UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

"CONSTRUCCIÓN DE REDES PRIVADAS VIRTUALES USANDO MPLS"

T E S I S

QUE PARA OBTENER EL TITULO DE:

INGENIERO EN TELECOMUNICACIONES

P R E S E N T A N :

SERGIO GOMEZ ESCARCEGA

HELIA LUCERO GONZALEZ ORDAZ

DIRECTOR DE TESIS: ING. RODOLFO ARIAS VILLAVICENCIO

FEBRERO DEL 2003

Estructura de la tesis

En esta tesis se realiza una investigación acerca de las Redes Privadas Virtuales (VPN) y de las distintas tecnologías en las que se pueden implementar. Para la comprensión de esta tesis se recomienda tener conocimientos básicos de redes de datos, principalmente del modelo de referencia OSI y de TCP/IP. Para entender la estructura de esta tesis es necesario saber qué son y cómo se clasifican las VPNs. En el Capítulo I se analizan las razones por las cuales surgieron las Redes Privadas Virtuales y también se hace una breve reseña histórica. En este capítulo se puede inferir el concepto de VPN. En el último punto del capítulo se justifica la existencia de las VPNs basadas en MPLS, que es el tema central de esta tesis.

Una Red Privada Virtual es una alternativa a la infraestructura WAN privada de una compañía. Utiliza infraestructura pública, pero las políticas de acceso y de seguridad que posee la hacen parecer una red privada. Una VPN puede utilizar cualquier tecnología de transporte de datos. De acuerdo a este aspecto, una VPN puede clasificarse en *Overlay* o en *Peer-to-Peer*. Con el fin de tener una idea sobre cada tipo de VPN y hacer un comparativo entre ellas, en el Capítulo II se describen las tecnologías con las que se puede implementar una VPN *Overlay*. De la misma manera, en el Capítulo III se describen las bases teóricas para entender el funcionamiento de las VPN *Peer-to-Peer*, aunque cabe hacer la aclaración que el contenido de este capítulo se enfoca a las VPNs MPLS, cuestión fundamental de esta tesis. En el Capítulo IV, se detalla la descripción, clasificación, operación y seguridad de las VPNs.

El capítulo central de esta tesis es el V, llamado “Operación de VPNs MPLS”. En él se describe una VPN MPLS y sus principales características, así como la operación, seguridad, escalabilidad, Calidad de Servicio y los pasos a seguir (sin abundar en el código) para la configuración de una VPN MPLS. Este capítulo trata por separado cada una de las nociones que implica una VPN MPLS, pero en la sección final, se hace una recapitulación en la que se unifican todos los conceptos.

En el Capítulo VI de esta tesis se analiza el estado actual y futuro de las VPNs MPLS. Para ello, se realizaron varias entrevistas a diferentes Proveedores de Servicios, a proveedores de equipo y a clientes. Por razones de confidencialidad, no se mencionan nombres de empresas o personas, ni cifras o precios, ya que esa fue una petición de los entrevistados. Sin embargo, se trata de hacer un comparativo de las VPNs en forma cualitativa. A partir de la información recabada en las entrevistas, se dedujeron los criterios de selección de un cliente para decidir cuál es el tipo de VPN que más le conviene y si una VPN MPLS ofrece ventajas sobre las demás. El Anexo de esta tesis es el cuestionario que se hizo a las personas entrevistadas. Uno de los objetivos de esta tesis era obtener información precisa, pero no se obtuvo respuesta a todas las preguntas formuladas. Finalmente, las Conclusiones muestran el análisis hecho a las ventajas, necesidades y expectativas de las VPNs MPLS.

Después de las Conclusiones se encuentra un Glosario. En él se describen brevemente los términos y las tecnologías mencionadas a lo largo de la tesis y que quizá no sean familiares para el lector.

Capítulo I Introducción

I.1. Historia de las redes privadas virtuales.

I.1.1. Razón y origen de las redes privadas virtuales.

Las presiones de la actual competencia en el mercado de muchas industrias han resultado en alianzas y sociedades entre las empresas. forzando a que corporaciones separadas actúen y funcionen como una misma al enfrentar a los clientes. Mientras dicha táctica ha incrementado la productividad y aprovechamiento en la mayoría de las corporaciones, al mismo tiempo se han creado nuevas demandas en las redes corporativas.

El mundo ha cambiado en las últimas dos décadas. En lugar de simplemente conectarse a sus sitios locales o regionales, la mayoría de las empresas tienen que pensar en conexiones más amplias debido a que sus mercados así lo requieren y para ello, necesitan que sea en una forma rápida, segura y confiable hacia cualquier lugar en el que se encuentren sus oficinas.

Una red enfocada únicamente en conectar sitios fijos de una corporación no es factible para muchas compañías pues, hoy en día, usuarios remotos y socios externos necesitan tener acceso a los recursos computacionales de la empresa y la clásica red WAN debe ser extendida para acomodar a estos usuarios. En consecuencia, muchas empresas están considerando las Redes Privadas Virtuales (*VPNs*) para complementar su infraestructura WAN.

Al principio, una comunicación confiable significaba el uso de líneas dedicadas para mantener una WAN. Las líneas dedicadas abarcaban desde ISDN (*Integrated Services Digital Network*, que corre a 144Kbps) hasta

OC-3 (*Optical Carrier 3*, que corre a 155Mbps), proporcionando un medio para expandir la red de una empresa en forma confiable, pero cuyo costo de mantenimiento es realmente elevado. Como la popularidad de Internet creció, éste comenzó a ser visto como un medio de expansión de las redes corporativas. Primero vinieron las intranets, cuyos sitios eran diseñados para uso exclusivo de los empleados de la compañía y ahora, son las Redes Privadas Virtuales (VPNs) las que cubren las necesidades de los empleados (incluso los remotos) y de las oficinas distantes.

Se espera que en un futuro no muy lejano, cerca del 100% de las empresas hayan sustituido su infraestructura WAN con VPNs. Desde una perspectiva de la arquitectura de una red, el motivo de esto está manifiesto en que una VPN puede cubrir mejor las necesidades de conectividad que cualquier tecnología WAN a un menor costo. Sin embargo, las ventajas de una VPN van más allá, ya que son más baratas para operar que una red privada desde el punto de vista de la administración, ancho de banda y capital. Por lo tanto, el período de recuperación de la inversión en el equipo de la VPN generalmente se lleva meses, en lugar de años. Además, una VPN opera de tal forma que permite que la compañía cliente se dedique a sus negocios por completo y pueda dejar de preocuparse por su red corporativa, siendo la principal ventaja ofrecida.

Primera forma de evolución de las VPNs

Una primera forma rústica de VPN fueron las líneas dedicadas. Toda la infraestructura estaba dedicada a un cliente y el ancho de banda se tenía disponible todo el tiempo, aunque no fuera aprovechado. Sin embargo, no podía ser vendido a nadie más. Este primer modelo de redes privadas actualmente se sigue dando en México, aunque con menor incidencia; en el resto del mundo ya se ha abandonado este concepto de VPN debido a que no es costeable para el proveedor, pues se tiene ancho de banda ocioso.

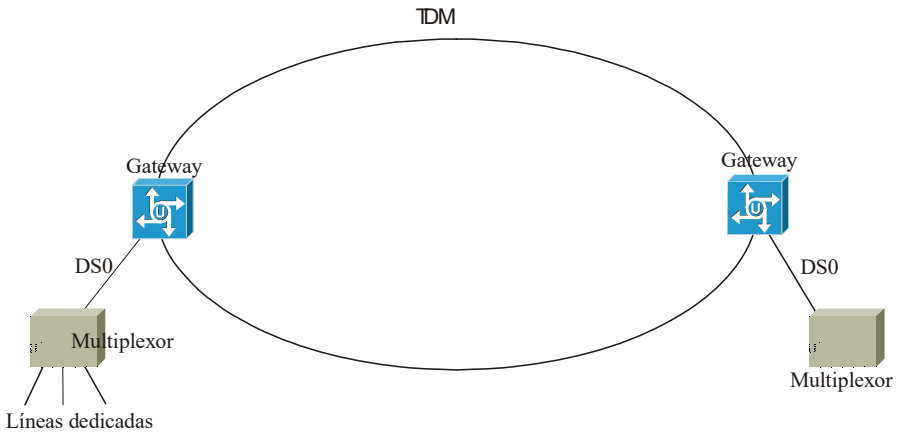


Figura I.1.1. VPN por líneas dedicadas

El problema con este tipo de red es que si el cliente quería servicio de datos y además quería videoconferencia, por ejemplo, se tenían que otorgar dos líneas dedicadas, una para cada tipo de servicio. Además, no es posible otorgar calidad de servicio y el costo de renta es elevado.

Modelo *Overlay*

El siguiente paso en la evolución fue la creación de redes privadas virtuales (llamadas así porque el tráfico entre sitios es transportado por una red pública) bajo el modelo *Overlay*. El desarrollo a esta tecnología se originó al buscar la disminución de costos (en cajas y en procesamiento) y no para ofrecer mayor cantidad de servicios. Con el modelo *Overlay* se ofreció el mismo servicio de "línea privada" pero en el medio físico se tenían varios circuitos virtuales. En este caso, si se requería servicio de datos y servicio de videoconferencia, éstos se ofrecían en el mismo tubo, pero en diferentes circuitos virtuales.

La distinción de los circuitos virtuales se hacía en el CE, el cual tenía cierta inteligencia e implicaba un costo. Realmente, el ahorro estaba en la distribución adecuada del ancho de banda y en la disminución de canales (por ejemplo, recordemos que en TDM, para servicio de datos y de videoconferencia se necesitaban dos canales, mientras que en *Overlay* sólo es necesario un canal, el cual tiene dos circuitos virtuales). En este modelo se desarrollaron X.25, Frame Relay, ATM y recientemente, L2TP.

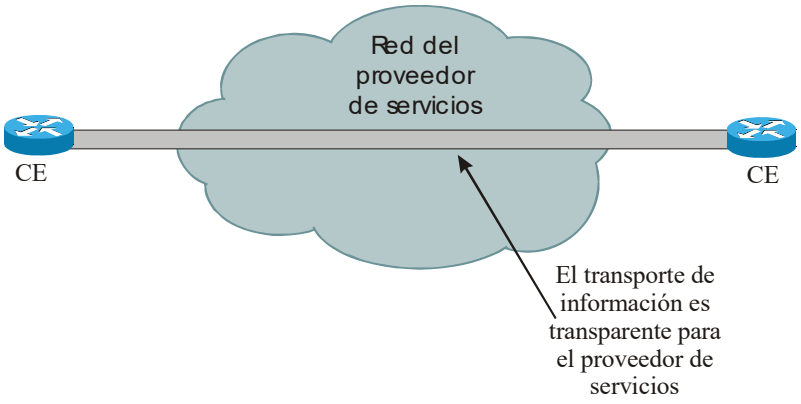


Figura I.1.2. VPN Overlay

La VPN se genera sobre la infraestructura del proveedor de servicios y no se hace *peering* o enrutamiento directo con el equipo del proveedor, sino que se pasa todo el tráfico en forma transparente a través de él. Con esto, el enrutamiento (definición de rutas y algoritmos) se ejecuta en el equipo del cliente, por lo que él se hace responsable de los problemas que surjan en el enrutamiento.

Las VPNs *Overlay* también son llamadas VPNs de capa 2. Las cajas de capa 2 no están posibilitadas para ofrecer un número ilimitado de circuitos virtuales, con lo que poco a poco se llega a su capacidad máxima. Esto significa un costo adicional debido a que es necesario añadir cajas (o módulos o interfaces) en la red del Proveedor De Servicios. Si el número de clientes no justifica hacer esta inversión, se presenta una problema de escalabilidad de negocio. Sin embargo, este modelo todavía es muy usado pues cubre necesidades que aún se tienen en muchos clientes.

Como las necesidades de las empresas clientes fueron cambiando, surgieron las VPDNs para cubrir las insuficiencias del modelo *Overlay*. Una VPDN es una red que extiende un acceso remoto de la red privada

usando una infraestructura compartida. En lugar de realizar conexiones directamente a la red privada, los usuarios de la VPDN sólo necesitan usar la Red Telefónica Pública de Conmutación para conectarse a un Punto de Presencia (POP) local del Proveedor de Servicios de Internet (ISP), el cual emplea el internet como medio de transmisión para comunicar a los usuarios con su red privada. El costo de este enlace es el de una llamada de larga distancia, lo que representa un ahorro considerable para el cliente comparándolo con la renta de un enlace dedicado.

Las VPDNs son una solución efectiva para aquellas empresas que cuenten con empleados remotos, es decir, que estén lejos de las oficinas corporativas y que necesitan tener acceso a su red. Una VPDN simplemente consiste en establecer una conexión punto a punto de larga distancia entre un usuario remoto y la red privada. Los protocolos más usados para este tipo de VPN´s son L2TP para el establecimiento de conexiones lógicas y IPsec para el cifrado de información.

Modelo peer-to-peer

A pesar de que las VPNs *Overlay* funcionan bien, pronto se requirió evolucionar a otro modelo ya que se requirió ampliar la escalabilidad por parte del proveedor, así como el aumento de ancho de banda y diferenciación de servicios debido al tipo de aplicaciones que fueron surgiendo. Poco a poco, los clientes comenzaron a comprar menos equipo y los Proveedores de Servicios se hicieron responsables de la adquisición de servidores, de *backups*, de discos con mayor capacidad para soportar las bases de datos de los clientes, etc. y todo ese equipo se dispone en renta a una cuota fija.

La tendencia industrial se encamina hacia IP. Por una lado se tienen protocolos *legacy* y por otro lado, se tiene IP. Debido a que IP es una tecnología barata (pues permite dar los servicios a precios accesibles para mercados masivos) surgió la idea de montar los servicios de una VPN sobre IP. Así, surgen las VPNs *peer-to-peer*. Este modelo es llamado también de capa 3 porque puede estar montado sobre GRE, IPsec o MPLS.

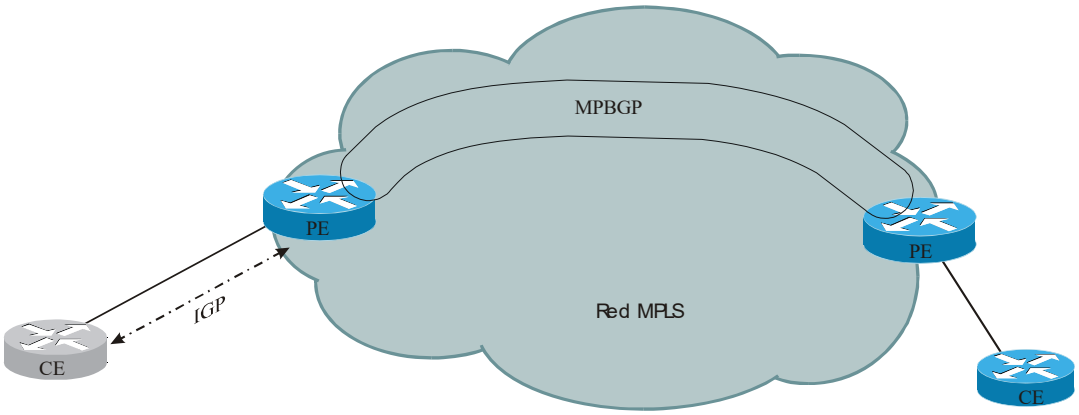


Figura I.1.3. VPN Peer-to-peer

En el modelo *peer-to-peer* , el proveedor tiene su propio equipo, al cual hace *peering* el equipo del cliente. En este modelo, el tráfico no es transparente para el proveedor, es decir, el cliente comparte información con el equipo del proveedor . Este paradigma es distinto al tradicional pues se comparte información de ruteo y por lo tanto, la responsabilidad de optimizar el enrutamiento (y cualquier problema relacionado) es del proveedor, no del cliente. El hecho de intercambiar rutas entre el cliente y el proveedor dio lugar a una controversia entre los paradigmas –tradicional vs. nuevo- y por ello, al principio el cliente no se sentía cómodo de intercambiar información pues no tiene forma de estar seguro de que el proveedor no esté haciendo *peering* con la competencia.

El desconocimiento de este modelo hace creer que es menos confiable que el *overlay*, pero no se así. De hecho, esto podría ser considerado como una desventaja que opaca los enormes beneficios que pueda ofrecer el modelo *peer-to-peer*. Es por eso que en el cambio de *overlay* a *peer-to-peer* se necesita labor de convencimiento y explicar al cliente los beneficios que ofrece el último modelo. En un futuro, se espera que se tenga tecnología IP montado sobre DWDM para abaratar costos.

MPLS en las VPNs

Actualmente, la tendencia de las VPNs está orientada al modelo *peer-to-peer*. En este modelo, MPLS ofrece mayores ventajas pues además de que soporta los mismos servicios que se tenían con las antiguas VPNs (como Frame Relay, ATM, etc.) también ofrece enrutamiento optimizado en el núcleo del proveedor y evita los *lookups* o búsquedas a nivel de capa 3 (eliminando un procesamiento lento en los enrutadores, lo cual es costoso). En resumen, MPLS es una tecnología que ofrece los mismos beneficios que las otras, pero a costos menores. Con lo anterior, los servicios que antes se contrataban con una red privada, ahora se montan sobre una infraestructura barata, permitiendo seguir invirtiendo sin los costos de las otras tecnologías.

No existe un antecedente como tal de MPLS, lo más parecido sería ATM, Frame Relay y/o X.25 aunque hacen cosas diferentes. Sin embargo, hace 4 ó 5 años, Cisco Systems desarrolló un protocolo llamado *Tag Switching*, que es el protocolo del que se originó MPLS. Lo que proponía *Tag Switching* es que en lugar de hacer *lookups* o búsquedas a nivel de capa 3 para conocer hacia dónde se tienen que conmutar los paquetes, se crea una tabla a nivel de capa 2 y se pone una etiqueta entre el encabezado de capa 2 y el encabezado de capa 3. Para ello, se crean tablas dinámicas en caché, evitando buscar en capa 3 la interfaz por la que se va a enrutar el paquete. Lo que se hacía era "marcar" los paquetes y con base en eso, se hacía el *lookup* de las etiquetas, no de las direcciones IP.

Así, en el núcleo del Proveedor de Servicios no se tenía que dar tantas funcionalidades pues la premisa de *Tag Switching* es que un paquete que llegue a la red, tiene que salir lo más rápido posible. Cisco mandó el draft de *Tag Switching* a la IETF y la norma surgió como "MPLS" (*Multiprotocol Label Switching*). Las diferencias entre *Tag Switching* y su sucesor MPLS, además del nombre, son el protocolo de distribución de etiquetas (para *Tag Switching* se llama *Tag Distribution Protocol*, TDP, y para MPLS se llama *Label Distribution Protocol*, LDP) y el número de puerto de TCP (para TDP el número de puerto es 711 y para LDP es 646).

Otros fabricantes han hecho su propia versión de MPLS, pero para que haya interoperabilidad, se tiene que hablar LDP.

I.2. Necesidades actuales para el transporte de información usando VPNs con MPLS

Inicialmente, las primeras redes conocidas propiamente como VPNs que se trataron de implementar, se desarrollaron sobre servicio Frame Relay y servicio ATM, ya que eran las únicas tecnologías que se tenían disponibles para tal fin en ese entonces. Estas dos tecnologías ganaron bastante popularidad alrededor del mundo, siendo Frame Relay la que se implementó con mayor incidencia en nuestro país. En los siguientes párrafos se analizará cada una de estas tecnologías y después de ello, se razonará sobre el porqué de una VPN MPLS.

Frame Relay da servicio de transporte sin calidad de servicio, mientras que ATM sí ofrece términos de QoS. Sin embargo, cuando se trata de transportar información sobre ATM lo que se tiene que hacer es un proceso de adaptación de la información de capa 3 a la capa 2, que es la que maneja ATM. Este proceso se lleva a cabo a través de una subcapa llamada *ATM Adaptation Layer* (AAL), la cual divide la información en bloques de 48 bytes y añade información de control en otros 5 bytes. Obviamente, este proceso de adaptación toma un tiempo adicional al tiempo que toma la conmutación de paquetes o celdas de un switch a otro, resultando en un retardo y en un *overhead* considerable.

Por su parte, en las redes IP los enrutadores han ido evolucionando de tal forma que tienen interfaces con mayor capacidad que la que tiene los switches ATM. Ha sido tal su crecimiento que pueden manejar interfaces desde STM-48 hasta OC-192, velocidades muy superiores comparadas con las de las interfaces ATM, que en algunos casos sólo puede manejar STM-16 o incluso más pequeñas. En conclusión, el desarrollo de enrutadores IP ha sido más rápido y controlado que el de los switches ATM, por lo que hay una mayor capacidad de transporte sobre una red IP que sobre una red ATM.

Cabe mencionar que el desarrollo de los enrutadores IP no se ha limitado únicamente a expandir su capacidad de transporte, sino que también se ha modificado su estructura interna para hacer que la conmutación de

paquetes sea más rápida y que pueda ser comparable con la conmutación que se hace en ATM, pero sin el retardo que incluye ATM al adaptar la información a la capa 2. Con ello, la velocidad de transporte sobre una red IP es comparable a la que se tiene sobre ATM, pero sin el costo elevado de una infraestructura con switches ATM, pues simplemente se montan los enrutadores IP sobre una red de transporte (como SDH) y se tiene una red de enrutadores conectados entre sí a través de SDH (STM-1, STM-16, STM-64, etc. según lo que se desee ocupar). Esta red es más barata que conectar un enrutador, colgarlo de un switch ATM y este switch ATM a su vez colgarlo a una red de transporte SDH (o SONET o lo que se desee). Realmente, la parte de ATM se libera (se quita) para reducir costos.

Sin embargo, IP no maneja calidad de servicio debido a que no tiene una parte dedicada a ello en forma estricta, desventaja que podrían encontrar los clientes con respecto a ATM, que sí maneja QoS. A pesar de ello, IP es capaz de manejar "niveles de servicio", en los cuales ciertos clientes pueden transportar su información de una manera privilegiada a través de la red, cumpliendo la mayor parte de los casos con las normas de calidad que se aplican en ATM.

Por las razones anteriormente expuestas, muchas empresas han decidido interconectarse por redes publicas en lugar de usar ATM para ese fin. IP está sobrepasando las ventajas que ofrece ATM y a costos menores, por lo que ATM ha dejado de desarrollarse. De hecho, en México ATM no logró obtener una parte significativa del mercado y en general, no existen muchos clientes para ATM.

Por otro lado, Frame Relay se desarrolló bastante bien, su implementación ha tenido gran aceptación y tiene presencia en muchos sitios, es decir, tiene una cobertura amplia, al grado que tal vez la red IP no la ha alcanzado aún. Por esta razón, las VPNs sobre Frame Relay se utilizan en gran medida todavía, en parte debido a que los costos son menores a los que se tienen en ATM e inclusive a los que se tienen en IP, y en parte también porque es una tecnología que tiene muchos años en el mercado y tiene bastantes clientes. Por ello, en cuanto a VPNs, la competencia está entre IP y Frame Relay, aunque éste último no compite en la capacidad de transmisión de una red IP, pero para empresas pequeñas, sigue siendo efectivo.

Con los textos anteriores, se concluye que una red VPN sobre IP es mejor que sobre ATM o Frame Relay en cuanto a la capacidad de transporte y costos (al menos respecto a ATM). Realmente, existen muchas técnicas de implementación de VPNs sobre IP. Una de ellas es a través de túneles, encapsulando la información IP en IP, enviándola por PPP de tal forma que se forme una conexión lógica a nivel IP sobre una red. Otra técnica de implementación de VPNs sobre IP es por medio de encapsulación MPLS.

Por mucho tiempo se han manejado las VPNs basadas en túneles, las cuales requerían de mucho tiempo de implementación (ya que era manual) y cuyo mantenimiento era laborioso (porque también era manual). Si una empresa tenía una matriz y *n* sucursales, se tenían que implementar $\left[\frac{n \times (n-1)}{2}\right]$ túneles, lo cual resulta impráctico en el caso de que *n* sea grande, modificándose la estructura de la red. Esto último se debe a que al agregar un nodo, se tiene que formar conexiones lógicas entre ese nodo y los demás de la red, aunque no sean de la misma VPN IP. Con ello, se originan problemas de administración (configuración y mantenimiento) pues si se tiene una falla, se tendría que revisar cada uno de los túneles (esto es, que estén bien construidos, que estén operando, que transporten información correctamente, etc.)

Posteriormente, debido a las complicaciones que presentaban las VPNs basadas en túneles, se empezaron a implementar nuevos esquemas en los que se aislaron los enrutadores y se agruparon en conjuntos para que manejaran la tabla de enrutamiento de una sola VPN y, al final, se optó por implementar un esquema mucho más sencillo usando VPNs sobre MPLS, lo cual elimina la creación de túneles, encapsula la información en un *frame* de MPLS que va entre la capa 2 y la capa 3 (lo que se llama “capa 2 y media”) y evita los problemas de administración que se tenían, con ayuda de otros medios como el aislamiento de información de enrutamiento en enrutadores por BGP.

Dado que es muy fácil implementar VPNs con MPLS, las empresas que optan pasar su tráfico por redes IP, encuentran que la mejor forma de hacerlo es por medio de VPNs basadas en MPLS. Esta tesis se concentra en explicar el funcionamiento de las VPNs MPLS y las expectativas que se tiene de ellas.

Capítulo II Bases teóricas para VPNs Extendidas

II.1 Encapsulación en la capa de enlace de datos

II.1.1 Frame Relay

Introducción

Frame Relay es una tecnología de alta velocidad que ofrece ancho de banda sobre demanda y que permite multiplexar estadísticamente diferentes circuitos virtuales sobre un mismo enlace de acceso a la red. La existencia de caminos redundantes en las redes públicas Frame Relay y el uso de protocolos dinámicos, proporcionan una alta disponibilidad de la red. Estas características la posicionan como una tecnología adecuada en términos de velocidad, costos y disponibilidad para las empresas en un gran número de aplicaciones.

Aunque bajo ciertas condiciones Frame Relay es capaz de transportar voz, al igual que Internet, es una tecnología pensada para el transporte de datos. Sin embargo, para la transmisión de voz y video, ATM es la mejor opción. La interoperabilidad entre Frame Relay y ATM está garantizada por la existencia de normas internacionales[1].

Topología

Los dispositivos asociados a las red FR se dividen en dos categorías: DTE (*Data Terminal Equipment*) y DCE (*Data Circuit-terminating Equipment*)[5].

Los DTEs generalmente son considerados como equipos terminales y Están localizados en el lado del cliente. De hecho, pueden ser propiedad del cliente. Ejemplos de DTEs son: routers, *bridges*, computadoras personales y terminales.

Los DCEs son dispositivos de intercomunicación del propietario de la portadora. Proporciona el reloj y los servicios de conmutación en la red, siendo realmente los dispositivos que transmiten los datos a través de la WAN. Generalmente son switches y routers.

Un ejemplo de su arquitectura se muestra a continuación[5]:

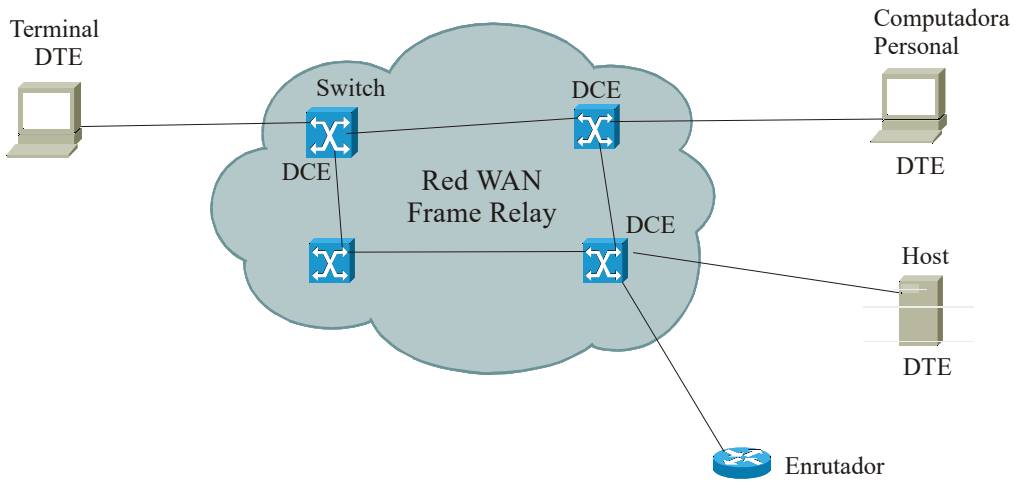


Figura II.1.1.1

Descripción y características:

Frame Relay ofrece un servicio orientado a conexión basado en el establecimiento de circuitos virtuales bidireccionales y en el intercambio de tramas tipo HDLC. Se puede pensar en FR como una línea virtual rentada, pues el cliente renta un circuito virtual permanente entre dos puntos y entonces puede enviar *frames* o paquetes de datos entre ellos. También es posible rentar circuitos virtuales permanentes entre un lugar

determinado y muchas otras localidades, de modo que cada *frame* lleve un identificador de 10 bits que le diga qué circuito virtual usar.

La diferencia entre una línea rentada real y una virtual es que con una real, el usuario puede enviar tráfico durante todo el día a máxima velocidad (enlace dedicado). Con una línea virtual, se pueden enviar ráfagas de datos a toda velocidad, pero el uso promedio a largo plazo deberá ser inferior a un nivel predeterminado. A cambio, la portadora cobra mucho menos por una línea virtual que por una física.

FR proporciona una forma de determinar el inicio y el fin de cada *frame* (por medio de bytes bandera en su encabezado) y de detectar errores de transmisión . Si se recibe un *frame* defectuoso, FR siempre lo descarta. FR no proporciona control de flujo. Sin embargo, tiene un bit en el encabezado que, en un extremo de la conexión, se puede encender para indicar al otro que hay problemas de congestión. El uso de ese bit es opción de los usuarios[3].

Las características más importantes de Frame Relay son[4]:

- Altas velocidades de transmisión
- Bajos retardos sobre la red
- Alta conectividad
- Uso eficiente del ancho de banda

Funcionamiento

Las conexiones de FR son implementadas usando circuitos virtuales, los cuales son conexiones lógicas creadas entre dos DTEs a través de la red de conmutación de paquetes FR. Los circuitos virtuales proporcionan una comunicación bidireccional de un dispositivo DTE al otro y son señalados únicamente por

un identificador del circuito virtual, llamado DLCI (*Data Link Connection Identifier*). Un cierto número de circuitos virtuales puede ser multiplexado en un mismo circuito físico para transmisión a través de la red. Esta capacidad ofrece reducir el equipo y la complejidad de la red requeridos para conectar varios dispositivos DTE. Los circuitos virtuales se clasifican en dos: en SVCs (*Switched Virtual Circuits*) y PVCs (*Permanent Virtual Circuit*).

Los SVCs son circuitos virtuales conmutados. La conexión es temporal y es usada sólo cuando es requerida esporádicamente para transmitir datos entre dos DTEs. Los PVCs son circuitos virtuales en los que las conexiones son establecidas permanentemente y son usados frecuentemente para transmitir datos entre dos DTEs.

Para poder utilizar una red FR, el cliente del servicio debe conectar su ambiente de cómputo interno a un enrutador (en el caso de una red local) que contenga una tarjeta FR o a un FRAD (*Frame Relay Access Device*). Estos elementos a su vez se conectan a la línea de acceso a la red a través de un DSU (*Data Service Unit*) o de un DSU/CSU (*Data Service Unit/Channel Service Unit*), que pueden ser dispositivos externos o estar integrados en los enrutadores y FRADs. Un FRAD es un dispositivo multiprotocolo que recibe datos por sus puertos seriales, los encapsula en tramas y los envía a la red FR. En el sentido inverso, recibe tramas de la red FR, desencapsula los datos y los envía al puerto correspondiente.

Entre los usuarios y los nodos de la red (*UNI, User-to-Network Interface*) se transmiten únicamente tramas de capa 2 (enlace de datos) del modelo OSI. Al establecer un circuito virtual, el usuario negocia con la red tres parámetros: CIR, Bc y Be, los cuales definen las características de las ráfagas de su tráfico. El CIR (*Commited Information Rate*) es la velocidad media de transferencia de información a la que el usuario desea transmitir. El CIR se mide sobre un intervalo de tiempo T que es proporcional al tamaño de las ráfagas que son transmitidas por la fuente de información. A este parámetro se le llama Bc (*Commited Burst Size*) y es el número máximo de bits que la red se compromete a transportar sobre cualquier intervalo de tiempo (normalmente inferior a 8 segundos).

$$Bc = AR * s$$

donde:

AR: velocidad de acceso

s: tiempo máximo de ráfagas

$$CIR = \frac{Bc}{T}$$

donde:

T: tiempo entre ráfagas

Por ejemplo, si la velocidad de acceso (AR) es de 64kbps, la duración de las ráfagas es de 1.5 segundos y el tiempo T entre ráfagas es de 6 segundos, entonces el Bc es de 96kb y el CIR es de 16kbps.

Si el tráfico de un usuario excede su CIR (es decir, Bc bits en T segundos), el nodo de acceso a la red enciende el indicador de elegibilidad (bit DE del encabezado del *frame*, ver figura II.1.1.2) para ser descartado de todas las tramas en exceso si es que no tiene una prioridad alta (en caso de que los paquetes estén marcados como prioritarios, no pueden ser descartados). Finalmente, el tráfico de un usuario que exceda Bc en más de una cierta cantidad Be (*Excess Burst Size*) durante un intervalo de tiempo T, es descartado en el nodo de acceso de la red. En algunas redes es posible programar el nodo de acceso para que deje pasar este tráfico en exceso con el bit DE encendido. A la cantidad $\frac{(Bc + Be)}{T}$ se conoce como EIR (*Excess Information Rate*).

En el diseño y operación de una red privada virtual que utilice una red pública de transporte FR, es de suma importancia ajustar los parámetros del servicio: CIR, Bc y Be. Este ajuste debe realizarse en función de las características del tráfico inicial esperado y de mediciones efectuadas continuamente durante la operación cotidiana de la red.

En una red FR se cobra la renta de la interfaz, el CIR, la distancia desde el primer punto del PVC hasta el último (se cobra por km), el enlace local (FR no se hizo para enlaces locales) y opcionalmente, el Be[3].

Formato del *frame*

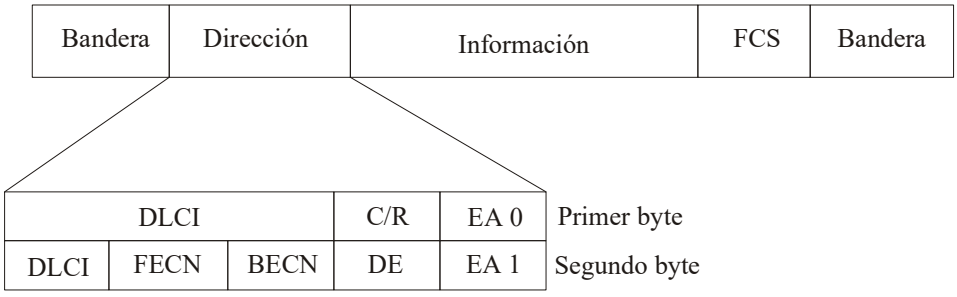


Figura II.1.1.2

En la parte superior de la figura II.1.1.1 se observa cómo el *frame* de FR sigue el mismo formato que HDLC, difiriendo únicamente en el campo Dirección. Los campos que componen al encabezado se describen a continuación:

- Bandera: un byte. Delimitan el inicio y el final del *frame*; es una secuencia de bits: 01111110.
- Información: variable. Contiene los datos de capas superiores.
- FCS (*Frame Check Sum*): un byte. Sirve para detectar errores en el *frame*, para ello se somete en el trasmisor a los campos del *frame* a un algoritmo polinomial y se obtiene un FCS, el cual es enviado en el *frame*. En el receptor se aplica nuevamente el algoritmo y su resultado lo compara con el FCS enviado. Si son distintos, hubo error.

Después de la primera bandera, siguen los dos bytes del campo Dirección. En FR, este campo está dividido, compuesto por:

- 10 bits correspondientes al identificador del circuito virtual, llamado DLCI (*Data Link Connection Identifier*). Estos diez bits son el corazón del encabezado FR, ya que identifica la conexión lógica que es multiplexada en el canal físico. En el modo básico de direccionamiento, DLCI tiene significado local, es

decir, los dispositivos finales en dos puntos diferentes de una conexión pueden usar un DLCI diferente para referirse a la misma conexión. En el primer byte del campo, el DLCI ocupa los primeros 6 bits, y en el segundo byte del campo, ocupa los primeros 4 bits.

- El bit C/R (*Command/Response*) casi no es usado.
    - Existen tres bits en el segundo byte destinados al control de congestión:El bit FECN (*Forward Explicit Congestion Notification*) encendido (es decir, igual a uno) señala que hay congestión en el sentido de la trayectoria seguida con lo que se le indica a la máquina transmisora que baje su tasa de transmisión.
    - El bit BECN (*Backward Explicit Congestion Notification*) encendido señala que hay congestión en el sentido inverso de la trayectoria seguida.
    - El bit DE (*Discard Elegibility*) es empleado por el equipo terminal FR para indicarle a la red que ciertos *frames* son menos importantes que otros. Así, en caso de sobrepasar la tasa permitida, se da prioridad a aquellos frames distinguidos como importantes y los demás se descartan.
- Al final de cada byte del campo dirección, está un bit de dirección extendida o *Extended Address* (EA). Si este bit es uno, el byte correspondiente es el último byte de un DLCI. (por esta razón, en el primer byte, el bit EA tiene un cero y, en el segundo byte, el bit EA tiene un uno). La mayorías de las implementaciones actualmente utilizan 2 bytes de DLCI, pero gracias a los bits EA en un futuro aquellos podrán ser más grandes[5].

Ventajas y desventajas

Una de las ventajas que ofrece Frame Relay es *Link Management Interface*. LMI es un conjunto de mejoras hechas a la especificación básica de FR. Ofrece ciertas características (llamadas extensiones) para administrar redes complejas. Las extensiones LMI incluyen direccionamiento global, mensajes de estado del circuito virtual y *multicasting*. Los mensajes LMI son enviados en *frames* caracterizados por un DLCI específico para LMI, definido como DLCI = 1023[2].

La extensión opcional de LMI más importante es el direccionamiento global (*global addressing*). La especificación básica de FR sólo soporta valores del campo DLCI que identifiquen a los PVCs con un significado local. Esto representa una desventaja, pues en este caso no hay direcciones que identifiquen a las interfaces de red o a los nodos asociados a dichas interfaces. Esto quiere decir que con un direccionamiento normal FR, deben crearse mapas estáticos para indicarle a los enrutadores qué DLCIs usar para encontrar un dispositivo remoto y sus direcciones de red asociadas. Con la extensión de direccionamiento global sí es posible identificar los nodos. Con esta extensión, los valores insertados en el campo DLCI de un *frame* son direcciones con significado global de dispositivos individuales de un usuario final[2].

Otra extensión opcional importante de LMI es *multicasting*. Los grupos *multicast* son designados por una serie de cuatro valores de DLCIs reservados (del 1019 al 1022). Los *frames* son enviados por un dispositivo usando uno de dichos DLCIs reservados hacia todos los puntos de salida del grupo *multicast*. La extensión *multicasting* define también mensajes LMI que notifican a los usuarios de dichos dispositivos de la presencia, supresión o agregación de grupos *multicast*. Esta extensión también reserva ancho de banda para permitir los mensajes de actualización de los protocolos de enrutamiento.

Por último, los mensajes de estado de los circuitos virtuales proporcionan comunicación y sincronización entre dispositivos DTE y DCE. Dichos mensajes son usados para reportar periódicamente el estado de los PVCs, lo cual previene que los datos sean enviados a PVCs que no existan[5].

Implementación

Una red típica Frame Relay consiste en cierto número de DTEs conectados a los puertos remotos en el equipo multiplexor por medio de los servicios tradicionales *point-to-point*, tales como T1 o circuitos de 56k. Una implementación común de una red privada Frame Relay es equipar un multiplexor T1 con interfaces Frame Relay y con otras interfaces diferentes. El tráfico Frame Relay es enviado a través de las interfaces FR. El tráfico que no es Frame Relay se envía por las demás interfaces, según la aplicación y servicio, como por

ejemplo un conmutador PBX para servicio telefónico o una aplicación de teleconferencia, tal como se muestra en la siguiente figura[5]:

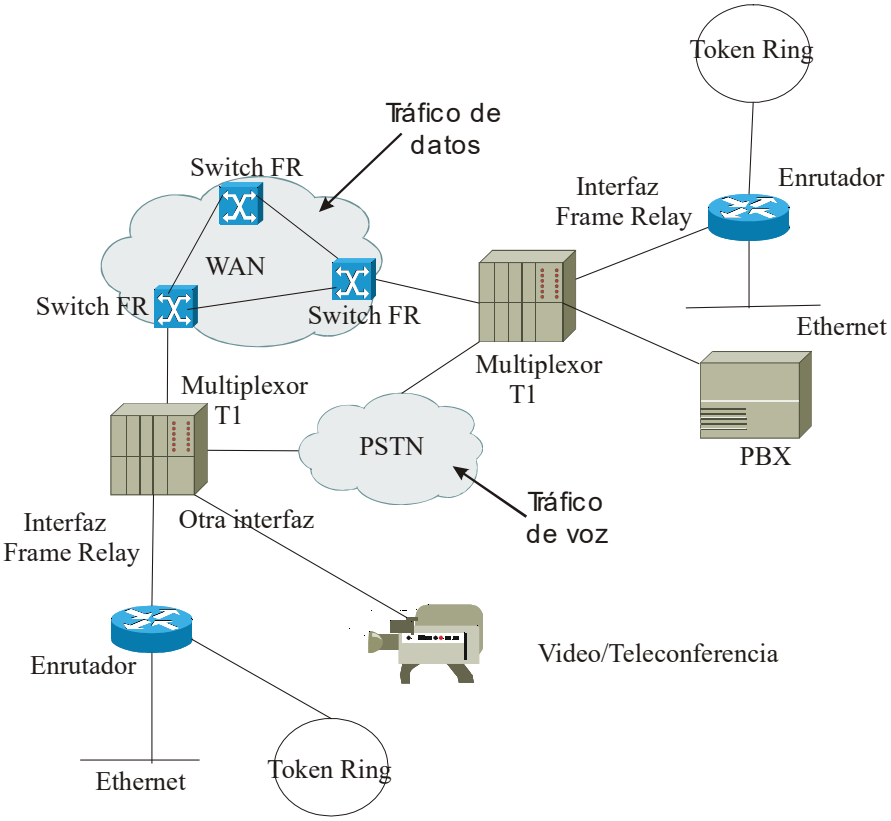


Figura II.1.1.3

La mayoría de las redes Frame Relay son proporcionadas por proveedores que ofrecen servicios de transmisión a sus clientes. Comúnmente a esto se le llama servicio público Frame Relay. Frame Relay puede ser implementado en servicios públicos o privados.

II.1.2 ATM

Introducción

Inicialmente propuesto por la Industria de las Telecomunicaciones, rápidamente se convirtió en la tecnología más promovida dentro de las industrias de Comunicaciones y Computadores. En su momento, las recomendaciones iniciales propuestas por el CCITT en 1988 fueron que, ATM y la Red Óptica Síncrona (SONET) formasen la base de la Red Digital de Servicios Integrados de Banda Ancha (B-ISDN), un nuevo estándar en desarrollo para la integración en red de datos, voz, imagen y vídeo, a velocidades de transmisión desde 34 Mbps a varios Gigabits por segundo.

ATM emplea el concepto de Conmutación de Celdas (*Cell Switching*), el cual combina los beneficios de la Conmutación de Paquetes tradicionalmente utilizada en redes de datos, y la Conmutación de Circuitos utilizada en redes de voz.

ATM se basa en el concepto de Conmutación Rápida de Paquetes (*Fast Packet Switching*) en el que se supone una fiabilidad muy alta a la tecnología de transmisión digital, típicamente sobre fibra óptica, y por lo tanto la no necesita la recuperación de errores en cada nodo. Ya que no hay recuperación de errores, no son necesarios los contadores del número de secuencia de las redes de datos tradicionales y tampoco se utilizan direcciones de red ya que ATM es una tecnología orientada a conexión. En su lugar se utiliza el concepto de Identificador de Circuito o Conexión Virtual (VCI)[8].

Topología

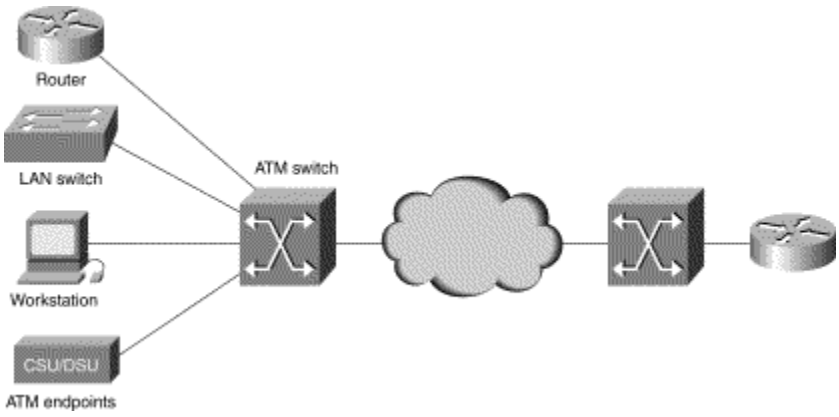


Figura II.1.2.1

Una red ATM esta constituida por un conmutador ATM y los puntos finales *o "endpoints*" ATM. Un conmutador ATM es responsable del tránsito de celdas o células a través de la red ATM. El trabajo de estos conmutadores está muy bien definido y consiste en aceptar las celdas o células desde un punto final o *"end point*" o de otro conmutador ATM. Éste a su vez, lee y actualiza la información del encabezado de la célula o celda y lo conmuta rápidamente a la interfaz de salida hasta su destino. Un punto final *o "endpoint*" contiene un adaptador de interfaz de red ATM, por ejemplo un punto final puede ser una estación de trabajo, un enrutador, una Unidad De Servicios Digitales (DSU, *Digital Service Units*), un conmutador LAN, un CODEC, etc.

Descripción y características

Los paquetes en ATM tienen una longitud fija de 53 bytes (ver figura II.1.2.7) permitiendo que la información sea transportada de una manera predecible. El hecho de que sea predecible permite diferentes tipos de trafico en la misma red.

Los paquetes están divididos en dos partes, el encabezado y el *payload*. El *payload* (que ocupa 48 bytes) es la parte del paquete donde viaja la información, ya sean datos, imágenes o voz. El encabezado (que ocupa 5 bytes) lleva el mecanismo de direccionamiento.

Otro concepto clave es que ATM está basado en el uso de conmutadores. Hacer la comunicación por medio de un conmutador (en vez de un *bus*) tiene ciertas ventajas: reservación de ancho de banda para la conexión, mayor ancho de banda, procedimientos de conexión bien definidos y velocidades de acceso flexibles. ATM es una arquitectura estructurada en capas que permite que múltiples servicios como voz y datos vayan mezclados en la misma red.

Existen tres tipos de servicios en ATM:

- Circuitos Virtuales Permanentes (*Permanent Virtual Circuits* o PVC): que permiten una conectividad directa entre los sitios. Este es similar a las líneas dedicadas y garantiza la disponibilidad de la conexión. No requiere procedimientos de instalación de la conexión.
- Circuitos Virtuales Conmutados (*Switched Virtual Circuits* o SVC): creados dinámicamente y permanecen en uso solo cuando los datos contenidos son transferidos. Se toma en cuenta como una llamada telefónica y requiere de un control que usa señalización entre los puntos finales de ATM y los conmutadores ATM.
- Servicios No Orientados a Conexión

Funcionamiento

[6]La figura II.1.2.2 muestra un formato básico y la jerarquía de ATM. Una conexión ATM, consiste en celdas de información contenidas en un circuito virtual (VC). Estas celdas provienen de diferentes fuentes representadas como generadores de bits a tasas de transferencia constantes (como la voz) y a tasas variables tipo ráfagas o *bursty traffic* (como los datos). Cada celda compuesta por 53 bytes, de los cuales 48 (opcionalmente 44) son para el intercambio de información y los restantes para el uso de campos de control (cabecera) con información de "quién soy" y "donde voy". Una celda es identificada por un *"Virtual Circuit Identifier*" VCI y un *"Virtual Path Identifier*" VPI dentro de esos campos de control, que incluyen tanto el enrutamiento de celdas como el tipo de conexión.

- *Virtual Channel Identifier (VCI) – Identificador de canal virtual (VCI)*[7]*:* conexión virtual establecida a través de la red desde el origen hacia el destino, donde los paquetes, tramas o celdas se enrutan sobre el mismo trayecto para la duración de la llamada. Estas conexiones parecen trayectos dedicados a los usuarios, pero en realidad son recursos de red compartidos por todos los usuarios. El ancho de banda en un circuito virtual no se asigna hasta que se use.

- *Virtual Path Identifier* (VPI) – Identificador de trayectos virtuales (VPI)[8]*:* en ATM, existe un campo dentro de un encabezamiento de una celda que se emplea para conmutar trayectos virtuales, definidos como grupos de canales virtuales.

La organización de la cabecera (*header*) variará levemente dependiendo de sí la información relacionada es para interfaces de red a red o de usuario a red. Las celdas son enrutadas individualmente a través de los conmutadores basados en estos identificadores, los cuales tienen significado local - ya que pueden ser cambiados de interfaz a interfaz.

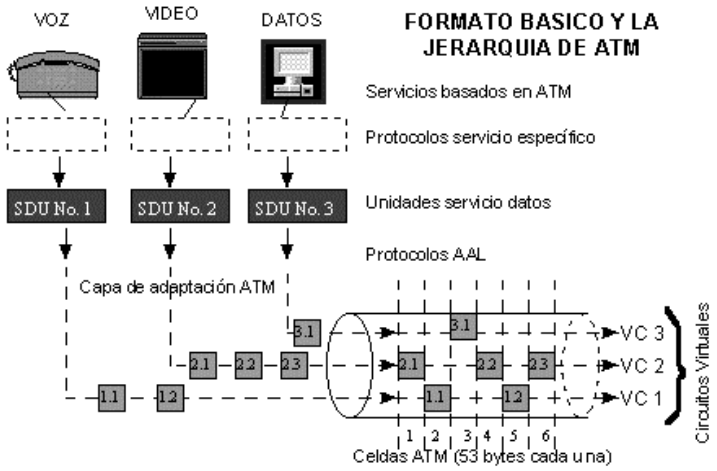


Figura II.1.2.2

La técnica ATM multiplexa muchas celdas de circuitos virtuales en una ruta (*path*) virtual colocándolas en particiones (*slots*), similar a la técnica TDM. Sin embargo, ATM llena cada *slot* con celdas de un circuito

virtual a la primera oportunidad, similar a la operación de una red conmutada de paquetes. En la figura siguiente se ilustran los procesos de conmutación implícitos, los conmutadores VC y los conmutadores VP.

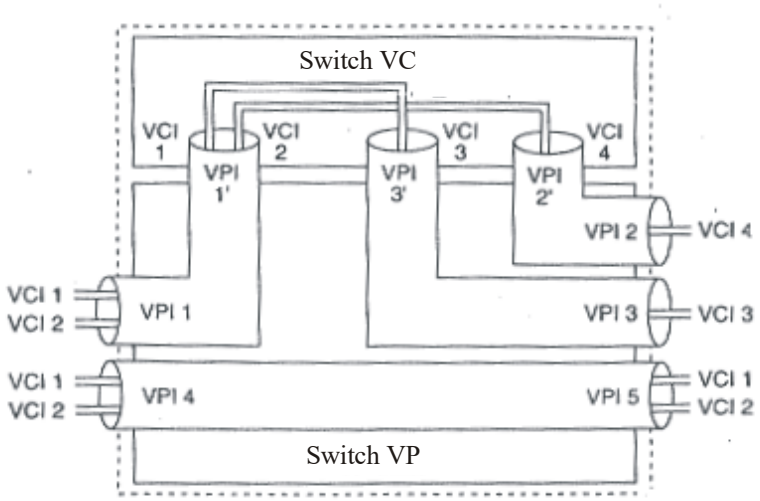


Figura II.1.2.3

Los *slots* de celda no usados son llenados con celdas "*idle*", identificadas por un patrón específico en la cabecera de la celda. Este sistema no es igual al llamado *bit stuffing* en la multiplexación Asíncrona, ya que aplica a celdas enteras.

Diferentes categorías de tráfico son convertidas en celdas ATM por medio la Capa de Adaptación de ATM (AAL - *ATM Adaptation Layer*), de acuerdo con el protocolo usado. (Más adelante se explica este protocolo).

El protocolo ATM consiste de tres niveles o capas básicas:

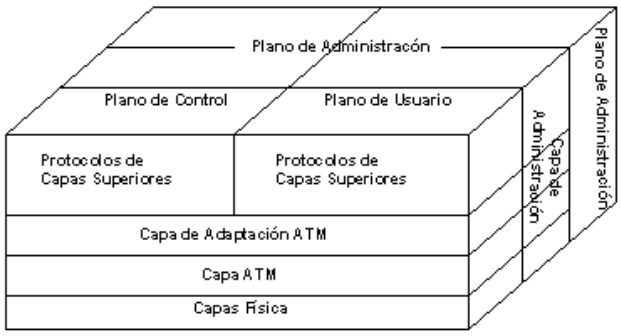


Figura II.1.2.4

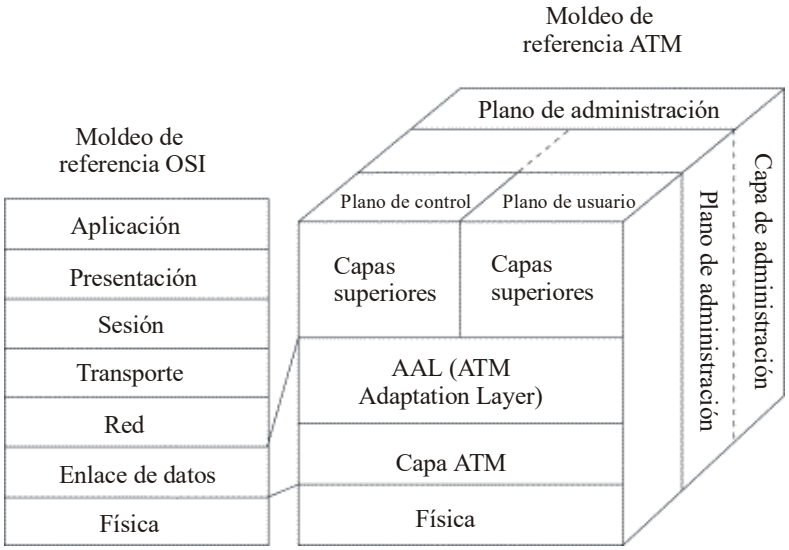


Figura II.1.2.5

La primera capa, llamada capa física (*Physical Layer*), define las interfaces físicas con los medios de transmisión. El protocolo de trama para la red ATM es responsable de la correcta transmisión y recepción de los bits en el medio físico apropiado. A diferencia de muchas tecnologías LAN como Ethernet que especifica ciertos medios de transmisión, (10baseT, 10base5, etc.) ATM es independiente del transporte físico. Las celdas ATM pueden ser transportadas en redes SONET (*Synchronous Optical Network*), SDH (*Synchronous Digital Hierarchy*, T3/E3, TI/EI) o aún en módems de 9600bps. Hay dos subcapas en la capa física que separan el medio físico de transmisión y la extracción de los datos:

La subcapa PMD (*Physical Medium Depedent*) tiene que ver con los detalles que se especifican para velocidades de transmisión, tipos de conectores físicos, extracción de reloj, etc., Por ejemplo, la tasa de datos SONET que se usa, es parte del PMD. La subcapa TC (*Transmission Convergence*) tiene que ver con la extracción de información contenida desde la misma capa física. Esto incluye la generación y el chequeo del Header Error Corrección (HEC), extrayendo celdas desde el flujo de bits de entrada y el procesamiento de celdas "*idles*", y el reconocimiento del límite de la celda. Otra función importante es intercambiar información de operación y mantenimiento (OAM) con el plano de administración.

La segunda capa es la capa ATM. Ella define la estructura de la celda y cómo las celdas fluyen sobre las conexiones lógicas en una red ATM; esta capa es independiente del servicio. Junto con la capa de adaptación de ATM, forman la capa de Enlace del modelo de referencia OSI. La capa ATM se encarga de compartir simultáneamente los circuitos virtuales a través de la capa física (multiplexado de celdas) y traspasa las celdas a través de la red ATM. Para hacer esto usa información de las VPI y VCI que se encuentran en el encabezado de la celda de ATM.

La tercer capa es la ATM Adaptation Layer (AAL). La AAL juega un rol clave en el manejo de múltiples tipos de tráfico para usar la red ATM, y es dependiente del servicio. Específicamente, su trabajo es adaptar los servicios dados por la capa ATM a aquellos servicios que son requeridos por las capas más altas, tales como emulación de circuitos, (*circuit emulation*), vídeo, audio, Frame Relay, etc.

AAL1[9]: es un servicio orientado a conexión, es usado para el manejo de recursos que usan una Tasa de Bits Constantes (*Constant Bit Rate* o CBR), como es la voz y las videoconferencias. ATM transporta tráfico CBR usando servicios de emulación de circuitos y también acomoda los equipos que se usan en las líneas dedicadas en la red central de ATM. AAL1 requiere una sincronización de tiempo entre el destino y el origen. Por esta razón, AAL1 depende de un medio, como SONET, que soporta reloj. El proceso de AAL1 prepara la celda para la transmisión en tres pasos. El primer proceso es la toma de muestras síncronas (por ejemplo, 1 byte de datos a una tasa de muestreo de 125 milisegundos) son insertadas dentro del campo *Payload*. El segundo paso es un Número de Secuencia (*Sequence Number* SN) y un Número de secuencia de Protección (*Sequence Number Protection* o SNP) que son agregados para proveer la información que el receptor AAL1 usa para verificar que las celdas se han recibido en un orden correcto. El tercer paso es el resto del campo *Payload* que está conformado por 47 bytes.

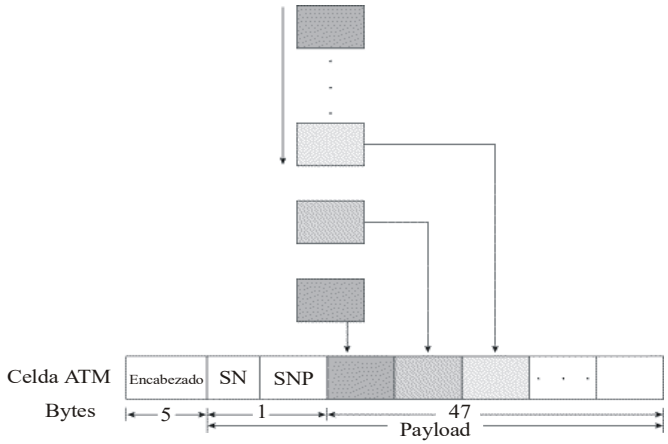


Figura II.1.2.6

AAL2[10]: es para uso de tráfico de Tasa de Bit Variable (*Variable Bit Rate* o VBR), que típicamente incluye servicios caracterizados como voz o video en paquetes que no tiene una velocidad constante de su transmisión de datos pero que tienen requerimientos similares a los servicios de bits constantes (CBR). AAL2 usa 44 bytes de la celda para datos y reserva 4 bytes para los procesos AAL2.

AAL3/4[11]: soporta dos tipos de datos, los orientados a conexión y los no orientados a conexión. Fue designado para las redes de los proveedores de servicios. AAL3/4 es usado para transmitir paquetes SMDS (*Switched Multimegabit Data Service*) a través de la red ATM.

AAL3/4 prepara a la celda para una transmisión en cuatro pasos. El primero es la Subcapa de Convergencia (*Convergence Sublayer* o CS) que crea una Unidad De Datos De Protocolo (*Protocol Data Unit*) usando una etiqueta antes para el encabezado y al último nos da la longitud del campo. EL segundo paso, es la Subcapa de Segmentación y Reensamblaje (*Segmentation and Reassembly* o SAR) que fragmenta al PDU y la CRC-32 a cada PDU para el control de error. Finalmente, el SAR PDU completo se adapta al formato de la celda ATM para que la capa de ATM lo pueda leer.

AAL5[12]: Es un protocolo para soportar transmisiones de datos con o sin conexión. Elimina parte de la complejidad y sobrecarga introducida por AAL3/4, proporcionando un nivel de adaptación simple y

eficiente para la transmisión de tramas de datos entre dispositivos tales como enrutadores, sobre una red ATM.

AAL5 define un formato de trama de longitud variable, así como los procedimientos para segmentar la trama en celdas para su transmisión sobre la red ATM, y el reensamblado en destino.

El Subnivel de Convergencia CS, para realizar sus funciones añade 8 bytes por trama: Un CRC-32 para detectar errores de trama y celdas perdidas, 2 bytes de para especificar la longitud de la trama (0-65.535 bytes), 2 bytes de control reservados. Hay un campo de relleno (*PAD*) conteniendo de 0 a 47 bytes con el fin del número total de bytes sea múltiplo de 48. La Unidad De Datos Del Protocolo así generada (CS-PDU), es transportada al subnivel SAR para su segmentación.

El subnivel SAR utiliza un bit del campo PT de la cabecera de la celda ATM, para indicar que es la última celda (EOM) perteneciente a la trama (PT = 0x1), o no es la última (not EOM, PT = 0x0). No consume ninguna parte de la carga útil de la celda para realizar esta función, obteniéndose una mejora de 4 bytes por celda frente a AAL3/4.

AAL5, a diferencia de AAL3/4, no permite la multiplexación de mensajes de diferentes usuarios (diferentes SDUs) dentro de un mismo VPI/VCI ya que no contiene el Identificador de Mensaje (MID), así que requiere un VPI/VCI dedicado.

Formato del *frame*

El formato de una celda ATM es muy simple. Consiste de 5 bytes de cabecera y 48 bytes para información.

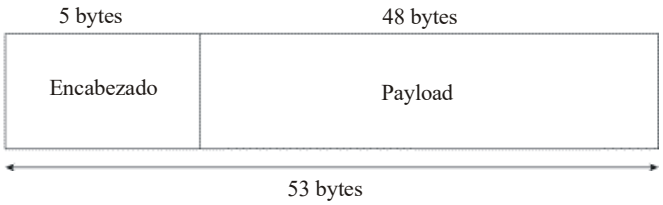


Figura II.1.2.7

Las celdas son transmitidas serialmente y se propagan en estricta secuencia numérica a través de la red. El tamaño de la celda ha sido escogido de tal forma que sea muy eficiente para transmitir largas tramas de datos y longitudes de celdas cortas que minimizan el retardo de procesamiento de extremo a extremo, que son buenas para voz, vídeo y protocolos sensibles al retardo. A pesar de que no se diseñó específicamente para eso, la longitud de la celda ATM acomoda convenientemente dos *Fast Packets IPX* de 24 bytes cada uno.

Los comités de estándares han definido dos tipos de cabeceras ATM: los User-to-Network Interface (UNI) y la Network to Network Interface (NNI). La UNI es un modo nativo de interfaz ATM que define la Interfaz En El Equipo Del Cliente (*Customer Premises Equipment*), tal como hubs o enrutadores ATM y la red de área ancha ATM (ATM WAN). La NNI define la interfaz entre los nodos de las redes (los conmutadores) o entre redes. La NNI puede usarse como una interfaz entre una red ATM de un usuario privado y la red ATM de un proveedor público (*carrier*). Específicamente, la función principal de ambos tipos de cabeceras de UNI y la NNI, es identificar las *"Virtual Paths Identifiers*" (VPIs) y los "Virtual Circuits" o Virtual Channels"(VCIs) como identificadores para el ruteo y la conmutación de las celdas ATM.

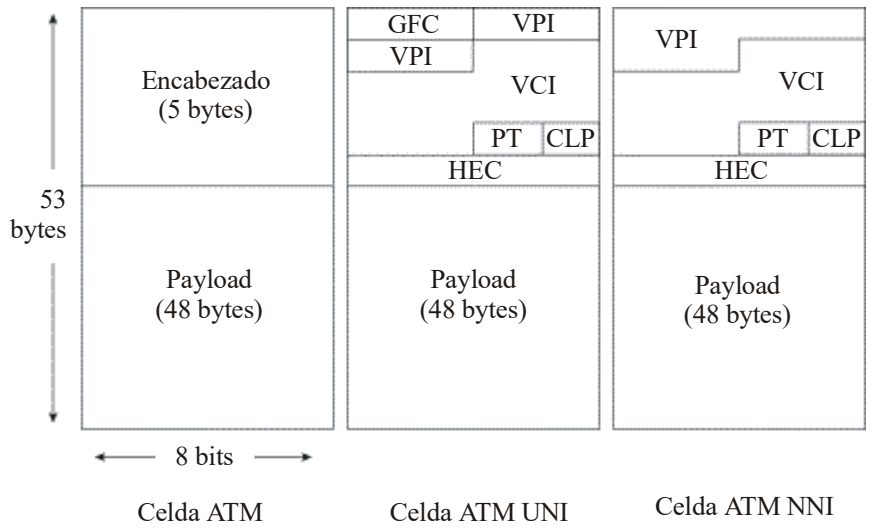


Figura II.1.2.8

- *Generic Flow Control (*GFC*):* Provee funciones locales, como identificar múltiples estaciones que comparten una misma interfase ATM. Este campo típicamente no es usado y su valor de default es 0 (0000 binario).
- *Virtual Path Identifier(VPI):* Junto con el campo VCI, identifica el próximo destino de la celda y pasa a través de una serie de conmutadores ATM en la dirección de su destinatario.
- *Virtual Channel Identifier (VCI):* Junto con VPI, identifica el próximo destino de la celda, que pasa a través de una serie de conmutadores en el sentido de su destinatario.
- *Payload Type (PT):* Indica en el primer bit de la celda si contiene datos de usuario o datos de control. Si la celda contiene datos del usuario, este bit es 0. Si contiene datos de control este bit es 1. El segundo bit indica la congestión (0= si no hay congestión y 1 si hay congestión), y el tercer bit indica si la celda es la última en una serie de células o celdas que representa un solo frame AAL5 (1= si es la última del *frame*).
- *Cell Loss Priority(CLP):* Indica si la celda debe ser descartada si existe una congestión extrema. Si el bit de CLP es 1, entonces la celda deberá de ser descartada dando preferencia a las celdas con el bit CLP igual a cero.

Header Error Control (HEC): Calcula el *checksum* sólo de los primeros 4 bytes del encabezado. HEC puede corregir solo un bit erróneo de estos 4 bytes, por eso preserva la celda en vez de descartarla.

Ventajas y desventajas

Ventajas[13]:

- Una única red ATM tendrá cabida a todo tipo de trafico(voz, datos y vídeo). ATM mejora la eficiencia y manejabilidad de la red.
- Capacita nuevas aplicaciones debido a su alta velocidad y a la integración de los tipos de trafico, ATM capacitará la creación y la expansión de nuevas aplicaciones como las multimedia.
- Compatibilidad, ya que ATM no está basado en un tipo específico de transporte físico, y por ende, es compatible con las actuales redes físicas que han sido desplegadas. ATM puede ser implementado sobre par trenzado, cable coaxial y fibra óptica.
- Simplifica el control de la red, pues ATM esta evolucionando hacia una tecnología estándar para todo tipo de comunicaciones. Esta uniformidad intenta simplificar el control de la red usando la misma tecnología para todos los niveles de la red.
- Largo periodo de vida de la arquitectura- Los sistemas de información y las industrias de telecomunicaciones se están centrando y están estandarizado el ATM. ATM ha sido diseñado desde el comienzo para ser flexible en distancias geográficas, número de usuarios, acceso y ancho de banda.

Desventajas[14]:

- Muchos analistas de la industria ven a ATM como un término largo, una tecnología estratégica, y hacia la que finalmente todas las LAN tenderán. Sin embargo, ATM es radicalmente distinto a las tecnologías LAN de hoy en día, lo cual hace que muchos conceptos tomen años en ser estandarizados. Los sistemas operativos actuales y las familias de protocolos en particular, requerirán de modificaciones significativas con el fin de soportar ATM. Esto será muy costoso, molesto y consumirá tiempo.
- Algunas personas pagarán mucho por estar en la punta de la tecnología, pero en estos momentos, las actuales tecnologías de alta velocidad como FDDI y *Fast Ethernet* proveerán rendimiento a precios que

los productos ATM no serán capaz de competir. Sólo una vez que las ventas de ATM alcancen volúmenes significativos, el costo de los productos podrán competir con la tecnología de hoy en día.

Implementación

Esta tecnología se puede implementar de la siguiente manera[8]:

Redes de empresa homogéneas:

ATM puede utilizarse para crear una verdadera red homogénea a través de una gran compañía. También puede utilizarse como una red de área local altamente efectiva, como un *backbone* en un campus, como red de área metropolitana, como red de área extensa, o como una combinación de todas las anteriores. Es concebible que redes de grandes empresas estén basadas principalmente en ATM, con una infraestructura que cubra la empresa entera. Esta red ATM soportaría tráfico multimedia, es decir, todo tipo de tráfico transportado por una red única y homogénea.

Grupos de trabajo virtuales:

Con ATM como núcleo principal de la red de una empresa, los usuarios remotos pueden pertenecer al mismo grupo de trabajo sin notar el impacto de la distancia geográfica mientras se comunican con miembros del mismo grupo. ATM conmuta y transmite las celdas sobre los enlaces de alta velocidad proporcionando un retardo muy bajo independientemente de la localización. Las limitaciones físicas de las redes de hoy desaparecen, y la red se convierte transparente para las aplicaciones remotas.

Desarrollos en colaboración:

Los departamentos de ingeniería de diferentes países pueden trabajar conjuntamente en la especificación de un nuevo diseño, utilizando una aplicación de conferencia para documentación sobre una red ATM. El documento podría ser un sencillo texto, o un documento complejo constando de una combinación de texto,

gráficos de alta resolución, anotaciones de voz y un vídeo clip. Los beneficios resultantes incluyen un mejor diseño, aumento de la productividad, y un menor tiempo para su comercialización.

Computación distribuida con uso intensivo de ancho de banda:

Con la difusión de la arquitectura cliente-servidor y el rápido aumento del número de servidores, se necesita un mayor ancho de banda. Con la escalabilidad de ATM, el ancho de banda de la red se puede incrementar añadiendo puertos de acceso a los conmutadores o incrementado la velocidad de algunos de los puertos. Cuando los 155 Mbps destinados a un servidor se convierten en un cuello de botella, se puede añadir una interfase de 622 Mbps sin impacto sobre el resto de la red. El beneficio es la protección de la inversión en la infraestructura de red.

Vídeo conferencia de sobremesa multi-ventana:

Una red ATM proporciona una alta calidad a un coste efectivo en el transporte de múltiples tipos de información. Por ejemplo, un grupo de ejecutivos podría revisar los planes comerciales de un nuevo producto, un equipo de científicos podría revisar los resultados de un nuevo experimento, un equipo de doctores podría diagnosticar a un paciente en una clínica remota. La información podría ser un documento complejo, un vídeo con movimiento en tiempo real de un experimento científico, o una combinación de radiografías, cardiogramas e imágenes TAC. Los beneficios serían menos viajes, mejor utilización de los recursos humanos costosos (tales como ejecutivos, científicos y doctores), y una comunicación muy superior a la de la voz.

Soporte y formación remota:

Un cliente llama, al centro de soporte del vendedor, con un problema. El vendedor inmediatamente obtiene sobre su pantalla la información acerca del cliente, y le transfiere al ingeniero de soporte apropiado para revisar su problema. El cliente envía un vídeo clip con los síntomas del problema, o muestra el problema en tiempo real según está ocurriendo en vídeo en movimiento, junto con los informes de diagnósticos previamente capturados. El suministrador trabaja con el cliente remotamente para resolver el problema en tiempo real. Los beneficios serían una rápida respuesta al cliente, una mejora de las relaciones entre el cliente y el suministrador, y ahorros de gastos para ambos.

Problemas para la implementación

En el pasado los protocolos de comunicaciones de datos evolucionaron en respuesta a circuitos poco confiables. Los protocolos en general detectan errores en bits y tramas perdidas y luego retransmiten los datos. Tal vez los usuarios jamás vean estos errores reportados y la degradación de la respuesta serían los únicos síntomas.

A diferencia de los mecanismos de control extremo a extremo que utiliza TCP en el *internerworking*, la capacidad de Gbit/seg de la red ATM genera un conjunto de requerimientos necesarios para el control de flujo. Si el control del flujo se hiciese como una realimentación del lazo extremo a extremo, en el momento en que el mensaje de control de flujo arribase a la fuente, ésta habría transmitido ya algunos Mbytes de datos en el sistema, exacerbando la congestión. Y en el momento en que la fuente reaccionase al mensaje de control, la condición de congestión hubiese podido desaparecer apagando innecesariamente la fuente. La constante de tiempo de la realimentación extremo a extremo en las redes ATM (retardo de realimentación por producto lazo - ancho de banda) debe ser lo suficientemente alta como para cumplir con las necesidades del usuario sin que la dinámica de la red se vuelva impráctica.

Las condiciones de congestión en las redes ATM están previstas para que sean extremadamente dinámicas requiriendo de mecanismos de *hardware* lo suficientemente rápidos para llevar a la red al estado estacionario, necesitando que la red en sí, éste activamente involucrada en el rápido establecimiento de este estado estacionario. Sin embargo, esta aproximación simplista de control reactivo de lazo cerrado extremo a extremo en condiciones de congestión no se considera suficiente para las redes ATM.

II.1.3 PPP

Introducción

El protocolo PPP (*Point- to- Point Protocol*) originalmente surgió como un protocolo de encapsulación para transportar tráfico IP a través de enlaces *point-to-point* (punto a punto). PPP también estableció un estándar para la asignación y gestión de las direcciones IP, encapsulación asíncrona (*start/stop*) y síncrona orientada a bit (*bit-oriented*), multiplexación de protocolos de red, configuración del enlace, pruebas de calidad de enlace, detección de errores y negociación de opciones para ciertas características como la negociación de las direcciones de la capa de red y la negociación de compresión de datos. PPP soporta estas funciones por medio de un Protocolo de Control de Enlace (*LCP, Link Control Protocol*) extendible y una familia de Protocolos de Control de Red (*NCP, Network Control Protocol*) para negociar las configuraciones opcionales de operaciones y facilidades. En adición a IP, PPP soporta otros protocolos, incluyendo IPX (*Internetwork Packet Exchange* de Novell) y DECnet[15].

Topología

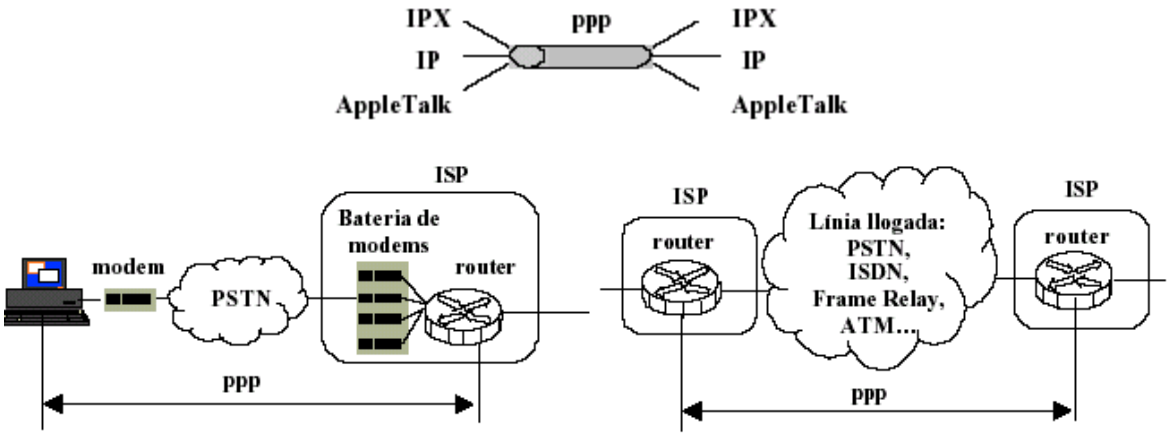


Figura II.1.3.1

PPP tiene una gran variedad de usos típicos pues es la tecnología con la que actualmente se logra una conexión a Internet desde cualquier hogar a través de un módem. La figura sólo muestra algunos ejemplos de los usos de PPP.[16]

Descripción y Características

PPP está diseñado para enlaces simples para el transporte de paquetes entre dos puntos, estos enlaces proveen una comunicación *full-duplex* simultánea y se asume que hace una entrega de paquetes en orden. PPP provee una solución para una fácil conexión de una gran variedad de *host*s, *bridges* y enrutadores. Se tienen las siguientes características:

- **Encapsulación**: La encapsulación en PPP provee un protocolo diferente de multiplexación simultánea de la capa de red a través del mismo enlace. Fue cuidadosamente diseñado tener compatibilidad con la mayoría de los dispositivos comunes.
- ***Link Control Protocol*** (LCP): Para ser suficientemente versátil y portable en una gran variedad de ambientes, PPP usa LCP. LCP se usa para hacer un acuerdo automático en las opciones del formato de encapsulación, tener un manejo de varios límites de tamaños de los paquetes, detectar los enlaces *Loop-Back* y los errores de configuración, y terminar el enlace. Otras facilidades opcionales pueden ser la verificación de la identidad de los puntos del enlace, y puede determinar cuándo un enlace es funcional y cuándo falla.
- **Network Control Protocols (NCP):** Los enlaces punto a punto (*point-to-point*) tienden a exacerbar muchos problemas con la familia de protocolos de red. Por ejemplo, la asignación y gestión de direcciones IP, el cual es un problema aún en ambientes LAN, es especialmente difícil sobre enlaces de conmutación de circuitos punto a punto (como mucho de los módems servidores *dial-up*) . Estos

problemas son manejados por una familia de NCP's, cada uno de los cuales maneja una necesidad específica requerida para su respectivo protocolo perteneciente a la capa de red[17].

Funcionamiento

Con el fin de establecer comunicaciones sobre un enlace punto a punto (*point-to-point*), cada punta del enlace PPP requiere mandar primero paquetes LCP para configurar y probar el enlace de datos. Después de que el enlace está establecido, el nodo puede ser verificado. Entonces, PPP requiere mandar paquetes NCP para escoger y configurar uno o más protocolos de la capa de red. Cada uno de estos protocolos de la capa de red escogidos debe ser configurado para que los paquetes de cada protocolo puedan ser enviados sobre este enlace.

El enlace podrá permanecer configurado para tener comunicación hasta que paquetes LCP o NCP cierren el enlace, o hasta que algún evento externo ocurra (la intervención del administrador de la red o cuando el tiempo de conexión expira).

En el proceso de configuración, mantenimiento y terminación de los enlaces punto a punto (*point-to-ponit)*, el enlace PPP va a través de distintas fases las cuales son especificadas en el siguiente diagrama:

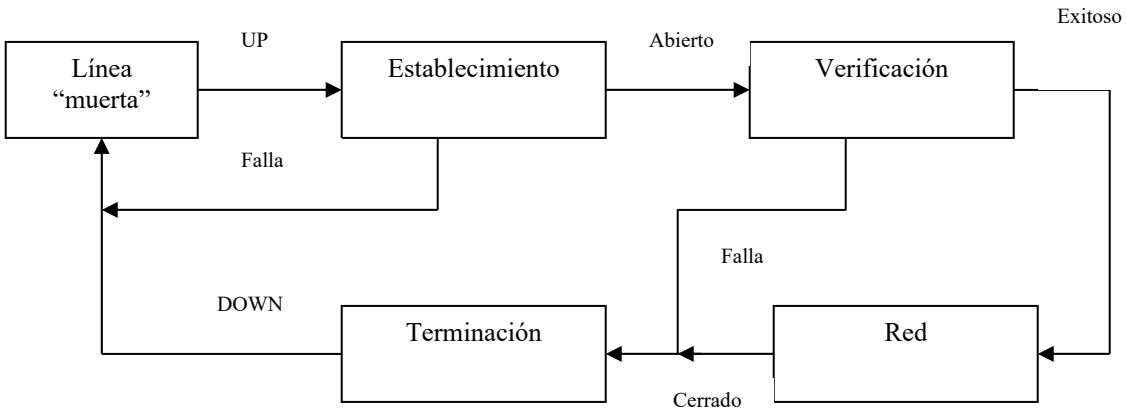


Figura II.1.3.2

- **Fase de Línea Muerta:** Es cuando la capa física no está lista. El enlace necesariamente debe terminar y comenzar en esta fase. Cuando un evento externo indica que la capa de red esta lista para ser usada, PPP procederá al establecimiento del enlace. Durante esta parte, el LCP pueden estar en los estados iniciales terminales. La transición a la fase de establecimiento señalará un evento *UP* al LCP. Típicamente, un enlace regresará a esta fase automáticamente después de la desconexión del módem. En el caso un enlace *hard-wired*, esta fase puede ser extremadamente corto (solo el que se necesita para detectar la presencia del dispositivo).
- **Fase de Establecimiento del enlace:** El LCP es usado para establecer la conexión a través de un intercambio de configuración de paquetes. Este intercambio es completo, y el LCP entra en un estado de "Abierto", una vez ocurrido esto un paquete *Configure-Ack* es enviado y recibido. Es importante notar que sólo las opciones de configuración, las cuales son independientes de los protocolos de la capa de red, son configurados por LCP. La configuración de un protocolo de la capa de red en particular es manejado por diferentes NCP durante la fase del protocolo de la capa de red. Cualquier paquete recibido durante esta fase que no sea LCP, requiere ser descartado. Recibir un paquete LCP *Configure-Request* causa un retorno a la fase de establecimiento del enlace desde la fase del protocolo de la capa de red o la fase de verificación.
- **Fase de verificación:** En algunos enlaces puede ser deseable requerir una verificación del nodo por sí mismo antes de permitir paquetes del protocolo de la capa de red. Por estándar, la verificación no es obligatoria. Sin embargo, si alguna implementación requiere que el nodo sea verificado con algún protocolo de seguridad, se requerirá en esta fase ese procedimiento. La verificación debe tomar lugar lo más pronto posible después del establecimiento del enlace. Sin embargo, la determinación de la calidad del enlace puede ocurrir consecuentemente. Una implementación no puede permitir el intercambio de paquetes de determinación de calidad del enlace para retardar la autentificación indefinidamente. El avance de la fase de verificación a la fase del protocolo de la capa de red puede no ocurrir hasta que la

verificación se complete. Si la verificación falla, se deberá proceder a ir a la fase de terminación del enlace.

- **Fase del protocolo de la capa de red:** Una vez que PPP termina las fases previas, cada protocolo de la capa de red (IP, IPX, AppleTalk) requiere ser configurado por separado para asignar un NCP apropiado. Cada NCP puede ser "Abierto" o "Cerrado" en cualquier tiempo. Después de que un NCP alcanza el estado "Abierto", PPP llevará los paquetes correspondientes del protocolo de la capa de red. Cualquier paquete del protocolo de la capa de red recibido cuando el NCP no está en estado "Abierto" debe ser descartado. Durante esta fase, el tráfico del enlace consiste en la combinación de paquetes LCP, NCP o del protocolo de la capa de red.
- **Fase de Terminación del enlace:** PPP puede terminar el enlace en cualquier momento. Esto puede pasar debido a la pérdida de la portadora, a una falla en la verificación, o en la calidad del enlace, por la terminación del tiempo de uso o en un cierre del enlace por medio del administrador. LCP es usado para cerrar el enlace a través del intercambio de paquetes de terminación. Cuando un enlace se cierra, PPP informa a los protocolos de la capa de red para que cada uno de ellos tome la acción apropiada. Después de intercambiar estos paquetes de terminación, la implementación debe mandar una señal a la capa física de desconexión para reforzar esta terminación, particularmente en el caso de una falla de verificación. El nodo que envía la propuesta de terminación debe desconectarse después de recibir el *Terminate-Ack*, o después de reiniciar el contador de expiración. El nodo que recibe la propuesta de terminación debe esperar al otro nodo para desconectarse y no requerirá la desconexión hasta que al menos el tiempo de reinicio haya pasado después de mandar un *Terminate-Ack*. Después de esto, PPP debe proceder a la fase de línea muerta. Cualquier paquete recibido que no sea LCP durante esta fase, tiene que ser descartado[17].

Formato del *frame*

El formato del *frame* completo es[18]:

Indicador (1 byte)	Dirección (1 byte)	Control (1 byte)	Protocolo (1 o 2 bytes)	Información (variable)	Suma (2 o 4 bytes)	Indicador (1 byte)

Figura II.1.3.3

En donde todas las tramas comienzan con el byte Indicador que es igual a “01111110”. Luego viene el campo Dirección, al que siempre se asigna el valor “11111111”. La Dirección va seguida del campo de Control, cuyo valor predeterminado es “00000011”. Este valor indica un marco sin número, ya que PPP no proporciona transmisión confiable (usando números de secuencia y acuses), pero en ambientes ruidosos se puede usar un modo numerado para transmisión confiable. El penúltimo campo es el de Suma de Comprobación, que normalmente es de 2 bytes, pero puede negociarse una suma de 4 bytes. La trama finaliza con otro byte Indicador “01111110”[18].

