3
Diseño de Redes

Antes de adquirir equipamiento o decidirse por una plataforma de soporte físico, se debe tener una clara idea de la naturaleza de sus problemas de comunicación. En realidad, si usted está leyendo este libro es porque necesita conectar sus redes de computadoras para compartir recursos y en última instancia acceder a Internet. El diseño de red que elija para implementarlo debe concordar con los problemas de comunicaciones que está tratando de resolver. ¿Necesita conectar un lugar remoto a una conexión de Internet en el centro de su campus? ¿Es probable que su red crezca para incluir varios lugares alejados? ¿La mayoría de los componentes de su red van a estar instalados en locaciones fijas, o se va a expandir para incluir cientos de computadoras portátiles itinerantes y otros dispositivos?

Cuando resolvemos un problema complejo, a menudo es útil hacer un dibujo de sus recursos y problemas. En este capítulo, veremos cómo otras personas han construido redes inalámbricas para resolver sus problemas de comunicación, incluyendo diagramas de la estructura esencial de la red. Vamos a cubrir los conceptos que definen TCP/IP, el principal lenguaje de programación hablado actualmente en Internet. Mostraremos varios métodos sencillos para hacer que la información fluya eficientemente por su red y por la del resto del mundo.

Diseñando la red física

Puede parecer raro que hablemos de la red “física” cuando construimos redes inalámbricas. Después de todo ¿dónde está la parte física de la red? En estas redes, el medio físico que utilizamos para la comunicación es obviamente la energía electromagnética. Pero en el contexto de este capítulo, la red física se refiere al tema mundano de dónde poner las cosas. ¿Cómo va a organizar el equipamiento de forma que pueda alcanzar a sus clientes inalámbricos? Sea que deba llegar hasta una oficina en un edificio o exten-

derse a lo largo de muchas millas, las redes inalámbricas son organizadas en estas tres configuraciones lógicas:

- Enlaces punto a punto
- Enlaces punto a multipunto
- Nubes multipunto a multipunto

El diseño de la red física que elija va a depender de la naturaleza del problema que esté tratando de resolver. Si bien diferentes partes de su red pueden aprovechar las tres configuraciones, los enlaces individuales van a estar dentro de una de esas topologías. La aplicación de estas topologías se describe mejor mediante un ejemplo.

Punto a punto

Los enlaces ***punto a punto*** generalmente se usan para conectarse a Internet donde dicho acceso no está disponible de otra forma. Uno de los lados del enlace punto a punto estará conectado a Internet, mientras que el otro utiliza el enlace para acceder al mismo. Por ejemplo, una Universidad puede tener una conexión *Frame Relay* o una conexión VSAT dentro del campus, pero difícilmente podrá justificar otra conexión de la misma índole a un edificio muy importante fuera del campus. Si el edificio principal tiene una visión libre de obstáculos hacia el lugar remoto, una conexión punto a punto puede ser utilizada para unirlos. Ésta puede complementar o incluso remplazar enlaces discados existentes.

Con antenas apropiadas y existiendo línea visual, se pueden hacer enlaces punto a punto seguros de más de treinta kilómetros.

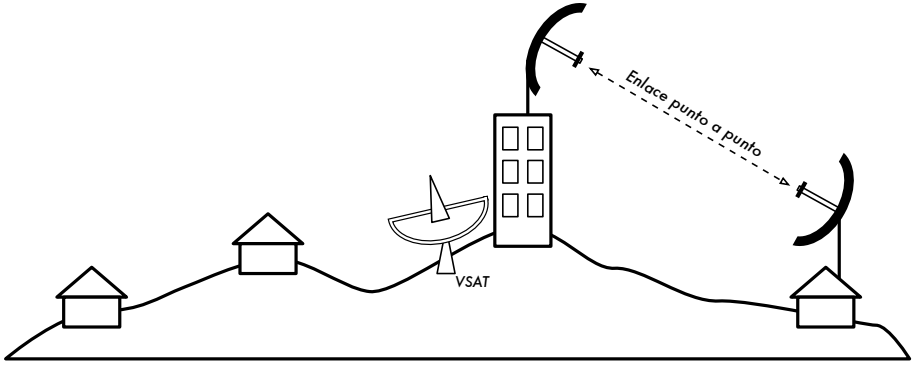

Figura 3.1: Un enlace punto a punto le permite a un lugar remoto compartir una conexión central a Internet.

Por supuesto, una vez hecha una conexión punto a punto, se pueden añadir otras para extender la red aún más. Si en nuestro ejemplo, un edificio alejado se encuentra en la cima de una gran colina, puede ser posible ver otras locaciones importantes que no pueden ser vistas directamente desde el campus central. Mediante la instalación de otro enlace punto a punto hacia el lugar remoto, se puede unir a la red otro nodo y hacer uso de la conexión central a Internet.

Los enlaces punto a punto no necesariamente tienen que estar relacionados con el acceso a Internet. Supongamos que debe desplazarse hasta una estación de monitoreo meteorológico alejada, –ubicada en lo alto de una colina–, para recolectar los datos que ella toma. Podría conectar el lugar con un enlace punto a punto, logrando la recolección y el monitoreo de datos en tiempo real, sin tener que ir hasta el lugar. Las redes inalámbricas pueden proveer suficiente ancho de banda como para transmitir grandes cantidades de datos (incluyendo audio y video) entre dos puntos, aún en ausencia de conexión a Internet.

Punto a multipunto

La siguiente red más comúnmente encontrada es la ***punto a multipunto*** donde varios nodos[1] están hablando con un punto de acceso central, esta es una aplicación punto a multipunto. El ejemplo típico de esta disposición es el uso de un punto de acceso inalámbrico que provee conexión a varias computadoras portátiles. Las computadoras portátiles no se comunican directamente unas con otras, pero deben estar en el rango del punto de acceso para poder utilizar la red.

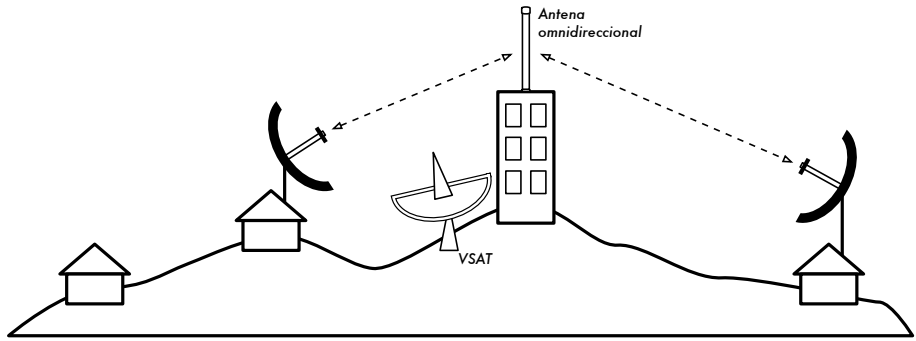

Figura 3.2: La conexión VSAT central es compartida por múltiples sitios remotos. Estos tres lugares también pueden comunicarse directamente a velocidades mucho más rápidas que las ofrecidas por VSAT.

1. Un ***nodo*** es todo dispositivo capaz de enviar y recibir datos en una red. Los puntos de acceso, enrutadores, computadoras y laptops son todos ejemplos de nodos.

La red punto a multipunto también puede ser aplicada a nuestro ejemplo anterior en la universidad. Supongamos que el edificio alejado en la cima de una colina está conectado con el campus central con un enlace punto a punto. En lugar de colocar varios enlaces punto a punto para conexión a Internet, se puede utilizar una antena que sea visible desde varios edificios alejados. Este es un ejemplo clásico de conexión de área extendida ***punto*** (sitio alejado en la colina) a ***multipunto*** (muchos edificios abajo en el valle).

Existen algunas limitaciones con el uso de punto a multipunto en distancias muy grandes, que van a ser tratadas más adelante en este capítulo. Estos enlaces son útiles y posibles en muchas circunstancias, pero no cometamos el clásico error de instalar una torre de radio de gran potencia en el medio de un pueblo esperando ser capaces de servir a miles de clientes, como podría hacerlo con una estación de radio FM. Como veremos, las redes de datos se comportan de forma muy diferente a las emisoras de radiodifusión.

Multipunto a multipunto

El tercer tipo de diseño de red es el ***multipunto a multipunto***, el cual también es denominado red ***ad hoc*** o en malla (***mesh***). En una red multipunto a multipunto, no hay una autoridad central. Cada nodo de la red transporta el tráfico de tantos otros como sea necesario, y todos los nodos se comunican directamente entre sí.

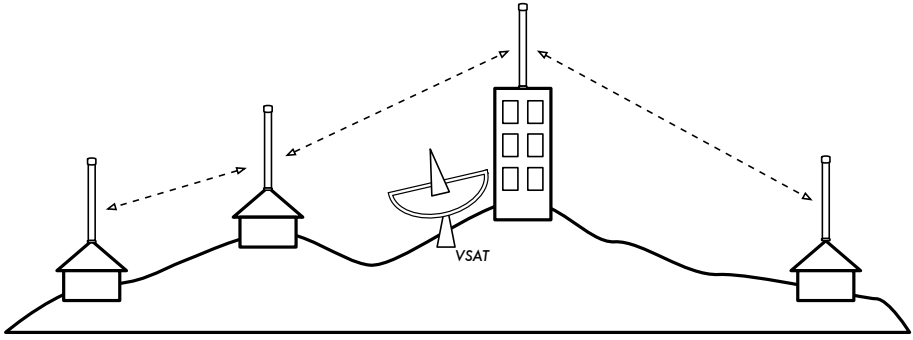

Figura 3.3: Una red en malla (mesh) multipunto a multipunto. Cada punto puede acceder a otro a gran velocidad, o utilizar la conexión central VSAT para acceder a Internet.

El beneficio de este diseño de red es que aún si ninguno de los nodos es alcanzable desde el punto de acceso central, igual pueden comunicarse entre sí. Las buenas implementaciones de redes *mesh* son auto-reparables, detectan automáticamente problemas de enrutamiento y los corrigen. Extender una red *mesh* es tan sencillo como agregar más nodos. Si uno de los nodos en la "nube" tiene acceso a Internet, esa conexión puede ser compartida por todos los clientes.

Dos grandes desventajas de esta topología son el aumento de la complejidad y la disminución del rendimiento. La seguridad de esta red también es un tema importante, ya que todos los participantes pueden potencialmente transportar el tráfico de los demás. La resolución de los problemas de las redes multipunto a multipunto tiende a ser complicada, debido al gran número de variables que cambian al moverse los nodos. Las nubes multipunto a multipunto generalmente no tienen la misma capacidad que las redes punto a punto o las redes punto a multipunto, debido a la sobrecarga adicional de administrar el enrutamiento de la red, y al uso más intensivo del espectro de radio.

Sin embargo, las redes *mesh* son útiles en muchas circunstancias. Al final de este capítulo, vamos a ver algunos ejemplos de cómo construir una red *mesh* multipunto a multipunto utilizando un protocolo de enrutamiento denominado OLSR.

Use la tecnología adecuada

Todos estos diseños de redes pueden ser usados para complementarse unos con otros en una gran red y, obviamente, también se pueden suplementar con técnicas tradicionales de cableado de redes. Es una práctica común, por ejemplo, usar un enlace inalámbrico de larga distancia para proveer acceso a Internet a una ubicación remota, y luego armar un punto de acceso en ese lugar para proveer acceso local. Uno de los clientes de este punto puede también actuar como nodo *mesh*, permitiendo que la red se disperse orgánicamente entre usuarios de computadoras portátiles quienes compartirán el enlace original de acceso a Internet punto a punto.

Ahora que tenemos una idea más clara de la configuración de las redes inalámbricas, podemos comenzar a entender como se realiza la comunicación en dichas redes.

La red lógica

La comunicación es posible sólo cuando los participantes hablan un lenguaje común. Pero una vez que la comunicación se torna más compleja que una simple radiodifusión, los ***protocolos*** se vuelven tan importantes como el lenguaje. Todas las personas en un auditorio pueden hablar inglés, pero sin un conjunto de reglas que establezca quién tiene el derecho a usar el micrófono, la comunicación de las ideas individuales a todo el auditorio es casi imposible. Ahora imagine un auditorio tan grande como el mundo, lleno de todas las computadoras que existen. Sin un conjunto común de protocolos de comunicación que regulen cuándo y cómo cada computador puede hablar, Internet sería una cacofonía, con cada máquina intentando hablar al mismo tiempo.

TCP/IP comprende el conjunto de protocolos que permiten que sucedan las conversaciones en Internet. Entendiendo TCP/IP, usted puede construir redes que virtualmente pueden crecer a cualquier tamaño, y en última instancia formar parte de la Internet global.

El Modelo TCP/IP

Las redes de datos se describen a menudo como construidas en muchas capas. Cada capa depende de la operación de todas las capas subyacentes antes de que la comunicación pueda ocurrir, pero sólo necesita intercambiar datos con la capa superior o la inferior. El modelo de redes TCP/IP[2] comprende 5 capas, como se muestra en este diagrama:

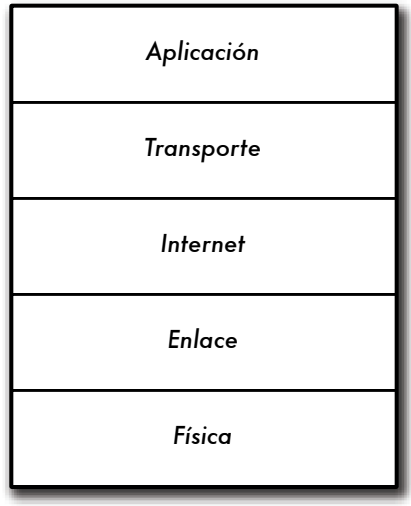

Figura 3.4: El modelo de redes TCP/IP.

