110 1.11111111 = 10.10.29.255
Rango: 10.10.28.0 - 10.10.29.255

10.10.23.200 /22
ID: 00001010.00001010.000101 00.00000000 = 10.10.20.0
Broadcast: 00001010.00001010.000101 11.11111111 = 10.10.23.255
Rango: 10.10.20.0 - 10.10.23.255

10.10.17.16 /21
ID: 00001010.00001010.00010 000.00000000 = 10.10.16.0
Broadcast: 00001010.00001010.00010 111.11111111 = 10.10.23.255
Rango: 10.10.16.0 - 10.10.23.255

10.10.1.253 /20

ID: 00001010.00001010.0000 0000.00000000 = 10.10.0.0

Broadcast: 00001010.00001010.0000 1111.11111111 = 10.10.15.255

Rango: 10.10.0.0 - 10.10.15.255

Paso 2: Crear un listado para compararlos:

Subred	**Rango**
Subred A	10.10.32.0 – 10.10.35.255
Subred B	10.10.28.0 – 10.10.29.255
Subred C	10.10.20.0 – 10.10.23.255
Subred D	10.10.16.0 – 10.10.23.255
Subred E	10.10.0.0 – 10.10.15.255

Examinándolo se puede concluir que entre las subredes C y D existe solapamiento de direcciones, ya que ambas hacen uso del rango comprendido entre las direcciones 10.10.20.0 y 10.10.23.255.

Reto 4.11. En una topología se ha aplicado VLSM sobre la red de clase C 192.168.255.0/24. Comprobar si entre las diferentes subredes existentes se produce solapamiento de direcciones.

Subred A: 192.168.255.0 /26
Subred B: 192.168.255.252 /30
Subred C: 192.168.255.128 /28
Subred D: 192.168.255.124 /30

Paso 1: Calcular el rango de cada una de las subredes creadas, incluyendo su ID y broadcast:

192.168.255.0 /26

ID: 11000000.10101000.11111111.00 000000 = 192.168.255.0

Broadcast: 11000000.10101000.11111111.00 111111 = 192.168.255.63

Rango: 192.168.255.0 - 192.168.255.63

192.168.255.252 /30

ID: 11000000.10101000.11111111.111111 00 = 192.168.255.252

Broadcast: 11000000.10101000.11111111.111111 11 = 192.168.255.255

Rango: 192.168.255.252 - 192.168.255.255

192.168.255.128 /28
ID: 11000000.10101000.11111111.1000 0000 = 192.168.255.128
Broadcast: 11000000.10101000.11111111.1000 1111 = 192.168.255.143
Rango: 192.168.255.128 - 192.168.255.143

192.168.255.124 /30
ID: 11000000.10101000.11111111.011111 00 = 192.168.255.124
Broadcast: 11000000.10101000.11111111.011111 11 = 192.168.255.127
Rango: 192.168.255.124 - 192.168.255.127

Paso 2: Crear un listado para compararlos:

Subred	**Rango**
Subred A	192.168.255.0 – 192.168.255.63
Subred B	192.168.255.252 – 192.168.255.255
Subred C	192.168.255.128 – 192.168.255.143
Subred D	192.168.255.124 – 192.168.255.127

Examinándolo se puede llegar a la conclusión de que no existe solapamiento de direcciones entre las subredes existentes.

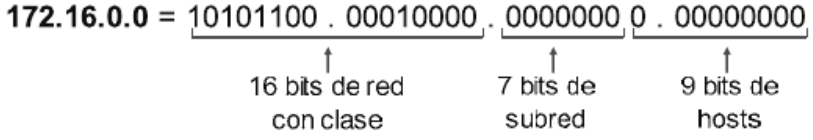
Reto 4.12 – Como administrador se le ha encomendado la tarea de agregar una nueva subred compuesta por 300 dispositivos a una topología VLSM ya existente formada por los siguientes segmentos:

172.16.4.0/22
172.16.8.0/24
172.16.10.128/25
172.16.255.0/30

Paso 1: Calcular una máscara de subred capaz de albergar 300 hosts:

$2^8 - 2 = 254$… No cumple los requisitos.
$2^9 - 2 = 510$… ¡Cumple los requisitos!

Destinando 9 bits para hosts restan 7 para subred, que sumados a los 16 de la máscara por defecto de clase B establecen un prefijo /23.



Paso 2: Identificar las subredes posibles en relación con la máscara calculada:

Subred Zero:
ID: 10101100.00010000.0000000 0.00000000 = 172.16.0.0
Broadcast: 10101100.00010000.0000000 1.11111111 = 172.16.1.255

Subred 1:
ID: 10101100.00010000.0000001 0.00000000 = 172.16.2.0
Broadcast: 10101100.00010000.0000001 1.11111111 = 172.16.3.255

Subred 2:
ID: 10101100.00010000.0000010 0.00000000 = 172.16.4.0
Broadcast: 10101100.00010000.0000010 1.11111111 = 172.16.5.255

Subred 3:
ID: 10101100.00010000.0000011 0.00000000 = 172.16.6.0
Broadcast: 10101100.00010000.0000011 1.11111111 = 172.16.7.255
…
… Resultado omitido por brevedad…

Subred 127:
ID: 10101100.00010000.1111111 0.00000000 = 172.16.254.0
Broadcast: 10101100.00010000.1111111 1.11111111 = 172.16.255.255

Paso 3: Calcular el rango de cada una de las subredes en uso:

172.16.4.0 /22
ID: 10101100.00010000.000001 00.00000000 = 172.16.4.0
Broadcast: 10101100.00010000.000001 11.11111111 = 172.16.7.255
Rango: 172.16.4.0 - 172.16.7.255

172.16.8.0 /24
ID: 10101100.00010000.00001000. 00000000 = 172.16.8.0
Broadcast: 10101100.00010000.00001000. 11111111 = 172.16.8.255
Rango: 172.16.8.0 - 172.16.8.255

172.16.10.128 /25
ID: 10101100.00010000.00001010.1 0000000 = 172.16.10.128
Broadcast: 10101100.00010000.00001010.1 1111111 = 172.16.10.255
Rango: 172.16.10.128 - 172.16.10.255

172.16.255.0 /30

ID: 10101100.00010000.11111111.000000 00 = 172.16.255.0

Broadcast: 10101100.00010000.11111111.000000 11 = 172.16.255.3

Rango: 172.16.255.0 - 172.16.255.3

Comparar las subredes existentes con las nuevas calculadas.

Rangos de red en uso
172.16.4.0 – 172.16.7.255
172.16.8.0 - 172.16.8.255
172.16.10.128 - 172.16.10.255
172.16.255.0 - 172.16.255.3

Rangos de nuevas subredes
172.16.0.0 - 172.16.1.255
172.16.2.0 - 172.16.3.255
172.16.4.0 - 172.16.5.255
172.16.6.0 – 172.16.7.255
…
…..
172.16.254.0 - 172.16.255.255

Paso 4: Descartar aquellas que produzcan solapamiento:

Rangos de nuevas subredes
172.16.0.0 - 172.16.1.255
172.16.2.0 - 172.16.3.255
~~172.16.4.0 - 172.16.5.255~~
~~172.16.6.0 – 172.16.7.255~~
…
…..
~~172.16.254.0 - 172.16.255.255~~

Paso 5: Seleccionar la subred que se aplicará a la topología.

La práctica solicita aquella con menor ID posible, por lo tanto, deberá ser implementada la subred zero (172.16.0.0 - 172.16.1.255).

TEST CAPÍTULO 4: SUBNETTING EN IPV4

1.- ¿Cuáles de los siguientes beneficios se obtienen mediante la aplicación de subredes? (Seleccionar dos respuestas)

A. Tablas de rutas más cortas.
B. Mayor velocidad de procesamiento de paquetes.
C. Mejor aprovechamiento del ancho de banda disponible.
D. Mejor aprovechamiento de direcciones IP.
E. Mayor disponibilidad de la red.

2.- Como máximo, ¿cuántos dispositivos permitirá la red con clase 192.168.1.0?

A. 255
B. 254
C. 253
D. 256

3.- Dada la siguiente topología…

¿Cuántas subredes son necesarias?

A. 8
B. 10
C. 18
D. 20
E. Ninguna de las respuestas anteriores es correcta.

4.- Se debe crear una subred que permita un máximo de 62 dispositivos. ¿Cuántos bits para hosts son necesarios para cumplir dicho requisito?

A. 5
B. 6
C. 7
D. 8

5.- Se ha segmentado la red con clase 172.20.0.0, aplicando sobre ella el prefijo /17. ¿Cuántas subredes se podrán crear?

A. 2
B. 4
C. 8
D. 64

6.- ¿Cuántos bits son necesarios para crear 20 subredes?

A. 1
B. 5
C. 10
D. 20

7.- Se requiere implementar una red con clase que permita albergar un mínimo de 1000 hosts. ¿Cuál de las siguientes opciones resulta la más adecuada teniendo en cuenta que se deben desaprovechar el menor número de direcciones posibles?

A. 10.0.0.0
B. 192.168.1.0
C. 172.20.0.0
D. 80.0.0.0

8.- A la red con clase 172.30.0.0 se le ha aplicado una máscara /20 con el fin de segmentarla. ¿Cuántas subredes se podrán crear y cuántos hosts como máximo permitirá cada una de ellas?

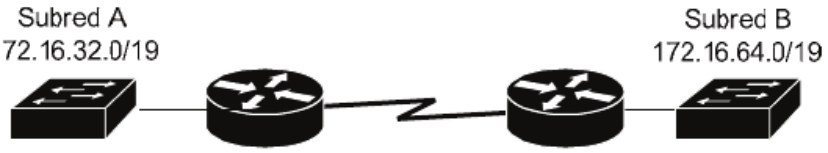
A. 16 subredes, 4096 hosts.
B. 16 subredes, 4094 hosts.
C. 32 subredes, 4096 hosts.
D. 32 subredes, 4094 hosts.
E. 14 subredes, 4096 hosts.
F. 14 subredes, 4094 hosts.

9.- Un dispositivo ha obtenido mediante DHCP la dirección IP 172.25.29.153 y máscara 255.255.240.0. ¿A qué subred pertenece?

A. 172.25.0.0
B. 172.25.128.0
C. 172.25.32.0

D. 172.25.192.0
E. 172.25.16.0

10.- Dada la siguiente topología…



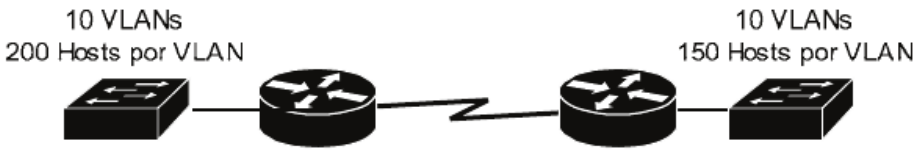
Un PC de la subred A con IP 172.16.45.100 desea comunicarse con otro de la subred B, con dirección 172.16.59.44. Sin embargo, la comunicación no tiene éxito. ¿A qué puede ser debido?

A. Ambas subredes deben estar conectadas físicamente al mismo router.
B. La dirección del PC ubicado en la subred B no pertenece a su rango.
C. Existe solapamiento de direcciones entre ambas subredes.
D. Ninguna de las respuestas anteriores es correcta.

11.- Un PC ha obtenido mediante DHCP los siguientes datos de conexión: IP – 10.10.10.10, Máscara – 255.255.128.0. ¿A qué subred pertenece?

A. 10.10.0.0
B. 10.0.0.0
C. 10.10.128.0
D. 10.10.10.0

12.- Como administrador de red se le ha encomendado la tarea de seleccionar una red con clase que permita su división en subredes cumpliendo los requisitos de la siguiente topología…



¿Cuál de las siguientes opciones se ajusta más a las necesidades, teniendo en cuenta que se deben desaprovechar el mínimo de direcciones posibles?

A. 192.168.100.0/24
B. 172.20.0.0/16

C. 10.0.0.0/8
D. Ninguna de las respuestas anteriores es correcta.

13.- ¿Qué máscara de subred utiliza la dirección IP 90.25.48.96/12?

A. 255.0.0.0
B. 255.16.0.0
C. 255.32.0.0
D. 255.64.0.0
E. 255.128.0.0

14.- Un PC con IP 172.16.125.1/18 desea enviar un broadcast a su subred. ¿A qué dirección debe hacerlo?

A. 172.16.255.255
B. 172.128.255.255
C. 172.16.125.255
D. 172.16.127.255
E. 172.16.0.255

15.- Un host realiza un ping con destino 127.0.0.1, obteniendo respuesta. ¿Qué conclusión se puede sacar de ello?

A. El host de destino está operativo y es accesible.
B. El router que actúa como puerta de enlace está operativo.
C. La tarjeta de red está instalada correctamente en el PC pero no dispone datos de red válidos.
D. El protocolo TCP/IP está instalado correctamente en el dispositivo.

16.- La máscara de red 11111111.11111111.11110000.00000000 es representada en formato decimal como:

A. 255.255.240.0
B. 255.255.192.0
C. 255.255.224.0
D. 255.255.248.0

17.- La máscara de red 11111111.11111111.11000000.00000000 también puede ser representada como: (Seleccionar dos respuestas)

A. /17

B. /18
C. /19
D. 255.255.128.0
E. 255.255.192.0
F. 255.255.64.0

18.- ¿Cuál de las siguientes direcciones corresponde a un broadcast de una red con clase?

A. 192.168.255.255
B. 172.16.16.255
C. 10.0.255.255
D. 192.168.1.0

19.- ¿Qué prefijo de red representa la máscara 255.255.248.0?

A. /20
B. /21
C. /22
D. /23

20.- Se ha configurado un servidor DHCP para la subred 192.168.1.128/27. ¿Cuáles de las siguientes direcciones IP no podrán ser suministradas a los clientes? (Seleccionar dos respuestas)

A. 192.168.1.159
B. 192.168.1.145
C. 192.168.1.129
D. 192.168.1.128
E. 192.168.1.132
F. 192.168.1.150

21.- ¿Por qué la dirección IP 192.168.100.31 no puede ser asignada a ningún host perteneciente a la subred 192.168.100.16/28?

A. Porque no pertenece a la subred mencionada.
B. Sí puede ser asignada.
C. Porque es un ID de red.
D. Porque es una dirección de broadcast.

22.- ¿Qué ventaja aporta VLSM respecto a un diseño de subredes sin máscara variable?

A. Gracias a VLSM se puede hacer uso de menos routers en la red.
B. Gracias a VLSM el tamaño de las tablas de rutas es menor y por lo tanto la velocidad de procesamiento mayor.
C. Gracias a su uso el desaprovechamiento de direcciones es mínimo.
D. Gracias a su uso se puede aplicar la misma subred en diferentes puntos de la topología.

23.- ¿Cuál de los siguientes representa el error más común en un diseño VLSM?

A. Hacer uso de la misma subred en diferentes segmentos físicos.
B. Aplicar la misma máscara en varias subredes.
C. Que las rutas no sean publicadas por los protocolos de enrutamiento.
D. Un cálculo erróneo que provoque solapamiento de direcciones.

24.- ¿En qué consiste la sumarización de rutas?

A. En seleccionar la mejor ruta hacia un destino e instalarla en la tabla.
B. Es el número total de rutas utilizadas por un dispositivo.
C. Consiste en definir una sola interfaz de salida para todas las rutas instaladas en la tabla.
D. Se basa en resumir en una sola entrada un conjunto de rutas.

25.- Una topología VLSM aplica en uno de sus segmentos la dirección 10.5.48.0/20. ¿Cuál de las siguientes subredes no produce solapamiento con ella?

A. 10.5.0.0/20
B. 10.4.0.0/15
C. 10.5.32.0/19
D. 10.5.0.0/17

26.- De las siguientes subredes, ¿cuál produce solapamiento de direcciones con la 172.30.8.0/22?

A. 172.30.11.0/25
B. 172.30.16.0/20
C. 172.30.6.0/23
D. 172.30.0.0/21

CONFIGURACIÓN INICIAL DE ROUTERS CISCO 5

INSTALACIÓN DE ROUTERS CISCO

Los routers, presentes en cualquier infraestructura de red y considerados elementos críticos, son los dispositivos encargados de llevar a cabo la comunicación entre diferentes redes. Operan en capa 3, siendo su función principal la de analizar la dirección IP de destino incluida en cada paquete y tomar la decisión de reenvío más adecuada, basándose para ello en la tabla de rutas almacenada en su propia memoria, la cual se genera y actualiza gracias a la configuración manual por parte del administrador, o de manera automática gracias a los protocolos de enrutamiento. El presente capítulo abordará la puesta en marcha, conceptos más básicos y rutas estáticas en dispositivos Cisco, mientras que los protocolos de enrutamiento, que requieren un análisis más detallado, serán objeto de estudio en el capítulo 6.

Un detalle importante antes de proceder a su configuración consiste en definir cuántos routers son necesarios y la ubicación física de estos. La instalación de un número muy elevado de ellos implica un menor rendimiento en la red, ya que por lo general (dependiendo del modelo) toman la decisión de reenvío mediante software, dando como resultado un procesamiento más lento que en los switchs, cuya manipulación de tramas se realiza directamente desde el hardware. Un diseño óptimo es aquel en el que se logra enrutar hacia las diferentes subredes de manera centralizada. Para llevarlo a cabo, lo más habitual consiste en hacer uso de una sede central o núcleo (CPD), el cual ubicará los elementos críticos de red y se encargará, entre otras, de esta misión. Para las sedes externas bastará con un router que conecte con la sede central.

Un ejemplo podría ser el siguiente:

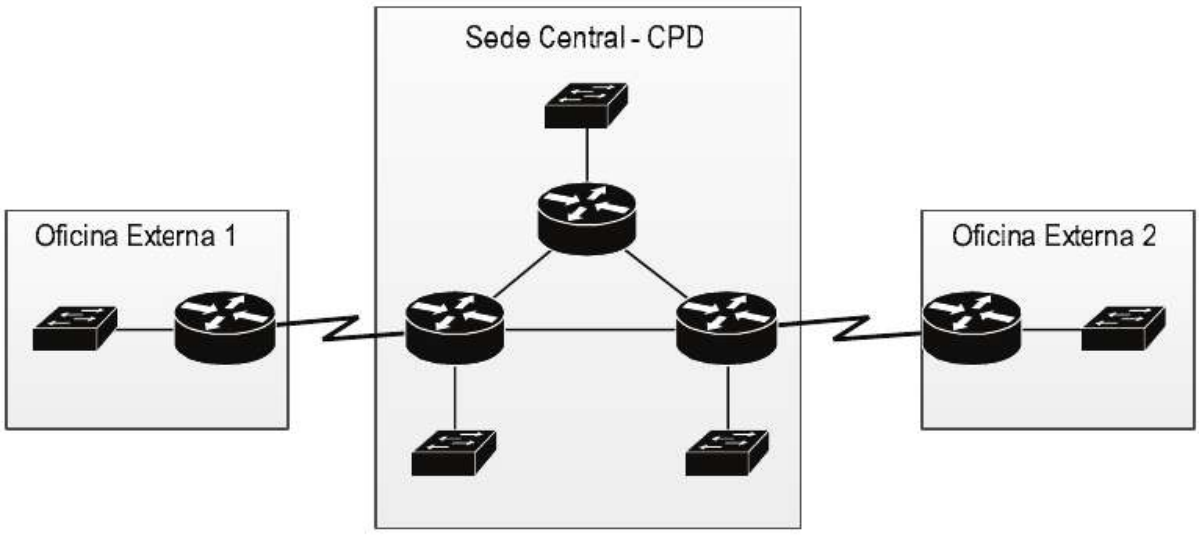


Fig. 5-1 Distribución de routers en red corporativa.

¿Qué router instalar en cada ubicación?

Cisco dispone de una amplia variedad de opciones, la gran mayoría calificados como "*Routers de servicios integrados*" *(ISR),* que son aquellos que no solo realizan la función básica de enrutar paquetes, sino que además agregan servicios adicionales como seguridad, VPN, DHCP, conexiones DSL, gestión de usuarios, etc. Con relación al modelo seleccionado se dispondrá de más o menos funciones. Para la sede central resulta recomendable la instalación de routers de alto rendimiento, ya que requieren mayor capacidad de hardware que aquellos ubicados en oficinas externas. Para estos últimos se podrá optar por modelos más sencillos, siempre y cuando se adapten a las necesidades de la compañía.

¿Cómo es físicamente un router Cisco corporativo?

También dependerá del modelo seleccionado, pero todos tienen en común una o varias conexiones Ethernet, puerto de consola, conexión auxiliar, y por lo general, slots de expansión denominados WIC (*Wan Interface Card*). El ejemplo mostrado a continuación corresponde a un router Cisco 2901 (*2900 Series*).

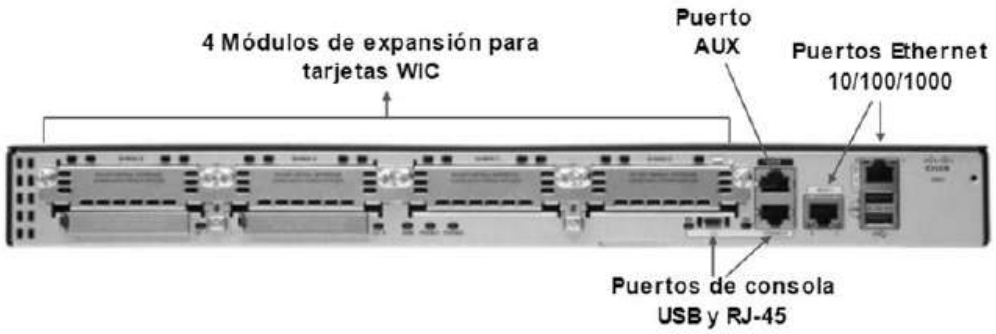


Fig. 5-2 Puertos disponibles en un router Cisco 2900 Series.

- *Puertos Ethernet 10/100/1000*: Son las interfaces que conectan directamente con las diferentes redes y a través de las cuales se reenvían y reciben paquetes.

- *Puertos de consola USB y RJ-45*: Son utilizados para acceder a la configuración (CLI) de manera local. Los modelos antiguos solo incluyen la interfaz RJ-45.

- *Puerto AUX:* Necesario si se desea habilitar el acceso remoto al router a través de un módem (*out of band*).

- *Módulos de expansión:* Es el hardware utilizado para agregar funcionalidades al router, como más puertos Ethernet o un módulo CSU/DSU. Su instalación se lleva a cabo mediante tarjetas denominadas WIC.

Instalación física de un router corporativo

Como se ha descrito, cada interfaz cumple un determinado propósito. De todas ellas, los puertos Ethernet son aquellos que permiten y establecen la comunicación entre las diferentes redes, ya que actúan como puerta de enlace para los dispositivos ubicados en cada una de ellas. Un ejemplo podría ser el siguiente:

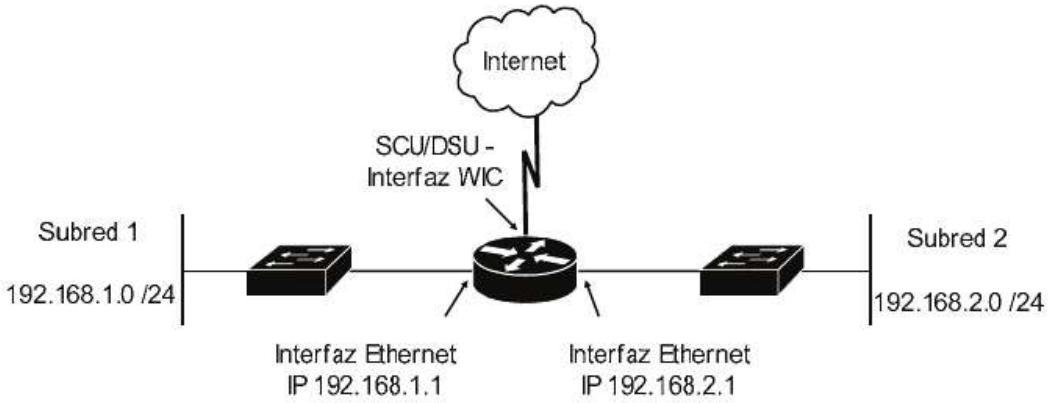


Fig. 5-3 Tipos de conexión a través de Interfaces en un router.

Donde cada interfaz dispone de una dirección IP perteneciente al rango de la subred con la cual conecta. Evidentemente, estas deben ser configuradas manualmente, lo cual implica la intervención del administrador para que las funciones del router se ejecuten. Este aspecto representa una gran diferencia respecto a los switchs, ya que estos, sin ningún tipo de configuración inicial, dotan de conexión a los dispositivos que conectan en sus interfaces.

Routers de acceso a Internet

Otro tipo de routers son los denominados SOHO *(Small office home office),* enfocados a pequeñas empresas y hogares y cuya función principal consiste en crear una pequeña red, dotando de conexión a Internet a los usuarios de esta mediante tecnologías como DSL, cable o fibra. De hecho, son los dispositivos instalados por los ISP en los hogares de sus suscriptores. Comúnmente reciben el nombre de "*routers*" pero realmente incorporan los siguientes elementos:

- Router.
- Switch.
- Módem.
- Punto de acceso.

Su instalación resulta, con creces, más sencilla que la llevada a cabo en una red corporativa y su utilización implica una serie de ventajas e inconvenientes. Como ventajas cabe destacar la ya mencionada facilidad de instalación y una administración centralizada de todas sus funciones. Sin embargo, como inconvenientes se podrían identificar un menor rendimiento durante el procesamiento de paquetes, la existencia de un único punto de falla y el nivel de seguridad permitido, que aunque en algunos modelos se considere aceptable, siempre será menor que aquel implementado sobre un diseño corporativo, ya que en este último cada dispositivo es administrado de manera individual, hecho que permite definir diferentes capas de seguridad.

Configuración básica de interfaces en routers Cisco

Los routers Cisco, al igual que los switchs, hacen uso de IOS como sistema operativo, por lo tanto, muchos de los comandos de configuración ya analizados en el capítulo 2 se aplicarán de igual manera a lo largo de la presente sección, como el acceso a las interfaces a través del comando **interface *[interfaz].***

En párrafos anteriores se ha mencionado que cada modelo de router dispondrá de determinadas características de hardware, lo cual incluye el número y tipo de

puertos disponibles. El primer paso a llevar a cabo consiste en determinar qué interfaces incluye el dispositivo en cuestión, tarea que se podrá realizar de dos maneras, ya sea físicamente, o bien a través de comando **show ip interfaces brief**, el cual una vez ejecutado mostrará en pantalla un listado de aquellas presentes y su estado actual.

```
Router#show ip interface brief
Interface                  IP-Address      OK? Method  Status
Protocol

FastEthernet0/0            unassigned      YES unset   administratively down down
FastEthernet0/1            unassigned      YES unset   administratively down down
Serial0/1/0                unassigned      YES unset   administratively down down
Serial0/1/1                unassigned      YES unset   administratively down down
```

En este caso, el router dispone de 2 interfaces FastEthernet y 2 Serial, ninguna de ellas con dirección IP asignada. "*Status"* hace referencia a la línea, es decir, capa 1, mientras que "*Protocol"* identifica los protocolos y configuración propia de capa 2. La combinación de dichos campos determinará el estado real de la interfaz, la cual únicamente estará operativa cuando el valor de ambos sea "*up*". Cualquier otro resultado identifica algún tipo de problema, siendo los más habituales los siguientes:

Status (Línea)	**Protocol (Protocolo)**	**Razón**
Administratively down	*Down*	La interfaz se encuentra administrativamente deshabilitada (*shutdown*).
Down	*Down*	Interfaz no operativa debido a algún error en capa 1, posiblemente cable no conectado, defectuoso o dispositivo del otro extremo apagado.
Up	*Down*	Falta de concordancia del protocolo aplicado en ambos extremos. Normalmente debidos a errores de configuración.
Up	*Up*	Interfaz operando con normalidad.

Inicialmente, todas las interfaces se encuentran en estado "*administratively down/down".* Ello es debido a la configuración por defecto incluida en routers Cisco. Para activarlas bastará con ejecutar el comando **no shutdown** sobre cada una de ellas.

CONFIGURACIÓN DE INTERFACES ETHERNET

Para que una interfaz comience a enrutar tráfico desde y hacia la red a la que pertenece tan solo bastará con asignarle una IP y máscara, para posteriormente habilitarla. Para ello se deberán ejecutar las siguientes acciones:

Paso 1: Acceder a la interfaz con el comando **interface *[interfaz]***, desde el modo de configuración global.

Paso 2: Asignarle una dirección IP y máscara de red, a través de la sentencia **ip address *[dir IP] [máscara]***. Evidentemente, la IP debe pertenecer al rango de red con el cual conecta con dicha interfaz.

Paso 3: Habilitarla con el comando **no shutdown.**

Ejemplo: Configurar R1 de tal manera que lleve a cabo el enrutamiento entre los diferentes segmentos de red.

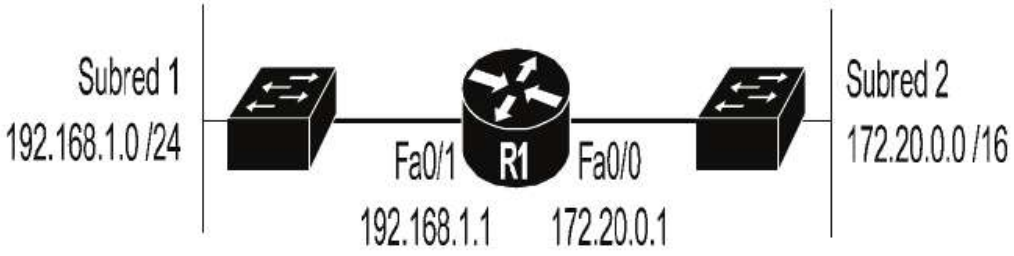


Fig. 5-4 Configuración de interfaces Ethernet.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface fastethernet 0/0
R1(config-if)#ip address 172.20.0.1 255.255.0.0
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#interface fastethernet 0/1
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shutdown
```

Aplicados los cambios y conectado correctamente el cableado en ambos extremos, se obtendría el siguiente resultado tras ejecutar un *show ip interface brief*.

```
R1#show ip interface brief
Interface                  IP-Address      OK? Method  Status
Protocol
```

```
FastEthernet0/0        172.20.0.1      YES manual  up                    up
FastEthernet0/1        192.168.1.1     YES manual  up                    up
Serial0/1/0            unassigned      YES unset   administratively down down
Serial0/1/1            unassigned      YES unset   administratively down down
```

CONFIGURACIÓN DE INTERFACES SERIAL

Su configuración se lleva a cabo de igual manera que la recién analizada, con una pequeña diferencia, y es que este tipo de enlaces son considerados WAN y como tal se debe establecer un ancho de banda entre ambos extremos.

La conexión física se realiza mediante un cable serial, que por un extremo actúa como DTE y por el otro como DCE. El DCE establece y controla el ancho de banda del enlace, mientras que el DTE lo acepta y aplica. Por lo tanto, a la hora de configurarlo, en la interfaz del router que actúa como DCE también se debe configurar el ancho de banda que se desee aplicar.

Para ello se debe proceder de la siguiente manera:

Paso 1: Acceder a la interfaz mediante el comando **interface *[interfaz]***, desde el modo de configuración global.

Paso 2: Asignarle una dirección IP y máscara a través del comando **ip address *[dir IP] [máscara]***.

Paso 3: Configurar el ancho de banda del enlace tan solo en el router que actúa como DCE, ejecutando para ello la sentencia **clock rate *[bps]***, donde *bps* indica la velocidad, en bits por segundo, que se aplicará. Por ejemplo, para un enlace de 56 Kbps se configurarán 56000 bps.

Paso 4: Habilitar la interfaz mediante un **no shutdown.**

Ejemplo: Se ha agregado un nuevo router a la red, el cual conecta con R1 a través de un enlace serial y cuyo ancho de banda será gestionado por R2, aplicando una velocidad de 64 Kb. Configurar ambos de acuerdo a la topología.

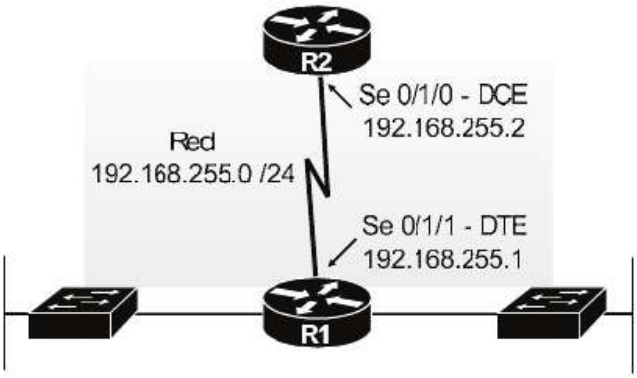


Fig. 5-5 Configuración de interfaces serial.

Configuración de R1 - DTE

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface Serial0/1/1
R1(config-if)#ip address 192.168.255.1 255.255.255.0
R1(config-if)#no shutdown
```

Configuración de R2 - DCE

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface Se0/1/0
R2(config-if)#ip address 192.168.255.2 255.255.255.0
R2(config-if)#clock rate 64000
R2(config-if)#no shutdown
```

El enlace WAN creado consta de 64 kbps. Esta velocidad sobre entornos reales resulta prácticamente nula. Se ha optado por ella como ejemplo para una mayor facilidad de comprensión durante la conversión Kb-bps. Normalmente, el ancho de banda (*clock rate*) es definido por el dispositivo CSU/DSU, por lo tanto, su configuración depende del ISP.

Una vez concluida su configuración, IOS dispone de diferentes opciones para verificarla y a su vez comprobar el estado de cada interfaz. Además del ya analizado "*show ip interfaces brief*", se podrá hacer uso de:

- *show protocols [interfaz]:* Muestra información básica como el estado de la interfaz, dirección IP, máscara, etc.

- *show interfaces [interfaz]:* Muestra información más detallada, incluyendo características de hardware, contadores, protocolos aplicados, dirección IP, máscara…

Una vez configuradas y operativas, el router es capaz de comunicar las diferentes redes gracias al enrutamiento, pero, ¿en qué consiste y cómo se lleva a cabo?

ENRUTAMIENTO Y RUTAS ESTÁTICAS

El enrutamiento se define como el proceso de reenvío de paquetes iniciado por el host y ejecutado por el router, teniendo como finalidad la correcta comunicación entre dispositivos pertenecientes a diferentes redes. En otras palabras, entregar y dirigir paquetes desde una red de origen hacia otra de destino. Para llevarlo a cabo, tanto hosts como routers aplican diferentes métodos, consistentes en:

Lógica de enrutamiento en hosts

El host de origen crea un paquete IP que debe ser enviado a un destino. Conforme a la ubicación de este el proceso varía, ejecutando la siguiente lógica.

Si ambos dispositivos pertenecen a la misma red, el origen solicita la MAC del destino haciendo uso del protocolo ARP. Este último responderá a la petición. Tras obtenerla, tanto en capa 2 como 3 se incluyen las direcciones del destinatario final.

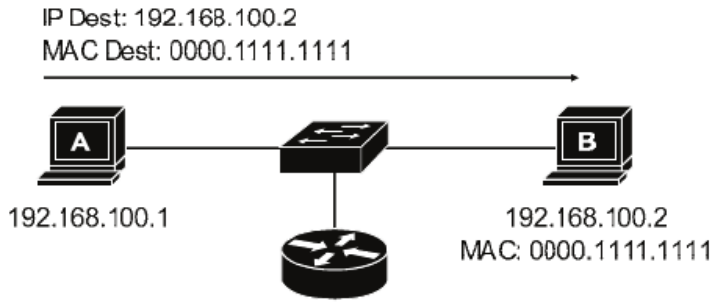


Fig. 5-6 Comunicación entre hosts ubicados en la misma red.

En este caso, A desea enviar un paquete a B. Como ambos forman parte de la misma red la comunicación se lleva a cabo de manera directa, sin la intervención del router. Para ello, A obtiene la MAC de B gracias al protocolo ARP, para luego agregarla en la trama creada en capa 2. Mientras, en capa 3, el destino también hace referencia a la dirección IP de B. Cuando los datos son enviados al medio son recibidos por el Switch, que lee la información incluida en capa 2, consulta su tabla de MACs y ejecuta el reenvío directamente a través de la interfaz que conecta con B.

Sin embargo, si el host de destino forma parte de una red diferente a la del origen, este debe enviar el paquete a su puerta de enlace, es decir, a un router. Para

ello solicita su MAC mediante una petición ARP. Tras obtenerla, el paquete es creado en capa 3 con la IP del host de destino y en capa 2 con la MAC de la puerta de enlace.

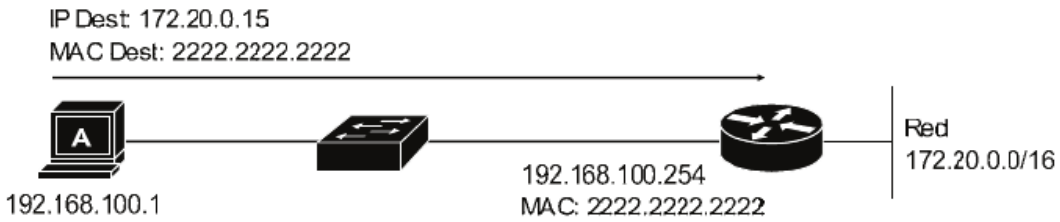


Fig. 5-7 Comunicación entre hosts ubicados en diferentes redes.

El host A desea enviar un paquete a la IP 172.20.0.15. Para ello calcula la red a la que pertenece dicha dirección y comprueba que no coincide con la suya propia, por lo que debe enviar los datos a su puerta de enlace, que en este caso es el router con IP 192.168.100.254. Acto seguido envía una petición ARP solicitando su MAC. La dirección obtenida es agregada como destino en la trama creada en capa 2, mientras que en capa 3 se mantiene la IP del host final, en este caso 172.20.0.15. Cuando los datos son enviados al medio son recibidos por el switch, que lee la dirección incluida en capa 2, consulta su tabla de MACs y los reenvía directamente a través de la interfaz que conecta con el router, el cual, a su vez, se encargará de redirigirlos hacia el destino correcto.

Lógica de enrutamiento en routers

Una vez los datos son recibidos por el router se aplica el procedimiento utilizado por estos dispositivos, algo más complejo que el llevado a cabo por los hosts y basado en:

Paso 1: Comprobar los campos FCS y MAC de cada trama recibida. Los routers reciben gran cantidad de tramas a través de sus interfaces, debiendo analizar cada una de ellas y seleccionar cuáles serán procesadas y cuáles no.

El primer paso llevado a cabo consiste en calcular el valor FCS y compararlo con el recibido. Si ambos no coinciden significa que se ha producido algún tipo de error durante la transmisión y por lo tanto se descarta la trama automáticamente.

Si por el contrario coinciden, se comprueba el campo “MAC de destino”, cuya dirección incluida debe corresponder con la MAC de la interfaz del

router por la cual fue recibida la comunicación, de no ser así, esta será descartada automáticamente.

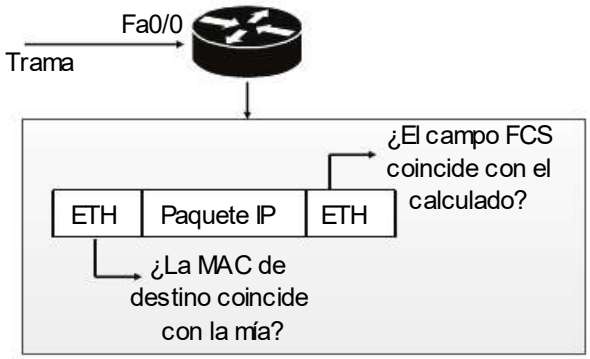


Fig. 5-8 Verificación de campos FCS y MAC.

Cuando ambas comprobaciones resultan correctas, el router continúa el procedimiento en el paso 2.

Paso 2: El siguiente paso consiste en obtener la IP del destinatario. Para ello, el router debe desencapsular la trama, gracias a lo cual accede al paquete IP y lee el campo "dirección de destino", proceso que se lleva a cabo eliminando la cabecera y el tráiler Ethernet (capa 2).

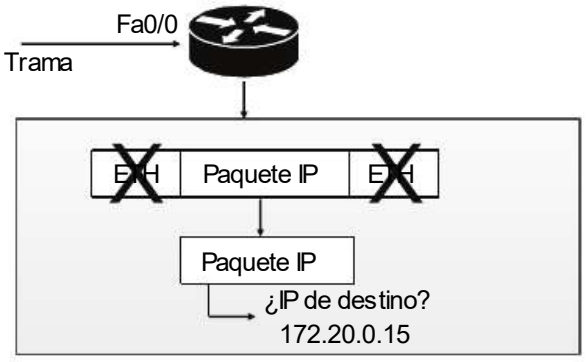


Fig. 5-9 Desencapsulado de trama.

Paso 3: Una vez obtenida la IP, el router debe seleccionar la interfaz por la cual reenviar los datos. Para ello, calcula la red a la que pertenece dicha dirección y comprueba en su tabla de rutas la interfaz necesaria para acceder a la misma.

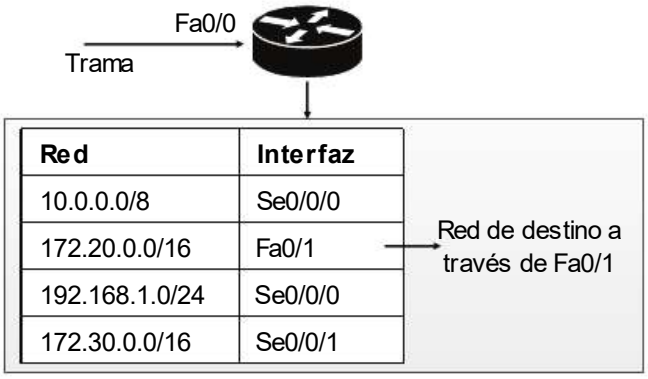


Fig. 5-10 Selección de interfaz de reenvío.

Si la red de destino no se encontrara en la tabla de rutas, el paquete es descartado. Para evitarlo se puede optar por definir una ruta por defecto en el dispositivo, configuración que será analizada en párrafos posteriores.

Paso 4: El router ha comprobado su tabla de rutas y ha seleccionado la interfaz Fa0/1 para reenviar los datos. El siguiente paso consiste en crear el paquete IP, el cual, a su vez, será encapsulado en una trama. En capa 3, las direcciones de origen y destino no se modifican, siempre se utilizarán las mismas que fueron incluidas por el host de origen. Sin embargo, en capa 2 sí que varían a lo largo de la comunicación. Al crear la trama, la MAC de origen corresponde a la dirección de la interfaz del router por la cual se va a realizar el reenvío, mientras que la de destino identifica al dispositivo a quien va dirigida la comunicación. Este último puede ser otro router, el propio host de destino o cualquier otro dispositivo intermediario.

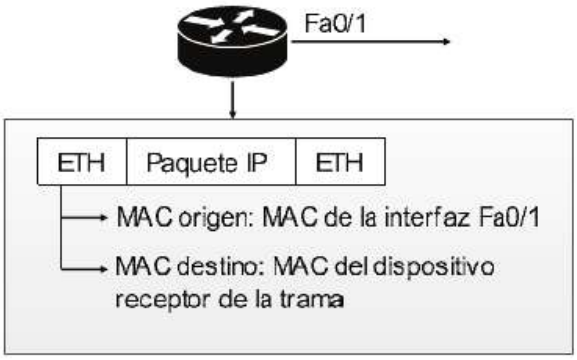


Fig. 5-11 Creación y reenvío de trama.

Paso 5: Por último, el router reenvía la trama creada a través de la interfaz seleccionada.

Procesamiento interno en routers Cisco

Para los routers, un volumen de tráfico normal en la red equivale a la recepción de miles y miles de paquetes por segundo a través de cada una de sus interfaces, debiendo ejecutar sobre cada uno de ellos la lógica de reenvío recién analizada. A este hecho hay que sumarle que en redes de gran tamaño la tabla de rutas puede almacenar cientos de entradas, lo que supone mayor lentitud a la hora de buscar la interfaz de salida para cada paquete. Una de las características más importantes que deben tener estos dispositivos es la velocidad de procesamiento, la cual puede verse afectada por dichas causas. Se estima que para que un router tenga un rendimiento aceptable debe ser capaz de reenviar decenas de miles de paquetes por segundo (pps).

Para lograrlo, Cisco ha desarrollado e integrado diversas técnicas a lo largo del tiempo, lo que se conoce como procesamiento interno de routers. La primera de ellas se basa en la lógica de enrutamiento recién analizada, denominada *process switching*.

Más adelante se introducen algunas mejoras sobre dicho modelo, entre las que se incluye la utilización de una lista que almacena las direcciones IP de los últimos paquetes enviados, lo que puede entenderse como una memoria caché y cuyo fin consiste en acelerar el reenvío de paquetes. Este nuevo modo fue denominado *fast switching.*

Por último, a finales de los 90, Cisco desarrolla un tercer método denominado *Cisco Express Forwarding (CEF),* el cual basa la diferencia principal con sus antecesores en la inclusión de tablas adicionales para realizar búsquedas más rápidas en la tabla de enrutamiento. Además, las cabeceras de capa 2 son almacenadas para cada destino con el fin de no tener que crearlas repetidamente para cada trama. A día de hoy los routers Cisco aplican CEF por defecto.

Configuración de rutas y enrutamiento InterVLAN

A lo largo del capítulo se ha mencionado la tabla de rutas en numerosas ocasiones, la cual almacena las diferentes redes existentes asociándolas con la interfaz de salida necesaria para llegar a cada una de ellas, siendo su función de extrema importancia para que el enrutamiento se lleve a cabo de manera correcta. Pero, ¿cómo aprende el router esta información?

Los routers crean, agregan y modifican su tabla de rutas conforme a:

- *Rutas directamente conectadas:* Son agregadas automáticamente e identifican la red a la que pertenece cada una de las interfaces configuradas con el comando *ip address.*

- *Rutas estáticas:* Son configuradas manualmente por el administrador.

- *Protocolos de enrutamiento:* Son las redes aprendidas de manera dinámica gracias a protocolos de enrutamiento.

RUTAS DIRECTAMENTE CONECTADAS

Las rutas directamente conectadas identifican las redes a las que pertenece cada una de las interfaces del router. Estas son agregadas automáticamente a la tabla siempre que se cumplan las siguientes condiciones:

- La interfaz que conecta con la red debe estar configurada con una IP perteneciente al mismo rango.
- La interfaz esté habilitada.
- La interfaz se encuentre en estado *"up" "up".*

Realmente el tercer requisito engloba los dos anteriores, por lo que puede afirmarse que para que una ruta directamente conectada sea agregada a la tabla de rutas, la interfaz que conecta con ella debe encontrarse en estado *"up" "up".*

La topología utilizada como ejemplo para este capítulo dispone de 2 routers con sus interfaces configuradas y operativas, por lo tanto, las redes con las que conectan deberían ser agregadas automáticamente a la tabla:

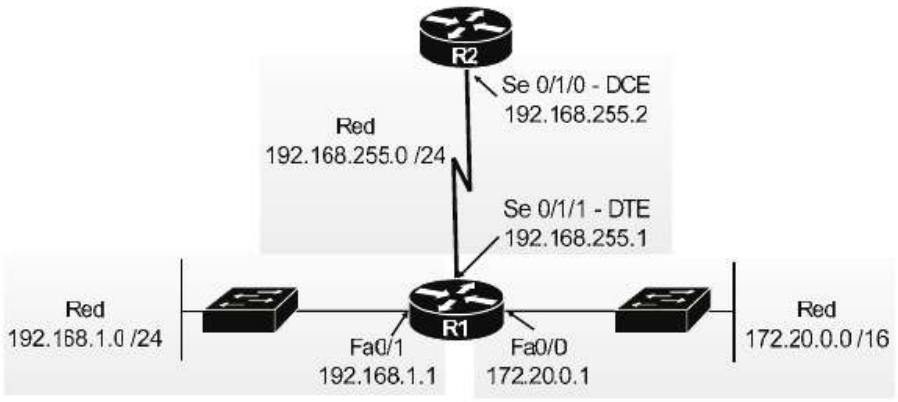


Fig. 5-12 Topología de red corporativa.

¿Cuáles son las redes directamente conectadas a R1?

La 172.20.0.0/16 a través de su interfaz Fa0/0
La 192.168.1.0/24 a través de su interfaz Fa0/1
La 192.168.255.0/24 a través de su interfaz Se0/1/1

¿Y a R2?

La red 192.168.255.0/24 a través de su interfaz Se0/1/0

Todas las interfaces se encuentran en estado "*up/up*" (comprobado anteriormente mediante un *show ip interfaces brief)*, por lo tanto, se cumplen las 3 condiciones para que las redes sean agregadas a la tabla de rutas de cada dispositivo. Para verificarlo bastará con ejecutar el comando *show ip route.*

```
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    172.20.0.0/16 is directly connected, FastEthernet0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/1
C    192.168.255.0/24 is directly connected, Serial0/1/1



R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.255.0/24 is directly connected, Serial0/1/0
```

Como se puede observar, la primera información mostrada corresponde a una serie de códigos, los cuales identifican la manera en que ha sido aprendida la ruta. En este caso, en todas ellas se aplica el mismo método, "C", que significa "*directamente conectada*".

Enrutamiento InterVLAN

En el capítulo 2 fueron analizados los métodos disponibles para ejecutar el enrutamiento entre diferentes VLANs, llegando a la conclusión de que el más apropiado y escalable es el denominado "*router-on-a-stick*", el cual hace uso de un solo enlace físico entre el switch y el router para llevar a cabo toda la comunicación, siendo necesario para ello la utilización de interfaces virtuales (subinterfaces), las cuales posibilitarán el enrutamiento.

El proceso de configuración necesario para lograr dicho propósito consta de las siguientes acciones:

- *Paso 1:* Configurar la interfaz del switch como enlace troncal, permitiendo el acceso de las VLANs que se desean comunicar. (Ya analizado en el capítulo 2).

- *Paso 2:* Crear las subinterfaces necesarias en el router con el comando **interface *[interfaz].[subinterfaz]*** desde el modo de configuración global. Debe definirse una para cada VLAN.

- *Paso 3:* En cada subinterfaz, aplicar el protocolo 802.1Q para que las tramas sean etiquetadas con la VLAN correspondiente. Para ello debe ejecutarse la sentencia **encapsulation dot1q *[vlan],*** donde *vlan* indica el id de esta que se desea aplicar.

- *Paso 4:* En cada subinterfaz configurar una IP con el comando **ip address *[ip] [máscara]***, la cual debe formar parte del rango de la subred de la VLAN a la que pertenece.

- *Paso 5:* Una vez configuradas todas las subinterfaces, ejecutar el comando **no shutdown** desde el modo de configuración de **la interfaz física**.

Si no existe ningún error en capa 1 entre el switch y el router, el proceso de configuración dará como resultado que la interfaz entre en estado *"up" "up"*, por lo que a su vez se habrán cumplido las tres condiciones para que se incluyan las diferentes subredes en la tabla de rutas como directamente conectadas.

Por ejemplo, en relación con la topología mostrada en la *Fig. 5-12,* imagina que la red crece y se agrega un nuevo switch, conectado a la interfaz Fa0/0 de R2 y configurado con 3 VLAN (10, 20 y 30), debiendo estar comunicadas entre sí. Configurar R2 para que el enrutamiento entre todas ellas se pueda llevar a cabo.

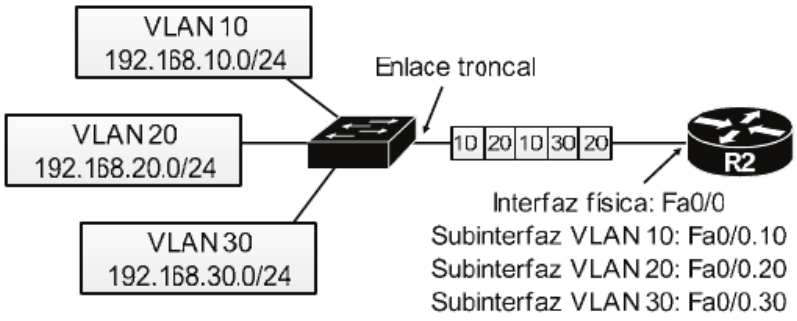


Fig. 5-13 Subinterfaces para enrutamiento InterVLAN.

Aplicando la configuración necesaria...

```
R2(config)#interface FastEthernet0/0.10
R2(config-subif)#encapsulation dot1q 10
R2(config-subif)#ip address 192.168.10.1 255.255.255.0
R2(config-subif)#exit
R2(config)#interface FastEthernet0/0.20
R2(config-subif)#encapsulation dot1q 20
R2(config-subif)#ip address 192.168.20.1 255.255.255.0
R2(config-subif)#exit
R2(config)#interface FastEthernet0/0.30
R2(config-subif)#encapsulation dot1q 30
R2(config-subif)#ip address 192.168.30.1 255.255.255.0
R2(config-subif)#exit
R2(config)#interface Fastethernet0/0
R2(config-if)#no shutdown
```

Tras ello, tanto la interfaz física como las subinterfaces deben entrar en estado *"up" "up"*. Para verificarlo bastará con ejecutar un *show ip interface brief…*

```
R2#show ip interface brief
Interface                  IP-Address      OK? Method Status
Protocol

FastEthernet0/0            unassigned      YES unset  up                    up
FastEthernet0/0.10         192.168.10.1    YES manual up                    up
FastEthernet0/0.20         192.168.20.1    YES manual up                    up
FastEthernet0/0.30         192.168.30.1    YES manual up                    up
FastEthernet0/1            unassigned      YES unset  administratively down down
Serial0/1/0                192.168.255.2   YES manual up                    up
Serial0/1/1                unassigned      YES unset  administratively down down
```

A su vez, las nuevas redes configuradas serán agregadas automáticamente en la tabla de rutas como directamente conectadas…

```
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.10.0/24 is directly connected, FastEthernet0/0.10
C    192.168.20.0/24 is directly connected, FastEthernet0/0.20
C    192.168.30.0/24 is directly connected, FastEthernet0/0.30
C    192.168.255.0/24 is directly connected, Serial0/1/0
```

¿Cómo procede R2 si recibe un paquete por la interfaz Se0/1/0 con IP de destino 192.168.20.50? Comprueba la tabla de rutas y selecciona la interfaz Fa0/0.20 como salida hacia la red de destino. En capa 2 agrega una etiqueta de VLAN 20 y lo reenvía por la interfaz física Fa0/0.

¿Y si recibe un paquete por la interfaz Fa0/0 con IP de destino 192.168.30.100? Elimina la trama en capa 2 y lee la dirección de destino en capa 3. Busca en la tabla de rutas y selecciona la interfaz Fa0/0.30 como salida hacia la red de destino. En capa 2 agrega una etiqueta de VLAN 30 y la reenvía por la interfaz física Fa0/0.

En las subinterfaces, la VLAN que se desea comunicar es definida mediante el comando *encapsulation dot1q [vlan]*. Por ejemplo, para la VLAN 10 se ejecuta la sentencia "*encapsulation dot1q 10*". Sin embargo, el número de subinterfaz no tiene por qué coincidir con el id de VLAN. Por facilidad de administración y comprensión se ha optado por asociarlas (VLAN 10, subinterfaz 0.10), aunque no fuera necesario.

ENRUTAMIENTO INTERVLAN EN SWITCHS DE CAPA 3

Cada vez resulta más común la implementación de switchs de capa 3 (también denominados multicapa) en algunos puntos de la red como alternativa a los routers. Estos, como su nombre indica, son aquellos capaces de operar con paquetes IP, por lo tanto, pueden ejecutar el enrutamiento entre diferentes redes. Esta característica los convierte en la opción ideal para comunicar VLANs sin la necesidad de un router para ello, ya que el mismo switch, correctamente configurado, llevaría a cabo dicha tarea.

Aplicado al ejemplo anterior *(Fig. 5-13):*

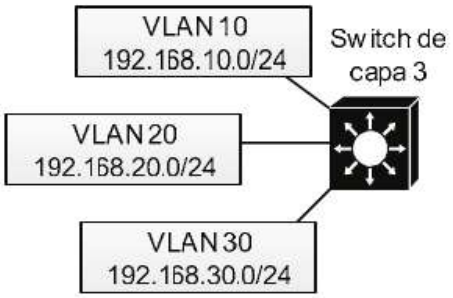


Fig. 5-14 Enrutamiento InterVLAN en switch de capa 3.

En este caso, el enrutamiento es ejecutado de manera interna por el switch, sin necesidad de ningún enlace físico hacia un router para ello. La configuración se basa en definir interfaces virtuales, una por cada VLAN, a través de las cuales se llevará a cabo la comunicación. Para lograrlo, bastará con aplicar el siguiente procedimiento:

Paso 1: Habilitar el enrutamiento con el comando **ip routing** desde el modo de configuración global.

Paso 2: Crear una interfaz virtual para cada una de las VLANs a enrutar, con el comando **interface vlan *[id vlan]*** ejecutado desde el modo de configuración global y donde *id vlan* hace referencia al número de esta.

Paso 3: Configurar una IP en cada interfaz con el comando **ip address *[ip] [máscara].*** Esta debe formar parte del rango de la subred de la VLAN a la que pertenece.

Paso 4: Habilitar cada interfaz con el comando **no shutdown**.

Aplicado en el switch de capa 3…

```
SwitchL3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SwitchL3(config)#ip routing
SwitchL3(config)#interface vlan 10
SwitchL3(config-if)#ip address 192.168.10.1 255.255.255.0
SwitchL3(config-if)#no shutdown
SwitchL3(config-if)#exit
SwitchL3(config)#interface vlan 20
SwitchL3(config-if)#ip address 192.168.20.1 255.255.255.0
SwitchL3(config-if)#no shutdown
SwitchL3(config-if)#exit
SwitchL3(config)#interface vlan 30
SwitchL3(config-if)#ip address 192.168.30.1 255.255.255.0
SwitchL3(config-if)#no shutdown
SwitchL3(config-if)#exit
```

El contenido de CCNA no profundiza en conceptos de switchs de capa 3, sin embargo, sí resulta necesario conocer esta configuración y la función de las interfaces virtuales, que no es otra que enrutar tráfico entre las diferentes VLANs.

RUTAS ESTÁTICAS

Las redes directamente conectadas son agregadas automáticamente a la tabla de rutas, sin embargo, no sucede lo mismo con aquellas remotas, las cuales pueden ser aprendidas en relación con dos métodos, manualmente mediante rutas estáticas o gracias a la aplicación de protocolos de enrutamiento.

En redes grandes con multitud de routers y subredes resulta prácticamente imposible realizar la configuración de manera manual, además de poco práctico y nada escalable. Este método suele ser más habitual y recomendable sobre entornos pequeños o rutas específicas, como puede ser el caso de aquellas por defecto, analizadas en párrafos posteriores.

Una ruta estática es configurada mediante el comando **ip route *[red destino] [máscara de red de destino] [interfaz de salida o IP de siguiente salto]*** desde el modo de configuración global, donde *red de destino* indica la red remota, con su máscara, e *interfaz de salida o IP de siguiente salto* hace referencia a:

- Interfaz de salida define la interfaz local del router por la cual debe ser reenviado el paquete para llegar a la red remota.
- IP de siguiente salto indica la dirección IP del router al cual debe ser reenviado el paquete para que llegue a la red de destino.

Ejemplo:

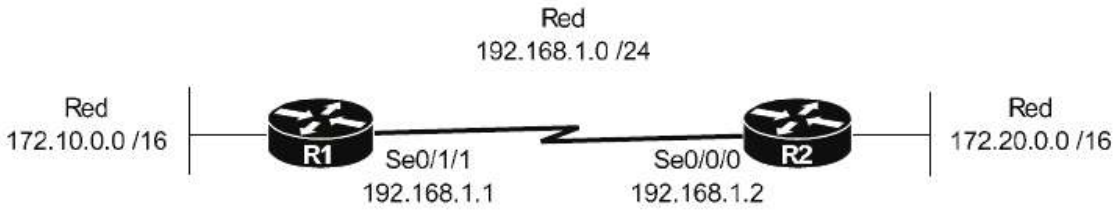


Fig. 5-15 Redes remotas. Configuración de rutas estáticas.

R1 conecta directamente con dos redes, la 172.10.0.0/16 y la 192.168.1.0/24. Sin embargo, existe otra remota a la que aún no tiene acceso porque no ha sido configurada, la 172.20.0.0/16. Para agregarla como ruta estática y con ello obtener

conectividad se debe aplicar el comando descrito anteriormente en cualquiera de sus dos formas, siendo estas:

```
R1(config)#ip route 172.20.0.0 255.255.0.0 Se0/1/1
ó
R1(config)#ip route 172.20.0.0 255.255.0.0 192.168.1.2
```

El primer método define que los paquetes con destino a la red 172.20.0.0/16 deben ser reenviados a través de la interfaz Se0/1/1, lo que dará como resultado que sean recibidos por R2. Mientras, el segundo comando indica que los paquetes con destino a la misma red deben ser reenviados a la dirección IP 192.168.1.2, que corresponde a R2, obteniendo el mismo resultado.

R2 también dispone de dos redes directamente conectadas, la 172.20.0.0/16 y la 192.168.1.0/24, y de otra remota a la que aún no tiene acceso, la 172.10.0.0/16. Para configurarla:

```
R2(config)#ip route 172.10.0.0 255.255.0.0 Se0/0/0
ó
R2(config)#ip route 172.10.0.0 255.255.0.0 192.168.1.1
```

Gracias a la configuración agregada manualmente, ambos routers disponen de conectividad total en la red.

De los dos métodos disponibles lo más recomendable es hacer uso de la interfaz local de salida. ¿Por qué? Cuando se indica la IP de siguiente salto, el router realiza una búsqueda recursiva en su tabla de rutas para averiguar la interfaz por la cual debe ser reenviado el paquete. Por ejemplo, en R1 se indicó como IP de siguiente salto la dirección 192.168.1.2. Bien, cuando tenga que enviar datos hacia la red 172.20.0.0/16, primero comprobará dicha IP, calculará la red a la que pertenece (192.168.1.0/24) y buscará en su tabla de rutas la interfaz que debe utilizar para llegar hacia la misma. Por último, reenviará el paquete a través de ella, siendo recibido por R2.

Las rutas estáticas son representadas en la tabla de rutas con el código *"S"* y permanecen configuradas hasta que el enlace caiga, en cuyo caso serían eliminadas automáticamente. Si fuera necesario mantenerlas activas aún sin enlace operativo se debe incluir el parámetro "*permanent*" al final de la sentencia *ip route*…

Volviendo a la topología de ejemplo para este capítulo…

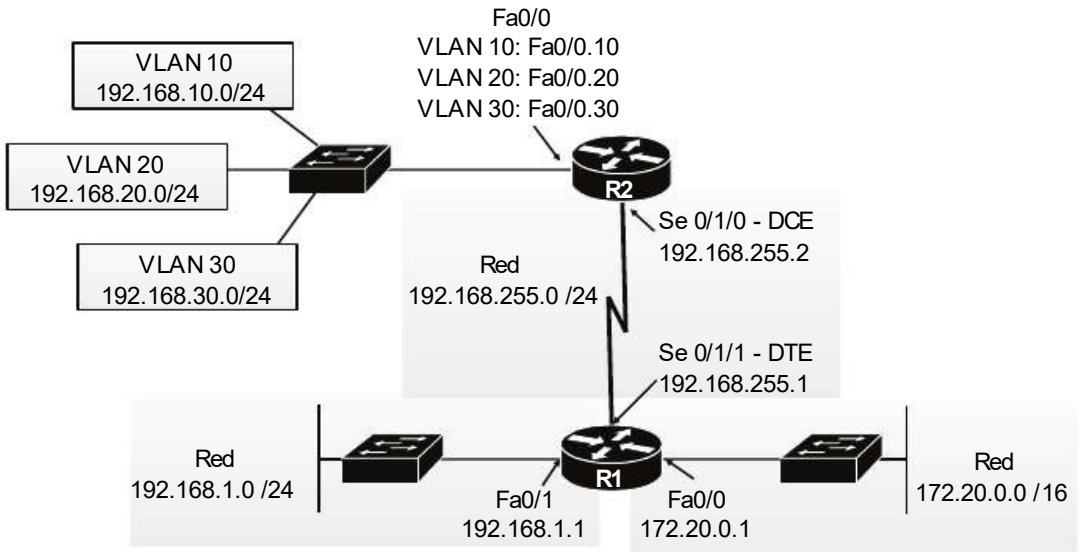


Fig. 5-16 Topología de red corporativa.

Para ambos routers existen redes remotas que aún no han sido configuradas, por lo tanto, no disponen de acceso a ellas. Una buena práctica antes de realizar cualquier configuración consiste en planificarla y documentarla, en cuyo caso se deberá identificar las redes de destino existentes para cada dispositivo y asociarlas con la interfaz de salida necesaria para llegar a cada una de ellas. Una vez localizadas, proceder a su configuración.

Router	Red remota	Interfaz para conectar con red remota
R1	192.168.10.0/24	Se0/1/1
R1	192.168.20.0/24	Se0/1/1
R1	192.168.30.0/24	Se0/1/1
R2	172.20.0.0/16	Se0/1/0
R2	192.168.1.0/24	Se0/1/0

Configuración en R1

```
R1#conf t
R1(config)#ip route 192.168.10.0 255.255.255.0 Se0/1/1
R1(config)#ip route 192.168.20.0 255.255.255.0 Se0/1/1
R1(config)#ip route 192.168.30.0 255.255.255.0 Se0/1/1


R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
```

```
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    172.20.0.0/16 is directly connected, FastEthernet0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/1
S    192.168.10.0/24 is directly connected, Serial0/1/1
S    192.168.20.0/24 is directly connected, Serial0/1/1
S    192.168.30.0/24 is directly connected, Serial0/1/1
C    192.168.255.0/24 is directly connected, Serial0/1/1
```

Configuración en R2

```
R2#conf t
R2(config)#ip route 172.20.0.0 255.255.0.0 Se0/1/0
R2(config)#ip route 192.168.1.0 255.255.255.0 Se0/1/0


R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set
S    172.20.0.0/16 is directly connected, Serial0/1/0
S    192.168.1.0/24 is directly connected, Serial0/1/0
C    192.168.10.0/24 is directly connected, FastEthernet0/0.10
C    192.168.20.0/24 is directly connected, FastEthernet0/0.20
C    192.168.30.0/24 is directly connected, FastEthernet0/0.30
C    192.168.255.0/24 is directly connected, Serial0/1/0
```

Tras los cambios realizados, tanto R1 como R2 disponen de rutas hacia todas las redes existentes en la topología, por lo que ambos disponen de conectividad total.

Rutas estáticas por defecto

En párrafos anteriores se ha descrito que la manera de proceder de un router ante un paquete cuya red de destino sea desconocida, es descartándolo. Este hecho puede evitarse configurando una ruta por defecto, que por definición es aquella que se utilizará para enviar los paquetes cuyo destino no coincida con ninguno de los existentes en la tabla de rutas.

Para agregarla es necesario ejecutar el comando **ip route 0.0.0.0 0.0.0.0** ***[interfaz salida]*** desde el modo de configuración global. Una vez aplicado, la nueva entrada será representada con el código *"S*"*. La "S" la identifica como una ruta estática,

mientras que el "*" la marca como principal. Ello es debido a que pueden ser configuradas varias rutas por defecto, sin embargo, solo se hará uso de una, la principal.

Imagina que el router R2 de la topología conecta con Internet a través de su interfaz Se0/1/1, siendo necesario configurar una ruta por defecto para que cualquier paquete que no coincida con ninguna de las entradas existentes sea reenviado a través de la misma.

```
R2(config)#ip route 0.0.0.0 0.0.0.0 Se0/1/1
R2(config)#exit

R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 0.0.0.0 to network 0.0.0.0

C    10.0.0.0/8 is directly connected, Serial0/1/1
S    172.20.0.0/16 is directly connected, Serial0/1/0
S    192.168.1.0/24 is directly connected, Serial0/1/0
C    192.168.10.0/24 is directly connected, FastEthernet0/0.10
C    192.168.20.0/24 is directly connected, FastEthernet0/0.20
C    192.168.30.0/24 is directly connected, FastEthernet0/0.30
C    192.168.255.0/24 is directly connected, Serial0/1/0
S*   0.0.0.0/0 is directly connected, Serial0/1/1
```

La configuración de rutas estáticas se lleva a cabo de manera relativamente sencilla, pero a su vez resulta un método muy poco escalable. La opción más recomendable a implementar, sobre todo en redes de gran tamaño, son los protocolos de enrutamiento dinámico, los cuales se encargan de aprender y completar la tabla de rutas de manera automática. Estos serán analizados en profundidad en el capítulo 6 *"Protocolos de enrutamiento"*.

PROTOCOLO DHCP: ANÁLISIS Y CONFIGURACIÓN

Un aspecto importante a tener en cuenta en cualquier red es la manera en la que los hosts obtienen la configuración para acceder a ella, la cual debe estar formada por una dirección IP, máscara de red, puerta de enlace y servidores DNS, estos últimos opcionales pero muy recomendables. Para llevar a cabo esta tarea se puede optar por dos métodos, bien realizar la configuración de manera manual en cada

dispositivo, o bien hacer uso del protocolo DHCP *(Dynamic Host Configuration Protocol)*, el cual se encarga de proporcionar dichos datos de conexión de manera automática a los miembros la red. La primera de las opciones, además de poco escalable, resulta muy difícil de administrar. Mientras, DHCP se convierte en la opción ideal debido su facilidad de administración y servicio centralizado. En la gran mayoría de redes, incluso aquellas domésticas, se opta por su implementación.

El protocolo DHCP basa su modo de operar en el intercambio de una serie de mensajes entre cliente y servidor con el fin de que el primero obtenga una configuración de red válida de manera automática. Hay que tener en cuenta que en un principio el cliente no dispone de dirección IP, ni máscara, ni puerta de enlace, ni si quiera conoce ningún dato sobre el servidor al cual va a solicitar la configuración. Es por ello que se establece un procedimiento bien definido para que la comunicación entre ambos pueda llevarse a cabo, consistente en:

- *Paso 1:* El cliente aún no dispone de configuración alguna, por lo que inicia una búsqueda de servidores DHCP. Para ello, envía un mensaje a la red denominado DHCPDISCOVER, con dirección de destino broadcast, tanto en capa 2 (FFFF.FFFF.FFFF.FFFF) como en capa 3 (255.255.255.255). Como origen, en capa 2 hace uso de su propia MAC, mientras que en capa 3 incluye la dirección 0.0.0.0, que corresponde a una IP reservada para este propósito.

- *Paso 2:* Cuando algún servidor DHCP recibe el DHCPDISCOVER, busca una dirección IP disponible para el cliente, crea una entrada ARP que la asocia con la MAC del host solicitante y acto seguido oferta los datos de conexión mediante un mensaje DHCPOFFER. Este es enviado en capa 2 agregando la MAC del cliente como destino y la MAC del servidor como origen.

- *Paso 3:* Una vez recibido, el cliente responde con un mensaje DHCPREQUEST, el cual tiene dos propósitos, aceptar los datos de conexión que se ofertaron en el paso 2 o solicitar la renovación de estos.

- *Paso 4:* Por último, el servidor responde con un DHCPACK, el cual incluye la configuración definitiva que el cliente aplicará para obtener acceso a la red.

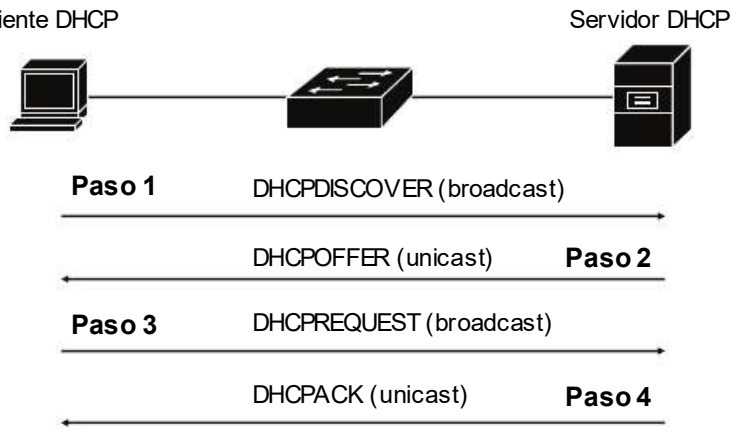


Fig. 5-17 Protocolo DHCP. Comunicación entre cliente y servidor.

Servidores DHCP ubicados en subredes remotas

El procedimiento recién analizado concluye con éxito cuando el servidor DHCP y el cliente pertenecen a la misma subred (sin ningún router de por medio). Sin embargo, cuando se encuentran en diferentes segmentos y por lo tanto la comunicación tiene que atravesar algún router resulta necesaria la configuración de estos para que el proceso pueda llevarse a cabo. ¿Por qué? Como se ha analizado, algunos de los mensajes utilizados por el protocolo son enviados mediante broadcast, siendo este el inconveniente, los routers no reenvían broadcast a través de sus interfaces.

En entornos corporativos lo más habitual consiste en plantear el diseño de tal manera que los servidores (incluido el DHCP) formen parte de una subred exclusiva, lo que significa que los clientes pertenecerán a otras diferentes y por lo tanto será necesaria la configuración de los routers intermediarios para que la comunicación sea posible.

Para ello bastará con ejecutar el comando **ip helper-address *[dir IP servidor DHCP]***, desde el modo de configuración de la interfaz por la cual se recibe el broadcast, indicando la dirección IP del servidor. Al aplicarlo, el router realiza las siguientes acciones:

- Acepta los mensajes DHCP con destino 255.255.255.255 en la interfaz donde fue configurado y modifica sus direcciones incluidas en capa 3. Como origen agrega la configurada en su propia interfaz y como destino la del servidor (definida en el comando).
- Reenvía el mensaje DHCP al servidor de manera unicast.
- Cuando el servidor responde al router, este lo reenvía a la subred mediante un broadcast (255.255.255.255) para que sea recibido por el cliente.

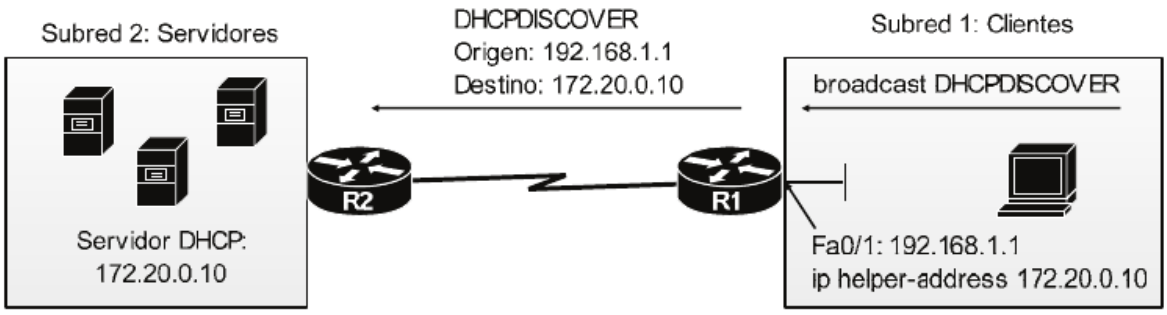


Fig. 5-18 DHCP Relay.

En el ejemplo, un cliente ubicado en la subred 1 inicia la búsqueda de algún servidor DHCP mediante un mensaje DHCPDISCOVER, enviado a la dirección broadcast 255.255.255.255. El router recibe el paquete a través de su interfaz Fa0/1, que previamente ha sido configurada con el comando *ip helper-address 172.20.0.10,* por lo tanto, R1 aceptará el broadcast, modificará la IP de origen y destino y acto seguido lo reenviará como mensaje unicast. Como origen aplica la dirección de su propia interfaz, 192.168.1.1, y como destino, aquella definida durante la configuración, en este caso 172.20.0.10.

Creado el nuevo paquete, comprueba su tabla de rutas y reenvía el mensaje a través de la interfaz necesaria para llegar a la red del servidor. Cuando este responda, lo hará directamente al router (192.168.1.1), el cual a su vez lo reenviará mediante broadcast a la subred del cliente.

La configuración necesaria en R1…

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int fa0/1
R1(config-if)#ip helper-address 172.20.0.10
```

¿Por qué en R2 no resulta necesaria la configuración del comando si también debe reenviar paquetes DHCP? No es necesaria porque dichos paquetes son enviados desde R1 como mensajes unicast, siendo estos aceptados y reenviados por los routers sin necesidad de configuración previa.

El proceso que ejecuta el router de modificar un paquete broadcast en otro unicast, y viceversa, es denominado *DHCP relay.*

CONFIGURACIÓN DHCP EN ROUTERS CISCO

Normalmente, las compañías instalan servidores DHCP dedicados que operan con sistemas operativos como Windows Server 2016, sin embargo, los routers Cisco también pueden ser configurados para ofrecer este servicio. Sea cual sea la opción elegida, definir los siguientes parámetros resulta requisito imprescindible:

- *ID de subred y máscara:* Gracias a ello el servidor conoce el rango de direcciones IP disponibles que puede ofrecer a los clientes. Por ejemplo, con el ID 192.168.10.0 y máscara 255.255.255.0 ofertará las direcciones incluidas en el rango 192.168.10.1 - 192.168.10.254.

- *Direcciones IP excluidas o reservadas:* Son direcciones pertenecientes al rango pero dedicadas a determinados dispositivos que requieren siempre la misma IP. Estas pueden ser divididas en dos grupos, excluidas o reservadas. Las excluidas son aquellas que el servidor DHCP no proporcionará a ningún cliente, debiendo ser configuradas manualmente, normalmente en routers, switchs, puntos de acceso, servidores, etc. Mientras, en las reservadas se crea una asociación en el servidor de IP-MAC del cliente, para que a este último siempre se le asigne la misma dirección.

- *Puerta de enlace predeterminada:* Es la dirección a la cual los clientes deberán enviar los paquetes con destino a redes remotas con el fin de que puedan ser enrutados. El dispositivo encargado de ello es el router, por lo que la puerta de enlace corresponde a la IP de la interfaz del router que conecta con la subred de los clientes.

- *IP de servidores DNS:* Como su nombre indica, define los servidores DNS que utilizarán los clientes.

Durante la configuración del servidor también pueden ser definidos diferentes parámetros, siendo el más destacado la vigencia de IPs ya asignadas, también denominado tiempo de arrendamiento. Esta función se encarga de mantener una reserva IP-Host de direcciones ya asignadas durante un tiempo establecido. Es decir, si un dispositivo ha sido desconectado de la red (o simplemente apagado), y vuelve a ser conectado, el servidor le ofertará la misma IP que obtuvo con anterioridad siempre y cuando el tiempo de arrendamiento no haya concluido. De lo contrario se le asignará cualquiera disponible dentro del rango.

Para configurar un router Cisco como servidor DHCP se deben llevar a cabo las siguientes acciones:

- *Paso 1:* Excluir las direcciones IP que no deben ser asignadas automáticamente, con el comando **ip dhcp excluded-address *[IP de inicio] [IP final]*** desde el modo de configuración global. *[IP de inicio] [IP final]* definen el rango deseado.

- *Paso 2:* Crear un pool DHCP a través de la sentencia **ip dhcp pool *[nombre]*** desde el modo de configuración global. El nombre tan solo es un identificativo a nivel local. Una vez creado se accede a un nuevo submodo de configuración desde el cual se ejecutarán todos los pasos restantes.

- *Paso 3:* Definir el ID de red y máscara, gracias a los cuales el router calculará el rango de direcciones disponibles a ofertar de manera automática a los clientes que lo soliciten, con el comando **network *[id de red] [máscara]***.

- *Paso 4:* Asignar la(s) puerta(s) de enlace necesaria, con el comando **default-router *[IP 1] [IP 2]…***

- *Paso 5:* Establecer los DNS, ejecutando la sentencia **dns-server [IP 1] [IP 2]…**

- *Paso 6 (opcional):* Configurar el tiempo de arrendamiento de direcciones IP asignadas, con el comando **lease *[días] [horas] [minutos]***.

- *Paso 7 (opcional):* Definir un nombre de dominio DNS, haciendo uso del comando **domain-name *[nombre]***.

Volviendo a la topología de ejemplo presente en este capítulo, configurar R1 como servidor DHCP para las redes de dispositivos finales a las que pertenece, siendo estas la 192.168.1.0/24 y la 172.20.0.0/16. Además, se han agregado dos hosts, uno en cada red, para comprobar que el servidor opera correctamente.

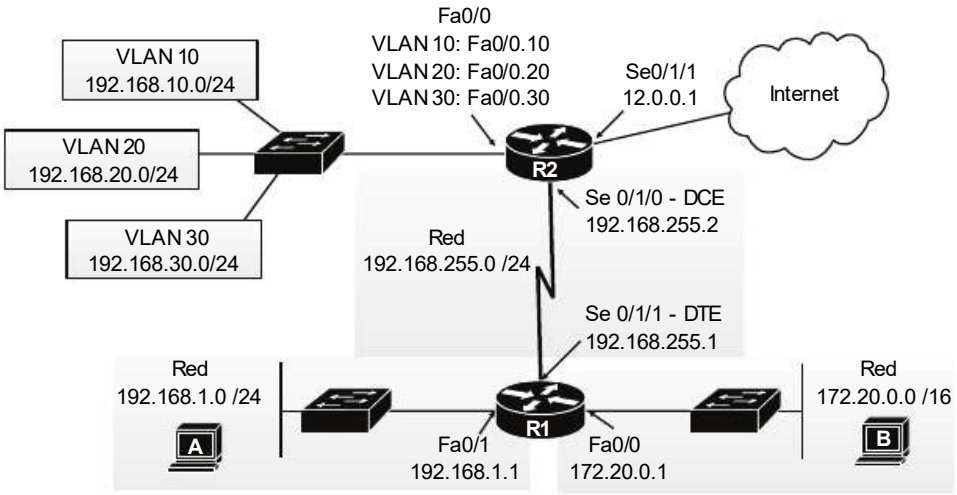


Fig. 5-19 Topología de red corporativa.

El primer paso a llevar a cabo consiste en planificar y documentar la configuración necesaria:

	Pool red 192.168.1.0/24	Pool red 172.20.0.0/16
Direcciones excluidas	192.168.1.1 – 192.168.1.10	172.20.0.1 – 172.20.0.10
Nombre del pool	SubredIzquierda	SubredDerecha
ID de Red / máscara	192.168.1.0 / 255.255.255.0	172.20.0.0 / 255.255.0.0
Puerta de enlace	192.168.1.1	172.20.0.1
Servidores DNS	192.168.10.254	192.168.10.254
Tiempo de arrendamiento	1 día	1 día

El servidor DNS se encuentra ubicado en la VLAN 10 conectado a R2. Se aplicará el mismo para todos los clientes.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip dhcp excluded-address 192.168.1.1 192.168.1.10
R1(config)#ip dhcp excluded-address 172.20.0.1 172.20.0.10

R1(config)#ip dhcp pool SubredIzquierda
R1(dhcp-config)#network 192.168.1.0 255.255.255.0
R1(dhcp-config)#default-router 192.168.1.1
R1(dhcp-config)#dns-server 192.168.10.254
R1(dhcp-config)#lease 1 0 0

R1(config)#ip dhcp pool SubredDerecha
R1(dhcp-config)#network 172.20.0.0 255.255.0.0
R1(dhcp-config)#default-router 172.20.0.1
R1(dhcp-config)#dns-server 192.168.10.254
R1(dhcp-config)#lease 1 0 0
```

Con los cambios realizados, los hosts A y B deberán obtener sus direcciones IP automáticamente a través R1. Para verificarlo bastará con ejecutar un *ipconfig /all* en ambos PCs.

Host A…

```
PCA>ipconfig /all

FastEthernet0 Connection:(default port)
Physical Address.....:  0040.0B90.8063
Link-local IPv6 Address.....:  FE80::240:BFF:FE90:8063
IP Address.....:  192.168.1.11
Subnet Mask.....:  255.255.255.0
Default Gateway.....:  192.168.1.1
DNS Servers.....:  192.168.10.254
DHCP Servers.....:  192.168.1.1
```

Host B…

```
PCB>ipconfig /all

FastEthernet0 Connection:(default port)
Physical Address.....:  0060.3E68.DCAE
Link-local IPv6 Address.....:  FE80::260:3EFF:FE68:DCAE
IP Address.....:  172.20.0.11
Subnet Mask.....:  255.255.0.0
Default Gateway.....:  172.20.0.1
DNS Servers.....:  192.168.10.254
DHCP Servers.....:  172.20.0.1
```

Ambos han obtenido correctamente la configuración por DHCP desde R1 y ahora disponen de conectividad. Se les ha asignado una dirección correspondiente a su subred y terminada en .11, coincidiendo con la primera disponible después del rango de excepciones establecido.

Para verificar tanto la configuración como operaciones de DHCP, IOS dispone de los siguientes comandos:

- *Show ip dhcp binding:* Muestra un listado de las direcciones IP que han sido asignadas, asociadas a la dirección MAC del cliente que hace uso de cada una de ellas.

- *Show ip dhcp pool [nombre de pool]:* Muestra información sobre un determinado pool, incluyendo datos como el id de red, máscara, número total de direcciones, IPs asignadas, etc.

- *Show ip dhcp server statistics:* Muestra datos estadísticos de operaciones ejecutadas por el servicio DHCP.

PRUEBAS DE CONECTIVIDAD

Una vez los hosts han obtenido sus datos de conexión deberían acceder a la red sin problemas. Aun así, para comprobarlo se puede hacer uso de dos comandos bastante útiles, *ping* y *traceroute*.

Ping

Ping es considerada la herramienta por excelencia necesaria para realizar testeos de conectividad de extremo a extremo en capa 3. Su modo de operar se basa en el protocolo ICMP (*Internet Control Message Protocol),* donde el emisor envía un paquete "*IP ICMP echo request*" al destino, y este, al recibirlo, responde con

un "*IP ICMP echo reply*". Si la comunicación concluye con éxito significa que existe conectividad entre ambos y por lo tanto entre las diferentes subredes a las que pertenecen.

El comando admite dos variantes: **ping *[nombre],*** o **ping *[dir IP].*** En la primera, el host debe realizar una consulta DNS para obtener la IP del nombre indicado, mientras que en la segunda, el paquete ICMP es creado directamente con la dirección introducida.

En el ejemplo recién analizado los dos hosts disponen de configuración de red y la tabla de rutas de R1 contiene la información necesaria para acceder a ambas subredes (rutas directamente conectadas), por lo tanto, un ping entre ambos debería concluir con éxito si todos los datos obtenidos por DHCP fueran correctos.

```
PCA>ping 172.20.0.11

Pinging 172.20.0.11 with 32 bytes of data:

Reply from 172.20.0.11: bytes=32 time=0ms TTL=127
Reply from 172.20.0.11: bytes=32 time=0ms TTL=127
Reply from 172.20.0.11: bytes=32 time=0ms TTL=127
Reply from 172.20.0.11: bytes=32 time=0ms TTL=127
```

Desde el PC con IP 192.168.1.11 se ha ejecutado un ping a la dirección 172.20.0.11 y se ha obtenido respuesta, por lo tanto existe conectividad entre ambos. ¿Qué proceso se ha llevado a cabo?

- El host A ha creado un paquete *ICMP echo Request* con dirección de destino 172.20.0.11.
- Acto seguido calcula que dicha IP pertenece a una red que no coincide con la suya propia, por lo tanto lo envía a su puerta de enlace, que en este caso es la 192.168.1.1.
- El paquete es recibido por el router R1 a través de su interfaz Fa0/1.
- Este lee la IP de destino y busca la red en su tabla de rutas. La localiza asociada a su interfaz Fa0/0, por lo tanto, reenvía el paquete a través de la misma.
- El host B lo recibe y responde con un mensaje *ICMP echo Reply,* ejecutando el mismo proceso.

Durante la comunicación, la IP de origen corresponde a la del dispositivo que inicia el ping. En este aspecto existe una peculiaridad en los routers, y es que estos pueden disponer de numerosas interfaces en estado "*up/up*", cada una de ellas con una IP diferente. Entonces, cuando el comando es ejecutado desde IOS, ¿cuál de ellas se utiliza como origen en el paquete ICMP? En este caso, el router examina la

dirección introducida, busca la red de destino en la tabla de rutas y envía el paquete a través de la interfaz adecuada para llegar a ella, utilizando como IP de origen aquella configurada en dicha interfaz.

IOS también dispone de la opción de ping extendido, la cual permite introducir de manera manual diferentes opciones del protocolo, como la dirección de origen, pudiendo ser esta la configurada en cualquier otra interfaz. Para ello bastará con ejecutar el comando *ping*, sin parámetros, y completar las opciones deseadas.

```
R1#ping

Protocol [ip]:
Target IP address: 172.20.0.11
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 192.168.1.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.20.0.11, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/1/5 ms
```

Traceroute

Otra utilidad de gran ayuda para la resolución de incidencias y pruebas de conectividad en capa 3 es la herramienta traceroute. Mientras que ping realiza un testeo de extremo a extremo, traceroute se encarga de mostrar un listado con los saltos que da un paquete desde el origen hasta el destino.

Su modo de operar se basa en el uso del campo TTL (*Time To Live),* incluido en capa 3 y formado por un valor numérico cuya misión consiste en controlar el número máximo de saltos que puede dar un paquete IP, con el objetivo de evitar loops de enrutamiento. Para ello, cada vez que el paquete atraviesa un router, al valor de TTL se le resta 1, de tal manera que cuando llegue a 0 se producen dos acciones, primero es descartado y segundo se informa al origen de la comunicación que se ha excedido el número de saltos permitidos, mediante un mensaje *ICMP TTL Exceeded.*

Traceroute se basa en la explotación de este mensaje, llevando a cabo las siguientes acciones:

- *Paso 1:* El origen crea un paquete con valor TTL=1 y lo envía a la dirección IP de destino.

- *Paso 2:* El primer router en recibirlo resta 1 al TTL, por lo tanto su nuevo valor es igual a 0, hecho que causa que el paquete sea descartado, informando al origen con un *ICMP TTL Exceeded.*

- *Paso 3:* Este recibe el mensaje y muestra en pantalla el primer salto realizado.

- *Paso 4:* Acto seguido crea otro paquete hacia el mismo destino pero con el campo TTL=2, de tal manera que el primer router resta uno (TTL=1) y lo reenvía. El segundo router hará lo mismo, restará 1 (TTL=0) e informará al origen con un mensaje *ICMP TTL Exceeded.* Este mostrará el segundo salto en pantalla.

El proceso se repite sucesivamente hasta que el paquete llegue a su destino, mostrando todos los saltos intermedios.

Cuando *traceroute* es ejecutado en Windows el comando necesario es **tracert *[ip de destino]***, mientras que en sistemas Linux e IOS la sentencia es **traceroute *[ip destino]***.

TEST CAPÍTULO 5: CONFIGURACIÓN INICIAL DE ROUTERS CISCO

1.- ¿Cuál es la función principal de los routers?

A. Aumentar el ancho de banda de la red.
B. Proveer de seguridad a los dispositivos.
C. Establecer la conexión con Internet.
D. Permitir la comunicación entre diferentes redes.

2.- ¿Cuál es la función de los módulos de expansión en routers Cisco?

A. Dotar de mayor capacidad de memoria al router.
B. Agregar nuevas funciones de hardware, como un módem o puertos Ethernet.
C. Agregar nuevas funciones de software al dispositivo, como un firewall, IPS, IDS, etc.
D. Monitorizar el tráfico que atraviesa el dispositivo.

3.- ¿Qué información se podrá obtener tras ejecutar un "show ip interfaces brief" en un router Cisco? (Seleccionar tres respuestas)

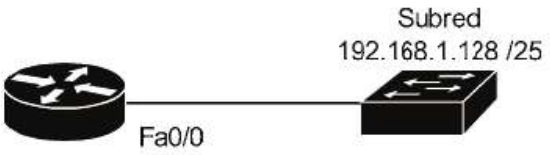
A. Interfaces instaladas en el dispositivo.
B. Dirección IP de cada interfaz.
C. Estado de las interfaces.
D. Paquetes recibidos en cada interfaz.
E. Paquetes enviados en cada interfaz.
F. Protocolos de enrutamiento habilitados.

4.- ¿Cuáles de las siguientes causas son las más probables cuando una interfaz se encuentra en estado "down/down"? (Seleccionar dos respuestas)

A. Velocidad del enlace diferente en ambos extremos.
B. Diferentes protocolos aplicados en los dos routers que forman parte del enlace.
C. Dispositivo del otro extremo apagado.
D. Interfaz apagada con el comando "shutdown".
E. Cable de red desconectado.

5.- La interfaz Fa0/0 del router de la siguiente topología debe ser configurada para que actúe como puerta de enlace para los dispositivos pertenecientes a la subred con

la que conecta. ¿Cuál de los siguientes comandos se debe aplicar en ella para lograr el objetivo?

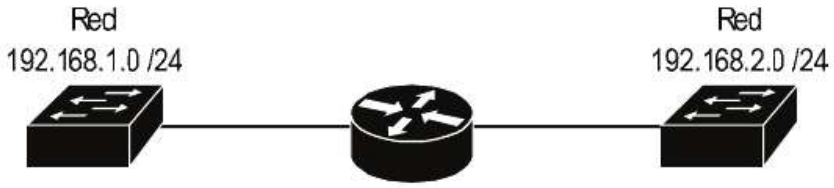


A. ip address 192.168.1.1 255.255.255.128
B. ip address 192.168.1.1 mask 255.255.255.0
C. ip address 192.168.1.254 255.255.255.0
D. ip address 192.168.1.135 255.255.255.128

6.- En un enlace serial entre dos routers, R1 actúa como DTE y R2 como DCE. ¿En cuál de ellos se debe configurar el comando "clock rate…"?

A. R1.
B. R2.
C. Ambos.
D. Ninguno.

7.- Dada la siguiente topología…



PC1, con IP 192.168.2.100 desea enviar un paquete a PC2, con IP es 192.168.2.150. ¿Cómo procederá?

A. Enviará el paquete directamente a PC2.
B. Lo enviará a su puerta de enlace predeterminada.
C. Realizará una consulta DNS antes de enviar el paquete.
D. Descartará la comunicación.

8.- ¿En qué campo de un paquete IP se basa un router para tomar la decisión de reenvío?

A. En la tabla de rutas.

B. En los protocolos de enrutamiento.
C. En la dirección MAC de destino.
D. En la dirección IP de destino.

9.- La interfaz de un router, perteneciente a la red 192.168.1.0/24, recibe un paquete con dirección IP de destino 192.168.1.255. ¿Cómo procederá?

A. Lo reenviará a través de todas sus interfaces.
B. Comprobará la tabla de rutas y lo reenviará a través de la interfaz adecuada.
C. Cambiará la dirección de capa 2, comprobará la tabla de rutas y lo reenviará a través de la interfaz adecuada.
D. Lo descartará.

10.- ¿Qué tipo de procesamiento interno aplican los routers Cisco actuales por defecto?

A. Fast Switching.
B. Process Switching.
C. Cisco Express Forwarding.
D. Cisco Express Switching.

11.- ¿Qué rutas son agregadas de manera automática a la tabla de enrutamiento? (Seleccionar dos respuestas)

A. Las rutas estáticas.
B. Las rutas directamente conectadas.
C. Las rutas por defecto.
D. Las rutas aprendidas por protocolos de enrutamiento.

12.- ¿Cuál de los siguientes requisitos resulta necesario para que un router pueda comunicar diferentes VLANs a través de una sola interfaz física?

A. Que el switch con el que conecta sea de capa 3.
B. Que se cree una subinterfaz por cada VLAN a enrutar.
C. Que el tráfico de las diferentes VLANs sea etiquetado con la VLAN nativa.
D. Que la velocidad de transferencia del enlace sea como mínimo de 100 Mbps.

13.- ¿Qué acción realiza el comando “ip route 192.168.1.0 255.255.255.0 Se0/1”?

A. Agrega una ruta estática a la tabla de rutas.
B. Agrega una ruta directamente conectada a la tabla de rutas.

C. Agrega la interfaz Se0/1 a la red 192.168.1.0 /24.
D. Agrega una ruta estática por defecto a la tabla de rutas.

14.- ¿Qué acción realiza el comando "ip route 0.0.0.0 0.0.0.0 Se0/1"?

A. Agrega una ruta estática a la tabla de rutas.
B. Agrega una ruta directamente conectada a la tabla de rutas.
C. Establece una conexión con Internet a través de la interfaz Se0/1.
D. Agrega una ruta estática por defecto a la tabla de rutas.

15.- ¿Cuáles de las siguientes direcciones de destino son incluidas en un mensaje DHCPDISCOVER? (Seleccionar dos respuestas).

A. Broadcast 255.255.255.255.
B. Multicast 224.0.0.10.
C. Unicast a la dirección IP del servidor DHCP.
D. Broadcast FFFF.FFFF.FFFF.
E. Unicast a la dirección IP de la puerta de enlace.

16.- Dada la siguiente configuración, ¿qué rango de direcciones IP serán suministradas mediante DHCP a los clientes que la soliciten?