Como los campos arriba descritos tienen valores predeterminados, es conveniente enfocarse en el campo de protocolo e información (Información y relleno):

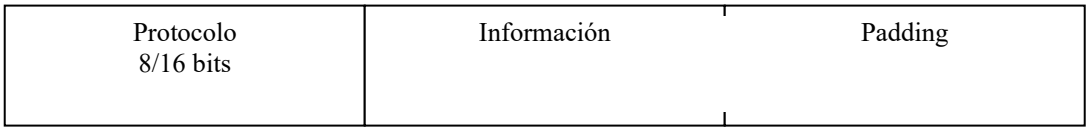


Figura II.1.3.4

- **Protocolo***:* Este campo es de uno o dos bytes. Estos valores identifican al paquete a encapsular dentro del campo de información. El campo es transmitido y recibido de acuerdo a su byte, siendo primero el más significativo. La estructura de esta campo depende del mecanismo de campos de dirección OSI 3309. El último bit significativo del último byte significativo debe de ser igual a “1”. También, todos los protocolos deben de ser asignados de forma que el bit menos significativo del byte más significativo es igual a “0”. Los *frames* recibidos sin estas características deben de ser tratados como protocolos desconocidos.

  Los valores del campo de protocolo en el rango de “0***” a “3***” son paquetes específicos de protocolos de la capa de red; los valores en el rango de “8***” a “b***” son paquetes pertenecientes a NCP . Los valores de “4***” a “7***” son usados para protocolos de un bajo volumen de tráfico, los cuales no están asociados a los NCP. Los valores en el rango de “c***” a “f***” identifican a paquetes del Protocolo de Control de la capa del enlace, como LCP.

- **Información:** Este campo puede ser de cero o más bytes. Este campo de información contiene los paquetes del protocolo especificado en el campo Protocolo. La máxima longitud es variable (incluye al campo *Padding*, pero no incluye al campo Protocolo) y está determinado por la MRU *o Maximum Recive Unit* (Unidad Máxima de Recepción) la cual por estándar es de 1500 bytes. Por medio de una negociación previa, PPP puede usar otros valores de MRU.

- ***Padding*:** En las transmisiones, el campo Información puede ser rellenado con un número arbitrario de bytes por encima del MRU. Cada protocolo debe ser capaz de distinguir estos bytes de la información real[17].

Ventajas y desventajas

Ventajas[19]:

- Efectúa detección de errores
- Reconoce múltiples protocolos
- Se negocian las direcciones IP al momento de la conexión
- Proporciona verificación de autenticidad.

Desventajas[18]:

- Complejo.
- Encabezado relativamente grande.

Implementación

El implementador puede especificar mejoras a la configuración estándar, la cual es transmitida al nodo sin la intervención del operador. Finalmente, el operador puede configurar explícitamente las opciones del enlace para habilitar al enlace para trabajar en ambientes donde puede ser imposible manejarlo con la implementación estándar. Esta configuración hecha es implementada a través de un mecanismo de opciones de negociación, en donde cada extremo del enlace describe sus características y requerimientos. Esto significa que el mecanismo de negociación de las opciones está determinado por los LCP. Las mismas facilidades estás diseñadas para ser usadas por otros protocolos, especialmente con la familia de NCPs[17].

II.2. Protocolos de tuneleo

II.2.1. IPsec

Introducción

[20]IPsec fue desarrollado para proporcionar funciones de verificación y cifrado para las redes IP existentes y futuras. El Grupo de Trabajo de Seguridad IP (*IP Security Working Group*), establecido por la IETF, desarrolló este sistema, teniendo una primera edición en 1995. El desarrollo de IPsec originalmente pensaba aplicarse al protocolo IP de próxima generación, IPv6. Esto es una razón de peso para emplear VPNs con IPsec. Como IPsec fue desarrollado principalmente para propósitos de verificación y cifrado, se le ha implementado una opción de modo para tuneleo, lo cual se explicará con detalle más adelante.

IPsec es una característica obligatoria en IPv6, pero no en IPv4, aunque se ha adaptado para trabajar también con esa versión. Como el "despegue" de IPv6 ha sido lento, es decir, aún no se ha expandido su uso, la necesidad de proteger paquetes transmitidos sobre Internet ha sido la principal causa de que IPsec se haya adaptado a IPv4. Esta característica hace a IPsec particularmente atractiva para ser el protocolo base de una VPN, pues con él se garantiza que la VPN funcionará tanto para las redes actuales como para las futuras. Sin embargo, cabe mencionar que es necesario modificar el *stack* de IPv4 para soportar IPsec.

Los servicios ofrecidos por IPsec incluyen control de accesos, integridad a las conexiones no orientadas a conexión, verificación del origen de los datos, protección contra reenvíos, confidencialidad (cifrado) y control limitado de flujo. Estos servicios se proporcionan en la capa de IP (capa 3), pero la protección también se ofrece para protocolos de capas superiores e inferiores.

Topología

Con IPsec es posible establecer un túnel para comunicar dos puntos remotos. La comunicación entre dichos puntos se puede dar a través de Internet, con lo cual se ahorra significativamente en costos. Un bosquejo de una comunicación lograda con IPsec, se muestra en la siguiente figura.

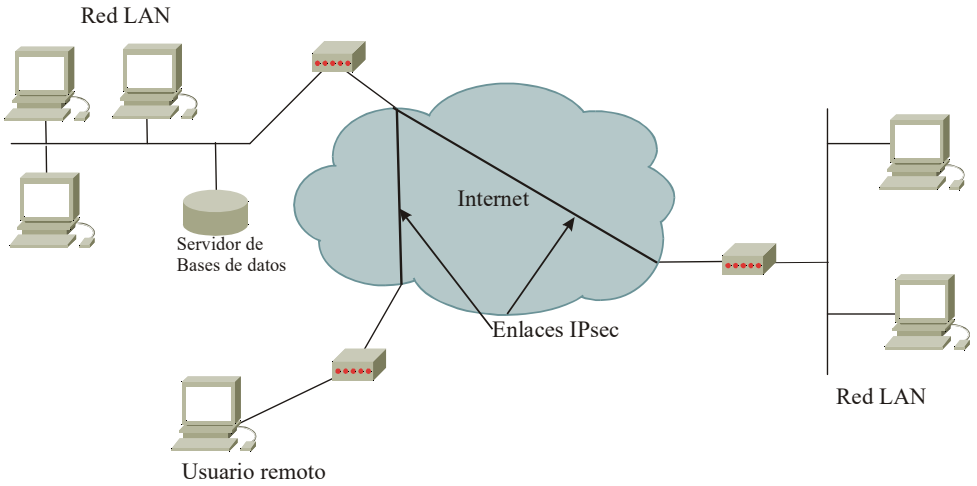


Figura II.2.1.1

Descripción

Para comprender IPsec, se puede describir su arquitectura dividiéndola en tres secciones importantes[21]:

- La primera parte (y la menos usada) se llama Encabezado de Verificación o Authentication Header (AH).
- La segunda sección se conoce como ESP o *Encapsulating Security Payload*, que proporciona el cifrado de los datos y, opcionalmente, los verifica (ampliamente recomendado).
- Finalmente, y muy importante, el sistema de administración de llaves.

El Encabezado de Verificación o AH proporciona integridad y verificación a los datos. La integridad de los datos ofrece una garantía de que el paquete no puede ser modificado mientras transita a través de la red. También provee al receptor con la capacidad de verificar al transmisor y protege contra reenvíos o ataques a la red.

El ESP se usa para obtener confidencialidad en los mensajes y, si se desea, también verificación. La confidencialidad se proporciona con el uso de un algoritmo de cifrado que no es fijo, pues el encabezado de ESP proporciona la capacidad de elegir diferentes tipos de algoritmos. El algoritmo de cifrado se aplica no sólo a los datos, sino a todo el paquete, incluyendo el encabezado y el relleno. La verificación ESP se proporciona usando el campo Verificación de Datos o *Checksum*, el cual verifica la integridad de los datos.

La última sección es una parte importante de IP. El sistema de administración de llaves puede dividirse en dos subsecciones: en sistemas automáticos y en sistemas manuales, ambos con uso común. Un sistema automático es más difícil de implementar, por lo que un sistema manual sería la mejor opción en compañías con redes pequeñas. El sistema manual requiere que el administrador del sistema configure cada una de las conexiones de la red con un conjunto propio de llaves para cada empleado y un servidor de verificación para la red privada. El sistema automático generalmente se emplea con el Protocolo de Determinación de Llaves Oakley (*Oakley Key Determination Protocol*) o con el algoritmo ISAKMP (*Internet Security Association and Key Management Protocol*).

El algoritmo *Oakley* utiliza el algoritmo de Intercambio de Llaves Diffie Hellman (*Diffie Hellman Key Exchange*) como un medio seguro para intercambiar datos entre dos partes. Desafortunadamente, este sistema no usa algún mecanismo de verificación, por lo que es vulnerable a ataques. La verificación en el intercambio de llaves se puede completar en tres diferentes formas[22]:

- La primera usa un intercambio público de llaves cifradas, lo cual significa que una punta del enlace cifra sus identificaciones con su propia llave de cifrado. Obviamente, las identificaciones se descifran sólo con la llave correspondiente.
- La segunda usa una firma digital, la cual se usa para colocar la información en una función *hash* y la cifra usando una llave privada.
- La última alternativa es el cifrado simétrico de llaves, el cual se emplea en conjunto con otro sistema para generar una llave, la cual es usada entonces para el cifrado simétrico en el intercambio de parámetros.

[23]El algoritmo de Intercambio de Llaves Diffie Hellman (*Diffie Hellman Key Exchange*) es el primer algoritmo público de llaves que se produjo. Es capaz de proporcionar una llave de cifrado para un intercambio seguro entre dos partes. Trabaja para cada usuario a partir de dos parámetros globales, q y α. Cuando dos partes, A y B, quieren intercambiar datos cifrados, el usuario A selecciona aleatoriamente un entero *Xa*, el cual será su llave privada, la combina con la variable global α y usa el resultado como su llave pública de cifrado. El usuario B hace lo mismo usando la variable goblal q. Este proceso tiene como resultado que ambas partes tengan una llave de cifrado que sólo ellos conocerán. La llave de cifrado cambia con cada intercambio de datos que se dé.

[21]ISAKMP no es muy diferente al sistema *Oakley*, aunque está más enfocado en la definición de procedimientos, protocolos y asociaciones de seguridad (simplemente conocidas como SA, *Security Associations*). Consta de dos fases, la primera de ellas se usa para establecer un canal seguro (el cual se hace con el sistema de intercambio Oakley). La segunda fase establece una asociación de seguridad del protocolo,

logrado a partir del conjunto de protocolos de seguridad y usados para transformación de paquetes, para los cuales son válidas las llaves.

Un bosquejo de la arquitectura de IPsec, según la adaptación de la IETF, se muestra en la siguiente figura:

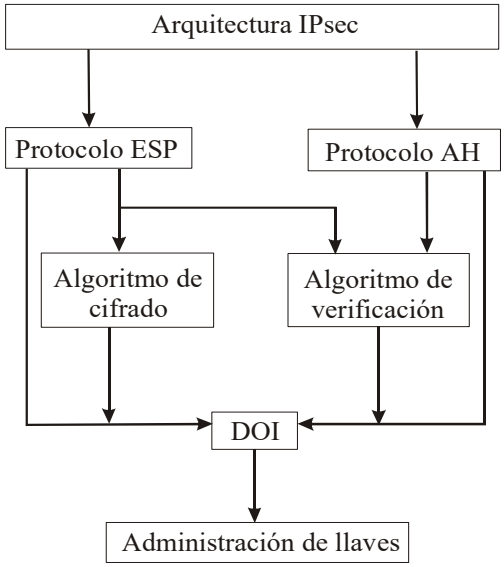


Figura II.2.1.2

La porción DOI agrupa los protocolos usados en la negociación de una SA. El DOI especifica efectivamente cómo entablar la comunicación y qué protocolos y formatos usar[22].

Funcionamiento

[20]Una implementación IPsec opera en un *host* o en un *gateway* de seguridad, buscando siempre la protección del tráfico IP. Dicha protección está basada en los requerimientos definidos en la Base de Datos de las Políticas de Seguridad (SPD, *Security Policy Database*), la cual es establecida y mantenida por un usuario o un administrador del sistema.

En general, los paquetes son seleccionados por uno de los tres modos de procesamiento basados en IP y en la información del encabezado de la capa de transporte que, a su vez, deben coincidir con las entradas de la

SPD. Cada paquete debe disponer de los servicios de seguridad de IPsec, pudiendo ser descartado o admitido por IPsec según las políticas aplicables de la base de datos.

IPsec permite que el usuario (o en su caso, el administrador del sistema) tenga el control de los servicios que se ofrecen. Por ejemplo, se puede crear un túnel para transportar todo el tráfico entre dos *gateways* de seguridad o se puede crear un túnel cifrado separado por cada conexión TCP entre cada par de *hosts* que se comunican a través de dichos *gateways*.

La administración de IPsec debe incorporar facilidades específicas, como qué servicios se pueden usar y en qué combinaciones, la forma en la que una protección de seguridad dada debería ser aplicada o los algoritmos usados para seguridad cifrada. Debido a que dichos servicios de seguridad usan valores secretos compartidos (llamados llaves cifradas[1]), IPsec cuenta con un conjunto de mecanismos separados para colocar tales llaves en su lugar.

Asociaciones de seguridad

Una asociación de seguridad, mejor conocida como SA (*Security Association*), es una conexión sencilla que transporta el tráfico de los servicios de seguridad logrados por AH o por ESP. En caso de que se aplique la protección de AH y de ESP al mismo tiempo, se crean dos o más SAs para brindar seguridad al tráfico. Para proteger una comunicación bidireccional típica entre dos *hosts*, son necesarios dos SAs (uno para cada dirección)[20].

Una SA se identifica únicamente por tres parámetros: un índice del parámetro de seguridad SPI (*Security Parameter Index*), una dirección IP destino y un identificador del protocolo de seguridad (ya sea AH o ESP). En principio, la dirección IP destino debe ser *unicast*, pues los mecanismos del sistema de administración de IPsec sólo están definidos para SAs *unicats*. Los SAs se describen para comunicación punto a punto, aunque el concepto también es aplicable a comunicaciones punto a multipunto.

[1] Estas llaves son usadas para servicios de verificación, integridad y cifrado.

Existen dos tipos de SAs: las modo transporte y las modo túnel:

Modo transporte:

Una SA modo transporte es una asociación de seguridad entre dos *hosts*. En este modo, los datos se verifican y se cifran únicamente en la capa de transporte (capa 4), lo cual deja disponible al encabezado IP para sufrir modificaciones, teniendo como riesgo que una VPN quede abierta y sea objetivo de varias formas de ataques[20].

En el caso de ESP, un SA modo transporte proporciona servicios de seguridad sólo para protocolos superiores y no para el encabezado IP o alguna extensión de los encabezados que precedan al encabezado ESP. En el caso de AH, la protección se extiende para ofrecerla a las porciones seleccionadas (previamente) del encabezado IP, además de las extensiones y opciones deseadas[20].

Modo túnel:

Un SA modo túnel es esencialmente un SA aplicado a un túnel IP. Si el extremo de un SA es un *gateway* de seguridad, el SA debe ser modo túnel. Esto significa que un SA entre dos *gateways* de seguridad o entre un *host* y un *gateway*, siempre es un SA modo túnel. Esto se debe a que se pretende evitar problemas potenciales con la fragmentación y el reensamblado de paquetes, lo que se dificulta en transmisiones con diferentes rutas. Existe un caso especial, pues si el tráfico de datos está destinado a un *gateway* que actúe como *host*, se permite el modo transporte (ya que el *gateway* no actúa como tal) [20].

En este modo se cifra el paquete IP completo y le agrega un nuevo encabezado IP. Debido a que el encabezado IP original está completamente cifrado y encapsulado, cualquier analizador de red que trate de determinar las redes LAN que se están comunicando no podrá saberlo, pues sólo es posible saber las direcciones IP de los enrutadores *gateways* a los que están conectadas las LANs. Sin embargo, sí es posible

obtener información a partir de los *gateways*, como los nombres de las compañías que se comunican y qué oficinas tienen exceso de tráfico.

Modos combinados:

Los modos túnel y transporte pueden ser combinados para añadir seguridad a la red. Un ejemplo común de este caso es usar el modo túnel para verificar o cifrar el paquete y procesarlo en el modo transporte para protección adicional[22].

Las SAs se pueden combinar en dos formas:

- **Adyacencia en el transporte***:* se refiere a la aplicación de dos o más protocolos de seguridad al mismo paquete IP sin involucrar tuneleo. En esta combinación se mezclan AH y ESP en un solo nivel.

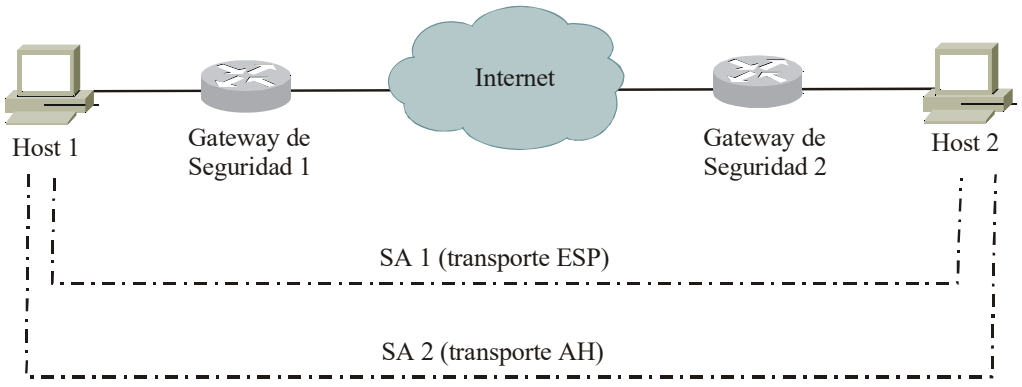


Figura II.2.1.3

- **Tuneleo iterado:** se refiere a la aplicación de múltiples capas de protocolos de seguridad efectuados a través del tuneleo IP. Se permiten varios niveles de anidación y cada túnel se puede originar o terminar en un sitio diferente IPsec a lo largo de la ruta. Existen tres casos básicos:

1. Los dos extremos de la SA son iguales: el túnel interno o el externo puede ser AH o ESP:

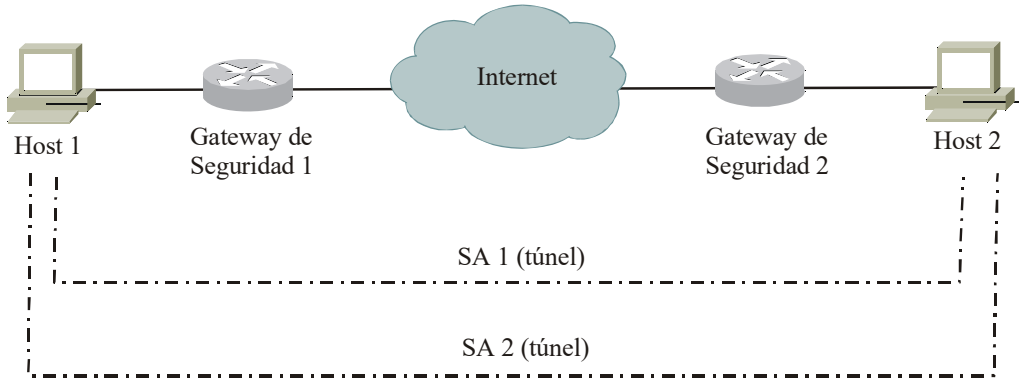


Figura II.2.1.4

2. Uno de los extremos de los SAs es el mismo:

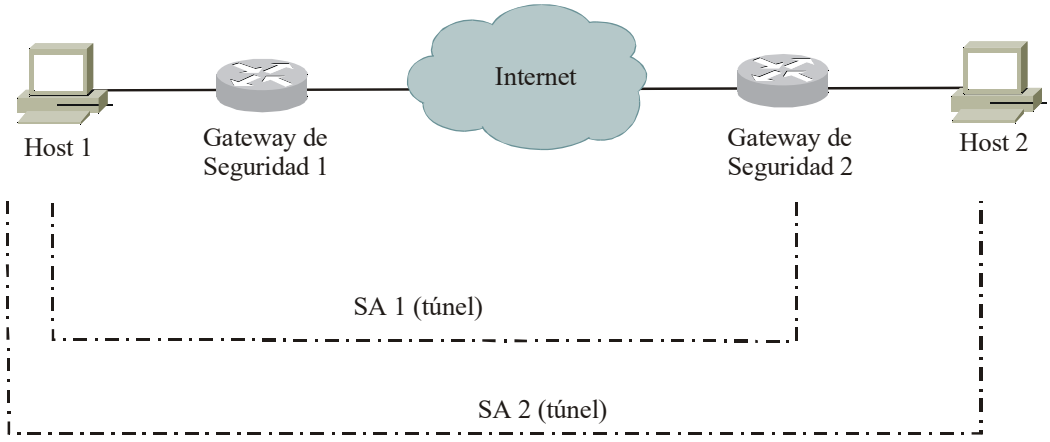


Figura II.2.1.5

3. Ninguno de los puntos es el mismo:

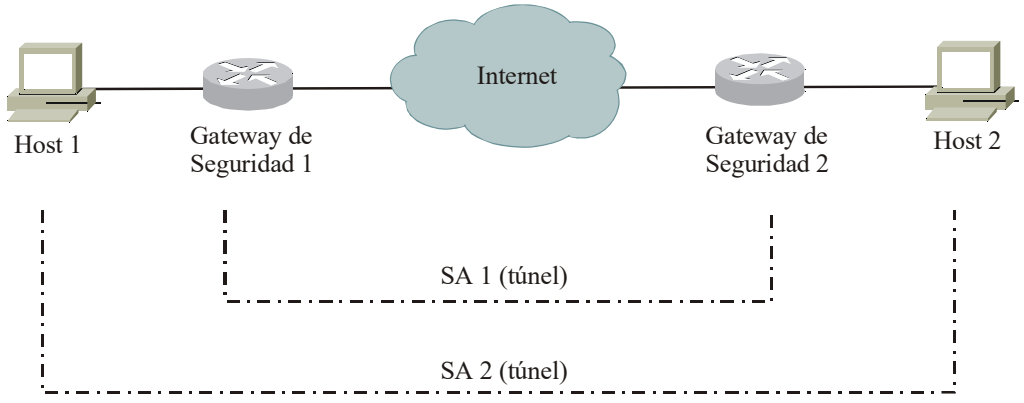


Figura II.2.1.6

Formato el *frame*

En el encabezado IP externo, los campos Dirección Destino y Dirección Origen identifican los extremos del túnel, es decir, el punto encapsulador y el desencapsulador. En el encabezado IP interno, esos mismo campos identifican el transmisor original y el "recipiente" del paquete (desde el punto de vista de ese túnel), respectivamente[20].

El encabezado IP interno no es modificado durante su transmisión por el túnel, excepto en el campo TTL. Tampoco sufren cambios las opciones IP o las extensiones de ese encabezado[23].

Para un modo transporte[22]:

En IPv4, aparece un encabezado del protocolo de seguridad en modo transporte inmediatamente después de las opciones del encabezado IP y antes de los encabezados de protocolos superiores. (como TCP o UDP). En IPv6, el encabezado del protocolo de seguridad en modo transporte aparece después de las extensiones del encabezado IP, pero antes de las opciones del encabezado del protocolo más alto (según el modelo OSI).

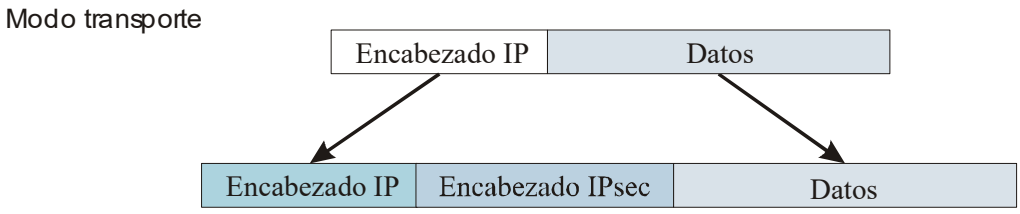


Figura II.2.1.7

Para el modo túnel[22]:

Existe un encabezado IP externo que especifica el destino del proceso IPsec y un encabezado IP interno que especifica el último destino del paquete. El encabezado del protocolo de seguridad aparece después del encabezado IP externo y antes del interno. Si se emplea AH en el modo túnel, el encabezado IP externo tiene protección, así como el paquete IP tuneleado, es decir, también el encabezado IP interno y los datos. Si se emplea ESP, la protección se ofrece solamente al paquete tuneleado y no al encabezado IP externo.

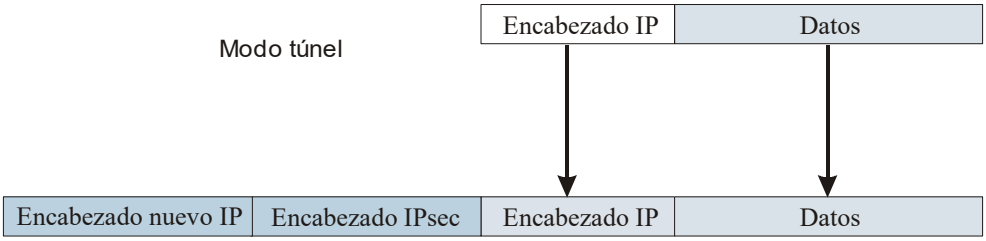


Figura II.2.1.8

En caso de ser necesario, se inserta el encabezado del protocolo de seguridad entre el encabezado IP externo y el interno.

En el proceso de IPsec se observan ciertas características en los encabezados[23]:

1. La versión IP en el encabezado del encapsulado puede ser diferente al valor que aparece en el encabezado interno.
2. El campo TTL en el encabezado interno se decrementa en el enrutador encapsulador antes de reenviar el paquete y en el enrutador desencapsulador al reenviarlo. El campo *Checksum* cambia, por lo tanto, cuando el campo TTL cambia.
3. Las direcciones destino y origen dependen de la SA, la cual se usa para definir la dirección destino que determina qué dirección origen (interfaz de red) se usa para reenviar el paquete.
4. La configuración del nodo determina si se copia o se limpia el encabezado IP interno.
5. Si el encabezado IP interno es versión cuatro (campo Protocolo igual a 4), se copia el campo *Type Of Service* (TOS). Si es versión seis (campo Protocolo igual a 41), se mapea la clase al campo TOS.

Ventajas y desventajas

Ventajas[21]:

- IPsec ofrece a los usuarios la posibilidad de cifrar datos y/o verificar todo el tráfico que sea recibido en el nivel IP, proporcionando seguridad a la red, si así se desea.
- Trabaja en ambas versiones de IP. Aunque fue establecido para IPv6, es compatible con IPv4.
- Puede soportar diferentes aplicaciones
- Principalmente, ofrece conexiones seguras a través de Internet, lo cual da la posibilidad de que las empresas usen Internet en lugar de costosas redes privadas.
- Soporta accesos remotos, característica importante para empresas que tengan empleados que trabajen a distancia. Así, la compañía no necesita establecer redes enormes, pues el empleado simplemente se conecta al ISP local y paga sólo la llamada local.
- Una compañía puede utilizar IPsec para mejorar su sistema de seguridad. Esta ventaja se puede aprovechar en proteger las transacciones de los clientes de empresas que empleen un tipo de comercio electrónico.

Desventajas[20], [22]:

- No existe un estándar de compresión para IPsec. La necesidad de un estándar de compresión puede parecer superflua, pero cuando se cifran los datos, se vuelve realmente difícil comprimirlos, teniendo una fragmentación más grande de datos.
- Aún no están completos todos los estándares relativos a esta tecnología.

- La habilidad de proporcionar servicios diferenciados está restringida debido a que el encabezado IP original está cifrado y es imposible interactuar con los enrutadores intermedios. Este problema sólo existe en IPv4, pues en IPv6 ya se eliminó.
- A pesar de que IPsec ofrece seguridad de alta calidad, ésta depende de la implementación de la tecnología, lo cual escapa de la vista de los estándares relacionados. Además, la seguridad de una red está en función de diversos factores, incluyendo los personales, físicos, de procedimientos y de las prácticas de seguridad en cómputo que tengan los administradores. Esto hace que IPsec sólo sea una parte de toda una arquitectura de seguridad.
- La seguridad de una red lograda por el uso de IPsec es críticamente dependiente del ambiente de operación, ya que los defectos de seguridad del sistema operativo en el cual se ejecuta IPsec afecta la calidad de la seguridad. Tampoco los atributos de los ambientes en los que se implementa IPsec están dentro de los estándares.

Implementación

[20]Existen varias formas en las que IPsec puede ser implementado en un *host*, en un enrutador o en *firewall* (para crear un *gateway* de seguridad). Se describirán los casos más comunes en la implementación de IPsec:

a) Integración de IPsec en una implementación nativa de IP: requiere tener acceso a la fuente IP y es aplicable tanto a los dos *hosts* que han establecido comunicación, como a los *gateways* de seguridad correspondientes.
b) Implementación BITS (*Bump-in-the-stack*): IPsec se implementa "debajo" de un *stack* existente de IP, entre el IP nativo y los controladores de la red local. No se requiere tener acceso al código fuente del origen, lo que hace que este sistema sea apropiado para usar en sistemas protegidos. Esta implementación es común en *hosts*.
c) Implementación BITW (*Bump-in-the-wire*): el uso de procesador de cifrado es una característica común en el diseño de redes con estrictos sistemas de seguridad, tales como los del comercio electrónico y los

sistemas militares. Esta implementación debe diseñarse para ser aplicada a un *host*, a un *gateway* o a ambos. En el caso de que se aplique a un *host*, la implementación es similar a BITS, pero se debe soportar un enrutador o un *firewall* que opere como un *gateway* de seguridad. Generalmente, el dispositivo BITW es direccionable en IP.

II.2.2. GRE

Introducción

Existen diferentes propuestas (como la RFC 1234 y la RFC 1226) para el encapsulamiento de un protocolo sobre otro protocolo y se han propuesto otros tipos de encapsulación (RFC1241, RFC1479) para el transporte de IP sobre IP para ciertas políticas.

La encapsulación GRE (*Generic Routing Encapsulation*) es un mecanismo ampliamente usado para transportar paquetes que no son IP, aunque también es posible encapsular IP sobre IP. Esta técnica está disponible cuando las direcciones de los *hosts* están en un espacio privado de direcciones IP, pero necesitan ser transportadas sobre Internet[24].

Topología

Un túnel GRE se puede establecer comunicación entre dos usuarios remotos a través de Internet, teniendo una red parecida a la mostrada en la siguiente figura:

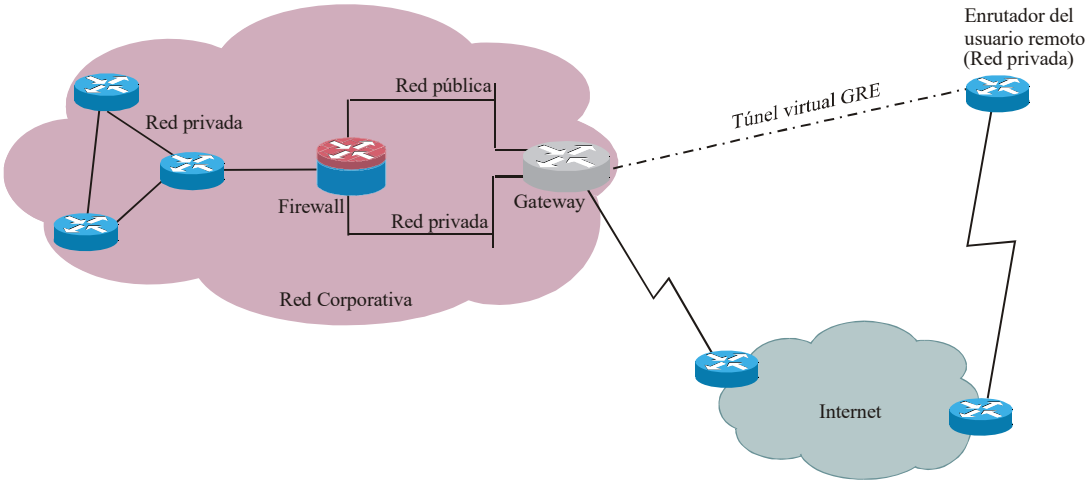


Figura II.2.2.1

En este escenario tenemos que todos los dispositivos verifican sus accesos, ya que todas las comunicaciones desde un usuario remoto hacia la red corporativa deben ser privadas y por ello, cifradas. Además, con GRE se puede determinar en la configuración de los enrutadores que las direcciones privadas se usen tanto en el usuario remoto como en la red corporativa[25].

Descripción

GRE fue desarrollado originalmente por Cisco y está definida en la RFC 1701. Este protocolo ha extendido su uso, aunque actualmente está siendo desplazado por IPsec. GRE llegó a ser parte de los estándares IETF cuando se encontró útil para ser la base de otros protocolos. El protocolo completo como base para otros está definido en la RFC 2784[25].

Desde la perspectiva de la "virtualización" de los enlaces, GRE da a los usuarios la apariencia de una línea dedicada IP o su equivalente en otras familias de protocolos, tal como IPX o AppleTalk, aunque lo más común es que GRE corra sobre IP, lo cual está definido en la RFC 1702. El contenido de los paquetes se identifica por un identificador del protocolo IP.

Funcionamiento

El protocolo GRE encapsula varios protocolos de red en túneles IP. Con GRE, un enrutador en cada sitio encapsula paquetes de protocolos específicos en un encabezado IP, creando un enlace virtual punto a punto a los enrutadores de los otros extremos de una nube IP, donde se retira el encabezado IP. GRE es capaz de manejar el transporte de tráfico multiprotocolo y tráfico IP *multicast* entre dos sitios que sólo tengan una conectividad IP *unicast*[26].

El tuneleo en GRE involucra tres tipos de protocolos[26]:

- Pasajero: es el protocolo encapsulado. Como ejemplo están IP, IPX, AppleTalk, etc.
- Portador: GRE proporciona los servicios de portador.
- Transporte: IP transporta al protocolo encapsulado.

El tuneleo GRE permite que otros protocolos diferentes a IP obtengan las funciones de seguridad construidas en IP, específicamente, permite la verificación de paquetes y proporciona confiabilidad a tales paquetes pues usa varias técnicas de cifrado diseñadas especialmente para el protocolo IP.

Después de que GRE encapsula paquetes dentro de paquetes IP estos son redireccionados a un *host* intermedio, el cual desencapsula el paquete y lo enruta hacia su siguiente salto. GRE es ejecutado por seudo-interfaces, las cuales simulan una conexión punto a punto. Son dos los modos de operación de estas interfaces:

- Encapsulación GRE (default): los paquetes salientes son encapsulados con un encabezado IP y el encabezado GRE especifica el tipo de paquete que se está encapsulando.
- Encapsulación móvil: sólo es aplicable para encapsulación IP. Los paquetes IP salientes son encapsulados con un encabezado pequeño y el encabezado original es modificado[27].

Para comprender el funcionamiento de GRE se puede considerar la red mostrada en la figura II.2.2.1. Los pasos que seguiría el establecimiento de un túnel para la comunicación entre un usuario remoto y la red corporativa serían[26]:

- Se construye un túnel entre el *gateway* y el usuario remoto. La función del túnel es proporcionar conectividad entre la red privada del usuario remoto y el espacio de direcciones de la red privada de la compañía.
- Cuando se configure el túnel entre las interfaces de los dos enrutadores, el origen y el destino del túnel se registran con direcciones IP.
- Si se desea, un enrutador remoto ejecuta NAT para que la comunicación entre el enrutador remoto e Intenet sea enrutada localmente y sólo las comunicaciones hacia la red corporativa se realicen a través del túnel encriptado.

Si se cuenta con un *firewall*, el *gateway* de la red corporativa tiene dos enlaces separados hacia el *firewall*: uno sobre una red que tiene una IP registrada y la otra sobre una red con una IP privada.

Se puede colocar un filtro en el *gateway* de la red corporativa para asegurarse de que únicamente las rutas del túnel GRE pasan por el enlace de la red privada hacia el *firewall*. En la base de datos local se puede colocar también un dispositivo de verificación[26].

Formato del *frame*

Un paquete GRE completo tiene un encabezado de entrega, un encabezado GRE y el encabezado original IP. La estructura general de un paquete GRE se muestra en la siguiente figura[24]:

Figura II.2.2.2

El encabezado de GRE versión cero es el siguiente[28]:

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
C	R	K	S	s	Recursión			Banderas					Versión			Tipo de protocolo															
Checksum																Offset															
Llave																															
Número de secuencia																															
Routing																															

Figura II.2.2.3

- C (*Checksum Present*): 1 bit. Si está en 1, entonces campo de *Checksum* está presente y contiene información válida. Si *C* o *R* están prendidos, los campos *Checksum* y *Offset* están presentes.
- R (*Routing Present*): 1 bit. Si está prendido, el campo de *Offset* está presente y contienen información válida. Si C o R están prendidos, los campos *Checksum* y *Offset* están presentes.
- K (*Key Present*): 1 bit. Si está prendido, el campo llave está presente y contiene información válida.
- S (*Sequence Number Present*): 1 bit. Si está prendido el campo Número de Secuencia está presente y contiene información válida.
- s (*Strict Source Route*): 1 bit. Es recomendable que este bit sólo se use si toda la información de ruteo consiste en Rutas Estrictas de Origen.
- Recursión (*Recursion Control*): 3 bits. Contiene el número de encapsulaciones adicionales que son permitidas. El valor por default es 0.
- Bandera :5 bits. Estos bits están reservados y por el momento deben de ser 0.
- Versión: 3 bits. Indica la versión de GRE, que en este caso debe ser 0.
- Tipo de protocolo: 16 bits.Contiene el tipo de protocolo del *Payload*. En general, el valor debe ser de Ethernet. Estos tipos se pueden ver en la RFC 1700.
- Checksum: 16 bits. Es opcional. Contiene el *checksum* del IP.
- Offset: 16 bits. Es opcional.

- Llave: 32 bits. Es opcional. Contiene un número que es insertado por el encapsulador. Este número puede ser usado por el receptor para hacer una verificación a la fuente de donde vino el paquete.
- Número de secuencia: 32 bits. Es opcional. Contiene un número que es insertado por el encapsulador. Este número puede ser usado por el receptor para establecer el orden en que el paquete fue transmitido.
- *Routing*: Variable. Es opcional, este campo en una lista de SRE (*Source Route Entry Packet*).

Ventajas y desventajas

Ventajas[24]:

GRE tiene una variedad de ventajas, tales como:

- Funciona como "remedio" para redes discontiguas
- Proporciona y administra VPNs
- Permite la existencia de partes discontiguas en áreas OSPF
- Maneja direcciones IP falsas y direcciones IP privadas sobre un núcleo público
- Transporta tráfico IP y no IP

Desventajas[24]:

El tuneleo GRE debería ser usado con cuidad debido a que puede distinguir la naturaleza del enlace, haciéndolo parecer más rápido, más lento o menos costoso de lo que pueda ser en realidad. Este cambio puede ocasionar problemas con el comportamiento del enrutamiento, además de que realiza más ciclos de CPU que si se enrutara con un protocolo nativo.

Implementación

[25]GRE se puede utilizar para establecer algunas configuraciones inusuales de áreas en OSPF. Por ejemplo, si se tienen dos *campi* que forman parte de la misma área OSPF, se puede crear un túnel GRE para lograr la comunicación entre ellos:

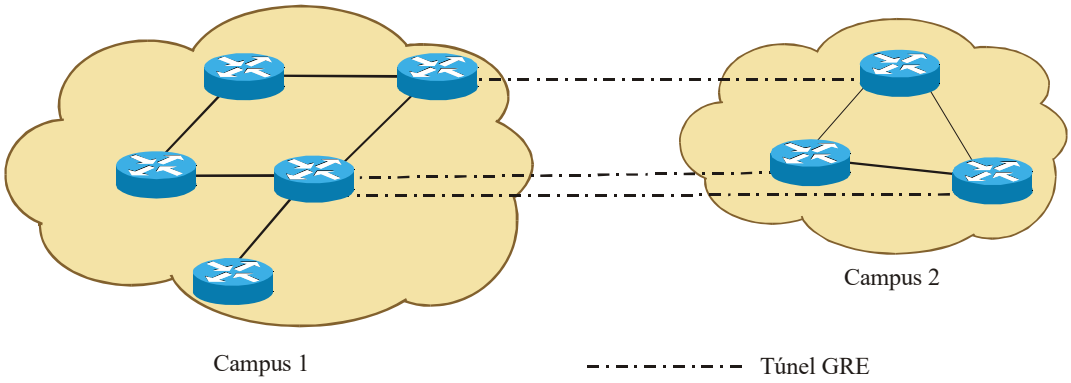


Figura II.2.2.4

En muchas circunstancias, se podrían conectar los dos *campi* a través de un *backbone* común, lo cual sería una técnica apropiada. Sin embargo, las redes pequeñas se pueden beneficiar con la simplicidad de una misma área (con el túnel GRE se sigue considerando una misma área OSPF). Esto puede ser provechoso en el caso de que algún enrutador no soporte funciones multi-áreas o que tenga una autoconfiguración limitada a operar en una misma área.

Otra aplicación de GRE es unir áreas discontiguas, pero esto sólo se puede hace si se corre un protocolo de enrutamiento que sea *classful* (o que respete las clases de IP).

II.2.3. L2F

Introducción

El servicio tradicional de la red *dial-up* en el Internet sólo funciona para las direcciones IP registradas, pero es deseable otra clase de aplicación virtual *dial-up* que permita protocolos múltiples y también direcciones IP no registradas en el Internet. Los ejemplos de esta clase de aplicación soportan direcciones IP privadas, IPX y AppleTalk vía SLIP/PPP por la infraestructura existente de Internet.

El apoyo de estas aplicaciones *dial-up* virtuales multiprotocolos tienen un beneficio significante ya que los usuarios finales y los proveedores de servicios de internet (ISPs) pueden compartir inversiones muy grandes hechas en el acceso y la infraestructura del núcleo de la red, además de permitir la realización de llamadas locales. También permite que las inversiones existentes en aplicaciones no hechas sobre IP sean soportadas en una manera segura[29].

Un ejemplo de los protocolos que permiten integrar múltiples servicios *dial-up* es L2F. L2F es un protocolo de capa 2 desarrollado por Cisco, independiente del medio físico que sentó las bases para establecer un nuevo protocolo de tuneleo llamado L2TP normalizado por la IETF (*Internet Engineering Task Force*) [30].

Topología

A continuación se muestra un ejemplo de topología en la cual se aplica tuneleo por L2F a través de un acceso PSTN (*Public Switched Telephone Network*) –es decir, a través de módems PPP-. Los usuarios remotos (ya sean PPP, SLIP o incluso ISDN) tienen acceso a una LAN sólo si marcan al enrutador *gateway*, aunque su conexión física sea a través de un NAS (*Network Access Server*) del ISP[29].

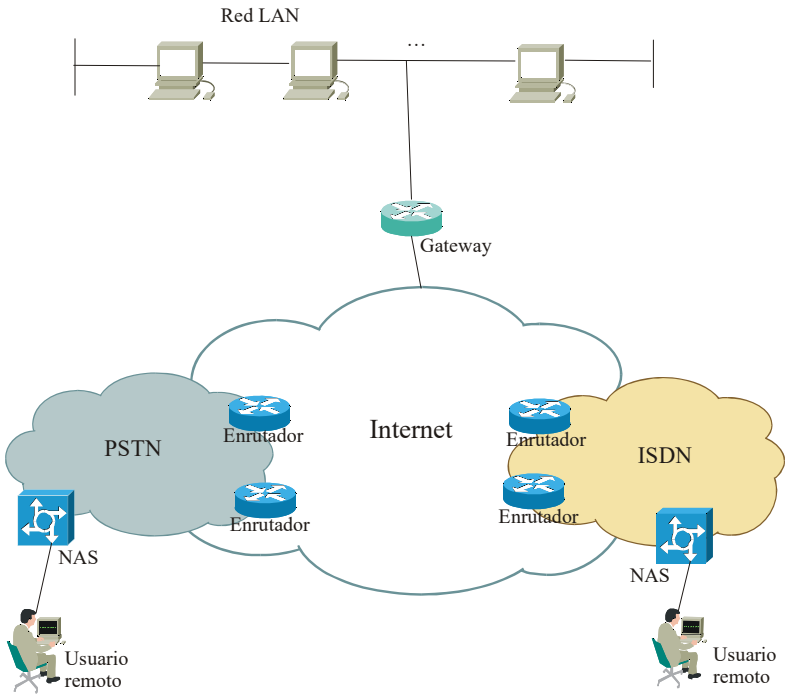


Figura II.2.3.1

Descripción y características:

El protocolo *Layer 2 Tunneling* provee a los clientes de redes conmutadas un amplio rango de beneficios, como por ejemplo, cuáles son los parámetros de la sesión negociados por el cliente en su propio enrutador de salida[33]. Los mecanismos tradicionales como autorización , negociación de la dirección, protocolo de acceso, cuenta y filtro de acceso son controlados en la propia red del cliente. Las tecnologías existentes de cifrado corren transparentemente punta a punta sobre los túneles de capa 2 asegurando privacidad y confiabilidad de la información[30]. L2F soporta varios túneles paralelos independientes y puede tunelear HDLC, PPP o SLIP sobre UDP[34]. Sin embargo, la verificación del usuario es más débil que en PPTP, además de que es necesario un código extra para ello[30]. L2F está definido en la RFC 2341 y utiliza el puerto 1701 de UDP[31].

Existen diferencias significativas entre los servicios de acceso estandarizados a Internet y los servicios virtuales *dial-up* con respecto a verificación, asignación de direcciones, autorización y contabilización. Los mecanismos usados por los servicios virtuales *dial-up* pueden coexistir con mecanismos más tradicionales,

con lo cual un PoP de un ISP puede servir simultáneamente tanto a sus clientes como a los clientes de *dial-up's* virtuales[29].

Seguridad:

Para el servicio virtual *dial-up*, el ISP realiza la verificación sólo en donde es requerida para descubrir la identidad aparente del usuario (y por tanto, el *gateway* deseado). Cuando es determinado, se inicia una conexión al *gateway* con la información de verificación reunida por el ISP. El *gateway* completa la verificación, ya sea aceptando o rechazando la conexión y debe proteger contra ataques al establecer túneles. El establecimiento del túnel involucra una fase de verificación entre el ISP y el *gateway* para proteger contra dichos ataques.

Asignación de direcciones:

Para un servicio de Internet, el usuario acepta que la dirección de Internet sea asignada dinámicamente por el ISP, lo que significa que el usuario remoto no tiene acceso a sus propios recursos de red, debido a los *firewalls* y otras políticas de seguridad aplicadas a la red local para tener acceso desde direcciones IP externas.

Para un servicio virtual *dial-up*, el *gateway* puede existir más allá del *firewall* local, asignando direcciones que son internas (pueden ser direcciones IP o no). Debido a que L2F tunelea en capa 2, las políticas de administración de direcciones son irrelevantes para corregir el servicio virtual *dial-up*. Para propósitos de manejo de PPP, el usuario *dial-up* aparece como si tuviera una conexión al *gateway*.

Verificación:

La verificación del usuario ocurre en tres fases: la primera en el ISP, la segunda en el *gateway* y la tercera (opcional) también en el *gateway*. El ISP usa el nombre de usuario para determinar si se requiere un *dial-up* virtual e inicia la conexión del túnel hacia el *gateway* adecuado. Ya establecido el túnel, se asigna un

identificador de multiplexación (*MID, Multiplex ID*) que no esté siendo usado y se inicia una sesión para enviar la información de verificación reunida.

El *gateway* lleva a cabo la segunda fase al decidir si acepta o no la conexión. La indicación de conexión puede incluir información de verificación, ya sea textual, por PAP o por CHAP. De acuerdo a esta indicación. El *gateway* acepta o rechaza la conexión. El rechazo puede deberse a que se encontró un nombre o password incorrecto. Si se aceptó la conexión, el *gateway* es libre para decidir si continúa con la tercera fase o no. La tercera fase se lleva a cabo en PPP.

Contabilización:

Un requisito que tanto el *gateway* de acceso como el *gateway* del ISP puedan proporcionar es la contabilización de datos, ya sea por paquetes, bytes o números de inicio/fin de conexiones. Como el servicio virtual *dial-up* es un servicio de acceso, el conteo de intentos de conexión (en particular, intentos fallidos de conexión) es de especial interés. El *gateway* del ISP puede rechazar nuevas conexiones de acuerdo a la información de verificación reunida por el ISP. Existe un caso en el que el *gateway* acepta la conexión y entonces continúa con la verificación, pero subsecuentemente, el *gateway* puede desconectar al cliente. Una desconexión debe incluir la indicación para el ISP con la razón correspondiente. Como el *gateway* puede declinar una conexión según la información recolectada por el ISP, la contabilización puede proporcionar una distinción entre los intentos de conexión fallidos y los exitosos. Sin esta facilidad, el *gateway* debe aceptar siempre las peticiones de conexión y por ende, necesitará intercambiar paquetes PPP con el sistema remoto.

Los paquetes PPP pueden ser encapsulados en L2F. El paquete encapsulado es el paquete como si fuera transmitido sobre un enlace físico, incluye[29]:

- Banderas
- Datos de transparencia (*bit stuffing* o bits de relleno para medios que requieren sincronización)
- CRC
- Valor del protocolo

Funcionamiento

La forma en la que se establecen una conexión virtual *dial-up* es la siguiente[29]:

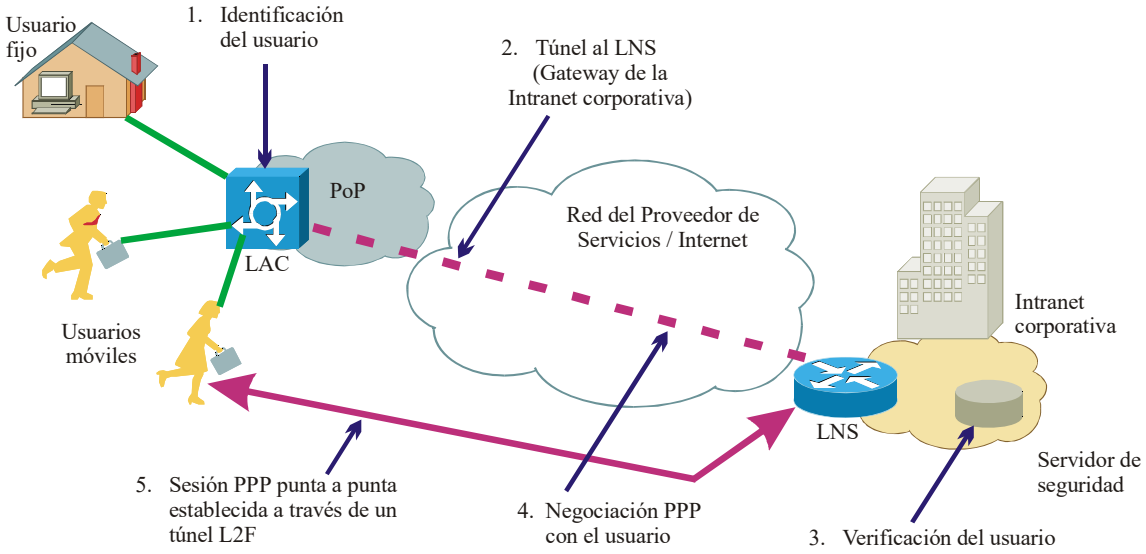


Figura II.2.3.2

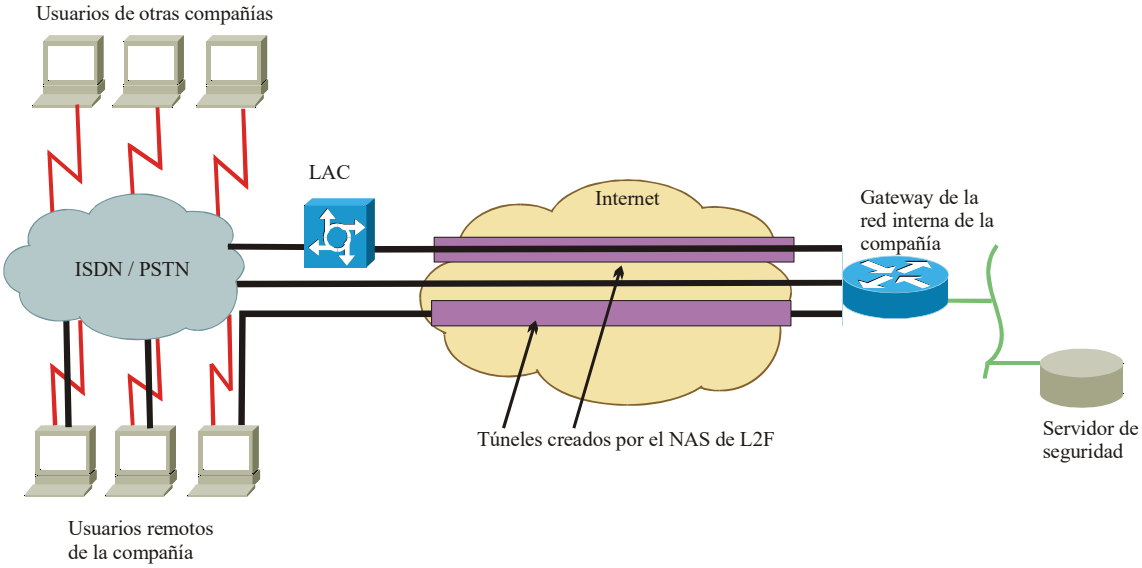


Figura II.2.3.3

[29]El usuario remoto inicia una conexión PPP a un ISP, ya sea por medio de una red PSTN o una ISDN. El NAS (*Network Access Server*) acepta la conexión y el enlace PPP es establecido. El ISP realiza una autenticación parcial del usuario y del sistema final con CHAP o con PAP[34]. Únicamente el campo del nombre del usuario es interpretado para determinar si el usuario requiere una conexión virtual *dial-up*. Es deseable, pero no necesario, que el nombre de usuario sea estructurado (por ejemplo, black@yahoo.com). Alternativamente, el ISP mantendrá una base de datos que mapee usuarios y servicios. En el caso de una conexión virtual *dial-up*, el mapeo será entre un punto final específico y el enrutador *gateway*. En caso de que no sea requerida una conexión virtual *dial-up*, se proporciona el acceso al Internet.

Si hasta el momento no existiera un túnel hacia el enrutador *gateway* deseado, se inicia uno. L2F está diseñado para ser altamente aislado de los detalles del medio sobre el cual es establecido el túnel, sólo requiere que el medio proporcione una conectividad orientada punto a punto. Ejemplos de estos medios son UDP, los PVCs de Frame Relay o los circuitos virtuales de X.25. Una vez que el túnel existe, se asigna un nuevo MID y se envía una notificación de la conexión para avisar al *gateway* de la nueva sesión *dial-up*. El *gateway* puede aceptar o no la conexión. En caso de rechazo, se despliega una notificación al usuario explicando la razón por la cual se declinó la conexión. Inmediatamente después de esa indicación, se termina la llamada.

La notificación inicial incluye la información de verificación requerida para permitir que el *gateway* verifique al usuario y decida aceptar o rechazar la conexión. En el caso de CHAP, también se incluye el nombre de usuario y la respuesta. Para PAP, se incluye el nombre de usuario y el *password*. El *gateway* puede escoger si utiliza toda la información para completar su verificación o si realiza un ciclo adicional de verificación. Para PPP, la notificación inicial incluye una copia del LCP CONFACK (*Configuration Acknowledgement*) enviados en cada dirección, con lo cual se completa la negociación LCP. El *gateway* puede elegir si usa esta información para inicializar su estado o si se inicia un nuevo intercambio de mensajes LCPs.

En caso de que el *gateway* acepte la conexión, se crea una interfaz virtual para PPP en forma análoga a la usada en una conexión *dial-up* directa. Con esta interfaz virtual, los *frames* de capa de enlace pueden pasar

por el túnel en ambas direcciones. Los *frames* del usuario remoto se reciben en el PoP, se les quita cualquier otro encabezado y se encapsulan en L2F, para finalmente ser enviados por el túnel correspondiente.

El *gateway* acepta estos *frames*, les quita el encapsulado L2F y los procesa como *frames* recibidos normalmente a su interfaz y su protocolo apropiados. La interfaz virtual se comporta como un interfaz física con la excepción de que en este caso, el hardware se localiza en el PoP del ISP. En dirección inversa, el *gateway* encapsula el paquete en L2F y el PoP lo desencapsula antes de transmirlo por la interfaz física hacia el usuario remoto. En este punto, la conectividad es PPP (*point-to-point*), cuyos puntos finales están en las aplicaciones del usuario remoto por un lado y, por el otro lado, en el *gateway*. Debido a que el usuario remoto llega a ser un cliente *dial-up* más del servidor de acceso del *gateway*, la conectividad del cliente puede ser administrada usando mecanismos tradicionales de autorización, de acceso y de filtrado. La contabilización del número de accesos puede ser ejecutado desde el NAS o desde el *gateway*.

Debido a que L2F conecta las notificaciones que PPP envía a los clientes y éstas contienen información suficiente para que el *gateway* realice la verificación e inicialice su máquina de estado LCP, no es necesario que el usuario remoto sea verificado por segunda vez con CHAP, ni que el cliente experimente múltiples ciclos de negociación LCP. Estas técnicas pueden aplicarse para optimizar la conexión, pero no son necesarias.

Formato del *frame*:

El paquete entero encapsulado tiene la forma[32]:

Figura II.2.3.4

El header o encabezado de L2F es[32]:

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
F	K	P	S	0	0	0	0	0	0	0	0	C	Versión			Protocolo								Número de secuencia							
Multiplex ID																Client ID															
Tamaño																Offset															
Llave																															

Figura II.2.3.5

Los campos del encabezado son[29] y [32]:

- F: 1bit. Si es uno, existe *offset* (especificado en el campo del mismo nombre).
- K: 1 bit. Si es uno, se presenta una llave (especificada en el campo del mismo nombre).
- P: 1 bit. Si es uno, indica que este paquete tiene prioridad.
- S, Secuencia válida: 1 bit. Si es uno, el número de secuencia es usado. Debe ser uno para todos los paquetes de administración L2F.
- C, *Checksum*: 1bit. Si es uno, existe *checksum*, es decir, existen dos bytes adicionales de checksum después del mensaje PPP/HDLC. El *checksum* se aplica desde el primer byte de las banderas L2F hasta el último byte del paquete.
- Versión: 3 bits. La versión del software de L2F que creó el paquete, el valor de este campo debe ser 001, de lo contrario el paquete es invalidado
- Protocolo: 8 bits. Específica el protocolo encapsulado en el paquete L2F:

Tabla II.2.3.1

Valor	*Descripción*
0	Ilegal
1	Paquete de administración L2F
2	Tuneleo PPP
3	Tuneleo SLIP

Si es un paquete ilegal, se elimina.

- Número de secuencia (*Sequence Number*): 8 bits, opcional. Se presenta si el bit *S* es 1; en ese caso, todos los paquetes que se reciban en el futuro con ese MID, debe tener el bit *S* igual a uno y por lo tanto usar el campo de Número de Secuencia. El número de secuencia comienza con cero para el primer paquete L2F

enviado con ese MID y los paquetes subsecuentes incrementan este campo en uno hasta 256. El contador es para cada MID.

- *Multiplex ID:* 16 bits. Identifica una conexión particular en un túnel. Cada conexión tiene asignada un MID que no sea usado en ese momento. El valor reservado para MID es cero, ya que este valor es usado para comunicar el estado del túnel a él mismo y a estos paquetes se les llama L2F_PROTO y son enviados con un MID igual a cero[3].
- *Client ID:* 16 bits. Es usado para la demultiplexación en los puntos finales de los túneles en caso de que las conexiones punto a punto no tengan una técnica para hacerlo directamente. Los paquetes con un cliente desconocido deben ser descartados.
- Tamaño: 16 bits. Tamaño en bytes del paquete completo, incluyendo encabezado, y demás campos adicionales. No incluye al *checksum*. Si se recibe un paquete más pequeño que lo indicado en este campo, se descarta.
- *Offset:* 16 bits, opcional. Especifica el número de bytes desde el último del header hasta el comienzo del mensaje. Cuando no existe *offset*, se supone que el mensaje inicia inmediatamente después del último byte del encabezado.
- Llave: 32 bits. El paquete está basado en la última respuesta de verificación dada. Su valor se determina tomando los 128 bits de la respuesta de verificación, se interpretan como 4 palabras de 32 bits cada una y se les aplica la función XOR a estas palabras juntas, resultando un valor de 32 bits.

Mensajes L2F de administración

Cuando se especifica en el campo Protocolo de un mensaje L2F que se trata de un mensaje de administración, el cuerpo del mensaje puede tener opciones o no. Una opción es un byte llamado Tipo de Mensaje *(Message Type)*, que puede estar o no seguido de "sub-opciones".

Cada sub-opción es un byte con determinado valor y según sea su valor, pueden seguir más bytes o no[29].

Tabla II.2.3.2

Valor Hexadecimal	*Abreviación*	*Significado*
0x00	Invalid	Invalid message
0x01	L2F_CONF	Request configuration
0x02	L2F_CONF_NAME	Name of peer sending L2F_CONF
0x03	L2F_CONF_CHAL	Random number peer challenges with
0x04	L2F_CONF_CLID	Assigned_CLID for peer to use
0x02	L2F_OPEN	Accept configuration
0x01	L2F_OPEN_NAME	Name received from client
0x02	L2F_OPEN_CHAL	Challenge client received
0x03	L2F_OPEN_RESP	Challenge response from client
0x04	L2F_ACK_LCP1	LCP CONFACK accepted from client
0x05	L2F_ACK_LCP2	LCP CONFACK sent to client
0x06	L2F_OPEN_TYPE	Type of authentication used
0x07	L2F_OPEN_ID	ID associated with authentication
0x08	L2F_REQ_LCP0	First LCP CONFREQ from client
0x03	L2F_CLOSE	Request disconnect
0x01	L2F_CLOSE_WHY	Reason code for close
0x02	L2F_CLOSE_STR	ASCII string description
0x04	L2F_ECHO	Verify presence of peer
0x05	L2F_ECHO_RESP	Respond to L2F_ECHO

Entrega de mensajes L2F:

L2F está diseñado para operar sobre enlaces punto a punto no orientados a conexión, por lo tanto, no está diseñado para proporcionar control de flujo del tráfico de datos ni para entregarlos confiablemente. Por esta razón, es deseable que cada protocolo tuneleado en L2F sea capaz de controlar el flujo de datos y retransmitirlos en caso necesario, pues únicamente los mensajes de control de L2F pueden ser retransmitidos.

Todos los mensajes de control L2F (es decir, los paquetes L2F con un valor 01 hexadecimal en el campo Tipo de Protocolo) son transmitidos con un número de secuencia. El Número de Secuencia es un contador que corre libremente por cada túnel L2F y que es incrementado (hasta 256) cada vez que se envía un paquete. Se utiliza para detectar paquetes duplicados o fuera de orden en el receptor.

Los mensajes de control L2F se intercambian hasta que el túnel L2F está completamente establecido y es hasta después de ser intercambiados dichos mensajes, que se pueden ofrecer los servicios de transporte de datos al cliente.

Tabla del estado del túnel:

A continuación se presenta en forma de tabla los mensajes L2F concernientes a la creación de un túnel. Un NAS inicia el estado Start0, enviando antes un paquete para esperar un primer evento. Un enrutador *gateway* inicia en el estado Start1, esperando inmediatamente un paquete inicial para comenzar el servicio. Si un evento no coincide con un evento en específico, el paquete asociado a ese evento es descartado.

Tabla II.2.3.3

Establecimiento del túnel (MID = 0) desde el lado del NAS			
Estado	***Evento***	***Acción***	***Nuevo estado***
Start0		Send CONF	Start1
Start1	CONF	Send OPEN	Start2
Start1	timeout 1-3	Send CONF	Start1
Start1	timeout 4	Clean up tunnel	(done)
Start2	OPEN	(initiate 1st client)	Open1
Start2	timeout 1-3	Send OPEN	Start2
Start2	timeout 4	Clean up tunnel	(done)
Open1	OPEN	Send OPEN	Open1
Open1	CLOSE	Send CLOSE	Close1

Open1	no MIDs open	Send CLOSE	Close2
Close1	CLOSE	Send CLOSE	Close1
Close1	timeout 4	Clean up tunnel	(done)
Close2	CLOSE	Clean up tunnel	(done)
Close2	timeout 1-3	Send CLOSE	Close2
Close2	timeout 4	Clean up tunnel	(done)

Tabla II.2.3.4

Establecimiento del túnel (MID = 0) desde el lado del gateway			
Estado	***Evento***	***Acción***	***Nuevo estado***
Start0	CONF	Send CONF	Start1
Start1	CONF	Send CONF	Start1
Start1	OPEN	Send OPEN	Open1
Start1	timeout 4	Clean up tunnel	(done)
Open1	OPEN	Send OPEN	Open1
Open1	OPEN (MID>0)	First client, below	Open2
Open1	CLOSE	Send CLOSE	Close1
Open1	timeout 4	Clean up tunnel	(done)
Open2	OPEN(MID>0)	Below	Open2
Open2	CLOSE	Send CLOSE	Close1
Close1	CLOSE	Send CLOSE	Close1
Close1	timeout 4	Clean up tunnel	(done)

Tabla del estado del cliente:

Esta tabla es similar a la anterior, pero en este caso se enumeran los estados de la conexión de un cliente en un túnel en estado abierto. Como esta secuencia se refiere a clientes, el campo MID no puede ser cero.

Tabla II.2.3.5

Establecimiento del túnel (MID ≠0) desde el lado del NAS			
Estado	***Evento***	***Acción***	***Nuevo estado***
Start0		Send OPEN	Start1
Start1	OPEN	Enable forwarding	Open1
Start1	CLOSE	Clean up MID	MID done
Start1	Timeout1-3	Send OPEN	Start1
Start1	timeout 4	Clean up MID	(MID done)
Start1	Client done	Send CLOSE	Close2
Open1	OPEN	No change	Open1
Open1	CLOSE	Send CLOSE	Close1
Open1	Client done	Send CLOSE	Close2
Close1	CLOSE	Send CLOSE	Close1
Close1	timeout 4	Clean up MID	(MID done)
Close2	CLOSE	Clean up MID	(MID done)
Close2	timeout 1-3	Send CLOSE	Close2
Close2	timeout 4	Clean up MID	(MID done)

ventajas y desventajas

Al proporcionar servicios virtuales de *dial-up*, L2F ofrece los siguientes atributos[29]:

- Encabezado pequeño: el protocolo debe imponer un encabezado adicional mínimo. Esto implica una encapsulación compacta y una estructura que omita algunas porciones de la encapsulación donde no sea requerida su función.
- Eficiencia: el protocolo debe ser encapsulado y desencapsulado fácil y eficientemente.

- Independencia del protocolo: deben hacerse pocas suposiciones sobre el medio en el que son transportados los paquetes.
- Desarrollo sencillo: no debe confiar en soportes adicionales de telecomunicaciones para operar (como número únicos de llamadas, o identificador de llamadas)
- Soporta múltiples protocolos
- Como está normalizado, es soportado por muchos vendedores

Las desventajas de L2F son:

- no ofrece control de flujo
- sus pares valor-atributo (AVP) no tienen seguridad, lo cual implica que los identificadores de túnel y de sesión pueden ser conocidos y tal vez alterados
- los datos no están cifrados
- la verificación que realiza no es muy robusta

II.2.4. PPTP

Introducción

Point-to-Point Tunneling Protocol o PPTP es un protocolo diseñado por Microsoft destinado a la creación de Redes Privadas Virtuales (VPN, del inglés Virtual Private Networks). Debido a que las VPN pueden incluir o soportar otros protocolos de red como IPX o NetBEUI dentro del protocolo TCP/IP, existen protocolos de red que pueden utilizarse para crear VPNs y, aunque PPTP no es el único, es un protocolo fácil de adquirir y de utilizar, lo que le hizo ganar muchos adeptos.

El protocolo PPTP permite utilizar enlaces económicos de Internet para crear conexiones seguras entre computadoras. Además, también se utiliza para crear redes privadas virtuales entre distintos sistemas operativos. Sin embargo, para establecer una conexión permanente PPTP hay que utilizar el servicio *Routing and Remote Access Service* (RRAS) de Windows NT[35].

Aunque este protocolo fue muy popular durante un tiempo, actualmente está siendo sustituido por L2TP. Sin embargo, la implementación de Microsoft sufre de varios importantísimos errores de diseño que hacen que su protección cifrada sea inefectiva para alguien más motivado que un simple observador casual, por lo que ya no se recomienda su uso[36].

Topología

La implementación PPTP sólo necesita de dos dispositivos, el PAC y el PNS. Estos dispositivos permiten que las funciones de un NAS (*Network Access Server*) sean separadas usando una arquitectura cliente-servidor. PPTP divide sus funciones de la siguiente manera:

- El PAC (*PPTP Access Concentrator*) es responsable de la interfaz física nativa de PSTN o de ISDN y controla los módems externos o los adaptadores de terminal (TAs), según sea el caso. Un NAS tendría una interfaz directa a un circuito analógico o digital de telecomunicación por medio de un módem externo o de un TA. Por ello, sus funciones son adaptar la tasa de transmisión, realizar la conversión analógica-digital y la conversión síncrona-asíncrona y todas las alteraciones necesarias a la cadena de datos. También es responsable por la terminación lógica de una sesión LCP PPP.

- El PNS *(PPTP Network Server*) se encarga de la agregación de canales y de la administración del paquete de un protocolo PPP en diferentes enlaces. Análogamente al PAC, es la terminación lógica de varios protocolos NCP PPP y hace el enrutamiento de múltiples protocolos entre interfaces NAS.

Los dos comparten la responsabilidad de verificar con PPP los protocolos de la VPN[37].

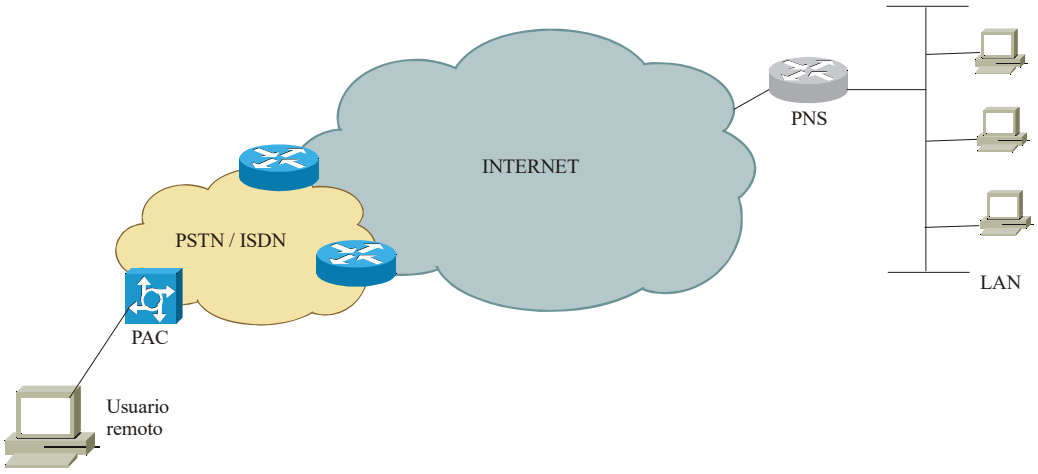


Figura II.2.4.1

Descripción y características

PPTP fue diseñado para proporcionar comunicaciones verificadas y cifradas entre un cliente y un enrutador *gateway* o entre dos *gateways* a través de un nombre de usuario y una contraseña. El objetivo primordial de PPTP era soportar múltiples protocolos con la mayor simplicidad posible, a la vez de tener la capacidad de abarcar todo el rango de direcciones IP. PPTP usa una conexión TCP para el mantenimiento del túnel y GRE para encapsular los *frames* PPP que sean tuneleados. La información de los *frames* PPP encapsulados pueden ser cifrados o comprimidos. El uso de PPP ofrece la posibilidad de negociar la verificación, cifrado y asignación de direcciones IP de los servicios de la VPN[38].

PPTP tiene verificación mutua cliente/servidor basada en contraseñas de usuarios y llaves de cifrado resultantes del proceso de verificación. PPTP es barato y sencillo de instalar para el administrador. Además, PPTP puede pasar a través de NAT (*Network Address Translation*), lo cual elimina la necesidad de que cada conexión PPTP tenga una dirección IP registrada a lo largo de Internet.

PPTP usa una versión extendida de GRE para transportar paquetes PPP. Las mejoras hechas a GRE permite tener bajo el nivel de congestión y control de flujo en los túneles establecidos entre el PAC y el PNS. Este mecanismo permite hacer uso eficiente del ancho de banda disponible para los túneles y evita las retransmisiones innecesarias. PPTP no determina los algoritmos específicos a ser usados para ellos, pero sí define los parámetros que deben ser comunicados con el fin de permitir que dichos algoritmos puedan trabajar. Sin embargo, comúnmente se utiliza el algoritmo de Ventana Deslizante para controlar el flujo de datos[39].

PPTP soporta una variedad de configuraciones de red en áreas anchas: líneas analógicas de teléfono o ISDN a través de la red de conmutación pública de teléfono (PSTN por sus siglas en inglés, Public Switched Telephone Network), Frame Relay y X.25. Para proporcionar esta tecnología, el ISP necesitará agregar o actualizar el software en sus servidores existentes de accesos remotos. Un ISP que realiza esta actualización en su punto de presencia proporcionaría un beneficio importante a sus clientes, pues podrían aprovechar las ventajas de una VPN sobre PPTP a sus clientes remotos.

Seguridad[40]:

PPTP hace uso de la seguridad proporcionada a través de PPP. MS-CHAP (verificación en PPP) se usa para validar la identidad del usuario contra los dominios de Windows NT y la llave de la sesión resultante es usada para cifrar los datos del usuario. La aplicación de Microsoft CCP (Protocolo de Mando de Condensación) tiene un bit que se usa para negociar el cifrado. Los clientes de RAS (*Remote Access Service*) pueden pedir conectarse sólo con cifrado habilitado. El servidor de RAS puede configurarse para permitir únicamente sesiones cifradas.

El cifrado de los datos se realiza usando los protocolos de RAS y mdash, llamados RSA RC4. Si el cifrado se negocia, RSA RC4 se usa con una llave por sesión de 40 bits, resultante de la primera verificación del usuario. La verificación adicional puede ser realizada por el Proveedor de Servicios de Internet en su punto de presencia (POP), si así se desea. PPTP complementa a los *firewalls* y dirige una necesidad diferente de seguridad. Los *firewalls* protegen a la red corporativa controlando estrictamente los datos que entran desde el Internet. PPTP trata de proporcionar seguridad a los datos intercambiados entre los usuarios remotos y la red corporativa.

Funcionamiento

En lugar de marcar un número de larga distancia para tener acceso remoto a una red corporativa, un usuario de PPTP marca un número local usando un módem V.34 o un módem ISDN para conectarse a un punto de presencia de un Proveedor de Servicios de Internet (ISP por sus siglas en inglés, Internet Service Provider). Una sesión PPTP proporciona una conexión segura a través del Internet hacia la red corporativa. La llamada local se conecta por medio de un dispositivo de hardware llamado Front-End Processor (*FEP*) el cual está situado en la misma ciudad del usuario. El FEP a su vez se conecta al servidor local (generalmente de Windows NT) localizado en una ciudad diferente a través de una red WAN, tal como Frame Realy o X.25. El FEP toma los paquetes PPP del usuario final y los conduce por un túnel (los "túnelea") que atraviesa la WAN. Debido a que PPTP soporta múltiples protocolos (IP, Ipx, y NetBUI), puede ser usado para tener acceso a una amplia variedad de infraestructuras LAN[38].

PPTP puede desplegarse en dos maneras:

- En una solución, la máquina del cliente y la del servidor usan los controladores de PPTP. Todo el cifrado se hace en el cliente, y el proceso inverso al cifrado se hace en el servidor. En este caso, no se necesita ningún cambio por parte del ISP para llevar a cabo esta solución.

- Como una alternativa, el ISP instala plataformas de marcación PPTP o procesadores Front-End. En esta solución, cualquier cliente de PPP que inicie una llamada, no sólo aquellos que entienden PPTP, puede establecer una conexión cifrada de PPTP al servidor de PPTP de la corporación[40].

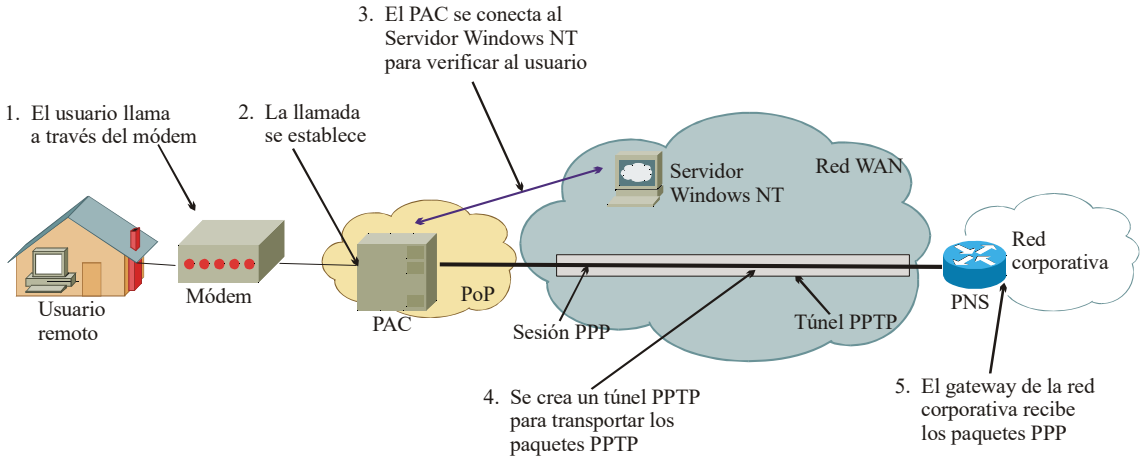


Figura II.2.4.2

Existen dos componentes paralelos en PPTP:

- una Conexión de Control entre cada par PAC-PNS operando sobre TCP y
- un túnel IP operando entre el mismo par PAC-PNS, el cual es usado para transportar paquetes PPP encapsulados en GRE para las sesiones de usuario entre el par.

Antes de que el túnel se cree entre un PAC y un PNS, se debe establecer una Conexión de Control entre ellos. La Conexión de Control es una sesión TCP estándar sobre la cual la llamada PPTP controla y administra la información que pasa. La sesión de control está lógicamente asociada (aunque de forma separada) con las sesiones del túnel PPTP. Para cada par PAC-PNS, deben existir tanto el túnel como una conexión de control. La conexión de control es responsable del establecimiento, administración y liberación de las sesiones transportadas a través del túnel. Esto significa que un PNS es notificado de una llamada llegante a un PAC asociado, así como un PAC es instruido para que tenga lugar una llamada saliente. Una conexión de control puede ser establecida tanto por el PAC como por el PNS. Enseguida del establecimiento de la conexión solicitada, el PNS y el PAC establecen su conexión de control por medio de mensajes *Start-Control-*

Connection-Request y *Reply*. Estos mensajes también son usados para intercambiar información sobre las capacidades básicas de operación del PAC y del PNS. Una vez que la conexión de control se ha establecido, se inician sesiones entre ambos dispositivos para atender las solicitudes. La conexión de control debe comunicar los cambios que ocurran acerca de las características de una sesión individual por medio de un mensaje *Set-Link-Info*. La conexión de control se mantiene a través de mensajes *keep-alive*, con lo que se logra detectar a tiempo una falla entre el PAC y el PNS.

Como se mencionó anteriormente, PPTP requiere el establecimiento de un túnel para cada par PAC-PNS. Dicho túnel se usa para transportar todos los paquetes de las sesiones PPP asociadas al par PAC-PNS. En el encabezado se distingue a qué sesión PPP pertenece un paquete por medio de un valor llamado llave (o "*key*") dentro del campo Llave. De esta manera, los paquetes PPP son multiplexados y demultiplexados sobre un túnel. El valor de la llave se establece durante la llamada de la conexión de control[37].

Operación de la conexión de control[37]

PPTP define un conjunto de mensajes enviados como datos TCP en la conexión de control entre un PNS y un PAC dado. La sesión TCP para la conexión de control se establece al iniciar la conexión TCP al puerto 1723. El puerto orígen asigna a cualquier otro puerto que no esté ocupado en ese momento. Cada mensaje PPTP de la conexión de control inicia con una porción fija de ocho bytes del encabezado. Dicha porción contiene el tamaño total del mensaje, el identificador del tipo de mensaje PPTP y una "*Magic Cookie*". Existen dos tipos de mensajes en la conexión de control: el Mensaje de Control y el Mensaje de Administración. El campo llamado *"Magic Cookie"* es una constante hexadecimal: 1A2B3C4D. El principal propósito de este campo es permitir que el receptor se sincronice apropiadamente con la cadena de datos TCP. Sin embargo, no debe ser usado para resincronizar una cadena de datos TCP en el caso de que se haya transmitido un mensaje mal formateado. La pérdida de sincronización implica el cierre inmediato de la conexión de control TCP.

La operación de la conexión de control es sencilla debido a que se usa TCP para proporcionar un mecanismo confiable de transporte. Sin embargo, la conexión TCP se puede cerrar en cualquier momento y para estos casos, se deben tener mecanismos de recuperación de errores. Además, los mensajes no se retransmiten ni se

ordenan. Algunos procedimientos de recuperación de errores son comunes en todos los estados de la conexión de control. Cabe señalar que si una respuesta no es recibida en 60 segundos, la conexión de control se cierra.

La conexión de control de PPTP no distingue entre un PNS y un PAC, pero sí es posible distinguir quién origina la conexión y quién la recibe. La terminal origen es aquella que intenta primero abrir TCP. Como es posible que cualquiera de los dos dispositivos (ya sea el PAC o el PNS) origine la conexión, también es factible que ocurra una colisión en TCP.

El esquema de los estados de la terminal origen de la conexión de control es[37]:

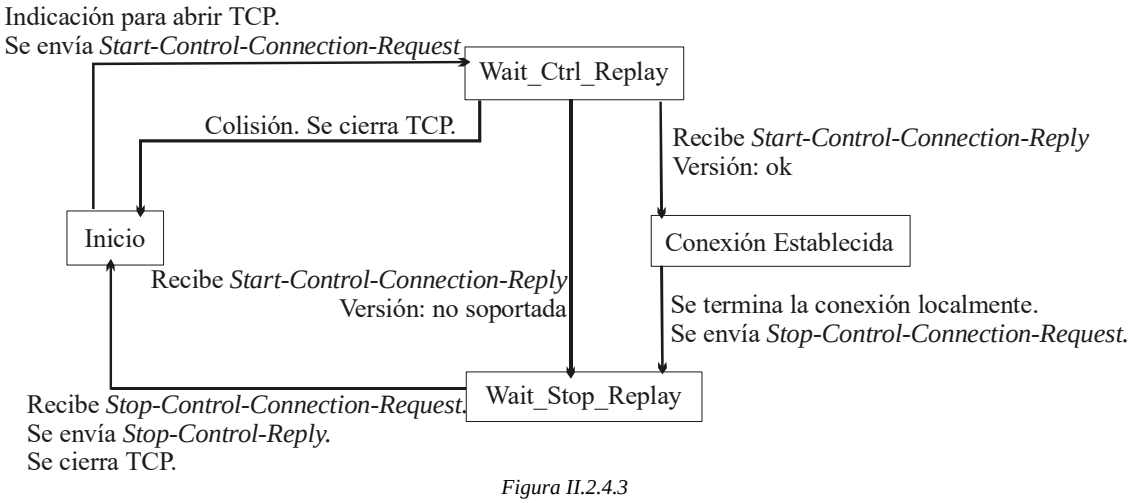


Figura II.2.4.3

Descripción de los estados:

Inicio

Durante este estado, la terminal que origina la conexión de control intenta abrir una conexión TCP hacia su punta vecina. Cuando la conexión TCP se abre, la terminal origen transmite un mensaje *Start-Control-Connection-Request* y pasa al estado Wait_Ctl_Reply.

Wait_Ctl_Reply

La terminal origen verifica si existe otra petición de conexión en su punta vecina con el fin de evitar colisiones. Cuando se recibe un mensaje *Start-Control-Connection-Request* se examina si la versión

del mensaje enviado es compatible con la respuesta. Si la versión de la respuesta es más antigua que la del mensaje enviado, ésa versión debe usarse. Si la versión de la respuesta es más reciente, se usa la versión del mensaje enviado (la versión más antigua), si es posible soportarla. En los dos casos anteriores, posteriormente se pasa al estado Conexión Establecida. En el caso de que no se pueda soportar la versión del mensaje, se envía un mensaje *Stop-Control-Connection-Request* y se pasa al estado *Wait-Stop-Reply*.