En la sección anterior sobre el diseño de redes se describió la capa uno: la ***capa física***. Este es el medio físico donde ocurre la comunicación. Puede ser un cable de cobre CAT5, un cable de fibra óptica, ondas de radio, o cualquier otro medio.

La siguiente capa se denomina ***capa de enlace***. Cuando dos o más nodos comparten el mismo medio físico (por ejemplo, varias computadoras conectadas a un concentrador (*hub*), o un cuarto lleno de computadoras portátiles usando el mismo canal de radio) la capa de enlace establece quién tiene el turno para transmitir en el medio. Ejemplos comunes de protocolos de enlace son Ethernet, Token Ring, ATM, y los protocolos de redes inalámbricas

2. El modelo TCP/IP no es un estándar internacional, y su definición varia. Aquí es incluido como un modelo pragmático utilizado para comprender y solucionar problemas en las redes Internet.

(802.11 a/b/g). La comunicación sobre esta capa se llama de ***enlace local***, ya que todos los nodos pueden comunicarse unos con otros directamente. En redes tipo Ethernet, cada nodo tiene su propia ***dirección MAC*** (***Media Access Control***), que es un número único de 48 bits asignado a cada dispositivo de red cuando es fabricado.

Justo sobre la capa enlace está la ***capa Internet***. Para TCP/IP, está constituido por el Protocolo Internet (**IP**). En la capa Internet, los paquetes pueden salir del enlace local de red y ser retransmitidos a otras redes. Los *enrutadores* realizan esta función teniendo por lo menos dos interfaces de red, una en cada una de las redes a ser interconectadas. Los nodos en Internet son especificados por su única ***dirección IP*** global.

Una vez establecido el enrutamiento en Internet, se necesita un método para alcanzar un servicio particular en una dirección IP dada. Esta función es realizada por la próxima capa, la ***capa de transporte***. TCP y UDP son ejemplos comunes de protocolos de la capa de transporte. Algunos protocolos de la capa de transporte (como el TCP) aseguran que todos los datos han llegado a su destino, y son reensamblados y entregados a la próxima capa en el orden correcto.

Finalmente, en la cima tenemos la ***capa de aplicación***. Esta es la capa con la que la mayoría de los usuarios tienen contacto, y es el nivel en el que ocurre la comunicación humana. HTTP, FTP, y SMTP son todos protocolos de la capa de aplicación. Las personas están por encima de todas estas capas, y necesitan poco o ningún conocimiento de las capas subyacentes para usar efectivamente la red.

Una manera de mirar al modelo TCP/IP es pensar en una persona que entrega una carta en un edificio de oficinas. Va a tener que interactuar primero con la calle (capa física), poner atención al tráfico de la misma (capa de enlace), doblar en los lugares correctos para conectarse con otras calles y llegar a la dirección correcta (capa Internet), ir al piso y oficina correcta (capa transporte), y finalmente encontrar el destinatario o recepcionista que puede recibir la carta (capa de aplicación). Las cinco capas pueden ser recordadas fácilmente usando la frase **F**avor **E**ntrar, **I**nmediatamente **T**omar el **A**scensor, para la secuencia de capas Física, Enlace de Datos, Internet, Transporte, y Aplicación, o en inglés “**P**lease **D**on't **L**ook **I**n **T**he **A**ttic,” que se usa por “**P**hysical / **D**ata **L**ink / **I**nternet / **T**ransport / **A**pplication”

Redes inalámbricas 802.11

Antes de que los paquetes puedan ser reenviados y enrutados en Internet, la capa uno (física) y dos (enlace) necesitan estar conectadas. Sin conectividad de enlace local, los nodos no pueden hablarse y enrutar paquetes.

Para proveer conectividad física, los dispositivos de redes inalámbricas deben operar en la misma porción del espectro de radio. Como pudimos ver en el capítulo dos, esto significa que los radios 802.11a se comunican con otro radio 802.11a en frecuencias de 5GHz, y que los radios 802.11b/g hablan con otros 802.11b/g en 2,4GHz, pero un dispositivo 802.11a no puede interoperar con uno 802.11b/g, puesto que usan porciones completamente diferentes del espectro electromagnético.

Más específicamente, las tarjetas inalámbricas deben concordar en un canal común. Si a una tarjeta de radio 802.11b se le asigna el canal 2 mientras que otra el canal 11, no podrán comunicarse.

Cuando dos tarjetas inalámbricas son configuradas para usar el mismo protocolo en el mismo canal de radio, están prontas para negociar conectividad al nivel de la capa de enlace. Cada dispositivo 802.11a/b/g puede operar en uno de los cuatro modos posibles:

1. El ***Modo maestro*** (también llamado ***AP*** o ***modo de infraestructura***) se utiliza para crear un servicio que parece un punto de acceso tradicional. La tarjeta de red crea una red con un canal y un nombre específico (llamado **SSID**), para ofrecer sus servicios. En el modo maestro, las tarjetas inalámbricas administran todas las comunicaciones de la red (autentificación de clientes inalámbricos, control de acceso al canal, repetición de paquetes, etc.). Las tarjetas inalámbricas en modo maestro sólo pueden comunicarse con tarjetas asociadas a ella en modo administrado.
2. El ***Modo administrado*** es denominado algunas veces ***modo cliente***. Las tarjetas inalámbricas en modo administrado sólo pueden unirse a una red creada por una tarjeta en modo maestro, y automáticamente cambiarán su canal para que corresponda con el de ésta. Luego ellas presentan las credenciales necesarias al maestro, y si estas credenciales son aceptadas, se dice que están asociadas con la tarjeta en modo maestro. Las tarjetas en modo administrado no se comunican unas con otras directamente, y sólo se van a comunicar con una tarjeta asociada en modo maestro.
3. El ***Modo ad hoc*** crea una red multipunto a multipunto donde no hay un único nodo maestro o AP. En el modo *ad hoc*, cada tarjeta inalámbrica se comunica directamente con sus vecinas. Cada nodo debe estar dentro del alcance de los otros para comunicarse, y deben concordar en un nombre y un canal de red.
4. El ***Modo Monitor*** es utilizado por algunas herramientas (tales como Kismet, descrito en el capítulo seis) para escuchar pasivamente todo el tráfico de radio en un canal dado. En el modo monitor, las tarjetas inalámbricas no trasmiten datos. Se utiliza para analizar problemas en un

enlace inalámbrico o para observar el uso del espectro en el área local. El modo monitor no es usado para las comunicaciones normales.

Cuando implementamos un enlace punto a punto, o punto a multipunto, un radio opera en modo maestro, mientras que los otros operan en modo administrado. En una red *mesh* multipunto a multipunto, todos los radios operan en modo *ad hoc* de manera que puedan comunicarse directamente.

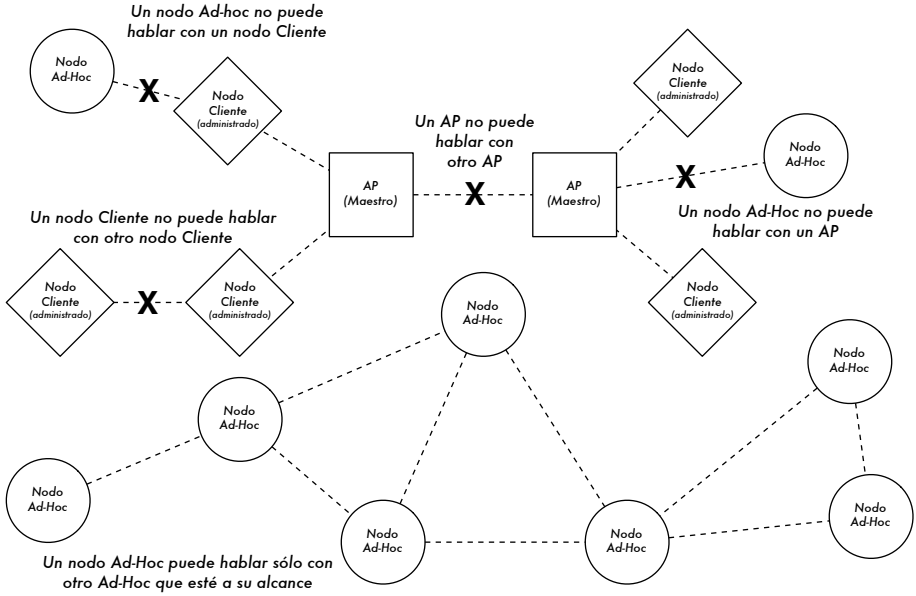

Figura 3.5: AP, clientes, y nodos ad hoc.

Es importante mantener estos modos en mente cuando realiza su diseño de red. Recuerde que los clientes en modo administrado no pueden comunicarse unos con otros directamente, por lo que es posible que quiera instalar un repetidor en modo maestro o *ad hoc*. Como veremos más adelante, el modo *ad hoc* es más flexible pero posee algunos problemas de prestaciones comparado con la utilización de los modos maestro o administrado.

Ahora que sus tarjetas inalámbricas proveen conectividad física y de enlace, están listas para comenzar a pasar paquetes a la capa 3: la capa Internet.

Redes Internet

Direcciones IP, direccionamiento de redes, enrutamiento y reenvío son conceptos relacionados e importantes en redes Internet. Una ***dirección IP*** es un identificador para un nodo de red como un PC, un servidor, un enrutador o un puente. El ***direccionamiento de redes*** es un sistema usado para asignar

estos identificadores en grupos convenientes. El ***enrutamiento*** mantiene un registro del lugar en la red donde están ubicados esos grupos. Los resultados del proceso de enrutamiento se guardan en una lista llamada ***tabla de enrutamiento***. El ***reenvío*** es la acción de usar la tabla de enrutamiento para mandar un paquete al destino final o al "próximo salto" en la dirección a ese destino.

Direcciones IP

En una red IP[3], la dirección es un número de 32 bits, usualmente escrito como 4 números de 8 bits expresados en forma decimal, separados por puntos. Algunos ejemplos de direcciones IP son 10.0.17.1, 192.168.1.1 ó 172.16.5.23.

Direccionamiento de redes

Las redes interconectadas deben ponerse de acuerdo sobre un plan de direccionamiento IP. En Internet, hay comités de personas que asignan las direcciones IP con un método consistente y coherente para garantizar que no se dupliquen las direcciones, y establecen nombres que representan a grupos de direcciones. Esos grupos de direcciones son denominados subredes, o subnets. Grandes subnets pueden ser subdivididas en subnets más pequeñas. Algunas veces un grupo de direcciones relacionadas se denomina espacio de direcciones.

En Internet, ninguna persona u organización posee realmente estos grupos de direcciones porque las direcciones sólo tienen significado si el resto de la comunidad de Internet se pone de acuerdo sobre su uso. Mediante acuerdos, las direcciones son asignadas a organizaciones en relación con sus necesidades y tamaño. Una organización a la cual se le ha asignado un rango de direcciones, puede asignar una porción de ese rango a otra organización como parte de un contrato de servicio. Las direcciones que han sido asignadas de esta manera, comenzando con comités reconocidos internacionalmente, y luego repartidas jerárquicamente por comités nacionales o regionales, son denominadas ***direcciones IP enrutadas globalmente***.

Algunas veces es inconveniente o imposible obtener más de una dirección IP enrutada globalmente para un individuo u organización. En este caso, se puede usar una técnica conocida como Traducción de Direcciones de Red o ***NAT*** (*Network Address Translation*). Un dispositivo NAT es un enrutador con dos puertos de red. El puerto externo utiliza una dirección IP enrutada globalmente, mientras que el puerto interno utiliza una dirección IP de un

---

3. En este libro vamos a tratar primariamente con IPv4, la versión de este protocolo de mayor uso hoy en día. Aunque IPv6 va a remplazar a IPv4 en algún momento futuro, discutir IPv6 está fuera del alcance de este libro.

rango especial conocido como ***direcciones privadas***[4]. El enrutador NAT permite que una única dirección global sea compartida por todos los usuarios internos, los cuales usan direcciones privadas. A medida que los paquetes pasan por él los convierte de una forma de direccionamiento a otra. Al usuario le parece que está conectado directamente a Internet y que no requieren software o controladores especiales para compartir una única dirección IP enrutada globalmente.

Enrutamiento

Internet está cambiando y creciendo constantemente. Continuamente se agregan nuevas redes, se añaden y remueven enlaces entre redes, que fallan y vuelven a funcionar. El trabajo del ***enrutamiento*** es determinar la mejor ruta al destino, y crear una tabla de enrutamiento que liste el mejor camino para todos los diferentes destinos.

Enrutamiento estático es el término utilizado cuando la tabla de enrutamiento es creada por configuración manual. Algunas veces esto es conveniente para redes pequeñas, pero puede transformarse rápidamente en algo muy dificultoso y propenso al error en redes grandes. Peor aún, si la mejor ruta para una red se torna inutilizable por una falla en el equipo u otras razones, el enrutamiento estático no podrá hacer uso de otro camino.