```
R1(config)#ip dhcp excluded-address 192.168.1.1 192.168.1.254

R1(config)#ip dhcp pool DHCP
R1(dhcp-config)#network 192.168.1.0 255.255.255.0
R1(dhcp-config)#default-router 192.168.1.1
R1(dhcp-config)#lease 1 0 0
```

A. El rango 192.168.1.1 – 192.168.1.255
B. El rango 192.168.1.0 – 192.168.1.255
C. El rango 192.168.1.1 – 192.168.1.254
D. Ninguna dirección IP.

17.- Un PC ha obtenido por DHCP sus datos de conexión, entre los que se encuentra el servidor DNS, con IP 172.30.1.1. ¿Qué comando ha sido ejecutado en el router para que el cliente pueda obtener dicha información?

A. "dns-server 172.30.1.1 255.255.255.0" desde el modo de configuración global.
B. "dns-server 172.30.1.1 255.255.255.0" desde el modo de configuración de la interfaz que conecta con el cliente.

C. “dns-server 172.30.1.1 255.255.255.0” desde el modo de configuración del pool DHCP.
D. Ninguna de las anteriores.

18.- En un switch de capa 3 han sido configuradas 3 interfaces virtuales para llevar a cabo el enrutamiento entre 3 VLANs, sin embargo, tras ejecutar un “show ip route” no aparece ninguna ruta y la comunicación entre ellas no se está realizando. ¿A qué puede ser debido?

A. En un switch de capa 3 no es posible ejecutar el comando “show ip route”.
B. La configuración se debe llevar a cabo en interfaces físicas, no virtuales.
C. No se ha ejecutado el comando “ip routing”.
D. Resulta necesario configurar rutas estáticas para cada subred.

19.- De los siguientes, ¿qué comando ejecuta una prueba de conectividad de extremo a extremo?

A. Ping.
B. ICMP.
C. Tracert.
D. TTL.

PROTOCOLOS DE ENRUTAMIENTO 6

CONCEPTOS BÁSICOS

Analizadas tanto las rutas directamente conectadas como aquellas estáticas, tan solo resta abarcar el estudio del tercer método posible de aprendizaje de rutas, los protocolos de enrutamiento. Estos se basan en el intercambio de información entre los diferentes dispositivos de capa 3 que componen la red, con el fin de completar y actualizar de manera dinámica y automática la tabla de rutas de cada uno de ellos, lo que sin duda lo convierte en el método más escalable y recomendable a aplicar sobre entornos corporativos.

Existen diferentes opciones para lograr dicho propósito, como RIP, OSPF o EIGRP, cada uno de ellos con características y modo de operar propio pero coincidiendo en el objetivo final, consistente en llevar a cabo las siguientes funciones:

- Aprender y agregar de manera automática información en la tabla de rutas.
- Si se produjera algún cambio en la topología (como una nueva subred o la eliminación de otra), informar al resto de routers para que apliquen los cambios oportunos.
- Si se elimina una ruta hacia una subred y existe otra disponible, hacer uso de ella.
- Evitar bucles de capa 3.
- Si existiera más de una ruta hacia la misma subred, instalar la mejor de ellas.

Los protocolos de enrutamiento pueden resultar complejos, pero definen su modo de operar en una lógica bastante sencilla basada en:

- *Paso 1:* Cada router agrega en la tabla de rutas sus redes directamente conectadas.

- *Paso 2:* Cada router comparte con sus vecinos su información de enrutamiento.

- *Paso 3:* Cuando se aprende una nueva ruta, se agrega a la tabla y se establece como próximo salto (por dónde debe ser enviado el paquete) la interfaz por la cual fue recibida.

Este mismo procedimiento analizado de manera práctica…

Paso 1: R1 agrega las redes 192.168.1.0/24 y 192.168.254.8/30 a su tabla de rutas y las asocia a sus interfaces Fa0/1 y Se0/0/1 respectivamente. R2 ejecuta la misma operación, sus redes directamente conectadas son agregadas a su tabla de rutas y asociadas a sus respectivas interfaces.

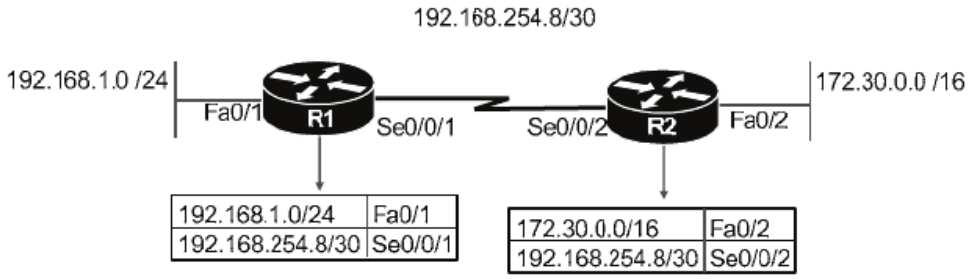


Fig. 6-1.1 Tabla de rutas con redes directamente conectadas.

Paso 2: R1 envía a R2 todas las redes incluidas en su tabla de rutas. R2 procede de la misma manera.

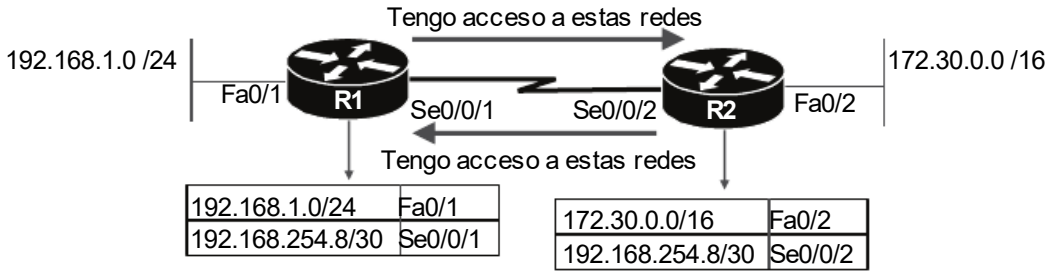


Fig. 6-1.2 Intercambio de rutas entre routers vecinos.

Paso 3: R1 analiza la información recibida y comprueba que existe una red remota de la cual no tenía conocimiento, por lo tanto, la agrega a su tabla de rutas, asociándola a la interfaz Se0/0/1 ya que la información fue recibida a través de la misma. A partir de ahora, cuando reciba un paquete con destino 172.30.0.0/16 lo reenviará a través de dicha interfaz, este llegará a R2, que a su vez lo reenviará a través de Fa0/2, siendo recibido por su destinatario. R2 lleva a cabo la misma operación, comprueba las rutas recibidas desde R1 y agrega la 192.168.1.0/24, asociándola a la interfaz Se0/0/2.

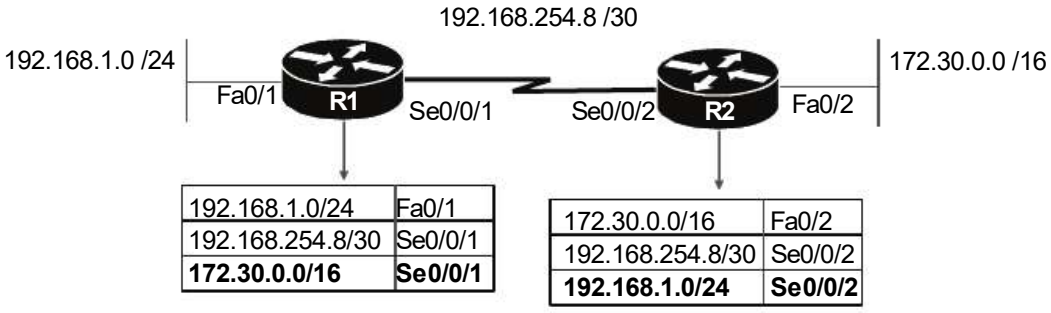


Fig. 6-1.3 Actualización de tabla de rutas.

Gracias a ello ambos han actualizado su tabla de rutas de manera automática. La comunicación permanecerá activa y cualquier cambio en la topología será notificado, en cuyo caso se realizarán las modificaciones oportunas.

Dependiendo del entorno donde deben ser aplicados, los protocolos de enrutamiento se agrupan en dos categorías: IGP *(Interior Gateway Protocol)* y EGP *(Exterior Gateway Protocol)*. Los primeros son aquellos diseñados para su implementación dentro del mismo sistema autónomo (AS). Mientras, los segundos han sido diseñados para su aplicación entre diferentes AS.

Un sistema autónomo *(Autonomous System - AS)* equivale a una red cuya administración depende por completo de una misma organización, por ejemplo, la red privada de cualquier compañía es administrada por la propia compañía, por lo tanto, es considerada un sistema autónomo. Los protocolos IGP han sido diseñados específicamente para ser aplicados en un mismo AS, en otras palabras, son aquellos a implementar en redes privadas. Actualmente, las opciones más comunes son RIPv2, EIGRP y OSPF. Mientras, los EGP son aquellos que permiten el intercambio de rutas entre diferentes sistemas autónomos y son utilizados normalmente por los ISP, siendo la única opción disponible a día de hoy BGP *(Border Gateway Protocol)*.

Protocolos de enrutamiento IGP

Como se ha mencionado anteriormente, los IGP han sido diseñados para su aplicación en redes privadas, siendo los más comunes RIPv2, EIGRP y OSPF. De todos ellos, RIPv2 es considerado inseguro y su sistema de métrica es poco eficiente. EIGRP es propietario de Cisco, por lo tanto solo puede ser configurado en sus routers, mientras que OSPF, al ser un protocolo abierto, representa la opción ideal sobre entornos multifabricante. EIGRP y OSPF hacen uso de sistemas de métrica eficientes, convirtiéndose en los protocolos más recomendables.

Las operaciones llevadas a cabo por todos ellos se basan por completo en algoritmos, los cuales definen la lógica y el procesamiento ejecutado para aprender rutas, seleccionar la mejor hacia cada subred y aplicar la convergencia cuando se produzca algún cambio en la topología. Existen 3 tipos principales de algoritmos IGP:

- *Vector distancia* (también conocido como *Bellman-Ford*, en honor a su creador).

- *Link state* (Estado del enlace).

- *Vector distancia avanzado* (también denominado balance híbrido).

El primero en ser desarrollado fue vector distancia, el cual basa su sistema de métrica principalmente en el conteo de saltos desde el origen hasta el destino. Se considera un salto cada vez que el paquete es reenviado en capa 3, por lo tanto, la mejor ruta será aquella cuyo número de routers, desde el origen hasta el destino, resulte menor. Los primeros protocolos en aplicarlo fueron, primero RIP, y posteriormente IGRP, ambos en desuso debido a la lenta convergencia y formación de loops de enrutamiento temporales.

Con el fin de solucionar dichos inconvenientes nace *link state*, el cual, en lugar del conteo de saltos, basa su cálculo en el estado de cada enlace, tomando como valores la velocidad, carga, retraso, etc. La ruta hacia un destino cuyos enlaces resulten más rápidos será considerada la mejor y por lo tanto la instalada y utilizada. Los protocolos que hacen uso de *link state* son OSPF *(Open Shortest Path First)* e IS-IS (*Intermediate System to Intermediate System).*

Por último, vector distancia avanzado fue desarrollado por Cisco para su aplicación en el protocolo EIGRP (*Enhanced Interior Gateway Protocol)* basando su modo de operar en una mezcla de características entre vector distancia y *link state*, aunque mayormente de este último.

Una de las funciones principales de todos ellos consiste en el cálculo de la métrica, siendo esta el valor numérico asignado por cada algoritmo para cada ruta hacia cada subred. Aquella cuyo valor resulte menor será la instalada en la tabla de rutas. Dicho cálculo varía dependiendo del protocolo aplicado, basándose cada uno de ellos en:

Protocolo	Métrica	Descripción
RIP-2	Conteo de saltos	Número de routers (saltos) entre el origen y el destino.
OSPF	Coste	La suma total del coste desde el origen hasta el destino. Su valor se basa, entre otros aspectos, en el ancho de banda de cada enlace.
EIGRP	Ancho de banda y retraso	La suma total del coste desde el origen hasta el destino. Su valor se basa en el ancho de banda y retraso en cada enlace.

En la práctica, la implementación de un protocolo u otro puede suponer la elección de diferentes rutas dentro de una misma topología, por ejemplo, ¿qué ruta instalará R1 para llegar hasta R3 en los siguientes supuestos?

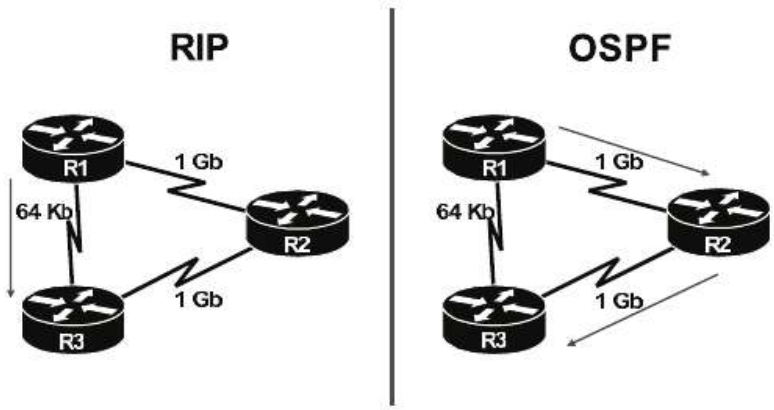


Fig. 6-2 Elección de rutas entre diferentes protocolos de enrutamiento.

Misma topología, diferentes protocolos de enrutamiento. Existen dos posibles rutas para llegar desde R1 hasta R3, una de ellas a través del enlace directo entre ambos (R1-R3) y otra mediante R2 (R1-R2-R3). En la topología con RIP, R1 instalará y hará uso de la ruta R1-R3 ya que tan solo existe un salto de diferencia entre el origen y el destino. Sin embargo, si el protocolo aplicado es OSPF, se instalará y utilizará la R1-R2-R3 porque es considerada más eficiente debido a la velocidad de los enlaces, siendo la comunicación más rápida aunque existan dos saltos de diferencia.

¿Y si se aplicaran los dos protocolos de manera simultánea en la misma topología? El resultado sería que se aprenderían rutas tanto desde RIP como desde OSPF, pero ¿cuál de ellas se instalará en la tabla de rutas? Para solucionar este problema se creó la distancia administrativa.

Distancia administrativa

La distancia administrativa (*AD - Administrative Distance)* consiste en un valor numérico en formato decimal que otorga una prioridad a cada ruta dependiendo del origen de esta, de manera que, cuando existen varias hacia un mismo destino, aprendidas por diferentes métodos, el router instalará y utilizará por defecto aquella cuyo AD sea menor.

Los valores para los diferentes tipos de rutas son:

Tipo de ruta	**Distancia administrativa**
Directamente conectada	0
Estática	1
BGP (uso externo)	20
EIGRP (rutas internas)	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
EIGRP (rutas externas)	170
BGP (rutas internas)	200

Volviendo a la topología de la *Fig. 6-2*, si se aplican los dos protocolos de enrutamiento de manera simultánea, RIP seleccionará la ruta R1-R3 como principal, mientras que OSPF la R1-R2-R3. En este caso, IOS analizará la procedencia de ambas, donde la primera obtiene un valor de AD de 120 mientras que la segunda de 110. Como esta es menor, R1 instalará y utilizará aquella aprendida por OSPF.

Tanto la distancia administrativa como la métrica de cada ruta pueden ser verificadas con el comando *show ip route*. De la información obtenida, sus valores son representados en el formato *[AD/métrica]*, ubicado al lado de la red de destino.

```
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
```

```
        * - candidate default, U - per-user static route, o - ODR
        P - periodic downloaded static route

Gateway of last resort is not set

C     172.20.0.0/16 is directly connected, FastEthernet0/1
O     192.168.1.0/24  [110/2] via 192.168.254.1, 00:00:21, FastEthernet0/0
C     192.168.254.0/24 is directly connected, FastEthernet0/0
```

La distancia administrativa también puede ser modificada manualmente. Para rutas estáticas bastará con agregar su valor al final de la sentencia, por ejemplo:

```
RouterA(config)# ip route 172.10.0.0 255.255.0.0 Se0/0/0 210
```

En este caso, la ruta configurada tendrá un valor de AD de 210.

Por último, los IGP también pueden ser catalogados como protocolos con clase *(classful)* o sin clase (*classless*). Un protocolo con clase es aquel que no envía la máscara de red en sus actualizaciones de enrutamiento, aplicando siempre la definida por defecto para cada tipo de dirección. Por ejemplo, para la red 172.20.0.0 automáticamente se aplica la máscara 255.255.0.0. ¿Qué consecuencias tiene este modelo? La desventaja de que VLSM (subredes con máscara variable) no es soportado, por lo tanto, no son protocolos aptos para su implementación en la gran mayoría de redes actuales. Un ejemplo de ello es RIP.

Por el contrario, los protocolos sin clase son aquellos que sí incluyen la máscara de red en sus actualizaciones de enrutamiento y por lo tanto pueden ser aplicados en redes con VLSM, algunos ejemplos son RIPv2, OSFP y EIGRP.

Protocolo	Classless	Algoritmo	Propietario
RIPv1	No	Ventor Distancia	Abierto
RIPv2	Sí	Vector Distancia	Abierto
EIGRP	Sí	Vector Distancia Avanzado	Cisco
OSPF	Sí	Link State	Abierto
IS-IS	Sí	Link State	Abierto

De los cuales, EIGRP, OSPF y RIPv2 forman parte del contenido de CCNA, siendo analizados en detalle en las próximas secciones.

EIGRP - ALGORITMO Y MODO DE OPERACIÓN

EIGRP es el protocolo de enrutamiento IGP propietario de Cisco nacido como mejora de su antecesor, IGRP. En párrafos anteriores se han mencionado algunas de

sus características de manera muy básica, como el algoritmo aplicado o su distancia administrativa. La presente sección será dedicada a realizar un análisis más profundo de cada una de sus funciones, ya que, junto con OSPF, representan las opciones más comunes a día de hoy.

Existen dos versiones, una para redes IPv4 (EIGRP), y otra para entornos IPv6 (EIGRPv6). El modo de operar en ambos casos coincide, pero adaptado a cada uno de los protocolos IP. Para comenzar se realizará un estudio de su algoritmo y modo de operación, continuando con su configuración en redes IPv4.

La configuración de EIGRPv6 será analizada en el capítulo 11, *"IP Versión 6"*.

Algoritmo aplicado en EIGRP

El algoritmo hace referencia al método utilizado para calcular las diferentes rutas y métricas hacia cada subred, con el fin de obtener e instalar la mejor de ellas hacia cada destino, logrando a su vez una topología libre de bucles de capa 3. Con anterioridad se ha hecho mención a las diferentes opciones existentes, siendo estas vector distancia, link state y vector distancia avanzado. EIGRP es el único protocolo que hace uso de vector distancia avanzado, el cual se puede entender como una mezcla de los dos anteriores, tomando las mejores características de cada uno de ellos y aplicando algunos cambios, lo que se traduce en ciertas mejoras. Su modo de operar se basa principalmente en cuatro características:

- Actualizaciones de enrutamiento parciales.
- Horizonte dividido.
- Envenenamiento de ruta.
- Cálculo de la métrica.

ACTUALIZACIONES DE ENRUTAMIENTO PARCIALES

Una gran diferencia entre vector distancia y vector distancia avanzado es la manera en la que ambos gestionan las actualizaciones de enrutamiento una vez finalizada la convergencia. El primero se basa en enviar la tabla de rutas completa a cada vecino (*full update*), además, también son utilizadas para mantener la adyacencia con cada uno de ellos, teniendo como consecuencia su envío cada 30 segundos por defecto y traduciéndose en un elevado consumo de ancho de banda y una lenta convergencia ante cualquier cambio en la red.

Vector distancia avanzado aplica una mejora sobre esta característica, basada en la utilización de actualizaciones de enrutamiento parciales, las cuales solo son

enviadas cada vez que se produzca algún cambio en la topología, y además, solo incluyen el cambio en cuestión. Este factor agrega beneficios sobre la red, como un consumo bastante reducido de ancho de banda y una convergencia más rápida ante cualquier cambio. La adyacencia con vecinos se mantiene a través de mensajes *hello*, que son más ligeros y no incluyen información de enrutamiento, siendo enviados por defecto cada 5 segundos en enlaces rápidos (fibra, Ethernet…) y cada 60 segundos en enlaces lentos (ISDN, ATM…).

HORIZONTE DIVIDIDO

Tanto vector distancia como vector distancia avanzado aplican la técnica de horizonte dividido, gracias a la cual se logra una topología libre de bucles de capa 3. Para ello, basa su modo de operar en un procedimiento tan sencillo como no reenviar una actualización de enrutamiento a través de la misma interfaz por la cual fue recibida.

Un ejemplo de cómo se genera un bucle sin aplicar horizonte dividido podría ser el siguiente:

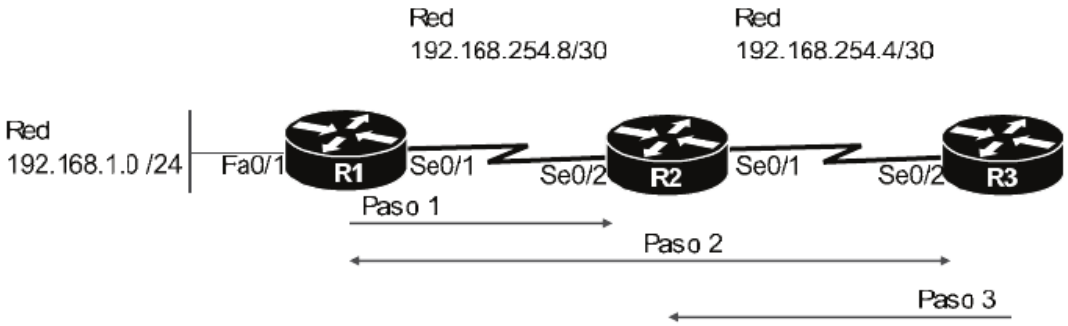


Fig. 6-3.1 Generación de un bucle en capa 3.

Con el fin de facilitar la comprensión, centrémonos tan solo en la red 192.168.1.0/24 y en el router R2.

Paso 1: R1 publica sus redes a sus vecinos directamente conectados.

Paso 2: R2 recibe la actualización, la examina y agrega a su tabla de rutas la red 192.168.1.0. Acto seguido la reenvía a sus vecinos. Como no se está aplicando el horizonte dividido, lo hace a través de todas sus interfaces, Se0/1 y Se0/2.

Paso 3: R3 recibe la actualización, la comprueba y agrega la red 192.168.1.0 a su tabla de rutas, para luego reenviarla a través de todas sus interfaces, en este caso Se0/2.

Llegados a este punto, R2 vuelve a recibir otra ruta hacia 192.168.1.0, pero esta vez a través de R3. ¿Cómo procede? Instalará y utilizará una, y la otra queda como respaldo. La recibida desde R1 obtiene mejor métrica, por lo tanto, hace uso de ella. Hasta ahora no existe ningún problema, pero ¿qué sucede si la red 192.168.1.0 cae?

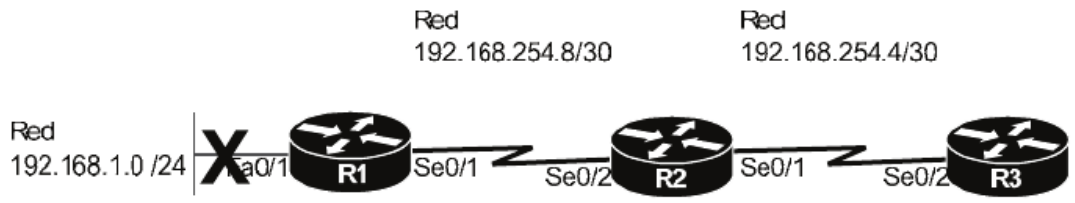


Fig. 6-3.2 Generación de un bucle en capa 3.

R1 informa a R2 y este elimina su ruta anteriormente recibida. Sin embargo, dispone de otra alternativa a través de R3, por lo tanto, la instala y utiliza, dando como resultado que los paquetes con destino a dicha red sean reenviados a R3, el cual a su vez comprueba su tabla de rutas y determina que para llegar a la 192.168.1.0 debe reenviar los paquetes a R2. Cuando R2 lo reciba hará lo mismo, comprueba su tabla y lo reenvía al mismo router. Se ha generado un bucle de capa 3.

Este hecho resulta fácilmente evitable aplicando la regla de horizonte dividido:

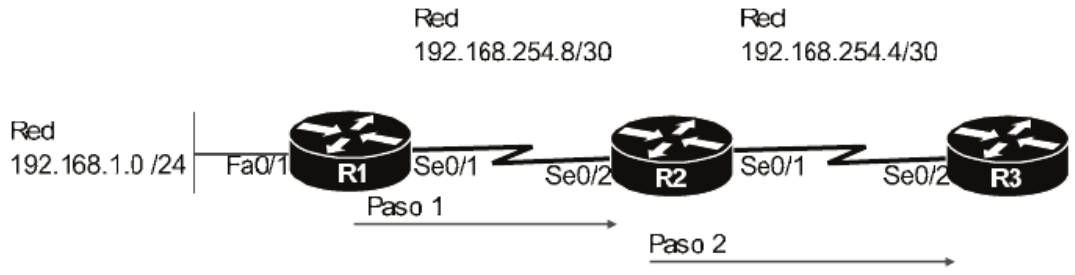


Fig. 6-4 Regla de horizonte dividido.

- *Paso 1:* R1 envía la actualización de enrutamiento a R2.

- *Paso 2:* R2 agrega la red 192.168.1.0 a su tabla de rutas. Como en este caso sí se aplica la regla de horizonte dividido, tan solo la publicará a través de la interfaz

Se0/1. R3 la recibe y agrega la nueva red. Como también aplica horizonte dividido no la reenvía a través de la misma interfaz por la cual fue recibida, de tal manera que R2 tan solo dispondrá de una ruta hacia 192.168.1.0, lo que se traduce en que en caso de caída, no se producirán bucles de capa 3.

ENVENENAMIENTO DE RUTA

Otro de los métodos llevados a cabo para evitar loops de enrutamiento consiste en el envenenamiento de ruta, el cual se basa en notificar a los vecinos de aquellas redes actualmente inaccesibles. Para ello, cuando un router detecta que una red ha caído, la marca como inalcanzable, asignándole una métrica preestablecida por el protocolo como infinita, por ejemplo, para RIP es aquella con valor 16, mientras que en EIGRP equivale a 2^{32}-1 (2^{56}-1 en algunas versiones de IOS). Aplicado el cambio envía una actualización a sus routers vecinos, y estos, al recibirla, la comprobarán y también la marcarán como inalcanzable. Acto seguido la reenvían a través de todas las interfaces, incluyendo aquella por la que fue recibida.

Continuando con el ejemplo anterior, cuando la red 192.168.1.0 cae (aplicando una métrica 16 para facilitar la comprensión):

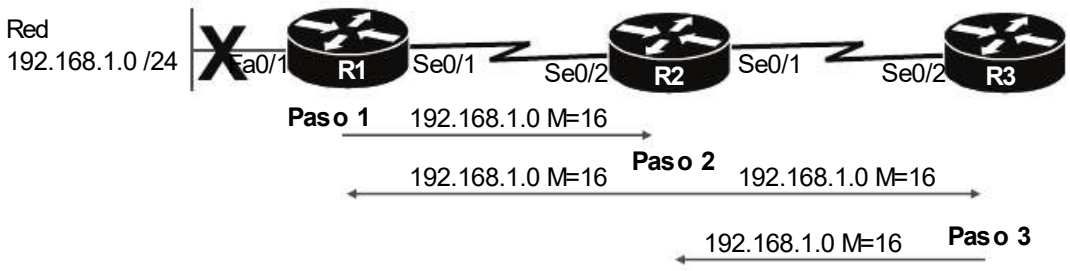


Fig. 6-5 Envenenamiento de ruta.

- *Paso 1:* R1 detecta que la red 192.168.1.0 ha caído, por lo tanto, envía una actualización a sus routers vecinos en la cual le asigna una métrica con valor infinito.

- *Paso 2:* R2 la examina y marca dicha red como inalcanzable. Acto seguido la reenvía a todos sus vecinos.

- *Paso 3:* La actualización es recibida por R1 y R3. El primero ya le había aplicado un valor infinito pero R3 no, por lo tanto, la marca como inalcanzable y la reenvía de nuevo. Esta llega a R2 que ya conoce este hecho y además ya lo ha notificado previamente, por lo que no reenvía de nuevo la actualización.

Dependiendo de la versión de IOS, cuando un router recibe una red como inalcanzable puede proceder de dos maneras, bien eliminarla de la tabla de rutas, o bien aplicarle métrica infinita durante un tiempo determinado esperando a que vuelva a estar activa. Si el tiempo concluye y aún no está operativa, es eliminada.

CÁLCULO DE MÉTRICA

En EIGRP, el cálculo de la métrica para cada una de las rutas se basa principalmente en el ancho de banda y retraso de los enlaces intervinientes, ejecutando una ecuación matemática donde se incluyen ambos factores. El resultado dará el valor a aplicar y gracias a ello se determinará cuál utilizar cuando se han aprendido varias hacia la misma subred.

Dicho cálculo se basa en la siguiente ecuación:

$$\textbf{Métrica} = \left(\left(\frac{10^7}{\text{Menor ancho de banda}} \right) + \text{retraso} \right) * 256$$

"Menor ancho de banda" hace referencia al enlace cuyo ancho de banda resulte el más bajo de todos los intervinientes, por ejemplo, para una ruta que atraviesa dos links, uno de 1 Gb y otro de 100 mb, la métrica se calculará aplicando el valor de 100 mb ya que es el más lento. Este debe ser aplicado en Kb, es decir, 100 Mb equivalen 100000 Kb, que a su vez es igual a 10^5.

"Retraso" es el tiempo que tarda un paquete desde el origen hasta el destino. Este factor es importante ya que las rutas con mayor número de enlaces o aquellas que presenten sobrecarga darán como resultado un retraso mayor y por lo tanto una métrica más elevada.

Por último, ambos valores pueden ser modificados mediante los comandos **bandwidth *[ancho de banda]*** y **delay *[retraso]***, desde el modo de configuración de cada interfaz, por lo que es posible influir de manera manual en el resultado de la métrica calculada por EIGRP.

En el examen de CCNA no es necesario realizar el cálculo de métrica, pero sí resulta importante conocer el significado de "menor ancho de banda" y "retraso", y cómo pueden los administradores de red influir manualmente sobre su resultado.

Modo de operación

En EIGRP, el modo de operación hace referencia principalmente al descubrimiento de vecinos, intercambio de información y selección de rutas.

DESCUBRIMIENTO DE VECINOS

Un aspecto imprescindible en cualquier IGP es la manera en la que los routers descubren y establecen asociaciones con otros dispositivos que ejecutan el mismo protocolo, con la finalidad de intercambiar información de enrutamiento entre ellos.

En párrafos anteriores se ha descrito que los mensajes *hello* son utilizados para mantener adyacencias. Bien, además de dicha función, también se encargan de descubrir nuevos vecinos, llevando a cabo el siguiente proceso:

- *Paso 1:* El router envía paquetes *hello* a la dirección multicast 224.0.0.10 (reservada para EIGRP) a través de todas las interfaces configuradas para ello.

- *Paso 2:* Estos son recibidos por los routers vecinos, que analizarán la información contenida y dependiendo de esta formarán adyacencia o no. El intercambio de rutas tan solo se llevará a cabo con aquellos con lo que se logre establecer relación.

Para gestionar tanto el envío de estos paquetes como las adyacencias establecidas, los routers Cisco hacen uso de dos temporizadores, *Hello Interval* y *Hold Interval.* El primero define un tiempo periódico, en segundos, utilizado para el envío de mensajes *hello*, siendo su valor por defecto de 5 segundos en enlaces rápidos como fibra o Ethernet. Mientras, *Hold Interval* hace referencia al tiempo de espera de un router para dar por concluida una adyacencia de la cual no se ha recibido ningún mensaje desde el otro extremo. Por defecto, este tiempo es de 15 segundos. Cada vez que se recibe un mensaje de algún vecino, el temporizador *hold* asociado al mismo es reiniciado a 0.

Los *hello* contienen información que será analizada por el router que lo recibe para determinar si se establece, o no, adyacencia. Los datos más relevantes son:

- Dirección IP de la interfaz por la cual se envía el mensaje.
- Autenticación de EIGRP (si es configurada).
- Número de ASN de EIGRP al que pertenece la interfaz desde la cual es enviado.

Conforme a ello, la asociación entre dos routers se llevará a cabo siempre y cuando se cumplan los siguientes requisitos:

- La dirección IP del mensaje forma parte del mismo rango de red que la IP configurada en la interfaz por la cual fue recibido.
- La autenticación, en caso de que fuera definida, coincide con la autenticación local de EIGRP.
- El número de ASN del paquete *hello* coincide con el configurado en la interfaz del router local a través de la cual fue recibido.

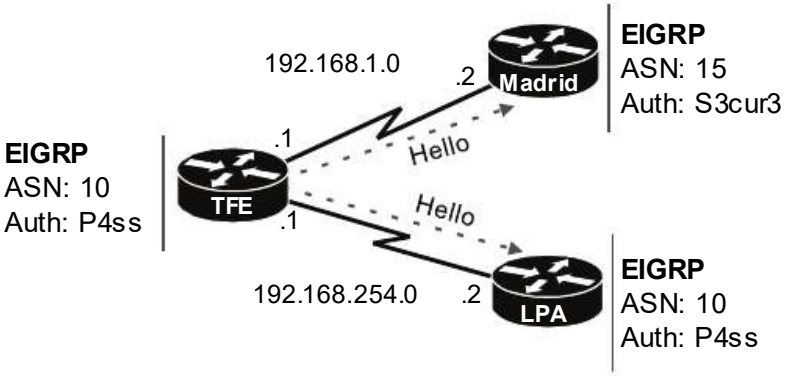


Fig. 6-6 Adyacencia de vecinos en EIGRP.

En el router TFE se ha configurado EIGRP como protocolo de enrutamiento, de tal manera que inicia la búsqueda de nuevos vecinos mediante el envío de mensajes *hello* a través de todas sus interfaces, los cuales serán recibidos por Madrid y LPA.

- Madrid analiza la información incluida, que contendrá los siguientes datos: IP: 192.168.1.1, ASN: 10, Auth: P4ss, y lo compara con su propia configuración de EIGRP. Madrid forma parte del ASN 15 y como autenticación aplica la cadena "S3cur3", parámetros que no coinciden con los recibidos. Entre Madrid y TFE **no** se establecerá adyacencia EIGRP.

- El router LPA también analiza el mensaje, el cual contendrá los siguientes datos: IP: 192.168.254.1, ASN: 10, Auth: P4ss, y los compara con su configuración. En este caso los parámetros sí coinciden y la IP de la interfaz por la cual fue recibido (192.168.254.2) forma parte del mismo rango de red que la IP incluida en el *hello.* Se cumplen todos los requisitos para que TFE y LPA formen adyacencia, por lo tanto se establece la relación y comienza el intercambio de rutas entre ambos.

El ASN y configuración serán analizados en la sección "EIGRP - Configuración y verificación en redes IPv4", a continuación en este mismo capítulo.

INTERCAMBIO DE INFORMACIÓN

Una vez se ha establecido la relación entre dos vecinos se procede al intercambio de información entre ambos a través de mensajes de actualización (*update messages*), que pueden ser enviados de manera multicast a la dirección 224.0.0.10 o unicast a la dirección IP del router vecino. El primer método suele resultar más habitual en enlaces multipunto donde la información debe ser recibida por más de un destinatario, mientras que el segundo es aplicado en enlaces punto a punto. Además, para este proceso, EIGRP hace uso del protocolo RTP *(Reliable Transport Protocol)* el cual provee mecanismos de control de errores, por lo que si un *update message* no fuera recibido por su destinatario, sería reenviado. Este detalle marca una diferencia importante respecto a otros protocolos de enrutamiento vector distancia, los cuales no incluyen esta característica, debido a lo cual se podrían producir bucles de capa 3 temporales.

En EIGRP, el intercambio de información entre routers vecinos consta del siguiente procedimiento:

- *Paso 1:* Cuando se establece la adyacencia, ambos routers envían una actualización completa (*full update*), la cual incluye toda la información de enrutamiento almacenada en sus tablas de rutas.

- *Paso 2:* Los vecinos continúan manteniendo la relación mediante mensajes *hello*.

- *Paso 3:* Ante cualquier cambio en la topología, como la caída de un enlace o la inclusión de una nueva red, se envían actualizaciones parciales *(partial updates)*, las cuales tan solo contienen la información del cambio en cuestión.

Es decir, la actualización completa únicamente es enviada cuando se establece la adyacencia, de ahí en adelante los routers simplemente se limitan a mantener la relación mediante mensajes *hello* y a notificar cualquier cambio en la topología mediante actualizaciones parciales.

El protocolo RTP *(Reliable Transport Protocol)* de EIGRP no debe ser confundido con *Real-Time Transport Protocol*. Aunque ambos hagan uso de las mismas siglas, este último es utilizado para transmisiones de voz y vídeo.

SELECCIÓN DE RUTAS

Por último, EIGRP debe seleccionar qué ruta, de todas las disponibles, instalará hacia cada subred. Esta decisión se basa por completo en el valor de métrica resultante para cada una de ellas, haciendo uso de la más baja hacia cada destino y

almacenando las restantes como respaldo. Pero, y si la ruta principal cae, ¿cuál de las disponibles toma su relevo? Para este propósito, y con el fin de evitar loops de enrutamiento, EIGRP define dos tipos de métricas, denominadas distancia factible y distancia reportada.

- *Distancia factible (Feasible Distance)*: La distancia factible (FD) es la métrica calculada desde el router local hasta cada subred.

- *Distancia reportada (Reported Distance)*: La distancia reportada (RD) es la mejor métrica desde el router del siguiente salto hasta cada subred.

Estas, que pueden llevar a confusión, se comprenden mejor de manera práctica:

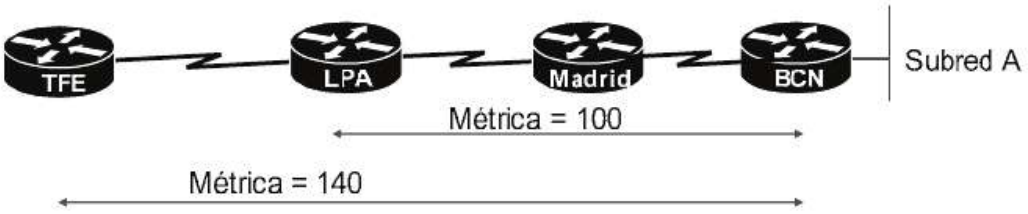


Fig. 6-7 Distancia factible y distancia reportada.

¿Qué valores de FD y RD almacenará el router TFE para la subred A?

La distancia factible es la métrica desde el router local hasta la subred, por lo tanto, obtiene un valor de 140, mientras que la distancia reportada es la mejor métrica desde el router del siguiente salto hacia la subred en cuestión. En este caso, LPA actúa como siguiente salto y su mejor métrica hacia A es de 100, por lo tanto, la distancia reportada de TFE para la subred A es igual a 100.

En relación con estos dos valores EIGRP seleccionará la ruta que instalará en la tabla, la cual es denominada “sucesor”, mientras que las posibles alternativas reciben el nombre de “sucesor factible”:

- *Sucesor:* Es la mejor opción disponible hacia cada subred y por lo tanto la que se instalará y utilizará. Corresponde a aquella con valor más bajo de distancia factible (FD).

- *Sucesor factible:* Son aquellas que EIGRP considera candidatas a utilizar en caso de que la principal caiga, es decir, las que se almacenan como respaldo a la actualmente en uso.

Sin embargo, para que EIGRP considere una ruta como sucesor factible debe cumplir una condición, y es que la distancia reportada (RD) debe ser menor que la distancia factible (FD) de la ruta actual. Dicho requisito se lleva a cabo con el fin de evitar bucles de capa 3.

En el siguiente ejemplo, ¿qué ruta seleccionará TFE como sucesor y cuáles como sucesor factible para la subred A?

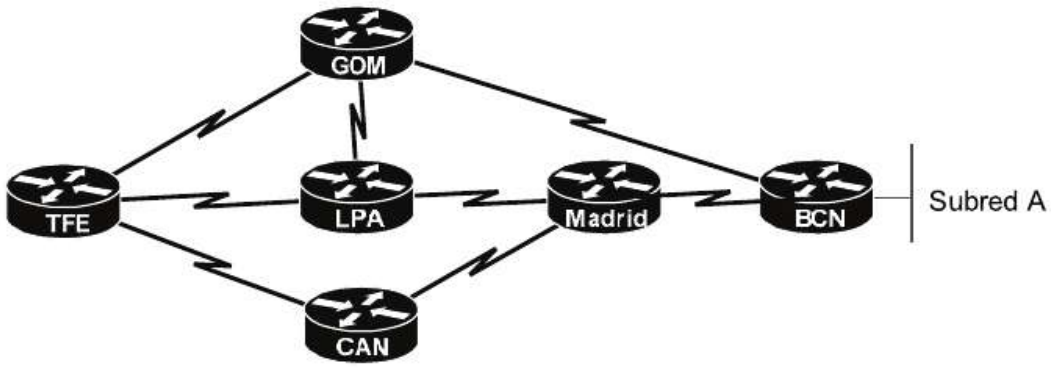


Fig. 6-8 Selección de rutas sucesor y sucesor factible.

EIGRP ha calculado y almacenado los siguientes valores de FD y RD:

Siguiente Salto	Distancia Factible (FD)	Distancia Reportada (RD)
GOM	8,000	6,000
LPA	13,000	9,500
CAN	9,100	7,500

En relación con ello, instala y utiliza como ruta principal (sucesor) aquella cuyo siguiente salto corresponde al router GOM (TFE-GOM-BCN) ya que esta obtuvo el mejor valor de FD.

¿Cuál o cuáles actuarán como sucesores factibles?

La condición para que una ruta sea considerada sucesor factible es que su RD resulte menor que la FD de aquella seleccionada como sucesor, cuyo valor en este caso es igual a 8,000. Por lo tanto, se analizan las distancias reportadas de las rutas restantes.

- A través de LPA, la RD hacia a la subred A es de 9,500, que es mayor a 8,000 y por lo tanto no cumple la condición para ser sucesor factible.

- A través del router CAN la distancia reportada obtiene un valor de 7,500. En este caso sí es menor a 8,000, por lo tanto cumple la condición y es considerada sucesor factible para la subred A.

Si la ruta TFE-GOM-BCN cae, EIGRP automáticamente instala aquella a través de CAN.

Por último, si un router no dispone de ningún sucesor factible hacia una determinada subred y su ruta sucesor cae, EIGRP ejecuta un algoritmo denominado DUAL *(Diffusing Update Algorithm)*, cuya función consiste en enviar mensajes de petición (*query messages)* a sus vecinos, consultando por alguna ruta disponible y a su vez libre de loops para la subred en cuestión. Estos responderán con un *reply message*, informando si disponen de ella, o no. En caso positivo, la instalará y utilizará.

EIGRP - CONFIGURACIÓN Y VERIFICACIÓN EN REDES IPV4

La configuración de EIGRP en IPv4 consta de un procedimiento bastante sencillo, donde se deberá definir un número de ASN y las redes directamente conectadas que publicará el protocolo. Además, IOS dispone de parámetros opcionales que permitirán optimizar su rendimiento al máximo.

Su configuración más básica consta de:

- *Paso 1:* Habilitar EIGRP con el comando **router eigrp *[número de ASN]***, desde el modo de configuración global. ASN hace referencia al número de sistema autónomo al que pertenecerá esta instancia del protocolo. Uno de los requisitos necesarios para que dos routers formen adyacencia es que coincidan en este parámetro, por lo tanto, todos los dispositivos de la misma topología EIGRP deben aplicar el mismo valor. Al ejecutar el comando se accede al modo de configuración de EIGRP.

- *Paso 2:* Definir qué redes directamente conectadas serán publicadas por el protocolo. Además, esta acción implica que las interfaces locales que conectan con las mismas formen parte de EIGRP, enviando mensajes en busca de posibles vecinos y adyacencias. Para llevarlo a cabo se debe aplicar el comando **network *[red] [wildcard]*** desde el modo de configuración de EIGRP. Se debe indicar la dirección de red a publicar y la máscara wildcard de esta, que corresponde al valor inverso de la máscara de red, por ejemplo, para una 255.255.0.0, la wildcard será 0.0.255.255.

Un ejemplo podría ser el siguiente:

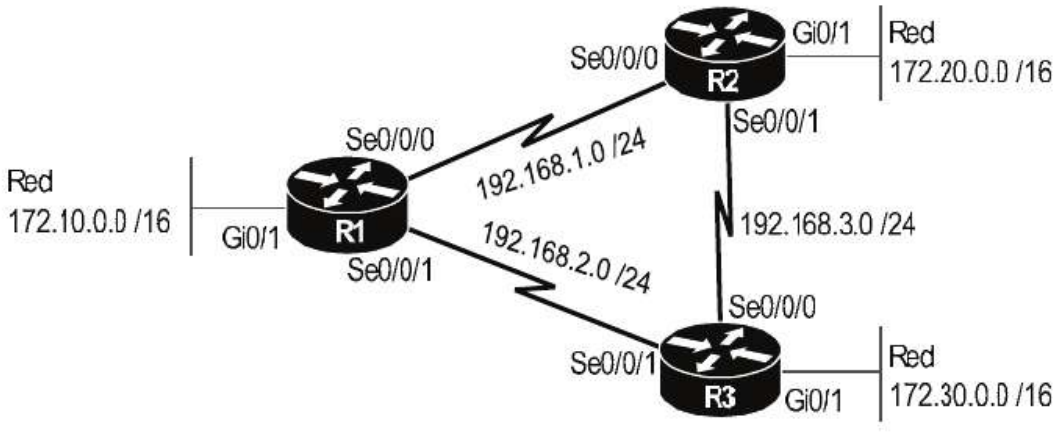


Fig. 6-9 Diseño de red para ejemplo de EIGRP.

Una buena práctica antes de proceder a su configuración consiste en planificarla, en este caso, al tratarse de una topología pequeña no resulta estrictamente necesario, pero en entornos corporativos se convierte en una tarea imprescindible. En cada router deberán definirse las redes a publicar, siendo aquellas directamente conectadas, de tal manera que:

Router	Redes a publicar	ASN de EIGRP
R1	172.10.0.0 /16 192.168.1.0 /24 192.168.2.0 /24	10
R2	172.20.0.0 /16 192.168.1.0 /24 192.168.3.0 /24	10
R3	172.30.0.0 /16 192.168.2.0 /24 192.168.3.0 /24	10

Como número de ASN se ha seleccionado el 10 pero puede ser cualquier otro, siempre y cuando se aplique el mismo en todos los routers que formarán parte de la misma topología.

```
---Configuración en R1---
R1(config)#router eigrp 10
R1(config-router)#network 172.10.0.0 0.0.255.255
R1(config-router)#network 192.168.1.0 0.0.0.255
R1(config-router)#network 192.168.2.0 0.0.0.255
```

```
---Configuración en R2---
R2(config)#router eigrp 10
R2(config-router)#network 172.20.0.0 0.0.255.255
R2(config-router)#network 192.168.3.0 0.0.0.255
R2(config-router)#network 192.168.1.0 0.0.0.255

%DUAL-5-NBRCHANGE: IP-EIGRP 10: Neighbor 192.168.1.1 (Serial0/0/0) is up: new
adjacency

---Configuración en R3---
R3(config)#router eigrp 10
R3(config-router)#network 172.30.0.0 0.0.255.255
R3(config-router)#network 192.168.2.0 0.0.0.255
R3(config-router)#network 192.168.3.0 0.0.0.255

%DUAL-5-NBRCHANGE: IP-EIGRP 10: Neighbor 192.168.2.1 (Serial0/0/1) is up: new
adjacency
%DUAL-5-NBRCHANGE: IP-EIGRP 10: Neighbor 192.168.3.1 (Serial0/0/0) is up: new
adjacency
```

El comando *network* también puede ser aplicado sin la wildcard, pero en este caso EIGRP aplicará la máscara por defecto para la red que se incluya.

Como se puede observar, a medida que se aplica la configuración en los diferentes routers se van estableciendo las adyacencias entre ellos, gracias a lo cual intercambian y aprenden rutas de manera automática.

Los comandos ejecutados resultan imprescindibles para que el protocolo lleve a cabo su finalidad. Conforme a ellos EIGRP dispone de diferentes características opcionales que permitirán optimizar y personalizar su modo de operar, siendo las siguientes:

Todos los comandos a continuación descritos son aplicados desde el modo de configuración de EIGRP, exceptuando los intervalos *hello* y *hold* y el retraso y ancho de banda de los enlaces.

1.- Interfaces pasivas: Son aquellas a través de las cuales EIGRP no enviará paquetes *hello*, por lo tanto, no se establecerá ninguna relación mediante ellas. Una buena práctica consiste en configurar como pasivas aquellas que únicamente conectan con dispositivos finales. Ello es debido a que estos no participan en el intercambio de rutas, por lo que resulta innecesario intentar formar adyacencias, evitando también posibles ataques de usuarios malintencionados. Un ejemplo es la interfaz Gi0/1 de R1 de la topología recién configurada, la cual conecta con la red 172.10.0.0 y que únicamente está compuesta por dispositivos finales (PCs, impresoras, etc.). Lo mismo es aplicable a las interfaces Gi0/1 de R2 y R3. Para llevar a cabo esta configuración se debe ejecutar el comando **passive-interface *[interfaz]***.

2.- Router ID: Todo router que forme parte de EIGRP debe ser identificado con un ID, denominado RID (*Router ID*). En versiones de IOS 12.1 o posteriores puede ser definido manualmente con el comando **eigrp router-id *[x.x.x.x]***, donde *x.x.x.x* hace referencia a un valor numérico, en formato decimal y dividido en 4 octetos de 8 bits cada uno. Es decir, igual que una dirección IPv4. Si no fuera configurado, el router seleccionará uno automáticamente con relación al siguiente criterio:

Se tomará como RID el valor de la dirección IP más alta configurada en interfaces loopback cuyo estado sea *up/up*.

Si no existen interfaces loopback o estas no se encuentran en estado *up/up*, se aplicará como RID la dirección IP más alta de las interfaces físicas configuradas cuyo estado sea up/up.

3.- Balanceo de carga: EIGRP por defecto ejecuta balanceo de carga en rutas cuya métrica coincida con aquella instalada en la tabla. Esta circunstancia no suele ser habitual, hecho por el cual IOS, con el fin de proporcionar un mejor aprovechamiento del ancho de banda y de los enlaces disponibles, permite su configuración para que también se lleve a cabo en rutas con diferente métrica. Para lograrlo resulta necesario aplicar el comando **variance *[x]***, donde *x* hace referencia a un valor decimal comprendido entre 1 y 128 que indica el factor a multiplicar por la mejor métrica. Es decir, imaginemos que R1 dispone de 3 rutas para llegar a la misma red, con métricas 10, 20 y 40, siendo la mejor de ellas y por lo tanto la utilizada aquella con valor 10. Sin embargo, deseamos que se realice balanceo de carga, incluyendo también la ruta con métrica 20. Para lograrlo bastará con ejecutar el comando *variance 2,* cuyo resultado será multiplicar por dos la mejor métrica (10*2 = 20), logrando así el objetivo deseado.

Un detalle importante es que para que una ruta sea candidata para balanceo de carga debe ser considerada sucesor factible. Por defecto, el valor de *variance* es igual a 1.

4.- Número máximo de rutas: Al igual que puede ser configurado manualmente el balanceo de carga entre diferentes métricas, también es posible limitar el número máximo de rutas para dicho balanceo gracias al comando **maximum-paths *[x]***. Su valor debe estar comprendido entre 1 y 32, y si no fuera configurado, el aplicado por defecto es 4. El valor 1 indica que no se realizará balanceo de carga.

5.- Intervalos hello y hold: Los valores *Hello Interval* y *Hold Interval*, analizados en párrafos anteriores, también pueden ser modificados de manera manual. Los comandos para ello son **ip hello-interval eigrp *[asn] [segundos]*** e **ip hold-time eigrp *[asn] [segundos]***, ambos desde el modo de configuración de la interfaz. Si no fueran

definidos se aplican los valores por defecto, siendo 5 segundos para *hello* y 15 para *hold*.

6.- Sumarización automática: Consiste en agrupar diferentes subredes, normalmente contiguas, en una sola entrada en la tabla de rutas. Este hecho, en entornos de gran tamaño y con un buen diseño de subnetting, se traduce en mayor velocidad de procesamiento y enrutamiento. Para que se lleve a cabo bastará con ejecutar el comando **auto-summary**.

7.- Retraso y ancho de banda de los enlaces: Con el fin de influir manualmente sobre la selección de rutas, resulta posible definir el ancho de banda y retraso (*delay*) de cada uno de los enlaces intervinientes hacia cada destino. Gracias a ello, y estableciendo los mismos valores, se obtendrá idéntica métrica sobre diferentes rutas, y por lo tanto, se realizará balanceo de carga sin necesidad de aplicar el comando *variance*. Su configuración se lleva a cabo mediante las sentencias **bandwidth *[ancho de banda]*** y **delay *[segundos],*** ambas desde el modo de configuración de la interfaz y donde *ancho de banda* debe ser definido en Kb.

Aunque dichos parámetros resulten opcionales, la mayoría son recomendables con el fin de lograr un mejor rendimiento del protocolo.

Aplicar los siguientes cambios en la topología EIGRP configurada en párrafos anteriores:

Router	Interfaces Pasivas	RID	Variance	Max. Paths	Sumarización
R1	Gi0/1	1.1.1.1	2	2	Sí
R2	Gi0/1	2.2.2.2	2	2	Sí
R3	Gi0/1	Por defecto	2	2	Sí

Los valores *hello, hold*, *bandwidth* y *delay* se mantendrán por defecto.

```
---Configuración R1---
R1(config)#router eigrp 10
R1(config-router)#passive-interface Gi0/1
R1(config-router)#eigrp router-id 1.1.1.1
R1(config-router)#variance 2
R1(config-router)#maximum-paths 2
R1(config-router)#auto-summary

---Configuración R2---
R2(config)#router eigrp 10
R2(config-router)#passive-interface Gi0/1
R2(config-router)#eigrp router-id 2.2.2.2
R2(config-router)#variance 2
R2(config-router)#maximum-paths 2
```

```
R2(config-router)#auto-summary

---Configuración R3---
R3(config)#router eigrp 10
R3(config-router)#passive-interface Gi0/1
R3(config-router)#variance 2
R3(config-router)#maximum-paths 2
R3(config-router)#auto-summary
```

El RID no ha sido definido manualmente en R3. ¿Qué valor tomará? Como tampoco dispone de interfaces loopback, buscará la IP más alta configurada en interfaces físicas operativas. En este caso, estas hacen uso de las direcciones 172.30.0.1, 192.168.3.1 y 192.168.2.2, por lo tanto, el RID de EIGRP en R3 será igual a 192.168.3.1.

Verificación de redes en IPv4

Concluida la configuración se deberá verificar que los cambios han sido aplicados según lo previsto. Para ello, IOS dispone los siguientes comandos:

- *show ip eigrp interfaces:* Muestra información en pantalla sobre cada proceso del protocolo configurado en el router y las interfaces que forman parte de él.

```
R1#show ip eigrp interfaces
IP-EIGRP interfaces for process 10

                        Xmit Queue   Mean   Pacing Time   Multicast    Pending
Interface        Peers  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Se0/0/0            1        0/0      1236       0/10          0           0
Se0/0/1            1        0/0      1236       0/10          0           0
```

La interfaz Gi0/1 no participa porque fue configurada como pasiva.

- *show ip eigrp topology:* Muestra en pantalla todas las rutas aprendidas por el protocolo, incluyendo información como su distancia factible, distancia reportada o interfaz a través de la cual fue recibida cada una de ellas.

```
R1#show ip eigrp topology
IP-EIGRP Topology Table for AS 10

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status

P 172.10.0.0/16, 1 successors, FD is 2816
         via Connected, GigabitEthernet0/1
P 192.168.1.0/24, 1 successors, FD is 2169856
         via Connected, Serial0/0/0
P 192.168.2.0/24, 1 successors, FD is 2169856
         via Connected, Serial0/0/1
P 172.20.0.0/16, 1 successors, FD is 2170112
```

```
        via 192.168.1.2 (2170112/2816), Serial0/0/0
P 192.168.3.0/24, 2 successors, FD is 2681856
        via 192.168.1.2 (2681856/2169856), Serial0/0/0
        via 192.168.2.2 (2681856/2169856), Serial0/0/1
P 172.30.0.0/16, 1 successors, FD is 2170112
        via 192.168.2.2 (2170112/2816), Serial0/0/1
```

En relación con el resultado obtenido se puede comprobar que EIGRP dispone de dos rutas hacia la red 192.168.3.0, una de ellas a través de R2 (192.168.1.2) y la otra mediante R3 (192.168.2.2). Para ambas, lo valores "(2681856/2169856)" corresponden a la distancia factible (FD) y la distancia reportada (RD) de cada ruta (en este caso coinciden).

- *show ip eigrp neighbors:* Muestra los vecinos con los cuales se ha establecido y mantiene adyacencia, incluyendo diferentes datos como su dirección IP y la interfaz local a través de la cual conectan.

```
R1#show ip eigrp neighbors
IP-EIGRP neighbors for process 10
H   Address         Interface       Hold Uptime    SRTT   RTO   Q   Seq
                                    (sec)          (ms)       Cnt Num
0   192.168.1.2     Se0/0/0         10   00:17:05  40     1000  0   7
1   192.168.2.2     Se0/0/1         11   00:17:02  40     1000  0   14
```

Otros comandos útiles son:

- *show ip eigrp traffic:* Muestra información sobre el tráfico generado por EIGRP, como paquetes *hello* enviados y recibidos, actualizaciones de enrutamiento, acks, etc.

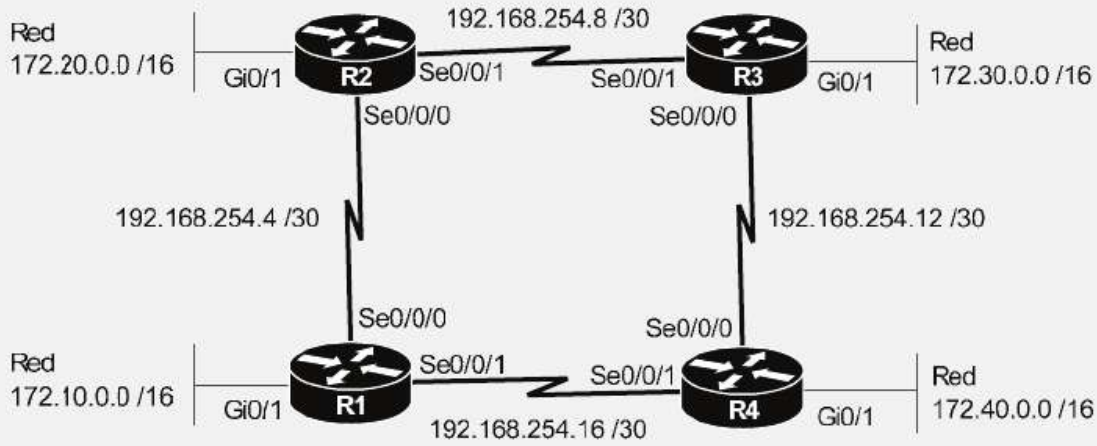
- *show ip protocols:* Muestra información sobre los diferentes protocolos que han sido configurados en el router.

- *show ip route eigrp:* Muestra en pantalla las entradas incluidas en la tabla de rutas con procedencia de EIGRP. Estas son representadas mediante el código "*D*", facilitando datos como su métrica, distancia administrativa, siguiente salto o IP del vecino a través de la cual fue aprendida la ruta.

Reto 6.1 – Configurar EIGRP como protocolo de enrutamiento en la siguiente topología de tal manera que:

- Todos los routers formen parte del ASN 15.
- Configurar como interfaces pasivas aquellas que se consideren oportunas.
- Se debe habilitar el balanceo de carga para un máximo de 5 rutas.

- El RID debe ser seleccionado automáticamente por cada router. ¿Qué valor se asignará en cada uno de ellos?
- Los intervalos *hello* y *hold* deben tomar los valores por defecto de EIGRP.
- El ancho de banda y el retraso de cada enlace no deben ser modificados.



Solución al final del capítulo.

OSPF - ALGORITMO Y MODO DE OPERACIÓN

OSPF *(Open Shortest Path First)* representa, junto a EIGRP, el protocolo de enrutamiento más habitual en redes IPv4, con una ventaja sobre este último, y es que al ser abierto puede ser aplicado en routers de cualquier fabricante, mientras que EIGRP es propiedad de Cisco. Sin embargo, la finalidad de ambos resulta ser la misma, consistente en el intercambio de rutas y en lograr una topología libre de bucles de capa 3. Para ello, y con el fin de ser soportado sobre cualquier entorno, OSPF dispone de dos versiones, OSPFv2 para IPv4 y OSPFv3 para IPv6, coincidiendo ambas en el algoritmo aplicado y diferenciándose tan solo en el proceso de configuración.

La presente sección será dedicada a realizar un estudio detallado de su algoritmo, *link state*, y el modo de operar del protocolo, para finalizar con su configuración en redes IPv4.

OSPFv3 será analizado en el capítulo 11, *"IP Versión 6".*

Algoritmo aplicado en OSPF

OSPF hace uso del algoritmo *link state* para llevar a cabo el intercambio de información entre sus miembros, atendiendo a dos propósitos, primero, lograr una topología libre de bucles de capa 3, y segundo, que todos los routers que la componen dispongan exactamente de la misma base de datos al finalizar la convergencia, la cual estará compuesta por los mensajes OSPF generados y enviados entre los diferentes vecinos. Este hecho marca una diferencia importante en cuanto a EIGRP, donde los datos obtenidos hacia cada destino resultan limitados (distancia factible y distancia reportada).

En *link state,* cada router recolecta información de sus redes directamente conectadas para luego distribuirla a lo largo de la topología. Gracias a ello, cada dispositivo genera una base de datos que incluye todos los mensajes recibidos, en relación con los cuales se calcularán las mejores rutas hacia cada subred. ¿Cómo se lleva a cabo este procedimiento? Para analizarlo en profundidad resulta necesario conocer y distinguir la función entre los siguientes elementos:

- *LSA:* El LSA (*Link State Advertisement*) es el mensaje que genera cada router para publicar una red, incluyendo en él toda la información que deben conocer los vecinos sobre la misma (ID, máscara, siguiente salto, métrica, etc.).

- *LSU:* Los LSU (*Link State Update*) son los paquetes intercambiados entre los routers para compartir información de enrutamiento. Están compuestos por LSA, por lo tanto, un paquete LSU puede incluir diferentes mensajes LSA.

- *LSDB:* La LSDB (*Link State Data Base*) es la base de datos almacenada por *link state* en cada router. Al inicio del proceso tan solo estará compuesta por las redes directamente conectadas del dispositivo, pero a ella se va agregando la información incluida en los LSU recibidos desde los diferentes vecinos. Al finalizar la convergencia, todos los routers deben haber generado la misma LSDB ya que todos deben haber recibido los mismos LSU.

OSPF basa por completo el intercambio de información entre vecinos en los LSA y LSU, con el fin de generar la LSDB, sin embargo, el procedimiento aplicado por el protocolo varía dependiendo del tipo de enlace, diferenciando entre punto a punto y multiacceso:

INTERCAMBIO DE RUTAS EN ENLACES PUNTO A PUNTO

Un enlace punto a punto identifica dos dispositivos directamente conectados, por lo que el procedimiento ejecutado por OSPF para el intercambio de información entre estos se torna bastante sencillo, pudiendo ser resumido en los siguientes 4 pasos:

- *Paso 1:* Los routers inician la búsqueda de vecinos OSPF mediante el envío de mensajes *hello* a través de las interfaces configuradas para tal propósito.

- *Paso 2:* Cada router genera un LSA para cada una de sus redes directamente conectadas, incluyendo toda la información necesaria sobre las mismas.

- *Paso 3:* Dichos LSA son agregados a uno o varios paquetes LSU y a su vez enviados a los diferentes vecinos a través de los enlaces punto a punto, los cuales ejecutan dos acciones, primero, agregan la información recibida a su propia base de datos, y segundo, la reenvían a los routers restantes. El proceso se repite hasta que todos los miembros de la topología dispongan de dicha información.

- *Paso 4:* Gracias a ello se genera la misma LSDB en cada dispositivo, la cual estará compuesta por todos los mensajes recibidos. De esta manera, cada router tiene conocimiento no solo de sus vecinos directamente conectados, sino de todos los dispositivos y rutas que forman la topología OSPF.

Para analizarlo de manera práctica centremos el siguiente ejemplo en el router R1:

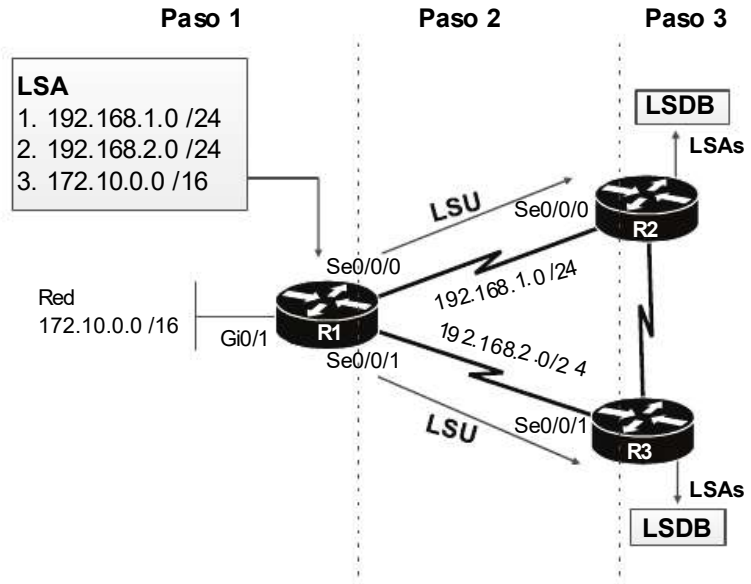


Fig. 6-10 OSPF. Intercambio de rutas en enlaces punto a punto.

R1 conecta directamente con R2 y R3 a través de enlaces punto a punto. Tras el descubrimiento de vecinos mediante el envío de mensajes *hello* comienza el proceso de intercambio de rutas. En el paso 1, genera un LSA por cada red directamente conectada. En el paso 2, los LSA son incluidos en un paquete LSU que es enviado directamente a R2 y R3. Por último, en el paso 3, ambos analizan el paquete recibido y agregan la información a sus LSDB. Además, reenvían el LSU a través de todas las interfaces pertenecientes a OSPF excepto por la cual fue recibido, en este caso, R2 lo reenviará a R3 y viceversa. Este hecho podría generar problemas, ya que existirían LSA duplicados dentro de la misma LSDB. Para evitarlo, cada router analiza la información recibida con la ya existente en su base de datos, para en caso de coincidencia, descartarla.

Una vez finalizada la convergencia, la adyacencia entre vecinos se mantiene mediante el envío periódico de mensajes *hello,* mientras que los LSU solo volverán a ser enviados cuando se produzca algún cambio en la topología.

INTERCAMBIO DE RUTAS EN ENTORNOS MULTIACCESO

En entornos multiacceso el proceso llevado a cabo por OSPF varía significativamente respecto a los enlaces punto a punto. En este caso, cada router tomará un rol, y dependiendo de este, actuará de una manera u otra en la topología. En este aspecto, OSPF selecciona un router denominado *DR* (*Designated Router*), cuya función consiste en llevar a cabo el intercambio de LSU entre todos los miembros de la topología. Para ello, cada dispositivo envía su información de enrutamiento solo a él, y este a su vez se encargará de reenviarla a los vecinos restantes. Como su función es crítica, también se selecciona un *BDR* (*Backup Designated Router*) el cual monitoriza al DR y en caso de caída toma inmediatamente su rol. El resto de routers, que actúan como *DRother*, simplemente se limitan al intercambio de LSA (en paquetes LSU) con el DR.

En este caso, el proceso de intercambio de rutas ejecutado por OSPF puede ser resumido en 5 pasos:

- *Paso 1:* Cada router inicia la búsqueda de vecinos mediante el envío de mensajes *hello* a través de las interfaces configuradas para ello.

- *Paso 2:* Entre todos los miembros de la topología OSPF, el protocolo selecciona uno que actuará como DR y otro como BDR.

- *Paso 3:* Cada router genera un LSA para cada una de sus redes directamente conectadas, los cuales incluyen toda la información necesaria sobre las mismas.

- *Paso 4:* Dichos LSA son agregados a uno o varios paquetes LSU y a su vez enviados tan solo al DR, que almacena la información recibida en su propia base de datos y acto seguido la reenvía a los vecinos restantes.

- *Paso 5:* Cada router, con los paquetes LSU recibidos, completa su LSDB.

En la siguiente topología el router "Central" actúa como DR. ¿Cómo se llevará a cabo el intercambio de rutas?

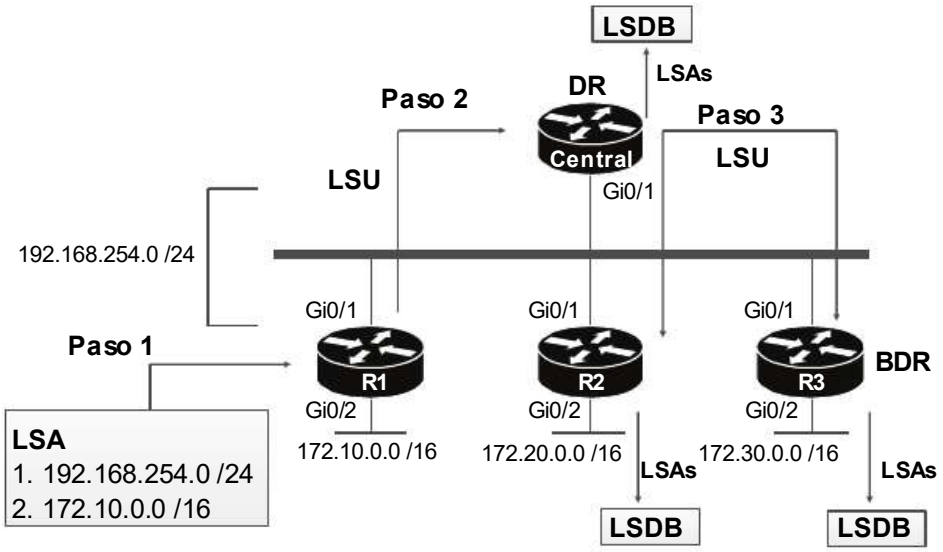


Fig. 6-11 OSPF. Intercambio de rutas en entornos multiacceso.

Al tratarse de un entorno multiacceso (bus), *link state* seleccionará un router que actuará como DR y otro como BDR, siendo en este caso "Central" el DR. Conforme a ello, todos los miembros llevaran a cabo el mismo procedimiento de intercambio de rutas. Para analizarlo, centremos el ejemplo en R1. En el paso 1 genera un LSA por cada ruta directamente conectada. En el paso 2, dichos LSA son incluidos en un paquete LSU que es enviado al DR. Por último, en el paso 3, el DR almacena los LSA recibidos en su propia LSDB y reenvía la información al resto de routers presentes en la topología OSPF, en este caso R2 y R3.

Al igual que sucede en los enlaces punto a punto, cuando un router recibe un LSA lo compara con los ya almacenados en su base de datos, y si existe, simplemente lo descarta.

Una vez finalizada la convergencia solo se enviarán paquetes LSU cuando se produzca algún cambio en la topología.

Proceso de elección del DR y BDR

En OSPF el router designado centraliza la distribución de rutas a lo largo de la topología, hecho por el cual es considerado un elemento crítico dentro de esta. Es por ello muy recomendable que cumpla los siguientes requisitos, primero, que disponga de alta disponibilidad y buena capacidad de memoria, y segundo, que su ubicación física sea la idónea para llevar a cabo sus funciones de manera eficiente. En ambos casos resulta imprescindible la intervención del administrador, pudiendo y debiendo influir sobre su elección.

El proceso llevado a cabo para determinar qué router tomará el rol de designado es el siguiente:

- El DR será aquel cuya prioridad de OSPF en la interfaz que conecta con la red multiacceso resulte la más alta de todos los routers que componen la topología.
- Si la prioridad no fuera configurada, el DR será aquel cuyo RID (*Router id*) resulte el más alto de todos los dispositivos que forman parte de la topología.

Si aun así dichos valores no fueran definidos manualmente por el administrador, el router seleccionará automáticamente un RID con relación a las direcciones IP configuradas tanto en interfaces loopback como físicas. Este proceso coincide con el llevado a cabo por EIGRP, ya analizado en este mismo capítulo. ¿Cómo conoce cada router la prioridad o RID de los restantes? Su valor es incluido en los paquetes *hello* durante el proceso de descubrimiento de vecinos.

Como se ha descrito con anterioridad, el intercambio de los LSA se lleva a cabo agregando estos en uno o varios paquetes LSU. Sin embargo, muchos libros y guías de estudio mencionan este proceso simplemente como intercambio de los LSA. Para afrontar el examen de CCNA se debe diferenciar entre LSA, LSU y LSDB.

CÁLCULO DE RUTAS

Una vez concluida la convergencia, todos los routers de la topología deben contener la misma información en sus LSDB. Conforme a ella, el protocolo aplica el algoritmo SPF (*Shortest Path First*) con el fin de calcular las posibles rutas y sus métricas hacia los diferentes destinos, para posteriormente hacer uso de aquella más baja hacia cada subred.

En OSPF, la métrica hace referencia al valor obtenido tras la suma de todos los costes notificados desde el router local hasta la red de destino, influyendo sobre ella factores como la velocidad, retraso o carga de cada enlace interviniente.

Un ejemplo podría ser el siguiente:

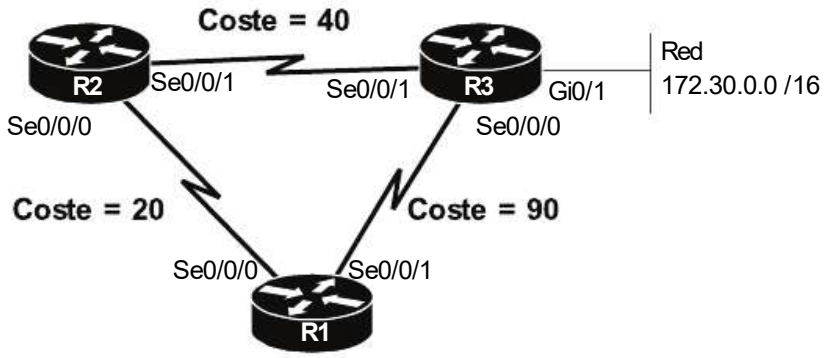


Fig. 6-12 Cálculo de rutas en OSPF.

Donde R1 aplica SPF sobre su LSDB e identifica dos posibles rutas hacia la red 172.30.0.0 /16, la R1-R3 y R1-R2-R3, pero ¿cuál de ellas instalará? Tras el cálculo de la métrica hará uso de aquella con menor valor, siendo en este caso la R1-R2-R3 (20+40=60).

Los costes definidos en el ejemplo son orientativos con el fin de facilitar la comprensión. OSPF aplica valores más elevados ya que el algoritmo ejecuta una ecuación matemática compleja, la cual no es contenido de CCNA.

Modo de operación

En este caso, el modo de operación hace referencia principalmente al descubrimiento de vecinos, distribución en áreas y tipos de LSA.

DESCUBRIMIENTO DE VECINOS

La primera acción que lleva a cabo un router configurado con OSPF consiste en enviar paquetes "*hello*" a la dirección multicast 224.0.0.5 a través de las interfaces configuradas para tal propósito, con el objetivo de establecer adyacencia con otros dispositivos. Estos mensajes incluyen datos como el RID, IP de la interfaz que lo envía, área y una serie de parámetros que deben coincidir en ambos extremos para que la relación concluya con éxito. Si ello ocurre comienza el intercambio de rutas, el

cual debe finalizar con la misma información de LSDB en ambos dispositivos, lo que se conoce como *full state.*

Esta transición, en detalle, consta de 5 estados:

1.- Init: Identifica el envío de mensajes *hello* en busca de posibles vecinos.

2.- 2-way: El dispositivo se encuentra en proceso de negociación con otro para establecer adyacencia.

3.- ExStart: Dicha adyacencia ha concluido con éxito y procede a intercambiar una breve descripción de sus LSDB.

4.- Loading: Tras ello, comienza el envío de paquetes LSU.

5.- Full state: Ambos routers disponen de la misma LSDB.

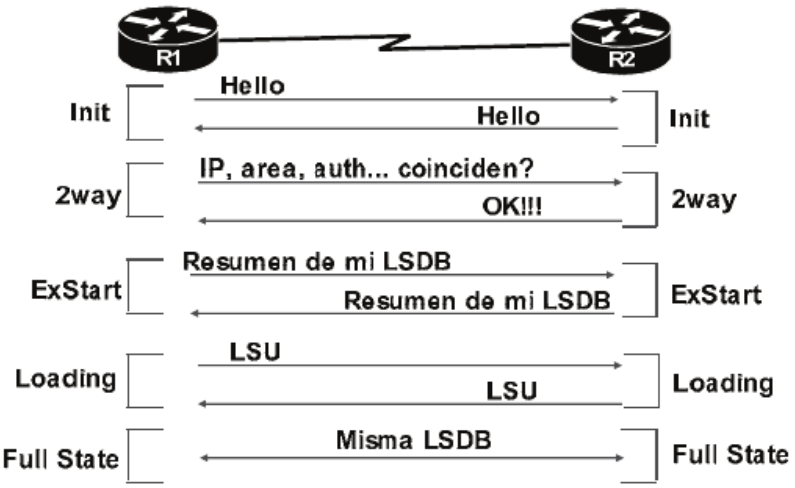


Fig. 6-13 Adyacencia entre vecinos y convergencia en OSPF.

Una vez concluido, ambos dispositivos mantienen la adyacencia mediante el envío periódico de mensajes *hello* de manera unicast o multicast dependiendo del medio utilizado. Además, al igual que ocurre en EIGRP, si un router no recibe mensajes de un determinado vecino durante un tiempo preestablecido, elimina automáticamente su adyacencia, debiendo efectuar un recálculo de rutas. Los temporizadores utilizados para tal propósito en OSPF son: "*Hello Interval*", que define el intervalo de tiempo para el envío de cada uno de estos mensajes, y "*Dead Interval",* que establece el tiempo máximo de espera antes de eliminar una adyacencia. Su valor

suele ser 4 veces superior al *Hello Interval* y es reseteado a cero cada vez que se recibe un mensaje del vecino en cuestión.

Por último, los LSA son reenviados cada 30 minutos por defecto, con la excepción de que se produzca algún cambio en la topología, en cuyo caso se enviarían de manera inmediata con el fin de notificar a los diferentes vecinos y evitar con ello posibles loops de enrutamiento.

DISTRIBUCIÓN EN ÁREAS

Otro de los aspectos más importantes del protocolo consiste en la distribución de la topología. En este sentido OSPF basa su modelo en áreas, con el fin de optimizar el rendimiento de la red y con ello la velocidad de enrutamiento e intercambio de información. El planteamiento llevado a cabo resulta sencillo, las interfaces deben pertenecer a una determinada área y solo establecerán adyacencia con vecinos que operen en la misma. Pero ¿qué ventajas aporta este modelo? Una de las características de OSPF es que la LSDB almacenada en cada router debe contener información de la topología completa, en relación con la cual se ejecuta el algoritmo SPF para obtener la métrica hacia cada destino. Bien, en redes muy grandes, si todos los routers pertenecieran al mismo área sus LSDB contendrían demasiada información, por lo que el algoritmo tardaría más tiempo en ejecutar los cálculos, afectando a su vez a la velocidad de enrutamiento. Además, en este tipo de entornos la probabilidad de que se produzca algún cambio aumenta, en cuyo caso se generaría un LSA que sería enviado inmediatamente a todos los vecinos, debiendo ejecutar el algoritmo nuevamente.

Lo ideal para evitar dicha situación consiste en dividir la topología, gracias a lo cual los routers tan solo establecerán adyacencia con aquellos pertenecientes a su misma área, logrando así que las LSDB sean de menor tamaño y obteniendo como beneficio un cálculo mucho más rápido del algoritmo SPF. Una vez hecho, la comunicación entre las diferentes áreas se llevará a cabo a través de uno o varios routers situados estratégicamente.

Los roles que puede tomar un dispositivo en este caso son:

- *Backbone Router (BR):* En OSPF, el *"backbone area"* hace referencia a aquella encargada de establecer la comunicación y enrutamiento entre diferentes áreas. Los routers que la integran, denominados BR, deben ser de alto rendimiento, con buena capacidad de memoria, CPU y redundancia, ya que centralizan gran parte del enrutamiento.

- *Router ABR (Area Border Router):* Son routers intermediarios cuyas interfaces pertenecen a diferentes áreas, entre ellas la *backbone*. Los ABR intercambian información de enrutamiento con los BR, gracias a lo cual se hace posible la comunicación entre diferentes áreas.

- *Internal Router:* Son aquellos cuyas interfaces solo pertenecen a una determinada área, siempre y cuando no sea la *backbone*.

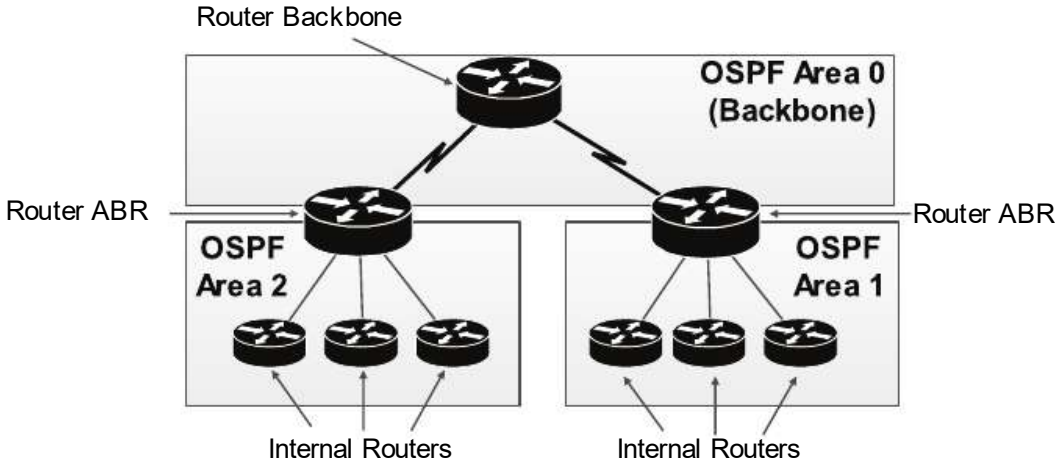


Fig. 6-14 Diseño OSPF distribuido en áreas.

Por último, OSPF puede ser configurado únicamente en un área y operaría sin problemas, este modelo es denominado *single area* y puede resultar útil en topologías pequeñas. En redes de gran tamaño no es recomendable su aplicación debido a los problemas ya mencionados en párrafos anteriores.

TIPOS DE LSA

A excepción de los mensajes *hello*, OSPF basa la comunicación entre vecinos en el intercambio de LSA. Estos, dependiendo del propósito o entorno pueden contener diferente información, por lo que el protocolo los divide en 11 tipos *(Type LSA),* de los cuales 3 son contenido de CCNA.

- *LSA Type 1 (Router LSA):* Son aquellos intercambiados por routers del mismo área y contienen información sobre el dispositivo que lo genera, como su RID, IP, máscara o interfaces.

- *LSA Type 2 (Network LSA):* Son los LSA generados y enviados por el router DR en entornos multiacceso para notificar a los demás dispositivos de su misma área de información como su propia dirección IP y la del BDR, ID de subred, máscara, etc.

- *LSA Type 3 (Summary LSA):* Son generados por los routers ABR para el intercambio de rutas entre diferentes áreas. Estos contienen información sumarizada (mediante un *auto-summary*) de los id y prefijos de cada una de ellas.

OSPF - CONFIGURACIÓN Y VERIFICACIÓN EN REDES IPV4

Concluido el estudio de su algoritmo y modo de operación, tan solo bastaría proceder a la configuración del protocolo y su verificación.

Aunque comúnmente sea denominado como OSPF, la versión del protocolo para redes IPv4 es OSPFv2 y su configuración se lleva a cabo aplicando el siguiente procedimiento:

- *Paso 1:* Crear una instancia del protocolo con el comando **router ospf *[número de proceso]*** desde el modo de configuración global. A diferencia de EIGRP, este valor tan solo tiene importancia a nivel local, es decir, diferentes routers pueden ser configurados con distintos procesos y podrán formar adyacencia sin problema. Ello es debido a que las relaciones entre vecinos se basan en el número de área a la que pertenecen. Al ejecutar el comando se accede al modo de configuración de OSPF, desde el cual se deberán ejecutar el resto de acciones.

- *Paso 2:* Definir las interfaces y redes que se publicarán, aplicando para ello el comando **network *[red] [wildcard] [área]***. Gracias al ID de red indicado, OSPF determina qué interfaz hace uso de una IP perteneciente a su rango, y a través de la misma enviará mensajes *hello* en busca de posibles vecinos. La *Wildcard,* al igual que ocurre en EIGRP, hace referencia al valor inverso de la máscara de red. Por último, *área* indica el número de esta a la que pertenecerá la interfaz. Uno de los requisitos para que dos routers establezcan adyacencia es que coincidan en este parámetro.

- *Paso 3 (Opcional):* Definir qué interfaces actuarán como pasivas, haciendo uso para ello de la sentencia **passive-interface *[interfaz]***. A través de estas no se establecerá adyacencia con ningún otro dispositivo ni se publicarán rutas. Normalmente identifican aquellas destinadas para la conexión con dispositivos finales.

- *Paso 4 (Opcional):* Establecer un RID mediante el comando **router-id [*x.x.x.x*]**, donde *x.x.x.x* indica el valor, en decimal, que se asignará, haciendo uso del mismo formato que una dirección IPv4. Si no fuera configurado se seleccionará uno automáticamente conforme al criterio ya analizado en párrafos anteriores. Aunque su configuración sea opcional resulta altamente recomendable llevarla a cabo ya que

la comunicación entre vecinos y la información listada al ejecutar los comandos show muestran el RID de los diferentes dispositivos. (en EIGRP se muestra la IP).

- *Paso 5 (Opcional):* Definir el número máximo de rutas que OSPF podrá utilizar de manera simultánea para realizar balanceo de carga, ejecutando el comando **maximum-paths [*x*]**. De lo contrario se aplicará el valor por defecto, siendo igual a 4.

Configurar OSPFv2 en la siguiente topología de tal manera que:

- Todas las interfaces pertenezcan al área 0 (*backbone*).
- Los routers apliquen los siguientes RID: R1 (1.1.1.1), R2 (2.2.2.2) y R3 (3.3.3.3).
- Configurar como interfaces pasivas aquellas que se consideren oportunas.

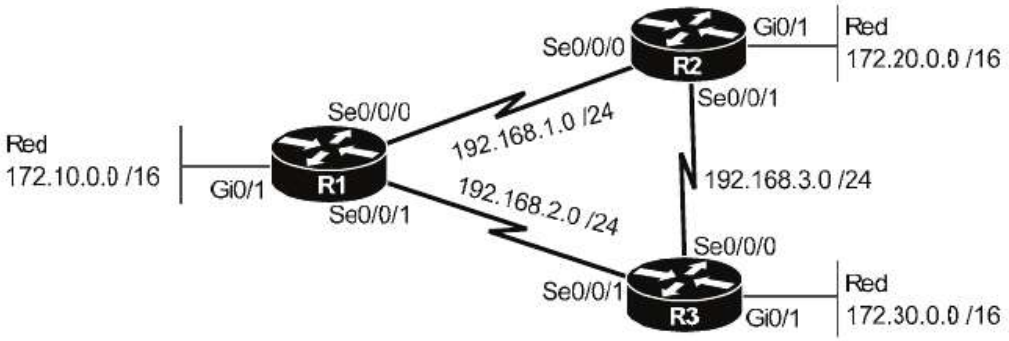


Fig. 6-15 Diseño de red para supuesto práctico de OSPF.

Como siempre, antes de proceder a su configuración deben ser documentados los cambios a realizar, para posteriormente aplicarlos.

Router	Proceso OSPF	RID	Red	Wildcard	Área	Interfaces Pasivas
R1	5	1.1.1.1	172.10.0.0 192.168.1.0 192.168.2.0	0.0.255.255 0.0.0.255 0.0.0.255	0 0 0	Gi0/1
R2	10	2.2.2.2	172.20.0.0 192.168.1.0 192.168.3.0	0.0.255.255 0.0.0.255 0.0.0.255	0 0 0	Gi0/1
R3	15	3.3.3.3	172.30.0.0 192.168.2.0 192.168.3.0	0.0.255.255 0.0.0.255 0.0.0.255	0 0 0	Gi0/1

```
---Configuración en R1---
R1(config)#router ospf 5
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 172.10.0.0 0.0.255.255 area 0
R1(config-router)#network 192.168.1.0 0.0.0.255 area 0
R1(config-router)#network 192.168.2.0 0.0.0.255 area 0
R1(config-router)#passive-interface Gi0/1

---Configuración en R2---
R2(config)#router ospf 10
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 172.20.0.0 0.0.255.255 area 0
R2(config-router)#network 192.168.1.0 0.0.0.255 area 0
R2(config-router)#network 192.168.3.0 0.0.0.255 area 0
R2(config-router)#passive-interface Gi0/1

---Configuración en R3---
R3(config)#router ospf 15
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 172.30.0.0 0.0.255.255 area 0
R3(config-router)#network 192.168.2.0 0.0.0.255 area 0
R3(config-router)#network 192.168.3.0 0.0.0.255 area 0
R3(config-router)#passive-interface Gi0/1
```

Verificación de OSPF en redes IPv4

Concluida su configuración resulta imprescindible verificar que los cambios se han aplicado correctamente y la red opera según lo previsto. Para ello, y con el fin de facilitar dicha tarea, IOS dispone de los siguientes comandos:

- *show ip ospf:* Muestra información sobre cada una de las instancias del protocolo configuradas en el router, facilitando datos como su número de proceso, RID, áreas, interfaces, autenticación o contadores LSA.

Aplicado sobre R1:

```
R1#show ip ospf
 Routing Process "ospf 5" with ID 1.1.1.1
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 Number of external LSA 0. Checksum Sum 0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 External flood list length 0
    Area BACKBONE(0)
        Number of interfaces in this area is 3
        Area has no authentication
        SPF algorithm executed 3 times
        Area ranges are
        Number of LSA 3. Checksum Sum 0x01fc83
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
```

```
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

- *show ip ospf neighbor:* Muestra información en pantalla sobre vecinos OSPF con los cuales se ha establecido y mantiene adyacencia. Estos son identificados por su RID, de ahí la importancia de su configuración, aunque resulte opcional. Además, facilita datos como sus direcciones IP y la interfaz local necesaria para establecer la conexión con cada uno de ellos.

```
R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
2.2.2.2           0   FULL/  -        00:00:39    192.168.1.2     Serial0/0/0
3.3.3.3           0   FULL/  -        00:00:31    192.168.2.2     Serial0/0/1
```

- *show ip ospf database:* Genera información sobre los routers de los cuales se ha recibido algún LSA. Si la convergencia ha concluido con éxito, deberán listarse los RID de todos los dispositivos pertenecientes a la misma área de OSPF.

```
R1#show ip ospf database
            OSPF Router with ID (1.1.1.1) (Process ID 5)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         158         0x80000007 0x00ccc3 5
3.3.3.3         3.3.3.3         158         0x80000007 0x00303e 5
2.2.2.2         2.2.2.2         59          0x80000009 0x00f587 5
```

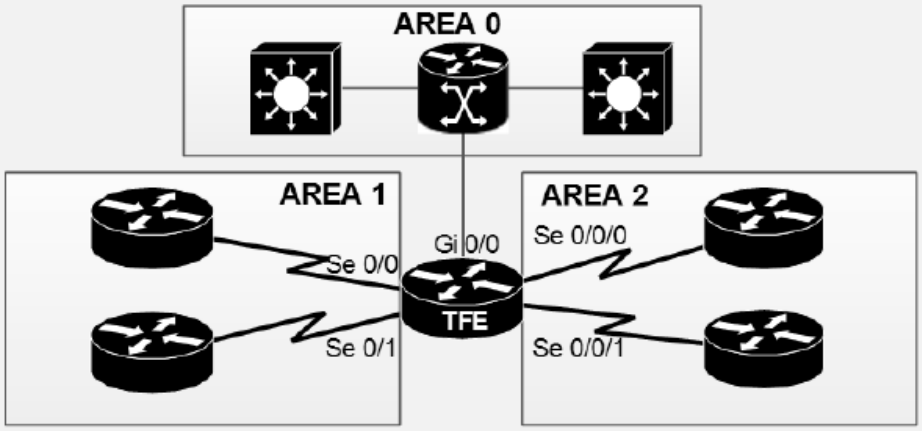
Otros comandos útiles son:

- *show ip ospf interface [interfaz]:* Muestra información referente a OSPF sobre una determinada interfaz. Entre los datos obtenidos se incluye su IP, área a la que pertenece, proceso de OSPF, RID, tipo de enlace, coste, valores *hello*, *dead* y número de adyacencias con vecinos a través de la misma.

- *show ip route ospf:* Muestra en pantalla las entradas incluidas en la tabla de rutas con procedencia de OSPF. Estas son representadas mediante el código *"O",* facilitando datos como la dirección de la red remota, métrica, distancia administrativa y siguiente salto.

Reto 6.2 - Dada la siguiente topología, configurar OSPF en el router TFE conforme a la siguiente documentación:

Router	Proceso OSPF	RID	Red - Interfaz
TFE	25	25.25.25.25	172.10.0.0/16 - Se0/0 172.20.0.0/16 - Se0/1 192.168.10.0/24 – Se0/0/0 192.168.20.0/24 – Se0/0/1 10.0.0.0/8 – Gi0/0



Solución al final del capítulo.