Conexión Establecida:

Una conexión establecida puede ser terminada por una condición local o por una solicitud *Stop-Control-Connection-Request* por parte del receptor. Si la conexión se termina debido a una condición local, la terminal origen envía un mensaje *Stop-Control-Connection-Request* y pasa al estado *Wait-Stop-Reply*. Si la conexión se termina por solicitud del receptor, se envía un mensaje *Stop-Control-Connection-Reply* y cierra la conexión TCP, asegurándose de que la última información TCP se transmitió correctamente.

Wait_Stop_Reply:

Si se recibe un mensaje *Stop-Control-Connection-Reply,* la conexión TCP se cierra y la conexión de control regresa al estado Inicio.

El esquema de los estados de la terminal receptora de la conexión de control es[37]:

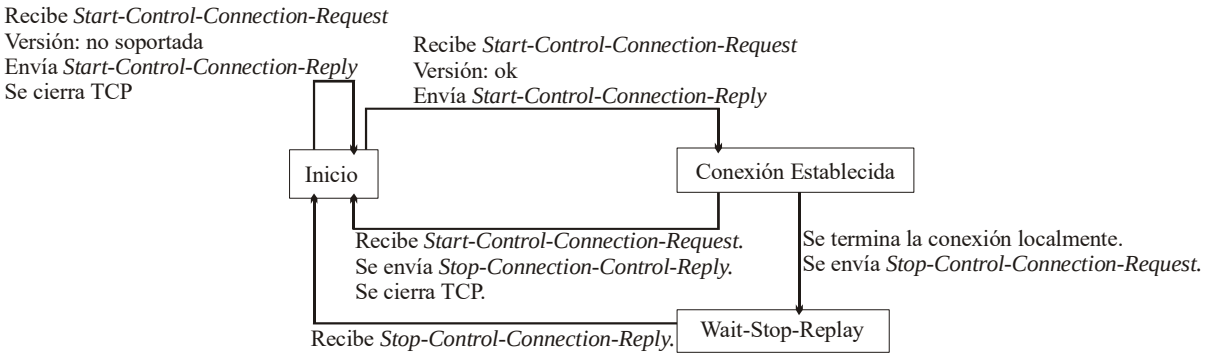


Figura II.2.4.4

Descripción de los estados:

Inicio:

El receptor de la conexión de control espera que se abra TCP en el puerto 1723. Cuando es notificado de que una conexión TCP se ha abierto, se preparan los mensajes PPTP que serán enviados. En el momento en que se recibe un mensaje *Start-Control-Connection-Request*, se debe examinar en el encabezado del paquete el campo correspondiente a la versión. Si la versión del mensaje recibido es más reciente que la versión soportada por el receptor, y además es posible soportar la versión reciente, el receptor envía un mensaje *Start-Control-Connection-Reply*. De la misma manera, se envía este mensaje si las versiones son iguales y en los dos casos anteriores, se pasa al estado Conexión Establecida. Si no fuera posible soportar la versión del mensaje enviado, el receptor envía un *Start-Connection-Reply*, cierra la conexión TCP y regresa al estado inicial.

Conexión Establecida

Una conexión establecida puede ser terminada por una condición local o por una solicitud *Stop-Control-Connection-Request*. Si la conexión se termina debido a una condición local, la terminal origen envía un mensaje *Stop-Control-Connection-Request* y pasa al estado *Wait-Stop-Reply*. Si la conexión se termina por solicitud del receptor, se envía un mensaje *Stop-Control-Connection-Reply* y cierra la conexión TCP, asegurándose de que la última información TCP se transmitió correctamente.

Wait-Stop-Reply:

Si se recibe un mensaje *Stop-Control-Connection-Reply*, la conexión TCP se cierra y la conexión de control regresa al estado Inicio.

Formato del *frame*

Los paquetes PPP que son transportados entre un PAC y un PNS son encapsulados en PPTP y después encapsulados en IP. En estos paquetes no se incluyen las banderas HDLC, los bits de inserción o caracteres de control, ni bits correspondientes a CRC. Los paquetes transmitidos sobre el túnel tienen generalmente una estructura como se ilustra a continuación[37]:

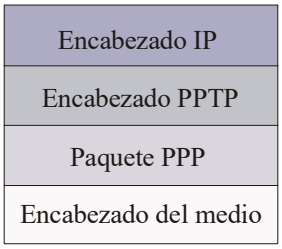


Figura II.2.4.5

El encabezado usado en PPTP está basado en las mejoras hechas a GRE. La principal diferencia entre un encabezado GRE y uno PPTP es que a éste último se le agregó un campo llamado Número de Reconocimiento (*Acknowledgement Number*) el cual tiene el propósito de determinar si un paquete de control o un conjunto de ellos se ha recibido en un extremo del túnel. Sin embargo, este campo no se usa para la retransmisión de paquetes de datos del usuario, pues en ese caso, su función es determinar la tasa a la cual los paquetes del usuario son transmitidos sobre el túnel para una sesión dada. El formato del encabezado se muestra en seguida[37]:

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td>C</td><td>R</td><td>K</td><td>S</td><td>s</td><td colspan="3">Recursión</td><td>A</td><td colspan="3">Banderas</td><td>C</td><td colspan="3">Versión</td><td colspan="16">Tipo de protocolo</td></tr>
<tr><td colspan="16">Llave de la longitud de los datos</td><td colspan="16">Llave del identificador de llamadas (Call ID)</td></tr>
<tr><td colspan="32">Número de secuencia</td></tr>
<tr><td colspan="32">Número de reconocimiento (Acknowledgement Number)</td></tr>
</table>

Figura II.2.4.6

- Bit C: si es igual a cero, indica que se ha verificado la trama.
- Bit R: si es igual a cero, indica que se ha enrutado el paquete.
- Bit K: si es igual a uno, los campos Llave están presentes.

- Bit S: si es igual a uno, indica que se trata de un paquete de datos y el campo Número de Secuencia está presente. Si es igual a cero, se trata de un paquete de reconocimiento.
- Bit s: si es igual a cero, indica que en el paquete se presenta la ruta exacta de enrutamiento desde el origen.
- Recursión: si es igual a cero, se tiene control de recursión.
- Bit A: indica que el campo de Número de Reconocimiento (*Acknowledgement Number*) está presente en el encabezado. Si es igual a uno, significa que se lleva un conteo de los reconocimientos transmitidos.
- Banderas: estos bits deben ser iguales a cero.
- Ver: estos bits deben ser iguales a uno.
- Tipo de protocolo: este campo es igual al número hexadecimal 880B.

El campo llave se divide en:

- Llave de la longitud de los datos: indica el tamaño del mensaje sin incluir el encabezado.
- Llave del identificador de la llamada (*Call ID*): estos dos bytes contienen el identificador de la llamada para la sesión a la cual pertenece el paquete.
- Número de secuencia: este campo está presente sólo si el bit S es igual a uno. Contiene el número de secuencia del paquete.

- Número de reconocimiento (*Acknowledgement Number*): está presente sólo si el bit A es iguala uno. Contiene el número de secuencia del paquete recibido numerado con el más alto valor by cada que se recibe otro, se incremente en uno..

Ventajas y desventajas

Ventajas:

Este protocolo beneficia principalmente a personas que trabajan en oficinas remotas, en sus casas, o que necesitan acceso a su LAN corporativa desde la calle. Las principales ventajas que ofrece PPTP se deben a la separación de las funciones de un NAS en dos dispositivos separados y éstas son[37]:

- Administración flexible del direccionamiento IP: los usuarios pueden mantener una sola dirección IP, incluso dentro de una red de conmutación entre diferentes PAC y llegar a un solo PNS. Si la red de una empresa utiliza direcciones no registradas, un PNS asociado a la empresa asigna las direcciones a la red privada.
- Soporte a protocolos diferentes a IP en redes conmutadas: lo cual permite que AppleTalk e IPX sean tunuelados a través de un solo ISP. No es necesario que el PAC sea capaz de procesar esos protocolos.
- Ofrece una solución a la división de múltiples enlaces PPP: para agregar canales B a una red ISDN, típicamente se usa *Multilink PPP*, lo cual requiere que todos los canales de un conjunto de múltiples enlaces sea agrupado en un solo NAS. Pero como en PPTP un PNS puede manejar un conjunto de múltiples enlaces, éste se comprime en esa terminal y se separan en el lado del PAC.

Desventajas:

Aunque PPTP es una implementación concreta cliente-servidor de la tecnología VPN, su sistema de seguridad no es suficiente, pues se han descubierto una serie de importantes vulnerabilidades en su implementación. A pesar de que se han corregido muchos de los problemas denunciados en la implementación original, algunos de ellos siguen presentes o incluso han debilitado considerablemente la calidad del sistema. Por esta razón, se considera que PPTP es un protocolo que puede evitar a un usuario curioso, pero no es rival ante un adversario determinado a acceder a la información que circule por el túnel. Este hecho es especialmente importante para las empresas que emplean PPTP para interconectar sus intranets entre sí, a través de infraestructuras públicas como es Internet. Esta razón es la principal por la que L2TP ha tenido buena aceptación, pues no presenta problemas en seguridad[36].

II.2.5. L2TP

Introducción

L2TP es un protocolo de capa 2 (según el modelo OSI), extensión del protocolo PPP. Es usado para Redes Privadas Virtuales (VPNs, Virtual Private Network) y reúne las mejores características de dos protocolos existentes de tuneleo[2]: PPTP (de Microsoft) y L2F (de Cisco)[41]. Usando el tuneleo de L2TP, un Proveedor de Servicios de Internet (ISP, Internet Service Provider) puede crear un túnel virtual para enlazar a los sitios remotos de un cliente o a usuarios remotos de una red corporativa[42].

Topología

[43]Los dispositivos principales propios de esta tecnología son:

- El LAC (*L2TP Access Concentrator*) es un dispositivo que conecta al cliente a una red pública o a otro dispositivo PPP. Está localizado en el punto de presencia (*POP, Point of Presence*) del ISP y se encarga de intercambiar mensajes PPP con los usuarios remotos. Soporta el tuneleo de cualquier protocolo en PPP, además de inicializar las llamadas salientes y recibir las llamadas entrantes. Sólo necesita implementar el medio de transmisión.

- El LNS (*L2TP Network Server*) es un dispositivo que maneja el servidor del protocolo L2TP, opera sobre cualquier plataforma que soporte una terminación PPP y su desempeño depende del medio sobre el que se creen los túneles, por lo tanto, tiene una interfaz WAN o LAN. Es el punto terminal de los túneles L2TP y punto de acceso en el que los *frames* PPP son procesados y dirigidos hacia protocolos de capas

[2] Un túnel es una interfaz lógica sobre la cual se establece una conexión con determinado ancho de banda reservado, por lo que el término *tuneleo* se refiere al tráfico de datos en dicha conexión.

superiores. Inicia las llamadas que llegan al LAC, es decir, llamadas entrantes para el LAC, y recibe las que inician en el LAC (o llamadas salientes para el LAC). Es capaz de terminar éstas últimas.

- El NAS (*Network Access Server*) es un dispositivo que proporciona a los usuarios un acceso temporal a la red según la demanda, por conexiones punto-a-punto, usando típicamente la red pública telefónica de conmutación (PSTN) o usando una red ISDN (*Integrated Services Digital Network*).

La arquitectura de acceso de L2TP a VPNs depende del ambiente en el que se establece la VPN:

- En un ambiente *dial* o de conmutación, el túnel L2TP puede ser iniciado desde un NAS o desde un *software* del cliente hacia un enrutador que actúe como punto de terminación del túnel.

- En un ambiente xDSL, los circuitos Virtuales Permanentes (PVC, *Permanent Virtual Circuit*) del usuario sobre ATM se extienden desde el CE (*Customer Edge*) hasta un NAS central , el cual origina los túneles L2TP hacia los LNS. Éste NAS puede ser operado para ofrecer servicios de ADSL.

Un bosquejo de la arquitectura es el siguiente:

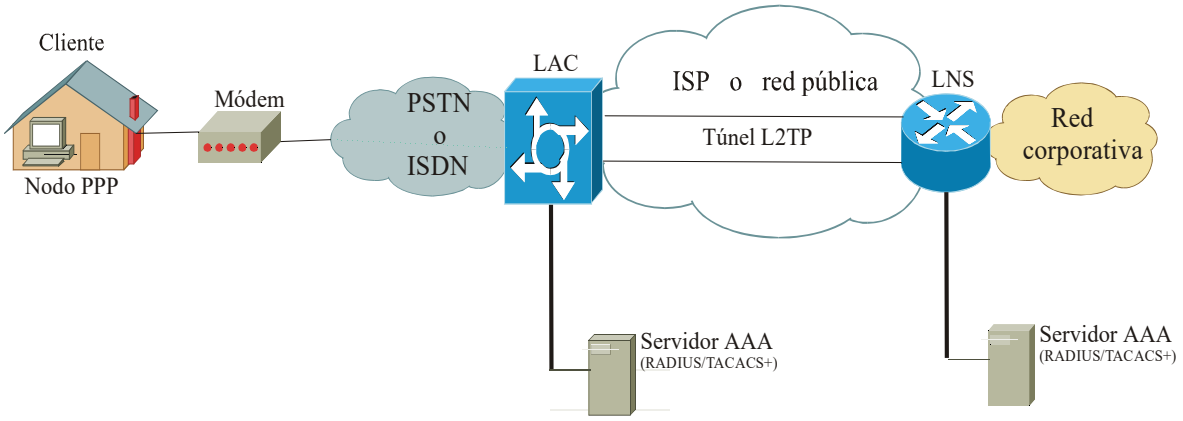


Figura II.2.5.1

Descripción y características

Los servicios tradicionales por marcación a redes telefónicas sólo soportan direcciones IP registradas, lo cual limita los tipos de aplicaciones que son implementados sobre VPNs. L2TP soporta múltiples protocolos y direcciones IP públicas o privadas sobre Internet, permitiendo que la infraestructura existente de acceso -como lo es Internet, los módems, los servidores de acceso y los adaptadores de terminal ISDN (*TAs, Terminal Adapters*)- sea empleada. Además, permite a los clientes de una empresa reducir el encabezado agregado a sus paquetes y explotar los recursos ofrecidos.

Las características principales de este protocolo son[41]:

- Soporta cualquier protocolo enrutable (IP, IPx, etc.)
- Es independiente del medio de transmisión: opera sobre cualquier red capaz de entregar paquetes de datos. (soporta cualquier tecnología LAN o WAN)

Las mejoras realizadas a L2F y PPTP que se añadieron a L2TP son: el control de flujo y el ocultamiento del Par "Valor-Atributo" (AVP*, Attribute-Value Pair*)[3] para alcanzar mayor seguridad, además de que se incluyeron también funciones que ya realizaban L2F y PPTP, como balanceo de carga en el enrutador *gateway* propio, configuración de un enrutador *gateway* primario y uno secundario como respaldo, soporte al nombre de dominio DNS, flexibilidad en el nombre de dominio y soporte de múltiples sesiones PPP, entre otras[42].

Para garantizar la seguridad se tiene la posibilidad de implementar cualquiera de las siguientes opciones[43]:

- AAA *(Authentication, Authorization, Accountig)*: La verificación se da por medio de protocolos como CHAP, PAP, MS-CHAP, etc. La autorización de acceso a los servicios de la VPN se otorga según el par *login/password* o según el número DNIS. La contabilización en el LAC se enfoca en el número de conexiones, es decir, en el número de inicios y terminaciones (*start/stop*) de una conexión, recabando información completa de los intentos fallidos de conexión.

[3] AVP es un par de valores genéricos enviados desde el servidor AAA hacia el cliente AAA. Por ejemplo, en un AVP "user = bill", *user* es el atributo y *bill*, el valor.

- IPsec: proporciona confiabilidad, integridad y verificación en cada nodo de la red. Para verificar cada nodo de la red con IPsec, se utiliza el algoritmo ISAKMP (*Internet Security Association Key Management Protocol*), conocido comúnmente como ISAKMP/Oakley o comúnmente IKE. Además, este algoritmo negocia políticas de seguridad y maneja el intercambio de llaves de una sesión.
- *Software* Cisco de *Firewall*: es una opción del *software* que se implementa con las características de seguridad. Este software realiza el control de tráfico, bloqueo de java (para controlar que los archivos que se bajen de la red no tengan virus), prevención y detección de virus, alertas de tiempo real y transacción de paquetes UDP para conocer las direcciones origen y destino de la conexión del usuario y el par de puertos correspondiente.
- Calidad de Servicio (QoS, *Quality of Service*): es posible soportar precedencia y prioridad IP, fragmentación y precedencia y prioridad para BGP4 con múltiples túneles para un LNS dado. Los proveedores de servicios pueden ofrecer a las empresas usuarias túneles diferenciados con niveles variables de ancho de banda.
- Administración de direcciones IP: L2TP soporta completamente la asignación dinámica de direcciones IP en cierto rango que esté bajo la administración de la empresa. También soporta direcciones IP privadas [41] y la asignación dinámica desde un servidor DHCP, además de la Traducción De La Dirección De Red (NAT, *Network Address Translation*) para evitar que las direcciones internas sean publicadas al exterior.
- Confiabilidad: la implementación de L2TP proporciona la capacidad de tener un respaldo, permitiendo que múltiples nodos LNS sean configurados como LNS de respaldo. Si la conexión al LNS primario es inalcanzable, el NAS establecerá una conexión con el LNS de respaldo.
- Escalabilidad: L2TP soporta un número de sesiones ilimitadas en cada LAC. Si se implementa L2TP con características de balanceo de carga (*load sharing*) y con un LNS "*stackable*" o apilable, varios LNS pueden ejecutar el balanceo de carga a través de múltiples conexiones tipo túnel entre un LAC y los LNS. La capacidad estadística del balanceo de carga a través de varios LNS otorga aún más confiabilidad y escalabilidad.

  La característica "*stackable*" de un LNS ayuda a dar un soporte adicional para múltiples sesiones PPP. Uno de los LNS tendrá la responsabilidad de ensamblar los paquetes fragmentados para cada sesión a través de varios túneles.

Funcionamiento

[43]Un proveedor de servicios de Internet u otros servicios de acceso pueden crear un túnel virtual para enlazar los sitios remotos del cliente o a usuarios remotos con las redes corporativas propias de la compañía. El LAC, localizado en el punto de presencia del ISP, intercambia mensajes PPP con los usuarios remotos y se comunica con el LNS por medio de solicitudes y respuestas L2TP para establecer el túnel. L2TP envía los paquetes a través del túnel establecido entre los puntos finales de una conexión punto a punto. Los paquetes de los usuarios remotos son aceptados en el punto de presencia del ISP, despojados de cualquier byte de relleno, encapsulados en L2TP y enviados sobre el túnel adecuado. El enrutador *gateway* de la red del cliente recibe los paquetes L2TP, los desencapsula de L2TP y los procesa para asignarles la interfaz apropiada. La estructura de un túnel L2TP, es parecida a la siguiente figura:

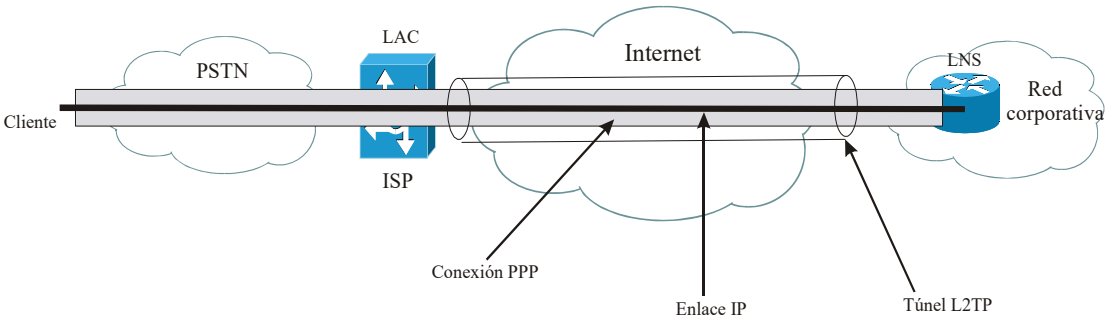


Figura II.2.5.2

Si la conexión de una VPN a un usuario remoto es por medio de L2TP, dicha conexión se establece entre un LAC (localizado en el punto de presencia del ISP) y el LNS de una empresa (localizado en la red corporativa). La secuencia[42] que se sigue para establecer la comunicación es:

1. El usuario remoto inicia una comunicación PPP al ISP, ya sea sobre PSTN o sobre ISDN
2. El LAC en la red del ISP acepta la comunicación al PoP y el enlace PPP se establece

3. El usuario final y el LNS negocian los parámetros del protocolo del control de enlace (LCP, *Link Control Protocol*)[4], el LAC verifica parcialmente al usuario final, ya sea con CHAP o con PAP. Para determinar si el usuario es un cliente autorizado para ingresar a la VPN, se utiliza el nombre de usuario, el nombre del dominio o el número DNIS (*Dial Number Identification Service*). En caso de que el usuario sea un cliente autorizado, el LNS lo mapeará a un punto final específico
4. Los puntos finales del túnel, el LAC y el LNS, se verifican entre sí antes de iniciar alguna sesión sobre la VPN. Si la comunicación es satisfactoria, se crea el túnel. El LNS puede aceptar la creación de un túnel sin haber realizado una comunicación previa al LAC
5. Una vez que el túnel existe, se crea una sesión L2TP para el usuario final
6. Entonces, el LAC propaga al LNS las opciones negociadas con LCP y la verificación parcial CHAP o PAP. El LNS concentrará las opciones negociadas y la verificación parcial en la interfaz de acceso. El LNS compara las opciones configuradas en dicha interfaz y las opciones enviadas por el LAC, si no coinciden, la comunicación se terminará y se desconectará del LAC

El resultado final es que el proceso de intercambio de datos parece darse entre el usuario final y el LNS, sin dispositivo intermediario (el LAC). Para ayudar a comprender este proceso, la siguiente figura muestra la secuencia que sigue una llamada saliente con su correspondiente numeración, según la secuencia:

[4] LCP, Link Control Protocol: protocolo componente de PPP que establece, configura y prueba los enlaces usados por PPP

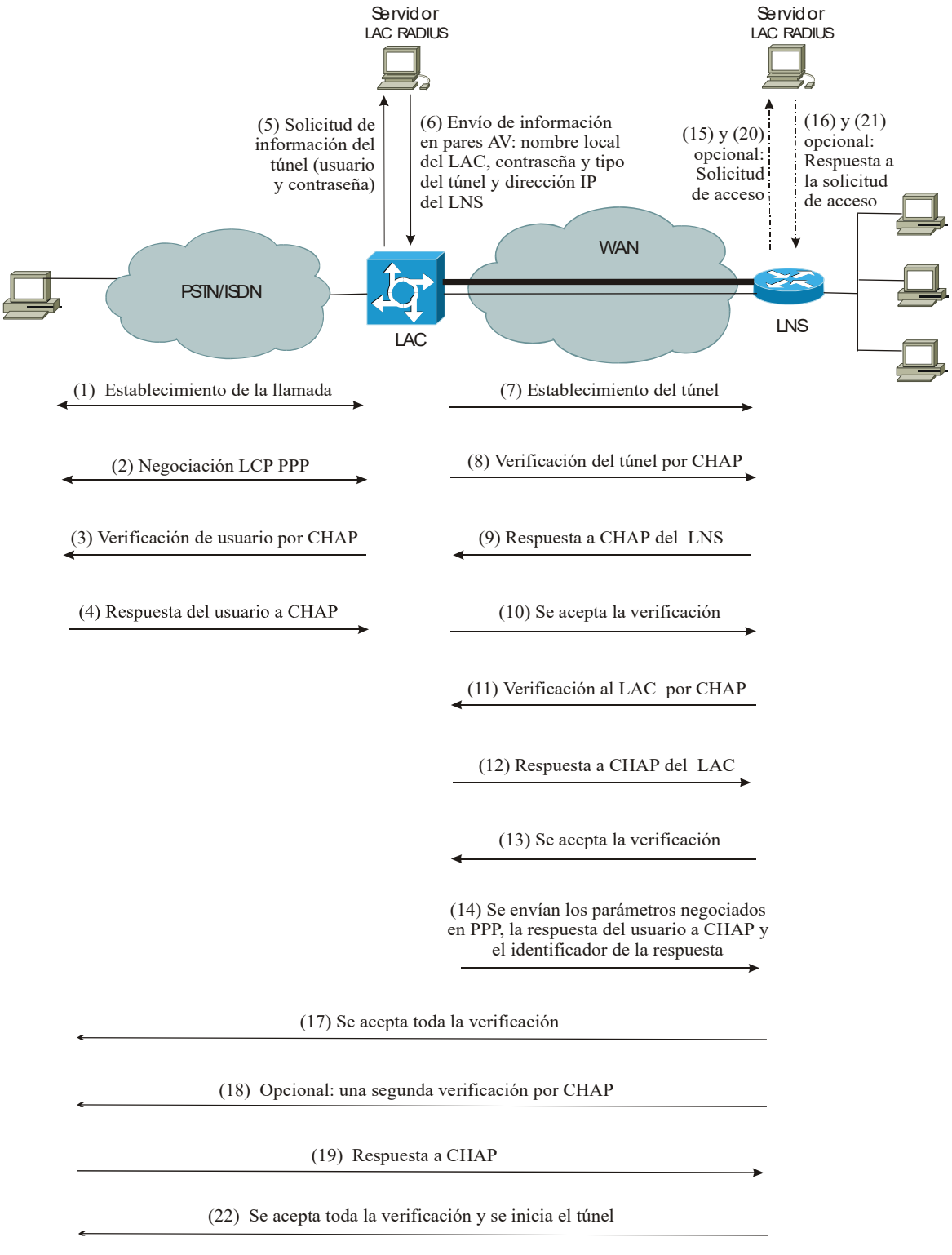


Figura II.2.5.3

Los mensajes de control AVP (*Attribute Value Pairs*) se utilizan para maximizar la extensibilidad sin afectar la interoperabilidad. Son intercambiados para crear un túnel o para establecer una sesión[44].

Formato del *Frame*

L2TP utiliza dos tipos de mensajes: los mensajes de control y los mensajes de datos. Los mensajes de control se usan en el establecimiento, mantenimiento y limpieza de túneles y llamadas y utilizan un canal de control en L2TP para garantizar la entrega. Los mensajes de datos son usados para encapsular *frames* PPP que serán transportados a través del túnel y en caso de que se pierdan, no se retransmiten[44].

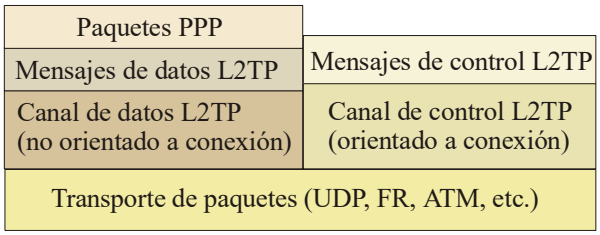


Figura II.2.5.4

La figura anterior es un bosquejo de la relación de los *frames* PPP y los mensajes de control sobre los canales de control y de datos de L2TP. Los *frames* PPP pasan por un canal de datos, encapsulados primero con un encabezado L2TP y después como paquetes en Frame Relay, UDP, ATM, etc. En cambio, los mensajes de control se envían en un canal confiable sobre el mismo transporte de paquetes (FR, ATM, UDP, etc). Generalmente se utilizan números de secuencia, los cuales se usan en los mensajes de control para garantizar la entrega de los paquetes y en los mensajes de datos para ordenar los paquetes y detectar aquellos que se perdieron[44].

Los paquetes L2TP, ya sean de control o de datos, comparten el mismo formato de encabezado. La estructura del encabezado es la siguiente:

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
T	L	X	X	S	X	O	P	X	X	X	X	Versión				Tamaño															
Identificador del túnel																Identificador de la sesión															
Ns																Nr															
Tamaño del Offset																Padding o relleno															

Figura II.2.5.5

Los campos del encabezado son[44]:

- T: 1 bit. Indica el tipo de mensaje. Si es cero, se trata de un mensaje de datos, si es uno, es un mensaje de control.
- L: 1 bit. Si es uno, el campo Tamaño está presente. Este bit debe ser uno para los mensaje de control.
- X: son bits reservados para futuras extensiones. Todos los bits reservados deben ser cero en los mensajes enviados e ignorados en los mensajes recibidos
- S: 1 bit. Si es uno, los campos Nr y Ns están presentes. Este bit debe ser uno para los mensajes de control.
- O: 1 bit. Si este bit es uno, el campo Tamaño del *Offset* está presente. Para los mensajes de control, este bit debe ser cero
- P: 1 bit. Indica prioridad; en caso de que sea uno, el mensaje debe tener preferencia en su transmisión. Este bit es usado solamente para los mensajes de datos, por lo tanto, los mensajes de control deben tener este bit con cero.
- Versión: 4 bits. Indica la versión del encabezado de L2TP. Si es uno, permite la detección de paquetes L2F mezclados con paquetes L2TP. Si es dos, indica que son paquetes L2TP.
- Tamaño (*Length*): indica el tamaño total en bytes del mensaje
- Identificador del túnel (*Tunnel ID*): Es el identificador para el control de la conexión. Los túneles L2TP son nombrados por identificadores que tienen sólo significado local. En la creación de un túnel, se intercambian los *tunnel ID* por medio de AVPs.
- Identificador de la sesión (*Session ID*): es el identificador de una sesión en un túnel. Las sesiones L2TP son nombradas por identificadores que tienen sólo significado local. En el establecimiento de una sesión, se intercambian los *session ID* por medio de AVPs.

- Ns: indica el número de secuencia para el mensaje de control o de datos, comenzando en cero e incrementado en uno por cada mensaje enviado.
- Nr: indica el número de secuencia esperado en el próximo mensaje de control que será recibido. Nr es igual al Ns del último mensaje recibido más uno. En los mensajes de datos debe ser ignorado este campo.
- Tamaño del *Offset* (*Offset Size*): Si está presente, especifica el número de bytes que hay después del encabezado para iniciar con el mensaje, es decir, indica el número de bytes de relleno que contiene el paquete. Después del último byte de relleno, inicia el mensaje.

los tipos de mensajes de control

Los mensajes de control se distinguen porque el bit T del encabezado L2TP es igual a uno. Para maximizar la operación sin sacrificar su interacción con otros protocolos, existe un método para codificar los tipos de mensajes en L2TP. Este método utiliza mensajes AVP, los cuales definen el tipo específico de mensaje de control que se envía[44].

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td>M</td><td>H</td><td colspan="3">Reservados</td><td colspan="12">Tamaño</td><td colspan="16">Identificador del vendedor</td></tr>
<tr><td colspan="16">Tipo de atributo</td><td colspan="16">Valor del atributo</td></tr>
<tr><td colspan="32">Relleno hasta que el tamaño se alcance</td></tr>
</table>

Figura II.2.5.6

Los primero seis bits del *frame* son bits de máscara. Dos de ellos sí están definidos, pero los cuatro restantes están reservados para extensiones futuras. Mientras tanto, dichos bits reservados deben ser igual a cero.

- Bit M (*Mandatori bit*): Regula el comportamiento necesario para una implementación que reciba un mensaje AVP y que no sea reconocido. Si el bit M pertenece a un mensaje asociado a una sesión en particular, esa sesión debe ser terminada. Si el bit M pertenece a un mensaje asociado a un túnel en general, todo el túnel (incluyendo las sesiones establecidas sobre él) debe terminarse. Si se recibe un mensaje AVP no reconocido y el bit M es cero, el mensaje se ignora.

- Bit H (*Hidden bit*): Identifica el escondite de los datos en el campo Valor del Atributo (*Attribute Value*) en un mensaje AVP. Esta capacidad puede ser usada para evitar el transporte de datos importantes (tal como son las contraseñas de usuarios) como un texto común dentro de un mensaje AVP.
- Tamaño (*Length*): Identifica el número de bytes (incluyendo los bits de máscara) que conforman el mensaje AVP. El tamaño de este campo es de 10 bits, permitiendo un total de 1023 bytes de datos en un mensaje AVP. El tamaño mínimo es de 6 bytes y, en este caso, se entiende que el campo de Valor del Atributo (*Attribute Value*) no está presente. El tamaño se puede calcular como 6 + el tamaño en bytes del campo Valor del Atributo (*Attribute Value*).
- Identificador del vendedor (*Vendor ID*): Identifica el valor asignado por la IANA para cada vendedor a través de su código "SMI *Network Management Private Enterprise Codes*" [RFC1700]. El cero corresponde a los valores de atributo adoptados por IETF. Cualquier vendedor que desee implementar sus propias extensiones L2TP, puede usar su propio identificador con valores de atributo privados, siempre y cuando se garantice que no invadirán extensiones de otros vendedores ni del IETF. En este campo se tienen 16 bits, por lo que esta característica se limita a las primeras 65,535 empresas registradas en la IANA.
- Tipo de atributo: Campo de dos bytes con un valor único cuya interpretación en todos los mensajes AVP se define según el campo Identificador del Vendedor (*Vendor ID*).
- Valor del Atributo (*Attribute Value*): Se refiere al valor indicado en los campos Tipo de atributo e Identificador del Vendedor. El tamaño en bytes de este campo es igual al tamaño total del *frame* menos seis. Si el tamaño total del *frame* es seis, este campo no existe.

Los tipos de mensajes de control se definen según el valor del atributo:

Tabla II.2.5.1

Mensajes referentes al control de la conexión		
Valor del atributo	*Abreviatura*	*Significado*

0	--	Reservado
1	SCCRQ	Start-Control-Connection-Request
2	SCCRP	Start-Control-Connection-Reply
3	SCCCN	Start-Control-Connection-Connected
4	StopCCN	Stop-Control-Connection-Notification
5	--	Reservado
6	Hello	Hello

Tabla II.2.5.2

Mensajes referentes a la administración de la llamada		
Valor del atributo	*Abreviatura*	*Significado*
7	OCRQ	Outgoing-Call-Request
8	OCRP	Outgoing-Call-Reply
9	OCCN	Outgoing-Call-Connected
10	ICRQ	Incoming-Call-Request
11	ICRP	Incoming-Call-Reply
12	ICCN	Incoming-Call-Connected
13	--	Reservado
14	CDN	Call-Disconnected-Notify

Tabla II.2.5.3

Mensajes referentes al reporte de errores		
Valor del atributo	*Abreviatura*	*Significado*
15	WEN	WAN-Error-Notify

Tabla II.2.5.4

Mensajes referentes al control de la sesión PPP		
Valor del atributo	*Abreviatura*	*Significado*

16	SLI	Set-Link-Info

Ventajas y desventajas:

[43]Debido a que L2TP es un protocolo normalizado, todos los clientes, inclusive los proveedores de servicios y los administradores de redes corporativas, pueden aprovechar los servicios ofrecidos por diversos vendedores. La interoperabilidad entre los vendedores ayudará a asegurar el desarrollo de un estándar internacional de servicios de acceso a VPNs.

La implementación de L2TP es una solución que proporciona una larga lista de beneficios para las empresas usuarias, incluyendo principalmente:

- Seguridad y prioridad garantizada para sus aplicaciones más críticas
- Conectividad mejorada, costos reducidos y libertad de administrar los recursos propios como mejor convenga a las necesidades
- Ambiente de acceso remoto escalable y flexible sin afectar la seguridad de la red o dañar a las aplicaciones críticas de la misma[42]

A la vez, los proveedores de servicios obtienen otros beneficios al emplear acceso a VPNs por medio de L2TP:

- La capacidad de "provisionar", facturar y administrar el acceso a VPNs, proporcionando ventajas competitivas, minimizando los costos de producción de los clientes e incrementando la rentabilidad.
- La flexibilidad para ofrecer una amplia variedad de servicios VPN a través de diferentes arquitecturas
- La capacidad de proporcionar servicios diferenciados para seguridad y accesos remotos a VPNs sobre el Internet público o sobre la red de un proveedor de servicios.
- Interoperabilidad, puede ser usado como parte de toda una solución de acceso[42]
- Permite múltiples conexiones PPP en varios enrutadores *gateway*. Con ello, si se apilan los enrutadores, todos son vistos como una sola identidad[2]

La desventaja que actualmente tiene L2TP es que si se habilita el control de flujo con el comando correspondiente y se le da un valor diferente a cero, la trayectoria de conmutación se procesa a nivel físico, no lógico[42]. Además, aún no es implementado en muchos productos y la última milla se considera insegura en L2TP.

Implementación:

El siguiente diagrama ilustra un escenario típico de L2TP. La meta es tunelear los *frames* PPP entre el sistema remoto o el LAC del cliente y un LNS localizado en la LAN.

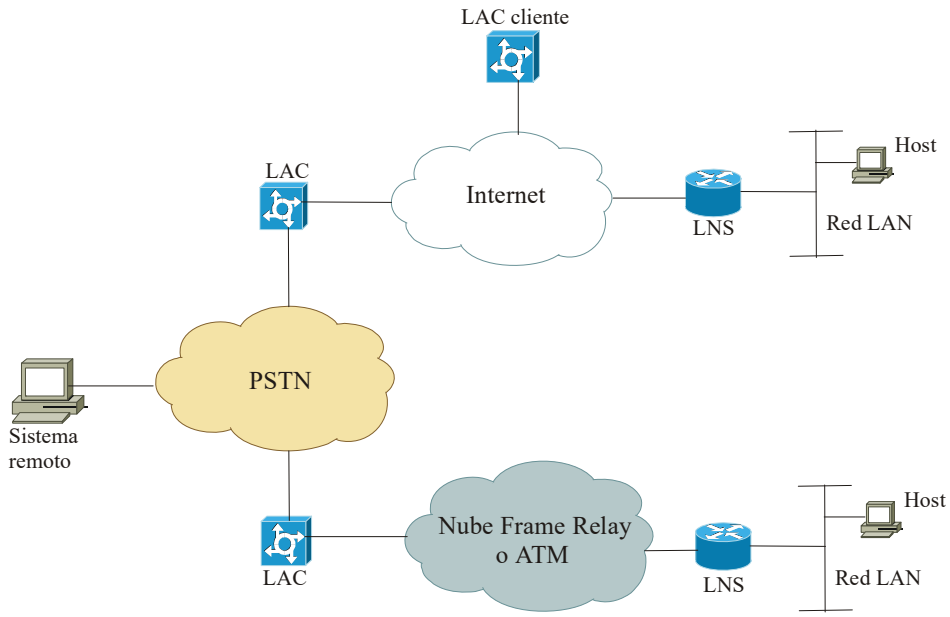


Figura II.2.5.7

El sistema remoto inicia una conexión PPP a través de la nube PSTN hacia un LAC. Entonces, el LAC tunelea la conexión PPP a través de Internet o de la nube Frame Relay o ATM hacia un LNS para lograr el acceso a la LAN deseada. Se le proporciona al sistema remoto con direcciones de la LAN por medio de la negociación NCP de PPP. Si el usuario estuviera conectado directamente a un NAS, el administrador de la

LAN podría efectuar AAA (*Authentication, Authorization y Accounting*) en ese momento, es decir, verificaría, autorizaría y contabilizaría la conexión.

Un LAC cliente (o sea, un *host* que corre L2TP) puede participar en el tuneleo a la LAN sin el uso de otro LAC. En este caso, el *host* que tiene el software necesario para parecer un LAC cliente ya tiene una conexión al Interner público. Entonces se crea una conexión virtual PPP y el *software* de L2TP en el LAC cliente crea un túnel hacia el LNS. También en este caso puede realizar la verificación, autorización y contabilización de la conexión por parte del administrador de la LAN

Referencias:

Frame Relay:

[1] "Frame Relay: Networks for Tomorrow and Today"

http://new.fr.forum.com/4000/4000index.html

[2] "NNI: The Key to Multi-Carrier Frame Relay"

l

[3] "A brief discussion of Switched Access to Frame Relay Services and Frame Relay Switched Virtual Circuits" http://new.frforum.com/4000/4007/4007.htm

[4] "FRFs" http://frame-relay.indiana.edu/5000/approved

[5] Internetworking Technologies Handbook. Segunda edición. Cisco Press. Cap. 10

ATM:

[6] http://www.monografias.com/trabajos/atm/atm.shtml

[7] http://www.coasin.cl/html/glosario/v.html

[8] http://www.coasin.cl/html/glosario/v.html

[9] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm

[10] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm

[11] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm

[12] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm

[13] http://neutron.ing.ucv.ve/revista-e/No4/ATM%20vs%20FR.html

[14] http://neutron.ing.ucv.ve/revista-e/No4/ATM%20vs%20FR.html

PPP:

[15] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ppp.htm

[16] http://studies.ac.upc.es/FIB/STD/lab/ppp.pdf

[17] http://www.ietf.org/rfc/rfc1661.txt?number=1661

[18] http://www.telecos-malaga.com/documentos.php?seccion=redes

[19] http://atenea.udistrital.edu.co/cursos/mt.redesI/grp04/ppp.html

IPsec:

[20] RFC 2401. Security Architecture for IP.

[21] Deploying Ipsec. Reference Guide. Cisco Systems.

[22] http://www.eng.uts.edu.au/~kumbes/ra/vpn/vpn02a.htm#VPN%20History

[23] RFC 2402. IP Authentication Header.

GRE:

[24] Berkowitz, Howard C. Designing Routing & Switching Architectures for Enterprise Networks. Network Architecture & Development Series. Ed. MacmillanTechnical Publishing USA, pp. 772-773

[25] Berkowitz, Howard C. WAN Survival Guide. Strategies for VPNs & Multiservice Networks. Ed. Wiley. USA, 2001. pp. 225-227

[26] Kaeo, Merike.Designing Network Security. A practical guide to creating a secure network infrastructure. Cisco Systems. USA, 1999. pp. 337, 340 y 341.

[27] http://www.qnx.com/developer/docs/momentics_nc_docs/neutrino/technotes/gre.html

[28] http://www.networksorcery.com/enp/protocol/gre.htm

L2F:

[29] RFC 2341, http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2341.html

[30] http://www-uka.rz.uni-karlsruhe.de/~Tobias.Zimmer/vpn/node8.html

[31] http://www.certifiednow.com/ExpertISP/l2f.htm

[32] http://www.networksorcery.com/enp/protocol/l2f.htm

[33] http://www.ee.ust.hk/~eejie/vpnmpls/tsld021.htm

[34] http://networking.champlain.edu/gck/security_sit/tsld018.htm

PPTP:

[35] http://www.windowstimag.com/atrasados/1999/29_mar99/articulos/internet.htm

[36] http://www.argo.es/~jcea/artic/vpn1.htm

[37] ftp://ftp.rediris.es/docs/rfc/26xx/2637

[38] http://www.microsoft.com/windows2000/docs/vpndeploy.doc

[39] http://www.microsoft.com/windows2000/technologies/communications/vpn/default.asp

[40] http://www.microsoft.com/ntserver/ProductInfo/faqs/PPTPfaq.asp#1

L2TP:

[41] RFC 1918

[42] Layer 2 Tunnel Protocol, Feature Summary.

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t1/l2tpt.pdf

[43] Layer 2 Tunnel Protocol, A Feature in Cisco IOS Software.

http://www.cisco.com/warp/public/cc/pd/iosw/prodlit/l2tun_ds.pdf

[44] RFC 2661 ftp://ftp.isi.edu/in-notes/rfc2661.txt

Capítulo III BASES TEÓRICAS PARA VPNs *PEER-TO-PEER*

Con formato: Español (México)

III.1.1. Introducción a MPLS

Para transmitir un paquete en IP tradicional, primero se revisa la dirección IP destino contenida en el encabezado de la capa de red en un paquete, transportándose así desde su origen hasta su destino final. El enrutador analiza la dirección IP destino independientemente de cada salto en la red. Los protocolos dinámicos de enrutamiento o las configuraciones estáticas construyen una base de datos conocida como Tabla de Enrutamiento, necesaria para analizar la dirección IP destino. Sin embargo, aun con su éxito y su amplio uso, este tipo de enrutamiento tiene restricciones que disminuyen su flexibilidad. Es por esta razón que han sido desarrolladas nuevas técnicas para incrementar la funcionalidad de las redes basadas en IP y para proveer soluciones a estos problemas se creó MPLS.

Para entender todas las cuestiones que afectan la escalabilidad y la flexibilidad de las redes basadas en envío IP, es necesario revisar algunos de los mecanismos básicos del envío basado en IP y sus interacciones con la infraestructura de las capas inferiores (LAN o WAN). Con esta información, se puede identificar cualquier inconveniente y tal vez proveer ideas alternativas para que pueda ser implementado.

esquema de enrutamiento de la capa de red

La transmisión tradicional de paquetes en la capa de red (IP a través de Internet), confía en la información que le proporciona el protocolo de enrutamiento de la capa de red (por ejemplo, OSPF o BGP) o el enrutamiento estático para hacer una decisión de envío independiente en cada salto (es decir, en cada enrutador) dentro de la red. Esta decisión está basada solamente en la dirección IP *unicast* destino. Todos los paquetes con el mismo destino siguen la misma ruta a través de la red si es que no existen costos asignados. Cuando un

enrutador tiene dos caminos con el mismo costo hacia un destino, los paquetes pueden tomar una o ambas rutas, teniendo como resultado un balanceo de carga.

Los enrutadores ejecutan el proceso de decisión que selecciona qué ruta debe tomar el paquete. Estos dispositivos participan en la recolección y distribución de la información al nivel de la capa de red y ejecutan la conmutación de acuerdo a los contenidos de los encabezados de la capa 3 en cada paquete. Se pueden conectar los enrutadores directamente por medio de enlaces punto a punto (point-to-point) o con redes de área local (por ejemplo, un *hub* compartido o un MAU) o pueden conectarse por medio de conmutadores LAN o WAN (por ejemplo conmutadores Frame Relay o ATM). Desafortunadamente, los conmutadores de Capa 2 (LAN o WAN) no tienen la capacidad de soportar información de enrutamiento de Capa 3 o de seleccionar el camino que tomará el paquete según el análisis de las direcciones destino de Capa 3. Por esta razón, los conmutadores de Capa 2 (LAN o WAN) no pueden estar involucrados en el proceso de decisión de envío de paquetes de Capa 3. En el caso de un ambiente WAN, el diseñador de la red tiene que establecer manualmente las rutas de capa 2 por las que se envían los paquetes de capa 3 entre los enrutadores, los cuales son conectados físicamente a la red de Capa 2.

Las rutas de una LAN Capa 2 son sencillas de establecer, ya que todos los conmutadores LAN son transparentes a los equipos conectados a ésta. En cambio, el establecimiento de las rutas WAN de Capa 2 es más complejo. Estas rutas usualmente están basadas en el esquema punto a punto (*point-to-point*; por ejemplo los circuitos virtuales en la mayoría de las redes WAN) y son establecidas sólo por demanda a través de una configuración manual. Cualquier dispositivo de enrutamiento (como el *ingress router* o enrutador de ingreso) de la red de Capa 2 que quiera enviar paquetes de Capa 3 a cualquier otro dispositivo de enrutamiento (como el *egress router* o enrutador de egreso) necesita establecer una conexión directa a través de la red hacia el dispositivo destino o mandar los datos a diferentes dispositivos para la transmisión hacia el destino final.

Se puede considerar el ejemplo mostrado en la figura III.1.1.1:

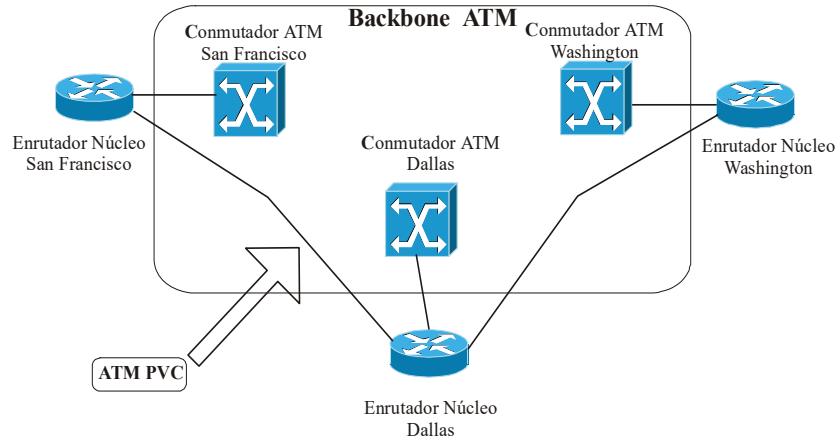


Figura III.1.1.1. Ejemplo de una red IP basada en núcleos ATM

La red ilustrada en la figura III.1.1.1 está basada en un núcleo ATM rodeada por enrutadores que ejecutan el envío de paquetes a nivel de la capa de red. Asumiendo que las conexiones entre los enrutadores solamente son como lo indica la figura III.1.1.1, todos los paquetes mandados desde San Francisco hacia Washington deben pasar por el enrutador de Dallas, en donde son analizados y enviados de regreso sobre la misma conexión ATM en Dallas hacia el enrutador en Washington. Este paso extra introduce un retardo en la red y sobrecarga al enrutador de Dallas innecesariamente, así como al enlace ATM entre el enrutador de Dallas y el conmutador adyacente ATM en Dallas.

Para lograr el envío óptimo de los paquetes en la red, debe existir un circuito virtual ATM entre cualquiera de los dos enrutadores conectados al núcleo ATM, lo cual puede ser fácil de implementar en redes pequeñas, como la mostrada en la figura III.1.1.1, pero hay serios problemas de escalabilidad cuando se quiere implementar en redes grandes (con 10 o cientos de enrutadores) que pertenezcan al mismo núcleo WAN.

Los problemas de escalabilidad surgen cuando una red crece al incrementarse el número de enrutadores. Por ejemplo, si se quiere un enrutamiento óptimo, cada vez que un nuevo enrutador es conectado al núcleo WAN de la red, se debe establecer un nuevo circuito virtual entre ese enrutador y el resto de la red. Además, con

cierta configuración del protocolo de enrutamiento, cada enrutador adjunto a la Capa 2 del núcleo WAN (construido con conmutadores ATM o Frame Relay) necesita un circuito virtual dedicado a los demás enrutadores dentro del mismo núcleo. Inclusive para lograr una redundancia deseable en el núcleo, cada enrutador también debe establecer una adyacencia en el protocolo de enrutamiento con cada uno de los enrutadores del mismo núcleo. El resultado es una adyacencia de enrutadores de malla completa (*Full-mesh*), teniendo como consecuencia que cada enrutador tenga un número grande de vecinos de protocolo de enrutamiento, es decir, una cantidad muy grande de tráfico de enrutamiento. Por ejemplo, si una red corre bajo OSPF o IS-IS como protocolos de enrutamiento, cada enrutador propaga cada cambio en la topología de la red hacia cada uno de los demás enrutadores conectados al mismo núcleo WAN, resultando en un tráfico de enrutamiento proporcional al cuadrado del número de enrutadores.

Con lo anterior, el mantenimiento y la administración de los circuitos virtuales entre los enrutadores es complejo, ya que es muy difícil predecir el monto exacto del tráfico entre cualquiera de los dos enrutadores de la red. Para simplificarlo, algunos Proveedores de Servicios han optado por una garantía del servicio en la red, como una CIR igual a Cero (*Zero Committed Information Rate*) en redes Frame Relay o una Tasa de Bits No Específica (*Unspecified Bit Rate*) para redes ATM.

La necesidad del intercambio de información entre los enrutadores y los conmutadores WAN no fue una cuestión sencilla para los ISPs que usaban *backbones* constituidos únicamente por enrutadores o para los Proveedores de Servicio tradicionales que daban sólo servicios WAN (Circuitos virtuales ATM o Frame Relay), pues debido a muchos factores ambos grupos fueron obligados a mezclarse en el diseño del *backbone*. Uno de dichos factores es que los Proveedores de Servicios tradicionales se vieron en la necesidad de ofrecer servicios IP, pero ellos querían que los nuevos servicios estuvieran basados en la infraestructura WAN que ya tenían para proteger su inversión, además de que querían ofrecer una Calidad de Servicio (QoS) robusta que es más fácil implementar en su infraestructura que en los enrutadores tradicionales.

El rápido incremento de las necesidades de ancho de banda hizo que la introducción de nuevos enrutadores con interfaces ópticas obligaran a los Proveedores de Servicios a empezar a confiar en la tecnología ATM,

pues en ese tiempo las interfaces de los enrutadores no daban las velocidades ofrecidas por los conmutadores ATM.

Con los argumentos anteriores, es claro que se deben usar mecanismos diferentes para habilitar el intercambio de información de capa de red entre los enrutadores y los conmutadores WAN y, además, permitirle a los conmutadores participar en el proceso de decisión de envío de paquetes sin ser necesarias las conexiones entre enrutadores frontera.

servicio de diferenciación de paquetes

El envío convencional de paquetes IP usa solamente la dirección IP destino contenida en el encabezado de Capa 3 del paquete para hacer la decisión de envío. En la figura III.1.1.2, por ejemplo, el enlace directo entre el enrutador núcleo de San Francisco y el enrutador núcleo de Washington envía el tráfico que entra a la red en cualquiera de los POP's del área, aunque este enlace pueda congestionarse y, en cambio, el enlace de San Francisco a Dallas y de Dallas a Washington pueden no saturarse tanto como el otro enlace.

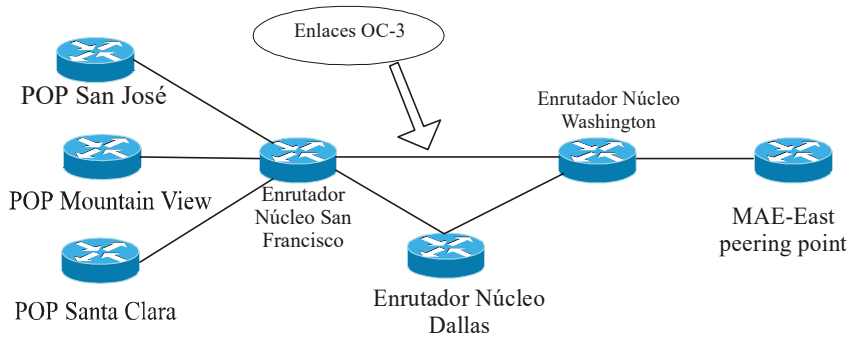


Figura III.1.1. 2.- Ejemplo de una red que podría beneficiarse de la ingeniería de tráfico

Aunque existen ciertas técnicas que afectan el proceso de decisión, tales como el Enrutamiento Basado en Políticas (*Policy Based Routing* o PBR), no hay una técnica que, únicamente con ella, se pueda tomar la

decisión en una ruta saturada por dónde va a ir el paquete hacia su destino final. En la red mostrada en la figura III.1.1.2, el proceso de enrutamiento debe desarrollarse en el enrutador núcleo de San Francisco para desviar un poco de tráfico del área a Washington a través de Dallas, reduciendo drásticamente el rendimiento en el enrutador núcleo. Idealmente, un enrutador frontera (por ejemplo, el POP Santa Clara) puede especificar por cuál enlace deben fluir los paquetes.

Debido a que la mayoría de los Proveedores de Servicios tienen redes con rutas redundantes, existe un requerimiento para permitir que el dispositivo de enrutamiento de ingreso (o *ingress router*) sea capaz de decidir el envío de paquetes, lo cual afecta la ruta que el paquete toma en la red. Una opción puede ser la aplicación al paquete de una etiqueta que indique a los otros dispositivos por cuál ruta se debe transmitir el paquete. Este requerimiento también puede permitir a los paquetes destinados a la misma red IP tomar diferentes caminos en vez de uno solo determinado por el protocolo de enrutamiento de Capa 3. Esta decisión debería estar basada en factores diferentes a la dirección destino IP, como el puerto por el que fue enviado el paquete o el nivel de Calidad de Servicio que el paquete necesita, etc.

control y envío independiente

Con el envío convencional de paquetes IP, cualquier cambio en la información que controla el envío de paquetes se comunica a todos los dispositivos dentro del dominio de enrutamiento. Este cambio siempre involucra un periodo de convergencia con el algoritmo de envío, por lo que es deseable un mecanismo que pueda cambiar la forma en la que el paquete es enviado sin afectar a otros dispositivos en la red.

Para implementar un mecanismo así, los enrutadores no deben confiar en la información del encabezado IP para transmitir el paquete, y en lugar de eso que mejor sea posible agregar una etiqueta adicional para indicar el comportamiento del paquete al ser reenviado. Cuando el envío está basado en etiquetas adjuntas al paquete IP original, cualquier cambio dentro del proceso de decisión puede comunicarse a otros dispositivos a través de las nuevas etiquetas de distribución y, debido a que estos dispositivos envían paquetes basados en

etiquetas, puede ocurrir un cambio sin impactar a todos los dispositivos que ejecutan la transmisión del paquete.

propagación de la información de enrutamiento externo

La transmisión convencional de paquetes IP requiere que la información de enrutamiento externo sea notificada a todos los dispositivos de enrutamiento. Esto es necesario porque los paquetes se enrutan según la dirección destino contenida en el encabezado de la capa de red del paquete. Para continuar con el ejemplo de la sección anterior, los enrutadores núcleo de la figura III.1.1.2 tienen que almacenar todas las rutas de Internet para que los enrutadores puedan propagar paquetes entre los clientes.

El intercambio de información de enrutamiento externo tiene implicaciones de escalabilidad en términos de la propagación de las rutas, la memoria a usar y la utilización del procesamiento del CPU de los enrutadores del núcleo de la red y, en realidad, es exagerado considerando que lo único que se quiere hacer es pasar los paquetes de enrutador a enrutador. Por ello, es muy recomendable un mecanismo que permita a los dispositivos de enrutamiento interno conmutar los paquetes a través de la red de un dispositivo de ingreso a través de otro de egreso, sin analizar la dirección destino de la capa de red del paquete.

III.1.2. Descripción de MPLS

Multiprotocol Label Switching o MPLS es una tecnología que emergió de las emisiones existentes asociadas a la transmisión de paquetes en el ambiente actual del internetworking. Los miembros de la IETF lograron hacer un estándar de todas las ideas y tecnologías relacionadas con la conmutación de etiquetas. Para optimizar los procesos de enrutamiento, MPLS propone una conmutación de paquetes basada en etiquetas o *labels*, en lugar de las direcciones IP.

La diferencia más significativa entre MPLS y las tecnologías tradicionales WAN es la forma en la que las etiquetas son asignadas y la capacidad de llevar una pila de etiquetas adjuntas al paquete. El concepto de la

pila de etiquetas implementa nuevas aplicaciones, tales como la ingeniería de tráfico, las Redes Virtuales Privadas, un re-enrutamiento rápido cuando hay una falla en el envío o en el nodo, etc.

La transmisión de paquetes en MPLS tiene un severo contraste con los ambientes de las redes no orientadas a conexión, en donde cada paquete es analizado de acuerdo a su siguiente salto y se revisa su encabezado de Capa 3 para tomar una decisión de envío basada en la información extraída con el algoritmo de enrutamiento de la capa de red. MPLS ejecuta la conmutación de etiquetas, combinando los beneficios de un envío de paquetes basado en Capa 2 con los beneficios del enrutamiento en Capa 3. Similarmente a las redes de Capa 2 (como Frame Relay o ATM), MPLS asigna una etiqueta a los paquetes para ser transportados a través de redes basadas en celdas o paquetes. El mecanismo de esta tecnología es la conmutación de etiquetas, en la cual las unidades de datos (paquetes o celdas) son llevadas por etiquetas cortas que le dicen a los nodos de conmutación cómo procesar y enviar los datos.

La arquitectura de MPLS está dividida en dos componentes separados: los componentes de envío (también llamado Plano de Datos) y los componentes de control (también llamado Plano de Control). El Plano de Datos usa una base de datos de etiquetas ubicada en el conmutador para ejecutar el envío de los paquetes según las etiquetas que porten. El Plano de Control es el responsable de la creación y el mantenimiento de la información de envío con etiquetas (referido como *binding*) por medio de un grupo de conmutadores de etiquetas interconectadas. La figura III.1.2. muestra la arquitectura básica de un nodo MPLS ejecutando enrutamiento IP.

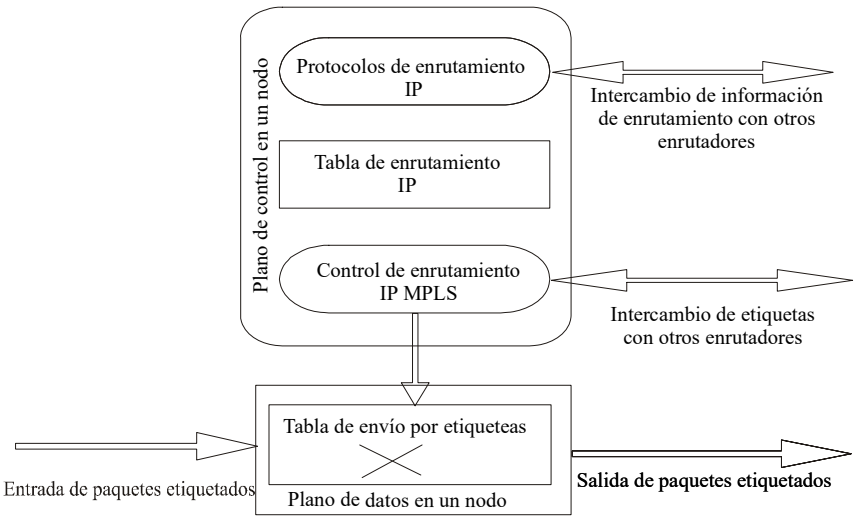


Figura III.1.2. Arquitectura básica de MPLS

Con MPLS, cada nodo debe ejecutar uno o más protocolos de enrutamiento IP (o emplear las rutas estáticas) para intercambiar información de enrutamiento IP con otros nodos MPLS en la red. En este sentido, cada nodo MPLS (incluyendo conmutadores ATM) es un enrutador IP en el Plano de Control.

En los enrutadores tradicionales, los protocolos de enrutamiento IP llenan las tablas de enrutamiento global. Esta tabla global es útil para construir la tabla de envío de IP (*Forwarding Information Base*, FIB) creada por el proceso de balanceo de carga CEF (*Cisco Express Forwarding*), necesario para implementar MPLS.

Por otra parte, en un nodo MPLS se usa la tabla de enrutamiento global para determinar el intercambio de etiquetas o *binding* con otros nodos adyacentes MPLS con el fin de conocer las subredes individuales que están contenidas dentro de la tabla de enrutamiento. El intercambio de etiquetas o *binding* para el enrutamiento basado en IP *unicast* es ejecutado usando el protocolo propietario de Cisco, *Tag Switching Protocol* (TDP), o el protocolo especificado por la IETF, *Label Distribution Protocol* (LDP).

El proceso de control de enrutamiento IP MPLS usa el intercambio de etiquetas entre los nodos para construir la tabla de envío de etiquetas, construida a partir de la información del plano de datos que es usada para el envío de paquetes etiquetados a través de la red MPLS.

La construcción de las tablas de "enrutamiento" que necesita MPLS se describe más adelante.

III.1.3. Arquitectura de MPLS

Como con cualquier tecnología nueva, se introducen muchos términos nuevos para describir los dispositivos que conforman a la arquitectura. Estos nuevos términos describen la funcionalidad de cada uno de los dispositivos y sus papeles dentro de la estructura MPLS.

Uno de los dispositivos es el Enrutador de Conmutación de Paquetes, mejor conocido como LSR (*Label Switch Router*). Cualquier enrutador o conmutador que implemente los procedimientos de distribución de etiquetas y que pueda enviar paquetes basados en ellas, entra dentro de esta categoría. La función básica de los procedimientos de distribución de etiquetas es permitir al LSR distribuir sus etiquetas hacia otros LSRs dentro de la red MPLS.

Existen diferentes tipos de LSR que se diferencian por la funcionalidad que proveen dentro de la infraestructura de la red. Estos diferentes tipos de LSR son descritos dentro de la arquitectura como *Edge-LSR*, ATM-LSR y *ATM-Edge-LSR*. La diferencia entre varios tipos de LSR es puramente en términos de arquitectura (una sola caja puede asumir muchos papeles).

Un *Edge-LSR* es un enrutador que ejecuta imposiciones de etiquetas (a veces referido como *PUSH*) o disposición de etiquetas (también llamado *POP*) en las fronteras de la red MPLS. La imposición de etiquetas o *push tag* es el acto de poner al principio la etiqueta o la pila de etiquetas al paquete en el punto de ingreso (con respecto al flujo desde la fuente hasta el destino) del dominio MPLS. La disposición de etiquetas o *pop*

tag es el inverso de la imposición de etiquetas, ya que es el acto de remover las etiquetas del paquete en el punto de egreso antes de ser enviado a su vecino que está fuera del dominio de MPLS.

Cualquier LSR que tenga un vecino que no pertenezca a MPLS es considerado un *Edge-LSR*, es decir, si un LSR tiene cualquier interfaz conectada a través de MPLS y otra interfaz conectada a una red diferente a MPLS, se considera un LSR frontera.

Los enrutadores ATM-LSR y los ATM Edge-LSR son equivalentes a los ya descritos, realizan las mismas funciones, pero para comprender su diferencia es importante aclarar que existen dos modos principales de MPLS: el *Frame Mode* y el *Cell Mode*. El *Frame Mode* es el modo de MPLS en el que la infraestructura sobre la que se implementó tiene enrutadores conectados por cualquier medio (FR, ISDN, etc.), excepto ATM. El *Cell Mode*, en cambio, es el modo en el que se tienen únicamente conmutadores ATM o, en su defecto, enrutadores con interfaces ATM, las cuales tienen configuradas MPLS. Se hace la distinción de los dos modos porque en el *Frame Mode* es posible retirar las etiquetas (*Pop Tag*) sin ningún problema, pero en el *Cell Mode* no, debido al uso de VCI y VPI en ATM, pues si se quitan las etiquetas, los paquetes se quedan sin identificadores y ya no es posible conmutarlos. Así, tenemos que el LSR y el *Edge-LSR* pertenecen al *Frame Mode*, y el ATM-LSR y el *ATM Edge-LSR* al *Cell Mode*.

El *Edge-LSR* usa una tabla de envío IP tradicional, con información de etiquetado, para poder poner etiquetas a los paquetes o para poder quitárselas, según sea el caso. La figura III.1.3 muestra la arquitectura de un *Edge-LSR*.

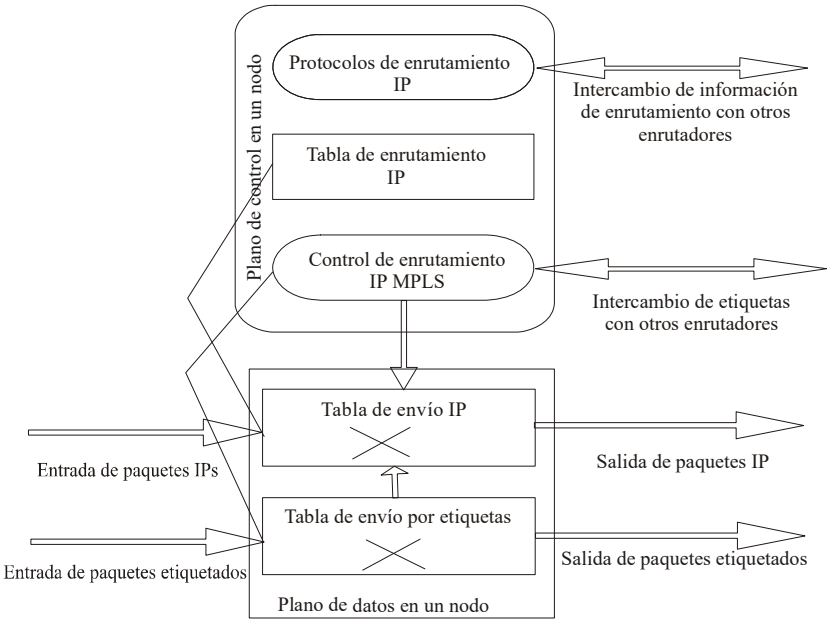


Figura III.1.3. Arquitectura de un Edge-LSR

Un *Edge-LSR* extiende la arquitectura de un nodo MPLS (como el mostrado en la figura Figura III.1.3) con componentes adicionales dentro del plano de datos. La tabla de envío IP estándar está construida a partir de la tabla de enrutamiento IP y se complementa con la información del etiquetado. Los paquetes entrantes IP pueden ser enviados como paquetes IP puros a los nodos que no pertenecen al dominio MPLS o pueden ser etiquetados y mandados como paquetes etiquetados para otros nodos MPLS. Los paquetes entrantes que tienen etiqueta MPLS se mandan como paquetes etiquetados hacia otros nodos MPLS y para paquetes con etiqueta MPLS destinados a nodos que no son MPLS, la etiqueta se remueve y se ejecuta un envío IP (Capa 3) para buscar el destino.

Un ATM-LSR es un conmutador ATM que puede actuar como un LSR. El ATM-LSR ejecuta el enrutamiento IP y la asignación de etiquetas en el Plano de Control, y manda los paquetes de datos usando las técnicas de envío tradicional de celdas ATM en el Plano de Datos. En otras palabras, la matriz de conmutación ATM del

conmutador ATM es usada como una tabla de envío de etiquetas de un nodo MPLS. Los conmutadores tradicionales ATM pueden hacer la función de ATM-LSR a través de una actualización de su *software*.

En la tabla III.1.3. se resumen las funciones ejecutadas por diferentes tipos de LSR. Cabe hacer la aclaración de que cada dispositivo puede ejecutar más de una función.

Tabla III.1.3.

Tipo de LSR	**Acciones ejecutadas según el tipo de LSR**
LSR	• Envía los paquetes etiquetados.
Edge LSR	• Puede recibir un paquete IP, ejecutar una búsqueda de Capa 3, e imponer una pila de etiquetas antes de enviar el paquete dentro del dominio del LSR. • Puede recibir un paquete etiquetado, remover la etiqueta, ejecutar búsqueda de destino en Capa 3 y enviar el paquete IP a hacia su siguiente salto.
ATM-LSR	• Corre protocolos MPLS en el plano de control para instalar circuitos virtuales ATM. • Transmite paquetes etiquetados como celdas ATM.
ATM Edge-LSR	• Puede recibir un paquete etiquetado o no, se segmenta en celdas ATM y se envía por medio de su próximo salto a un ATM-LSR • Puede recibir celdas ATM desde su ATM-LSR adyacente, reensamblar estas celdas en el paquete original y entonces enviar el paquete como un paquete etiquetado o no.

La arquitectura MPLS usa la asignación de etiquetas "*downstream*" por demanda para tráfico *unicast*. Además de este tipo de asignación, existe otra opción llamada asignación "no solicitada", la cual permite que las etiquetas sean únicas globalmente o únicas por nodo, o únicas por interfaz. El *stack* de etiquetas de MPLS es del tipo *Last-In-First-Out* o LIFO por sus siglas, el cual indica que la última etiqueta en ser recibida, es la primera en salir. El *stack* LIFO se emplea como mecanismo para soportar jerarquías en las etiquetas. Con él, las decisiones de envío se realizan siempre con base en la etiqueta superior del *stack*.

Para la arquitectura existen dos mecanismos de selección de la trayectoria: salto a salto (o *hop-by-hop*) y el enrutamiento explícito. Con el mecanismo *hop-by-hop*, el siguiente salto se elige usando los resultados del cálculo normal del enrutamiento de la capa de red. Con el mecanismo del enrutamiento explícito, la ruta se especifica por el origen y es necesario que todos los LSRs sean capaces de enviar paquetes enrutados explícitamente, pero no tienen que ser capaces de originarlos.

La arquitectura no define una encapsulación para datos etiquetados, pero permite dos opciones: usar una encapsulación desarrollada específicamente para MPLS o usar "lugares disponibles" en las encapsulaciones de la capa de enlace de datos. Un ejemplo obvio de este último caso son los campos VCI/VPI de ATM. A propósito de ATM, existe un documento dentro del conjunto que define a MPLS que trata el caso especial de ATM, pues se discuten las posibles codificaciones de etiquetas en el VCI y en el VPI.

III.1.4. Encapsulación

Para aquellos tipos de enlaces en los que que no se pueden acomodar etiquetas en el encabezado de Capa 2 (es decir, para todos los tipos de enlaces excepto ATM y Frame Relay) MPLS usa un encabezado que consiste en palabras de 32 bits. El formato de dicho encabezado se muestra a continuación:

Figura III.1.4. Encabezado MPLS

La descripción se detalla a continuación:

- Etiqueta: 20 bits. Número aleatorio entre 16 y 2^{20} que se utiliza para la conmutación de paquetes.
- Experimental: 3 bits. El uso de estos tres bits aún no se decide por el grupo de trabajo MPLS, Sin embargo, es común que se usen para soportar calidad de servicio.
- *Stack:* 1 bit. Indica que se ha alcanzado la última etiqueta del *stack*.

- TTL*:* 8 bits. Es el campo que indica el número de saltos que ha dado el paquete. El máximo es 255.

La especificación de la encapsulación incluye un conjunto de reglas para el procesamiento TTL. El objetivo de dichas reglas es que una red MPLS se comporte de manera similar a una red IP convencional, tanto como sea posible desde el punto de vista del TTL. Esto último significa que un paquete IP que cruce un dominio MPLS debe salir con un campo TTL modificado por el número de Ps o saltos en el dominio MPLS. Esto se logra copiando el campo TTL del encabezado IP en el campo TTL de MPLS cuando el paquete se etiqueta por primera vez, se decrementa el campo TTL de MPLS en cada P y éste se copia nuevamente al encabezado IP cuando la etiqueta se remueve en su egreso del dominio MPLS. Una alternativa es decrementar en uno el campo TTL en el encabezado IP cada vez que la etiqueta se remueve, teniendo el efecto de que la trayectoria de conmutación de etiquetas o LSP (*Label Switch Path*), se observe como un solo salto.

Un tema importante dentro de la encapsulación es la fragmentación. Es posible que se requiera fragmentación para paquetes ya etiquetados, tal como con los paquetes IP, pues al añadir una o más etiquetas a los paquetes, se puede ocasionar que éste sea demasiado grande o, incluso, un paquete sin etiqueta puede ser tan grande que para su transmisión se necesite fragmentarlo. Si un paquete etiquetado necesita ser fragmentado, la fragmentación se aplica al paquete IP sobre el que está el paquete MPLS, resultando un *stack* para cada fragmento.

Por último, dentro de la especificación de la encapsulación, se definen valores reservados para las etiquetas, los cuales son:

- 0: Valor nulo explícito para IPv4
- 1: Alerta del enrutador
- 2: Valor nulo explícito para IPv6
- 3: Valor nulo implícito

Las etiquetas de valor nulo explícito se usan en casos donde la encapsulación de etiquetas no necesita ser validada. La alerta del enrutador se usa como una opción en el paquete IP para señalar que un enrutador

necesita poner más atención en ese paquete para simplificar su envío. El valor nulo implícito aparece en el encabezado del paquete transmitido, pero se reserva su uso en el protocolo LDP (*Label Distribution Protocol*).

III.1.5. protocolo de distribución de etiquetas (LDP)

Existe una gran variedad de protocolos que se usan para la distribución de etiquetas, tales como BGP y RSVP, pero el protocolo LDP es típicamente el más conocido de estos mecanismos, siendo el protocolo principal de aquellos creados especialmente para MPLS y uno de los que soportan enrutamiento *unicast*. LDP tiene las siguientes características básicas:

- El LSR proporciona un mecanismo de descubrimiento de vecinos para establecer comunicación.
- Se definen cuatro tipos de mensajes:
  - Mensajes DISCOVERY
  - Mensajes ADJACENCY (usados para inicialización, keepalive y términos de sesión entre LSRs).
  - Mensajes de anuncios de etiquetas o LABEL.
  - Mensajes NOTIFICATION.
- Se ejecuta sobre TCP para proporcionar una entrega confiable de mensajes, con excepción de los mensajes DISCOVERY.
- Está diseñado para ser extensible fácilmente, usando mensajes conocidos como colecciones TLV (*Type, Lenght, Value*).

III.1.5.1. descubrimiento de vecinos en el núcleo MPLS

Para generalizar la explicación y que ésta sea coincidente para los dos tipos de modos de MPLS, el término de los enrutadores en la red MPLS será LSR, como se hacía hasta antes de su distinción entre enrutadores LSR, Edge-LSR, ATM LSR y ATM Edge LSR.

El protocolo para descubrir vecinos en LDP se ejecuta sobre UDP. Para ello, un LSR envía periódicamente un mensaje *multicast* HELLO a un puerto UDP conocido por todos los enrutadores de esa red. Los demás LSRs escuchan los mensajes HELLO en ese puerto, con lo que un LSR puede aprender cuáles son los LSRs a los que está directamente conectado. Cuando un LSR aprende la dirección de otro LSR por medio de este mecanismo, se establece una conexión TCP a aquél LSR. Después de ello, es posible el establecimiento de una sesión LDP entre los dos LSRs. Una sesión LDP es bidireccional, por lo que el LSR de cada extremo puede anunciar o solicitar *bindings* hacia el otro extremo.

El mecanismo de descubrimiento de vecinos LSRs permite que éstos se conozcan aún cuando no están directamente conectados o cuando no tienen una subred en común. En este caso, un LSR envía periódicamente mensajes *unicast* HELLO hacia un puerto UDP con una dirección IP específica, la cual debe ser aprendida por otro medio (como por ejemplo, por configuración). El receptor del mensaje HELLO debe responder con otro mensaje *unicast* HELLO y hasta después de ello, se procede a establecer la sesión TCP y, Posteriormente, la LDP.

Una situación típica en la cual el mecanismo de descubrimiento de vecinos es útil se da cuando se configura ingeniería de tráfico entre dos LSRs y se desea enviar mensajes ya etiquetados sobre ese enlace.

III.1.5.2. transporte confiable de datos

LDP se ejecuta sobre TCP debido a que se necesita confiabilidad en la entrega de datos. Si un paquete etiquetado no se entrega oportunamente, el tráfico no puede conmutar la etiqueta y el paquete tendrá que ser manejado por el procesador de control o se desechará. Además de la confiabilidad, se emplea TCP para asegurar que la entrega de datos se haga en orden, pues LDP no puede realizar esta funcionalidad por sí solo. Sin embargo, la confiabilidad en LDP se construye hasta el nivel de funcionalidad necesario, pues TCP proporciona un mecanismo para evitar congestión, el cual no es estrictamente necesario para un protocolo de

control de vecino a vecino y, en cambio, el orden completo de mensajes que da es más estricto que el requerido para la distribución de etiquetas.

Las ventajas de la confiabilidad en LDP pesan más que las desventajas. Por ejemplo, debido a que cada mensaje etiquetadas debe ser reconocido, Sería necesario un contador de tiempo para cada mensaje no reconocido, pero LDP delega esta función a TCP, el cual puede usar un solo contador para toda una sesión, debido a que el sobreflujo generado por muchos contadores puede ser significativo.

TCP proporciona una gran cantidad de funciones que LDP es capaz de usar libremente, tales como la encapsulación eficiente de mensajes de capas superiores en paquetes IP y el control de flujo. Exceptuando el control de congestión, el control de tráfico es necesario para un protocolo de control, siempre y cuando se asegure que un transmisor no sobrepase la capacidad del receptor.

III.1.5.3. modos de distribución de etiquetas

La distribución y asignación de etiquetas pueden ser ejecutados en diferentes maneras. Una de las alternativas es el modo no solicitado *downstream* y su contraparte, el modo de asignación de etiquetas bajo demanda, las cuales ya fueron explicadas. Otra alternativa es el modo ordenado y su contraparte, el modo independiente. Cuando un conmutador de etiquetas es usado para soportar el enrutamiento según su destino, cada LSR puede hacer una decisión independiente para asignar una etiqueta a una dirección IP y para anunciar tal asignación a sus vecinos. A este modo de asignación se le llama independiente y, de esta manera, el establecimiento de una trayectoria o LSP (*Label Switched Path*) logra una convergencia casi inmediata.

Por otro lado, en el modo ordenado, la asignación de etiquetas procede, como su nombre lo indica, de forma ordenada de un LSR a otro. En este modo de asignación, los LSRs de egreso son los únicos que puede iniciar el proceso de establecimiento del LSP o ruta, Es decir, los LSRs frontera de un sitio dado asignan una etiqueta para su dirección IP y anuncian dicha asignación a sus vecinos LSRs. Este proceso es repetido en la misma

forma por los demás LSRs de la red. De esta manera, la asignación de etiquetas procede en forma ordenada del egreso al ingreso. Además, la selección de FECs puede ser realizada en el LSR que se encuentre al inicio de la ruta y dicha LSR será empleada por los demás LSRs que formen parte de esa ruta.

En general se tiene un modo de asignación de paquetes llamado *Label Space* que indica el tipo de interfaz con el que se está trabajando. Si se tiene cero al final del identificador de la interfaz, se entiende que se trabaja con cualquier tipo de interfaz y se tiene el *Frame Mode*. Si es uno, se tiene *Cell Mode* pues la interfaz es de ATM.

En forma resumida, en el *Frame Mode* se tiene dos modos de asignación de etiquetas: el modo "*per platfform allocation*" en el que la etiqueta se genera en cada enrutador y es única en él, y el modo "*unsolicited*" en el que se genera la etiqueta aunque no haya tráfico. La generación de etiquetas es por ruta, no por interfaz. Por otro lado, en el *Cell Mode* también tenemos dos modos de asignación de etiquetas: el modo "*per interface allocation*" y el modo "*on demand*". En el modo *"per interface allocation",* la asignación de etiquetas se hace para cada puerto ATM. En el modo "*on demand*" se tienen dos divisiones. Una de ellas es "*ordered mode*", en la que se manda investigar la etiqueta de la red destino desde el primer conmutador ATM de MPLS. Hasta que se conocen todas las etiquetas, se toma el paquete y se envía, lo cual requiere mucho tiempo. La otra división es "*independent mode*" en la que el primer conmutador ATM de MPLS le da una etiqueta al paquete que llega y se transmite hasta llegar a su destino, pero no se puede distinguir el origen del paquete.

III.1.5.4. Tipos de mensajes LDP

Existen cuatro tipos básicos de mensajes: los mensajes DISCOVERY, los mensajes ADJACENCY (los cuales se involucran en la inicialización y cierres de sesión entre LSRs y en los mensajes *Keepalive*), los mensajes de anuncio de etiquetas (los cuales también se emplean para solicitudes, retiros y liberación de etiquetas, no sólo en su anuncio) y los mensajes NOTIFICATION (empleados para informar mensajes de error). Dentro de estos rubros, los mensajes que comúnmente se utilizan son:

Mensajes INITIALIZATION:

Son los mensajes que se envían al inicio de una sesión LDP para permitir que dos LSRs acuerden los parámetros y opciones de la sesión, los cuales incluyen los valores de los contadores KEEPALIVE y el rango de etiquetas a ser usadas en el enlace entre los dos LSRs. Ambos enrutadores pueden enviar los mensajes INITIALIZATION y, Al recibirlos, deberán responder con mensajes KEEPALIVE si los parámetros son aceptables. Si algún parámetro no fuera aceptable, el LSR responde con una notificación de error y la inicialización de la sesión se da por terminada.

Mensajes KEEPALIVE:

Son mensajes enviados periódicamente en ausencia de algún otro mensaje con el fin de asegurar que cada enrutador conoce a sus demás vecinos y que se encuentran trabajando correctamente. En la ausencia de un mensaje KEEPALIVE o algún otro mensaje LDP en el intervalo apropiado de tiempo, un LSR concluye que su vecino, o su conexión a él, se ha caído y termina la sesión.

Mensajes MAPEO DE ETIQUETA:

Estos mensajes son el corazón de la distribución de etiquetas, pues son usados para anunciar una relación entre una dirección IP y una etiqueta (Estos es, define una FEC).

Mensajes RETIRO DE ETIQUETA:

Son usados para revocar el anuncio previo de una etiqueta o de una FEC (relación dirección IP-etiqueta o *binding*). Las razones para retirar una etiqueta o un *binding* pueden ser el cambio del prefijo de una dirección IP en la tabla de enrutamiento de los LSRs debido a algún cambio en el enrutamiento o el cambio en la configuración del LSR que cause el cese de conmutación de etiquetas de los paquetes en esa FEC.

Mensajes LIBERACIÓN DE ETIQUETA:

Son usados por un LSR que previamente recibió un mapeo de etiqueta y que ahora ya no tiene necesidad de mapearla. Esto sucede comúnmente cuando el LSR que libera la etiqueta encuentra que el siguiente salto para cierta FEC no es la que anuncia dicho LSR.

Mensajes SOLICITUD DE ETIQUETA:

Son mensajes que se emplean únicamente en el modo de asignación de etiquetas *downstream* bajo demanda. Como su nombre lo indica, se envían estos mensajes para solicitar una etiqueta a un paquete para transmitirse en la red MPLS.

Mensajes SOLICITUD DE ABORTO:

Estos mensajes se emplean cuando se necesita revocar la solicitud de una etiqueta antes de que ésta sea completada, debido por ejemplo, a que el siguiente salto para la FEC en cuestión cambie.

III.1.6. Funcionamiento

III.1.6.1. Construcción de tablas de enrutamiento en MPLS

En el enrutamiento convencional se tiene una tabla global en cada enrutador de la red, en las cuales se tiene tres direcciones IP: *Source Address* (o dirección IP origen), *Destination Address* (o dirección IP destino) y el *Next Hop* (o dirección IP del siguiente salto), pero básicamente se consideran las últimas direcciones IP para el proceso de enrutamiento "básico".