Enrutamiento dinámico es un método en el cual los elementos de la red, en particular los enrutadores, intercambian información acerca de su estado y el estado de sus vecinos en la red, y luego utilizan esta información para automáticamente tomar la mejor ruta y crear la tabla de enrutamiento. Si algo cambia, como un enrutador que falla, o uno nuevo que se pone en servicio, los protocolos de enrutamiento dinámico realizan los ajustes a la tabla de enrutamiento. El sistema de intercambio de paquetes y toma de decisiones es conocido como protocolo de enrutamiento. Hay muchos protocolos de enrutamiento usados en Internet hoy en día, incluyendo OSPF, BGP, RIP, y EIGRP.

Las redes inalámbricas asemejan a las redes cableadas, en el sentido de que necesitan protocolos de enrutamiento dinámicos, pero tienen suficientes diferencias para requerir protocolos de enrutamiento orientados a sus necesidades específicas. En particular, las conexiones de las redes cableadas generalmente funcionan bien o no funcionan (por ejemplo, un cable Ethernet está enchufado o no). Las cosas no son tan claras cuando se trabaja con redes inalámbricas. La comunicación inalámbrica puede ser afectada por objetos en movimiento en el camino de la señal, o por señales que interfieren. Consecuentemente, los enlaces pueden no funcionar bien, o funcionar pobremente, o variar entre los dos extremos. Ya que los protocolos de red

---

4. El término direcciones privadas es definido en RFC 1918, http://www.ietf.org/rfc/rfc1918

existentes no toman en cuenta la calidad de un enlace cuando realizan decisiones de enrutamiento, el comité IEEE 802.11 y el IETF están trabajando en estandarizar protocolos para redes inalámbricas. En la actualidad está poco claro cuándo va a surgir un estándar único que tome en cuenta los enlaces de calidad variable.

Mientras tanto, hay muchos intentos de programación *ad hoc* que quieren solucionar el problema. Algunos ejemplos incluyen ***Hazy Sighted Link State*** (***HSLS***) 'Visión Borrosa del Estado del Enlace', ***Ad-hoc On-demand Distance Vector*** (***AODV***) 'Vector de Distancia bajo Demanda *ad hoc*', y ***Optimized Link State Routing*** (***OLSR***) 'Enrutamiento Optimizado según el Estado de la Red'. Otro es el ***SrcR***, una combinación de DSR y ETX implementada por el proyecto Roofnet del MIT. Más adelante en este capítulo vamos a ver ejemplos de cómo implementar una red utilizando OLSR para realizar decisiones de enrutamiento.

Reenvío

El ***reenvío*** es mucho más sencillo que el direccionamiento y el enrutamiento. Cada vez que un enrutador recibe un paquete, consulta su tabla de enrutamiento interna. Comenzando con el bit más significativo (de mayor orden), escudriña la tabla de enrutamiento hasta encontrar la entrada que tenga el mayor número de bits coincidentes con la dirección destinataria. A esto se le llama ***prefijo*** de la dirección. Si en la tabla se encuentra una entrada que coincide con el prefijo, el campo ***hop count*** (***cuenta de salto***) o ***TTL*** (***tiempo de vida***) se decrementa. Si el resultado es cero, el paquete se descarta y se envía una notificación de error al emisor del mismo. De lo contrario, el paquete se envía al nodo o interfaz especificado en la tabla de enrutamiento. Por ejemplo, si la tabla de enrutamiento contiene estas entradas:

```
Destination     Gateway         Genmask         Flags Metric  Iface
10.15.6.0       0.0.0.0         255.255.255.0   U     0       eth1
10.15.6.108     10.15.6.7       255.255.255.255 UG    1       eth1
216.231.38.0    0.0.0.0         255.255.255.0   U     0       eth0
0.0.0.0         216.231.38.1    0.0.0.0         UG    0       eth0
```

... y el paquete llega con la dirección de destino 10.15.6.23, el enrutador sería enviado por la interfaz eth1. Si el paquete tiene un destino de 10.15.6.108, sería reenviado al gateway (pasarela) 10.15.6.7 (ya que es más específica y hay más coincidencia de bits de alto orden que la ruta a la red 10.15.6.0.).

El destino 0.0.0.0 es una convención especial denominada ***gateway por omisión***. Si ningún prefijo corresponde a la dirección de destino, el paquete es enviado al gateway por omisión. Por ejemplo, si un destino fuera 72.1.140.203, el enrutador reenviaría el paquete a 216.231.38.1 (que pre-

sumiblemente acercaría el paquete a su último destino, y así sucesivamente).

Si un paquete llega y no se encuentra una entrada apropiada (por ej. no se ha definido un gateway por omisión y ningún prefijo corresponde a una ruta conocida), se descarta el paquete y se regresa un paquete de error al emisor inicial.

El campo TTL se utiliza para detectar bucles de enrutamiento. En su ausencia, un paquete podría circular indefinidamente entre dos enrutadores que se listan mutuamente como el mejor próximo salto. Esta clase de bucles puede causar mucho tráfico innecesario en la red y constituye una amenaza a su estabilidad. Usar el campo TTL no soluciona los bucles de enrutamiento, pero ayuda a prevenir la destrucción de una red debido a una mala configuración.

Unificando todo

Una vez que todos los nodos de la red tienen una dirección IP, pueden enviar paquetes de datos a cualquier otro nodo. Mediante el enrutamiento y el reenvío, esos paquetes pueden llegar a nodos en redes que no están conectadas físicamente con el nodo original. Este proceso describe mucho de lo que "sucede" en Internet. Esto es ilustrado en la siguiente figura:

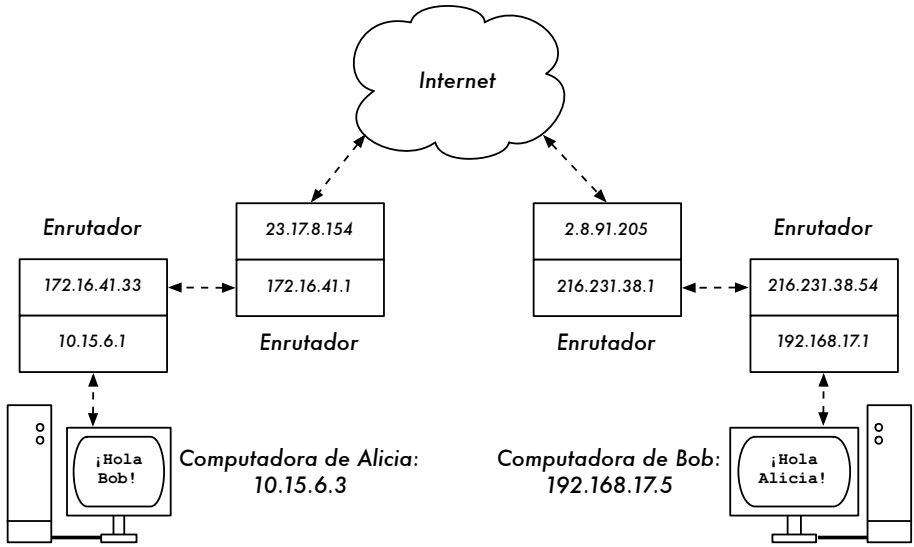

Figura 3.6: Redes Internet. Cada segmento de la red tiene un enrutador con dos direcciones IP, realizando un "enlace local" a dos redes diferentes. Los paquetes son reenviados entre enrutadores hasta que encuentran su destino.

En este ejemplo se puede ver el camino que toman los paquetes cuando Alicia habla con Bob utilizando un servicio de mensajería instantánea. Cada línea punteada representa un cable Ethernet, un enlace inalámbrico, o cualquier otro tipo de red física. El símbolo de la nube es usado comúnmente para "La Internet", y representa cualquier número de redes IP involucradas. Ni Alicia ni Bob necesitan preocuparse de cómo operan esas redes, siempre que los enrutadores reenvíen el tráfico IP hasta el destino final. Si no fuera por los protocolos de Internet y la cooperación de todos en la red, este tipo de comunicación sería imposible.

Ahora que hemos visto cómo fluyen los paquetes en las redes IP, vamos a ver un tipo de red IP muy especializada: una red mallada (*mesh*) OLSR.

Redes mesh con OLSR

La mayoría de las redes WiFi operan en el modo infraestructura: consisten en un punto de acceso en algún lugar (con un radio operando en el modo maestro), conectado a una línea DSL u otra red cableada de larga distancia. En un "hot spot" el punto de acceso generalmente actúa como una estación master que distribuye el acceso a Internet a sus clientes, que operan en el modo administrado. Esta topología es similar al servicio GSM de teléfonos móviles. Los teléfonos móviles se conectan a una estación base sin la cual no se pueden comunicar entre sí. Si hace una llamada en broma a un amigo que está del otro lado de la mesa, su teléfono envía los datos a la estación base de su proveedor que puede estar a una milla de distancia. Luego la estación base reenvía los datos al teléfono de su amigo.

Las tarjetas WiFi en el modo administrado tampoco pueden comunicarse directamente. Los clientes –por ejemplo, dos computadoras portátiles en la misma mesa– tienen que usar un punto de acceso como intermediario. Todo el tráfico entre dos clientes conectados a un punto de acceso debe ser enviado dos veces. Si los clientes A y C se comunican, el cliente A envía datos al punto de acceso B, y luego el punto de acceso va a retransmitir los datos al cliente C. Una transmisión puede tener una velocidad de 600 kbyte/seg (que es prácticamente la máxima velocidad que podemos obtener con 802.11b). En nuestro ejemplo, puesto que los datos deben ser repetidos por el punto de acceso antes de que lleguen a su objetivo, la velocidad real entre ambos clientes va a ser de sólo 300 kbyte/seg.

En el modo *ad hoc* no hay una relación jerárquica entre maestro-cliente. Los nodos pueden comunicarse directamente si están dentro del rango de su interfaz inalámbrica. Por lo tanto, en nuestro ejemplo ambas computadoras podrían conectarse a la velocidad máxima cuando operan en *ad hoc* bajo circunstancias ideales.

La desventaja del modo *ad hoc* es que los clientes no repiten el tráfico destinado a otros clientes. En el ejemplo del punto de acceso, si dos clientes A y C no pueden "verse" directamente con su interfaz inalámbrica, todavía se pueden comunicar si el AP está dentro del rango inalámbrico de ambos clientes.

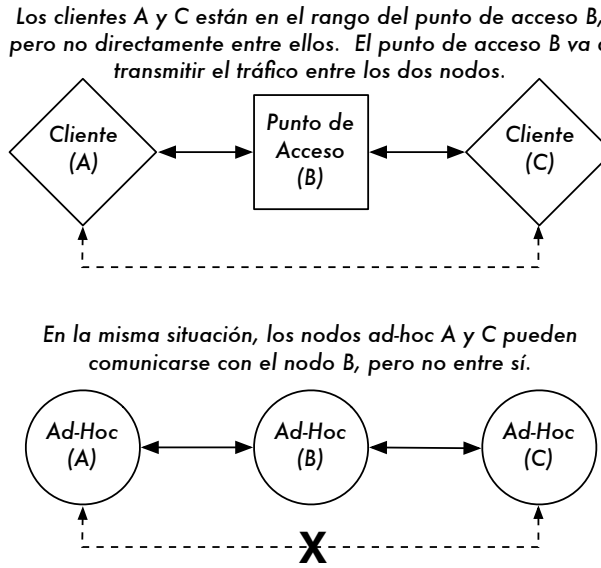

Figura 3.7: El punto de acceso B va a transmitir el tráfico entre los clientes A y C. En el modo ad hoc, el nodo B no va a transmitir el tráfico entre A y C por omisión.

Los nodos *ad hoc* no repiten datos por omisión, pero pueden hacerlo si se aplica el ***enrutamiento***. Las redes malladas (*mesh*) están basadas en la estrategia de que cada nodo actúa como un relevo para extender la cobertura de la red inalámbrica. Cuantos más nodos, mejor será la cobertura de radio y rango de la nube mallada.

Hay un tema álgido que debe ser mencionado en este punto. Si el dispositivo utiliza solamente una interfaz de radio, el ancho de banda disponible se ve reducido significativamente cada vez que el tráfico es repetido por los nodos intermedios en el camino desde A hasta B. Además, va a haber interferencia en la transmisión de esos nodos compartiendo el mismo canal. Por lo tanto, las económicas redes *malladas ad hoc* pueden suministrar muy buena cobertura de radio a una red inalámbrica comunitaria a expensas de la velocidad –especialmente si la densidad de los nodos y la potencia de transmisión son elevadas. Si una red *ad hoc* consiste sólo en unos pocos nodos que están funcionando simultáneamente, si no se mueven y siempre tienen radioenlaces estables –y una larga lista de otras condicionantes– es posible escribir a mano una tabla de enrutamiento individual para todos los nodos.

Desafortunadamente, esas condiciones raramente se encuentran en el mundo real. Los nodos pueden fallar, los dispositivos WiFi pueden cambiar de lugar, y la interferencia puede hacer que los radioenlaces estén inutilizados en cualquier momento. Además nadie quiere actualizar varias tablas de enrutamiento a mano si se adiciona un nodo a la red. Mediante la utilización de protocolos que mantienen automáticamente las tablas de enrutamiento individuales de cada nodo involucrado, podemos olvidarnos de esos temas. Los protocolos de enrutamiento más comunes en el mundo cableado (como el OSPF) no funcionan bien en este ambiente porque no están diseñados para tratar con enlaces perdidos o con topologías que cambian rápidamente.