RIP- ROUTING INFORMATION PROTOCOL

A día de hoy, EIGRP y OSPF representan los protocolos de enrutamiento IGP más habituales, debido en gran medida a la velocidad de convergencia y fiabilidad ofrecida por sus sistemas de métrica. Dichas características nacen como evolución a sus antecesores, más concretamente, RIP, RIPv2 e IGRP. La historia comienza en 1980, con la aparición de RIP *(Routing Information Protocol)*, siendo continuado por IGRP (propietario de Cisco) y posteriormente, la versión 2 del primero, RIPv2. Este, junto a EIGRP y OSPF, constan como los últimos protocolos de enrutamiento IPv4 en ser desarrollados, por la década de los 90.

Debido a su antigüedad, RIP ha caído en desuso, hasta tal punto de ser considerado obsoleto. Sin embargo, muchas de sus características aún son aplicadas sobre los protocolos más actuales, como el envenenamiento de ruta o el horizonte dividido, ya analizados en este mismo capítulo.

La presente sección abordará el estudio y características más destacadas tanto de RIPv1 como de RIPv2, llevando a cabo una comparación entre ambos, para posteriormente proceder a la configuración de la versión 2 del protocolo.

IGRP no forma parte del contenido de CCNA.

Comparación entre RIPv1 y RIPv2

Como se ha mencionado anteriormente, RIPv1 consta como el primer protocolo de enrutamiento IGP en ser desarrollado, por lo que resulta evidente que con el paso del tiempo requirió ciertas mejoras, implementadas en la versión 2. La comparación entre ambas se llevará a cabo mediante el análisis de sus características comunes, para, a posteriori, estudiar las mejoras que agrega RIPv2 sobre su antecesor.

Tanto RIPv1 como RIPv2 basan su sistema de métrica en el conteo de saltos desde el origen hasta el destino, entendiéndose como un salto cada vez que un paquete atraviesa un dispositivo de capa 3, de tal manera que, si entre A y B existieran 4 routers, la métrica aplicada sobre su ruta es igual a 4. Además, se establece un máximo de 15 saltos, descartando cualquier ruta que requiera una métrica mayor. ¿Por qué resulta necesaria dicha medida? Debido a que, si no fuera aplicada, la convergencia sería más lenta, pudiendo generar bucles de capa 3.

Otro factor en el que coinciden es en la manera de notificar a routers vecinos ante cualquier cambio acaecido en la red. En este caso, ambas versiones envían *full updates* cada 30 segundos, es decir, la tabla de rutas completa. Este hecho tiene como consecuencia una convergencia más lenta y una mayor utilización del ancho de banda disponible.

Por último, también aplican envenenamiento de ruta y horizonte dividido. El primero consiste en asignar una métrica infinita sobre aquellas rutas que el dispositivo ha detectado como caídas, evitando de esta manera el envío de paquetes a través de enlaces no operativos y a su vez posibles bucles de enrutamiento, forzando al dispositivo a hacer uso de alguna ruta alternativa (si existiera) hacia el destino en cuestión. En este caso, la métrica infinita obtiene un valor igual a 16. Mientras, el horizonte dividido consiste en no reenviar una ruta a través de la interfaz por la cual fue recibida, evitando gracias a ello loops de capa 3.

Las características recién analizadas coinciden en ambas versiones, sin embargo, RIPv2 implementa ciertas mejoras con el fin de optimizar el rendimiento y funcionalidades del protocolo. Estas son:

- *VLSM:* RIPv1 no envía la máscara de red en sus actualizaciones de enrutamiento, aplicando aquella por defecto para cada clase. Por su contra, RIPv2 sí que lo hace, habilitando la utilización de subredes de máscara variable.

- *Autenticación:* RIPv2 agrega una capa de seguridad al permitir la autenticación entre routers vecinos (opcional).

- *Sumarización:* También aplica la sumarización de rutas, gracias a lo cual resulta posible reducir el número de entradas en la tabla y con ello actualizaciones de enrutamiento de menor tamaño. Esta característica es ejecutada por defecto, sin embargo, en entornos compuestos por redes no contiguas es necesario deshabilitarla.

- *Dirección multicast 224.0.0.9:* Los mensajes *full update* son enviados a la dirección multicast 224.0.0.9, la cual identifica al grupo de routers que operan con RIPv2. Con ello se logra que tan solo sean recibidos por los destinatarios correctos. Esta misma operación, en la versión 1, se lleva a cabo mediante un broadcast, lo que implica mayor utilización del ancho de banda disponible y menor seguridad al ser recibidos por todos los dispositivos de la red.

En resumen:

Característica	RIPv1	RIPv2
Envía la máscara de red en sus actualizaciones de enrutamiento (VLSM)	No	Sí
Aplica sumarización	No	Sí
Permite autenticación	No	Sí
Actualizaciones de enrutamiento a la dirección multicast 224.0.0.9	No	Sí
Envenenamiento de ruta	Sí	Sí
Horizonte dividido	Sí	Sí
Actualizaciones de enrutamiento completas *(full updates)*	Sí	Sí
Métrica basada en conteo de saltos (vector distancia)	Sí	Sí

Como se puede observar, RIPv2 dispone prácticamente de las mismas características que OSPF y EIGRP, entonces, ¿por qué su aplicación resulta menos común? Simplemente por su sistema de métrica y actualizaciones de enrutamiento

completas. Estas dos características se traducen en un cálculo de rutas poco eficiente y convergencia lenta. Ambas deficiencias son resueltas tanto en OSPF como EIGRP, los cuales basan su métrica en diferentes factores como el ancho de banda o retraso de cada enlace y a su vez hacen uso de actualizaciones de enrutamiento parciales.

Configuración y verificación de RIPv2

El proceso de configuración de RIPv2 en routers Cisco consta de los siguientes pasos:

Paso 1: Acceder al modo de configuración del protocolo a través del comando **router rip**, desde el cual se deberán ejecutar el resto de acciones.

Paso 2: Habilitar la versión 2 mediante la sentencia **version 2.**

Paso 3: Especificar las redes directamente conectadas a publicar, con el comando **network [*id de red*].**

Paso 4 (Opcional): Definir las interfaces pasivas, a través de las cuales no se enviarán actualizaciones de enrutamiento ni se establecerán relaciones entre vecinos. Para ello, ejecutar el comando **passive-interface *[interfaz]***.

Paso 5 (Opcional): Deshabilitar la sumarización automática, la cual es aplicada por defecto. Dicha acción se lleva a cabo a través de la sentencia **no auto-summary.**

Paso 6 (Opcional): Si el dispositivo dispone de una ruta por defecto, puede ser publicada por RIPv2 aplicando la sentencia **default-information originate**.

La autenticación en RIPv2 consta de diferentes parámetros para crear la clave y su posterior habilitación. Estos no forman parte del contenido de CCNA.

Ejemplo: Configurar RIPv2 en los routers TFE, Central y EXT, cumpliendo los siguientes requisitos:

- Definir como interfaces pasivas aquellas que se consideren oportunas.
- Propagar la ruta por defecto, ubicada en EXT.
- Deshabilitar la sumarización automática en todos los routers.

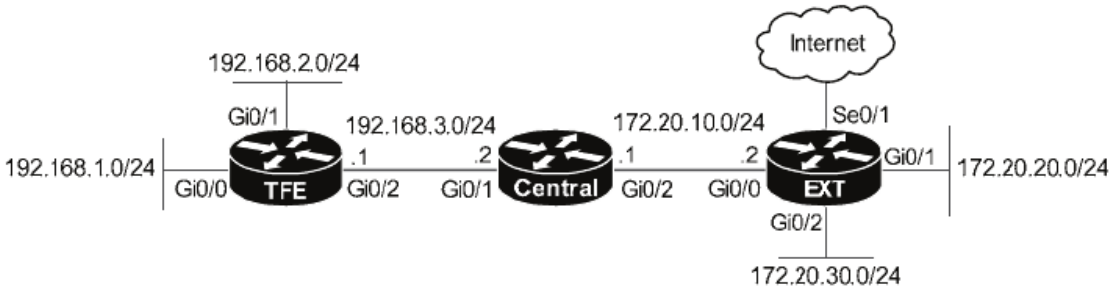


Fig. 6-16 Diseño de red para supuesto práctico de RIPv2.

```
---Configuración en TFE---
TFE(config)#router rip
TFE(config-router)#version 2
TFE(config-router)#network 192.168.1.0
TFE(config-router)#network 192.168.2.0
TFE(config-router)#network 192.168.3.0
TFE(config-router)#no auto-summary
TFE(config-router)#passive-interface Gi0/0
TFE(config-router)#passive-interface Gi0/1

---Configuración en Central---
Central(config)#router rip
Central(config-router)#version 2
Central(config-router)#network 192.168.3.0
Central(config-router)#network 172.20.10.0
Central(config-router)#no auto-summary

---Configuración en EXT---
EXT(config)#router rip
EXT(config-router)#version 2
EXT(config-router)#network 172.20.10.0
EXT(config-router)#network 172.20.20.0
EXT(config-router)#network 172.20.30.0
EXT(config-router)#no auto-summary
EXT(config-router)#default information-originate
EXT(config-router)#passive-interface Gi0/1
EXT(config-router)#passive-interface Gi0/2
EXT(config-router)#passive-interface Se0/1
```

Gracias a los cambios aplicados, y una vez finalizada la convergencia, todos los routers dispondrán de acceso a las diferentes redes remotas existentes, las cuales podrán ser verificadas ejecutando un **show ip route**. Por ejemplo, en TFE.

```
TFE#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
```

```
Gateway of last resort is 192.168.3.2 to network 0.0.0.0

     172.20.0.0/24 is subnetted, 3 subnets
R       172.20.10.0 [120/1] via 192.168.3.2, 00:00:03, GigabitEthernet0/2
R       172.20.20.0 [120/2] via 192.168.3.2, 00:00:03, GigabitEthernet0/2
R       172.20.30.0 [120/2] via 192.168.3.2, 00:00:03, GigabitEthernet0/2
C    192.168.3.0/24 is directly connected, GigabitEthernet0/2
C    192.168.2.0/24 is directly connected, GigabitEthernet0/1
C    192.168.1.0/24 is directly connected, GigabitEthernet0/0
R*   0.0.0.0/0 [120/2] via 192.168.3.2, 00:00:03, GigabitEthernet0/2
```

Donde aquellas aprendidas por el protocolo son representadas mediante el código *R*, facilitando información bastante útil como su distancia administrativa, métrica o siguiente salto. Un análisis de ruta podría ser el siguiente:

```
R       172.20.30.0 [120/2] via 192.168.3.2, 00:00:03, GigabitEthernet0/2
```

- *R*: Ruta aprendida por el protocolo RIP. En este caso, en su versión 2.

- *172.20.30.0:* Corresponde al id de la red remota.

- *[120/2]:* Hace referencia a la distancia administrativa de la ruta y su métrica. En este caso, al ser aprendida por RIP, su AD obtiene un valor de 120, mientras que la métrica es igual a 2 porque existen dos saltos de diferencia entre TFE y la red 172.20.30.0.

- *vía 192.168.3.2:* Indica el siguiente salto necesario para llegar a la red remota en cuestión. En este caso, la IP 192.168.3.2 hace referencia al router Central, por lo tanto, cualquier paquete con destino 172.20.30.0 será reenviado al mismo.

- *00:00:03:* Es el tiempo transcurrido, en segundos, desde que se recibió la última actualización de enrutamiento para esta ruta.

- *GigabitEthernet0/2:* Identifica la interfaz local necesaria por la cual deben ser reenviados los paquetes con destino 172.20.30.0/24.

Por lo tanto, si se realizara un ping desde TFE hacia cualquier host ubicado en dicha red se debería obtener respuesta, por ejemplo, a la interfaz Gi0/2 del router EXT, con IP 172.20.30.1.

```
TFE#ping 172.20.30.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.20.30.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/1 ms
```

Por último, IOS también dispone del comando **show ip rip database**, el cual muestra en pantalla todas las rutas aprendidas por el protocolo, facilitando información sobre cada una de ellas como el siguiente salto o el tiempo transcurrido desde su última actualización de enrutamiento.

BGP - BORDER GATEWAY PROTOCOL

Una característica común tanto en EIGRP, como OSPF y RIPv2, es que todos ellos han sido desarrollados para su aplicación sobre entornos privados administrados por una única compañía, es decir, constan como protocolos IGP. Sin embargo, las redes WAN públicas también requieren el intercambio de rutas entre los dispositivos de capa 3 ubicados en la misma, gracias a lo cual se hace posible la comunicación entre los diferentes ISP y con ello el enrutamiento a nivel global. Con dicho propósito fue desarrollado BGP *(Border Gateway Protocol)*, cuyas características y configuración serán objeto de estudio a lo largo de la presente sección.

Modo de operación

BGP consta como el único protocolo EGP *(Exterior Gateway Protocol)* implementado actualmente como método de intercambio de rutas entre los ISP ubicados en las diferentes regiones, países o continentes. Su finalidad, al igual que los IGP, consiste en aprender rutas de manera dinámica y automática, seleccionar la mejor de ellas hacia cada destino y finalizar la convergencia libre de bucles. Para lograrlo, basa su modo de operar en la siguiente lógica:

- Cada ISP dispone de un rango de direcciones IPv4 públicas, asignado por IANA, pero gestionado por él mismo para dar servicio a sus clientes.
- Cada ISP publica su rango al resto de ISPs, de tal manera que cada uno de ellos dispone de la información necesaria para llevar a cabo el enrutamiento a nivel global a través de la red pública.
- Cuando un ISP requiere enviar un paquete a una dirección no perteneciente a su mismo rango, consulta su tabla BGP y lo reenvía al siguiente salto necesario.

Aunque su concepto pueda resultar sencillo, BGP aplica algoritmos complejos para determinar la mejor ruta hacia cada destino, además, hace uso de diferentes terminologías y mensajes de actualización, los cuales son abordados de manera muy superficial en el CCNA. Su estudio en profundidad es contemplado en certificaciones más avanzadas como el CCNP.

INTERCAMBIO DE RUTAS

BGP hace uso de una topología física dividida en multitud de ASNs *(Autonomous System Number)*, donde cada uno de ellos simplemente hace referencia a un valor numérico y único a nivel global otorgado a cada compañía que participa en el intercambio de rutas públicas, siendo en la mayor parte de las ocasiones ISPs. Un ejemplo muy básico podría ser el siguiente:

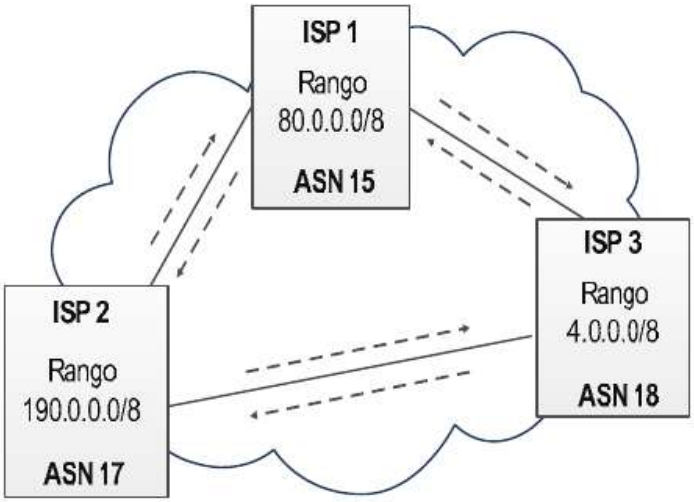


Fig. 6-17 ASNs en BGP.

Donde cada ISP dispone de un rango IPv4 y a su vez es identificado mediante un ASN. Todos ellos compartirán su información de enrutamiento y al finalizar la convergencia dispondrán de acceso hacia las diferentes redes remotas.

Este proceso es llevado a cabo mediante actualizaciones de enrutamiento *(BGP update messages)* compuestas por el id, con su correspondiente máscara, de cada una de las redes a publicar. Además, todas ellas incluyen una serie de valores, denominados *path attributes*, utilizados por el destino para determinar las mejores rutas. Dichas actualizaciones varían dependiendo del entorno, diferenciando entre *iBGP* y *eBGP*:

- *iBGP (Internal BGP)* hace referencia a la comunicación llevada a cabo de manera interna entre los dispositivos pertenecientes a un mismo ASN. Por ejemplo, un ISP con routers en diferentes regiones de un mismo país.

- *eBGP (External BGP)* identifica el intercambio de rutas entre diferentes ASN.

Por lo tanto:

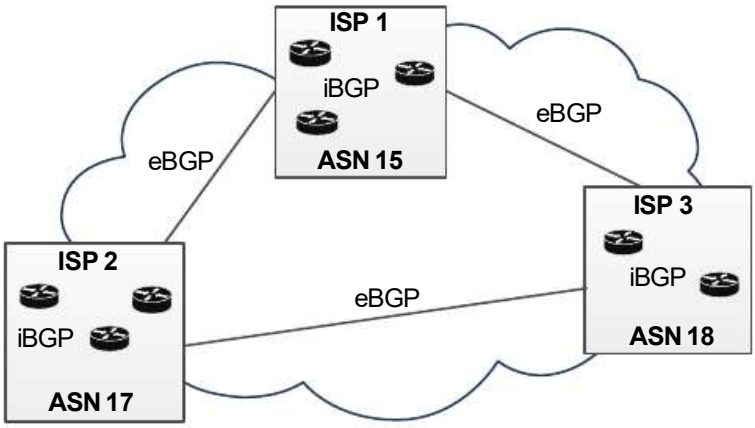


Fig. 6-18 Entornos iBGP comunicados mediante eBGP.

Con el fin de facilitar la comprensión, el estudio se centrará en el ISP1. Imagina que este opera a lo largo del territorio español y ha dividido el rango 80.0.0.0/8 en multitud de subredes, repartidas a lo largo de las diferentes comunidades. Todos los routers de su propiedad compartirán rutas de manera interna *(iBGP),* excepto aquellos que conectan con otros ISP, que lo harán de manera externa *(eBGP)* y además, mediante ruta sumarizada.

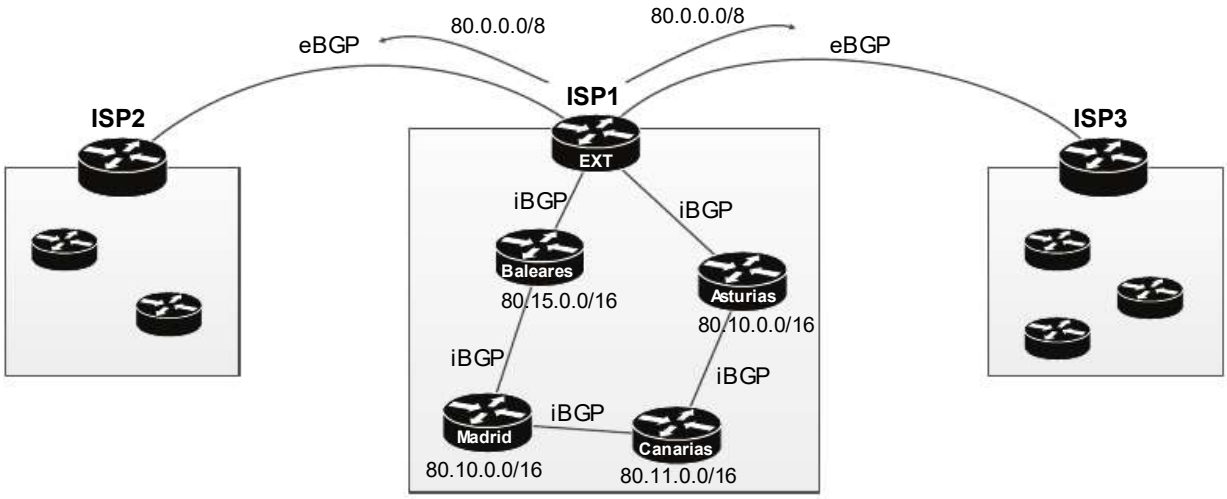


Fig. 6-19 Publicación de rutas sumarizadas en eBGP.

ISP1 divide el rango asignado por IANA, repartiéndolo entre las diferentes regiones y creando un entorno *iBGP* entre todos sus dispositivos. Sin embargo, de

manera externa *(eBGP)* tan solo publica la sumarización de todas estas subredes, en el prefijo 80.0.0.0/8.

En BGP se hace uso del término *prefijo, bloque de direcciones* o *NLRI (Network Layer Reachability Information)* para referirse a las subredes publicadas.

El ejemplo recién analizado y los datos en él expuestos no corresponden con la realidad, optando por ello con el fin de facilitar la compresión. Los entornos BGP resultan infinitamente más complejos y requieren mayor conocimiento del protocolo.

Como se ha mencionado en párrafos anteriores, sobre cada ruta se incluyen una serie de valores denominados *path attributes.* Uno de ellos es el "*AS_Path",* cuya función coincide con la métrica en los protocolos IGP, de hecho, el término *"métrica"* no existe en BGP. Cuando se reciben diferentes rutas hacia un mismo destino, el protocolo ejecuta un proceso denominado *best path selection*, el cual determina la mejor de ellas en base al *AS_Path* incluido en las mismas.

Configuración básica de eBGP

BGP dispone de numerosas opciones y parámetros a implementar dependiendo de factores como la topología, router, tipo de vecinos o rangos a publicar, pudiendo convertirse en un proceso realmente complejo. CCNA trata este protocolo de manera muy superficial, por lo que la configuración a continuación detallada resulta la más básica posible. Esta tendrá como escenario un router corporativo que conecta directamente con el ISP, con el cual intercambiará información de enrutamiento haciendo uso de eBGP.

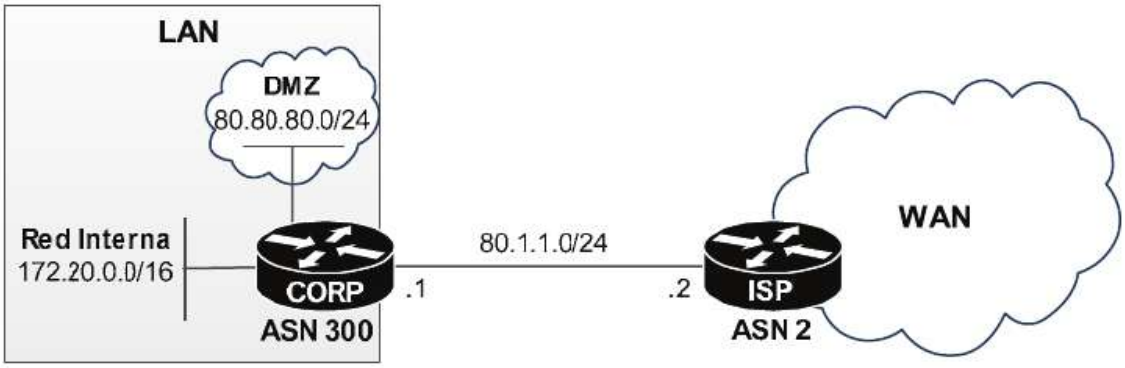


Fig. 6-20 Diseño de red para supuesto práctico de BGP.

El objetivo consiste en lograr que CORP e ISP establezcan relación de vecinos con el fin de que el primero notifique al segundo el rango de direcciones públicas que dispone en la DMZ. Para ello, se deben llevar a cabo las siguientes acciones.

- *Paso 1:* Acceder al modo de configuración del protocolo con el comando **router bgp *[asn]***.

- *Paso 2:* Definir cada uno de los vecinos de manera estática, ejecutando para ello la sentencia **neighbor *[ip vecino]* remote-as *[asn vecino]***. Esta representa una gran diferencia respecto a los protocolos IGP, donde el descubrimiento se lleva a cabo de manera dinámica.

- *Paso 3:* Agregar los rangos que se desean publicar a la tabla de rutas BGP, mediante el comando **network *[id red]* mask *[máscara]***.

De dicho proceso resulta necesario detallar los pasos 2 y 3. En BGP, los vecinos deben ser definidos de manera estática mediante el comando recién descrito. Una vez aplicado en ambos extremos, comienza una fase de negociación a través del puerto TCP 179, la cual concluirá, o no, con la adyacencia. Si finaliza de manera satisfactoria, el estado del protocolo se torna "e*stablished*" y comienza el intercambio de rutas. Para ello, previamente se lleva a cabo el siguiente proceso de transición (exceptuando el estado *"idle"*):

Estado	**Motivo**
Idle	La relación con el vecino en cuestión ha sido deshabilitada mediante el comando neighbor *[ip vecino]* shutdown.
Connect	La conexión TCP se ha establecido, pero no ha finalizado.
Active	La conexión TCP ha finalizado, pero aún no se ha enviado ningún mensaje BGP.
Opensent	La conexión TCP ha finalizado y se ha enviado el primer mensaje BGP para establecer adyacencia con el vecino.
Openconfirm	Se ha enviado el primer mensaje BGP para establecer adyacencia, pero aún no se ha recibido respuesta por parte del vecino.
Established	Se ha establecido adyacencia entre ambos routers y comienza el intercambio de información.

Otra gran diferencia entre BGP y los protocolos IGP consiste en las funciones desarrolladas por el comando *network*:

- En los IGP identifica la red local que se desea publicar, la cual debe pertenecer a alguna de las interfaces del dispositivo. Además, si dicha interfaz no fuera configurada como pasiva, participará activamente en la búsqueda de vecinos y publicación de rutas.

- Mientras, BGP simplemente se limita a compartir su tabla de rutas, por lo tanto, cualquier rango a publicar primero debe ser agregado a dicha tabla. Ello es posible gracias a diferentes métodos, siendo uno de ellos, el comando *network*.

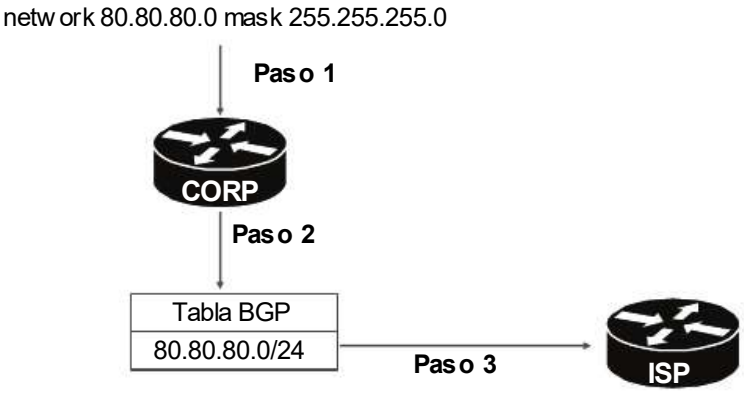


Fig. 6-21 Publicación de tabla de rutas en BGP.

- *Paso 1:* Se ejecuta el comando "*network 80.80.80.0 mask 255.255.255.0*" en la configuración BGP del router CORP.

- *Paso 2:* Dicho rango es agregado a la tabla de rutas.

- *Paso 3:* CORP comparte con ISP la información incluida en su tabla de rutas BGP.

Con ello, las acciones a llevar a cabo en ambos routers son las siguientes:

```
---Configuración en CORP---
CORP(config)#router bgp 300
CORP(config-router)#neighbor 80.1.1.2 remote-as 2
CORP(config-router)#network 80.80.80.0 mask 255.255.255.0


---Configuración en ISP---
ISP(config)#router bgp 2
ISP(config-router)#neighbor 80.1.1.1 remote-as 300
```

Una vez concluido el proceso, IOS dispone los siguientes comandos de verificación:

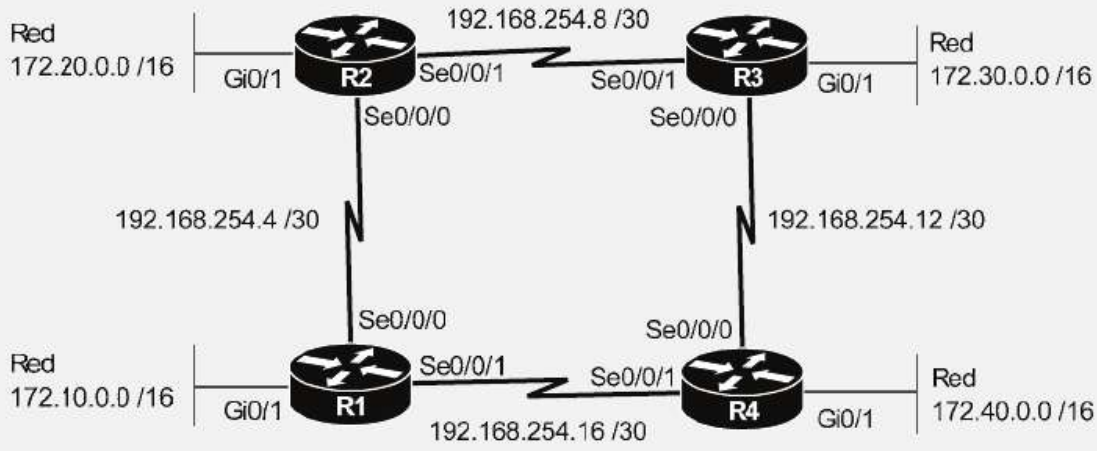
- *show ip bgp summary:* Muestra en pantalla un listado de vecinos BGP, facilitando información como su ASN, IP, o estado de la conexión TCP, entre otros.

- *show ip bgp:* Muestra la tabla de rutas BGP, donde aquellas representadas con el código "*>" identifican la mejor opción hacia el destino en cuestión.

SOLUCIÓN DE RETOS: PROTOCOLOS DE ENRUTAMIENTO

Reto 6.1 – Configurar EIGRP como protocolo de enrutamiento en la siguiente topología de tal manera que:

- Todos los routers formen parte del ASN 15.
- Configurar como interfaces pasivas aquellas que se consideren oportunas.
- Se debe habilitar el balanceo de carga para un máximo de 5 rutas.
- El RID debe ser seleccionado automáticamente por cada router. ¿Qué valor se asignará en cada uno de ellos?
- Los intervalos *hello* y *hold* deben tomar los valores por defecto de EIGRP.
- El ancho de banda y el retraso de cada enlace no deben ser modificados.




```
---Configuración en R1---
R1(config)#router eigrp 15
R1(config-router)#network 192.168.254.16 0.0.0.3
R1(config-router)#network 192.168.254.4 0.0.0.3
R1(config-router)#network 172.10.0.0 0.0.255.255
```

```
R1(config-router)#passive-interface Gi0/1
R1(config-router)#maximum-paths 5

---Configuración en R2---
R2(config)#router eigrp 15
R2(config-router)#network 192.168.254.8 0.0.0.3
R2(config-router)#network 192.168.254.4 0.0.0.3
R2(config-router)#network 172.20.0.0 0.0.255.255
R2(config-router)#passive-interface Gi0/1
R2(config-router)#maximum-paths 5

---Configuración en R3---
R3(config)#router eigrp 15
R3(config-router)#network 192.168.254.8 0.0.0.3
R3(config-router)#network 192.168.254.12 0.0.0.3
R3(config-router)#network 172.30.0.0 0.0.255.255
R3(config-router)#passive-interface Gi0/1
R3(config-router)#maximum-paths 5

---Configuración en R4---
R4(config)#router eigrp 15
R4(config-router)#network 192.168.254.16 0.0.0.3
R4(config-router)#network 192.168.254.12 0.0.0.3
R4(config-router)#network 172.40.0.0 0.0.255.255
R4(config-router)#passive-interface Gi0/1
R4(config-router)#maximum-paths 5
```

¿Qué RID aplicará cada router?

En ninguno de ellos ha sido definido manualmente, por lo tanto, será asignado de manera automática en relación con el criterio de selección llevado a cabo por el protocolo. Como tampoco existen interfaces loopback, se aplicará el valor de la IP más alta configurada en interfaces físicas operativas, por lo tanto:

- R1 aplicará como RID la IP configurada en su interfaz Se0/0/1. Para llegar a esta conclusión bastará con analizar los diferentes rangos de red a los que pertenece el router, donde cualquier IP perteneciente a las redes 192.68.254.4/30 y 172.10.0.0/16 nunca podrá ser mayor que aquellas direcciones incluidas en la 192.168.254.16/30 (Se0/0/1).

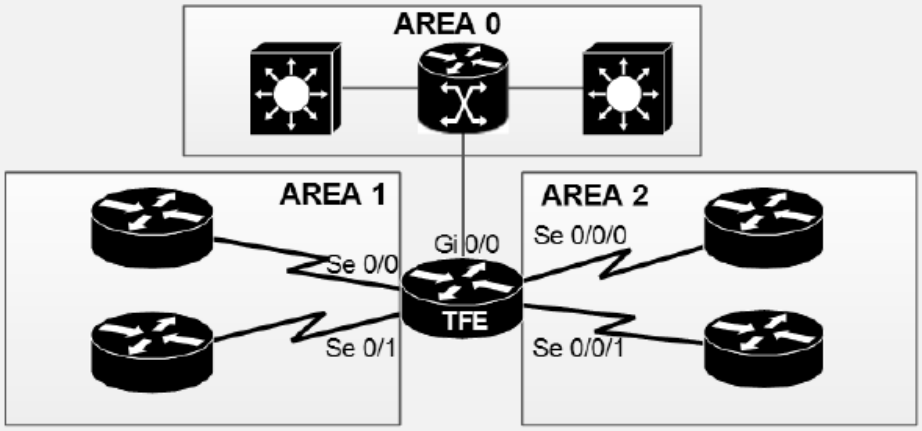
- Por la misma razón, R2 tomará como RID la IP configurada en su interfaz Se0/0/1.

- Así mismo, R3 aplicará como RID la IP asignada a su interfaz Se0/0/0.

- Por último, y conforme al mismo criterio, R4 tomará como RID la IP configurada en su interfaz Se0/0/1.

Reto 6.2 - Dada la siguiente topología, configurar OSPF en el router TFE conforme a la siguiente documentación:

Router	Proceso OSPF	RID	Red - Interfaz
TFE	25	25.25.25.25	172.10.0.0/16 - Se0/0 172.20.0.0/16 - Se0/1 192.168.10.0/24 – Se0/0/0 192.168.20.0/24 – Se0/0/1 10.0.0.0/8 – Gi0/0




```
---Configuración en TFE---
TFE(config)#router ospf 25
TFE(config-router)#router-id 25.25.25.25
TFE(config-router)#network 172.10.0.0 0.0.255.255 area 1
TFE(config-router)#network 172.20.0.0 0.0.255.255 area 1
TFE(config-router)#network 192.168.10.0 0.0.0.255 area 2
TFE(config-router)#network 192.168.20.0 0.0.0.255 area 2
TFE(config-router)#network 10.0.0.0 0.255.255.255 area 0
```

TEST CAPÍTULO 6: PROTOCOLOS DE ENRUTAMIENTO

1.- ¿Cuál es la diferencia principal entre los protocolos de enrutamiento IGP y EGP?

A. Los IGP son aquellos necesarios para comunicar diferentes sistemas autónomos, mientras que los EGP son implementados dentro de un mismo AS.
B. Los IGP hacen uso del algoritmo vector distancia mientras que los EGP de link state.
C. Todos los IGP se basan en un modelo en áreas, mientras que los EGP en el número de proceso configurado.
D. Ninguna de las anteriores.

2.- ¿Cuál de los siguientes protocolos no es IGP?

A. OSPF.
B. RIP.
C. EIGRP.
D. IS-IS.
E. Ninguna de las anteriores.

3.- ¿Cuál de los siguientes protocolos resulta la opción idónea a implementar sobre un entorno compuesto por routers de diferentes fabricantes?

A. BGP.
B. OSPF.
C. EIGRP.
D. RIP.

4.- Un router ha aprendido dos rutas hacia la misma subred, la primera de ellas mediante EIGRP y la segunda a través de OSPF. ¿Cuál instalará en su tabla?

A. La aprendida desde EIGRP.
B. La aprendida desde OSPF.
C. Ambas, realizando balanceo de carga.
D. Si la métrica coincide, se instalarán ambas.
E. Aquella con menor métrica.

5.- En un router se ha configurado EIGRP como protocolo de enrutamiento y este ha aprendido 5 rutas hacia una misma subred. ¿Cuál de ellas instalará en su tabla?

A. Aquella con menor distancia administrativa.

B. Aquella con menor métrica.
C. Aquella con menor número de saltos.
D. Todas.

6.- EIGRP hace uso del algoritmo…
A. Vector distancia avanzado.
B. Bellman-Ford.
C. Link State.
D. Vector distancia.

7.- De las siguientes, ¿cuál puede identificar una causa que limite el uso de EIGRP?
A. El sistema de métrica utilizado.
B. La lenta convergencia.
C. La creación de bucles temporales en redes de gran tamaño.
D. Que solo pueda ser configurado en routers Cisco.

8.- Diferentes protocolos de enrutamiento hacen uso del horizonte dividido para evitar bucles de capa 3. ¿Cuál de las siguientes afirmaciones define mejor esta técnica?
A. El horizonte dividido es la técnica mediante la cual una actualización de enrutamiento no es reenviada a través de la misma interfaz por la que fue recibida.
B. El horizonte dividido es la técnica utilizada para notificar a los vecinos de rutas inalcanzables, asignándoles una métrica infinita.
C. Es la técnica utilizada para notificar a los vecinos de manera inmediata cuando se ha producido algún cambio en la topología.
D. Ninguna de las anteriores.

9.- En RIP, ¿qué significado tiene una métrica con valor 16?
A. Que desde el origen hasta el destino existen 16 saltos de diferencia.
B. Que el paquete podrá dar un máximo de 16 saltos antes de ser descartado.
C. Que el paquete podrá dar un máximo de 15 saltos antes de ser descartado.
D. Que la red de destino es inalcanzable.

10.- ¿Qué dirección multicast es utilizada por el protocolo EIGRP?
A. 224.0.0.5
B. 224.0.0.9

C. 224.0.0.10
D. 224.0.0.8

11.- ¿Cuáles de las siguientes opciones identifican requisitos para que dos routers establezcan adyacencia en EIGRP? (Seleccionar dos respuestas)

A. Coincidir en el número de ASN.
B. Que las interfaces pertenezcan al mismo área.
C. Que la dirección IP de ambos dispositivos pertenezcan al mismo rango de red.
D. Que la velocidad del enlace coincida en ambos extremos.

12.- La ruta sucesor utilizada por EIGRP dispone de una distancia factible de 215555. Sin embargo, se ha aprendido otra hacia la misma subred con una RD de 215550 y FD de 262000. ¿Cómo procederá EIGRP con esta nueva ruta?

A. La utilizará como sucesor, eliminando aquella con valor FD 215555.
B. Descarta la nueva ruta.
C. La asigna como sucesor factible.
D. Ninguna de las anteriores.

13.- De las siguientes, ¿qué interfaces se recomiendan configurar como pasivas en cualquier protocolo de enrutamiento que lo permita?

A. Aquellas que tan solo conectan con dispositivos finales.
B. Las interfaces seriales.
C. Aquellas que no actúan como puerta de enlace.
D. Todas las anteriores.

14.- Un router configurado con OSPF aún no ha establecido ninguna adyacencia con vecinos. ¿Qué rutas incluirá su LSDB?

A. Ninguna.
B. Su LSDB no ha sido creada porque aún no ha recibido ningún LSA.
C. Su LSDB no ha sido creada porque aún no ha recibido ningún LSU.
D. Sus redes directamente conectadas.

15.- En un entono multiacceso, ¿qué función desempeña el router BDR?

A. El intercambio de rutas con el resto de routers de la topología.
B. El intercambio de rutas con el resto de routers de su misma área.

C. El intercambio de rutas con el DR y en caso de caída de este, tomar su rol inmediatamente.
D. El intercambio de rutas con el DR.

16.- ¿Qué valor de distancia administrativa asigna IOS a OSPF?
A. 90.
B. 100.
C. 110.
D. 120.

17.- Dos routers vecinos han establecido adyacencia en OSPF y ambos se encuentran en estado *Loading.* ¿Qué proceso están llevando a cabo?
A. El intercambio de paquetes LSU.
B. La negociación de parámetros OSPF.
C. Ambos disponen de la misma LSDB.
D. Ninguna de las anteriores.

18.- ¿Qué beneficios aporta la distribución en áreas de OSPF? (Seleccionar dos respuestas)
A. Mejor aprovechamiento del ancho de banda.
B. Mayor rapidez de convergencia.
C. Mayor seguridad.
D. Cálculo del algoritmo SPF más rápido.

19.- Dos routers han sido configurados con OSPF, pero no logran establecer adyacencia. ¿Cuál de los siguientes motivos puede ser la causa del problema?
A. RID no configurado.
B. Las interfaces pertenecen a diferentes áreas.
C. El proceso de OSPF no coincide en ambos routers.
D. Todas las anteriores.

20.- En una tabla de rutas se ha asignado el código “O” en varias de sus entradas. ¿Qué significado tiene?
A. Que se trata de una ruta sumarizada.
B. Que la ruta ha sido aprendida e instalada mediante el protocolo EIGRP.
C. Que el valor del campo TTL para los paquetes que hagan uso de dicha ruta será igual a 0.
D. Que la ruta ha sido aprendida e instalada a través del protocolo OSPF.

21.- Configurando EIGRP a través de la CLI se ha recibido el siguiente mensaje en pantalla…

```
%DUAL-5-NBRCHANGE: IP-EIGRP 1: Neighbor 192.168.6.25 (Serial 0/0/0) is up: new
adjacency
```

En relación con la información incluida, ¿cuáles de las siguientes afirmaciones son correctas? (Seleccionar 3 respuestas)

A. Se ha establecido adyacencia con un vecino cuya IP es 192.168.6.25.
B. Se ha establecido adyacencia con un vecino a través de la IP local 192.168.6.25.
C. Se ha establecido adyacencia con un vecino a través de su interfaz Se0/0/0.
D. Se ha establecido adyacencia con un vecino a través de la interfaz local Se0/0/0.
E. Se ha establecido adyacencia con un vecino perteneciente al proceso 1 de EIGRP.
F. Se ha establecido adyacencia con un vecino perteneciente al proceso 5 de EIGRP.

22.- Un router ha sido configurado con una interfaz loopback (10.20.20.20) y otra física (192.168.254.254), ambas en estado operativo. Si en EIGRP no fuera configurado el RID, ¿qué valor tomará automáticamente?

A. 10.20.20.20.
B. 192.168.254.254.
C. Ninguno.
D. El RID por defecto, 0.0.0.0.

23.- ¿Qué código es utilizado por IOS para mostrar en la tabla de rutas aquellas aprendidas e instaladas por EIGRP?

A. O
B. D
C. E
D. EI

24.- De las siguientes, ¿cuál define mejor la diferencia entre LSU y LSA?

A. Los LSU contienen información sobre cada red del router local y topología en general, mientras que los LSA son necesarios para mantener la adyacencia entre routers vecinos.
B. Los LSA contienen información sobre cada red, mientras que los LSU están compuestos por uno o más LSA.
C. La LSDB de cada router se genera y completa gracias a los LSU, mientras que los LSA son agregados directamente a la tabla de rutas.
D. Existen diferentes tipos de LSU, mientras que de LSA tan solo uno.

25.- Una topología corporativa está compuesta por 500 routers y 2000 subredes. Tras implementar OSPF se ha comprobado que la convergencia resulta muy lenta, los routers operan al 100% de CPU y ante cualquier cambio el enrutamiento falla hasta que finaliza la convergencia. De las siguientes opciones, ¿cuál es la idónea para aliviar o solucionar el problema?

A. Ampliar la memoria de los routers.
B. Crear un diseño de OSPF basado en áreas.
C. Disminuir el número de subredes.
D. Ampliar el ancho de banda de los enlaces entre routers.

26.- Configurando un router a través de la CLI se obtiene el siguiente mensaje constantemente…

```
00:06:01: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from
backbone area must be virtual-link but not found from 192.168.1.2, Serial0/0/0
spf 2
```

¿Qué significado tiene?

A. Identifica a un router ABR que ha establecido adyacencia con algún vecino de un área diferente.
B. El router no puede establecer adyacencia con el vecino debido a un error en los RID de ambos.
C. El router ha logrado formar adyacencia con un vecino a través de la interfaz Se0/0/0.
D. El router no puede establecer adyacencia con el vecino porque pertenecen a diferentes áreas.

27.- ¿Qué dirección de destino es utilizada por RIPv2 para enviar actualizaciones de enrutamiento?

A. 255.255.255.255.
B. 224.0.0.9.
C. 224.0.0.10.
D. La dirección IP de cada router vecino.

28.- ¿Cuáles de las siguientes características son soportadas por RIPv2 y no por la versión 1? (Seleccionar dos respuestas)

A. Horizonte dividido.
B. VLSM.
C. Envenenamiento de ruta.
D. Sumarización.

29.- ¿Qué código es utilizado en la tabla de rutas para identificar aquellas aprendidas por RIPv2?

A. R.
B. R2.
C. S.
D. C.

30.- Tras ejecutar el comando *"show ip bgp summary"* en R1, se comprueba que uno de los vecinos mantiene el estado *"idle"*. ¿A qué puede ser debido?

A. La conexión TCP entre ambos se ha establecido, pero no ha concluido.
B. La conexión TCP ha finalizado, pero R1 aún no ha recibido ningún mensaje BGP del vecino en cuestión.
C. Se está llevando a cabo la negociación entre ambos para establecer adyacencia.
D. En la configuración BGP de R1 se ha ejecutado el comando *shutdown* sobre dicho vecino.

31.- ¿Qué función desarrolla el comando *"network [id red] mask [máscara]"* en BGP?

A. Publica la red en cuestión a los vecinos BGP.
B. Identifica la interfaz configurada con dicha red, a través de la cual se establecerán adyacencias y enviarán actualizaciones de enrutamiento.
C. Agrega la red definida a la tabla de rutas BGP.
D. Ninguna de las respuestas anteriores es correcta.

32.- R1, a través del protocolo BGP, ha recibido 3 rutas hacia la misma red. Tras ejecutar el comando *"show ip bgp"* se comprueba que una de ellas es representada mediante el código *"*>"*. ¿A qué puede ser debido?

A. Sobre la misma se ha aplicado envenenamiento de ruta, marcándola como inalcanzable.
B. Representa la mejor ruta hacia el destino en cuestión.
C. Se está ejecutando balanceo de carga hacia la red de destino.
D. Dicha ruta es local, pertenece a R1.

33.- ¿Qué función ejerce el valor "*AS_Path"* sobre cada ruta en BGP?

A. Si hacia un mismo destino existiera más de una ruta, gracias a dicho valor se determina la mejor de ellas.
B. Facilita información del entorno sobre el cual debe ser aplicada la ruta en cuestión, pudiendo ser iBGP o eBGP.
C. Facilita información como su métrica, número de saltos máximos permitidos o velocidad de cada enlace.
D. Las respuestas A y B son correctas.

SEGURIDAD EN CAPA 3

7

LISTAS DE CONTROL DE ACCESO: CONCEPTOS BÁSICOS

La seguridad presente en cualquier infraestructura de red representa uno de los elementos más importantes y críticos de esta, hasta el punto de influir de manera directa en el éxito o fracaso de cualquier compañía, siendo un claro ejemplo de ello aquellas que basan su modelo de negocio en servicios online. Sea cual sea el propósito, el objetivo siempre será el mismo, proteger la información ante el robo o acceso no autorizado. Para lograrlo, la estrategia llevada a cabo suele ser definida en relación con diferentes factores, como la política de la compañía, dispositivos disponibles para aplicarla o el modo en que los usuarios acceden a la información. Sea como fuere, un buen plan deberá abarcar las 7 capas del modelo OSI.

A nivel de red la gran mayoría de dispositivos disponen de mecanismos de seguridad, siendo algunos de ellos diseñados exclusivamente para tal propósito, como es el caso de los firewalls, IDS o IPS, gracias a los cuales se hace posible garantizar un nivel de protección adecuado sobre entornos corporativos. Además, esta puede y debe ser complementada con las características propias de otros dispositivos como switchs o routers. Dentro de este ámbito IOS dispone de diferentes opciones, entre las que se encuentran el acceso autorizado mediante usuario/contraseña y sus privilegios, cifrado de la comunicación o ACL, representando estas últimas el método por excelencia de filtrado de paquetes en capa 3. Debido a ello, el presente capítulo será dedicado prácticamente en su

totalidad a su estudio y configuración, para a posteriori analizar diferentes servicios vulnerables y finalizar con NAT.

Las ACL, o listas de control de acceso, representan el mecanismo mediante el cual IOS permite o descarta una comunicación conforme a una serie de criterios, como una dirección IP, red o protocolo. Estos filtros son aplicados a nivel de interfaz y pueden ser definidos en dos direcciones, entrada *(in)* y salida *(out)*. Una ACL de entrada examina los paquetes recibidos a través de un determinado enlace y por lo tanto su filtrado se ejecuta antes de que el router tome una decisión de reenvío. Por su contra, las ACL de salida aplican su criterio a los paquetes que el router se dispone a enviar a través de la interfaz donde fuera aplicada, y por supuesto, antes de que dicho reenvío se lleve a cabo.

Un ejemplo gráfico de ambas podría ser el siguiente, donde R1 debe ser configurado con una ACL para filtrar el tráfico desde el Host A hacia el servidor. ¿En qué interfaces y dirección se podrá aplicar?

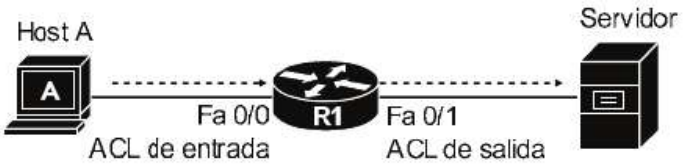


Fig. 7-1 Dirección de las ACL.

- Si la ACL fuera configurada en la interfaz Fa0/0 debe ser de entrada porquE el tráfico enviado desde Host A hacia el router es recibido por dicha interfaz.

- Sin embargo, si fuera aplicada en Fa0/1 debe ser de salida porque el tráfico generado por el Host A ya ha sido procesado y será reenviado a través de la misma.

Tipos de ACL

IOS permite la configuración de dos tipos de ACL, estándar y extendida. Las primeras resultan las más sencillas, donde el criterio de filtrado tan solo se basa en el origen de la comunicación (IP o red). Por su contra, las extendidas también permiten definir el destino, así como los puertos utilizados. Ambas son clasificadas por Cisco de la siguiente manera:

- ACL estándar numerada (1-99 o 1300-1999).
- ACL extendida numerada (100-199 o 2000-2699).
- ACL nombradas: estándar o extendida.

ACL	Filtrado	Numeradas	Nombradas
Estándar	IP/red de origen	El ID de ACL es un valor comprendido entre 1-99 o 1300- 1999. Su configuración se lleva a cabo en el modo de configuración global.	El ID de ACL es un nombre. Modo de configuración propio de la ACL.
Extendida	IP/red de origen/destino Puerto origen/destino Otros	El ID de ACL es un valor comprendido entre 100-199 o 2000-2699. Su configuración se lleva a cabo en el modo de configuración global.	ID de ACL es un nombre. Modo de configuración propio de la ACL.

ACL ESTÁNDAR NUMERADA

Como recién se ha mencionado, una ACL estándar representa la solución de seguridad en capa 3 más básica disponible en IOS. Ello es debido a que simplemente basa su filtrado en el origen de la comunicación, ya sea una dirección IP o un ID de red. La presente sección será dedicada a su análisis y configuración, enfocando ambos casos de manera práctica.

Lógica aplicada en una ACL estándar

Una lista de control de acceso está compuesta por una serie de sentencias que el administrador de red configura individualmente y que el router agrega de manera secuencial a medida que estas son definidas, creando así, para una misma ACL, diferentes filtros listados en el mismo orden en que han sido configurados. conforme a dicho orden, cada paquete será inspeccionado hasta que se produzca la primera coincidencia, en cuyo caso se ejecuta la acción configurada en la misma.

Por ejemplo, en la siguiente ACL, los paquetes con dirección de origen 192.168.1.50 y 192.168.1.110 serán descartados, mientras que aquellos con IP 192.168.1.25 serán permitidos.

```
1.- Dir. Origen: 192.168.1.50 -> Denegar
2.- Dir. Origen: 192.168.1.25 -> Permitir
3.- Dir. Origen: 192.168.1.110 -> Denegar
```

Sin embargo, el administrador de red decide agregar una nueva sentencia para permitir también la IP 192.168.1.50, quedando definida de la siguiente manera:

```
1.- Dir. Origen:  192.168.1.50 -> Denegar
2.- Dir. Origen:  192.168.1.25 -> Permitir
3.- Dir. Origen:  192.168.1.110 -> Denegar
4.- Dir. Origen:  192.168.1.50 -> Permitir
```

Ahora existen 2 entradas para una misma dirección. ¿Cómo actuará el router? La lógica consiste en aplicar la primera coincidencia, que en este caso define la denegación, por lo tanto, aquellos paquetes con origen 192.168.1.50 continuarán siendo descartados. Una vez aplicado un filtro sobre un paquete, el resto de sentencias no son aplicables sobre el mismo.

En este caso, el modo de proceder correcto consiste en eliminar el filtro de denegación y posteriormente crear uno nuevo que permita la comunicación.

Un detalle importante a tener en cuenta es que todas las ACL definen de manera automática una denegación implícita (*deny any*) al final de la lista, de tal manera que si un paquete no coincide con ninguno de los filtros configurados, automáticamente será descartado.

La denegación implícita puede ser evitada configurando manualmente un *permit any* como última entrada en la ACL, sin embargo, su aplicación resulta poco o nada recomendable.

CÓMO DEFINIR UNA ACL ESTÁNDAR

Tanto definir una ACL estándar como agregar nuevos filtros sobre alguna ya existente se lleva a cabo a través del comando **access-list *[1-99 | 1300-1999] [permit | deny] [IP/red origen]***, ejecutado desde el modo de configuración global, donde:

- *[1-99 | 1300-1999]:* Define el ID de la ACL a la cual se agregará la sentencia en cuestión. Su valor debe formar parte de alguno de los dos rangos, ya que son los destinados por IOS para la creación de las ACL estándar.

- *[permit | deny]:* Identifica la acción a ejecutar.

- *[IP/red origen]:* Como su nombre indica, representa la dirección IP o red sobre la cual se aplicará la acción de permitir o denegar. Esta puede identificar a un solo host, un conjunto de ellos o cualquier dispositivo.

ACL estándar numerada para un host

Son aquellas cuya acción configurada tan solo afecta a un determinado dispositivo, identificado mediante su dirección IP.

Ejemplo: Configurar una ACL estándar para que los paquetes con origen 192.168.50.50 sean permitidos.

```
Router(config)# access-list 1 permit 192.168.50.50
```

En versiones más recientes de IOS también se debe agregar el parámetro *host* antes de la IP, de tal manera que:

```
Router(config)# access-list 1 permit host 192.168.50.50
```

ACL estándar numerada para un conjunto de hosts

Cuando la acción configurada debe ser aplicada sobre un conjuntos de hosts, como todos los dispositivos de una misma subred, se deberá agregar el ID acompañado de la máscara wildcard. En capítulos anteriores se ha mencionado que esta representa el valor inverso de la máscara de red, por ejemplo, para una 255.255.255.0, su wildcard sería 0.0.0.255. Sin embargo, en las ACL dicha definición varía significativamente, y es que en este caso pueden identificar un rango de direcciones IP que incluso, no tienen por qué pertenecer a la misma subred.

Para lograrlo, los bits son analizados de la siguiente manera:

- *Binarios 0:* Se comparan con el ID de red incluido en la sentencia y en caso de coincidencia se aplica la acción configurada.

- *Binarios 1:* No son analizados.

Por ejemplo, en una ACL se ha aplicado la acción de denegar para el ID 172.20.0.0 con wildcard 0.0.255.255, dando como resultado que cualquier paquete cuya IP comience por 172.20 sea descartado. Ello es debido a que los dos primeros bytes de la wildcard tienen el valor 0.0 (todos los bits a 0 en binario) lo que significa que a cualquier IP cuyos dos primeros bytes coincidan con aquellos del ID configurado se les aplicará la acción definida, en este caso la denegación. Los dos siguientes bytes tienen el valor 255.255 (todos los bits a 1 en binario) por lo que no serán analizados. El resultado final será que la ACL descartará cualquier paquete con IP de origen 172.20.x.x.

Otros ejemplos podrían ser:

1.- Denegar el acceso a todos los hosts pertenecientes a la subred 192.168.1.0/24.

```
Router(config)# access-list 1 deny 192.168.1.0  0.0.0.255
```

- Se solicita la denegación a una subred específica con máscara /24. En estos casos bastará con definir como wildcard el valor inverso de la máscara de red.

2.- Permitir el acceso a todos los hosts cuya dirección IP comience por 10.10.

```
Router(config)# access-list 1 permit 10.10.0.0  0.0.255.255
```

- Se solicita permitir el acceso a un conjunto de hosts que no tienen por qué pertenecer a la misma subred. Es por ello que en este caso, la wildcard necesaria es 0.0.255.255, la cual dará como resultado que los paquetes cuyos dos primeros bytes de su IP coincidan con aquellos del ID de red indicado sean permitidos. El resto de bytes son ignorados, logrando de esta manera el objetivo de permitir el acceso a cualquier IP perteneciente al rango 10.10.x.x.

ACL estándar numerada para cualquier host

IOS también dispone la opción de ejecutar sobre cualquier paquete la misma acción. Para ello se debe hacer uso del parámetro *any,* siendo su aplicación más lógica (aunque poco común) la creación de una sentencia al final de la ACL para permitir cualquier comunicación. ¿Por qué? Por defecto, cada ACL define una denegación implícita como último filtro, con el objetivo de bloquear cualquier paquete que no coincida con ninguna de las entradas anteriores. Si se deseara que estos fueran permitidos se hace necesario configurar una sentencia para ello, la cual debe posicionarse al final de la ACL.

Un ejemplo puede ser:

```
Router(config)# access-list 1 permit any
```

Configuración de ACL estándar numerada

El procedimiento, al completo, consta de las siguientes acciones:

Paso 1: Identificar en qué router, interfaz y dirección será aplicada. Las ACL estándar deben ser configuradas lo más cercano posible al destino, con el fin de evitar bloqueos no deseados.

Paso 2: Definir cada entrada de la ACL con el comando y parámetros analizados en párrafos anteriores, teniendo en cuenta el orden de creación de las sentencias y el *deny* implícito al final de todas ellas.

Paso 3: Aplicar la ACL en la interfaz seleccionada con el comando **ip access-group *[número ACL] [in | out]***, desde el modo de configuración de la propia interfaz y donde *in* | *out* hace referencia a la dirección deseada.

Supuesto práctico 1: Conforme a la siguiente topología, configurar una ACL que cumpla los siguientes requisitos:
- Todo el tráfico de la VLAN 10 con destino a la red 172.20.0.0/16 debe ser bloqueado, exceptuando el host con IP 192.168.10.100, que sí tendrá acceso.
- Cualquier otro tráfico también debe ser permitido.

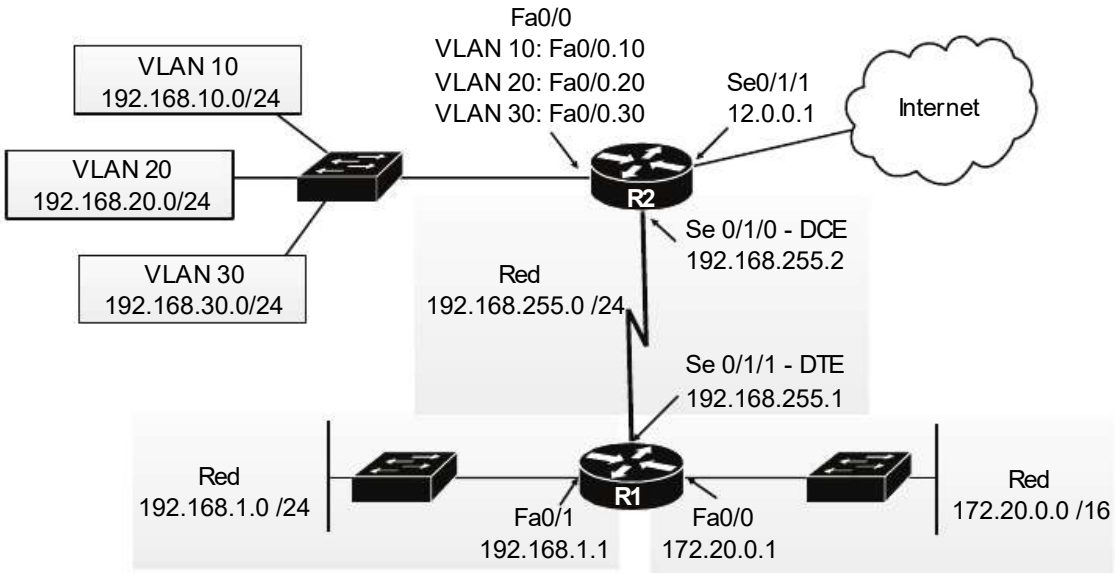


Fig. 7-2 Diseño de red para supuestos prácticos de las ACL.

Paso 1: Identificar en qué router será aplicada.

La primera acción a llevar a cabo consiste en recopilar toda la información necesaria. En este caso los datos a tener en cuenta son los siguientes:

Origen: 192.168.10.0/24 (VLAN10), denegar.
Origen: 192.168.10.100 (Host en VLAN 10), permitir.
Origen: Cualquiera, permitir.
Destino: Red 172.20.0.0/16.

¿En qué router, interfaz y dirección será configurada? Una ACL estándar debe estar ubicada lo más cercano posible al destino, siendo en este caso la red 172.20.0.0, la cual forma parte de R1. Por lo tanto, la ACL será aplicada en dicho router. La interfaz correcta es aquella que conecta directamente con el destino, en este caso la Fa0/0, y la dirección de salida, porque los paquetes de la VLAN10 son procesados por R1 y deben ser reenviados a través de la misma.

¿En qué orden deben ser definidas las entradas en la ACL? A excepción de un host, se solicita bloquear una subred al completo. Los filtros son procesados en el mismo orden en que han sido configurados, y desde que exista una coincidencia se ejecuta la acción establecida en él. Es por ello que la primera sentencia debe ser aquella que permita el tráfico del host, la segunda, bloquear la subred completa, y por último, permitir cualquier tipo de comunicación, de tal manera que:

```
1.- Dir. Origen: 192.168.10.100 -> Permitir
2.- Dir. Origen: Red 192.168.10.0/24 -> Denegar
3.- Dir. Origen: any -> Permitir
```

¿Por qué una ACL estándar debe estar ubicada lo más cercano posible al destino? Las ACL estándar son muy sencillas y tan solo basan su filtrado en una dirección de origen, ya sea de host, o de red. Imagina, que por error, fuera configurada en la interfaz Fa0/0 de R2 en dirección de entrada. Ello daría como resultado que todos los paquetes de la VLAN 10 fueran descartados de manera inmediata sin tener en cuenta el destino de cada uno de ellos. El objetivo consiste en bloquearlos solo para la red 172.20.0.0, sin embargo, en este caso se denegarían para cualquier comunicación.

Paso 2: Configurar la ACL en el router.

Decididos el orden de las sentencias y el router donde será aplicada, tan solo bastaría proceder a ello haciendo uso de los comandos analizados en párrafos anteriores.

```
R1(config)# access-list 1 permit 192.168.10.100
R1(config)# access-list 1 deny 192.168.10.0  0.0.0.255
R1(config)# access-list 1 permit any
```

Paso 3: Aplicar la ACL en la interfaz y dirección seleccionada.

Por último, aplicarla en Fa0/0 con dirección de salida.

```
R1(config)# int Fa0/0
R1(config-if)# ip access-group 1 out
```

Una vez concluida la configuración, podrá ser verificada gracias al comando **show ip access-list**, el cual muestra en pantalla un listado de todas las ACL creadas en el router y el orden de sus filtros.

```
R1#show ip access-list
Standard IP access list 1
    permit host 192.168.10.100
    deny 192.168.10.0 0.0.0.255
    permit any
```

Supuesto práctico 2: Conforme a la topología de la *Fig. 7-2*, denegar el acceso a Internet a todos los hosts cuya IP comience por 192.168.x.x y permitirlo para cualquier otra dirección.

En este caso el destino hace referencia a Internet, que conecta a través de la interfaz Se0/1/1 de R2, por lo tanto, la ACL debe ser configurada en dicho router e interfaz, en dirección de salida y donde la denegación debe ser la primera sentencia, para luego permitir cualquier comunicación restante.

```
R2(config)# access-list 5 deny 192.168.0.0  0.0.255.255
R2(config)# access-list 5 permit any
R2(config)# int Se0/1/1
R2(config-if)# ip access-group 5 out
```

Reto 7.1 – Conforme a la misma topología, configurar dos ACL estándar numeradas cumpliendo los siguientes requisitos:
- ACL1: Impedir el acceso de la red 192.168.20.0/24 y del host 192.168.30.150 al destino 192.168.1.0/24. Permitir cualquier otro tipo de tráfico.
- ACL2: Bloquear toda comunicación de la red 172.20.0.0/16 hacia cualquier destino.

Solución al final del capítulo.

Cálculo de rangos mediante la máscara wildcard

Es posible que en determinadas ocasiones resulte necesario calcular el rango de hosts definido en sentencias ACL ya creadas. Dicha tarea se lleva a cabo con relación al ID y máscara wildcard configurada en ella, y consta de un proceso tan sencillo como:

- El inicio del rango corresponde al ID de red definido en la sentencia.

- El fin del rango es igual al resultado del ID más la wildcard.

Para el siguiente filtro:

```
R2(config)# access-list 5 deny 192.168.0.0  0.0.255.255
```

¿Qué direcciones IP serán denegadas?

- El inicio de rango es el ID configurado: 192.168.0.0
- El fin del rango es igual al ID más la wildcard, por lo tanto: 192.168.0.0 + 0.0.255.255 = 192.168.255.255
- Rango definido en esta sentencia: 192.168.0.0 - 192.168.255.255

Otros ejemplos:

Sentencia ACL	**Rango**
access-list 1 deny 10.10.0.0 0.0.128.255	10.10.0.0 – 10.10.128.255
access-list 2 permit 192.168.128.0 0.0.0.127	192.168.128.0 – 192.168.128.127
access-list 3 deny 172.15.0.0 0.0.0.255	172.15.0.0 – 172.15.0.255
access-list 4 deny 172.30.30.0 0.0.1.255	172.30.30.0 – 172.30.31.255
access-list 5 permit 10.10.10.10	10.10.10.10
access-list 6 permit 192.168.128.0 0.0.127.255	192.168.128.0 – 192.168.255.255
access-list 1 deny 10.1.0.0 0.127.255.255	10.1.0.0 – 10.128.255.255

ACL EXTENDIDA NUMERADA

Las ACL extendidas resultan algo más complejas que las estándar, pero a su vez permiten un mayor abanico de opciones gracias a las cuales se podrán ajustar las necesidades de filtrado con mayor precisión. Aun así, mantienen bastantes similitudes, entre las que se encuentran:

- Ambas son aplicadas a nivel de interfaz y en una dirección.

- Ambas generan un listado de filtros a medida que estos son configurados, para a posteriori analizar los paquetes conforme al mismo orden. Desde que exista una coincidencia, se ejecuta la acción definida en la misma.

- Ambas incorporan un *deny* implícito al final de la lista. Cualquier paquete que no coincida con ninguno de los filtros configurados manualmente será descartado.

Mientras que las ACL estándar tan solo basan su filtrado en la dirección de origen, las extendidas permiten identificar tanto origen como destino, así como el protocolo o puerto utilizado durante la comunicación. Todo ello será objeto de estudio a lo largo de los siguientes párrafos, prestando especial atención a las opciones disponibles y sus variantes de configuración.

Filtrado basado en protocolo y direcciones de origen y destino

Una de las características principales de las ACL extendidas consiste en la posibilidad de definir el filtrado con relación al protocolo utilizado durante la comunicación, siendo los más comunes tcp, udp o icmp, entre otros. Ello, junto a las direcciones de origen y destino, permiten identificar con exactitud el tipo de tráfico sobre el cual se desea ejecutar una determinada acción. Su configuración se lleva a cabo mediante la sintaxis **access-list *[número] [permit | deny] [protocolo] [ip/red origen] [ip/red destino]***, desde el modo de configuración global, donde:

- *[número]:* Hace referencia al ID de ACL a la cual se agregará la sentencia en cuestión. Al ser extendida debe estar comprendido entre los rangos 100-199 o 2000-2699.

- *[permit | deny]:* La acción a ejecutar con aquellos paquetes que coincidan con todos los parámetros definidos.

- *[protocolo]:* El protocolo sobre el cual se desea aplicar el filtro.

- *[ip/red origen]:* La IP de origen o conjunto de hosts que serán examinados.

- *[ip/red destino]:* La IP, o conjunto de hosts identificados como destino de la comunicación.

Resulta importante recalcar que la acción de permitir o denegar tan solo se llevará a cabo cuando todos los campos definidos (protocolo, origen y destino) coincidan con aquellos analizados en el paquete.

Algunos ejemplos podrían ser:

Sentencia ACL	Resultado
access-list 101 deny tcp any any	Descarta paquetes con cabecera TCP, desde cualquier origen hacia cualquier destino.
access-list 101 deny udp any any	Descarta tráfico con cabecera UDP, desde cualquier origen hacia cualquier destino.
access-list 101 deny icmp any any	Descarta comunicaciones con cabecera ICMP, desde cualquier origen hacia cualquier destino.
access-list 101 permit ip any any	Permite comunicaciones IP (lo que incluye TCP, UDP e ICMP) desde cualquier origen hacia cualquier destino.
access-list 101 deny ip 192.168.1.100 any	Descarta paquetes IP con dirección de origen 192.168.1.100 hacia cualquier destino.
access-list 101 permit icmp any 10.10.10.10	Permite paquetes ICMP desde cualquier dirección de origen hacia el host de destino 10.10.10.10.

Filtrado basado en números de puerto TCP y UDP

En capa 4 los puertos resultan necesarios para identificar cada una de las aplicaciones o servicios ejecutados en un host, permitiendo de esta manera que el tráfico enviado a dicho dispositivo a través de un determinado puerto sea recibido y procesado por el software correcto. Una de las mayores ventajas de las ACL extendidas es que también permiten la opción de definir los puertos de origen y destino, logrando gracias a ello un filtrado más eficiente y específico, ya que habilita la posibilidad de ejecutar acciones sobre el tráfico generado o recibido por una aplicación o servicio en concreto. El comando necesario para llevar a cabo su configuración es **access-list *[número] [permit | deny] [protocolo] [IP/red origen] [puerto origen] [IP/red destino] [puerto destino]***, donde, como se puede observar, la mayoría de parámetros coinciden con el ya analizado filtrado de protocolo, desarrollando también la misma función, de modo que únicamente se agregan las opciones de puerto de origen y puerto de destino:

- *[puerto origen]: (opcional)* Como su nombre indica, hace referencia al puerto de origen de la comunicación.

- *[puerto destino]: (opcional)* Indica el puerto de destino a filtrar.

Además, ambos deben ser definidos con un parámetro adicional que desarrolla la función de identificar qué puerto o rango será analizado. Las opciones disponibles en este caso son las siguientes:

Parámetro	**Función**
eq *[puerto]*	Igual
ne *[puerto]*	Desigual
it *[puerto]*	Menor que...
gt *[puerto]*	Mayor que...
range *[x] [y]*	Rango de puertos *x - y*

Por ejemplo, un servidor web que utiliza el puerto 80 para dar servicio a los clientes de la red 192.168.10.0 /24 y que por motivos de seguridad tan solo debe recibir peticiones HTTP. Cualquier otro tipo de tráfico será bloqueado.

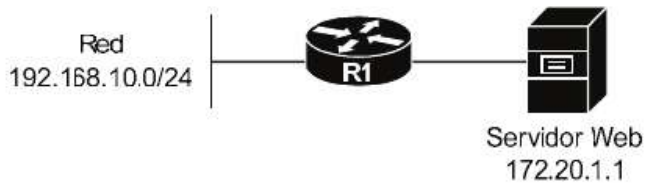


Fig. 7-3 Supuesto práctico filtrado basado en puerto.

Conforme a las condiciones descritas, se hace necesaria una ACL que permita a la red 192.168.10.0/24 comunicarse con el host 172.20.1.1 a través del puerto 80, es decir, tráfico HTTP. Con una ACL estándar no resulta posible cumplir el objetivo, sin embargo, con una extendida sí.

```
access-list 101 permit tcp 192.168.10.0  0.0.0.255  host 172.20.1.1 eq 80
```

Donde cada uno de los parámetros desarrolla la siguiente función:

- *access-list 101:* Crea una ACL numerada con ID 101 (extendida).

- *permit:* Define la acción a ejecutar sobre aquellos paquetes que coincidan con **todos** los parámetros configurados en la sentencia.

- *tcp:* Identifica el protocolo que debe ser analizado. En este caso TCP porque HTTP hace uso del mismo.

- *192.168.10.0 0.0.0.255:* Hace referencia al conjunto de dispositivos de origen que serán inspeccionados.

- *host 172.20.1.1:* Define el host de destino, en este caso la dirección IP del servidor web.

- *eq 80:* Identifica el puerto de destino que será examinado.

En la sentencia no ha sido configurada la opción "puerto de origen" porque los clientes web aplican valores aleatorios en dicho campo durante la comunicación con el servidor, además, lo que realmente interesa filtrar es el puerto de destino, que debe ser igual a 80.

Cualquier paquete que no coincida con los parámetros definidos será descartado gracias al *deny* implícito ubicado al final de la ACL.

IOS también permite la opción de identificar los "puertos bien conocidos" por su nombre. Estos representan los servicios más comunes a nivel de red, debido a lo cual disponen de un valor numérico único y reservado para cada uno de ellos, entre los que se encuentran:

Número de puerto	**Protocolo**	**Servicio**	**Comando a aplicar en la ACL**
20	TCP	FTP (datos)	ftp-data
21	TCP	FTP (control)	ftp
23	TCP	Telnet	telnet
25	TCP	SMTP	smtp
53	TCP	DNS	domain
69	UDP	TFTP	tftp
80	TCP	HTTP (WWW)	www
110	TCP	POP3	pop3
161	UDP	SNMP	snmp

De tal manera que ante cualquiera de los listados se podrá hacer uso del nombre en lugar de su valor numérico. En el ejemplo recién analizado se aplicó el parámetro "*eq 80*", pero también podría haber sido definido como "*eq www*". El resultado en ambos casos sería exactamente el mismo.

Configuración de ACL extendida numerada

En este caso, se deberán llevar a cabo las siguientes acciones:

Paso 1: Planificar en qué router, interfaz y dirección será aplicada. Las ACL extendidas, a diferencia de las estándar, deben ser ubicadas lo **más cercano posible al origen**, gracias a lo cual se evita que circulen por la red paquetes que posteriormente serán descartados.

Paso 2: Crear la ACL desde el modo de configuración global haciendo uso de los comandos recién analizados.

Paso 3: Aplicarla en la interfaz seleccionada con el comando **ip access-group *[número ACL] [in | out]***, desde el modo de configuración de la propia interfaz y donde *in* | *out* hace referencia a la dirección deseada.

Supuesto práctico 1: Conforme a la topología definida en la *Fig. 7-2*, definir una ACL que afecte a todos los hosts de la red 172.20.0.0 /16, cumpliendo los siguientes requisitos:
- Bloquear la comunicación TCP con la red de destino 192.168.1.0.
- Permitir paquetes UDP a través de los puertos 67 y 68 para el host de destino 192.168.10.10.
- Bloquear todo el tráfico UDP hacia cualquier destino.
- Permitir cualquier otro tipo de comunicación.

Paso 1: Planificar en qué router será configurada.

Las ACL extendidas deben ser ubicadas lo más cercano posible al **origen**, que en este caso es la red 172.20.0.0/16, conectada a R1 a través de la interfaz Fa0/0. Por lo tanto, será configurada en dichos router e interfaz. Los paquetes son recibidos a través de la misma, por lo que la dirección debe ser de entrada.

¿En qué orden serán aplicadas las sentencias? Como ya se ha mencionado, estas son ejecutadas secuencialmente en el mismo orden en que han sido configuradas. Se debe permitir la comunicación a través de dos puertos UDP, sin embargo, también se solicita bloquear todo el tráfico del mismo protocolo. Por lo tanto, resulta importante establecer como primer filtro aquel que permita los puertos solicitados, de lo contrario serán bloqueados. Otro detalle a tener en cuenta consiste en ubicar la sentencia que permite todo el tráfico en la última posición del listado, ya que el propósito de esta es no ejecutar el *deny* implícito presente en toda ACL.

Conforme a dicha teoría, el orden a aplicar es el siguiente:

1.- Permitir el tráfico UDP con puerto 67 y 68 para el host de destino 192.168.10.10.
2.- Bloquear la comunicación TCP con la red de destino 192.168.1.0.
3.- Bloquear el tráfico UDP hacia todos los destinos.
4.- Permitir cualquier otro tipo de comunicación.

En este caso, la sentencia que bloquea el tráfico TCP puede ser definida en cualquier otra posición de la lista, siempre y cuando sea antes del *permit any*.

Paso 2: Configurar la ACL en el router.

Decididos el orden de las sentencias y el router donde será ubicada, tan solo bastaría proceder a ello aplicando el comando de configuración ya analizado.

```
R1(config)#   access-list   110   permit   udp   172.20.0.0   0.0.255.255   host
192.168.10.10 eq 67
R1(config)#   access-list   110   permit   udp   172.20.0.0   0.0.255.255   host
192.168.10.10 eq 68
R1(config)#  access-list  110  deny  tcp  172.20.0.0  0.0.255.255  192.168.1.0
0.0.0.255
R1(config)# access-list 110 deny udp 172.20.0.0 0.0.255.255 any
R1(config)# access-list 110 permit ip 172.20.0.0 0.0.255.255 any
```

Paso 3: Aplicar la ACL en la interfaz y dirección seleccionada.

Por último, aplicarla en Fa0/0 con dirección de entrada.

```
R1(config)# int Fa0/0
R1(config-if)# ip access-group 110 in
```

Supuesto práctico 2: Conforme a la misma topología *(Fig. 7-2*). Se sospecha que dos usuarios, con direcciones IP 192.168.30.100 y 192.168.1.215, hacen uso de Internet para juegos online. Gracias a un analizador de protocolos se comprueba que el puerto de destino utilizado en dicha comunicación es el TCP 8545. Bloquearlo para ambos dispositivos.

Se solicita bloquear el puerto TCP 8545 para dos direcciones IP ubicadas en diferentes subredes, que a su vez están conectadas a distintos routers, R1 y R2. Las ACL extendidas deben situarse lo más cercano posible al origen con el fin de evitar tráfico innecesario en la red. Por lo tanto, resulta necesario configurar dos ACL, una en la interfaz Fa0/1 de R1 para bloquear el host 192.168.1.215, y otra en la interfaz Fa0/0 de R2 para bloquear aquel con IP 192.168.30.100, ambas en dirección de entrada.

```
R1(config)#access-list 101 deny tcp host 192.168.1.215 any eq 8545
R1(config)#access-list 101 permit ip any any
R1(config)#int Fa0/1
```

```
R1(config-if)# ip access-group 101 in

R2(config)#access-list 101 deny tcp host 192.168.30.100 any eq 8545
R2(config)#access-list 101 permit ip any any
R2(config)#int Fa0/0
R2(config-if)# ip access-group 101 in
```

Reto 7.2 – Haciendo uso de la misma topología. Se han detectado diferentes ataques procedentes de la subred 192.168.1.0/24 contra el servidor FTP de la compañía, ubicado en la VLAN20 con IP 192.168.20.15. Bloquear cualquier intento de conexión procedente de dicha subred a los puertos TCP 20 y 21 del servidor FTP. Cualquier otro tipo de tráfico debe ser permitido.

Solución al final del capítulo.

Reto 7.3 – Conforme a la misma topología. Los dispositivos de la red 192.168.1.0/24 tan solo deben disponer de acceso a los siguientes servicios:

1.- Al servidor DNS cuya IP es 192.168.10.250.
2.- Al servidor de correo POP3 y SMTP con dirección 192.168.10.251.
3.- Al servidor web de la empresa con IP 192.168.10.252.
4.- Cualquier otro tipo de comunicación para esta subred debe ser bloqueado.

Configurar una ACL en el router que proceda para lograr el resultado deseado.

Solución al final del capítulo.

Para dar fin a esta sección, resulta imprescindible analizar dos opciones disponibles durante la creación de una ACL, remark y log.

Remark habilita la posibilidad de agregar una descripción a la ACL, la cual puede servir de ayuda para identificarla cuando existen múltiples de ellas configuradas. La sintaxis para ello es **access-list *[ID]* remark *[descripción].***

El siguiente parámetro, *log*, muestra en pantalla y en tiempo real información sobre paquetes analizados que han coincidido con alguna de las sentencias presentes en la ACL, ya sea para permitirlo o denegarlo. Esta opción es recomendable utilizarla con prudencia y solo en casos de comprobación y verificación, ya que consume bastantes recursos en el dispositivo y resulta prácticamente imposible operar en la consola IOS debido a la cantidad de mensajes que serán mostrados, sobre todo en routers con gran volumen de tráfico. Para habilitarla, bastará con incluir el parámetro *log* al final de la sentencia.

ACL NOMBRADA

El segundo método disponible para la creación de listas de control de acceso, tanto estándar como extendidas, es el llevado a cabo mediante la asignación de un nombre en lugar de un ID. El objetivo, operación y resultado coinciden con el recién analizado, sin embargo, difieren en los siguientes aspectos:

- Son identificadas a través de un nombre, en lugar de un valor numérico.

- Los filtros son definidos desde el modo de configuración propio de la ACL.

- Resulta más sencillo editar las sentencias, pudiendo modificar o alterar el orden de estas.

El primer paso a realizar consiste en la creación de la ACL. Para ello se debe hacer uso del comando **access-list *[standard | extended] [nombre]***, ejecutado desde el modo de configuración global, donde *[standard | extended]* identifica el tipo de ACL deseada, mientras que *[nombre]* le asigna un identificativo a la misma.

```
Router(config)# ip access-list standard PRUEBA-ESTANDAR
Router(config-std-nacl)#

Router(config)# ip access-list extended PRUEBA-EXTENDIDA
Router(config-ext-nacl)#
```

Como se puede observar, una vez creada se accede a un modo de configuración propio desde el cual se configurarán los filtros necesarios.

Configuración de filtros en ACL nombradas

Durante el estudio de las ACL numeradas se podía observar como cada sentencia debe incluir el ID de ACL a la cual será agregada, haciendo uso para ello del comando *"access-list [ID]…".* Esta identificación resulta totalmente innecesaria en las nombradas, ya que, gracias al modo de configuración propio, el administrador opera directamente sobre la lista en cuestión, de tal modo que la sintaxis y parámetros necesarios para definir un filtro coincide en ambos modelos, exceptuando dicha identificación.

Un ejemplo podría ser el siguiente, donde se crea una ACL estándar, con los mismos filtros pero de manera numerada y nombrada.

```
---ACL estándar numerada---
Router(config)# access-list 1 permit 10.10.1.1
Router(config)# access-list 1 permit 192.168.2.3
```

```
---ACL estándar nombrada---
Router(config)# access-list standard PRUEBA-ESTANDAR
Router(config-std-nacl)# permit 10.10.1.1
Router(config-std-nacl)# permit 192.168.2.3
```

Exactamente lo mismo ocurre con aquellas extendidas, creada la ACL resulta innecesario identificarla en cada sentencia.

```
---ACL extendida numerada---
Router(config)# access-list 101 deny tcp host 10.10.1.1 10.0.0.0
0.255.255.255 eq 80
Router(config)# access-list 101 deny udp any any
Router(config)# access-list 101 deny tcp 192.168.1.0 0.0.0.255 any eq
telnet
Router(config)# access-list 101 permit ip any any

---ACL extendida nombrada---
Router(config)# access-list extended PRUEBA-EXTENDIDA
Router(config-ext-nacl)# deny tcp host 10.10.1.1 10.0.0.0
0.255.255.255 eq 80 Router(config-ext-nacl)# deny udp any any
Router(config-ext-nacl)# deny tcp 192.168.1.0 0.0.0.255 any eq telnet
Router(config-ext-nacl)# permit ip any any
```

Una vez definida, bastará con aplicarla en la interfaz y dirección necesaria haciendo uso del comando ***ip access-group [nombre] [in|out]***:

```
Router(config)# int Fa0/0
Router(config-if)# ip access-group PRUEBA-EXTENDIDA out
```

Edición de ACL ya creadas

Una de las grandes ventajas que ofrece este método de configuración consiste en la posibilidad de controlar el orden de las sentencias, permitiendo definir la posición que ocupará un determinado filtro a la hora de crearlo. Por ejemplo, imagina que resulta necesario permitir el tráfico de la IP 10.10.1.2 en la ACL estándar recién configurada, debiendo ocupar el lugar más alto de la lista.

La primera acción a llevar a cabo consiste en comprobar el orden actual de los filtros, objetivo que se logra ejecutando un *show ip access-list [nombre]*.

```
Router# show ip access-list PRUEBA-ESTANDAR
Standard IP access list PRUEBA-ESTANDAR
        10 permit 10.10.1.1
        20 permit 192.168.2.3
```

Los números 10 y 20 hacen referencia a la posición de cada uno de ellos dentro de la ACL, los cuales han sido asignados automáticamente por IOS en relación con el orden de configuración. Sin embargo, se ha solicitado habilitar el tráfico de una

determinada dirección y además ubicarla en la primera posición. ¿Cómo lograrlo? Para ello, simplemente se debe indicar el lugar que ocupará, que en este caso, y para cumplir el objetivo, debe ser menor de 10.

```
Router(config)# access-list standard PRUEBA-ESTANDAR
Router(config-std-nacl)# 3 permit 10.10.1.2
```

De tal manera que:

```
Router# show ip access-list PRUEBA-ESTANDAR
Standard IP access list PRUEBA-ESTANDAR
        3  permit 10.10.1.2
        10 permit 10.10.1.1
        20 permit 192.168.2.3
```

Si fuera necesario mover una sentencia, por ejemplo, posicionar la número 20 en segundo lugar, primero debe ser eliminada haciendo uso del comando **no *[número sentencia]***, para posteriormente volver a crearla indicando el nuevo lugar que ocupará en la lista.

```
Router(config)# access-list standard PRUEBA-ESTANDAR
Router(config-std-nacl)# no 20
Router(config-std-nacl)# 5 permit 192.168.2.3

Router# show ip access-list PRUEBA-ESTANDAR
Standard IP access list PRUEBA-ESTANDAR
        3  permit 10.10.1.2
        5  permit 192.168.2.3
        10 permit 10.10.1.1
```

Ejemplo 1: En relación con la topología definida en la *Fig. 7-2,* configurar una ACL con nombre “BLOQUEO-MSN” que cumpla los siguientes requisitos:

- Descartar la comunicación de todos los hosts pertenecientes a la red 192.168.1.0/24 hacia cualquier dirección de destino a través del rango de puertos TCP 6891-6900. Esta sentencia debe ocupar la posición número 1 en la lista.
- Descartar el tráfico de cualquier host ubicado en la red 192.168.1.0/24 hacia la dirección de destino 10.255.255.1 a través del puerto TCP 8080. Esta sentencia debe ocupar la posición número 2 en la lista.
- Permitir cualquier otro tipo de comunicación. Esta sentencia debe ocupar la posición 50 en la lista.

¿Dónde aplicarla? Se solicita bloquear una serie de puertos, por lo tanto se hace necesaria una ACL extendida, la cual debe estar ubicada lo más cercano posible al origen. Dicho origen es la red 192.168.1.0/24, conectada a R1 a través de su interfaz Fa0/1, siendo la dirección por la que fluye el tráfico de entrada.

```
R1(config)# access-list extended BLOQUEO-MSN
R1(config-ext-nacl)# 1 deny tcp 192.168.1.0  0.0.0.255 any range 6891 6900
R1(config-ext-nacl)#  2  deny  tcp  192.168.1.0   0.0.0.255  host  10.255.255.1  eq
8080
R1(config-ext-nacl)# 50 permit ip any any
R1(config-ext-nacl)# exit
R1(config)# int Fa0/1
R1(config-if)# ip access-group BLOQUEO-MSN in
```

SEGURIDAD DE ACCESO Y SERVICIOS VULNERABLES

Las ACL representan uno de los elementos más importantes en cuanto a seguridad de red se refiere, sin embargo, también se deben tener en cuenta otros aspectos con el fin de asegurar tanto el acceso al dispositivo como los servicios ejecutados en el mismo. En capítulos anteriores han sido analizadas diferentes configuraciones relativas a ello, como el cifrado de contraseñas o ssh. La presente sección estará dedicada a identificar aquellos servicios vulnerables, asegurar el acceso a través de las líneas vty mediante listas de control de acceso y analizar el protocolo NTP *(Network Time Protocol)* prestando especial atención a los beneficios que aporta su aplicación sobre entornos corporativos.

Servicios en routers y switchs

Todos los dispositivos incorporan una configuración por defecto, la cual incluye una serie de servicios que se mantendrán en ejecución siempre que esté operativo. Este simple hecho puede suponer una brecha de seguridad, ya que en muchas ocasiones representan protocolos vulnerables fácilmente explotables, teniendo como consecuencia el acceso no autorizado al dispositivo o la obtención de información de red por parte del atacante. Es por ello que una de las primeras acciones a llevar a cabo, en cuanto a seguridad se refiere, consiste en deshabilitar aquellos servicios innecesarios y asegurar aquellos otros que sí serán de utilidad.

Un claro ejemplo de ello es HTTP. IOS soporta una interfaz gráfica (GUI) desde la cual es posible modificar o verificar configuraciones, de la misma manera que si se hiciera a través de la CLI. Su acceso, por defecto, es mediante HTTP, por lo tanto, el dispositivo debe ejecutar dicho servicio. El problema radica en que este protocolo no cifra la comunicación entre cliente-servidor, por lo que si fuera interceptada el atacante podría obtener información valiosa y/o acceso como administrador. Para evitarlo es recomendable deshabilitar su ejecución mediante el comando **no ip http server,** y a su vez, habilitar HTTPS, haciendo uso de la sentencia **ip http secure-server,** ambos desde el modo de configuración global, logrando con ello que todas las conexiones a la GUI se lleven a cabo de manera cifrada. Su aplicación también

requiere crear un usuario con el máximo nivel de privilegios, sin embargo, esta configuración no es contenido de CCNA.

Otro de los servicios habilitados por defecto es CDP *(Cisco Discovery Protocol),* utilizado por switchs y routers Cisco para obtener información de dispositivos vecinos, como su dirección IP, interfaces o modelo. Resulta evidente que gracias al mismo cualquier atacante podría obtener información relevante de la red, por lo tanto, se desaconseja su ejecución. En caso de necesitar dichos datos y no poder obtenerlos por otros medios, se recomienda habilitarlo puntualmente y acto seguido volver a desactivarlo, aplicando para ello la sentencia **no cdp run,** desde el modo de configuración global, o **no cdp enable**, desde el modo de configuración de la interfaz implicada.

Por último, Cisco ofrece la posibilidad de deshabilitar de manera automática y a través de un solo comando una serie de servicios en general poco utilizados y de baja utilidad, denominados "*small servers*", tanto en tcp como udp. La manera de llevarlo a cabo es mediante la sentencia **no service tcp-small-servers** y **no service udp-small-servers** respectivamente, ambos ejecutados desde el modo de configuración global.

```
// Deshabilitar HTTP
Router(config)# no ip http server

// Deshabilitar CDP de manera global
Router(config)# no cdp run

// Deshabilitar CDP a nivel de interfaz
Router(config)# interface gigabitEthernet 0/0
Router(config-if)# no cdp enable

// Deshabilitar small servers...TCP y UDP
Router(config)# no service tcp-small-servers
Router(config)# no service udp-small-servers
```

Asegurar el acceso a través de las líneas VTY

Las líneas VTY son interfaces lógicas utilizadas por IOS para permitir el acceso remoto a la CLI, es decir, representan una vía directa hacia la configuración del dispositivo, a la cual solo debería poder acceder el personal autorizado a ello. Para lograrlo se hace necesaria la aplicación de una ACL que controle el tráfico sobre dichas líneas, permitiendo tan solo la conexión de los hosts definidos en ella. Su configuración consta de:

- *Paso 1:* Crear una ACL estándar, la cual permita el acceso a las direcciones IP deseadas.

- *Paso 2:* Aplicarla en las líneas VTY con el comando **access-class *[ID ACL]* in**, desde el modo de configuración de estas y donde *ID ACL* identifica aquella creada en el paso 1. En este caso siempre se define el parámetro *in*, ya que el objetivo es controlar las conexiones entrantes.

Ejemplo: El router "Central" representa un elemento crítico en la red, por lo que su acceso debe estar asegurado. Una de las configuraciones a llevar a cabo consiste en habilitar las conexiones remotas hacia la CLI a todos los hosts pertenecientes al departamento de informática, el cual se encuentra ubicado en la VLAN 50, subred 172.50.1.0/24. Cualquier otro intento de conexión será bloqueado. Ejecutar las acciones necesarias.

```
Central(config)# access-list 10 permit 172.50.1.0  0.0.0.255
Central(config)# line vty 0 15
Central(config-line)# access-class 10 in
```

Con ello, cualquier intento de conexión remota a la CLI que no pertenezca a la subred 172.50.1.0/24 será rechazado gracias al *deny* implícito definido por defecto al final de toda ACL.

Reto 7.4 – Como administrador de red has sido contratado por la empresa "Corp" para asegurar uno de sus routers. Una de las tareas encomendadas consiste en que solo puedan acceder remotamente a su configuración las direcciones IP 10.15.10.1 y 10.15.10.2. Realizar las acciones necesarias.

Solución al final del capítulo.

NTP (Network Time Protocol)

El protocolo NTP por sí mismo no representa ningún sistema de protección sobre el dispositivo, pero sí que resulta un complemento imprescindible a la hora de analizar la información generada en respuesta a diferentes eventos de seguridad, como la caída de una interfaz, accesos al dispositivo, coincidencias en ACL, etc. Para cada uno de dichos sucesos, IOS genera un mensaje log, los cuales normalmente son enviados a algún servidor en la red con el fin de almacenarlos de manera permanente y poder ser analizados en cualquier momento. Cada uno de estos mensajes contiene diferente información, entre la que se encuentra la fecha y hora exacta en la que ocurrió el evento.

Imagina que la red sufre un ataque y se debe investigar qué ha ocurrido y cuándo sucedió. Dicha información se obtiene gracias a la lectura y análisis de los diferentes logs generados por los dispositivos implicados. Ahora imagina que cada uno de ellos tiene configurada una hora diferente, o simplemente no coinciden por minutos o incluso segundos. Este hecho dará como resultado que los logs difieran en el tiempo, resultando mucho más complicado definir el ataque y mitigarlo.

Debido a ello, se debe lograr que todos los elementos de red dispongan exactamente la misma hora, tarea que resulta prácticamente imposible mediante la configuración manual en cada dispositivo.

Con dicho objetivo nace NTP *(Network Time Protocol)*, un protocolo bastante sencillo pero tremendamente útil basado en el modelo cliente-servidor, donde el servidor establece la fecha y hora y los clientes la aplican sobre sí mismos, logrando con ello que todos dispongan exactamente la misma configuración.

El comando necesario en IOS para configurar un dispositivo como cliente es **ntp server *[IP servidor ntp]*** desde el modo de configuración global.