Source Address	Destination Address	Next Hop	Output Interface	# Hops	⋯
Dirección IP . . .	*Dirección IP* . . .	*Dirección IP* . . .	*puerto* . . .	*Número de saltos*	⋯ . . .

Figura III.1.6.1.1. Tabla global de enrutamiento

La razón por la cual se utilizan principalmente la dirección destino y el siguiente salto en el enrutamiento es porque es primordial saber por dónde va a salir un paquete que llega al enrutador (*Next Hop*) según sea su destino (*Destination Address*). El mecanismo de enrutamiento se hace a nivel de capa 3, pero debido a que la

búsqueda de dichas direcciones en una tabla global es muy lenta y requiere demasiado tiempo de procesamiento, los fabricantes de cajas han buscado otros mecanismos que simplifiquen este proceso. Uno de estos mecanismos (y el que se usa en MPLS) es *el Cisco Express Forwarding* o CEF, el cual, como su nombre lo indica, es propietario de Cisco.

En CEF se hace balanceo por carga, ya sea por sesión o por la relación origen-destino, e inclusive por paquete (sin impactar al CPU de las cajas). La asignación de las interfaces a los pares origen-destino es aleatorio en cada sesión, es decir, al inicio de cada sesión se asigna aleatoriamente la interfaz a cada par. Los equipos que manejan CEF se pueden configurar para hacer el balanceo de carga, ya sea por paquete o por sesión.

Con CEF, el primer paquete que llega al enrutador es analizado para determinar su destino, y con ello se hace una búsqueda en la tabla global de enrutamiento del siguiente salto correspondiente y de la interfaz por la que saldrá del enrutador. Con esos datos, se crea una "entrada" para ir construyendo una tabla en memoria caché. Los demás paquetes que lleguen al enrutador también son analizados y si algunos de ellos van dirigidos a una dirección IP que anteriormente se había analizado, ya no se busca en la tabla global el siguiente salto y la interfaz de salida que le corresponden, sino que se busca en alguna de las entradas de la tabla de la memoria caché. Esta tabla es como una tabla de flujos que se construye conforme van llegando los paquetes y así, cuando llega un paquete al enrutador, primero se busca en la tabla de flujos su dirección IP destino. Si se encuentra, inmediatamente se le asocia el *Next Hop* y la interfaz de salida que le corresponden, pero en el caso de que no se encuentre una entrada para su dirección IP destino, se realiza la búsqueda en la tabla global.

Con lo anterior, podemos afirmar que CEF permite un enrutamiento basado en la memoria caché para construir tablas más pequeñas que faciliten la búsqueda de los datos necesarios para el enrutamiento. En MPLS se toma la tabla construida gracias a CEF y la llama FIB (*Forwarding Information Base*), la cual básicamente tiene los siguientes campos:

Destination Address	Next Hop	Output Interface
Dirección IP	*Dirección IP*	*puerto*
.	.	.
.	.	.
.	.	.

Figura III.1.6.1.2. Tabla FIB

Un enrutador *Edge LSR* asigna a cada una de las entradas de la FIB una etiqueta. Dicha etiqueta es un valor numérico que se genera aleatoriamente en el enrutador y tiene un rango de 2^4 a 2^{20}, con algunos números reservados. Una vez generadas las etiquetas, éstas se asocian a la FIB, formando una nueva tabla llamada LIB (*Label Information Base*). La asociación de etiquetas a las direcciones IP destino se lleva a cabo a través de LDP y con ello se forma la LIB, la cual tiene los siguientes campos:

Destination Address	Local Label	Next Hop Label
Dirección IP	*Etiqueta*	*Etiqueta*
.	.	.
.	.	.
.	.	.

Figura III.1.6.1.3. Tabla LIB

Con la LIB podemos observar cómo los datos de la tabla global de enrutamiento se traducen a etiquetas. En la LIB ya se tiene una etiqueta local, asignada según la dirección IP destino. A esta etiqueta local se le llama etiqueta de entrada u *outgoing tag* (porque al llegar el paquete, esta etiqueta será retirada), mientras que a la etiqueta del *Next Hop* se le llama etiqueta de salida o *local tag* (porque es la etiqueta con la que va a viajar el paquete en el segmento de red directamente conectado), pero cada una de ellas tiene significado local. Las etiquetas en MPLS son similares a los DLCIs en Frame Relay.

Es importante señalar que en la asignación de etiquetas también se puede considerar la dirección IP origen con el fin de hacer una distinción del tráfico. Con esta opción, es posible asignar etiquetas diferentes a paquetes que tengan la misma dirección IP destino, pero diferente origen, resultando seguramente en trayectorias distintas para los paquetes según su origen (aunque sea el mismo destino). Esta ventaja permite establecer prioridades de tráfico y marcado de paquetes.

La LIB es una tabla de etiquetas a partir de la cual se construye la LFIB (*Label Forwarding Information Base)*. Es equivalente a la FIB pues maneja los mismos campos, pero en la LFIB ya no se manejan direcciones IP como en la FIB, sino etiquetas. En la LFIB se asocian las etiquetas de entrada y de salida con la interfaz por la que saldrá el paquete:

Local Label	Next Hop Label	Output Interface
Etiqueta	*Etiqueta*	*puerto*

Figura III.1.6.1.4. Tabla LFIB

Las LFIBS son las tablas que se intercambian los vecinos de MPLS pues en ese ambiente sólo se necesita conocer las etiquetas y sus interfaces de salida. Esto brinda una seguridad inherente a la red, pues si se conecta una *sniffer* a la red, sólo vería etiquetas, es decir, podría conocer números con valor local a cada enrutador y no las direcciones IP destino y origen.

Cuando un paquete llega a una red MPLS, éste se topa primero con un *Edge LSR*. La tabla con la que inicia la conmutación de etiquetas el *Edge LSR* es la LIB y de ahí mapea la entrada a la LFIB. Sin embargo, si en la LIB no existe la entrada que corresponde al paquete que llegó al enrutador, se debe regresar a IP a hacer la búsqueda en la tabla global de enrutamiento para obtener los datos correspondientes a dicho paquete. A partir de esa búsqueda y por medio de CEF, se crea una entrada en la FIB para mapear los datos extraídos a una

entrada en la LIB y que así se le asignen etiquetas de entrada y de salida. Finalmente, la LFIB toma la entrada recién creada en la LIB y la asocia al puerto de salida que la FIB indica como su correspondiente, con lo que se tendría también en la LFIB una entrada para dicho paquete y para los demás paquetes que lleguen a la red MPLS con la misma dirección destino.

En la arquitectura MPLS, un enrutador *Edge LSR* está dividido en dos planos, como ya se mencionó. En el Plano de Control se tienen los procedimientos de enrutamiento para definir la ruta que va a seguir un paquete desde su origen hasta su destino. Estos procedimientos son los protocolos IGP y por ello, a este plano le corresponde la tabla global de enrutamiento. Además, se realiza la generación, asignación, distribución, conmutación y transporte de etiquetas. Por otro lado, el Plano de Datos se encarga de la transmisión efectiva sobre el medio físico de los paquetes en MPLS. Este plano controla el medio físico en el que se colocan los paquetes y las tablas FIB, LIB y LFIB pertenecen a este plano.

III.1.6.2. imposición de etiquetas en la frontera de la red

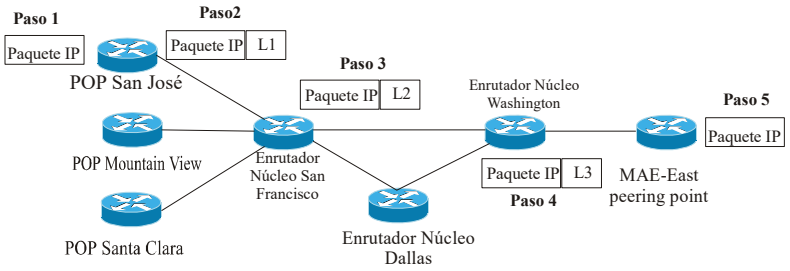
La imposición de etiquetas se ha descrito como el acto de poner en el paquete una etiqueta para que pueda entrar al dominio MPLS. Esta función se realiza en la frontera, lo cual significa que el paquete es etiquetado antes de ser enviado al dominio MPLS. Para ejecutar esta función, el *Edge-LSR* necesita entender en dónde está el encabezado del paquete y qué etiqueta o pila de etiquetas tiene. En el envío tradicional IP de Capa 3, cada salto en la red ejecuta una búsqueda dentro de la tabla de envío IP convencional para encontrar la dirección IP destino contenida en el encabezado del paquete de Capa 3. Se selecciona la dirección IP del siguiente salto en cada interacción de la búsqueda y eventualmente se manda el paquete a la interfaz correspondiente hasta su destino final.

La elección del siguiente salto para el paquete IP es una combinación de dos funciones. La primera función fragmenta todo el envío en paquetes más adecuados y les asigna un prefijo IP destino. La segunda función mapea cada prefijo IP destino hacia la dirección del siguiente salto IP. Esto significa que cada destino en la

red es alcanzable por un camino en el sentido del flujo del tráfico desde un dispositivo de ingreso hacia un dispositivo de egreso destino (pueden estar disponibles varias rutas si el balanceo de carga es ejecutado usando costos iguales o costos diferentes en las rutas, como en algunos protocolos IGP, tal como EIGRP)

En la arquitectura MPLS, el resultado de la primera función es conocida como Equivalencia de Clases de Envío (*Forwarding Equivalence Classes* o FEC). Una FEC puede visualizarse como un grupo de paquetes IP con una etiqueta asociada, que son enviados de la misma manera a través del mismo camino, con el mismo tratamiento de envío. La FEC es codificada como un identificador de longitud fija.

Cuando el paquete es enviado hacia su siguiente salto, la etiqueta es antepuesta en el paquete IP y, entonces, el próximo dispositivo en la ruta del paquete puede enviarlo de acuerdo a una etiqueta codificada, en vez de hacer el análisis del encabezado de Capa 3. La figura III.1.6.1 ilustra todo el proceso de la imposición de etiquetas y su envío.



Paso 1.- El paquete Ip llega al enrutador de San José.
Paso 2.-El enrutador San José hace la búsqueda de Capa 3, le pone una etiqueta y se envía el paquete hacia el enrutador San Francisco.
Paso 3.- El enrutador San Francisco hace un reconocimiento de la etiqueta, la intercambia y envía el paquete hacia el enrutador en Washington.
Paso 4.- El enrutador en Washington hace un reconocimiento de etiqueta, la intercambia y envía el paquete hacia el enrutador MAE-East
Paso 5.- El enrutador MAE-East hace el reconocimiento de etiqueta, le quita la etiqueta, hace una búsqueda de Capa 3 y envía el paquete hacia el próximo enrutador externo.

Figura III.1.6.1. Imposición y envío de etiquetas MPLS.

III.1.6.2. Envío de paquetes MPLS y caminos etiquetados conmutados

Cada paquete entra a la red MPLS en el LSR de ingreso y sale por el LSR de egreso. Este mecanismo crea lo que se conoce como Camino Etiquetado Conmutado (*Label Switched Path* o LSP), el cual esencialmente describe el conjunto de LSRs a través de los cuales el paquete transita hasta un LSR de egreso para un FEC particular o, en otras palabras, es la ruta que sigue un paquete etiquetado a través de la red MPLS. El LSP es unidireccional, lo cual significa que se usa un LSP diferente para regresar el tráfico desde una FEC particular.

La creación de los LSP es orientada a conexión, ya que el camino es dado de alta antes de que el tráfico fluya, es decir, esta conexión está basada en una topología de información, lo que significa que el camino es creado indiferentemente de que cualquier tipo de tráfico actual sea requerido para fluir a lo largo de una ruta hacia una FEC particular.

Cuando un paquete atraviesa la red MPLS, cada LSR intercambia la etiqueta de entrada con la etiqueta de salida, parecido al mecanismo utilizado en ATM en donde los VPI/VCI son intercambiados a diferentes VPI/VCI cuando existe una conmutación en ATM. Este proceso continúa hasta que se sabe que el LSR es de egreso (*Edge LSR*).

Cada LSR mantiene dos tablas, en las cuales se mantiene la información que es relevante en los componentes de envío de MPLS. La primera es la LIB y mantiene todas las etiquetas asignadas por el LSR y hace el mapeo de las etiquetas recibidas desde sus vecinos. Este mapeo de etiquetas se lleva a cabo a través del uso de protocolos de distribución de etiquetas. Múltiples vecinos pueden enviar etiquetas con el mismo prefijo de IP, pero éste no debe ser el próximo salto IP que esté activo (o en uso) en ese momento en la tabla de enrutamiento para el destino. No todas las etiquetas en LIB necesitan ser usadas para el envío de paquetes. La segunda tabla, la LFIB se emplea durante un envío de paquetes y sólo retiene las etiquetas que están en uso por los componentes MPLS.

Usando los términos anteriores, la arquitectura del Edge-LSR en la figura III.1.3 puede ser re-dibujada como se muestra en la Figura III.1.6.2.

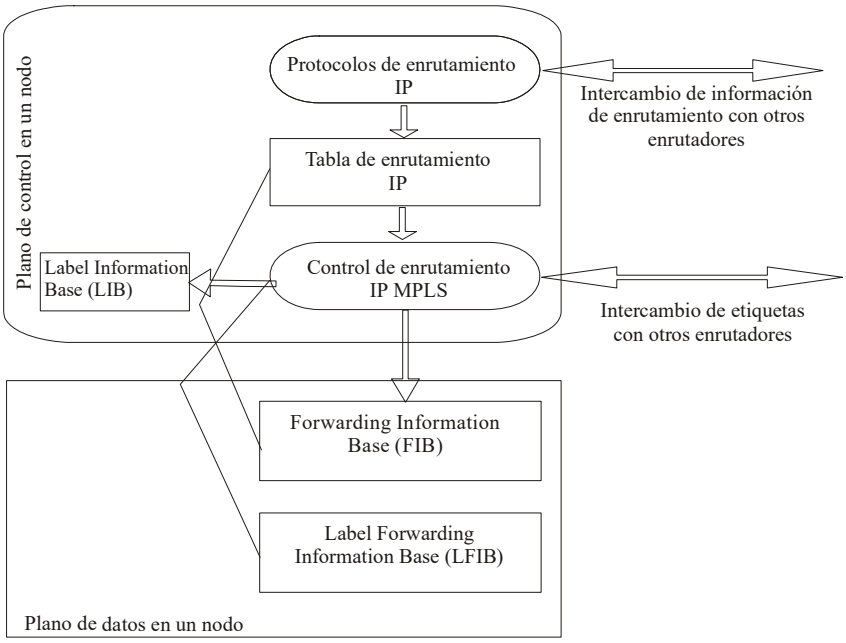


Figura III.1.6.2. Arquitectura del Edge LSR

III.1.7. Aplicaciones de MPLS

La arquitectura MPLS no sólo implementa una integración de los enrutadores tradicionales y conmutadores ATM en un *backbone* IP unificado (Arquitectura IP + ATM). El verdadero poder de MPLS es poner otras aplicaciones, en donde sea posible, desde Ingeniería de Tráfico hasta VPN (Virtual Private Network). Todas las aplicaciones MPLS usan la funcionalidad del plano de control similar al plano de control de enrutamiento IP. La Figura III.1.7. muestra la interacción entre estas aplicaciones y la matriz de conmutación de etiquetas.

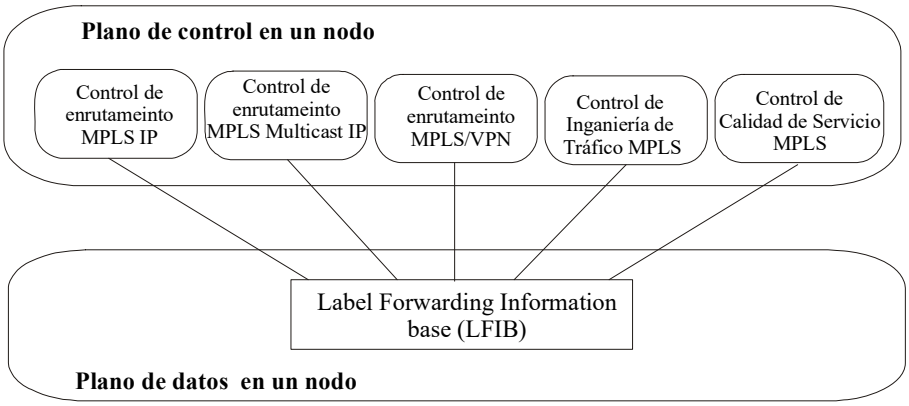


Figura III.1.7. Varias aplicaciones de MPLS y sus interacciones.

Cada aplicación MPLS tiene los mismos componentes que las aplicaciones del enrutamiento IP:

- Una base de datos definida por la tabla de enrutamiento IP en una aplicación de enrutamiento IP.
- Los protocolos de control que intercambian los contenidos de las tablas entre los LSRs (protocolos enrutamiento IP o enrutamiento estático IP).
- Proceso de control que ejecuta el mapeo de etiquetas a las FEC y un protocolo que intercambia etiquetas entre LSRs (TDP o LDP en una aplicación de enrutamiento IP).
- Opcionalmente, una base de datos interna que mapea etiquetas con LIB en una aplicación de enrutamiento IP.

Cada aplicación usa su propio conjunto de protocolos para intercambiar tablas FEC o hacer el mapeo de etiquetas con el FEC entre nodos. La tabla III.1.7. resume los protocolos y las estructuras de los datos.

Tabla III.1.7. Protocolos de control usados en varias aplicaciones MPLS

Aplicación	**Tabla asociada**	**Protocolo de control usado para construir la tabla asociada**	**Protocolo de Control usado para intercambiar etiquetas mapeadas al FEC**
Enrutamiento IP	Tabla de enrutamiento IP	Cualquier Protocolo IP	TDP o LDP
Enrutamiento VPN	Tabla de enrutamiento VPN	La mayoría de los protocolos de enrutamiento entre el proveedor de servicios y el cliente. BGP Multiprotocolo dentro de la red del proveedor de servicios	BGP Multiprotocolo
Ingeniería de tráfico	Definiciones de túneles MPLS	Definición manual de interfaces, extensiones IS-IS o OSPF	RSVP o CR-LDP
Calidad de Servicio MPLS	Tabla de enrutamiento IP	Protocolos de enrutamiento IP	Extensiones LDP o TDP

III.2.1. Introducción a BGP

Los protocolos de enrutamiento EGPs (*Exterior Gateway Protocol*) fueron introducidos debido a que los protocolos IGPs (*Interior Gateway Protocol*) no escalan en redes que van más allá del nivel de la empresa. Los IGPs no fueron diseñados para propósitos de tareas globales de *internetworking* pues no tenían la necesidad de conectarse a diferentes administraciones de empresas que son técnica y políticamente independientes una de la otra.

Los protocolos de enrutamiento exterior se crearon para controlar la expansión de las tablas de enrutamiento y así proporcionar una vista más estructurada de Internet al dividir a los dominios de enrutamiento en administraciones separadas llamadas Sistemas Autónomos o AS (*Autonomous Systems*), las cuales, cada una de ellas, tienen sus políticas independientes de enrutamiento.

Actualemente BGP4 es el estándar de facto para el enrutamiento en Internet. Se trata de un protocolo EGP que le proporciona a Internet una topología controlada y libre de *loops*.

III.2.2. Conceptos básicos

Un sistema autónomo (AS) es un conjunto de enrutadores con las mismas políticas de enrutamiento bajo una misma administración técnica. El AS puede ser una colección de IGPs trabajando juntos para proporcionar enrutamiento interior. Para el mundo exterior, el AS completo es visto como una sola entidad. Cada AS tiene su número de identificación, el cual es asignado por un proveedor o por un Registro de Internet. La información de enrutamiento es intercambiada entre ASs por medio de un protocolo de enrutamiento exterior, tal como BGP, como se ilustra en la siguiente figura:

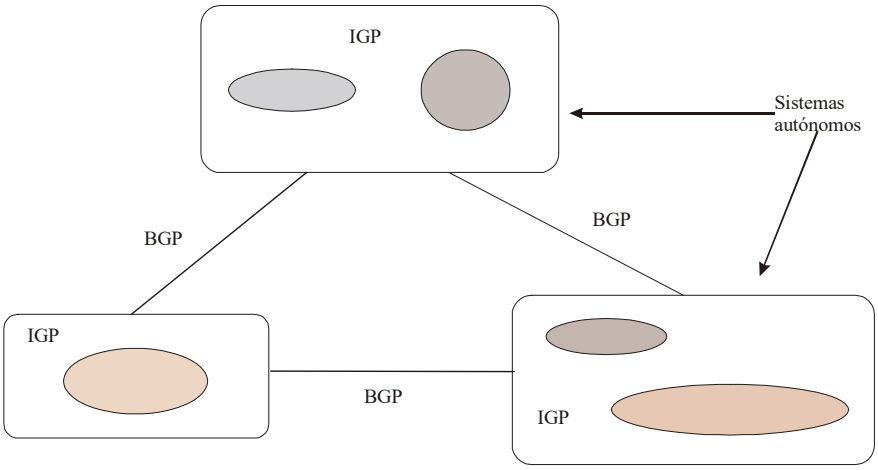


Figura III.2.2. Ejemplo de sistemas autónomos

Al segregar la red mundial en diferentes administraciones, se puede ver una red enorme dividida en redes más pequeñas y manejables. Dichas redes pequeñas, llamadas ASs, tienen su propio conjunto de reglas y políticas que las distinguen de las demás ASs y cada una de ellas ejecuta su propio IGP, independiente del IGP en otro AS.

Un AS se considera *stub* cuando alcanza redes externas a su dominio por medio de un solo punto de salida. Este tipo de AS no tiene que aprender rutas de Internet de su proveedor de servicios, pues ya que hay un camino único de salida, todo el tráfico al exterior se va por esa ruta *default*.

En el caso de que el AS no tenga una ruta *default* para el tráfico exterior, sino muchos puntos de salida, existen diferentes métodos para anunciar y aprender rutas. Uno de los métodos por los cuales el ISP. Uno de los métodos por los que un ISP puede aprender y anunciar las rutas de un cliente es por medio de BGP. En un AS *stub* es difícil obtener el número del AS registrado ante InterNIC debido a que las políticas de enrutamiento del cliente son una extensión del proveedor. Por tal motivo, el proveedor le da al cliente un número de AS de su conjunto privado de número de ASs que va del 65412 al 65535.

III.2.3. Descripción

BGP (*Border Gateway Protocol*) ha tenido varias fases y mejoras desde su versión inicial BGP1 en 1989 hasta su versión actual, BGP4 cuyo desarrollo inició en 1993. BGP4 es la primera versión de BGP que soporta sumarización CIDR y subneteo.

BGP no impone restricciones en la topología de Internet, pues supone que el enrutamiento en un sistema autónomo se realiza por medio de un protocolo de enrutamiento interior, es decir, adentro del sistema autónomo. BGP construye un bosquejo del sistema autónomo de acuerdo a la información intercambiada entre vecinos BGP. Dicho bosquejo generalmente es referido como "árbol". Así, podemos pensar en Internet como todo un árbol de Ass, con cada AS identificado con un número de AS. Las conexiones entre dos ASs forman una trayectoria y la colección de información referente a las trayectorias forman una ruta para alcanzar un destino específico. Además, BGP garantiza el enrutamiento entre dominios libre de *loops*.

BGP es un protocolo *path vector* usado para transportar información de enrutamiento entre sistemas autónomos. El término *path vector* viene del hecho de que la información de enrutamiento BGP transporta una secuencia de números de AS, los cuales indican la trayectoria que una ruta ha atravesado. BGP utiliza TCP como su protocolo de transporte (puerto 179), lo que asegura que toda la confiabilidad del transporte, como la retransmisión de paquetes, se lleva a cabo por TCP y no necesita que se implemente en BGP algún mecanismo que dé confiabilidad al transporte de datos.

Dos enrutadores BGP forman una conexión entre cada uno del protocolo de transporte. Estos enrutadores reciben el nombre de vecinos o *peers*. Los enrutadores vecinos intercambian múltiples mensajes para abrir y confirmar los parámetros de una conexión, tales como la versión de BGP que es está ejecutando entre ellos (por ejemplo, versión 4 para BGP4). En caso de que exista algún error entre los vecinos, se envían notificaciones de error y la conexión entre los *peers* no se establece.

En los enrutadores BGP se anuncian las rutas por medio de los mensajes UPDATEs. Este tipo de mensajes contiene, entre otras cosas, una lista de pares <tamaño, prefijo> que indican la lista de destinos alcanzables por cada sistema. El mensaje UPDATE también contiene los atributos de la trayectoria, los cuales incluyen información del grado de preferencia para cada ruta en particular.

En el caso de que ocurran cambios de información, como una ruta que se hace inalcanzable o que surja una trayectoria mejor, BGP informa a sus vecinos retirando las rutas inválidas (rutas que no están disponibles para usarse) e inyecta nueva información de enrutamiento como parte de un mensaje UPDATE. Si no ocurren cambios, los enrutadores sólo intercambian paquetes KEEPALIVE.

Los mensajes KEEPALIVE se envían periódicamente entre vecinos BGP para asegurar que la conexión se mantiene arriba. Los paquetes KEEPALIVE (19 bytes cada uno) no deberían causar ningún inconveniente en el CPU del enrutador o en el ancho de banda del enlace, pues consume un mínimo ancho de banda de aproximadamente 2.5bits por segundo cada 60 segundos.

Aunque BGP es un protocolo exterior, puede ser usado adentro de un sistema autónomo como un túnel para intercambiar actualizaciones BGP. Las conexiones BGP en el interior de un Sistema Autónomo se llaman IBGP (*Internal BGP*), mientras que las conexiones BGP entre AS, se llaman EBGP (*External BGP*). Los enrutadores que ejecutan IBGP se conocen como enrutadores de tránsito pues transportan el tráfico que transita a través del AS. De la misma manera, los enrutadores que ejectuan EBGP con otro sistema autónomo generalmente se conocen como enrutadores de frontera.

III.2.4. Formato del encabezado

El formato del mensaje BGP es un campo Marcador de 16 bytes, seguido por un campo llamado Tamaño de 2 bytes y un campo nombrado Tipo, de un byte. La figura III.2.4 ilustra el formato básico del encabezado de un mensaje BGP:

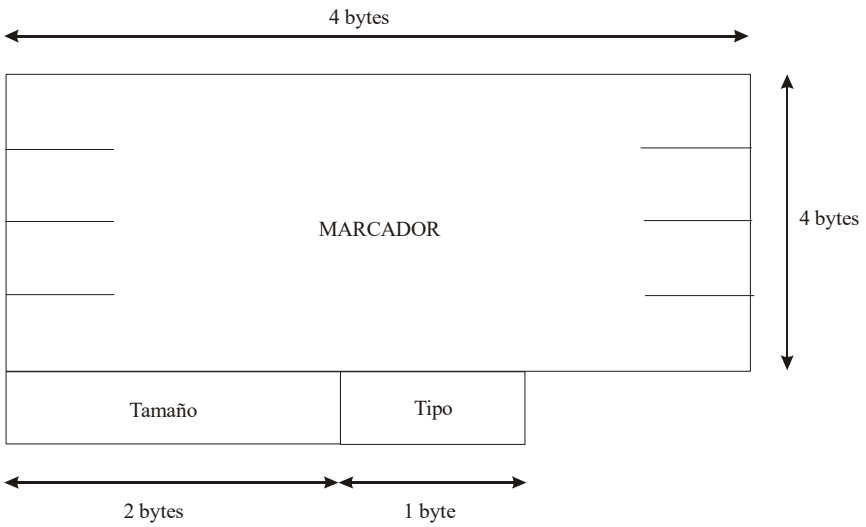


Figura III.2.4. Encabezado de BGP

Puede haber o no datos en la porción que sigue al encabezado, dependiendo del tipo de mensaje. Los mensajes KEEPLALIVE, por ejemplo, consisten únicamente del encabezado del mensaje y no lleva datos.

El campo Marcador es usado para verificar los mensajes BGP o para detectar la pérdida de sincronización entre dos vecinos BGP. Este campo puede tener dos formatos:

- Si el tipo de mensaje es OPEN o si el mensaje OPEN no tiene información de verificación, el campo Marcador debe llenarse con puros unos.
- De otra manera, el campo Marcador debe ser calculado con el mecanismo de verificación empleado.

El campo Tamaño indica la longitud total del mensaje BGP, incluyendo al encabezado. El mensaje BGP más pequeño no puede ser menor a 19 bytes (16 de Marcador + 2 de Tamaño + 1 de Tipo) y el más grande no es mayor a 4096 bytes.

El campo Tipo indica el tipo de mensaje BGP con las siguientes posibilidades:

- OPEN
- NOTIFICATION
- KEEPALIVE
- UPDATE

III.2.5. Tipos de mensajes

III.2.5.1. Mensaje OPEN

El formato del mensaje se muestra en la siguiente figura:

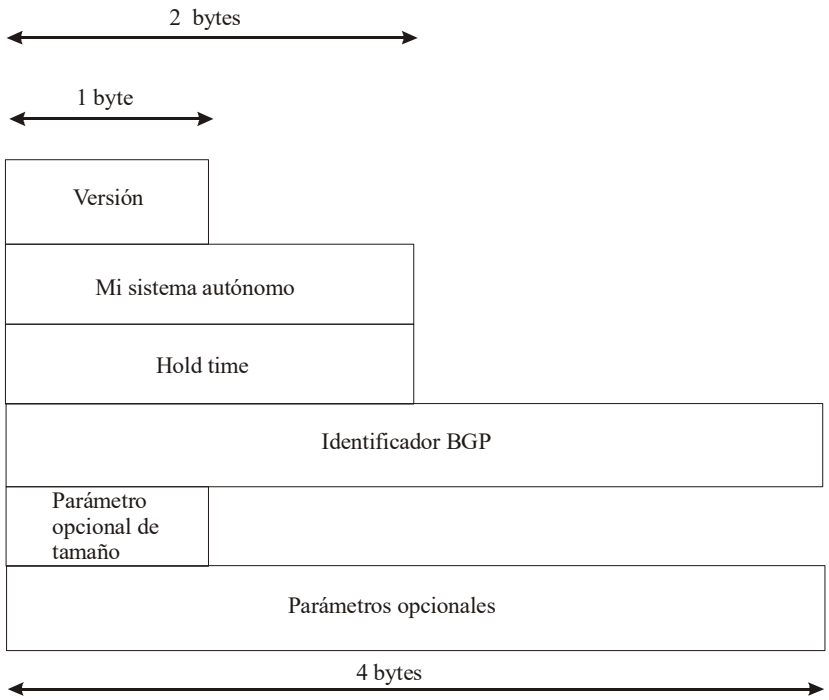


Figura III.2.5.1. Mensaje OPEN

La descripción de los campos se detalla a continuación:

- Versión: 1 byte. Entero sin signo que indica la versión del protocolo BGP. Durante la negociación entre vecinos, los *peers* BGP acuerdan el número de la versión. Los vecinos BGP siempre negocian la versión más reciente que tengan en común.
- Mi Sistema Autónomo: 2 bytes. Campo que indica el número de AS del enrutador BGP.
- Tiempo de Espera (*Hold Time*): 2 bytes. Entero sin signo que indica la cantidad máxima de tiempo en segundos que debe pasar entre la recepción de mensajes sucesivos KEEPALIVE o UPDATE. El contador que lleva la cuenta del tiempo incrementa de cero hasta el valor del *Hold Time* y en cuanto se recibe un mensaje KEEPALIVE o UPDATE, se resetea a cero. Si el tiempo de espera de un vecino en particular se excede, tal vecino debe considerarse muerto.

  El enrutador BGP negocia con sus vecinos el valor del *Hold Time*, tomando el valor más bajo que tenga alguno de ellos. Si dicho tiempo se acuerda en cero, se considera que la conexión siempre está arriba. El mínimo valor recomendado para el *Hold Time* es de tres segundos.
- Identificador BGP: 4 bytes. Entero sin signo que indica el identificador o ID del transmisor. El identificador generalmente es la dirección *loopback* del enrutador.

III.2.5.2. Mensaje NOTIFICATION

Un mensaje NOTIFICATION siempre se envía cuando se detecta un error, después del cual se cierra la conexión entre los vecinos. Los administradores de la red tendrán que evaluar el mensaje NOTIFICATION para determinar la naturaleza específica de los errores que emerjan en el protocolo de enrutamiento. En la figura III.2.5.2 se ilustra el formato general del mensaje:

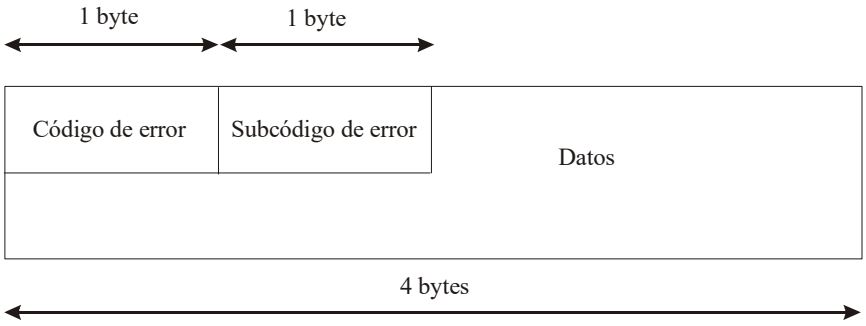


Figura III.2.5.2. Mensaje NOTIFICATION

El mensaje NOTIFICATION se compone de un byte para el Código de Error, un byte para el Subcódigo de Error y de un campo variable para Datos. El Código de Error indica el tipo de notificación y el Subcódigo de Error proporciona información más específica acerca de la naturaleza del error. En la Tabla III.2.5.2 se enlistan los posibles errores y sus subcódigos.

Tabla III.2.5.2

Código de Error		***Subcódigos de Error***	
1	Error en el encabezado del mensaje	1	Conexión no sincronizada
		2	Error en el tamaño del mensaje
		3	Error en el tipo del mensaje
2	Error en el mensaje OPEN	1	Número de versión no soportado
		2	Error en el vecino AS
		3	Error en el identificador BGP
		4	Parámetro opcional no soportado
		5	Falla en la verificación
		6	Tiempo de Espera no válido
3	Error en el mensaje UPDATE	1	Lista de atributos mal formada
		2	Atributo **bien-conocido** desconocido
		3	Falta atributo **bien-conocido**
		4	Error en las banderas del atributo

		5	Error en el tamaño del atributo
		6	Atributo Origen inválido
		7	Loop en el enrutamiento AS
		8	Atributo *Next-Hop* inválido
		9	Error en un atributo opcional
		10	Campo de Red inválido
		11	Atributo AS_*path* mal formado
4	Tiempo de Espera expirado	No aplica	
5	Error en la Máquina de Estado Finito	No aplica	
6	Cierre de la sesión (por errores fatales)	No aplica	

III.2.5.3. Mensaje KEEPALIVE

Los mensajes KEEPALIVE son mensajes periódicos intercambiados entre vecinos para determinar si siguen alcanzables. Como ya se había mencionado, el tiempo de espera (o *hold time*) es la cantidad máxima de tiempo que debe pasar entre la recepción de mensajes KEEPALIVE o UPDATES sucesivos. Los mensajes KEEPALIVE son enviados a una tasa que garantice que el tiempo de espera no expirará antes de recibir estos mensajes, pues si no, la sesión se terminará. Una tasa recomendada es la tercera parte del intervalo del tiempo de espera. Un mensaje KEEPALIVE tiene los 19 bytes mínimos del encabezado.

III.2.5.4. Mensaje UPDATE

Las actualizaciones de enrutamiento contienen toda la información necesaria que usa BGP para construir un bosquejo del Internet. A continuación se enlistan los bloques básicos de un mensaje UPDATE:

- NLRI (*Network Layer Reachability Information*)
- Rutas inalcanzables

- Atributos de las rutas

En la figura III.2.5.4.1 se ilustran los componentes en el contexto del formato de un mensaje UPDATE.

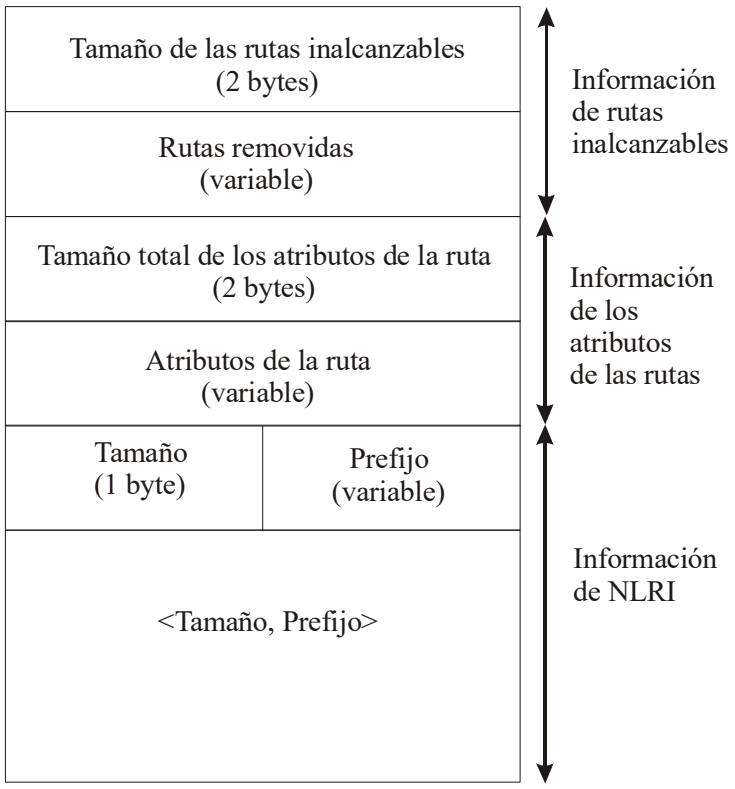


Figura III.2.5.4.1. Mensaje UPDATE

- NLRI:

El NLRI es una indicación, en la forma de un prefijo IP, de las redes que son anunciadas. Como BGP4 soporta CIDR y subneteo, el NLRI es el mecanismo con el cual se logra eso. El NLRI es la parte de las actualizaciones de enrutamiento BGP que enlista el conjunto de destinos a los cuales BGP está tratando de informar sus otros vecinos BGP. El NLRI consiste de múltiples instancias de un par <tamaño, prefijo>, donde

tamaño es el número de bits de máscara que un prefijo en particular tiene. Por ejemplo, en el NLRI <19, 198.24.160.0>, el prefijo es 198.24.160.0 y el tamaño es una máscara de 19 bits, siendo más conocido si se escribe 198.24.160.0/19.

- Rutas inalcanzables:

Esta parte es una lista de rutas que se ha vuelto inalcanzables, o en términos de BGP, retiradas. Las rutas inalcanzables no están más en servicio y necesitan ser removidas de las tablas de enrutamiento BGP. Las rutas retiradas tienen el mismo formato que NLRI: <tamaño, prefijo>. De la misma manera, <18,192.213.134.0> indica que se ha retirado la ruta 192.213.134.0/18.

Un mensaje UPDATE que no tiene NLRI o atributos de las rutas se usa para anunciar únicamente las rutas removidas de servicio.

- Atributos:

Los atributos BGP son un conjunto de parámetros usados para mantener el rastro de la información específica de una ruta. Estos parámetros son usados en el proceso de filtrado y decisión de rutas de BGP. Cada mensaje UPDATE tiene una longitud y una secuencia variable de los atributos. Un atributo tiene la forma <Tipo de Atributo, Tamaño del Atributo, Valor del Atributo>. El Tipo de Atributo es un campo de dos bytes que consta de una bandera de atributos de un byte y un código del tipo de atributo de un byte. La figura III.2.5.4.2 ilustra la forma general del campo Tipo de Atributo:

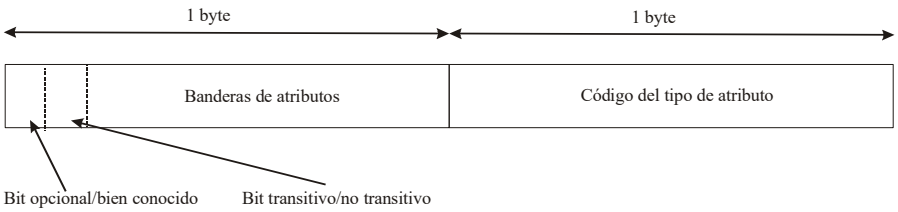


Figura III.2.5.4.2. Tipo de atributo

Existen cuatro categorías de atributos: los bien conocidos mandatorios, bien conocidos discrecionales, opcionales transitivos y los opcionales no transitivos. Estas cuatro categorías están descritas por los dos primeros bits del campo Banderas del campo Tipo de Atributo. El primer bit del campo Banderas indica si el atributo es opcional (1) o bien conocido (0). El segundo bit indica si el atributo opcional es transitivo (1) o no transitivo (0). Los atributos bien conocidos siempre son transitivos, por lo que el segundo bit es siempre uno. El tercer bit indica si la información en el atributo opcional transitivo es parcial (1) o completo (0). El cuarto bit define si la longitud del atributo es de un byte (0) o dos bytes (1). Los otros cuatro bits del campo Banderas siempre son cero. Los atributos se describirán con mayor detalle más adelante, en la sección III.2.9.

III.2.6. Proceso de establecimiento de vecinos

La negociación de vecinos BGP se efectúa a través de diferentes etapas después de que la conexión es completamente establecida. Para ilustrar este proceso es común hacerlo en forma de una máquina de estado finito que resalta los principales eventos del proceso con una indicación de los mensajes enviados (OPEN, KEEPALIVE, NOTIFICATION) al vecino en la transición de un estado a otro.

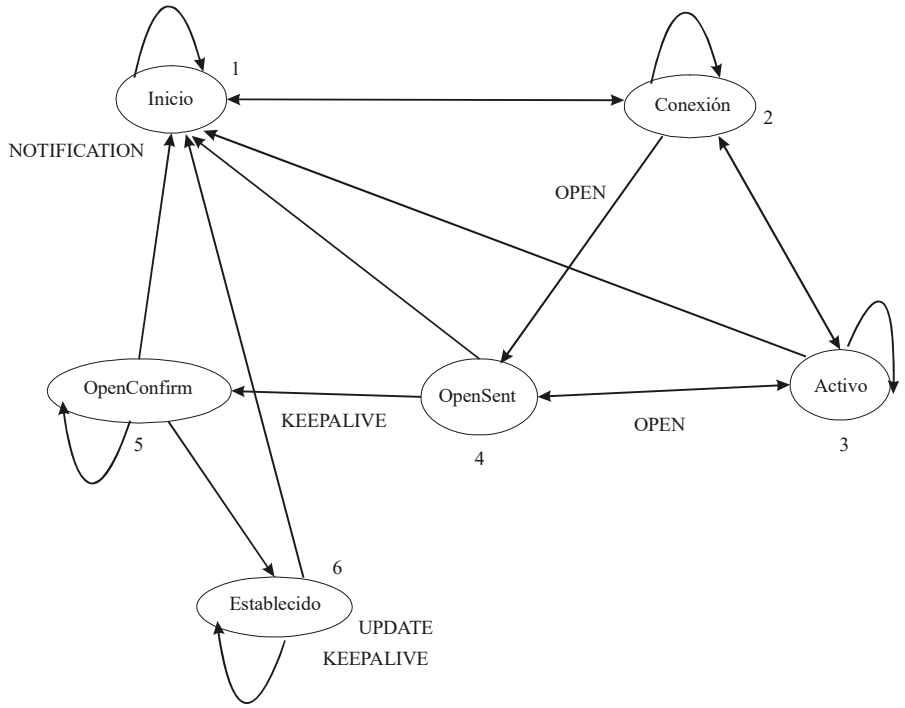


Figura III.2.6. Máquina de estado finito del proceso de establecimiento de vecinos

Inicio

Esta es la primera etapa de la conexión. BGP espera que suceda un evento para pasar a otro estado. Un evento de inicio es causado por un administrador al establecer una sesión BGP a través de la configuración del enrutador o reiniciando una sesión ya existente. Después del evento de inicio, BGP inicializa su conexión a TCP, un contador de tiempo de intento de conexión y sus recursos, además de que comienza a esperar la conexión iniciada por un vecino remoto. Una vez realizado esto, BGP transita al estado Conexión. En caso de errores, BGP continúa en el estado Inicio.

Conexión

BGP espera a que la conexión de TCP se complete. Si la conexión TCP es exitosa, se pasa al estado *OpenSent* (que es donde se envía el mensaje OPEN). Si la conexión TCP no es exitosa, se pasa al estado

Activo. Si el contador de tiempo de intento de conexión expira, el estado permanece en Conexión, se reinicia el contador y se debe iniciar una nueva conexión TCP. En caso de que suceda cualquier otro evento, se regresa al estado Inicio.

Activo

BGP trata de adquirir un vecino al inicializar una conexión en un protocolo de transporte. Si es exitosa tal conexión, se pasa al estado *OpenSent* y se envía un nuevo mensaje OPEN. Si el contador de tiempo de intento de conexión expira, BGP lo reinicia y regresa al estado Conexión. En caso de otros eventos (como una interrupción ocasionada por el sistema o un operador), se regresa al estado Inicio.

En general, un estado que oscila entre Conexión y Activo es una indicación de que algo anda mal con la conexión TCP, la cual no se está estableciendo. Esto se puede deber a muchas retransmisiones TCP o a la inhabilidad de un vecino para alcanzar la dirección IP de su vecino.

OpenSent

BGP espera un mensaje OPEN de su vecino. Este mensaje es verificado para detectar algún error. En caso de encontrarse alguno (como un número de versión incorrecto o un AS inaceptable), el sistema envía un mensaje NOTIFICATION y regresa al estado Inicio. Si no existen errores, BGP envía mensajes KEEPALIVE y reinicia el contador *Keepalive*. En esta etapa se negocia el tiempo de espera y se elige el valor menor.

En este estado, BGP reconoce (comparando su número de AS y el número AS de su vecino) si el vecino pertenece al mismo AS (IBGP) o a diferentes AS (EBGP). Cuando se detecta una desconexión TCP, se regresa al estado Activo. En caso de que ocurran otros errores (como la expiración del contador del tiempo de espera), BGP envía un mensaje NOTIFICATION con el código correspondiente del error y regresa al estado Inicio. Además, en respuesta a una interrupción iniciada por el sistema o por un operador, también se regresa al estado Inicio.

OpenConfirm

BGP espera por un mensaje KEEPALIVE o NOTIFICATION. Si se recibe un KEEPALIVE, se pasa al estado Establecido y se completa la negociación con el vecino. Si el sistema recibe un mensaje KEEPALIVE, se reinicia el contador del tiempo de espera. Si se recibe un mensaje NOTIFICATION, se regresa al estado Inicio.

El sistema envía mensajes KEEPALIVE periódicamente a la tasa que indica el contador *Keepalive*. En caso de que se reciba una notificación de una desconexión TCP o en respuesta a alguna interrupción, también se regresa al estado Inicio, después de ser enviado un mensaje NOTIFICATION con su respectivo código.

Establecido

Esta es la etapa final en la negociación entre vecinos BGP. En este estado, BGP inicia el intercambio de paquetes UPDATE con sus vecinos. El contador del tiempo de espera se reinicia cuando se recibe un mensaje KEEPALIVE o un mensaje UPDATE. Si el sistema recibe algún mensaje NOTIFICATION, es decir, que un error se ha producido, el estado pasa a Inicio.

III.2.7. Sesiones entre vecinos

Construcción de la sesión

Aunque BGP se emplea entre Sistemas Autónomos para proporcionar una topología interdominios libre de lazos o *loops*, también puede ser usado en un AS como un túnel entre enrutadores de frontera que ejecuten EBGP hacia otros ASs.

Bajo el establecimiento de la sesión entre vecinos y durante el intercambio de mensajes OPEN en la negociación, los enrutadores vecinos comparan sus números de AS y determinan si son vecinos dentro del mismo AS o de diferentes ASs. La diferencia entre EBGP e IBGP se manifiesta en cómo procesa cada vecino las actualizaciones proporcionadas por otros y en la forma en la que los diferentes atributos BGP se transportan en sus enlaces internos o externos.

El proceso de negociación es esencialmente el mismo para vecinos internos o externos, así como la construcción de la conexión TCP en la capa de transporte. Es primordial tener una conectividad IP entre los dos vecinos para que la sesión tenga lugar. La conectividad IP es llevada a cabo por medio de protocolos diferentes a BGP y para lograrla, se puede configurar un protocolo IGP (*Interior Gateway Protocol*) o un enrutamiento estático.

Conexiones físicas y lógicas

Los vecinos EBGP tienen una restricción al ser conectados directamente. BGP desecha cualquier actualización de su vecino externo si el vecino no está conectado directamente. Sin embargo, existen algunas situaciones donde los vecinos externos no están en el mismo segmento físico y entonces se consideran vecinos lógicos, no físicos.

Para establecer vecinos lógicos se ha introducido el concepto de interfaz *loopback*, la cual es una intefaz virtual que se supone activa todo el tiempo. Al relacionar la conexión con un vecino a su interfaz lógica, se garantiza que la sesión no depende de ninguna interfaz física que pueda ser problemática.

No es necesario añadir interfaces *loopback* en cada situación. Si los vecinos externos están directamente conectados y la dirección IP del segmento directamente conectado se usa para la negociación entre vecinos, la interfaz *loopback* sobra. Si el enlace físico entre los dos vecinos es problemático, entonces la sesión se interrumpirá con o sin interfaz *loopback*.

Verificación de la sesión

El encabezado de los mensajes BGP permiten llevar a cabo una verificación. La verificación es una medida de precaución contra *hackers* que puedan presentarse como uno de los vecinos BGP y alimentar el enlace con información falsa. La verificación entre dos vecinos BGP otorga la capacidad de validar la sesión pues usa una combinación de contraseñas y llaves. Un vecino que trata de establecer una sesión sin el uso de

contraseñas y llaves específicas no tendrá permiso de entrar. La verificación utiliza el algoritmo MD5 (*Message-Digest* versión 5).

Continuidad de BGP dentro de un AS

Para evitar lazos o *loops* de enrutamiento creados dentro de un AS, BGP no anuncia a sus vecinos internos las rutas que aprendió por medio de otros enrutadores IBGP, lo cual es importante para mantener una malla completa IBGP dentro del AS, es decir, cada enrutador BGP en el AS tiene que construir una sesión BGP con todos los otros enrutadores dentro del AS.

Sincronización dentro de un AS

BGP debe estar sincronizado con el IGP del AS de tal forma que BGP espere hasta que el IGP haya propagado la información de enrutamiento a través del Sistema Autónomo antes de anunciar sus rutas a otros AS. En el momento en el que un enrutador reciba una actualización acerca de un destino de un *peer* IBGP, el enrutador trata de comprobar el alcance interno para tal destino antes de anunciarlo a otros vecinos EBGP. El enrutador entonces verifica la existencia del destino en IGP, proporcionando información de si los enrutadores que no son BGP pueden entregar tráfico a tal destino. Suponiendo que IGP reconoce el destino, el enrutador lo anunciará a otros vecinos EBGP. De otra manera, el enrutador considerará que la ruta no está sincronizada con el IGP y no la anunciará.

Las reglas BGP establecen que un enrutador BGP no debería anunciar a sus vecinos externos los destinos que haya aprendido de sus vecinos internos a menos que esos destinos también sean conocidos por IGP. Si un enrutador conoce los destinos por medio de IGP, se asume que la ruta ya ha sido propagada dentro del AS y se garantiza su alcance interno.

La consecuencia de inyectar rutas BGP dentro de un AS es costosa. El redistribuir rutas de BGP al IGP resulta en un sobreflujo en los enrutadores internos, los cuales seguramente no están equipados para manejar tantas rutas. Además, realmente no es necesario transportar todas las rutas externas dentro del AS.

El enrutamiento puede ser efectuado fácilmente si se tienen enrutadores que no son BGP que liberen de un poco de tráfico a los enrutadores BGP. Obviamente esto resulta en enrutamiento sub-óptimo (no hay garantía de que se elige la trayectoria más corta para cada ruta), pero el costo es mínimo comparado con el mantenimiento de miles de rutas en el AS.

III.2.8. Proceso de enrutamiento

BGP es un protocolo sencillo y flexible. Como ya se mencionó, las rutas se intercambian entre BGPs por medio de mensajes UPDATEs que al ser recibidos, se ejecutan políticas y filtros sobre ellos para ser transmitidos posteriormente a otros vecinos BGP.

Generalmente se guardan todas las actualizaciones BGP en una tabla de enrutamiento BGP separada de la tabla de enrutamiento IP. En el caso de que existan múltiples rutas con el mismo destino, BGP no inunda su *peer* con todas esas rutas, sino que toma la mejor de ellas y la envía. Además de que se transmiten las rutas a través de todos los enrutadores, un enrutador BGP puede originar actualizaciones de enrutamiento para anunciar las redes que pertenecen a su propio AS. Las rutas locales válidas originadas en el sistema y las mejores rutas aprendidas de los vecinos BGP se emplean para tomar una decisión final en el proceso de enrutamiento.

Para representar el proceso BGP, se puede considerar que cada enrutador BGP tiene diferentes rutas y diferentes motores de políticas asociadas a esas rutas. Un modelo podría involucrar los siguientes componentes:

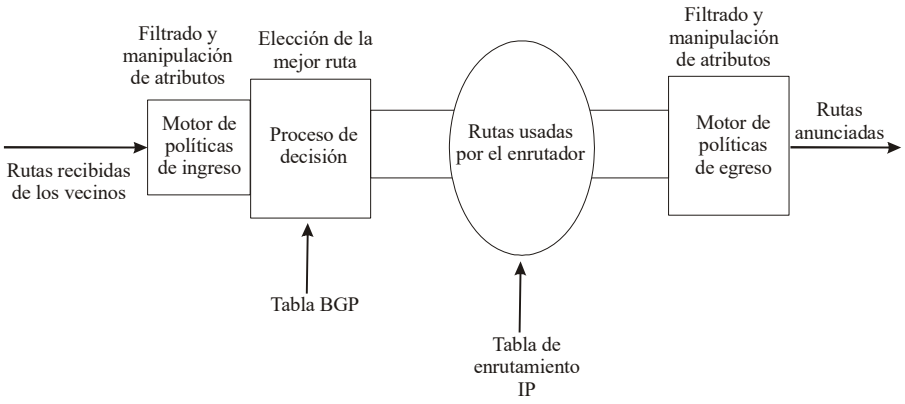


Figura III.2.8. Modelo del proceso de enrutamiento

- Un conjunto de rutas que el enrutador recibe de sus vecinos

  BGP recibe rutas de sus vecinos internos o externos. Según la configuración del motor de políticas de entradas, algunas o todas estas rutas construyen la tabla BGP del enrutador.

- Un motor de políticas de ingreso

  Este motor maneja el filtrado de las rutas y la manipulación de sus atributos. El filtrado es realizado de acuerdo a varios parámetros, tales como los prefijos IP, la información AS_*path* y la información de otros atributos. BGP también usa este motor para manipular los atributos y poder influir en su propio proceso de decisión para que las rutas alcancen su destino.

- Un proceso de decisión que determine qué rutas usará el enrutador

  BGP utiliza un proceso de decisión para determinar qué rutas quiere usar para alcanzar cierto destino. El proceso de decisión está basado en las rutas que se conservan en el enrutador después de que el motor de políticas de ingreso haya sido aplicado. El proceso de decisión se ejecuta en la tabla de enrutamiento BGP y busca todas las rutas disponibles para el mismo destino, compara los diferentes atributos asociados a cada ruta y elige la mejor.

- Un conjunto de rutas usadas por el enrutador

  Las mejores rutas que hayan sido identificadas en el proceso de decisión son las rutas que el enrutador usa y son candidatas para ser anunciadas a otros vecinos y también para ser colocadas en la tabla de enrutamiento IP. Además de que las rutas son transmitidas a otros vecinos, el enrutador (si está configurado para ello) origina actualizaciones de las redes de su Sistema Autónomo. Esta es la forma en la que un AS inyecta sus rutas a las redes exteriores.

- Un motor de políticas de egreso

  Es el mismo motor que el de políticas de ingreso, pero aplicado al lado de salida. Las rutas usadas por el enrutador (las mejores rutas) y las rutas que el enrutador genera localmente se someten al proceso de este motor, el cual aplica filtros y puede cambiar algunos atributos antes de enviar la actualización.

- Un conjunto de rutas que el enrutador anuncia a otros vecinos

  Es el conjunto de rutas que resultan del motor de políticas de egreso y que son anunciadas a los vecinos BGP, ya sean internos o externos.

III.2.9. Tipos de atributos en BGP

Los atributos BGP son parte de un paquete UPDATE. Son un conjunto de parámetros que describen las características de un prefijo o ruta, usados por el proceso de decisión para seleccionar la mejor. El tráfico dentro de un AS fluye de acuerdo al mapa que han construido los enrutadores con sus rutas. Si se alteran estas rutas, se cambia el comportamiento del tráfico.

Los atributos BGP pueden resolver problemas como la forma en la que se pueda evitar que una red privada sea anunciada, la manera de filtrar las actualizaciones de enrutamiento provenientes de un vecino en particular o cómo se puede garantizar que el enlace o el proveedor que se usa es el que se indica. Los atributos BGP se clasifican en:

- Atributos bien conocidos mandatorios:

  Son los atributos que tienen que existir en el paquete UPDATE de BGP. Debe ser reconocido por todas las implementaciones BGP. Si se pierde u omite un atributo bien conocido mandatorio, debe generarse un mensaje NOTIFICATION para asegurarse que todas las implementaciones BGP tengan un estándar del conjunto de atributos.

- Atributos bien conocidos discrecionales:

  Son los atributos reconocidos por todas las implementaciones BGP, pero que pueden o no ser enviados en el mensaje UPDATE.

Además de los atributos bien conocidos, una ruta debe contener uno o más atributos opcionales. Los atributos opcionales no requieren ser soportados por todas las implementaciones BGP. Los atributos opcionales pueden ser transitivos o no transitivos.

- Atributos opcionales transitivos:

  En el caso de que un atributo opcional no sea reconocido por la implementación BGP, se debe buscar una bandera transitiva para saber si tal implementación corresponde a un atributo en particular. Si es así (la bandera transitiva es uno), la implementación BGP debe aceptar el atributo y pasarlo hacia otros enrutadores BGP.

- Atributos opcionales no transitivos:

  Cuando un atributo opcional no es reconocido y la bandera transitiva no es igual a cero, el atributo debe ser ignorado y no puede pasar hacia otros enrutadores BGP.

A continuación se presenta una tabla con el tipo de tributo y el valor del código que les corresponde:

Tabla III.2.9. Tipos de atributos

Nombre del atributo	Tipo de atributo	Código del tipo de atributo
ORIGIN	Bien conocido mandatorio	1
AS_*path*	Bien conocido mandatorio	2
NEXT_HOP	Bien conocido mandatorio	3
MULTI_EXIT_DISC	Opcional no transitivo	4
LOCAL_PREF	Bien conocido discrecional	5
ATOMIC_AGGREGATE	Bien conocido discrecional	6
AGGREGATOR	Opcional transitivo	7
COMMUNITY	Opcional transitivo	8
ORIGINATOR_ID	Opcional no transitivo	9
Lista del grupo	Opcional no transitivo	10
Destino de preferencia	---	Definido por MCI
Anunciante	---	Definido por Baynet
rcid_path	---	Definido por Baynet
Reservado para desarrollo	---	255

III.2.9.1. Atributo ORIGIN

El atributo origen es bien conocido mandatorio con código 1 de tipo de atributo. Indica el origen de una actualización de enrutamiento (NLRI, el cual tiene prefijo y máscara IP) con respecto al Sistema Autónomo que lo originó. BGP considera tres tipos de orígenes:

- IGP: el NLRI es interno al AS origen
- EGP: el NLRI es aprendido por medio de un EGP
- INCOMPLETO: el NLRI es aprendido por otro medio

BGP considera el atributo ORIGIN en su proceso de decisión para establecer una preferencia entre varias rutas. Específicamente, BGP prefiere la ruta con el menor tipo de origen, donde IGP es menor a EGP y EGP es menor a INCOMPLETE.

III.2.9.2. Atributo AS_path

Con formato: Español (México)

Un atributo AS_path es bien conocido mandatoio con código 2 del tipo de atributo. Es una secuencia de números de los Sistemas Autónomos que una ruta ha atravesado para alcanzar su destino. El Sistema Autónomo que origina la ruta añade su propio número de AS cuando envía la ruta a sus vecinos externos BGP. A partir de entonces, cada AS que recibe la ruta y la transmite a sus vecinos BGP, también añadirá su propio número de AS a la lista. La lista final contiene todos los números de los AS que una ruta ha atravesado y al final de ella está el número del AS que la originó. Este tipo de lista se llama AS_sequence debido a que los números de los AS están ordenados secuencialmente.

BGP emplea este atributo como parte de las actualizaciones de enrutamiento (paquetes UPDATE) para asegurar la topología libre de lazos que requiere Internet. Cada ruta que es transmitida entre dos vecinos BGP, transporta una lista de todos los AS por los que ha cruzado. Si la ruta se anuncia al AS que la originó, ese AS ya es parte de la lista del atributo AS_path y no aceptará la ruta. La información AS_path es uno de los atributos BGP que busca determinar la mejor ruta para alcanzar un destino. Si se comparan dos o más rutas diferentes que tengan los demás atributos idénticos, se prefiere aquella ruta que tenga la lista AS_path menor.

Para conservar los números AS, generalmente InterNIC no asigna un número AS legal a los clientes cuyas políticas son una extensión de otras políticas de su propio proveedor. Esto significa que usualmente el proveedor necesita que el clientes use un número AS tomado del conjunto privado de ASs (64512-65535). Por lo tanto, todas las actualizaciones BGP que recibe el proveedor de su cliente, contienen números AS privados. Los números AS privados no pueden ser filtrados a Internet debido a que no son únicos. Por esta razón, se ha implementado una característica para retirar el AS privado de la lista AS_path antes de que la ruta sea propagada a Internet.

III.2.9.3. Atributo NEXT_HOP

Con formato: Español (México)

Este atributo es bien conocido mandatorio con código de tipo 3. En IGP el next_hop para alcanzar una ruta es la dirección IP de la interfaz conectada al enrutador que ha anunciado la ruta, pero en BGP el concepto de BGP es más elaborado y toma una de las siguientes tres formas:

1. Para sesiones EBGP: el next_hop es la dirección IP del vecino que anuncia la ruta.
2. Para sesiones IBGP: para rutas originadas en el AS, el next_hop es la dirección IP del vecino que anuncia la ruta. Para rutas inyectadas en el AS por medio de EBGP, el next_hop aprendido de EBGP es transportado inalterado en IBGP. El next_hop es la dirección IP del vecino EBGP de donde se aprendió la ruta.
3. Cuando la ruta se anuncia en un medio multiacceso (tal como Ethernet, Frame Relay y demás), el next_hop es generalmente la dirección IP de la interfaz del enrutador conectado al medio que origina la ruta.

Para comprender mejor lo anterior, se puede considerar un ejemplo:

El enrutador SF ejecuta una sesión EBGP con el enrutador LA y una sesión IBGP con el enrutador SJ. El enrutador SF aprende la ruta 128.213.1.0/24 del enrutador LA. Por otra parte, el enrutador SF inyecta la ruta local 192.212.1.0/24 a BGP.

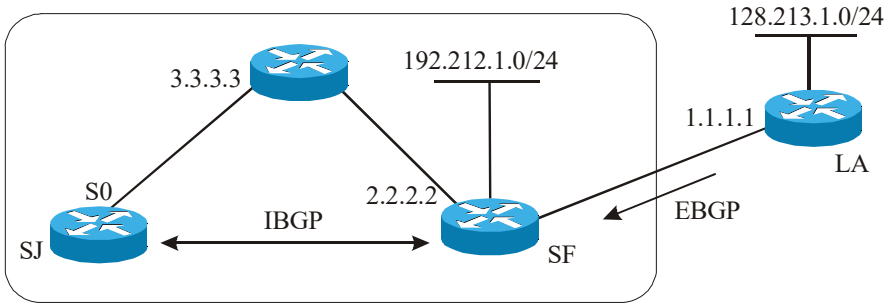


Figura III.2.9.3

El enrutador SJ aprende la ruta 192.212.1.0/24 por medio de 2.2.2.2, la dirección IP del vecino IBGP que anuncia la ruta. En este caso, según la definición, el next_hop de SJ para alcanzar a 192.212.1.0/24 es 2.2.2.2. Similarmente, el enrutador SF aprende 128.213.1.0/24 del enrutador LA por medio de su next_hop 1.1.1.1. Cuando SF transmite una actualización a SJ, no altera la información del next_hop y así, SJ conoce que el siguiente salto para alcanzar 128.213.1.0/24 es 1.1.1.1. En este caso se observa que el siguiente salto no necesariamente es alcanzable por una conexión directa.

En el caso de tener una red multiaccesos (Ethernet, Frame Relay), el next_hop no es sencillo de reconocer. Un medio es considerado multiacceso si los enrutadores conectados a dicho medio tienen la capacidad de intercambiar datos en una relación muchos a muchos. Los enrutadores en el medio multiacceso comparten la misma subred IP y tienen acceso físico a todos los otros enrutadores del medio en un solo salto.

III.2.9.4. Atributo MULTI_EXIT_DISC (MED)

MED es un atributo opcional no transitivo con código 4 de tipo de atributo. El MED es una indicación para los vecinos externos acerca de qué tan preferida es una trayectoria en un AS que tiene múltiples puntos de entrada. Al MED se le conoce también como la métrica exterior de una ruta. Un valor menor de MED es preferido sobre otro valor mayor.

A diferencia de la preferencia local, el atributo MED sí se intercambia entre Ass, pero un MED que ingrese a un AS, no sale de él. Cuando una actualización entra a un AS con cierto valor MED, éste se usa para hacer las decisiones dentro del AS. Cuando BGP transmite las actualizaciones de enrutamiento a otro AS, el MED se pone a cero a menos que el MED saliente tenga un valor específico.

Cuando la ruta es originada por el AS, el valor del MED sigue la métrica interna IGP de la ruta, lo cual es útil cuando un cliente tiene múltiples conexiones al mismo proveedor. La métrica IGP refleja qué tan cercana o

lejana se encuentra una red al punto de salida. Una red que está más cercana al punto de salida A que al punto de salida B, tendrá una métrica IGP menor en el enrutador de frontera conectado al punto A. Cuando la métrica IGP se traduce a MED, el tráfico que llegue al AS puede entrar por el enlace más cercano al destino debido a que el MED menor es preferido para el mismo destino, lo cual puede aprovechar tanto el proveedor como el cliente para balancear el tráfico sobre múltiples enlaces entre dos ASs.

A menos de que sea especificado de otra manera, los enrutadores comparan los atributos MED para las rutas a sus vecinos externos que están en el mismo AS. Los MEDs de diferentes AS no son comparables ya que el MED asociado con una ruta usualmente tiene alguna indicación de la topología interna del AS.

Para ilustrar la definición de MED se puede considerar un ejemplo:

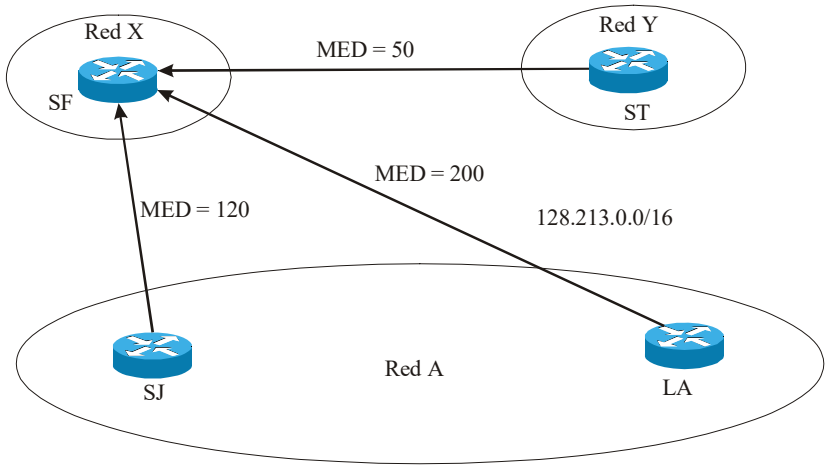


Figura III.2.9.4

La red X recibe actualizaciones de 128.213.0.0/16 de tres diferentes fuentes: SJ (métrica 120), LA (métrica 200) y ST (métrica 50). SF compara los dos valores de las métricas que vienen de la red A y preferirá el enrutador SJ debido a que tiene la métrica menor (120).

Los clientes que se conecten al mismo proveedor en múltiples puntos, pueden intercambiar métricas con sus proveedores para influir al otro en su tráfico al exterior con el fin de obtener un mejor balanceo de carga.

III.2.9.5. Atributo LOCAL_PREF

La preferencia local es un atributo bien conocido disecrecional con código 5 del tipo de atributo. La preferencia local o LOCAL_PREF es el grado de preferencia dado a una ruta para compararla con otras que tengan el mismo destino. Un valor mayor de preferencia local india que la ruta es más preferida. La preferencia local, como lo indica el nombre, es local al Sistema Autónomo y sólo es intercambiada entre vecinos IBGP y nunca es transmitida a vecinos EBGP.

La preferencia local generalmente es usada para establecer el punto de salida de un sistema autónomo para alcanzar cierto destino. Debido a que este atributo es comunicado a todos los enrutadores BGP dentro del AS, todos ellos tendrán una vista común de cómo salir del AS.

III.2.9.6. Atributo ATOMIC_AGGREGATE

Con formato: Español (México)

La sumarización (o agregación) de rutas causa pérdida de información debido a que al sumarizar rutas que provienen de diferentes fuentes, se tienen diferentes atributos. Este atributo es bien conocido discrecional con código 6 del tipo de atributo y es una indicación de pérdida de información. Básicamente, si un sistema propaga una sumarización, causa pérdida de información y es necesario añadir el atributo ATOMIC_AGGREGATE a la ruta.

El atributo ATOMIC_AGGREGATE no debe ser enviado cuando la sumarización transporta alguna información extra que da una indicación de dónde viene la información sumarizada.

III.2.9.7. Atributo AGGREGATE

Este atributo es opcional transitivo con código 7. especifica el Sistema Autónomo y el enrutador que ha generado una sumarización. Un enrutador BGP que ejecuta sumarización de rutas debe añadir el atributo AGGREGATE, el cual contiene el número de Sistema Autónomo y la dirección IP de la *loopback* del enrutador.

III.2.9.8. Atributo COMMUNITY

En el contexto de BGP, una comunidad es un grupo de destinos que comparten algunas propiedades comunes. Una comunidad no se restringe a una sola red o a un solo Sistema Autónomo, pues no tiene límites físicos. Un ejemplo es un grupo de redes que pertenecen a la misma comunidad educacional o del gobierno y pueden pertenecer a cualquier Sistema Autónomo. Las comunidades se usan para simplificar las políticas de enrutamiento identificando las rutas de acuerdo a las propiedades lógicas, más que al prefijo IP o al número de AS. Un enrutador BGP puede usar este atributo en conjunto con otros atributos para controlar qué rutas va a aceptar, a preferir y a transmitir a otros vecinos BGP.

El atributo COMMUNITY tiene código 8 de tipo de atributo y es opcional transitivo con una longitud variable. Sus valores constan de cuatro bytes y tienen un rango de 00000000 a 0000FFFF hexadecimal y los valores reservados van de FFFF0000 a FFFFFFFF, todo en hexadecimal. Estas comunidades tienen un significado global. Un ejemplo de comunidades conocidas son:

- NO_EXPORT (FFFFFF01 hex): una ruta que transporte este valor de comunidad no debe ser anunciada a sus vecinos de confederación o de AS si sólo existe un AS en la confederación.
- NO_ADVERTISE (FFFFFF02 hex): una ruta que transporte este valor de comunidad, cuando se reciba, no debe ser anunciada a ningún otro vecino BGP.

Se pueden definir atributos de comunidades privadas para suso especiales. Una práctica común es usar los dos primeros bytes del atributo comunidad para el número del AS y los dos últimos bytes para definir un valor en relación al AS.

Una ruta puede tener más de una comunidad. Un enrutador BGP que encuentre múltiples atributos de comunidades en una ruta puede actuar basado en una de ellas, en algunas o en todas. Un enrutador tiene la opción de añadir o modificar la comunidad antes de transmitirla a otros vecinos.

Una vez descritos todos los atributos, es posible entender el proceso de decisión de BGP:

BGP basa su proceso de decisión en los valores de los atributos. Si se encuentra con varias rutas con el mismo destino, BGP elige la mejor ruta según el criterio que a continuación se presenta a grandes rasgos:

1. Si el next_hop es inaccesible, la ruta es ignorada (esta es la razón por la que es importante tener una ruta IGP al siguiente salto).
2. Se prefiere la ruta con la preferencia local mayor.
3. Si las rutas tienen la misma preferencia local, se elige la ruta que haya sido originada localmente, es decir, por ese enrutador.
4. Si la preferencia local es la misma, se elige la ruta con la menor lista AS_path.
5. Si la AS_path es la misma, se prefiere la ruta con el menor tipo de origen.
6. Si el tipo de origen es el mismo, se prefiere la ruta con menor MED.
7. Si las rutas tienen el mismo MED, se prefiere a las EBGP antes que a las IBGP.
8. Si todos los escenarios anteriores son idénticos, se prefiere la ruta que pueda ser alcanzada por medio del vecino IGP más cercano, es decir, que tome la trayectoria más pequeña dentro del AS para alcanzar el destino.
9. Si la ruta interna es la misma, el identificador del enrutador BGP será el que decida. Se elige a la ruta que provenga del enrutador con menor *loopback*.

III.3. Distribución de etiquetas MPLS usando BGP

Existen situaciones en las que los vecinos de BGP intercambian tanto etiquetas como NLRIs (como formalmente se les conoce a las rutas), las cuales son intercambiadas convencionalmente por BGP. El grupo de trabajo MPLS ha definido una pequeña extensión a BGP4 para habilitar el uso del mecanismo de distribución de etiquetas.

Debido a que BGP4 es extensible, MPLS hace uso de las extensiones multiprotocolo de BGP. Estas extensiones permiten que BGP soporte diferentes familias de direcciones, tal como IPv4 o IPv6. MPLS simplemente define una nueva familia de direcciones en la que las direcciones no solamente incluyen prefijos IP, sino también una o más etiquetas. La codificación de los prefijos IP y las etiquetas se muestra en la siguiente figura:

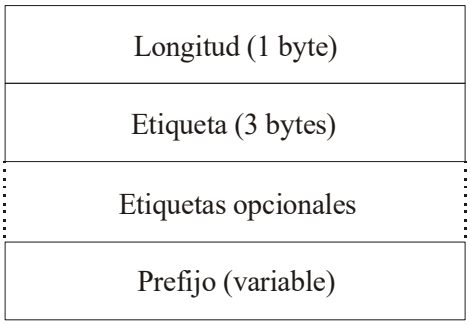


Figura III.3.1. Codificación de etiquetas y prefijos IP en BGP

La Longitud total es la longitud de la(s) etiqueta(s) más la longitud de los prefijos de dirección IP. Cada Etiqueta se codifica en tres bytes, con 20 bits para el número de etiqueta (ver la figura III.1.4.). El bit más significativo de los restantes cuatro bits de esos tres bytes cumple la misma función que el bit *Stack* del encabezado MPLS. Los tres bits restantes de los tres bytes se ponen en cero. Después de la etiqueta sigue el prefijo de la dirección IP (que puede ser de cualquier longitud. Por ejemplo, para IPv4 es de 32 bits) y se rellena con bits ceros hasta completar un número par de bytes.

Con la codificación anterior de etiquetas y de prefijos de dirección IP, cualquier enrutador que hable BGP y que anuncie una ruta, puede conocer una etiqueta o un *stack* de etiquetas para que sea usado por el paquete en dicha ruta. Todos los mecanismos de BGP se usan normalmente.

III.3.1. MPBGP

BGP version 4 actualmente es el protocolo de enrutamiento exterior que se usa de facto en Internet y usa un protocolo *Path Vector* sobre TCP para transportar información de ruteo entre sistemas autónomos (AS). BGP ha sido extendido para soportar VPNs con MPLS, "convirtiéndose" en MPBGP, *Multiprotocol Border Gateway Protocol*.

BGP es capaz de transportar información solamente para IPv4 y las extensiones que lo convierten en MPBGP lo hacen capaz de transportar también información de enrutamiento para múltiples protocolos de la capa de red. Los enrutadores que tienen configuradas dichas extensiones pueden interoperar con enrutadores que no las tienen.

En MPBGP se introducen dos nuevos atributos: *Multiprotocol Reachable* NLRI (MP_REACH_NLRI) y *Multiprotocol Unreachable* NLRI (MP_UNREACH_NLRI). El primero es usado para transportar el conjunto de destinos alcanzables junto con la información del siguiente salto para enviar dichos destinos. El segundo de ellos se emplea para el transporte de los destinos inalcanzables. Ambos atributos son opcionales y no transitivos. De esta forma, un enrutador que hable BGP y que no soporte las capacidades multiprotocolos, solamente ignora la información transportada en estos atributos y no la pasa a otros enrutadores BGP.

La información del siguiente salto transportada en el atributo MP_REACH_NLRI define la dirección en la capa de red del enrutador frontera que puede ser usado como el siguiente salto para los destinos listados en el mensaje UPDATE del atributo MP_REACH_NLRI. Cuando se anuncia un atributo MP_REACH_NLRI a un vecino externo, un enrutador puede usar una de sus direcciones de interfaz en el campo del siguiente salto del

atributo, dado que el vecino externo al cual la ruta está siendo anunciada comparte una subred común con la dirección del siguiente salto.

Un enrutador que hable BGP puede anunciar a un vecino externo una interfaz de cualquier enrutador interno en el campo del siguiente salto del atributo, dado que la dirección de capa de red de ese enrutador frontera fue aprendido de un vecino externo y que el vecino externo al cual la ruta está siendo anunciada comparte una subred común con la dirección del siguiente salto. Sin embargo, un enrutador que hable BGP también debe ser capaz de deshabilitar el anuncio de las direcciones de interfaz por razones de políticas de enrutamiento o para manejar un desperfecto en el "puenteo" de información.

Desde el punto de vista de las VPNs, *Multiprotocol BGP* es un protocolo de distribución de información de enrutamiento que, a través del uso de extensiones multiprotocolos y atributos de comunidades, define quién le puede hablar a quién. Los miembros de una VPN dependen de los puertos lógicos al inicio de la VPN, que es donde MPBGP asigna un valor único llamado *Route Distinguisher* (RD). Los RDs son desconocidos para los usuarios, haciendo imposible que un usuario de cierta VPN ingrese a otra.

En una VPN MPLS, MPBGP distribuye las tablas FIB (*Forwarding Information Base*) de cierta VPN únicamente a los miembros de dicha VPN, proporcionando una seguridad nativa por medio de la separación lógica del tráfico en la VPN. Además, los enrutadores frontera del proveedor (nube MPLS) establecen trayectorias hacia otro de ellos usando LDP para comunicar información *binding* o de envío de etiquetas. La información *binding* de etiquetas para rutas externas, es decir, para rutas del cliente, se distribuyen hacia todos los PEs usando MPBGP en lugar de LDP debido a que es más fácil añadir esa información a la información IP de la VPN, la cual ya está siendo distribuida.

El atributo comunidad fuerza la extensión del alcance de la información. MPBGP mapea las tablas FIB a los enrutadores PE pertenecientes a una VPN en particular, en lugar de actualizar todos los enrutadores frontera en la red MPLS.

Referencias

Davie, Bruce y Rekhter, Yakov. *MPLS, Technology and Applications.*

Morgan Kaufmann Publishers, E.U.

Capítulo 5, pp. 121-145.

Guichard, Jim y Pepelnjak, Ivan. MPLS and VPN Architectures.
Cisco Press, E.U., 2000.
Capítulo 8, pp 141-163.

(falta la bibliografía de un libro)

http://www.ietf.org/rfc/rfc2283.txt

http://www.webtorials.com/main/resource/papers/BCR/paper23.htm

Capítulo IV Redes Privadas Virtuales

IV.1. Introducción

Las Redes Privadas Virtuales o VPN (*Virtual Private Network*) son una alternativa a la infraestructura WAN que está reemplazando a las redes privadas actualmente existentes por medio de líneas rentadas, redes Frame Relay o ATM propias de una empresa. Una VPN puede utilizar cualquier tecnología de transporte disponible, ya sea el Internet público, o los *backbones* o las redes Frame Relay o ATM de un proveedor de servicios. Sin embargo, la funcionalidad de una VPN está definida principalmente por el equipo instalado en la frontera de la red corporativa y de la integración de las características a través de la WAN y no del protocolo de transporte WAN. [1]

Las VPNs pueden conectar a usuarios remotos fijos, a usuarios remotos móviles y a oficinas remotas pequeñas con la red corporativa o, inclusive, puede lograr la conexión de dos empresas socias con el fin de compartir información. Según la VPN que se implemente, se tienen que considerar diferentes características de seguridad y de administración del ancho de banda.

IV.2. Descripción

Una Red Privada Virtual (VPN) está definida como una red en la cual la conexión del cliente entre múltiples sitios está desarrollada sobre una infraestructura compartida con las mismas políticas de acceso y seguridad de una red privada[2].

Una solución VPN se define según la extensión de las características ofrecidas. Una VPN debe ser segura de ataques a la información, asegurar la entrega confiable de datos en situaciones críticas y ser fácilmente administrable para la empresa. Una solución VPN tiene ciertos componentes[1]:

- **El proveedor de servicios**, que es una organización dueña de la infraestructura (el equipo y el medio de transmisión) que proporciona líneas dedicadas a sus clientes. El proveedor de servicios ofrece al cliente un servicio de Red Privada Virtual.
- **El cliente**, quien se conecta a la red del proveedor de servicio a través del "Equipo de Permiso a Clientes" (*CPE, Customer Permises Equipment*),. El dispositivo CPE es también llamado CE (*Customer Edge*). El cliente es quien paga por los servicios de una VPN.
- **El dispositivo CE**, el cual es conectado a través del medio de transmisión (usualmente una línea dedicada, pero también puede ser una conexión de línea conmutada) hacia el equipo del proveedor de servicios, el cual puede ser un conmutador X.25 o Frame Relay o ATM, o incluso un enrutador IP. El CE es generalmente un dispositivo ensamblador y desensamblador de paquetes (*PAD, Packet Assembly and Disassembly*) que provee la conectividad del puente o el enrutador en la terminal.
- **El dispositivo PE**, que es la frontera del proveedor de servicio. El acrónimo es por *Provider Edge*. El PE interconecta a los dispositivos CE con la red del proveedor de servicios.
- **Dispositivos P**. l proveedor de servicio frecuentemente tiene un equipo adicional en el corazón de su red (conocida también como *P-Network* o *Red-P*). Estos dispositivos son llamados Dispositivos-P (*P-Devices*), y como ejemplo están los Enrutadores-P (*P-Enrutadors*) o Conmutadores-P (*P-Switches*).
- **El sitio**. La parte contigua a la red del cliente (*Costumer Network*) es llamada "sitio" (site). Uno se puede conectar a la Red-P (*P-network*) a través de una o varias líneas de transmisión, usando uno o varios dispositivos CE y PE, según los requerimientos de redundancia.
- **Las líneas dedicadas**, las cuales son proporcionadas al cliente por el proveedor de servicios sobre el modelo VPN (*Virtual Private Network*).recuentemente son llamadas VC (*Virtual Circuit*) o "Circuito Virtual". El VC (*Virtual Circuit*) puede estar disponible todo el tiempo (como los Circuitos Virtuales Permanentes o *Permanent Virtual Circuit, PVC*) o bien, puede estar establecido bajo demanda (*Switched*

Virtual Circuit SVC o Circuito Virtual Conmutado). Algunas tecnologías usadas en términos especiales para las VC's, son, por ejemplo, *Data Link Connection Identifier* (*DLCI*) para Frame Relay.

- **La tasa de tráfico.** El proveedor de servicios puede dar una tasa plana para un servicio VPN, el cual normalmente depende del ancho de banda disponible para el cliente, o usar una tasa que dependa del uso, la cual puede depender del volumen o de la duración de los datos intercambiados.

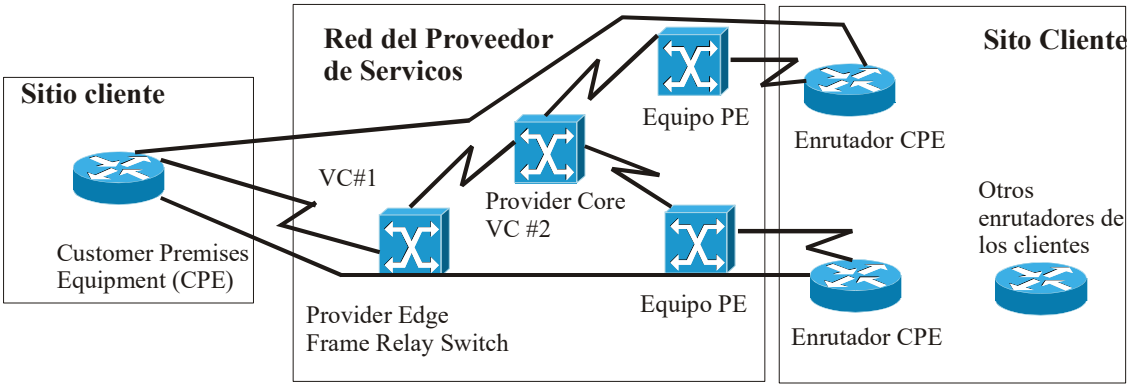


Figura IV.2.1

Una VPN típica debe tener una red LAN principal en los edificios de la compañía, otras LANs en oficinas remotas y usuarios individuales que se conectan remotamente. Básicamente, una VPN es una red privada que usa una red pública (usualmente Internet) para conectarse a sitios remotos, pero en lugar de usar líneas dedicadas, la VPN emplea conexiones virtuales enrutadas a través del Internet desde la LAN principal, hasta el sitios remoto[2].