Enrutamiento mallado con olsrd

El Optimized Link State Routing Daemon –olsrd– (Demonio de Enrutamiento de Estado de Enlace) de *olsr.org* es una aplicación desarrollada para el enrutamiento de redes inalámbricas. Nos vamos a concentrar en este software de enrutamiento por varias razones. Es un proyecto fuente abierta que soporta Mac OS X, Windows 98, 2000, XP, Linux, FreeBSD, OpenBSD y NetBSD. Olsrd está disponible para puntos de acceso que corren Linux como Linksys WRT54G, Asus Wl500g, AccessCube o Pocket PCs que corren Linux Familiar, y viene incluido en los equipos Metrix que corren Metrix Pebble. Olsrd puede manejar interfaces múltiples y puede extenderse con diferentes *plug-ins*. Soporta IPv6 y está siendo desarrollado y utilizado activamente en redes comunitarias alrededor del mundo.

Existen varias implementaciones para OLSR, que comenzaron como un borrador IETF escrito en el INRIA en Francia. La implementación de *olsr.org* comenzó como la tesis de máster de Andreas Toennesen en la Universidad UniK. El demonio de enrutamiento se modificó con base en la experiencia práctica de las redes comunitarias gratuitas. El olsrd actual difiere significativamente del borrador original porque incluye un mecanismo denominado Link Quality Extension (Extensión de la Calidad del Enlace) que mide la cantidad de paquetes perdidos entre nodos y calcula las rutas de acuerdo con esta información. Esta extensión rompe la compatibilidad con los demonios de enrutamiento que adhieren al borrador del INRIA. El olsrd disponible en *olsr.org* puede ser configurado para comportarse de acuerdo al borrador del IETF que carece de esta característica –pero no hay una razón para deshabilitar el Link Quality Extension (Extensión de la Calidad del Enlace), a menos que se requiera la compatibilidad con otras implementaciones.

Teoría

Después de haber corrido olsrd por un rato, cada nodo adquiere conocimiento acerca de la existencia de los otros nodos en la nube *mallada*, y sabe cuáles nodos pueden ser utilizados para enrutar el tráfico hacia ellos. Cada nodo mantiene una tabla de enrutamiento que cubre la totalidad de la nube

mesh. Este enfoque de enrutamiento mallado es denominado ***enrutamiento proactivo***. En contraste, los algoritmos de ***enrutamiento reactivo*** buscan rutas sólo cuando es necesario enviar datos a un nodo específico.

Hay argumentos en favor y en contra del enrutamiento proactivo, y hay muchas otras ideas acerca de cómo hacer el enrutamiento mallado que vale la pena mencionar. La ventaja más grande del enrutamiento proactivo es que sabemos quién está dentro o fuera de la red y no debemos esperar hasta que se encuentre una ruta. El alto tráfico de protocolo y la mayor cantidad de carga de CPU son algunas de las desventajas. En Berlín, la comunidad de Freifunk está operando una nube mallada donde olsrd tiene que administrar más de 100 interfaces. El promedio de carga del CPU causada por olsrd en un Linksys WRT54G corriendo a 200 MHz es aproximadamente del 30% en la *mesh* de Berlín. Hay un límite al grado hasta el cual la extensión de un protocolo proactivo puede escalar –dependiendo de cuántas interfaces estén involucradas y cuán a menudo se actualizan las tablas de enrutamiento.

Mantener rutas en una nube mallada con nodos estáticos toma menos esfuerzo que hacerlo en una *mesh* compuesta de nodos que están en constante movimiento, ya que la tabla de enrutamiento no necesita ser actualizada tan a menudo.

Mecanismo

Un nodo que corre olsrd envía constantemente mensajes de "*Hello*" con un intervalo dado para que sus vecinos puedan detectar su presencia. Cada nodo computa una estadística de cuántos "Hellos" ha recibido y perdido desde cada vecino –de esta forma obtiene información sobre la topología y la calidad de enlace de los nodos en el vecindario. La información de topología obtenida es difundida como mensajes de control de topología (TC messages) y reenviada por los vecinos que olsrd ha elegido para ser relevadores "multipunto".

El concepto de relevadores multipunto es una nueva idea en el enrutamiento proactivo que viene desde el borrador de OLSR. Si cada nodo retransmite la información de topología que ha recibido, se puede generar una sobrecarga innecesaria. Dichas transmisiones son redundantes si un nodo tiene muchos vecinos. Por esta razón, un nodo olsrd decide cuáles vecinos serán designados "relevadores multipunto favorables", encargados de reenviar los mensajes de control de topología. Nótese que los relevadores multipunto son elegidos exclusivamente con el propósito de reenviar mensajes de CT, la carga útil (payload) se enruta utilizando todos los nodos disponibles.

Existen otros dos tipos de mensajes en OLSR que informan cuándo un nodo ofrece una pasarela (*gateway*) a otras redes (mensajes HNA) o tiene múlti-

ples interfaces (mensajes MID). No hay mucho más que decir acerca de estos mensajes más allá del hecho de que existen. Los mensajes HNA hacen al olsrd muy conveniente para conectarse a Internet con un dispositivo móvil. Cuando un nodo *mesh* se mueve detectará pasarelas a otras redes y siempre elegirá la pasarela a la que tenga la mejor ruta. No obstante, olsrd no es a prueba de balas. Si un nodo anuncia que es una pasarela a Internet –cuando en realidad no lo es, porque nunca tuvo acceso o lo perdió– los otros nodos van a creer esta información de todas formas. La seudo-pasarela es un agujero negro. Para solucionar este problema se desarrolló una aplicación de pasarela dinámica. La aplicación detecta automáticamente si la pasarela está verdaderamente conectada y si el enlace está activo. Si no es así, olsrd interrumpe el envío de mensajes HNA falsos. Es muy recomendable construir y utilizar esta aplicación en lugar de depender de los mensajes HNA estáticos.

Práctica

Olsrd implementa enrutamiento IP en una aplicación interna de los usuarios –la instalación es bastante sencilla. Los paquetes de instalación están disponibles para OpenWRT, AccessCube, Mac OSX, Debian GNU/Linux y Windows. OLSR es una parte estándar de Metrix Pebble. Si usted debe compilar desde la fuente, por favor lea la documentación que viene con el paquete. Si todo está configurado correctamente, lo único que tiene que hacer es iniciar el programa OLSR.

En primer lugar debe asegurarse de que cada una de las interfaces del nodo de la *mesh* tiene asignada una dirección IP estática. No se recomienda (ni es práctico) utilizar DHCP en una red IP mallada. Una solicitud DHCP no va a ser contestada por un servidor DHCP si el nodo que la solicita necesita un enlace de múltiples saltos para alcanzarlo, y aplicar relevo de DHCP (DHCP relay) en toda una malla es poco práctico. El problema podría ser resuelto utilizando IPv6, puesto que se dispone de suficientes direcciones para generar una IP a partir de la dirección MAC para cada tarjeta involucrada (como se sugiere en "IPv6 Stateless Address Autoconfiguration in large mobile *ad hoc* networks" por K. Weniger y M. Zitterbart, 2002).

Una página-wiki donde todas las personas interesadas pueden elegir una dirección IPv4 para cada interfaz que esté corriendo OLSR daemon puede ayudar al propósito bastante bien. No existe una manera sencilla de automatizar el proceso cuando se utiliza IPv4.

En general, la dirección de difusión en las interfaces *mesh* debe ser 255.255.255.255, por convención. No hay una razón para ingresar explícitamente la dirección de difusión, ya que olsrd puede ser configurado para reemplazar cualquier dirección de difusión con su valor por convención. Sólo debemos asegurarnos de que las configuraciones son las mismas en todos

lados. Olsrd puede hacer esto por sí mismo. Cuando se establece un archivo de configuración olsrd por omisión, esta característica debe ser habilitada para eliminar confusiones del tipo "¿por qué los otros nodos no pueden ver mi máquina?"

Configuremos ahora la interfaz inalámbrica. Aquí hay un comando que ejemplifica como configurar una tarjeta WiFi con el nombre wlan0 utilizando Linux:

```
iwconfig wlan0 essid olsr.org mode ad-hoc channel 10 rts 250 frag 256
```

Verifique que la parte inalámbrica de la tarjeta WiFi ha sido configurada de manera que tenga una conexión *ad hoc* con otros nodos *mesh* dentro del rango directo (salto único). Asegúrese de que la interfaz usa el mismo canal inalámbrico, el mismo nombre de red inalámbrica ESSID (Extended Service Set IDentifier) y tiene la misma Cell-ID (Identificación de la Célula) que todas las otras tarjetas WiFi que conforman la malla. Muchas tarjetas WiFi o sus respectivos drivers no actúan de acuerdo con el estándar 802.11 para redes *ad hoc* y por lo tanto no pueden conectarse a una célula. Por otro lado pueden ser incapaces de conectarse con otros dispositivos en la misma tabla, aún si están configurados con el canal y el nombre de la red inalámbrica correctos. Incluso pueden confundir otras tarjetas que se comportan de acuerdo con el estándar creando su propio Cell-ID en el mismo canal y con el mismo nombre de red inalámbrica. Las tarjetas WiFi hechas por Intel que son distribuidas en Notebooks Centrino tienen esta falla.

Para comprobar esto puede utilizar el comando **`iwconfig`** cuando utiliza Linux GNU. Aquí están lo resultados de mi computadora:

```
wlan0   IEEE 802.11b  ESSID:"olsr.org"
        Mode:Ad-Hoc  Frequency:2.457 GHz  Cell: 02:00:81:1E:48:10
        Bit Rate:2 Mb/s   Sensitivity=1/3
        Retry min limit:8   RTS thr=250 B   Fragment thr=256 B
        Encryption key:off
        Power Management:off
        Link Quality=1/70  Signal level=-92 dBm  Noise level=-100 dBm
        Rx invalid nwid:0  Rx invalid crypt:28  Rx invalid frag:0
        Tx excessive retries:98024 Invalid misc:117503 Missed beacon:0
```

Es importante configurar el valor umbral "RTS" '*Request To Send*' para una malla, con el fin de mitigar el efecto de las colisiones entre las transmisiones de los nodos del mismo canal. RTS/CTS establece un procedimiento antes de la transmisión de cada paquete para estar seguro de que el canal está libre. Esto implica una sobrecarga, pero incrementa la prestación en el caso de nodos ocultos –¡y éstos son inherentes a una *mesh*! Este parámetro establece el tamaño del paquete más pequeño (en bytes) para el cual el nodo envía RTS. El valor umbral de RTS debe ser menor que IP-Packet Size" –Tamaño del paquete IP– y que el "Fragmentation Threshold" –Umbral de

Fragmentación–; en caso contrario estaría deshabilitado. En nuestro ejemplo este valor es de 256 bytes. TCP es muy sensible a las colisiones, por lo tanto es importante habilitar RTS.

La fragmentación permite dividir un paquete IP en una ráfaga de paquetes más pequeños para su transmisión. Si bien implica una sobrecarga, en un medio ambiente ruidoso esto reduce la penalización por los errores y le permite a los paquetes atravesar ráfagas de interferencia. Las redes *mesh* son muy ruidosas porque los nodos utilizan el mismo canal y por lo tanto las transmisiones están predispuestas a interferir unas con otras. Este parámetro configura el tamaño máximo antes de que un paquete de datos sea dividido y enviado en una ráfaga –un valor igual al tamaño máximo del paquete IP deshabilita el mecanismo, por lo tanto el umbral de fragmentación debe ser menor que el tamaño del paquete IP. Se recomienda utilizar el umbral de fragmentación.

Una vez que se asigna una dirección IP válida y una *máscara de red*, y que la interfaz inalámbrica está funcionando, el archivo de configuración de olsrd debe ser cambiado para que éste encuentre y utilice las interfaces sobre las cuales debe trabajar.

Para Mac OS-X y Windows se dispone de una buena guía para la configuración y el monitoreo del demonio. Desafortunadamente, esto lleva a que los usuarios que tienen poco conocimiento previo hagan mal las cosas –como permitir agujeros negros. En BSD y Linux el archivo de configuración **`/etc/olsrd.conf`** tiene que ser editado con el editor de texto.

Una configuración olsrd simple

No vamos a mostrar un archivo de configuración completo. Aquí hay algunas de las cosas esenciales que deben ser chequedas.

```
UseHysteresis           no
TcRedundancy            2
MprCoverage             3
LinkQualityLevel        2
LinkQualityWinSize      20

LoadPlugin "olsrd_dyn_gw.so.0.3"
{
        PlParam     "Interval"   "60"
        PlParam     "Ping"       "151.1.1.1"
        PlParam     "Ping"       "194.25.2.129"
}

Interface "ath0" "wlan0" {
        Ip4Broadcast 255.255.255.255
}
```

Hay muchas más opciones disponibles en el archivo **`olsrd.conf`**, pero estas opciones básicas le van a permitir comenzar. Después de realizar estos pasos, olsrd puede ser iniciado con un simple comando en el terminal:

```
olsrd -d 2
```

Personalmente, cuando usamos una estación de trabajo recomiendo correrlo con la opción de depuración –d 2, especialmente la primera vez. Podemos ver qué es lo que hace olsrd y monitorear cómo están funcionando los enlaces a sus vecinos. En dispositivos integrados el nivel de depuración debe ser 0 (apagado), porque genera mucha carga en la CPU.