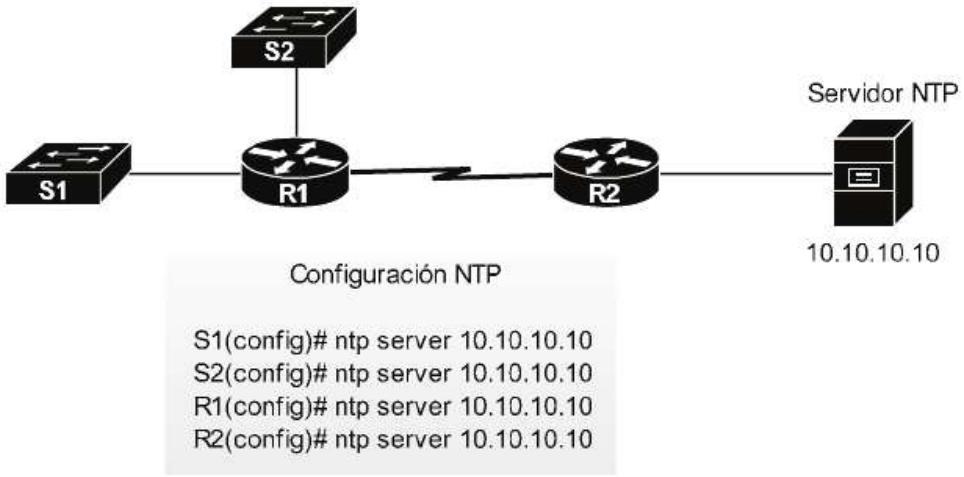


Fig. 7-4 Configuración de NTP.

Una vez aplicado es posible verificar su correcto funcionamiento gracias a los comandos *show ntp status* o *show ntp associations*, mostrando información sobre el servidor utilizado para la sincronización.

NAT (NETWORK ADDRESS TRANSLATION)

Actualmente, Internet representa la red WAN por excelencia a nivel mundial. Sus orígenes constan de un modelo de comunicación basado en el protocolo IPv4 donde cada dispositivo disponía de una dirección pública para acceder a ella. Con el paso del tiempo, y debido al gran auge que experimentó la red, dichas direcciones se fueron agotando, convirtiéndose en un sistema muy poco escalable y totalmente inviable a día de hoy. Como soluciones a ello nacen NAT e IPv6. A continuación se procederá al estudio y configuración de NAT en routers Cisco, mientras que IPv6 será objeto de análisis en el capítulo 11 "*IP Versión 6*".

Modo de operar

NAT puede ser definido como una solución a corto y medio plazo que nace con el propósito de evitar el desgaste de direcciones en redes públicas, basándose para ello en la asignación de la misma IPv4 sobre múltiples dispositivos de la misma red privada. Este nuevo modelo, implementado actualmente en el 99% de hogares y compañías, consiste en la asignación, por parte del ISP, de una o varias direcciones IP públicas, que serán configuradas en la interfaz del router que conecta con la WAN. Este, a su vez, también pertenece a la LAN y será el encargado de permitir la comunicación entre ambos tipos de redes, aplicando las direcciones adecuadas sobre cada entorno. Para ser más exactos, la función de NAT consiste en ejecutar una traducción de direcciones, de privadas a públicas y viceversa. Su modo de operar consiste en:

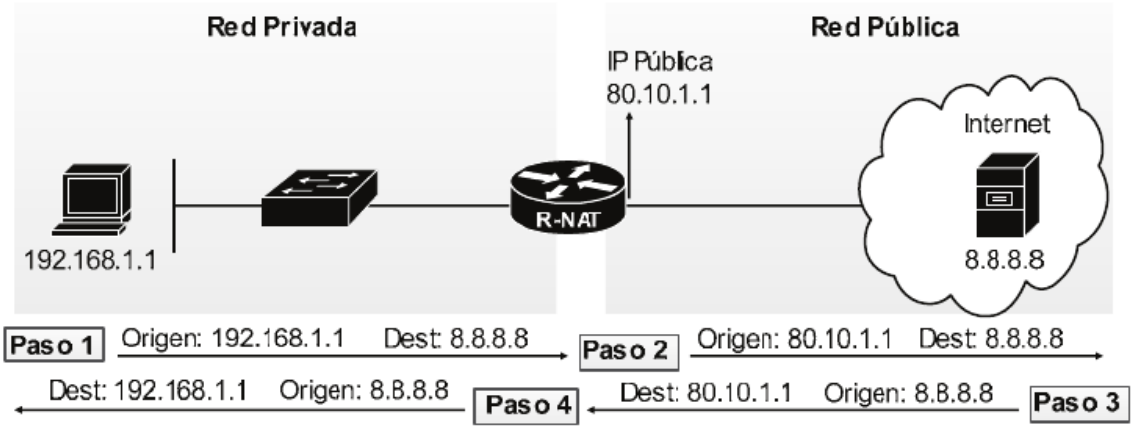


Fig. 7-5 NAT. Proceso de traducción de direcciones.

- *Paso 1:* Un host de la red privada desea comunicarse con un servidor ubicado en Internet. Para ello crea un paquete en capa 3 con su propia IP como origen (privada) y la del servidor como destino (pública), para luego enviarlo a su puerta de enlace, que en este caso corresponde a R-NAT.

- *Paso 2:* El router lo recibe, analiza la dirección de destino y comprueba que debe reenviarlo a través de Internet. Para ello crea un nuevo paquete utilizando como origen la IP pública configurada en su interfaz WAN, acto seguido lo reenvía a través de la misma.

- *Paso 3:* El paquete es recibido por el servidor, el cual responde directamente a R-NAT, más concretamente a su dirección pública, la 80.10.1.1.

- *Paso 4:* El router recibe el paquete, examina su tabla NAT y comprueba que la comunicación fue iniciada por el host 192.168.1.1, por lo cual crea un nuevo paquete y lo reenvía hacia la red privada, utilizando como destino la IP 192.168.1.1.

El mismo proceso se aplicaría con cualquier otro dispositivo de la red privada que desee comunicarse a través de Internet. NAT implica una serie de ventajas pero también algunos inconvenientes. Como ventajas cabe destacar su simplicidad de uso, ahorro de direcciones, administración centralizada de una o más direcciones públicas y mayor seguridad sobre la red privada. Sin embargo, el rendimiento del router disminuye en función del número de dispositivos que hagan uso simultáneo; a mayor número de estos, menor rendimiento. También se pierde capacidad de rastreo en conexiones de extremo a extremo.

En routers Cisco es posible configurar tres tipos de NAT. Estático, dinámico y con sobrecarga (PAT).

NAT ESTÁTICO

Este modelo es el más sencillo y consiste en mapear una dirección pública con otra privada, de tal manera que el uso de la primera queda reservado exclusivamente para un dispositivo de la LAN.

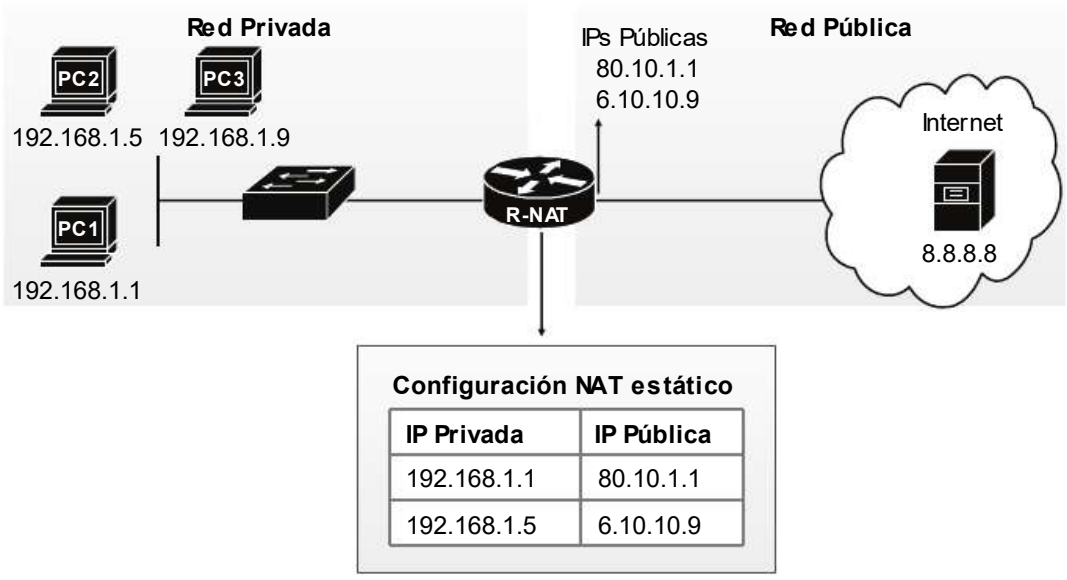


Fig. 7-6 NAT estático.

R-NAT ha sido configurado con NAT estático, mapeando las direcciones 192.168.1.1 y 192.168.1.5 a dos IPv4 públicas. El resultado será que PC1 accederá a Internet a través de la IP 80.10.1.1, mientras que PC2 lo hará mediante la 6.10.10.9. Por su contra, PC3 no podrá acceder a la red pública ya que no se le ha asignado ninguna dirección para ello.

En NAT estático los mapeos son "uno a uno", es decir, una misma IP pública no podrá ser mapeada con más de una privada. En este caso, para que PC3 obtenga acceso a Internet se deberá contratar otra dirección al ISP o bien configurar otro tipo de NAT.

NAT DINÁMICO

El modelo dinámico consiste en configurar un conjunto (*pool*) de direcciones públicas que podrán ser utilizadas por cualquier dispositivo de la red privada pero nunca de manera simultánea. La principal diferencia entre el modelo estático y el dinámico es que en el primero se establecen mapeos "uno a uno" permanentes, mientras que en el segundo estos mapeos, que también son "uno a uno", resultan temporales.

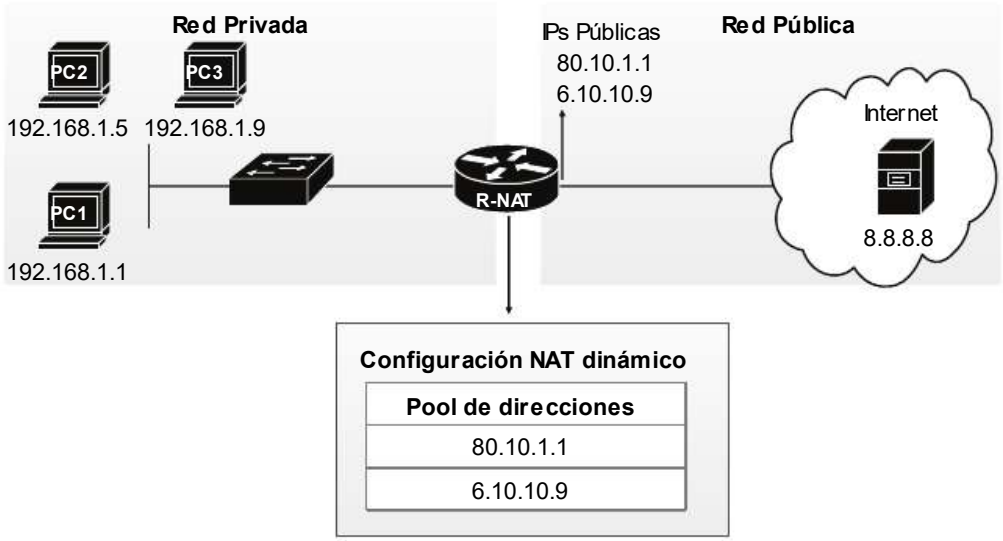


Fig. 7-7 NAT dinámico.

PC1 y PC2 acceden a Internet y R-NAT les asigna las direcciones 80.10.1.1 y 6.10.10.9 respectivamente. Si en este momento PC3 intenta conectar con la red pública no podrá hacerlo ya que las dos IPs del pool configurado están en uso. Pasado un tiempo, PC1 finaliza su comunicación en Internet y PC3 lo vuelve a intentar, en este caso sí obtendrá acceso ya que una dirección ha quedado disponible.

NAT CON SOBRECARGA O PAT

El problema principal en NAT estático y dinámico consiste en que ambos métodos establecen mapeos "uno a uno", resultando imposible el uso simultáneo de la misma IP pública por más de un dispositivo de la red privada. Este hecho, en entornos corporativos, representa un modelo poco escalable, ya que sería demasiado complicado y costoso garantizar el acceso a Internet a todos los dispositivos.

Con el fin de solucionarlo nace NAT con sobrecarga, también denominado PAT *(Port Address Translation)* gracias al cual una o varias IPs públicas pueden ser utilizadas de manera simultánea por multitud de dispositivos. Para lograrlo, el router establece un mapeo para cada cliente de la red privada, asignando sobre cada sesión un número de puerto diferente, los cuales serán utilizados para identificar cada una de las comunicaciones llevadas a cabo hacia la red pública.

Un ejemplo podría ser el siguiente:

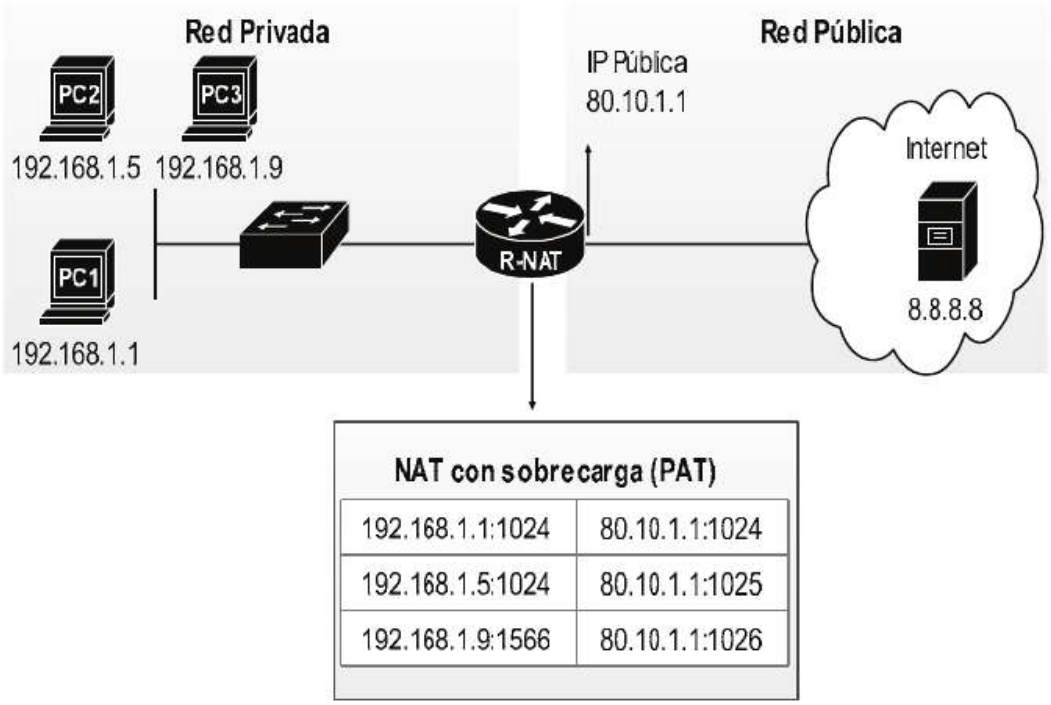


Fig. 7-8 PAT.

Donde R-NAT tan solo dispone de una IP pública y aplica PAT para brindar acceso a Internet a todos los dispositivos de la red privada. Imagina que los tres PCs acceden al mismo servidor de manera simultánea y dos de ellos mediante el mismo puerto de origen (1024). Cuando el router reciba las respuestas, ¿cómo determina a qué dispositivo de la red privada debe redirigir cada una de ellas? Para lograrlo, PAT modifica el puerto de origen de cada comunicación que debe ser reenviada a través de la WAN, y a su vez lo asocia con el dispositivo de la red privada que inició el proceso. De esta manera, dependiendo del puerto de respuesta utilizado por el servidor, el router identifica el dispositivo de la red privada al cual debe ser reenviado. Por ejemplo, los paquetes recibidos desde Internet con puerto 1025 serán redirigidos al socket 192.168.1.5:1024.

Imagina que la IP 8.8.8.8 corresponde a un servidor Web, dando servicio a través del puerto 80, y que los 3 PCs de la red privada acceden él. El destino coincide, 8.8.8.8:80, mientras que el origen varía dependiendo del PC que inicia la comunicación.

La primera acción que lleva a cabo el router al recibir los paquetes consiste analizar su origen, para luego definir y asociar un número de puerto a cada uno de ellos. Dicha asociación es almacenada en una tabla ubicada en la memoria interna del dispositivo, la cual será analizada cada vez que se reciba o envíe un paquete. Para este ejemplo, R-NAT ha generado la siguiente tabla:

Origen red privada	Puerto público
192.168.1.1:1024	1024
192.168.1.5:1024	1025
192.168.1.9:1566	1026

Tras ello, y con el fin de enviarlos a través de la WAN, el router modifica la dirección y puerto de origen de los paquetes. Como dirección hace uso de la misma IP pública, y como puerto aplica el asociado a cada dispositivo en la tabla generada. Acto seguido los paquetes son enviados al medio y recibidos por el servidor:

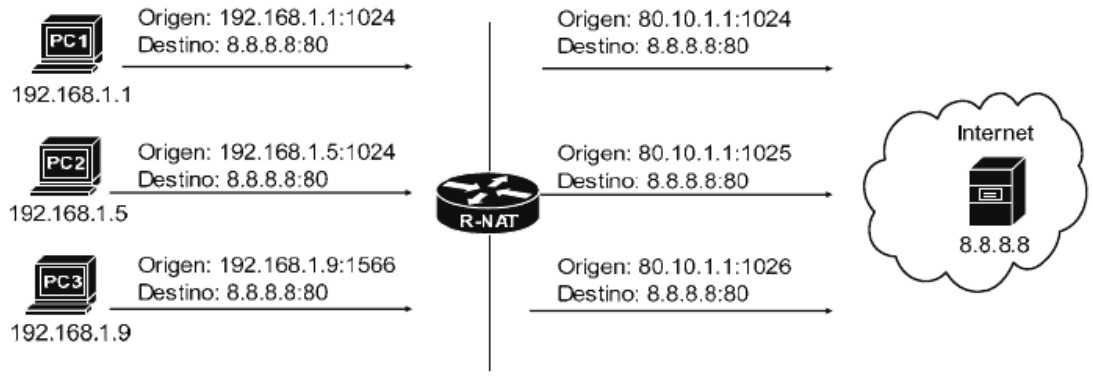


Fig. 7-9 PAT. Traducción de direcciones y puertos LAN-WAN.

El cual responderá directamente al router:

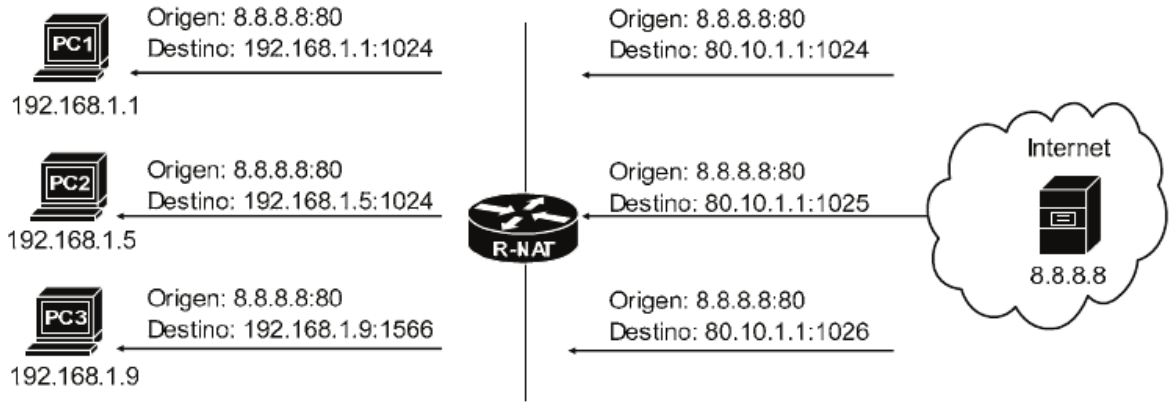


Tabla PAT	
192.168.1.1:1024	1024
192.168.1.5:1024	1025
192.168.1.9:1566	1026

Fig. 7-10 PAT. Traducción de direcciones y puertos WAN-LAN.

Para cada paquete de respuesta, R-NAT analiza el puerto de destino y consulta su tabla PAT para determinar la dirección privada a la que debe ser reenviado:

- Para el puerto 1024, R-NAT creará un paquete con destino 192.168.1.1:1024, y lo reenviará a la red privada, siendo recibido por PC1.

- Para el puerto 1025, creará un paquete con destino 192.168.1.5:1024, lo reenviará a la red privada y será recibido por PC2.

- Por último, para el puerto 1026, creará un paquete con destino 192.168.1.9:1566, y lo reenviará a la red privada, siendo recibido por PC3.

Un detalle a tener en cuenta es que la numeración de puertos definida por PAT se lleva a cabo de manera continua, es decir, desde que es asignado el primer valor (en este caso 1024) los siguientes son creados simplemente sumando 1.

La función principal de NAT, sea cual sea el modelo aplicado, consiste en la traducción de direcciones con el fin de permitir la comunicación entre dispositivos ubicados en redes públicas y privadas. Durante el proceso llevado a cabo, las direcciones y puertos varían, es por ello que el protocolo define una serie de términos con el fin de identificar cada uno de ellos, siendo los siguientes:

Dirección	**Definición**
Inside Local	Hace referencia a la IP privada del host de origen que inicia la comunicación. También denominada "*Inside private*".
Inside Global	Identifica la dirección pública que asigna NAT al host de la red privada para su comunicación a través la WAN. También denominada "*Inside public*".
Outside Global	Representa la dirección pública del host de destino utilizada para la comunicación a través de la WAN. También denominada "*Outside public*" (ver ejemplo).
Outside Local	Representa la dirección pública del host de destino definida por el host de la red privada. También denominada "*Outside private*". Normalmente esta coincide con la Outside Global (ver ejemplo).

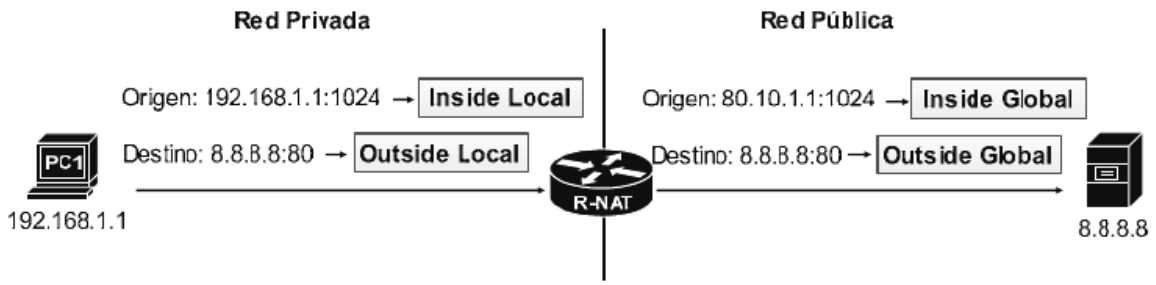


Fig. 7-11 PAT. Tipos de direcciones.

Configuración de NAT estático

NAT estático consiste en mapear una dirección privada con otra pública de manera exclusiva y permanente, por lo tanto, el objetivo principal consiste en identificar ambas IPs y aplicarlo en las interfaces correctas. Para ello se deben ejecutar las siguientes acciones:

- *Paso 1:* Establecer el mapeo de las direcciones privada y pública con el comando **ip nat inside source static *[IP privada] [IP pública]*** desde el modo de configuración global.

- *Paso 2:* Definir la interfaz de salida de NAT, que debe ser aquella que conecta directamente con la red pública. Para ello se debe aplicar la sentencia **ip nat outside** desde el modo de configuración de la propia interfaz.

- *Paso 3:* Identificar la interfaz de entrada de NAT, que será aquella que conecta con la red privada en la cual reside el host definido en el paso 1. Aplicar sobre la misma el comando **ip nat inside**.

Ejemplo: Configurar R1 para que el servidor con IP 172.20.0.1 utilice de manera exclusiva y permanente la dirección pública 12.12.12.12 cuando requiera acceder a la WAN.

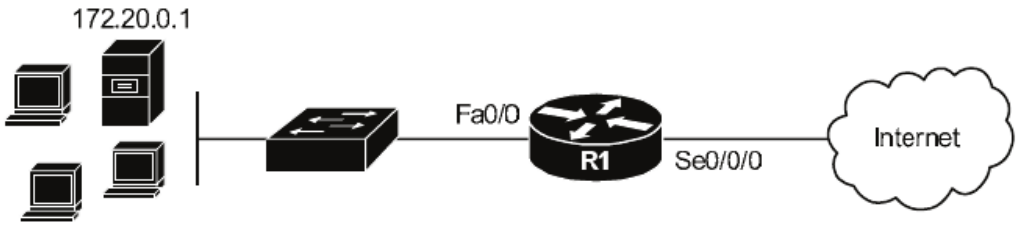


Fig. 7-12 Diseño de red para supuesto práctico de NAT estático.

```
R1(config)# ip nat inside source static 172.20.0.1 12.12.12.12

R1(config)# interface Se0/0/0
R1(config-if)# ip nat outside

R1(config)# interface Fa0/0
R1(config-if)# ip nat inside
```

Configuración de NAT dinámico

El objetivo en NAT dinámico consiste en definir un conjunto de direcciones públicas que serán utilizadas para dar acceso a la WAN a los hosts de la red privada, sin embargo, la misma IP no podrá ser asociada de manera simultánea a más de un dispositivo. Su configuración consta de:

- *Paso 1:* Definir el pool de direcciones con el comando **ip nat pool** ***[nombre] [IP inicio] [IP fin]*** **netmask** ***[máscara]*** desde el modo de configuración global.

- *Paso 2:* Crear una ACL para identificar qué hosts podrán acceder a la red pública. Normalmente bastará con una estándar.

- *Paso 3:* Vincular dicha ACL a NAT. El comando necesario para ello es **ip nat inside source list** ***[num acl]*** **pool** ***[nombre pool]*** desde el modo de configuración global y donde *[num acl]* hace referencia al ID creado en el paso 2 y *[nombre pool]* a aquel definido en el paso 1.

- *Paso 4:* Identificar la interfaz de salida de NAT con el comando **ip nat outside** desde el modo de configuración de la propia interfaz.

- *Paso 5:* Definir la interfaz de entrada de NAT mediante el comando **ip nat inside** desde el modo de configuración de dicha interfaz.

Ejemplo: Una compañía dispone de un pool de direcciones públicas que comprende el rango 8.8.8.1 - 8.8.8.6, con máscara /29 (datos facilitados por el ISP). La organización desea utilizarlas para que los hosts de la red 172.20.0.0/16 accedan a Internet. Sin embargo, y para no sobrecargar el router, cada IP solo puede ser asociada a un dispositivo de manera simultánea.

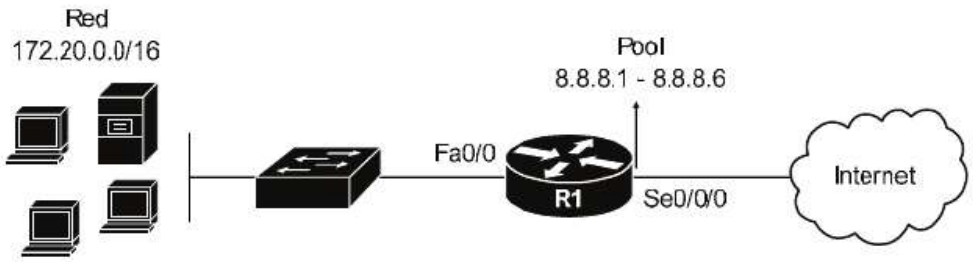


Fig. 7-13 Diseño de red para supuesto práctico de NAT dinámico.

```
R1(config)# ip nat pool INTERNET 8.8.8.1  8.8.8.6 netmask
255.255.255.248

R1(config)# access-list 10 permit 172.20.0.0  0.0.255.255

R1(config)# ip nat inside source list 10 pool INTERNET

R1(config)# interface Se0/0/0
R1(config-if)# ip nat outside

R1(config)# interface Fa0/0
R1(config-if)# ip nat inside
```

Configuración de NAT con sobrecarga o PAT

Por último, NAT con sobrecarga logra la utilización simultánea de la misma IP pública por múltiples dispositivos de la red privada. Su configuración se basa en:

Paso 1: Crear una ACL, normalmente estándar, para definir qué hosts pueden acceder a la WAN.

Paso 2: Vincular dicha ACL a NAT, a través de la sentencia **ip nat inside source list *[num acl]* interface *[interfaz de salida]* overload** desde el modo de configuración global y donde *[num acl]* identifica el ID de esta (paso 1) e interfaz de salida hace referencia a aquella que conecta con la red pública. El parámetro *overload* aplica NAT con sobrecarga (PAT).

Paso 3: Definir la interfaz de salida ejecutando el comando **ip nat outside** sobre el modo de configuración de la propia interfaz.

Paso 4: Identificar la interfaz de entrada haciendo uso del comando **ip nat inside** desde el modo de configuración de dicha interfaz.

Ejemplo: Una compañía dispone de la dirección pública 80.80.80.80/8 y desea que todos los hosts de la red privada 172.20.0.0/16 puedan acceder de manera simultánea a Internet. Configurar R1 para lograr dicho objetivo.

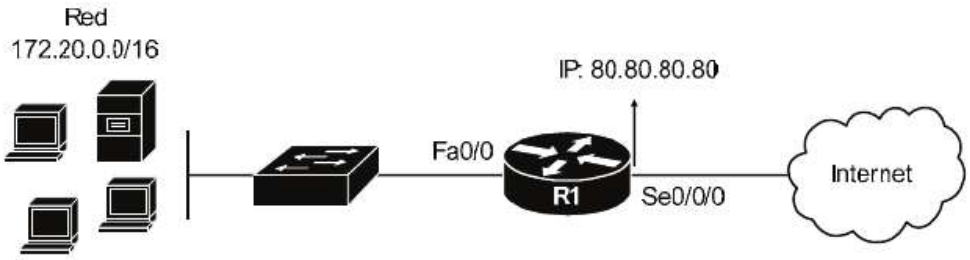


Fig. 7-14 Diseño de red para supuesto práctico de PAT.

```
R1(config)# access-list 10 permit 172.20.0.0  0.0.255.255

R1(config)# ip nat inside source list 10 interface Se0/0/0 overload

R1(config)# interface Se0/0/0
R1(config-if)# ip nat outside

R1(config)# interface Fa0/0
R1(config-if)# ip nat inside
```

También es posible configurar NAT con sobrecarga haciendo uso de un pool de direcciones públicas. Para lograrlo bastará con ejecutar el mismo procedimiento que en NAT dinámico, agregando el parámetro *overload* al final del comando "*ip nat inside source list [num acl] pool [nombre pool] overload*".

Resolución de problemas en NAT

En la mayor parte de los casos, las incidencias relacionadas con NAT son debidas a errores de configuración. Una buena práctica a llevar a cabo si los resultados no fueran los deseados consiste en verificar los siguientes aspectos:

- Interfaces aplicadas en NAT: Un error bastante común es la asignación de estas de manera inversa, es decir, la red privada como *outside* y la pública como *inside*.

- En NAT estático, verificar los mapeos de direcciones privadas - públicas.

- En NAT dinámico, comprobar el pool de direcciones y la ACL asociada.

- En NAT con sobrecarga, verificar la ACL y que el parámetro *overload* ha sido agregado.

- Asegurar que ninguna otra ACL aplicada sobre las interfaces que operan con NAT bloquea su tráfico. El router, antes de llevar a cabo la traducción de direcciones, ejecuta dichas ACL.

Además, IOS dispone de los siguientes comandos de verificación:

show ip nat translations: Genera un listado con los mapeos de direcciones que ejecuta NAT actualmente.

show ip nat statistics: Muestra en pantalla datos estadísticos del protocolo, como el número total de traducciones, interfaces utilizadas, IP o rango de IPs públicas en uso, mapeos etc.

SOLUCIÓN DE RETOS: SEGURIDAD EN CAPA 3

Reto 7.1 – Conforme a la misma topología, configurar dos ACL estándar numeradas cumpliendo los siguientes requisitos:
- ACL1: Impedir el acceso de la red 192.168.20.0/24 y del host 192.168.30.150 al destino 192.168.1.0/24. Permitir cualquier otro tipo de tráfico.
- ACL2: Bloquear toda comunicación de la red 172.20.0.0/16 hacia cualquier destino.

```
---ACL 1---
R1(config)# ip access-list 1 deny 192.168.20.0  0.0.0.255
R1(config)# ip access-list 1 deny host 192.168.30.150
R1(config)# ip access-list 1 permit any
R1(config)# int Fa0/1
R1(config-if)# ip access-group 1 out

---ACL 2---
R1(config)#ip access-list 2 deny 172.20.0.0  0.0.255.255
R1(config)# int Fa0/0
R1(config-if)# ip access-group 2 in
```

Los números de ACL pueden variar, siempre y cuando no coincidan en ambas y a su vez formen parte del rango 1-99 o 1300-1999.

Reto 7.2 – Haciendo uso de la misma topología. Se han detectado diferentes ataques procedentes de la subred 192.168.1.0/24 al servidor FTP de la compañía, ubicado en la VLAN20 con IP 192.168.20.15. Bloquear cualquier intento de conexión

procedente de dicha subred a los puertos TCP 20 y 21 del servidor FTP. Cualquier otro tipo de tráfico debe ser permitido.

```
R1(config)# ip access-list 110 deny tcp any host 192.168.20.15 eq 20
R1(config)# ip access-list 110 deny tcp any host 192.168.20.15 eq 21
R1(config)# ip access-list 110 permit ip any any
R1(config)# int Fa0/1
R1(config-if)# ip access-group 110 in
```

El número de ACL puede variar, siempre y cuando forme parte del rango 100-199 o 2000-2699.

Reto 7.3 – Conforme a la misma topología. Los dispositivos de la red 192.168.1.0/24 tan solo deben disponer de acceso a los siguientes servicios:

1.- Al servidor DNS cuya IP es 192.168.10.250.
2.- Al servidor de correo POP3 y SMTP con dirección 192.168.10.251.
3.- Al servidor web de la empresa con IP 192.168.10.252.
4.- Cualquier otro tipo de comunicación para esta subred debe ser bloqueado.

Configurar una ACL en el router que proceda para lograr el resultado deseado.

```
R1(config)#ip  access-list  120  permit  tcp  192.168.1.0    0.0.0.255    host
192.168.10.250 eq 53
R1(config)#ip  access-list  120  permit  tcp  192.168.1.0    0.0.0.255    host
192.168.10.251 eq 110
R1(config)#ip  access-list  120  permit  tcp  192.168.1.0    0.0.0.255    host
192.168.10.251 eq 25
R1(config)#ip  access-list  120  permit  tcp  192.168.1.0    0.0.0.255    host
192.168.10.252 eq 80
R1(config)#int fa0/1
R1(config-if)# ip access-group 120 in
```

El número de ACL puede variar, siempre y cuando forme parte del rango 100-199 o 2000-2699.

Reto 7.4 – Como administrador de red has sido contratado por la empresa “Corp” para asegurar uno de sus routers. Una de las tareas encomendadas consiste en que solo puedan acceder remotamente a su configuración las direcciones IP 10.15.10.1 y 10.15.10.2. Realizar las acciones necesarias.

```
Router(config)# ip access-list 20 permit host 10.15.10.1
Router(config)# ip access-list 20 permit host 10.15.10.2
Router(config)# line vty 0 15
Router(config-line)# ip access-class 20 in
```

El número de ACL puede variar, siempre y cuando forme parte del rango 1-99 o 1300-1999.

TEST CAPÍTULO 7: SEGURIDAD EN CAPA 3

1.- ¿Qué tráfico es analizado por una ACL de entrada?

A. Aquel generado y reenviado por el router a través de una o varias interfaces.
B. Aquel recibido por el router a través de una o varias interfaces.
C. Todo tipo de tráfico que atraviese una interfaz.
D. Ninguna de las anteriores.

2.- ¿En qué modo de configuración se crea una ACL?

A. Modo de configuración global.
B. Modo de configuración de la interfaz donde será aplicada.
C. Modo privilegiado.
D. Ninguna de las anteriores.

3.- Revisando el fichero startup-config de un router se observa una ACL con ID 1315. ¿Qué significado tiene?

A. Que la ACL contiene 1315 entradas.
B. Que se trata de una ACL extendida.
C. Que se trata de una ACL estándar.
D. Que la ACL ha filtrado 1315 paquetes.

4.- Una ACL contiene una sola entrada que deniega cualquier paquete de la red 192.168.1.0. ¿Qué sucederá con el tráfico restante?

A. Será descartado.
B. Será permitido.
C. Será descartado, pero se informará al origen.
D. Dependerá de la dirección de destino.

5.- La siguiente ACL ha sido aplicada en dirección de entrada en la interfaz Fa0/0. ¿Qué acción llevará a cabo el router cuando reciba un paquete con IP 192.168.50.50 a través de la misma?

```
access-list 1 permit 192.168.50.50
access-list 1 deny 192.168.50.50
```

A. Permitirlo.
B. Denegarlo.
C. Permitirlo, pero a la hora de reenviarlo será descartado.
D. Denegarlo e informar al origen.

6.- ¿Qué tipo de filtrado permite una ACL estándar? (Seleccionar dos respuestas)

A. Dirección IP de origen.
B. Dirección IP de destino.
C. Red de origen.
D. Red de destino.
E. Puerto de origen.
F. Puerto de destino.

7.- ¿Qué rangos pueden ser utilizados para configurar una ACL extendida numerada? (Seleccionar dos respuestas)

A. 1-99.
B. 1300-1999.
C. 100-199.
D. 2000-2699.

8.- ¿En qué capa del modelo OSI opera una ACL estándar?

A. Capa 1.
B. Capa 2.
C. Capa 3.
D. Capa 4.

9.- ¿Qué acción llevará a cabo la siguiente ACL sobre un paquete con IP 10.10.255.254?

```
access-list 1 deny 10.10.0.0 0.0.255.255
access-list 1 permit any
```

A. Denegar.
B. Permitir.
C. La IP 10.10.255.254 corresponde a un broadcast y es descartada por el router sin necesidad de ACL.
D. Es denegada gracias al “deny” implícito ubicado al final de toda ACL.

10.- ¿Qué afirmación es correcta respecto a la sentencia mostrada a continuación?

```
access-list 1 permit any 192.168.1.0 0.0.0.255
```

A. Permite el tráfico desde cualquier origen hacia la red de destino 192.168.1.0 /24.

B. Permite el tráfico desde cualquier origen hacia la red de destino 192.168.1.0 /8.
C. Permite el tráfico desde cualquier origen hacia la red de destino 192.168.1.0 /16.
D. No es posible crear una ACL estándar de estas características.

11.- ¿Dónde debe ser configurada una ACL estándar?

A. En el router e interfaz más cercano a la red de origen.
B. En el router e interfaz más cercano a la red de destino.
C. En cualquier router de la red.
D. En cualquier router de la red ubicado entre el origen y el destino.

12.- Dada la siguiente ACL:

```
access-list 1 permit 172.20.10.100
access-list 1 deny 172.20.10.0  0.0.0.255
access-list 1 permit any
```

Se debe agregar una nueva sentencia que bloquee el tráfico de la subred 192.168.1.0/24. ¿Cuál de las siguientes opciones representa la manera correcta de proceder?

A. Aplicar una nueva sentencia que bloquee dicho tráfico con el comando "*access-list 1 deny 192.168.1.0 0.0.0.255*".
B. Eliminar el filtro "*access-list 1 permit any*", para luego agregar la nueva sentencia "*access-list 1 deny 192.168.1.0 0.0.0.255*".
C. Eliminar el filtro "*access-list 1 permit any*", agregar la sentencia "*access-list 1 deny 192.168.1.0 0.0.0.255*" y volver a crear el "*access-list 1 permit any*".
D. Con una ACL estándar no es posible realizar el filtrado que se solicita, se hace necesaria una ACL extendida.

13.- Se ha aplicado una ACL sobre la interfaz Fa0/0 mediante el comando "ip access-class 1 in", sin embargo, no se está llevando a cabo el filtrado deseado. ¿Cuál de las siguientes causas puede ser la más probable?

A. La interfaz Fa0/0 está apagada.
B. La ACL 1 no ha sido configurada en el router.
C. El comando utilizado no aplica la ACL en interfaces FastEthernet.
D. La ACL está configurada con un "*permit any*" que acepta todo el tráfico y evita el filtrado.

14.- En una ACL extendida se debe bloquear cualquier comunicación cuyo puerto de destino sea mayor que 80. ¿Qué parámetro es necesario para ello?

A. eq 80.
B. ne 80.
C. it 80.
D. gt 80.

15.- Un administrador de red desea bloquear todo el tráfico FTP. ¿Qué puertos debe filtrar? (Seleccionar dos respuestas)

A. 20 – TCP.
B. 20 – UDP.
C. 21 – TCP.
D. 21 – UPD.

16.- Un administrador de red desea bloquear todo el tráfico telnet y permitir el SSH. ¿Qué acción debe llevar a cabo? (Seleccionar dos respuestas)

A. Bloquear el puerto TCP 23.
B. Permitir el puerto TCP 23.
C. Bloquear el puerto TCP 53.
D. Permitir el puerto TCP 53.
E. Bloquear el puerto TCP 22.
F. Permitir el puerto TCP 22.

17.- ¿Qué número de puerto está siendo filtrado en la siguiente ACL?

```
access-list 101 deny tcp 10.0.0.0 0.255.255.255 host 192.168.1.50 eq
pop3
```

A. 443.
B. 110.
C. 25.
D. 161.

18.- ¿Qué protocolo resulta imprescindible aplicar en la red con el fin de lograr ordenar cronológicamente los eventos de seguridad generados por diferentes dispositivos?

A. ACL.
B. SMNP.
C. SMTP.
D. NTP.

E. OSPF.
F. CDP.

19.- Una buena práctica para asegurar cualquier dispositivo consiste en:
A. Aplicar un firewall sobre él.
B. Deshabilitar los servicios que no serán utilizados.
C. Comprobar y analizar los logs mensualmente.
D. Ninguna de las anteriores.

20.- ¿Qué acción lleva a cabo la siguiente configuración?

```
R1(config)# access-list 1 permit 10.10.10.0 0.0.0.255
R1(config)# line vty 0 15
R1(config-line)# access-class 1 in
```

A. Permite el acceso remoto al dispositivo al rango de IPs 10.10.10.0 - 10.10.10.255.
B. Permite el acceso remoto al dispositivo al rango de IPs 10.10.10.0 - 10.10.10.254.
C. Permite el acceso remoto al dispositivo al rango de IPs 10.10.10.1 - 10.10.10.254.
D. Permite el acceso remoto al dispositivo a los hosts ubicados en la red 10.10.10.0 /16.

21.- La red privada de una compañía está compuesta por 500 hosts y tan solo dispone de 2 direcciones públicas para brindar acceso a Internet. ¿Qué solución se debe aplicar para que todos ellos puedan acceder a la red pública de manera simultánea?

A. NAT estático.
B. NAT dinámico.
C. PAT.
D. Contratar una dirección pública para cada dispositivo de la red privada.

22.- ¿En qué interfaz debe ser configurado el comando “ip nat outside”?

A. En aquella que conecta directamente con la red pública y a través de la cual se ejecutará la traducción de direcciones.
B. En aquella que conecta con la red privada y cuyos dispositivos accederán a la WAN.
C. En todas las interfaces configuradas con una IP pública.
D. Ninguna de las anteriores.

23.- ¿En qué consiste NAT estático?

A. En asignar una dirección pública a un dispositivo de la red privada para su acceso a la WAN.
B. En realizar la traducción de una dirección pública a otra privada y viceversa, las cuales no varían.
C. En establecer un mapeo entre una dirección pública y otra privada de manera exclusiva y permanente.
D. Todas las respuestas anteriores son correctas.

24.- ¿Qué máscara wildcard se debe configurar en una ACL para agrupar todos los hosts pertenecientes a la subred 172.20.20.0/25?

A. 0.0.0.255.
B. 0.0.255.255.
C. 0.0.0.63.
D. 0.0.0.127.

25.- ¿Qué ubicación resulta la idónea para configurar y aplicar una ACL extendida?

A. El router e interfaz más cercano a la red de origen.
B. El router e interfaz más cercano a la red de destino.
C. Cualquier router.
D. Cualquier router ubicado entre el origen y el destino definido en la ACL.

REDUNDANCIA EN PUERTAS DE ENLACE

8

CONCEPTO DE REDUNDANCIA

La redundancia puede ser definida como el método llevado a cabo para garantizar la continuidad de un servicio aun cuando uno o varios de sus componentes fallan o no operan con normalidad. Su función resulta de vital importancia, sobre todo en elementos críticos de cualquier tipo de infraestructura. Por ejemplo, un avión está compuesto por dos motores, si uno presenta errores o simplemente no es utilizado, puede continuar volando perfectamente haciendo uso del otro. En el ámbito de las comunicaciones, la redundancia puede y debe ser implementada en prácticamente la totalidad de sus componentes, desde múltiples enlaces para no perder conectividad de extremo a extremo hasta evitar problemas de hardware mediante la duplicidad de fuentes de alimentación o procesadores. Gracias a su aplicación a todos los niveles se hace posible lograr una de las mayores metas en cualquier red, que esta se mantenga operativa el 99,9% del tiempo, traduciéndose en mayor productividad para la compañía, mejor escalabilidad y menor tiempo durante la resolución de problemas.

Un ejemplo bastante sencillo de la diferencia entre una red no redundante (a nivel de enlaces entre switchs) y otra que sí lo es podría ser el siguiente:

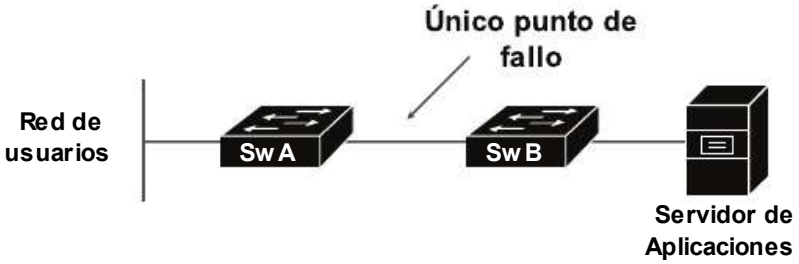


Fig. 8-1 Enlace no redundante.

Donde SwA y SwB conectan entre sí a través de un solo enlace, lo que representa un único punto de fallo *(single point of failure)*. En la práctica, si este cae, la comunicación entre la red de usuarios y el servidor de aplicaciones no podrá llevarse a cabo. Imagina que las operaciones dependen por completo de dicho servidor, el problema resulta evidente.

Una manera de solucionarlo consiste en implementar redundancia de enlace entre ambos dispositivos, tal que:

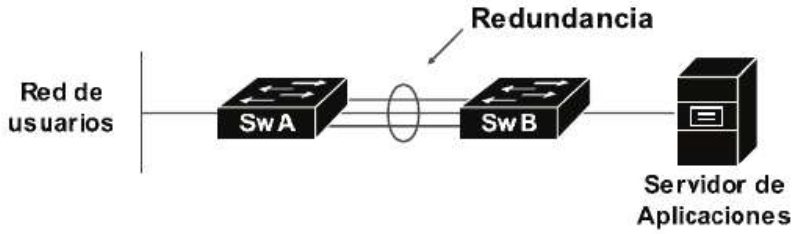


Fig. 8-2 Enlace redundante.

Gracias a lo cual, si alguno de los tres enlaces entre SwA y SwB cae, la comunicación entre la red de usuarios y el servidor de aplicaciones no se vería afectada. Incluso fallando un segundo, se mantendría operativa a través del link restante.

Durante el diseño de una topología resulta sumamente importante tener en cuenta y planificar la redundancia. En capítulos anteriores han sido analizados diferentes aspectos sobre un diseño de red eficiente, haciendo mención a las diferentes capas que este debe contemplar. Bien, la capa de acceso está compuesta por dispositivos finales y un fallo en cualquiera de ellos no supone la pérdida de disponibilidad en la red. Sin embargo, la caída de cualquier enlace físico que intervenga en la comunicación entre las diferentes capas tendrá como consecuencia problemas graves de conectividad, por lo que en este caso resulta imprescindible

aplicar sistemas redundantes. Además, en elementos críticos como switchs o routers también se hace necesaria a nivel de hardware.

Entre los diferentes tipos de redundancia podrían destacar los siguientes:

- La ya mencionada redundancia de enlaces.

- De dispositivos: Si un router o switch falla, es posible acceder al destino a través de otro.

- De servicios: En servidores críticos como DHCP, DNS, Active Directory o aplicaciones, la redundancia consiste en que varios desarrollen exactamente la misma función y a su vez se mantengan sincronizados, de tal manera que, si uno cae, el servicio se mantiene operativo a través de otro.

- De hardware: Cualquier elemento físico del dispositivo que pueda ser duplicado, como tarjetas de red, fuentes de alimentación, procesadores, almacenamiento, etc.

- De puerta de enlace: Si el router que actúa como *gateway* para una determinada red cae, otra toma su rol inmediatamente, evitando con ello la pérdida de conectividad de los hosts ubicados en dicha red.

De todos ellos, los más importantes para el examen de CCNA son la redundancia de enlaces, de la que ya se ha hecho mención en capítulos y párrafos anteriores, y la redundancia en puertas de enlace, analizada a continuación.

El término informático "puerta de enlace" (o *gateway*) simplemente hace referencia a una dirección IP, normalmente configurada en un router y utilizada por los dispositivos para enviar cualquier comunicación con destino hacia una red diferente a la propia, es decir, remota. La acción más habitual consiste en que todos los hosts pertenecientes a una misma subred hagan uso del mismo *gateway*, por lo que su función es considerada crítica y por consiguiente muy recomendable aplicar sobre la misma tolerancia a fallos.

Con dicho objetivo nacen diferentes protocolos, como HSRP *(Hot Standby Router Protocol),* VRRP *(Virtual Router Redundancy Protocol)* y GLBP *(Gateway Load Balancing Protocol),* cada uno de ellos con características y modo de operar propio, pero compartiendo finalidad, consistente en evitar el aislamiento, en caso de fallo, de todos los hosts ubicados en una determinada subred. El concepto llevado a cabo resulta sencillo, basado en definir un grupo de routers físicos que crearán uno virtual, con una única IP que será la que actúe como puerta de enlace, de tal manera que, si

un dispositivo del grupo cae, otro asume automáticamente su rol, sin causar pérdida de conectividad y siendo un proceso totalmente transparente para los hosts.

Los protocolos encargados de ofrecer redundancia de puerta de enlace son denominados FHRP *(First-Hop Redundancy Protocols)* y pueden ser configurados tanto en routers como en switchs de capa 3. De todos ellos, HSRP y GLBP son propietarios de Cisco, por lo tanto, su utilización se limita a sus dispositivos, mientras, VRRP es un estándar abierto definido en el RFC3768, pudiendo ser aplicado sobre entornos multifabricante.

Un ejemplo de redundancia de *gateway*:

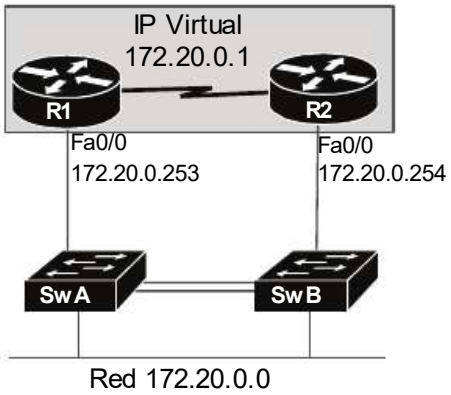


Fig. 8-3 Redundancia en puerta de enlace.

La red 172.20.0.0 dispone de dos routers físicos. La configuración de algún protocolo FHRP sobre los mismos supondría la creación entre ambos de otro virtual, el cual será utilizado por los hosts como puerta de enlace. En este caso, su IP es la 172.20.0.1.

De los tres protocolos mencionados, HSRP y GLBP forman parte del material de estudio de CCNA, siendo analizados a continuación.

PROTOCOLO HSRP: CARACTERÍSTICAS Y CONFIGURACIÓN

HSRP *(Hot Standby Router Protocol)* es un protocolo de capa 3 propietario de Cisco desarrollado con la finalidad de ofrecer redundancia de puerta de enlace a los dispositivos ubicados en una determinada subred. A lo largo de los siguientes

párrafos serán analizadas sus características, para acto seguido proceder a su configuración aplicándola sobre un supuesto práctico.

HSRP: Modo de operar

El objetivo del protocolo consiste en definir un router virtual, cuyo control podrá ser asumido por cualquier router físico perteneciente al mismo grupo HSRP, de tal manera que, si uno cae, sus funciones serán continuadas por otro, logrando así la redundancia. Para que ello sea posible cada dispositivo ejercerá un determinado rol, el cual definirá su función a desarrollar. Estos son:

- *Active:* Es el router que toma el control de la IP y MAC virtuales y por lo tanto el que ejecuta el enrutamiento. Tan solo un dispositivo del grupo asumirá esta tarea.

- *Standby:* Es aquel que monitoriza al activo con el fin de tomar su rol en caso de caída.

Dicha asignación se lleva a cabo en relación con la prioridad HSRP de cada dispositivo, donde el activo resultará aquel con el valor más alto. Además, es un parámetro configurable, por lo que se puede y debe influir sobre la elección de roles. Aun así, si no fuera definido manualmente, IOS aplica por defecto el valor 100, y en caso de coincidencia, el proceso se llevará a cabo conforme a la dirección IP configurada en sus interfaces físicas, donde el dispositivo con aquella más alta toma el rol de activo.

El modelo "*active/standby*" también es denominado como "*active/passive*".

Un ejemplo de HSRP:

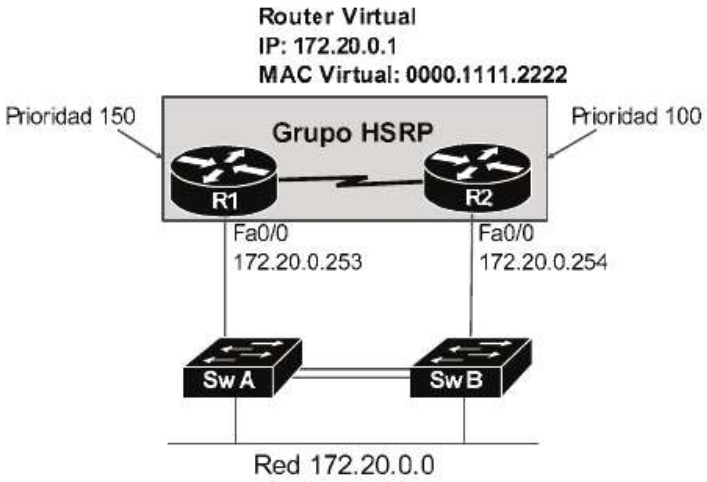


Fig. 8-4 Router virtual definido por HSRP.

Entre R1 y R2 se ha configurado un grupo HSRP con la finalidad de ofrecer redundancia de puerta de enlace a la red 172.20.0.0, creando un router virtual entre ambos, con IP 172.20.0.1 y MAC 000.1111.2222. El control de este será asumido por R1 *(active)*, ya que su prioridad es mayor, mientras que R2 tomará el rol *standby* y monitorizará al primero para ejercer sus funciones en caso de caída. Pero ¿cómo determinan los switchs a qué router físico reenviar las tramas cuyo destino sea la MAC virtual? Para ello, el seleccionado como activo ejecuta las siguientes acciones:

- R1 envía una trama a la red con MAC de origen 0000.1111.2222. Esta será recibida por SwA y SwB, los cuales la agregan a su tabla de MACs y la asocian a la interfaz por la cual fue recibida. A partir de ahora, las tramas con destino 0000.1111.2222 serán reenviadas a R1.

- Los hosts de la red harán uso de la IP del router virtual como *gateway*, en este caso la 172.20.0.1. Cuando los dispositivos envían paquetes a su puerta de enlace deben utilizar como destino en capa 2 la MAC de esta, la cual obtienen enviando una consulta ARP a la red. En un grupo HSRP, dichas consultas solo serán respondidas por el router activo.

¿Qué sucede si R1 cae?

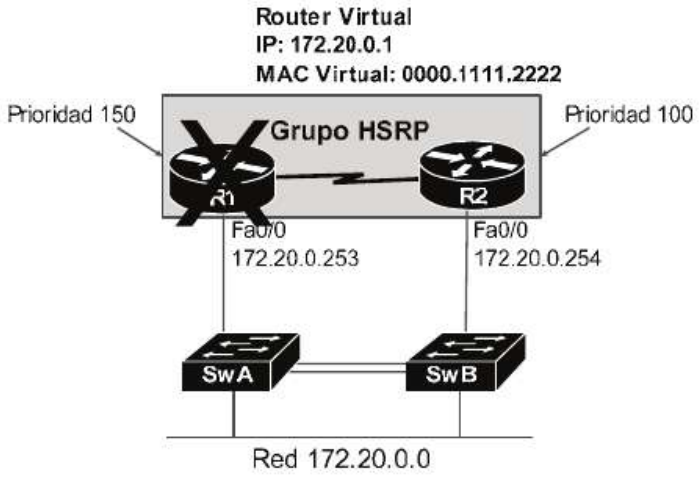


Fig. 8-5 Cambio de rol en HSRP.

Por algún motivo R1 no dispone de conexión a la red (link caído, fallo de configuración, router apagado, etc.). R2, tras monitorizarlo y detectar su pérdida de conectividad, toma el rol de activo y por lo tanto el control de la puerta de enlace. Sin embargo, los switchs desconocen dicho cambio y aún asocian la MAC virtual con la interfaz de R1, por lo tanto, la primera acción que ejecuta R2 consiste en enviar una

trama al medio con MAC de origen 0000.1111.2222. Los switchs la recibirán, eliminarán la entrada anterior y la asociarán a la nueva interfaz por la cual fue recibida, siendo en este caso aquella que conecta con R2.

Una de las mayores desventajas de HSRP consiste en que el enrutamiento solo es llevado a cabo por un router físico, es decir, no se ejecuta balanceo de carga. Si se deseara que ambos operen como activos de manera simultánea, resulta necesaria la configuración de dos grupos, cada uno de ellos con una IP virtual diferente, y gracias a la modificación de la prioridad, lograr que cada router actúe como activo en un grupo diferente. Como punto negativo a ello cabe destacar que habría que configurar diferentes puertas de enlace a los dispositivos de la red, por ejemplo, si está compuesta por 100 hosts, a 50 de ellos aplicar una IP virtual y a los otros 50 la otra.

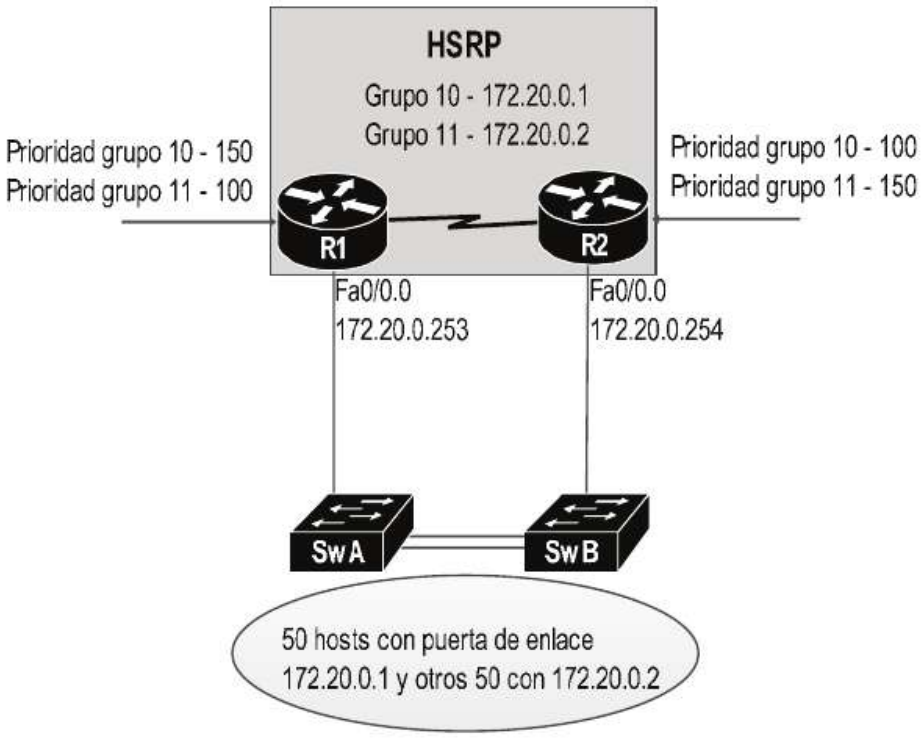


Fig. 8-6 Balanceo de carga en HSRP.

La topología dispone de dos routers que pueden actuar como puerta de enlace activa de manera simultánea, además, cada uno de ellos ejerce como *standby* del otro, por lo que, si alguno cae, el restante enrutaría el tráfico de las dos IPs virtuales.

Para lograrlo, el proceso de configuración debe contemplar los siguientes aspectos:

- Se deben definir dos grupos HSRP, cada uno de ellos con una IP virtual perteneciente al mismo rango de red.

- Modificar las prioridades manualmente, de tal manera que cada router resulte seleccionado como activo para cada una de las puertas de enlace. En el ejemplo, R1 será el activo para el grupo 10 y *standby* para el 11, mientras que R2 a la inversa, activo en el grupo 11 y *standby* en el 10. Gracias a ello, R1 enrutará el tráfico de los hosts con puerta de enlace 172.20.0.1 y actuará como respaldo para la 172.20.0.2. Por su contra, R2 enrutará el tráfico de los hosts con *gateway* 172.20.0.2 y ejercerá como *standby* para la 172.20.0.1.

Configuración y verificación de HSRP

Para habilitar HSRP se deben llevar a cabo las siguientes acciones, todas ellas ejecutadas desde el modo de configuración de la interfaz que conecta con la red a la cual se ofrecerá redundancia:

- *Paso 1:* Definir tanto el grupo al que pertenecerá el dispositivo como su IP virtual, con el comando **standby *[num grupo]* ip *[ip virtual]***.

- *Paso 2 (Opcional):* Modificar la prioridad del router para un determinado grupo, gracias a la sentencia **standby *[num grupo]* priority *[prioridad]***.

- *Paso 3 (Opcional):* Configurar la autenticación necesaria, ejecutando el comando **standby [*num grupo*] authentication *[password]***.

Ejemplo: Aplicar redundancia de puerta de enlace sobre la siguiente red, haciendo uso de HSRP y cumpliendo los siguientes requisitos:

- Grupo HSRP: 10.
- IP virtual: 192.168.10.254.
- Router activo: TFE.
- Standby: LPA.
- Contraseña de autenticación: “P4sS”.

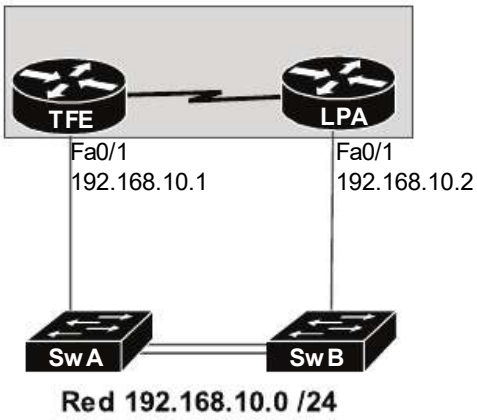


Fig. 8-7 Diseño de red para supuesto práctico de HSRP.

```
--- Configuración en TFE ---
TFE(config)#int fa0/1
TFE(config-if)#ip address 192.168.10.1 255.255.255.0
TFE(config-if)#no shutdown
TFE(config-if)#standby 10 ip 192.168.10.254
TFE(config-if)#standby 10 priority 150
TFE(config-if)#standby 10 authentication P4sS

--- Configuración en LPA ---
LPA(config)#int fa0/1
LPA(config-if)#ip address 192.168.10.2 255.255.255.0
LPA(config-if)#no shutdown
LPA(config-if)#standby 10 ip 192.168.10.254
LPA(config-if)#standby 10 authentication P4sS
```

Una vez aplicados los cambios, la primera acción que ejecutan ambos dispositivos consiste en iniciar una breve negociación con el fin de verificar que se cumplen las condiciones necesarias para establecer el grupo HSRP. Esta se basa en:

1.- Comprobar que ambos disponen de la misma IP virtual e idéntica autenticación. Si cualquiera de estos dos parámetros no coincide, la negociación se detiene y no se establece el grupo.

2.- Si concluye con éxito, continúan el proceso determinando la MAC virtual y versión del protocolo a aplicar (por defecto la v2).

3.- Por último, y en relación con las prioridades, se define el rol adoptado por cada dispositivo.

En el ejemplo recién analizado, TFE y LPA coinciden en configuración, perteneciendo al grupo 10 de HSRP con IP virtual 192.168.10.254 y autenticación "P4sS", por lo tanto, la negociación concluye con éxito. La única diferencia radica en que TFE dispone de una prioridad de 150, siendo seleccionado como activo ya que LPA mantiene el valor por defecto, 100.

Con el fin de verificar que los cambios han sido aplicados según lo previsto, IOS dispone de los siguientes comandos de verificación:

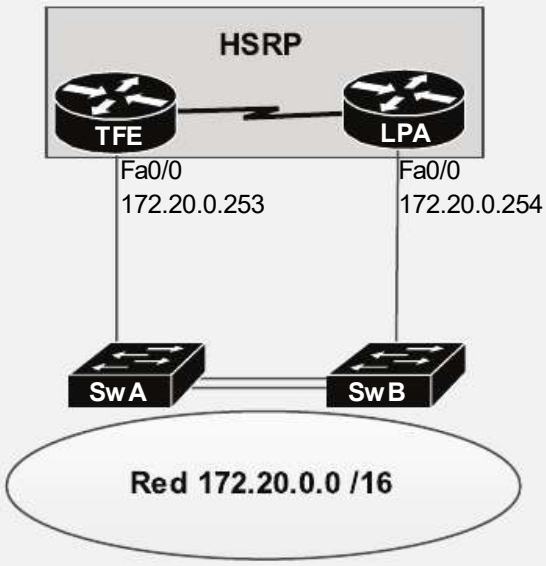
- *show standby:* Muestra información relevante sobre cada una de las interfaces donde HSRP ha sido configurado, como el grupo al cual pertenece, IP y MAC virtual, rol, prioridad, etc.

- *show standby brief:* Muestra un listado en pantalla incluyendo aquellas interfaces configuradas en HSRP, facilitando información como el grupo al cual pertenecen, IP local y virtual, rol, etc.

```
--- Verificación en TFE ---

TFE#show standby
FastEthernet0/1 - Group 10 (version 2)
  State is Active
    6 state changes, last state change 00:25:47
  Virtual IP address is 192.168.10.254
  Active virtual MAC address is 0000.0C9F.0000
    Local virtual MAC address is 0000.0C9F.F00A (v2 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.386 secs
  Preemption disabled
  Active router is local
  Standby router is 192.168.10.2
  Priority 150 (configured 150)
  Group name is hsrp-Fa0/1-10 (default)

TFE#show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State    Active          Standby         Virtual IP
Fa0/1       10   150   Active   local           192.168.10.2    192.168.10.254




--- Verificación en LPA ---

LPA#show standby
FastEthernet0/1 - Group 10 (version 2)
  State is Standby
    4 state changes, last state change 00:28:52
```

```
  Virtual IP address is 192.168.10.254
  Active virtual MAC address is unknown
    Local virtual MAC address is 0000.0C9F.F00A (v2 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.476 secs
  Preemption disabled
  Active router is 192.168.10.1
  Standby router is local
  Priority 100 (default 100)
  Group name is hsrp-Fa0/1-10 (default)
LPA#show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State    Active          Standby         Virtual IP
Fa0/1       10   100   Standby  192.168.10.1    local           192.168.10.254
```

Reto 8.1 – Configurar HSRP en los routers TFE y LPA de tal manera que se ejecute balanceo de carga entre ambos. Para ello, se deben cumplir los siguientes requisitos:
- Definir dos grupos, el 20 con IP virtual 172.20.0.1, y el 30 con IP virtual 172.20.0.2.
- TFE debe tomar el rol activo para el grupo 20 y standby para el 30.
- LPA debe tomar el rol activo para el grupo 30 y standby para el 20.
- La autenticación para el grupo 20 debe ser “P4ss20”, mientras que para el grupo 30 “W0rd30”.



Solución al final del capítulo.

PROTOCOLO GLBP: CARACTERÍSTICAS Y CONFIGURACIÓN

Otro de los protocolos disponibles en cuanto a redundancia de puerta de enlace se refiere es GLBP *(Gateway Load Balancing Protocol).* Este comparte finalidad con HSRP y VRRP, sin embargo, sus características lo convierten en la opción más avanzada para este propósito. A lo largo de la presente sección serán analizados tanto su modo de operar como su configuración, abordando esta última mediante un supuesto práctico.

GLBP: Modo de operar

GLBP, con el fin de lograr el objetivo de ofrecer redundancia en puerta de enlace a los hosts de una determinada subred, define un grupo compuesto por varios routers físicos que hacen uso de una misma IP virtual, de tal manera que, si uno de ellos cae, otro toma su rol sin interrumpir la conectividad de los dispositivos finales. Entonces, ¿en qué se diferencia respecto a HSRP? Principalmente en un factor, y es que GLBP aplica balanceo de carga por defecto, sin necesidad de crear más de un grupo.

Para ello genera diferentes MACs virtuales, tantas como routers físicos disponga, de tal manera que los hosts, al realizar una consulta ARP sobre la IP virtual, recibirán una u otra MAC. Tras ello, la comunicación en capa 2 de cada dispositivo con la puerta de enlace queda vinculada a un determinado router, logrando de esta manera el balanceo de carga. El proceso llevado a cabo consta de:

- *Paso 1:* Para cada router del grupo se genera una MAC virtual diferente.

- *Paso 2:* Cada consulta ARP a la puerta de enlace será respondida de manera sucesiva con una MAC virtual. Es decir, si por ejemplo existen dos routers en una red con 4 hosts, la primera consulta ARP será respondida con la MAC virtual del primer router, la segunda consulta con la MAC virtual del segundo router, la tercera con la MAC del primero, y así sucesivamente a medida que se reciban peticiones ARP.

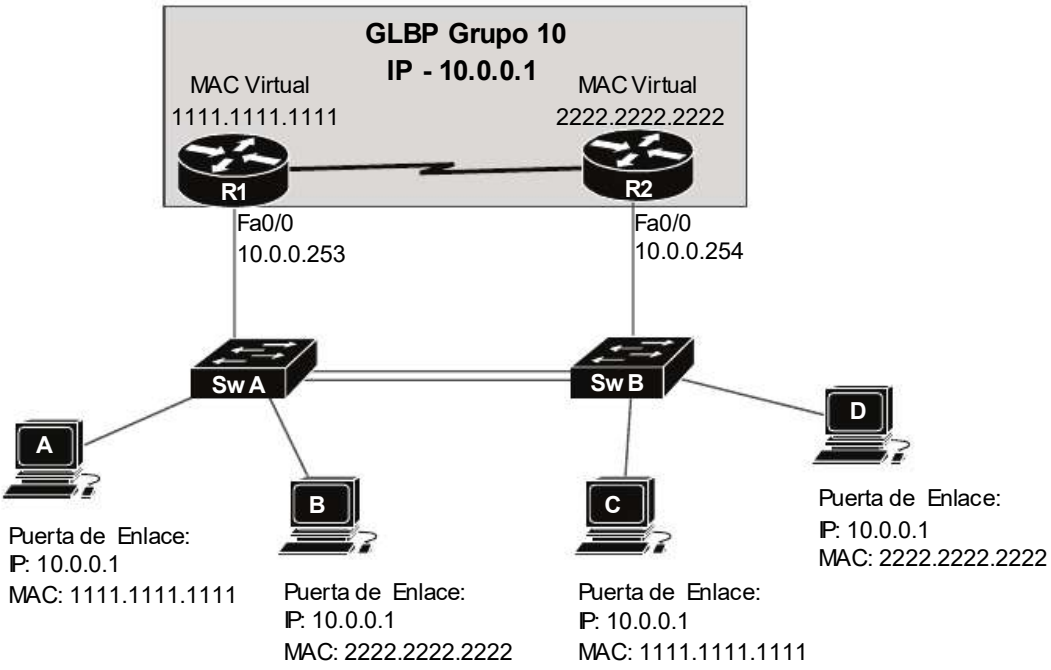


Fig. 8-8 Router virtual definido por GLBP.

En la topología mostrada, los hosts han sido configurados con la IP 10.0.0.1 como puerta de enlace, por lo tanto, envían una petición ARP a la misma con el fin de obtener su MAC. A y C han recibido la 1111.1111.1111, la cual es utilizada por R1, mientras que B y D han recibido la 2222.2222.2222, lo que significa que su comunicación será a través de R2. Gracias a ello, ambos dispositivos ejecutan el enrutamiento de manera simultánea.

¿Qué sucede si algún router cae? En este caso, si alguno perdiera conectividad, el restante lo detecta y automáticamente asume su control, logrando que la comunicación de los hosts no se interrumpa. Por ejemplo, si R1 cae, R2 tomaría posesión de dos MACs virtuales, la suya propia y la de R1.

Se ha mencionado que las consultas ARP a la puerta de enlace son respondidas de manera sucesiva, respetando un orden establecido. Para que ello sea posible resulta necesario que un solo router sea el encargado de responder dichas peticiones. Por lo tanto, este debe conocer todas las MACs virtuales generadas. Esta función será determinada con relación al rol de cada uno de ellos, que en GLBP son:

- *Router AVG:* El AVG *(Active Virtual Gateway)* es el encargado de generar y asignar las MACs virtuales a los demás routers del grupo, así como de responder a las consultas ARP. Solo un dispositivo puede tomar este rol, siendo aquel cuya prioridad resulte la más alta.

- *Router AVF:* Los AVF *(Active Virtual Forwarder)* son todos los routers que componen un grupo GLBP. Su función se basa en asumir el control de una MAC virtual y ejercer como puerta de enlace para los hosts que hagan uso de la misma. El seleccionado como AVG también actúa como AVF.

Otra de las grandes diferencias entre HSRP y GLBP es que en el primero solo dos routers asumen un rol (*active* o *standby*), y de estos, tan solo uno ejerce como puerta de enlace, de ahí que el modelo implementado en este caso sea denominado como activo/pasivo. Sin embargo, en GLBP todos los dispositivos que componen el grupo toman el rol AVF, participando en el enrutamiento, lo cual corresponde a un modelo activo/activo.

Por último, y al igual que HSRP, GLBP es un protocolo propietario de Cisco, por lo que su aplicación se limita a dispositivos de dicho fabricante.

Configuración y verificación de GLBP

El proceso de configuración de GLBP resulta muy similar al llevado a cabo en HSRP. Como requisito obligatorio se deberá especificar el grupo e IP virtual a utilizar, mientras que como parámetros opcionales se podrá establecer su prioridad y autenticación. Todo ello desde el modo de configuración de la interfaz que conecta con la red a la cual se ofrecerá redundancia:

- *Paso 1:* Definir tanto el grupo al que pertenecerá el dispositivo como su IP virtual, haciendo uso del comando **glbp** ***[num grupo]*** **ip** ***[ip virtual]***.

- *Paso 2 (Opcional):* Modificar la prioridad para un determinado grupo, a través de la sentencia **glbp** ***[num grupo]*** **priority** ***[prioridad]***. Si no fuera definida se aplicará el valor por defecto, 100, y en caso de coincidencia la elección se llevará a cabo conforme a la IP configurada en las interfaces físicas, siendo el router con el valor más alto aquel seleccionado como AVG.

- *Paso 3 (Opcional):* Establecer la autenticación necesaria con el comando **glbp [*num grupo]* authentication text *[contraseña]***.

Ejemplo: Aplicar redundancia de puerta de enlace en la siguiente topología, cumpliendo los siguientes requisitos:

- Grupo GLBP: 5.
- IP Virtual: 192.168.1.1.
- Router AVG: TFE.
- Autenticación: “P4ss”.

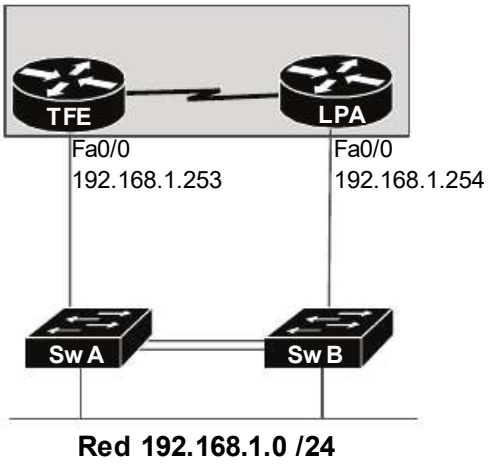


Fig. 8-9 Diseño de red para supuesto práctico de GLBP.

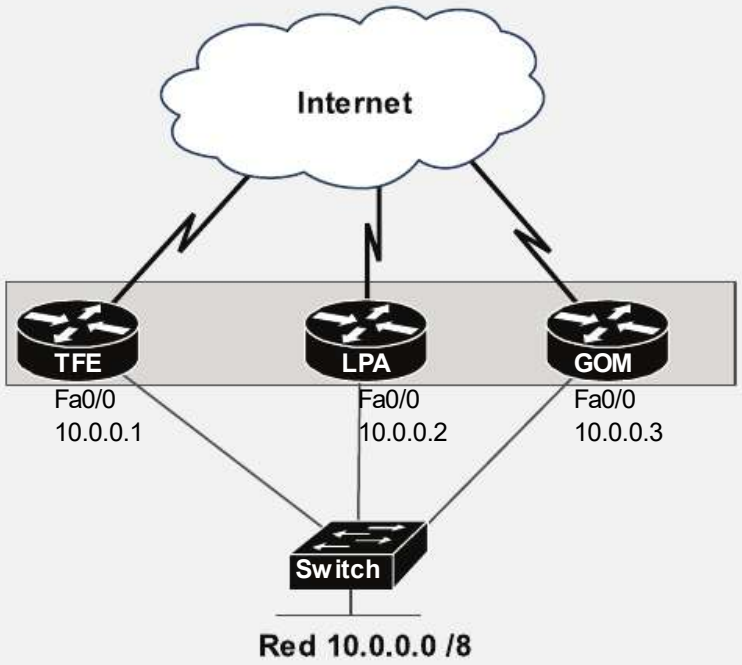
```
--- Configuración en TFE ---

TFE(config)#int fa0/0
TFE(config-if)#ip address 192.168.1.253 255.255.255.0
TFE(config-if)#no shutdown
TFE(config-if)#glbp 5 ip 192.168.1.1
TFE(config-if)#glbp 5 priority 150
TFE(config-if)#glbp 5 authentication text P4ss

--- Configuración en LPA ---

LPA(config)#int fa0/0
LPA(config-if)#ip address 192.168.1.254 255.255.255.0
LPA(config-if)#no shutdown
LPA(config-if)#glbp 5 ip 192.168.1.1
LPA(config-if)#glbp 5 authentication text P4ss
```

Para verificar que la configuración ha sido aplicada según lo previsto se podrá hacer uso de los siguientes comandos:

- *show glbp:* Muestra información del protocolo sobre cada una de las interfaces donde ha sido configurado, facilitando datos como la IP y MACs virtuales, grupo, prioridad, rol, etc.

- *show glbp brief:* Genera un listado con información relevante sobre los routers pertenecientes al mismo grupo GLBP, incluyendo datos como su estado, grupo, prioridad, IP, etc.

Reto 8.2 – Configurar GLBP en la siguiente topología, teniendo en cuenta los siguientes requisitos:
- Todos los routers deben pertenecer al grupo 10.
- TFE debe responder cualquier consulta ARP hacia la IP virtual.
- Los hosts harán uso de la IP 10.0.0.254 como puerta de enlace.
- La autenticación necesaria para formar parte del grupo debe ser la cadena "Acc3$$".

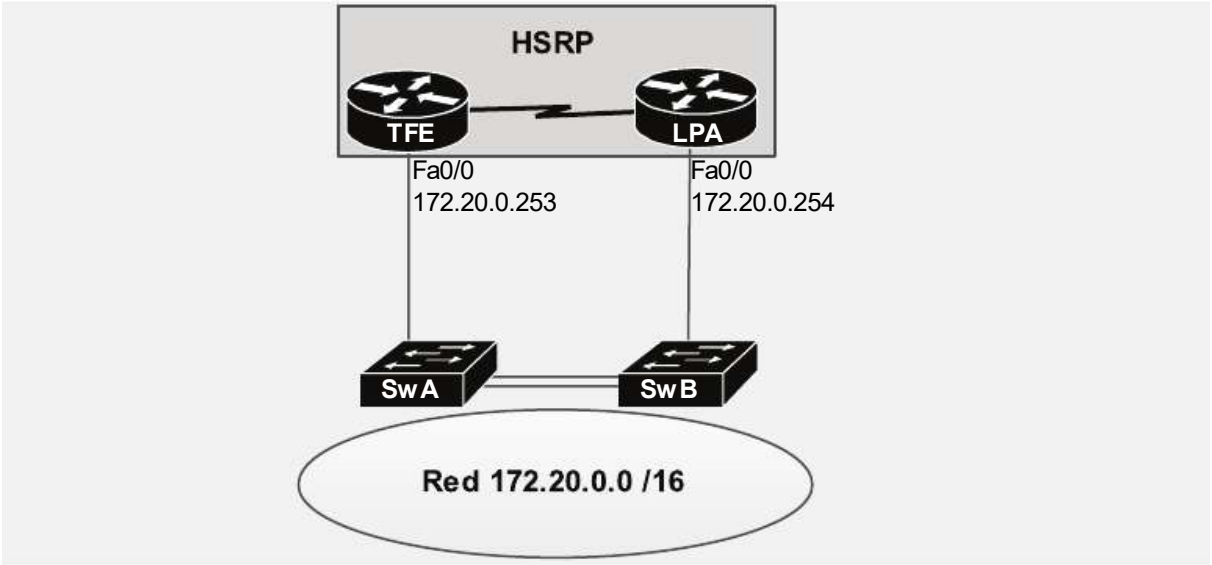


Solución al final del capítulo.

SOLUCIÓN DE RETOS: HSRP Y GLBP

Reto 8.1 – Configurar HSRP en los routers TFE y LPA de tal manera que se ejecute balanceo de carga entre ambos. Para ello, se deben cumplir los siguientes requisitos:

- Definir dos grupos, el 20 con IP virtual 172.20.0.1, y el 30 con IP virtual 172.20.0.2.
- TFE debe tomar el rol activo para el grupo 20 y standby para el 30.
- LPA debe tomar el rol activo para el grupo 30 y standby para el 20.
- La autenticación para el grupo 20 debe ser "P4ss20", mientras que para el grupo 30 "W0rd30".

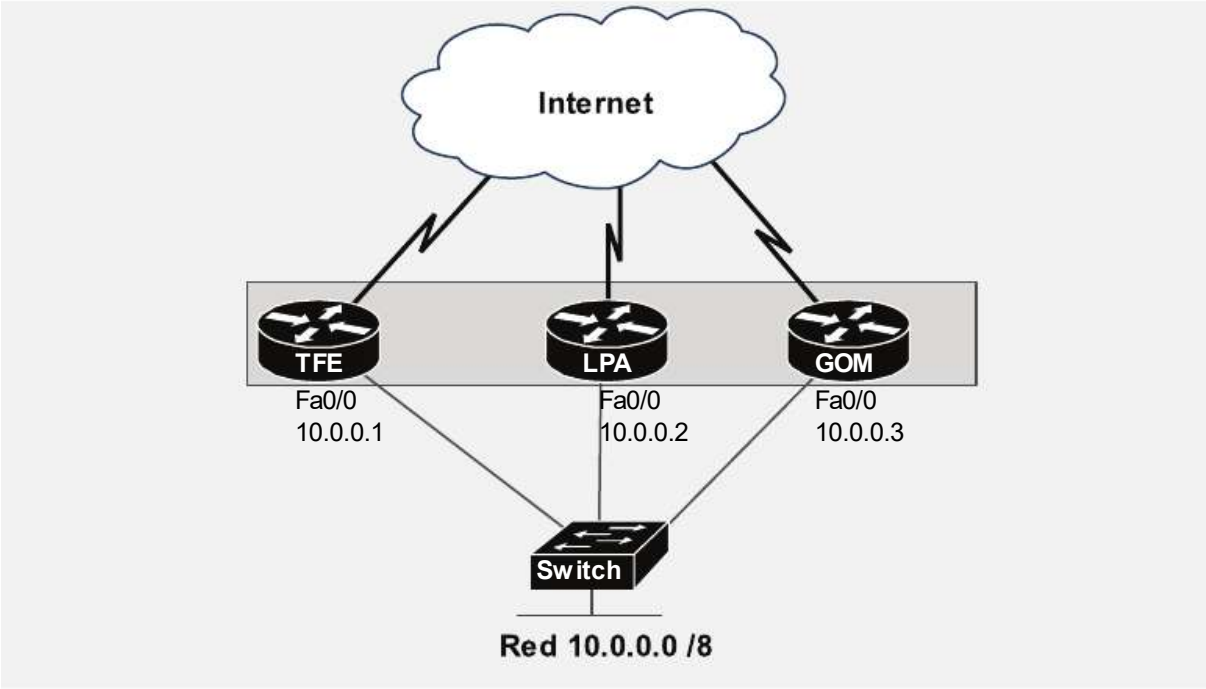



```
--- Configuración en TFE ---

TFE(config)#int fa0/0
TFE(config-if)#ip address 172.20.0.253 255.255.0.0
TFE(config-if)#no shutdown
TFE(config-if)#standby 20 ip 172.20.0.1
TFE(config-if)#standby 20 priority 150
TFE(config-if)#standby 20 authentication P4ss20
TFE(config-if)#standby 30 ip 172.20.0.2
TFE(config-if)#standby 30 authentication W0rd30

--- Configuración en LPA ---

LPA(config)#int fa0/0
LPA(config-if)#ip address 172.20.0.254 255.255.0.0
LPA(config-if)#no shutdown
LPA(config-if)#standby 20 ip 172.20.0.1
LPA(config-if)#standby 20 authentication P4ss20
LPA(config-if)#standby 30 ip 172.20.0.2
LPA(config-if)#standby 30 priority 150
LPA(config-if)#standby 30 authentication W0rd30
```

Las prioridades pueden variar, siempre y cuando TFE obtenga un valor mayor de 100 para el grupo 20 y LPA para el grupo 30.

Reto 8.2 – Configurar GLBP en la siguiente topología, teniendo en cuenta los siguientes requisitos:

- Todos los routers deben pertenecer al grupo 10.
- TFE debe responder cualquier consulta ARP hacia la IP virtual.
- Los hosts harán uso de la IP 10.0.0.254 como puerta de enlace.
- La autenticación necesaria para formar parte del grupo debe ser la cadena "Acc3$$".




```
--- Configuración en TFE ---

TFE(config)#int fa0/0
TFE(config-if)#ip address 10.0.0.1 255.0.0.0
TFE(config-if)#no shutdown
TFE(config-if)#glbp 10 ip 10.0.0.254
TFE(config-if)#glbp 10 priority 150
TFE(config-if)#glbp 10 authentication text Acc3$$

--- Configuración en LPA ---

LPA(config)#int fa0/0
LPA(config-if)#ip address 10.0.0.2 255.0.0.0
LPA(config-if)#no shutdown
LPA(config-if)#glbp 10 ip 10.0.0.254
LPA(config-if)#glbp 10 authentication text Acc3$$

--- Configuración en GOM ---

GOM(config)#int fa0/0
GOM(config-if)#ip address 10.0.0.3 255.0.0.0
GOM(config-if)#no shutdown
GOM(config-if)#glbp 10 ip 10.0.0.254
```

La prioridad en TFE puede variar, siempre y cuando su valor sea mayor de 100.

TEST CAPÍTULO 8: HSRP Y GLBP

1.- ¿Cuál de los siguientes objetivos resulta posible lograr mediante la aplicación de redundancia?

A. Aumentar la seguridad en la red.
B. Evitar que elementos críticos dispongan de un único punto de fallo.
C. Ampliar el ancho de banda.
D. Mayor velocidad de enrutamiento.

2.- De las siguientes opciones, ¿cuál identifica un protocolo FHRP?

A. HSRP.
B. SFTP.
C. HTTP.
D. STP.

3.- Se desea implementar FHRP sobre una topología compuesta por 3 routers de diferentes fabricantes. ¿Cuál de los siguientes protocolos resulta la opción idónea?

A. HSRP.
B. VRRP.
C. GLBP.
D. Cualquiera de los anteriores.

4.- Un grupo HSRP ha sido configurado con la IP 10.1.1.1. ¿Qué MAC virtual utilizará?

A. La MAC de la interfaz del router que obtenga el rol de activo.
B. La MAC más alta de los routers que formen parte del mismo grupo.
C. La MAC 0010:0001:0011.
D. Ninguna de las anteriores.

5.- Un grupo HSRP está compuesto por los routers R1 (activo), R2 (standby) y R3. Pasado un tiempo, R1 sufre un fallo de hardware y debe ser retirado. ¿Cuál de las siguientes afirmaciones describe con mayor exactitud el proceso llevado a cabo por HSRP tras la caída de su router activo?

A. R2 toma el control de la IP y MAC virtual.
B. R2 toma el control de la IP y MAC virtual y cambia su rol a activo.
C. R2 toma el control de la IP y MAC virtual y cambia su rol a activo. R3 actúa como *standby*.
D. Se ejecuta una nueva negociación entre R2 y R3 para concluir cuál de ellos tomará el rol de activo.

6.- En la red 172.20.0.0/16 se ha configurado redundancia de puerta de enlace sobre un grupo de 3 routers físicos, participando todos ellos de manera activa y simultánea en el enrutamiento. ¿Qué protocolo ha sido aplicado?

A. VRRP.
B. GLBP.
C. HSRP.
D. SFTP.

7.- ¿Qué router es el encargado de responder a las peticiones ARP dirigidas hacia la IP virtual en un grupo HSRP?

A. Router activo.
B. Router pasivo.
C. Ambos.
D. Ninguno.

8.- En un grupo HSRP compuesto por dos routers no ha sido configurada la prioridad en ninguno de ellos. R1 hace uso de la IP virtual 172.10.0.1/16, mientras que R2 de la 172.20.0.1/16. Tras la negociación llevada a cabo por el protocolo ¿Cuál de ellos será seleccionado como router activo?

A. R1.
B. R2.
C. Ambos, se ejecutará balanceo de carga.
D. Ninguno, no se establecerá el grupo HSRP.

9.- ¿Cuál es la función del router seleccionado como *standby* en HSRP?

A. No ejerce ninguna función.
B. Responder consultas ARP.
C. Monitorizar al router activo.
D. Generar la MAC virtual.

10.- ¿Qué método se deberá llevar a cabo en HSRP para lograr balanceo de carga?

A. Configurar un grupo, con tantas IPs virtuales como routers físicos disponga, de tal manera que se generará una MAC virtual para cada uno de ellos y todos podrán enrutar tráfico.
B. En HSRP el balanceo de carga se ejecuta por defecto entre todos los routers pertenecientes al mismo grupo.
C. Definir un grupo HSRP con una IP virtual y conectar todos los routers físicos al mismo segmento de red.
D. Ninguna de las anteriores.

11.- El grupo 10 de HSRP requiere como autenticación la cadena "T3sT". ¿Qué comando se deberá aplicar sobre sus routers para definirla?

A. Router(config)# standby 10 authentication T3sT
B. Router(config-rtr)# standby 10 authentication T3sT
C. Router(config-line)# standby 10 authentication T3sT
D. Router(config-if)# standby 10 authentication T3sT

12.- De las siguientes opciones, ¿cuál identifica una diferencia entre HSRP y GLBP?

A. HSRP es un protocolo propietario de Cisco, GLBP no.
B. HSRP aplica balanceo de carga por defecto, GLBP no.
C. HSRP hace uso de un modelo activo/pasivo, GLBP no.
D. HSRP permite autenticación. GLBP no.

13.- R1 ha sido seleccionado como router activo para el grupo 10 de HSRP, formando parte del mismo a través de su interfaz Fa0/1. El administrador de red desea conocer qué MAC virtual ha generado. ¿Cuál de los siguientes comandos de verificación facilitará dicha información?

A. R1#show interface Fa0/1
B. R1#show interface Fa0/1 detail
C. R1#show ip interface brief
D. R1# show hsrp virtual mac
E. Ninguno de los anteriores.

14.- ¿Qué diferencia a un router AVG de otro AVF en GLBP? (Seleccionar dos respuestas)

A. El AVG actúa como puerta de enlace activa, el AVF solo como respaldo.
B. El AVG tiene una prioridad mayor que el AVF.
C. El AVG responde a las consultas ARP dirigidas hacia la IP virtual.
D. El AVG monitoriza a los routers AVF, y si alguno cae, toma el control de su MAC virtual.

15.- ¿Cómo logra el protocolo GLBP que varios routers físicos actúen como puerta de enlace activa de manera simultánea?

A. Mediante la aplicación de múltiples IP virtuales.
B. Mediante la modificación de la prioridad en cada router.
C. Mediante la asignación de diferentes MACs virtuales.
D. Mediante la implementación de enlaces redundantes hacia los switchs.

16.- Se ha configurado como IP virtual la dirección 192.168.10.1 para ofrecer redundancia de puerta de enlace a los hosts pertenecientes a la red 192.168.11.0/24. Sin embargo, estos no obtienen conectividad con ninguna otra red. ¿A qué puede ser debido?

A. Los routers son de diferentes fabricantes y por lo tanto incompatibles.
B. La prioridad coincide en todos los dispositivos, generando conflicto a la hora de seleccionar al activo.
C. La IP virtual no comparte el mismo rango de red que los hosts.
D. Se ha configurado algún protocolo FHRP incompatible con los routers de la topología.

17.- GLBP hace uso de un modelo…

A. Activo/activo.
B. Pasivo/activo.
C. Activo/pasivo.
D. AVF/pasivo.
E. AVG/pasivo.

18.- Un conjunto de tres routers ha sido configurado con HSRP y GLBP para ofrecer redundancia de puerta de enlace a los hosts de una misma subred. HSRP utiliza como IP virtual la 10.0.0.1 mientras que GLBP la 10.0.0.2. ¿Qué dirección IP aplicarán los hosts como gateway?

A. 10.0.0.1
B. 10.0.0.2
C. HSRP y GLBP no pueden ser configurados en los mismos routers para ofrecer redundancia a una misma subred.
D. La dirección IP definida en los hosts, ya sea manualmente o de manera automática mediante DHCP.

19.- Un conjunto de tres routers ha sido configurado en el grupo 1 de GLBP, de los cuales se desea que R1 resulte seleccionado como AVG. R2 y R3 mantienen la prioridad por defecto. ¿Cuál de los siguientes comandos se deberá aplicar para lograr dicho propósito?

A. R1(config-if)# glbp 1 priority 100
B. R1(config-if)# glbp 1 priority 90
C. R1(config)# glbp 1 priority 120
D. R1(config)# glbp 1 priority 50
E. Ninguno de los anteriores.

20.- ¿Qué protocolo FHRP dispone de mejores características y por lo tanto resulta recomendable a implementar en una topología compuesta únicamente por routers Cisco?

A. VRRP.
B. HSRP.
C. GLBP.

REDES PRIVADAS VIRTUALES 9

VPN: CONCEPTOS BÁSICOS

Actualmente, la distribución física llevada a cabo por gran parte de las compañías consta de una sede central y el resto remotas, todas ellas distanciadas geográficamente y resultando imprescindible su interconexión con el fin de formar parte de la misma red privada y con ello poder compartir datos, aplicaciones o políticas de seguridad, entre otras. Para lograr dicho propósito existen diferentes opciones, como los circuitos dedicados, la aplicación de MPLS *(Multiprotocol Label Switching)* o la implementación de VPNs. De los 3 métodos, los dos primeros pueden suponer un coste elevado, por lo que en muchas ocasiones se suele optar por las redes privadas virtuales, las cuales serán analizadas a lo largo del presente capítulo.