Los elementos esenciales de una VPN deben tener ciertas características[4]:

- Escalabilidad en la plataforma
- Seguridad
- Confiabilidad
- Administración
- Políticas de seguridad

En general, una VPN ofrece más ventajas que las redes basadas en líneas dedicadas, teniendo cuatro principales:

- Costos más bajos que en las redes privadas: el costo total se reduce con el bajo costo del transporte de los datos, del ancho de banda, del equipo para el *backbone* y de operaciones.
- Las VPNs tienen inherentemente arquitecturas más flexibles y escalables que las clásicas redes WAN, además de que habilita a las empresas para extender su conectividad rápidamente y a costos efectivos y facilita la conexión o desconexión a oficinas remotas, locaciones internacionales, usuarios móviles o a redes de las empresas socias, según se requiera.
- Reducida administración de la carga en comparación con las redes privadas propias, pues las empresas pueden relegar la administración de su red a un proveedor de servicios y enfocarse más en los negocios que les competen.
- Topologías de red más simples, resultado de una reducida administración de la carga. El utilizar un *backbone* IP elimina los Circuitos Virtuales Permanentes (PVCs) asociados a los protocolos orientados a conexión, tales como ATM o Frame Relay, además de que crea una topología de malla completa que disminuye el costo y la complejidad de la red.

Además, una VPN bien diseñada puede ofrecer grandes beneficios a una empresa, como por ejemplo[2]:

- Conectividad en una extensa área geográfica
- Seguridad en la información
- Reducidos tiempos en el transporte de datos y bajos costos para los usuarios remotos
- Incrementa la productividad
- Simplifica la topología de la red
- Oportunidad de interconectarse globalmente

IV.3 Clasificación de las VPNs

[1]Como existe una gran variedad de tecnologías y topologías para VPN, la única manera de manejar exitosamente esta diversidad, es introduciendo una clasificación de VPNs, la cual se puede realizar de acuerdo a los siguientes criterios:

- **El problema del negocio de VPNs que se esté tratando de solucionar**. La mayoría de los problemas surgen en la comunicación entre una misma compañía (también llamada comunicación intranet), en la comunicación entre compañías (también llamada extranet) y en el acceso para usuarios móviles (mejor conocido como Red VPN *Dialup*).
- **La capa del modelo OSI** en la cual el proveedor de servicio intercambia la información de topología con el cliente. La mayoría de las categorías son modelos *overlay* (extendidos), donde el proveedor de servicios atiende al cliente únicamente con un conjunto de enlaces punto a punto (o multipunto) entre los *sites*, o son modelos "*peer-to-peer*" (vecino-a-vecino o igual-a-igual), donde el proveedor de servicios y el cliente intercambian información de enrutamiento de capa 3.
- **La tecnología de capa 2 o de capa 3 usada** para implementar el servicio VPN dentro de la red proveedora de servicio. la cual puede ser X.25, Frame Relay, SMDS, ATM o IP.
- **La topología de la red**, la cual puede ser desde una topología simple con un *hub* hasta una red con una malla completa o topologías multiniveles jerárquicas en redes más grandes.

Las Redes Privadas Virtuales (VPNs) pueden ser clasificadas en varias formas. La clasificación tecnológica más amplia es aquella basada en la forma en la que se intercambia información en la VPN. En el modelo VPN *peer-to-peer*, la información de enrutamiento del cliente es intercambiada entre los enrutadores del cliente y los enrutadores del proveedor de servicios. En el modelo VPN *overlay*, el proveedor de servicios proporciona únicamente VCs (líneas lógicas rentadas) y la información de enrutamiento es intercambiada directamente entre los enrutadores de los clientes en la frontera (enrutadores CE). Los dos modelos pueden ser combinados en una red más grande de Proveedor de Servicios: el modelo VPN *peer-to-peer* puede usar VPN *overlay* en sus partes de acceso (por ejemplo, enlace de los enrutadores del Proveedor de Servicios a través de ATM).

La clasificación más detallada de las VPNs, mostrada en la figura que se muestra abajo (figura IV.3.1), se enfoca en la tecnología de capa 3 que es usada para el transporte de paquetes sobre la VPN. El modelo VPN *overlay* puede ser implementado con tecnologías WAN de conmutación de capa 2 (como X.25, Frame Relay, SMDS o ATM) o con tecnologías de encapsulado de capa 3 (como IP sobre IP, IPsec). El modelo VPN *peer-to-peer* puede ser implementado tradicionalmente con complejos trucos de enrutamiento o con listas de acceso IP. El Multiprotocolo de Conmutación de Etiquetas, (mejor conocido como MPLS, *Multiprotocol Label Switching*) basado en VPNs, supera la mayoría de las fallas de otras tecnologías VPN *peer-to-peer*, permitiendo a los Proveedores de Servicios combinar los beneficios del modelo *peer-to-peer* (más sencillo de enrutar, implementación más sencilla de los requerimientos del cliente) con la seguridad y el aislamiento inherente del modelo VPN *overlay*.

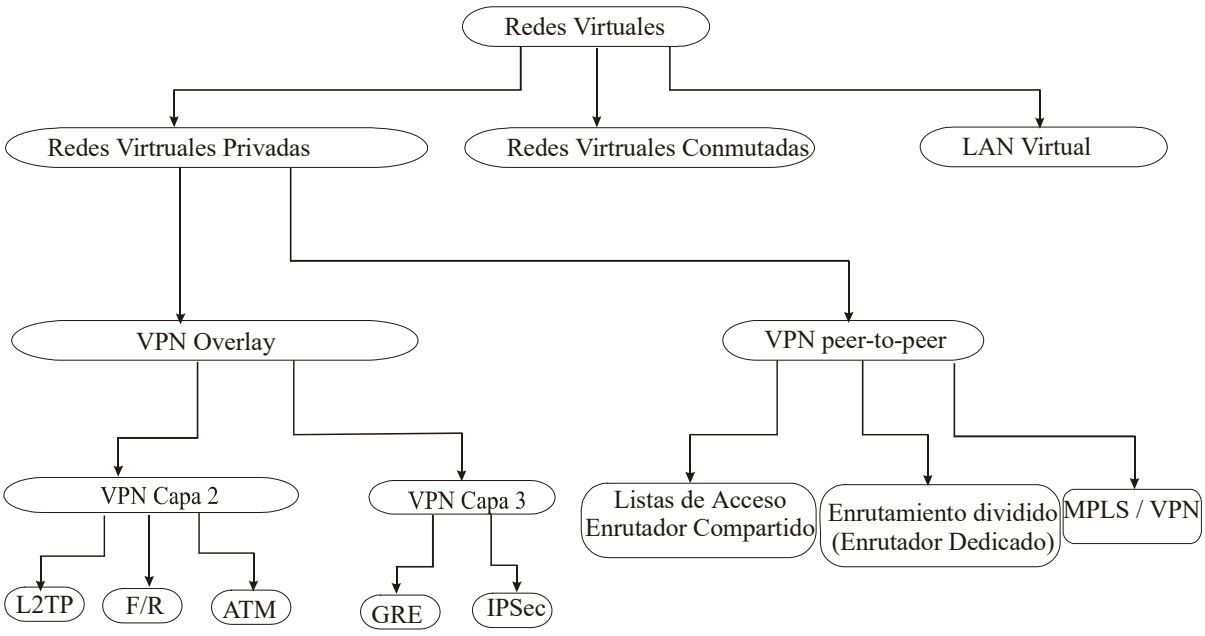


Figura IV.3.1

IV.3.1 Clasificación de VPNs basada en el problema a solucionar

Los tres problemas típicos de una empresa que se tratan de resolver con una red privada virtual (*Virtual Private Network*) son:

- Comunicación interna en la organización (intranet)
- Comunicación con otras organizaciones (extranet)
- Acceso de usuarios móviles, trabajadores, oficinas remotas y más a través de un medio de conmutación barato

Los tres tipos de soluciones con VPNs usualmente explotan la mayoría de las topologías y tecnologías ofrecidas por los Proveedores de Servicios VPNs, pero difieren grandemente en el nivel de seguridad requerido en su implementación.

Las comunicaciones internas en una organización frecuentemente no están bien protegidas por los *hosts* finales o por los *firewalls*. El servicio VPN usado para implementar la comunicación intranet además debe ofrecer altos niveles de aislamiento y seguridad. La comunicación intranet también requiere calidad de servicio garantizada para los procesos críticos. Éstas son las razones principales por las cuales existen muchas organizaciones que usen Internet para comunicarse. ya que éste no puede ofrecer calidad de servicio punto a punto, aislamiento o seguridad, así como una infraestructura adecuada para la comunicación intranet. Las Redes Privadas Virtuales o VPNs fueron implementadas comúnmente con tecnologías tradicionales, como X.25, Frame Relay o ATM.

Las comunicaciones extranet (o inter-organizacionales) frecuentemente toman lugar entre los sitios centrales de las organizaciones. Usualmente se usan dispositivos dedicados de seguridad, tales como *firewalls* o equipos de cifrado similares, tal como se muestra en la figura IV.3.1.1:

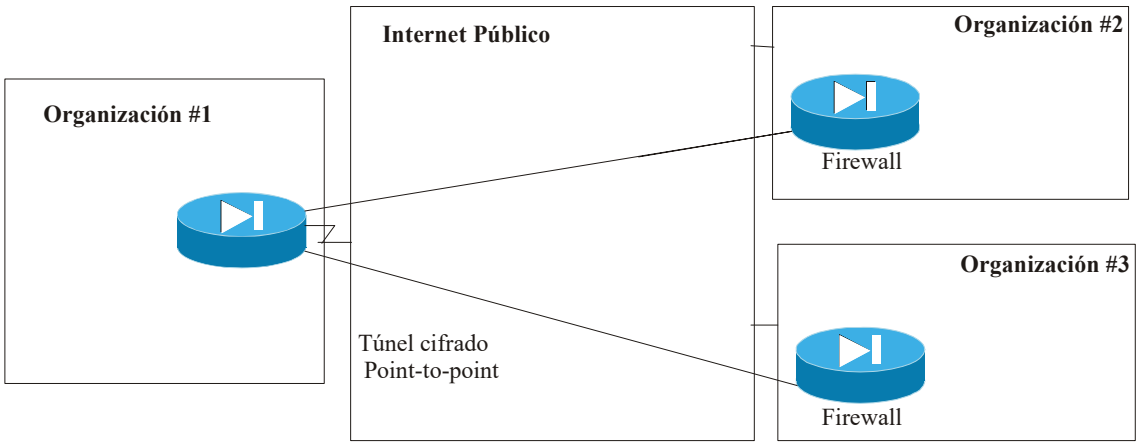


Figura IV.3.1.1

Este tipo de comunicaciones puede exigir también menos requisitos de Calidad de Servicio, por lo que el Internet tal vez sea más conveniente para las comunicaciones extranet; por lo tanto, no es una sorpresa que cada vez más y más tráfico de empresa a empresa tome lugar sobre Internet.

El acceso remoto del usuario dentro de una red corporativa, típicamente desde direcciones cambiantes o desconocidas, siempre filtrado con elementos de seguridad, obtenidos a lo largo del enlace punta a punta usando tecnologías de cifrado o una contraseña de un solo tiempo (*one-time password*). De esta manera, los requerimientos de seguridad para los servicios VPDN son tan rigurosos para las comunicaciones intranet que la mayoría de los servicios estén implementados actualmente sobre IP (*Internet Protocol*), sobre Internet o usando el *backbone* privado de un Proveedor de Servicios, tal como se muestra en la siguiente figura:

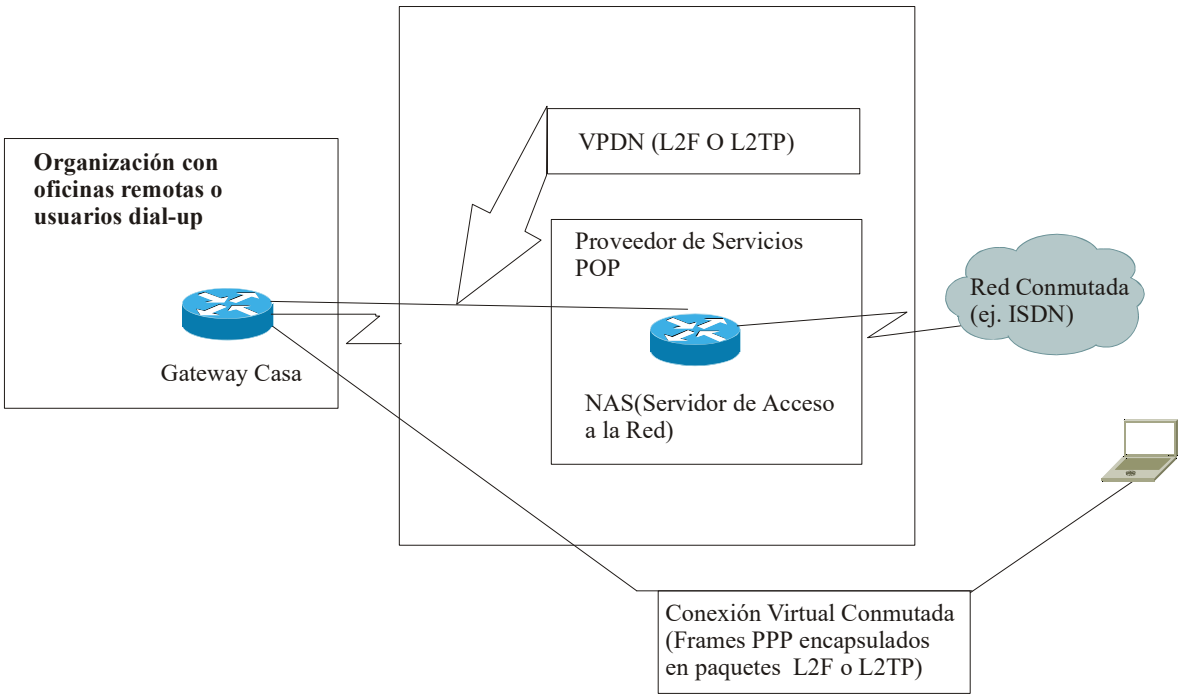


Figura IV.3.1.2

Los protocolos usados para implementar servicios de VPNs sobre IP incluyen L2F (*Layer 2 Forward*), PPTP (*Point-to-Point Tunneling Protocol*) o L2TP (*Layer 2 Trasnport Protocol*). La tecnología VPDN usa un número especial de términos que son únicos al mundo VPDN:

- ***Network Access Server* (NAS):** es el Servidor de Acceso Remoto (RAS, *Remote Access Server*) es administrado por el proveedor de servicios que acepta la llamada del cliente, llevando a cabo la verificación y enviando la llamada (por L2F o por L2TP) hacia el *gateway* del cliente.
- ***Home Gateway*:** es un enrutador administrado por un cliente que acepta la llamada enviada por el NAS, ejecuta una verificación y una autorización adicional y termina la sesión PPP del usuario. Los parámetros de la sesión PPP (incluyendo las direcciones de red, tales como la dirección IP) son negociadas entre el usuario y el gateway propio; NAS sólo envía paquetes PPP entre los dos.

IV.3.2 Modelos VPN *Peer-to-Peer* y Overlay

Los dos modelos de implementación que han expandido su uso son:

- El modelo overlay o extendido, en el cual el Proveedor de Servicios proporciona al cliente líneas rentadas emuladas.
- El modelo *peer-to-peer*, donde el Proveedor de Servicios y el cliente intercambian información de enrutamiento de capa 3 y el proveedor enruta los datos entre los sitios (o sites) de los clientes en la trayectoria óptima sin la participación del cliente.

Modelo VPN *Overlay*

El modelo VPN overlay es el más sencillo de comprender debido a que proporciona claramente la separación de las responsabilidades entre el cliente y el proveedor de servicios pues el Proveedor de Servicios proporciona al cliente un conjunto de líneas contratadas emuladas. Dichas líneas son llamadas VCs (Circuitos Virtuales), los cuales pueden estar disponibles permanentemente (PVCs) o establecidos por demanda (SVCs).

La siguiente figura muestra la topología de una VPN overlay y los circuitos virtuales usados en ella:

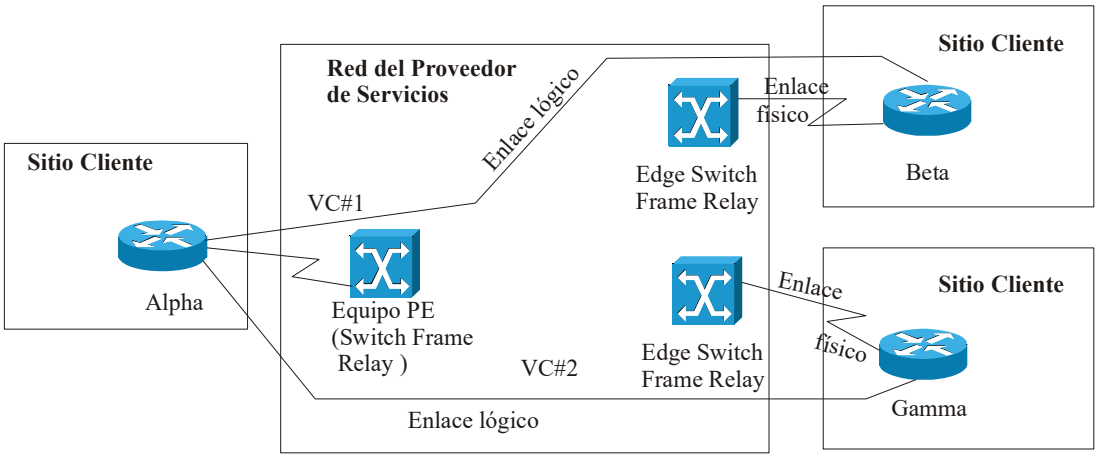


Figura IV.3.2.1

El cliente establece la comunicación enrutador a enrutador entre los equipos CE del cliente, dispositivos sobre los cuales se instauran los circuitos virtuales por medio del Proveedor de Servicios. Los datos del protocolo de enrutamiento siempre son intercambiados entre los dispositivos del cliente y el Proveedor de Servicios no tiene conocimiento de la estructura interna de la red del cliente. La siguiente figura muestra la topología de los enrutadores de la red VPN de la figura anterior:

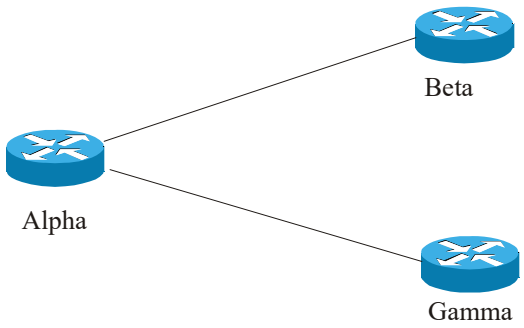


Figura IV.3.2.2

La Calidad de Servicio (QoS) garantiza que el modelo VPN overlay sea expresado comúnmente en términos del ancho de banda comprometido en un circuito virtual determinado (CIR, *Committed Information Rate*) y un máximo ancho de banda disponible en el circuito virtual (PIR, *Peak Information Rate*). La garantía del ancho de banda consignado es proporcionada usualmente a través de la naturaleza estadística de los servicios de capa 2, pero depende de la estrategia de sobreventa del proveedor de servicios. Esto significa que la tasa consignada no está realmente garantizada aunque el proveedor pueda proporcionar una Tasa Mínima de Información (MIR, *Minimum Information Rate*) que es obtenida a través de la infraestructura de capa 2.

La garantía del ancho de banda consignada es también una garantía del ancho de banda entre dos puntos en la red del cliente. Sin una matriz completa de tráfico para todas las clases de tráfico, es difícil para el cliente maniobrar las garantías en la mayoría de las redes overlay. También es difícil proporcionar las múltiples clases de servicio debido a que el Proveedor de Servicios no puede diferenciar el tráfico en medio de la red.

Trabajar así, creando múltiples conexiones (por ejemplo, en Frame Relay los PVCs) entre los sitios de los clientes, sólo incrementa el costo general de la red.

Las redes VPNs overlay pueden ser implementadas con un gran número de tecnologías de conmutación de WAN de capa 2, incluyendo a X.25, Frame Relay, ATM o SDMS. En los últimos años, las redes VPNs *overlay* también han sido implementadas con métodos de tuneleo de IP sobre IP, todos en *backbones* privados de IP sobre el Internet público. Los dos métodos más comunes de tuneleo IP sobre IP son *Generic Route Encapsulation* (GRE) y la encriptación con *IP Security* (IPSec).

A pesar de que es relativamente fácil de entender e implementar, el modelo VPN overlay tiene desventajas:

- Es apropiado para configuraciones no redundantes con pocos sitios centrales y sitios bastantes remotos, pero llega a ser extremadamente difícil de administrar en una configuración más compleja.
- La implementación propia de las capacidades del circuito virtual requiere un conocimiento detallado de los perfiles de tráfico de sitio a sitio, los cuales no siempre están disponibles.
- Cuando es implementado con tecnologías de capa 2, el modelo VPN overlay introduce otra capa innecesaria de complejidad en las redes *New World Service Provider* que, en su mayoría, están basadas en IP, lo que incrementa los costos operacionales de tal red.

Modelo VPN *Peer-to-Peer*

Este modelo fue introducido hace pocos años para superar las desventajas del modelo VPN overlay. En el modelo *peer-to-peer*, el dispositivo límite o de frontera (PE) del proveedor es un enrutador (PE-enrutador) que intercambia directamente la información de enrutamiento con el enrutador CE. La siguiente figura muestra un ejemplo de VPN *peer-to-peer*, la cual es equivalente a la figura ilustrativa del modelo VPN overlay.

El modelo *peer-to-peer* proporciona ciertas ventajas sobre el modelo overlay tradicional:

- El enrutamiento (desde el punto de vista del cliente) llega a ser extremadamente simple, ya que el enrutador CE del cliente intercambia información de enrutamiento con sólo uno (o unos cuantos) enrutadores PE (PE enrutador) mientras que en las redes VPN overlay, el número de enrutadores vecinos pueden crecer a un número grande.
- El enrutamiento entre los sitios del cliente es siempre óptimo, ya que el enrutador PE del proveedor conoce la topología de la red del cliente y puede también establecer un enrutamiento óptimo entre sitios.
- El suministro del ancho de banda es más simple, ya que el cliente tiene que especificar sólo el ancho de banda de entrada y de salida para cada sitio (tasa de acceso Comprometido o *Committed Access Rate*, CAR, y Tasa de Entrega Comprometida, *Committed Delivery Rate*, CDR) y no los perfiles de tráfico exactos de sitio a sitio.
- La adición de un nuevo sitio es sencilla ya que el Proveedor de Servicio acondiciona sólo un sitio adicional y cambia la configuración en el enrutador PE adjunto. En el modelo VPN overlay, el Proveedor de Servicio debe proporcionar todo el conjunto de VCs manejado desde este sitio hacia otros sitios de clientes de la VPN.

Hay dos opciones disponibles para el modelo de VPN *peer-to-peer*:

- Enrutador compartido, donde varios clientes VPN comparten el mismo enrutador PE.
- Enrutador dedicado, donde cada cliente VPN tiene un enrutador PE dedicado.

Modelo *peer-to-peer* con enrutador compartido

Cuando se tiene un enrutador compartido, varios clientes pueden ser conectados al mismo enrutador PE. Las listas de acceso deben de ser configuradas en todas las interfaces PE-CE en los enrutadores PE para asegurar la separación de los clientes de la VPN y para prevenir que un cliente de la VPN pueda irrumpir en otra red VPN, o también para prevenir que un cliente de una VPN no pueda atacar otra red VPN. La siguiente figura ilustra un ejemplo de la configuración de enrutador compartido.

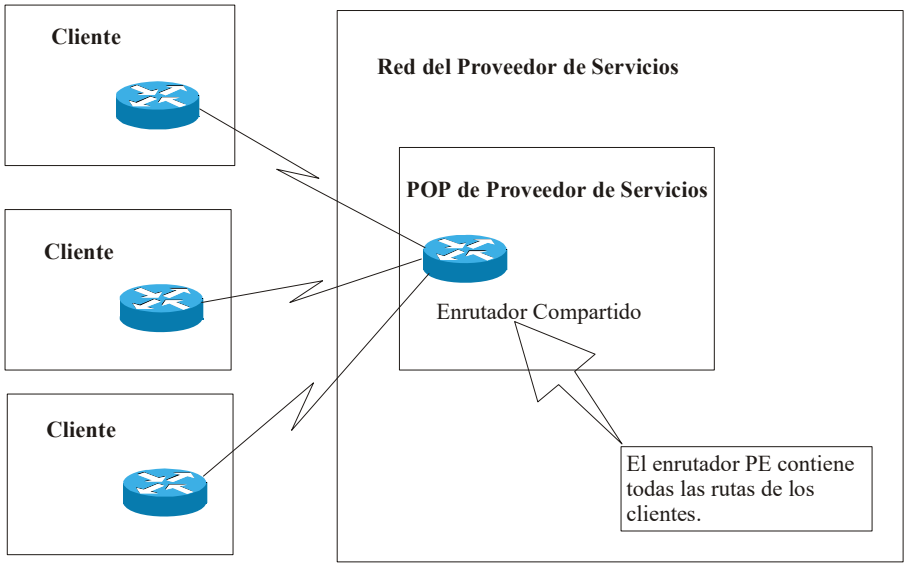


Figura IV.3.2.3

Modelo *peer to peer* con enrutador dedicado

En el modelo *peer to peer* con enrutador dedicado, cada cliente VPN tiene su propio enrutador PE dedicado (figura IV.3.2.4) y, sin embargo, sólo tiene acceso a los enrutadores que se encuentran en la tabla de enrutamiento del enrutador PE.

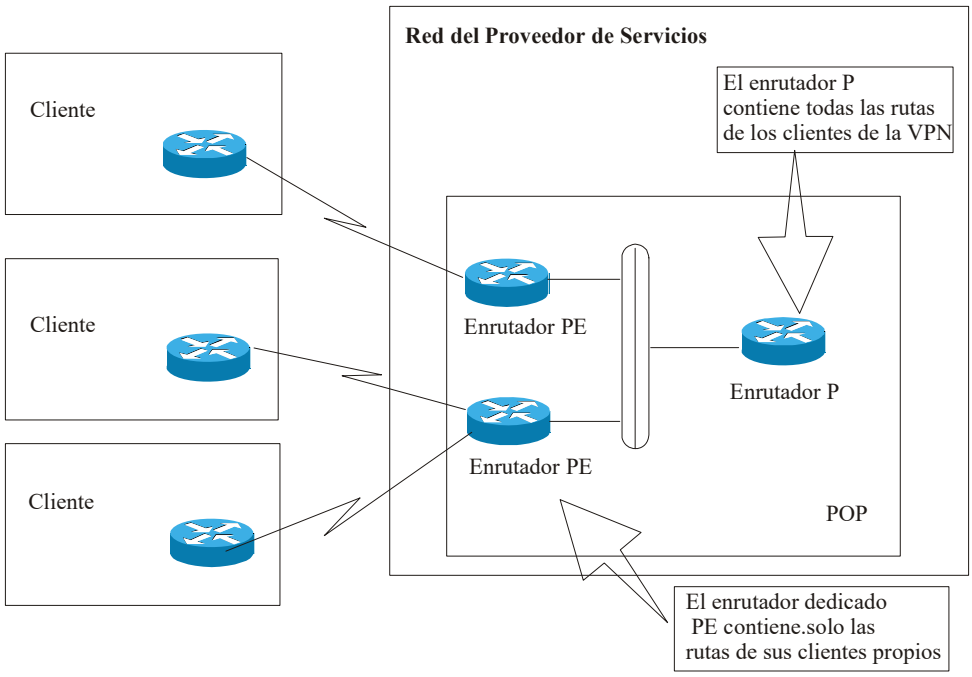


Figura IV.3.2.4

En el modelo de enrutador dedicado se usa un protocolo de enrutamiento para crear la tabla de enrutamiento de la VPN en los enrutadores PE. Las tablas de ruteo en un enrutador PE contienen sólo los enrutadores anunciados por los clientes VPN conectados a él, resultando en un aislamiento casi perfecto entre los clientes de la VPN (asumiendo que tiene que estar basado en una tabla de enrutamiento). Dentro de esta modalidad, el enrutador dedicado puede ser implementado como sigue:

- Cada protocolo de enrutamiento es ejecutado entre el enrutador PE y el enrutador CE.
- BGP es ejecutado entre el enrutador PE y el enrutador P
- El enrutador-PE redistribuye rutas recibidas desde el enrutador-CE encapsuladas en BGP, marcadas con una identificación del cliente (ID, comunidad BGP) y propaga las rutas a los enrutadores-P. De esta manera, el enrutador P contiene todas las rutas de todos los clientes VPN.
- Los enrutadores P sólo propagan rutas en comunidades BGP apropiadas por los enrutadores PE. Así, los enrutadores PE sólo reciben las rutas que se originaron desde los enrutadores CE dentro de la VPN.

Comparación de modelos *peer to peer*

El modelo *peer to peer* con enrutador compartido es muy difícil de mantener debido a que requiere el empleo de listas de acceso largas y complejas en casi cada interfaz del enrutador. El modelo de enrutador dedicado, aunque más sencillo de configurar y de mantener, llega a ser muy caro para el proveedor de servicios cuando trata de servir a un gran número de clientes con sitios geográficamente dispersos.

Ambos modelos comparten varias desventajas que evitan la expansión de su uso:

- Todos los clientes comparten el mismo espacio de direcciones IP, evitando que se usen direcciones IP privadas de acuerdo a la RFC 1918. Los clientes deben usar direcciones IP ya sean públicas o privadas para ser localizados por el proveedor de servicios.
- Los clientes no pueden insertar la ruta de *defalut* en su VPN. Esta limitación evita que tengan acceso a internet por medio de otro Proveedor de Servicios.

Además de estas dos ventajas, el modelo de enrutador compartido sufre de complejidad cuando varios clientes usan protocolos de enrutamiento (RIP, RIPv2, BGP y IS-IS) en donde existen varias instancias pero no son soportados por el *software* del enrutador.

IV.3.3. Topologías típicas de redes VPN

La topología VPN necesaria por una organización debería ser dictada por el problema que la organización está tratando de resolver, sin embargo, varias topologías conocidas son empleadas tan frecuentemente que serán discutidas a continuación. Como se podrá observar, una misma topología soluciona una gran variedad de diferentes contratiempos en diferentes niveles del mercado o la industria.

Las topologías más comunes se pueden dividir en tres categorías:

- Topologías influenciadas por el modelo VPN overlay o extendido, las cuales incluyen la topología *Hub-and-Spoke*, malla parcial o completa y topología híbrida.

- Topologías extranet, las cuales incluyen Extranet *any-to-any* y Extranet Servicios Centrales.
- Topologías de Propósito Especial, tales como el *Backbone* VPDN y topología de Red Administrada

IV.3.3.1 Topologías para VPNs Intranet

Topología *Hub-and-Spoke*

La topología más usada comúnmente es la topología *Hub-and-Spoke*, donde cierto número oficinas remotas (*spokes*) son conectadas a un sitio central (*hub*), similar a la figura siguiente:

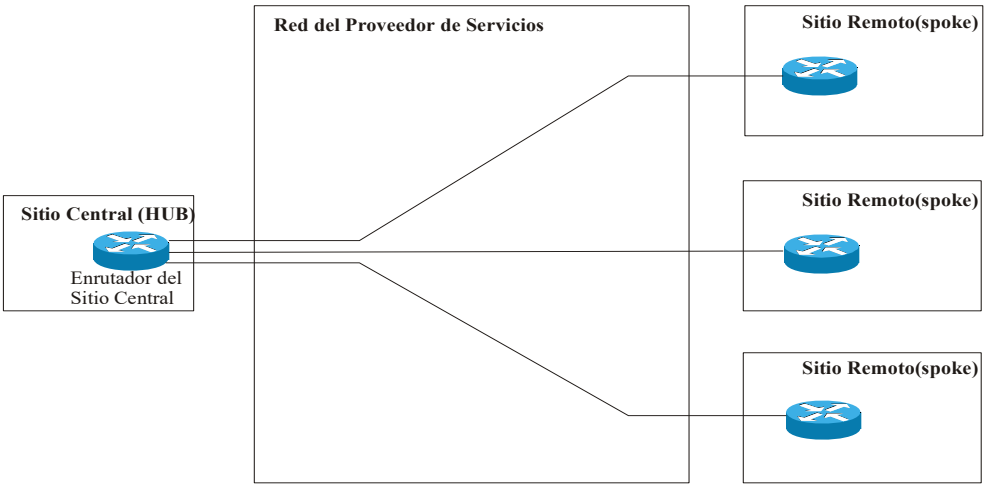


Figura IV.3.3.1.1

Las oficinas remotas usualmente pueden intercambiar datos. No hay restricciones específicas de seguridad en tráfico entre oficinas y la cantidad de datos intercambiados entre ellas es insignificante. Esta topología es usada típicamente en organizaciones con estructuras jerárquicamente estrictas, por ejemplo, bancos, oficinas de gobierno, almacenes, organizaciones internacionales con pequeñas oficinas en cada país, y más.

Al emplear VPNs basadas en tecnologías de capa 2, tales como Frame Relay o ATM, la topología *Hub-and-Spoke* es más común de lo que se podría esperar. Esta topología está basada puramente en necesidades

debidas a costos muy altos o a un incremento en la complejidad en el enrutamiento asociadas a otras topologías que usan otros tipos de tecnologías. En otras palabras, existen muchos ejemplos donde el cliente puede beneficiarse con otra topología diferente pero no tiene más elección que la *Hub-and-Spoken* por razones de costo o complejidad.

Con requerimientos de redundancia, la topología sencilla *Hub-and-Spoke* de la figura anterior frecuentemente es mejorada con un enrutador adicional en el sitio central, tal como se muestra en la figura IV.3.3.3.1.2:

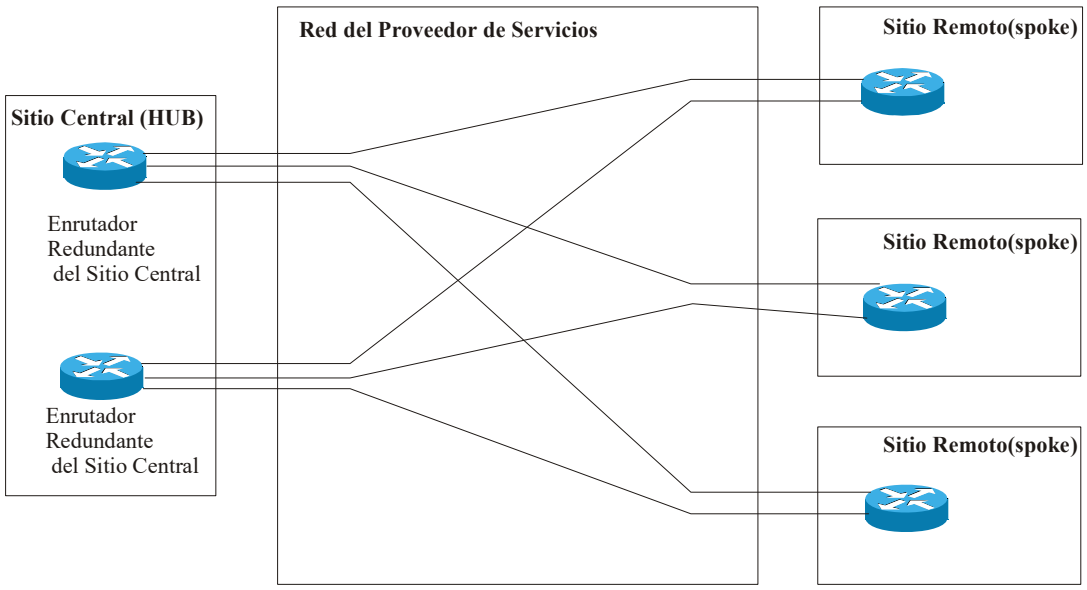


Figura IV.3.3.1.2

Otra forma es agregando un sitio central de respaldo, el cual es conectado con el sitio central primario a través de una conexión con velocidad más alta:

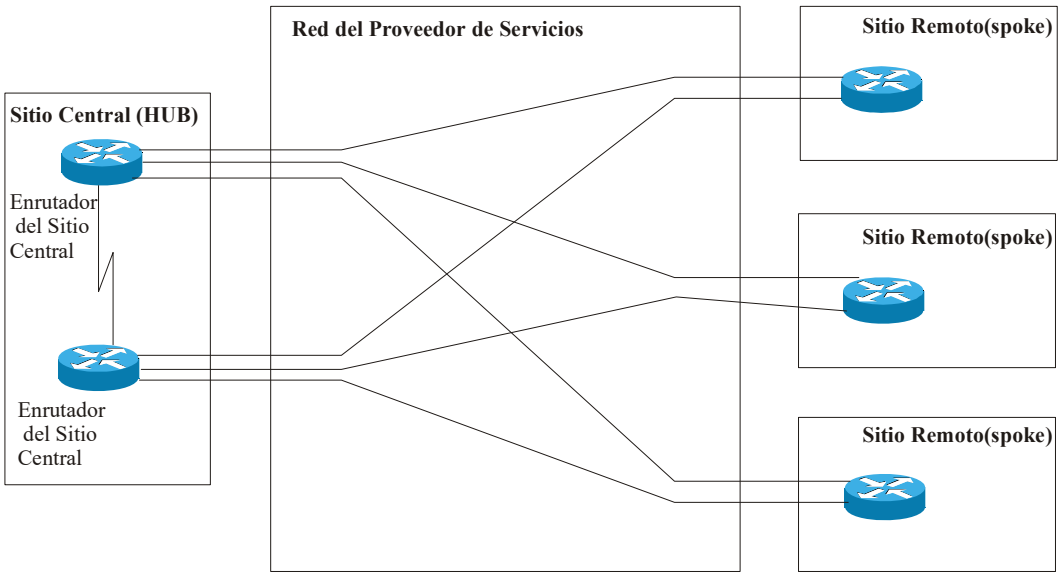


Figura IV.3.3.1.3

El implementar una topología *Hub-and-Spoke* redundante con un modelo VPN overlay basado en VC (*Virtual Circuits*) siempre presenta ciertos retos. Cada *hub* requiere al menos un VC a dos enrutadores centrales. Dichos VCs pueden ser provisionales en la configuración *backbone* primaria o en una configuración de balanceo de carga con un número de desventajas con una u otra solución:

- En la configuración *backbone* primaria, el circuito virtual (VC) de respaldo no es usado mientras el circuito virtual primario está activo, lo que resulta en gastos innecesarios adquiridos por el cliente.
- En la configuración de carga compartida, los sitios *"spokes*" o secundarios encuentran salidas reducidas si uno de los circuitos virtuales (o uno de los enrutadores centrales) fallan. Esta configuración no es apropiada para las topologías con un sitio central de respaldo similar al de la figura anterior.

Los proveedores de servicios de la más alta calidad tratan de cumplir con los requerimientos de redundancia de los clientes ofreciendo un servicio mejorado llamado *Shadow PVC*. Con un *Shadow PVC,* el cliente obtiene dos circuitos virtuales por el precio de uno con la condición de que sólo pueden usar uno a la vez para tráfico de datos (aunque una pequeña cantidad de datos es permitida en el segundo PVC para habilitar los intercambios del protocolo de enrutamiento).

Los requerimientos de redundancia pueden complicar aún más la topología *Hub-and-Spoke* con la introducción de características *dial-backup* (o respaldo de marcación). La solución *dial-backup* implementada en la red del Proveedor de Servicios (por ejemplo, una conexión ISDN respaldando una línea dedicada Frame Relay, como muestra en la figura IV.3.3.1.4)es transparente para el cliente, pero esto no ofrece una verdadera redundancia punta punta, ya que no puede detectar las fallas potenciales (por ejemplo, las fallas en el protocolo de enrutamiento). Una verdadera redundancia punta apunta sobre un modelo de VPN overlay puede ser archivado solo por los dispositivos CE estableciendo una conexión conmutada afuera de la VPN.

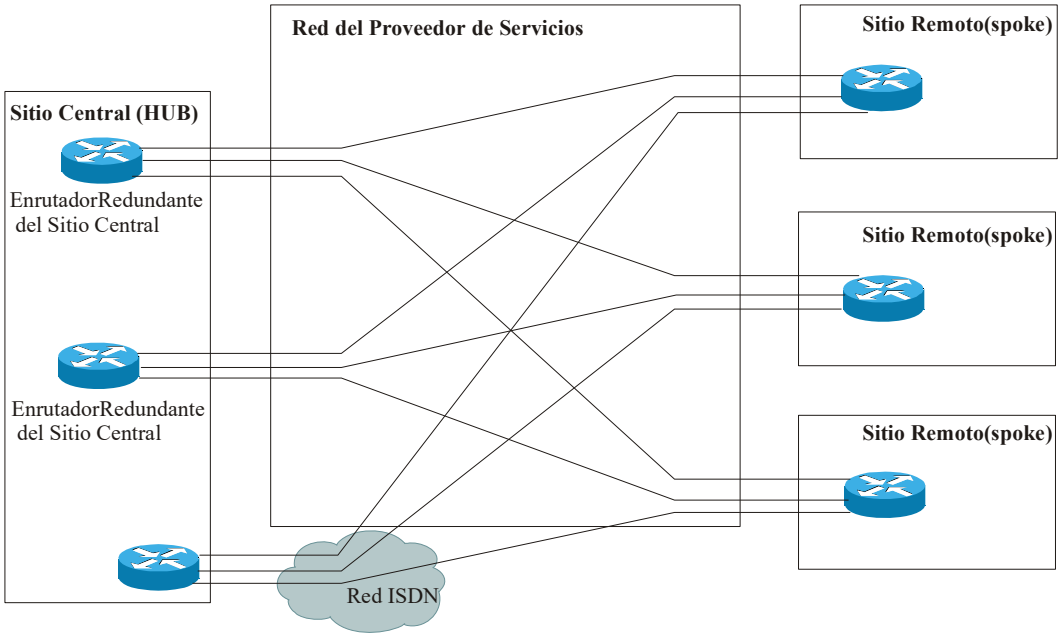


Figura IV.3.3.1.4

Usualmente, una topología simple *hub-and-spoke* se transforma a una topología multinivel conforme va creciendo la red. La topología multinivel puede ser una topología *hub-and-spoke* recursiva, similar a la mostrada en la figura IV.3.3.1.4, o puede ser también una topología híbrida. La reestructuración puede ser integrada por escalabilidad de restricciones de los protocolos de enrutamiento IP o por la escalabilidad de niveles de aplicación (por ejemplo, la introducción de una implementación de tres hileras cliente-servidor).

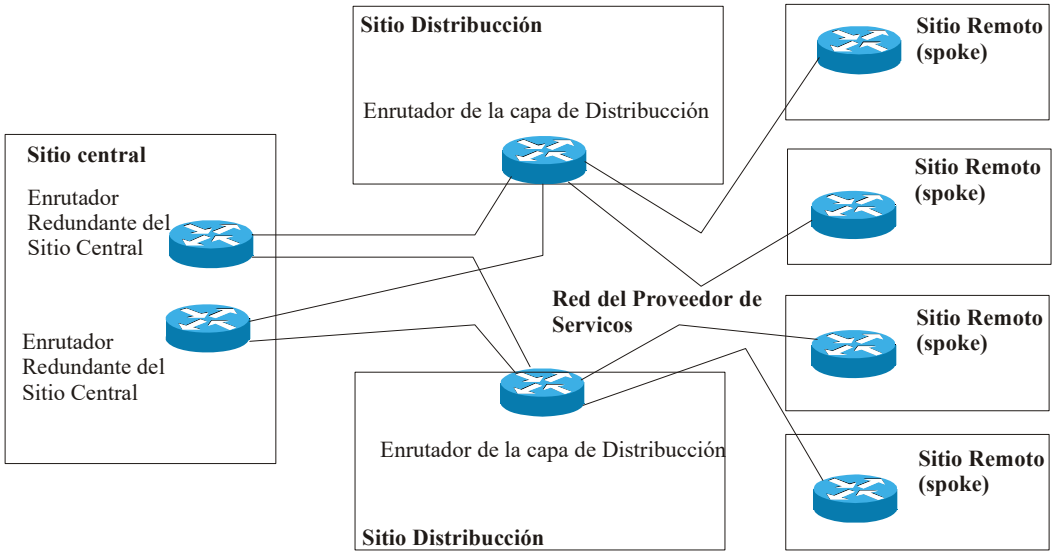


Figura IV.3.3.1.5

La implementación de la topología *hub-and-spoke* junto con un modelo VPN overlay es muy conveniente en entornos donde las oficinas remotas casi siempre intercambian datos con los puntos centrales y no con cualquier otro. Por ejemplo, el intercambio de datos entre las oficinas centrales siempre son transportadas por medio de sitios centrales si la cantidad de los datos intercambiados entre las oficinas remotas representa una proporción significativa del tráfico de la red extendida. Sin embargo, una topología de malla parcial o completa (*partial-mesh* y *full-mesh*) podría ser más conveniente.

Topología de malla parcial o completa (*partial-mesh* y *full-mesh*)

No todos los clientes pueden implementar sus redes con topologías *hub-and-spoke*:

- La organización puede ser menos jerárquica en estructura, requiriendo intercambio de datos entre varios puntos de la organización.
- Las aplicaciones usadas en la organización necesitan comunicación *peer-to-peer*, como los sistemas de mensajería o colaboración.
- Para algunas corporaciones multinacionales, el costo de la topología *hub-and-spoke* podría ser excesiva, así como el costo elevado de los enlaces internacionales.

En estos casos, el modelo VPN overlay más apropiado para las necesidades de una organización podría ser un modelo de malla parcial, donde los sitios dentro de la VPN son conectados por contenedores virtuales (VC) dictaminados por requerimientos de tráfico (donde eventualmente son dictados por la necesidades de negocios). Si no todos los sitios tienen una conectividad directa a todos los sitios (como por ejemplo, en la figura IV.3.3.1.6) la topología es llamada de malla parcial. Si todos los sitios tienen conectividad con todos los demás, es de malla completa.

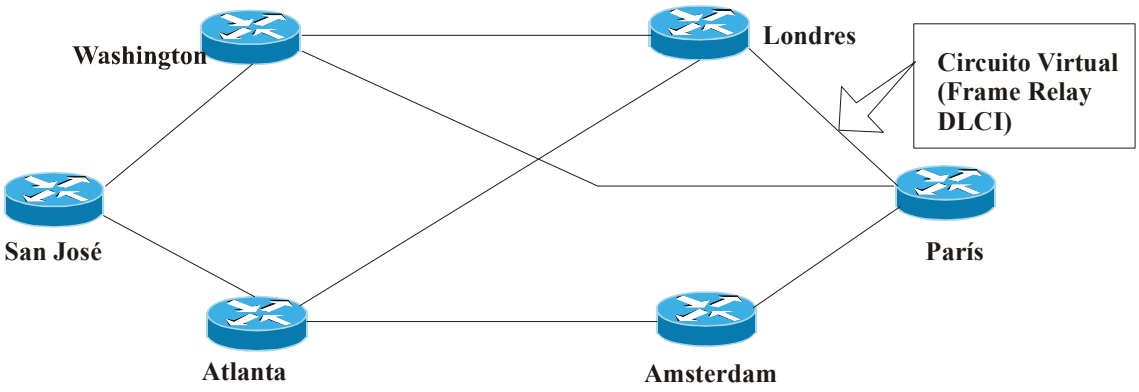


Figura IV.3.3.1.6

No se implementan muchas redes de malla completa debido al muy alto costo de esa implementación y la complejidad introducida por el gran número de VCs. Con ese tipo de topología, el número de VCs es: $VC = \left[\frac{(n-1)n}{2} \right]$ donde n es el número de dispositivos adjuntos. La mayoría de los clientes tienen que instalar una topología de malla parcial, la cual usualmente es afectada por compromisos o por parámetros externos, como la disponibilidad de enlaces y los costos de los VCs.

Implementar una topología de malla completa es bastante simple, sólo se necesita una matriz de tráfico indicando el ancho de banda requerido entre un par de sitios dentro de la VPN, pudiendo a empezar a ordenar los VCs al Proveedor del Servicio. Para el caso de una malla parcial, la implementación no es tan sencilla, ya que se debe de hacer lo siguiente:

1. Resolver la matriz de tráfico.

2. Proponer una topología de malla parcial basada en esa matriz de tráfico (por ejemplo, instalando un VC sólo entre sitios con altos requerimientos de tráfico) y en los requerimientos de redundancia.
3. Determinar exactamente sobre qué VC va a fluir el tráfico entre dos sitios. Este paso también puede involucrar una afinación del protocolo de enrutamiento para hacer más fluido el tráfico sobre las propias VCs.
4. Hacer el tamaño de VCs acorde a la matriz de tráfico y a la agregación del tráfico alcanzado sobre el VC.

Las características del protocolo de enrutamiento en una (usualmente multinacional) malla parcial larga puede crecer a la proporción donde es extremadamente difícil predecir el flujo de tráfico sin usar avanzadas herramientas de simulación. Esto obliga a los clientes a migrar a BGP sólo para manejar los problemas de ingeniería de tráfico en sus topologías de mallas parciales.

Topología híbrida

Las redes VPN grandes construidas con un modelo de VPN overlay tienden a combinar topología *hub-and-spoke* con topología de malla parcial. Por ejemplo, una gran organización multinacional puede tener acceso a las redes de cada país implementando una topología *hub-and-spoke* donde el núcleo de la red internacional puede ser implementada con una topología de malla parcial. La siguiente figura muestra este ejemplo.

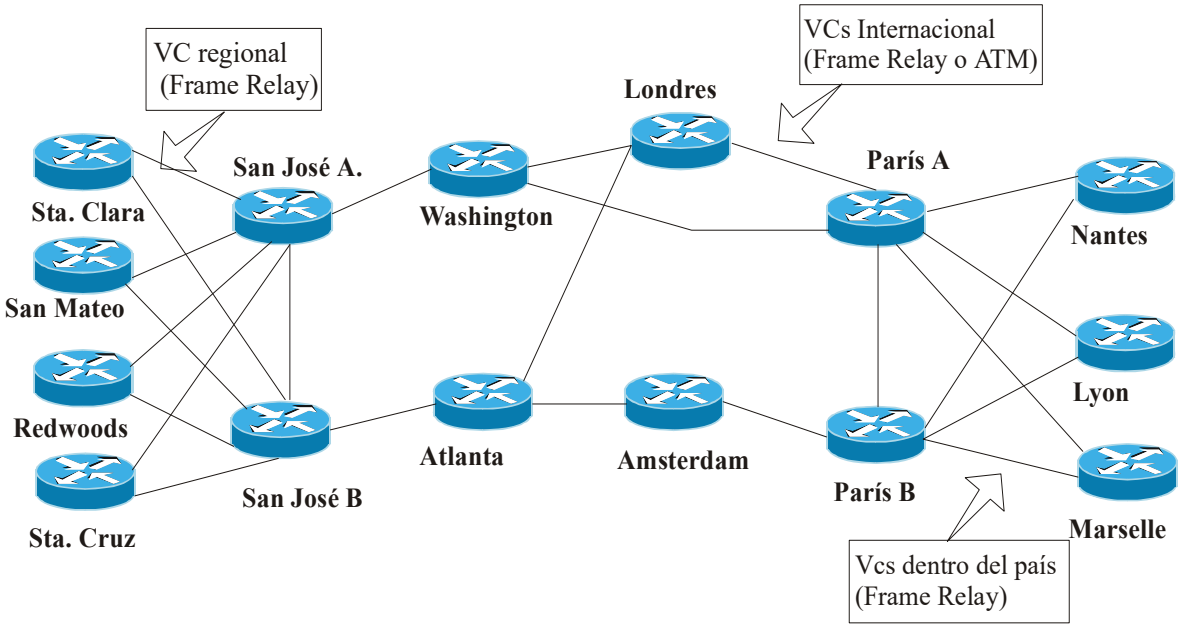


Figura IV.3.3.1.7

La mejor aproximación de un diseño de topología híbrida es siguiendo un diseño de red modular:

- Dividir la red extendida en núcleo, distribución y acceso
- Diseñar el núcleo y los accesos de cada red individualmente (por ejemplo, una topología *hub-and-spoke* doble con un respaldo conmutado en la red de acceso y una malla parcial en el núcleo de la red)
- Conectar la red núcleo y las redes de acceso a través de una capa de distribución tratando de aislarlas lo mejor posible. Por ejemplo, una falla dentro del *loop* local en algún lugar de la oficina remota no debe ser propagada dentro de la red núcleo. Los enrutadores de las oficinas remotas no deben ver las fallas de los enlaces internacionales.

IV.3.3.2. Topologías para VPNs Extranets

Topología simple extranet

Las topologías de intranet conciernen a la topología física y lógica de la red VPN, dictadas por la tecnología VC por la que el modelo VPN overlay es implementado. Las topologías extranet están más enfocadas a los

requerimientos de seguridad de la red VPN, la cual puede ser implementado con diferentes topologías, ya sea con el modelo overlay o *peer-to-peer*.

La topología tradicional extranet puede ser una extranet cualquiera permitiendo a cierto número de compañías intercambiar datos cualquiera-a-cualquiera (*any-to-any*). Los ejemplos pueden incluir comunidades de interés (compañías manufactureras de aeroplanos) o una cadena de proveedores (por ejemplo, una manufacturera de automóviles y sus proveedores).

Los datos en la extranet pueden ser intercambiados entre cualquier número de sitios. La extranet por sí misma no impone restricciones en el intercambio de datos. Usualmente, cada sitio es responsable de su propia seguridad, filtrado de tráfico y *firewalling*. La única razón para usar una extranet en lugar de una internet pública son las garantías de la Calidad de Servicio de sensibilidad de los datos intercambiados a través de la VPN, la cual sigue siendo más flexible a los ataques de la captura de datos que el internet genérico.

Si la extranet es implementada por el modelo VPN *peer-to-peer* (como el ejemplo de la extranet de la figura IV.3.3.2.1), cada organización sólo especifica cuánto tráfico va a ser recibido y enviado por cada uno de los sitios, es decir, la proporción de recursos en el lado del cliente y del Proveedor de Servicios es muy sencilla y efectiva.

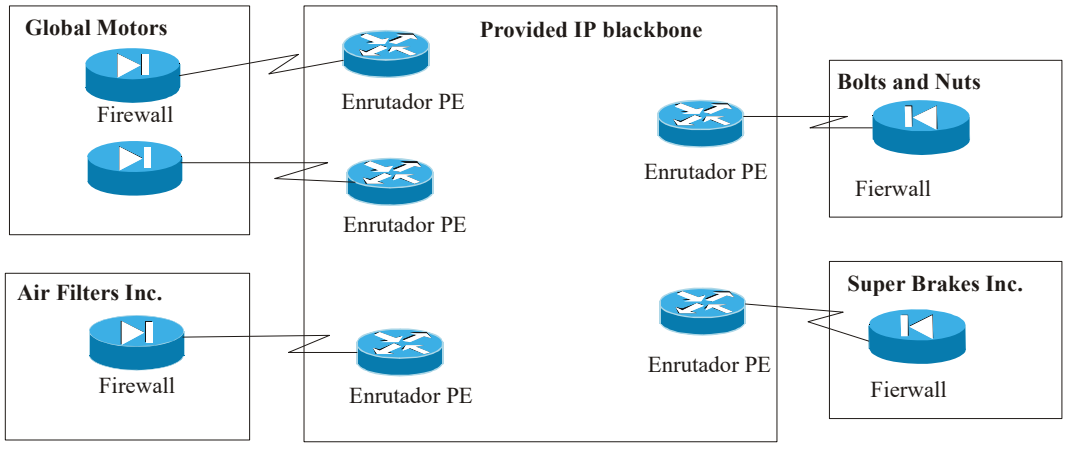


Figura IV.3.3.2.1

En el modelo VPN overlay, el tráfico entre sitios es intercambiado a través de VCs punto a punto (*point-to-point*) como lo muestra la siguiente figura:

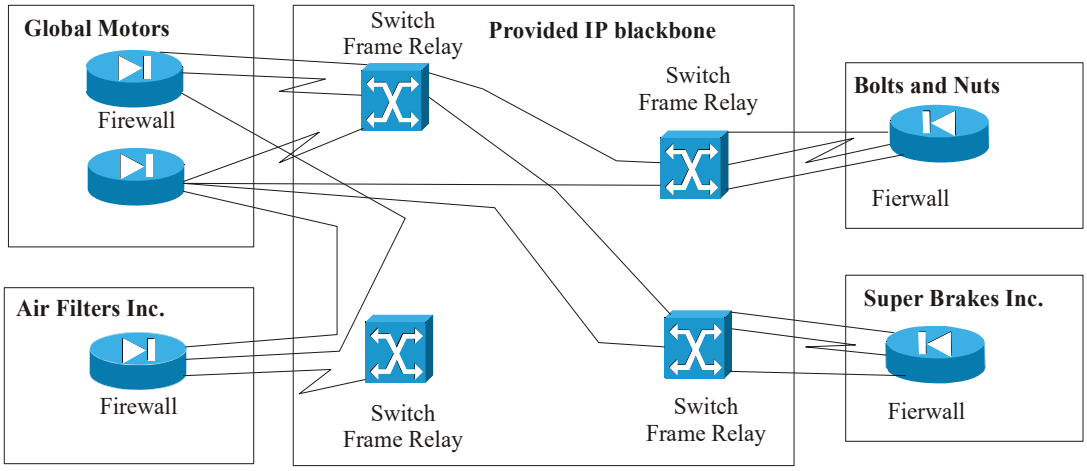


Figura IV.3.3.2.2

En la topología extranet similar a la de la figura anterior, cada organización participante paga por las VCs que usa. Obviamente, sólo se instalan los VCs más necesarios para minimizar costos. Además, los participantes en las VPNs podrían tratar de prevenir el tráfico de tránsito entre otros participantes siguiendo, a través de las VCs, aquellas que pagaron, usualmente resultando en una conectividad parcial entre los sitios de la extranet y, a veces, resultando en complejos problemas de enrutamiento. Por ello, el modelo de VPN *peer-to-peer* es el camino preferido para la implementación de una extranet *any-to-any*.

servicios centrales de extranet

Las extranets que enlazan organizaciones pertenecientes a la misma comunidad de interés, comúnmente son muy abiertas, permitiendo la conectividad *any-to-any* entre las organizaciones. Las extranets de propósito dedicado (por ejemplo, una red de administración de una cadena de proveedores enlazando a una gran organización con todos sus proveedores) tiende a ser más centralizada y permite la comunicación sólo entre

la organización, patrocinando la extranet y todos los demás participantes, similar al ejemplo de la figura siguiente:

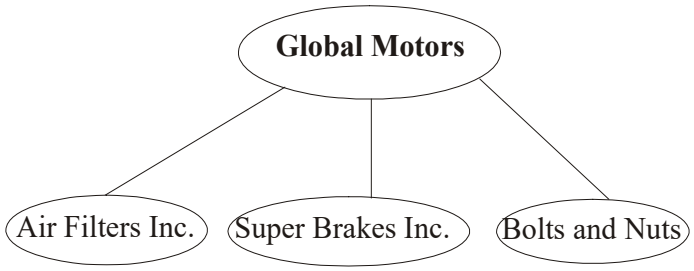


Figura IV.3.3.2.3

La seguridad en los Servicios Centrales de la extranet típicamente es proporcionada por la organización central que patrocina la extranet. Otros participantes con redes internas con misiones críticas (por ejemplo, bancos comerciales) también deberían permitir implementar sus propias medidas de seguridad (como por ejemplo, un *firewall* entre sus redes internas y la extranet)

Similar a cualquier otra red VPN, los Servicios Centrales de la extranet pueden ser implementados ya sea con el modelo VPN overlay o con el modelo *peer-to-peer*. Sin embargo, en este caso, el modelo *peer-to-peer* tiene desventajas definitivas debido a que el Proveedor de Servicios debe tener mucho cuidado en que los participantes de la extranet no puedan alcanzar a otro de ellos. Por el contrario, la implementación de los Servicios Centrales de la extranet sobre un modelo VPN overlay es extremadamente directo:

- Los VCs entre todos los sitios participantes y los sitios centrales son mantenidos y administrados. El tamaño de cada VC corresponde a los requerimientos de tráfico entre el sitio participante y el sitio central.
- El sitio central anuncia subredes disponibles únicamente del sitio central hacia sitios participantes.
- Los filtros del tráfico en el sitio central recibidos por otros participantes para hacer seguro un problema de enrutamiento o los ataques útiles de robo de servicio no influyen en la estabilidad de la VPN.

Cuidando estos tres aspectos, la red VPN se la figura anterior (figura IV.3.3.2.3), se transforma en la topología de VCs de la figura que la precede (figura IV.3.3.2.2).

Bajo el modelo extranet *any-to-any*, la red de la figura IV.3.3.2.2 tendría un número limitado de VCs (resultando en una topología redundante *hub-and-spoke*) debido a los inconvenientes del costo. Bajo el modelo de los Servicios Centrales de la extranet, la misma VPN tendría el mismo número de VCs debido a restricciones de seguridad. Así, podemos observar que un número de requerimientos diferentes pueden ser dictados por una misma topología VC.

Una topología extranet un poco más compleja de Servicios Centrales puede contener cierto número de servidores dispersos a través de varios sitios y cierto número de sitios clientes que tienen acceso a esos servidores, similar a la figura IV.3.3.2.4. Los ejemplos típicos que pudieran requerir esta topología son las redes sobre IP, donde un número de usuarios tiene acceso a *gateways* comunes en diferentes ciudades o países pero no está permitido que se vean entre ellos.

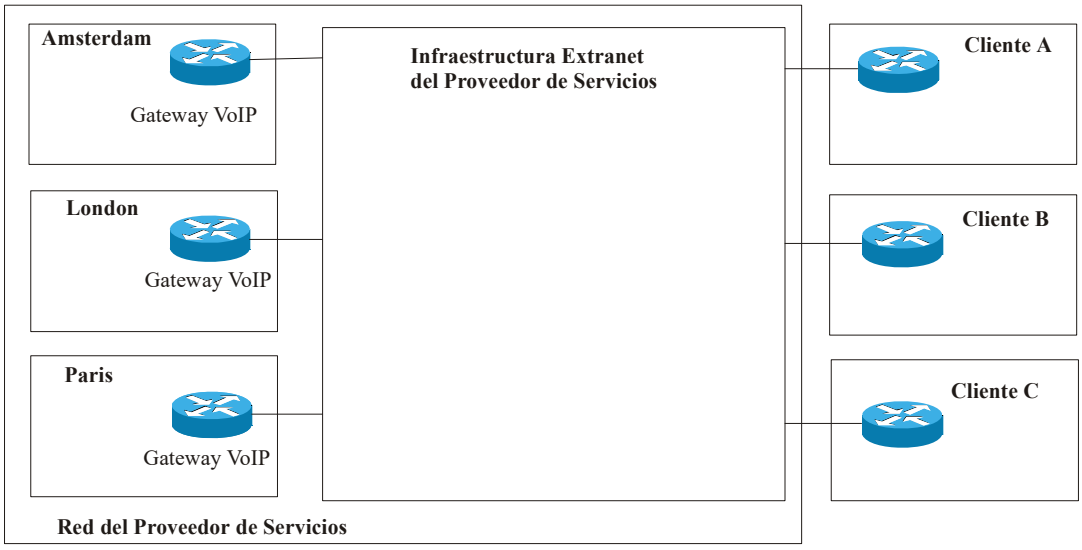


Figura IV.3.3.2.4

Una extranet puede ser implementada, ya sea con un modelo VPN *peer-to-peer* o con un modelo overlay. El número de VCs en el modelo VPN overlay (en el que un VC separado es necesario para cada sitio cliente y

para cada sitio del servidor) y la correspondiente complejidad de repartición de recursos usualmente evita el desarrollo de un modelo VPN overlay en estos escenarios. Una instalación más manipulable podría usar un modelo *peer-to-peer* o una combinación de ambos modelos, como se ilustra en la siguiente figura:

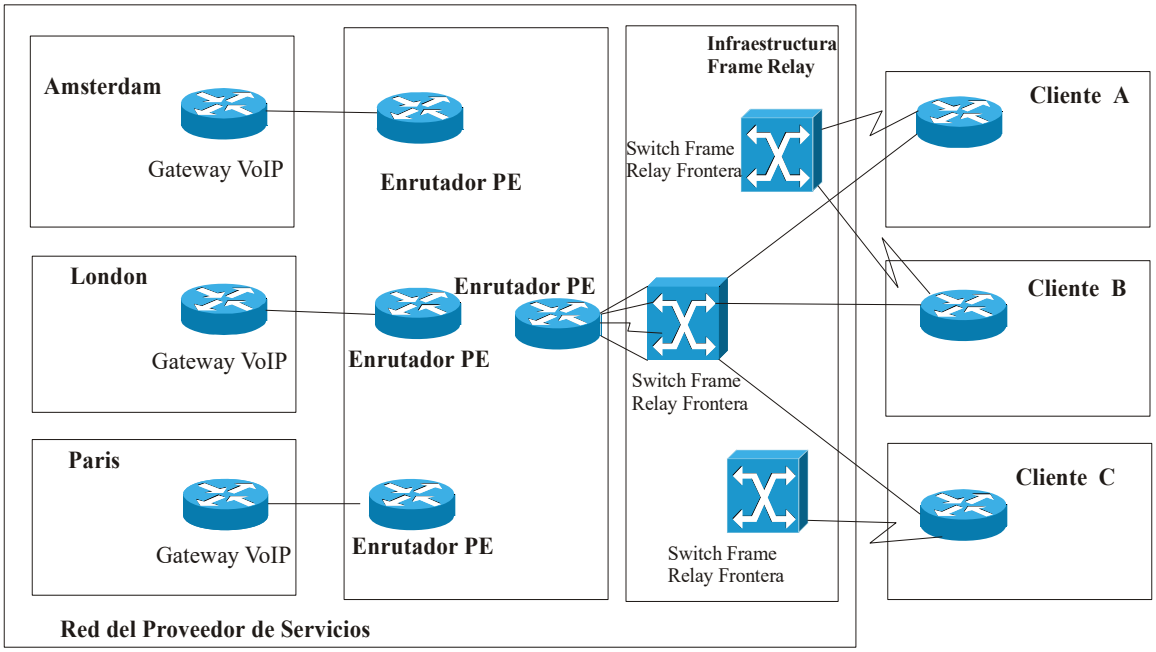


Figura IV.3.3.2.5

Lógicamente, la red en la figura anterior usa un modelo VPN *peer-to-peer* con enrutadores de distribución en el papel de los enrutadores PE (*Provider Edge*). La actual topología física difiere del punto de vista de la topología lógica: los enrutadores de distribución son enlazados con los sitios clientes (los enrutadors CE o *Customer Edge*) a través del modelo VPN overlay. Un ejemplo de esto, es la red Frame Relay.

IV.3.3.3 Topologías para VPDNs

Topología VPDN

Una VPDN (VPDN, *Virtual Private Dial-up Network*) es una VPN que usa como medio de conmutación una red de líneas telefónicas de par de cobre (PSTN) para establecer un enlace a la red corporativa que desea. El servicio de la Red Privada Virtual Conmutada usualmente se implementa usando túneles PPP para el

intercambio de *frames* PPP entre usuarios conmutados y sus *gateways* locales en paquetes IP intercambiados con el servidor de la red de acceso, como se muestra en la figura IV.3.3.3.1:

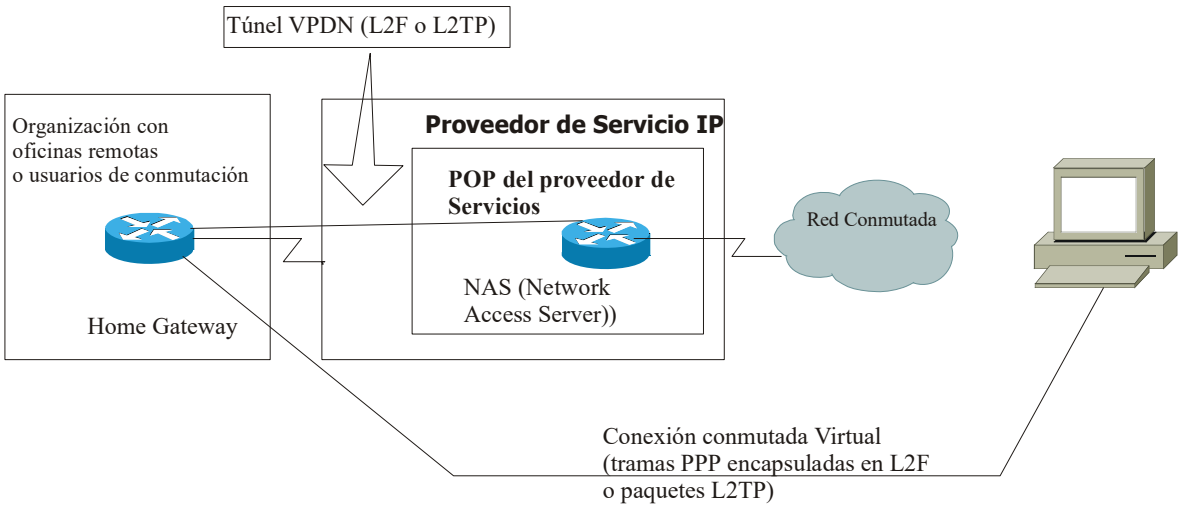


Figura IV.3.3.3.1

El usuario conmutado y el *gateway* local establece una conectividad IP (o IPx, AppleTalk, etc.) a través de enlaces por túneles PPP e intercambiando paquetes de datos a través de éstas. La figura que se muestra a continuación (IV.3.3.3.2) detalla la pila *o stack* del protocolo usado entre varias partes de la solución VPDN.

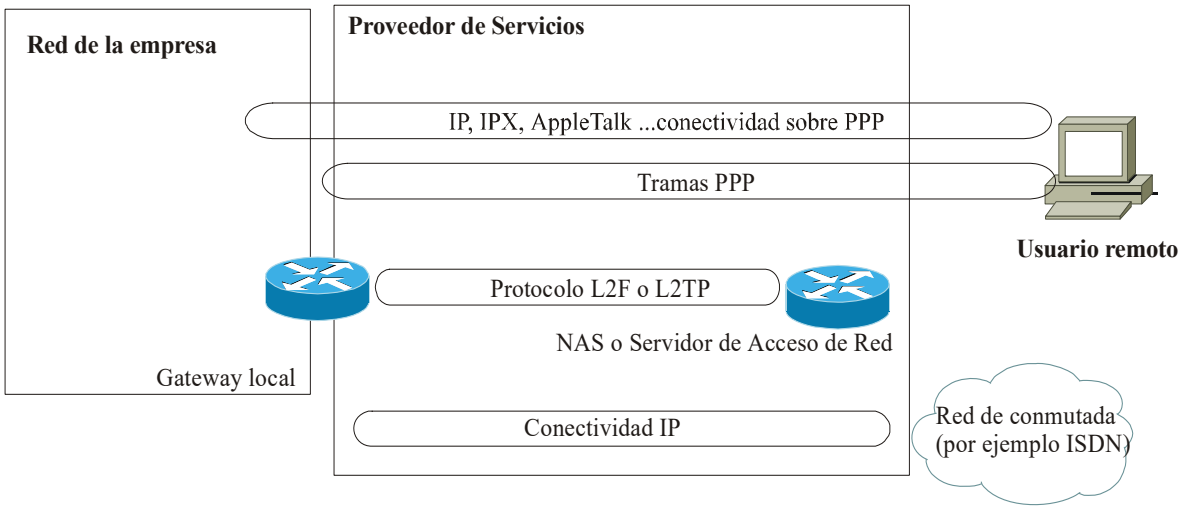


Figura IV.3.3.3.2

Cada solución VPDN requiere una infraestructura basada en IP para intercambiar datos por medio de túneles PPP entre el NAS (*Network Access Server*) y el *gateway* local. En el escenario más sencillo posible, el internet público puede ser usado como la infraestructura necesaria. Cuando los requerimientos de seguridad son más estrictos, una Red Privada Virtual puede ser construida para intercambiar *frames* PPP encapsulados. La estructura resultante para algunos diseñadores de redes resulta muy compleja, porque tratan de entender todo el escenario y todos los detalles al mismo tiempo, pero la complejidad puede ser reducida grandemente desarticulando las partes:

- El NAS y el *gateway* local utilizan sin diferencia la infraestructura IP que está disponible para intercambiar datos VPDN, la cual puede ser pensada como una aplicación ubicada en la parte más alta de la pila o *stack* IP. Consecuentemente, la estructura interna de la red IP no afecta el intercambio de los datos de aplicación y los contenidos de los datos de aplicación (paquetes IP en *frames* PPP encapsulados en una envolvente VPDN) no interactúan con los enrutadores que proveen los servicios IP
- La red basada en IP es efectivamente una extranet de Servicios Centrales con muchos sitios servidores (NAS, *Network Access Server*) y *gateways* locales actuando como sitios cliente. Esta infraestructura puede ser implementada en varias de maneras, desde modelo VPN overlay o con el modelo *peer-to-peer*.

Topología VPN de redes administradas

Este tipo de topología VPN es usada por los Proveedores de Servicios para administrar los enrutadores *customer-premises* (cliente-permisos) en un servicio de red administrada. Típicamente, como se muestra en la figura IV.3.3.3.3, el Proveedor de Servicios proporciona un número de enrutadores al sitio cliente, conectándolos a través de VCs implementados con Frame Relay o ATM y construidos con topologías *hub-and-spoke* separadas, conectando cada enrutador cliente con el Centro de Administración de la Red (NMC, *Network Management Center*).

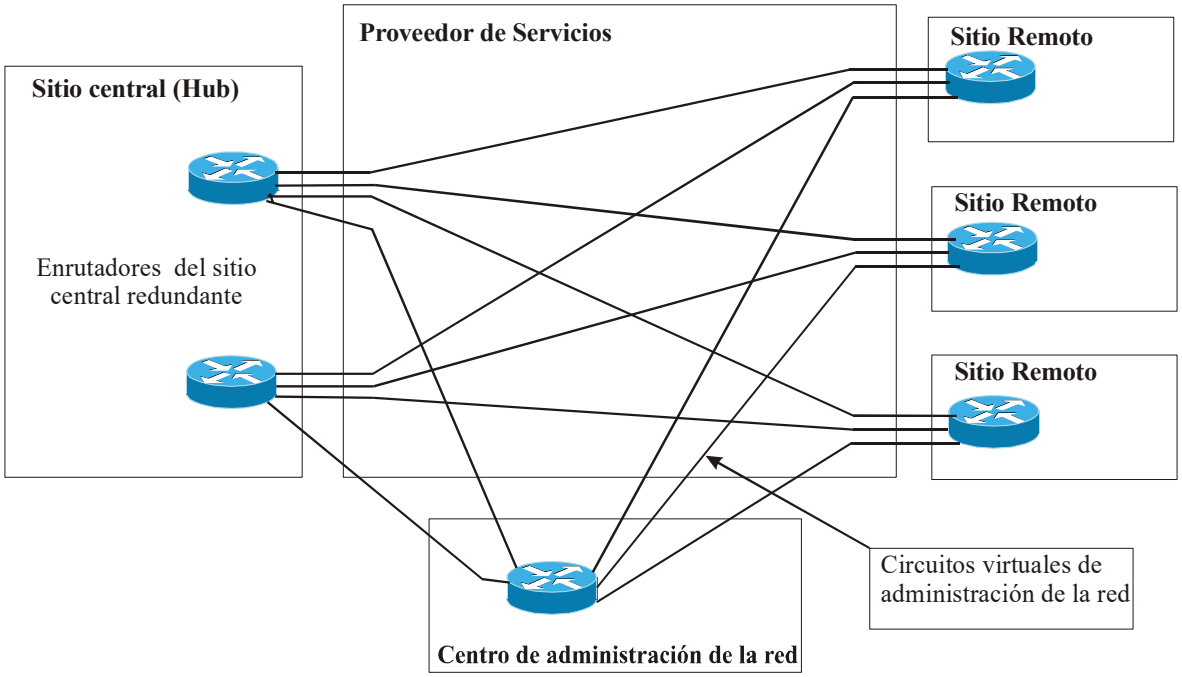


Figura IV.3.3.3.3

La topología VPN usada en la parte del cliente de la red puede ser cualquier topología que soporte el modelo VPN, yendo desde la topología *hub-and-spoke* hasta la topología de malla completa o *full mesh*. La topología usada en la parte de la red que administra la CPE efectivamente puede ser una topología extranet de Servicios Centrales con enrutadors actuando como clientes y NMCs (*Network Management Center*) siendo el punto central de la administración de la extranet.

Como ya se explicó, la topología de Servicios Centrales de Extranet es más sencilla de implementar con una topología *hub-and-poke* del modelo VPN overlay, lo cual también explica porqué la mayoría de los proveedores de servicios de redes administradas usan el arreglo de la figura IV.3.3.3.3.

La topología de Red administrada también puede ser implementada con varias tecnlogías VPN *peer-to-peer*, a pesar de que no es tan sencillo como con el modelo VPN overlay.

IV.4. Operación de las VPNs

Los factores principales que componen a una VPN son *el tuneleo o tunnelling* y los servicios de seguridad. En esta sección se describirá el aspecto del tuneleo y en la siguiente, se explicará la seguridad en una VPN.

[2]Existen dos tipos comunes de uso de VPNs: las VPDNs y las VPNs que se enlazan sitio a sitio. Una VPDN es una conexión de acceso remoto usuario-a-LAN usada por una compañía que tiene empleados que necesitan conectarse a su red privada desde lugares remotos. Típicamente, una corporación que desea establecer varios accesos remotos a su VPN proporciona a sus usuarios algún número para entrar a Internet por medio de un ISP. Entonces, los usuarios remotos marcan tal número para ingresar a Internet y usan el *software* de los clientes de la VPN para tener acceso a la red corporativa. Un ejemplo de alguna compañía que necesita varios accesos remotos a una VPN podría ser una firma con cientos de vendedores en toda una ciudad. Los accesos remotos a la VPN permiten conexiones seguras y cifradas entre la red corporativa y sus usuarios remotos a través de un proveedor de servicios.

Con una VPN sitio-a-sitio, una compañía puede conectarse a varios sitios fijos a través de una red pública (como Internet) con el uso de equipo dedicado y un cifrado a gran escala de la información. Cada sitio necesita solamente de una conexión local a la misma red pública, con lo que se logra un ahorro considerable de líneas dedicadas. Las VPNs sitio-a-sitio pueden ser construidas entre oficinas de la misma compañía, o por ejemplo, de una oficina de la compañía hacia un tercer sitio para compartir una base de datos.

El tuneleo o *tunnellig* es el proceso de encapsular un paquete entero dentro de otro paquete y enviarlo sobre una red. El tuneleo por sí sólo no proporciona seguridad a la información, pues aún después de que un paquete haya sido encapsulado en otro, sigue siendo visible para el dispositivo que recibe la encapsulación. Los protocolos de cifrado usan el tuneleo como medio para transferir los datos cifrados a través de una red pública, por lo que son una parte esencial en una VPN[2].

El tuneleo es la característica que hace posible la construcción de una VPN. Este proceso oculta la arquitectura y la operación de las redes inmediatas (como el Internet) de los dispositivos o redes conectadas a una VPN, lo cual logra que la implementación de la VPN sea sencilla debido a que no es necesario conocer los detalles de cómo se interconectan las redes, y tampoco necesitan conocer la existencia de la VPN los dispositivos conectados a ella, exceptuando al *gateway* de seguridad. Esto significa que la VPN puede ser conectada, desconectada, modificada o reemplazada sin alteraciones a la LAN que esté directamente conectada a ella[5].

Existen dos tipos de túneles: los permanentes (o estáticos) y los temporales (o dinámicos). Los túneles estáticos limitan la aplicación de las VPNs y por ello no son usados en una configuración VPN. En cambio, los túneles dinámicos reducen la utilización del ancho de banda y el costo, pues sólo se emplean cuando se requiere.

El corazón del tuneleo es la encapsulación de paquetes LAN en otro paquete, lo que implica que los detalles de la VPN no son importantes para la LAN y viceversa. La información que se encapsule depende de la capa en la cual opere la VPN. La encapsulación en capa 2 ó capa 3 es común en los protocolos VPN. Paja ilustrar la encapsulación tenemos:

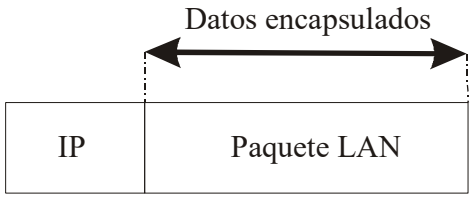


Figura IV.4.1

La capa en la cual tome lugar la encapsulación depende de la configuración de las redes directamente conectadas y de los dispositivos que situados en los extremos del túnel. Por ejemplo, si dos redes LAN que se conectan a través de una VPN tienen diferentes protocolos de capa 2, pero su protocolo de capa 3 es el mismo, se utiliza el protocolo de capa 3 para la encapsulación. Alternativamente se podría usar el protocolo de capa 2 para encapsular, pero sería necesario emplear un dispositivo de interconexión (tal como un

enrutador) que tradujera la información entre los dos protocolos de la capa de enlace de datos. Sin embargo, es importante considerar el flujo extra en un análisis para determinar cuál protocolo sería empleado para la encapsulación, pues mientras más baja sea la capa del protocolo, más grande es el encabezado y el *trailer* del paquete encapsulado, resultando en más flujo extra[5].

Con lo anterior, se puede destacar que una de las características más importantes de la arquitectura de una VPN es la capacidad de comunicar redes diferentes que usen distintos protocolos de capa de enlace de datos, pues la VPN se comporta similar a un enrutador en ese sentido, traduciendo ambos protocolos para la comunicación. Un ejemplo sería una VPN que conectara a una red LAN Ethernet y a una red LAN FDDI, ambas con el mismo protocolo de la capa de red. Cuando los datos se transmiten sobre la VPN, sólo la información de la capa 3 (como IP o IPX) puede ser intercambiada y cuando dicha información alcance su destino, es enviada para ser procesada en la capa 2. La encapsulación se realiza en el paquete entero, incluyendo los encabezados y las colas de las redes interconectadas[5].

Para tunelear un paquete, se necesitan tres diferentes protocolos:

- **Protocolo pasajero**: en el cual los datos originales son transportados (como por ejemplo, IP, IPX o NetBEUI)
- **Protocolo de encapsulación:** con el cual se "envuelve" al paquete original (como por ejemplo, GRE, IPsec, L2F, PPTP, L2TP)
- **Protocolo portador:** que es usado por la red sobre la cual se transporta la información

El paquete original (en el protocolo pasajero) es colocado dentro del protocolo de encapsulación, el cual a su vez es puesto dentro del encabezado del protocolo portador (como IP) para transmitirlo sobre una red pública. Algunos protocolos de encapsulación llevan a cabo el cifrado de los datos (como L2TP, L2F, PPTP, IPsec, etc.), por lo que se le agrega seguridad a la transmisión. Para VPNs sitio-a-sitio, generalmente el protocolo de encapsulación es GRE o IPsec. En VPDNs , el tuneleo usualmente se lleva a cabo con PPP. Como parte del

stack de TCP/IP, PPP es el portador para otros protocolos IP en la comunicación entre el *host* y el sistema remoto. Además, PPP usa L2TP, L2F o PPTP para tunelear el paquete.