El resultado debe ser algo parecido a esto:

```
--- 19:27:45.51 ------------------------------------------- DIJKSTRA

192.168.120.1:1.00 (one-hop)
192.168.120.3:1.00 (one-hop)

--- 19:27:45.51 ------------------------------------------------ LINKS

IP address       hyst   LQ     lost   total  NLQ    ETX
192.168.120.1    0.000  1.000  0      20     1.000  1.00
192.168.120.3    0.000  1.000  0      20     1.000  1.00

--- 19:27:45.51 ---------------------------------------- NEIGHBORS

IP address       LQ     NLQ    SYM   MPR   MPRS  will
192.168.120.1    1.000  1.000  YES   NO    YES   3
192.168.120.3    1.000  1.000  YES   NO    YES   6

--- 19:27:45.51 ---------------------------------------- TOPOLOGY

Source IP addr   Dest IP addr     LQ     ILQ    ETX
192.168.120.1    192.168.120.17   1.000  1.000  1.00
192.168.120.3    192.168.120.17   1.000  1.000  1.00
```

Utilizar OLSR en Ethernet y en interfaces múltiples

No es necesario tener una interfaz inalámbrica para probar o utilizar olsrd, aunque fue diseñado para éstas. También puede ser utilizado en cualquier NIC. Las interfaces WiFi no tienen que operar siempre en el modo *ad hoc* para formar una malla cuando los nodos *mesh* tienen más de una interfaz. Para los enlaces dedicados puede ser una buena opción que corran en el modo de infraestructura. Muchas tarjetas y manejadores (drivers) WiFi tienen problemas en el modo *ad hoc*, pero el modo de infraestructura trabaja bien –porque todos esperamos que al menos esta característica funcione. El modo *ad hoc* no ha tenido muchos usuarios hasta ahora, por lo que la implementación del mismo ha sido descuidada por muchos fabricantes. Actu-

almente, debido al aumento de la popularidad de las redes *mesh*, se está mejorando esta situación.

Muchas personas utilizan olsrd en interfaces cableadas así como inalámbricas porque no piensan en la arquitectura de red. Simplemente conectan antenas a sus tarjetas WiFi, cables a sus tarjetas Ethernet, habilitan olsrd para que corra en todas las computadoras e interfaces y arrancan. Esto es abusar de un protocolo que fue diseñado para hacer redes inalámbricas en enlaces con pérdidas; pero, ¿por qué no?

Se espera que olsrd sea un superprotocolo. Evidentemente no es necesario enviar mensajes de "Hello" cada dos segundos en una interfaz cableada –pero funciona. Esto no debe ser tomado como una recomendación –simplemente es sorprendente lo que la gente hace con este protocolo y todavía les funciona. De hecho la idea de tener un protocolo que hace todo es muy atractiva para los novatos que quieren tener una LAN enrutada de tamaño pequeño a mediano.

Aplicaciones (*plug-ins*)

Existen varias aplicaciones para olsrd. Para obtener una lista completa, puede chequear el sitio web *olsr.org*. Aquí hay unas instrucciones resumidas para la aplicación de visualización de la topología de la red **`olsrd_dot_draw`**.

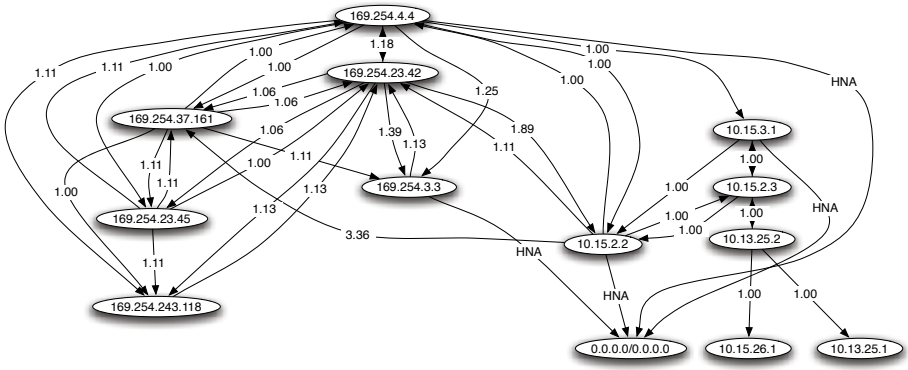

Figura 3.8: Una topología de red OLSR generada automáticamente.

A menudo es muy bueno para la comprensión de una red mallada poder mostrar la topología de la red gráficamente. El olsrd_dot_draw produce la topología en un archivo de formato dot en el puerto TCP 2004. Las herramientas *graphviz* pueden utilizarse para dibujar los gráficos.

Instalar la aplicación dot_draw

Compile todas las aplicaciones OLSR por separado e instálelas. Para cargarlas agregue las siguientes líneas a **`/etc/olsrd.conf`**

```
LoadPlugin     "olsrd_dot_draw.so.0.3"
{
       PlParam "accept" "192.168.0.5"
       PlParam "port" "2004"
}
```

El parámetro "*accept*" especifica el host que fue aceptado para visualizar la Información Topológica (por el momento, uno solo) y es el "localhost" (host local) por omisión. El parámetro "port" especifica el puerto TCP.

Luego reinicie OLSR y chequee si tiene un resultado en el Puerto TCP 2004

```
telnet localhost 2004
```

Después de un rato debe aparecer algún texto.

Puede guardar las descripciones gráficas resultantes y correr las herramientas **dot** o **neato** del paquete *graphviz* para obtener imágenes.

Bruno Randolf ha escrito un pequeño programa perl que obtiene continuamente la información topológica desde olsrd y la despliega utilizando las herramientas gráficas *graphviz* e *ImageMagick*.

Primero instale los siguientes paquetes en su estación de trabajo:

- graphviz, *http://www.graphviz.org/*
- ImageMagick, *http://www.imagemagick.org/*

Descargue el programa en:
http://meshcube.org/nylon/utils/olsr-topology-view.pl

Ahora usted puede correr el programa con **`./olsr-topology-view.pl`** y visualizar la topología actualizada casi en tiempo real.

Resolución de problemas

Siempre que las tarjetas WiFi pueden “verse” directamente con sus radios, la herramienta “ping” funcionará sea que olsrd esté corriendo o no. Esto es así porque las máscaras de red grandes efectivamente hacen de cada nodo un enlace local, por lo que los temas de enrutamiento son eludidos en el primer salto. Esto debe ser chequeado en primer lugar, si las cosas no funcionan

como se espera. La mayoría de los dolores de cabeza que la gente enfrenta con WiFi en el modo *ad hoc* son causados por el hecho de que este modo ha sido implementado descuidadamente en los manejadores (*drivers*) y las tarjetas. Si no es posible hacer *ping* a los nodos que están en el rango, es probable que sea un problema de las tarjetas o los manejadores, o que la configuración de la red esté mal.

Si cada máquina puede hacer *ping* a las otras, pero olsrd no encuentra las rutas, entonces deben chequearse las direcciones IP, la máscara de red y la dirección de difusión.

¿Está utilizando un cortafuego? Asegúrese de que no bloquee el puerto UDP 698.

¡Que se divierta!

Estimando la capacidad

Los enlaces inalámbricos pueden proveer a los usuarios un rendimiento real significantemente mayor que las conexiones tradicionales a Internet, tales como VSAT, discado, o DSL. El rendimiento también se denomina ***capacidad del canal***, o simplemente ***ancho de banda*** (aunque este término no está relacionado con el ancho de banda de las ondas de radio). Es importante comprender que la velocidad listada de los dispositivos inalámbricos (la tasa ***de datos***) se refiere a la tasa a la cual los radios pueden intercambiar símbolos, no al rendimiento que va a observar el usuario. Como mencionamos antes, un enlace 802.11g puede utilizar 54Mbps en el radio, pero el rendimiento real será de unos 22Mbps. El resto es la tara (*overhead*) que necesitan los radios 802.11g para coordinar sus señales.

El rendimiento es una medida de bits por tiempo. 22Mbps significa que en un segundo dado pueden ser enviados hasta 22 megabits desde un extremo del enlace al otro. Si los usuarios intentan enviar más de 22 megabits a través del enlace, va a demorar más de un segundo. Si los datos no pueden ser enviados inmediatamente, son puestos en una ***cola de espera***, y transmitidos tan pronto como sea posible. Esta cola de datos incrementa el tiempo que se necesita para que los bits puestos en la cola más recientemente atraviesen el enlace. El tiempo que le toma a los datos atravesar el enlace es denominado ***latencia***, y una latencia muy grande es denominada comúnmente demora (*lag*). El enlace va a enviar todo el tráfico en espera, pero sus clientes seguramente se quejen al incrementar la demora.

¿Cuánto rendimiento van a necesitar sus usuarios realmente? Esto depende de cuántos usuarios existen y de cómo usan su enlace inalámbrico. Las

diversas aplicaciones de Internet requieren diferentes cantidades de rendimiento.

Aplicación	Ancho de Banda/ Usuario	Notas
Mensajería de texto / IM	< 1 Kbps	Como el tráfico es infrecuente y asincrónico, IM va a tolerar mucha latencia.
Correo electrónico	1 to 100 Kbps	Al igual que IM, el correo electrónico es asincrónico e intermitente, por lo tanto va a tolerar la latencia. Los archivos adjuntos grandes, los virus y el correo no deseado aumentan significativamente la utilización del ancho de banda. Los servicios de correo web (tales como Yahoo o Hotmail) deben ser considerados como navegadores web, no como correo electrónico.
Navegadores web	50 - 100+ Kbps	Los navegadores web sólo utilizan la red cuando se solicitan datos. La comunicación es asincrónica, por lo que se puede tolerar una buena cantidad de demora. Cuando los navegadores web, buscan datos voluminosos (imágenes pesadas, descargas largas, etc.) la utilización del ancho de banda aumenta significativamente.
Flujo de audio (streaming)	96 - 160 Kbps	Cada usuario de un servicio de flujo de audio va a utilizar una cantidad constante de una relativamente gran cantidad de ancho de banda, durante el tiempo que está activo. Puede tolerar algo de latencia pasajera mediante la utilización de mucha memoria de almacenamiento temporal en el cliente (buffer). Pero extensos períodos de espera van a hacer que el audio “salte” o que se den fallos en la sesión.

Aplicación	Ancho de Banda/ Usuario	Notas
Voz sobre IP (VoIP)	24 - 100+ Kbps	Como con el flujo de audio, VoIP dedica una cantidad constante de ancho de banda de cada usuario mientras dura la llamada. Pero con VoIP, el ancho de banda utilizado es aproximadamente igual en ambas direcciones. La latencia en una conexión VoIP molesta inmediatamente a los usuarios. Para VoIP una demora mayor a unas pocas decenas de milisegundos es inaceptable.
Flujo de video (streaming)	64 - 200+ Kbps	Como el flujo de audio, un poco de latencia intermitente es superada mediante la utilización de la memoria de almacenamiento temporal del cliente. El flujo de video requiere de alto rendimiento y baja latencia para trabajar correctamente.
Aplicaciones para compartir archivos Par-a-par (BitTorrent, KaZaA, Gnutella, eDonkey, etc.)	0 - infinitos Mbps	Si bien las aplicaciones par a par (peer-to-peer) toleran cualquier cantidad de latencia, tienden a utilizar todo el rendimiento disponible para transmitir datos a la mayor cantidad de clientes y lo más rápido como les sea posible. El uso de estas aplicaciones causa latencia y problemas de rendimiento para todos los otros usuarios de la red, a menos que se utilice un conformador de ancho de banda adecuado.

Para estimar el rendimiento necesario para su red, multiplique el número esperado de usuarios por el tipo de aplicación que probablemente vayan a usar. Por ejemplo, 50 usuarios quienes están principalmente navegando en la web, en los momentos pico van a consumir entre 2.5 a 5Mbps o más de rendimiento, y se va a tolerar algo de latencia. Por otro lado, 50 usuarios simultáneos de VoIP van a requerir de 5Mbps o más de rendimiento **en ambas direcciones** sin absolutamente nada de latencia. Debido a que el equipamiento inalámbrico 802.11g es *half duplex* (esto es, sólo transmite o recibe, nunca las dos cosas a la vez) debe duplicar el rendimiento requerido por un total de **10Mbps**. Sus enlaces deben proveer esa capacidad cada segundo, o las conversaciones van a tener demora.

Ya que es poco probable que todos sus usuarios utilicen la conexión precisamente al mismo momento, una práctica normal es la de ***sobresuscribir***

el rendimiento disponible por algún factor (esto es, permitir más usuarios de los que el máximo de ancho de banda disponible puede soportar). La sobresuscripción en un factor que va desde 2 a 5 es bastante normal. Probablemente usted utilice sobresuscripción cuando construya su infraestructura de red. Si es cuidadoso en el monitoreo del rendimiento real de su red, va a poder planificar cuándo actualizar diferentes partes de la red, y cuántos recursos adicionales va a necesitar.

Es de esperar que, sin importar cuánta capacidad provea, sus usuarios encuentren aplicaciones que utilicen la totalidad de la misma. Como veremos al final de este capítulo, las técnicas de conformación del ancho de banda pueden ayudar a mitigar algunos problemas de latencia. Mediante la conformación de ancho de banda, ***almacenamiento temporal*** (*caching*) web, así como otras técnicas, se puede reducir significativamente la latencia e incrementar el rendimiento global de su red.

Para tener una experiencia de cómo es una demora en conexiones muy lentas, el ICTP ha creado un simulador de ancho de banda. El mismo descarga una página web a toda velocidad y por otro lado a la tasa reducida que usted elija. Esa demostración le da una visión de cómo el bajo rendimiento y la alta latencia reducen la utilidad de Internet como una herramienta de comunicación. El mismo se encuentra disponible en
http://wireless.ictp.trieste.it/simulator/

Planificar enlaces

Un sistema básico de comunicación consiste de dos radios, cada uno con su antena asociada, separados por la trayectoria que se va a cubrir. Para tener una comunicación entre ambos, los radios requieren que la señal proveniente de la antena tenga un valor por encima de cierto mínimo. El proceso de determinar si el enlace es viable se denomina cálculo del *presupuesto de potencia*. Que las señales puedan o no ser enviadas entre los radios dependerá de la calidad del equipamiento que se esté utilizando y de la disminución de la señal debido a la distancia, denominada ***pérdida en la trayectoria***.