Una VPN *(Virtual Private Network)* puede ser definida como la tecnología que habilita el transporte de datos de manera segura entre dispositivos de una red privada a través de un medio público como Internet. Resulta la solución de interconexión remota económicamente menos costosa y por ello se ha convertido en la opción idónea para pequeñas y medianas empresas, debido en gran parte al nivel de seguridad y fiabilidad logrado gracias a los protocolos aplicados sobre las mismas, tanto para el cifrado como para el transporte de datos. Bien es cierto que estos, al ser enviados a través de un medio público son susceptibles a ataques, pero difícilmente legibles y manipulables.

Las VPN aportan beneficios como:

- *Ahorro de costes:* Su puesta en marcha tan solo requiere conexión a Internet en ambos extremos y dispositivos que soporten su implementación.

- *Escalabilidad:* Su configuración, además de resultar un proceso bastante sencillo, depende por completo del administrador de la red, no de terceros como puede ser el caso de los circuitos dedicados. Este hecho, unido a la baja inversión económica que requieren, las convierten en una solución altamente escalable.

- *Compatibilidad:* Al tratarse de una tecnología que hace uso de Internet como medio de transporte, es compatible con todos los dispositivos que tengan a acceso a dicho medio, como portátiles, teléfonos móviles, routers, etc.

- *Seguridad:* Las VPNs se apoyan en otros protocolos para el cifrado de datos y autenticación, previniendo el robo de estos y el acceso no autorizado a la red.

Precisamente la seguridad representa uno de los elementos más importantes sobre cualquier VPN, la cual podrá ser implementada gracias a la combinación de diferentes protocolos, disponiendo, a su vez, de varios métodos para ello. Sea cual sea la opción elegida, la comunicación entre ambos extremos debe garantizar los siguientes principios de seguridad:

- *Privacidad (Confidencialidad):* Cifrar los datos con el objetivo de que solo sean legibles por el destinatario de los mismos.

- *Integridad:* Asegurar que la comunicación no sufre ninguna modificación desde el origen hasta el destino, ya sea intencionadamente o debido a problemas de transmisión (como interferencias). La integridad se logra gracias al campo "*CRC Checksum*", donde el origen aplica un valor generado conforme a un algoritmo matemático ejecutado sobre los bits que integran el paquete. El destino ejecuta el mismo proceso sobre los datos recibidos y si el resultado no coincide significa que se ha producido algún error durante la transmisión.

- *Autenticación:* Como su nombre indica, autenticar ambos extremos de la conexión con el fin de verificar que los datos que atraviesan el enlace provienen de dispositivos autorizados.

- *Anti-Replay:* Prevenir que los datos sean capturados por un tercero y reenviados falsificando la identidad del origen o modificando su contenido, es decir, evitar ataques *man-in-the-middle.*

En resumen, el objetivo principal de las VPNs consiste en establecer la conexión entre dos ubicaciones remotas y permitir la comunicación de sus redes privadas a

través de un medio público como Internet, asegurando los datos para que no sean ni legibles ni falsificados por terceros. Pero, si una IP privada no es transportable a través de una red pública, ¿cómo es posible lograr dicha comunicación? ¿Qué direcciones se utilizan durante el enrutamiento? Para ello, el software o hardware VPN debe encapsular el paquete IP de la red privada en otro cuyas direcciones sean públicas, con el fin de poder enviarlo a través de Internet. Además, el contenido del paquete original es cifrado. En el destino, el software o hardware VPN desencapsulará el paquete y descifrará los datos originales, reenviándolos al destino correcto.

Un ejemplo del proceso llevado a cabo por ambos extremos de la comunicación podría ser el siguiente, donde el Host A desea comunicarse con el servidor SMTP (172.10.0.1), ubicado en una sede remota conectada a través de una VPN entre R1 y R2.

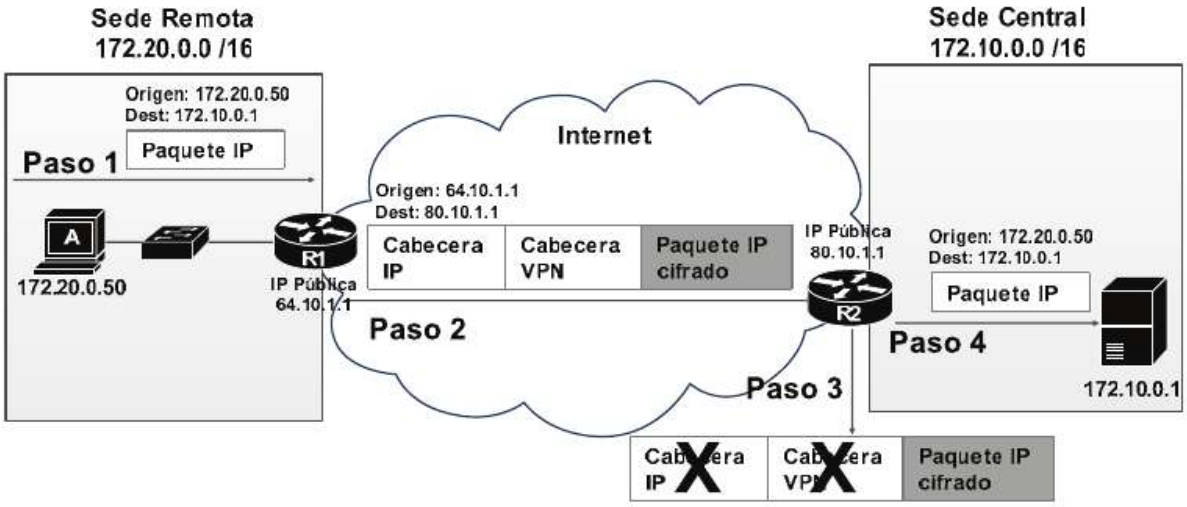


Fig. 9-1 Comunicación entre los dos extremos de la VPN.

- *Paso 1:* El Host A desea comunicarse con el servidor SMTP. Para ello, genera un paquete IP con origen 172.20.0.50 y destino 172.10.0.1 (direcciones privadas) y lo envía a R1.

- *Paso 2:* R1 lo examina y comprueba que la comunicación debe atravesar la VPN, por lo tanto, cifra el paquete original y lo encapsula en otro nuevo, el cual estará compuesto por dos cabeceras, una VPN y otra IP. Esta última incluye las direcciones públicas de origen (64.10.1.1) y destino (80.10.1.1). Acto seguido lo envía a través de Internet hacia R2.

- *Paso 3:* R2 recibe el paquete y ejecuta varias acciones, primero, verifica la autenticación del origen (R1), y segundo, comprueba que los datos no han sufrido ninguna modificación o error durante la transmisión. Si ambas concluyen con éxito, desencapsula y descifra el paquete IP original creado por el Host A.

- *Paso 4:* Por último, R2 reenvía el paquete original hacia la red privada, siendo recibido por el servidor 172.10.0.1.

Dependiendo de su propósito, las VPNs pueden ser clasificadas en dos tipos:

- *site-to-site:* Son aquellas que establecen la comunicación entre dos redes privadas. En este tipo de VPNs la conexión suele ser permanente y para su configuración se hace uso de dispositivos dedicados y especializados para ello, como routers. A su vez, pueden identificarse dos subtipos, la *Intranet site-to-site*, que conecta dos redes privadas administradas por la misma compañía y normalmente disponen de acceso total entre ellas (al igual que si estuvieran físicamente unidas) y la *Extranet site-to-site*, que define conexiones entre redes privadas de diferentes compañías, disponiendo normalmente de acceso limitado (determinados recursos o servidores) con el fin de compartir cierta información o proyectos.

- *acceso remoto:* En este caso, el propósito consiste en permitir la conexión y comunicación de usuarios individuales con la red privada de la compañía, de tal manera que puedan acceder a los recursos que esta ofrece sin necesidad de estar físicamente en ella. Es decir, habilitar la movilidad de los trabajadores, pudiendo ejercer sus funciones sin importar su ubicación. Este tipo de VPN suele ser configurada mediante software (cliente VPN) instalado en los dispositivos de los usuarios, mientras que en el otro extremo el acceso es gestionado por algún dispositivo especializado para ello, como un router o firewall.

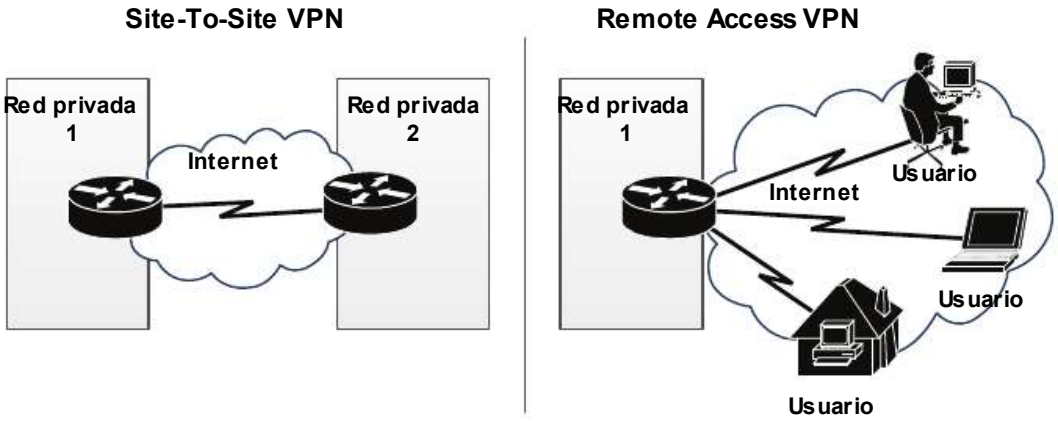


Fig. 9-2 Diferencia entre VPN site-to-site y de acceso remoto.

Como se ha mencionado en párrafos anteriores, el medio utilizado por las VPNs para llevar a cabo sus operaciones es Internet, por lo que resulta imprescindible contar con mecanismos de seguridad eficientes para que el transporte de datos se realice de manera fiable. Dos métodos para lograrlo son IPSec y SSL, protocolos que serán analizados a continuación.

Protocolos de seguridad: IPSec y SSL

Comprendidos el concepto, las funciones y los tipos de VPN, resulta necesario profundizar en una de sus características principales, la seguridad, la cual puede ser implementada en relación con diferentes protocolos. Dos de ellos, IPSec y SSL son contenido de CCNA y por lo tanto analizados a continuación.

IPSec

IPSec *(IP Security)* no identifica un protocolo en sí, sino un conjunto de ellos (suite o pila) que operan en capa 3 y que implementados de manera conjunta garantizan los principios de privacidad, integridad, autenticación y anti-replay a los datos sobre los cuales se aplica, es decir, aseguran la comunicación entre ambos extremos de la VPN.

Una de las mayores ventajas es que se trata de una suite abierta, por lo tanto, permite la inclusión y actualización de los protocolos ya existentes, traduciéndose en mejoras de seguridad. Por ejemplo, para el cifrado de datos IPSec comenzó con DES como única opción, el cual aplica una clave de 56 bits, sin embargo, con el paso del tiempo se desarrollaron métodos más avanzados, como 3DES y AES, los cuales fueron agregados a la suite, de tal manera que actualmente se puede optar por cualquiera de estos 3 protocolos y posiblemente en el futuro se incluirá alguno más. 3DES hace uso una clave de 56 bits ejecutada 3 veces sobre los mismos datos, mientras que en AES lo son de 128 o 256 bits.

Su modo de operación consiste en definir una clave de cifrado *(encryption key, session key o shared key),* que debe coincidir en ambos extremos de la VPN, la cual será utilizada junto a un algoritmo matemático para cifrar los paquetes IP originados en la red privada. Acto seguido son encapsulados en un nuevo paquete que incluye las cabeceras VPN e IP. En el destino, se aplica el mismo algoritmo en conjunto con la clave de cifrado para obtener los datos originales.

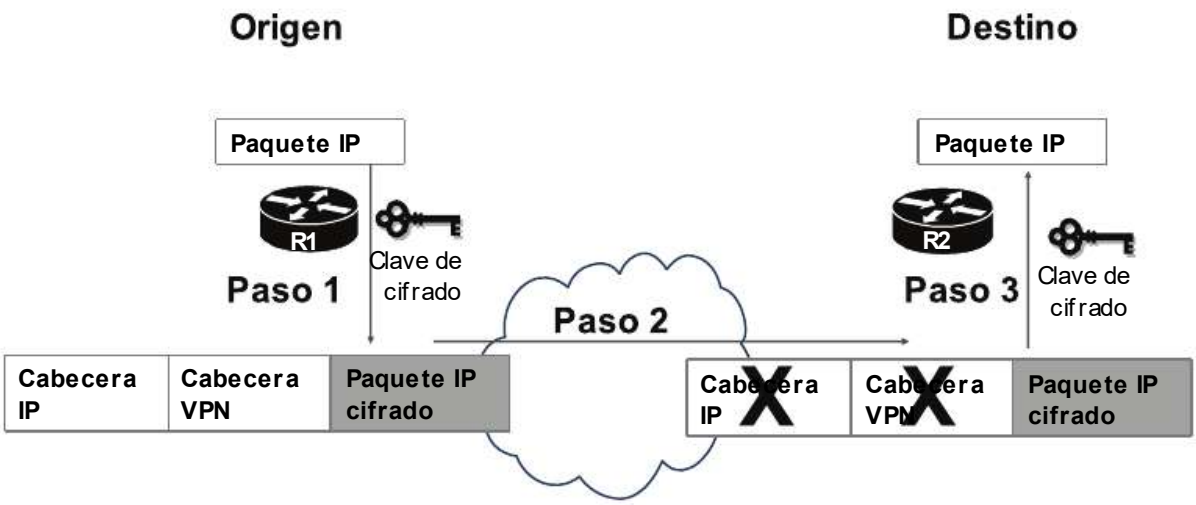


Fig. 9-3 IPSec. Procedimiento de cifrado - descifrado.

En certificaciones específicas de seguridad, como CCNA Security, se profundiza más sobre IPSec y su modo de operación, configuración y protocolos que lo conforman.

SSL

Otro de los protocolos disponible para el transporte de datos de manera segura a través de Internet es SSL *(Secure Sockets Layer)*. A diferencia de IPSec, SSL opera en capa 4 y es ampliamente utilizado en comunicaciones Web para establecer conexiones seguras sobre una gran variedad de servicios, entre los que se incluyen correo electrónico, transacciones bancarias, formularios, identificación, etc. Hace uso del puerto TCP 443 (HTTPS) y provee cifrado de datos.

Además de ello, también puede ser implementado para asegurar el tráfico de una VPN, cumpliendo con las exigencias de seguridad que las mismas requieren. Su modo de operar se basa en la utilización de certificados digitales, los cuales cifrarán y a su vez autenticarán la comunicación llevada a cabo entre ambos extremos. Gracias a ello, resulta posible establecer accesos vía web a servicios corporativos, evitando la instalación y configuración de software cliente VPN en los dispositivos de los usuarios.

Al igual que en IPSec, los detalles técnicos, operación y configuración de SSL son contenido de certificaciones de seguridad más específicas como CCNA Security.

TÚNELES GRE: CONFIGURACIÓN Y VERIFICACIÓN

La comunicación a través de una VPN se hace posible gracias a la aplicación conjunta de diferentes protocolos y procesos, cada uno de ellos con una función específica. Los analizados hasta ahora, de manera muy superficial, proveen seguridad sobre los datos, sin embargo, no ejecutan el proceso de encapsulación y envío de estos. Para ello debe ser aplicado algún otro protocolo que establezca, mantenga y gestione la conexión entre ambos extremos de la VPN. Uno de ellos es GRE, el cual será analizado a continuación.

Protocolo GRE: Conceptos básicos

GRE *(Generic Routing Encapsulation)*, es un protocolo propietario de Cisco que aplicado sobre una VPN define el proceso de encapsulación de datos necesario para que la comunicación entre ambos extremos pueda llevarse a cabo. Su función principal consiste en dar formato a las cabeceras IP y VPN agregadas sobre el paquete IP original ya cifrado, aunque también provee los mecanismos necesarios para el establecimiento, transmisión y gestión de la conexión.

Dicho proceso, denominado *tunneling*, consiste en encapsular un protocolo sobre otro, gracias a lo cual resulta posible transportar datos entre redes de diferente tipo, por ejemplo, que un paquete LAN pueda atravesar un entorno WAN. Para ello, GRE ejecuta las siguientes acciones:

- Primero, sobre los datos ya cifrados, agrega una cabecera propia de GRE, con un tamaño de 4 bytes (32 bits). Esta, entre otros, incluye los campos autenticación (opcional), número de secuencia o protocolo encapsulado, siendo todos ellos necesarios para la comunicación y control entre ambos extremos.

- Segundo, agrega otra cabecera, esta vez IP y denominada "*delivery header*", con una longitud de 20 bytes e incluyendo en la misma, entre otros, los campos dirección de origen y destino. Estas direcciones corresponden a aquellas públicas asignadas a los routers para su comunicación a través de Internet. Los dispositivos intermediarios entre el origen y el destino tomarán la decisión de reenvío en relación con la información incluida en esta cabecera.

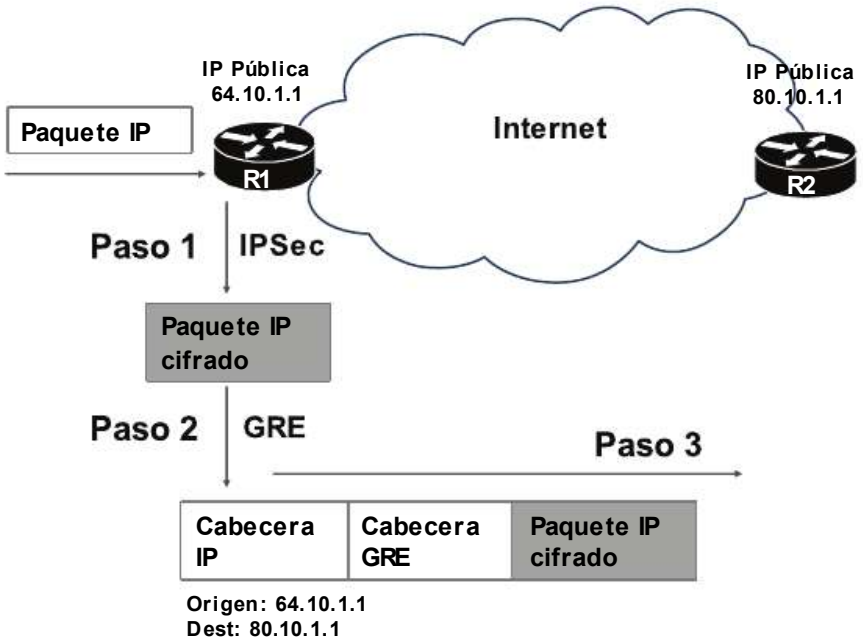


Fig. 9-4 Encapsulado GRE.

- *Paso 1:* R1 recibe un paquete IP que debe ser enviado a través de la VPN. La primera acción que lleva a cabo consiste en cifrarlo con el protocolo configurado para ello, en este caso IPSec.

- *Paso 2:* Los datos cifrados son encapsulados en un nuevo paquete IP por el protocolo de *tunneling* implementado, en este caso GRE, el cual agrega las cabeceras necesarias para su transporte a través de la WAN.

- *Paso 3:* El nuevo paquete es enviado hacia la red pública.

Cada protocolo de *tunneling* soporta diferentes encapsulaciones, siendo este uno de los puntos fuertes de GRE, ya que es capaz de hacerlo hasta con 20 protocolos de capa 3 diferentes.

Por último, GRE no cifra ni provee seguridad alguna sobre los datos, de ello se encarga IPSec o cualquier otro destinado a tal propósito. Los procesos de cifrado y encapsulación son totalmente diferentes y no deben ser confundidos.

Un túnel no establece un enlace físico ni dedicado entre ambos extremos de la VPN. El medio es Internet y por lo tanto los paquetes pueden tomar diferentes rutas para llegar a su destino.

Configuración y verificación de un túnel GRE

El proceso de configuración de GRE en routers Cisco consta de los siguientes pasos, todos ellos ejecutados en los dispositivos de ambos extremos:

- *Paso 1:* Habilitar una interfaz lógica, que será utilizada por el router para gestionar la VPN. Para ello se debe aplicar el comando **interface tunnel *[num],*** desde el modo de configuración global y donde el valor introducido tan solo tiene importancia a nivel local, es decir, si en ambos routers no coincide, el enlace se establecerá sin problemas.

- *Paso 2:* Configurar una IP a dicha interfaz con el comando **ip address *[ip] [máscara]***. Esta debe ser privada y en ambos extremos pertenecer a la misma subred.

- *Paso 3:* Definir la interfaz física o IP pública que utilizará el router para la comunicación a través de la VPN, con el comando **tunnel source *[interfaz o IP pública]***. Si se opta por la interfaz, debe ser aquella que conecta directamente con Internet, en cuyo caso GRE hará uso de su IP pública (previamente configurada). Si por el contrario se definiera directamente la IP pública, GRE haría una búsqueda de la interfaz a la cual pertenece, para acto seguido enviar los paquetes a través de la misma.

- *Paso 4:* Indicar el destino con el comando **tunnel destination *[IP]***, la cual hace referencia a la IP pública del otro extremo de la VPN.

Dada la siguiente topología, configurar una VPN GRE entre TFE y LPA.

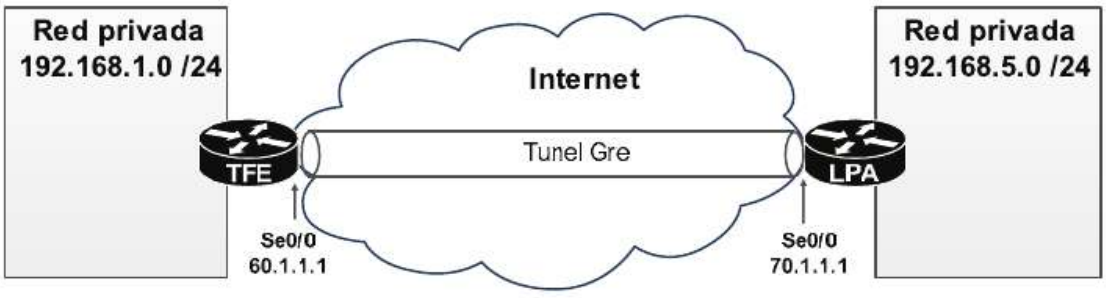


Fig. 9-5 Diseño de red para supuesto práctico de GRE.

```
---Configuración en TFE---

TFE(config)# interface Se0/0
TFE(config-if)#ip address 60.1.1.1 255.0.0.0
TFE(config-if)# exit
```

```
TFE(config)# interface tunnel 1
TFE(config-if)#ip address 192.168.255.1 255.255.255.0
TFE(config-if)# tunnel source Serial0/0
TFE(config-if)# tunnel destination 70.1.1.1

---Configuración en LPA---

LPA(config)# interface Se0/0
LPA(config-if)#ip address 70.1.1.1 255.0.0.0
LPA(config-if)# exit
LPA(config)# interface tunnel 5
LPA(config-if)#ip address 192.168.255.2 255.255.255.0
LPA(config-if)# tunnel source Serial0/0
LPA(config-if)# tunnel destination 60.1.1.1
```

Para que la comunicación concluya con éxito también resulta imprescindible que las tablas de enrutamiento de ambos dispositivos contengan el direccionamiento hacia la red privada del otro extremo, ya sea mediante rutas estáticas o a través de protocolos de enrutamiento dinámico. Para el ejemplo, y haciendo uso de configuración estática…

```
TFE(config)# ip route 192.168.5.0 255.255.255.0 192.168.255.2
LPA(config)# ip route 192.168.1.0 255.255.255.0 192.168.255.1
```

Con los comandos aplicados tan solo se ha creado y establecido el túnel GRE, para el cifrado y seguridad de datos se requiere una configuración adicional que no forma parte del contenido de CCNA.

Una vez creado el túnel podrá ser verificado gracias a los comandos que IOS dispone para ello. En primer lugar, se deberá ejecutar un ping hacia la IP privada del otro extremo, si no existe ningún error y las tablas de enrutamiento son correctas se debería obtener respuesta.

```
TFE#ping 192.168.255.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.255.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/9 ms
```

El túnel se establece a través de una interfaz lógica, por lo tanto, si opera con normalidad su estado debe ser *up/up*. Así pues, otro método de comprobación es mediante un *show ip interface brief.*

```
TFE#show ip interface brief

Interface                  IP-Address      OK? Method Status
Protocol

FastEthernet0/0            192.168.1.1     YES manual up                    up
```

```
FastEthernet0/1        unassigned      YES unset  administratively down down
Serial0/1              unassigned      YES unset  administratively down down
Serial0/0              60.1.1.1        YES manual up                    up
Tunnel1                192.168.255.1   YES manual up                    up
```

Por último, para obtener todos los detalles del túnel se podrá ejecutar un *show interface tunnel [num].*

Gracias a la configuración aplicada, los hosts de las redes privadas 192.168.1.0/24 y 192.168.5.0/24 pueden comunicarse a través de una red pública como Internet.

Supuesto práctico 1: Configurar un túnel VPN GRE entre R1 y R2 haciendo uso de la red 192.168.255.32/30 para comunicar ambos extremos.

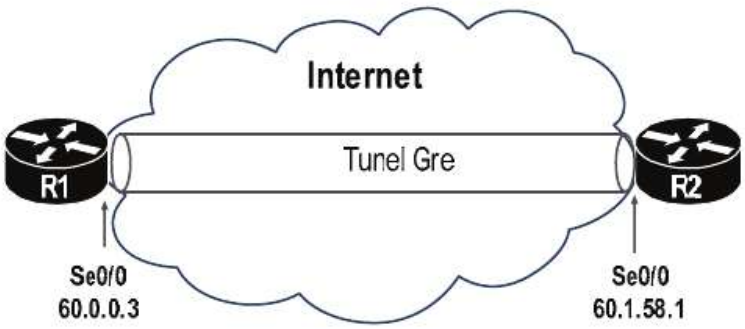


Fig. 9-6 Diseño de red para supuesto práctico 1 de GRE.

```
---Configuración en R1---

R1(config)# interface tunnel 1
R1(config-if)#ip address 192.168.255.33 255.255.255.252
R1(config-if)# tunnel source Serial0/0
R1(config-if)# tunnel destination 60.1.58.1

---Configuración en R2---

R2(config)# interface tunnel 1
R2(config-if)#ip address 192.168.255.34 255.255.255.252
R2(config-if)# tunnel source Serial0/0
R2(config-if)# tunnel destination 60.0.0.3
```

- El número utilizado para la creación de las interfaces túnel puede variar.

- Las direcciones IP privadas pueden ser configuradas a la inversa, es decir, la 192.168.255.33 en R2 y la 192.168.255.34 en R1.

TEST CAPÍTULO 9: REDES PRIVADAS VIRTUALES

1.- ¿Qué beneficios aporta el uso de las VPNs en comparación con otras tecnologías como MPLS? (Seleccionar dos respuestas)

A. Mayor ancho de banda.
B. Escalabilidad.
C. Ahorro de costes.
D. Enlace dedicado punto a punto.
E. Red administrada por terceros.

2.- ¿Cuál es el objetivo principal de una VPN?

A. La comunicación entre diferentes redes privadas a través de una red pública como Internet.
B. Asegurar la comunicación entre redes remotas a través de una red pública como Internet.
C. Que los usuarios de una red privada puedan acceder a contenidos y servicios de otra red privada.
D. Asegurar los datos en capa 3 para su transporte a través de redes públicas.

3.- ¿Qué es IPSec?

A. Un conjunto de protocolos, con diferentes funciones, utilizados para establecer la conexión de extremo a extremo en una VPN, así como garantizar los principios de seguridad exigidos durante la comunicación.
B. Un protocolo de seguridad utilizado por GRE para el cifrado de datos en el túnel VPN.
C. Un conjunto de protocolos de seguridad, cada uno de ellos con una función específica y con el objetivo de garantizar los principios de privacidad, integridad, autenticación y anti-replay exigidos para que una comunicación sea considerada segura.
D. La versión más reciente del protocolo IP.

4.- ¿Qué ataque previenen los protocolos encargados de la función anti-replay?

A. Ataques DDoS.
B. Ataques de diccionario.
C. Ataques de fuerza bruta.
D. Ataques Man-in-the-Middle.

5.- ¿Qué función de seguridad es la encargada de verificar que los datos no sufren ningún cambio ni modificación desde el origen hasta el destino?

A. Privacidad.
B. Integridad.
C. Autenticación.
D. Anti-Replay.

6.- ¿Cómo es posible que una IP privada pueda comunicarse con otra a través de Internet?

A. Gracias al protocolo IPv4.
B. Gracias a la encapsulación llevada a cabo por protocolos específicos para ello.
C. Gracias a la modificación del paquete en capa 3, agregando las direcciones de origen y destino públicas.
D. Gracias a NAT y PAT, que habilitan la comunicación de la red privada con Internet.

7.- Un usuario accede a servicios y aplicaciones de la red privada de su compañía desde su hogar y a través de un navegador web. ¿Qué tipo de VPN está utilizando?

A. SSL.
B. Site-to-site.
C. HTTPS.
D. IPSec.

8.- De las siguientes funciones, ¿cuál es llevada a cabo por GRE?

A. Cifrado.
B. Encapsulación.
C. Entrega fiable.
D. Todas las anteriores.

9.- La configuración de un túnel GRE se lleva a cabo mediante…

A. Una interfaz física.
B. El protocolo de enrutamiento configurado en el dispositivo.
C. La tabla de rutas.
D. Una interfaz lógica previamente creada.

10.- Los routers A y B disponen de la siguiente configuración:

```
---Router A---
A(config)# interface tunnel 1
A(config-if)#ip address 192.168.255.6 255.255.255.252
A(config-if)# tunnel source Serial0/0
A(config-if)# tunnel destination 1.1.1.1
```

```
---Router B---
B(config)# interface tunnel 5
B(config-if)#ip address 192.168.255.9 255.255.255.252
B(config-if)# tunnel source Serial0/0
B(config-if)# tunnel destination 2.2.2.2
```

Sin embargo, no es posible la comunicación entre ambos a través del túnel VPN. ¿A qué puede ser debido?

A. Los números de interfaz en ambos routers no coinciden.
B. Falta por configurar la autenticación en ambos extremos.
C. Las direcciones IP privadas configuradas en ambos extremos no pertenecen a la misma subred.
D. Las IP públicas no pertenecen al mismo rango de red.
E. Ninguna de las anteriores.

11.- Los routers A y B disponen de la siguiente configuración:

```
---Router A---
A(config)# interface tunnel 1
A(config-if)#ip address 192.168.255.6 255.255.255.128
A(config-if)# tunnel source Serial0/0
A(config-if)# tunnel destination 5.5.5.5

---Router B---
B(config)# interface tunnel 5
B(config-if)#ip address 192.168.255.9 255.255.255.128
B(config-if)# tunnel source Serial0/0
B(config-if)# tunnel destination 4.4.4.4
```

Sin embargo, no es posible la comunicación entre ambos a través del túnel VPN. ¿A qué puede ser debido? (Seleccionar dos respuestas)

A. La interfaz serial en uno o ambos routers no ha sido configurada con una IP pública o su estado no es *up/up*.
B. Falta por definir el método de cifrado en ambos extremos.
C. Las direcciones IP públicas configuradas corresponden a una clase A de rango privado.
D. No se ha configurado el enrutamiento, estático o dinámico, para que el router A conozca las redes de B y viceversa.

12.- La configuración de un túnel GRE incluye el comando “tunnel source Se0/1”. ¿Qué IP utilizará el protocolo para encapsular los paquetes que deban ser enviados a través del túnel?

A. Aquella configurada mediante el comando *tunnel source [ip].*
B. La dirección IP pública que utiliza el router para comunicarse en Internet.
C. La dirección IP pública configurada en la interfaz Se0/1.
D. Cualquier IP pública disponible en el pool de direcciones configurado.

13.- ¿Cuál de los siguientes comandos verifica que una interfaz túnel se encuentra en estado up/up? (Seleccionar dos respuestas)

A. Show ip protocols.
B. Show ip interface brief.
C. Show interface tunnel *[num].*
D. Show running-config.

14.- De las siguientes características, ¿cuál identifica una ventaja de GRE sobre otros protocolos de encapsulado VPN?

A. La velocidad de envío.
B. La robustez en el cifrado de datos.
C. El tamaño de tramas.
D. La cantidad de protocolos que es capaz de encapsular.
E. La escalabilidad.
F. La interoperabilidad.

15.- ¿Cuál de los siguientes protocolos puede ser configurado para establecer un túnel VPN?

A. IPSec.
B. SSL.
C. Site-to-Site.
D. DES.
E. Ninguno de los anteriores.

REDES WAN. TIPOS Y PROTOCOLOS 10

CONCEPTOS BÁSICOS

A lo largo de los diferentes capítulos han sido analizados multitud de protocolos y tecnologías, la gran mayoría de ellas aplicables tan solo en redes privadas cuyo alcance geográfico abarca distancias relativamente cortas. Para muchas compañías este hecho supone una limitación en cuanto a conexión entre sedes se refiere, en ocasiones situadas a cientos o miles de kilómetros de distancia, optando por la implementación de una red WAN para solventar el problema.

Por definición, una red WAN *(Wide Area Networks)* es aquella que abarca una distancia geográfica tan amplia que resulta prácticamente ilimitada, siendo capaz de interconectar dispositivos ubicados en cualquier parte del mundo. Ello es posible gracias a la implementación de medios físicos y protocolos específicos en capa 1 y 2, siendo analizados en párrafos posteriores. Su acceso se lleva a cabo a través de algún proveedor de servicios (ISP - *Internet Service Provider*), los cuales disponen de la tecnología necesaria para que la conexión resulte viable y fiable, tanto física (cableado, hardware, etc.) como administrativamente (enrutamiento, seguridad, etc.). Sus características más destacadas son:

- Soportan diferentes tipos de tráfico, como voz, vídeo o datos.
- Soportan áreas geográficas extensas.
- Su acceso se lleva a cabo a través de un ISP, disponiendo de diferentes tecnologías para ello, como cable, cobre, fibra, 3G/4G o satélite, entre otras.
- Operan en capa 1 (medios físicos) y capa 2 (protocolos).

En capa 1, los elementos más importantes y a través de los cuales fluye la comunicación son los siguientes, todos ellos propiedad y administrados por el ISP:

CO (Central Office): Es el edificio central donde se ubican los dispositivos encargados de la comunicación a través de la WAN. Su función es la de recibir y enrutar tráfico hacia su destino correcto.

Switch WAN: Los switchs WAN son conmutadores que forman parte del ISP pero que no se encuentran ubicados en el CO. Son distribuidos a lo largo de la ciudad, provincia o país y su función consiste en reenviar la comunicación desde los clientes hasta el CO.

Punto de demarcación: Es el elemento de la LAN que da acceso al cableado WAN. Por ejemplo, un cajetín de telefonía (cobre) o una ONT (fibra). El punto de demarcación establece el límite entre la LAN, administrada por la compañía, y la WAN, administrada por el ISP.

CSU/DSU: Es un dispositivo instalado por el ISP y ubicado físicamente en la red del cliente que ejerce la función de establecer la conexión entre la WAN y la LAN. Para ello, por un extremo conecta con el punto de demarcación y por el otro con el router de la red local.

Router: El router que tendrá acceso a la WAN. Este debe conectar a través de alguna de sus interfaces con el CSU/DSU.

CPE: Los CPE identifican a todos los dispositivos ubicados físicamente en la LAN del cliente, pero cuya misión consiste en establecer una conexión con la WAN. De los analizados hasta ahora, los CPE son el CSU/DSU y el router.

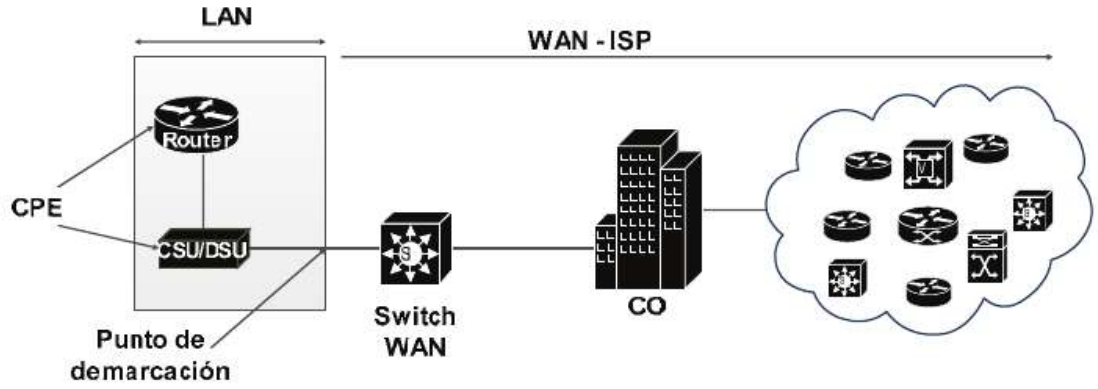


Fig. 10-1 Elementos físicos en red WAN.

En cuanto a interconexión entre sitios remotos se refiere, las opciones disponibles permiten la creación de dos tipos de entornos, privados, para los cuales se hace necesario el uso de tecnologías como Frame Relay, MPLS, Ethernet WAN o VSAT, entre otros, y públicos, como Internet, cuyas modalidades de acceso más comunes son DSL, ISDN, Cable o 3G/4G. Todos ellos forman parte del contenido de CCNA y por lo tanto analizados a continuación.

TECNOLOGÍAS DE ACCESO A REDES WAN

Dependiendo del método de acceso y tecnología aplicada, el propósito de la red WAN varía, diferenciando entre aquellas de ámbito privado y aquellas otras destinadas al intercambio de información y servicios públicos, como Internet. La principal diferencia entre ambas, aparte de la ya mencionada tecnología, reside en el nivel de seguridad disponible. En las privadas, la comunicación entre ambos extremos debe llevarse a cabo a través de algún medio o protocolo que garantice la privacidad total de los datos, a los que solo podrá acceder el cliente que lo ha contratado. El encargado de proveerla es el ISP. Mientras, en las públicas, el mismo medio es compartido por todos los clientes que acceden a la misma, convirtiéndolas en un entorno inseguro y poco fiable. En estos casos, además de las medidas tomadas por el ISP, los clientes deberán implementar los métodos necesarios para asegurar sus comunicaciones, como puede ser el uso de HTTPS, SFTP, etc.

La creación de una WAN privada puede llevarse a cabo mediante las siguientes tecnologías:

- Líneas arrendadas (*leased lines*).
- Frame Relay.
- Ethernet WANs.
- MPLS.
- VSAT.

Mientras, para acceder a Internet (WAN pública) se podrá optar por alguna de las siguientes:

- ISDN.
- DSL.
- Cable.
- 3G/4G.

Redes WAN Privadas

LÍNEAS ARRENDADAS (LEASED LINES)

Las líneas arrendadas, también denominadas enlaces seriales *(serial links)* representan un circuito privado y dedicado entre dos extremos a través de una WAN, estableciendo un enlace punto a punto con el fin de habilitar una red privada entre ambos. Se puede entender como una conexión directa entre dos routers, siempre activa y sin límite de distancia. También reciben el nombre de "líneas de datos" o "circuitos privados" y es considerado un servicio de capa 1 entre cliente y proveedor, aunque evidentemente para el transporte de datos resultan necesarios protocolos de capa 2, normalmente HDLC o PPP.

En capa 1, la tecnología aplicada dependerá del ISP, siendo la más común TDM *(Multiplexación por división de tiempo)*, con el fin de garantizar la operatividad y fiabilidad del enlace.

En este tipo de circuito, la conexión del router con la red WAN se lleva a cabo a través de una interfaz serial.

FRAME RELAY

Otra manera de crear redes WAN privadas consiste en la aplicación de Frame Relay. Este modelo, a diferencia de las líneas arrendadas, se basa en un entorno punto-multipunto, es decir, un mismo origen puede enlazar con múltiples destinos.

El diseño de red resulta sencillo, donde cada dispositivo conecta a través de un solo enlace con un switch FR, y este, a su vez, establece un circuito virtual hacia cada destino, asignando a cada uno de ellos un identificador denominado DLCI *(Data Link Connection Identifier)*, gracias al cual se lleva a cabo el direccionamiento y cuyas funciones coinciden con las llevadas a cabo por las direcciones MAC en una LAN, por lo que se puede deducir que Frame Relay opera en capa 2. Los switchs se encuentran ubicados en la WAN y son administrador por el ISP.

Aunque FR ha caído en desuso debido al desarrollo de métodos más eficientes, aún resulta útil para determinados fines.

ETHERNET WAN

Ethernet define una serie de estándares en capa 2 comúnmente aplicados sobre redes LAN, cuyos detalles han sido analizados en capítulos anteriores. Una de sus

características hace referencia al cableado y su longitud máxima, siendo de 100 metros para UTP y limitando con ello el alcance geográfico de este tipo de redes. Sin embargo, gracias a la aparición de la fibra dichas distancias aumentan de manera considerable, permitiendo establecer un entorno WAN privado con Ethernet como tecnología de comunicación.

Para lograrlo, el ISP instala switchs en la WAN haciendo uso de fibra como medio físico, hecho que habilita una velocidad y rendimiento superior. Dichos conmutadores establecen una red punto-multipunto cuyo diseño lógico se asemeja bastante a Frame Relay, no así los protocolos y medios necesarios en capa 1 y 2. Además, el direccionamiento se lleva a cabo igual que en la LAN, mediante direcciones MAC.

Diferentes servicios ofrecidos por los ISP, como Metro Ethernet *(MetroE)* o VPLS *(Virtual Private Lan Service)* hacen uso de esta tecnología.

MPLS

MPLS *(Multiprotocol Label Switching)* se basa en un modelo muy similar a Frame Relay y Ethernet WAN, pero incluyendo una característica que lo diferencia de los anteriores, y es que opera en capa 3, lo que se traduce en nuevas opciones y beneficios sobre la comunicación.

Existen diferentes tipos y servicios basados en esta tecnología, sin embargo, el CCNA tan solo abarca las MPLS VPN, cuyo propósito consiste en crear redes privadas con la capacidad de transportar y comunicar múltiples protocolos y tipos de tráfico, habilitando también la posibilidad de aplicar QoS *(Quality of Service)* sobre el mismo.

En este caso, los routers conectan con la WAN mediante cualquier interfaz que soporte tráfico IP, para a posteriori ser transportado y enrutado por dispositivos pertenecientes al ISP.

VSAT

Por último, VSAT permite la creación de redes WAN privadas sin necesidad de cableado, normalmente utilizada en lugares remotos o poco accesibles donde no existe instalación física por parte del ISP, bien porque no es posible realizarla, bien porque no resulta rentable.

En estos casos, la comunicación se lleva a cabo a través de un satélite. Para ello, el router ubicado en cada sede debe conectar con un terminal VSAT (similar a una

antena parabólica), el cual será el encargado de establecer el enlace con el satélite y este último de comunicar las diferentes sedes.

En los ejemplos de Frame Relay, Ethernet WANs y MPLS se representan 3 switchs/routers ubicados en la WAN con el fin de facilitar la compresión. Evidentemente, la comunicación de un extremo a otro atraviesa multitud de dispositivos más.

Acceso a redes WAN públicas (Internet)

Internet es, sin lugar a duda, la red pública más conocida y accesible a nivel mundial, compuesta por millones de dispositivos e incontables servicios y prestaciones de los que hacen uso diariamente infinidad de personas. Ello es posible gracias a la interconexión entre los ISP de los diferentes países, habilitando una red sin límites geográficos, y lo que es más importante, permitiendo el acceso desde cualquier parte del planeta, el cual deberá llevarse a cabo mediante la contratación de alguno de los servicios ofrecidos por los ISP para ello. Los más comunes son:

ISDN

ISDN *(Integrated Services for Digital Networks)* consta como una de las primeras tecnologías desarrolladas para este propósito, la cual hace uso de la red de telefonía (circuitos de cobre) como medio de transporte y señales analógicas como método de transmisión. El problema reside en que los PCs y dispositivos operan con señales digitales, por lo que resulta necesaria la instalación de un módem ISND que lleve a cabo la conversión entre ambas. El proceso es el siguiente:

- *Paso 1:* El cliente, por ejemplo, un PC, genera señales digitales cada vez que requiera acceder a algún servicio de datos.

- *Paso 2:* Estas son recibidas por un módem ISDN, que se encargará de convertirlas en analógicas para acto seguido enviarlas a la red WAN a través de la línea telefónica.

- *Paso 3:* Tras ello, la comunicación es procesada y enrutada por el ISP.

- *Paso 4:* La respuesta será recibida mediante señales analógicas, que el módem transforma en digitales para acto seguido reenviarlas al PC.

Pero, si a través del cobre solo se transmiten señales analógicas, ¿cómo diferencia el ISP que la comunicación es de datos y no de voz? Realmente la conexión de datos fluye a través de una llamada telefónica. Para ello, el módem ISND realiza una

llamada a un PoP *(Point of Presence)* ubicado por el ISP en algún punto de la región. A partir de ahí se establece un enlace físico entre ambos y toda la comunicación que fluya a través del mismo será considerada de datos. La función del PoP consiste en conectar analógicamente con los módems de los clientes, convertir las señales recibidas a digitales y enrutarlas hacia Internet.

En cuanto al ancho de banda, existen diferentes modalidades disponibles en ISDN. La más básica, BRI *(Basic Rate Interface)*, hace uso de una línea dividida en dos canales de 64 Kb cada uno, por lo tanto, alcanza una velocidad de 128 Kb. Posteriormente fueron desarrolladas opciones más avanzadas, entre las que se encuentran las líneas T1, las cuales disponen de 24 canales de datos, cada uno de 64 Kb, lo que suman un total de 1536 Kb.

A día de hoy, y gracias a la aparición de nuevas tecnologías, estas velocidades son consideradas excesivamente bajas. Debido a ello, ISDN se ha convertido en un servicio en desuso, aunque aún resulta posible su contratación a precios relativamente económicos.

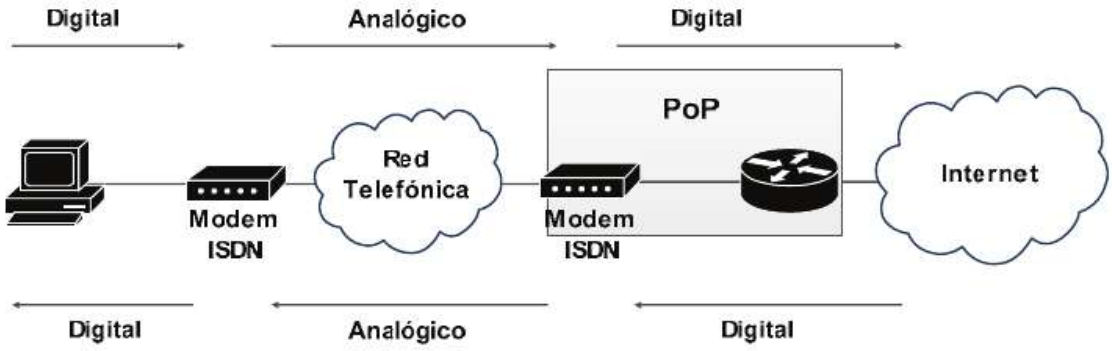


Fig. 10-2 Comunicación ISDN.

DSL

El desarrollo de la tecnología DSL *(Digital Subscriber Line)* supuso una mejora bastante significativa en cuanto a métodos de acceso a Internet se refiere. Esta también hace uso de la red de telefonía como medio físico, sin embargo, y a diferencia de ISDN, envía las señales de datos directamente en formato digital, obteniendo gracias a ello notables ventajas como un mayor ancho de banda.

El modo de operar puede ser dividido en dos. Primero, en el lado del cliente, los datos generados para el acceso a Internet son enviados al módem DSL en formato

digital, y este a su vez los reenvía a la red de telefonía sin aplicar sobre ellos ningún tipo de conversión. Sin embargo, las llamadas telefónicas (voz) continúan haciendo uso de transmisión analógica. Es decir, el cliente generará dos tipos de señal que enviará a través del mismo medio físico. Segundo, el ISP deberá dividirlas y gestionarlas por separado. Este proceso se lleva a cabo gracias a un dispositivo denominado DSLAM, el cual las detecta y reenvía a la red adecuada para que la comunicación concluya con éxito. Los datos son reenviados a un router y este a su vez a la red WAN, mientras que las señales analógicas a un switch de voz. El DSLAM es instalado por el ISP en algún punto de la región y todas las comunicaciones generadas por los clientes que han contratado el servicio serán recibidas y gestionadas a través del mismo.

En cuanto al ancho de banda disponible, cada ISP suele ofrecer diferentes modalidades, que normalmente varían entre 1 y 20 mb de bajada, mientras que la velocidad de subida suele ser siempre inferior.

Por último, la distancia física entre el cliente y el DSLAM determinará la velocidad máxima de acceso al servicio DSL, a mayor distancia, menor velocidad.

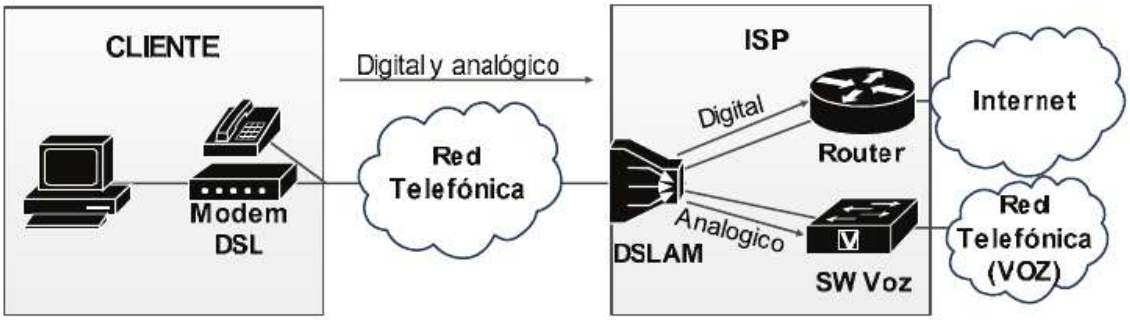


Fig. 10-3 Comunicación DSL.

CABLE

Cable representa una tecnología completamente diferente a ISND y DSL en cuanto a medio físico se refiere, sin embargo, mantiene bastantes similitudes con el modo de operación de DSL.

En este caso, el medio físico corresponde a un cable coaxial, a través del cual se enviarán dos tipos de señal, ambas digitales, pero una de datos y otra de TV. La manera de diferenciarlas consiste en la aplicación de diferentes frecuencias, siendo una de ellas dedicada exclusivamente para datos. Estas, al ser enviadas a través del

mismo circuito, serán recibidas conjuntamente por el ISP, el cual se encargará de redirigirlas hacia la red correcta.

En cuanto a ancho de banda, cable ofrece velocidades muy similares a DSL, resultando también conexiones asimétricas.

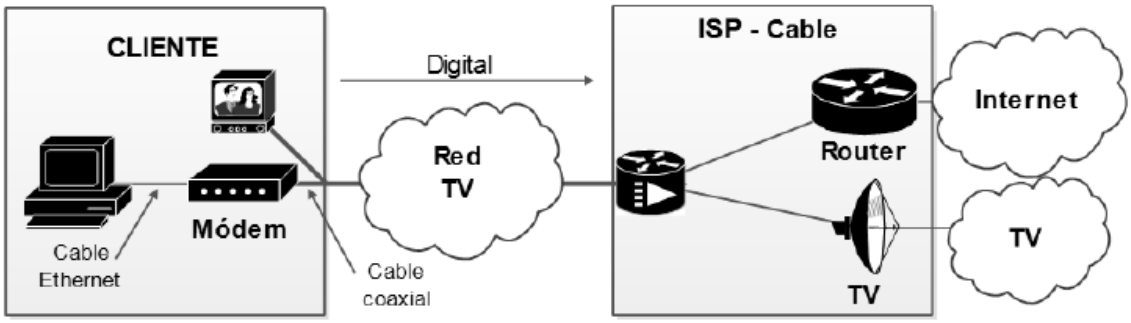


Fig. 10-4 Comunicación cable.

Cable y DSL representan tecnologías de acceso a Internet de conexión permanente y siempre operativas, sin embargo, en ISDN resulta necesario establecer una llamada (el módem) para iniciar la comunicación, por lo que no es considerado un enlace permanente.

COMUNICACIÓN MÓVIL

A día de hoy, el 99% de las operadoras de telefonía móvil incluyen conexión de datos en la totalidad de sus tarifas, es más, el mercado y la comunicación en general se enfocan cada vez más en poder acceder a cualquier tipo de servicios a través del Smartphone. Correo electrónico, banca online, juegos, redes sociales, mensajería, web y aplicaciones requieren acceso a Internet, para lo cual han sido desarrollados diferentes protocolos. Dos de ellos, 3G y 4G, son los más conocidos e implementados.

Este tipo de comunicación resulta posible gracias a torres de telefonía equipadas con potentes antenas instaladas por el ISP a lo largo de la región. Estas se comunican entre sí creando “celdas”, todas ellas interconectadas y desarrollando las mismas funciones. Cada celda abarca un espacio físico limitado, y a su vez, todas ellas unidas determinan el rango de cobertura móvil que ofrece el ISP.

En cuanto a los dispositivos, cada uno dispone de una pequeña antena, de potencia limitada pero suficiente para establecer la conexión inalámbrica con la torre más cercana. Este proceso puede ser resumido en los siguientes pasos:

- *Paso 1:* El dispositivo móvil inicia una búsqueda de señales de radio transmitidas por las torres de telefonía que tiene a su alcance. Si localiza aquellas de su proveedor significa que dispone de cobertura por lo que continúa con el paso 2, de lo contrario no conectará con ninguna red y prosigue analizando señales hasta detectar la adecuada.

- *Paso 2:* Se inicia un proceso de negociación entre la torre y el dispositivo que concluye con el registro de este en la red de telefonía móvil del proveedor contratado. En este paso, el dispositivo pertenece a una determinada celda, sin embargo, puede cambiar físicamente de lugar, comunicándose a través de diferentes torres sin perder el registro en la red ni mucho menos la conectividad.

- *Paso 3:* El dispositivo puede realizar llamadas (voz) y acceder a Internet (datos) de manera inalámbrica.

La comunicación se lleva a cabo mediante señales de radio, donde el cliente emite voz o datos que son recibidos por la antena más cercana, y esta, que estará cableada a dispositivos de enrutamiento, dividirá y enviará la comunicación a la red que considere oportuna. En el caso de Internet, a routers propiedad del ISP.

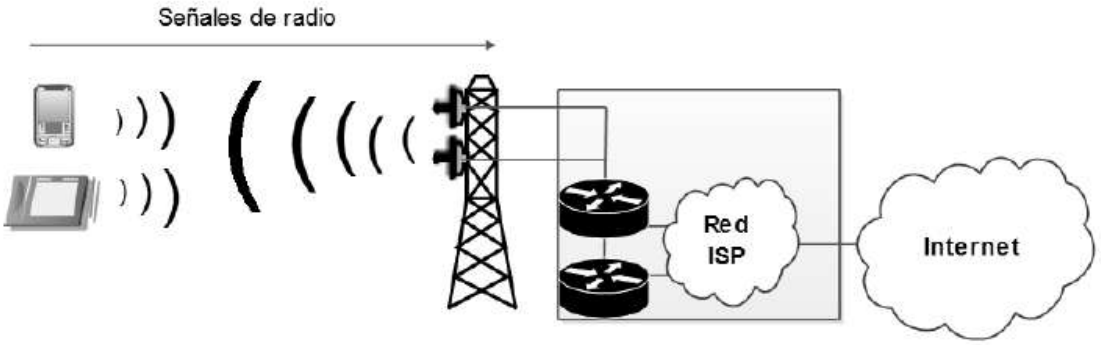


Fig. 10-5 Comunicación en red móvil.

Los protocolos de conectividad móvil no forman parte del contenido de CCNA.

PROTOCOLOS WAN EN CAPA 2: HDL, PPP Y PPPOE

Uno de los métodos disponibles para la creación de una red WAN privada son las líneas arrendadas, cuya misión, como se ha analizado recientemente, consiste en establecer un circuito dedicado entre dos dispositivos, normalmente routers, con el objetivo de habilitar un enlace entre ambos sin límite de distancia y utilizando para ello diferentes medios en capa 1 y protocolos en capa 2.

En capa física, el circuito será seleccionado y depende por completo del operador de servicios, siendo un enlace siempre disponible, de conexión permanente, y haciendo uso de *full duplex* como modo de transferencia. Mientras, los dispositivos encargados de gestionar la comunicación serán el router y el CSU/DSU instalados por el ISP en ambos extremos de la comunicación, donde el CSU/DSU desarrolla dos funciones principales:

- Primero, establece la frecuencia de reloj, actuando como DCE en el enlace serial que conecta con el router, el cual toma el rol de DTE, aceptando y transmitiendo a la velocidad establecida. Cualquier otra combinación (ambos extremos en DCE, ambos extremos en DTE, DCE sin frecuencia de reloj configurada, etc.) dará como resultado que la comunicación no sea posible.

- Segundo, aplica TDM como método de transmisión de datos a través de la WAN.

Resumiendo, el CSU/DSU conecta por un extremo con un router y por otro extremo con la WAN. Con el primero marca la frecuencia de reloj y con el segundo aplica TDM para el envío de datos.

A día de hoy, los routers corporativos Cisco suelen incorporar la función de CSU/DSU, o bien permiten agregar módulos de expansión para este y otros propósitos, logrando que un solo dispositivo disponga de todas las funciones anteriormente descritas.

El objetivo principal en capa 1 consiste en definir los medios necesarios para establecer la conexión física y transportar los datos de un extremo a otro. Mientras, en capa 2, los protocolos se encargarán del direccionamiento y transmisión de la información. Los más comunes en enlaces punto a punto en redes WAN son HDLC, PPP y PPPoE, analizados a continuación.

HDLC: Características y configuración

HDLC *(High Level Data Link Protocol)* es uno de los protocolos disponibles para el direccionamiento de datos en enlaces WAN punto a punto. Su misión principal consiste en definir el formato a aplicar en capa 2 para que la comunicación entre ambos extremos resulte posible. Para ello, el paquete IP es encapsulado en una trama, la cual estará compuesta por los siguientes campos:

Delimitador 8 bits	Dirección 8 bits	Control 8 bits	Datos Long. Variable	CRC 16 o 32 bits	Delimitador 8 bits

- *Delimitador:* Señalizan el principio y final de la trama. Su valor siempre equivale a 01111110.

- *Dirección:* Hace referencia a la dirección de destino. Al tratarse de un enlace punto a punto tan solo existe un posible destinatario, por lo que su valor suele ser 11111111, que equivale a un broadcast.

- *Control:* Identifica el tipo de trama HDLC, pudiendo ser de información, control, o no numeradas. Los detalles de estas no forman parte del contenido de CCNA.

- *Datos:* Contiene el paquete encapsulado proveniente desde capa 3. Es de longitud variable, pero debe cumplir la condición de que el número de bits que lo componen debe ser múltiplo de 8. Si los datos incluidos no cumplen dicho requisito se aplica un relleno de bits.

- *CRC:* Necesario para el control de errores. Contiene el resultado de un algoritmo matemático aplicado sobre el número total de bits de la trama (excluyendo los delimitadores). En el destino, el router que la recibe ejecuta el mismo proceso sobre los datos recibidos. Si el valor coincide significa que no se han producido errores durante la transmisión, sin embargo, si no fuera así, la trama es descartada.

De todos ellos, el delimitador inicial, dirección y control definen la cabecera de la trama, mientras que los campos CRC y delimitador final componen el tráiler.

La primera acción que lleva a cabo HDLC antes de proceder al envío de datos consiste en llevar a cabo una negociación entre ambos extremos con el fin de establecer el enlace y en la cual se definen diferentes parámetros como el protocolo de capa 3 a transportar. Esta precisamente identifica una de las mayores desventajas

de HDLC, ya que tan solo encapsulará y transportará uno, suponiendo un problema sobre entornos multiprotocolo, por ejemplo, redes que operan con IPv4 e IPv6 de manera simultánea.

Con el fin de solucionarlo, Cisco modifica la trama e incluye un nuevo campo, utilizado para identificar el tipo de datos encapsulados. Este detalle supone una mejora significativa ya que gracias a ello resulta posible transportar diferentes protocolos de capa 3 a través del mismo enlace. Aun siendo propietario de Cisco, se permite su aplicación en diferentes fabricantes, por lo este tipo de HDLC está disponible en gran cantidad de dispositivos.

El formato, en este caso, es el siguiente:

Delimitador 8 bits	Dirección 8 bits	Control 8 bits	Tipo 16 bits	Datos Long. Variable	CRC 16 o 32 bits	Delimitador 8 bits

Sea cual sea el tipo de HDLC aplicado, una vez negociado y establecido el enlace, el proceso de envío de una trama recibida desde la LAN a través de un enlace WAN consta de:

- *Paso 1:* La trama recibida normalmente será Ethernet, por lo que la primera acción a realizar consiste en eliminarla.

- *Paso 2:* Crear una trama HDLC para encapsular el paquete IP.

- *Paso 3:* Enviarla través del enlace WAN.

- *Paso 4:* Una vez recibida, el router del otro extremo elimina la trama HDLC y encapsula el paquete IP dentro de una nueva trama 802.3 para acto seguido reenviarla a la LAN.

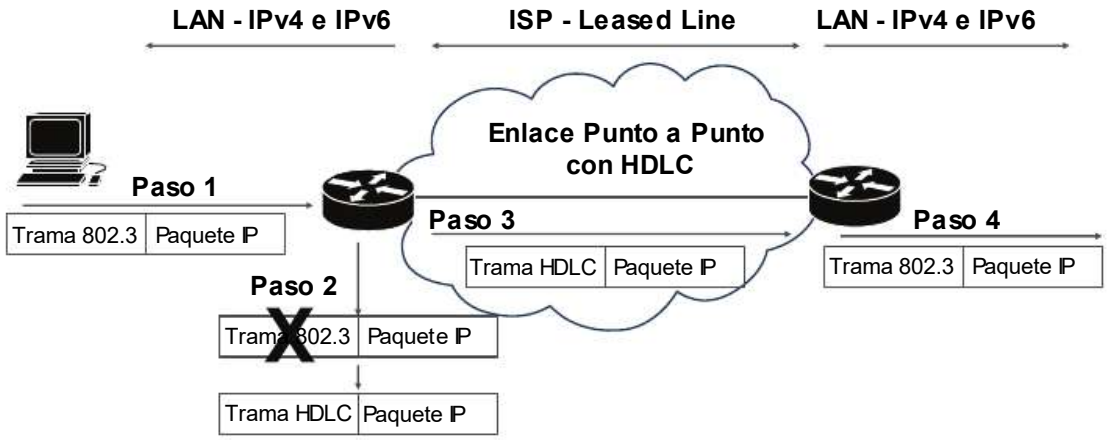


Fig. 10-6 Proceso de creación de trama HDLC.

CONFIGURACIÓN DE HDLC

HDLC es el protocolo aplicado por defecto en routers Cisco sobre enlaces seriales, aun así, su configuración manual se lleva a cabo a través del siguiente procedimiento:

- *Paso 1:* Definir una dirección IP a la interfaz serial con el comando **ip address *[dir IP]* *[máscara]***. Ambos extremos deben pertenecer al mismo rango de red.

- *Paso 2:* Aplicar HDLC mediante la sentencia **encapsulation hdlc**, desde el modo de configuración de la interfaz.

- *Paso 3:* Habilitarla con el comando **no shutdown**.

En entornos de prácticas de laboratorio como Packet Tracer o simulaciones en el examen de certificación también resulta necesario configurar los routers para que actúen como DCE o DTE. El primero marcará la frecuencia de reloj que se utilizará para el envío de datos, mientras que el DTE simplemente la acepta y aplica. En estos casos tan solo se debe ejecutar, en el router que actúa como DCE, el comando **clock rate *[frecuencia]*** desde el modo de configuración de la interfaz serial, donde *frecuencia* indica la velocidad de reloj, en bps. En entornos reales esta configuración es definida por el ISP.

Ejemplo: Configurar el enlace serial entre TFE y LPA para que opere con HDLC a una frecuencia de reloj de 64 kbps.

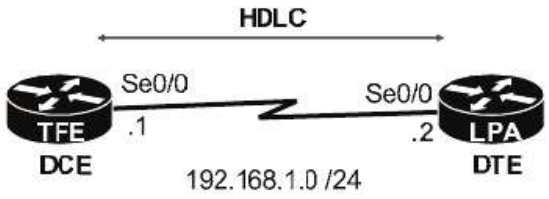
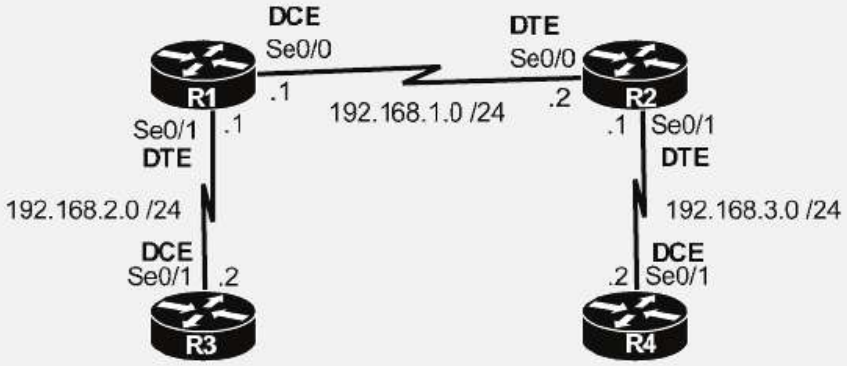


Fig. 10-7 Enlace entre routers para supuesto práctico de HDLC.

```
---Configuración en TFE---
TFE(config)#int se0/0
TFE(config-if)#ip address 192.168.1.1 255.255.255.0
TFE(config-if)#encapsulation hdlc
TFE(config-if)#clock rate 64000
TFE(config-if)#no shutdown

---Configuración en LPA---
LPA(config)#int se0/0
LPA(config-if)#ip address 192.168.1.2 255.255.255.0
LPA(config-if)#encapsulation hdlc
LPA(config-if)#no shutdown
```

Reto 10.1 - Configurar la siguiente topología de tal manera que todos los enlaces seriales hagan uso de HDLC con una frecuencia de reloj de 56000 bps.



Solución al final del capítulo.

PPP: Características y configuración

HDLC es considerado un protocolo muy básico y con seguridad nula, debido en gran parte a su fecha de creación, donde la simplicidad en las comunicaciones era mayor y las amenazas de seguridad prácticamente inexistentes. Con el paso del tiempo se hizo necesario solucionar sus carencias, y a razón de ello nace

PPP *(Point-to-Point Protocol)*, un protocolo de capa 2 para enlaces WAN punto a punto cuya función, al igual que en HDLC, consiste en encapsular paquetes de capa 3 en tramas de capa 2 para posteriormente ser transmitidas a través del medio físico. Es más, el formato es idéntico al utilizado en la versión HDLC de Cisco, con el campo "*Tipo*" para identificar el protocolo encapsulado, de tal manera que:

Delimitador 8 bits	Dirección 8 bits	Control 8 bits	Tipo 16 bits	Datos Long. Variable	CRC 16 o 32 bits	Delimitador 8 bits

Donde cada uno de los campos mantiene la misma función que los ya analizados en HDLC.

Sin embargo, aunque la trama defina el mismo formato, se agregan algunas características que mejoran y aseguran la comunicación, como:

- Métodos más eficientes durante la negociación del enlace, incluyendo el establecimiento, mantenimiento y finalización de este, gracias al protocolo LCP.

- Mecanismos de control sobre los protocolos de capa 3 que serán transportados, encargándose de ello NCP.

- Autenticación entre ambos extremos haciendo uso de PAP o CHAP.

PROTOCOLO LCP (LINK CONTROL PROTOCOL)

Como su nombre indica, LCP provee los mecanismos de control necesarios para llevar a cabo las funciones de establecimiento, mantenimiento y finalización del enlace cuando resulte necesario. La primera acción que lleva a cabo PPP es hacer uso del mismo para iniciar la negociación entre ambos extremos, la cual se basa principalmente en identificar al dispositivo vecino, determinar el tamaño de tramas y detectar errores o configuraciones no coincidentes. Si concluye con éxito, se establece el enlace en capa 2 entre ambos routers, de lo contrario, se detiene el proceso de manera inmediata.

LCP desarrolla siempre las mismas funciones y las ejecuta de la misma manera sin importar los protocolos de capas superiores encapsulados.

PROTOCOLOS NCP (NETWORK CONTROL PROTOCOLS)

Una de las características y mejoras que agrega PPP sobre la comunicación con respecto a HDLC es el control y soporte de protocolos de capas superiores, mejorando el encapsulado y transporte de cada uno de ellos. Dicha función es posible gracias a NCP, el cual define un conjunto de protocolos dedicados a manejar las necesidades específicas para el transporte de datos de capa 3 sobre cada tipo de entorno. Por ejemplo, para encapsular IPv4, NCP hace uso de IPCP, para IPv6, IPv6CP, para CDP, CDPCP, etc. Gracias a ello se logra una comunicación más fiable y efectiva sobre enlaces punto a punto.

PROTOCOLOS DE AUTENTICACIÓN PAP Y CHAP

En cuanto a seguridad se refiere, PPP incluye la opción de autenticación entre ambos extremos, de tal manera que solo los dispositivos autorizados podrán formar parte del enlace. Dicha característica agrega una capa de seguridad importante, ya que cada uno de ellos verifica la identidad del otro para establecer el link y por supuesto antes de enviar cualquier tipo de datos a través del mismo, evitando con ello determinados ataques. Aunque conste como un parámetro opcional, su configuración resulta altamente recomendable.

Existen dos protocolos para tal propósito, PAP *(Password Authentication Protocol)* y CHAP *(Challenge-Handshake Authentication Protocol)*. El modo de operar en ambos casos se basa en un intercambio de mensajes entre los dispositivos, el cual se lleva a cabo durante el proceso de negociación ejecutado por LCP. Si la autenticación falla, no se establece el enlace.

Aunque el propósito final coincida, disponen de características propias, siendo la diferencia más notable que PAP envía los mensajes en texto plano, mientras que CHAP aplica el algoritmo de cifrado MD5 sobre las credenciales de autenticación, convirtiéndolo en la opción más segura.

En resumen, PPP es un protocolo de capa 2 que tiene como objetivo habilitar enlaces WAN punto a punto y que a su vez hace uso de diferentes protocolos para garantizar una comunicación fiable y segura. Para establecerlo, mantenerlo y finalizarlo hace uso de LCP, que a su vez aplica PAP o CHAP como método de autenticación. Además, gracias a NCP da soporte a los diferentes protocolos de capa 3 encapsulados.

CONFIGURACIÓN DE PPP CON AUTENTICACIÓN CHAP

En routers Cisco, el proceso de configuración de PPP con autenticación CHAP sobre enlaces seriales consta de las siguientes acciones:

- *Paso 1:* Habilitar PPP como protocolo en capa 2, con el comando **encapsulation ppp** desde el modo de configuración de la interfaz serial.

- *Paso 2:* Aplicar CHAP como método de autenticación, gracias a la sentencia **ppp authentication chap,** también desde el modo de configuración de la interfaz serial.

- *Paso 3:* Crear un usuario local y contraseña para verificar las credenciales del otro extremo, mediante el comando **username *[usuario]* password *[pass],*** ejecutado desde el modo de configuración global y donde *"usuario"* corresponde al nombre del router remoto, mientras que "*pass"* establece la contraseña a aplicar en PPP. Por ejemplo, en un enlace formado entre R1 y R2 se desea aplicar la contraseña "secret". En R1 se deberá ejecutar el comando *"username R2 password secret",* mientras que en R2 *"username R1 password secret".*

Al igual que en HDLC, las prácticas de laboratorio y simulaciones de examen de CCNA requieren la configuración de la frecuencia de reloj en la interfaz del router que actúe como DCE.

Ejemplo: Configurar PPP con autenticación CHAP sobre el enlace serial entre TFE y LPA cumpliendo los siguientes requisitos:

- Frecuencia de reloj en el router que actúa como DCE: 64Kbps.
- Nombres de los routers: TFE y LPA.
- Contraseña CHAP: "secretPASS".

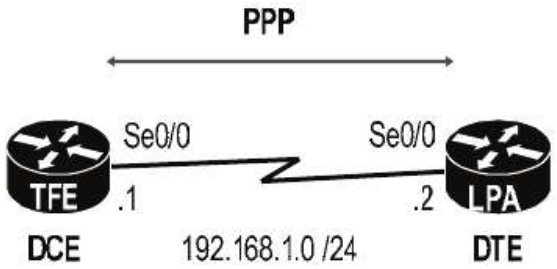


Fig. 10-8 Enlace entre routers para supuesto práctico de PPP.

```
--- Configuración en TFE---

Router(config)#hostname TFE
TFE(config)#int Se0/0
TFE(config-if)#ip address 192.168.1.1 255.255.255.0
TFE(config-if)#clock rate 64000
TFE(config-if)#encapsulation ppp
TFE(config-if)#ppp authentication chap
TFE(config-if)#no shutdown
TFE(config-if)#exit
TFE(config)#username LPA password secretPASS

---Configuración en LPA---

Router(config)#hostname LPA
LPA(config)#int se 0/0
LPA(config-if)#ip address 192.168.1.2 255.255.255.0
LPA(config-if)#encapsulation ppp
LPA(config-if)#ppp authentication chap
LPA(config-if)#no shutdown
LPA(config-if)#exit
LPA(config)#username TFE password secretPASS
```

Una vez aplicados los cambios podrán ser verificados mediante un *show interfaces [interfaz].* El resultado mostrará en pantalla un listado con la configuración y características de la interfaz en cuestión, entre los que se encuentra el tipo de encapsulación, protocolos utilizados, estado, etc.

Para comprobar que el enlace está operativo (*up/up*) bastará con ejecutar un *show ip interface brief.* Cualquier otra combinación significa que el enlace presenta errores, en cuyo caso se deberá identificar y aplicar la solución más adecuada para resolverlo. En PPP las incidencias más comunes son:

Estado	**Protocolo**	**Error en capa**	**Causa más común**
Administratively Down	Down	-	Interfaz apagada. Aplicar el comando "*no shutdown*".
Down	Down	Capa 1	Interfaz apagada en el router del otro extremo. Error de cableado. Problemas de hardware en alguno de los dispositivos, etc.
Up	Down	Capa 2	Diferente encapsulación en ambos extremos, error de autenticación PAP/CHAP o de configuración de mensajes *keepalive*.
Up	Up	-	Enlace operativo.

Errores en capa 2

Cuando el enlace no se establece debido a errores en capa 2 puede ser por diferentes motivos. ¿Cómo identificar cada uno de ellos para proceder a solucionarlo?

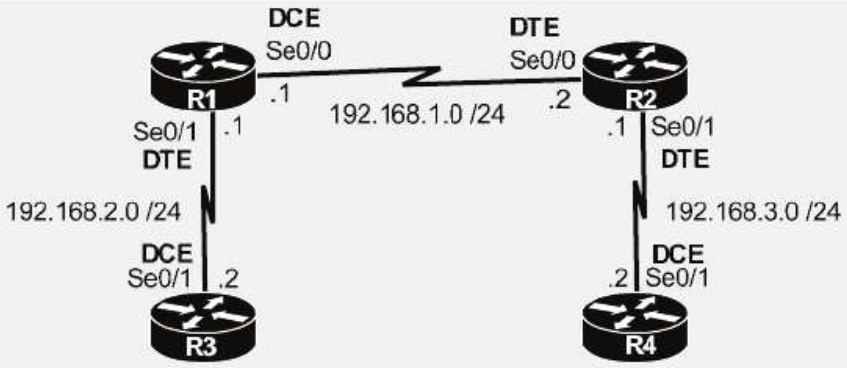
En aquellos provocados por diferente encapsulación, por ejemplo, un extremo en PPP y el otro en HDLC, el estado de línea y protocolo en ambos routers será *up/down,* sin embargo, puede variar a *up/up, up/down, up/up,* de manera intermitente, manteniendo cada uno de ellos durante apenas segundos. Para solucionarlo bastará con aplicar el mismo protocolo en ambos extremos.

En errores de autenticación PAP/CHAP, el estado de línea y protocolo será *up/down,* de manera permanente. Ello es debido a que operan con la misma encapsulación, por lo tanto, el enlace existe, pero no se torna operativo debido a credenciales incorrectas. Para solucionarlo, hay que comprobar la configuración en ambos dispositivos y aplicar la misma clave de autenticación.

Por último, los errores en mensajes *keepalive* son aquellos que se producen porque uno de los routers deja de recibirlos durante un tiempo determinado. Estos son enviados periódicamente con el fin de mantener el enlace activo, siendo el tiempo por defecto de 10 segundos. En HDLC solo son utilizados por la versión propietaria de Cisco, mientras que en PPP forman parte de LCP. Sin embargo, se puede deshabilitar su utilización mediante el comando *no keepalive,* ejecutado desde el modo de configuración de la interfaz. El problema surge cuando solo se aplica en un extremo, de tal manera que un router los envía y el otro no. En este caso, aquel configurado con *"no keepalive"* mantendrá el estado *"up/up",* mientras que en el otro extremo *"up/down".* Para solucionarlo bastará con configurar ambos de la misma manera, ya sea mediante el envío de dichos mensajes o sin los mismos.

Reto 10.2 - Configurar el enlace serial entre R1 y R2 haciendo uso de PPP y autenticación CHAP conforme a los siguientes requisitos:

- Frecuencia de reloj: 56 kbps.
- Contraseña CHAP: "CHAPpass".
- Nombres de los routers: indicados en la topología.



Solución al final del capítulo.

PPPoE: Características y configuración

Como se ha mencionado, PPP agrega numerosos beneficios sobre la comunicación, entre los que se incluyen soporte en capa 2 y 3 y mecanismos de autenticación. Ello lo ha convertido en el protocolo por excelencia implementado a lo largo del tiempo por ISPs para establecer la conexión con sus clientes y autenticar a estos en la red, permitiendo, o no, su acceso. Sin embargo, el hecho de que únicamente opere sobre enlaces punto a punto y a su vez de manera analógica impide su aplicación sobre nuevas tecnologías, como DSL, cuyo modo de transmisión resulta digital y multipunto.

Con ello se hizo imprescindible el desarrollo de un nuevo protocolo, con idénticas funcionalidades, pero siendo soportado por los nuevos métodos de conexión, dando lugar a PPPoE *(PPP over Ethernet)*, definido en el RFC 2516 y el cual provee el mecanismo necesario para llevar a cabo el transporte PPP sobre Ethernet.

Para ser más exactos, PPPoE establece un túnel entre el router cliente y el del ISP, encapsulando las tramas PPP sobre tramas Ethernet, permitiendo así su transmisión y con ello la aplicación de todas sus características. Además, dicho túnel define un circuito lógico denominado *"PPPoE session"*, a través del cual fluye la comunicación entre origen y destino.

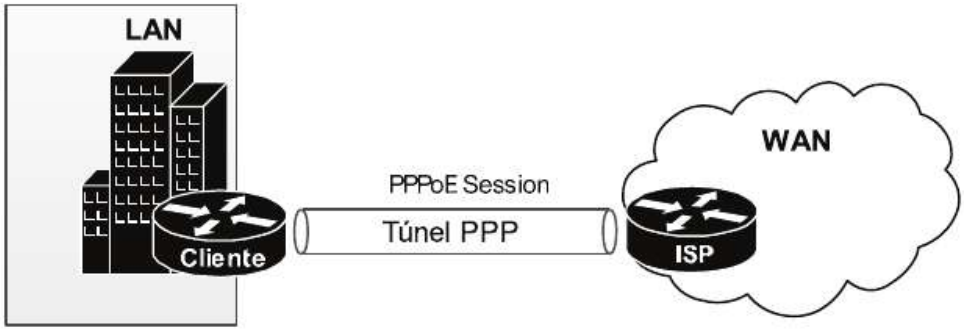


Fig. 10-9 PPPoE Session.

CONFIGURACIÓN DE PPPOE

La aplicación de PPPoE en routers Cisco consta de dos partes. Primero, habilitar una interfaz virtual, denominada *dialer*, sobre la cual será configurado PPP, definiendo parámetros como la autenticación CHAP o dirección IP. Segundo, dicha interfaz debe ser asociada con una física, utilizada para establecer el túnel a través del cual serán encapsuladas y enviadas las tramas PPP creadas por la anterior. Para ello se deberán llevar a cabo las siguientes acciones:

- *Paso 1:* Crear una interfaz virtual con el comando **interface dialer *[num]***, desde el modo de configuración global, donde *[num]* hace referencia a un valor decimal que otorga un id a la interfaz en cuestión. Acto seguido se accede a un nuevo modo de configuración desde el cual se deberán ejecutar los pasos 2, 3, 4, 5 y 6.

- *Paso 2:* Habilitar PPP como protocolo de encapsulación en capa 2, aplicando para ello el comando **encapsulation ppp.**

- *Paso 3:* Establecer CHAP como método de autenticación, definiendo el usuario y contraseña necesarios para identificar el router ante el otro extremo (ISP). Para ello bastará con ejecutar las sentencias **ppp chap hostname *[usuario]*** y **ppp chap password *[contraseña]***. El proceso de autenticación concluirá satisfactoriamente si el ISP dispone de dichas credenciales en su base de datos.

- *Paso 4:* Asignar una dirección IP a la interfaz virtual. En este caso, el comando de configuración a aplicar varía en relación con las características de conexión contratadas. Si se tratara de una IP dinámica, se hace necesaria la sentencia **ip address negotiated,** en cuyo caso PPP hará uso del protocolo IPCP, siendo el ISP quien asigne una dirección al cliente de la misma manera que lo haría un servidor DHCP. Si por el contrario se dispone de una IP estática, el comando a ejecutar es **ip address *[dir.IP] [máscara]***.

- *Paso 5:* Establecer el tamaño máximo permitido para paquetes procedentes de capa 3 (MTU), el cual, en ppp, es igual 1492 bytes. Para ello, aplicar la sentencia **mtu 1492.**

- *Paso 6:* Definir un *pool* con el comando **dialer pool *[num]***. Este tiene como finalidad asociar la interfaz virtual recién creada con la interfaz física que establecerá el túnel (ver paso 11). El valor *[num]* le otorga un id, que no tiene por qué coincidir con el definido en el paso 1.

- *Paso 7:* Volver al modo de configuración global.

- *Paso 8:* Acceder al modo de configuración de la interfaz física que conecta directamente con el ISP, a través del comando **interface *[interfaz física]***.

- *Paso 9:* No asignar ninguna dirección IP, y en el caso de que la tuviera, eliminarla con el comando **no ip address.** Ello es debido a que esta es definida en la interfaz virtual, ya sea de manera estática o dinámica.

- *Paso 10:* Habilitar PPPoE sobre la interfaz, ejecutando la sentencia **pppoe enable.**

- *Paso 11:* Asociar la interfaz física con la virtual, a través del comando **pppoe-client dial-pool-number *[num]***, donde *[num]* hace referencia al *pool* creado en el paso 6.

- *Paso 12:* Habilitar la interfaz mediante un **no suhtdown.**

- *Paso 13 (opcional):* Si la conexión recién configurada tuviera como propósito brindar salida hacia Internet a los dispositivos de la LAN, se hace necesario configurar una ruta estática por defecto, a través del comando **ip route 0.0.0.0 0.0.0.0 dialer *[num interfaz virtual]***, siendo esta la creada en el paso 1.

Ejemplo. Habilitar PPPoE sobre la interfaz Gi0/1 de TFE, donde la dirección IP pública es asignada de manera dinámica por el ISP, siendo necesarias las siguientes credenciales de autenticación para que la conexión a la red WAN se lleve a cabo de manera correcta:

- Usuario: TFE
- Password: Acc3s0

```
TFE(config)# interface dialer 7
TFE(config-if)# encapsulation ppp
TFE(config-if)# ppp chap hostname TFE
TFE(config-if)# ppp chap password Acc3s0
TFE(config-if)# ip address negotiated
TFE(config-if)# mtu 1492
TFE(config-if)# dialer pool 5
TFE(config-if)# exit
```

```
TFE(config)# interface Gi0/1
TFE(config-if)# no ip address
TFE(config-if)# pppoe enable
TFE(config-if)# pppoe-client dial-pool-number 5
TFE(config-if)# no shutdown
TFE(config-if)#exit

TFE(config)# ip route 0.0.0.0 0.0.0.0 dialer 7
```

Una vez concluido, el estado de la interfaz y sus detalles pueden ser verificados con los comandos *show ip interface brief* y *show interface dialer [num].*

FRAME RELAY: CONFIGURACIÓN Y VERIFICACIÓN

Frame Relay puede ser definido como un protocolo de capa 2 desarrollado con el fin de comunicar y conectar diferentes redes privadas de manera segura y fiable a través de la WAN. Basa su modo de operar en la creación de una red multiacceso donde cada dispositivo conectará con los restantes mediante circuitos virtuales, logrando gracias a ello múltiples comunicaciones de manera simultánea. Este hecho implica que la trama en capa 2 incluya un campo "dirección", en el que tan solo podrán ser definidos valores unicast o multicast, nunca broadcast, ya que en Frame Relay no se permite su envío, lo que identifica a este tipo de redes como *nonbroadcast multiaccess (NBMA).*

Al tratarse de un protocolo antiguo y gracias a la aparición de nuevas tecnologías que incluyen características más avanzadas y que a su vez resultan más sencillas de implementar, como las VPNs o MPLS, Frame Relay ha caído en desuso. Sin embargo, aún es posible optar por este modelo de comunicación, el cual está compuesto por los siguientes elementos:

Data Terminal Equipment (DTE): Es el router ubicado en la red LAN y que conecta con el dispositivo Frame Relay del ISP, ubicado en la WAN.

Data Communications Equipment (DCE): Es el Switch Frame Relay, encargado de gestionar los circuitos virtuales a través de la WAN. Su administración depende por completo del ISP.

Access Link: Es el circuito físico que conecta el DTE con el DCE.

Access Rate (AR): La velocidad definida para el *Access Link*.

Local Management Interface (LMI): Es el protocolo utilizado para la comunicación entre el DTE y el DCE. Será analizado en párrafos posteriores.

Virtual Circuit (VC): Es un circuito lógico que representa la ruta establecida entre los DTE de ambos extremos de la comunicación.

Permanent Virtual Circuit (PVC): Es una ruta creada entre los DTE de ambos extremos. La diferencia entre los VC y los PVC es que estos últimos establecen y utilizan siempre el mismo circuito físico, de manera similar a una línea arrendada.

Data link Connection Identifier (DLCI): Son las direcciones utilizadas por Frame Relay en capa 2 para identificar a cada uno de los circuitos.

Switched Virtual Circuit (SVC): Un SVC es un circuito virtual (VC) creado de manera dinámica cada vez que resulte necesario.

Commited Information Rate (CIR): Es el ancho de banda definido para los circuitos virtuales.

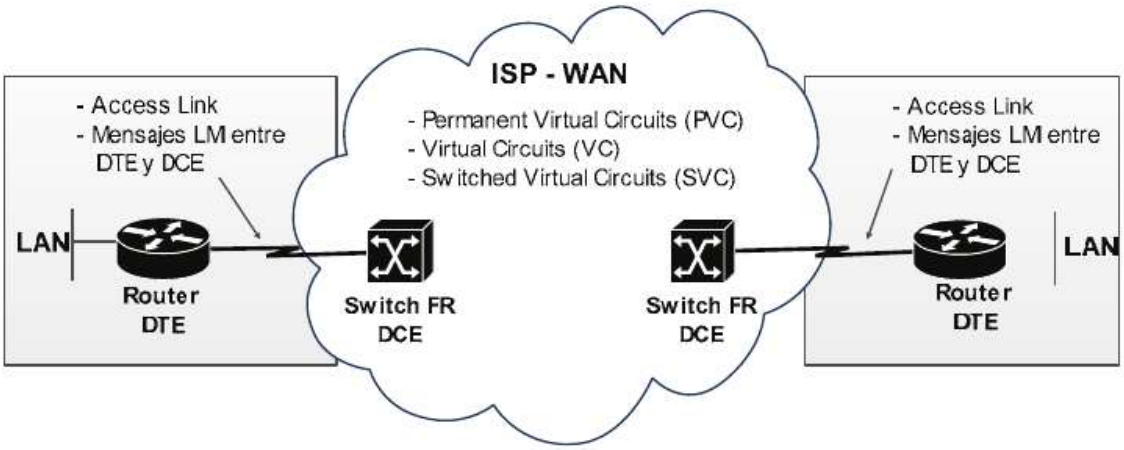


Fig. 10-10 Elementos presentes en una topología Frame Relay.

Una de las mayores ventajas de Frame Relay respecto a las conexiones punto a punto reside en los circuitos virtuales (*VC*), logrando que a través de un solo link de acceso el router pueda comunicar con multitud de ellos, lo que convierte a FR en una opción de interconexión altamente escalable y económicamente más accesible que otros tipos de tecnología.

La comunicación a través de los VC recorrerá la WAN del ISP, compuesta por multitud de Switchs FR. Dicho entorno es accesible por todos o gran cantidad de los clientes que contratan este tipo de servicios, lo que lo convierte en un medio

compartido. Sin embargo, se asegura la privacidad y seguridad en la comunicación gracias a los protocolos que intervienen en la misma. Cuando un cliente contrata FR, el ISP normalmente establece un enlace virtual predefinido entre ambos extremos, denominado PVC *(Permanent Virtual Circuit)*.

Por último, para el diseño de una red FR se puede optar por tres métodos, una malla completa *(full-mesh)*, una malla parcial *(partial-mesh)* o un modelo híbrido, basado en una mezcla de los dos anteriores. La malla completa establece circuitos virtuales entre cada par de DTEs que compongan la red, mientras que en la parcial pueden omitirse enlaces, de tal manera que algunos DTEs, para llegar a un determinado destino, deberán hacerlo a través de diferentes circuitos virtuales intermediarios. Por ejemplo…

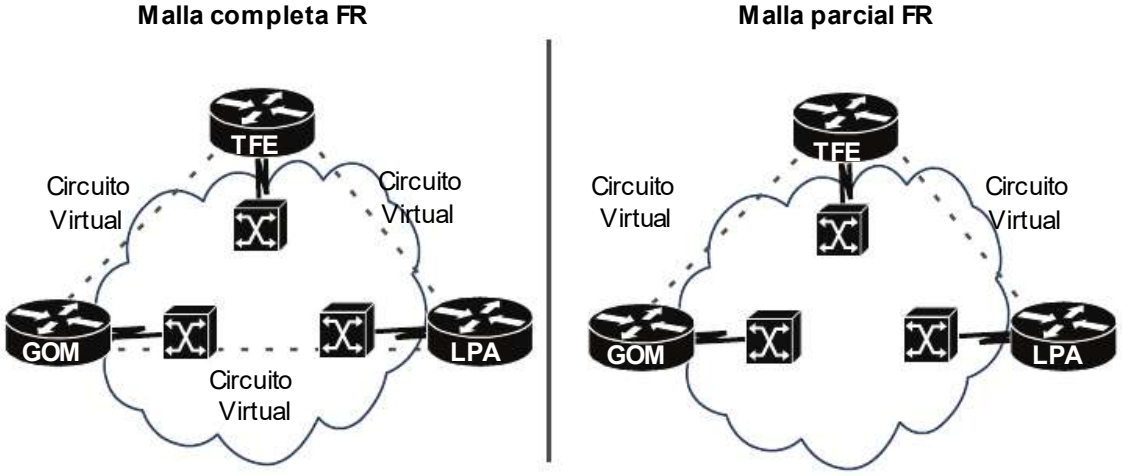


Fig. 10-11 Tipos de diseño Frame Relay.

PROTOCOLO LMI

LMI *(Local Management Interface)* es el protocolo utilizado por Frame Relay para la gestión del enlace entre el DTE y el DCE. Sus funciones principales son:

- El envío de mensajes *keepalive* entre ambos dispositivos con el fin de mantener la adyacencia.

- Identificar los circuitos virtuales caídos, con el fin de que no sean utilizados hasta que vuelvan a operar con normalidad.

Los routers Cisco disponen de 3 opciones para tal propósito: Cisco, ITU (q933a) y ANSI, aunque normalmente se hace uso de la opción por defecto, denominada "*LMI*

Autosense", cuya función consiste en aplicar de manera automática el mismo protocolo que el utilizado por el switch Frame Relay ubicado en la WAN. Aun así, el comando necesario para definirlo manualmente es **frame-relay lmi-type [*cisco / ansi / q933a*],** desde el modo de configuración de la interfaz serial.

FORMATO DE TRAMA

Al tratarse de un protocolo en capa 2, FR debe encapsular en tramas los paquetes procedentes de capa 3. El formato definido para ello resulta bastante sencillo, donde los campos "dirección" y "FCS" hacen posible la comunicación. En el primero se identifica el DLCI del circuito virtual necesario para el direccionamiento, mientras que el segundo se encarga de la detección de errores.

En detalle, la trama consta de los siguientes campos:

Delimitador 8 bits	Dirección 16 bits	Datos Long. Variable	FCS 16 bits	Delimitador 8 bits

- *Delimitador:* Al igual que en HDLC, establece el inicio y fin de una trama, su valor es 01111110.

- *Dirección:* A su vez está compuesto por dos subcampos del tamaño de 8 bits cada uno, entre los cuales se encuentra la dirección DLCI.

- *Datos:* Son los datos procedentes de capa 3, es decir, el paquete encapsulado.

- *FCS:* También denominado CRC, es el encargado del control de errores. Al igual que en HDLC, contiene el resultado de un algoritmo matemático aplicado sobre los bits de la trama (excluyendo los delimitadores). En el destino, el router que la recibe aplica el mismo algoritmo a los bits recibidos. Si el resultado coincide significa que la trama ha sido recibida sin errores.

DIRECCIONAMIENTO

Frame Relay ejecuta el direccionamiento en relación con el DLCI del circuito virtual, el cual es configurado en formato decimal y representado en el campo "dirección" en binario, con un máximo de 10 bits. Este hecho supone una gran diferencia respecto a la mayoría de protocolos, los cuales hacen uso de direcciones MAC o IP para tal propósito.

Los valores de DLCI son gestionados por el ISP, asignando dos direcciones sobre cada circuito virtual, una en cada extremo. Este concepto puede asemejarse con las direcciones IP en un enlace punto a punto, donde cada router debe hacer uso de una diferente, de lo contrario la comunicación no sería posible. Además, un router puede estar configurado con tantos DLCI como circuitos virtuales disponga, sin embargo, es importante que estos no coincidan en su valor, de lo contrario se producirán conflictos de comunicación.

Un ejemplo podría ser…

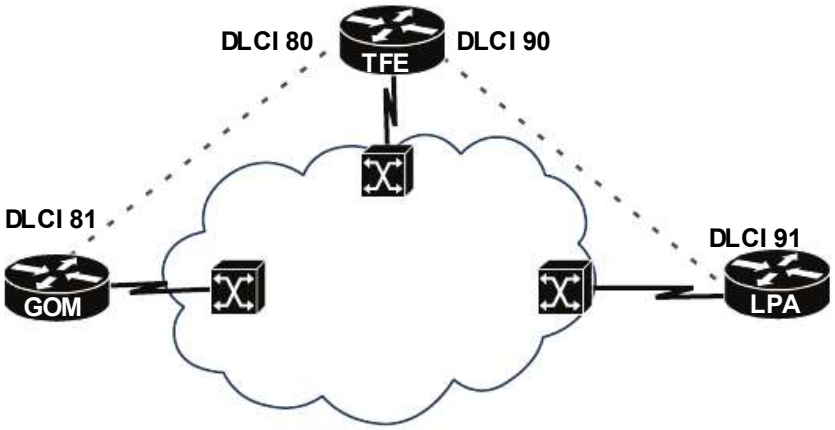


Fig. 10-12 Identificación de circuitos virtuales FR.