IV.5.Seguridad en las VPNs

[6]Hace cinco años, las redes estaban "cerradas" al tráfico externos, es decir, tenían un comportamiento parecido al que se muestra en la figura:

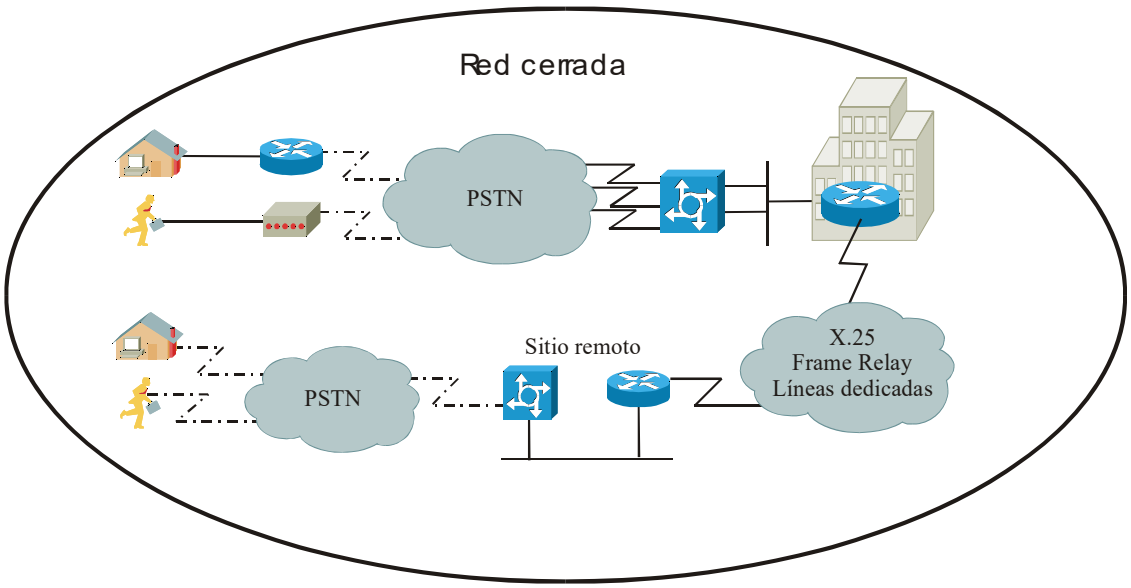


Figura IV.5.1

Actualmente, las redes se han "abierto", permitiendo el libre tránsito de paquetes a través de redes compartidas, teniendo un escenario similar al que a continuación se ilustra:

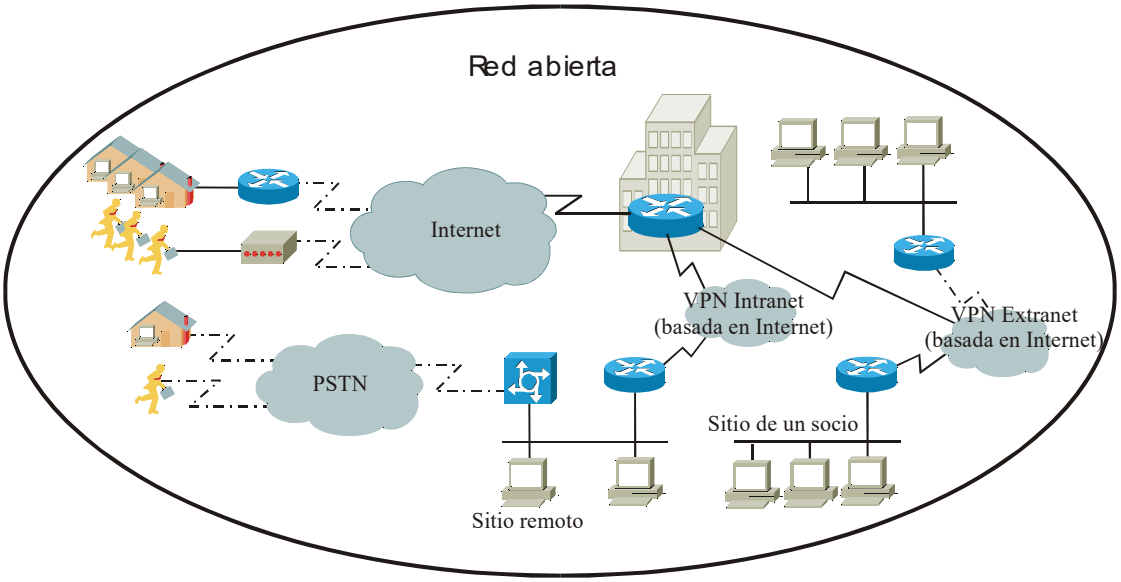


Figura IV.5.2

El papel de las redes ha cambiado en años recientes tomando cada vez mayor importancia, pues se han extendido a las áreas comerciales y financieras (entre otras), dando lugar a los *e-bussiness*. Debido a que son cada vez más las empresas que utilizan las redes de datos para concretar operaciones, ya sean muy sencillas (como la confirmación de una cita, por ejemplo) o de vital importancia (como muestra, una fuerte transacción), la seguridad de los datos es un tema primordial para cualquier administrador de una red. Se busca principalmente que la comunicación en una red sea segura aún en ambientes abiertos y un proceso continuo que se establezca una fuerte política de seguridad y una administración centralizada. Como ya se explicado, las VPNs pueden ser una buena opción en el transporte de datos confidenciales.

[4]Las empresas que tengan VPNs deben asegurarse de que éstas están a salvo de observadores prohibidos que perpetren los datos confidenciales que se transportan sobre la VPN y debe protegerse de los usuarios no autorizados que ingresen a la red y tengan acceso a sus recursos. Para garantizar la seguridad en una VPN, se deben cuidar cuatro aspectos importantes:

- Los túneles y el cifrado de datos
- Verificación de los paquetes

- *Firewalls* y detección de invasiones a la red
- Verificación del usuario

Estos mecanismos se complementan unos con otros, proporcionando seguridad en diferentes puntos de la red. [5]Una solución VPN debe ofrecer cada una de estas características de seguridad para ser considerada una solución viables para utilizar la infraestructura de una red pública. Un escenario ideal para una VPN, sería como se muestra en la figura IV.5.3:

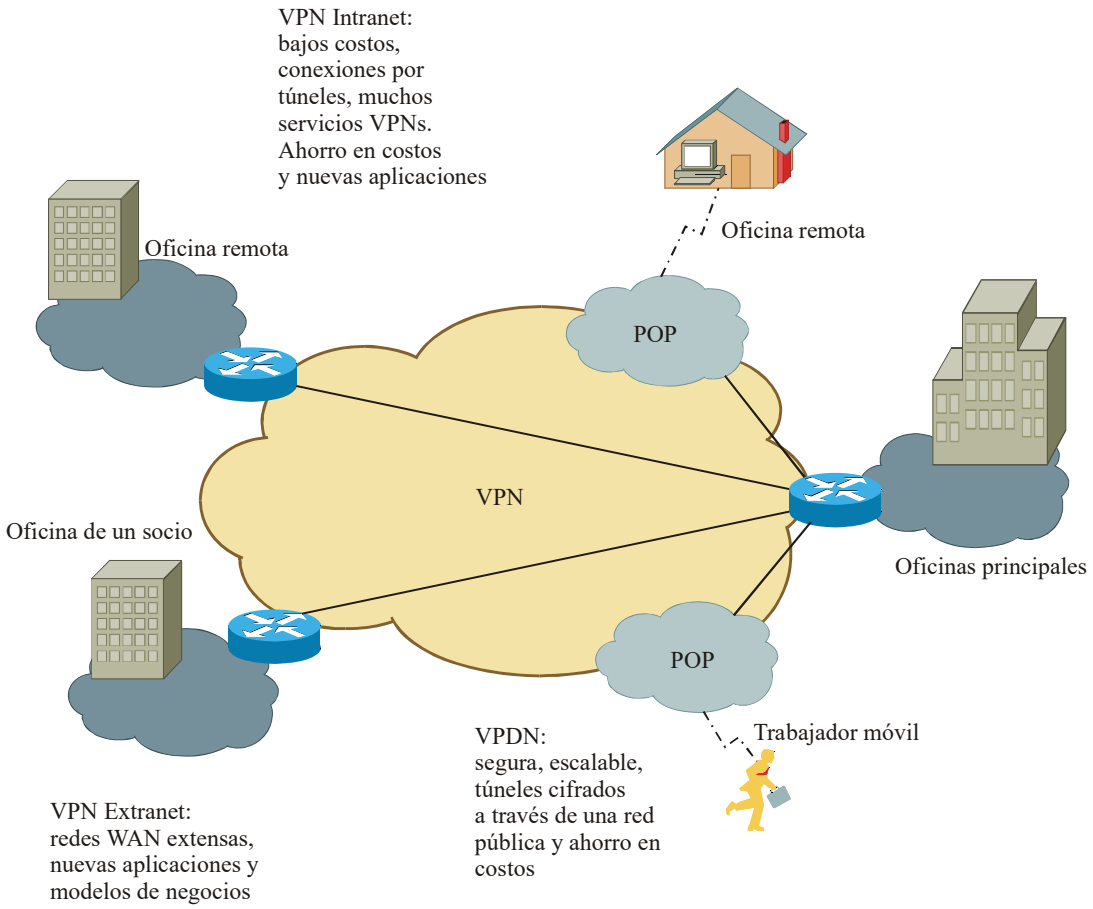


Figura IV.5.3

Túneles y cifrado de datos

Una solución VPN emplea túneles cifrados (con múltiples protocolos de encapsulación si fuera necesario) para proteger a los datos de ser interceptados y vistos por entidades no autorizadas. Los túneles proporcionan conexiones lógicas punta a punta a través de la red IP no orientada a conexión, habilitando la aplicación ed avanzadas características de seguridad. El cifrado se aplica a la conexión tuneleada para desordenar los datos y así hacerlos legibles sólo a los transmisores y receptores autorizados. En las aplicaciones en las que la seguridad no es un asunto preocupante, los túneles se pueden emplear sin cifrar los datos para proporcionar soporte sin seguridad a múltiples protocolos[4].

Verificación de los paquetes

A pesar de que la intervención de los datos en una red compartida es la preocupación principal para las empresas, la integridad de los datos también es importante. En una red insegura, los paquetes pueden ser interceptados y los datos pueden sufrir alteraciones para ser posteriormente enviados a su destino con información errónea. Por ejemplo, una factura ordenada al proveedor de una empresa que sea enviada sobre una red insegura, puede ser modificada cambiando la cantidad de la orden de 1000 a 100.

La verificación de los paquetes los protege contra intromisiones al aplicar encabezados al paquete IP con el fin de asegurar su integridad. Los componentes de IPsec, como *Authentication Header* (AH) y *Encapsulation Security Protocol* (ESP) son usados en conjunto con algoritmos *hash* (o de desorden) estándares, tales como MD5 (*Message Digset* 5), para garantizar la integridad de los paquetes transmitidos sobre un *backbone* IP compartido[4].

Firewalls y detección de invasiones a la red

En una aplicación VPN, los *firewalls* protegen a la empresa de accesos no autorizados y de ataques a su red, simplemente negando el acceso a la VPN. Además, para el tráfico autorizado, los *firewalls* verifican el origen de los datos y privilegia el acceso a los usuarios que son aceptados.

Un elemento de adicional de seguridad es la detección de invasiones. Mientras los *firewalls* permiten o niegan el paso de tráfico según su origen, destino, puerto y otros criterios, no analizan su contenido. Los sistemas de detección de invasión operan conjuntamente con los *firewalls* para extender el perímetro de seguridad al paquete, pues se analizaría el contenido y el contexto de cada uno de ellos en forma individual y así se determinaría si el tráfico es autorizado. Si la cadena de datos en una red experimenta actividad no autorizada el software de detección de invasiones automáticamente aplica políticas de seguridad en tiempo real, como por ejemplo, la desconexión de la sesión, y notifica al administrador de la red del incidente.

El uso de *firewalls* y del *software* de detección de invasiones proporcionan fuertes mecanismo de defensa contra los ataques a la red, pero una seguridad fuerte comienza adentro de la compañía al minimizar desde adentro las vulnerabilidades de la seguridad. Existen sistemas de auditoría de seguridad que revisa toda la red e identifica riesgos potenciales[4].

Verificación del usuario

Un componente clave de la seguridad de las VPNs es la garantía de que sólo los usuarios autorizados tengan acceso a los recursos de la empresa, mientras que los usuarios no autorizados son bloqueados totalmente para entrar a la red. Las soluciones a este problema son la verificación, autorización y contabilización de los accesos, de tal forma que la verificación de los usuarios determina el nivel e acceso y archiva toda la información necesaria de la contabilización de los datos[4]. Estas capacidades se encuentran reunidas en un servidor AAA (*Authentication, Authorization, Accounting*), el cual es usado para incrementar la seguridad en

un ambiente VPDN. Sin la verificación del usuario, cualquier computadora preconfigurada como un cliente de la VPN puede establecer una conexión segura en la red remota. Sin embargo, con la verificación de usuario, su nombre y su contraseña tienen que darse aún antes de que la conexión se complete. Los nombre de usuarios y las contraseñas pueden ser almacenados en un dispositivo terminal de la VPN o en un servidor AAA, que puede proveer verificación a numerosas bases de datos, tales como Windows NT, Novell, etc.

Cuando se solicita el establecimiento de un túnel, el dispositivo VPN pide el nombre de usuario y la contraseña. Esto puede ser verificado localmente o enviado a un servidor externo AAA, en el que se verifica lo siguiente:

- ¿quién eres? (Verificación)
- ¿qué tienes permitido hacer? (Autorización)
- ¿qué Qué estás haciendo? (Contabilización)

La información e contabilización es especialmente útil para propósitos de reportes, facturación o auditoría[2].

Referencias

[1] MPLS Technologies

[2] How Virtual Private Networks Work. http://www.cisco.com/warp/public/471/how_vpn_works.pdf

[3] Understanding VPDN. http://www.cisco.com/warp/public/471/vpdn_20980.html

[4] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/vpn.pdf

[5] http://www.eng.uts.edu.au/~kumbes/ra/vpn/vpn02a.htm#VPN%20History

[6] Seminarios Tecnológicos Cisco Systems. Cisco SAFE: Solución Integral de Seguridad y VPNs

Capítulo IV Redes Privadas Virtuales

IV.1. Introducción

Las Redes Privadas Virtuales o VPN (*Virtual Private Network*) son una alternativa a la infraestructura WAN que está reemplazando a las redes privadas actualmente existentes por medio de líneas rentadas, redes Frame Relay o ATM propias de una empresa. Una VPN puede utilizar cualquier tecnología de transporte disponible, ya sea el Internet público, o los *backbones* o las redes Frame Relay o ATM de un proveedor de servicios. Sin embargo, la funcionalidad de una VPN está definida principalmente por el equipo instalado en la frontera de la red corporativa y de la integración de las características a través de la WAN y no del protocolo de transporte WAN. [1]

Las VPNs pueden conectar a usuarios remotos fijos, a usuarios remotos móviles y a oficinas remotas pequeñas con la red corporativa o, inclusive, puede lograr la conexión de dos empresas socias con el fin de compartir información. Según la VPN que se implemente, se tienen que considerar diferentes características de seguridad y de administración del ancho de banda.

IV.2. Descripción

Una Red Privada Virtual (VPN) está definida como una red en la cual la conexión del cliente entre múltiples sitios está desarrollada sobre una infraestructura compartida con las mismas políticas de acceso y seguridad de una red privada[2].

Una solución VPN se define según la extensión de las características ofrecidas. Una VPN debe ser segura de ataques a la información, asegurar la entrega confiable de datos en situaciones críticas y ser fácilmente administrable para la empresa. Una solución VPN tiene ciertos componentes[1]:

- **El proveedor de servicios**, que es una organización dueña de la infraestructura (el equipo y el medio de transmisión) que proporciona líneas dedicadas a sus clientes. El proveedor de servicios ofrece al cliente un servicio de Red Privada Virtual.
- **El cliente**, quien se conecta a la red del proveedor de servicio a través del "Equipo de Permiso a Clientes" (*CPE, Customer Permises Equipment*),. El dispositivo CPE es también llamado CE (*Customer Edge*). El cliente es quien paga por los servicios de una VPN.
- **El dispositivo CE**, el cual es conectado a través del medio de transmisión (usualmente una línea dedicada, pero también puede ser una conexión de línea conmutada) hacia el equipo del proveedor de servicios, el cual puede ser un conmutador X.25 o Frame Relay o ATM, o incluso un enrutador IP. El CE es generalmente un dispositivo ensamblador y desensamblador de paquetes (*PAD, Packet Assembly and Disassembly*) que provee la conectividad del puente o el enrutador en la terminal.
- **El dispositivo PE**, que es la frontera del proveedor de servicio. El acrónimo es por *Provider Edge*. El PE interconecta a los dispositivos CE con la red del proveedor de servicios.
- **Dispositivos P**. l proveedor de servicio frecuentemente tiene un equipo adicional en el corazón de su red (conocida también como *P-Network* o *Red-P*). Estos dispositivos son llamados Dispositivos-P (*P-Devices*), y como ejemplo están los Enrutadores-P (*P-Enrutadors*) o Conmutadores-P (*P-Switches*).
- **El sitio**. La parte contigua a la red del cliente (*Costumer Network*) es llamada "sitio" (site). Uno se puede conectar a la Red-P (*P-network*) a través de una o varias líneas de transmisión, usando uno o varios dispositivos CE y PE, según los requerimientos de redundancia.
- **Las líneas dedicadas**, las cuales son proporcionadas al cliente por el proveedor de servicios sobre el modelo VPN (*Virtual Private Network*).recuentemente son llamadas VC (*Virtual Circuit*) o "Circuito Virtual". El VC (*Virtual Circuit*) puede estar disponible todo el tiempo (como los Circuitos Virtuales Permanentes o *Permanent Virtual Circuit, PVC*) o bien, puede estar establecido bajo demanda (*Switched*

Virtual Circuit SVC o Circuito Virtual Conmutado). Algunas tecnologías usadas en términos especiales para las VC's, son, por ejemplo, *Data Link Connection Identifier* (*DLCI*) para Frame Relay.

- **La tasa de tráfico.** El proveedor de servicios puede dar una tasa plana para un servicio VPN, el cual normalmente depende del ancho de banda disponible para el cliente, o usar una tasa que dependa del uso, la cual puede depender del volumen o de la duración de los datos intercambiados.

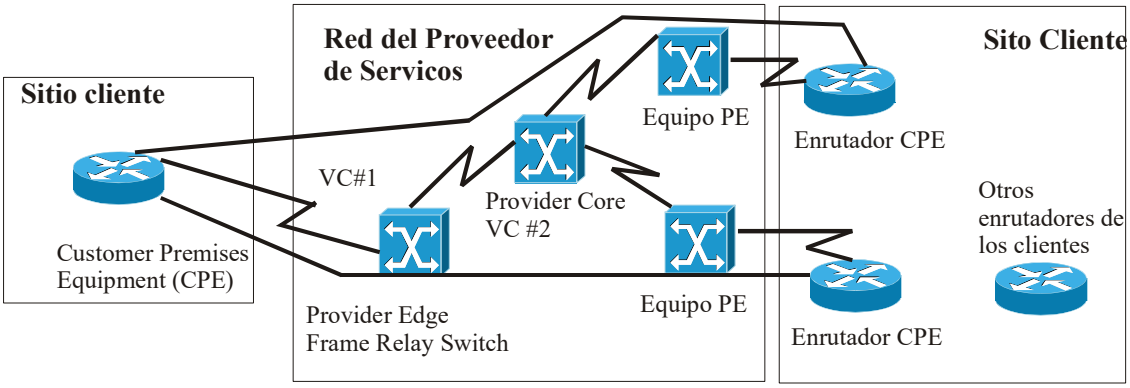


Figura IV.2.1

Una VPN típica debe tener una red LAN principal en los edificios de la compañía, otras LANs en oficinas remotas y usuarios individuales que se conectan remotamente. Básicamente, una VPN es una red privada que usa una red pública (usualmente Internet) para conectarse a sitios remotos, pero en lugar de usar líneas dedicadas, la VPN emplea conexiones virtuales enrutadas a través del Internet desde la LAN principal, hasta el sitios remoto[2].

Los elementos esenciales de una VPN deben tener ciertas características[4]:

- Escalabilidad en la plataforma
- Seguridad
- Confiabilidad
- Administración
- Políticas de seguridad

En general, una VPN ofrece más ventajas que las redes basadas en líneas dedicadas, teniendo cuatro principales:

- Costos más bajos que en las redes privadas: el costo total se reduce con el bajo costo del transporte de los datos, del ancho de banda, del equipo para el *backbone* y de operaciones.
- Las VPNs tienen inherentemente arquitecturas más flexibles y escalables que las clásicas redes WAN, además de que habilita a las empresas para extender su conectividad rápidamente y a costos efectivos y facilita la conexión o desconexión a oficinas remotas, locaciones internacionales, usuarios móviles o a redes de las empresas socias, según se requiera.
- Reducida administración de la carga en comparación con las redes privadas propias, pues las empresas pueden relegar la administración de su red a un proveedor de servicios y enfocarse más en los negocios que les competen.
- Topologías de red más simples, resultado de una reducida administración de la carga. El utilizar un *backbone* IP elimina los Circuitos Virtuales Permanentes (PVCs) asociados a los protocolos orientados a conexión, tales como ATM o Frame Relay, además de que crea una topología de malla completa que disminuye el costo y la complejidad de la red.

Además, una VPN bien diseñada puede ofrecer grandes beneficios a una empresa, como por ejemplo[2]:

- Conectividad en una extensa área geográfica
- Seguridad en la información
- Reducidos tiempos en el transporte de datos y bajos costos para los usuarios remotos
- Incrementa la productividad
- Simplifica la topología de la red
- Oportunidad de interconectarse globalmente

IV.3 Clasificación de las VPNs

[1]Como existe una gran variedad de tecnologías y topologías para VPN, la única manera de manejar exitosamente esta diversidad, es introduciendo una clasificación de VPNs, la cual se puede realizar de acuerdo a los siguientes criterios:

- **El problema del negocio de VPNs que se esté tratando de solucionar**. La mayoría de los problemas surgen en la comunicación entre una misma compañía (también llamada comunicación intranet), en la comunicación entre compañías (también llamada extranet) y en el acceso para usuarios móviles (mejor conocido como Red VPN *Dialup*).
- **La capa del modelo OSI** en la cual el proveedor de servicio intercambia la información de topología con el cliente. La mayoría de las categorías son modelos *overlay* (extendidos), donde el proveedor de servicios atiende al cliente únicamente con un conjunto de enlaces punto a punto (o multipunto) entre los *sites*, o son modelos "*peer-to-peer*" (vecino-a-vecino o igual-a-igual), donde el proveedor de servicios y el cliente intercambian información de enrutamiento de capa 3.
- **La tecnología de capa 2 o de capa 3 usada** para implementar el servicio VPN dentro de la red proveedora de servicio. la cual puede ser X.25, Frame Relay, SMDS, ATM o IP.
- **La topología de la red**, la cual puede ser desde una topología simple con un *hub* hasta una red con una malla completa o topologías multiniveles jerárquicas en redes más grandes.

Las Redes Privadas Virtuales (VPNs) pueden ser clasificadas en varias formas. La clasificación tecnológica más amplia es aquella basada en la forma en la que se intercambia información en la VPN. En el modelo VPN *peer-to-peer*, la información de enrutamiento del cliente es intercambiada entre los enrutadores del cliente y los enrutadores del proveedor de servicios. En el modelo VPN *overlay*, el proveedor de servicios proporciona únicamente VCs (líneas lógicas rentadas) y la información de enrutamiento es intercambiada directamente entre los enrutadores de los clientes en la frontera (enrutadores CE). Los dos modelos pueden ser combinados en una red más grande de Proveedor de Servicios: el modelo VPN *peer-to-peer* puede usar VPN *overlay* en sus partes de acceso (por ejemplo, enlace de los enrutadores del Proveedor de Servicios a través de ATM).

La clasificación más detallada de las VPNs, mostrada en la figura que se muestra abajo (figura IV.3.1), se enfoca en la tecnología de capa 3 que es usada para el transporte de paquetes sobre la VPN. El modelo VPN *overlay* puede ser implementado con tecnologías WAN de conmutación de capa 2 (como X.25, Frame Relay, SMDS o ATM) o con tecnologías de encapsulado de capa 3 (como IP sobre IP, IPsec). El modelo VPN *peer-to-peer* puede ser implementado tradicionalmente con complejos trucos de enrutamiento o con listas de acceso IP. El Multiprotocolo de Conmutación de Etiquetas, (mejor conocido como MPLS, *Multiprotocol Label Switching*) basado en VPNs, supera la mayoría de las fallas de otras tecnologías VPN *peer-to-peer*, permitiendo a los Proveedores de Servicios combinar los beneficios del modelo *peer-to-peer* (más sencillo de enrutar, implementación más sencilla de los requerimientos del cliente) con la seguridad y el aislamiento inherente del modelo VPN *overlay*.

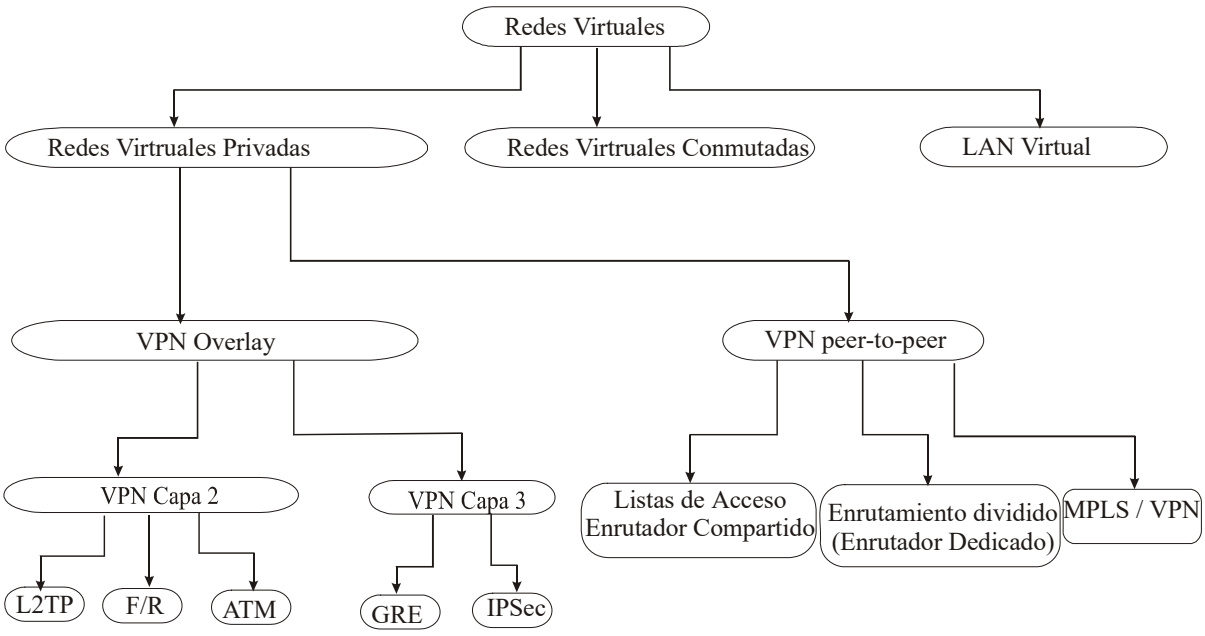


Figura IV.3.1

IV.3.1 Clasificación de VPNs basada en el problema a solucionar

Los tres problemas típicos de una empresa que se tratan de resolver con una red privada virtual (*Virtual Private Network*) son:

- Comunicación interna en la organización (intranet)
- Comunicación con otras organizaciones (extranet)
- Acceso de usuarios móviles, trabajadores, oficinas remotas y más a través de un medio de conmutación barato

Los tres tipos de soluciones con VPNs usualmente explotan la mayoría de las topologías y tecnologías ofrecidas por los Proveedores de Servicios VPNs, pero difieren grandemente en el nivel de seguridad requerido en su implementación.

Las comunicaciones internas en una organización frecuentemente no están bien protegidas por los *hosts* finales o por los *firewalls*. El servicio VPN usado para implementar la comunicación intranet además debe ofrecer altos niveles de aislamiento y seguridad. La comunicación intranet también requiere calidad de servicio garantizada para los procesos críticos. Éstas son las razones principales por las cuales existen muchas organizaciones que usen Internet para comunicarse. ya que éste no puede ofrecer calidad de servicio punto a punto, aislamiento o seguridad, así como una infraestructura adecuada para la comunicación intranet. Las Redes Privadas Virtuales o VPNs fueron implementadas comúnmente con tecnologías tradicionales, como X.25, Frame Relay o ATM.

Las comunicaciones extranet (o inter-organizacionales) frecuentemente toman lugar entre los sitios centrales de las organizaciones. Usualmente se usan dispositivos dedicados de seguridad, tales como *firewalls* o equipos de cifrado similares, tal como se muestra en la figura IV.3.1.1:

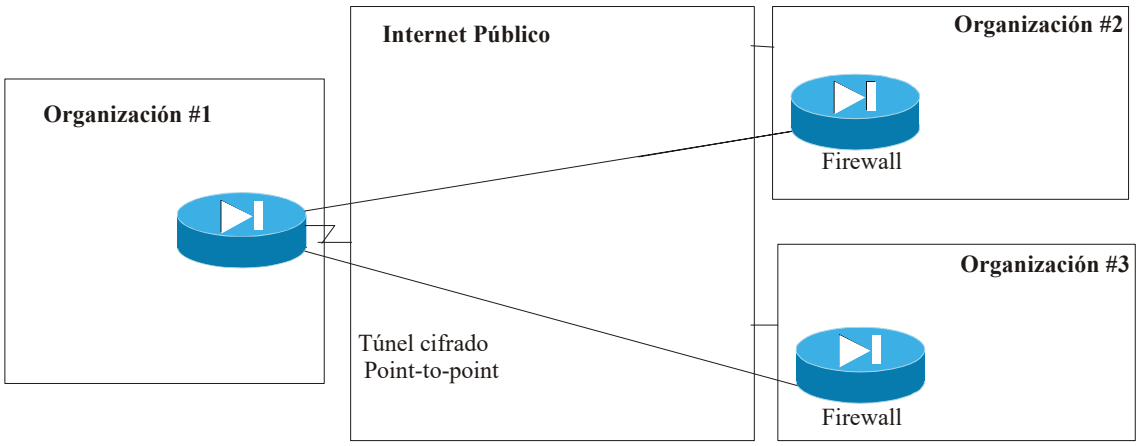


Figura IV.3.1.1

Este tipo de comunicaciones puede exigir también menos requisitos de Calidad de Servicio, por lo que el Internet tal vez sea más conveniente para las comunicaciones extranet; por lo tanto, no es una sorpresa que cada vez más y más tráfico de empresa a empresa tome lugar sobre Internet.

El acceso remoto del usuario dentro de una red corporativa, típicamente desde direcciones cambiantes o desconocidas, siempre filtrado con elementos de seguridad, obtenidos a lo largo del enlace punta a punta usando tecnologías de cifrado o una contraseña de un solo tiempo (*one-time password*). De esta manera, los requerimientos de seguridad para los servicios VPDN son tan rigurosos para las comunicaciones intranet que la mayoría de los servicios estén implementados actualmente sobre IP (*Internet Protocol*), sobre Internet o usando el *backbone* privado de un Proveedor de Servicios, tal como se muestra en la siguiente figura:

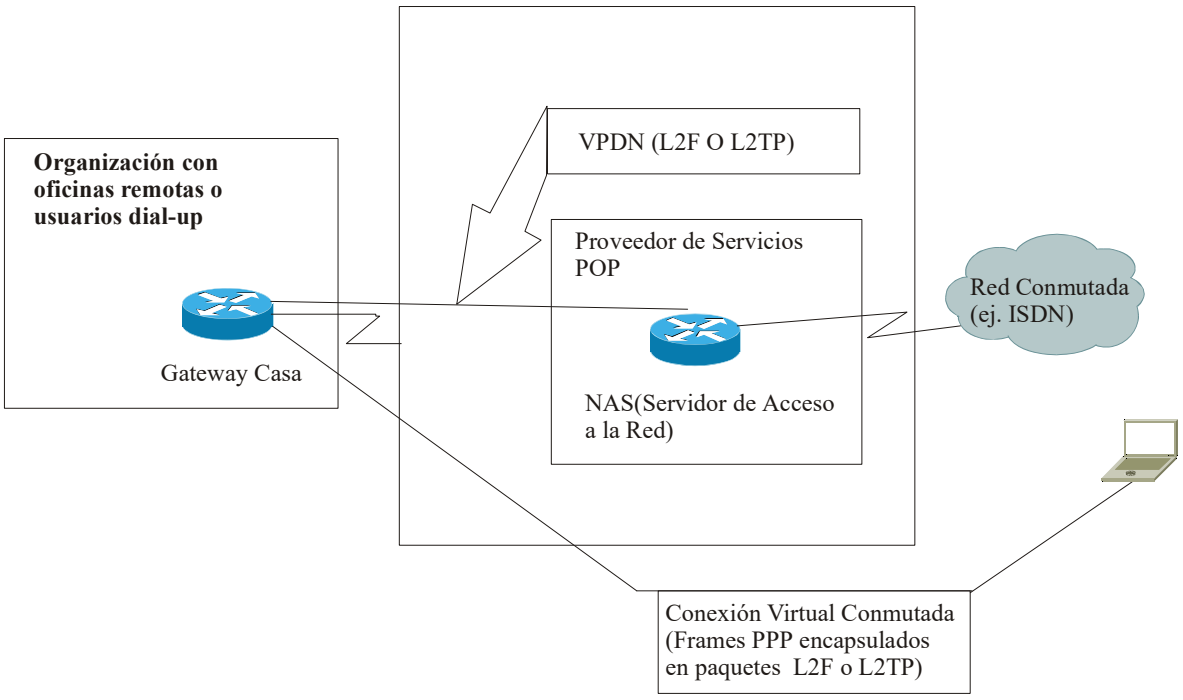


Figura IV.3.1.2

Los protocolos usados para implementar servicios de VPNs sobre IP incluyen L2F (*Layer 2 Forward*), PPTP (*Point-to-Point Tunneling Protocol*) o L2TP (*Layer 2 Trasnport Protocol*). La tecnología VPDN usa un número especial de términos que son únicos al mundo VPDN:

- ***Network Access Server* (NAS):** es el Servidor de Acceso Remoto (RAS, *Remote Access Server*) es administrado por el proveedor de servicios que acepta la llamada del cliente, llevando a cabo la verificación y enviando la llamada (por L2F o por L2TP) hacia el *gateway* del cliente.
- ***Home Gateway*:** es un enrutador administrado por un cliente que acepta la llamada enviada por el NAS, ejecuta una verificación y una autorización adicional y termina la sesión PPP del usuario. Los parámetros de la sesión PPP (incluyendo las direcciones de red, tales como la dirección IP) son negociadas entre el usuario y el gateway propio; NAS sólo envía paquetes PPP entre los dos.

IV.3.2 Modelos VPN *Peer-to-Peer* y Overlay

Los dos modelos de implementación que han expandido su uso son:

- El modelo overlay o extendido, en el cual el Proveedor de Servicios proporciona al cliente líneas rentadas emuladas.
- El modelo *peer-to-peer*, donde el Proveedor de Servicios y el cliente intercambian información de enrutamiento de capa 3 y el proveedor enruta los datos entre los sitios (o sites) de los clientes en la trayectoria óptima sin la participación del cliente.

Modelo VPN *Overlay*

El modelo VPN overlay es el más sencillo de comprender debido a que proporciona claramente la separación de las responsabilidades entre el cliente y el proveedor de servicios pues el Proveedor de Servicios proporciona al cliente un conjunto de líneas contratadas emuladas. Dichas líneas son llamadas VCs (Circuitos Virtuales), los cuales pueden estar disponibles permanentemente (PVCs) o establecidos por demanda (SVCs).

La siguiente figura muestra la topología de una VPN overlay y los circuitos virtuales usados en ella:

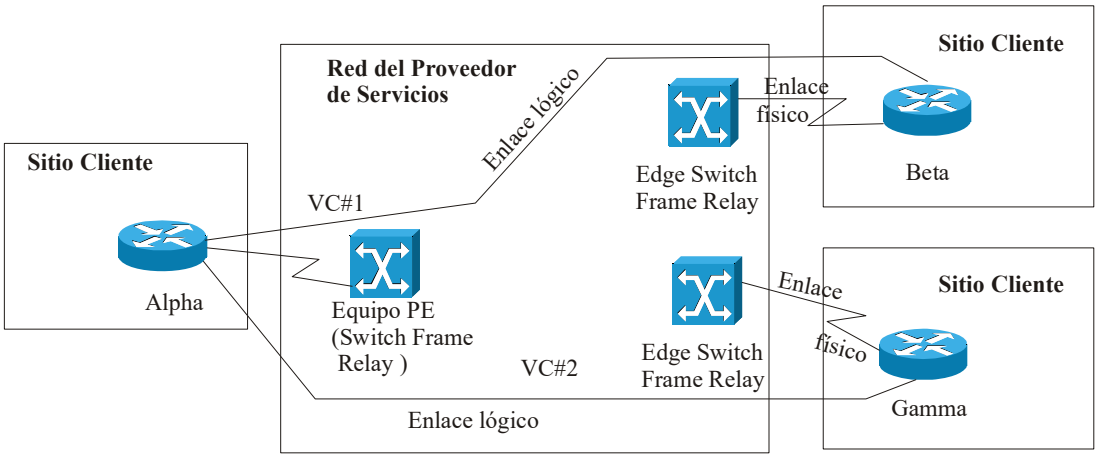


Figura IV.3.2.1

El cliente establece la comunicación enrutador a enrutador entre los equipos CE del cliente, dispositivos sobre los cuales se instauran los circuitos virtuales por medio del Proveedor de Servicios. Los datos del protocolo de enrutamiento siempre son intercambiados entre los dispositivos del cliente y el Proveedor de Servicios no tiene conocimiento de la estructura interna de la red del cliente. La siguiente figura muestra la topología de los enrutadores de la red VPN de la figura anterior:

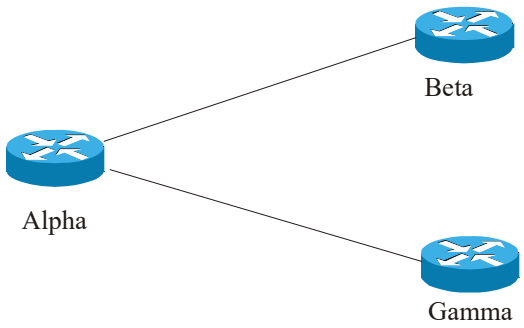


Figura IV.3.2.2

La Calidad de Servicio (QoS) garantiza que el modelo VPN overlay sea expresado comúnmente en términos del ancho de banda comprometido en un circuito virtual determinado (CIR, *Committed Information Rate*) y un máximo ancho de banda disponible en el circuito virtual (PIR, *Peak Information Rate*). La garantía del ancho de banda consignado es proporcionada usualmente a través de la naturaleza estadística de los servicios de capa 2, pero depende de la estrategia de sobreventa del proveedor de servicios. Esto significa que la tasa consignada no está realmente garantizada aunque el proveedor pueda proporcionar una Tasa Mínima de Información (MIR, *Minimum Information Rate*) que es obtenida a través de la infraestructura de capa 2.

La garantía del ancho de banda consignada es también una garantía del ancho de banda entre dos puntos en la red del cliente. Sin una matriz completa de tráfico para todas las clases de tráfico, es difícil para el cliente maniobrar las garantías en la mayoría de las redes overlay. También es difícil proporcionar las múltiples clases de servicio debido a que el Proveedor de Servicios no puede diferenciar el tráfico en medio de la red.

Trabajar así, creando múltiples conexiones (por ejemplo, en Frame Relay los PVCs) entre los sitios de los clientes, sólo incrementa el costo general de la red.

Las redes VPNs overlay pueden ser implementadas con un gran número de tecnologías de conmutación de WAN de capa 2, incluyendo a X.25, Frame Relay, ATM o SDMS. En los últimos años, las redes VPNs *overlay* también han sido implementadas con métodos de tuneleo de IP sobre IP, todos en *backbones* privados de IP sobre el Internet público. Los dos métodos más comunes de tuneleo IP sobre IP son *Generic Route Encapsulation* (GRE) y la encriptación con *IP Security* (IPSec).

A pesar de que es relativamente fácil de entender e implementar, el modelo VPN overlay tiene desventajas:

- Es apropiado para configuraciones no redundantes con pocos sitios centrales y sitios bastantes remotos, pero llega a ser extremadamente difícil de administrar en una configuración más compleja.
- La implementación propia de las capacidades del circuito virtual requiere un conocimiento detallado de los perfiles de tráfico de sitio a sitio, los cuales no siempre están disponibles.
- Cuando es implementado con tecnologías de capa 2, el modelo VPN overlay introduce otra capa innecesaria de complejidad en las redes *New World Service Provider* que, en su mayoría, están basadas en IP, lo que incrementa los costos operacionales de tal red.

Modelo VPN *Peer-to-Peer*

Este modelo fue introducido hace pocos años para superar las desventajas del modelo VPN overlay. En el modelo *peer-to-peer*, el dispositivo límite o de frontera (PE) del proveedor es un enrutador (PE-enrutador) que intercambia directamente la información de enrutamiento con el enrutador CE. La siguiente figura muestra un ejemplo de VPN *peer-to-peer*, la cual es equivalente a la figura ilustrativa del modelo VPN overlay.

El modelo *peer-to-peer* proporciona ciertas ventajas sobre el modelo overlay tradicional:

- El enrutamiento (desde el punto de vista del cliente) llega a ser extremadamente simple, ya que el enrutador CE del cliente intercambia información de enrutamiento con sólo uno (o unos cuantos) enrutadores PE (PE enrutador) mientras que en las redes VPN overlay, el número de enrutadores vecinos pueden crecer a un número grande.
- El enrutamiento entre los sitios del cliente es siempre óptimo, ya que el enrutador PE del proveedor conoce la topología de la red del cliente y puede también establecer un enrutamiento óptimo entre sitios.
- El suministro del ancho de banda es más simple, ya que el cliente tiene que especificar sólo el ancho de banda de entrada y de salida para cada sitio (tasa de acceso Comprometido o *Committed Access Rate*, CAR, y Tasa de Entrega Comprometida, *Committed Delivery Rate*, CDR) y no los perfiles de tráfico exactos de sitio a sitio.
- La adición de un nuevo sitio es sencilla ya que el Proveedor de Servicio acondiciona sólo un sitio adicional y cambia la configuración en el enrutador PE adjunto. En el modelo VPN overlay, el Proveedor de Servicio debe proporcionar todo el conjunto de VCs manejado desde este sitio hacia otros sitios de clientes de la VPN.

Hay dos opciones disponibles para el modelo de VPN *peer-to-peer*:

- Enrutador compartido, donde varios clientes VPN comparten el mismo enrutador PE.
- Enrutador dedicado, donde cada cliente VPN tiene un enrutador PE dedicado.

Modelo *peer-to-peer* con enrutador compartido

Cuando se tiene un enrutador compartido, varios clientes pueden ser conectados al mismo enrutador PE. Las listas de acceso deben de ser configuradas en todas las interfaces PE-CE en los enrutadores PE para asegurar la separación de los clientes de la VPN y para prevenir que un cliente de la VPN pueda irrumpir en otra red VPN, o también para prevenir que un cliente de una VPN no pueda atacar otra red VPN. La siguiente figura ilustra un ejemplo de la configuración de enrutador compartido.

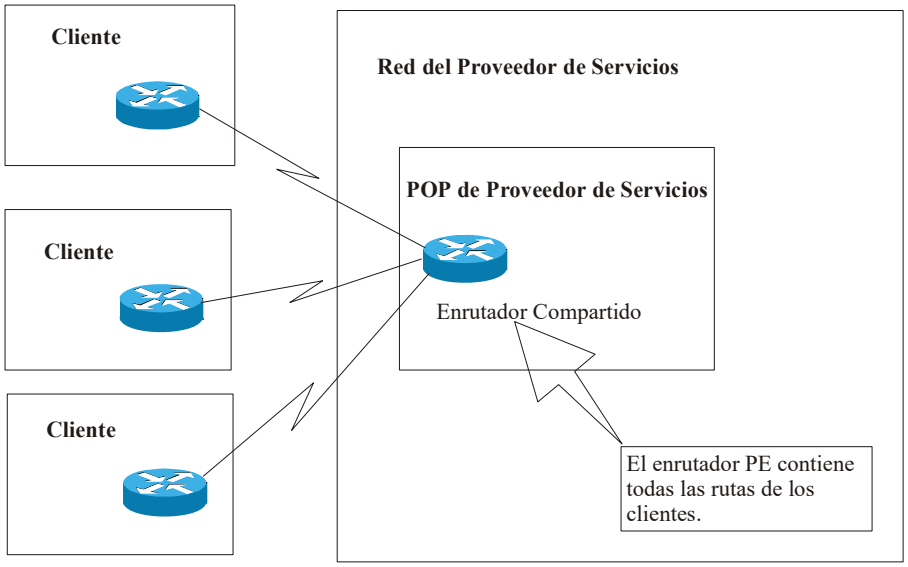


Figura IV.3.2.3

Modelo *peer to peer* con enrutador dedicado

En el modelo *peer to peer* con enrutador dedicado, cada cliente VPN tiene su propio enrutador PE dedicado (figura IV.3.2.4) y, sin embargo, sólo tiene acceso a los enrutadores que se encuentran en la tabla de enrutamiento del enrutador PE.

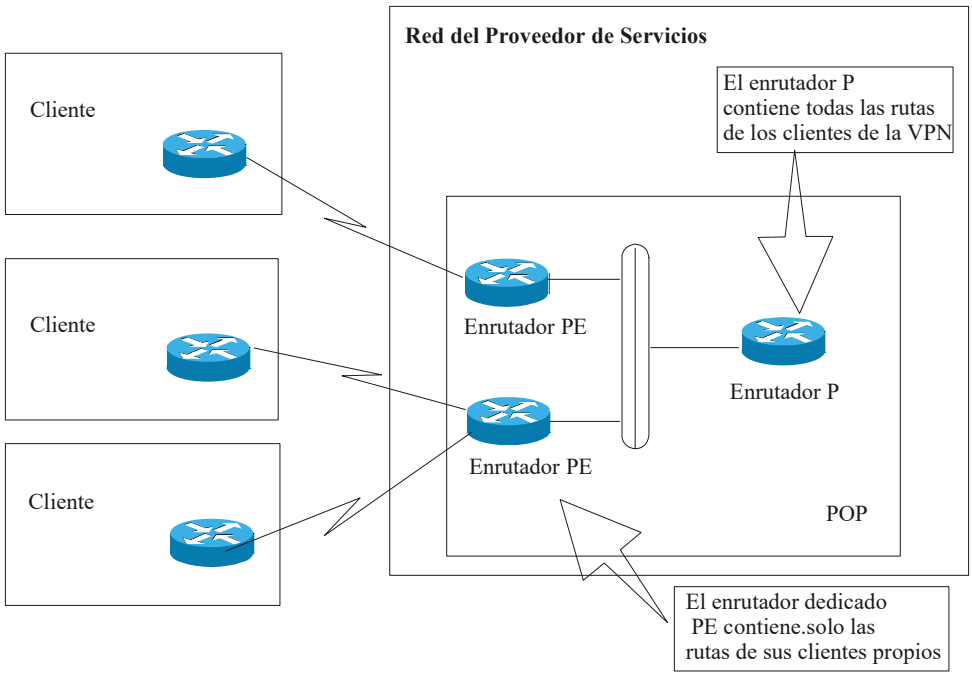


Figura IV.3.2.4

En el modelo de enrutador dedicado se usa un protocolo de enrutamiento para crear la tabla de enrutamiento de la VPN en los enrutadores PE. Las tablas de ruteo en un enrutador PE contienen sólo los enrutadores anunciados por los clientes VPN conectados a él, resultando en un aislamiento casi perfecto entre los clientes de la VPN (asumiendo que tiene que estar basado en una tabla de enrutamiento). Dentro de esta modalidad, el enrutador dedicado puede ser implementado como sigue:

- Cada protocolo de enrutamiento es ejecutado entre el enrutador PE y el enrutador CE.
- BGP es ejecutado entre el enrutador PE y el enrutador P
- El enrutador-PE redistribuye rutas recibidas desde el enrutador-CE encapsuladas en BGP, marcadas con una identificación del cliente (ID, comunidad BGP) y propaga las rutas a los enrutadores-P. De esta manera, el enrutador P contiene todas las rutas de todos los clientes VPN.
- Los enrutadores P sólo propagan rutas en comunidades BGP apropiadas por los enrutadores PE. Así, los enrutadores PE sólo reciben las rutas que se originaron desde los enrutadores CE dentro de la VPN.

Comparación de modelos *peer to peer*

El modelo *peer to peer* con enrutador compartido es muy difícil de mantener debido a que requiere el empleo de listas de acceso largas y complejas en casi cada interfaz del enrutador. El modelo de enrutador dedicado, aunque más sencillo de configurar y de mantener, llega a ser muy caro para el proveedor de servicios cuando trata de servir a un gran número de clientes con sitios geográficamente dispersos.

Ambos modelos comparten varias desventajas que evitan la expansión de su uso:

- Todos los clientes comparten el mismo espacio de direcciones IP, evitando que se usen direcciones IP privadas de acuerdo a la RFC 1918. Los clientes deben usar direcciones IP ya sean públicas o privadas para ser localizados por el proveedor de servicios.
- Los clientes no pueden insertar la ruta de *defalut* en su VPN. Esta limitación evita que tengan acceso a internet por medio de otro Proveedor de Servicios.

Además de estas dos ventajas, el modelo de enrutador compartido sufre de complejidad cuando varios clientes usan protocolos de enrutamiento (RIP, RIPv2, BGP y IS-IS) en donde existen varias instancias pero no son soportados por el *software* del enrutador.

IV.3.3. Topologías típicas de redes VPN

La topología VPN necesaria por una organización debería ser dictada por el problema que la organización está tratando de resolver, sin embargo, varias topologías conocidas son empleadas tan frecuentemente que serán discutidas a continuación. Como se podrá observar, una misma topología soluciona una gran variedad de diferentes contratiempos en diferentes niveles del mercado o la industria.

Las topologías más comunes se pueden dividir en tres categorías:

- Topologías influenciadas por el modelo VPN overlay o extendido, las cuales incluyen la topología *Hub-and-Spoke*, malla parcial o completa y topología híbrida.

- Topologías extranet, las cuales incluyen Extranet *any-to-any* y Extranet Servicios Centrales.
- Topologías de Propósito Especial, tales como el *Backbone* VPDN y topología de Red Administrada

IV.3.3.1 Topologías para VPNs Intranet

Topología *Hub-and-Spoke*

La topología más usada comúnmente es la topología *Hub-and-Spoke*, donde cierto número oficinas remotas (*spokes*) son conectadas a un sitio central (*hub*), similar a la figura siguiente:

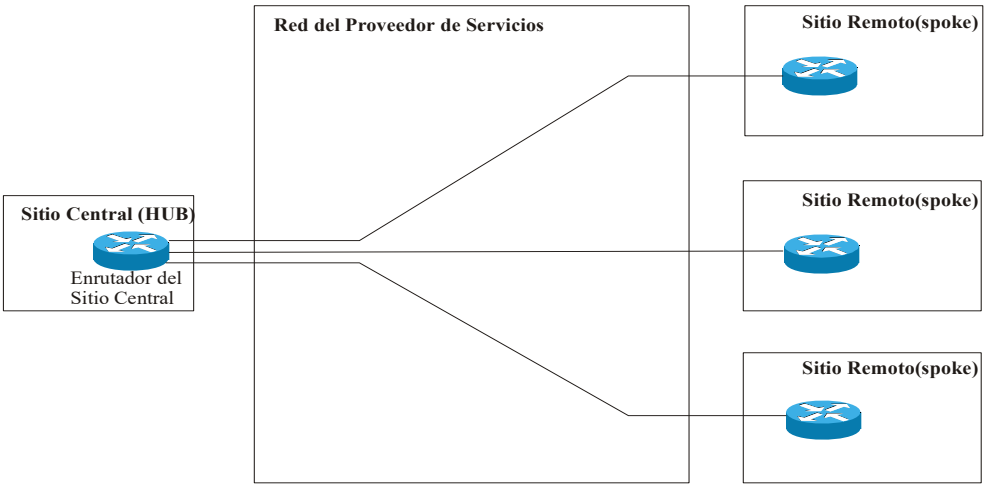


Figura IV.3.3.1.1

Las oficinas remotas usualmente pueden intercambiar datos. No hay restricciones específicas de seguridad en tráfico entre oficinas y la cantidad de datos intercambiados entre ellas es insignificante. Esta topología es usada típicamente en organizaciones con estructuras jerárquicamente estrictas, por ejemplo, bancos, oficinas de gobierno, almacenes, organizaciones internacionales con pequeñas oficinas en cada país, y más.

Al emplear VPNs basadas en tecnologías de capa 2, tales como Frame Relay o ATM, la topología *Hub-and-Spoke* es más común de lo que se podría esperar. Esta topología está basada puramente en necesidades

debidas a costos muy altos o a un incremento en la complejidad en el enrutamiento asociadas a otras topologías que usan otros tipos de tecnologías. En otras palabras, existen muchos ejemplos donde el cliente puede beneficiarse con otra topología diferente pero no tiene más elección que la *Hub-and-Spoken* por razones de costo o complejidad.

Con requerimientos de redundancia, la topología sencilla *Hub-and-Spoke* de la figura anterior frecuentemente es mejorada con un enrutador adicional en el sitio central, tal como se muestra en la figura IV.3.3.3.1.2:

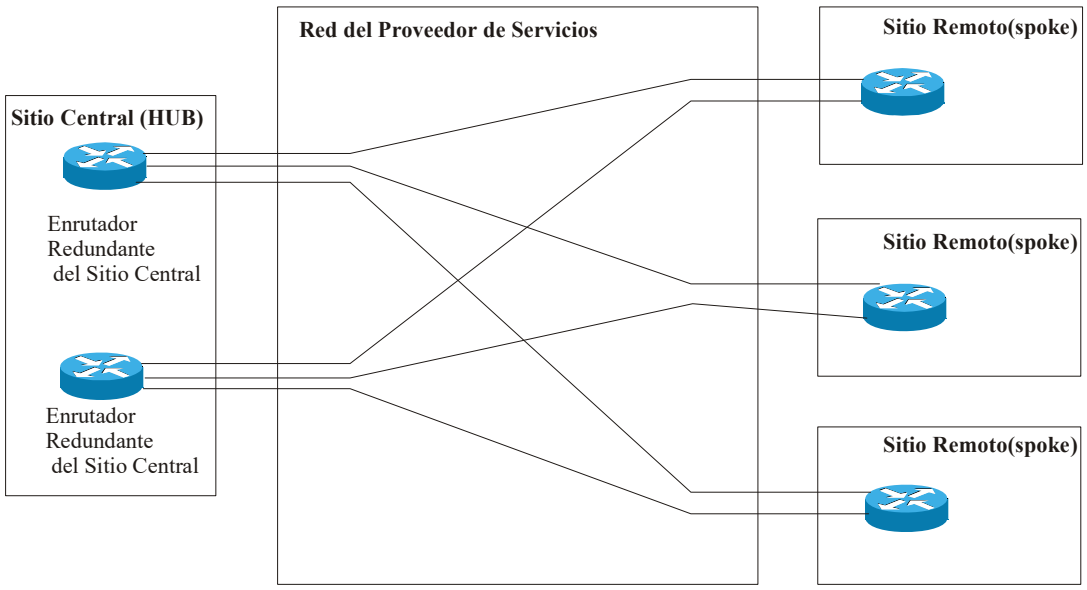


Figura IV.3.3.1.2

Otra forma es agregando un sitio central de respaldo, el cual es conectado con el sitio central primario a través de una conexión con velocidad más alta:

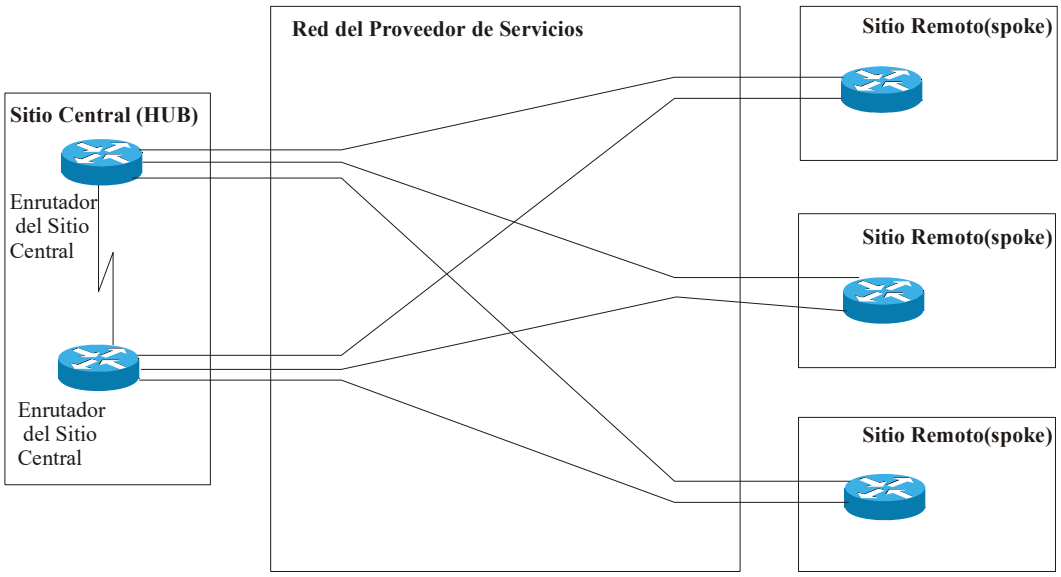


Figura IV.3.3.1.3

El implementar una topología *Hub-and-Spoke* redundante con un modelo VPN overlay basado en VC (*Virtual Circuits*) siempre presenta ciertos retos. Cada *hub* requiere al menos un VC a dos enrutadores centrales. Dichos VCs pueden ser provisionales en la configuración *backbone* primaria o en una configuración de balanceo de carga con un número de desventajas con una u otra solución:

- En la configuración *backbone* primaria, el circuito virtual (VC) de respaldo no es usado mientras el circuito virtual primario está activo, lo que resulta en gastos innecesarios adquiridos por el cliente.
- En la configuración de carga compartida, los sitios *"spokes*" o secundarios encuentran salidas reducidas si uno de los circuitos virtuales (o uno de los enrutadores centrales) fallan. Esta configuración no es apropiada para las topologías con un sitio central de respaldo similar al de la figura anterior.

Los proveedores de servicios de la más alta calidad tratan de cumplir con los requerimientos de redundancia de los clientes ofreciendo un servicio mejorado llamado *Shadow PVC*. Con un *Shadow PVC,* el cliente obtiene dos circuitos virtuales por el precio de uno con la condición de que sólo pueden usar uno a la vez para tráfico de datos (aunque una pequeña cantidad de datos es permitida en el segundo PVC para habilitar los intercambios del protocolo de enrutamiento).

Los requerimientos de redundancia pueden complicar aún más la topología *Hub-and-Spoke* con la introducción de características *dial-backup* (o respaldo de marcación). La solución *dial-backup* implementada en la red del Proveedor de Servicios (por ejemplo, una conexión ISDN respaldando una línea dedicada Frame Relay, como muestra en la figura IV.3.3.1.4)es transparente para el cliente, pero esto no ofrece una verdadera redundancia punta punta, ya que no puede detectar las fallas potenciales (por ejemplo, las fallas en el protocolo de enrutamiento). Una verdadera redundancia punta apunta sobre un modelo de VPN overlay puede ser archivado solo por los dispositivos CE estableciendo una conexión conmutada afuera de la VPN.

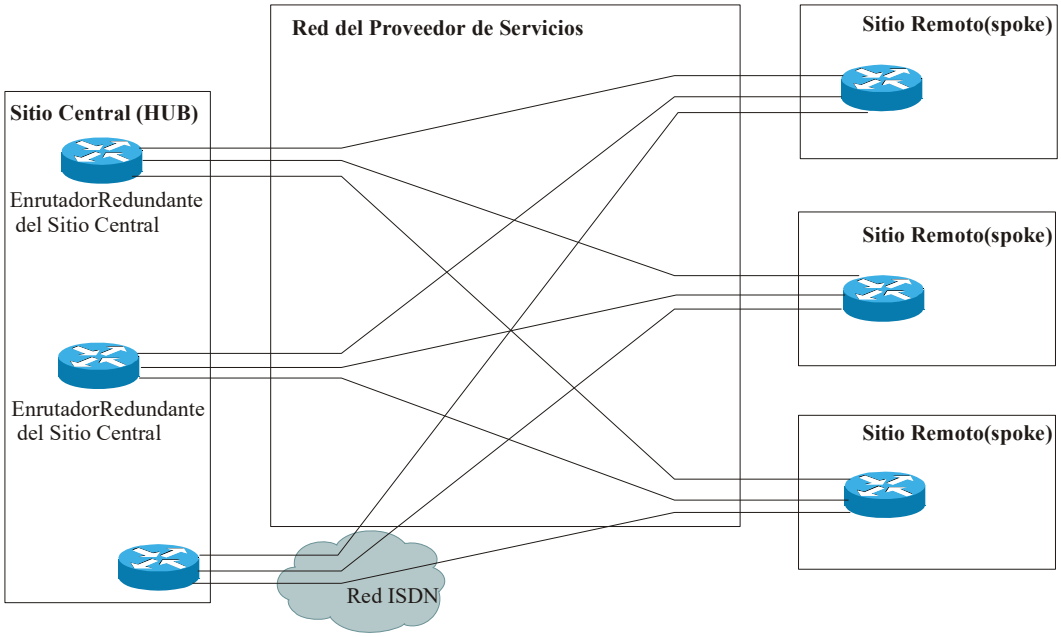


Figura IV.3.3.1.4

Usualmente, una topología simple *hub-and-spoke* se transforma a una topología multinivel conforme va creciendo la red. La topología multinivel puede ser una topología *hub-and-spoke* recursiva, similar a la mostrada en la figura IV.3.3.1.4, o puede ser también una topología híbrida. La reestructuración puede ser integrada por escalabilidad de restricciones de los protocolos de enrutamiento IP o por la escalabilidad de niveles de aplicación (por ejemplo, la introducción de una implementación de tres hileras cliente-servidor).

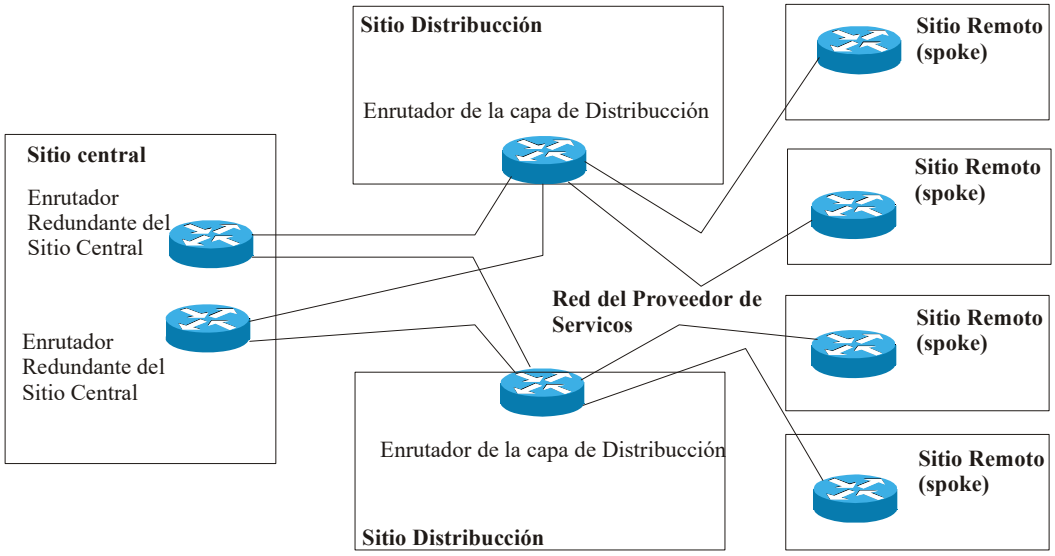


Figura IV.3.3.1.5

La implementación de la topología *hub-and-spoke* junto con un modelo VPN overlay es muy conveniente en entornos donde las oficinas remotas casi siempre intercambian datos con los puntos centrales y no con cualquier otro. Por ejemplo, el intercambio de datos entre las oficinas centrales siempre son transportadas por medio de sitios centrales si la cantidad de los datos intercambiados entre las oficinas remotas representa una proporción significativa del tráfico de la red extendida. Sin embargo, una topología de malla parcial o completa (*partial-mesh* y *full-mesh*) podría ser más conveniente.

Topología de malla parcial o completa (*partial-mesh* y *full-mesh*)

No todos los clientes pueden implementar sus redes con topologías *hub-and-spoke*:

- La organización puede ser menos jerárquica en estructura, requiriendo intercambio de datos entre varios puntos de la organización.
- Las aplicaciones usadas en la organización necesitan comunicación *peer-to-peer*, como los sistemas de mensajería o colaboración.
- Para algunas corporaciones multinacionales, el costo de la topología *hub-and-spoke* podría ser excesiva, así como el costo elevado de los enlaces internacionales.

En estos casos, el modelo VPN overlay más apropiado para las necesidades de una organización podría ser un modelo de malla parcial, donde los sitios dentro de la VPN son conectados por contenedores virtuales (VC) dictaminados por requerimientos de tráfico (donde eventualmente son dictados por la necesidades de negocios). Si no todos los sitios tienen una conectividad directa a todos los sitios (como por ejemplo, en la figura IV.3.3.1.6) la topología es llamada de malla parcial. Si todos los sitios tienen conectividad con todos los demás, es de malla completa.

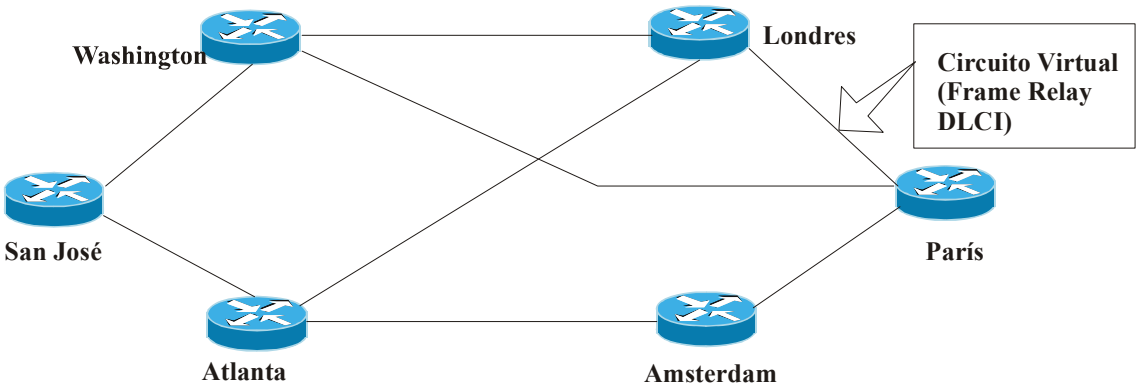


Figura IV.3.3.1.6

No se implementan muchas redes de malla completa debido al muy alto costo de esa implementación y la complejidad introducida por el gran número de VCs. Con ese tipo de topología, el número de VCs es: $VC = \left[\frac{(n-1)n}{2} \right]$ donde n es el número de dispositivos adjuntos. La mayoría de los clientes tienen que instalar una topología de malla parcial, la cual usualmente es afectada por compromisos o por parámetros externos, como la disponibilidad de enlaces y los costos de los VCs.

Implementar una topología de malla completa es bastante simple, sólo se necesita una matriz de tráfico indicando el ancho de banda requerido entre un par de sitios dentro de la VPN, pudiendo a empezar a ordenar los VCs al Proveedor del Servicio. Para el caso de una malla parcial, la implementación no es tan sencilla, ya que se debe de hacer lo siguiente:

1. Resolver la matriz de tráfico.

2. Proponer una topología de malla parcial basada en esa matriz de tráfico (por ejemplo, instalando un VC sólo entre sitios con altos requerimientos de tráfico) y en los requerimientos de redundancia.
3. Determinar exactamente sobre qué VC va a fluir el tráfico entre dos sitios. Este paso también puede involucrar una afinación del protocolo de enrutamiento para hacer más fluido el tráfico sobre las propias VCs.
4. Hacer el tamaño de VCs acorde a la matriz de tráfico y a la agregación del tráfico alcanzado sobre el VC.

Las características del protocolo de enrutamiento en una (usualmente multinacional) malla parcial larga puede crecer a la proporción donde es extremadamente difícil predecir el flujo de tráfico sin usar avanzadas herramientas de simulación. Esto obliga a los clientes a migrar a BGP sólo para manejar los problemas de ingeniería de tráfico en sus topologías de mallas parciales.

Topología híbrida

Las redes VPN grandes construidas con un modelo de VPN overlay tienden a combinar topología *hub-and-spoke* con topología de malla parcial. Por ejemplo, una gran organización multinacional puede tener acceso a las redes de cada país implementando una topología *hub-and-spoke* donde el núcleo de la red internacional puede ser implementada con una topología de malla parcial. La siguiente figura muestra este ejemplo.

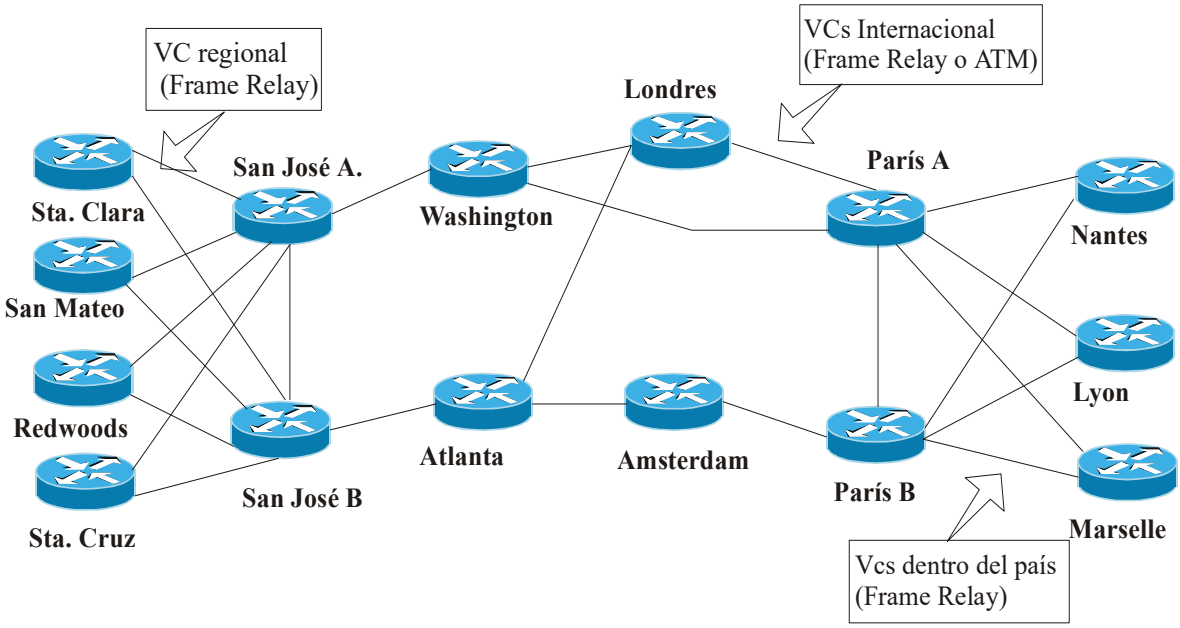


Figura IV.3.3.1.7

La mejor aproximación de un diseño de topología híbrida es siguiendo un diseño de red modular:

- Dividir la red extendida en núcleo, distribución y acceso
- Diseñar el núcleo y los accesos de cada red individualmente (por ejemplo, una topología *hub-and-spoke* doble con un respaldo conmutado en la red de acceso y una malla parcial en el núcleo de la red)
- Conectar la red núcleo y las redes de acceso a través de una capa de distribución tratando de aislarlas lo mejor posible. Por ejemplo, una falla dentro del *loop* local en algún lugar de la oficina remota no debe ser propagada dentro de la red núcleo. Los enrutadores de las oficinas remotas no deben ver las fallas de los enlaces internacionales.

IV.3.3.2. Topologías para VPNs Extranets

Topología simple extranet

Las topologías de intranet conciernen a la topología física y lógica de la red VPN, dictadas por la tecnología VC por la que el modelo VPN overlay es implementado. Las topologías extranet están más enfocadas a los

requerimientos de seguridad de la red VPN, la cual puede ser implementado con diferentes topologías, ya sea con el modelo overlay o *peer-to-peer*.

La topología tradicional extranet puede ser una extranet cualquiera permitiendo a cierto número de compañías intercambiar datos cualquiera-a-cualquiera (*any-to-any*). Los ejemplos pueden incluir comunidades de interés (compañías manufactureras de aeroplanos) o una cadena de proveedores (por ejemplo, una manufacturera de automóviles y sus proveedores).

Los datos en la extranet pueden ser intercambiados entre cualquier número de sitios. La extranet por sí misma no impone restricciones en el intercambio de datos. Usualmente, cada sitio es responsable de su propia seguridad, filtrado de tráfico y *firewalling*. La única razón para usar una extranet en lugar de una internet pública son las garantías de la Calidad de Servicio de sensibilidad de los datos intercambiados a través de la VPN, la cual sigue siendo más flexible a los ataques de la captura de datos que el internet genérico.

Si la extranet es implementada por el modelo VPN *peer-to-peer* (como el ejemplo de la extranet de la figura IV.3.3.2.1), cada organización sólo especifica cuánto tráfico va a ser recibido y enviado por cada uno de los sitios, es decir, la proporción de recursos en el lado del cliente y del Proveedor de Servicios es muy sencilla y efectiva.

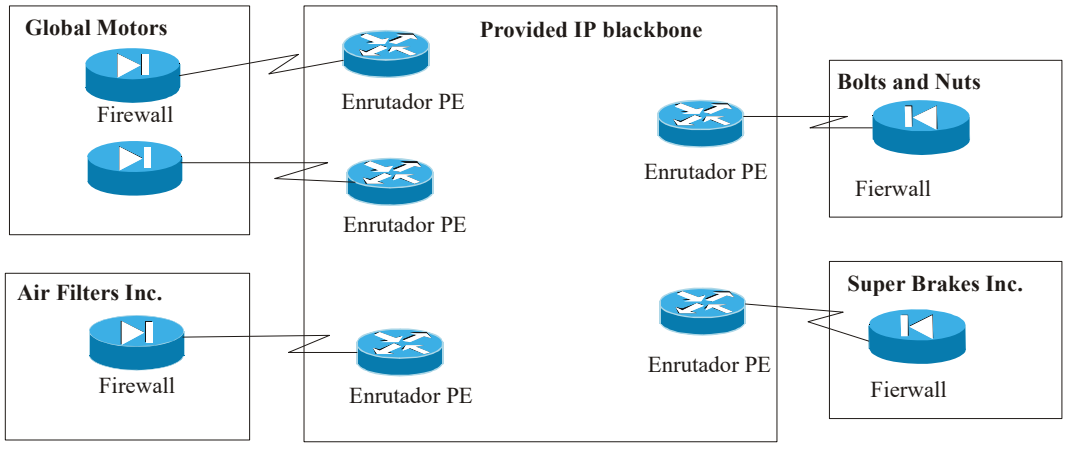


Figura IV.3.3.2.1

En el modelo VPN overlay, el tráfico entre sitios es intercambiado a través de VCs punto a punto (*point-to-point*) como lo muestra la siguiente figura:

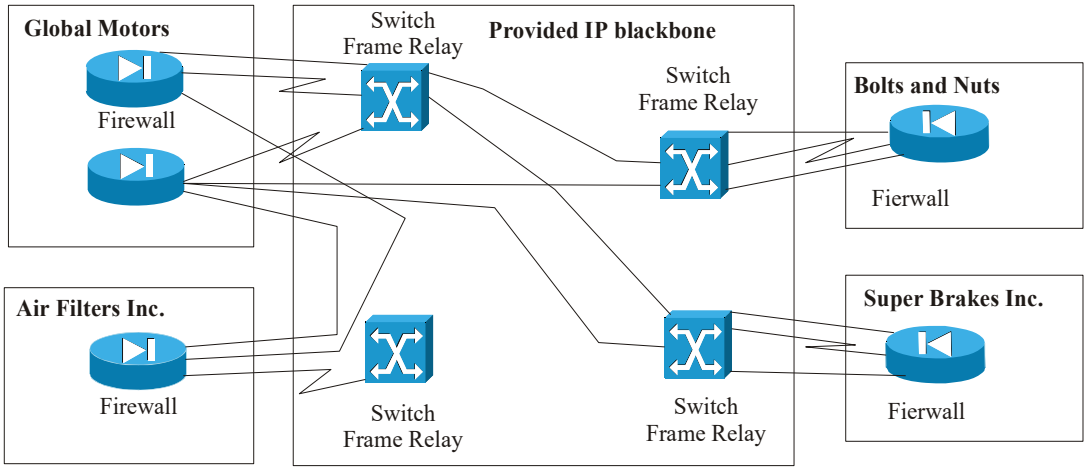


Figura IV.3.3.2.2

En la topología extranet similar a la de la figura anterior, cada organización participante paga por las VCs que usa. Obviamente, sólo se instalan los VCs más necesarios para minimizar costos. Además, los participantes en las VPNs podrían tratar de prevenir el tráfico de tránsito entre otros participantes siguiendo, a través de las VCs, aquellas que pagaron, usualmente resultando en una conectividad parcial entre los sitios de la extranet y, a veces, resultando en complejos problemas de enrutamiento. Por ello, el modelo de VPN *peer-to-peer* es el camino preferido para la implementación de una extranet *any-to-any*.

servicios centrales de extranet

Las extranets que enlazan organizaciones pertenecientes a la misma comunidad de interés, comúnmente son muy abiertas, permitiendo la conectividad *any-to-any* entre las organizaciones. Las extranets de propósito dedicado (por ejemplo, una red de administración de una cadena de proveedores enlazando a una gran organización con todos sus proveedores) tiende a ser más centralizada y permite la comunicación sólo entre

la organización, patrocinando la extranet y todos los demás participantes, similar al ejemplo de la figura siguiente:

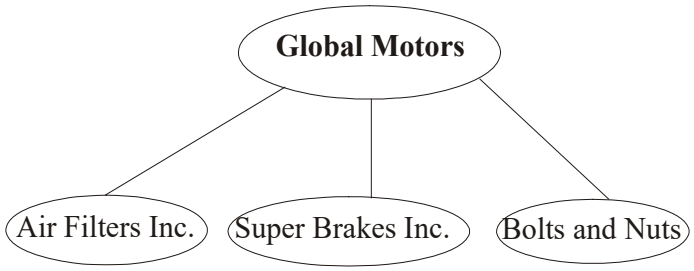


Figura IV.3.3.2.3

La seguridad en los Servicios Centrales de la extranet típicamente es proporcionada por la organización central que patrocina la extranet. Otros participantes con redes internas con misiones críticas (por ejemplo, bancos comerciales) también deberían permitir implementar sus propias medidas de seguridad (como por ejemplo, un *firewall* entre sus redes internas y la extranet)

Similar a cualquier otra red VPN, los Servicios Centrales de la extranet pueden ser implementados ya sea con el modelo VPN overlay o con el modelo *peer-to-peer*. Sin embargo, en este caso, el modelo *peer-to-peer* tiene desventajas definitivas debido a que el Proveedor de Servicios debe tener mucho cuidado en que los participantes de la extranet no puedan alcanzar a otro de ellos. Por el contrario, la implementación de los Servicios Centrales de la extranet sobre un modelo VPN overlay es extremadamente directo:

- Los VCs entre todos los sitios participantes y los sitios centrales son mantenidos y administrados. El tamaño de cada VC corresponde a los requerimientos de tráfico entre el sitio participante y el sitio central.
- El sitio central anuncia subredes disponibles únicamente del sitio central hacia sitios participantes.
- Los filtros del tráfico en el sitio central recibidos por otros participantes para hacer seguro un problema de enrutamiento o los ataques útiles de robo de servicio no influyen en la estabilidad de la VPN.

Cuidando estos tres aspectos, la red VPN se la figura anterior (figura IV.3.3.2.3), se transforma en la topología de VCs de la figura que la precede (figura IV.3.3.2.2).

Bajo el modelo extranet *any-to-any*, la red de la figura IV.3.3.2.2 tendría un número limitado de VCs (resultando en una topología redundante *hub-and-spoke*) debido a los inconvenientes del costo. Bajo el modelo de los Servicios Centrales de la extranet, la misma VPN tendría el mismo número de VCs debido a restricciones de seguridad. Así, podemos observar que un número de requerimientos diferentes pueden ser dictados por una misma topología VC.

Una topología extranet un poco más compleja de Servicios Centrales puede contener cierto número de servidores dispersos a través de varios sitios y cierto número de sitios clientes que tienen acceso a esos servidores, similar a la figura IV.3.3.2.4. Los ejemplos típicos que pudieran requerir esta topología son las redes sobre IP, donde un número de usuarios tiene acceso a *gateways* comunes en diferentes ciudades o países pero no está permitido que se vean entre ellos.

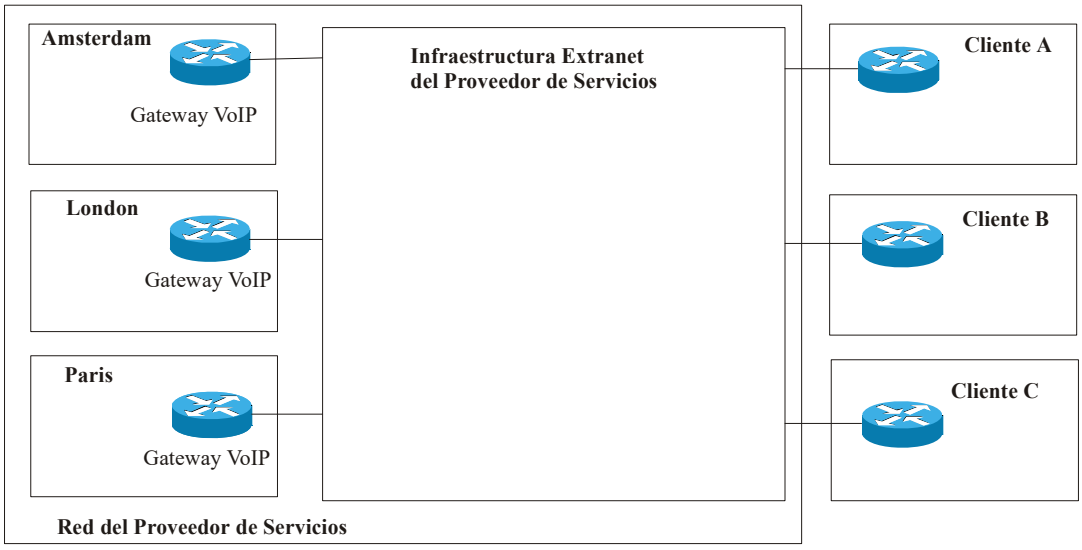


Figura IV.3.3.2.4

Una extranet puede ser implementada, ya sea con un modelo VPN *peer-to-peer* o con un modelo overlay. El número de VCs en el modelo VPN overlay (en el que un VC separado es necesario para cada sitio cliente y

para cada sitio del servidor) y la correspondiente complejidad de repartición de recursos usualmente evita el desarrollo de un modelo VPN overlay en estos escenarios. Una instalación más manipulable podría usar un modelo *peer-to-peer* o una combinación de ambos modelos, como se ilustra en la siguiente figura:

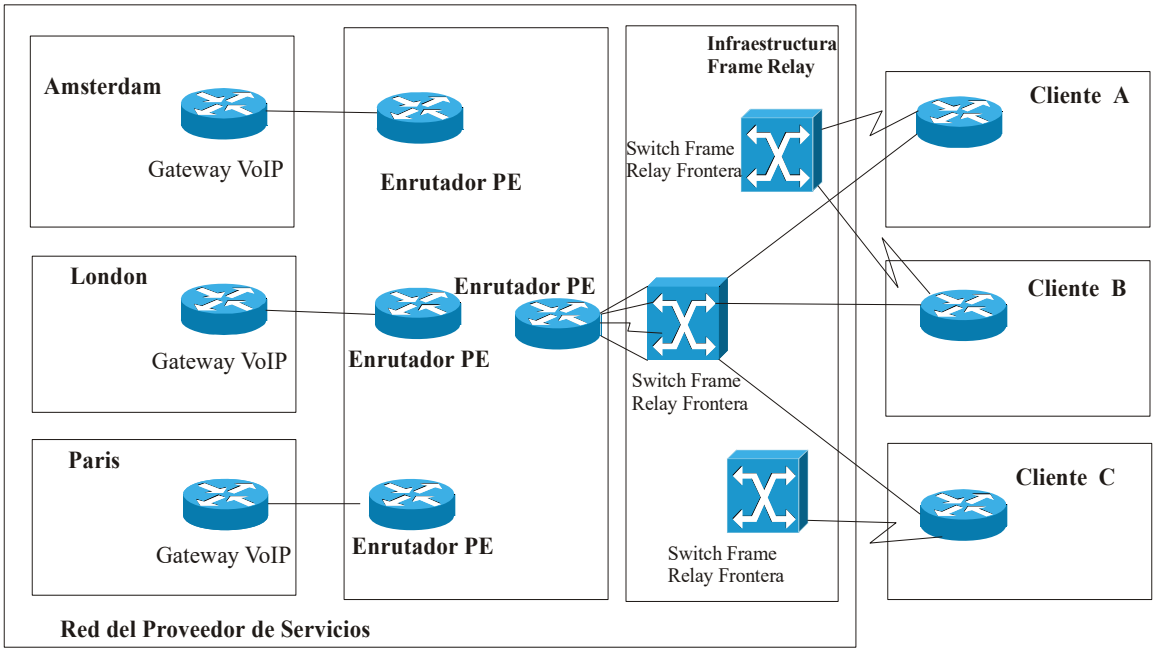


Figura IV.3.3.2.5

Lógicamente, la red en la figura anterior usa un modelo VPN *peer-to-peer* con enrutadores de distribución en el papel de los enrutadores PE (*Provider Edge*). La actual topología física difiere del punto de vista de la topología lógica: los enrutadores de distribución son enlazados con los sitios clientes (los enrutadors CE o *Customer Edge*) a través del modelo VPN overlay. Un ejemplo de esto, es la red Frame Relay.