Cálculo del presupuesto del enlace

La potencia disponible en un sistema 802.11 puede caracterizarse por los siguientes factores:

- ***Potencia de Transmisión***. Se expresa en milivatios o en dBm. La Potencia de Transmisión tiene un rango de 30mW a 200mW o más. La potencia TX a menudo depende de la tasa de transmisión. La potencia TX de un dispositivo dado debe ser especificada en los manuales provistos por el

fabricante, pero algunas veces puede ser difícil de encontrar. Algunas bases de datos en línea pueden ayudarlo, una de ellas es la provista por SeattleWireless (*http://www.seattlewireless.net/HardwareComparison*).

- **Ganancia de las Antenas**. Las antenas son dispositivos pasivos que crean el efecto de amplificación debido a su forma física. Las antenas tienen las mismas características cuando reciben que cuando transmiten. Por lo tanto, una antena de 12 dBi simplemente es una antena de 12 dBi, sin especificar si esto es en el modo de transmisión o de recepción. Las antenas parabólicas tienen una ganancia de 19-24 dBi, las antenas omnidireccionales de 5-12 dBi, y las antenas sectoriales, de 12-15 dBi.
- ***El Mínimo Nivel de Señal Recibida***, o simplemente, la sensibilidad del receptor. El RSL (*por su sigla en inglés*) mínimo es expresado siempre como dBm negativos (- dBm) y es el nivel más bajo de señal que la red inalámbrica puede distinguir. El RSL mínimo depende de la tasa de transmisión, y como regla general la tasa más baja (1 Mbps) tiene la mayor sensibilidad. El mínimo va a ser generalmente en el rango de -75 a -95 dBm. Al igual que la potencia TX, las especificaciones RSL deben ser provistas por el fabricante del equipo.
- ***Pérdidas en los Cables***. Parte de la energía de la señal se pierde en los cables, conectores y otros dispositivos entre los radios y las antenas. La pérdida depende del tipo de cable utilizado y de su longitud. La pérdida de señal para cables coaxiales cortos incluyendo los conectores es bastante baja, del rango de 2-3 dB. Lo mejor es tener cables lo más cortos como sea posible.

Cuando calculamos la pérdida en la trayectoria, se deben considerar varios efectos. Algunos de ellos son ***pérdida en el espacio libre***, ***atenuación*** y ***dispersión***. La potencia de la señal se ve disminuida por la dispersión geométrica del frente de onda, conocida comúnmente como pérdida en el espacio libre. Ignorando todo lo demás, cuanto más lejanos los dos radios, más pequeña la señal recibida debido a la pérdida en el espacio libre. Esto es independiente del medio ambiente, se debe solamente a la distancia. Esta pérdida se da porque la energía de la señal radiada se expande en función de la distancia desde el transmisor.

Utilizando los decibeles para expresar la pérdida y utilizando 2,45 GHz como la frecuencia de la señal, la ecuación para la pérdida en el espacio libre es:

```
Lfsl = 40 + 20*log(r)
```

Donde Lfsl (pérdida de señal en el espacio libre, *por su sigla en inglés*) es expresada en dB y r es la distancia en metros entre el transmisor y el receptor.

La segunda contribución para la pérdida en el camino está dada por la atenuación. Esto ocurre cuando parte de la potencia de la señal es absorbida al pasar a través de objetos sólidos como árboles, paredes, ventanas y pisos de edificios. La atenuación puede variar mucho dependiendo de la estructura del objeto que la señal está atravesando, y por lo tanto es muy difícil de cuantificar. La forma más conveniente de expresar esta contribución a la pérdida total es agregando una "pérdida permitida" a la del espacio libre. Por ejemplo, la experiencia demuestra que los árboles suman de 10 a 20 dB de pérdida por cada uno que esté en el camino directo, mientras que las paredes contribuyen de 10 a 15 dB dependiendo del tipo de construcción.

A lo largo del trayecto del enlace, la potencia de RF (radio frecuencia) deja la antena transmisora y se dispersa. Una parte de la potencia de RF alcanza a la antena receptora directamente, mientras que otra rebota en la tierra. Parte de esa potencia de RF que rebota alcanza la antena receptora. Puesto la señal reflejada tiene un trayecto más largo, llega a la antena receptora más tarde que la señal directa. Este efecto es denominado ***multitrayecto***, desvanecimiento o dispersión de la señal. En algunos casos las señales reflejadas se añaden y no causan problemas. Cuando se suman fuera de la fase, la señal recibida es prácticamente nula. En algunos casos, la señal en la antena receptora puede ser anulada por las señales reflejadas. Este fenómeno es conocido como ***anulación***. Existe una técnica simple utilizada para tratar con el multitrayecto, llamada ***diversidad de antena***. Consiste en agregar una segunda antena al radio. De hecho, el Multitrayecto es un fenómeno muy localizado. Si dos señales se suman fuera de fase en una locación, no lo harán en otra locación en las cercanías. Si tenemos dos antenas, al menos una de ellas será capaz de recibir una señal utilizable, aún si la otra está recibiendo una señal distorsionada. En aplicaciones comerciales se utiliza diversidad de antenas conmutadas: tienen múltiples antenas en múltiples entradas con un único receptor. Por lo tanto, la señal es recibida por una única antena a un mismo tiempo. Cuando se transmite, el radio utiliza la última antena usada para la recepción. La distorsión generada por el multitrayecto degrada la habilidad del receptor de recuperar la señal de manera similar a la pérdida de señal. Una forma simple de tomar en cuenta los efectos de la dispersión en el cálculo de la pérdida en el trayecto es cambiar el exponente del factor de la distancia en la fórmula de pérdida en el espacio libre. El exponente tiende a incrementarse con la distancia en un medio ambiente con mucha dispersión. En el exterior con árboles se puede utilizar un exponente de 3, mientras que en el caso de un medio ambiente interno puede usarse uno de 4.

Cuando se combinan pérdida en el espacio libre, atenuación y dispersión, la pérdida en el camino es:

```
L(dB) = 40 + 10*n*log(r) + L(permitida)
```

Donde n es el exponente mencionado.

Para realizar una estimación aproximada de la viabilidad del enlace, se puede evaluar solamente la pérdida en el espacio libre. El medio ambiente puede generar pérdida adicional de señal, y debe ser considerado para una evaluación exacta del enlace. De hecho el medio ambiente es un factor muy importante, y nunca debe ser descuidado.

Para evaluar si un enlace es viable, debemos conocer las características del equipamiento que estamos utilizando y evaluar la pérdida en el trayecto. Cuando hacemos este cálculo, la potencia TX debe ser sumada sólo en uno de los lados del enlace. Si está utilizando diferentes radios en cada lado del enlace, debe calcular la pérdida para cada dirección (utilizando la potencia TX adecuada para cada cálculo). Sumar todas las ganancias y restar las pérdidas resulta en:

```
  TX Potencia de Radio 1
+ Ganancia de la Antena de Radio 1
- Pérdida en los Cables de Radio 1
+ Ganancia de la Antena de Radio 2
- Pérdida en los Cables de Radio 2
_____________________________________

          = Ganancia Total
```

Restar la Pérdida en el trayecto de la Ganancia Total da:

```
          Ganancia Total
    - Pérdida en el trayecto
_____________________________

= Nivel de Señal en un lado del enlace
```

Si el nivel de señal resultante es mayor que el nivel mínimo de señal recibido, entonces ¡el enlace es viable! La señal recibida es suficientemente potente para que los radios la utilicen. Recuerde que el RSL mínimo se expresa siempre como dBm negativos, por lo tanto -56dBm es mayor que -70dBm. En un trayecto dado, la variación en un período de tiempo de la pérdida en el trayecto puede ser grande, por lo que se debe considerar un margen (diferencia entre el nivel de señal recibida y el nivel mínimo de señal recibida). Este margen es la cantidad de señal por encima de la sensibilidad del radio que debe ser recibida para asegurar un enlace estable y de buena calidad durante malas situaciones climáticas y otras anomalías atmosféricas. Un margen de 10-15 dB está bien. Para brindar algo de espacio para la atenuación y el multitrayecto en la señal de radio recibida, se debe tener un margen de 20dB.

Una vez que ha calculado el presupuesto del enlace en una dirección, debe hacer lo mismo en el otro sentido. Substituya la potencia de transmisión del

segundo radio y compare los resultados con el nivel mínimo de señal recibido en el primer radio.

Ejemplo de cálculo del presupuesto del enlace

Como ejemplo, queremos estimar la viabilidad de un enlace de 5km con un punto de acceso y un cliente. El punto de acceso está conectado a una antena omnidireccional de 10dBi de ganancia, mientras que el cliente está conectado a una antena sectorial de 14dBi de ganancia. La potencia de transmisión del AP es 100mW (o 20dBm) y su sensibilidad es -89dBm. La potencia de transmisión del cliente es de 30mW (o 15dBm) y su sensibilidad es de -82dBm. Los cables son cortos, con una pérdida de 2dB a cada lado.

Sumar todas las ganancias y restar todas las pérdidas desde el AP hasta el cliente nos da:

```
  20 dBm (TX Potencia del Radio 1)
+ 10 dBi (Ganancia de la Antena de Radio 1)
-  2 dB  (Pérdida en los Cables de Radio 1)
+ 14 dBi (Ganancia de la Antena de Radio 2)
-  2 dB  (Pérdida en los Cables de Radio 2)
————————
  40 dB = Ganancia Total
```

La pérdida en el trayecto de un enlace de 5km, considerando sólo la pérdida en el espacio libre:

```
Pérdida en el trayecto = 40 + 20log(5000) = 113 dB
```

Restamos la pérdida en el trayecto de la ganancia total

```
40 dB - 113 dB = -73 dB
```

Ya que -73dB es mayor que la sensibilidad del receptor del cliente (-82dBm), el nivel de señal es justo el suficiente para que el cliente sea capaz de oír al punto de acceso. Solamente hay 9dB de margen (82dB - 73dB) que nos permite trabajar bien con buen tiempo, pero probablemente no sea suficiente para enfrentar condiciones climáticas extremas.

Ahora debemos calcular la ganancia desde el cliente hacia el punto de acceso:

```
   15 dBm (TX Potencia del Radio 2)
+ 14 dBi (Ganancia de la Antena de Radio 2)
-  2 dB  (Pérdida en los Cables de Radio 2)
+ 10 dBi (Ganancia de la Antena de Radio 1)
-  2 dB  (Pérdida en los Cables de Radio
_____________

   35 dB = Ganancia Total
```

Obviamente, la pérdida en el camino es la misma en el viaje de vuelta. Por lo tanto, nuestro nivel de señal recibido en el punto de acceso es:

```
35 dB - 113 dB = -78 dB
```

Si la sensibilidad de recepción del AP es -89dBm, nos deja un margen de desvanecimiento de 11dB (89dB - 78dB). En general este enlace probablemente va a funcionar pero podría utilizar un poco más de ganancia. Si usamos un plato de 24dBi en el lado del cliente en lugar de una antena sectorial de 14dBi, vamos a tener una ganancia adicional de 10dBi en ambas direcciones del enlace (recuerde que la ganancia de la antena es recíproca). Una opción más cara puede ser la de utilizar radios de más potencia en ambos extremos del enlace, pero nótese que si agregamos un amplificador o una tarjeta de más potencia en uno sólo de los extremos, esto no ayuda a mejorar la calidad global del enlace.

Existen herramientas en línea que pueden ser utilizadas para calcular el presupuesto del enlace. Por ejemplo, el Green Bay Professional Packet Radio's Wireless Network Link Analysis (*http://my.athenet.net/~multiplx/cgi-bin/wireless.main.cgi*) (Paquete Profesional de Análisis de Enlaces de Redes Inalámbricas de Radio de Green Bay) es una excelente herramienta. La Edición Super genera un archivo PDF que contiene las gráficas de la zona de Fresnel y el trayecto de las ondas de radio. El programa de cálculo también puede ser descargado desde el sitio web e instalado localmente. Veremos en más detalle una excelente herramienta en línea en la siguiente sección, ***Software de planificación de enlace***.

El sitio web de Terabeam también tiene muy buenos calculadores disponibles en línea (*http://www.terabeam.com/support/calculations/index.php*).

Tablas para calcular el presupuesto del enlace

Para calcular el presupuesto del enlace, simplemente estime la distancia y complete las siguientes tablas:

Pérdida en el espacio libre a 2,4GHz

Distancia (m)	100	500	1,000	3,000	5,000	10,000
Pérdida (dB)	80	94	100	110	114	120

Ganancia de la Antena:

Antena Radio 1 (dBi)	+ Antena Radio 2 (dBi)	= Ganancia Total de la Antena

Pérdidas:

Radio 1 + Pérdida en los Cables (dB)	Radio 2 + Pérdida en los Cables (dB)	Pérdida en el espacio libre (dB)	= Pérdida Total (dB)

Presupuesto para el enlace de Radio 1 → Radio 2:

Potencia TX de Radio 1	+ Ganancia de la Antena	- Pérdida Total	= Señal	> Sensibilidad del Radio 2

Presupuesto para el enlace de Radio 2 ➜ Radio 1:

Potencia TX de Radio 2	+ Ganancia de la Antena	- Pérdida Total	= Señal	> Sensibilidad del Radio 1

Si la señal recibida es mayor que la intensidad mínima de señal recibida en ambas direcciones del enlace, entonces el enlace es viable.