El modo de direccionamiento en FR resulta totalmente diferente al llevado a cabo por otras tecnologías. En este caso, los routers utilizarán su propio DLCI como dirección en capa 2, gracias al cual el switch FR podrá gestionar la comunicación. El proceso consta de los siguientes pasos:

- *Paso 1:* El router crea una trama Frame Relay, utilizando como dirección su propio DLCI del circuito virtual por el cual desea enviar los datos.

- *Paso 2:* La trama atraviesa el link de acceso y es recibida por el switch FR. La configuración de este debe incluir los DLCI de cada extremo para cada circuito virtual, creando una asociación entre ambos. Cuando el switch recibe la trama, analiza la dirección incluida y la sustituye por el DLCI del otro extremo del circuito virtual, para acto seguido reenviarla a través de la red Frame Relay.

- *Paso 3:* El router de destino recibe la comunicación a través de su propio número de DLCI.

Para comprenderlo mejor, centrémonos en la comunicación entre el router TFE y LPA…

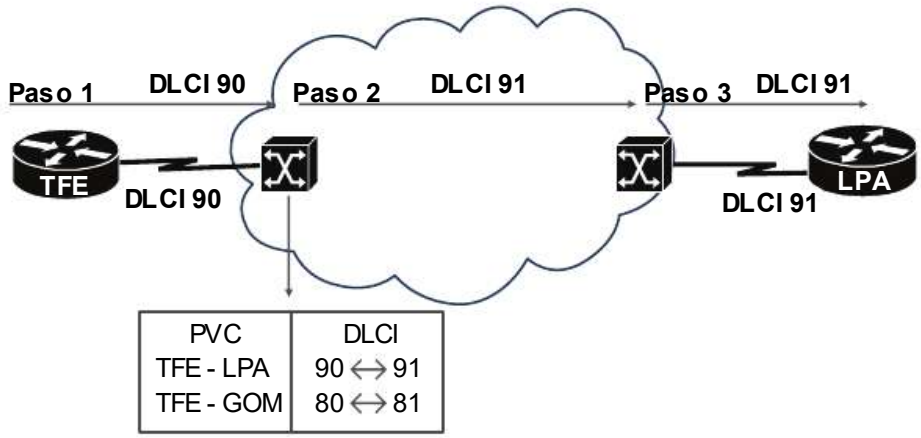


Fig. 10-13 Direccionamiento en Frame Relay.

- *Paso 1:* Para comunicarse con LPA, el router TFE crea una trama utilizando como dirección el valor 90, que corresponde a su propio DLCI para ese circuito virtual.

- *Paso 2:* La trama es recibida por el switch FR. Este analiza la dirección, comprueba que su valor es 90 y gracias a su configuración conoce tanto el circuito virtual que se debe utilizar como el DLCI del router de destino. El switch cambia el valor 90 por 91, ya que es el definido por LPA para ese mismo circuito.

- *Paso 3:* LPA recibe la trama.

La comunicación a la inversa se llevaría a cabo de la misma manera, donde LPA hará uso de su DLCI local (91) y el switch lo modificará por el valor 90.

Diseño en capa 3 de una red Frame Relay

Aunque la comunicación en Frame Relay se lleve a cabo en capa 2 y a través de circuitos virtuales, el modo de operar una vez establecidos resulta bastante similar a los enlaces punto a punto, y al igual que en estos, se hace necesario configurar una dirección IP en los routers de ambos extremos. Sin embargo, en FR, una sola interfaz física puede albergar multitud de circuitos. Entonces, ¿qué IP configurar o cómo proceder cuando conecta con diferentes VC?

El primer factor a tener en cuenta para ello es el diseño de la red en capa 3, pudiendo optar por los siguientes modelos…

- Una misma subred para todos los routers pertenecientes a la red FR.
- Una subred para cada circuito virtual.
- Varias subredes, algunas de ellas compartidas y otras dedicadas para determinados circuitos virtuales. (Una mezcla de los dos modelos anteriores).

MODELO DE UNA SUBRED PARA TODOS LOS DTE

Es el más sencillo de implementar y consiste, como su propio nombre indica, en hacer uso de una sola subred para todos los routers que formen parte del mismo diseño FR. En este caso, si un DTE dispone de varios circuitos virtuales, aplicará la misma IP en capa 3 para comunicarse a través de todos ellos. Este suele ser el diseño más habitual en redes Frame Relay de malla completa.

Un ejemplo podría ser el siguiente, donde todos los routers conectan a FR a través de su interfaz serial, y estas a su vez pertenecen a la subred 10.1.1.0/24.

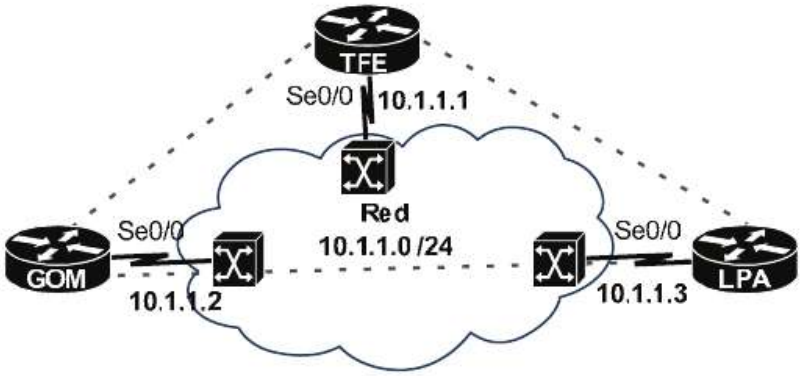


Fig. 10-14 Una única subred para todos los DTE.

MODELO DE UNA SUBRED PARA CADA CIRCUITO VIRTUAL

Suele ser aplicado sobre diseños parcialmente mallados y consiste en hacer uso de una subred para cada uno de los circuitos virtuales disponibles. Para lograrlo resulta necesaria la configuración de subinterfaces, donde cada una de ellas comunicará un determinado VC a través de la subred correspondiente.

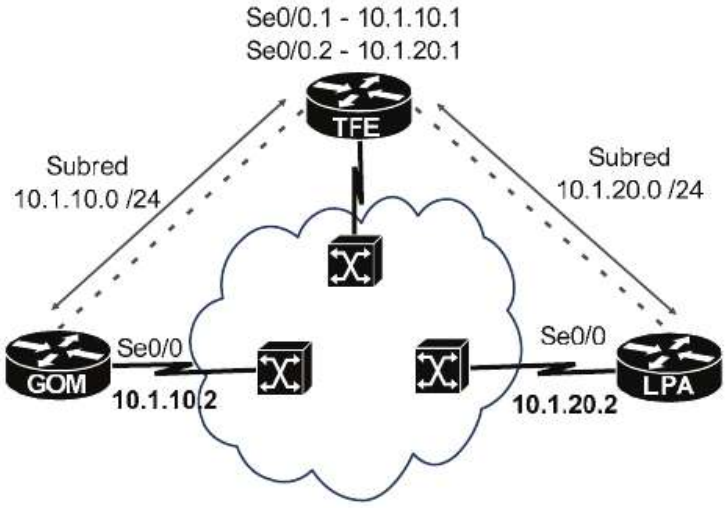


Fig. 10-15 Una subred para cada circuito virtual.

MODELO HÍBRIDO

El diseño híbrido consiste en una mezcla de los dos modelos anteriores, donde los circuitos virtuales que formen una malla completa compartirán la misma subred, mientras que los parcialmente mallados dispondrán de rangos dedicados. La configuración en este caso también incluye la utilización de subinterfaces.

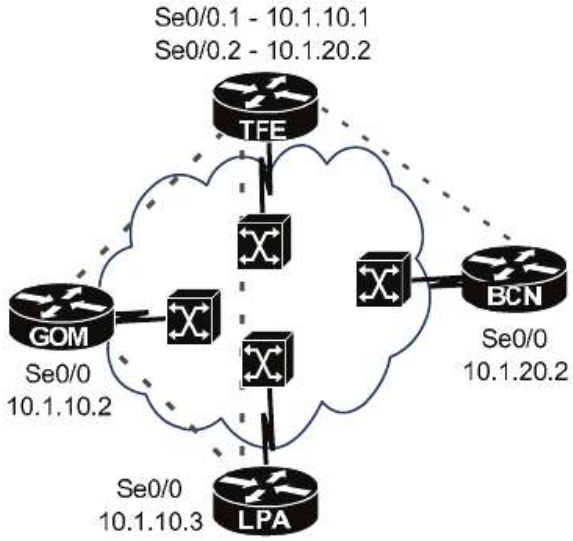


Fig. 10-16 Diseño híbrido FR.

En el ejemplo, TFE, GOM y LPA definen una malla completa y por lo tanto son configurados en la misma subred, sin embargo, BCN no forma parte de la misma, por lo que el circuito virtual que lo une con TFE es configurado haciendo uso de un rango dedicado.

Configuración y verificación de Frame Relay

En entornos reales la administración de Frame Relay depende por completo del ISP, tanto la asignación de los DLCI como la gestión de los switchs FR. Aun así, resulta necesario conocer su configuración en routers Cisco para afrontar el examen de CCNA, la cual varía dependiendo del diseño implementado.

CONFIGURACIÓN DE FR EN REDES TOTALMENTE MALLADAS

Las redes totalmente malladas hacen uso de la misma subred para la comunicación entre todos sus miembros, lo que significa que cada uno de ellos dispondrá de una única dirección IP que será utilizada en capa 3 sobre la totalidad de circuitos virtuales disponibles. En estos casos, la configuración se lleva a cabo directamente a través de la interfaz física, debiendo aplicar el siguiente procedimiento:

- *Paso 1:* Configurar una dirección IP correspondiente al rango utilizado en la red FR.

- *Paso 2:* Definir el tipo de encapsulado en capa 2, con el comando **encapsulation frame-relay [*cisco* / *ietf*]**. También es posible ejecutarlo sin parámetros ("*encapsulation frame-relay*"), en cuyo caso se aplicará el protocolo "Cisco" por defecto. Sin embargo, la opción recomendada es *ietf*.

- *Paso 3 (opcional):* Definir el tipo de LMI necesario para la gestión del enlace con el DCE, haciendo uso del comando **frame-relay lmi-type [*cisco* / *ansi* / *q933a*]**. Si no fuera configurado, por defecto se ejecuta la función *autosense*, la cual aplicará el mismo protocolo que el utilizado por el switch FR.

- *Paso 4:* Configurar la dirección DLCI para cada circuito virtual, con el comando **frame-relay interface-dlci *[num DLCI]***. El valor asignado debe formar parte del rango 16-1007.

- *Paso 5:* Habilitar la interfaz con el comando **no shutdown.**

Ejemplo: En la siguiente red totalmente mallada, configurar Frame Relay en el router TFE haciendo uso del protocolo IETF en capa 2. El tipo de LMI debe ser

seleccionado automáticamente mediante la función *autosense*. Los routers GOM y LPA ya han sido configurados y operan correctamente.

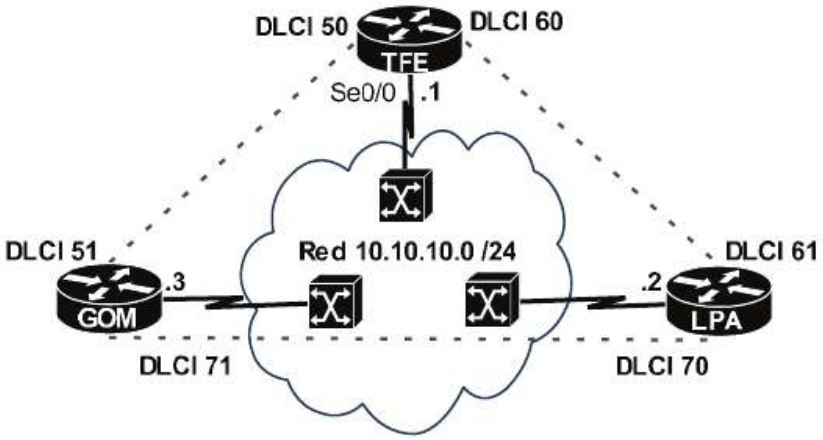


Fig. 10-17 Diseño FR para supuesto práctico.

```
TFE(config)#interface Se0/0
TFE(config-if)#ip address 10.10.10.1 255.255.255.0
TFE(config-if)#encapsulation frame-relay ietf
TFE(config-if)#frame-relay interface-dlci 50
TFE(config-if)#frame-relay interface-dlci 60
TFE(config-if)#no shutdown
```

Una de las funciones del comando *"frame-relay interface-dlci [num DLCI]"* consiste en ejecutar un ARP inverso, con la finalidad de identificar la dirección de capa 3 del router vecino y asociarla con el número de DLCI del circuito virtual, estableciendo con ello un mapeo de capa 3 con capa 2. Cuando un router no soporta esta función, o simplemente deseamos definir dicho mapeo de manera manual, se podrá llevar a cabo ejecutando el comando **frame-relay map *[protocolo] [IP de destino] [DLCI local] [broadcast]***, desde el modo de configuración de la interfaz física y donde:

- *Protocolo* indica el protocolo sobre el cual opera la red, pudiendo ser ip, ipx, appletalk, etc.

- *IP de destino* indica la dirección IP del router del otro extremo, la cual será mapeada con el DLCI local.

- *DLCI local* define la dirección DLCI del circuito virtual, la cual será mapeada con la dirección IP del router vecino.

- *Broadcast* habilita el envío de mensajes broadcast y multicast a través del enlace, utilizados, por ejemplo, por los protocolos de enrutamiento. Aunque su configuración

resulte opcional se recomienda su aplicación, ya que de lo contrario Frame Relay bloquearía este tipo de mensajes.

Con ello, si la configuración en el router TFE debiera llevarse a cabo mediante mapeos estáticos, quedaría definida de la siguiente manera:

```
TFE(config)#interface Se0/0
TFE(config-if)#ip address 10.10.10.1 255.255.255.0
TFE(config-if)#encapsulation frame-relay ietf
TFE(config-if)#frame-relay map ip 10.10.10.2 60 broadcast
TFE(config-if)#frame-relay map ip 10.10.10.3 50 broadcast
TFE(config-if)#no shutdown
```

Además, en estos casos, también se podrá deshabilitar la función de ARP inverso con el comando **no frame-relay inverse-arp**, desde el modo de configuración de la interfaz.

CONFIGURACIÓN DE FR EN REDES PARCIALMENTE MALLADAS

En el diseño parcialmente mallado se hace uso de diferentes subredes, una para cada circuito virtual, por lo tanto, resulta necesaria la configuración de diferentes direcciones IP, y con ello, subinterfaces, en las cuales se definirán los datos necesarios para cada circuito. Mientras, la interfaz física que alberga a todas ellas se encargará del encapsulado y tipo de LMI.

Su configuración consta de los siguientes pasos:

- *Paso 1:* En la interfaz física, definir el tipo de encapsulado en capa 2, con el comando **encapsulation frame-relay [*cisco* | *ietf*]**.

- *Paso 2 (opcional):* En la interfaz física, aplicar el tipo de LMI necesario para la gestión del enlace con el DCE, con el comando **frame-relay lmi-type [*cisco* | *ansi* | *q933a*]**.

- *Paso 3:* Crear la subinterfaz necesaria con el comando **interface [*interfaz serial].[subinterfaz*] [*point-to-point* | *multipoint*]** desde el modo de configuración global. Por ejemplo, "*Se0/0.10*" identifica la subinterfaz 10 en la interfaz física Se0/0. Mientras, [*point-to-point* | *multipoint*] hace referencia al tipo de enlace del circuito virtual. Punto a punto es aquel cuyo VC tan solo tiene un destino, mientras que multipunto puede ser utilizado para la conexión con múltiples de ellos.

- *Paso 4:* Desde el modo de configuración de la subinterfaz, asignar una dirección IP, que deberá pertenecer al rango de red utilizado por el circuito virtual que comunicará.

- *Paso 5:* Configurar el DLCI necesario para el circuito virtual, con el comando **frame-relay interface-dlci *[num DLCI],*** desde el modo de configuración de la subinterfaz.

- *Paso 6:* En la interfaz física, aplicar el comando **no shutdown.**

Los pasos 3, 4 y 5 deberán ser ejecutados sobre cada una de las subinterfaces definidas.

Ejemplo: En la siguiente red parcialmente mallada, configurar Frame Relay en el router TFE haciendo uso del protocolo IETF en capa 2. El tipo de LMI debe ser seleccionado automáticamente mediante la función *autosense*. Los routers GOM y LPA ya han sido configurados y operan correctamente.

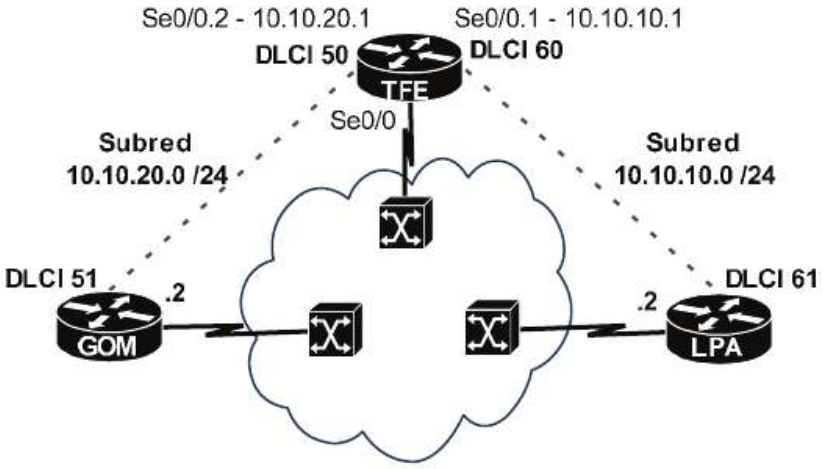


Fig. 10-18 Diseño FR para supuesto práctico.

```
TFE(config)#int se0/0
TFE(config-if)#encapsulation frame-relay ietf
TFE(config-if)#exit

TFE(config)#interface Se0/0.1 point-to-point
TFE(config-subif)#ip address 10.10.10.1 255.255.255.0
TFE(config-subif)#frame-relay interface-dlci 60
TFE(config-subif)#exit

TFE(config)#interface Se0/0.2 point-to-point
TFE(config-subif)#ip address 10.10.20.1 255.255.255.0
TFE(config-subif)#frame-relay interface-dlci 50
TFE(config-subif)#exit

TFE(config)#int se0/0
TFE(config-if)#no shutdown
```

Una vez finalizado el proceso de configuración, la mejor manera de comprobar que existe conectividad entre los diferentes extremos es mediante un ping. Obtener respuesta indica que el enlace se ha establecido correctamente, de lo contrario habría que revisar la configuración en ambos routers para determinar la causa del fallo.

Ejecutado desde TFE hacia los routers LPA y GOM…

```
TFE#ping 10.10.10.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/8/12 ms

TFE#ping 10.10.20.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 6/12/15 ms
```

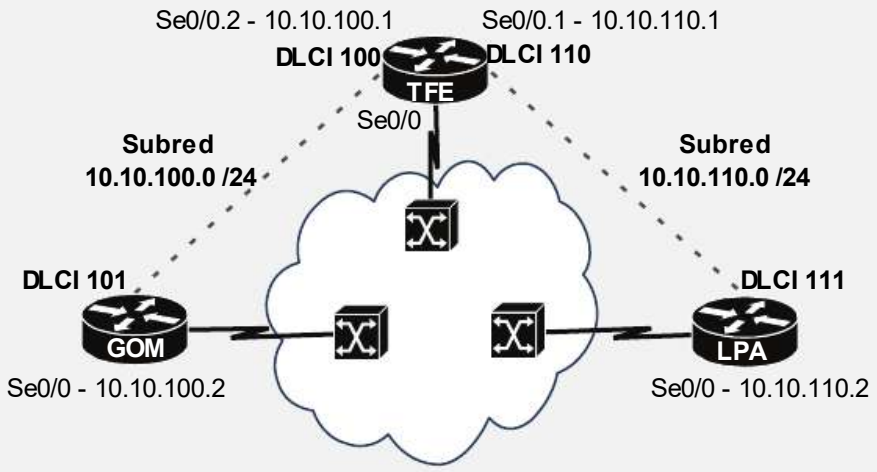
Otras opciones de verificación disponibles en IOS son:

- *show frame-relay map:* Muestra en pantalla un listado de los routers con los cuales se ha establecido un mapeo, ya sea mediante la función de ARP inverso o de manera estática (configuración manual), incluyendo información como la IP del router vecino, protocolos aplicados, DLCI, estado del enlace, etc.

- *show frame-relay pvc:* Muestra un listado con estadísticas sobre cada uno de los circuitos virtuales, facilitando información como el número de DLCI local, estado del enlace, interfaz física, etc.

Reto 10.3 – Configurar Frame Relay en todos los routers de la siguiente topología de tal manera que:

- El protocolo de encapsulado en capa 2 sea ietf.
- Se aplique la función autosense para seleccionar el tipo de LMI.
- En el router TFE, configurar los circuitos virtuales mediante subinterfaces, mientras que en GOM y LPA, realizar la configuración directamente desde la interfaz física, llevando a cabo un mapeo de IP – DLCI.



Solución al final del capítulo.

SERVICIOS WAN - CLOUD COMPUTING

A lo largo de la historia, y especialmente en las últimas décadas, el desarrollo llevado a cabo sobre la totalidad de componentes que conforman la WAN unido al constante crecimiento tanto de usuarios como de infraestructuras ha propiciado la aparición de innumerables oportunidades de negocio online, y con ello, diferentes nomenclaturas que identifican cada una de las opciones disponibles. Una de estas, *"Cloud Computing",* hace referencia al conjunto de servicios ubicados en la nube, abarcando un rango tan amplio que alberga soluciones de todo tipo y complejidad. Como ejemplo se podría hacer mención a la multitud de opciones de almacenamiento, correo, voz, vídeo, hosting, aplicaciones o incluso servidores dedicados.

Son muchos los factores que han posibilitado tal avance, especialmente dos, primero, la creciente velocidad de acceso a Internet mediante las tecnologías ya analizadas en este mismo capítulo, y segundo, la virtualización, gracias a la cual un único dispositivo físico puede albergar multitud de servidores, y con ello, aplicaciones y servicios. Pero ¿en qué consiste dicha virtualización?

La arquitectura de todo sistema informático comprende un determinado hardware, cuyas capacidades serán gestionadas mediante software (sistema operativo), que a su vez, permitirá la instalación y ejecución de aplicaciones y servicios sobre el mismo. Conforme a ello, el pensamiento más común identifica un dispositivo físico por cada S.O. Un claro ejemplo de ello podría ser cualquier PC doméstico.

Dicha teoría también es aplicable sobre servidores, implementando sobre estos soluciones más robustas y de mayores prestaciones. En este aspecto, aquellos destinados a uso corporativo normalmente presentan un diseño físico adaptado para su instalación en armarios de comunicaciones *(racks)*, ubicados en el centro de proceso de datos de cada compañía. Es decir, los CPD están compuestos por racks, los cuales albergan un número limitado de dispositivos físicos, entre ellos, servidores.

Fig. 10-19 Armario de comunicaciones (Rack).

Fig. 10-20 Servidor Cisco UCS C220 M4.

El tamaño de un rack, y por consiguiente su capacidad total, varía dependiendo del modelo, fabricante, propósito, etc. Existen multitud de opciones disponibles, debiendo optar por aquella que más se adecue a las necesidades de la red o entorno físico donde será instalado. Lo mismo sucede con los servidores, donde el tamaño difiere con relación al fabricante, prestaciones o hardware, entre otras.

Con todo ello se podría deducir que cada servidor dispondrá de un sistema operativo y será instalado físicamente en un armario de comunicaciones, por lo que si dicho rack, por ejemplo, albergara 5 servidores, se obtendrían el mismo número de S.O., sobre los cuales ofrecer los servicios deseados a la red.

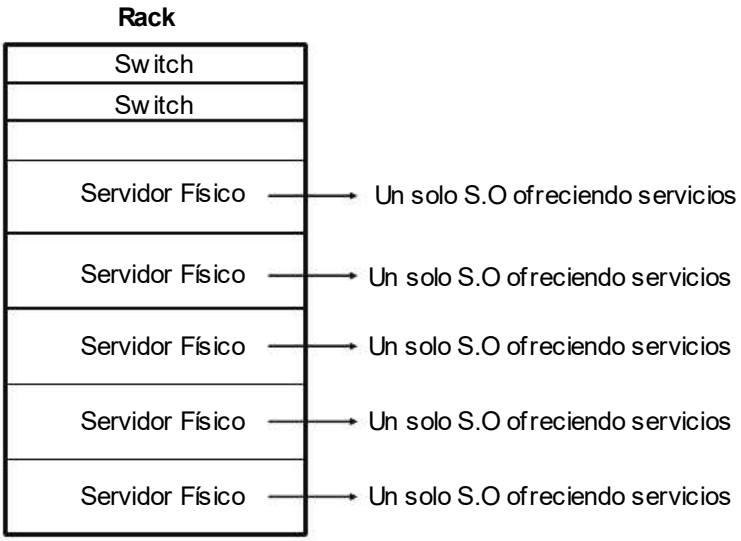


Fig. 10-21 Servidores no virtualizados.

Este modelo, perfectamente operativo, era el utilizado antiguamente, cuando las necesidades de la red no resultaban tan exigentes y el desarrollo tecnológico no permitía opciones más avanzadas. Sin embargo, a día de hoy, la innumerable cantidad de servicios existentes y la multitud de servidores necesarios para implementarlos hace totalmente inviable y nada escalable su aplicación. Imagina un centro de datos ubicado en la nube que requiera cientos o incluso miles de servidores. Las limitaciones físicas resultan evidentes.

La solución a ello nace en la virtualización, la cual, en su definición más básica, consiste en dividir y gestionar la totalidad del hardware presente en un dispositivo físico con el fin de permitir la creación de diferentes servidores virtuales, cada uno de ellos con capacidades de hardware dedicado y por su puesto con total autonomía en cuanto a sistema operativo, aplicaciones y servicios se refiere. Dicha gestión es posible gracias a la función desarrollada por el *hypervisor*, el cual administra cada uno de los recursos asignados sobre cada máquina virtual.

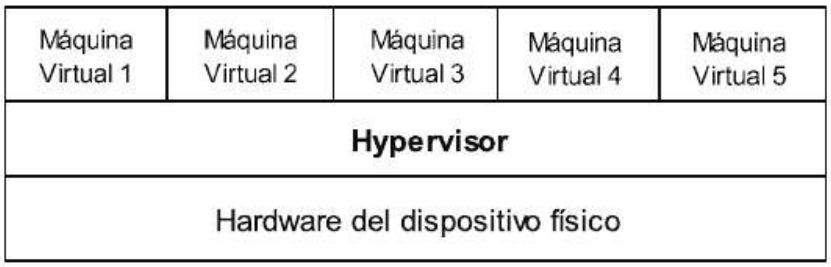


Fig. 10-22 Administración de hardware mediante el Hypervisor.

Las máquinas virtuales también pueden ser denominadas como VM (*Virtual Machine*). A lo largo de la presente sección se hará uso de ambas nomenclaturas.

De tal manera que, si sobre el modelo propuesto en la *Fig. 10-21* se aplicara virtualización, se podría disponer de un número más elevado de servidores en el mismo espacio físico.

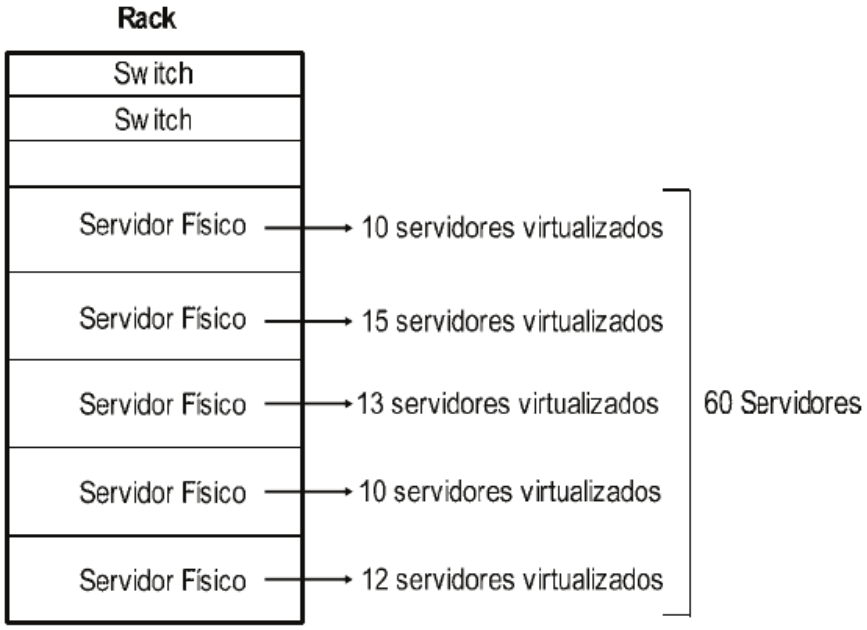


Fig. 10-23 Servidores virtualizados.

En cuanto a su implementación sobre entornos corporativos, las soluciones más comunes actualmente son *VMWare vCenter, Microsoft HyperV, Citrix XenServer* y *Red Hat KVM*. Todas ellas presentan sistemas robustos con un modelo de gestión intuitivo y eficaz que habilita múltiples opciones como la creación de máquinas virtuales de manera sencilla y rápida, mover cualquier de ellas hacia otro servidor físico, modificar sus características de hardware en cualquier momento o copias de seguridad automatizadas, entre muchas otras.

Evidentemente, este nuevo modelo aporta numerosas ventajas, entre las que cabe destacar:

- Ahorro de diferentes recursos, como espacio físico, cableado o sistema eléctrico.

- Mayor facilidad para ofrecer redundancia y con ello alta disponibilidad.

- Seguridad y aislamiento de cada máquina virtual. Un fallo en una de ellas no supone la caída del resto.

- Administración centralizada e intuitiva gracias al *hypervisor*.

- Mayor agilidad en cuanto a la puesta en marcha de nuevos servidores. Simplemente bastará con crear la máquina virtual e iniciarla. Un proceso que puede durar escasos minutos.

- Ahorro de costes. Aunque bien es cierto que el servidor físico requiere capacidad y rendimiento de hardware muy elevado, además de alta redundancia en la práctica totalidad de sus componentes, el coste total siempre será menor que la adquisición de servidores dedicados.

- Tan solo se requieren varias interfaces de switch para dar servicio a la totalidad de máquinas virtuales ubicadas en un servidor físico, sin importar el número total de estas.

Conforme a esta última afirmación, ¿Cómo se lleva a cabo la comunicación a nivel de red de todas las máquinas virtuales? Hay que tener en cuenta que un servidor físico dispone de un número limitado de conexiones de red, y evidentemente, la totalidad de VMs que alberga harán uso de las mismas para enviar y recibir cualquier tipo de tráfico. Sin embargo, una de las características de la virtualización consiste en la asignación de hardware dedicado a cada máquina virtual, logrando así el aislamiento entre todas ellas. Imagina que el servidor físico dispone de dos conexiones de red, y alberga 5 VMs, ¿cómo lograr la comunicación?

La solución consiste, primero, en la asignación de interfaces de red virtuales *(vNIC)* a cada una de las VMs, y segundo, en la creación, por parte del *hypervisor*, de un switch, también virtual *(vSwitch)*, que gestionará y comunicará las interfaces virtuales con aquellas físicas. De tal manera que:

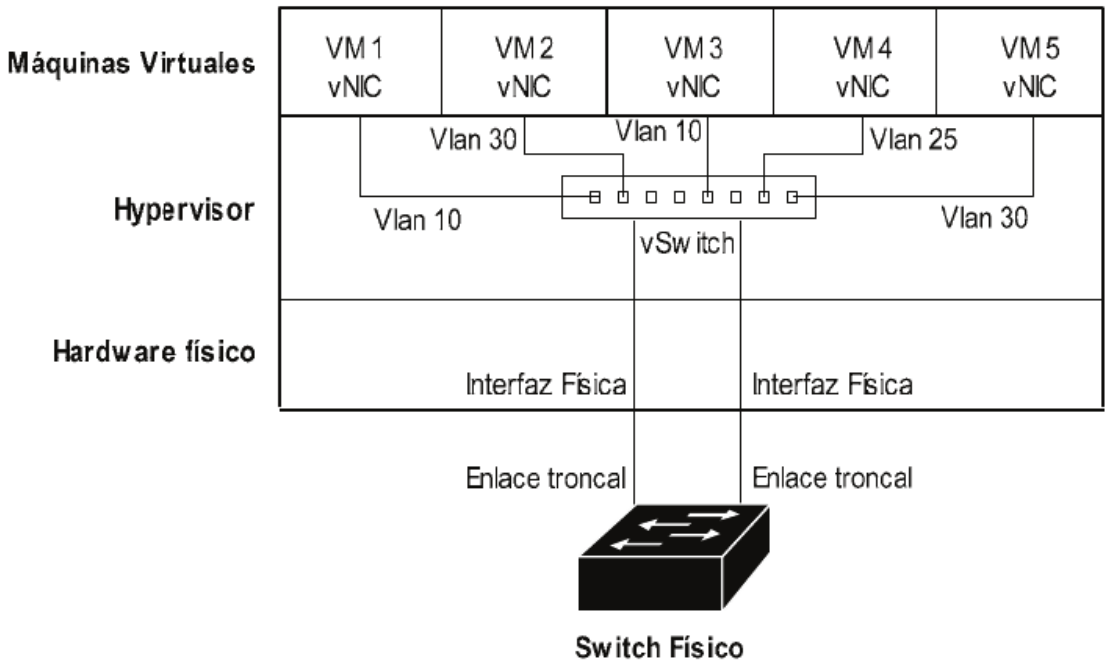


Fig. 10-24 Comunicación entre vSwitch y switch físico.

Obteniendo, gracias a ello, las siguientes características y beneficios:

- Posibilidad de crear el número de máquinas virtuales que se estimen oportunas sin tener en cuenta la cantidad de interfaces de red físicas disponibles.

- Posibilidad de ubicar las VMs en diferentes VLANs.

- Asignar cuántas vNIC fueran necesarias sobre cada máquina virtual.

Por último, las interfaces del servidor físico deben disponer de un rendimiento relativamente alto con el fin de ofrecer prestaciones aceptables a la totalidad de máquinas virtuales. En este aspecto, lo más habitual consiste en interfaces de 10 Gb, además, han de ser configuradas como enlaces troncales con el fin de transportar el tráfico de todas las VLANs.

En cuanto a redundancia y disponibilidad a nivel de red, la actuación más común corresponde a la instalación de dos switchs físicos, donde cada uno de ellos conecta con cada máquina virtual y a su vez con el resto de la topología, de tal manera que:

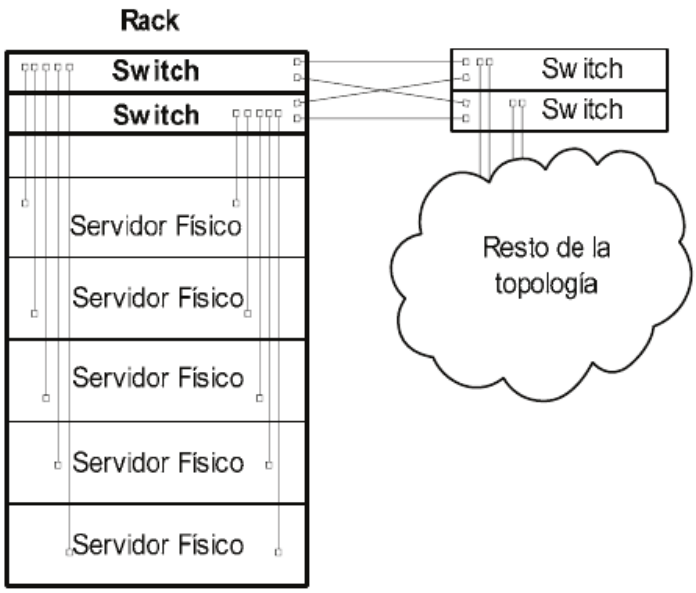


Fig. 10-25 Redundancia de enlace en sistemas virtualizados.

Una vez implementada y operativa la infraestructura necesaria, los servicios que alberga estarán a disposición de los usuarios, que, sobre entornos corporativos, serán aquellos ubicados en la red privada, mientras que para soluciones WAN, cualquiera con acceso a la red pública. En este último caso, todo servicio ubicado en la nube debe cumplir una serie de requisitos con el fin de ser considerado como parte de *"Cloud Computing",* siendo los siguientes:

- Ha de ser compatible con diferentes tipos de dispositivo, sistemas operativos y modos de conexión.

- Resulta imprescindible que su acceso sea bajo demanda y mediante un proceso totalmente automatizado. Por ejemplo, la solicitud de una cuenta de correo electrónico normalmente se lleva a cabo mediante un formulario vía web. Una vez cumplimentado, esta será creada automáticamente y el usuario podrá hacer uso de la misma de manera inmediata. Este mismo procedimiento es aplicado sobre servicios más avanzados, como las diferentes opciones de *hosting*.

- Tener la capacidad de soportar un aumento constante del número de usuarios sin que ello influya sobre el rendimiento del servicio.

- Ofrecer alta disponibilidad gracias a la redundancia tanto en servidores como conexiones.

En párrafos anteriores se ha mencionado el amplio rango de servicios de todo tipo y complejidad ubicados en la nube. Muchos de ellos, orientados hacia soluciones corporativas o profesionales, reciben diferentes nomenclaturas conforme a sus características. Tres de ellos, *"SaaS", "IaaS"* y *"PaaS"* forman parte del contenido de CCNA, siendo abordados de manera meramente informativa.

SOFTWARE AS A SERVICE (SAAS)

Como su nombre indica, *SaaS* hace referencia a la contratación, por parte del cliente, de un determinado software, el cual será ejecutado sobre una máquina virtual ubicada en la nube cuyas características de hardware y sistema operativo dependen por completo del proveedor. Es decir, el cliente tan solo obtiene acceso a una aplicación ya instalada, configurada, licenciada y gestionada. Resultan innumerables las opciones disponibles en la nube consideradas como *SaaS*, siendo algunas de ellas *Drop Box, Google Drive* o *Microsoft Exchange*.

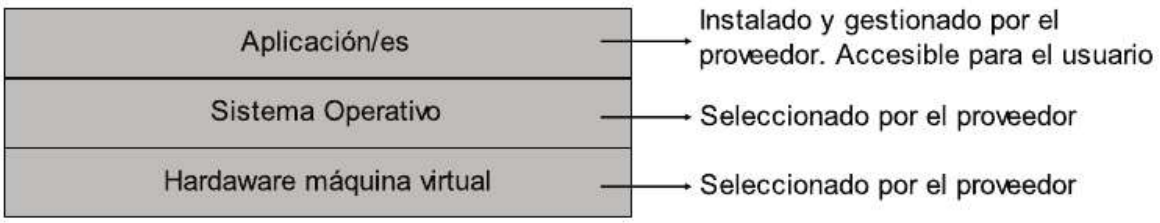


Fig. 10-26 Software as a service.

INFRAESTRUCTURE AS A SERVICE (IAAS)

IaaS hace referencia al servicio mediante el cual se habilita la posibilidad de contratar una máquina virtual ubicada en la nube, la cual será de uso privado y cuyas características de hardware y sistema operativo son seleccionadas por el propio cliente. Tras ello, el acceso a su administración se lleva a cabo vía web, pudiendo instalar y ejecutar las aplicaciones y servicios que considere oportuno.

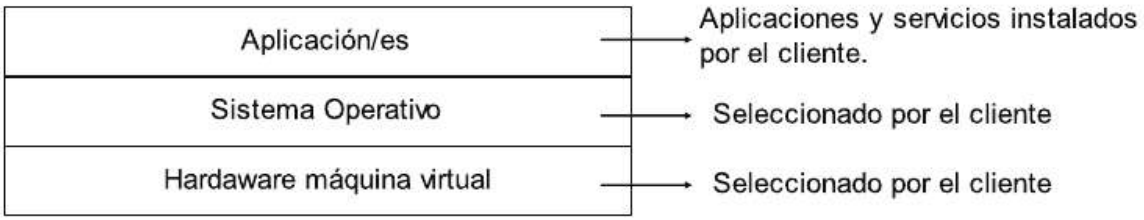


Fig. 10-27 Infrastructure as a service.

PLATFORM AS A SERVICE (PAAS)

PaaS resulta muy similar a *IaaS*, donde el cliente selecciona el hardware y sistema operativo de la máquina virtual a contratar. Sin embargo, en este caso, el proveedor incluye preinstalada una serie de aplicaciones (*Suite*), las cuales, dependiendo del propósito de la VM, podrán variar. Por ejemplo, las opciones *PaaS* orientadas a la creación de plataformas web normalmente incluirán software útil para ello, como *WordPress, MySQL*, y métodos de pago online, entre otros. Además de la Suite, el cliente también podrá instalar las aplicaciones que considere oportunas.

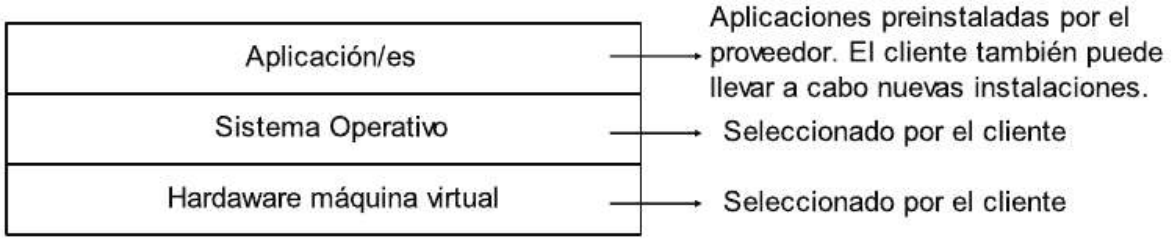
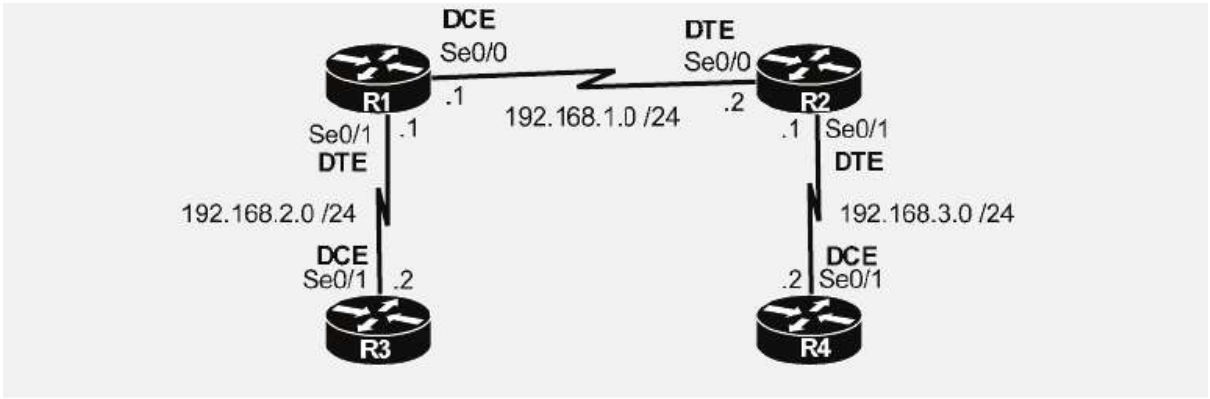


Fig. 10-28 Platform as a service.

La puesta en marcha de prácticamente todos los servicios ubicados en la nube normalmente consta de un formulario, que tras ser cumplimentado y abonada la tasa correspondiente, ejecuta un proceso automático que concluye con la creación del servicio en cuestión, el cual será ubicado sobre los servidores del proveedor, siendo su acceso y administración únicamente autorizada al cliente que lo ha contratado.

SOLUCIÓN DE RETOS: REDES WAN

Reto 10.1 – Configurar la siguiente topología de tal manera que todos los enlaces seriales hagan uso de HDLC con una frecuencia de reloj de 56000 bps.

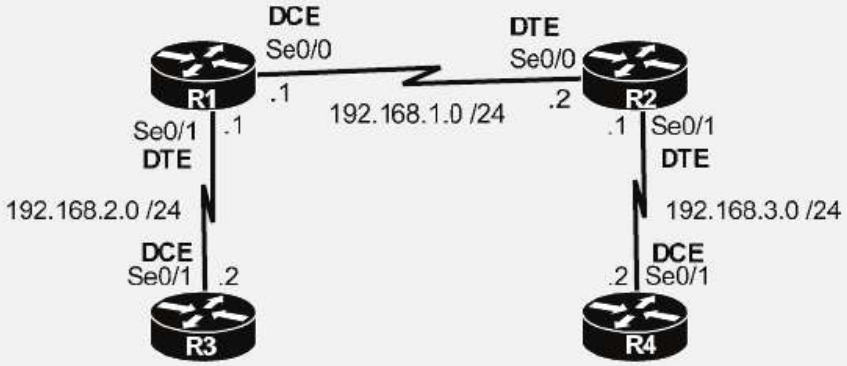



```
---Configuración en R1---
R1(config)#int se0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#encapsulation hdlc
R1(config-if)#clock rate 56000
R1(config-if)#no shutdown
R1(config)#int se0/1
R1(config-if)#ip address 192.168.2.1 255.255.255.0
R1(config-if)#encapsulation hdlc
R1(config-if)#no shutdown

---Configuración en R2---
R2(config)#int se0/0
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#encapsulation hdlc
R2(config-if)#no shutdown
R2(config)#int se0/1
R2(config-if)#ip address 192.168.3.1 255.255.255.0
R2(config-if)#encapsulation hdlc
R2(config-if)#no shutdown

---Configuración en R3---
R3(config)#int se0/1
R3(config-if)#ip address 192.168.2.2 255.255.255.0
R3(config-if)#encapsulation hdlc
R3(config-if)#clock rate 56000
R3(config-if)#no shutdown

---Configuración en R4---
R4(config)#int se0/1
R4(config-if)#ip address 192.168.3.2 255.255.255.0
R4(config-if)#encapsulation hdlc
R4(config-if)#clock rate 56000
R4(config-if)#no shutdown
```

Reto 10.2 – Configurar el enlace serial entre R1 y R2 haciendo uso de PPP y autenticación CHAP conforme a los siguientes requisitos:

- Frecuencia de reloj: 56 kbps.
- Contraseña CHAP: “CHAPpass”.

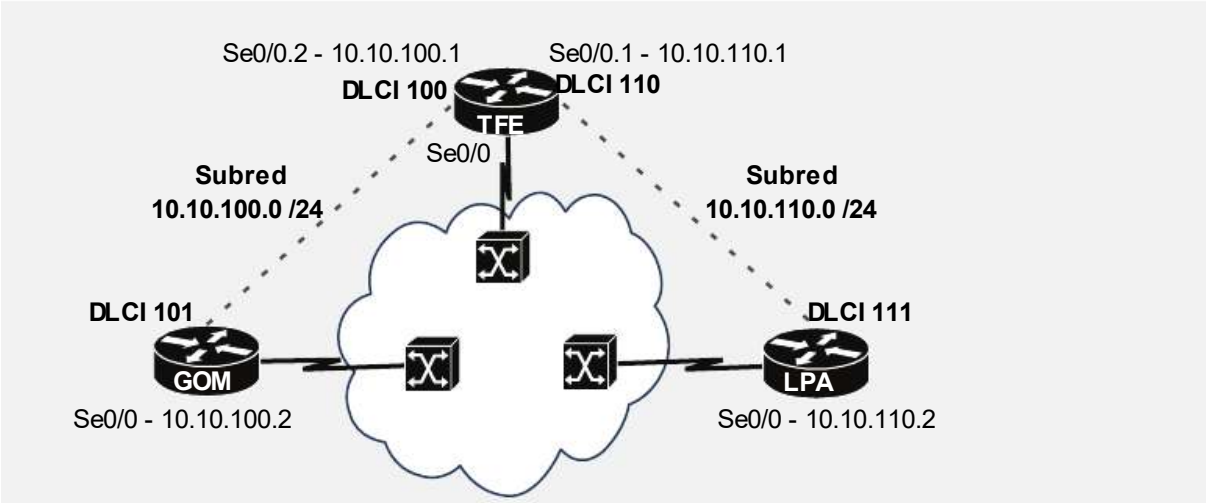
- Nombres de los routers: indicados en la topología.




```
---Configuración en R1---
R1(config)#int se0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#encapsulation ppp
R1(config-if)#ppp authentication chap
R1(config-if)#clock rate 56000
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)# username R2 password CHAPpass

---Configuración en R2---
R2(config)#int se0/0
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication chap
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)# username R1 password CHAPpass
```

Reto 10.3 – Configurar Frame Relay en todos los routers de la siguiente topología de tal manera que:

- El protocolo de encapsulado en capa 2 sea ietf.
- Se aplique la función autosense para seleccionar el tipo de LMI.
- En el router TFE, configurar los circuitos virtuales mediante subinterfaces, mientras que en GOM y LPA, realizar la configuración directamente desde la interfaz física, llevando a cabo un mapeo de IP – DLCI.




```
---Configuración en TFE---
TFE(config)#int se0/0
TFE(config-if)#encapsulation frame-relay ietf
TFE(config-if)#no shutdown
TFE(config-if)#exit
TFE(config)#interface Se0/0.1 point-to-point
TFE(config-subif)#ip address 10.10.110.1 255.255.255.0
TFE(config-subif)#frame-relay interface-dlci  110
TFE(config-subif)#exit
TFE(config)#interface Se0/0.2 point-to-point
TFE(config-subif)#ip address 10.10.100.1 255.255.255.0
TFE(config-subif)#frame-relay interface-dlci  100

---Configuración en GOM---
GOM(config)#int se0/0
GOM(config-if)#encapsulation frame-relay ietf
GOM(config-if)#ip address 10.10.100.2 255.255.255.0
GOM(config-if)#frame-relay map ip 10.10.100.1 101 broadcast
GOM(config-if)#no shutdown

---Configuración en LPA---
LPA(config)#int se0/0
LPA(config-if)#encapsulation frame-relay ietf
LPA(config-if)#ip address 10.10.110.2 255.255.255.0
LPA(config-if)#frame-relay map ip 10.10.110.1 111 broadcast
LPA(config-if)#no shutdown
```

TEST CAPÍTULO 10: REDES WAN

1.- ¿Cuál de las siguientes características identifica la diferencia principal entre las redes MAN y WAN?

A. Dispositivos.
B. Tamaño.
C. Número de usuarios.
D. Medios físicos.

2.- ¿En qué capas del modelo OSI operan los enlaces punto a punto en redes WAN? (Seleccionar dos respuestas)

A. Capa 1.
B. Capa 2.
C. Capa 3.
D. Capa 4.
E. Capa 5.

3.- ¿En qué lugar se encuentran ubicados físicamente los switchs FR?

A. En la red LAN.
B. En el CO del ISP.
C. En la red WAN.
D. Ninguna de las anteriores.

4.- De las siguientes opciones, ¿cuáles representan tecnologías de acceso a Internet? (Seleccionar dos respuestas)

A. VSAT.
B. ISDN.
C. Ethernet WAN.
D. MPLS.
E. Cable.

5.- ¿Qué modo de transmisión es aplicado comúnmente en capa 1 sobre las líneas arrendadas para realizar el envío de datos a través del medio físico?

A. PPP.
B. HDLC.
C. TDM.
D. DLCI.

6.- HDLC es un protocolo…

A. De capa 1 aplicado en enlaces WAN.
B. De capa 1 aplicado en redes Ethernet.
C. De capa 1 aplicado en redes Frame Relay.
D. De capa 2.

7.- ¿Qué identifican las direcciones DLCI en redes Frame Relay?

A. Un router vecino.
B. La dirección IP del router local.
C. La dirección IP del router remoto.
D. Un circuito virtual.

8.- De los siguientes servicios ofrecidos por los ISP, ¿cuáles hacen uso de la tecnología Ethernet WAN? (Seleccionar dos respuestas)

A. MetroE.
B. VPLS.
C. VPN.
D. MPLS.

9.- ¿Sobre qué medio físico opera la tecnología VSAT?

A. Fibra.
B. Cableado UTP.
C. Cableado STP.
D. Ninguna de las anteriores.

10.- ¿Cuál es la función principal de un módem?

A. Enrutar paquetes desde la LAN hacia la WAN.
B. Establecer la comunicación DSL entre cliente e ISP.
C. Convertir señales analógicas en digitales y viceversa.
D. Permitir el uso de voz y datos de manera simultánea.

11.- ¿Qué es el DSLAM?

A. El dispositivo, ubicado físicamente en la red LAN, encargado de dividir los datos de voz y vídeo antes de ser enviados a la WAN.
B. El dispositivo, ubicado físicamente en la red LAN, encargado de dividir la comunicación digital y analógica antes ser enviada a la WAN.
C. El dispositivo, ubicado físicamente en la red WAN, encargado de reenviar los datos recibidos desde el cliente a la red adecuada (voz o datos).
D. El dispositivo, ubicado físicamente en la red WAN, encargado de verificar que el cliente que accede a Internet ha contratado el servicio previamente.

12.- En un enlace serial, ¿qué router establece la frecuencia de reloj?

A. Aquel configurado con mayor ancho de banda mediante comando *bandwidth*.
B. Aquel que actúa como DTE.
C. Aquel que actúa como DCE.
D. Aquel seleccionado por el protocolo HDLC previa negociación entre ambos.

13.- ¿Qué diferencia existe entre una trama HDLC estándar y otra propietaria de Cisco?

A. Ninguna.
B. El campo *Type*.
C. El número de bits destinados para cada campo.
D. El algoritmo aplicado para el cálculo del CRC.

14.- En un enlace serial entre dos routers Cisco no ha sido configurado ningún protocolo de encapsulado en capa 2. ¿Cuál será aplicado?

A. PPP.
B. Ninguno, no se establece el enlace.
C. HDLC.
D. PPP o HDLC, previa negociación entre ambos.

15.- ¿Qué tipo de autenticación es considerada más segura y por lo tanto recomendada para el protocolo HDLC?

A. PAP.
B. CHAP.
C. LDAP.
D. Ninguna de las anteriores.

16.- ¿Qué protocolos son utilizados por PPP para proveer control y soporte a los diferentes protocolos de capa 3?

A. Protocolos CHAP.
B. Protocolos LCP.
C. Protocolos NCP.
D. Protocolos IP.

17.- De las siguientes opciones, ¿cuál no identifica una característica de PPP?

A. Detección de errores.
B. Encapsulado de paquetes en capa 3.
C. Soporte multi-link.
D. Autenticación.

18.- IPCP es un ejemplo de protocolo…

A. NCP.
B. LCP.
C. IP.
D. De capa 3.

19.- ¿Por qué es recomendada la aplicación de CHAP en lugar de PAP?

A. Porque no es propietario de Cisco y puede ser utilizado en un entorno multi-fabricante.
B. Porque la negociación entre ambos extremos se lleva a cabo en capa 3.
C. Porque el proceso de validación pueden ser resuelto a través de un servidor externo RADIUS o TACACS+.
D. Ninguna de las anteriores.
E. Las respuestas A y B son correctas.

20.- En un router configurado con PPP se ha aplicado el comando "*username CARL password SECRET*" como credenciales de autenticación CHAP. Sin embargo, el enlace no se establece, indicando CARL como nombre de usuario incorrecto. ¿Qué acción se podrá llevar a cabo para solucionar el problema?

A. Aplicar el comando "*username CARL password SECRET*" en el router del otro extremo, desde el modo de configuración de la interfaz serial.
B. Aplicar el comando "*username CARL password SECRET*" en el router del otro extremo, desde el modo de configuración global.
C. Aplicar el comando "*hostname CARL*" en el router del otro extremo.
D. La configuración CHAP no acepta el comando "*username….",* debe aplicarse la sentencia "*ppp authentication username CARL password SECRET*".

21.- Tras ejecutar un "*show ip interface brief*" se comprueba que un enlace serial punto a punto se encuentra en estado "*up/down*" en ambos extremos de manera permanente. ¿Cuáles de las siguientes opciones identifican la causa más probable de dicha situación? (Seleccionar dos respuestas)

A. Error de autenticación.
B. Configuración no coincidente en mensajes *keepalive*.
C. Diferente encapsulación en ambos extremos.
D. Error en capa 1.
E. Error en capa 2.
F. Error en capa 3.

22.- En redes Frame Relay, ¿qué es el DLCI?

A. El protocolo utilizado entre el DTE y el DCE para la comunicación y mantenimiento del enlace.
B. La velocidad establecida para el circuito virtual.
C. Una dirección de capa 2.
D. La asociación entre capa 2 y capa 3 en un circuito virtual.

23.- De las siguientes opciones, ¿cuál representa una función propia de los mensajes LMI?

A. Identificar la dirección IP del router vecino.
B. Mantener la adyacencia entre el DTE y el DCE.
C. Establecer un circuito virtual y mantenerlo.
D. Identificar el protocolo de capa 3 que será encapsulado dentro de la trama Frame Relay.

24.- En un diseño de red parcialmente mallado en Frame Relay…

A. Se aplica la misma subred sobre todos los circuitos virtuales.
B. La comunicación entre determinados routers se lleva a cabo a través de otro intermediario.
C. Todos los miembros disponen de un circuito virtual hacia el router con el rol de master, que será el encargado de reenviar la comunicación.
D. Los circuitos virtuales deben ser configurados como multiacceso.

25.- Las siglas PVC significan…

A. Private Virtual Circuit.
B. Personal Virtual Circuit.
C. Permanent Virtual Circuit.
D. Ninguna de las anteriores.

26.- ¿Qué propósito desarrolla la función "*LMI autosense*"?

A. La selección automática del protocolo de encapsulado Frame Relay a aplicar.
B. La selección automática del tipo de LMI, basándose en el protocolo utilizado por el router del otro extremo.
C. La asignación automática del número de DLCI sobre cada circuito virtual.
D. Ninguna de las anteriores.

27.- ¿En qué modelos de Frame Relay se suele utilizar una única subred para todos los enlaces?

A. En los diseños totalmente mallados.

B. En los diseños parcialmente mallados.
C. En los diseños híbridos.
D. Todas las respuestas anteriores son correctas.

28.- En un router se ha ejecutado el comando "*frame-relay map ip 10.10.10.10 40 broadcast*". ¿Qué acción se llevará a cabo?

A. Asociar la IP local 10.10.10.10 al DLCI local 40.
B. Asociar la IP remota 10.10.10.10 al DLCI local 40.
C. Asocial la IP local 10.10.10.10 al DLCI remoto 40.
D. Asociar la IP remota 10.10.10.10 al DLCI remoto 40.

29.- El comando *"encapsulation frame-relay*" debe ser aplicado…

A. En el modo de configuración global.
B. En el modo de configuración de la interfaz física.
C. En el modo de configuración de la subinterfaz.
D. En el modo de configuración del protocolo.

30.- En sistemas virtualizados, ¿cómo gestiona el *hypervisor* la comunicación de red de todas las máquinas virtuales? (Seleccionar dos respuestas)

A. A través de un switch físico.
B. A través de un switch virtual.
C. Mediante la asignación de interfaces de red físicas.
D. Mediante la asignación de interfaces de red virtuales.

31.- *IaaS* hace referencia al servicio ubicado en la nube que…

A. Permite al cliente acceder a un determinado software, ya instalado y con todas sus funciones operativas.
B. Permite al cliente contratar una máquina virtual, pudiendo seleccionar tanto sus características de hardware como sistema operativo, sobre el cual el proveedor preinstala una *Suite* de aplicaciones.
C. Permite al cliente contratar una máquina virtual, totalmente operativa y cuyo software es instalado y gestionado por el proveedor.
D. Ninguna de las anteriores.

IP VERSIÓN 6 11

PROTOCOLO IPV6: CONCEPTOS BÁSICOS

Desde su aparición, el protocolo por excelencia de capa 3 tanto en redes públicas como privadas ha sido IPv4, cuyo formato parecía albergar un rango de direcciones lo suficientemente amplio como para permitir el acceso de cualquier dispositivo a la red pública a lo largo del tiempo. Sin embargo, el gran auge experimentado por Internet y la infinidad de dispositivos ubicados en la nube han propiciado el agotamiento de las direcciones disponibles, y con ello, la necesidad de desarrollar un nuevo protocolo que permita continuar con el crecimiento de la red y que a su vez asegure que no se producirá un nuevo desgaste.

Con ello nace IPv6 *(IP versión 6),* el cual será objeto de estudio durante el presente capítulo y cuyas funciones coinciden con las llevadas a cabo por su antecesor, es decir, definir diferentes tipos de redes y el formato de sus direcciones, permitiendo la comunicación entre todas ellas. Además, su aparición conlleva la necesidad de actualizar multitud de protocolos diseñados específicamente para IPv4, como ICMP, creando una nueva versión, ICMPv6. De igual manera se desarrollan OSPFv3, EIGRPv6 o DHCPv6, todos ellos serán analizados en párrafos posteriores.

Para comenzar, nada mejor que analizar el nuevo formato de direcciones y su método de enrutamiento, para a posteriori profundizar en aspectos más técnicos como el direccionamiento, subnetting, DHCPv6, EIGRPv6 y OSPFv3.

Formato de direcciones

Uno de los objetivos principales de IPv6 consiste en asegurar que en el futuro no se producirá un desgaste como el sufrido por IPv4. Para lograrlo, el nuevo protocolo hace uso de direcciones con una longitud de 128 bits, frente a los 32 utilizados por su antecesor, logrando de esta manera un rango infinitamente mayor y difícilmente agotable. En este caso, las IP son representadas en formato hexadecimal, dividiéndolas en 8 bloques de 4 dígitos cada uno, separados entre ellos por el signo de puntuación ":". Algunos ejemplos podrían ser:

- 2031:0000:130F:0000:0000:09C0:876A:130B
- 0983:FB61:1111:5555:00AD:F166:0011:7433
- DEFF:0000:0000:8632:AAAA:0000:0000:EF4A
- 00FA:0000:0000:0000:0000:0000:0000:0000

Donde cada dígito hexadecimal equivale a 4 bits, de tal manera que la suma de todos ellos (32x4) hacen el total de 128. La conversión a binario, aunque rara vez resulte necesaria, se lleva a cabo modificando cada valor hexadecimal por su correspondiente en binario, representados en la siguiente tabla:

Hexadecimal	Binario	Hexadecimal	Binario
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

Sin embargo, para el ser humano, trabajar con direcciones de este tamaño y mucho más memorizarlas se convierte en una tarea extremadamente complicada. Debido a ello, IPv6 también permite su representación de manera abreviada conforme a los siguientes criterios:

- Los bloques o cuartetos (cuartetos de ahora en adelante) que comiencen por uno o varios ceros pueden ser representados sin incluir dichos valores. Por ejemplo, "00A0" puede ser abreviado como "A0". Hay que tener en cuenta que solo pueden ser suprimidos los valores iniciales.

- Un cuarteto compuesto únicamente por ceros puede ser abreviado con tan solo uno de sus valores. De esta manera, “0000” puede ser representado simplemente como “0”.

- Dos o más cuartetos sucesivos compuestos únicamente por ceros pueden ser representados como “::”, suprimiendo todos sus valores. Sin embargo, esta condición tan solo puede ser aplicada una vez en cada dirección IP**.** Por ejemplo, “0000:0000:AAAA:0000:0000” puede ser abreviado como “::AAAA:0:0” o “0:0:AAAA::”…nunca como “::AAAA::”.

Aplicando dichos criterios sobre las direcciones anteriormente representadas:

IPv6 Completa	**IPv6 abreviada**
2031:0000:130F:0000:0000:09C0:876A:130B	2031:0:130F::9C0:876A:130B
0983:FB61:1111:5555:00AD:F166:0011:7433	983:FB61:1111:5555:AD:F166:11:7433
DEFF:0000:0000:8632:AAAA:0000:0000:EF4A	DEFF::8632:AAAA:0:0:EF4A
00FA:0000:0000:0000:0000:0000:0000:0000	FA::

LONGITUD Y PREFIJO DE RED

IPv6 incluye dos nuevos términos relacionados con cada dirección IP:

Longitud: La longitud identifica el número de bits destinados a la parte de red y parte de hosts de una determinada dirección, es decir, desarrolla la misma función que la máscara en IPv4. Es representada como “/x”, siendo *x* el número total de bits que la componen.

Prefijo: El prefijo hace referencia a la red a la cual pertenece el host.

Por ello, resulta posible calcular el prefijo gracias a la longitud aplicada en cada dirección, para lo cual bastará con llevar a cabo el siguiente procedimiento:

- *Paso 1:* Dada una longitud, identificar su número total de bits en la dirección IP. Las longitudes deben ser múltiplo de 4, lo cual facilita el cálculo, ya que 1 dígito hexadecimal equivale a 4 bits.

- *Paso 2:* Asignar el valor 0 sobre los bits restantes.

Supuesto práctico 1: Determinar el prefijo IPv6 de la dirección 983:FB61:1111:5555:AD:F166:11:7433/64

Paso 1: Para iniciar el cálculo es recomendable disponer de la IP completa. En este caso ha sido abreviada, por lo que la primera acción a llevar a cabo consiste en modificar su representación:

983 = 0983
FB61= FB61
1111= 1111
5555= 5555
AD= 00AD
F166= F166
11= 0011
7433= 7433

Lo que es igual a 0983:FB61:1111:5555:00AD:F166:0011:7433/64.

La longitud indica que el prefijo hace uso de 64 bits, que corresponden a 16 dígitos hexadecimales (64/4), por lo tanto:

0983:FB61:1111:5555:00AD:F166:0011:7433

Paso 2 : Asignar el valor cero al resto de dígitos de la dirección:

0983:FB61:1111:5555:0000:0000:0000:0000, que abreviada corresponde a 983:FB61:1111:5555::

El prefijo de la IP 0983:FB61:1111:5555:00AD:F166:0011:7433/64 es **983:FB61:1111:5555::**

Supuesto práctico 2: Calcular el prefijo IPv6 de la dirección 2031:0:130F::9C0:876A:130B/56

Paso 1: En este caso, la dirección completa corresponde al valor hexadecimal 2031:0000:130F:0000:0000:09C0:876A:130B, sobre la cual se deberán identificar los 56 primeros bits, ya que son aquellos que hacen referencia al prefijo, de tal manera que:

56 /4 = 14

2031:0000:130F:0000:0000:09C0:876A:130B

Paso 2: El resto de la dirección a cero:

2031:0000:130F:0000:0000:0000:0000:0000 que abreviado es igual a 2031:0:130F::

El prefijo de la IP 2031:0000:130F:0000:0000:09C0:876A:130B es 2031:0:130F::

Otros ejemplos…

Dirección / Longitud	**Prefijo**
89BB:2011:0AAA:F621:0000:ABDC:3199:86BB /64	89BB:2011:AAA:F621::
34AC:3666:BABA:FECE:0690:9999:AA9A:DEDE /16	34AC::
B163:2011:0AAA:6AA6:BBBB:CCCC:DDDD:EEEE /4	B::
89BB:0000:0000:F621:0000:ABDC:3199:86BB /48	89BB:0:0::
89BB:0000:0000:0621:0000:ABDC:3199:86BB /88	89BB:0:0:621:0:AB::

Enrutamiento

El proceso de enrutamiento llevado a cabo en IPv6 coincide con el ejecutado por IPv4, con la única diferencia que en la tabla de rutas se almacenan y vinculan prefijos, en lugar de IDs de subred. El procedimiento consta de:

- *Paso 1:* Cuando un paquete IPv6 es recibido por un router, este examina la dirección de destino en capa 3, y conforme a su longitud calcula el prefijo.

- *Paso 2:* Busca el resultado en la tabla de rutas.

- *Paso 3:* Si existe, reenvía el paquete a través la interfaz asociada, de lo contrario lo descarta.

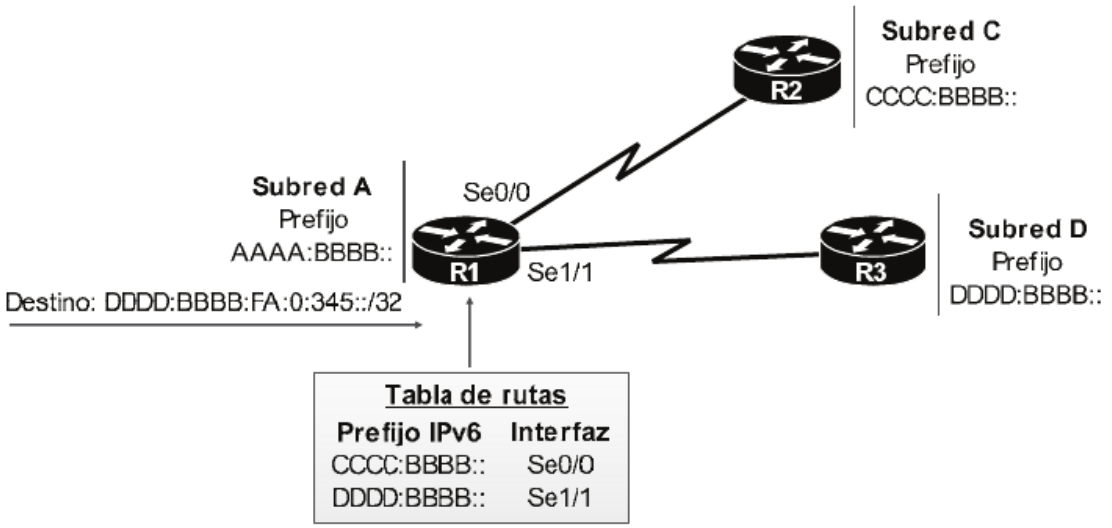


Fig. 11-1 Enrutamiento en IPv6.

Un equipo de la subred A envía un paquete a R1 con dirección de destino DDDD:BBBB:FA:0:345:: /32, el cual lo recibe y calcula su prefijo, siendo el resultado DDDD:BBBB::, ya que la longitud indicada es de 32 bits. Acto seguido lo busca en su tabla de rutas y concluye que el destino está asociado a la interfaz de salida Se1/1, enviando el paquete a través de la misma.

Protocolos de enrutamiento

Los protocolos de enrutamiento son aquellos encargados de aprender rutas hacia redes remotas de manera automática, asociando cada destino con la interfaz de salida local necesaria para llegar al mismo. En IPv6, la finalidad de estos coincide con la ya analizada en capítulos anteriores para IPv4, sin embargo, ha sido necesario desarrollar nuevas versiones con el fin de soportar las características del nuevo protocolo, siendo las siguientes:

Protocolo en IPv4	Protocolo actualizado para IPv6
RIP	RIPng (*RIP next generation)*
OSPFv2	OSPFv3
EIGRP	EIGRPv6
BGP	MP BGP-4

De los cuales, OSPFv3 y EIGRPv6 serán objeto de estudio a lo largo del presente capítulo.

DIRECCIONAMIENTO Y SUBNETTING EN IPV6

El protocolo IPv6 define dos tipos de direcciones, denominadas *global unicast addresses* y *unique local addresses.* Las primeras tan solo pueden ser utilizadas en redes públicas, mientras que las segundas son aquellas destinadas únicamente para entornos privados. Además, no son divididas en clases (como en IPv4), eliminando de esta manera la clasificación llevada a cabo por su antecesor. Este hecho, unido al formato de 128 bits de longitud, establece un rango de direcciones prácticamente inagotable que permite definir el direccionamiento en relación con dos modelos.

El primero de ellos, denominado *global unicast,* consiste en que todos los dispositivos de la compañía hagan uso de direcciones públicas, sin necesidad de aplicar NAT para acceder a Internet. En este caso, el reparto de IPs debe ser gestionado de manera global para que la misma dirección no pueda ser asignada sobre más de un dispositivo a nivel mundial. El organismo encargado de ello es IANA y su sistema de gestión de IPs será analizado en párrafos posteriores.

Mientras que el segundo, denominado *unique local*, se basa en el modelo actual de IPv4, es decir, el uso de una red privada cuyos miembros acceden a Internet a través de una o varias IPs públicas gracias a NAT.

Los siguientes párrafos serán dedicados a analizar en detalle ambos tipos, incluyendo el cálculo de subnetting en cada caso.

Global unicast

En *global unicast* cada dispositivo hace uso de una dirección pública diferente, por lo que todos ellos podrán comunicarse en Internet sin necesidad de aplicar técnicas como NAT o PAT. Para lograrlo resulta imprescindible mantener el control sobre la asignación de direcciones, ya que la misma IP no podrá ser utilizada por más de un dispositivo a nivel mundial. Para ello, IANA *(Internet Assigned Numbers Authority*) en colaboración con los RIRs *(Regional Internet Registries)* y los ISP llevan a cabo la siguiente política:

- *Paso 1:* IANA asigna prefijos de red a los RIR de cada país/continente.

- *Paso 2:* Estos a su vez lo dividen, definiendo un rango para cada uno de los ISP.

- *Paso 3:* Los ISP lo gestionan de tal manera que a cada uno de sus clientes se le concede un prefijo único y exclusivo.

- *Paso 4:* El propio cliente será el encargado de administrar su rango de direcciones.

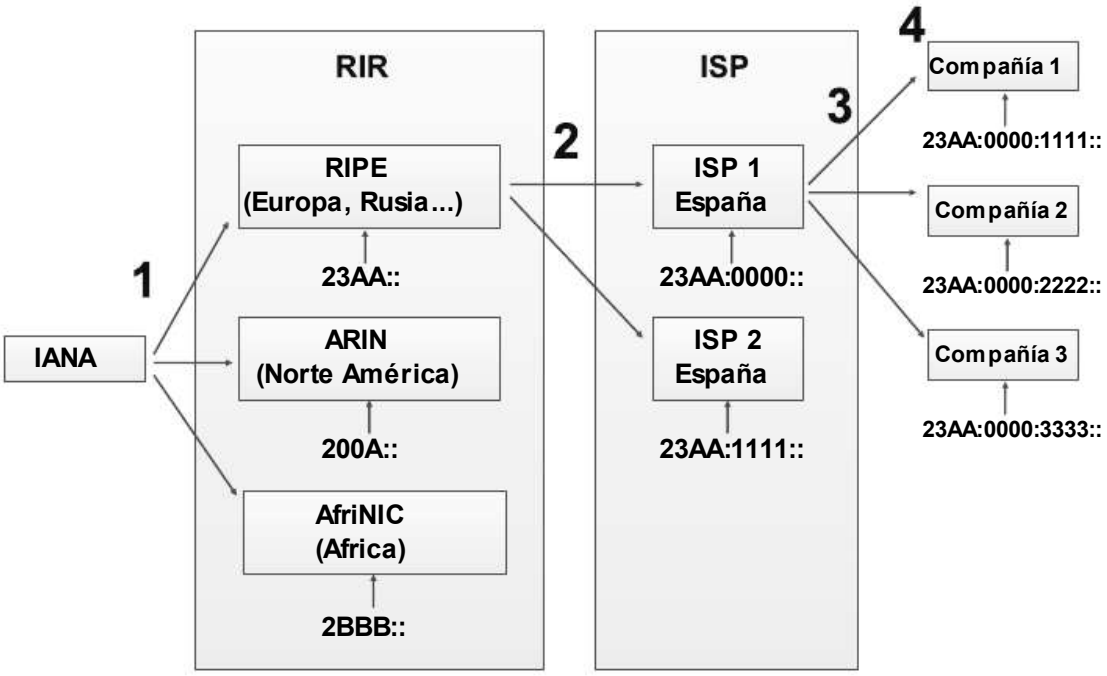


Fig. 11-2 Modo de asignación de rangos IPv6.

La compañía 1 ha obtenido el prefijo 23AA:0000:1111::/48. Este recibe el nombre de *global routing prefix* y debe ser único y exclusivo a nivel mundial. La longitud identifica tanto la red como el número de direcciones disponibles, donde los primeros 48 bits no pueden ser modificados, obteniendo con ello el rango 23AA:0:1111:: - 23AA:0:1111:FFFF:FFFF:FFFF:FFFF:FFFF. Con relación al mismo, la práctica más habitual consiste en crear subredes con el fin de facilitar la administración, mejorar la seguridad y optimizar el ancho de banda.

RANGO DE DIRECCIONES PÚBLICAS

Como se ha mencionado en párrafos anteriores, IPv6 no divide sus direcciones en clases, sin embargo, sí que distingue entre aquellas públicas y privadas. Para diferenciar ambas, el protocolo simplemente reserva ciertos dígitos hexadecimales para su aplicación sobre un determinado entorno, por ejemplo, todas las IP que comiencen por FD corresponden a direcciones de red privadas. En cuanto a las públicas, pueden ser asignadas todas aquellas que no hayan sido reservadas para otro propósito, los cuales se muestran en la siguiente tabla:

Tipo de dirección	Primeros dígitos hexadecimales
Global Unicast (pública)	Cualquiera que no esté reservada
Unique Local (privada)	FD
Multicast	FF
Link-Local	FE80::/10

De tal manera que IANA podrá asignar como dirección pública cualquiera que no comience con los dígitos “FD”, “FF” o “FE80::/10”.

SUBNETTING CON DIRECCIONES GLOBAL UNICAST

Una vez la compañía disponga de su prefijo de red, ella misma será la encargada de gestionarlo y administrarlo. La acción más común en estos casos consiste en dividirlo en subredes, obteniendo así los beneficios que su uso conlleva. El cálculo se efectúa de la misma manera que en IPv4, es decir:

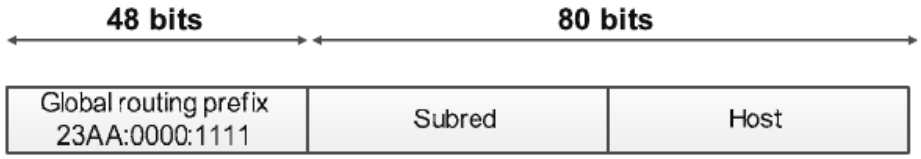
- El número de bits necesarios para crear subredes se calcula aplicando la fórmula 2^n.
- Para determinar el número máximo de hosts por subred se ejecuta la fórmula $2^n - 2$.

De tal manera que los 128 bits de una IP serán divididos en 3 partes. La primera identifica el *global routing prefix*, el cual no podrá ser modificado bajo ningún concepto. La segunda, los bits utilizados para subred y la tercera aquellos destinados para hosts.

Cálculo de subredes

La primera acción a llevar a cabo consiste en determinar el número de bits disponibles para iniciar el cálculo. Para ello, bastará con restar a 128 la longitud utilizada por el *global routing prefix.* La compañía 1 obtuvo el prefijo 23AA:0000:1111::/48, de tal manera que:

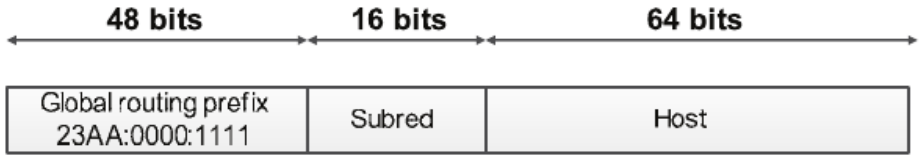
128 – 48 = 80



Tras ello, calcular cuántos de los bits disponibles son necesarios para poder crear el número total de subredes requeridas. Imagina que la compañía 1 es una multinacional con 5000 sucursales, necesitando la misma cantidad de subredes. Por lo tanto, la fórmula 2^n debe dar como resultado un valor igual o superior al indicado, teniendo en cuenta que "*n*" debe ser múltiplo de 4, ya que las direcciones son representadas en formato hexadecimal.

2^4 = 16. No cumple el requisito.
2^8 = 256. No cumple el requisito.
2^{12} = 4.096. No cumple el requisito.
2^{16} = 65.536. Cumple el requisito, se destinarán 16 bits para subred.

Los restantes formarán la parte de hosts, es decir, 128 - 48 - 16 = 64 bits para hosts.



¿Cuántos dispositivos podrá albergar cada subred?

2^{64} - 2 = 18.446.744.073.709.551.614 hosts.

Por último, tan solo bastaría identificar el prefijo de cada una de las subredes. Para una mayor comodidad y facilidad resulta recomendable sustituir los bits por dígitos hexadecimales y operar directamente sobre ellos. Continuando con el ejemplo:

Identificando el prefijo para cada subred

Los 16 bits destinados para la creación de subredes más los 48 utilizados por el prefijo global establecen una longitud de 64 bits, que corresponden a 16 dígitos hexadecimales, de los cuales, los 4 últimos identifican a cada una de las subredes. Para calcularlas simplemente hay que modificar sus valores, desde 0000 hasta FFFF:

Global routing prefix	Dígitos de subred	Prefijo de subred /Longitud
23AA:0000:1111	0000	23AA:0000:1111:0000:: /64
23AA:0000:1111	0001	23AA:0000:1111:0001:: /64
23AA:0000:1111	0002	23AA:0000:1111:0002:: /64
23AA:0000:1111	0003	23AA:0000:1111:0003:: /64
23AA:0000:1111	0004	23AA:0000:1111:0004:: /64
23AA:0000:1111	0005	23AA:0000:1111:0005:: /64
23AA:0000:1111	0006	23AA:0000:1111:0006:: /64
23AA:0000:1111	0007	23AA:0000:1111:0007:: /64
23AA:0000:1111	0008	23AA:0000:1111:0008:: /64
23AA:0000:1111	0009	23AA:0000:1111:0009:: /64
23AA:0000:1111	000A	23AA:0000:1111:000A:: /64
23AA:0000:1111	000B	23AA:0000:1111:000B:: /64
23AA:0000:1111	000C	23AA:0000:1111:000C:: /64
23AA:0000:1111	000D	23AA:0000:1111:000D:: /64
23AA:0000:1111	000E	23AA:0000:1111:000E:: /64
23AA:0000:1111	000F	23AA:0000:1111:000F:: /64
23AA:0000:1111	0010	23AA:0000:1111:0010:: /64
…Resto de subredes omitidas por brevedad…		

Supuesto práctico 1: La compañía “Canarias SL” dispone del prefijo global 2AAA:1515:FFFF:3131:0000:: /80**,** el cual desea dividir en mil subredes con un mínimo de 700 hosts en cada una de ellas.

Paso 1: Calcular cuántos bits quedan disponibles para subredes, restando a los 128 de la dirección los 80 utilizados por el prefijo global.

128 - 80 = 48 bits disponibles.

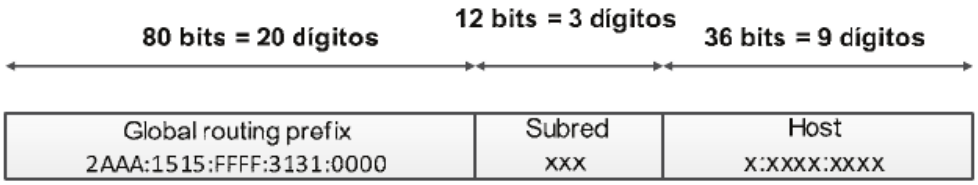
Paso 2: Determinar el número de bits necesarios para crear 1000 subredes, a través de la fórmula 2^n.

2^4 = 16. No cumple el requisito.
2^8= 256. No cumple el requisito.
2^{12} = 4.096. Cumple el requisito, se reservan 12 bits para subred, de tal manera que los 36 restantes (128 - 80 - 12) serán destinados para hosts.

Paso 3: Calcular el número de dispositivos que puede albergar cada subred, mediante la fórmula 2^n -2.

2^{36} - 2 = 68.719.476.734 hosts.

Con ello, la dirección mantendrá el siguiente formato.



Paso 4: Identificar el prefijo de cada subred. En este caso, la longitud es /92 (80 bits de prefijo global más 12 de subred).

Global routing prefix	**Dígitos de subred**	**Prefijo de subred /Longitud**
2AAA:1515:FFFF:3131:0000	000	2AAA:1515:FFFF:3131:0000:000:: /92
2AAA:1515:FFFF:3131:0000	001	2AAA:1515:FFFF:3131:0000:001:: /92
2AAA:1515:FFFF:3131:0000	002	2AAA:1515:FFFF:3131:0000:002:: /92
2AAA:1515:FFFF:3131:0000	003	2AAA:1515:FFFF:3131:0000:003:: /92
2AAA:1515:FFFF:3131:0000	004	2AAA:1515:FFFF:3131:0000:004:: /92
2AAA:1515:FFFF:3131:0000	005	2AAA:1515:FFFF:3131:0000:005:: /92
2AAA:1515:FFFF:3131:0000	006	2AAA:1515:FFFF:3131:0000:006:: /92
2AAA:1515:FFFF:3131:0000	007	2AAA:1515:FFFF:3131:0000:007:: /92
2AAA:1515:FFFF:3131:0000	008	2AAA:1515:FFFF:3131:0000:008:: /92
2AAA:1515:FFFF:3131:0000	009	2AAA:1515:FFFF:3131:0000:009:: /92
2AAA:1515:FFFF:3131:0000	00A	2AAA:1515:FFFF:3131:0000:00A:: /92
… Resto de subredes omitidas por brevedad…		

Supuesto práctico 2: A un ISP se le ha asignado el prefijo 3888:DDDD:: /32. Dicho proveedor de servicios dispone de 15 millones de clientes, sin embargo, se estima que esta cifra aumentará con el tiempo. Como administrador de red se le ha encomendado la tarea de dividir el rango adquirido en 200 millones de subredes. Cada una de las cuales será asignada a un cliente diferente. ¿Qué longitud de prefijo será la adecuada para lograr tal objetivo?

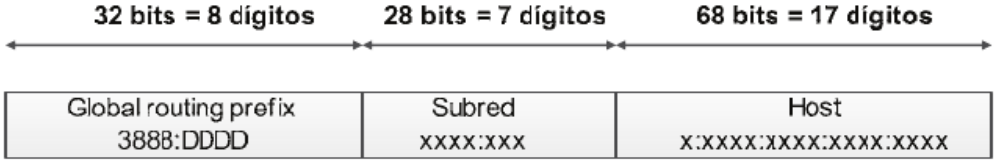
Paso 1: Calcular el número de bits disponibles para subredes.

128 - 32 = 96 bits.

Paso 2: Identificar cuántos de ellos son necesarios para lograr el objetivo.

2^{12} = 4096. No cumple el requisito.
2^{16} = 65.536. No cumple el requisito.
2^{20} = 1048576. No cumple el requisito.
2^{24} = 16.777.216. No cumple el requisito.
2^{28} = 268.435.456. Cumple el requisito.

De tal manera que la longitud adecuada es **/60**, compuesta por los 32 bits del prefijo global más los 28 destinados para subredes. La dirección mantendrá el siguiente formato:



Unique local

El modelo *unique local* resulta muy similar al implementado actualmente en IPv4, donde las compañías hacen uso de redes privadas y el acceso a la pública se lleva a cabo mediante NAT o PAT.

En IPv6, la diferencia entre una dirección pública y otra privada radica en los primeros dígitos que la componen, de tal manera que todas aquellas que comiencen por FD las identifican como privadas. Estas, al no ser enrutables sobre entornos públicos, pueden ser utilizadas por cualquier compañía y por supuesto no son asignadas por IANA ni ISPs, siendo las necesarias a aplicar en este modelo de direccionamiento.

Una dirección IPv6 *unique local* (privada) debe cumplir los siguientes requisitos:

- Comenzar con los valores hexadecimales **FD** (8bits).

- Disponer de un identificador único global, con una longitud de 40 bits y ubicado justo a continuación de los dígitos FD, para, entre ambos, completar los primeros 48 bits de la dirección. Este ID será utilizado como prefijo y con relación al mismo se crearán las subredes necesarias.

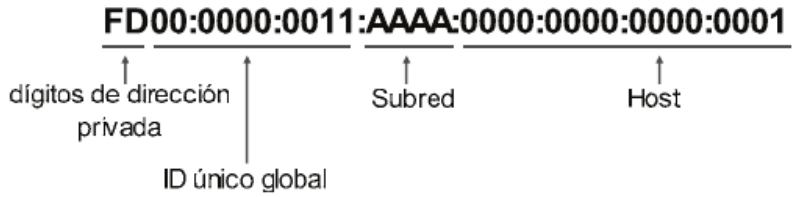
- Identificar la cantidad de bits reservados para subred. Normalmente 16.

- Identificar la cantidad de bits destinados para hosts. Normalmente 64.

De tal manera que una IP privada mantendrá el siguiente formato, donde la parte de subred y hosts puede variar dependiendo de las necesidades de cada compañía:

Fig. 11-3 Formato de dirección IPv6 Unique local.

Un ejemplo podría ser:



ID ÚNICO GLOBAL

Del formato recién analizado, lo que mayor confusión puede generar es el ID único global, el cual está compuesto por 10 dígitos hexadecimales seleccionados libremente por cada compañía y que tan solo tienen importancia a nivel local. Este, junto a los valores FD identifican el prefijo de la red privada, siendo lo más lógico aplicar combinaciones sencillas de recordar como FD00:1:1:: /48, FD22:2:2::/48 o similares. Una vez decidido, se crearán las subredes necesarias conforme al mismo.

Entonces, si la elección de dígitos depende por completo de la compañía y además solo es enrutable sobre entornos privados, ¿por qué se denomina "ID único global"? Es cierto que el término global hace referencia a direcciones públicas y ello puede generar cierta confusión, pero en este caso tan solo se trata de una recomendación. Aunque no resulte necesario ni mucho menos un requisito imprescindible, se considera una buena práctica que el prefijo seleccionado difícilmente coincida con cualquier otro. Ello es posible gracias a los 40 bits destinados a tal propósito, los cuales permiten un total de miles de millones de combinaciones diferentes.

Este hecho nace como consecuencia de que actualmente, en IPv4, cuando dos o más compañías se fusionan es muy común que el direccionamiento privado de cada una de ellas resulte idéntico, por ejemplo, que ambas hagan uso del rango 172.16.x.x, causando problemas de solapamiento o loops de enrutamiento, siendo necesario en estos casos un rediseño de la topología lógica. En IPv6, si dos o más prefijos coinciden generarían los mismos problemas.

SUBNETTING CON DIRECCIONES UNIQUE LOCAL

El cálculo de subnetting con direcciones privadas se lleva a cabo de la misma manera que el ya analizado para las globales. Bastará con determinar el número de bits disponibles y en relación con los mismos calcular cuántos serán necesarios para subred y cuántos para hosts.

El formato de direcciones IPv6 *unique local* incluye 48 bits inamovibles, de los cuales 8 corresponden a los dígitos FD y 40 al prefijo seleccionado por la compañía, por lo tanto, siempre se dispondrá del resto para la práctica de subnetting.

128 - 48 = 80 bits.

Imagina que la compañía 1, con diez mil subredes, basará su direccionamiento sobre un entorno privado, seleccionando como prefijo el ID FD00:000A:000A:: /48.

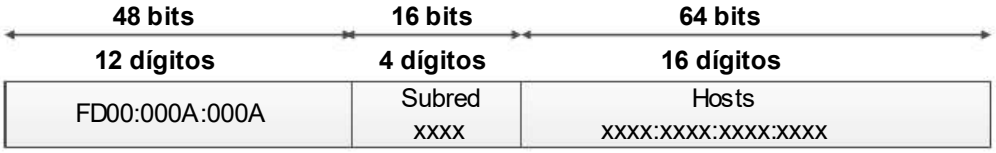
Paso 1: ¿Cuántos bits son necesarios para subred?

2^4 = 16. No cumple el requisito.
2^8= 256. No cumple el requisito.
2^{12} = 4.096. No cumple el requisito.
2^{16} = 65.536. Cumple el requisito.

Paso 2: ¿Cuántos para hosts?

80 -16 = 64 bits para hosts

De tal manera que:



Paso 3: Identificar los prefijos para cada subred.

Prefijo	Dígitos de subred	Prefijo de subred /Longitud
FD00:000A:000A	0000	FD00:000A:000A:0000:: /64
FD00:000A:000A	0001	FD00:000A:000A:0001:: /64
FD00:000A:000A	0002	FD00:000A:000A:0002:: /64
FD00:000A:000A	0003	FD00:000A:000A:0003:: /64
FD00:000A:000A	0004	FD00:000A:000A:0004:: /64
FD00:000A:000A	0005	FD00:000A:000A:0005:: /64
FD00:000A:000A	0006	FD00:000A:000A:0006:: /64
FD00:000A:000A	0007	FD00:000A:000A:0007:: /64
FD00:000A:000A	0008	FD00:000A:000A:0008:: /64
FD00:000A:000A	0009	FD00:000A:000A:0009:: /64
FD00:000A:000A	000A	FD00:000A:000A:000A:: /64
…Resto de subredes omitidas por brevedad…		

¿Por qué la longitud es /64? 48 bits del ID único global más 16 de subred.

CONFIGURACIÓN DE IPV6 EN ROUTERS CISCO

Aunque actualmente el protocolo en capa 3 más utilizado continúa siendo IPv4, resulta inevitable su sustitución en un futuro próximo, lo que conlleva adaptar todas las redes, tanto públicas como privadas, al nuevo modelo de direccionamiento. Proceder a dicho cambio, sobre todo en entornos corporativos de gran tamaño requiere tiempo y trabajo, lo que se traduce en una red inoperativa, reduciendo su producción a cero. Evidentemente, este hecho resulta totalmente inviable, por lo que la mejor alternativa consiste en utilizar ambos de manera simultánea hasta lograr la implementación total del nuevo protocolo. Esta técnica, denominada *dual stack*, permite a los hosts operar en IPv4 y/o IPv6, mientras que los dispositivos intermediarios serán capaces de enrutar ambos tipos de direcciones.

Para lograrlo bastará con configurar dos IP en cada interfaz, una por protocolo, que evidentemente deberán pertenecer al rango de red con la cual conectan. Además, en dispositivos Cisco también resulta necesario habilitar el enrutamiento IPv6, de tal manera que el proceso de configuración consta de dos partes:

- Habilitar el enrutamiento IPv6 en routers Cisco.
- Configuración de interfaces en IPv6.

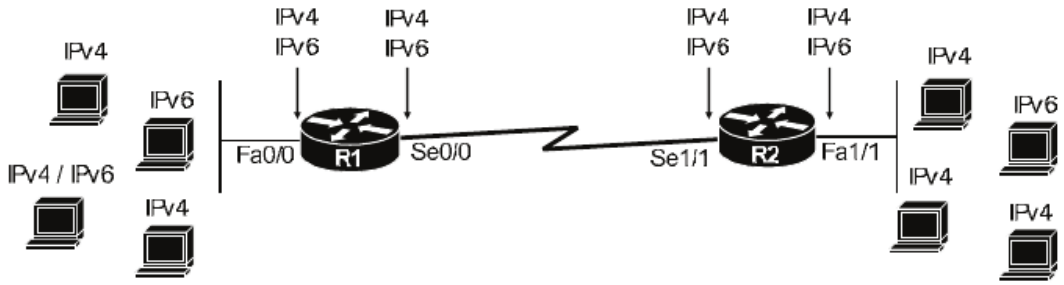


Fig. 11-4 Entorno dual-stack.

Habilitar enrutamiento IPv6 en routers Cisco

Un hecho a tener en cuenta es que la configuración por defecto en dispositivos Cisco de capa 3 tan solo incluye el enrutamiento IPv4, por lo que resulta necesario habilitar IPv6 de manera manual a través del comando **ipv6 unicast-routing,** ejecutado desde el modo de configuración global.

```
R1(config)#ipv6 unicast-routing
```

Su aplicación no conlleva deshabilitar IPv4. Ambos protocolos operarán de manera simultánea, es decir, en *dual-stack*.

Configuración de interfaces en IPv6

Habilitado el enrutamiento, la siguiente acción a llevar a cabo consiste en configurar las direcciones IPv6 en las interfaces necesarias. Para ello se podrá optar por diferentes métodos, como la aplicación manual, EUI-64, DHCP o SLAAC.