IV.3.3.3 Topologías para VPDNs

Topología VPDN

Una VPDN (VPDN, *Virtual Private Dial-up Network*) es una VPN que usa como medio de conmutación una red de líneas telefónicas de par de cobre (PSTN) para establecer un enlace a la red corporativa que desea. El servicio de la Red Privada Virtual Conmutada usualmente se implementa usando túneles PPP para el

intercambio de *frames* PPP entre usuarios conmutados y sus *gateways* locales en paquetes IP intercambiados con el servidor de la red de acceso, como se muestra en la figura IV.3.3.3.1:

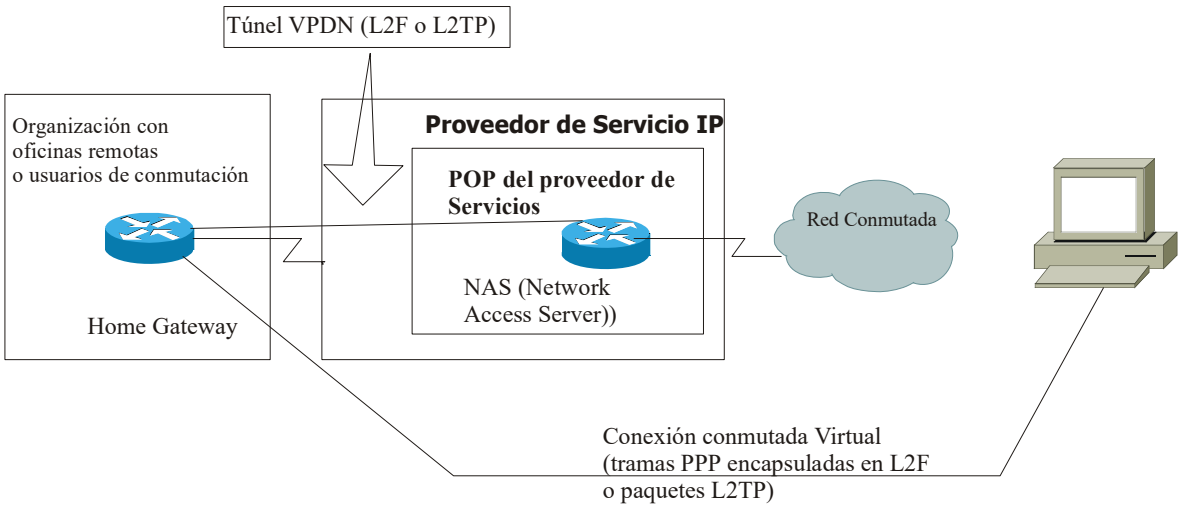


Figura IV.3.3.3.1

El usuario conmutado y el *gateway* local establece una conectividad IP (o IPx, AppleTalk, etc.) a través de enlaces por túneles PPP e intercambiando paquetes de datos a través de éstas. La figura que se muestra a continuación (IV.3.3.3.2) detalla la pila *o stack* del protocolo usado entre varias partes de la solución VPDN.

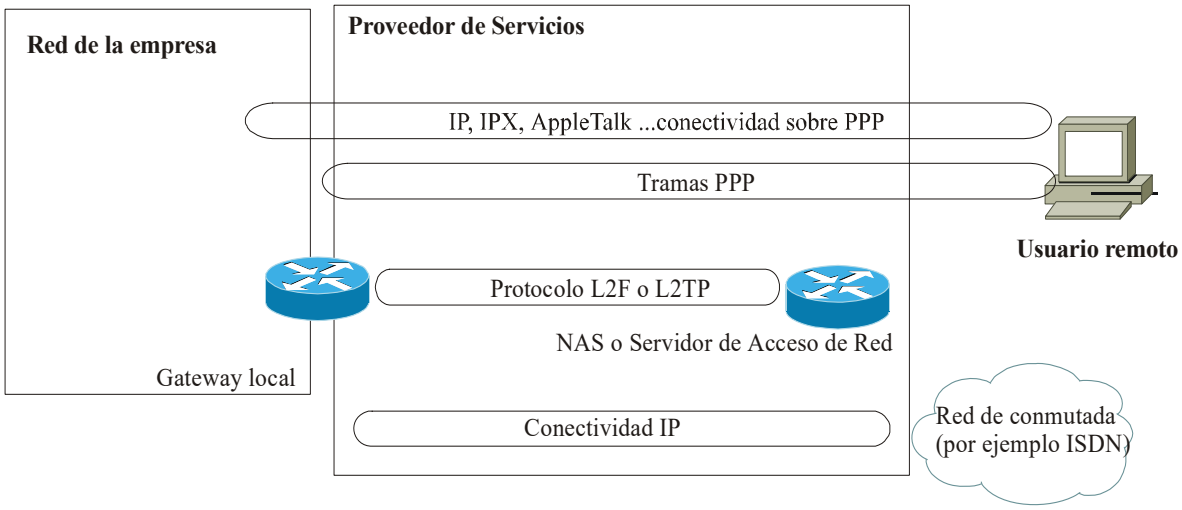


Figura IV.3.3.3.2

Cada solución VPDN requiere una infraestructura basada en IP para intercambiar datos por medio de túneles PPP entre el NAS (*Network Access Server*) y el *gateway* local. En el escenario más sencillo posible, el internet público puede ser usado como la infraestructura necesaria. Cuando los requerimientos de seguridad son más estrictos, una Red Privada Virtual puede ser construida para intercambiar *frames* PPP encapsulados. La estructura resultante para algunos diseñadores de redes resulta muy compleja, porque tratan de entender todo el escenario y todos los detalles al mismo tiempo, pero la complejidad puede ser reducida grandemente desarticulando las partes:

- El NAS y el *gateway* local utilizan sin diferencia la infraestructura IP que está disponible para intercambiar datos VPDN, la cual puede ser pensada como una aplicación ubicada en la parte más alta de la pila o *stack* IP. Consecuentemente, la estructura interna de la red IP no afecta el intercambio de los datos de aplicación y los contenidos de los datos de aplicación (paquetes IP en *frames* PPP encapsulados en una envolvente VPDN) no interactúan con los enrutadores que proveen los servicios IP
- La red basada en IP es efectivamente una extranet de Servicios Centrales con muchos sitios servidores (NAS, *Network Access Server*) y *gateways* locales actuando como sitios cliente. Esta infraestructura puede ser implementada en varias de maneras, desde modelo VPN overlay o con el modelo *peer-to-peer*.

Topología VPN de redes administradas

Este tipo de topología VPN es usada por los Proveedores de Servicios para administrar los enrutadores *customer-premises* (cliente-permisos) en un servicio de red administrada. Típicamente, como se muestra en la figura IV.3.3.3.3, el Proveedor de Servicios proporciona un número de enrutadores al sitio cliente, conectándolos a través de VCs implementados con Frame Relay o ATM y construidos con topologías *hub-and-spoke* separadas, conectando cada enrutador cliente con el Centro de Administración de la Red (NMC, *Network Management Center*).

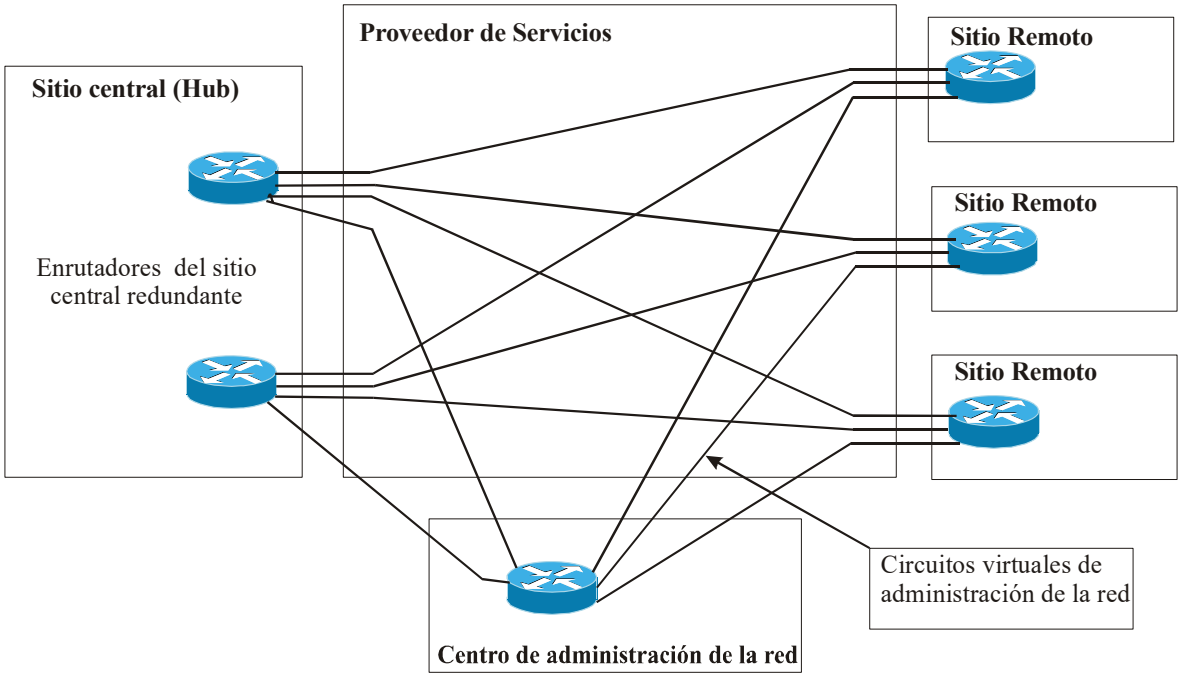


Figura IV.3.3.3.3

La topología VPN usada en la parte del cliente de la red puede ser cualquier topología que soporte el modelo VPN, yendo desde la topología *hub-and-spoke* hasta la topología de malla completa o *full mesh*. La topología usada en la parte de la red que administra la CPE efectivamente puede ser una topología extranet de Servicios Centrales con enrutadors actuando como clientes y NMCs (*Network Management Center*) siendo el punto central de la administración de la extranet.

Como ya se explicó, la topología de Servicios Centrales de Extranet es más sencilla de implementar con una topología *hub-and-poke* del modelo VPN overlay, lo cual también explica porqué la mayoría de los proveedores de servicios de redes administradas usan el arreglo de la figura IV.3.3.3.3.

La topología de Red administrada también puede ser implementada con varias tecnlogías VPN *peer-to-peer*, a pesar de que no es tan sencillo como con el modelo VPN overlay.

IV.4. Operación de las VPNs

Los factores principales que componen a una VPN son *el tuneleo o tunnelling* y los servicios de seguridad. En esta sección se describirá el aspecto del tuneleo y en la siguiente, se explicará la seguridad en una VPN.

[2]Existen dos tipos comunes de uso de VPNs: las VPDNs y las VPNs que se enlazan sitio a sitio. Una VPDN es una conexión de acceso remoto usuario-a-LAN usada por una compañía que tiene empleados que necesitan conectarse a su red privada desde lugares remotos. Típicamente, una corporación que desea establecer varios accesos remotos a su VPN proporciona a sus usuarios algún número para entrar a Internet por medio de un ISP. Entonces, los usuarios remotos marcan tal número para ingresar a Internet y usan el *software* de los clientes de la VPN para tener acceso a la red corporativa. Un ejemplo de alguna compañía que necesita varios accesos remotos a una VPN podría ser una firma con cientos de vendedores en toda una ciudad. Los accesos remotos a la VPN permiten conexiones seguras y cifradas entre la red corporativa y sus usuarios remotos a través de un proveedor de servicios.

Con una VPN sitio-a-sitio, una compañía puede conectarse a varios sitios fijos a través de una red pública (como Internet) con el uso de equipo dedicado y un cifrado a gran escala de la información. Cada sitio necesita solamente de una conexión local a la misma red pública, con lo que se logra un ahorro considerable de líneas dedicadas. Las VPNs sitio-a-sitio pueden ser construidas entre oficinas de la misma compañía, o por ejemplo, de una oficina de la compañía hacia un tercer sitio para compartir una base de datos.

El tuneleo o *tunnellig* es el proceso de encapsular un paquete entero dentro de otro paquete y enviarlo sobre una red. El tuneleo por sí sólo no proporciona seguridad a la información, pues aún después de que un paquete haya sido encapsulado en otro, sigue siendo visible para el dispositivo que recibe la encapsulación. Los protocolos de cifrado usan el tuneleo como medio para transferir los datos cifrados a través de una red pública, por lo que son una parte esencial en una VPN[2].

El tuneleo es la característica que hace posible la construcción de una VPN. Este proceso oculta la arquitectura y la operación de las redes inmediatas (como el Internet) de los dispositivos o redes conectadas a una VPN, lo cual logra que la implementación de la VPN sea sencilla debido a que no es necesario conocer los detalles de cómo se interconectan las redes, y tampoco necesitan conocer la existencia de la VPN los dispositivos conectados a ella, exceptuando al *gateway* de seguridad. Esto significa que la VPN puede ser conectada, desconectada, modificada o reemplazada sin alteraciones a la LAN que esté directamente conectada a ella[5].

Existen dos tipos de túneles: los permanentes (o estáticos) y los temporales (o dinámicos). Los túneles estáticos limitan la aplicación de las VPNs y por ello no son usados en una configuración VPN. En cambio, los túneles dinámicos reducen la utilización del ancho de banda y el costo, pues sólo se emplean cuando se requiere.

El corazón del tuneleo es la encapsulación de paquetes LAN en otro paquete, lo que implica que los detalles de la VPN no son importantes para la LAN y viceversa. La información que se encapsule depende de la capa en la cual opere la VPN. La encapsulación en capa 2 ó capa 3 es común en los protocolos VPN. Paja ilustrar la encapsulación tenemos:

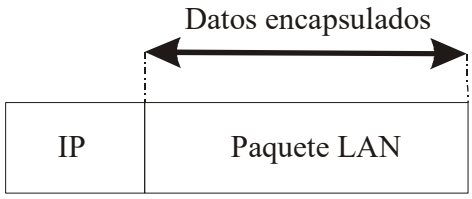


Figura IV.4.1

La capa en la cual tome lugar la encapsulación depende de la configuración de las redes directamente conectadas y de los dispositivos que situados en los extremos del túnel. Por ejemplo, si dos redes LAN que se conectan a través de una VPN tienen diferentes protocolos de capa 2, pero su protocolo de capa 3 es el mismo, se utiliza el protocolo de capa 3 para la encapsulación. Alternativamente se podría usar el protocolo de capa 2 para encapsular, pero sería necesario emplear un dispositivo de interconexión (tal como un

enrutador) que tradujera la información entre los dos protocolos de la capa de enlace de datos. Sin embargo, es importante considerar el flujo extra en un análisis para determinar cuál protocolo sería empleado para la encapsulación, pues mientras más baja sea la capa del protocolo, más grande es el encabezado y el *trailer* del paquete encapsulado, resultando en más flujo extra[5].

Con lo anterior, se puede destacar que una de las características más importantes de la arquitectura de una VPN es la capacidad de comunicar redes diferentes que usen distintos protocolos de capa de enlace de datos, pues la VPN se comporta similar a un enrutador en ese sentido, traduciendo ambos protocolos para la comunicación. Un ejemplo sería una VPN que conectara a una red LAN Ethernet y a una red LAN FDDI, ambas con el mismo protocolo de la capa de red. Cuando los datos se transmiten sobre la VPN, sólo la información de la capa 3 (como IP o IPX) puede ser intercambiada y cuando dicha información alcance su destino, es enviada para ser procesada en la capa 2. La encapsulación se realiza en el paquete entero, incluyendo los encabezados y las colas de las redes interconectadas[5].

Para tunelear un paquete, se necesitan tres diferentes protocolos:

- **Protocolo pasajero**: en el cual los datos originales son transportados (como por ejemplo, IP, IPX o NetBEUI)
- **Protocolo de encapsulación:** con el cual se "envuelve" al paquete original (como por ejemplo, GRE, IPsec, L2F, PPTP, L2TP)
- **Protocolo portador:** que es usado por la red sobre la cual se transporta la información

El paquete original (en el protocolo pasajero) es colocado dentro del protocolo de encapsulación, el cual a su vez es puesto dentro del encabezado del protocolo portador (como IP) para transmitirlo sobre una red pública. Algunos protocolos de encapsulación llevan a cabo el cifrado de los datos (como L2TP, L2F, PPTP, IPsec, etc.), por lo que se le agrega seguridad a la transmisión. Para VPNs sitio-a-sitio, generalmente el protocolo de encapsulación es GRE o IPsec. En VPDNs , el tuneleo usualmente se lleva a cabo con PPP. Como parte del

stack de TCP/IP, PPP es el portador para otros protocolos IP en la comunicación entre el *host* y el sistema remoto. Además, PPP usa L2TP, L2F o PPTP para tunelear el paquete.

IV.5.Seguridad en las VPNs

[6]Hace cinco años, las redes estaban "cerradas" al tráfico externos, es decir, tenían un comportamiento parecido al que se muestra en la figura:

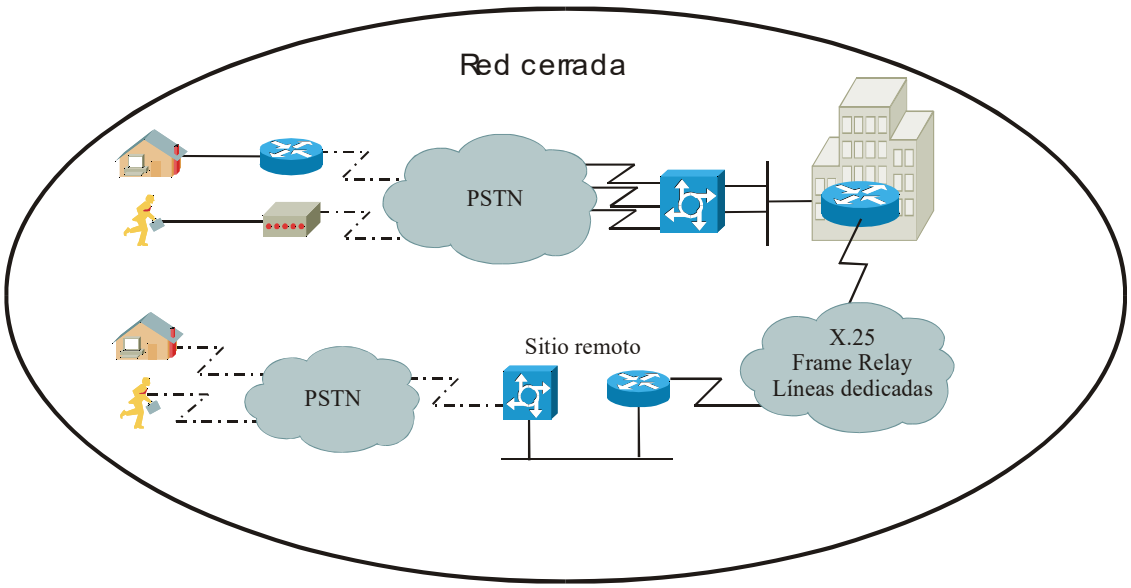


Figura IV.5.1

Actualmente, las redes se han "abierto", permitiendo el libre tránsito de paquetes a través de redes compartidas, teniendo un escenario similar al que a continuación se ilustra:

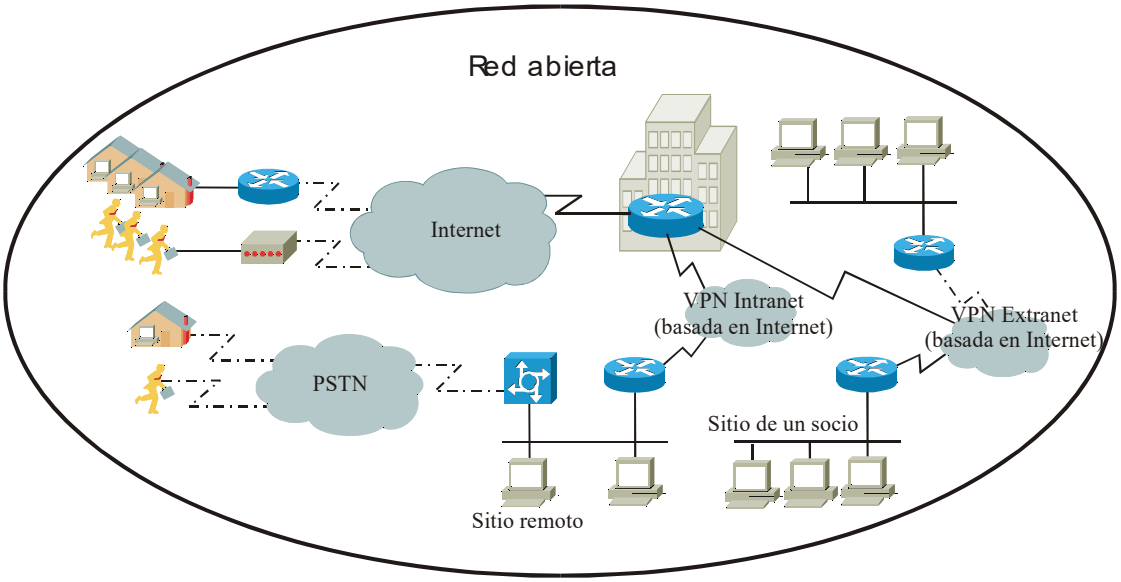


Figura IV.5.2

El papel de las redes ha cambiado en años recientes tomando cada vez mayor importancia, pues se han extendido a las áreas comerciales y financieras (entre otras), dando lugar a los *e-bussiness*. Debido a que son cada vez más las empresas que utilizan las redes de datos para concretar operaciones, ya sean muy sencillas (como la confirmación de una cita, por ejemplo) o de vital importancia (como muestra, una fuerte transacción), la seguridad de los datos es un tema primordial para cualquier administrador de una red. Se busca principalmente que la comunicación en una red sea segura aún en ambientes abiertos y un proceso continuo que se establezca una fuerte política de seguridad y una administración centralizada. Como ya se explicado, las VPNs pueden ser una buena opción en el transporte de datos confidenciales.

[4]Las empresas que tengan VPNs deben asegurarse de que éstas están a salvo de observadores prohibidos que perpetren los datos confidenciales que se transportan sobre la VPN y debe protegerse de los usuarios no autorizados que ingresen a la red y tengan acceso a sus recursos. Para garantizar la seguridad en una VPN, se deben cuidar cuatro aspectos importantes:

- Los túneles y el cifrado de datos
- Verificación de los paquetes

- *Firewalls* y detección de invasiones a la red
- Verificación del usuario

Estos mecanismos se complementan unos con otros, proporcionando seguridad en diferentes puntos de la red. [5]Una solución VPN debe ofrecer cada una de estas características de seguridad para ser considerada una solución viables para utilizar la infraestructura de una red pública. Un escenario ideal para una VPN, sería como se muestra en la figura IV.5.3:

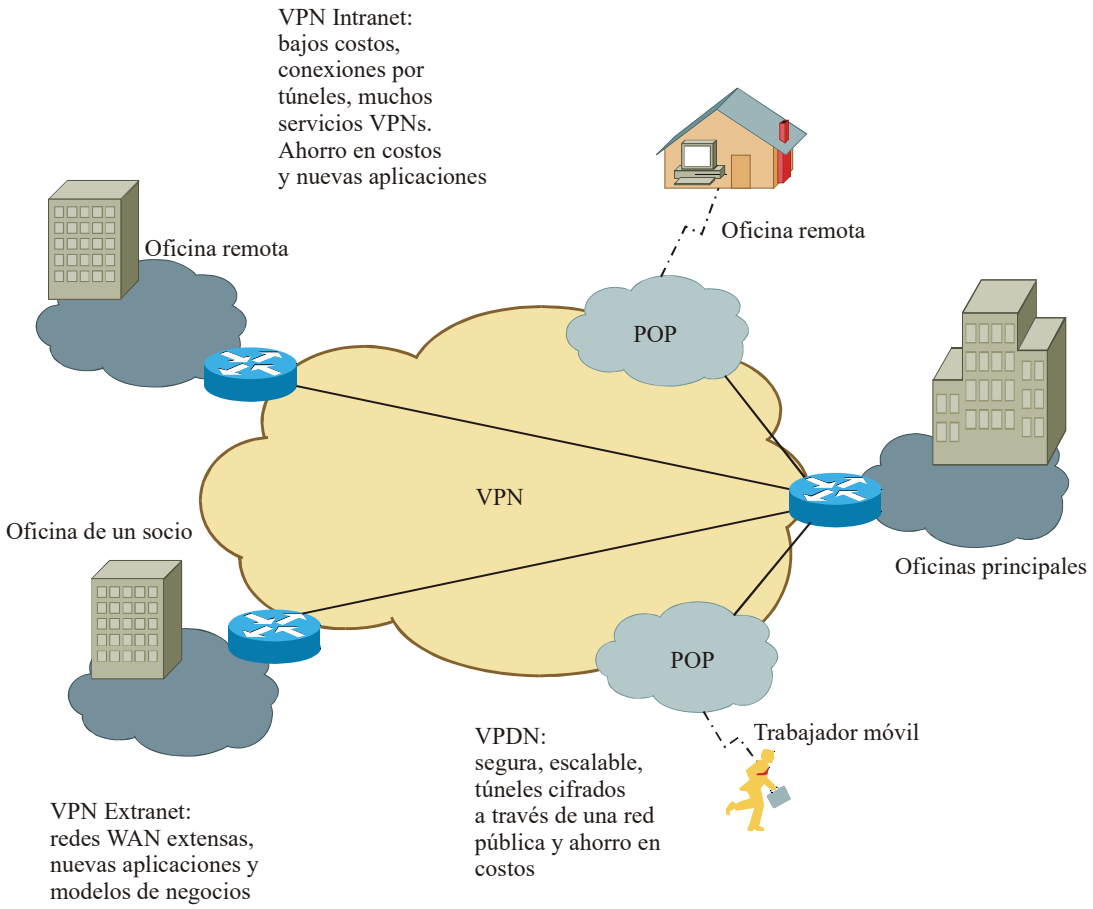


Figura IV.5.3

Túneles y cifrado de datos

Una solución VPN emplea túneles cifrados (con múltiples protocolos de encapsulación si fuera necesario) para proteger a los datos de ser interceptados y vistos por entidades no autorizadas. Los túneles proporcionan conexiones lógicas punta a punta a través de la red IP no orientada a conexión, habilitando la aplicación ed avanzadas características de seguridad. El cifrado se aplica a la conexión tuneleada para desordenar los datos y así hacerlos legibles sólo a los transmisores y receptores autorizados. En las aplicaciones en las que la seguridad no es un asunto preocupante, los túneles se pueden emplear sin cifrar los datos para proporcionar soporte sin seguridad a múltiples protocolos[4].

Verificación de los paquetes

A pesar de que la intervención de los datos en una red compartida es la preocupación principal para las empresas, la integridad de los datos también es importante. En una red insegura, los paquetes pueden ser interceptados y los datos pueden sufrir alteraciones para ser posteriormente enviados a su destino con información errónea. Por ejemplo, una factura ordenada al proveedor de una empresa que sea enviada sobre una red insegura, puede ser modificada cambiando la cantidad de la orden de 1000 a 100.

La verificación de los paquetes los protege contra intromisiones al aplicar encabezados al paquete IP con el fin de asegurar su integridad. Los componentes de IPsec, como *Authentication Header* (AH) y *Encapsulation Security Protocol* (ESP) son usados en conjunto con algoritmos *hash* (o de desorden) estándares, tales como MD5 (*Message Digset* 5), para garantizar la integridad de los paquetes transmitidos sobre un *backbone* IP compartido[4].

Firewalls y detección de invasiones a la red

En una aplicación VPN, los *firewalls* protegen a la empresa de accesos no autorizados y de ataques a su red, simplemente negando el acceso a la VPN. Además, para el tráfico autorizado, los *firewalls* verifican el origen de los datos y privilegia el acceso a los usuarios que son aceptados.

Un elemento de adicional de seguridad es la detección de invasiones. Mientras los *firewalls* permiten o niegan el paso de tráfico según su origen, destino, puerto y otros criterios, no analizan su contenido. Los sistemas de detección de invasión operan conjuntamente con los *firewalls* para extender el perímetro de seguridad al paquete, pues se analizaría el contenido y el contexto de cada uno de ellos en forma individual y así se determinaría si el tráfico es autorizado. Si la cadena de datos en una red experimenta actividad no autorizada el software de detección de invasiones automáticamente aplica políticas de seguridad en tiempo real, como por ejemplo, la desconexión de la sesión, y notifica al administrador de la red del incidente.

El uso de *firewalls* y del *software* de detección de invasiones proporcionan fuertes mecanismo de defensa contra los ataques a la red, pero una seguridad fuerte comienza adentro de la compañía al minimizar desde adentro las vulnerabilidades de la seguridad. Existen sistemas de auditoría de seguridad que revisa toda la red e identifica riesgos potenciales[4].

Verificación del usuario

Un componente clave de la seguridad de las VPNs es la garantía de que sólo los usuarios autorizados tengan acceso a los recursos de la empresa, mientras que los usuarios no autorizados son bloqueados totalmente para entrar a la red. Las soluciones a este problema son la verificación, autorización y contabilización de los accesos, de tal forma que la verificación de los usuarios determina el nivel e acceso y archiva toda la información necesaria de la contabilización de los datos[4]. Estas capacidades se encuentran reunidas en un servidor AAA (*Authentication, Authorization, Accounting*), el cual es usado para incrementar la seguridad en

un ambiente VPDN. Sin la verificación del usuario, cualquier computadora preconfigurada como un cliente de la VPN puede establecer una conexión segura en la red remota. Sin embargo, con la verificación de usuario, su nombre y su contraseña tienen que darse aún antes de que la conexión se complete. Los nombre de usuarios y las contraseñas pueden ser almacenados en un dispositivo terminal de la VPN o en un servidor AAA, que puede proveer verificación a numerosas bases de datos, tales como Windows NT, Novell, etc.

Cuando se solicita el establecimiento de un túnel, el dispositivo VPN pide el nombre de usuario y la contraseña. Esto puede ser verificado localmente o enviado a un servidor externo AAA, en el que se verifica lo siguiente:

- ¿quién eres? (Verificación)
- ¿qué tienes permitido hacer? (Autorización)
- ¿qué Qué estás haciendo? (Contabilización)

La información e contabilización es especialmente útil para propósitos de reportes, facturación o auditoría[2].

Referencias

[1] MPLS Technologies

[2] How Virtual Private Networks Work. http://www.cisco.com/warp/public/471/how_vpn_works.pdf

[3] Understanding VPDN. http://www.cisco.com/warp/public/471/vpdn_20980.html

[4] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/vpn.pdf

[5] http://www.eng.uts.edu.au/~kumbes/ra/vpn/vpn02a.htm#VPN%20History

[6] Seminarios Tecnológicos Cisco Systems. Cisco SAFE: Solución Integral de Seguridad y VPNs

Capítulo V. Operación De VPNs MPLS

V.1. Introducción

Una VPN contiene dispositivos de los clientes conectados a los enrutadores CE. Estos dispositivos usan VPNs para intercambiar información entre dispositivos, pero únicamente los enrutadores PE están "enterados" de las VPNs. En el capítulo III se explicó el funcionamiento de los dispositivos que integran una red MPLS y en el capítulo IV, se describieron los dispositivos que componen una VPN. En una red VPN MPLS, se emplean los términos de enrutadores CE o *Customer Edge* (los cuales pertenecen al cliente y están conectados a la nube MPLS), enrutadores PE o *Provider Edge* (enrutadores *Edge-LSR*, según la terminología MPLS) y enrutadores P o *Provider* (de acuerdo a la terminología MPLS, enrutadores LSR). Un ejemplo de red VPN MPLS se muestra a continuación, donde se observa el *backbone* de enrutadores P, los enrutadores PE y los enrutadores CE:

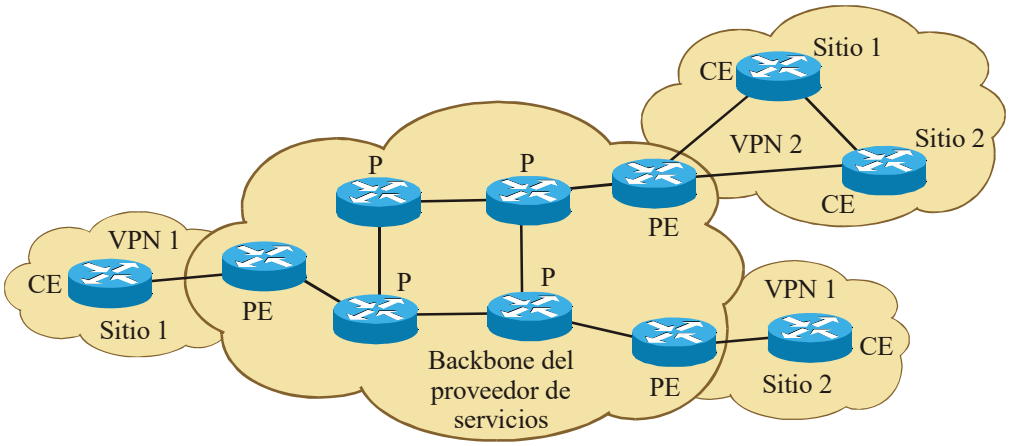


Figura V.1.1. Ejemplo de VPN MPLS

Desde el punto de vista del cliente, en la VPN se observa que sus enrutadores internos se comunican con el enrutador CE de un sitio hacia otro a través de la VPN, administrada por el proveedor de servicios:

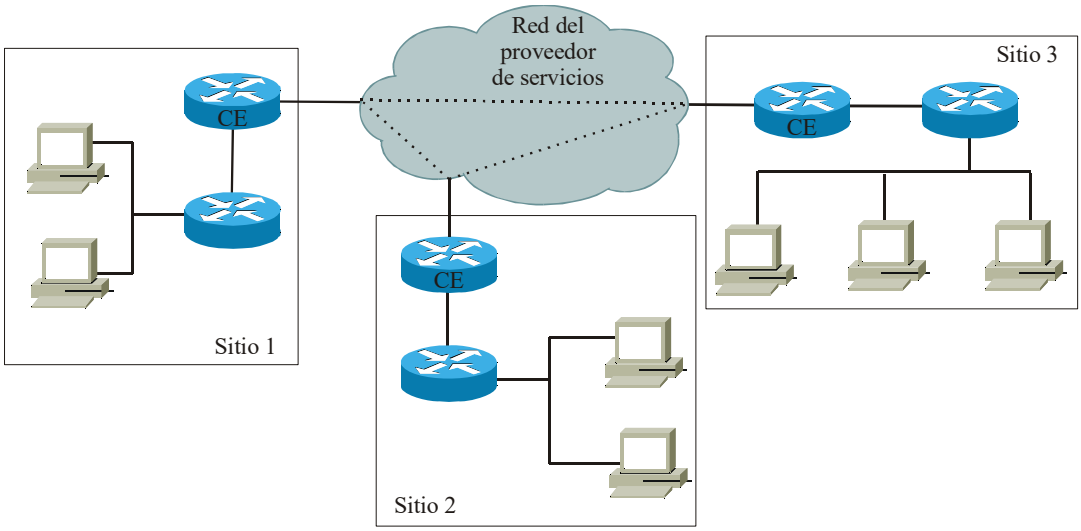


Figura V.1.2. VPN MPLS desde el punto de vista del cliente.

Esta apariencia sencilla de la red que se le da al cliente es la ventaja de emplear VPNs pues el cliente experimenta comunicación directa con sus sitios, tal como si tuviera una red privada a pesar de que su tráfico atraviesa una red pública y de que está compartiendo infraestructura con otros clientes.

El punto de vista del proveedor es naturalmente diferente, como se muestra en la figura V.1.3. la cual ilustra los diferentes clientes y con cada uno tiene una VPN. Sin embargo, un cliente puede tener múltiples VPNs:

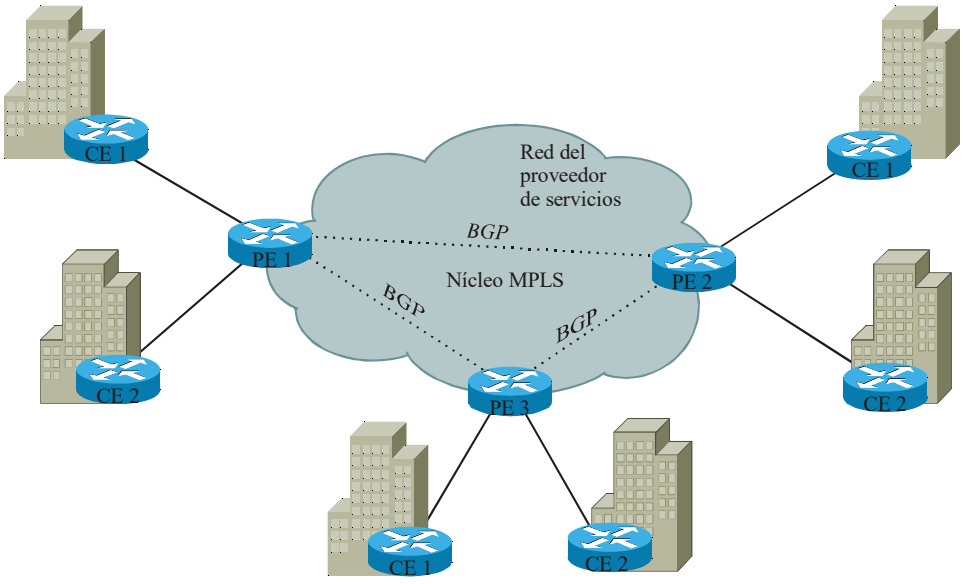


Figura V.1.3. VPN MPLS desde el punto de vista del proveedor de servicios.

Las VPNs MPLS permiten a los proveedores de servicio desarrollar VPNs escalables y entregar servicios de valor agregado, incluyendo:

Servicio no orientado a conexión:

Una ventaja técnica significativa de las VPNs MPLS es que son no orientadas a conexión. Internet debe su éxito a su tecnología básica, TCP/IP, la cual está construida en un paradigma de red no orientado a conexión basado en paquetes. Esto significa que no es necesaria una acción previa para establecer comunicación entre dos *hosts*, facilitando la comunicación para ambas partes.

Para establecer privacidad en un ambiente IP no orientado a conexión, las soluciones actuales de VPNs imponen una red *overlay* o extendida, con enlaces orientados a conexión, punta a punta. Incluso si una VPN se implementara sobre una red no orientada a conexión, ésta no podría tomar ventaja de la facilidad en la conectividad y de los múltiples servicios disponibles en una red no orientada a conexión. Cuando se crea una VPN no orientada a conexión, no son necesarios los túneles o el cifrado de los datos para obtener privacidad en la red, lo cual reduce considerablemente la complejidad.

Servicio centralizado:

La construcción de VPNs en capa 3 permite entregar servicios especiales a un grupo de usuarios representados por una VPN. Una VPN debe dar servicio por parte de los proveedores más allá de los mecanismos que requieran los usuarios conectados privadamente a la red. Además, debe proporcionar una manera flexible de entregar servicios de valor agregado a los clientes. La escalabilidad es crítica ya que los clientes quieren usar servicios privados en sus intranets y en sus extranets. Debido a que una VPN MPLS es vista como una Intranet privada, es posible usar servicios tales como *multicast*, calidad de servicio (QoS), soporte telefónico en una VPN, etc.

Escalabilidad:

Si se crea una VPN orientada a conexión, punto a punto, usando el modelo *overla*y como, por ejemplo, con Frame Relay o ATM, la principal desventaja que se tiene es la escalabilidad. Específicamente, no son óptimas las VPNs orientadas a conexión, aun sin conexiones de malla completa entre los sitios del cliente. Por esta razón, las VPNs MPLS usan el modelo *peer-to-peer* y arquitectura de capa 3 no orientada a conexión para ofrecer una escalabilidad adecuada, ya que el modelo *peer-to-peer* requiere una conexión entre sólo un sitio del cliente y un enrutador PE, a diferencia del modelo *overlay* en el que todos los sitios del cliente se conectan a un PE. Además, la arquitectura no orientada a conexión permite la creación de VPNs en la capa 3, eliminando la necesidad de túneles o circuitos virtuales.

Otras características de escalabilidad de las VPNs MPLS se deben a la partición de rutas de la VPN en los enrutadores PE y, además, la partición de la VPN en rutas IGP entre los enrutadores PE y los enrutadores P, cuidando dos aspectos:

- Los enrutadores PE deben mantener las rutas VPN para aquellas rutas a las cuales las VPN son miembros.
- Los enrutadores P no deben mantener alguna ruta VPN.

Esto asegura la escalabilidad en el núcleo de la red MPLS y asegura que ningún dispositivo sea un cuello de botella en la escalabilidad.

Seguridad:

Las VPNs MPLS ofrecen el mismo nivel de seguridad que las VPNs orientadas a conexión, pues los paquetes de una VPN no pueden pasar a otra. La seguridad se proporciona:

- En la frontera de la nube MPLS, asegurando que los paquetes recibidos de un cliente sean colocados en la VPN correcta.
- En el *backbone*, pues el tráfico de las VPNs es separado. Algún intento para obtener acceso a un enrutador PE es imposible debido a que los paquetes recibidos de los clientes son paquetes IP, los cuales deben ser recibidos en una interfaz específica que sea identificada únicamente con una etiqueta MPLS.

Facilidad de creación:

Debe ser fácil para los clientes crear nuevas VPNs y comunidades de usuarios. Debido a que las VPNs MPLS son no orientadas a conexión, una conexión no específica punto a punto mapea la topología, según sea requerido y se pueden añadir sitios a la red para formar grupos cerrados de usuarios. Cuando se administra la VPN de esta manera, se habilita a cualquier sitio como miembro de la VPN, maximizando la flexibilidad en la construcción de VPNs.

Direccionamiento flexible:

Para hacer un servicio de VPN más flexible, los clientes de un proveedor de servicios puede diseñar su propio plan de direccionamiento, independiente de los planes de otros clientes del proveedor de servicios. Muchos clientes emplean espacios de direcciones privados, tal como se define en la RFC 1918, y no quieren invertir tiempo y dinero en migrar a un espacio público de direcciones para poder habilitar su conectividad. Las VPNs MPLS permiten que el cliente continúe usando su espacio de direcciones sin la intervención de un traductor de direcciones (NAT, *Network Address Translation*) y éste solamente es requerido cuando se tienen dos VPNs que se quieren comunicar entre sí y cuyos espacios de direcciones se traslapan. Con lo anterior, es posible que el cliente use sus propias direcciones privadas no registradas y pueda comunicarse libremente a través de una red pública IP.

Soporte Integrado De la Clase de Servicio (Cos):

CoS es un requerimiento importante para muchos clientes de VPNs IP ya que proporciona la habilidad de dirigir dos aspectos importantes de las VPNs:

- Comportamiento predecible e implementación de políticas de enrutamiento.
- Soporte a múltiples niveles de servicio en una VPN MPLS.

El tráfico de la red se clasifica y etiqueta en la frontera de la nube MPLS antes de que sea agregado de acuerdo a las políticas definidas por los suscriptores, implementado por el proveedor y transportado a través

del núcleo de la nube MPLS. El tráfico en la frontera y en el núcleo puede ser diferenciado en distintas clases por la probabilidad de retardo o caída.

Migración directa:

Para los proveedores de servicios que desarrollen servicios VPN, se tiene una migración directa. Las VPNs MPLS son únicas debido a que pueden ser construidas sobre múltiples arquitecturas de red, incluyendo IP, ATM, Frame Relay y redes híbridas. La migración para los clientes es sencilla ya que no es necesario soportar MPLS en el enrutador CE y no se requieren modificaciones en la Intranet del cliente.

V.2. Descripción de una VPN MPLS

Una Red Privada Virtual (VPN) es una red en la cual la conectividad del cliente en múltiples sitios es desarrollada sobre una infraestructura compartida con las mismas políticas administrativas que una red privada. La trayectoria entre dos sistemas en una VPN y las características de dicha trayectoria pueden ser determinadas parcial o totalmente por políticas. Si se permite que un sistema en una VPN en particular se comunique con otros sistemas no pertenecientes a la misma VPN, esto se determina en las políticas.

En VPNs MPLS, una VPN generalmente consiste en un conjunto de sitios interconectados por medio del núcleo de una red MPLS de un proveedor de servicios, pero también es posible aplicar diferentes políticas para diferenciar los sistemas que están localizados en el mismo sitio. Las políticas administrativas también pueden ser aplicadas a sistemas que se conecten por par de cobre, pero estas políticas deben basarse en los procesos de verificación *dial-in*.

Un conjunto dado de sistemas puede ser una o más VPNs. Una VPN puede consistir en sitios (o sistemas) de una misma empresa (intranet) o de diferentes empresas (extranet), además de poseer sitios (o sistemas) conectados al *backbone* de un proveedor de servicios o a diferentes *backbone*s de varios proveedores de servicios.

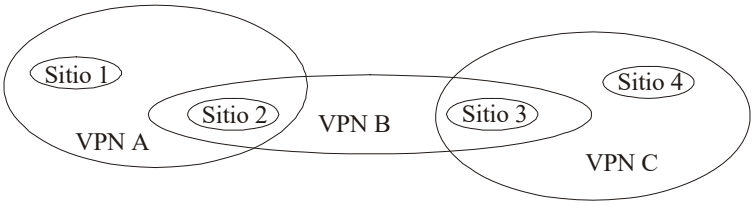


Figura V.2. Conjunto de VPNs.

Las VPNs basadas en MPLS son creadas en la capa 3 y están basadas en el modelo *peer-to-peer* de VPNs, lo cual las hace más escalables y más sencillas de construir y administrar que las VPNs convencionales. Además, los servicios de valor agregado, como el comercio en la red y los servicios de telefonía puede ser fácilmente desarrollados sobre una VPN MPLS en particular debido a que el *backbone* del proveedor de servicios reconoce cada VPN MPLS como una red IP no orientada a conexión segura.

El modelo VPN MPLS realmente es un modelo *peer-to-peer* de VPN que fuerza a que el tráfico sea separado asignando una única tabla de enrutamiento y envío VRF (*VPN Routing & Forwarding*) a cada cliente de la VPN. Esto implica que lo usuarios de una VPN específica no pueden ver el tráfico de afuera de su VPN. la separación de tráfico ocurre sin tuneleo o cifrado de datos debido a que es realizado directamente sobre la red. El *backbone* del proveedor de servicios consta de los enrutadores PE y los enrutadores P. Las VPNs MPLS proporcionan la habilidad de que la información de enrutamiento de una VPN específica esté presente únicamente en aquellos enrutadores PE que están directamente conectados a dicha VPN.

Mientras que la unidad básica de interconexión es el sitio, la arquitectura VPN MPLS permite un control más fino de la interconectividad. Por ejemplo, en un sitio dado, puede ser deseable restringir que sólo algunos sitios o sistemas se conecten a otros sitios. Esto es, ciertos sistemas en el mismo sitio pueden o no ser restringidos a pertenecer únicamente a una intranet.

Un enrutador CE puede pertenecer a múltiples VPNs, aunque sólo esté en un sitio. Cuando un enrutador CE pertenece a múltiples VPNs, una de esas VPNs es considerada la VPN primaria. En general, una VPN primaria de un enrutador CE es la intranet que incluye el sitio del enrutador CE. Un PE puede conectarse a

diferentes CE en cualquier número de sitios diferentes si aquellos CE están en una misma o en diferentes VPNs. Por robustez, un enrutador CE puede estar conectado a diferentes PEs. Un enrutador PE se conecta a una VPN en particular si es un enrutador adyacente al enrutador CE que está en la VPN.

Características de las VPNs MPLS

Las VPNs MPLS tienen las siguientes características:

- Las extensiones MPBGP son usadas para codificar los prefijos de las direcciones IPv4 de los clientes en valores únicos NLRI de VPN-IPv4.

  NLRI se refiere a una dirección destino en MPBGP, por lo que NLRI es considerada “una unidad de enrutamiento”. En el contexto de MPBGP IPv4, NLRI se refiere a un par <Prefijo de red, Tamaño del prefijo> que es transportado en las actualizaciones de enrutamiento en BGP4.
- Los atributos extendidos de comunidad en MPBGP son usados para controlar la distribución de las rutas del cliente.
- Cada ruta del cliente está asociada con una etiqueta MPLS, la cual es asignada por el enrutador PE que origine la ruta. La etiqueta es empleada en dirigir los paquetes de datos hacia el enrutador CE correcto.

  Cuando un paquete es enviado a través de la red del proveedor de servicios, se emplean dos etiquetas. La primera de ellas dirige el paquete al PE de egreso apropiado. La segunda etiqueta indica cómo debe enviar el paquete el enrutador PE de egreso.
- Los mecanismo de Clase de Servicio (CoS) y de Calidad de Servicio (QoS)proporcionan servicio de diferenciación a todos los paquetes de datos del cliente.
- El enlace entre el enrutador PE y el enrutador CE usa el envío tradicional IP.

  El PE asocia a cada CE con una tabla de envío por sitio que contiene solamente el conjunto de rutas disponibles en tal CE.

Principales tecnologías

Son cuatro las tecnologías principales que hacen posible construir VPNs basadas en MPLS:

- MPBGP (*Multiprotocol BGP*) entre PEs que transporten información de enrutamiento de los CEs.
- Filtrado de rutas basado en el atributo extendido comunidad de MPBGP.
- Envío MPLS, que transporta paquetes entre PEs (a través del *backbone* del proveedor de servicios).
- Cada PE tiene múltiples VRFs (instancias de enrutamiento y envío).

V.3. Operación de una VPN MPLS

V.3.1. Introducción

Cada VPN está asociada con una o más VRFs (instancias de enrutamiento y envío). Una VRF define los miembros de una VPN en un sitio del cliente conectado a un enrutador PE. Una VRF consiste en una tabla de enrutamiento IP, una tabla derivada de CEF, un conjunto de interfaces que usan la tabla de envío y un conjunto de reglas y parámetros del protocolo de enrutamiento que controla la información que se incluye en la tabla de enrutamiento.

No es necesario que exista una relación uno a uno entre los sitios del cliente y las VPNs. Un sitio dado puede ser miembro de múltiples VPNs, tal como se muestra en la figura V.3.1. Sin embargo, un sitio sólo puede estar asociado con una (y solo una) VRF. La VRF de un sitio del cliente contiene todas las rutas disponibles para el sitio de la VPN del cual es miembro.

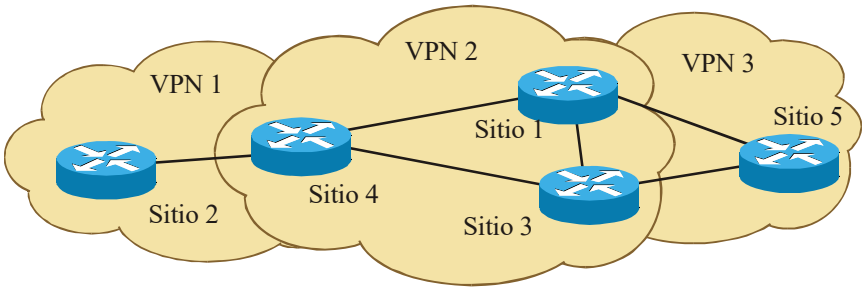


Figura V.3.1. Ejemplo de sitios en varias VPNs

La información de envío de paquetes es almacenada en la tabla de enrutamiento IP y en la tabla derivada de CEF para cada VRF. Un conjunto separado de tablas de enrutamiento y tablas CEF se mantiene para cada VRF. Estas tablas evitan que la información sea enviada hacia fuera de la VPN y también evita que los paquetes ajenos a la VPN sean enviados a un enrutador perteneciente a la VPN.

V.3.2. VRFs. Tablas de VPNs de enrutamiento y envío

La distribución limitada de la información de enrutamiento es claramente necesaria, pero por sí sola no es suficiente para controlar la conectividad. Esto se debe a que un enrutador PE puede tener sitios de diferentes VPNs conectados a él. Si éste PE tiene una sola tabla de enrutamiento, entonces esta tabla debería contener todas las rutas de todas esas VPN. Para poder solucionar este problema y que las VPN queden separadas por completo, es necesario que los enrutadores PE utilicen múltiples tablas de enrutamiento (llamadas VRFs), cada una pertenecientes a una VPN.

En un caso extremo, cada sitio conectado al enrutador PE puede tener su propia tabla de enrutamiento (que es lo que pasa, por ejemplo, cuando cada sitio conectado a un enrutador PE dado está en su propia VPN). Sin embargo, cuando un enrutador PE tiene muchos sitios con una VPN común, todos estos podrían compartir su tabla de enrutamiento.

La VRF es un elemento clave en la tecnología VPN MPLS. Las VRFs existen únicamente en los enrutadores PE y puede haber más de una VRF en un PE. Una VRF es una tabla de enrutamiento que contiene rutas que están disponibles para un conjunto particular de sitios. Las VRFs usan la tecnología CEF (para el caso de equipos Cisco), por lo que en la VPN debe estar habilitado el CEF.

Una VRF está asociada con los siguientes elementos:

- Tabla de enrutamiento IP
- Tabla de envío, derivada de la tecnología CEF
- Un conjunto de interfaces que usan la tabla de envío
- Un conjunto de protocolos de enrutamiento y vecinos de enrutamiento que inyectan información a la VRF

Cada VRF en el enrutador PE está basada en dos fuentes de información. La primera fuente es un conjunto de rutas que el enrutador PE recibe desde sus clientes CE directamente conectados. Una tabla de enrutamiento asociada a una VPN en particular podría ser poblada por rutas que el enrutador PE recibe de los enrutadores CE que tiene directamente conectados y que pertenecen a la VPN. La segunda fuente es el conjunto de rutas que el enrutador PE recibe de los demás enrutadores PE. Para esta segunda fuente, el filtrado de rutas está basado en atributos de comunidades BGP que determinan las rutas que un enrutador PE podría poner en su tabla de enrutamiento.

Cada PE mantiene una o más VRFs. El *software* de la VPN busca la dirección IP destino de un paquete específico en la VRF apropiada solamente si el paquete llega por medio de una interfaz directamente conectada que esté asociada con dicha VRF. Cuando se envía el paquete al PE destino (esto es, al PE que tiene conectado el CE destino), se le añade una etiqueta MPLS que indica a qué VRF pertenece el paquete y cómo manejarlo para entregarlo al CE correcto y de ahí, al *host* final.

Un enrutador PE usa sus múltiples tablas de enrutamiento para poder maniobrar los paquetes que recibe desde sus sitios que están directamente conectados. En la mayoría de los casos, cada puerto del cliente en el

enrutador PE está asociado a un tiempo provisional dentro de una tabla de enrutamiento particular. En el tiempo de envío el puerto de entrada en el enrutador PE determina qué tabla de enrutamiento va a usar el enrutador PE para enviar el paquete.

Una VRF es bautizada de acuerdo a la VPN o a los servicios que preste la VPN y al papel del CE en la topología.

V.3.3. *Route Distinguishers, Route Targets* y direcciones VPN-IPv4

Las VPNs basadas en MPLS emplean BGP para comunicarse entre Pes y así facilitar el intercambio de rutas del cliente. Esto es llevado a cabo por medio de extensiones BGP que transportan otras direcciones además de las direcciones IPv4.

BGP asume que las direcciones IP son únicas. Esta suposición es incorrecta para el ambiente VPN del proveedor de servicios, en donde el mismo bloque de direcciones IP (direccionamiento privado) puede ser simultáneamente ocupado por múltiples clientes VPN. Así que al usar BGP se necesita ver cómo hacerle para usar BGP en un ambiente en donde las direcciones IP no son únicas. Una solución obvia sería cambiar el direccionamiento para que las direcciones IP que se utilizan sean únicas, lo cual lleva a las direcciones VPN-IP.

Por definición, una dirección VPN-IP se constituye concatenando a una dirección IP un campo de longitud fija que es el *Route Distinguisher*, el cual está estructurado para permitir que cada proveedor de servicios pueda crear su propio RD sin el riesgo de que este el mismo RD sea asignado por otro proveedor. Por definición, un *Route Distinguisher* consiste en tres campos:

- Tipo: 2 bytes.
- Número de Sistema Autónomo: 2 bytes.
- Número asignado: 4 bytes.

El campo del Número de Sistema Autónomo contiene el número del sistema autónomo del proveedor de servicios. El Número Asignado es controlado por el proveedor de servicios. En los casos comunes, un proveedor asigna sólo un valor a cada VPN, lo que significa que dos VPNs no pueden tener *Route Distinguishers* iguales ya que es lo que hace única a la VPN.

Un enrutador PE del proveedor de servicios puede aprender un prefijo IP de un enrutador CE por medio de rutas estáticas, a través de sesiones BGP con el CE o a través de algún protocolo de enrutamiento establecido entre el PE y el CE. El prefijo IP es un miembro de la familia de direcciones IPv4.

Después de que el prefijo IP es aprendido, el PE lo convierte en un prefijo VPN-IPv4 combinándolo con un valor de ocho bytes que es el *Route Distinguisher*. El prefijo generado ahora es un miembro de la familia de direcciones VPN-IPv4, el cual sirve para identificar como única a la dirección del cliente, incluso si el sitio del cliente está usando una dirección IP globalmente privada no registrada y, por lo tanto, que no es única. El RD tiene significado local, es decir, sólo tiene significado en el PE donde se configura. El RD sirve para hacer únicas las direcciones IP de una VPN en un PE particular.

Para transportar información a través del *backbone*, se emplean además otros identificadores llamados *Route Target*. Los *Route Target* son comunidades empleadas en por MPBGP y tienen el propósito de indicar la membresía a cierta VPN, permitiendo así el soporte de VPN´s con topologías complejas. Cuando se exporta una dirección VPNv4 en un PE, se asocia esta un RT. Cuando el PE en el otro extremo recibe la ruta VPNv4, el criterio para aceptarla en una u otra VRF es el RT que tiene asociado. Si la dirección es aceptada, se extirpa de esta la parte correspondiente al RD inicialmente asociado por el PE que la origino, para posteriormente añadirle el RD asociado a la VRF que acepto tal dirección.La de hacer único (en todo el *backbone*) el valor del prefijo de la dirección IPv4. El valor del RT debe ser un valor único globalmente para evitar conflictos con otros prefijos y puede o no ser el mismo que el valor del *Route Distinguisher*, pero ambos están asociados desde la configuración.

La etiqueta MPLS es parte de una actualización de enrutamiento en BGP. Dicha actualización transporta también el direccionamiento y la información sobre qué tan alcanzable es un destino. Cuando el RT es único a través de toda la red VPN MPLS, la conectividad es establecida aún si diferentes clientes usan direcciones IP no únicas.

Desde el punto de vista de BGP, el manejo de rutas para direcciones VPN-IP no es diferente que manejar rutas de direcciones IP. MPBGP tiene la capacidad de hacer que BGP sea capaz de manipular rutas para distintas familias de múltiples direcciones. Cabe resaltar que la estructura de las direcciones VPN-IP así como la del *Route Distinguisher* es totalmente opaca a BGP (cuando BGP compara dos prefijos de direcciones VPN-IP, se ignora la estructura). En este sentido, no se introduce ningún mecanismo, sino que se usan los existentes, como por ejemplo, las comunidades BGP, el filtrado de rutas basado en comunidades, uso de BGP *Route Reflectors*, *BGP Refresh*, etc. Todos ellos son aplicables para las VPN-IP.

El uso de las direcciones VPN-IP es para el proveedor de servicios. La conversión de VPN-IP a IP se realiza en los enrutadores PE. Para cada VPN directamente conectada, un enrutador PE es configurado con un *Route Distinguisher*. Cuando el enrutador PE recibe una ruta de un enrutador CE directamente conectado, el enrutador PE identifica la VPN a la que el enrutador CE pertenece y antes de exportar esta ruta a MPBGP, convierte la información alcanzable de esta ruta de IP a VPN-IP usando el *Route Distinguisher* que fue configurado para esa VPN. Por otro lado, cuando un enrutador PE tiene que importar rutas desde MPBGP, el enrutador PE convierte la información alcanzable de estas rutas de VPN-IP a IP.

Cuando se desea intercambiar información entre PEs, se emplea el *Route Target*, para importar y exportar rutas a ciertas VRF´s distintivo que emplea MPBGP para intercambiar rutas en la nube MPLS. El mecanismo que se sigue para intercambiar información en una nube MPLS empieza en un enrutador PE. Al momento de iniciar el intercambio de información, la dirección VPN-IP sufre un cambio: el *Route Distinguisher* se retira y se sustituye por el *Route Target*. El RT se concatena al prefijo IP y sólo así la "nueva" dirección VPN-IP podrá transmitirse en la red MPLS. Al llegar al enrutador PE destino, el RT es retirado y en su lugar se coloca el RD que le corresponde.

V.3.4. Distribución limitada de la información de enrutamiento

BGP distribuye información sobre qué tan alcanzable es un prefijo VPN-IPv4 para cada VPN. la comunicación BGP toma lugar en dos niveles: en los dominios IP conocidos como Sistemas Autónomos (IBGP) y entre Sistemas Autónomos (EBGP). Las sesiones entre PE y PE o entre un PE y un *Route Reflector* son sesiones IBGP, mientras que las sesiones entre PE y CE son EBGP.

BGP propaga información sobre qué tan alcanzable es un prefijo VPN-IPv4 entre los enrutadores PE por medio de extensiones de BGP, es decir, por medio de MPBGP, el cual define el soporte para otras familias que no son IPv4. Esto se hace de tal forma que se asegure que las rutas de una VPN dada sean aprendidas solamente por miembros de esa VPN, habilitando la comunicación entre todos los miembros de la VPN.

Dentro de una VPN se necesita de un mecanismo que permita controlar la conectividad entre los sitios, al cual se le llama distribución limitada de la información de enrutamiento. La razón por la cual la distribución limitada de la información de enrutamiento da un control en la conectividad es porque el flujo de los datos está determinado por el flujo de la información de enrutamiento, es decir, al limitar el flujo de la información de enrutamiento, se limita el flujo de los datos. En otras palabras, la conectividad y el flujo de los datos depende de las tablas de enrutamiento y el contenido de estas tablas puede ser controlado limitando el flujo de la información de enrutamiento. Para entender cómo la distribución limitada de la información de enrutamiento es usada en el contexto de VPN MPLS, se enfatiza la distribución de la información de enrutamiento, que está compuesta de cinco partes:

1. La información de enrutamiento es propagada desde el sitio cliente hasta el proveedor de servicios. Esto es, la información es propagada desde el enrutador CE hasta el enrutador PE. Hay muchas maneras de propagar esta información, como es RIP, OSPF, BGP o rutas estáticas.
2. En el enrutador de ingreso PE la información es exportada a MPBGP.

3. Esta información es distribuida dentro del proveedor de servicios a través de los enrutadores PE usando MPBGP.
4. Este paso es exactamente opuesto al paso dos, ya que en el enrutador de egreso PE, la información de enrutamiento es importada de MPBGP.
5. Este paso es opuesto al paso 1, ya que la información de enrutamiento es mandada desde los enrutadores PE de egreso a los enrutadores CE de los sitios. Hay varias opciones para hacer esto, como RIP, OSPF, BGP o por rutas estáticas.

Al limitar la distribución de la información de enrutamiento se usa una técnica de filtrado de rutas basado en el atributo comunidad de BGP en donde la comunidad BGP actúa como un identificador que se añade a una ruta anunciada.

En el paso 2, como resultado de su configuración local, el enrutador de ingreso PE adjunta los atributos a la ruta y, así, la ruta es exportada a MPBGP. En el paso 4, como resultado de la configuración, el enrutador de egreso PE usa los atributos contenidos por una ruta para controlar la importación de rutas de MPBGP hacia el sitio del cliente (enrutador CE).

Hay una flexibilidad significativa en el mecanismo del filtrado de rutas basado en el atributo comunidad de BGP. En una punta del extremo, un enrutador de ingreso PE puede aplicar un atributo particular a todas las rutas provenientes de un sitio particular (desde un enrutador particular CE), mientras que en la otra punta de la red, el enrutador podría aplicar una comunidad distinta a cada ruta individual. La flexibilidad del mecanismo del filtrado de rutas permite flexibilidad en la conectividad entre sitios dentro de una VPN, así como también se controla la conectividad por el filtrado del enrutamiento basado en el atributo comunidad. Esto le permite a un proveedor de servicios usar un sólo mecanismo común para soportar clientes VPN con diversas políticas de conectividad intersitios.

Una distribución limitada de la información de enrutamiento se ejecuta en los pasos 2 y 4, los cuales son controlados por el proveedor de servicios. Como este mecanismo de limitación de flujo de la información de

enrutamiento puede ser manejado por el proveedor de servicios, el cliente VPN no tiene que estar involucrado dentro de este mecanismo, por lo que se pueden tener clientes que no sean expertos en enrutamiento IP.

Para mostrar el funcionamiento de este mecanismo de distribución limitada de la información de enrutamiento en un contexto de VPN basado en MPLS se puede ver la figura V.3.4. y examinar el flujo de la información de enrutamiento del Sitio 1 al Sitio 3 de la VPN A.

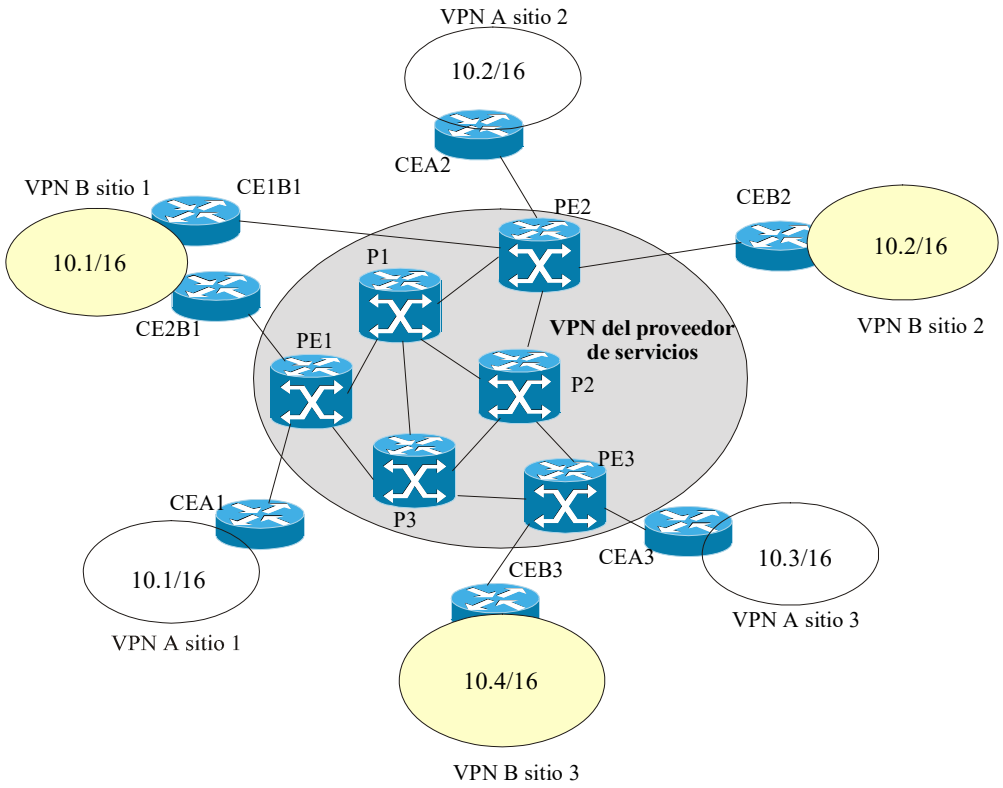


Figura V.3.4. Ejemplo de distribución limitada de la información de enrutamiento

En el primer paso, la ruta 10.1/16 es distribuida desde el enrutador CE en el Sitio 1 de la VPN A (CEA1) al enrutador PE al cual está conectado su sitio (PE1). Esta distribución podría ser, por ejemplo, con RIP. En el paso 2, esta ruta es exportada a MPBGP; el enrutador de ingreso PE (PE1), controlado por su propia configuración, adjunta el atributo BGP a la ruta. En el paso 3 esta ruta es distribuida a otro enrutador PE por

procedimientos BGP normales. En el paso 4, en el enrutador de egreso PE (PE3), la ruta es importada desde MPBGP. Esta importación es controlada por el filtrado de rutas que se ejecuta (bajo el control de su propia configuración) por PE3, y está basado en el atributo comunidad que transporta la ruta. Finalmente en el paso 5, la ruta es distribuida desde el enrutador PE3 hacia el enrutador CEA3, en el Sitio 3 de la VPN . Esta distribución puede ser por medio de RIP (como la que se usó en el paso 1) o por otro protocolo de enrutamiento (OSPF o BGP).

Hay características muy importantes en el mecanismo usado por VPN MPLS para controlar la interconectividad intersitios y que tienen implicaciones en la capacidad de escalamiento. Primero que nada, en una VPN, un enrutador CE mantiene el enrutamiento directo o *peer* sólo con el enrutador PE directamente conectado, pero no con los enrutadores CE en otros sitios de la VPN. Por ejemplo, en la figura V.3.4., el enrutador CEA1 tiene un enrutamiento directo o *peer* sólo con PE1, pero no con CEA2 o CEA3. Como resultado, el número de vecinos de enrutamiento que los enrutadores CEs deben mantener es constante e independiente del número total de sitios en la VPN. Esto facilita soportar grandes dimensiones de VPN (de alrededor de cientos o miles de sitios por VPN), ya que la cantidad de enrutamiento directo o *peer* que una ruta CE tiene que ejecutar es independiente de este número total de sitios en la VPN. En contraste, cuando se quiere una conectividad de malla completa entre sitios con el modelo *extendido*, la necesidad de tener un enrutamiento directo o *peer* de malla completa entre todos los sitios implica que la cantidad de enrutamiento directo crece con el número de sitios. Esto significa que la escalabilidad del modelo *extendido* en el área del enrutamiento directo es inferior al de una VPN MPLS.

La segunda característica tiene que ver con agregar o eliminar sitios dentro de una VPN dada. Para un proveedor de servicios de VPNs MPLS, agregar un sitio implica sólo cambiar la configuración del PE al que se conectará, ya que la configuración es independiente del número de sitios en la VPN. En contraste, cuando se tiene una malla completa en el modelo *extendido*, los cambios en la configuración son proporcionales al número de sitios que son agregados. Por esta razón, en cuanto a cambios o agregaciones en el número de sitios dentro de una VPN determinada, la aproximación por medio de VPN MPLS resulta mejor que con una conectividad de malla completa de un modelo de VPN *extendido*.

Finalmente se observa que el enrutador PE tiene que mantener sólo las rutas para las VPNs cuyos sitios están directamente conectados al enrutador PE (los sitios tienen enrutadores CE conectados a enrutadores PE). En la figura V.3.4. PE1 tiene que mantener rutas solo para las VPN A y B. Aunque tal vez haya algunas otras VPNs (no mostradas en la figura) que no tienen sitios conectados a PE1, para estas VPN no se tiene que mantener las rutas.

Un problema al usar filtrado de rutas basado en atributos BGP es que se tienen 2^{16} atributos por proveedor de servicios. Cada comunidad es de 32 bits de contenido y está estructurada como una concatenación de 16 bits para el número de Sistema Autónomo y 16 bits de asignación local. Es decir, si se necesita un atributo por VPN, lo que se obtiene es que cada proveedor de servicios tiene una posibilidad de 2^{16} clientes. Pero hay otra opción que son los atributos extendidos de comunidad en BGP, con los cuales esta cifra se puede ampliar a 2^{32} posibilidades por proveedor de servicios. Se puede notar que en cada comunidad extendida existe el número de Sistema Autónomo y es globalmente único, así que cada proveedor de servicio debe cuidar que efectivamente sean únicos.

Un caso sencillo del empleo del atributo comunidad en BGP es cuando tenemos sitios dentro de una VPN de malla completa. En este ejemplo, para esta VPN se usa solo una comunidad (a la cual se le da el nombre en este ejemplo de $C_{cerrado}$). En un enrutador PE que tiene sitios de esa VPN conectados a él, todas las rutas de los sitios son exportadas a MPBGP con la comunidad $C_{cerrado}$. Por otro lado, en todos estos enrutadores PE las rutas que son importadas dentro de la tabla de enrutamiento asociada con esta VPN, son sólo las rutas que tiene esa comunidad. Usando la figura V.3.4., si la VPN A necesita una conectividad de malla completa entre sus tres sitios, el proveedor de servicios asigna sólo una comunidad BGP para esta VPN ($C_{VPN\ A}$). Cuando el enrutador PE1 exporta las rutas que recibe de CEA1 a MPBGP, el proveedor las exporta (bajo el control de su propia configuración) con la comunidad $C_{VPN\ A}$. Por otro lado, las únicas rutas que importa el enrutador PE1 desde el MPBGP dentro de su tabla de enrutamiento asociada con la VPN A, son las rutas que tienen la comunidad $C_{VPN\ A}$.

Otro ejemplo es cuando una VPN requiere de una conectividad entre sus sitios *hub-and-spoke*, donde todos los sitios *spoke* pueden comunicarse con cualquier otro solo por medio de su sitio *hub*. En este caso, se necesita no solo uno, sino dos diferentes comunidades. Una comunidad es asociada al *hub* (C_{HUB}) y la otra con el *spoke* (C_{SPOKE}). Un enrutador PE que tiene sitios *spoke* conectados a él puede exportar a MPBGP las rutas recibidas desde estos sitios con la comunidad BGP C_{SPOKE} y puede importar desde MPBGP (dentro de la tabla de enrutamiento asociada con la VPN) sólo las rutas de la comunidad C_{HUB}. El enrutador PE que tiene el sitio *hub* conectado a él puede exportar rutas recibidas desde estos sitios con la comunidad BGP C_{HUB} y también puede importar dentro de la tabla de enrutamiento asociada con esa VPN sólo las rutas cuya comunidad sea C_{SPOKE}. Estos dos ejemplos ilustran algunas de las formas de conectividad que pueden ser controlados usando atributos de comunidades.

En este punto se pueden comparar los roles jugados por los *Route Distinguishers* y las comunidades BGP. Se tienen dos problemas separados y, por lo tanto, se tienen dos mecanismos. El primer problema es el cómo tratar con direcciones globales que no son únicas. Para resolver este problema se introduce un nuevo tipo de dirección, VPN-IP y se usa el *Route Distinguisher* para hacer esas direcciones únicas. El *Route Distinguisher* no es usado para limitar la conectividad, así que no es usado para el filtrado de rutas. El segundo problema es cómo limitar la conectividad, esto se soluciona usando un filtrado de rutas basado en los atributos de comunidad de BGP (*Route Target*), pero a su vez este no ayuda en el problema de las direcciones únicas.

Se puede ver que el *Route Distinguisher* no puede ser usado por más de una VPN, mientras que una VPN dada puede usar múltiples *Route Distinguishers*. Y además, un mismo atributo de comunidad BGP no puede ser usado por más de una VPN, mientras que una VPN puede usar múltiples comunidades extendidas. En general, un *Route Distinguisher* o una comunidad extendida por separado no pueden identificar una VPN.

Comunidades de Route Targets y de CEs

El mecanismo por el cual una VPN MPLS controla la distribución de la información de enrutamiento de la VPN es a través de comunidades extendidas MPBGP de RTs. Una comunidad extendida MPBGP es un valor con una estructura de ocho bytes. Una VPN MPLS usa las comunidades RT como a continuación se comenta:

- Cuando una ruta VPN aprendida de un enrutador CE es inyectada a BGP, se asocia a ella una lista de atributos extendidos de comunidades *Routes Targets* VPN. Típicamente, la lista de los valores de la comunidad RT es establecida a partir de una lista de exportación de RTs asociados a la VRF de la cual la ruta fue aprendida.
- Una lista de importación de comunidades extendidas RT es asociada a cada VRF. La lista de importación define los atributos extendidos de comunidades RT que una ruta debe tener para ser importada a la VRF. Por ejemplo, si la lista de importación para una VRF particular incluye las comunidades RT A, B y C, entonces cualquier ruta que transporte alguna de estas comunidades RT –A, B o C- es importada a la VRF.

Además, una VPN puede ser organizada en subconjuntos llamados Comunidades de enrutamiento CE o CERCs (por su nombre en inglés, *CE Routing Communities*). Una CERC describe cómo se pueden comunicar los CEs de una VPN entre sí, es decir, describe la topología lógica de la VPN. En este caso, se puede emplear un software para formar una variedad de topologías VPN entre los CEs para construir comunidades *hub & spoke* o comunidades *full mesh*. Las CERCs se construyen por bloques para permitir el desarrollo de topologías más complejas y que sea posible la conectividad entre los CEs.

Los tipos más comunes de VPNs son el *hub & spoke* y el *full mesh*:

- Una CERC *hub & spoke* es aquella en la que uno o algunos CEs actúan como *hubs* y los CEs restantes como *spokes*, éstos comunicándose únicamente con el o los *hubs*, pero nunca entre sí.

  En un ambiente VPN MPLS *hub & spoke*, los enrutadores *spoke* tienen un mismo RT con el que exportan sus rutas y el enrutador PE *hub* usa otro RT (diferente al de los *spokes*) para exportar sus rutas. Por otro lado, el enrutador PE *hub* sólo importa las rutas exportadas por los enrutadores PE *spoke*, y los

enrutadores PE *spoke* sólo importan las rutas que el enrutador *hub* PE exportó. Tanto los *spokes* como el *hub* emplean el mismo RD.

Para poder usar el sitio del hub como un punto de tránsito para la conectividad en ese ambiente, los sitios spoke exportan sus rutas al hub. Los *spokes* pueden hablar con el *hub*, pero los *spokes* nunca pueden hablar con otros *spokes*. Así, cuando el hub exporta sus rutas, todos los spokes las conocen y pueden importarlas. Cada spoke tiene las mismas rutas, así que cuando uno de ellos las exporta, el hub las advierte y no hay necesidad de que los demás spokes también exporten sus rutas.

Debido a la implementación actual de VPN MPLS, se deben aplicar diferentes RD para las VRFs de cada spoke.

- Una CERC *full mesh* es aquella en la cual todos los CEs están conectados entre sí, es decir existe comunicación de todos contra todos.

  Cada CE VRF tiene un RT y un RD únicos. De esta forma, cuando una VRF CE desea dar a conocer sus rutas, las exporta indicando el RT que le pertenece y viceversa, al importar rutas provenientes de cierto CEPE, se indica elige el RT correspondiente al CE a la VRF con la que se desea comunicación; en este caso se usara el mismo valor que el RD localmente configurado como el RT para importar rutas.

Para construir topologías muy complejas, es necesario dividir la conectividad requerida entre los CEs en grupos, donde cada grupo tiene una topología *full mesh* o un patrón *hub & spoke*. (un CE puede estar en uno o más grupos al mismo tiempo). Cada subgrupo en la VPN necesita su propia CERC. Cualquier CE que se encuentre solamente en un grupo debe unirse a su CERC correspondiente (como un *spoke* si fuera necesario), pero si el CE pertenece a varios grupos, aquél se añade a todos los grupos relevantes en una petición de servicio. Dada esta información, el *software* hace el resto, asignando valores de RTs y tablas VRFs para arreglar la conectividad tal como el cliente lo requiere.

V.3.5. MPLS como mecanismo de envío

Para poder enviar paquetes IP por las rutas expresadas en términos de direcciones VPN-IP, se utiliza MPLS. MPLS permite asignar una etiqueta a la información que tiene. Para ilustrar cómo funciona este mecanismo, se muestra la siguiente figura:

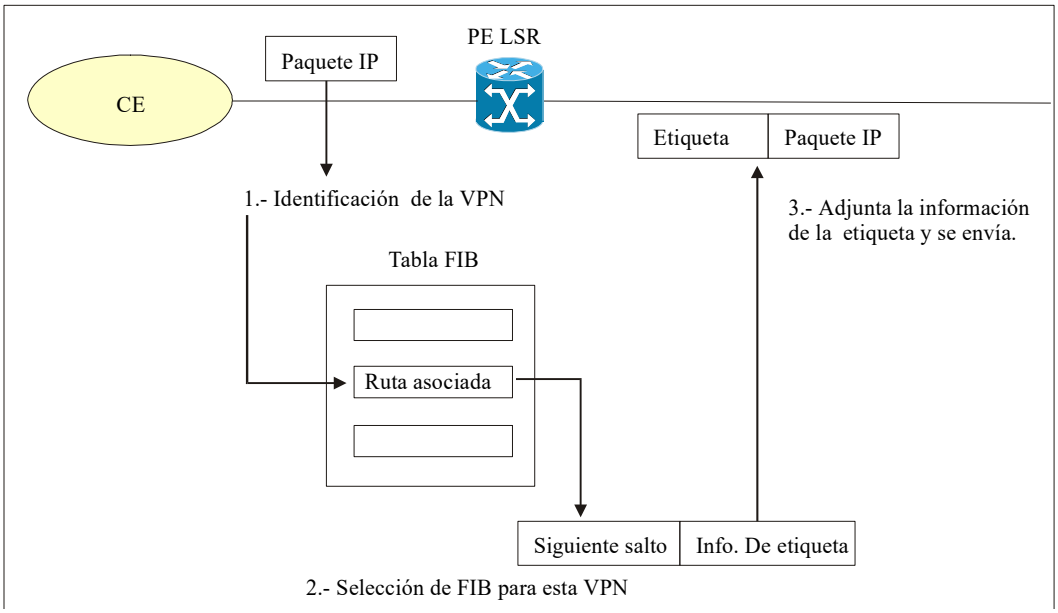


Figura V.3.5.1. Imposición de la etiqueta en un enrutador PE

Desde el punto de vista de MPLS el enrutador PE es un LSR frontera. Este enrutador convierte un paquete que no está etiquetado en uno etiquetado y viceversa.

Cuando un enrutador CE manda un paquete IP a su enrutador PE que está directamente conectado, el enrutador PE usa el puerto de ingreso (la interfaz en la cual el enrutador PE recibe el paquete), identifica a qué VPN pertenece ese CE, o dicho en forma más precisa, identifica la tabla de enrutamiento FIB (*Forwarding Information Base*) asociada con la VPN. Cuando la FIB es identificada, el enrutador PE ejecuta una búsqueda normal de direcciones IP en esta FIB usando el destino del paquete IP. Entonces, la FIB hace su búsqueda y el enrutador PE añade la información de la etiqueta apropiada al paquete y lo manda.

De acuerdo a la información de enrutamiento almacenada en la tabla VRF de enrutamiento IP y en la tabla VRF derivada de CEF, los paquetes son enviados a su destino usando MPLS. Un enrutador PE asigna una

etiqueta a cada prefijo de un cliente aprendido de un enrutador CE e incluye la etiqueta en la información sobre qué tan alcanzable es una red para el prefijo que anuncia a otros PEs. Cuando un PE envía un paquete recibido de un CE a través de la red del proveedor, se asigna al paquete la etiqueta aprendida del PE destino. Cuando el enrutador PE destino recibe el paquete etiquetado, le retira la etiqueta y lo usa para dirigir el paquete al enrutador CE adecuado. El envío de etiquetas a través del *backbone* del proveedor está basado en la conmutación dinámica de etiquetas o en las trayectorias de ingeniería de tráfico

Para mejorar el escalamiento, se emplea una jerarquía de conocimiento de enrutamiento. De esta forma, los enrutadores P no manejan la información de la VPN, reduciendo la carga en los enrutadores P. Para implementar este tipo de jerarquías, se utilizan dos niveles de etiquetas. La primer etiqueta está asociada con la ruta del enrutador PE de egreso y provee el envío desde un enrutador PE de ingreso a un enrutador PE de egreso. La segunda etiqueta controla el envío en el enrutador PE de egreso. La etiqueta de primer nivel puede ser distribuida vía LDP, o si el proveedor de servicio quiere, por medio de RSVP o LDP. La etiqueta de segundo nivel es distribuida por medio de BGP, junto a las rutas VPN-IP.

Cabe hacer la aclaración de que una ruta VPN-IP distribuida vía BGP lleva como atributo de siguiente salto la dirección del enrutador PE que origina la ruta. Esta ruta o siguiente salto es provisto por medio de los procedimientos de enrutamiento entre dominios del proveedor de servicios. Esta información (contenida en el atributo del siguiente salto) provee una unión entre el enrutamiento interno del proveedor (enrutamiento entre dominios) y las rutas VPN (que desde el punto de vista del proveedor de servicios, son rutas externas).

Para ilustrar el concepto de jerarquía de conocimiento de enrutamiento se muestra la figura V.3.5.2.