Software de planificación de enlace

Si bien calcular el presupuesto de un enlace a mano es sencillo, existen algunas herramientas que ayudan a la automatización del proceso.

Además de calcular la pérdida en el espacio libre, esas herramientas también van a tomar en cuenta otros factores relevantes (tales como absorción de los árboles, efectos del terreno, clima, y además estiman la pérdida en el trayecto en áreas urbanas). En esta sección, vamos a discutir dos herramientas gratuitas que son útiles para planificar enlaces inalámbricos: Green Bay Professional Packet Radio la de utilidades interactivas en línea de diseño de redes, y Radio Mobile.

CGIs para diseño interactivo

El grupo Profesional de Radio de Paquetes de Bahía Verde (GBPRR, *por su sigla en inglés*) ha generado una variedad de herramientas de planificación de enlaces que se encuentran gratuitas en línea. Las mismas se encuentran disponibles en *http://www.qsl.net/n9zia/wireless/page09.html*. Como están disponibles en línea, trabajan con cualquier dispositivo que tenga un navegador web y acceso a Internet.

Veremos la primera herramienta, ***Wireless Network Link Analysis*** (***Análisis de Enlaces de Redes Inalámbricas***), en detalle. La encontrará en línea en *http://my.athenet.net/~multiplx/cgi-bin/wireless.main.cgi*

Para comenzar ingrese el canal que va a ser usado por el enlace. El mismo puede ser especificado en MHz o GHz. Si no conoce la frecuencia, consulte la tabla en el Apéndice B. Tenga en cuenta que la tabla lista la frecuencia central del canal, mientras que la herramienta le solicita la frecuencia de transmisión más alta. De todos modos la diferencia es mínima, por lo que puede utilizar la frecuencia central. Para encontrar la frecuencia más alta de transmisión para un canal agregue 11MHz a la frecuencia central.

Luego ingrese los detalles del lado transmisor del enlace, incluyendo el tipo de línea de transmisión, la ganancia de la antena y otros detalles. Intente completar la mayor cantidad de datos que sepa o que pueda estimar. También puede ingresar la altura de la antena y la elevación para ese lugar. Estos datos van a ser usados para calcular el ángulo de inclinación de la antena. Para calcular el despeje de la zona de Fresnel, va a necesitar el Calculador de la Zona de Fresnel de GBPRR.

La siguiente sección es muy similar, pero incluye información acerca del otro extremo del enlace. Ingrese todos los datos disponibles en los campos apropiados.

Finalmente, la última sección describe el clima, el terreno, y la distancia del enlace. Ingrese todos los datos que conozca o que pueda estimar. La distancia del enlace la puede calcular el programa si usted especifica la latitud y la longitud de ambos lugares. Haga clic en el botón de aceptar para obtener un reporte detallado del enlace propuesto. Éste incluye todos los datos ingresados, así como las pérdidas en el trayecto proyectadas, tasas de error y tiempo que el enlace funcionará satisfactoriamente. Esos números son completamente teóricos, pero le darán una idea general de la viabilidad de enlace. Ajustando los valores de la planilla, puede jugar a "¿y qué pasa sí…?" para ver cómo cambiando los parámetros se afecta la conexión.

Además de la herramienta básica de análisis de enlaces, GBPRR provee una "edición súper" que produce un reporte en formato PDF, así como otras herramientas muy útiles (incluyendo el Calculador de la Zona de Fresnel, Calculador de Distancia y de Rumbo, y Calculador de Conversión de Decibeles, por nombrar algunos). También se provee el código fuente para la mayoría de las herramientas.

Radio Mobile

Radio Mobile es una herramienta para el diseño y simulación de sistemas inalámbricos. Predice las prestaciones de radio enlaces utilizando información acerca del equipamiento y un mapa digital del área. Es un software de dominio público que corre con Windows, pero puede utilizarse en Linux con el emulador Wine.

Radio Mobile usa el ***modelo digital de elevación del terreno*** para el cálculo de la cobertura, indica la intensidad de la señal recibida en varios puntos a lo largo del trayecto. Construye automáticamente un perfil entre dos puntos en el mapa digital mostrando el área de cobertura y la primera zona de Fresnel. Durante la simulación chequea la línea visual y calcula la Pérdida en el trayecto, incluyendo pérdidas debido a los obstáculos. Es posible crear redes de diferentes topologías, incluyendo *master/slave* (maestro/esclavo), punto a punto y punto a multipunto.

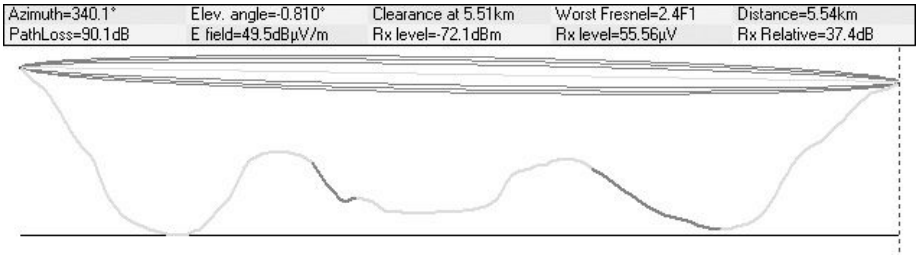

Figura 3.9: Viabilidad del enlace, incluye la zona de Fresnel y estimación de la línea visual, utilizando Radio Mobile

El software calcula el área de cobertura desde la estación de base en un sistema punto a multipunto. Trabaja para sistemas que tienen frecuencias desde 20 kHz a 200 GHz. Los ***Mapas de elevación digital*** (***DEM*** *por su sigla en inglés*) están disponibles gratuitamente desde variadas fuentes y para la mayor parte del mundo. Los DEMs no muestran las líneas costeras u otras fronteras identificables, pero pueden ser combinados fácilmente con otro tipo de datos (como fotos aéreas o cartas topográficas) en varias capas para obtener una representación más útil y rápidamente reconocible. Incluso usted puede digitalizar sus propios mapas y combinarlos con DEMs. Los mapas de elevación digitales pueden combinarse con mapas escaneados, fotos satelitales y servicios de mapas de Internet (tales como Mapquest) para producir predicciones de cobertura precisas.

Radio Mobile puede ser descargado en:
http://www.cplus.org/rmw/download.html

La página principal de Radio Mobile, con ejemplos y tutoriales está disponible en: *http://www.cplus.org/rmw/english1.html*

Radio Mobile bajo Linux

Radio Mobile también funciona utilizando Wine bajo Linux Ubuntu. Si bien las aplicaciones funcionan, algunas etiquetas de los botones pueden quedar mal ubicadas en el marco del botón, lo que puede dificultar su lectura.

Para utilizar Radio Mobile con Linux debemos tener el siguiente entorno:

- IBM Thinkpad x31
- Ubuntu Breezy (v5.10), *http://www.ubuntu.com/*
- Wine versión 20050725, desde el repositorio de Ubuntu Universe

Para instalar Radio Mobile en Windows encontrará instrucciones detalladas en *http://www.cplus.org/rmw/download.html*. Debe seguir todos los pasos excepto el paso 1 (ya que es difícil extraer un DLL desde el archivo

VBRUN60SP6.EXE bajo Linux). Va a tener que copiar el archivo MSVBVM60.DLL desde una computadora con Windows que ya tenga instalado Visual Basic 6 run-time, o buscar en Google el archivo MSVBVM60.DLL y descargarlo.

Continúe con el paso 2 desde la URL anterior, asegúrese de descomprimir los archivos descargados en el mismo directorio dentro del cual ha colocado los archivos DLL. No debe preocuparse por los pasos que siguen al 4; esos son pasos extra, necesarios sólo para los usuarios de Windows.

Finalmente puede iniciar Wine desde una terminal con el comando:

```
# wine RMWDLX.exe
```

En este punto debe ver Radio Mobile corriendo en su sesión XWindows.

Evitando el ruido

Las bandas libres de licenciamiento ISM y U-NII representan una porción muy pequeña del espectro electromagnético conocido. Debido a que esta región puede ser utilizada sin pagar costos de licenciamiento, muchos dispositivos comerciales la utilizan para un amplio rango de aplicaciones. Teléfonos inalámbricos, transmisores de video analógicos, *Bluetooth*, monitores de bebés, e incluso los hornos de microondas compiten con las redes de datos inalámbricas por el uso de la muy limitada banda de 2,4GHz. Esas señales, así como otras redes inalámbricas locales, pueden causar problemas significativos para los enlaces inalámbricos de largo alcance. Para reducir la recepción de señales no deseadas le describimos algunos pasos que puede utilizar.

- **Incremente la ganancia de la antena en ambos extremos del enlace punto a punto.** Las antenas no sólo agregan ganancia a un enlace, sino que el aumento de la directividad tiende a rechazar el ruido proveniente de los alrededores del enlace. Dos platos de alta ganancia que están enfocados uno al otro, rechazarán el ruido desde direcciones que están fuera del trayecto del enlace. Si utilizamos antenas omnidireccionales recibiremos ruido de todas las direcciones.
- **No utilice un amplificador.** Como veremos en el capítulo cuatro, los amplificadores pueden hacer que los problemas de interferencia empeoren con la amplificación indiscriminada de todas las señales recibidas. Al mismo tiempo, causan problemas de interferencia para los otros usuarios de la banda que se encuentren cerca.
- **Utilice antenas sectoriales en lugar de omnidireccionales.** Haciendo uso de varias antenas sectoriales puede reducir el ruido global recibido en un punto de distribución. Si organiza los canales utilizados en cada antena

sectorial, también puede incrementar el ancho de banda disponible para sus clientes.

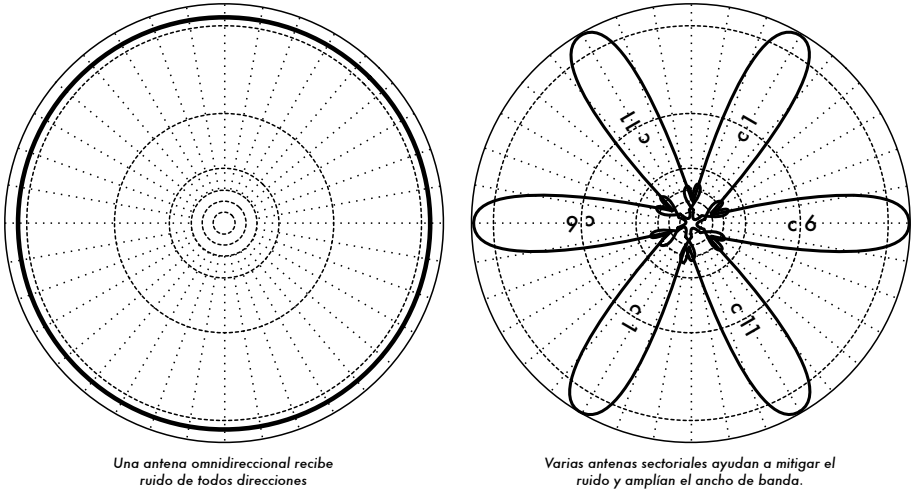

Figura 3.10: Una sola antena omnidireccional vs. múltiples antenas sectoriales.

- **Utilice el mejor canal disponible.** Recuerde que los canales 802.11b/g tienen un acho de 22MHz, pero están separados sólo por 5MHz. Realice una prospección del sitio (como se detalla en el capítulo ocho), y seleccione el canal que esté tan lejos como sea posible de las fuentes de interferencia existentes. Tenga en cuenta que el paisaje inalámbrico puede cambiar en cualquier momento ya que la gente puede agregar nuevos dispositivos (teléfonos inalámbricos, otras redes, etc.). Si de pronto su enlace tiene problemas para enviar paquetes, es posible que deba realizar otra prospección y tomar un canal diferente.
- **Utilice pequeños saltos y repetidores, en lugar de una única tirada a larga distancia. Mantenga sus enlaces punto a punto lo más corto posible.** Si bien es posible crear un enlace de 12km que cruce por el medio de una ciudad, es muy probable que tenga todo tipo de problemas de interferencia. Si puede quebrar ese enlace en dos o tres saltos más cortos, el enlace va a ser más estable. Obviamente, esto es imposible en enlaces rurales a larga distancia, donde se carece de las estructuras de montaje y de energía en los puntos intermedios, pero en estos casos los problemas de ruido son improbables.
- **Si es posible, utilice las bandas 5,8GHz, 900MHz, u otra banda sin licenciamiento.** Si bien esta es una solución a corto plazo, actualmente la mayor parte del equipamiento instalado utiliza 2,4GHz. Utilizar 802.11a, o un dispositivo de convertidor de 2,4GHz a 5,8GHz le va a permitir eludir esta congestión. Si usted puede encontrarlo, existe equipamiento 802.11 viejo que usa el espectro sin licenciamiento a 900MHz (desafortunadamente con un muy baja velocidad). Otras tecnologías tales como Ronja

(*http://ronja.twibright.com/*) usan tecnología óptica para enlaces a corta distancia libres de ruido.

- **Si todo esto falla, utilice un espectro con licenciamiento.** Hay lugares donde todo el espectro sin licenciamiento está siendo utilizado. En esos casos, puede tener sentido gastar el dinero adicional para tener un equipamiento propio que utilice una banda menos congestionada. Para enlaces punto a punto a larga distancia que requieren de muy alto rendimiento y máximo tiempo de disponibilidad, esta es, ciertamente, una opción. Por supuesto esto implica un precio mucho mayor comparado con el equipamiento sin licenciamiento.