CONFIGURACIÓN MANUAL

Como su nombre indica, consiste en asignar manualmente la dirección IP. Esta puede ser definida de manera completa o abreviada, debiendo especificar también su longitud. El comando necesario para ello es **ipv6 address [*ip/longitud prefijo*]**, ejecutado desde el modo de configuración de la interfaz.

```
R1(config)#interface Fa0/0
R1(config-if)#ipv6 address 2AAA:0000:0000:AAAA:0000:0000:0000:0010/64
```

De manera abreviada...

```
R1(config)#interface Fa0/0
R1(config-if)#ipv6 address 2AAA:0:0:AAAA::10/64
```

CONFIGURACIÓN AUTOMÁTICA MEDIANTE EUI-64

El segundo método de configuración se basa en EUI-64, técnica que consiste en generar automáticamente los bits correspondientes a la parte de host de una dirección IPv6.

En este caso, el formato utilizado debe ser obligatoriamente de 16 dígitos hexadecimales (64 bits) para el prefijo y otros 16 para hosts. Los primeros deberán ser definidos manualmente, mientras que los segundos serán generados por EUI-64 en relación con la MAC del dispositivo. El proceso consta de los siguientes pasos:

- *Paso 1:* Los 12 dígitos de la MAC son divididos en dos partes de 6 cada una.

- *Paso 2:* Entre ambas partes se insertan los valores FFFE.

- *Paso 3:* De los 16 dígitos obtenidos, se convierten los dos primeros a binario, obteniendo 8 bits, de los cuales el valor del séptimo se invierte, si es 0 se establece en 1 y viceversa.

- *Paso 4:* Volver a realizar la conversión binario-hexadecimal. Con ello, la dirección queda definida.

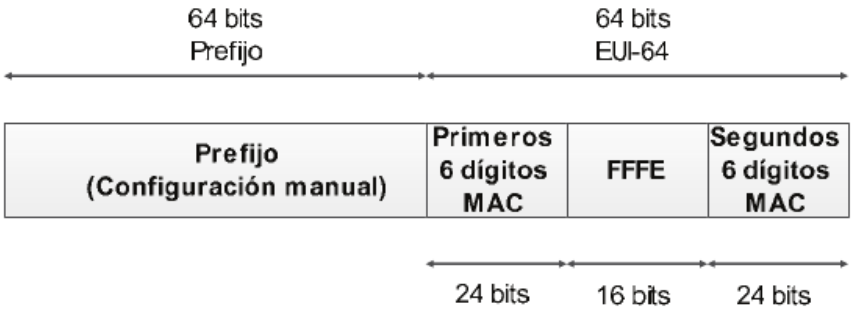


Fig. 11-5 Formato de dirección IPv6 EUI-64.

Supuesto práctico: Se ha configurado una interfaz para que obtenga su IP automáticamente mediante EUI-64, haciendo uso del prefijo AAAA:AAAA:AAAA:AAAA::/64. La MAC del dispositivo es 0844.6955.ABCD. ¿Qué dirección será generada?

Paso 1: Dividir la MAC 0844.6955.ABCD en dos partes de 6 dígitos hexadecimales cada una:

Parte 1: 084469
Parte 2: 55ABCD

Paso 2: Insertar los valores FFFE entre ambas:

084469 FFFE 55ABCD

Paso 3: Convertir los dos primeros dígitos a binario:

084469 FFFE 55ABCD
0 en binario equivale a 0000.
8 en binario equivale a 1000.

Localizar el séptimo bit (0000 1000) e invertir su valor, quedando definido como 0000 1010.

Paso 4: Volver a realizar la conversión binario-hexadecimal.

0000 en hexadecimal equivale a 0.
1010 en hexadecimal equivale a A.

Con ello, los 64 bits de la parte de host obtienen los siguientes valores: 0A4469 FFFE 55ABCD

Por lo tanto, la dirección IP generada por EUI-64 para dicho host es: AAAA:AAAA:AAAA:AAAA:0A44:69FF:FE55:ABCD /64

Este cálculo es ejecutado de manera automática por el dispositivo donde sea configurado.

El comando necesario en routers Cisco para que una interfaz ejecute EUI-64 es **ipv6 address *[prefijo /64]* eui-64,** desde el modo de configuración de la interfaz.

```
R1(config)#interface Fa0/0
R1(config-if)#ipv6 address 2AAA:0000:0000:AAAA::/64
```

OTROS MÉTODOS DE CONFIGURACIÓN

Cisco ofrece dos opciones más para que las interfaces obtengan su dirección IPv6 de manera automática e incluyendo el prefijo. Estas son el ya conocido DHCP y SLAAC.

El proceso llevado a cabo por DHCP coincide con el ya analizado en capítulos anteriores, pero adaptado al nuevo protocolo, es decir, el router envía un mensaje de solicitud a la red y el servidor DHCPv6 responderá a la misma, gracias a lo cual la interfaz obtendrá los datos de configuración de manera automática. Para ello, bastará con aplicar el comando **ipv6 address dhcp,** desde el modo de configuración de la interfaz en cuestión.

Por último, SLAAC *(Stateless Address Auto Configuration)* se basa en una técnica que consiste en generar automáticamente una IP conforme a la información incluida en los mensajes RA (*Router Advertisement),* los cuales son enviados por diferentes routers de manera periódica a la red. El comando necesario en este caso es **ipv6 address autoconfig**. Además, para poder aplicarlo, el dispositivo debe disponer de una dirección *Link-Local*, las cuales serán analizadas en párrafos posteriores. El modo de operación de SLAAC también será objeto de estudio en este mismo capítulo.

```
R1(config)# int Se0/0
```

```
R1(config-if)# ipv6 address dhcp
R1(config-if)# exit
R1(config)#int Fa0/0
R1(configif)# ipv6 address autoconfig
```

Verificación de configuración

Una vez aplicada cualquiera de las configuraciones anteriores, la manera más sencilla de verificarla es a través de los siguientes comandos:

- *show ipv6 interface brief:* Muestra un listado en pantalla que incluye aquellas interfaces configuradas con IPv6, facilitando datos como su estado, dirección IP, etc.

- *show ipv6 interface [interfaz]:* Muestra información detallada del protocolo IPv6 sobre una determinada interfaz, incluyendo su dirección IP, prefijo, dirección Link-local, estado, tamaño de MTU, etc.

Tipos de direcciones IPv6

Los diferentes métodos de configuración recién analizados, ya sea de manera manual o automática, identifican direcciones unicast, las cuales son utilizadas por los dispositivos para comunicaciones entre un origen y un destino definido, es decir, de un único emisor a un único receptor.

Sin embargo, IPv6, al igual que el resto de protocolos de capa 3, también requiere otros tipos de direcciones, cada una de ellas destinada a un propósito diferente. Las más destacables son:

- Link-Local.
- IPv6 multicast.
- IPv6 broadcast.
- ":" y "::1".

DIRECCIONES LINK-LOCAL

En IPv6, todos los hosts configurados con una IP también dispondrán de otra *Link-Local* de manera automática, es decir, el mismo dispositivo hará uso de dos direcciones, ambas unicast pero destinadas a diferentes tipos de comunicación. En este aspecto, la finalidad de una IP *global* o *unique* consiste en permitir el flujo de datos entre dispositivos ubicados en diferentes redes, mientras que las *Link-Local* suelen ser utilizadas para la comunicación entre hosts pertenecientes a la **misma** subred o para que ciertos protocolos hagan uso de las mismas con determinados fines. Solo forman parte de IPv6 (en IPv4 no existen), y sus características son:

- Son unicast.

- No son enrutables. Los routers no reenvían paquetes con direcciones *Link-Local* a través de sus interfaces, por lo que solo pueden ser utilizadas para la comunicación entre miembros de la misma subred.

- Son generadas automáticamente en relación con métodos definidos por cada fabricante. En dispositivos Cisco también puede ser configurada manualmente, como se verá a continuación.

- Son utilizadas por diferentes protocolos para determinados fines, siempre dentro de la misma subred. También los routers hacen uso de las mismas en ciertos casos para indicar el siguiente salto dentro de la tabla de rutas.

El formato mantiene 64 bits para el prefijo y otros 64 para la parte de hosts, debiendo cumplir los siguientes requisitos:

- El prefijo debe comenzar por FE80::/10.

- Cada dirección *Link-Local* dentro de la misma subred debe ser única. Para lograrlo, cada fabricante puede optar por diferentes métodos, por ejemplo, Cisco hace uso de EUI-64, sin embargo, Microsoft aplica otro proceso para completar los 64 bits de la parte de host.

Además, IOS también permite su configuración manual a través del comando **ipv6 address *[dir link-local]* link-local** desde el modo de configuración de la interfaz.

Una manera de proceder a su verificación es mediante el comando **show ipv6 interface brief**.

```
R1(config)#interface Fa0/0
R1(config-if)#ipv6 address 2AAA:0:0:AAAA::10/64
R1(config-if)#ipv6 address FE80:0:0:0:AAAA:AAAA:AAAA:AAAA/64 link-local
R1(config-if)# exit
R1(config)# int Se0/0
R1(config-if)# ipv6 address 2AAA:0:0:BBBB::1/64
R1(config-if)# exit
R1(config)# exit
R1# show ipv6 interface brief

FastEthernet0/0 [up/up]
FE80:0:0:0:AAAA:AAAA:AAAA:AAAA    //Link-Local definida manualmente.
2AAA:0:0:AAAA::10

Serial0/0 [up/up]
FE80::1FF:FE01:101    //Link-Local generada automáticamente.
2AAA:0:0:BBBB::1
```

DIRECCIONES IPV6 MULTICAST

IPv6 también opera con direcciones multicast, las cuales son utilizadas mayoritariamente por protocolos o aplicaciones cuando el destino de la comunicación debe identificar a un grupo de hosts. En este caso, el formato tan solo especifica los dígitos hexadecimales con los cuales debe comenzar la dirección, de tal manera que todas aquellas cuyos valores iniciales sean igual a FF hacen referencia a una IP multicast. Estas son definidas directamente por cada protocolo o servicio, y al igual que en IPv4, las más conocidas son reservadas por IANA para un fin específico, siendo algunas de las más importantes las siguientes:

Uso o protocolo	Dirección Multicast	Uso
Todos los hosts	FF02::1	El paquete será recibido por todos los hosts que operan con IPv6 en la misma subred.
Todos los routers	FF02::2	El paquete será recibido por todos los routers que operan con IPv6.
OSPFv3	FF02::5 FF02::6	FF02::5 es la dirección utilizada por OSPFv3 para el intercambio de mensajes entre los diferentes routers. Mientras, FF02::6 es aquella necesaria para la comunicación con los routers designados (DR).
EIGRPv6	FF02::A	Es la dirección utilizada por EIGRPv6 para el intercambio de mensajes entre los diferentes routers.

Otro tipo de direcciones son las denominadas "*solicited-node*", las cuales no son ni reservadas por IANA ni definidas por ningún protocolo o aplicación. En este caso, son generadas automáticamente por cada host con relación a los 6 últimos dígitos de su dirección IP, de tal manera que todos aquellos que coincidan en dichos valores pertenecerán al mismo grupo multicast *solicited-node* dentro de la misma subred. Estas suelen ser utilizadas para procesos como la traducción de direcciones IP a MAC, analizada en párrafos posteriores.

DIRECCIONES IPV6 BROADCAST

Las direcciones broadcast son aquellas utilizadas cuando la comunicación debe ser recibida por todos los hosts pertenecientes a la misma subred. Estas no son enrutables, por lo que el límite de un dominio broadcast lo establece el router más cercano. Su cálculo, al igual que en IPv4, se lleva a cabo asignando el valor 1 a todos

los bits de la parte de hosts. Por ejemplo, ¿cuál es la dirección broadcast de la subred AAAA::/96?

128-96 = 32 bits para hosts, todos con valor 1.

Obteniendo como resultado la IP: AAAA::FFFF:FFFF/96.

DIRECCIONES "::" Y "::1

Para finalizar con los tipos de direcciones pertenecientes a IPv6 resulta necesario hacer mención a la no especificada y a la loopback, ambas no configurables y que pueden servir de ayuda para identificar determinadas incidencias.

La dirección no especificada es la 0:0:0:0:0:0:0:0, abreviada a "**::**", y significa que por algún motivo el host aún no ha obtenido ninguna IP válida, por ejemplo, mientras espera a que un servidor DHCP le facilite los datos de conexión.

La dirección loopback es la ::1, utilizada para comprobar el funcionamiento del protocolo IPv6 en el propio host. Ejecutando un ping con éxito hacia la misma se verifica que está instalado y opera con normalidad.

CONFIGURACIÓN DE IPV6 EN HOSTS

Los dispositivos que operan en IPv6 requieren los mismos datos de conexión que aquellos en IPv4, es decir, dirección IP, longitud de prefijo, puerta de enlace y servidores DNS, pudiendo optar por su configuración de manera manual o automática gracias a DHCPv6 o SLAAC. Este último lleva a cabo un nuevo método basado en obtener el prefijo, IP y puerta de enlace sin necesidad de servidores DHCP. Para lograrlo hace uso del protocolo NDP *(Neighbor Discovery Protocol)* el cual será objeto de estudio a lo largo de los siguientes párrafos, para acto seguido proceder al análisis de DHCPv6 y SLAAC.

NDP - Neighbor Discovery Protocol

NDP es el protocolo utilizado por IPv6 para desarrollar dos de sus funciones principales. Primero, ejecuta el procedimiento necesario para la obtención de direcciones de manera automática, y segundo, incluye diferentes utilidades en torno a estas, como la detección de IPs duplicadas o el descubrimiento de MACs. Tanto DHCP como SLAAC hacen uso del mismo, por lo que la primera acción que lleva a

cabo un dispositivo configurado para obtener sus datos IPv6 de manera automática consiste en ejecutar NDP.

El protocolo define y desarrolla las siguientes funciones:

- Descubrimiento de routers.
- Descubrimiento del prefijo y longitud.
- Descubrimiento de direcciones MAC.
- Detección de direcciones duplicadas.

DESCUBRIMIENTO DE ROUTERS

Se basa en detectar los routers pertenecientes a su misma subred, y conforme a los resultados obtenidos establecer uno de ellos como puerta de enlace predeterminada. Para lograrlo, NDP ejecuta el siguiente procedimiento:

- *Paso 1:* El host envía un mensaje RS *(Router Solicitation)* a la red con destino FF02::2, la cual hace referencia a una dirección multicast reservada para la comunicación con routers. El propósito consiste en identificar aquellos pertenecientes a su misma subred y obtener sus direcciones, con el objetivo de aplicar una de ellas como puerta de enlace.

- *Paso 2:* Los routers responderán a la solicitud a través de un mensaje RA *(Router Advertisement)* que puede ser enviado de manera unicast al host que lo solicitó o mediante multicast a la IP FF02::1, la cual identifica a todos los dispositivos de la misma subred. Entre la información incluida se encuentra su dirección *Link-Local*. Además, estos mensajes también son enviados periódicamente a la red sin necesidad de haber recibido ningún RS.

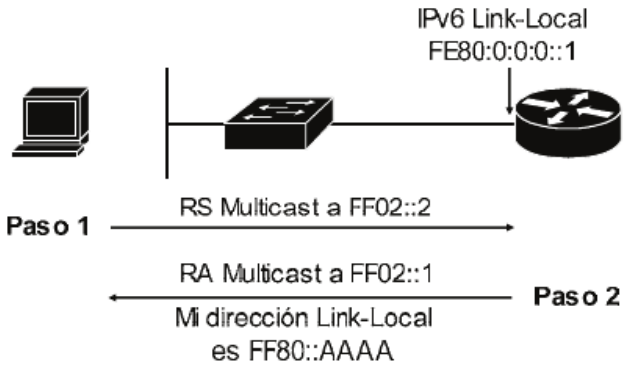


Fig. 11-6 NDP. Descubrimiento de routers.

DESCUBRIMIENTO DEL PREFIJO Y LONGITUD

Otro de los datos incluidos en los mensajes RA es el prefijo y su longitud. Con ello, el dispositivo ya ha obtenido de manera automática y sin necesidad de DHCP tanto la puerta de enlace como el prefijo de la red a la que pertenece. Este proceso es el utilizado por SLAAC durante el proceso de autoconfiguración.

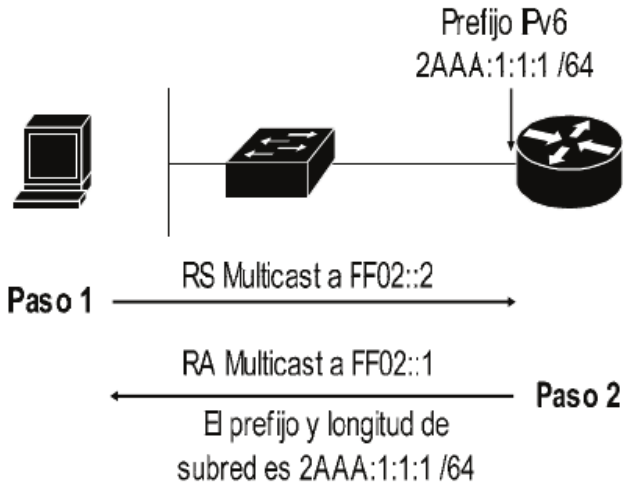


Fig. 11-7 NDP. Descubrimiento de prefijo y longitud.

DESCUBRIMIENTO DE DIRECCIONES MAC

La finalidad de los mensajes RS y RA consiste en recopilar información de la red en relación con los routers ubicados en la misma. Sin embargo, NDP también desarrolla la función de obtener datos directamente de los hosts vecinos, como la traducción de direcciones IP a MAC (ARP en IPv4). Para ello, ejecuta el siguiente procedimiento:

- *Paso 1:* El dispositivo envía un mensaje NS *(Neighbor Solicitation),* definiendo como destino la dirección multicast *solicited-node* del host al cual se requiere su MAC. Dicha dirección es calculada en base a su IP.

- *Paso 2:* Este será respondido mediante un mensaje NA *(Neighbor Advertisement)* unicast, el cual incluye la información solicitada.

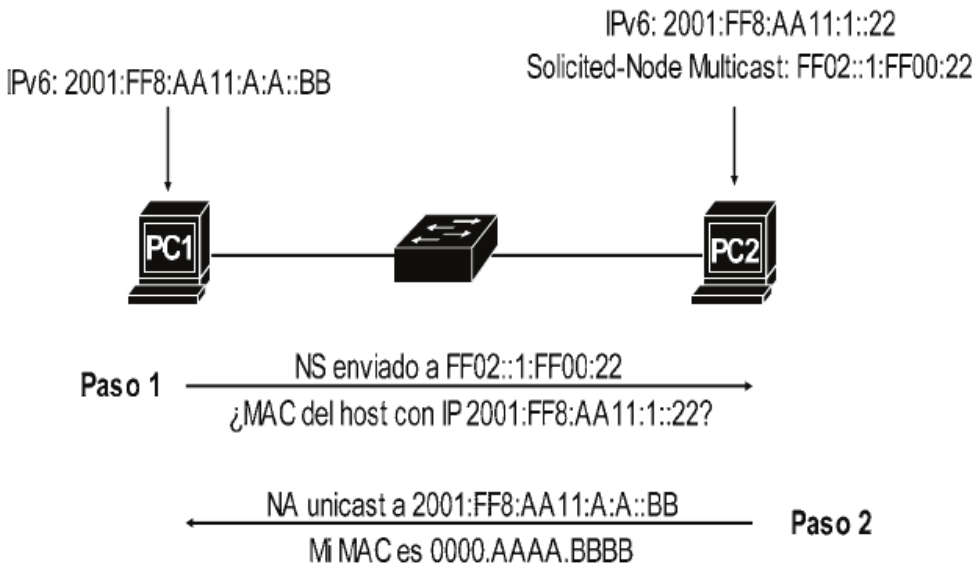


Fig. 11-8 NDP. Descubrimiento de direcciones MAC.

DETECCIÓN DE DIRECCIONES IP DUPLICADAS

Por último, NDP también es el protocolo encargado de detectar direcciones IP duplicadas dentro de la misma subred, utilizando para ello una función denominada DAD *(Duplicate Address Detection)*, la cual ejecuta las siguientes acciones:

Paso 1: El host, conforme a la información obtenida mediante los diferentes procesos calcula automáticamente una dirección IP para la subred a la que pertenece.

Paso 2: Antes de aplicarla sobre sí mismo debe verificar que no está siendo utilizada por ningún otro dispositivo.

Paso 3: Para ello, envía un mensaje NS a la red solicitando información sobre dicha IP.

Paso 4: Si algún otro miembro hace uso de la misma, responderá con un mensaje NA, por lo que el dispositivo deberá calcular otra y repetir el proceso hasta que se solucione el conflicto. De lo contrario, se autoasigna la dirección.

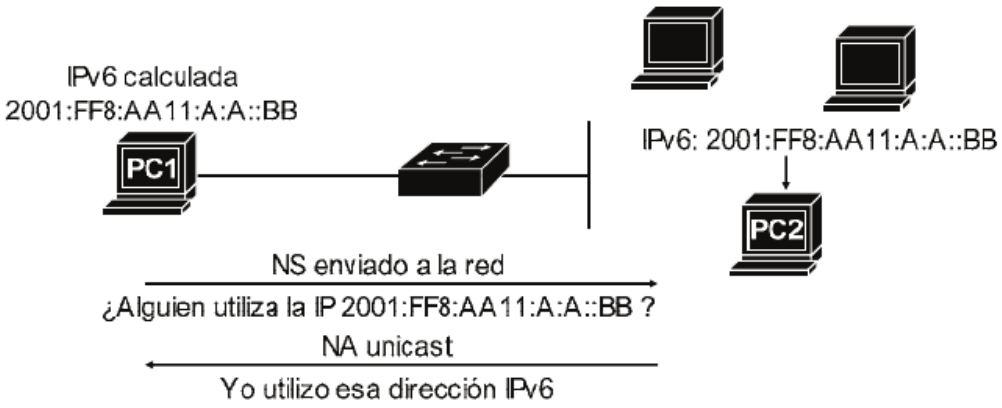


Fig. 11-9 NDP. Detección de direcciones IP duplicadas.

PC1 calcula automáticamente la IP 2001:FF8:AA11:A:A::BB. Antes de aplicarla sobre sí mismo, debe verificar que no está siendo utilizada por ningún otro miembro de la red, para lo cual envía un mensaje NS solicitando información sobre la misma. PC2 recibe el mensaje y comprueba que su IP coincide, por lo que informa a PC1 con un mensaje NA. En este caso, PC1 deberá calcular otra IP y volver a ejecutar el proceso hasta que su dirección sea única.

Para llevar a cabo sus funciones, NDP hace uso de mensajes ICMPv6. El método recién analizado también es ejecutado para direcciones *Link-Local*.

DHCPv6: Modo de operar

A lo largo del tiempo, el método aplicado por excelencia para que los hosts obtengan sus de datos de conexión de manera automática ha sido DHCP, debido en gran parte a su sencillez y administración centralizada, convirtiéndolo en la opción ideal sobre entornos corporativos. Este protocolo, al igual que muchos otros, ha sido actualizado para soportar las características de IPv6, desarrollando una nueva versión para tal propósito, DHCPv6.

A nivel de red, el modo de operar coincide con el llevado a cabo por su antecesor, es decir:

- El cliente envía un mensaje a la red en busca de algún servidor DHCPv6.

- El servidor responde a la solicitud ofreciendo datos de conexión.

- El cliente los analiza y acepta.

- El servidor envía la configuración necesaria.

- El cliente la aplica en el dispositivo.

Sin embargo, los mensajes utilizados durante la negociación entre ambos, aunque mantienen la misma finalidad, reciben diferente denominación. Mientras que en IPv4 estos son *DHCPDiscover, DHCPOffer, DHCPRequest* y *ACK,* en DHCPv6 son nombrados como *Solicit, Advertise, Request y Reply*.

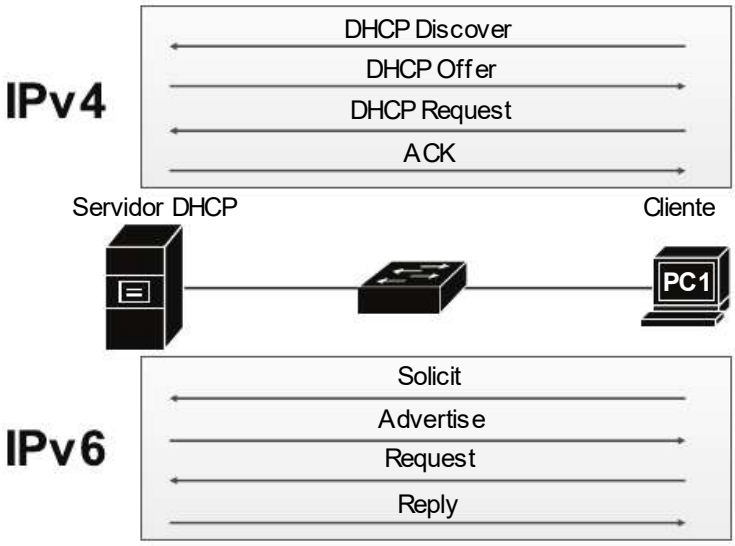


Fig. 11-10 Comparación entre DHCP en IPv4 e IPv6.

Además, la nueva versión permite su implementación con relación a dos modos, *"Stateful"* y "*Stateless"*, cada uno de ellos operando de la siguiente manera.

STATEFUL DHCPV6

Resulta muy similar al aplicado sobre redes IPv4, donde el servidor DHCP proporciona todos los datos de configuración necesarios al cliente y a su vez almacena información sobre cada uno de ellos. En DHCPv6, y gracias a NDP, se ha incluido una pequeña modificación, consistente en que los hosts deben obtener la puerta de enlace predeterminada mediante el intercambio de mensajes RS y RA.

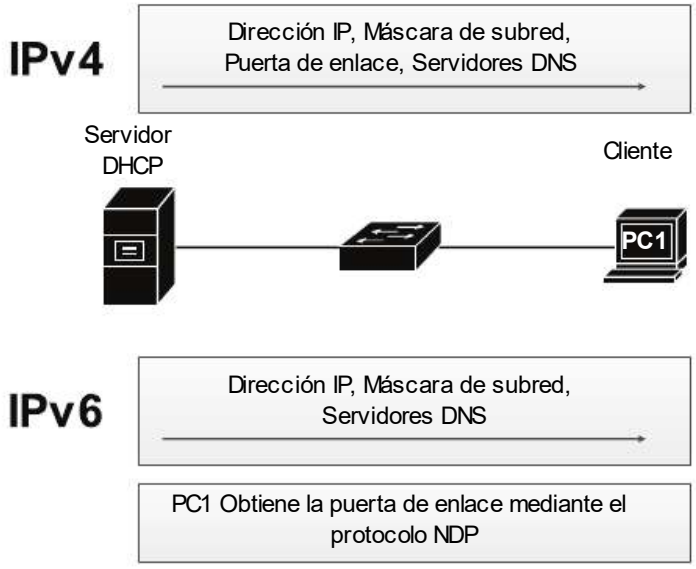


Fig. 11-11 Stateful DHCPv6.

STATELESS DHCPV6 Y SLAAC (STATELESS ADDRESS AUTO CONFIGURATION)

Uno de los mayores inconvenientes en DHCP reside en la cantidad de información que disponen los servidores sobre los clientes, convirtiéndolos en puntos sensibles de ataque. Para intentar solventarlo, DHCPv6 agrega el modo *Stateless*, el cual se apoya en el protocolo NDP para que el host pueda obtener sus datos de conexión. Gracias a ello, la única información que se debe proveer son los servidores DNS, por lo que no resulta necesario almacenar ningún tipo de datos sobre sus clientes.

Para ser más exactos, el procedimiento llevado a cabo en este caso consta de:

- *Paso 1:* El host detecta automáticamente el prefijo aplicado en la red gracias al intercambio de mensajes RS y RA.

- *Paso 2:* Del mismo modo localiza la puerta de enlace.

- *Paso 3:* Conforme al prefijo, calcula una dirección IP aplicando el método instaurado por cada fabricante, en el caso de Cisco, EUI-64.

- *Paso 4:* Antes de operar con la misma, ejecuta la función DAD de NDP para verificar que es única en la red.

- *Paso 5:* Por último, solicita al servidor DHCPv6 *Stateless* los servidores DNS. Este responderá proporcionando dicha información.

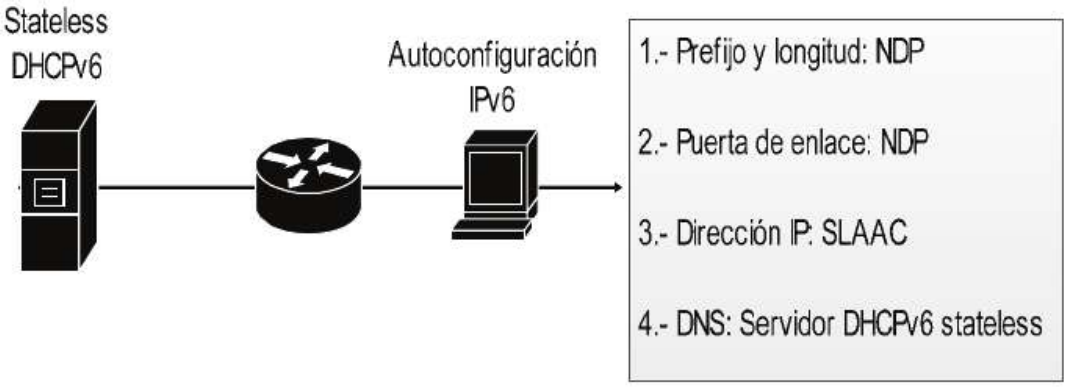


Fig. 11-12 Stateless DHCPv6.

Comparando ambos modos:

Característica	**Stateful DHCPv6**	**Stateless DHCPv6**
Almacena información de clientes	Sí	No
Proporciona arrendamiento de direcciones IP	Sí	No
Proporciona listado de servidores DNS	Sí	Sí
Comúnmente utilizado junto a SLAAC	No	Sí

DHCP RELAY

La primera acción que ejecuta un cliente para obtener una IP mediante DHCPv6 en modo *Stateful* consiste en enviar un mensaje *Solicit* a la red con destino FF02::1:2, la cual identifica a servidores y routers que actúan como agente *relay*. El problema reside en que estas no son enrutables, teniendo como consecuencia que, si el servidor fuera ubicado en una subred diferente a la del host, la comunicación no se llevará a cabo.

Para solucionarlo, al igual que en IPv4, bastará con configurar el router para que los mensajes *Solicit* sean reenviados a una dirección específica, aplicando el comando **ipv6 dhcp relay destination *[ip servidor DHCPv6]*** desde el modo de configuración de la interfaz necesaria.

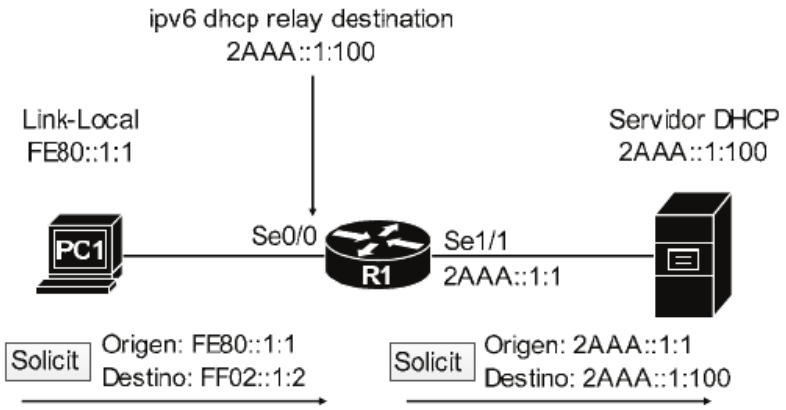


Fig. 11-13 DHCP Relay en IPv6.

```
R1(config)# interface Se0/0
R1(config-if)# ipv6 dhcp relay destination 2AAA::1:100
```

VERIFICACIÓN DE CONECTIVIDAD

Una vez el dispositivo ha obtenido sus datos de conexión podrá comunicarse en la red sin mayor problema. Aun así, como buena práctica resulta recomendable realizar las pruebas de conectividad necesarias con el fin de asegurar que la configuración obtenida es realmente válida.

Cada sistema operativo dispone de comandos propios para tal propósito, como **ipconfig /all** en Windows, o **ifconfig** en Linux. Ambos mostrarán en pantalla la información de red aplicada en el dispositivo, lo cual incluye su dirección IP, prefijo, servidores DNS, puerta de enlace, MAC, etc.

Tras ello, nada mejor que hacer uso de las utilidades ping y traceroute. Ping verifica la conectividad de extremo a extremo entre dos hosts, siendo ejecutado mediante el comando **ping *[dir IP destino]*** en sistemas Windows y **ping6 *[dir IP destino]*** en Linux. Mientras, traceroute testea los saltos realizados por un paquete desde el origen hasta el destino, siendo necesaria la sentencia **tracert *[dir IP destino]*** en Windows, y **traceroute6 *[dir IP destino]*** en Linux.

En IOS, el sistema operativo de Cisco ejecutará los comandos **ping** y **traceroute *[IP destino]*** para sendas funciones, con la peculiaridad que en ping se deberá introducir manualmente el protocolo IPv6.

ENRUTAMIENTO IPV6

Uno de los datos obtenidos por los hosts gracias a NDP es la puerta de enlace predeterminada, la cual hace referencia a una dirección *Link-Local* que identifica a un router y que será utilizada para enviar los paquetes cuyo destino en capa 3 corresponda a una subred diferente a la propia.

En estos casos la comunicación será recibida por el router en cuestión, el cual deberá disponer de la información necesaria para reenviarla hacia su destino. Dicho proceso, al igual que en IPv4, depende por completo de la tabla de rutas, que a su vez es gestionada y mantenida en base a los mismos métodos que su antecesor, mediante:

- Rutas directamente conectadas y locales.
- Rutas estáticas.
- Protocolos de enrutamiento dinámico.

Rutas directamente conectadas y locales

Como su nombre indica, este tipo de rutas hace referencia a aquellas redes directamente conectadas al dispositivo a través de alguna de sus interfaces, las cuales deben cumplir las siguientes condiciones:

- Disponer de una dirección IP perteneciente al mismo rango de red.

- Estar habilitada.

- Que se encuentre en estado "*up/up*".

Realmente, la tercera condición engloba las dos anteriores, por lo que se puede afirmar que una red directamente conectada será agregada automáticamente a la tabla de rutas cuando la interfaz necesaria para acceder a la misma se encuentra en estado "*up/up*".

Cuando ello ocurre el router genera dos nuevas entradas. La primera asocia el prefijo y longitud de la subred con la interfaz de salida, mientras que la segunda identifica la dirección IPv6 asignada en dicha interfaz.

En la siguiente topología, R1 no ha sido configurado, por lo que su tabla de rutas se encuentra vacía…

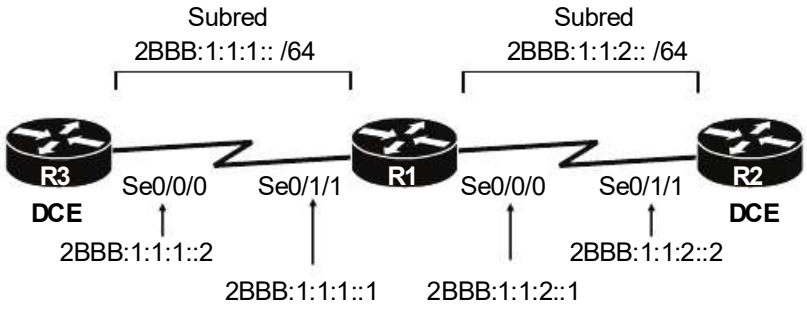


Fig. 11-14 DHCP Relay en IPv6.

El primer paso consiste en asignar las direcciones IP a las interfaces correspondientes, para acto seguido habilitarlas. De tal manera que:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int Se0/0/0
R1(config-if)#ipv6 address 2BBB:1:1:2::1/64
R1(config-if)# no shutdown
R1(config-if)#exit
R1(config)#int se0/1/1
R1(config-if)#ipv6 address 2BBB:1:1:1::1/64
R1(config-if)# no shutdown
```

Con ello, ambas deberían obtener el estado "*up/up*". Para verificarlo bastará con ejecutar el comando *show ipv6 interface brief.*

```
R1#show ipv6 interface brief
Serial0/0/0                [up/up]
    FE80::200:CFF:FE0E:1901
    2BBB:1:1:2::1
Serial0/1/1                [up/up]
    FE80::2E0:F9FF:FEB0:C202
    2BBB:1:1:1::1
```

Por lo que dichas redes serán agregadas a la tabla de rutas. El comando necesario para acceder a su contenido en IPv6 es *show ipv6 route.*

```
R1#show ipv6 route
IPv6 Routing Table - 5 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2BBB:1:1:1::/64 [0/0]
     via ::, Serial0/1/1
L   2BBB:1:1:1::1/128 [0/0]
     via ::, Serial0/1/1
```

```
C    2BBB:1:1:2::/64 [0/0]
     via ::, Serial0/0/0
L    2BBB:1:1:2::1/128 [0/0]
     via ::, Serial0/0/0
```

En ambos casos se han generado dos entradas, una con código C *(Connected)* que asocia el prefijo y longitud de la red en cuestión con la interfaz de salida necesaria para acceder a la misma, y otra con código L *(Local)* que hace referencia a la dirección IPv6 de dicha interfaz. En este caso la longitud siempre será igual a 128, ya que identifica únicamente una IP.

Por último, dos características de este tipo de rutas son:

- Son creadas y eliminadas automáticamente sin la intervención del administrador. Si una interfaz cae, la red con la cual conecta es borrada de la tabla hasta que se resuelva el problema.

- No hacen uso de direcciones *Link-local* para llevar a cabo el enrutamiento, a diferencia de otros métodos, analizados a continuación.

Rutas estáticas

Una ruta estática es aquella definida manualmente con el fin de asociar una red remota con la interfaz de salida necesaria para acceder a la misma. Una vez creada es agregada a la tabla de rutas, de tal manera que los paquetes que coincidan con su prefijo serán reenviados en relación con la configuración establecida, la cual podrá llevarse a cabo de dos maneras, directamente a través de la ya mencionada interfaz de salida, o mediante la IP de siguiente salto. El objetivo y la finalidad en ambos casos coinciden, aun así serán analizados por separado haciendo uso de la siguiente topología, donde R1 debe ser configurado para acceder a la red remota 2BBB:1:1:3::/64.

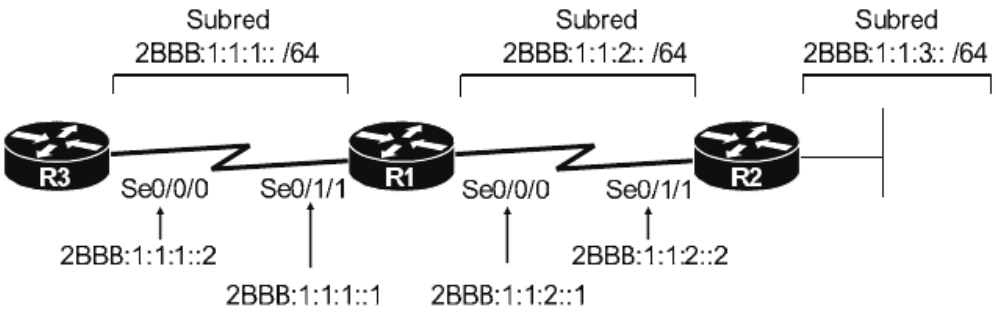


Fig. 11-15 Diseño de red para supuesto práctico de rutas estáticas.

RUTAS ESTÁTICAS CON INTERFAZ DE SALIDA

Como su nombre indica, consiste en asociar directamente una determinada subred con la interfaz de salida necesaria para acceder a la misma. Su configuración se lleva a cabo a través del comando **ipv6 route** ***[prefijo/longitud] [interfaz salida]*** desde el modo de configuración global, que aplicado sobre R1:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ipv6 route 2BBB:1:1:3::/64 Se0/0/0
```

Este hecho dará como resultado la creación de una nueva entrada en la tabla de rutas, identificada mediante el código S *(Static)*.

```
R1#show ipv6 route
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2BBB:1:1:1::/64 [0/0]
     via ::, Serial0/1/1
L   2BBB:1:1:1::1/128 [0/0]
     via ::, Serial0/1/1
C   2BBB:1:1:2::/64 [0/0]
     via ::, Serial0/0/0
L   2BBB:1:1:2::1/128 [0/0]
     via ::, Serial0/0/0
S   2BBB:1:1:3::/64 [1/0]
     via ::, Serial0/0/0
```

RUTAS ESTÁTICAS CON IP DE SIGUIENTE SALTO

El segundo método consiste en definir la IP de siguiente salto necesaria para acceder a la red remota deseada. Dicha dirección corresponde con aquella utilizada por el router al cual se debe reenviar el paquete, y que a su vez conecta directamente a través de alguna de las interfaces físicas. Por ejemplo, para la red 2BBB:1:1:3::/64, R1 accede a través de R2, con el cual comparte el mismo segmento de red, estableciendo su IP como siguiente salto.

Su configuración consta de la sentencia **ipv6 route** ***[prefijo/longitud] [IP de siguiente salto]*** desde el modo de configuración global, permitiendo una peculiaridad, y es que puede ser definida tanto la IP *global* o *unicast* como la dirección *Link-Local*.

Aplicado sobre R1 haciendo uso de la dirección global de R2:

```
R1#conf t
R1(config)#ipv6 route 2BBB:1:1:3::/64 2BBB:1:1:2::2

R1#show ipv6 route
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B – BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2BBB:1:1:1::/64 [0/0]
     via ::, Serial0/1/1
L   2BBB:1:1:1::1/128 [0/0]
     via ::, Serial0/1/1
C   2BBB:1:1:2::/64 [0/0]
     via ::, Serial0/0/0
L   2BBB:1:1:2::1/128 [0/0]
     via ::, Serial0/0/0
S   2BBB:1:1:3::/64 [1/0]
     via 2BBB:1:1:2::2
```

RUTAS ESTÁTICAS POR DEFECTO

Una ruta estática por defecto es aquella que el router utiliza para reenviar paquetes cuya red de destino no coincide con ninguna de las incluidas en la tabla de rutas. De no existir, dichos paquetes serían descartados.

En IPv6 su configuración también se lleva a cabo de manera manual, mediante el comando **ipv6 route ::/0** ***[interfaz de salida o IP de siguiente salto]*** desde el modo de configuración global. Si la opción deseada fuera el siguiente salto, podrá ser definida tanto la IP como la dirección *Link-Local* del router al cual debe ser reenviada la comunicación.

Imagina que la interfaz Fa0/3 de R2 conecta directamente con Internet, debiendo establecer una ruta por defecto para que todos los paquetes con destino desconocido sean reenviados a través de la misma.

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ipv6 route ::/0 Fa0/3
R2(config)#exit
R2# show ipv6 route
IPv6 Routing Table - 8 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
```

```
        D - EIGRP, EX - EIGRP external
S    ::/0 [1/0]
      via ::, FastEthernet0/3
C    2BBB:1:1:1::/64 [0/0]
      via ::, Serial0/1/1
L    2BBB:1:1:1::2/128 [0/0]
      via ::, Serial0/1/1
C    2BBB:1:1:2::/64 [0/0]
      via ::, Serial0/1/1
L    2BBB:1:1:2::2/128 [0/0]
      via ::, Serial0/1/1
C    2BBB:1:1:AAAA::/64 [0/0]
      via ::, FastEthernet0/3
L    2BBB:1:1:AAAA::1/128 [0/0]
      via ::, FastEthernet0/3
```

Las rutas estáticas pueden representar la solución ideal en topologías compuestas por varios routers y pocas subredes. Sin embargo, sobre entornos corporativos resulta un modelo muy complicado de mantener y nada escalable, siendo lo más recomendable, en estos casos, la aplicación de protocolos de enrutamiento dinámico.

Enrutamiento dinámico en IPv6

Los protocolos de enrutamiento dinámico son aquellos cuya misión consiste en aprender redes remotas de manera automática mediante el intercambio de información entre los diferentes dispositivos de capa 3 ubicados en la red, desarrollando también la función de completar y actualizar la tabla de rutas siempre que fuera necesario. Estos también han sido actualizados para su implementación sobre redes IPv6, dando lugar a las versiones RIPng, EIGRPv6 y OSPFv3, las cuales no dan soporte a IPv4, por lo que en entornos *dual stack* también resulta necesaria la configuración de sus antecesores. De todos ellos, EIGRPv6 y OSPFv3 representan las opciones más comunes, formando parte del contenido de CCNA.

Antes de continuar se recomienda volver a leer el modo de operar y configuración de EIGRP y OSPF, incluidos en el capítulo 6 "*Protocolos de enrutamiento".*

EIGRPV6. CONFIGURACIÓN Y VERIFICACIÓN

EIGRPv6 es la versión del protocolo adaptada para su aplicación exclusiva sobre entornos IPv6. Su modo de operar coincide totalmente con el ya analizado en el capítulo 6 "*Protocolos de enrutamiento*", exceptuando los siguientes detalles:

EIGRP publica el ID de red con su correspondiente máscara, mientras que EIGRPv6 publica el prefijo con su longitud.

- En EIGRP, la sumarización automática se aplica manualmente con el comando *auto-summary,* mientras que en EIGRPv6 no resulta necesaria ninguna configuración para tal propósito.

- En EIGRP, uno de los requisitos para que dos routers establezcan adyacencia es que ambos pertenezcan a la misma subred. En EIGRPv6 dicha condición no resulta necesaria.

Respecto a su configuración, el proceso necesario para llevarla a cabo consta de:

- *Paso 1:* Crear una instancia del protocolo a través del comando **ipv6 router eigrp *[número de ASN]***, desde el modo de configuración global. Al igual que sucede con su antecesor, uno de los requisitos para que dos dispositivos establezcan adyacencia es que formen parte del mismo sistema autónomo (ASN). Una vez ejecutado se accede al modo de configuración propio del protocolo, desde el cual se deberán llevar a cabo la totalidad de acciones restantes a excepción del paso 3.

- *Paso 2:* Habilitar EIGRPv6 mediante la sentencia **no shutdown.**

- *Paso 3:* Especificar las interfaces que participarán activamente en el intercambio de rutas, ejecutando para ello el comando **ipv6 eigrp *[número de ASN]*** desde el modo de configuración de cada interfaz. Dicho detalle supone la mayor diferencia en comparación con EIGRP, donde el mismo proceso se lleva a cabo indirectamente desde la configuración del propio protocolo.

- *Paso 4 (Opcional):* Definir el RID mediante la sentencia **eigrp router-id [*x.x.x.x*]**. Este tiene una longitud de 32 bits, haciendo uso del mismo formato que una dirección IPv4.

- *Paso 5 (Opcional):* Identificar las interfaces pasivas con el comando **passive-interface *[interfaz]***.

- *Paso 6 (Opcional):* Configurar el número máximo de rutas para ejecutar balanceo de carga con el comando **maximum-paths [*num*]**.

- *Paso 7 (Opcional):* Configurar el balanceo de carga para rutas con métricas desiguales mediante la sentencia **variance [*x*],** donde *x* indica el valor a multiplicar sobre la menor métrica.

Si los pasos opcionales no fueran configurados el router aplicará los valores por defecto para cada caso, exceptuando el RID, que será definido llevando a cabo el mismo criterio que la versión anterior del protocolo, es decir, conforme a las

direcciones IPv4 de las interfaces loopback o físicas. Sin embargo, es muy probable que en IPv6 el router no disponga de dichas direcciones, teniendo como consecuencia que el protocolo no opere correctamente, no se establezcan relaciones con vecinos o no se intercambien rutas. Es por ello que, aunque resulte un parámetro opcional, se recomienda su aplicación de manera manual.

Ejemplo. Configurar EIGRPv6 en la siguiente topología de tal manera que:
- Todos los routers deben pertenecer al ASN número 5, haciendo uso de los siguientes RID: 1.1.1.1 (R1), 2.2.2.2 (R2) y 3.3.3.3 (R3).
- Todas las interfaces activas formen parte del protocolo.
- Configurar balanceo de carga con un máximo de 2 rutas y un *variance* de 3.
- Definir como interfaces pasivas aquellas que conecten con dispositivos finales.

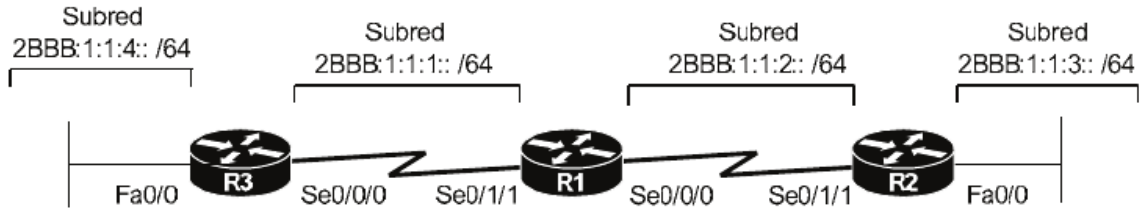


Fig. 11-16 Supuesto práctico para protocolos de enrutamiento.

```
---Configuración en R1---
R1(config)#ipv6 router eigrp 5
R1(config-rtr)#eigrp router-id 1.1.1.1
R1(config-rtr)#variance 3
R1(config-rtr)#maximum-paths 2
R1(config-rtr)#no shutdown
R1(config-rtr)#exit
R1(config)# interface Se0/0/0
R1(config-if)#ipv6 eigrp 5
R1(config-if)#exit
R1(config)# interface Se0/1/1
R1(config-if)#ipv6 eigrp 5

---Configuración en R2---
R2(config)#ipv6 router eigrp 5
R2(config-rtr)#eigrp router-id 2.2.2.2
R2(config-rtr)#variance 3
R2(config-rtr)#maximum-paths 2
R2(config-rtr)#passive-interface Fa0/0
R2(config-rtr)#no shutdown
R2(config-rtr)#exit
R2(config)#interface Fa0/0
R2(config-if)#ipv6 eigrp 5
R2(config-if)#exit
R2(config)# interface Se0/1/1
R2(config-if)#ipv6 eigrp 5
```

```
---Configuración en R3---
R3(config)#ipv6 router eigrp 5
R3(config-rtr)#eigrp router-id 3.3.3.3
R3(config-rtr)#passive-interface Fa0/0
R3(config-rtr)#variance 3
R3(config-rtr)#maximum-paths 2
R3(config-rtr)#no shutdown
R3(config-rtr)#exit
R3(config)#interface Fa0/0
R3(config-if)#ipv6 eigrp 5
R3(config-if)#exit
R3(config)# interface Se0/0/0
R3(config-if)#ipv6 eigrp 5
R3(config-if)#exit
```

La configuración aplicada dará como resultado la adyacencia e intercambio de mensajes EIGRPv6 entre los diferentes routers, logrando el aprendizaje de redes remotas que serán agregadas a la tabla de rutas de manera automática e identificadas con el código D (*Eigrp),* de tal manera que ejecutando un *show ipv6 route* en cualquiera de los dispositivos se podrán visualizar aquellas generadas por el protocolo.

Ejecutado sobre R1…

```
R1#show ipv6 route
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2BBB:1:1:1::/64 [0/0]
     via ::, Serial0/1/1
L   2BBB:1:1:1::1/128 [0/0]
     via ::, Serial0/1/1
C   2BBB:1:1:2::/64 [0/0]
     via ::, Serial0/0/0
L   2BBB:1:1:2::1/128 [0/0]
     via ::, Serial0/0/0
D   2BBB:1:1:3::/64 [90/20514560]
     via FE80::2D0:BCFF:FEB4:6989, Serial0/0/0
D   2BBB:1:1:4::/64 [90/20514560]
     via FE80::230:F2FF:FE93:8501, Serial0/1/1
```

En relación con los datos obtenidos se puede concluir que EIGRPv6 hace uso de direcciones *Link-Local* para definir el siguiente salto hacia aquellas redes aprendidas por el protocolo.

Otros comandos de verificación disponibles en IOS son:

- *show ipv6 eigrp interfaces:* Al igual que en IPv4, muestra información en pantalla sobre cada interfaz que forma parte del protocolo.

- *show ipv6 eigrp topology:* Genera un listado con todas las rutas aprendidas por EIGRPv6, incluyendo datos como su distancia factible, distancia reportada etc.

- *show ipv6 eigrp neighbors:* Facilita información sobre cada vecino con el cual se mantenga adyacencia.

- *show ipv6 route eigrp:* Muestra en pantalla aquellas redes remotas agregadas a la tabla de rutas por el protocolo EIGRPv6, incluyendo su prefijo y longitud, siguiente salto, métrica y distancia administrativa.

OSPFV3. CONFIGURACIÓN Y VERIFICACIÓN

Al igual que ocurre con EIGRP, OSPF también ha requerido el desarrollo de una nueva versión que permita su aplicación sobre entornos IPv6, siendo esta OSPFv3. Su modo de operar y finalidad coincide con la de su antecesor, ya analizada en el capítulo 6 "*Protocolos de enrutamiento*", a excepción de los siguientes detalles:

- Ambos hacen uso de mensajes LSA para el intercambio de información entre vecinos, sin embargo, OSPFv3 modifica la estructura de algunos de ellos.

- OSPFv2 publica el ID de red IPv4 con su correspondiente máscara, mientras que la nueva versión publica el prefijo con su longitud.

- OSPFv3 incluye un nuevo método de autenticación basado en el protocolo IPSec.

En cuanto a su configuración, la mayor diferencia entre ambos radica en cómo identificar las redes a publicar. Mientras que en OSPFv2 dicho proceso se efectúa desde el modo de configuración propio del protocolo, en OSPFv3 se lleva a cabo directamente desde la propia interfaz. El procedimiento al completo consta de las siguientes acciones:

- *Paso 1:* Crear una instancia de OSPFv3 con el comando **ipv6 router ospf *[num proceso]*** desde el modo de configuración global, donde el valor definido tan solo tendrá importancia a nivel local, ya que la adyacencia entre vecinos dependerá del número de área a la que pertenecen. Tras su ejecución se accede al modo de configuración del protocolo, desde el cual se deberán aplicar los pasos 3 y 4.

- *Paso 2:* Habilitar OSPFv3 en aquellas interfaces que participarán activamente en el intercambio de rutas, con el comando **ipv6 ospf *[num proceso]* area *[num área]*** desde el modo de configuración de la interfaz. Uno de los requisitos necesarios para que dos routers establezcan adyacencia es que coincidan en el número de área.

- *Paso 3 (Opcional):* Definir un RID mediante la sentencia **router id *[id]***, con longitud de 32 bits e idéntico formato que una dirección IPv4.

- *Paso 4 (Opcional):* Identificar las interfaces pasivas con el comando **passive-interface *[interfaz]***.

Al igual que sucede en EIGRPv6, si el RID no fuera configurado el router buscará direcciones IPv4 en sus interfaces loopback o físicas con el fin de asignar un valor de manera automática. Sin embargo, si el dispositivo no dispone de ellas el protocolo dará problemas, mostrando el siguiente mensaje en pantalla:

```
%OSPFv3-4-NORTRID: OSPFv3 process 1 could not pick a router-id, please configure
manually
```

Por lo que resulta muy recomendable su aplicación de manera manual.

Ejemplo: En relación con la topología mostrada en la *Fig. 11-16*, configurar la siguiente topología de tal manera que:
- Las interfaces de todos los dispositivos pertenezcan al área 0 de OSPFv3.
- Cada router haga uso del siguiente RID: 1.1.1.1 (R1), 2.2.2.2 (R2) y 3.3.3.3 (R3).
- Las interfaces que conectan con dispositivos finales formen parten de OSPFv3 pero no participen en el intercambio de rutas ni establezcan adyacencias.

```
---Configuración en R1---
R1(config)# ipv6 router ospf 1
R1(config-rtr)# router-id 1.1.1.1
R1(config-rtr)#exit
R1(config)# int Se0/0/0
R1(config-if)# ipv6 ospf 1 area 0
R1(config-if)# exit
R1(config)# int Se0/1/1
R1(config-if)# ipv6 ospf 1 area 0

---Configuración en R2---
R2(config)# ipv6 router ospf 2
R2(config-rtr)# router-id 2.2.2.2
R2(config-rtr)#passive-interface fa0/0
R2(config-rtr)#exit
R2(config)# int Se0/1/1
R2(config-if)# ipv6 ospf 2 area 0
R2(config-if)# exit
R2(config)# int Fa0/0
R2(config-if)# ipv6 ospf 2 area 0

---Configuración en R3---
R3(config)# ipv6 router ospf 3
R3(config-rtr)# router-id 3.3.3.3
R3(config-rtr)#passive-interface fa0/0
R3(config-rtr)#exit
R3(config)# int Se0/0/0
```

```
R3(config-if)# ipv6 ospf 3 area 0
R3(config-if)# exit
R3(config)# int Fa0/0
R3(config-if)# ipv6 ospf 3 area 0
```

La configuración aplicada dará como resultado el intercambio de rutas de manera automática, siendo agregadas en la tabla mediante el código O *(OSPF).* Para comprobarlo, al igual que en ejemplos anteriores, bastará con ejecutar el comando *show ipv6 route,* obteniendo el siguiente resultado en R1.

```
R1#show ipv6 route
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2BBB:1:1:1::/64 [0/0]
     via ::, Serial0/1/1
L   2BBB:1:1:1::1/128 [0/0]
     via ::, Serial0/1/1
C   2BBB:1:1:2::/64 [0/0]
     via ::, Serial0/0/0
L   2BBB:1:1:2::1/128 [0/0]
     via ::, Serial0/0/0
O   2BBB:1:1:3::/64 [110/65]
     via FE80::2D0:58FF:FE22:668C, Serial0/0/0
O   2BBB:1:1:4::/64 [110/65]
     via FE80::201:C9FF:FE58:7011, Serial0/1/1
```

Como se puede observar, OSPFv3 también hace uso de direcciones *Link-Local* para definir el siguiente salto hacia una red remota.

Otros comandos útiles de verificación son:

- *show ipv6 ospf neighbor:* Facilita datos sobre vecinos con los cuales se ha establecido y mantiene adyacencia. Estos son identificados por su RID, aunque también se muestra su dirección IPv6 y la interfaz local necesaria para conectar con cada uno de ellos.

- *show ipv6 ospf interface [interfaz]:* Muestra información referente al protocolo sobre una determinada interfaz, incluyendo su IP, área, RID, tipo de enlace, coste, valores *hello* y *dead*, contadores, etc.

- *show ipv6 route ospf:* Muestra las redes remotas que han sido agregadas a la tabla de rutas por OSPFv3. Estas son representadas mediante el código *"O"*, facilitando información como su prefijo y longitud, métrica, siguiente salto, etc.

Reto 11.1 – Dada la siguiente topología, configurar los routers de tal manera que:

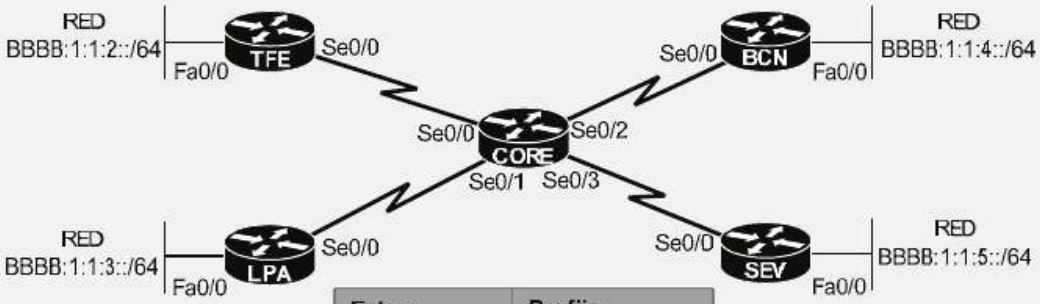
- Todas las interfaces deben obtener su IP de manera automática mediante EUI-64.

- TFE, LPA y CORE formen parte del ASN 10 de EIGRPv6, haciendo uso de los siguientes RID: TFE (3.3.3.3), LPA (2.2.2.2) y CORE (1.1.1.1).

- BCN, SEV y CORE pertenezcan al área 0 de OSPFv3, con RIDs 3.3.3.3 (BCN), 2.2.2.2 (SEV) y 1.1.1.1 (CORE).

- En ambos protocolos se deben definir como interfaces pasivas aquellas que conecten con dispositivos finales.

- CORE actúe como DCE en todos los enlaces seriales, con una velocidad de 64000 bps.



Enlace	Prefijo
CORE - TFE	AAAA:1:1:2::/64
CORE - LPA	AAAA:1:1:3::/64
CORE - BCN	AAAA:1:1:4::/64
CORE - SEV	AAAA:1:1:5::/64

Solución al final del capítulo.

SEGURIDAD IPV6: LISTAS DE CONTROL DE ACCESO

En cuanto a seguridad en capa 3 se refiere, las ACLs representan el método por excelencia disponible en dispositivos intermediarios de red, más concretamente, en routers. Su modo de operar y configuración en IPv4 forman parte del capítulo 7 "*Seguridad en capa 3*", sin embargo, con la aparición de IPv6 ha sido necesaria su adaptación al nuevo protocolo, incluyendo nuevas características y funcionalidades. La presente sección será dedicada a ello, abordando las diferencias más destacables y su configuración tanto en modo estándar como extendido.

En este caso, el objetivo y la finalidad coinciden sin importar el protocolo IP sobre el cual sean aplicadas, es decir, permitir el filtrado de paquetes conforme a direcciones de origen, destino, o puertos, entre otros. Sin embargo, algunas características varían entre IPv4 e IPv6, entre las que se encuentran:

- Mientras que en IPv4 su identificación puede llevarse a cabo con relación a un número o nombre, en IPv6 tan solo es posible su definición por nombre.

- IPv6 incluye 3 sentencias implícitas al final de cada ACL, mientras que IPv4 tan solo una (*deny any any*).

- El formato de paquete definido en ambos protocolos incluye campos propios y únicos para cada uno de ellos, por ejemplo, *TTL* de IPv4 no existe en IPv6, siendo sustituido por *Next Hop*, cuya función resulta similar. Por lo tanto, algunos de los filtros disponibles en relación con campos del paquete no coinciden en las ACLs de IPv4 e IPv6.

- La creación de una ACL en IPv4 requiere su identificación como estándar o extendida, ya sea mediante la asignación numérica del rango disponible para cada una de ellas, o mediante los parámetros *standard* o *extended* en aquellas nombradas. En IPv6 dicho concepto desaparece, no siendo necesario ningún tipo de identificación para tal propósito.

- Mientras que en IPv4 resulta necesario definir la dirección *wildcard* de la red o hosts a filtrar, IPv6 tan solo requiere el prefijo de estos.

- Las posibilidades de filtrado disponibles en ACLs extendidas de IPv6 permiten un mayor abanico de opciones que aquellas en IPv4.

Evidentemente, una ACL IPv6 no detiene paquetes IPv4, ni viceversa. Este hecho resulta importante tenerlo en cuenta sobre entornos *dual stack,* donde se hace necesario configurarlas y aplicarlas para ambos protocolos en aquellas interfaces que operen con los mismos de manera simultánea.

En cuanto a la dirección e interfaces donde pueden ser aplicadas, ambos tipos de ACLs coinciden, siendo las direcciones disponibles de entrada y/o salida, permitiendo como máximo una ACL por dirección y siendo aplicables en cualquier interfaz del dispositivo, incluso aquellas virtuales o destinadas a gestión remota, como las líneas VTY.

REGLAS IMPLÍCITAS EN ACLS IPV6

Una de las mayores diferencias radica en las reglas implícitas pertenecientes en IPv4 e IPv6. Mientras que el primero tan solo incluye un "*deny any any*" como última sentencia, IPv6 agrega 3, definidas en el siguiente orden:

```
permit icmp any any nd-na
permit icmp any any nd-ns
deny any any
```

¿Por qué resulta necesario permitir paquetes ICMP por defecto? ICMPv6 alberga numerosas utilidades, influyendo de manera directa sobre operaciones básicas del protocolo IPv6. Una de estas operaciones consiste en el descubrimiento de vecinos, ejecutado por NDP y gracias al cual los hosts pueden obtener de manera automática información de la red a la que pertenecen, como su id, prefijo, o el cálculo de una dirección IP sin necesidad de DHCP. Para lograrlo, NDP hace uso de los paquetes *icmp nd-na* e *icmp nd-ns*. Es por ello que dichos mensajes deben ser permitidos de manera implícita en IPv6.

ACL IPV6 ESTÁNDAR

Una ACL estándar en IPv4 es aquella que tan solo basa su filtrado en la dirección de origen del paquete, sin necesidad de especificar ningún otro tipo de parámetro. Este concepto cambia radicalmente en IPv6, donde, primero, no resulta necesario identificar la ACL como estándar, y segundo, puede ser definido tanto el origen como el destino de la comunicación. En este caso, el comando necesario para cada una de las sentencias es **[permit | deny] ipv6 [*origen*] [*destino*]**, donde:

- *[permit | deny]* indica la acción a ejecutar ante paquetes que cumplan todas las condiciones definidas en la sentencia.

- *ipv6* identifica el protocolo sobre el cual se llevará a cabo la acción. IPv6 en este caso.

- *[origen]* hace referencia al origen de la comunicación, pudiendo definir 3 valores:
  - *any*, para cualquier dirección de origen.
  - *host [ip/prefijo],* para un dispositivo en concreto.
  - *[red/prefijo],* cuando el origen representa una red o conjunto de hosts.

- *[destino]*, hace referencia al destino de la comunicación, permitiendo las mismas opciones, es decir:
  - *any*, para cualquier dirección.

- *host [ip/prefijo],* para un dispositivo en concreto.
- *[red/prefijo],* cuando se requiere identificar una red o conjunto de hosts.

Ejemplo: Definir una sentencia que permita la comunicación de la subred FD00:A:A:1::/64 hacia cualquier destino**.**

```
permit ipv6 FD00:A:A:1::/64 any
```

Con ello, el proceso de configuración completo de una ACL estándar consta de las siguientes acciones:

- *Paso 1:* Crear la ACL con el comando **ipv6 access-list *[nombre]***, desde el modo de configuración global.

- *Paso 2:* Definir cada uno de los filtros necesarios a través del comando recién analizado.

- *Paso 3:* Aplicarla en la interfaz y dirección deseada, ejecutando para ello la sentencia **ipv6 traffic-filter *[nombre ACL] [in | out]***, desde el modo de configuración de la interfaz en cuestión, siendo *[nombre ACL]* aquel definido en el paso 1 e *[in | out]* la dirección de análisis de tráfico deseada.

Ejemplo: Aplicar una ACL en R1 que permita la comunicación de todos los hosts pertenecientes a la subred FD00:A:A:1::/64 hacia los destinos FD00:A:A:2::/64 y FD00:A:A:A::10/64.

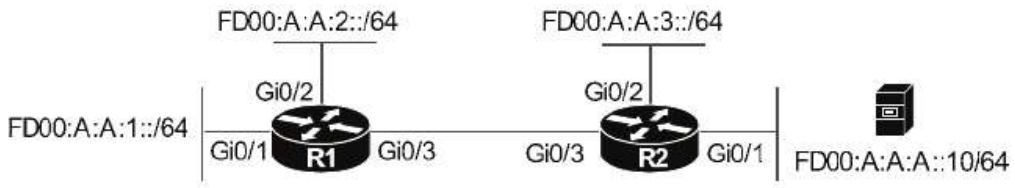


Fig. 11-17 Diseño de red para supuestos prácticos de ACLs.

```
R1(config)#ipv6 access-list PERMITIR-ACCESO
R1(config-ipv6-acl)#permit ipv6 FD00:A:A:1::/64 FD00:A:A:2::/64
R1(config-ipv6-acl)#permit ipv6 FD00:A:A:1::/64 host FD00:A:A:A::10/64
R1(config-ipv6-acl)#exit

R1(config)# interface Gi0/1
R1(config-if)#ipv6 traffic-filter PERMITIR-ACCESO in
```

ACL IPV6 EXTENDIDA

En IPv6, la cabecera de un paquete contiene numerosos campos inexistentes en IPv4. Ello da como resultado que las ACLs extendidas dispongan de multitud de opciones, lo cual permite una definición más exacta del filtrado, pero a su vez implica mayor complejidad en cuanto a configuración se refiere. Algunas de dichas opciones comprenden los puertos de origen y/o destino, protocolo, campo *dscp, flow-label*, secuencia, tipos de mensaje icmp, rango de tiempo o fragmentos, entre muchos otros. Sin embargo, para afrontar la certificación CCNA bastará con llevar a cabo el filtrado en base al protocolo, puerto y direcciones de origen y destino, asemejándose en gran medida a las ACL IPv4 ya analizadas. En este caso, el comando necesario para definir una sentencia es **[permit | deny] [protocolo] [*origen*] *[puerto o protocolo de origen]* [*destino*]] *[puerto o protocolo de destino],*** donde:

- *[permit | deny]* indica la acción a ejecutar ante paquetes que cumplan todas las condiciones definidas en la sentencia.

- *[protocolo]* identifica el protocolo sobre el cual se llevará a cabo la acción, pudiendo ser ipv6, udp, tcp o icmp, entre otros.

- *[origen]* hace referencia al origen de la comunicación, permitiendo definir 3 valores:
  - *any*, para cualquier dirección de origen.
  - *host [ip/prefijo],* para un dispositivo en concreto.
  - *[red/prefijo],* cuando el origen representa una red o conjunto de hosts.

- *[puerto o protocolo de origen]* especifica el número de puerto de origen o nombre del servicio asociado al mismo, permitiendo los siguientes parámetros:
  - *eq [número de puerto o nombre],* para un puerto igual al indicado.
  - *gt [número de puerto],* para números de puerto igual o mayor al definido.
  - *lt [número de puerto],* filtra los puertos con un valor igual o menor al indicado.
  - *range [rango de puertos],* establece un rango de puertos.

- *[destino]*, hace referencia al destino de la comunicación, permitiendo las mismas opciones que para el origen, es decir:
  - *any*, para cualquier dirección.
  - *host [ip/prefijo],* para un dispositivo en concreto.
  - *[red/prefijo],* cuando se requiere identificar una red o conjunto de hosts.

- *[puerto o protocolo de destino]* especifica el número de puerto de destino o nombre del servicio asociado al mismo, permitiendo los mismos parámetros que en el origen:

- *eq [número de puerto o nombre],* para un puerto igual al indicado.
- *gt [número de puerto],* para números de puerto igual o mayor al definido.
- *lt [número de puerto],* filtra los puertos con un valor igual o menor al indicado.
- *range [rango de puertos],* establece un rango de puertos.

Ejemplo: Definir una sentencia que permita la comunicación tcp de la subred FD00:A:A:1:: /64 a través del puerto 22 hacia cualquier destino**.**

```
permit tcp FD00:A:A:1::/64 eq 22 any
```

Su configuración al completo se lleva a cabo a través de las siguientes acciones:

- *Paso 1:* Crear la ACL con el comando **ipv6 access-list *[nombre]***, desde el modo de configuración global.

- *Paso 2:* Definir cada uno de los filtros necesarios a través del comando recién analizado.

- *Paso 3:* Aplicarla en la interfaz y dirección deseada, ejecutando para ello la sentencia **ipv6 traffic-filter *[nombre ACL] [in | out]***, desde el modo de configuración de la interfaz en cuestión, siendo *[nombre ACL]* aquel definido en el paso 1 e *[in | out]* la dirección de análisis de tráfico deseada.

Como se puede observar, su configuración y aplicación coincide con el procedimiento llevado a cabo para una estándar. Ello es debido a que IPv6 identifica ambas en relación con las sentencias incluidas, las cuales sí presentan diferencias.

Ejemplo: Conforme a la topología presente en la *Fig. 11-17*, aplicar una ACL en R1 cumpliendo los siguientes requisitos:
- Permitir la comunicación TCP mediante el puerto 5555 a todos los hosts pertenecientes a la subred FD00:A:A:1::/64 con destino FD00:A:A:2::/64.
- Permitir la comunicación web de todos los dispositivos pertenecientes a la subred FD00:A:A:1::/64, hacia el puerto 80 del host de destino FD00:A:A:A::10/64.

```
R1(config)#ipv6 access-list ACL-EXT
R1(config-ipv6-acl)#permit tcp FD00:A:A:1::/64 eq 55555 FD00:A:A:2::/64
R1(config-ipv6-acl)#permit tcp FD00:A:A:1::/64 host FD00:A:A:A::10/64 eq 80
R1(config-ipv6-acl)#exit

R1(config)# interface Gi0/1
R1(config-if)#ipv6 traffic-filter ACL-EXT in
```

Por último, IOS dispone el comando *show ipv6 access-list*, el cual muestra en pantalla un listado de las ACLs creadas en el dispositivo y las sentencias que componen cada una de ellas.

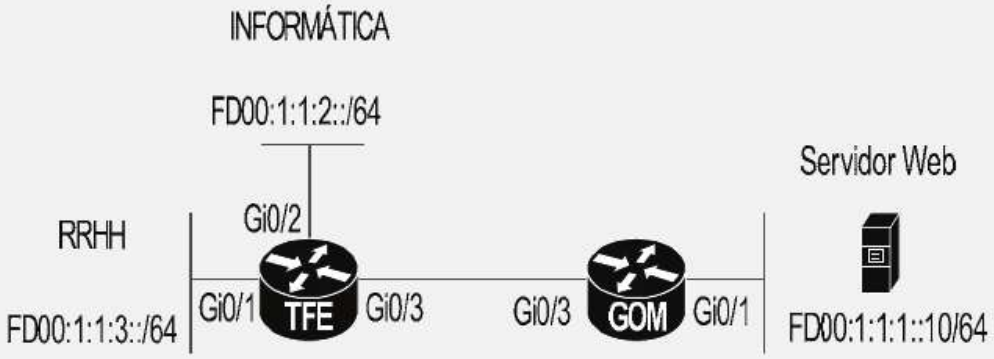
Reto 11.2 – Conforme a la siguiente topología, configurar las ACLs necesarias para cumplir los siguientes objetivos:

- Todos los dispositivos ubicados en las subredes FD00:1:1:2::/64 y FD00:1:1:3::/64 deben acceder al servidor web con dirección FD00:1:1:1::10/64 a través del puerto TCP 80.

- Además, los hosts ubicados en la subred de informática también deben tener acceso vía SSH al mismo servidor.

- Permitir cualquier tipo de tráfico entre las subredes de Informática y RRHH.

- Bloquear cualquier otro tipo de comunicación.



Solución al final del capítulo

SOLUCIÓN DE RETOS: IP VERSIÓN 6

Reto 11.1 – Dada la siguiente topología, configurar los routers de tal manera que:

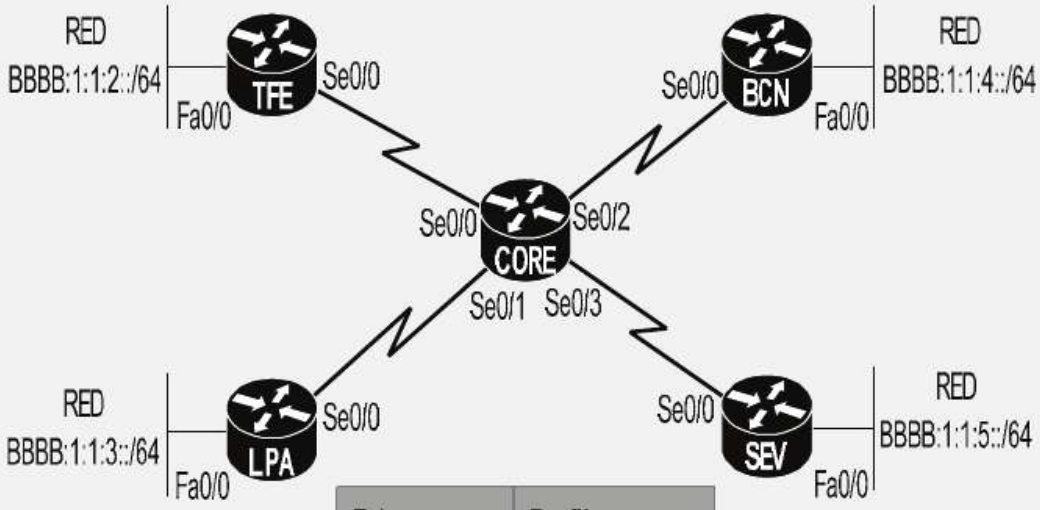
- Todas las interfaces deben obtener su IP de manera automática mediante EUI-64.

- TFE, LPA y CORE formen parte del ASN 10 de EIGRPv6, haciendo uso de los siguientes RID: TFE (3.3.3.3), LPA (2.2.2.2) y CORE (1.1.1.1).

- BCN, SEV y CORE pertenezcan al área 0 de OSPFv3, con RIDs 3.3.3.3 (BCN), 2.2.2.2 (SEV) y 1.1.1.1 (CORE).

- En ambos protocolos se deben definir como interfaces pasivas aquellas que conecten con dispositivos finales.

- CORE actúe como DCE en todos los enlaces seriales, con una velocidad de 64000 bps.



Enlace	Prefijo
CORE-TFE	AAAA:1:1:2::/64
CORE-LPA	AAAA:1:1:3::/64
CORE-BCN	AAAA:1:1:4::/64
CORE-SEV	AAAA:1:1:5::/64

```
---Configuración CORE---
CORE#conf t
CORE(config)#ipv6 unicast-routing
CORE(config)#ipv6 router eigrp 10
CORE(config-rtr)#router-id 1.1.1.1
CORE(config-rtr)#no shutdown
CORE(config-rtr)#exit

CORE(config)#ipv6 router ospf 1
CORE(config-rtr)#router-id 1.1.1.1
CORE(config-rtr)#exit

CORE(config)#int se0/0
CORE(config-if)#ipv6 address AAAA:1:1:2::/64 eui-64
```

```
CORE(config-if)#clock rate 64000
CORE(config-if)#ipv6 eigrp 10
CORE(config-if)#no shutdown
CORE(config-if)#exit
CORE(config)#int se0/1
CORE(config-if)#ipv6 address AAAA:1:1:3::/64 eui-64
CORE(config-if)#clock rate 64000
CORE(config-if)#ipv6 eigrp 10
CORE(config-if)#no shutdown
CORE(config-if)#exit
CORE(config)#int se0/2
CORE(config-if)#ipv6 address AAAA:1:1:4::/64 eui-64
CORE(config-if)#clock rate 64000
CORE(config-if)#ipv6 ospf 1 area 0
CORE(config-if)#no shutdown
CORE(config-if)#exit
CORE(config)#int se0/3
CORE(config-if)#ipv6 address AAAA:1:1:5::/64 eui-64
CORE(config-if)#clock rate 64000
CORE(config-if)#ipv6 ospf 1 area 0
CORE(config-if)#no shutdown

---Configuración TFE---
TFE#conf t
TFE(config)#ipv6 unicast-routing
TFE(config)#ipv6 router eigrp 10
TFE(config-rtr)#router-id 3.3.3.3
TFE(config-rtr)#passive-interface fa0/0
TFE(config-rtr)#no shutdown
TFE(config-rtr)#exit

TFE(config)#int se0/0
TFE(config-if)#ipv6 address AAAA:1:1:2::/64 eui-64
TFE(config-if)#ipv6 eigrp 10
TFE(config-if)#no shutdown
TFE(config-if)#exit
TFE(config)#int fa0/0
TFE(config-if)#ipv6 address BBBB:1:1:2::/64 eui-64
TFE(config-if)#ipv6 eigrp 10
TFE(config-if)#no shutdown


---Configuración LPA---
LPA#conf t
LPA(config)#ipv6 unicast-routing
LPA(config)#ipv6 router eigrp 10
LPA(config-rtr)#router-id 2.2.2.2
LPA(config-rtr)#passive-interface fa0/0
LPA(config-rtr)#no shutdown
LPA(config-rtr)#exit

LPA(config)#int se0/0
LPA(config-if)#ipv6 address AAAA:1:1:3::/64 eui-64
LPA(config-if)#ipv6 eigrp 10
LPA(config-if)#no shutdown
LPA(config-if)#exit
LPA(config)#int fa0/0
LPA(config-if)#ipv6 address BBBB:1:1::3/64
LPA(config-if)#ipv6 eigrp 10
LPA(config-if)#no shutdown
```

```
---Configuración BCN---
BCN#conf t
BCN(config)#ipv6 unicast-routing
BCN(config)#ipv6 router ospf 2
BCN(config-rtr)#router-id 3.3.3.3
BCN(config-rtr)#passive-interface fa0/0
BCN(config-rtr)#exit

BCN(config)#int se0/0
BCN(config-if)#ipv6 address AAAA:1:1:4::/64 eui-64
BCN(config-if)#ipv6 ospf 2 area 0
BCN(config-if)#no shutdown
BCN(config-if)#exit
BCN(config)#int fa0/0
BCN(config-if)#ipv6 address BBBB:1:1:4::/64 eui-64
BCN(config-if)#ipv6 ospf 2 area 0
BCN(config-if)#no shutdown

---Configuración SEV---
SEV#conf t
SEV(config)#ipv6 unicast-routing
SEV(config)#ipv6 router ospf 3
SEV(config-rtr)#router-id 2.2.2.2
SEV(config-rtr)#passive-interface fa0/0
SEV(config-rtr)#exit

SEV(config)#int se0/0
SEV(config-if)#ipv6 address AAAA:1:1:5::/64 eui-64
SEV(config-if)#ipv6 ospf 3 area 0
SEV(config-if)#no shutdown
SEV(config-if)#exit
SEV(config)#int fa0/0
SEV(config-if)#ipv6 address BBBB:1:1:5::/64 eui-64
SEV(config-if)#ipv6 ospf 3 area 0
SEV(config-if)#no shutdown
```

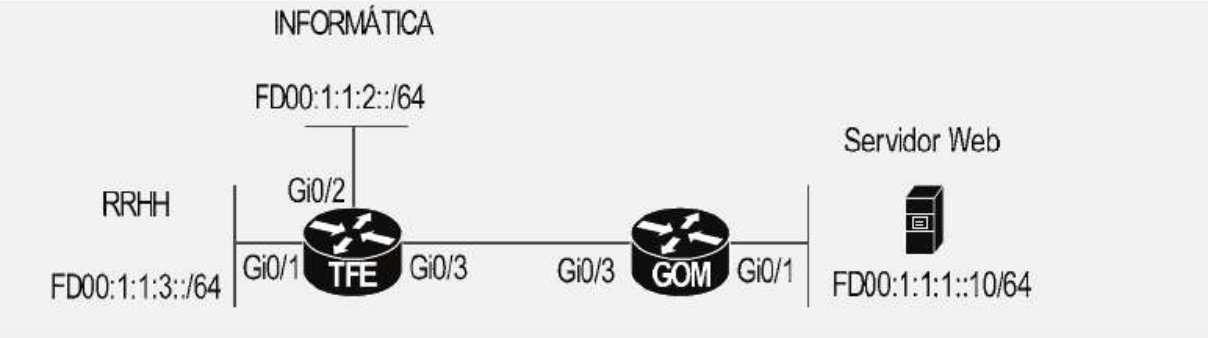
Reto 11.2 – Conforme a la siguiente topología, configurar las ACLs necesarias para cumplir los siguientes objetivos:

- Todos los dispositivos ubicados en las subredes FD00:1:1:2::/64 y FD00:1:1:3::/64 deben acceder al servidor web con dirección FD00:1:1:1::10/64 a través del puerto TCP 80.

- Además, los hosts ubicados en la subred de informática también deben tener acceso vía SSH al mismo servidor.

- Permitir cualquier tipo de tráfico entre las subredes de Informática y RRHH.

- Bloquear cualquier otro tipo de comunicación.




```
TFE(config)#ipv6 access-list ACCESO-RRHH
TFE(config-ipv6-acl)#permit tcp FD00:1:1:3::/64 host FD00:1:1:1::10/64 eq 80
TFE(config-ipv6-acl)#permit ipv6 FD00:1:1:3::/64 FD00:1:1:2::/64
TFE(config-ipv6-acl)#exit

TFE(config)#ipv6 access-list ACCESO-INFORMATICA
TFE(config-ipv6-acl)#permit tcp FD00:1:1:2::/64 host FD00:1:1:1::10/64 eq 80
TFE(config-ipv6-acl)#permit tcp FD00:1:1:2::/64 host FD00:1:1:1::10/64 eq 22
TFE(config-ipv6-acl)#permit ipv6 FD00:1:1:2::/64 FD00:1:1:3::/64
TFE(config-ipv6-acl)#exit

TFE(config)# interface Gi0/1
TFE(config-if)#ipv6 traffic-filter ACCESO-RRHH in
TFE(config-if)#exit

TFE(config)# interface Gi0/2
TFE(config-if)#ipv6 traffic-filter ACCESO-INFORMATICA in
TFE(config-if)#exit
```

TEST CAPÍTULO 11: IP VERSIÓN 6

1.- ¿De cuántos dígitos está compuesta una dirección IPv6? (Seleccionar dos respuestas)

A. 64 dígitos decimales.
B. 32 dígitos hexadecimales.
C. 96 dígitos binarios.
D. 128 dígitos hexadecimales.
E. 128 dígitos binarios.

2.- La dirección A00A:0000:0000:0000:FF5B:00AA:0000:0000 también puede ser representada como…

A. AA::FF5B:AA::
B. AA::FF5B:AA:0:0
C. A00A::FF5B:AA:0:0
D. A00A::FF5B:AA::

3.- ¿Cuáles de los siguientes protocolos de enrutamiento pueden ser implementados sobre entornos IPv6? (Seleccionar dos respuestas)

A. RIPng.
B. OSPFv2.
C. RIPv3.
D. EIGRPv6.

4.- Si a una compañía se le ha asignado el prefijo global AAAA:: /80, ¿de cuántos bits dispone para crear subredes?

A. 64.
B. 80.
C. 48.
D. 16.

5.- Una dirección IPv6 que comienza con los dígitos hexadecimales FD identifica:

A. Una dirección *global unicast*.
B. Una dirección *unique local*.
C. Una dirección *multicast*.
D. Una dirección *Link-Local*.

6.- ¿En qué se basa el método EUI-64?

A. En generar una dirección IPv6 automáticamente con relación a la MAC del dispositivo.

B. En generar una dirección IPv6 de manera automática asignando a la parte de host los valores hexadecimales de la MAC.
C. En utilizar la MAC como dirección IPv6.
D. En generar un prefijo IPv6 automáticamente en relación con la MAC del dispositivo.

7.- ¿Con qué código es identificada una red remota agregada por EIGRPv6 a la tabla de rutas?

A. OE.
B. EI.
C. D.
D. E.

8.- De las siguientes opciones, ¿cuál representa un método de transición de IPv4 a IPv6?

A. Dual configuration.
B. Dual protocols.
C. Dual stack.
D. Dual comunication.
E. EIGRPv6 y OSPFv3.

9.- El prefijo de la IP A00A:0000:0000:0000:FF5B:00AA:0000:0000/56 es…

A. A00A::0:0::
B. A00A:0000:0000:0000:FF00:0000:0000:0000
C. A00A:0:0:0:FF::
D. A00A::

10.-Tras realizar los cálculos necesarios, una compañía concluye que destinará 20 dígitos hexadecimales para subred y los restantes para hosts. ¿Cuál es la longitud de prefijo aplicada por dicha compañía?

A. /64.
B. /60.
C. /128.
D. /96.
E. /80.

11.- ¿Qué diferencia hay entre el comando “ipv6 address 2ABA:10:0:FFFF::10/64” e “ipv6 address 2ABA:0010:0000:FFFF:0000:0000:0000:0010/64”?

A. En la primera se configura un prefijo y en la segunda una dirección IP.

B. La primera identifica un método de autoconfiguración y la segunda una IP fija.
C. La primera es una dirección *Link-Local,* mientras que la segunda una *unique local*.
D. En ambos se configura la misma IP.

12.- Se ha aplicado el siguiente comando sobre una interfaz:

```
ipv6 address AABB:1:1:1::/64 eui-64
```

¿Qué acción llevará a cabo?

A. Generará una dirección IPv6 automáticamente gracias al protocolo NDP.
B. Generará una dirección IPv6 automáticamente en relación con la MAC del dispositivo.
C. Se calculará el prefijo necesario para que la interfaz forme parte de la red AABB:1:1:1::/64.
D. Ninguna de las anteriores.

13.- El prefijo de la IP FFFF:54A0:0000:4322:FBAA:00AA:0000:0000/4 es…

A. FFFF::
B. FFFF:54::
C. FF::
D. F::

14.- Una dirección IPv6 cuyos primeros dígitos hexadecimales corresponden a FF identifica:

A. Una dirección *global unicast*.
B. Una dirección *unique local*.
C. Una dirección *multicast*.
D. Una dirección *Link-Local*.

15.- ¿Qué datos de conexión son proporcionados por un servidor *DHCP Stateless* a un host?

A. Dirección IP.
B. Servidores DNS.
C. Puerta de enlace predeterminada.
D. Prefijo de red.
E. Todas las anteriores.

16.- ¿Cuáles de las siguientes condiciones se deben cumplir para que una red directamente conectada sea agregada a la tabla de rutas en IPv6? (Seleccionar dos respuestas)

A. Que el enrutamiento IPv6 se encuentre habilitado.
B. Que la interfaz que conecta con la red se encuentre en estado *"up/up".*
C. Que la interfaz que conecta con la red opere en dual stack.
D. Que la interfaz sea FastEthernet o Gigabit Ethernet.

17.- ¿En qué modo de configuración deben ser definidas las interfaces que formarán parte de OSPFv3?

A. Modo de configuración global.
B. Modo de configuración del protocolo.
C. Modo de configuración de cada una de las interfaces.
D. Ninguna de las anteriores.

18.- La IP ::1 identifica una dirección…

A. No especificada.
B. Privada.
C. Pública.
D. Link-Local.
E. Loopback.

19.- ¿Qué protocolo de enrutamiento hace uso de la dirección multicast FF02::A?

A. RIPng.
B. OSPFv3.
C. IS-IS.
D. EIGRPv6.

20.- Una dirección "global unicast"…

A. Pertenece a una red privada.
B. Es aquella necesaria para la comunicación en redes públicas.
C. Es reservada por IANA para un fin específico.
D. Se genera automáticamente conforme a la MAC del dispositivo.

21.- Las interfaces de un router Cisco han sido configuradas correctamente para operar en IPv6. Sin embargo, no se logra la comunicación a través de ninguna de ellas. ¿A qué puede ser debido?

A. Alguna interfaz serial ha sido deshabilitada con el comando "*shutdown*".
B. No se ha instalado el medio físico correcto para IPv6.

C. Los protocolos de enrutamiento configurados no son compatibles con IPv6.
D. No se ha habilitado el enrutamiento IPv6 desde el modo de configuración global.

22.- Tras realizar el cálculo oportuno, una compañía dispone de 30 dígitos hexadecimales para subred y 2 para hosts. ¿Cuántos dispositivos podrá albergar cada una de las subredes?

A. 254.
B. 4094.
C. 14.
D. 65534.

23.- Los mensajes RS de NDP, ¿a qué dirección son enviados?

A. FF02::5
B. FF02::4
C. FF02::A
D. FF02::2

24.- ¿Cuáles de los siguientes métodos pueden ser utilizados por IPv6 para que los hosts obtengan una dirección IP de manera automática? (Seleccionar dos respuestas)

A. NDP.
B. SLAAC.
C. DHCP Stateful.
D. NIC.
E. DHCP Stateless.

25.- ¿Cuáles de las siguientes características corresponden a direcciones *Link-Local*? (Seleccionar tres respuestas)

A. Son generadas automáticamente.
B. Utilizadas para el enrutamiento entre diferentes redes privadas.
C. Son direcciones unicast.
D. Son direcciones multicast.
E. Comienzan con los dígitos hexadecimales FF.
F. Comienzan con los dígitos hexadecimales FE.

26.- ¿Qué función lleva a cabo la característica DAD incluida en el protocolo NDP?

A. Calcular automáticamente la parte de host para una dirección IPv6.
B. Realizar la traducción de direcciones IP a MAC.

C. Detectar direcciones duplicadas en la red.
D. Obtener automáticamente la puerta de enlace predeterminada.

27.- ¿Cuál es el formato utilizado por el RID tanto en EIGRPv6 como en OSPFv3?

A. 128 bits divididos en 4 octetos de 32 bits cada uno.
B. 128 bits divididos en 8 octetos de 16 bits cada uno.
C. 64 bits divididos en 4 octetos de 16 bits cada uno.
D. 32 bits divididos en 4 octetos de 8 bits cada uno.

28.- ¿Cuántos bits son reservados para prefijo y cuántos para host en las direcciones *Link-Local*?

A. 80 bits de prefijo y 48 bits de host.
B. 48 bits de prefijo y 80 bits de host.
C. 32 bits de prefijo y 96 bits de host.
D. 64 bits de prefijo y 64 bits de host.

GESTIÓN DE IOS 12

PROTOCOLOS DE MONITORIZACIÓN

Una de las funciones más importantes a desarrollar en toda red consiste en la monitorización y control de los elementos que la conforman. Un fallo en cualquiera de ellos puede suponer la falta de conectividad o caída de algún servicio, hecho que en entornos de producción equivale a la pérdida de bienes por parte de la compañía, convirtiendo esta tarea en un procedimiento imprescindible y prioritario para cualquier administrador. Para llevarla a cabo bastará con analizar los logs generados por cada dispositivo, los cuales mantienen un registro sobre todos los eventos que suceden en el mismo, al cual, en dispositivos Cisco, se accede a través del comando *show logging.*

Dicho modo de supervisión puede resultar viable, pero nada recomendable teniendo en cuenta los siguientes factores. Primero, las redes corporativas están compuestas por multitud de dispositivos, acceder a cada uno de ellos día tras día para leer sus logs resulta una tarea bastante tediosa y nada escalable. Y segundo, la memoria para su almacenamiento es limitada, pudiendo agotarse el espacio reservado para ello y teniendo como consecuencia la sobreescritura de los eventos más antiguos.

Con el objetivo de facilitar y optimizar dicho procedimiento han sido desarrollados diferentes protocolos, todos ellos con la misión de monitorizar la red, pero diferenciados en su modo de operar y utilidad final. Cinco de ellos son Syslog, SNMP, IPSLA, NetFlow y SPAN analizados a continuación.

Syslog

Syslog es el protocolo encargado de recopilar y almacenar de manera centralizada los *logs* generados por los distintos dispositivos ubicados en la red, facilitando, gracias a ello, su lectura y análisis.

Su modo de operar resulta bastante sencillo, donde los clientes envían sus registros a un servidor, de tal manera que el administrador dispondrá de todos ellos en la misma ubicación. Sin embargo, hay que tener en cuenta que cualquier suceso ocurrido en el dispositivo genera una entrada de log, por ejemplo, el envío o recepción de cualquier paquete a través de las interfaces. La consecuencia de ello es que los ficheros contendrán millones de registros, resultando imposible de analizar y a su vez ocupando demasiada memoria. Además, la gran mayoría de ellos identifican comportamientos normales en la red.

La información que realmente interesa es aquella que alerte sobre incidencias en el dispositivo, con el fin de solucionarlas a la mayor brevedad posible. Para ello, syslog permite el filtrado de mensajes, dividiéndolos en 8 niveles con relación al tipo de evento ocurrido. Estos son:

Nivel	Nombre	Evento
0	Emergency	Dispositivo o sistema caído.
1	Alert	Se requiere alguna acción inmediata sobre el dispositivo.
2	Critical	Suceso crítico. Se requiere intervención inmediata.
3	Error	Algún error que requiere la intervención del administrador.
4	Warning	Eventos que deben ser analizados.
5	Notification	Eventos normales, pero poco usuales o significativos.
6	Informational	Mensajes de información sobre algún suceso acaecido.
7	Debugging	Todos los eventos y paquetes/tramas que atraviesan el dispositivo.

De todos ellos, los niveles comprendidos entre el 0 y el 4 representan incidencias que requieren la intervención del administrador para resolverlas, mientras que

aquellos del 5 al 7 hacen referencia a operaciones normales de red, generando infinidad de entradas.

Un ejemplo de registro syslog en dispositivos Cisco podría ser el siguiente:

```
*jul 13, 05:02:29.022: %LINK-5-CHANGED: Interface GigabitEthernet0/0, changed
state to administratively down
```

Dónde:

- *jul 13, 05:02:29.022* indica la fecha y hora exacta en la que se produjo el evento.

- *%LINK* hace referencia al elemento sobre el cual se ha generado la acción.

- *5* indica el tipo de evento.

- *CHANGED* es la nomenclatura utilizada para identificar la acción que propició el registro.

- *Interface GigabitEthernet0/0, changed state to administratively down* es una breve descripción de lo ocurrido.

¿Qué conclusiones se pueden obtener en relación con la información recibida? Cuando una interfaz se encuentra en dicho estado es porque ha sido deshabilitada administrativamente. Es decir, alguien ha aplicado el comando *shutdown* sobre la interfaz Gi0/0 el día 13 de julio a las 05:02:29.022 horas.

Uno de los detalles más importantes a tener en cuenta durante la lectura y análisis de logs es la fecha y hora en la que se produce cada evento, gracias a la cual se podrá establecer una correlación de mensajes, ayudando en gran medida a resolver el problema y determinar la causa que lo motivó, sobre todo en cuestiones de seguridad. En Cisco, para que dicho dato sea incluido debe ser habilitado previamente el servicio *timestamps*, a través del comando **service timestamps log datetime msec**, desde el modo de configuración global. Además, resulta altamente recomendable que todos los dispositivos mantengan su hora sincronizada, logrando dicho objetivo mediante la aplicación del protocolo NTP, ya analizado en el capítulo 7 "*Seguridad en capa 3*".

CONFIGURACIÓN DE SYSLOG

Su aplicación se lleva a cabo mediante las siguientes acciones, todas ellas ejecutadas desde el modo de configuración global:

- *Paso 1:* Identificar el servidor syslog al cual serán enviados los mensajes, con el comando **logging *[ip servidor]***.

- *Paso 2:* Seleccionar el tipo de sucesos que serán filtrados, mediante la sentencia **logging trap *[nivel],*** donde *nivel* hace referencia a un valor decimal comprendido entre el 0 y el 7, el cual también incluye los niveles inferiores.

- *Paso 3 (Opcional):* Si no lo está, habilitar el servicio *timestamps* con el comando **service timestamps log datetime msec**.

Ejemplo: Configurar syslog en LPA, TFE y SW1 para que los eventos de nivel 0, 1, 2, 3 y 4 sean enviados y almacenados en el servidor 10.10.10.10. El servicio *timestamps* ya se encuentra habilitado en todos los dispositivos.

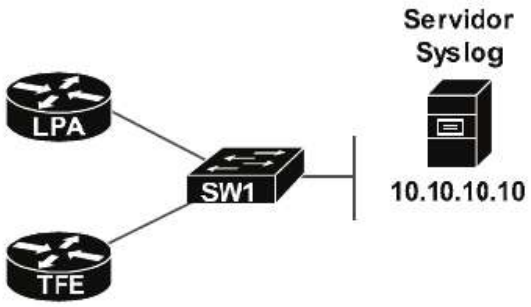


Fig. 12-1 Diseño de red para supuesto práctico de syslog.