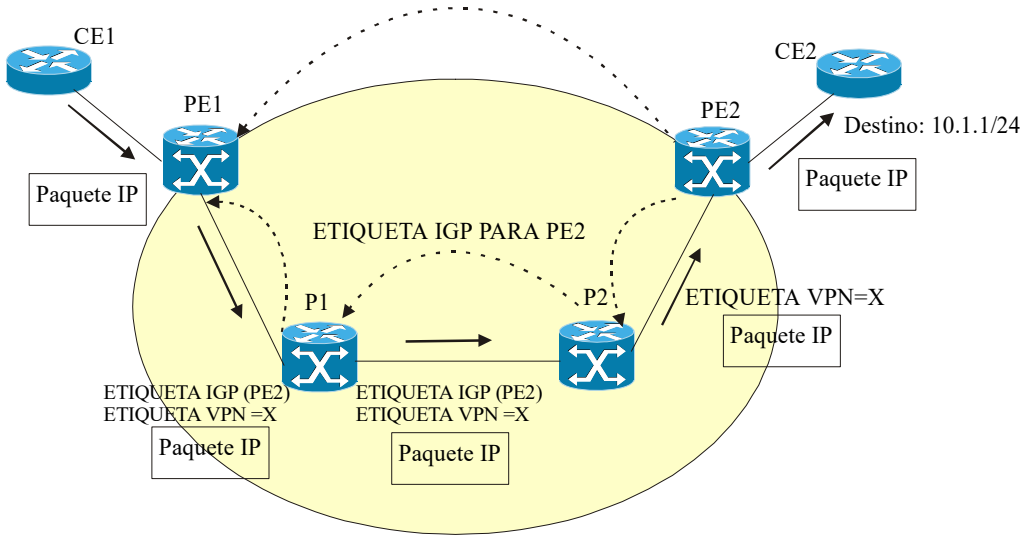


Figura V.3.5.2. Imposición de etiquetas en el enrutador PE

Se muestran dos sitios dentro de una VPN en particular, en donde cada sitio está representado sólo por sus enrutadores CE (CE1 y CE2). Ambos enrutadores PE1 y PE2 están configurados con un *Route Distinguisher* apropiado para ser usado por la VPN, y también con una comunidad BGP apropiada para ser usada para exportar rutas a MPBGP. En el PE1, la interfaz que conecta a PE1 con CE1 está asociada con la tabla de enrutamiento para esa VPN.

Cuando PE2 recibe desde CE2 una ruta con la información de alcance 10.1.1/24, PE2 convierte la información de alcance de esta ruta de una dirección IP a una dirección VPN-IP, adjunta el atributo de la comunidad BGP y se exporta esta ruta a MPBGP. El atributo BGP del siguiente salto de esta ruta es un conjunto de direcciones de PE2 (usualmente esta dirección es la dirección *loopback* del enrutador). Además de toda la información convencional de BGP, la ruta también lleva una etiqueta que está asociada con la ruta VPN-IP. Esta información es distribuida al enrutador PE1 usando BGP (líneas punteadas de la figura V.3.5.2). Cuando PE1 recibe la ruta, convierte la ruta de VPN-IP a IP y la usa para completar su tabla de enrutamiento asociada con la VPN.

Existe una trayectoria o LSP desde PE1 hasta PE2, la cual está asociada con la ruta a PE2 y está establecida y mentada por LDP o ingeniería de tráfico MPLS. El PE1 tendrá una etiqueta asociada a la dirección IP del PE2, la cual será usada para enviar todo el tráfico correspondiente a redes IP aprendidas por medio de BGP

desde el PE2. A su vez, el PE2 envía etiquetas asociadas a las redes que anuncia a través de BGP. En este punto, la tabla de enrutamiento en PE1 contiene una ruta para 10.1.1/24 y la pila de etiquetas, donde la primera es la etiqueta que PE1 recibe vía BGP y la segunda es la etiqueta asociada con el enrutador PE2.

Si CE1 manda un paquete con una dirección destino 10.1.1.1, al llegar el paquete al PE1, éste determina la tabla de enrutamiento apropiada y entonces ejecuta la búsqueda dentro de esta tabla. Como resultado de esta búsqueda, PE1 adjunta dos etiquetas al paquete y manda el paquete a P1. P1 usa la segunda etiqueta cuando hace la decisión de envío y luego envía el paquete a P2. P2 es el penúltimo salto con respecto al LSP asociado con una ruta a PE2. Por ello, P2 retira la etiqueta de salida antes de mandar el paquete a PE2 y entonces recibe el paquete. Éste usa la etiqueta que lleva el paquete (la etiqueta que PE2 distribuyó a PE1 por medio de BGP) para hacer la decisión. Posteriormente, PE2 desmonta la etiqueta y manda el paquete a CE2.

Se puede hacer un análisis sencillo del uso de la jerarquía de conocimientos de enrutamiento: si suponemos que la red del proveedor de servicios consiste en 200 enrutadores (enrutadores P y PE), los cuales soportan 10,000 VPNs, y cada VPN con 100 rutas, sin el uso de una jerarquía de conocimiento de enrutamiento cada enrutador P tendría que necesitar en su tabla de enrutamiento 10,000*100=1,000,000 de rutas. En cambio, con MPLS solo se necesitarán 200 rutas.

V.4. Seguridad en VPNs MPLS

V.4.1. Introducción

La seguridad es un componente muy importante para una solución efectiva de VPN y, por esta razón, la meta de una VPN MPLS es hacer que su seguridad sea comparable con aquella que las VPN ATM o Frame Relay ofrecen, es decir, en rasgos generales se pretende evitar que por la interconexión o una mala configuración los datos se mezclen entre VPNs.

Para empezar, el envío dentro de una VPN de un proveedor de servicios está basado en una conmutación de etiquetas y no en envío tradicional IP, es decir, el envío dentro del proveedor de servicios no está determinado por las direcciones de IP de los paquetes. Las LSPs asociadas con las rutas VPN-IP son originadas y terminadas sólo en los enrutadores PE. En un enrutador PE, estas LSP están asociadas con tablas particulares de envío y las tablas de envío están asociadas con las interfaces en el enrutador PE. Con lo anterior, se observa que estas interfaces están asociadas con ciertas VPN.

V.4.2. Características de seguridad de MPBGP

MPBGP es un protocolo de distribución de información que define quién puede hablar con quién. Los miembros de una VPN dependen de los puertos lógicos al inicio de la VPN, que es donde MPBGP asigna un valor único llamado *Route Distinguisher* (RD).

Los RDs son desconocidos para los usuarios, haciendo imposible que un usuario de cierta VPN ingrese a otra. Sólo si se pre-asignan los puertos, es posible que participen en la VPN. En una VPN MPLS, MPBGP distribuye las tablas FIB de una VPN a los miembros de dicha VPN., proporcionando seguridad nativa por medio de la separación lógica del tráfico en la VPN. Además, los vecinos IBGP de los enrutadores PE pueden ejecutar MD5 al momento de establecer las relaciones IBGP entre vecinos, reduciendo la probabilidad de que se introduzca un spoof en segmentos TCP de las conexiones IBGP de los PEs.

El proveedor de servicios, no el cliente, asocia una VPN específica con cada interfaz de un PE. Los usuarios sólo pueden participar en una intranet o en una extranet si residen en el puerto lógico o físico correcto y si tiene el RD adecuado, lo cual hace prácticamente imposible que se pueda ingresar a una VPN ajena.

En el núcleo, un IGP estándar, tal como OSPF o IS-IS, distribuye la información de enrutamiento. Los enrutadores PE establecen trayectorias entre ellos a través del núcleo (enrutadores P) usando LDP para

comunicar información de conmutación de etiquetas. La información de conmutación de etiquetas para rutas externas (esto es, para el cliente) es distribuida entre los enrutadores PE usando MPBGP en lugar de LDP debido a que la información VPN IP ya distribuida es fácil de adjuntar.

V.4.3. Seguridad a través de la resolución de direcciones IP

Las redes VPNs MPLS son más fáciles de integrar con clientes que posean redes basadas en IP. Los clientes pueden interconectarse con un proveedor de servicios sin cambiar sus aplicaciones intranet debido a que las redes MPLS están construidas de tal forma, que no es importante el tipo de aplicaciones de los clientes, pues inclusive pueden usar transparentemente su espacio de direcciones IP sin ser traducidas por NAT ya que las VPNs tienen un identificador único.

Las VPNs MPLS no se enteran de la existencia de otras VPNs MPLS. El tráfico es separado en las VPNs usando una tabla de envío distinta lógicamente de las demás y un RD para cada VPN. De acuerdo a la interfaz de entrada, el enrutador PE selecciona una tabla de envío específica, la cual lista únicamente los destinos válidos en la VPN. Para crear extranets, un proveedor debe configurarlas explícitamente.

La tabla de envío para un PE contiene como entradas solamente las direcciones de los miembros de la misma VPN y se rechazan las peticiones para direcciones no listadas en dicha tabla de envío. Al implementar una tabla de envío separada lógicamente para cada VPN, cada VPN se convierte en una red privada, no orientada a conexión construida sobre una infraestructura compartida.

IP limita el tamaño de una dirección a 32 bits en el encabezado del paquete. Las direcciones VPN IP añaden 64 bits al inicio del encabezado, creando una dirección extendida que el envío clásico IP no puede despachar. MPLS resuelve este problema enviando el tráfico según las etiquetas, con lo que se puede usar MPLS para que las rutas VPN IP sean conmutadas por medio de las etiquetas. Los enrutadores PE realmente se fijan en

las etiquetas MPLS, no en el encabezado del paquete. Como las etiquetas sólo existen para destinos válidos, MPLS asegura que la entrega es confiable.

Cuando se proporciona un circuito virtual usando el modelo *overlay* de VPNs, la interfaz de egreso para un paquete de datos en particular es una función únicamente de la interfaz de ingreso; la dirección IP destino no determina la trayectoria del paquete a través de la nube MPLS, previniendo que se efectúe comunicación no autorizada.

En las VPNs MPLS, un paquete recibido en el *backbone* es asociado con una VPN estipulando que todos los paquetes recibidos por cierta interfaz pertenecen a la misma VPN. Entonces, su dirección IP se busca en la tabla de envío asociada a la VPN. Las rutas de dicha tabla son específicas para la VPN del paquete recibido. De esta forma, la interfaz de ingreso determina un conjunto de posibles interfaces de egreso y el destino IP del paquete se usa para elegir la interfaz correspondiente entre todas las del conjunto previamente determinado. Este punto ayuda a prevenir aún más la comunicación no autorizada.

V.4.4. Aislamiento de las VPNs

Para mantener un aislamiento adecuado de una VPN de las demás, es importante que los enrutadores P no acepten un paquete etiquetado de algún PE adyacente, a menos de que se cumplan las siguientes condiciones:

- La etiqueta al inicio del *stack* haya sido distribuida del enrutador P al enrutador PE
- El enrutador P pueda determinar el uso que la etiqueta le dará al paquete al salir del *backbone*, antes de que cualquier etiqueta más baja en el *stack* sea inspeccionada.

Estas restricciones son necesarias para prevenir que los paquetes que lleguen a una VPN a la que no pertenecen. Las tablas VRFs en los PEs son usadas únicamente para los paquetes que lleguen provenientes de

un CE directamente conectado a tal PE, pero no son usadas para los paquetes que lleguen provenientes de otros enrutadores que pertenezcan al *backbone* del proveedor de servicios.

V.5. Escalabilidad

Como ya se mencionó anteriormente, al utilizar MPLS se mantiene a los enrutadores P libres de cualquier información de enrutamiento de la VPN. Esto significa que los enrutadores PE son los únicos dispositivos que contienen esta información.

Los enrutadores PE tienen que mantener la información de enrutamiento VPN y los sitios que están directamente conectados, pero no deben tener todas las rutas de todas las VPNs del proveedor de servicios. Si el volumen de información de enrutamiento VPN dentro de un PE en particular es muy grande y supera la capacidad de ese enrutador PE, lo que se hace es añadir un nuevo enrutador PE y dividir la carga entre el viejo enrutador y el nuevo enrutador.

Para anular la situación en la que un *Route Reflector* en particular pudiera ser requerido para manejar la información de enrutamiento de todas las VPNs soportadas por el proveedor de servicios, se particionan los *Route Reflectors* entre las VPN soportadas por el proveedor. De este modo, un conjunto de *Route Reflectors* puede contener las primeras 100 VPNs, el segundo conjunto las siguientes 100 VPNs y así sucesivamente. Como resultado, si el volumen de la información de enrutamiento mantenida por un conjunto particular de *Route Reflectors* es muy alto, se puede adicionar un nuevo conjunto de *Route Reflectors* y mover algunas VPN del conjunto viejo al conjunto nuevo.

Los equipos disponibles a la venta tienen ciertas características que pueden ser superadas en capacidad como consecuencia del escalamiento del tamaño de una VPN que ofrece un proveedor de servicios, pero existe la

posibilidad de agregar nuevos equipos para expandir esta capacidad, lo que da un gran margen para que crezcan los servicios de la VPN.

V.6. Calidad de Servicio (QoS) y Clase de Servicio (CoS)

Una VPN tiene un conjunto de políticas administrativas que controlan tanto la conectividad como la QoS (*Quality of Service* o Calidad de Servicio) entre los sitios. El reto de QoS es desarrollar un conjunto de mecanismos que lo soporten con la flexibilidad para poseer un gran rango de clientes VPN. Por ello, un proveedor de servicios debe ser capaz de ofrecer múltiples Clases de Servicios para cada VPN, es decir, debe permitir diferentes aplicaciones (en una misma VPN) que reciban diferentes Clases de Servicios. Por ejemplo, el correo electrónico puede ser una Clase de Servicio, mientras que las aplicaciones sensibles a retardo y de gran demanda , como VoIP, pueden ser otra clase de servicio.

El QoS y la CoS permiten que el proveedor de servicios ofrezca niveles de servicio basados en paquetes IP diferenciados. QoS se refiere a la habilidad de una red para proveer mejores servicios a cierto tráfico seleccionado. En particular, las características de QoS proporcionan mejores servicios y más predecibles en la red a partir de lo siguiente:

- Ancho de banda dedicado
- Mejoramiento de características que producen pérdidas
- Prevención y administración de congestión en la red
- Establecimiento de prioridades de tráfico a través de la red.

CoS se refiere a los métodos que proporcionan servicio diferenciado, en los cuales la red entrega un tipo particular de servicio basado en la clase de servicio especificado en cada paquete. CoS provee diferentes categorías específicas de servicio.

Una clase de servicio que una aplicación puede tener en una VPN tal vez sea poco diferente a la Clase de Servicio que puede tenerse dentro de otra VPN. Esto es, que el conjunto de mecanismos de soporte en QoS debe permitir la decisión acerca de qué tipo de tráfico toma una clase de servicio en específico para cada VPN. No todas las VPN tienen que usar todas las clases de servicios que un proveedor de servicios VPN ofrece, por eso, el conjunto de mecanismos de soporte en QoS debe permitir la decisión del tipo de clase de servicio que se va a usar para una VPN.

Existen dos modelos empleados para describir el QoS en el contexto de las VPN:

- *Pipe* (que en un acercamiento al español sería tubería)
- *Hose* (regadera)

En el modelo "*Pipe*" un proveedor de servicios proporciona a un cliente VPN ciertas garantías de QoS para el tráfico desde un enrutador CE del cliente hacia otro, es decir, se establece una tubería o "*pipe*" conectada a los dos enrutadores y el tráfico que entra dentro de esta tubería tiene ciertas garantías QoS. Por ejemplo, estas garantías QoS de las que se habla es de dar un mínimo ancho de banda entre los dos sitios dentro de esta tubería.

El modelo "*pipe*" se puede refinar haciendo un solo subconjunto de todo el tráfico (únicamente para aplicaciones en específico) desde un CE hacia otro CE, ambos usando esta tubería o "pipe". La última decisión sobre qué tipo de tráfico puede ser usado por la tubería es local al enrutador PE al final de la misma. Este modelo es muy similar (pero no idéntico) al modelo QoS que los clientes VPN tienen hoy en día basado en Frame Relay o ATM. La diferencia esencial radica en que en Frame Relay o en ATM la conexión es bidireccional, mientras que en el modelo de tubería o "*pipe*" se ofrecen garantías a través de la implementación en una sola dirección. El hecho de que una tubería sea unidireccional permite una asimetría con respecto al patrón de tráfico, en donde la cantidad de tráfico de un sitio a otro puede ser diferente a la cantidad de tráfico en sentido contrario.

Para ilustrar este modelo, se tiene la figura V.6.1. En este ejemplo, un proveedor de servicio provee a la VPN A con una tubería o "*pipe*" que garantiza 7Mbps de ancho de banda para el tráfico del Sitio 3 al Sitio 1 (para ser más preciso, desde CEA3 a CEA1) y otra tubería que garantiza 10Mbps de ancho de banda para el tráfico desde el Sitio 3 al Sitio 2 (desde CEA3 a CEA2). La figura muestra que un enrutador CE puede tener más de una tubería o "pipe" que se originen en él, pues en este caso hay dos tuberías que se originan desde CEA3, pero también puede haber más de una tubería que termine en un sitio o cierto enrutador CE.

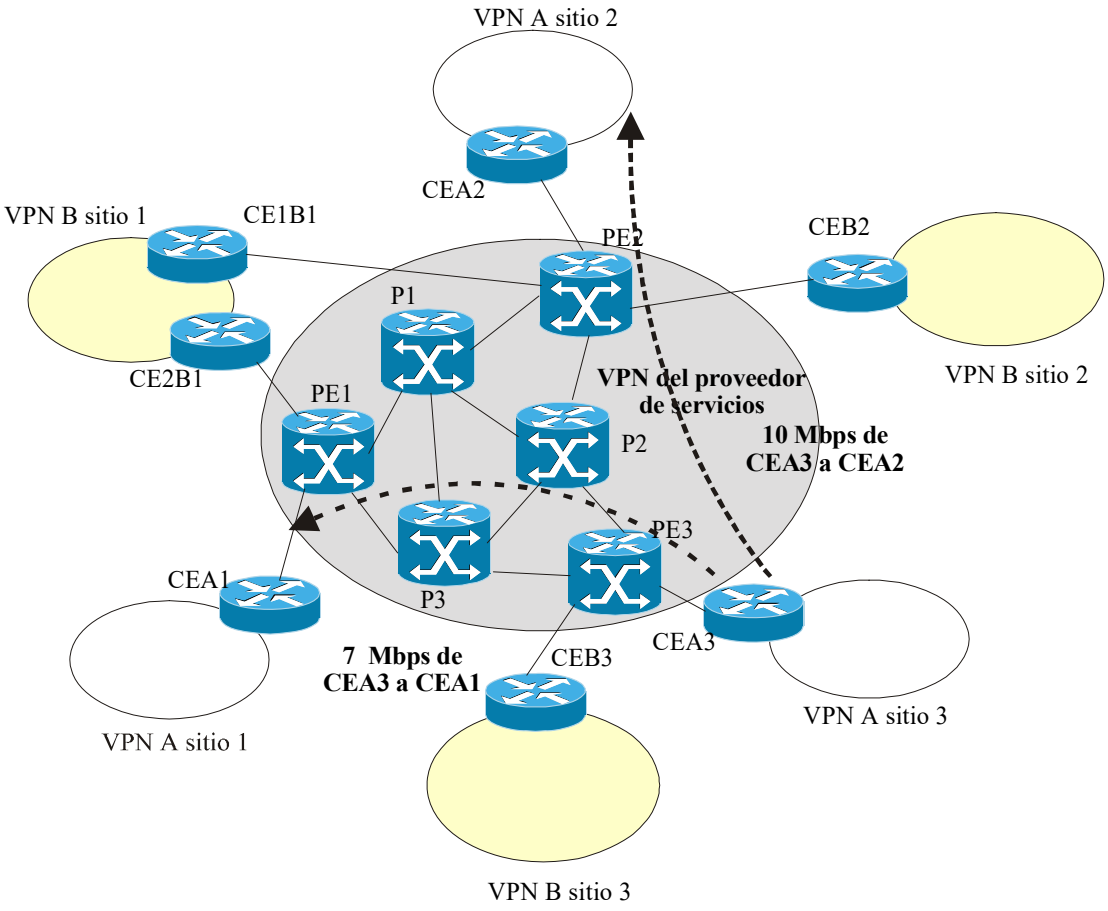


Figura V.6.1. Modelo pipe o tubería.

Una de las ventajas del modelo "*pipe*" es que es muy parecido a lo que ofrece Frame Relay o ATM y, lo más importante, es que es muy fácil de entender por los clientes. Sin embargo, existen inconvenientes, ya que se asume que el cliente conoce el tráfico que va de uno de sus sitios a otro. Desafortunadamente, esta

información en muchas ocasiones no existe o está fuera de actualización, por lo que es muy difícil de asignar cierto ancho de banda, pues se puede rebasar lo requerido o no llegar a lo que el cliente requiere.

Con el otro modelo, el modelo "*hose*" o regadera, no se necesitan conocer las necesidades de los clientes. En este modelo se usan dos parámetros:

- ICR (*Ingress Committed Rate* o Tasa de Ingreso Comprometido)
- ECR (*Egress Committed Rate* o Tasa de Egreso Comprometido)

El ICR es la cantidad de tráfico que un enrutador CE en particular puede mandar a otros CE, mientras que el ECR es la cantidad de tráfico que cierto enrutador CE puede recibir de otros CE. Dentro de un CE no existe el requerimiento de que su ICR debe ser igual a su ECR.

En la figura V.6.2. se muestra este modelo, en el cual el proveedor de servicios provee a la VPN B con ciertas garantías por encima de los 15Mbps para el tráfico que el Sitio 2 manda a otros sitios (ICR = 15Mbps). Este tráfico va al Sitio 1, o al Sitio 3, o es distribuido (arbitrariamente) entre el Sitio 1 y el Sitio 3. Por otro lado, el proveedor de servicios provee a la VPN B con ciertas garantías por encima de los 7Mbps para el tráfico que el Sitio 3 manda hacia otros sitios en esta VPN (ICR = 7Mbps). Este tráfico va al Sitio1, o al Sitio 2 o es distribuido (arbitrariamente) entre del Sitio 1 y el Sitio 2. Similarmente, el proveedor de servicios provee a la VPN B con ciertas garantías por encima de los 15Mbps para el tráfico que otros sitios mandan al Sitio 2 (ECR= 15Mbps). Este tráfico se origina desde el Sitio 1 o el Sitio 3 o es distribuido (arbitrariamnete) entre el Sitio 1 y el Sitio 3.

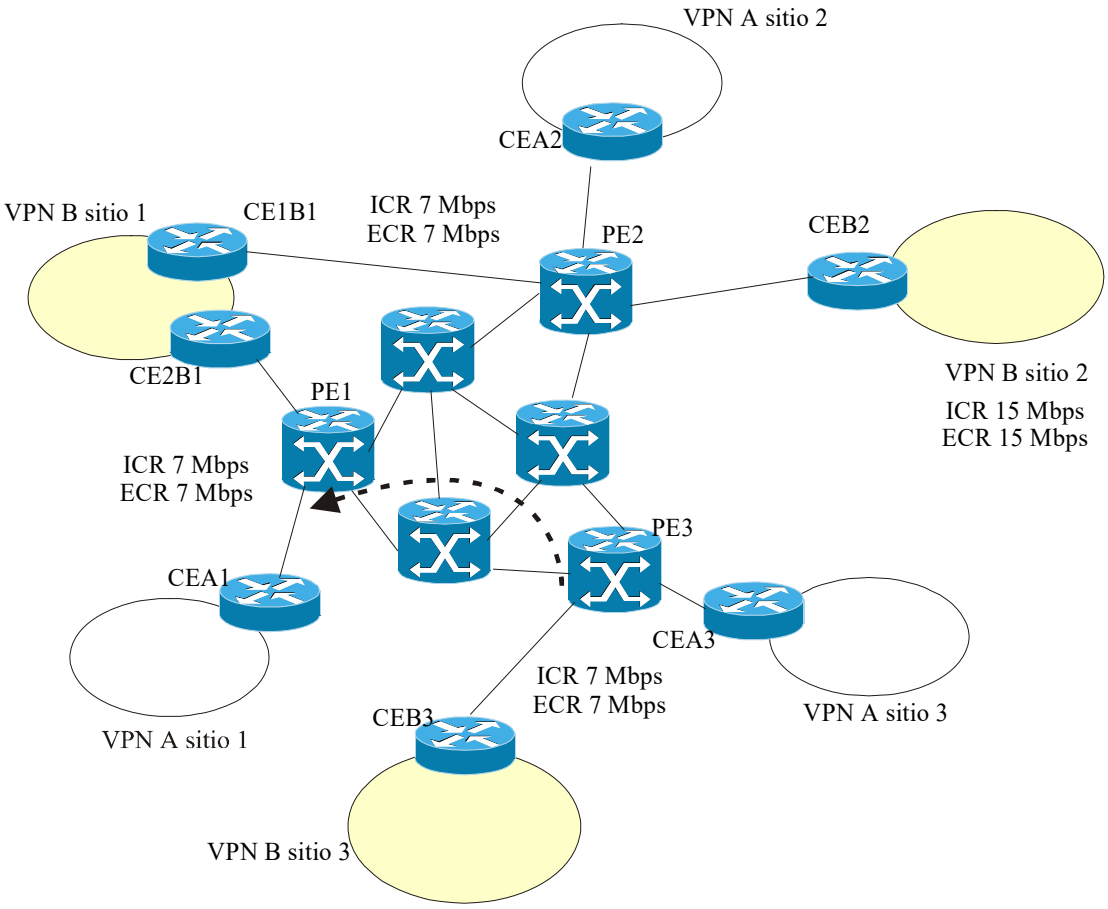


Figura V.6.2. Modelo hose o regadera.

Este modelo soporta múltiples clases de servicios diferenciados por sus características relativas. Por ejemplo, un servicio puede necesitar una pérdida de paquetes bajo otro servicio. Para servicios que requieren una garantía muy grande en ancho de banda, el modelo de tubería es la mejor opción.

Estos modelos no son mutuamente exclusivos, por lo que un proveedor de servicios debe permitir que un cliente elija entre ellos o que pueda tener una combinación de ambos, así como dar al cliente la opción de decidir qué servicio comprar y qué tráfico debe tener una clase de servicio.

Para que el modelo de tubería se pueda soportar, se usa una trayectoria LSP de ancho de banda garantizado, la cual se origina y se termina en los enrutadores PE y es usada para proporcionar una ancho de banda garantizado para todas las tuberías que van de un enrutador PE hacia otro. Para cada par dado de enrutadores PE, puede haber múltiples enrutadores CE conectados a estos enrutadores PE, los cuales tienen tuberías entre ellos y para eso, en vez de usar una trayectoria LSP de ancho de banda garantizado para cada tubería, se usa una sola trayectoria LSP de ancho de banda garantizado para todas estas.

Usar una sola trayectoria LSP de ancho de banda garantizado para llevar múltiples tuberías entre un par de enrutadores PE mejora las propiedades de escalamiento de la solución . Esto es porque el número de LSPs de ancho de banda garantizado que el proveedor de servicios tiene que establecer y mantener tiene su límite en el número de pares de enrutadores PE que posee este proveedor de servicios, en vez de tener límite por el número de tuberías que los clientes VPN puedan tener. Los procedimientos por los cuales un enrutador de ingreso PE determina qué tráfico recibe de una clase de servicios en particular, son locales a ese enrutador PE. Para desarrollar adecuadamente QoS, es necesario que la arquitectura sea *end-to-end* o extremo a extremo, ya que los mecanismos QoS deben estar implementados tanto en la frontera como en el núcleo de la nube MPLS.

Para los proveedores de servicios es deseable el QoS, ya que es una ayuda potencial para soportar muchos tipos de tráfico (datos, voz, video) sobre la misma infraestructura de la red. En un ambiente VPN MPLS, el proveedor de servicios debe considerar tanto enrutadores de paquetes, como enrutadores de celdas. En un ambiente de paquetes, CoS en MPLS es bastante confiable. Un enrutador PE simplemente copia la precedencia IP al campo CoS de MPLS. Con ello, es posible tener seis clases de servicio usando los tres bits del campo Tipo de Servicio de IP, pero dos de ellas están reservadas para el uso interno de la red. Las tecnologías que hacen encolamiento pueden usar esta señal para proporcionar el manejo apropiado a los paquetes.

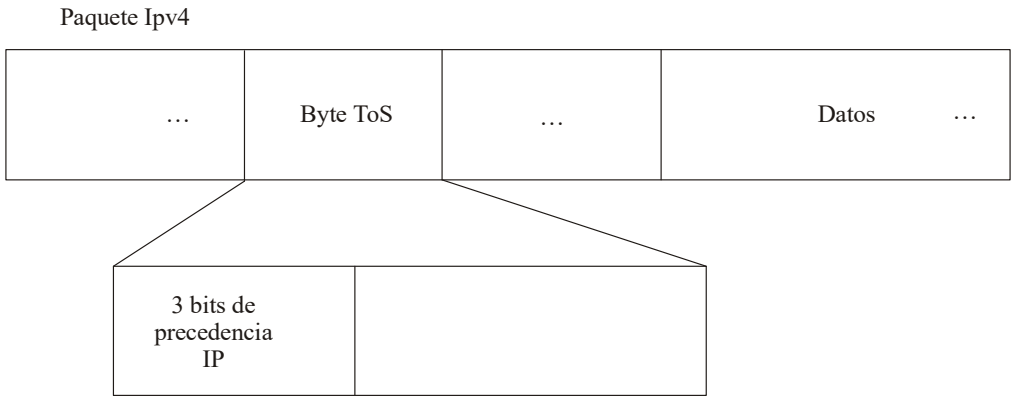


Figura V.6.3. ToS del encabezado IP

V.7. Recapitulación y Configuración de una VPN MPLS

V.7.1. Recapitulación

A lo largo de este capítulo se ha descrito el funcionamiento de una VPN MPLS por secciones, así que a continuación se hará una recapitulación general de la operación de una VPN MPLS para tener una visión global de ella:

MPLS tiene dos aspectos muy importantes que lo hacen una buena solución para los problemas de seguridad y de tráfico en la red. El primer aspecto importante es que es considerado un protocolo muy seguro, garantizando que una VPN MPLS sea tan confiable como cualquier VPN del modelo *overlay* o aun más, a pesar de lo que los clientes puedan pensar. Existe la falsa creencia de que el modelo *peer* no es tan seguro como el *overlay* pues en el primero se tiene que hacer *peering* o enrutamiento directo a uno de los enrutadores de la red del proveedor de seguros, mientras que en el *overlay* el transporte de información es transparente para el proveedor de servicios. La seguridad de una VPN MPLS radica en que en una nube MPLS, la conmutación se hace por medio de etiquetas o *labels*, no por direcciones IP.

20 bytes de encabezado IP	Etiqueta MPLS	Encabezado capa 2

Figura V.7.1.1. Formato del paquete transportado en una VPN MPLS.

Para que se pueda “jalar” la información, es necesario conocer la dirección IP origen y la destino, pero como MPLS maneja etiquetas, un intruso de la red sólo verá números cuyo significado es local al enrutador que lo originó (etiquetas locales). Por lo tanto, las etiquetas otorgan seguridad a la información pues lo que fluye en el núcleo de la nube MPLS son números sin sentido para el intruso.

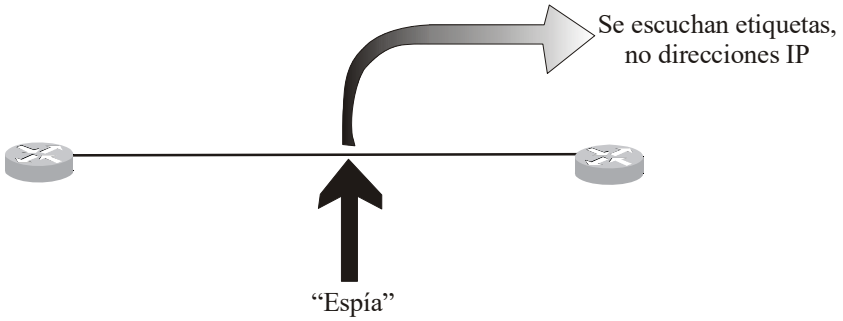


Figura V.7.1.2. Uno de los aspectos de seguridad en MPLS.

El segundo aspecto importante que se tiene en una VPN MPLS es la creación de pequeñas tablas con el fin de optimizar el enrutamiento y también ofrecer seguridad, pues al tener dichas tablas independientes unas de las otras, se asegura el aislamiento de la información que el cliente comparte con el proveedor (al momento del enrutamiento directo o *peering*).

Cuando se configura un equipo PE, se “levanta” un protocolo, que es el programa que genera una tabla global, la cual se usa para hacer el enrutamiento local en la red. Además, también se tiene un protocolo para que la nube MPLS hable con el cliente (por ejemplo, RIPv2 o IS-IS), el cual genera una tabla que comparten ambos (el proveedor y el cliente) en el PE.

Al crear VPNs con MPLS, lo que se hace es crear “espacios” de enrutamiento dentro de ese algoritmo, es decir, se crean pequeñas tablas en ese protocolo y se asignan a cada VPN. Estas pequeñas tablas se llaman VRFs. Con las VRFs ya no se comparte una tabla global entre el cliente y el proveedor en la cual se tienen todas las rutas, sino que se tienen espacios particulares ligados a una VPN. Las VRFs utilizan el mismo protocolo para crearse, pero son independientes unas de otras, lo cual ofrece otro punto de seguridad.

Con las VRFs se tienen tablas más pequeñas, provocando que la búsqueda de cierta ruta sea más rápido. Inclusive, con las VRFs es posible proporcionar el servicio de VPNs MPLS a dos o más clientes con el mismo esquema de direccionamiento, en donde un cliente tiene homologado tal esquema de direccionamiento y el (o los) otro(s) lo tiene(n) “prestado”. Esta es una gran ventaja, ya que el cliente espera de una VPN MPLS conectarse a la red y hacer uso de ella sin tener que configurar nada antes de ello. Es como el caso en el que al comprar un aparato eléctrico, el cliente espera poder conectarlo a la energía eléctrica y que funcione inmediatamente sin tener que modificarle algo previamente. De la misma manera, con MPLS se esperaría que ninguno de los dos clientes con el mismo esquema de direccionamiento tenga que reconfigurar las direcciones IP de su red.

Normalmente, es una regla básica de enrutamiento que una misma red no sea vista por dos puntos diferentes, lo cual no permite tener como clientes a dos redes con el mismo esquema de direccionamiento. Además, en el modelo *peer-to-peer*, el proveedor de servicios no le puede pedir al cliente que cambie algo de su red, pues en dicho modelo una de las premisas que se tiene es que el cliente renta una VPN ya construida (armada y configurada) y no tiene por qué involucrarse en su construcción. Entonces, con el enrutamiento convencional el proveedor no le podría dar servicio a uno de ese clientes, lo cual es negativo desde el punto de vista de los negocios. Sin embargo, con MPLS se supera este problema debido a que no se emplean direcciones IP, eliminando el conflicto de enrutar paquetes a una misma dirección IP (e inclusive con la misma etiqueta) si éstos van “marcados” para distinguir su destino.

En resumen, MPLS cambia el paradigma de enrutamiento desde cuatro puntos:

1. No se utilizan direcciones IP.
2. Se optimizan de los procesos de enrutamiento.
3. Se separan los procesos de enrutamiento de cada cliente.
4. El procesamiento es más rápido debido a que se tiene tablas de enrutamiento más pequeñas.

En MPLS se tiene un etiqueta asociada a cada dirección IP destino y a la asociación que forma ese par <Dirección IP destino, Etiqueta> se le llama *Forwarding Equivalence Class* o FEC. En el argot de MPLS no se hablan de direcciones IP destino, sino de FECs.

En una red MPLS se tiene tres componentes (clasificados según su arquitectura): los CE (equipos que pertenecen al cliente), los PE (equipos que interactúan con el cliente) y los P (cajas que sólo hablan etiquetas y de acuerdo a ellas hacen la conmutación; no tienen interacción con el usuario).

Virtualmente se tiene dos partes, una red IP y una red MPLS. La división de esas dos redes se da en los enrutadores PE, lugar donde se originan las VPN pues en estas cajas se tienen las cuatro tablas necesarias para lograr el transporte de paquetes en MPLS (tabla global, FIB, LIB, LFIB). Como se explicó en el capítulo III, cuando un paquete del cliente llega al enrutador PE, lo primero que se hace es buscar en la tabla de enrutamiento global los datos que le correspondan. Sin embargo, si anteriormente ya se habían recibido paquetes con el mismo destino y el mismo origen IP a través de la misma interfaz de ingreso, sus datos seguramente ya están almacenados en la tabla construida por CEF. La tabla derivada por CEF se conoce como FIB en el argot de MPLS. En la FIB se tienen únicamente los datos que son importantes para la conmutación en MPLS. Tanto la tabla global como la FIB pertenecen a IP.

La FIB sirve para crear otra tabla llamada LIB, en la cual se asignan las etiquetas previamente creadas por el PE, esto es, a cada destino IP se le asigna una etiqueta local y una etiqueta para el siguiente salto. Con esta tabla y con algunas de las entradas de la FIB se completa la tabla que finalmente será la que permita la conmutación en MPLS. Esta tabla se llama LFIB y contiene etiqueta local, etiqueta del siguiente salto y la

interfaz de salida correspondiente para cada paquete. Estas dos últimas tablas (LIB y LFIB) pertenecen a la tecnología MPLS.

Es importante mencionar nuevamente que en el Plano de Control se realiza la generación, asignación, transporte y ruteo de etiquetas, perteneciendo a este plano la tabla global. El plano de datos se encarga de la transmisión efectiva de etiquetas, por lo que las tablas FIB, LIB y LFIB pertenecen a este plano.

Para comprender cómo es que una VPN surge en un PE, se consideran los planos de la arquitectura MPLS –Plano de datos y Plano de control-. Ambos planos se encuentran presentes en los enrutadores PE.

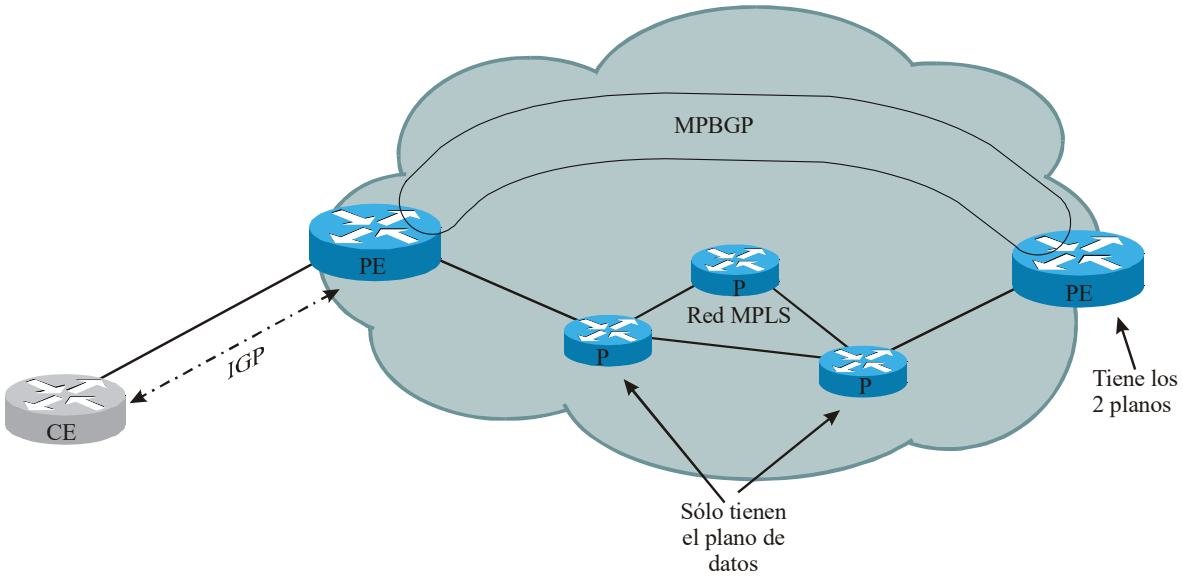


Figura V.7.1.3. Planos en los enrutadores de un proveedor.

Todo el proceso de IP (algoritmos del IGP y a la creación de las tablas) y el de MPLS (creación de VRFs, configuración e MPLS a nivel global) se asocian a las interfaces a las que se van a aplicar estos procesos.

Como se observa en la figura V.7.1.4., es posible pensar en un enrutador PE como un enrutador dividido en dos partes. La primera de ellas con una interfaz de ingreso por la que ingresan los paquetes IP y por la que el

PE está directamente conectado al cliente. En esta parte, el PE tiene la tarea de asignar etiquetas a los paquetes IP, por lo que el plano de control es el que está presente en esta parte del enrutador. La segunda parte del enrutador PE tiene una interfaz de egreso por la cual salen los paquetes etiquetados por MPLS. El PE está conectado al núcleo P por medio de esta interfaz. El plano de datos es el que gobierna en esta parte del enrutador, pues se realiza la transmisión efectiva de etiquetas. En el centro del enrutador PE se lleva a cabo la construcción de tablas necesarias para la conmutación en MPLS.

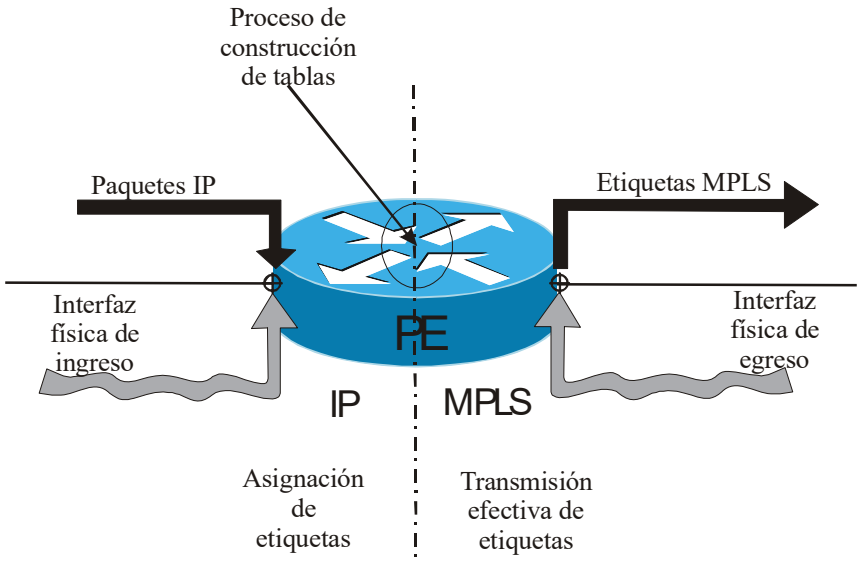


Figura V.7.1.4. Análisis de un enrutador PE.

Se puede tener una interfaz de ingreso para todo el proceso de MPLS. Si se desea añadir una interfaz a la red MPLS, se configura la interfaz para soportar MPLS.

Una VPN surge en el momento en el que a una dirección IP se le asigna una etiqueta local por medio de LDP (lo cual se refleja en la LIB), formando una FEC. Cuando se crea una VPN, se le pone un nombre, el cual es sensible, es decir, distingue mayúsculas y minúsculas. Además de la etiqueta o *label*, se le asigna un distintivo a la dirección de red del cliente para distinguirla en la arquitectura MPLS. En realidad se tiene dos distintivos:

- Primer etiquetado: *Route Distinguisher*, cuyo significado es local. Al nombre de la VPN se le pone una etiqueta para que signifique algo a las cajas. Esta etiqueta es el RD, el cual permite que dos redes sean distintas a pesar de tener el mismo esquema de direccionamiento.
- Etiquetado de plano mayor: *Route Target,* el cual tiene significado global. El RT es necesario en topologías complejas y/o en *multicast*. El RT es una etiqueta de orden mayor que agrupa a redes o subredes del mismo grupo *multicast*.

Entonces, una VPN tiene nombre, *Route Distinguisher* y *Route Target*. En el caso de redes planas (es decir, sin subredes) el *Route Disntinguisher* es igual al *Route Target*. Existen dos formatos para el RD de una VPN:

16:32

Indica que se tienen 16 bits al inicio para nombrar el sistema autónomo de BGP al que se pertenece y 32 bits para nombrar un dirección IP significativa para el proveedor de servicios relacionada a la VPN.

32:16

Indica que se tienen 32 bits para nombrar una dirección IP ligada al cliente y otros 16 bits para mencionar un número significativo para el proveedor de servicios.

Los formatos anteriores son recomendaciones por lo que es posible modificar la designación de los campos. El formato más empleado es el 16:32.

Como se mencionó anteriormente, dentro del protocolo IGP se crean VRFs para cada VPN. De algún modo se le tiene que indicar a los enrutadores que dentro del algoritmo IGP que va a ejecutar, existen estas VRFs. Al formarse la VRF se le indica que va a funcionar con formato VPN IPv4, es decir, con direcciones de 96 bits asociadas a las VRFs correspondientes.

El protocolo IGP del CE al PE puede o no ser el mismo IGP que se hable en el núcleo de la red MPLS. Las tablas de enrutamiento IGP del cliente deben ser transportadas hacia sus otros puntos y para ello se utiliza MPBGP. Sin embargo, este proceso no es transparente, pues se necesita redistribuir las tablas del cliente construidas por el IGP, al MPBGP. Si no se habilita MPBGP, las tablas IGP del cliente se mezclarían con las tablas IGP del núcleo al momento de que un sitio del cliente quiera dar a conocer dichas tablas a otro de sus

sitios, pues para ello sería necesario transmitirlas a través del núcleo MPLS. Lo que pasaría es que el algoritmo de enrutamiento del núcleo aprendería las tablas IGP del cliente y agregaría las rutas a su tabla. Por lo anterior, es esencial tener un protocolo que permita transportar la tabla de enrutamiento IGP del cliente a través del núcleo MPLS, pero que mantenga independientes la tabla del cliente y la tabla del núcleo como si el núcleo y el cliente fueran dos dominios distintos (tal como en BGP se consideran dos Sistemas Autónomos diferentes).

La extensión de BGP que permite la transmisión de la tabla IGP del cliente es *Multiprotocol BGP*. Su nombre se debe a que transporta un protocolo IGP dentro de BGP. No se manda la información del cliente, sino la información de control del otro protocolo. Un requerimiento de MPLS es que la nube tenga topología *full mesh* (igual que en BGP) para que el enrutamiento sea óptimo, pero el MPBGP sólo se configura entre todos los PEs y no se incluyen a los P, pues el intercambio de información se da exclusivamente entre PEs.

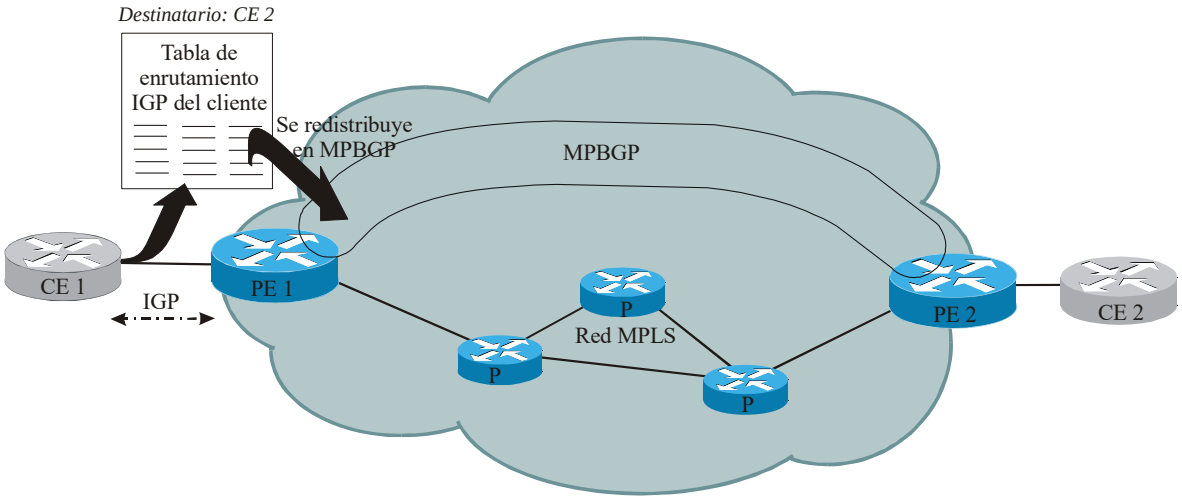


Figura V.7.1.5. Exportación a MPBGP de una tabla de enrutamiento IGP del cliente.

La redistribución de IGP a MPBGP es unidireccional, por lo cual se hace una redistribución para un sentido de la transmisión y otra redistribución para el otro sentido. Así, los dos extremos pueden ver las tablas. Si no

se hace la redistribución dos veces, en un extremo se vería la tabla de enrutamiento del otro CE y en el otro extremo, no.

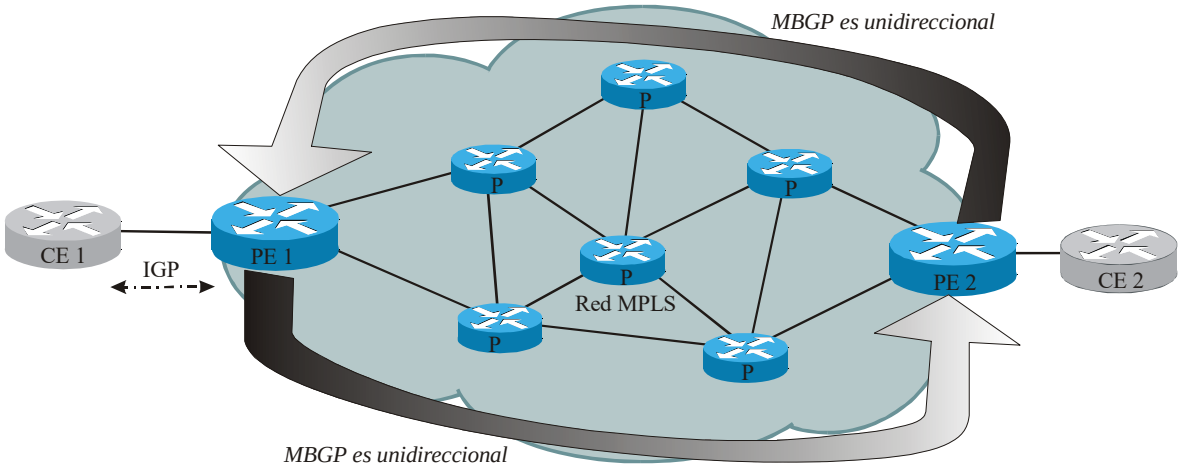


Figura V.7.1.6. Redistribución unidireccional de MPBGP.

Aún no están definidos los mecanismos necesarios para ofrecer Calidad de Servicio en el núcleo de la red MPLS, tal como en ATM o en Frame Relay. Lo que sí se puede hacer es un mapeo de los parámetros de QoS de otros protocolos hacia MPLS. Sin embargo, en México todavía no se llega a los niveles de congestión que requieran "echar mano" de los mecanismos de QoS.

Una de las posibilidades de QoS en MPLS es el "marcaje" de paquetes en el CE para establecer prioridades de tráfico. Para ello, se mapean los datos el campo *Type of Service* del encabezado IP a MPLS:

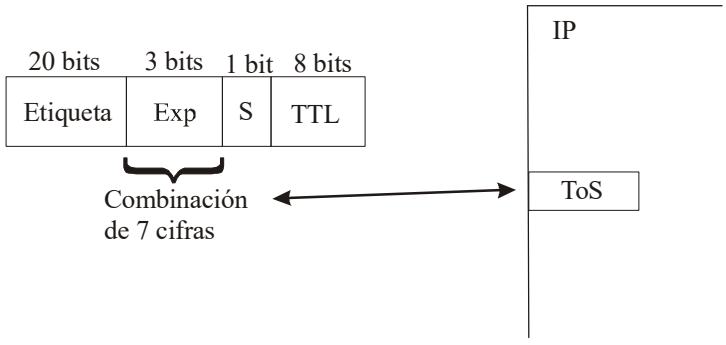


Figura V.7.1.7. Mapeo del ToS de IP a MPLS.

Los tres bits de precedencia en el campo ToS del encabezado IP se copian al campo Experimental del encabezado MPLS, también de tres bits. Estos tres bits ofrecen una combinación de siete cifras con las que se puede marcar a los paquetes según la criticidad del tráfico. Por ejemplo, al transportar voz sobre IP, los paquetes de esta aplicación se marcan con prioridad más crítica (la 5) para que los enrutadores que reciben dichos paquetes sepan que deben ser enviados inmediatamente, ya que la voz es sensible al retardo.

V.7.2. Configuración

MPLS es una tecnología que sigue en evolución. En la IETF existen dos *drafts* sobre el modelo de la creación (configuración) de VPNs. Uno de ellos está basado en el modelo *overlay* y el otro en el modelo *peer*. El *draft* Martini (nombrado así por el ingeniero italiano llamado Luca Martini que lo creó) es el correspondiente al modelo *peer*. Los pasos de la configuración de una VPN MPLS:

1. Configurar CEF (para crear FIBs) y MPLS.
2. Crear VRFs.

   Por medio de un comando se le asigna a cada VRF un nombre, un *Route Distinguisher* y un *Route Target*, tanto de importación como de exportación.
3. Levantar IGP entre los CEs y los PEs y en el núcleo P.
4. Habilitar MPBGP entre los PEs.
5. Definir las VRFs.
6. Redistribuir IGP en MPBGP.
7. Asociar a cada VRF una interfaz física de ingreso y prender MPLS en la interfaz de egreso y en todos los enrutadores P.

Referencias

Guichard, Jim y Pepelnjak, Ivan. MPLS and VPN Architectures.

Cisco Press, E.U., 2000.

Capítulo , pp .

(falta el capítulo y las páginas de referencia de este libro)

Davie, Bruce y Rekhter, Yakov. MPLS, Technology and Applications.

Morgan Kaufmann Publishers, E.U.

Capítulo 8, pp. 211-239.

MPLS's Newest Application. Webtutorials.

http://www.webtorials.com/main/resource/papers/BCR/paper23.htm

Multiprotocol BGP Support for CLNS. Cisco Systems.

http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080087d4f.html

Introduction to MPLS VPN Technology. Cisco VPN Solution Center Software.

http://www.cisco.com/en/US/products/sw/netmgtsw/ps2327/products_technical_reference_chapter09186a0080087c9c.html

Building MPLS-Based Virtual Private Networks and Services for Service Provider Core Networks. Cisco VPN Solution Center Software.

http://www.cisco.com/en/US/products/sw/netmgtsw/ps2327/products_technical_reference_chapter09186a0080087c9c.html

Capítulo VI s

pvce
afMPLS

VI.1. Criterios de selección de una VPN desde el punto de vista del cliente

A continuación se se tratará de desglosar desde el punto de vista de los clientes los criterios de selección principales para elegir el tipo de VPN que se puede usar. Los criterios de selección se definieron a partir de la información recabada en una serie de entrevistas realizadas a los principales Proveedores de Servicios y de equipo del país (Ver Anexo). Dichos criterios de selección son:

- Sencillez de implementación y de operación
- Cobertura
- Costo
- Servicios

La sencillez de implementación se refiere a la capacidad de un cliente para adquirir una tecnología. La capacidad del cliente se puede determinar por medio de una evaluación de su infraestructura (enrutadores, enlaces, conmutadores, etc.). de esta manera, cada cliente debe determinar la cantidad de hardware que va a emplear en la implementación de la tecnología nueva y el diferendo con su propia infraestructura, esto es, si tiene disponible equipo para la nueva implementación o si será necesario adquirirlo o modificarlo. Con base en esto, el cliente definirá la sencillez o complejidad de cada tecnología o servicio que desee adquirir.

En cuanto a la sencillez de operación, ésta se determina de acuerdo a los recursos humanos y de *software* necesarios para mantener el correcto funcionamiento de la VPN. Con respecto a este punto, la dimensión de la

VPN y de la empresa son un factor importante, ya que quizás mientras más grande sea la VPN, mayor cantidad de personas serán requeridas para supervisar la operación de la VPN y deberán desarrollarse herramientas de *software* para cumplir con las necesidades propias de la VPN (en el caso de que estas obligaciones recaigan en el cliente y no en el proveedor de servicios). La complejidad o sencillez de la operación la determinará cada cliente de acuerdo a su habilidad para adaptarse a las exigencias.

Adicionalmente, el cliente que desee implementar una VPN en su red corporativa debe asegurarse que la cobertura de la tecnología de la VPN llegue a los sitios en donde se requiere comunicación. Por ejemplo, si un cliente decide que una VPN con cierta tecnología es la solución más adecuada, la cobertura de esa tecnología tendrá que abarcar los puntos a interconectar. Por lo contrario, es mejor buscar otra opción.

Por otro lado, el cliente debe determinar el uso que va a dar a su red y con base en esto, elegir la tecnología que más se adapte a los requerimientos. Algunos clientes satisfacen sus necesidades con los servicios más básicos, mientras que otros prefieren pagar más con tal de obtener cierta calidad de servicio.

De todo lo anterior, también es necesario un análisis económico para conocer la factibilidad de implementación de la VPN. Esto significa que cada cliente debe evaluar los costos de adquisición de equipo nuevo, de desarrollo de herramientas de *software*, de sueldos mensuales de las personas encargadas del mantenimientos de la VPN y de los costos propios de la VPN. Estos son (según el plan tarifario de cada VPN): renta de enlaces, de puertos, cargos de largas distancias, etc.. En general, se debe hacer una evaluación de los costos generados por los criterios de selección y los costos que el Proveedor de Servicios absorbe.

Finalmente, para elegir la VPN el cliente debe hacer una valoración de los criterios y decidir qué características va a tener su VPN, según sus necesidades. El cliente debe decidir qué criterio es prioritario para él y con base en ello elegir el tipo de VPN que implementará.

VI.2. Estrategias de selección

Sencillez de implementación

Tanto en VPNs con líneas dedicadas, como de Frame Relay o MPLS, la implementación es muy parecida. Para cada sitio se necesita un enrutador (el uso de lo que hay después del enrutador dependerá de las necesidades propias del cliente). El cliente puede optar por un equipo que ya se tiene (si es el caso), rentarlo o comprarlo, tomando en cuenta las características de la red para seleccionar tanto caja como procesamiento, memoria e interfaces.

Enlaces dedicados:

Se requiere comprar, rentar o instalar una vía física de enrutador a enrutador para aquellos sitios que se quieran intercomunicar. Además, hay Proveedores de Servicios que no establecen el enlace como tal, sino que utilizan su propia red de datos para ofrecer este servicio. En este caso, se debe contemplar el enlace de cada uno de los sitios a esta red, así como su longitud y velocidad de transmisión.

Frame Relay y MPLS:

Se debe contemplar el enlace del enrutador al puerto del Proveedor de Servicios, su distancia y su velocidad de transmisión. Además, es importante considerar que el enrutador a emplear debe soportar el total de los circuitos virtuales que se necesitan según los requerimientos del cliente.

Sencillez de operación

Se divide en tres rubros: el monitoreo de los enlaces, supervisión del estado de los enrutadores y agregación de servicios nuevos. El cliente tiene la opción de monitorear sus enlaces o dejarle al Proveedor de Servicios esta tarea. Si la decisión del cliente es encargarse del monitoreo de sus enlaces, debe establecer un centro dedicado a ello con gente capacitada y con las herramientas adecuadas. Por otro lado, si se decide que el Proveedor de Servicios sea el encargado del monitoreo de sus enlaces, el cliente debe investigar las condiciones del proveedor y determinar si es conveniente para él o no.

En el caso de Frame Relay, el cliente debe tener conocimiento del enrutamiento de la red, ya que cualquier cambio que desee realizar en sus enrutadores debe reflejarse en los enrutadores del proveedor de servicios. Así, el cliente que cuente con Frame Relay necesita tener una persona capacitada para cubrir este aspecto. En cambio, con MPLS no es necesario tener un conocimiento profundo de la tecnología, ya que con este modelo, el proveedor de servicios vende la VPN de tal forma que el cliente se dedica exclusivamente a su red y las modificaciones al enrutamiento de su CE las realiza el proveedor.

Para el caso de los enrutadores, también se debe realizar un monitoreo del estado de éstos para evitar y detectar fallos. Si el cliente renta el enrutador al proveedor de servicios, podrá tener dos opciones: contar con alguien que los esté monitoreando y detectar las fallas para que el proveedor de servicios lo corrija, o para que el cliente lo haga,PSPS Si el enrutador es del cliente, él deberá tener personal que revise constantemente el estado de los enrutadores y establecer un plan de soporte técnico entre el proveedor de equipo y la empresa cliente.

En la agregación de un servicio nuevo, se debe contemplar que la red sea capaz de dar nuevos servicios. Para ello es importante definir quién agregaría los nuevos servicios, si el cliente o el Proveedor de Servicios. Por ejemplo, para Frame Relay, para cada nuevo sitio que se agregue a la red, se necesita añadir un circuito virtual hacia cada sitio ya existente. En cambio, para MPLS sólo será necesario darlo de alta en el PE correspondiente a su VPN.

Cobertura

Actualmente, Frame Relay tiene una cobertura mayor que MPLS. En ambos casos, se cubren las principales ciudades del país. Como cliente,, se debe hacer un análisis y establecer las localidades de las oficinas remotas para decidir cuál de las dos tecnologías abarca los puntos que se desean interconectar.

Ningún Proveedor de Servicio ofrece una cobertura que cubra el 100% del país. Hay localidades en las cuales no se registra actividad de las empresas y por lo tanto, no tiene infraestructura. Sin embargo, si alguna de las oficinas remotas del cliente no se encuentra dentro de la cobertura, el Proveedor de Servicios puede optar por comprarle la última milla (de la oficina remota a un nodo) a terceros. Generalmente, la cobertura de los proveedores abarca como mínimo las 33 ciudades principales del país. Los usuarios del servicio podrán conectarse a su red corporativa de datos desde cualquier ciudad de la República Mexicana marcando un número 800.

Costo

Es difícil determinar el costo de contratación de Frame Relay o de MPLS ya que para ello es necesario que asista un agente del Proveedor de Servicios a la empresa y evalúe las necesidades y los costos de la red corporativa. En el caso de Frame Relay, comúnmente se cobra por los puertos de acceso con cargos mensuales fijos según la capacidad contratada o por los circuitos virtuales permanentes con cargos mensuales (cargos fijos en función del ancho de banda requerido o cargos variables en función del tráfico transportado). En el caso de MPLS, depende mucho cómo se cobra ya que se considera el lugar (si hay cobertura o no), del número de puntas y del ancho de banda. No se puede especificar en sí una sola modalidad de cobro pues, además, se necesita analizar las características del enlace del cliente hacia el punto de presencia del proveedor de servicios.

Tanto para Frame Relay como para VPN MPLS, el enlace físico del cliente a la red del Proveedor de Servicios puede variar mucho tanto en costo como en características. Este puede ser montado sobre cobre, coaxial o fibra óptica. La distancia también influye en el cobro. Los costos de instalación pueden ser nulos si el contrato es a varios años.

Servicios

La característica más importante de VPN MPLS es el servicio en IP. Quizá su ventaja más importante sea que ofrece una Calidad de Servicio que Frame Relay no entrega, lo cual permite al cliente tener servicios de voz, datos y video ya que QoS diferencia el tráfico y asigna prioridades, es decir,Si el tráfico es sensible (por ejemplo, voz y video), es tratado inmediatamente. Además, para VPN MPLS el cliente no necesita tener conocimiento de la configuración de la tecnología, sólo se dedica al mantenimiento de su red. Por su parte, Frame Relay brinda la posibilidad de trabajar con otros protocolos diferentes a IP.

- PSen una varios (Ver Anexo).Se investigó Estas se describen a continuación.Relay.Lpor ser, del,PS,,PS, etc.,n,,ndoEn consecuencia,PSsfuturo,rámás baratos ,Por esta razón, líen acuerdo,é,de las empresas.,a.eldarles. Esto tiene la finalidad dede los Proveedores de Servicios P,dirigir esta.E,onsuficientes ,,s,.Estolos existentesse tuvo queaplicar.Ete

proceso,óaplicaciones,ieron.E,.T,Con en,.Esto significa complicado.Por ello,nsolucionarlos.é,; estoresueltoOrepresentabaal rseen una red real varioslos tiemposa,.E,, ,investigación,.Lnstalarobservaronsimultáneamente suieactualmente investigación,permitir í.Lo anterior tiene como objetivo ,, seel,.Tantas rutas nlos enrutadores,m ,enofreceí,()e,consehaciade ,,.Por otra parte,con Relay

la relación ,ífue que. , Relay, por lo que se recuperará la inversión,,SS seua,de las VPNs MPLS importanteslas consultadas:,n lass ,planeans aááde administrarPcada vez e

En las conclusiones se realizó un análisis más detallado de los requerimientos, impacto y perspectivas de las VPNs MPLS.

Conclusiones

Actualmente, en el mercado existen distintos tipos de Redes Privadas Virtuales, tales como VPNs con IPsec, con Frame Relay, con MPLS, con X.25 (ya en desuso) o incluso todavía con enlaces dedicados, aunque sin duda Frame Relay todavía abarca la mayor parte del mercado dentro de México. Sin embargo, las necesidades recientes de transporte de voz y de aplicaciones multimedia utilizan IP, además de que las aplicaciones cada vez son más pesadas y el Internet a veces no permite su correcta ejecución por requerir mayor ancho de banda y garantía, es decir que para poder utilizar este tipo de aplicaciones se necesita de la Calidad de Servicio.

El control de la red y la oferta de calidad de servicio se han vuelto un requerimiento importante, así que los clientes empezaron a preferir las VPNs IP (debido a su sencillez de manejo), por lo que los proveedores de servicios analizaron cuál sería la mejor opción de VPN IP y determinaron que MPLS es lo que estaban buscando, pues esta tecnología facilita soluciones a las solicitudes de los clientes, siendo posible utilizar el equipo del cliente (en caso de que ya posea infraestructura) y un *backbone* público para crear accesos privados. Si a eso se agrega que MPLS es muy escalable y que permite separar el tráfico de áreas distintas de la empresa que realicen diferentes funciones, resulta ser una opción prometedora.

En una VPN se tiene comunicación con todos los que forman esa red. Por ejemplo, en una red Frame Relay, si se tienen tres entidades para intercomunicarlas se debe establecer un PVC entre cada una de ellas, esto no es problema cuando la red es de tamaño mediano pero en el momento en que empieza a crecer el trabajo de dar de alta comunicación entre sitios con Frame Relay se complica cada vez más. En cambio, en una VPN MPLS, por su naturaleza, se tiene conectividad con todos los integrantes de la VPN. Aunado a este punto, se cuenta con una calidad de servicio en voz, datos, video, Internet, etc., característica que le conviene a ciertas empresas. Supóngase el caso de un banco que necesita compartir su base de datos entre varias sucursales y además usar el servicio de voz entre ellas. Encuentra que MPLS es una buena solución ya que permite definir

que el tráfico sea constante (para servicios de voz y video) o irregular (Internet), además de ser una red muy robusta, pues se tiene mayor disponibilidad que con una red Frame Relay.

Seguramente en la mayoría de los proveedores de servicios se decidió implementar VPNs MPLS para buscar más mercado, pues actualmente el mercado tiende a MPLS debido a la reciente exigencia de otro tipo de aplicaciones, otro tipo de prioridad y otro tipo de manejo de los datos en la red pública que Frame Relay no puede ofrecer, además de que hay posibilidades de poder trasmitir varios protocolos dentro de MPLS por lo que se tendría un muy gran beneficio.

Una vez que se ha decidido que es competitivo para un proveedor de servicios el implementar VPNs con MPLS, se debe cubrir con todos lo requerimientos de su implementación, y para esto hay que hacer una selección de un proveedor de equipos después de un análisis de las características de los equipos a utilizar así como su rendimiento. Al adquirir la tecnología, se capacita al personal adecuado para que pueda manejarla, además de que el que venda el equipo debe de dar distintas soluciones como capacitación, soporte técnico y resolución de conflictos que pudieran llegar a surgir. El conocimiento de la tecnología permitirá gestionar la red, ya sea creando sistemas o herramientas que permitan monitorear los equipos, detectar las fallas, facturar los servicios y, en general, será posible administrar la red con la gente capacitada.

Para este tipo de VPNs es necesario adquirir equipo que reconozca la funcionalidad de MPLS. Con la infraestructura de una red Frame Relay no es posible montar completamente una red MPLS pues Frame Relay maneja conmutadores de capa 2 y MPLS tiene funcionalidades que son también de capa 3. Cuando ya se haya instalado toda la red MPLS es importante definir la prioridad del tráfico o Calidad de Servicio (QoS).

Comúnmente, la implementación de una VPN está definida por los requerimientos que dependan de la aplicación y de las necesidades del cliente. Para ello es importante determinar si el proveedor de servicios puede soportar con sus recursos actuales las necesidades del cliente. Por ejemplo, tal vez la versión del sistema operativo del *software* de los enrutadores no soporte una aplicación que requiera el cliente. Si la necesidad del cliente es muy grande y el proveedor no cuenta con los recursos de red, tal vez se agregue

equipo o se evalúe si con el equipo del cliente se puede hacer algo, o si se tiene que contemplar equipos de otras marcas.

Desde hace 3 ó 4 años se realizan pruebas para MPLS y su introducción como servicio en México fue hace dos años aproximadamente. Sin embargo, cabe mencionar que el proceso de implementación fue más largo de lo que se esperaba debido a errores de la propia tecnología pues, en un principio, los equipos tenían muchos errores en lo que respecta a los sistemas operativos ya que se tenían "bugs" que fueron resueltos por medio de una comunicación permanente del proveedor de equipo y proveedore de servicios . El diseño de una red MPLS y su implementación puede ir de los 6 meses hasta los casi dos años, lo cual depende de la empresa.

Al implementar una red MPLS no sólo es necesario capacitar a la gente adecuada, sino que también es preciso reestructurar la organización de ella. Las áreas que debería tener una empresa que sea proveedor del servicio de VPNs MPLS se concentran en tres grandes rubros. El primer rubro debe definir el producto o servicio: su diseño, requerimientos, plan de equipo, configuraciones y demás variables importantes. El segundo rubro es referente a la construcción: compra de equipo, asignación de espacio, montaje y energización del mismo, etc. Por último, el rubro de administración: configuración de equipo, mantenimiento, etc. Las áreas que suelen ocupar estas empresas para una VPN MPLS son el área de operaciones (a cargo del monitoreo), área de soporte (encargado del mantenimiento del equipo), área de aprovisionamiento (para agregar clientes), mercadotecnia (ventas) y área de ingeniería (con funciones de diseño y análisis), aunque su organización puede cambiar de una empresa a otra. Antes, bastaba con una sola persona para hacerse cargo de una VPN, sin embargo, eso ya no es factible por la cantidad de funciones y la complejidad de las mismas, para lo cual las empresas han estructurado en varias áreas todas las actividades relacionadas.

Por otro lado, la relación costo-beneficio de una VPN MPLS ha resultado satisfactoria en todas las empresas que la manejan y se espera que a la larga de más ganancias con la infraestructura ya instalada. Es difícil definir exactamente esta relación ya que no solo se considera el costo del equipo, sino también el del soporte, mantenimiento, aprovisionamiento, monitoreo, mercadotecnia y todas las actividades que realiza la empresa para poder ofrecer el servicio . Por ejemplo, se puede usar equipo de diferentes proveedores y tal vez un

equipo necesite más del soporte de su proveedor que otro. Por ello, es difícil sopesar el monto exacto del costo. La cuestión del soporte ha sido crítica. Quizás un proveedor ofrezca un buen soporte, pero su equipo no ofrece tantas funcionalidades como otro cuyo soporte no es bueno. A veces, se tiene que recurrir a intermediarios.

Cuando se solicita una red (o un cambio en ella) se busca la solución más económica y que cubra los requerimientos mínimos. Muchas veces, esto se hace con lo que ya se tiene (infraestructura y servicios), por lo que los costos se consideran un gasto marginal. Cuando se defina individualmente el costo de una red se podrá determinar el monto exacto de la relación costo-beneficio de la misma. El beneficio monetario se define comparando los gastos con el precio que se le da al cliente por puerto, que en algunas ocasiones es el mismo que en Internet.

A pesar de que económicamente sea más redituable una VPN MPLS, tal vez sea más cara su gestión que una red con Frame Relay. Las necesidades del cliente se pueden adaptar a la mejor solución para que el proveedor de servicios tenga los costos más bajos. En Frame Relay se pagan los enlaces locales, no las largas distancias. Si se implementa una aplicación con MPLS y no lo requería, puede resultar muy caro y quizás con Frame Relay hubiera sido más barato. Todo depende del tipo de aplicación y de cómo se implemente la tecnología.

En el mercado de las VPNs, Frame Relay abarca una mayor proporción que MPLS. Es una tecnología madura, lleva ocho años en el mercado y durante ese tiempo fue la mejor opción para el transporte de tráfico en redes públicas punto a punto. Hasta la fecha, la tasa de crecimiento de Frame Relay se ha mantenido constante y tal vez los próximos años comience a decaer pues las necesidades de los clientes se están enfocando hacia MPLS. Se espera que en un futuro MPLS alcance la cantidad de clientes que tiene Frame Relay y que incluso lo supere, pero este lapso será largo.

Con respecto a los obstáculos más comunes que se presentan en la implementación de una VPN MPLS se encuentra el desconocimiento de la tecnología y por ende, la incapacidad de trasladar correctamente la teoría a la práctica. El mismo desconocimiento puede repercutir en una falta de estrategia, convirtiéndose en un

problema mayor, por ejemplo al integrar todos los elementos de la red (*software*, equipos, personal capacitado) para hacerla funcionar. Otro obstáculo que presentan las VPNs MPLS es la dificultad para convencer al cliente de que MPLS es la opción ideal para su VPN (obviamente basado en un análisis previo). En este punto, la mercadotecnia juega un papel muy importante, pues de ella dependen las ventas de esta tecnología.

Cuando se introdujo MPLS en México, la tecnología no estaba lista, principalmente los sistemas de gestión de la red. No existía un sistema que permitiera realizar la facturación de los servicios y que a la vez fuera compatible con el sistema de facturación del *carrier*. En el 2000, la tecnología era nueva y el equipo que se adquirió para la red MPLS también era nuevo. Al inicio de la implementación de una red MPLS se tuvieron muchas caídas debidas a fallas en el *software* (*bugs*) y a muchos problemas políticos que el proveedor no soportaba adecuadamente. Se tuvieron que buscar alternativas a esos problemas.

El soporte que da el proveedor de equipo es primordial para una implementación exitosa de VPNs. Si algo se sale de control, el proveedor debe ayudar a solucionarlo. Si surge un problema en una red que está en construcción, no sólo causa malestar al usuario final, sino también a todos aquellos por donde está transitando. La topología jerárquica ayuda a solucionar problemas rápidamente pues es fácil hacer diseños y aislar problemas.

Para la implementación es necesario primero un diseño en papel de los servicios que se pretendían resolver y a partir de ello se trabaja en una maqueta (a escala reducida) para hacer pruebas de laboratorio. Se carga en los equipos aplicaciones reales, funcionalidades sumamente críticas o funcionalidades que interferirán entre ellas con el fin de conocer el comportamiento del equipo, utilizando analizadores de protocolo y equipo generador de tráfico para simular lo mejor posible un ambiente real en el que se vislumbren los detalles a corregir y los imprevistos. Cada versión del sistema operativo tiene más funcionalidades que la versión anterior, por eso se tienen que hacer pruebas de laboratorio para saber si es cierto que se realizan las funciones que dice tener. Este proceso se comparte con el proveedor de equipo para que dé una solución adecuada a los *bugs*. Pero cabe hacer la aclaración de que aún habiendo trabajado con maquetas, en el momento en el que se

instala ya a la escala real surgen otros problemas, es decir, es de mucha ayuda utilizar maquetas montadas sobre un laboratorio pero no garantiza el correcto funcionamiento en su instalación real.

Existen implementaciones que afectan la seguridad porque rompen el esquema de la VPN, la cual tiene que estar aislada de la red normal. A veces se desea optimizar equipo y emplear uno mismo para realizar varias funciones, lo cual no es correcto desde el punto de vista de la seguridad (por ejemplo, equipos compartidos con acceso a la red normal y a la VPN, lo cual no es recomendable). Se considera aceptable agregar un equipo nuevo al que se le cargan las funciones que en otros equipos provocan conflictos y, de esta manera, se mantiene aislada la VPN. Sin embargo, se necesita cierto capital para adquirir equipo y en época de crisis económica, generalmente el equipo "nuevo" es reasignado a la VPN desde otra área de la empresa para ahorrar.

Es de gran importancia tener un constante monitoreo dentro de la red para encontrar problemas tanto en los equipos como en los enlaces. Los proveedores han empezado a sacar al mercado herramientas que son sistemas de ayuda en la gestión de la red, en la administración de procesos y en la facturación de los servicios, además de sistemas de simulación de la VPN y de monitoreo de enrutadores. Como MPLS tiene varios niveles de abstracción, todos deben ser cubiertos para detectar las fallas de los enrutadores en los diferentes niveles. En general, estos sistemas han ayudado principalmente en la administración de la red. No existe una herramienta propia para VPNs, pero sí para facilitar el aprovisionamiento de una red cualquiera. Por su parte, uno de los principales proveedores de servicios de México ha hecho un desarrollo propio para formar un mapa de la red y así, facilitar la detección de fallas.

Otro de los problemas que se ha presentado con los clientes de las VPN MPLS es el servicio de cifrado de datos. Aunque en MPLS no es necesario porque la seguridad que ofrece es elevada, si un cliente considera que una aplicación es muy importante y que debe ser cifrada, él tendrá que implementar un software que realice esta tarea. Se ha contemplado la posibilidad de "unir" IPsec y MPLS, aunque esto sólo es un plan a futuro. IPsec se implementaría entre los puntos de interconexión que la VPN pudiera tener con el Internet, con el único fin de dar soporte a las aplicaciones que necesiten confidencialidad en el acceso a Internet. Esto no

será una labor sencilla ya que IPsec es un estándar poco conocido hasta la fecha, no existe uniformidad en los clientes en cuanto a la definición de sus interconexiones y porque, por algún motivo, algunos clientes no cumplen con el estándar. Además, hasta la fecha no se ha analizado cómo se "unirían" IPsec y MPLS.

Este deseo de "unir" IPsec y MPLS responde al deseo por parte de los proveedores de servicios de estar a la vanguardia con los clientes, teniendo una red multiservicios, completamente IP y que cumpla con las necesidades del cliente. En un futuro se espera que VPN MPLS sea la VPN de mayor uso a nivel mundial ya que tiene la premisa "construye una vez, vende muchas veces", facilitando la tarea del proveedor.

Como ya se mencionó, en la introducción de MPLS al mercado los sistemas operativos no soportaban todas las funcionalidades, pero poco a poco se fueron incluyendo en las distintas versiones del sistema operativo hasta que todas las funcionalidades se unificaron en un solo sistema operativo. Esto ha evolucionado hasta el punto que surgió una versión que soporta el transporte de cualquier tecnología de capa 2 sobre MPLS (por ejemplo, Frame Relay sobre MPLS, AAL2 sobre MPLS, Ethernet sobre MPLS). Lo más común es que los paquetes IP se monten sobre MPLS, pero si un cliente tiene una interfaz diferente (como Frame Relay, ATM o Ethernet), no se puede transportar la información. Por esto , surgió AToM (*Any Transport over MPLS*). Independientemente del protocolo de capa 2 que se tenga, se puede montar de manera transparente sobre MPLS. Así, con AToM se pueden unificar las redes que antes eran diferentes y que el cliente consideraba separadas. Esto representa una ventaja económica ya que el operador sólo necesita una caja para transportar paquetes de varios protocolos, en lugar de tener varias cajas. Es más barato y por lo tanto, mejor desde el punto de vista de los negocios. Actualmente son pocos los proveedores que ofrecen esta ventaja.

El modelo de negocios obedece a poder ofrecer más servicios a menor costo. Se impone la tecnología que puede dar más a menor costo. Por eso, aunque IP no es la tecnología más óptima para dar calidad de servicio, domina el mercado porque es posible montar sobre él varios servicios con costos bajos. La tendencia de las tecnologías se mueve según el modelo de negocio. Por esta razón, la tendencia será alejarse de Frame Relay y hablar MPLS de forma nativa, quitando un *overhead* en la red.

Las VPNs se crearon para uso interno de la empresa y su crecimiento depende de la empresa. Una VPN de cierta compañía puede estar dedicada a capacitación y por ello, sería posible conectarla a la red de Internet 2. Si el tráfico de capacitación no estuviera separado del tráfico de las demás áreas de la empresa, no podría conectarse a Internet 2. En este aspecto, la Universidad Nacional Autónoma de México, quien es el principal actor de Internet 2 en el país, ha realizado pruebas con MPLS para conocer su funcionamiento y la posibilidad de implementarlo en el *backbone* de esta red en el caso de que sea necesario. En estos momentos no se tiene un flujo de datos tal que requiera la solución de MPLS; de hecho, no existen todavía problemas de tráfico en esta red que ameriten pensar en otras alternativas o soluciones.

En general, las telecomunicaciones tuvieron un *boom* a partir de 1991 hasta 1995 ó 1996. Desde entonces, algunas tecnologías se estabilizaron y otras fueron decayendo. Actualmente, el desarrollo de las redes de telecomunicación ha sido muy bajo. Para que una tecnología se desarrolle, es necesaria una demanda de servicio y, como tal, no la ha habido para VPNs. Existen muchas empresas que están requiriendo el servicio de transporte de información entre varias ciudades a través de VPNs. Sin embargo, el desarrollo de la tecnología no ha sido tan rápido como hubiera sucedido en años anteriores.

Si se tuviera la misma demanda de creación de VPNs como en años anteriores, la misma demanda de servicios hubiera obligado a mejorar la tecnología a algunos puntos de Calidad de Servicio y de manejo de rutas dentro de la VPN superiores a los actuales o inclusive a desarrollar servicios que aún no imaginamos porque no han sido necesitados por alguien. Cuando surge la necesidad, surge el desarrollo y, en estos momentos, la necesidad es muy pequeña o al menos no se ha podido satisfacer como se debiera debido al aspecto económico.

Actualmente, las redes basadas en VPNs en México están montadas sobre Frame Relay y sobre X.25 (estas últimas ya casi extintas). En México no existen VPNs montadas sobre ATM pues en nuestro país no se desarrolló (en Estados Unidos sí). Hasta hace dos años se desarrollaron los servicios de VPNs montadas sobre IP basadas en MPLS. El problema aquí es que hay poca demanda de crear VPNs y si surgen, seguramente serán sobre MPLS, pero las compañías que ya tiene montadas sus VPNs montadas sobre Frame Relay (o en el

caso e Estados Unidos, sobre ATM) ven muy costoso el migrar a VPNs MPLS, ya que implica mucho dinero llevar una VPN de una tecnología a otra, además de que conlleva problemas técnicos también. Por eso, aunque una VPN MPLS sea la mejor opción de para algunas empresas, no es ni siquiera considerada porque probablemente no tengan los recursos para hacerlo. El análisis que se haga, Seguramente resultará en que no es factible económicamente, además de que inicialmente se presentan más problemas técnicos que beneficios en la migración como tal, no en la implementación.

Por las razones anteriores, las VPNs MPLS apenas están captando la demanda de los clientes de VPNs. Si la situación de las empresas de telecomunicaciones mejora, se podrá esperar mayor demanda de servicios y por lo tanto, de creación de VPNs MPLS.

Anexo

A continuación se muestra el cuestionario que se presentó a cada uno de los entrevistados. Según la posición de cada persona (si era cliente, proveedor de equipo o de servicio), las preguntas se modificaban.

Estas son las preguntas que se le harán a las personas que decidan aportar información a la tesis "Construcción de VPNs usando MPLS". La finalidad que tienen estas preguntas es obtener la información necesaria para comparar los distintos tipos de VPNs (con Frame Relay, con ATM, con MPLS, VPDNs, etc.) y con ello concluir sobre el futuro que pueden tener las VPNs con MPLS en nuestro país, considerando los aspectos de la construcción de una VPN que generan costos, los beneficios que se logran, qué empresa es la principal proveedora de equipo para ello, etc.

Cabe señalar que si el entrevistado considera que cierta información no puede ser proporcionada por ser confidencial para la empresa, será respetado.

1. ¿Qué tipos de VPNs maneja la empresa?
2. ¿Cuáles fueron las causas por las que se crearon las VPNs?
3. ¿Qué se requiere para implementar una VPN?
4. ¿Qué obstáculos se presentaron en la implementación de una VPN?
5. ¿Cómo se resolvieron?
6. ¿Cuánto duró el proceso de implementación?
7. ¿Cuál es la cobertura de las VPNs?
8. ¿Se han obtenido otros beneficios además de haberse satisfecho las necesidades por las que se implementaron las VPNs?
9. ¿La relación costo-beneficio ha sido satisfactoria?
10. Si es el caso, ¿con qué tipo de VPN de las que maneja la empresa se abarca mayor mercado?
11. ¿Han surgido herramientas que faciliten la implementación de las VPNs?
12. ¿Cuáles son los principales proveedores de equipo?
13. ¿Cuántas áreas de la empresa se necesitan para la implementación, desarrollo y mantenimiento de las VPNs?
14. ¿Se maneja un gran número de clientes? ¿cuántos?
15. Comentarios generales del backbone (topología, protocolos, ancho de banda, etc.)
16. ¿Cuáles son las expectativas tanto comerciales como técnicas de las VPNs?