```
---Configuración en TFE---
TFE(config)# logging 10.10.10.10
TFE(config)# logging trap 4

---Configuración en LPA---
LPA(config)# logging 10.10.10.10
LPA(config)# logging trap 4

---Configuración en SW1---
SW1(config)# logging 10.10.10.10
SW1(config)# logging trap 4
```

SNMP

SNMP *(Simple Network Management Protocol)* se ha convertido en uno de los protocolos imprescindibles en cualquier red, implementado para monitorizar los elementos ubicados en la misma. Su misión consiste en el envío de notificaciones que alertan al administrador sobre sucesos ocurridos en el dispositivo, con el fin de que sean solucionados a la mayor brevedad posible. Aunque su finalidad pudiera resultar similar, SNMP y syslog desarrollan funciones bien diferenciadas, entre las que se encuentran:

- Syslog almacena un registro sobre todos los sucesos ocurridos en el dispositivo, el cual debe ser analizado en busca de aquellos que indiquen alguna incidencia. Mientras, SNMP alerta directamente al administrador sobre los eventos que este considere oportunos, normalmente aquellos que identifiquen errores.

- Syslog basa su modo de operar en el modelo cliente-servidor, mientras que SNMP define dos roles, mánager y agente. El mánager es quien solicita y recibe alertas de los agentes, mientras que estos últimos son los dispositivos monitorizados, como routers, switchs, servidores, etc. El intercambio de información entre ambos se basa por completo en la MIB, la cual será objeto de estudio en párrafos posteriores.

- De ambos protocolos, SNMP es considerado la opción ideal para dar respuesta inmediata ante cualquier suceso acaecido en la red, mientras que syslog representa la utilidad por excelencia para realizar un análisis de eventos o comportamiento de un determinado dispositivo.

Las funciones llevadas a cabo por SNMP resultan posibles gracias a una base de datos almacenada en cada agente, la cual define los elementos a monitorizar y genera la información que se enviará al mánager. Esta recibe el nombre de MIB *(Management Information Base)* y hace uso de una estructura jerárquica dividida en grupos, cada uno de ellos compuesto por objetos. A su vez, cada objeto obtiene un ID *(Object ID - OID)* que identifica el elemento a monitorizar o del cual se solicita información.

Los grupos en los que se estructura una MIB son los siguientes:

- System: Alberga aquellos objetos referentes al software y hardware presente en dispositivo (versión, tiempo operativo, último reinicio, etc.).
- Interfaces: Agrupa los objetos que facilitan información sobre cada una de las interfaces disponibles (estado, carga, velocidad, etc.).
- ATT (Address Translation Table): Direcciones de red, MACs…

- IP: Tablas de rutas, paquetes IP generados o recibidos…
- ICMP: Paquetes ICMP recibidos, errores, etc.
- TCP: Objetos referentes al protocolo TCP.
- UDP: Objetos referentes al protocolo UDP.
- EGP: Objetos referentes al protocolo EGP.
- Transmission: Objetos que hacen referencia a métodos de transmisión o comunicación.
- SNMP: Objetos propios del protocolo SNMP.

En total, 10 grupos que albergan un total de 185 objetos de los cuales se podrá obtener información mediante dos métodos, bien siendo solicitada, o bien siendo recibida de manera automática. Para el primer caso, el protocolo define el comando GET, el cual debe ser enviado al agente indicando el ID de objeto del cual se requiere información. Este responderá proporcionándola. Mientras, el segundo método es considerado la mejor utilidad de SNMP, consistente en alertar al mánager cuando ocurre alguna incidencia y sin necesidad de que este lo haya solicitado previamente. Este proceso se lleva a cabo mediante mensajes *SNMP Trap*, que son aquellos generados y enviados automáticamente por el protocolo cuando algún elemento presenta errores, por ejemplo, la caída de una interfaz. En este caso, se deberán definir previamente los objetos de los cuales se generarán este tipo de mensajes.

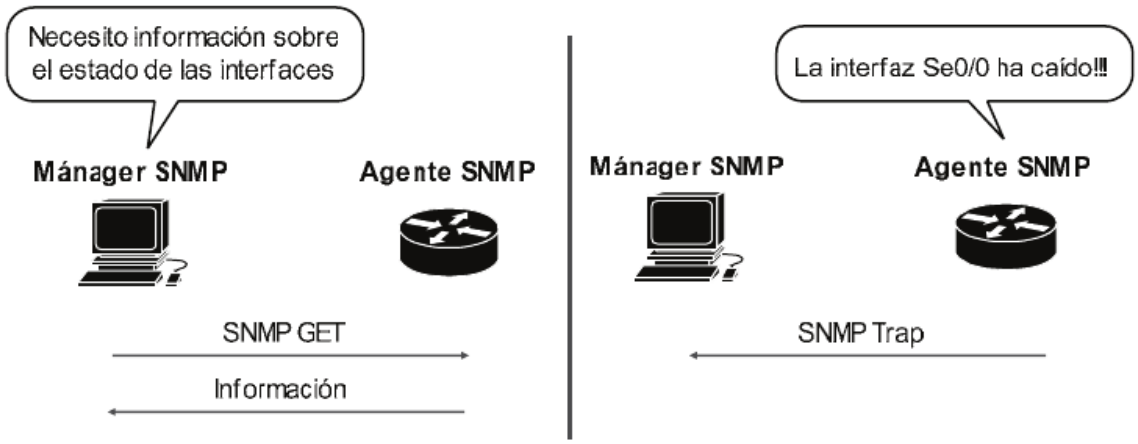


Fig. 12-2 Comunicación entre mánager y agente SNMP.

VERSIONES DE SNMP

SNMP dispone de tres versiones, cada cual desarrollada para solventar carencias o mejorar aspectos de sus antecesores. De todas ellas, la 1 y 2c resultan muy básicas, siendo posible la autenticación en la MIB en relación con dos tipos de permiso:

- De solo lectura (RO), mediante el cual solo se podrá obtener información.

- De lectura y escritura (RW), que también permite aplicar cambios de configuración, como habilitar o deshabilitar una interfaz.

El problema reside en que dicha autenticación es enviada en texto plano, lo que a día de hoy representa una vulnerabilidad grave, hecho por el cual en estas versiones se recomienda el modo de solo lectura.

La versión 3 del protocolo solventa dicha carencia e implementa seguridad a todos los niveles, agregando las siguientes características sobre la comunicación:

- Integridad: Evita que el mensaje sufra modificaciones desde el origen hasta el destino.
- Autenticación: Permite la creación de usuarios como método de autenticación sobre la MIB. En este caso, la contraseña nunca es enviada en texto plano.
- Cifrado: Como su nombre indica, permite aplicar mecanismos de cifrado sobre la comunicación entre mánager y agente.

La versión 2 de SNMP no fue aceptada ni implementada por fabricantes debido a la complejidad presente en su modo de operar, hecho que dio lugar a la v.2c.

CONFIGURACIÓN DE SNMP VERSIÓN 2C

Su aplicación se lleva a cabo mediante el comando **snmp-server community *[string] [RO|RW] [Nombre_ACL]***, desde el modo de configuración global y donde:

- *[String]* hace referencia a la cadena de texto necesaria para la autenticación en la MIB.

- *[RO][RW]* establece el tipo de permiso que se concederá sobre los objetos, de solo lectura o lectura y escritura.

- *[Nombre ACL]* es un parámetro opcional que aplica una ACL sobre la MIB. Su configuración resulta recomendable ya que permite habilitar el acceso solo a determinados hosts, agregando una capa de seguridad sobre la comunicación.

Ejemplo: Configurar TFE y LPA como agentes SNMP cumpliendo los siguientes requisitos:

- LPA requiere la palabra “PassLPA” como autenticación, obteniendo permisos de solo lectura.

- La MIB de TFE permitirá la modificación de sus objetos, siendo necesaria la cadena “PassTFE” para acceder a la misma. Además, solo podrá llevarlo a cabo el host con IP 192.168.1.1.

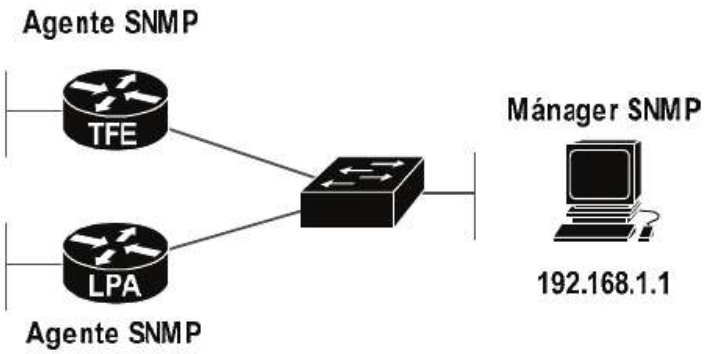


Fig. 12-3 Diseño de red para supuesto práctico de SNMP.

```
---Configuración en LPA---
LPA(config)# snmp-server community PassLPA RO

---Configuración en TFE---
TFE(config)# ip access-list standard ACL_SNMP
TFE(config-std-nacl)# permit host 192.168.1.1
TFE(config-std-nacl)# exit
TFE(config)# snmp-server community PassTFE RW ACL_SNMP
```

USUARIOS Y GRUPOS EN SNMPV3

Una de las grandes novedades incluidas en la última versión del protocolo consiste en la eliminación del concepto *community*, gracias al cual se definían los permisos y cadena de autenticación necesaria sobre la MIB. En este caso, dicha validación es sustituida por un sistema basado en usuarios, los cuales formarán parte de grupos, que a su vez dispondrán de los permisos que el administrador considere oportuno.

La creación de grupos se lleva a cabo mediante el comando **snmp-server group *[nombre]* v3 *[noauth | auth | priv]* read *[nombre]* write *[nombre]* access *[nombre acl]*.** Donde *[noauth | auth | priv]* establecen el nivel de seguridad deseado en relación con las siguientes características:

	Integridad	Autenticación	Cifrado
noauth	Sí	No	No
auth	Sí	Sí	No
priv	Sí	Sí	Sí

El resto de parámetros incluidos en la sentencia resultan opcionales, siendo necesarios para los siguientes propósitos:

- read *[nombre]*: Establece permisos de lectura sobre la MIB. Aunque no fuera configurada, el permiso de lectura es aplicado por defecto con la vista denominada "*V1Default*".

- write *[nombre]*: Establece permisos de escritura sobre la MIB. Si no fuera configurada el usuario no podría modificar objetos (modo RO en la versión 2c). Al definirla, se debe especificar un nombre de vista de escritura en el campo *[nombre].*

- access *[nombre acl]*: Aplica una ACL, previamente definida, sobre la MIB.

Cisco incluye diferentes vistas predefinidas tanto de lectura como de escritura que pueden ser aplicadas sobre cada grupo, por ejemplo, la denominada *"V1Default".* Sin embargo, también pueden ser creadas manualmente. Su configuración no forma parte del contenido de CCNA.

Varios ejemplos de grupos podrían ser:

Comando	Resultado
`snmp-server group TENERIFE v3 auth write v1default`	Crea un grupo denominado TENERIFE, con autenticación, integridad y permisos de lectura/ escritura sobre la MIB.
`snmp-server group LPA v3 noauth access ADMINS`	Crea un grupo denominado LPA, aplicando integridad, pero sin autenticación, con permisos de solo lectura y ejecutando la ACL "ADMINS" sobre la MIB.
`snmp-server group BCN v3 priv`	Crea un grupo denominado BCN, con permisos de solo lectura, aplicando integridad, autenticación y cifrado sobre la comunicación.

Una vez creados los grupos necesarios deben ser definidos los usuarios que formarán parte de cada uno de ellos. Dicha acción es llevada a cabo mediante el comando **snmp-server user *[nombre usuario] [grupo]* v3 *[auth | priv] [algoritmo] [cadena de texto],*** donde:

- *[nombre usuario]:* Como su nombre indica, hace referencia al usuario que será creado.

- *[grupo]:* El grupo al que pertenecerá.

- *[auth | priv]:* Identifica el método de autenticación y/o cifrado que se llevará a cabo con dicho usuario. Este parámetro depende del permiso definido en el grupo al que pertenece, si fuera *auth* tan solo será necesario definir la autenticación, sin embargo, si fuera *priv,* se hace necesario asignar tanto la autenticación como el método de cifrado.

- *[algoritmo] [cadena de texto]:* Asigna los algoritmos a ejecutar durante el proceso de autenticación y/o cifrado. Como autenticación se podrá optar por MD5 o SHA, mientras que, para el cifrado, DES, 3DES o AES. *[Cadena de texto]* hace referencia a la contraseña o valor de llave para cada uno de los casos. Además, si el método de cifrado fuera AES, también se deberá indicar una longitud de clave (ver ejemplos).

Ejemplos de usuarios para el grupo “TENERIFE”:

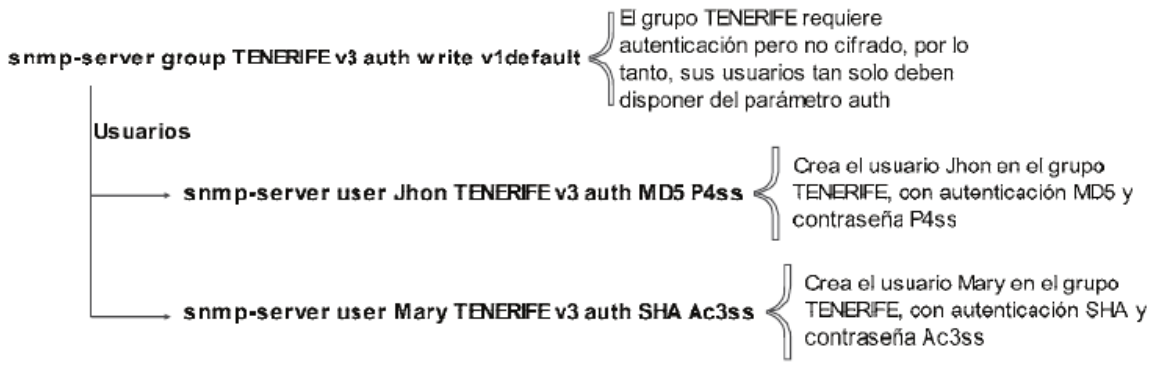


Fig. 12-4 Grupo SNMPv3 con autenticación sin cifrado.

Ejemplos de usuarios para el grupo “LPA”:

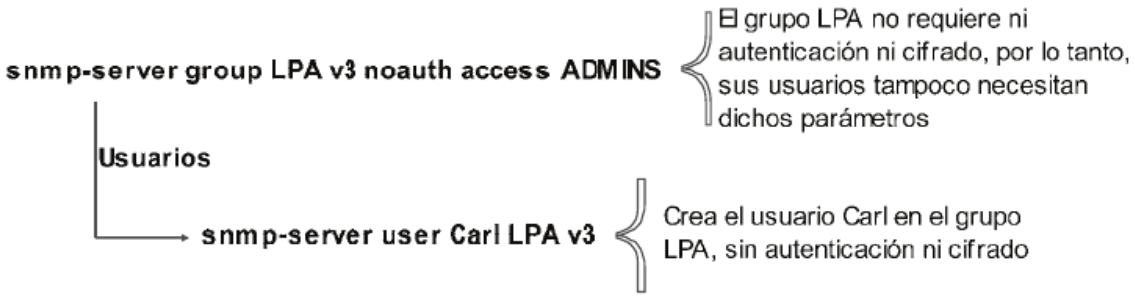


Fig. 12-5 Grupo SNMPv3 sin autenticación ni cifrado.

Ejemplos de usuarios para el grupo "BCN":

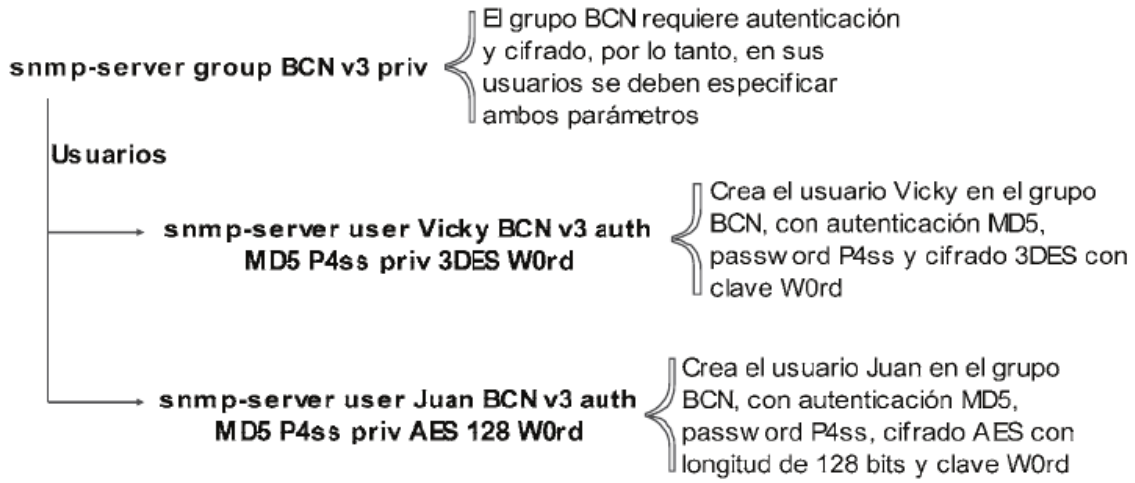


Fig. 12-6 Grupo SNMPv3 con autenticación y cifrado.

CONFIGURACIÓN DE SNMPV3

Por último, la aplicación de SNMPv3 se lleva a cabo a través del siguiente procedimiento:

- *Paso 1:* Definir los grupos necesarios con el comando recién analizado.

- *Paso 2:* Crear y asignar usuarios a dichos grupos mediante la sentencia y proceso recién analizado.

- *Paso 3:* Ejecutar el comando **snmp-server enable traps**, gracias al cual se habilita el envío de mensajes *trap* ante cualquier suceso acaecido sobre los objetos presentes en la MIB.

- *Paso 4:* Identificar el host de destino hacia el cual serán enviadas las notificaciones, mediante la sentencia **snmp-server host *[ip destino] [informs | traps]* version 3 *[noauth | auth | priv] [nombre usuario]***, donde *[ip destino]* identifica el host remoto e *[informs | traps]* hace referencia al modo de envío de mensajes. Ambas opciones alertan sobre los mismos eventos, sin embargo, *inform* hace uso de ACKs para verificar que la comunicación ha sido recibida por el destinatario. *[noauth | auth | priv] [nombre usuario]* identifica el usuario y características de seguridad del mismo, las cuales deben coincidir con las definidas previamente en el grupo al que pertenece.

Ejemplo: Configurar SNMPv3 en R1 de tal manera que:

- Se deben crear los grupos ADMINS y EVENTS, donde el primero aplicará integridad, cifrado, autenticación y permisos de lectura/escritura, mientras que el segundo integridad, autenticación y permisos de solo lectura.
- El usuario Billy pertenecerá al grupo EVENTS y aplicará autenticación MD5 con contraseña “4cc3so”.
- El usuario Adminsys pertenecerá al grupo ADMINS y aplicará autenticación SHA con contraseña “4cc3so”, haciendo uso del algoritmo de cifrado 3DES con clave “P4ss”.
- Los eventos ocurridos deben ser enviados a las direcciones IP 10.10.10.250 (haciendo uso del usuario Billy) y 10.10.10.251 (con usuario Adminsys y uso de ACKs).

```
R1(config)#snmp-server group ADMINS v3 priv write v1default
R1(config)#snmp-server group EVENTS v3 auth

R1(config)#snmp-server user Billy EVENTS v3 auth md5 4cc3so
R1(config)#snmp-server user Adminsys ADMINS v3 auth sha 4cc3so priv 3des P4ss

R1(config)#snmp-server enable traps

R1(config)#snmp-server host 10.10.10.250 traps version 3 auth Billy
R1(config)#snmp-server host 10.10.10.251 informs version 3 priv Adminsys
```

IPSLA

Otro de los métodos disponibles en cuanto a monitorización se refiere es el denominado IPSLA *(IP Service Level Agreement)*, aplicable tan solo en routers Cisco y gracias al cual se hace posible llevar a cabo mediciones de tiempo de respuesta ante diferentes tipos de tráfico. Con ello, el administrador dispone de información útil para determinar incidencias de red, enlaces congestionados o insuficiencia de ancho de banda.

El modo de operar resulta sencillo, donde el router genera un determinado tráfico de la misma manera que lo haría un dispositivo final, para acto seguido enviarlo hacia un destino definido. Es en este momento cuando IPSLA mide el tiempo de respuesta ante cualquier paquete enviado, almacenando los resultados en contadores que podrán ser analizados por el administrador en cualquier momento, en relación con los cuales se hace posible determinar posibles anomalías, por ejemplo, que los tiempos de respuesta varíen considerablemente ante un mismo tráfico.

Como recién se ha mencionado, las mediciones se llevan a cabo entre un origen y un destino definidos, donde el primero recibe el nombre de *"SLA Source",* mientras que el segundo es denominado *"SLA Responder"*. Este último normalmente identifica a otro router que debe ser configurado para tal propósito, es decir, responder al tráfico IPSLA generado por el origen. Sin embargo, el *Responder* puede variar dependiendo del protocolo a monitorizar, por ejemplo, si se tratara de ICMP, el destino puede ser cualquier dispositivo, sin necesidad de requerir ningún tipo de configuración previa.

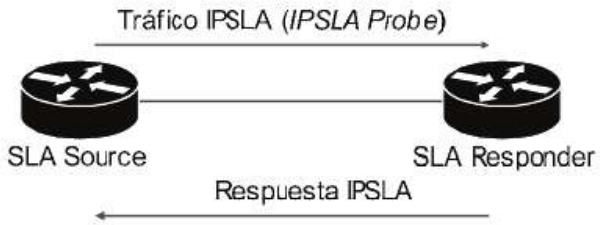


Fig. 12-7 Monitorización IPSLA.

CONFIGURACIÓN DE IPSLA ICMP

Una de las configuraciones más básicas consiste en ejecutar un ping de manera periódica hacia un mismo destino. Para ello bastará con llevar a cabo el siguiente procedimiento.

- *Paso 1:* Crear un proceso de IPSLA con el comando **ip sla *[id],*** desde el modo de configuración global y donde *id* hace referencia a un valor numérico que identificará dicha instancia a nivel local. Acto seguido se accede a un modo de configuración propio desde el cual se deben ejecutar los pasos 2 y 3.

- *Paso 2:* Seleccionar el destino mediante la sentencia **icmp-echo *[ip destino]* source-ip *[ip origen]*** o **icmp-echo *[ip destino]* source-interface *[interfaz origen]*.** Ambos métodos ejecutan la misma acción, variando en la manera de identificar la dirección IP de origen.

- *Paso 3:* Definir el intervalo de tiempo que se aplicará para el envío de cada paquete ICMP, con el comando **frequency *[segundos].***

- *Paso 4:* Por último, iniciar el servicio con el comando **ip sla schedule *[id]* life forever start-time now**, desde el modo de configuración global y donde *id* hace referencia a aquel creado en el paso 1. IPSLA permite programar el inicio y fin de un proceso. En este caso, se ha definido su ejecución inmediata (*start-time now*) e infinita (*life forever*). Para detenerlo bastará con ejecutar un **no ip sla schedule *[id].***

La configuración de IPSLA contempla multitud de protocolos y parámetros disponibles para cada uno de ellos. Sin embargo, el contenido de CCNA tan solo abarca los conceptos más básicos. Su análisis en detalle forma parte de certificaciones más específicas.

Ejemplo: Configurar IPSLA en R1 para que lleve a cabo mediciones ICMP cada 30 segundos al destino con IP 172.10.1.1, haciendo uso de la dirección de origen 172.10.1.254.

```
R1(config)# ip sla 5
R1(config-ip-sla)# icmp-echo 172.10.1.1 source-ip 172.10.1.254
R1(config-ip-sla)# frequency 30
R1(config-ip-sla)# exit
R1(config)# ip sla schedule 5 life forever start-time now
```

Una vez concluido, los resultados, estadísticas y contadores podrán ser visualizados ejecutando un *show ip sla summary*.

NetFlow

La finalidad de los protocolos recién analizados consiste en la monitorización y análisis, obteniendo y almacenando registros, en el caso de syslog, alertando al administrador cuando se produzca algún evento que requiera intervención, como SNMP, o llevando a cabo mediciones de tiempo de respuesta, como es el caso de IPSLA. Sin embargo, en entornos corporativos también resulta necesario controlar o al menos disponer de información sobre el flujo de datos al que está sometido cada dispositivo.

Para dicho propósito se podrá hacer uso de NetFlow, un protocolo desarrollado por Cisco con el objetivo de monitorizar el tráfico IP que atraviesa cada interfaz, obteniendo mediciones que podrán ser utilizadas para diferentes propósitos, como optimizar el rendimiento de la red instalando los mejores dispositivos en aquellos puntos donde más tráfico se genere, o detectar ataques con relación a flujos de

datos anómalos o desproporcionados, como los llevados a cabo por la denegación de servicio. En definitiva, disponer de información estadística y conforme a la misma efectuar las acciones de mejora que se consideren oportunas.

NetFlow ejecuta las mediciones en dirección de entrada o salida sobre cada interfaz haciendo uso del modelo cliente-servidor, donde el cliente es el dispositivo monitorizado, que enviará los datos a un servidor denominado "*NetFlow Collector*". Este simplemente hace referencia a un software que interpretará la información recibida, disponiendo de diferentes características con relación a la versión instalada, como la identificación del tráfico generado por cada uno de los hosts, sitios web más visitados, contenido más descargado, porcentaje de ancho de banda disponible en cada dispositivo, generación de gráficos e informes, etc.

Resulta importante remarcar que el protocolo no ejecuta ninguna acción sobre el tráfico monitorizado, ni detiene ningún tipo de ataque, simplemente realiza mediciones con el fin de generar datos estadísticos.

CONFIGURACIÓN DE NETFLOW

Su configuración consta de:

- *Paso 1:* Definir la dirección en la cual será monitorizado el tráfico en cada una de las interfaces deseadas, con el comando **ip flow [*ingress* / *egress*]**, desde el modo de configuración de la interfaz. *Ingress* analiza la entrada, mientras que *egress* la salida, permitiendo también aplicar ambas direcciones.

- *Paso 2:* Identificar el servidor al cual serán enviadas las mediciones mediante la sentencia **ip flow-export destination *[ip servidor] [puerto UDP]***, desde el modo de configuración global.

- *Paso 3:* Seleccionar la IP que se utilizará como origen para los paquetes enviados al *NetFlow Collector.* Dicha dirección debe estar asignada a alguna de las interfaces disponibles en el dispositivo, debiendo definir esta como origen. Para ello se debe ejecutar el comando **ip flow-export source *[interfaz]*** desde el modo de configuración global, pudiendo ser física o loopback.

- *Paso 4:* Seleccionar la versión del protocolo a utilizar, con el comando **ip flow-export version *[versión]*** desde el modo de configuración global. Las características de cada una de ellas no forman parte del contenido de CCNA, pero por razones de seguridad y funciones más avanzadas se recomienda la v9.

Ejemplo: Configurar NetFlow en el router TFE para monitorizar el tráfico de entrada y salida hacia Internet, enviando las mediciones al servidor con IP 192.168.1.1 y puerto UDP 5560. Además, se debe aplicar la versión 9 del protocolo y hacer uso de una interfaz loopback con IP 10.1.1.1 /24 como origen de la comunicación hacia el *Collector*.

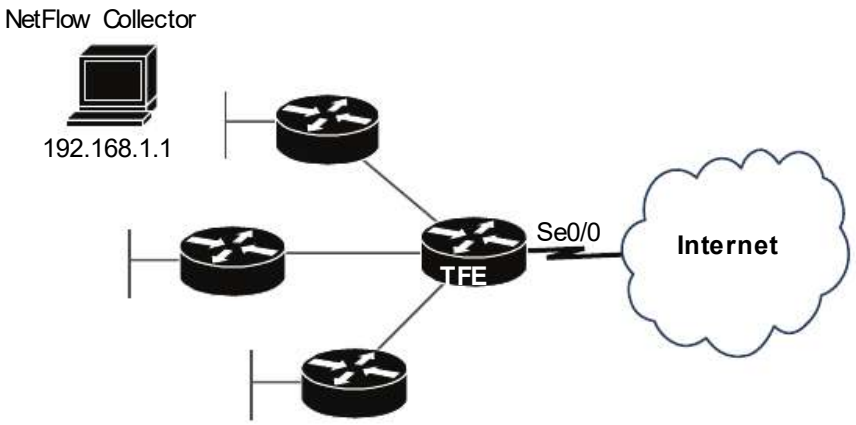
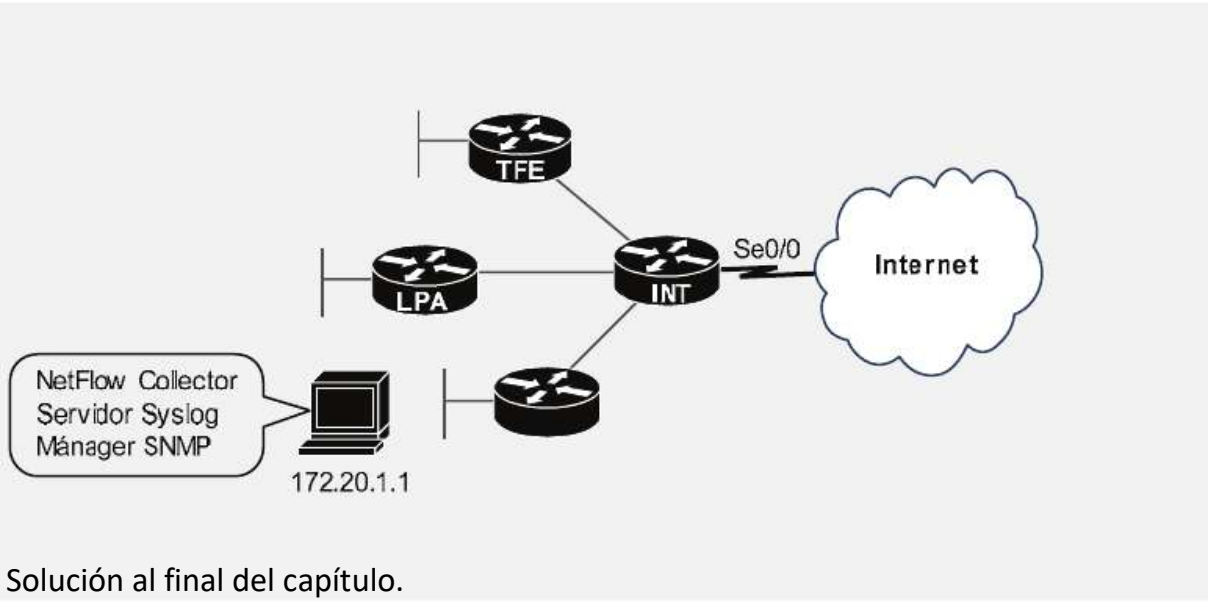


Fig. 12-8 Diseño de red para supuesto práctico NetFlow.

```
TFE(config)# interface Se0/0
TFE(config-if)# ip flow ingress
TFE(config-if)# ip flow egress
TFE(config-if)# exit
TFE(config)# interface loopback 1
TFE(config-if)# ip address 10.1.1.1 255.255.255.0
TFE(config-if)# no shutdown
TFE(config-if)# exit
TFE(config)# ip flow-export destination 192.168.1.1 5560
TFE(config)# ip flow-export source loopback 1
TFE(config)# ip flow-export version 9
```

Reto 12.1 – Configurar la siguiente topología de tal manera que:

- TFE envíe al servidor syslog los eventos ocurridos de nivel 4 o inferior. También se debe habilitar el servicio *timestamps*.
- LPA haga uso de SNMP, con autenticación “P4Ss” y permisos de lectura y escritura al que solo podrá acceder el mánager, con IP 172.20.1.1.
- INT monitorice el tráfico de entrada y salida hacia Internet. El *Netflow Collector* opera en el puerto UDP 4000 y la versión que se debe aplicar es la 9, configurando una interfaz loopback cuya IP (172.30.10.10/24) actuará como origen de la comunicación hacia el servidor.



Solución al final del capítulo.

SPAN

Los métodos de monitorización analizados hasta ahora basan su funcionalidad en la recopilación de datos, llevando a cabo diferentes acciones en torno a estos, donde:

- Syslog almacena un registro de *logs* generados por el dispositivo.

- SNMP alerta al administrador ante eventos que requieran intervención.

- IPSLA lleva a cabo mediciones de tiempo de respuesta ante diferentes tipos de tráfico.

- Netflow provee estadísticas en relación con el flujo de datos analizado.

De tal modo que todos ellos requieren "interactuar" bien con el tráfico, o bien con los elementos de hardware presentes en el dispositivo para lograr su finalidad.

SPAN (*Switched Port Analyzer*) se basa en un concepto totalmente diferente a los anteriores. En este caso, el dispositivo realiza una copia de las tramas recibidas a través de una o varias interfaces, para acto seguido reenviarlas a través de otra. Gracias a ello se hace posible llevar a cabo un análisis sobre dicho tráfico, siendo el administrador de red quien decida las acciones a tomar sobre el mismo. Precisamente este aspecto marca la diferencia con los métodos anteriores, donde la información es recibida directamente (alertas, logs, estadísticas…) sin la posibilidad de operar con los datos que la generan. Sus características más destacadas son:

- Tan solo puede ser aplicado en switchs.
- Cuando una interfaz ha sido configurada como destino de SPAN, el Switch no aprende ni asocia ninguna MAC con dicho puerto. Simplemente reenvía la copia del tráfico hacia el mismo.
- Las interfaces pueden ser monitorizadas en dirección de entrada, salida, o ambas.
- SPAN también permite monitorizar VLANs. En este caso, simplemente identifica aquellas interfaces asociadas a la VLAN en cuestión, realizando una copia del tráfico de todas ellas.
- El enlace de las interfaces configuradas como destino debe operar en modo troncal.

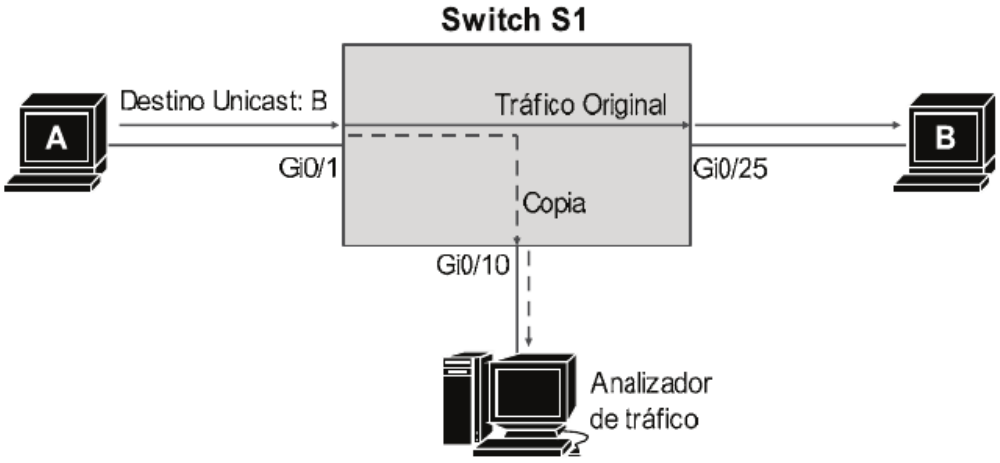


Fig. 12-9 SPAN. Modo de operar.

Con todo ello, la práctica más habitual consiste en la conexión de algún dispositivo con capacidad de capturar tráfico, como un PC con el software *"WireShark"*.

La solución de SPAN analizada en el CCNA tan solo contempla el modo local. Sin embargo, también puede ser configurado entre switchs remotos. Para dicho propósito Cisco dispone dos soluciones, RSPAN (*Remote SPAN*) donde el tráfico entre los dispositivos es enviado a través de un enlace troncal en capa 2, y ERSPAN *(Encapsulated RSPAN)* donde dicho tráfico es enviado a través de un túnel GRE, en capa 3.

CONFIGURACIÓN DE SPAN

El proceso consta de las siguientes acciones, ejecutadas desde el modo de configuración global:

- *Paso 1:* Habilitar SPAN con el comando **monitor session *[id]* source interface *[interfaz o rango] [rx | tx | both]*** o **monitor session *[id]* source vlan *[vlan id] [rx | tx | both]***. Dependiendo de si el tráfico a monitorizar corresponde a una interfaz o conjunto de estas, o a una determinada VLAN. *Id* establece un identificador a nivel local, permitiendo crear varias sesiones diferentes, mientras que *[rx | tx | both]* hace referencia a la dirección en la cual será copiado el tráfico.

- *Paso 2:* Definir la interfaz de salida, ejecutando para ello el comando **monitor session *[id]* destination interface *[interfaz],*** donde *id* corresponde a aquel creado en el paso 1.

Ejemplo: Habilitar SPAN en el Switch S1 para que todo el tráfico generado y recibido por las interfaces Gi0/1 y Gi/20 sea copiado y reenviado hacia la interfaz Gi0/5.

```
S1(config)# monitor session 3 source interface gi0/1 both
S1(config)# monitor session 3 source interface gi0/20 both
S1(config)# monitor session 3 destination interface gi0/5
```

Con el fin de verificar que los cambios han sido aplicados de manera correcta, IOS dispone del comando *show monitor session [id],* el cual muestra en pantalla un breve resumen de aquellos puertos pertenecientes al *id* definido.

```
S1# show monitor session 3
Session 3
---------
Type: Local Session
Source Ports:
RX Only: None
TX Only: None
Both: Gi0/1, Gi0/20
Destination Ports: Gi0/5
```

SECUENCIA DE ARRANQUE Y RECUPERACIÓN DE CONTRASEÑAS

Los routers mantienen una gran similitud con los PCs convencionales en cuanto a elementos que lo conforman se refiere. A nivel de hardware, ambos hacen uso de memoria RAM, procesador, fuente de alimentación o tarjetas de red, entre otros. A su vez, requieren de un sistema operativo para poder gestionar todas sus funciones. En ambos casos, el arranque y puesta en marcha obedece a un procedimiento bien definido y ejecutado de manera secuencial, que dará como resultado la operatividad, o no, del dispositivo. Para un administrador su conocimiento resulta imprescindible, es por ello que la presente sección abordará a su estudio, abarcando los siguientes aspectos:

- Secuencia de arranque en routers Cisco.
- Recuperación de contraseñas.

Secuencia de arranque en routers Cisco

Cisco dispone de una amplia variedad de dispositivos, y a su vez, de modelos de estos, cada uno de ellos con características y capacidades diferenciadas del resto, tanto en elementos de hardware como en la versión de IOS aplicada. Sin embargo, todos coinciden en un mismo comportamiento durante la secuencia de arranque, la cual consta del siguiente procedimiento:

PASO 1: POST

La primera acción llevada a cabo por un router Cisco durante la secuencia de arranque consiste en la ejecución del POST (*Power On Self Test*), el cual verifica y realiza una serie de chequeos sobre cada elemento de hardware presente en el dispositivo.

Si concluye con éxito, el proceso de arranque continúa, sin embargo, si existe algún error puede derivar en diferentes comportamientos, entre ellos:

- Continuar sin cargar algún elemento de hardware, por ejemplo, una interfaz.

- Continuar mostrando un mensaje de error, el cual informa del suceso acaecido.

- Detener el arranque por falta o fallo de algún componente imprescindible de hardware.

- Reinicio constante del dispositivo *(Reboot Loop)*.

- Detener el procedimiento e iniciar el router en modo ROMMON.

De los cuales, el suceso más común consiste en informar de lo ocurrido a través de un mensaje en pantalla, de formato similar a los logs ya analizados en este mismo capítulo.

La función POST se encuentra ubicada y es ejecutada desde la memoria ROM (solo lectura).

El modo ROMMON *(ROM Monitor)* hace referencia a una consola visualmente semejante a la CLI, pero ejecutada cuando se produce algún error durante el proceso de arranque del dispositivo (también puede ser forzada manualmente). Únicamente dispone de comandos de recuperación y reparación, por lo que no presenta ninguna función de red.

PASO 2: CARGA Y EJECUCIÓN DEL BOOTSTRAP

Una vez concluido el POST el router comienza la ejecución del bootstrap, el cual desarrolla dos funciones, inicializar el hardware y localizar el sistema operativo para acto seguido proceder a su carga.

Si concluye con éxito, el proceso de arranque continúa, de lo contrario el dispositivo será iniciado en modo ROMMON. En este caso, las incidencias más frecuentes son:

- No localizar la imagen IOS, bien porque ha sido eliminada, o bien porque no se tenga acceso a su ubicación.

- Que la IOS resulte ilegible (imagen corrupta).

El bootstrap es almacenado en la memoria ROM, pero copiado a la RAM para posteriormente ser ejecutado.

PASO 3: CARGA DE LOS FICHEROS DE CONFIGURACIÓN

Una vez ejecutado el sistema operativo se procede a la carga de los ficheros de configuración, los cuales definen el modo de operar y funcionalidades llevadas a cabo por el dispositivo. Para ello, IOS ejecuta las siguientes acciones:

- Primero, localiza el fichero *startup-config,* ubicado en la memoria NVRAM y compuesto por una serie de comandos que serán ejecutados de manera secuencial.

- Posteriormente hace una copia del mismo sobre el fichero *running-config,* ubicado en la memoria RAM y con el cual operará el dispositivo hasta que sea reiniciado.

Si el proceso concluye con éxito se accede directamente a la CLI y con ello a todas las funciones de IOS y el router. De lo contrario, el sistema iniciará un asistente de configuración guiada a través del siguiente mensaje en pantalla.

```
--- System Configuration Dialog ---

Continue with configuration dialog? [yes/no]:
```

Si la respuesta es no, IOS aplica la configuración de fábrica por defecto e inicia la CLI en modo usuario, a raíz de la cual se podrán configurar manualmente todas las funciones necesarias aplicando los comandos ya propuestos a lo largo de capítulos anteriores.

Si por el contrario la respuesta es sí, IOS guiará al usuario a través de un cuestionario con el fin de configurar automáticamente diferentes elementos del router.

En ambos casos se generará un nuevo fichero *running-config* que almacenará la secuencia de comandos introducidos.

El procedimiento de arranque definido es aplicable sobre routers Cisco. En switchs, aunque resulte prácticamente igual, difiere en algunos detalles

En resumen…

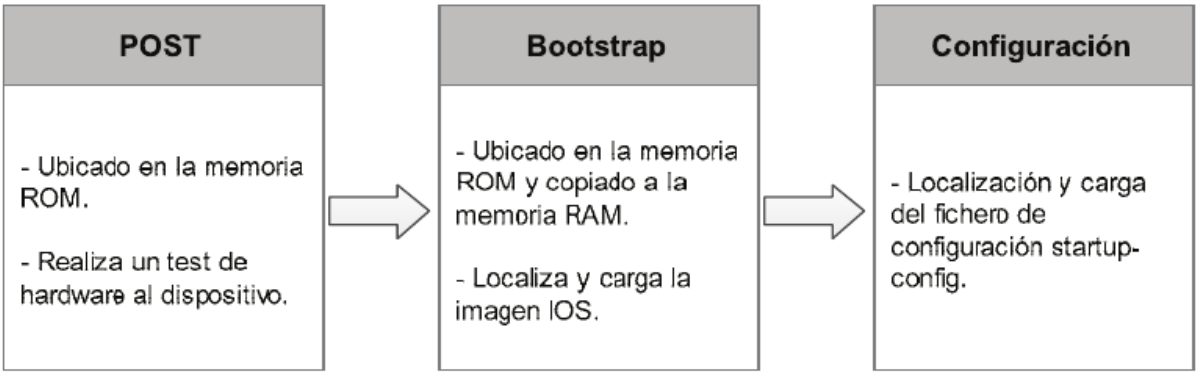


Fig. 12-10 Secuencia de arranque en routers Cisco.

Recuperación de contraseñas

La clave de acceso al modo privilegiado es aquella que otorga permisos de modificación sobre la configuración del sistema, siendo considerada un elemento crítico debido a su importancia, lo que conlleva que cuantas menos personas dispongan de la misma, mayor seguridad.

Este hecho puede desembocar en dos consecuencias, bien que la contraseña sea olvidada, o bien que no fuera compartida cuando se haga necesario, por ejemplo, si la red cambia de administradores. Sea como fuere, el acceso a la gestión del dispositivo resultará una tarea imposible, con el problema evidente que ello acarrea. Con el fin de solucionar este tipo de situaciones, Cisco dispone de un procedimiento de recuperación de contraseñas, mediante el cual se podrá modificar la clave y con ello tomar el control sobre su configuración. Este se basa en:

- *Paso 1:* Establecer una conexión física a través del puerto de consola y algún software de emulación de terminal como *"Putty".*

- *Paso 2:* Apagar y encender el dispositivo.

- *Paso 3:* Durante la carga de IOS, cancelar el proceso mediante la combinación de teclas "**Ctrl + Pausa**". Como consecuencia a dicha acción se forzará el inicio en modo ROMMON.

- *Paso 4:* Una vez en el mismo, aplicar el comando **confreg 0x2142**. Ello dará como resultado que durante el próximo reinicio se ignore el fichero *startup-config*, el cual incluye la clave de acceso al modo privilegiado.

- *Paso 5:* Resetear el dispositivo con el comando **reset**.

- *Paso 6:* IOS se ejecutará con normalidad, pero no cargará el *startup*, dando como resultado el acceso a la CLI a través del diálogo de configuración inicial, al cual se deberá responder **no.**

- *Paso 7:* Desde el modo usuario, acceder al modo privilegiado mediante la sentencia **enable**. No solicitará contraseña ya que no fue cargada durante el arranque.

- *Paso 8:* Ejecutar el comando **copy startup-config running-config** para recuperar toda la configuración que fue ignorada.

- *Paso 9:* Modificar la contraseña enable mediante la sentencia **enable secret *[nueva contraseña*]** o **enable password [*nueva contraseña*]**.

- *Paso 10:* Aplicar el comando **config-register 0x2102**, gracias al cual se volverá a cargar el fichero de configuración inicial durante el próximo reinicio.

- *Paso 11:* Por último, copiar la nueva configuración al *startup*, ejecutando un **copy running-config startup-config**.

- *Paso 12:* Reiniciar el dispositivo y acceder con la nueva clave.

Ejemplo: Ejecutar el procedimiento de recuperación de contraseña en el router TFE, aplicando como nueva clave la cadena “123456”.

```
Self decompressing the image :
############
monitor: command "boot" aborted due to user interrupt // PASO 3
rommon 1 > confreg 0x2142  // PASO 4
rommon 2 > reset  // PASO 5

System Bootstrap, Version 12.3(8r)T8, RELEASE SOFTWARE (fc1)
Cisco 1841 (revision 5.0) with 114688K/16384K bytes of memory.

Self decompressing the image :
############################################################################ [OK]
             Restricted Rights Legend


         --- System Configuration Dialog ---

Continue with configuration dialog? [yes/no]: no  //PASO 6


Press RETURN to get started!


Router>enable  //PASO 7
Router#copy startup-config running-config //PASO 8
Destination filename [running-config]?
```

```
486 bytes copied in 0.416 secs (1168 bytes/sec)
TFE#
%SYS-5-CONFIG_I: Configured from console by console

TFE#config t
Enter configuration commands, one per line.  End with CNTL/Z.
TFE(config)#enable secret 123456  //PASO 9
TFE(config)#config-register 0x2102  //PASO 10
TFE(config)#exit
TFE#
%SYS-5-CONFIG_I: Configured from console by console

TFE#copy running-config startup-config  //PASO 11
Destination filename [startup-config]?
Building configuration...
[OK]

TFE# reload  //PASO 12
Proceed with reload? [confirm]
System Bootstrap, Version 12.3(8r)T8, RELEASE SOFTWARE (fc1)
Cisco 1841 (revision 5.0) with 114688K/16384K bytes of memory.

Self decompressing the image :
############################################################################ [OK]
```

ADMINISTRACIÓN DE FICHEROS E IMÁGENES IOS

Una de las múltiples tareas a desarrollar como administrador consiste en la aplicación del IOS más adecuado en cada uno de los dispositivos Cisco que componen la topología. Esta decisión influye de manera directa sobre las capacidades y funciones disponibles, y con ello, sobre el rendimiento y optimización de la red. Su elección se considera sumamente importante, por lo que resulta necesario abarcar las características y gestión de este tipo de ficheros, dividiendo la presente sección en dos apartados dedicados a ello:

- Gestión de imágenes IOS.
- Gestión de licencias IOS.

Gestión de imágenes IOS

Una imagen IOS simplemente identifica un fichero con extensión *.bin*, el cual almacena el sistema operativo que debe ser cargado en el dispositivo. La gestión y desarrollo de estos ha sufrido diferentes variaciones a lo largo del tiempo, pudiendo ser dividido en dos modelos.

El primero, actualmente en desuso, se basaba en la creación de una imagen para cada serie de routers o switchs, que a su vez era dividida en versiones, cada una de ellas con funciones específicas como seguridad, VoIP, multiprotocolo, etc. De tal manera que un mismo dispositivo disponía de diferentes IOS, siendo el administrador quien debía seleccionar cuál de ellas aplicar.

Sin embargo, dicho modelo ha sido discontinuado y sustituido por un sistema de gestión basado en una sola imagen universal que incorpora todas las funciones dedicadas que antes se implementaban por separado, logrando gracias a ello una gestión más sencilla y eficiente, tanto para su administración como para su actualización.

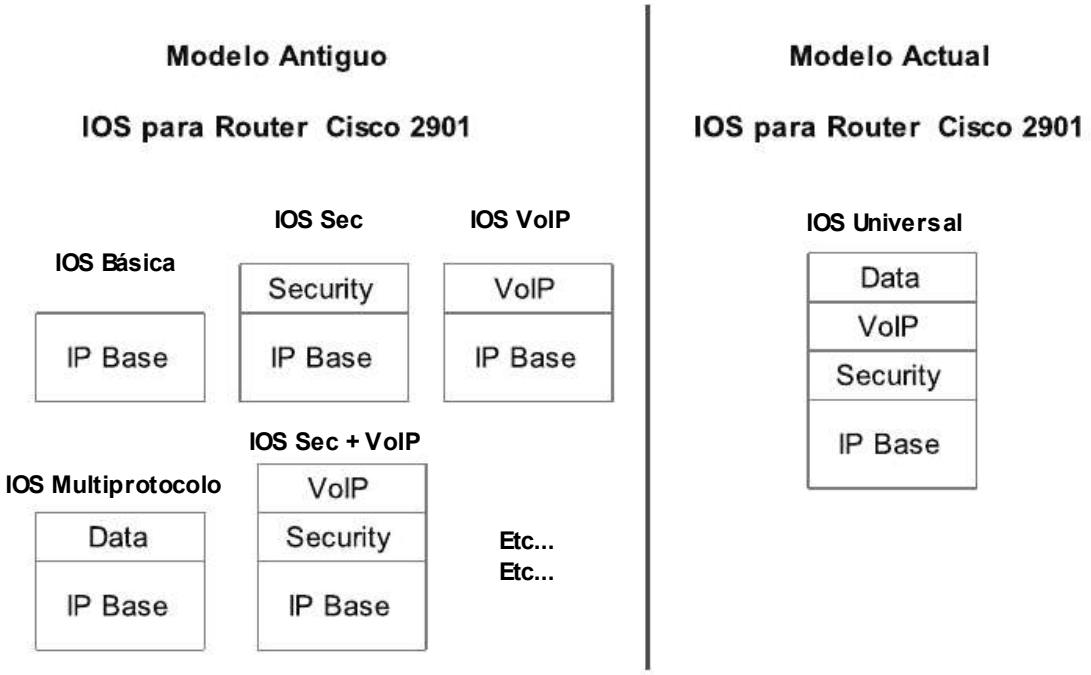


Fig. 12-11 Modelos de imágenes IOS.

"*IP Base*" dispone de las funciones básicas de configuración necesarias en cualquier dispositivo, es por ello que forma parte de todas las versiones de IOS, incluso en el modelo antiguo. Conforme al mismo es posible agregar diferentes paquetes con características específicas, entre los que se encuentran:

- *securityk9:* Dedicado a la seguridad, incluye firewall, IPS, IPSec, 3DES, VPN, etc.

- *datak9:* Específico para el tratamiento de datos, como entornos multiprotocolo, ATM, MPLS, soporte IBM, etc.

- *uck9:* Compuesto por tecnologías y protocolos necesarios para ofrecer soporte VoIP.

Actualmente los dispositivos incorporan la imagen universal por defecto, resultando innecesaria su instalación de manera manual. Sin embargo, cuando Cisco publica alguna actualización de IOS, su descarga y aplicación depende por completo del administrador, quien deberá llevar a cabo el procedimiento necesario para su instalación, el cual varía en relación con la ubicación del fichero, pudiendo identificar un servidor TFTP o la memoria flash.

ACTUALIZACIÓN DE IOS UBICADA EN TFTP

En este caso se deberán llevar a cabo las siguientes acciones:

- *Paso 1:* Descargar la nueva imagen desde la web de Cisco (software.cisco.com).

- *Paso 2:* Copiarla al servidor TFTP.

- *Paso 3:* Configurar el dispositivo para que proceda a su carga, ejecutando para ello el comando **boot system tftp *[imagen] [ip servidor TFTP]***, donde *imagen* identifica el nombre exacto y extensión del fichero descargado.

- *Paso 4:* Reiniciar el dispositivo.

Ejemplo:

```
Router(config)# boot system tftp c800-universalk9-mz.SPA.153-3.M5.bin
10.10.10.1
```

Durante el próximo reinicio el router buscará la imagen con nombre "c800-universalk9-mz.SPA.153-3.M5.bin" en el servidor 10.10.10.1, para acto seguido ejecutarla en el dispositivo. Si no fuera localizada, se iniciará en modo ROMMON.

ACTUALIZACIÓN DE IOS UBICADA EN LA MEMORIA FLASH

La memoria flash es la unidad de almacenamiento por defecto para imágenes IOS, siendo lo más habitual su ejecución desde la misma. El proceso de actualización en este caso consta de los siguientes pasos:

- *Paso 1:* Descargar la nueva imagen desde la web de Cisco (software.cisco.com).

- *Paso 2:* Almacenarla en un servidor TFTP.

- *Paso 3:* Copiarla a la memoria flash, a través del comando **copy tftp flash**. El proceso mostrará un diálogo que solicitará la IP del servidor y el nombre del fichero.

- *Paso 4:* Aplicar la carga del nuevo IOS mediante la sentencia **boot system flash:*[nombre de imagen]***.

- *Paso 5:* Reiniciar el dispositivo.

Ejemplo:

```
Router#copy tftp flash
Address or name of remote host [ ]? 10.10.10.1
Source filename [ ]?c800-universalk9-mz.SPA.153-3.M5.bin
Destination filename[c800-universalk9-mz.SPA.153-3.M5.bin]?
Accessing tftp://10.10.10.1/c800-universalk9-mz.SPA.153-3.M5.bin...
Loading    c800-universalk9-mz.SPA.153-3.M5.bin    from    10.10.10.1    (via
FastEthernet0/1):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 64202786 bytes]

Router# show flash
-#- --length-- -----date/time------ path
1 74503236 Mar 11 2015 21:05:33 +00:00 c800-universalk9-mz.SPA.153-3.M3.bin
2 84789908 Mar 12 2015 11:53:02 +00:00 c800-universalk9-mz.SPA.153-3.M4.bin
3 64202786 Mar 13 2015 07:30:09 +00:00 c800-universalk9-mz.SPA.153-3.M5.bin

Router# config t
Router(config)# boot system flash:c800-universalk9-mz.SPA.153-3.M5.bin
```

Gestión de licencias IOS

Aunque las IOS universales agrupen todos los paquetes en la misma imagen, resulta necesaria la activación de cada uno de ellos para disponer de las características específicas que ofrecen. Este proceso se lleva a cabo mediante la adquisición de licencias y posterior aplicación en el dispositivo, de tal manera que:

ADQUISICIÓN DE LICENCIAS

El proceso de compra se divide en los siguientes pasos:

- *Paso 1:* Ejecutar el comando **show license udi**, gracias al cual se obtiene el PID *(Product ID)*, SN *(Serial Number)* y UDI *(Unique Device Identifier)* del dispositivo.

- *Paso 2:* Acceder a la web www.cisco.com/go/license (se requiere cuenta de usuario).

- *Paso 3:* Completar el formulario, incluyendo en el campo UDI el valor obtenido en el paso 1.

- *Paso 4:* Seleccionar el PAK *(Product Authorization Key)* necesario, el cual define el tipo de funciones que se agregarán al dispositivo.

- *Paso 5:* Una vez abonado el monto correspondiente se mostrará la licencia, que deberá ser descargada o copiada a un fichero.

ACTIVACIÓN DE LA LICENCIA

Una vez adquirida bastará con activarla para poder hacer uso del PAK en cuestión:

- *Paso 1:* Copiar el fichero que contiene la licencia en alguna ubicación accesible desde el dispositivo, como un servidor TFTP, FTP o una memoria USB.

- *Paso 2:* Ejecutar el comando **license install *[ruta fichero]*** desde el modo privilegiado, por ejemplo, "*license install tftp://10.10.10.1/licencia_router.lic*".

- *Paso 3:* Reiniciar el dispositivo.

Tras ello, el router dispondrá de las nuevas funciones.

QOS - CONCEPTOS BÁSICOS

A día de hoy, prácticamente la totalidad de aplicaciones generan tráfico de red, ya sea de manera ininterrumpida o puntualmente para determinadas operaciones u actualizaciones. Sea como fuere, la comunicación debe ser procesada por los dispositivos intermediaros ubicados entre el origen y destino, pudiendo acaparar excesivo ancho de banda y con ello retrasos en las comunicaciones. En este sentido, el método de reenvío de paquetes y tramas llevado a cabo por defecto tanto en routers como switchs consiste en un sistema denominado FIFO *(first in, first out)* donde los datos son almacenados en un *buffer*, para posteriormente reenviarlos en el mismo orden en que han sido recibidos, es decir, no se aplica ningún tipo de prioridad sobre los mismos. Este hecho implica que FIFO no sea considerado el modelo más adecuado a implementar sobre entornos corporativos, ya que la productividad de cada compañía se basa en un tráfico específico para ello. Por

ejemplo, una empresa dedicada a la venta de productos vía web, requiere priorizar el tráfico de pedidos antes que cualquier otro.

En estos casos, la solución ideal consiste en aplicar políticas QoS *(Quality of Service)* gracias a las cuales el administrador de la red puede definir qué tráfico será tratado con prioridad y con ello procesado antes que el resto, logrando una comunicación más fiable, eficiente y centrada en los objetivos propios de cada compañía. Además, resulta especialmente útil sobre entornos o dispositivos susceptibles a congestión, siendo el siguiente escenario un claro ejemplo de ello:

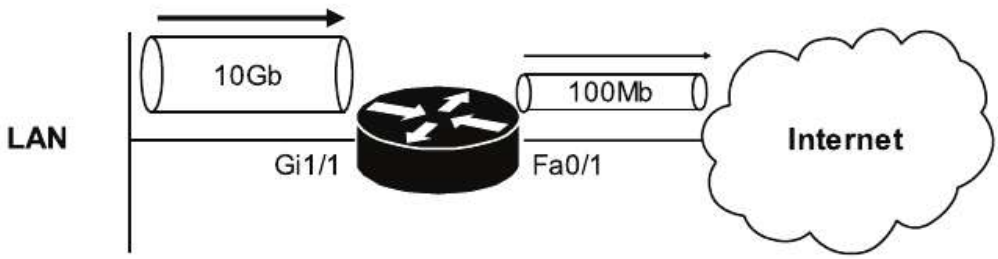


Fig. 12-12 Enlace (Fa0/1) propenso a congestión.

Donde la interfaz LAN opera a 10 Gb mientras que la salida a Internet a 100 Mb. Tal diferencia de velocidad propicia que ante elevados picos de tráfico, Fa0/1 no pueda procesar la totalidad de paquetes recibidos desde Gi1/1 a una velocidad aceptable, congestionando el enlace e impactando seriamente sobre los cuatro factores que determinan el rendimiento de toda red: Ancho de banda, Retraso, Jitter y Pérdida de paquetes.

- *Ancho de banda:* Como ya se ha mencionado en diferentes capítulos, el ancho de banda *(bandwidth)* hace referencia a la velocidad máxima de transmisión, en bits por segundo, de un determinado enlace.

- *Retraso unidireccional (One-way Delay)*: Indica el tiempo que tarda un paquete en llegar desde el origen hasta el destino de la comunicación.

- *Retraso de ida y vuelta (Round-trip Delay):* El tiempo que tarda un paquete desde que es enviado por el origen hasta que es recibida la respuesta desde el destino.

- *Jitter*: Hace referencia a la variación de retraso sufrido por paquetes generados para una misma comunicación entre un mismo origen y destino. Por ejemplo, un *ping* cuyo primer paquete tarda 5 ms, el segundo, 3 ms, el tercero, 6 ms, etc. La diferencia de tiempo entre todos ellos es denominada *jitter*.

- *Pérdida de paquetes:* Indica la totalidad de paquetes no recibidos por el destinatario durante una comunicación, siendo representado normalmente mediante porcentaje. Su impacto en TCP resulta menor que en UDP, ya que, gracias al control de errores, los paquetes perdidos son reenviados.

Una de las grandes ventajas de QoS es que permite gestionar el ancho de banda de un determinado enlace, mejorando con ello los resultados de la totalidad de factores recién descritos y logrando una comunicación fiable y segura de tráfico prioritario. Su modo de operar puede ser dividido en dos, primero, la clasificación e identificación de dicho tráfico, y segundo, la gestión de su envío.

Clasificación e identificación de tráfico

El primer detalle a tener en cuenta es que QoS puede operar tanto en capa 2 como 3, por lo tanto, su configuración resulta posible sobre diferentes tipos de dispositivo, como switchs y routers. Una vez implementado, la primera acción que lleva a cabo consiste en analizar el tráfico y marcar aquel que coincida con las políticas aplicadas. Dicho proceso simplemente se basa en asignar un valor de prioridad a un determinado campo ubicado en la cabecera del paquete o trama, el cual puede variar dependiendo del protocolo sobre el cual opera el enlace, siendo varios ejemplos 802.1Q, 802.11, IPv4 o IPv6, entre otros. Tras ello, el tráfico es clasificado, lo cual consiste en ordenar los paquetes en relación con la prioridad de cada uno de ellos, con el fin de gestionar su posterior envío. Un ejemplo podría ser el siguiente.

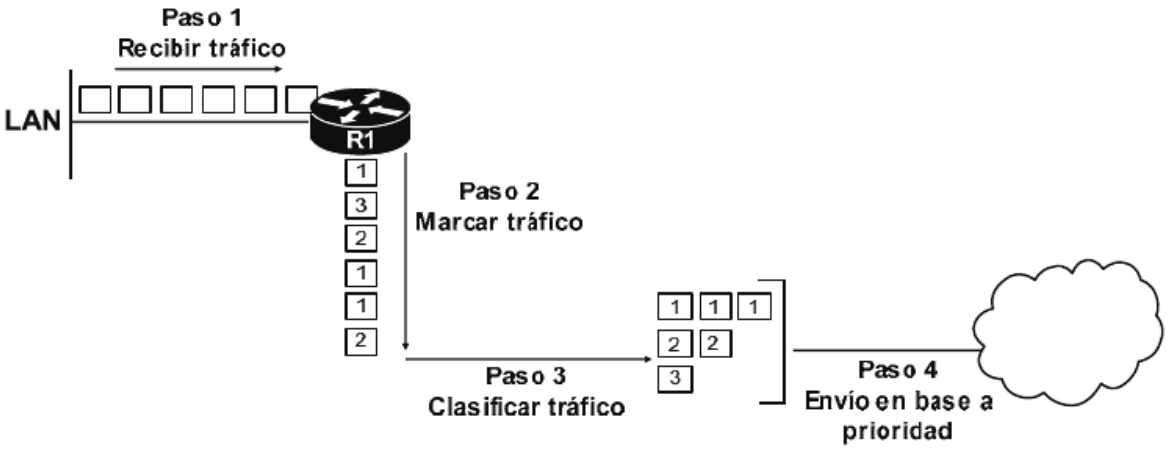


Fig. 12-13 Modo de operar de QoS.

Sobre R1 se han aplicado políticas QoS, de tal manera que:

- *Paso 1:* R1 recibe tráfico a través de la interfaz que conecta directamente con la LAN.

- *Paso 2:* QoS lo analiza y establece la prioridad de cada paquete conforme a las políticas definidas en el dispositivo.

- *Paso 3:* Tras ello, es clasificado, ordenándolo con relación a dicha prioridad.

- *Paso 4:* Se ejecuta el proceso de envío, el cual es gestionado por QoS. Dicho sistema será analizado en la próxima sección de este mismo capítulo.

Como se ha mencionado antes, QoS puede operar tanto en capa 2 como 3, sin embargo, aunque el propósito coincida, cada protocolo establece un determinado campo en su cabecera con el fin de definir la prioridad, siendo los más comunes los siguientes:

Protocolo	**Campo**	**Longitud**	**Capa**
IPv4, IPv6	DSCP	6 bits	Capa 3
IPv4, IPv6	IPP	3 bits	Capa 3
802.1Q	CoS	3 bits	Capa 2
802.11	TID	3 bits	Capa 2

¿Por qué IPv4 e IPv6 disponen de dos campos, DSCP e IPP, para el mismo propósito? A lo largo del tiempo, el protocolo IP ha sufrido diferentes variaciones con el fin de mejorar el servicio y calidad del mismo. Una de ellas corresponde al campo IPP, el cual, originariamente era la única opción disponible en capa 3 para marcar prioridad QoS. Sin embargo, el constante aumento de servicios y necesidades en esta capa trajo consigo que los 3 bits destinados a ello resultaran insuficientes. La solución nace con DCSP, el cual habilita 6 bits a tal propósito, aumentando las opciones de manera considerable. Actualmente, ambos campos forman parte de IPv4 e IPv6, siendo la opción más común DSCP.

CAMPO COS EN 80.2.1Q

802.1Q hace referencia al protocolo en capa 2 necesario sobre enlaces troncales, el cual permite el transporte de múltiples VLANs. Este fue objeto de estudio en el capítulo 2 *"Configuración de Switchs Cisco"*, analizando, entre otros, el formato de cabecera que agrega sobre la trama Ethernet, resultando ser el siguiente:

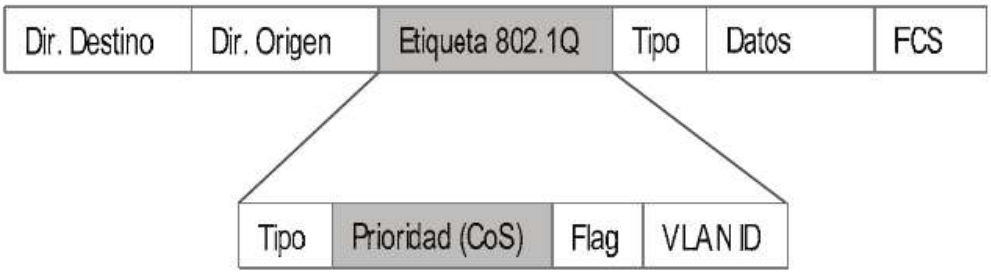


Fig. 12-14 Campo CoS en trama 802.1Q.

En dicho capítulo se prestó especial atención sobre el campo "VLAN ID", sin embargo, en esta ocasión, interesa analizar la prioridad (CoS).

CoS *(Class of Service)*, también denominado PRI *(User Priority)* hace referencia al campo destinado en 802.1Q para definir una determinada prioridad a la trama. Este tiene una longitud de 3 bits, por lo que tan solo permite un total de 8 valores, siendo los siguientes:

Valor Binario	Valor decimal	Prioridad
000	0	*Best Effor Data*
001	1	*Bulk Data*
010	2	*Critical Data*
011	3	*Call Signaling*
100	4	*Video*
101	5	*Voice*
110	6	*Routing*
111	7	*Reserved*

CAMPOS IPP Y DSCP EN IPV4

En IPv4, el formato de paquete incluye una cabecera denominada ToS *(Type of Service)*, con longitud de 8 bits, siendo reservados dos de ellos para control de flujo. La misma está compuesta por los campos IPP *(IP Precedence)* y DSCP *(Differentiated Service Code Point)*, a través de los cuales se podrá definir la prioridad del paquete en cuestión.

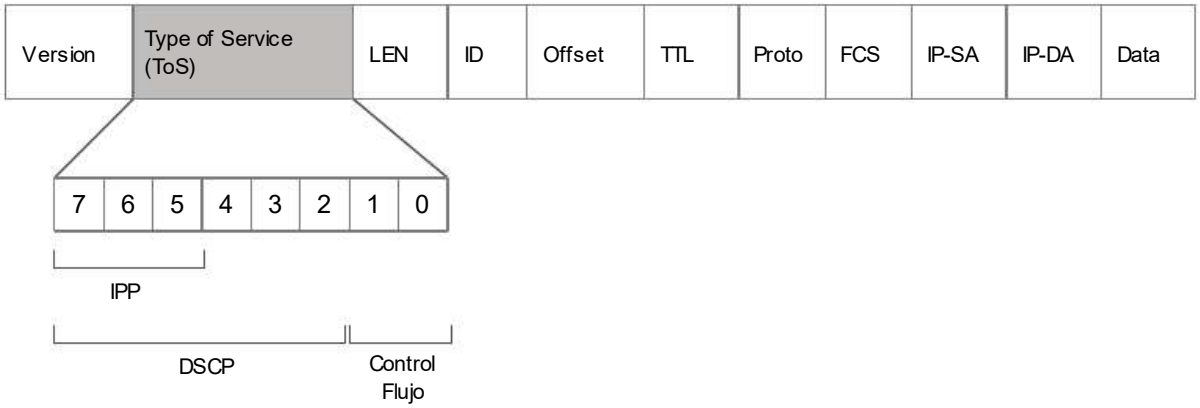
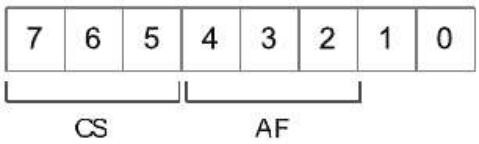


Fig. 12-15 Campo Type of Service en IPv4.

QoS suele operar con IPP en dispositivos anticuados, poco comunes sobre entornos corporativos. Este hace uso de los 3 primeros bits de la cabecera ToS, lo que habilita un total de 8 tipos de prioridad, siendo las siguientes:

Valor Binario	Valor decimal	Prioridad
000	0	*Routine*
001	1	*Priority*
010	2	*Inmediate*
011	3	*Flash*
100	4	*Flash Override*
101	5	*Critic/ECP*
110	6	*Internetwork Control*
111	7	*Network Control*

El otro campo en cuestión, DSCP, se basa en el valor de estos 3 primeros bits, a los que denomina CS *(Class Selector)* para agregar determinadas características sobre su prioridad, haciendo uso para ello de los 3 bits restantes, denominados AF *(Assured Forwarding)*.



DSCP habilita multitud de opciones y su estudio en detalle resulta complejo. Es por ello que no es contemplado en el temario de CCNA, siendo incluido en certificaciones más avanzadas.

CISCO NBAR

QoS mantiene cierta semejanza con las ACLs en cuanto a modo de operar se refiere. En este aspecto, ambos analizan tráfico y aplican una determinada acción sobre aquel que coincida con alguna de las políticas definidas. Sin embargo, en muchas ocasiones resulta complejo establecer los criterios adecuados para identificar un flujo específico. Por ejemplo, ¿cómo diferenciar el tráfico generado por aplicaciones de *streaming* de vídeo como webcams, sistemas de vídeo vigilancia o portales tipo YouTube?

Con el fin de facilitar dicha tarea, Cisco dispone de la herramienta NBAR *(Network Based Application Recognition)* gracias a la cual el dispositivo detecta de manera automática un amplio abanico de aplicaciones con relación al flujo de datos generado por las mismas. Actualmente su desarrollo se encuentra en la versión 2, NBAR2, también denominado *"next-generation NBAR"*, capaz de analizar y diferenciar entre alrededor de mil tipos de tráfico diferente, divididos a su vez en las siguientes categorías.

Categorías de tráfico disponibles en NBAR	
browsing	*business-and-productivity-tools*
email	*file-sharing*
gaming	*industrial-protocols*
instant-messaging	*internet-privacy*
layer3-over-ip	*location-based-services*
net-admin	*newsgroup*
social-networking	*streaming*
voice-and-video	

El proceso de configuración tanto de QoS como de NBAR no forma parte del contenido de CCNA.

Gestión de envío

Una vez clasificado el tráfico, QoS procede a su envío en relación con las prioridades establecidas. En párrafos anteriores se ha mencionado que el método por defecto aplicado tanto en routers como switchs consiste en el modelo FIFO *(first in, first out)* donde los paquetes recibidos son almacenados en un *buffer* para su posterior envío en el mismo orden en que han sido recibidos. Realmente este sistema habilita una cola de espera *(queuing)* de la siguiente manera:

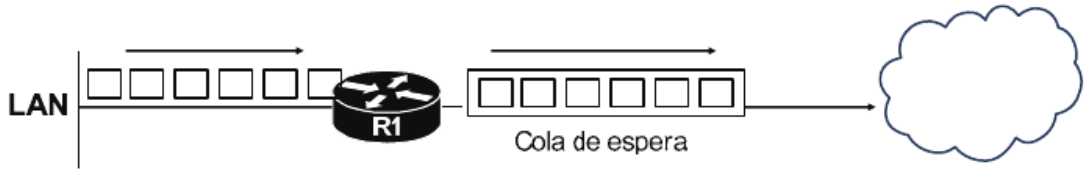


Fig. 12-16 Sistema de cola de envío en FIFO.

QoS también basa su modelo en la gestión de este tipo de colas, sin embargo, en lugar de una, habilita cuantas fueran necesarias para ubicar aquellos paquetes con diferente prioridad en diferentes listas de espera, de tal manera que:

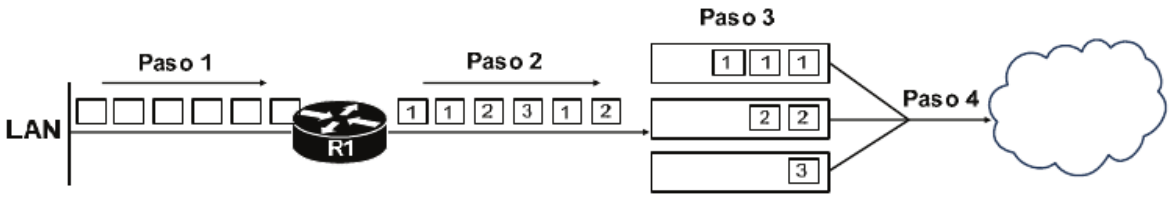


Fig. 12-17 Sistema de colas de envío en QoS.

- *Paso 1:* R1 recibe tráfico.

- *Paso 2:* Establece la prioridad de cada paquete conforme a las políticas QoS definidas en el dispositivo.

- *Paso 3:* Lo clasifica, habilitando diferentes colas para ubicar los paquetes con la misma prioridad de manera conjunta.

- *Paso 4:* Procede a su envío.

Dicho envío, en dispositivos Cisco, se basa en un sistema denominado *Round Robin,* el cual establece un determinado factor que será ejecutado de manera secuencial sobre cada una de las colas habilitadas. Por ejemplo, si como factor se toma un número de bytes, y a la cola 1 se le debe otorgar mayor prioridad que a cualquier otra, se podría establecer un total de 10000 bytes sobre la misma, 5000 sobre la cola 2 y 1500 sobre la cola 3. De esta manera, QoS primero enviará 10000 bytes de la cola 1, luego continuará con 5000 bytes de la cola 2 y por último 1500 de la 3, repitiendo el proceso de manera continua.

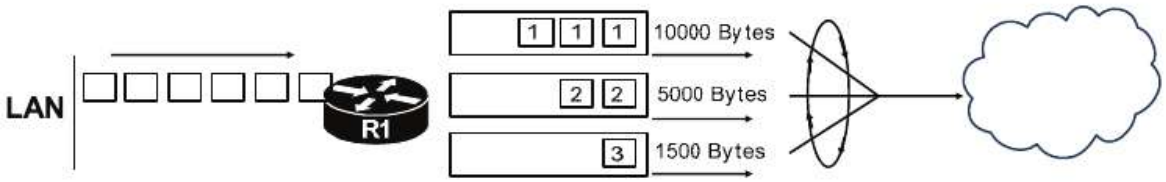


Fig. 12-18 Modelo Round-Robin basado en total de bytes.

Pudiendo también ser implementado mediante la reserva de un determinado ancho de banda para cada cola.

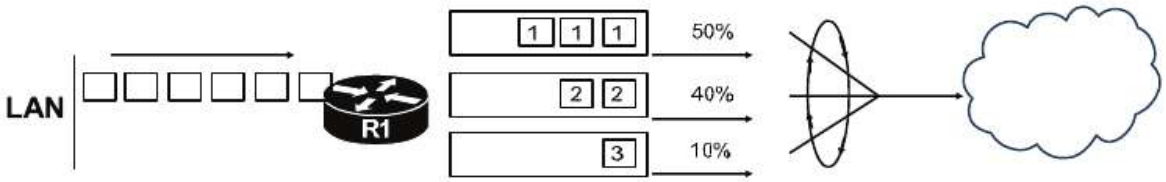


Fig. 12-19 Modelo Round-Robin basado en ancho de banda.

Ambos sistemas resultan eficaces sobre enlaces congestionados, sin embargo, no es del todo útil si el tráfico a tratar fuera de voz o vídeo, es decir, en tiempo real. En estos casos se requiere el envío inmediato del mismo, de lo contrario se producirá una elevada latencia y *jitter*, derivando en la pérdida de paquetes y una baja calidad de experiencia para el usuario final. La solución a ello consiste en implementar LLQ *(Low Latency Queuing),* la cual simplemente establece una prioridad especial sobre determinado tráfico, habilitando su envío inmediato sin ejecutar sobre el mismo la lógica *Round Robin.*

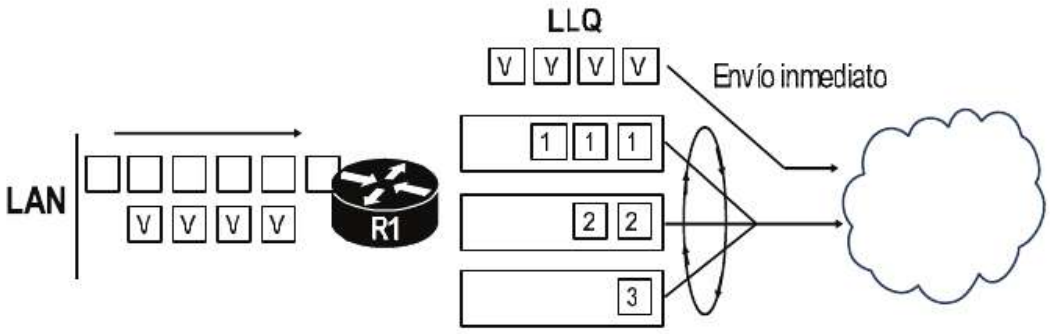


Fig. 12-20 Modelo LLQ sobre Round-Robin.

Por último, QoS va estrechamente ligado a QoE *(Quality of Experience)*, el cual mide la calidad del servicio experimentada desde el punto de vista del usuario final, es decir, el nivel de satisfacción de este a la hora de trabajar con todo aquello que suponga tráfico de red. Un ejemplo muy claro de QoE se encuentra en los sistemas de telefonía IP (VoIP). Este, al tratarse de un servicio en tiempo real y su vez recibido directamente por el usuario en formato comprensible (voz), permite evaluarlo de manera inmediata. En este aspecto, una buena práctica sobre entornos corporativos consiste en aplicar LLQ sobre el tráfico VoIP.

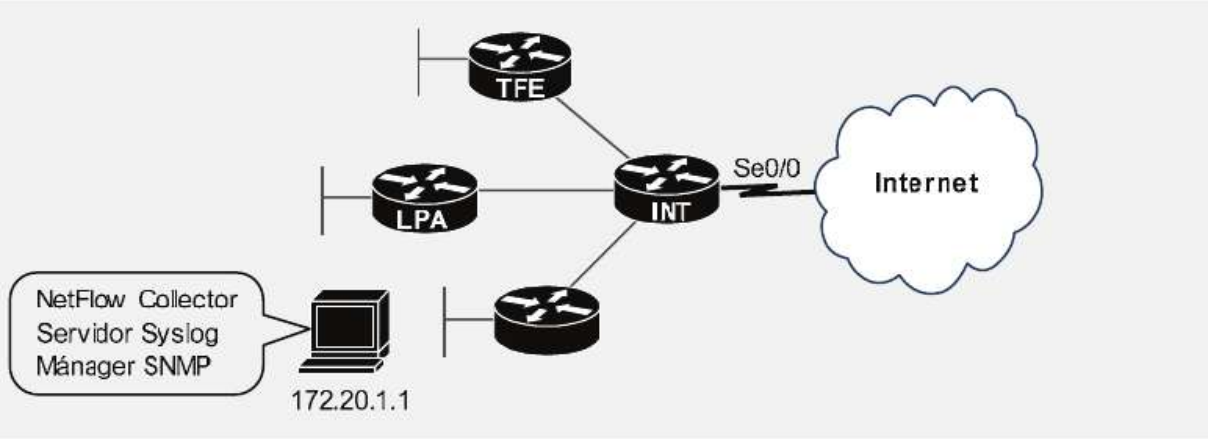
SOLUCIÓN DE RETOS: GESTIÓN DE IOS

Reto 12.1 – Configurar la siguiente topología de tal manera que:

- TFE envíe al servidor syslog los eventos ocurridos de nivel 4 o inferior. También se debe habilitar el servicio *timestamps*.

- LPA haga uso de SNMP, con autenticación “P4Ss” y permisos de lectura y escritura al que solo podrá acceder el mánager, con IP 172.20.1.1.

- INT monitorice el tráfico de entrada y salida hacia Internet. El *Netflow Collector* opera en el puerto UDP 4000 y la versión que se debe aplicar es la 9, configurando una interfaz loopback cuya IP (172.30.10.10/24) actuará como origen de la comunicación hacia el servidor.




```
---Configuración en TFE---
TFE(config)# logging 172.20.1.1
TFE(config)# logging trap 4
TFE(config)# service timestamps log datetime msec

---Configuración en LPA---
LPA(config)# ip access-list standard ACCESO_SNMP
LPA(config-std-nacl)# permit host 172.20.1.1
LPA(config-std-nacl)# exit
LPA(config)# snmp-server community P4Ss RW ACCESO_SNMP

---Configuración en INT---
INT(config)# interface Se0/0
INT(config-if)# ip flow ingress
INT(config-if)# ip flow egress
INT(config-if)# exit
INT(config)# interface loopback 0
INT(config-if)# ip address  172.30.10.10 255.255.255.0
INT(config-if)# no shutdown
INT(config-if)# exit
INT(config)# ip flow-export destination 172.20.1.1 4000
INT(config)# ip flow-export source loopback 0
INT(config)# ip flow-export version 9
```

El nombre de la ACL en LPA puede variar, al igual que el número de interfaz loopback configurado en el router INT.

TEST CAPÍTULO 12: GESTIÓN DE IOS

1.- ¿Cuál de los siguientes protocolos permite filtrar y almacenar registros conforme al tipo de evento ocurrido?

A. SNMP.
B. Syslog.
C. SFTP.
D. NetFlow.

2.- ¿Cuál de los siguientes protocolos envía alertas directamente al administrador cuando se produce algún evento que requiere intervención?

A. SNMP.
B. Syslog.
C. SFTP.
D. NetFlow.
E. Ninguna de las anteriores.

3.- En un router configurado con syslog se ha introducido el comando "*logging trap 4*". ¿Cuál de las siguientes afirmaciones al respecto resulta la más correcta?

A. El mánager enviará al agente eventos de nivel 4,5,6 y 7.
B. El servidor enviará al cliente eventos de nivel 4,5,6 y 7.
C. El agente enviará al mánager eventos de nivel 4, 3, 2, 1 y 0.
D. El servidor enviará al cliente eventos de nivel 4, 3, 2, 1 y 0.
E. El cliente enviará al servidor eventos de nivel 4.
F. Ninguna de las anteriores es correcta.

4.- Un administrador de red desea monitorizar y a su vez bloquear paquetes sospechosos recibidos a través de la interfaz que conecta con Internet. ¿Qué protocolo deberá aplicar para lograr dicho propósito?

A. NetFlow.
B. SNMP.
C. Syslog.
D. NetFlow para el bloqueo de paquetes y SNMP para la monitorización.
E. Ninguna de las anteriores es correcta.

5.- ¿En qué modo se inicia un router en caso de error durante la secuencia de arranque?

A. Modo enable.
B. Modo monitor.

C. Modo de configuración.
D. Modo de recuperación.

6.- ¿Cuál de los siguientes protocolos permite monitorizar el tráfico de una determinada interfaz?

A. SNMP.
B. Syslog.
C. SFTP.
D. NetFlow.
E. Ninguna de las anteriores.

7.- ¿A qué secuencia de arranque obedecen los dispositivos Cisco?

A. POST, ficheros de configuración, bootstrap.
B. Bootstrap, POST, ficheros de configuración.
C. POST, bootstrap, ficheros de configuración.
D. Ficheros de configuración, POST, bootstrap.

8.- ¿En qué memoria del dispositivo es ubicada la imagen IOS?

A. RAM.
B. ROM.
C. Flash.
D. NVRAM.

9.- ¿Cuál de los siguientes protocolos basa su modo de operación en la MIB?

A. SNMP.
B. Syslog.
C. SFTP.
D. NetFlow.

10.- Un administrador requiere ejecutar el proceso de recuperación de contraseñas en un router Cisco. ¿Cómo deberá proceder?

A. Estableciendo una conexión mediante telnet.
B. Estableciendo una conexión mediante SSH.
C. Estableciendo una conexión mediante cable de consola.
D. Todas las anteriores son correctas.

11.- La clave de acceso a un router ha sido modificada mediante el proceso de recuperación de contraseñas, sin embargo, el administrador ha olvidado restablecer el valor 0x2102 en el registro de configuración. ¿Qué ocurrirá durante el próximo reinicio?

A. El router solicitará la ubicación del fichero *startup-config*.
B. Se iniciará en modo ROMMON.
C. Se cargará el fichero *startup-config*.
D. Se mostrará el diálogo de configuración inicial.

12.- ¿Cuál de los siguientes protocolos de monitorización se basa en el almacenamiento de logs?

A. SNMP.
B. Syslog.
C. SFTP.
D. NetFlow.

13.- Un agente SNMP con permisos RW otorga al mánager funciones de…

A. Monitorización y autocorrección.
B. Monitorización y gestión de las diferentes MIBs.
C. Monitorización y configuración.
D. Monitorización y auditoría.

14.- ¿Cuál es la función del POST?

A. Localiza y carga la imagen IOS.
B. Localiza y carga los ficheros de configuración.
C. Localiza la memoria disponible y la utiliza como almacenamiento de IOS.
D. Realiza un test de hardware al dispositivo.

15.- Tras examinar la configuración de un router, se comprueba que dispone de un registro con valor 0x2102. ¿Qué acción llevará a cabo cuando sea reiniciado?

A. Ignorará el fichero *startup-config* y mostrará un diálogo de configuración inicial.
B. Cargará el fichero *startup-config* y aplicará la configuración incluida en el mismo.
C. Iniciará el router en modo ROMMON.
D. Ninguna de las anteriores.

16.- De los siguientes campos, ¿cuál es utilizado por QoS para marcar tráfico sobre enlaces 802.1Q?

A. IPP.
B. ToS.
C. DSCP.
D. Ninguno de los anteriores.

17.- ¿Cuáles de los siguientes factores presentan mejores resultados gracias a la aplicación de QoS? (Seleccionar dos respuestas)

A. Jitter.
B. Ancho de banda.
C. Retraso.
D. Seguridad.

18.- ¿De cuántos bits dispone el campo DSCP?

A. 3
B. 6
C. 8
D. 2

APÉNDICE

SOLUCIÓN DE TESTS

Capítulo 1: Redes informáticas. Conceptos básicos

1. D
2. A
3. B
4. B,E
5. C
6. C
7. C
8. A
9. B
10. A
11. B,D
12. B
13. C
14. B
15. D
16. B
17. A
18. C
19. B
20. B

21. C
22. D
23. E
24. A
25. A
26. D
27. B
28. A
29. C,D
30. C
31. B
32. D
33. B
34. A,B
35. C
36. A,D
37. F,G,H
38. C,F
39. C
40. A
41. D
42. C
43. A,C
44. E
45. D,E
46. D,E
47. C
48. D,H
49. G

Capítulo 2: Configuración de switchs Cisco

1. C
2. A
3. C
4. C
5. B
6. D
7. A

8. A
9. B
10. B
11. E
12. A
13. D
14. B
15. A
16. B
17. E
18. C
19. D
20. A
21. B,D,F
22. A
23. D
24. D
25. C,D
26. D
27. A
28. E
29. D
30. E
31. B
32. C
33. D
34. A
35. C
36. E
37. B

Capítulo 3: Spanning Tree Protocol

1. B
2. D
3. D
4. C,D
5. A,E
6. D

7. B
8. A
9. B,E
10. D
11. E
12. C
13. D
14. C
15. B,C
16. C
17. C
18. A
19. A
20. C
21. A,C,E
22. C
23. B

Capítulo 4: Subnetting en IPv4

1. C,D
2. B
3. D
4. B
5. A
6. B
7. C
8. B
9. E
10. B
11. A
12. B
13. B
14. D
15. D
16. A
17. B,E
18. A
19. B

20. A,D
21. D
22. C
23. D
24. D
25. A
26. A

Capítulo 5: Configuración inicial de routers Cisco

1. D
2. B
3. A,B,C
4. C,E
5. D
6. B
7. A
8. D
9. D
10. C
11. B,D
12. B
13. A
14. D
15. A,D
16. D
17. D
18. C
19. A

Capítulo 6: Protocolos de enrutamiento

1. D
2. E
3. B
4. A
5. B
6. A

7. D
8. A
9. D
10. C
11. A,C
12. C
13. A
14. D
15. C
16. C
17. A
18. B,D
19. B
20. D
21. A,D,E
22. A
23. B
24. B
25. B
26. D
27. B
28. B,D
29. A
30. D
31. C
32. B
33. A

Capítulo 7: Seguridad en capa 3

1. B
2. A
3. C
4. A
5. A
6. A,C
7. C,D
8. C
9. A

10. D
11. B
12. C
13. C
14. D
15. A,C
16. A,F
17. B
18. D
19. B
20. C
21. C
22. A
23. D
24. D
25. A

Capítulo 8: Redundancia en puertas de enlace

1. B
2. A
3. B
4. D
5. C
6. B
7. A
8. D
9. C
10. D
11. D
12. C
13. E
14. B,C
15. C
16. C
17. A
18. D
19. E
20. C

Capítulo 9: Redes privadas virtuales

1. B,C
2. A
3. C
4. D
5. B
6. B
7. A
8. B
9. D
10. C
11. A,D
12. C
13. B,C
14. D
15. E

Capítulo 10: Redes WAN. Tipos y protocolos

1. B
2. A,B
3. C
4. B,E
5. C
6. D
7. D
8. A,B
9. D
10. C
11. C
12. C
13. B
14. C
15. D
16. C
17. B
18. A

19. D
20. C
21. A,E
22. C
23. B
24. B
25. C
26. D
27. A
28. B
29. B
30. B,D
31. D

Capítulo 11: IP versión 6

1. B,E
2. C
3. A,D
4. C
5. B
6. A
7. C
8. C
9. D
10. E
11. D
12. B
13. D
14. C
15. B
16. A,B
17. C
18. E
19. D
20. B
21. D
22. A
23. D

24. B,C
25. A,C,F
26. C
27. D
28. D

Capítulo 12: Gestión de IOS

1. B
2. A
3. F
4. E
5. B
6. D
7. C
8. C
9. A
10. C
11. D
12. B
13. C
14. D
15. B
16. D
17. A,C
18. B

ÍNDICE ANALÍTICO

D

E

F

G

H

I

L

LI

M

N

O

P

Q

R

S

T

U

V

W