Para identificar las fuentes del ruido, necesita herramientas que le muestren qué está sucediendo en el aire a 2,4GHz. Vamos a ver algunos ejemplos de estas herramientas en el capítulo seis.

Repetidores

El componente más crítico para construir un enlace de red a larga distancia es la existencia de ***línea visual*** (a menudo abreviada como ***LOS*** *por su sigla en inglés*). Los sistemas de microondas terrestres simplemente no pueden tolerar colinas altas, árboles, u otros obstáculos en el camino de un enlace a larga distancia. Es necesario que se tenga una idea del relieve de la tierra entre dos puntos antes de poder determinar si un enlace es posible.

Pero aún si hay una montaña entre dos puntos, debemos tener presente que los obstáculos pueden ser transformados en activos. Las montañas pueden bloquear la señal, pero suponiendo que se pueda proveer energía, también pueden actuar como muy buenos ***repetidores***.

Los repetidores son nodos que están configurados para transmitir el tráfico que no es destinado al nodo. En una red mallada, cada nodo es un repetidor. En una red de infraestructura tradicional, los nodos deben ser configurados específicamente para poder pasar el tráfico a otros nodos.

Un repetidor puede usar uno o más dispositivos inalámbricos. Cuando utiliza un sólo radio (denominado ***repetidor de una mano***), la eficiencia global es ligeramente menor que la mitad del ancho de banda disponible, puesto que el radio puede enviar o recibir datos, pero no simultáneamente. Esos dispositivos son baratos, simples y tienen bajos requerimientos de potencia. Un repetidor con dos (o más) tarjetas de radio puede operar todos los radios a toda capacidad, siempre que los mismos estén configurados para usar canales que no se superpongan. Por supuesto, los repetidores también pueden proveer una conexión Ethernet para conectividad local.

Los repetidores pueden ser adquiridos como un juego completo, o fácilmente ensamblados conectando dos o más nodos inalámbricos con un cable de Ethernet. Cuando planee usar un repetidor construido con tecnología 802.11, tenga en cuenta que cada nodo debe ser configurado en el modo maestro, administrado o *ad hoc* que le corresponda. Generalmente, ambos radios en el repetidor están configurados en el modo maestro para permitir que los múltiples clientes puedan conectarse a cualquier lado del repetidor. Pero dependiendo de su diseño de red, uno o más dispositivos van a necesitar utilizar el modo *ad hoc* o el modo cliente. En general, los repetidores son utilizados para evitar obstáculos en el camino de un enlace a larga distancia. Los mismos pueden ser edificios en el camino, pero esos edificios contienen gente. A menudo podemos hacer acuerdos con los dueños de los edificios para proveerles de ancho de banda a cambio de utilizar la azotea y la electricidad. Si el dueño del edificio no está interesado, podemos intentar persuadir a los inquilinos de los pisos más altos para instalar equipamiento en una ventana.

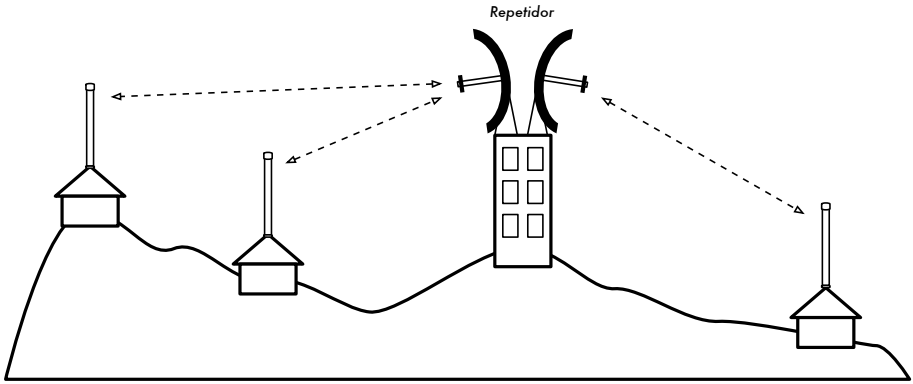

Figura 3.11: El repetidor reenvía los paquetes por el aire entre los nodos que no tienen una línea visual directa.

Si usted no puede pasar sobre, o a través de un obstáculo, a menudo lo puede rodear. En lugar de usar un enlace directo, intente hacer un salto múltiple para eludir el obstáculo.

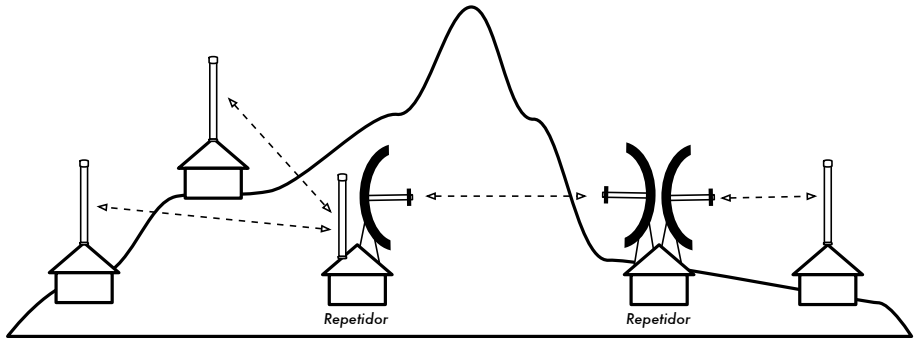

Figura 3.12: No había energía disponible en lo alto de la colina, pero fue circunvalada con el uso de múltiples repetidores ubicados alrededor de la base.

Finalmente, usted podría necesitar ir hacia atrás para poder avanzar. Si tenemos un lugar alto en una dirección diferente, y ese lugar puede ver más allá del obstáculo, se puede hacer un enlace estable a través de una ruta indirecta.

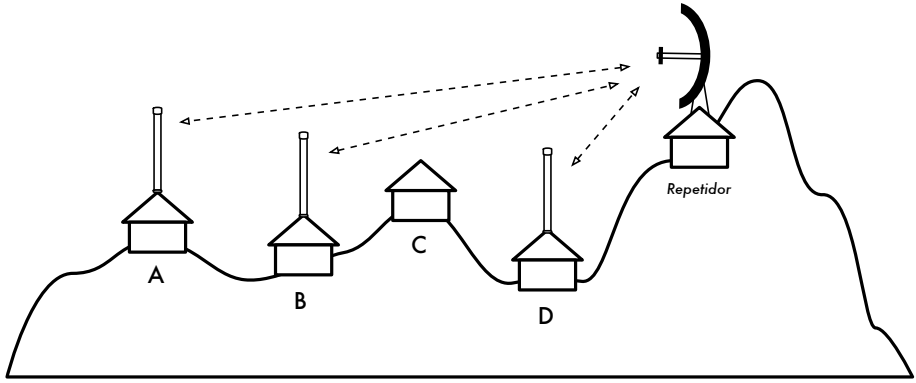

Figura 3.13: El lugar D no puede ver al lugar A o el B, porque el lugar C está en el camino y no está interesado en tener un nodo. Al instalar un repetidor en un lugar alto los nodos A, B, y D se pueden comunicar. El tráfico desde el nodo D en realidad viaja más lejos que el del resto de la red antes de que el repetidor reenvíe esos datos.

Los repetidores en la red me recuerdan el principio de “los seis grados de separación”. Esta idea dice que no importa a quién está buscando, sólo necesita contactar cinco intermediarios antes de encontrar a la persona. Los repetidores pueden “ver” una gran cantidad de intermediarios, y si su nodo está dentro del rango podrá comunicarse con cualquier nodo que el repetidor pueda alcanzar.

Optimización del Tráfico

El ancho de banda se mide como un cociente de número de bits transmitidos en un segundo. Esto significa que dado suficiente tiempo, la cantidad de información transmisible en cualquier enlace se acerca al infinito. Desafortunadamente, para un período de tiempo finito, el ancho de banda provisto por una conexión de red cualquiera no es infinito. Siempre puede descargar (o cargar) tanto tráfico como quiera; sólo que debe esperar todo lo que sea necesario. Por supuesto que los usuarios humanos no son tan pacientes como las computadoras, y no están dispuestos a esperar una infinita cantidad de tiempo para que su información atraviese la red. Por esta razón, el ancho de banda debe ser gestionado y priorizado como cualquier otro recurso limitado.

Se puede mejorar significativamente el tiempo de respuesta y maximizar el rendimiento disponible mediante la eliminación del tráfico indeseado y redundante de nuestra red. Esta sección describe varias técnicas comunes para asegurarse de que nuestra red solamente está transportando el tráfico que debe y no otro.

Almacenamiento Web temporal

Un servidor web *proxy* es un servidor en la red local que mantiene copias de lo que se ha leído recientemente, páginas web que son utilizadas a menudo, o partes de esas páginas. Cuando la siguiente persona busque esas páginas, las mismas se recuperan desde el servidor *proxy* local sin ir hasta Internet. Esto resulta, en la mayoría de los casos en un acceso al web más rápido, al mismo tiempo que se reduce significativamente la utilización del ancho de banda con Internet. Cuando se implementa un servidor *proxy*, el administrador debe saber que existen algunas páginas que no son almacenables, por ejemplo, páginas que son el resultado de programas del lado del servidor, u otros contenidos generados dinámicamente.

Otra cosa que también se ve afectada es la manera como se descargan las páginas web. Con un enlace a Internet lento, una página normal comienza a cargarse lentamente, primero mostrando algo de texto y luego desplegando los gráficos uno por uno. En una red con un servidor *proxy*, puede haber un retraso durante el cual parece que nada sucede, y luego la página se carga por completo rápidamente. Esto sucede porque la información es enviada a la computadora tan rápido que para el rearmado de la página se toma una cantidad de tiempo perceptible. El tiempo global que toma este procedimiento puede ser sólo de diez segundos (mientras que sin un servidor *proxy*, puede tomar 30 segundos cargar la página gradualmente). Pero a menos que esto se explique a algunos usuarios impacientes, estos pueden decir que el servidor *proxy* está haciendo las cosas más lentamente. General-

mente es tarea del administrador lidiar con la percepción de los usuarios acerca de temas como éste.

Servidores *proxy*

Existen varios servidores proxy disponibles. Los que siguen son los paquetes de software utilizados más comúnmente:

- **Squid.** El software libre Squid es el estándar de facto en las universidades. Es gratuito, confiable, sencillo de utilizar y puede ser mejorado (por ejemplo, añadiendo filtros de contenido y bloqueos de publicidad). Squid produce bitácoras (*logs*) que pueden ser analizadas utilizando software como Awstats, o Webalizer, los cuales son de fuente libre y producen buenos reportes gráficos. En la mayoría de los casos, es más fácil instalarlo como parte de la distribución en lugar de descargarlo desde *http://www.squid-cache.org/* (la mayoría de las distribuciones Linux como Debian, así como otras versiones de Unix como NetBSD y FreeBSD vienen con Squid). Una buena guía de configuración de Squid se puede encontrar en: *http://squid- docs.sourceforge.net/latest/book-full.html*.
- **Servidor Proxy Microsoft 2.0.** No está disponible para instalaciones nuevas porque ha sido reemplazado por el servidor Microsoft ISA y ha dejado de tener soporte. Si bien es utilizado por algunas instituciones es mejor no considerarlo para instalaciones nuevas.
- **Servidor Microsoft ISA.** ISA es un muy buen programa de servidor *proxy*, pero demasiado caro para lo que hace. Sin embargo, con descuentos académicos puede ser accesible para algunas instituciones. Produce sus propios reportes gráficos, pero sus archivos de bitácora (log) también pueden ser analizados con el popular software Sawmill (*http://www.sawmill.net/*). Los administradores de un sitio con un Servidor MS ISA deben dedicar tiempo suficiente para obtener la configuración adecuada; por otra parte, el Servidor MS ISA Server puede utilizar gran cantidad de ancho de banda. Por ejemplo, una instalación por omisión puede consumir fácilmente más ancho de banda que lo que el sitio ha utilizado anteriormente, porque las páginas comunes con fechas de expiración cortas (tales como los sitios de noticias) se actualizan continuamente. Por lo tanto, es importante que la captura preliminar (prefetching) se configure correctamente, para que sea realizada durante la noche. El servidor ISA también puede ser asociado a productos de filtrado de contenidos tales como WebSense. Para más información vea el sitio: *http://www.microsoft.com/isaserver/* y *http://www.isaserver.org/*.

Evitando que los usuarios evadan el servidor proxy

Si bien eludir la censura de Internet y las políticas de acceso restrictivo a la información son un laudable esfuerzo político, los servidores *proxy* y los

firewalls son herramientas necesarias en áreas con anchos de banda extremadamente limitados. Sin ellos la estabilidad y la usabilidad de la red se ven amenazadas por los propios usuarios legítimos de la red. Las técnicas para eludir un servidor *proxy* pueden ser encontradas en: *http://www.antiproxy.com/*. Este sitio es útil para que los administradores vean cómo sus redes pueden enfrentarse a estas técnicas.

Para reforzar el uso del almacenamiento ***temporal proxy*** (*caching proxy*), puede simplemente considerarse instaurar una política de acceso a la red y confiar en sus usuarios. En el diseño que sigue, el administrador debe confiar en que los usuarios no van a eludir el servidor *proxy*.

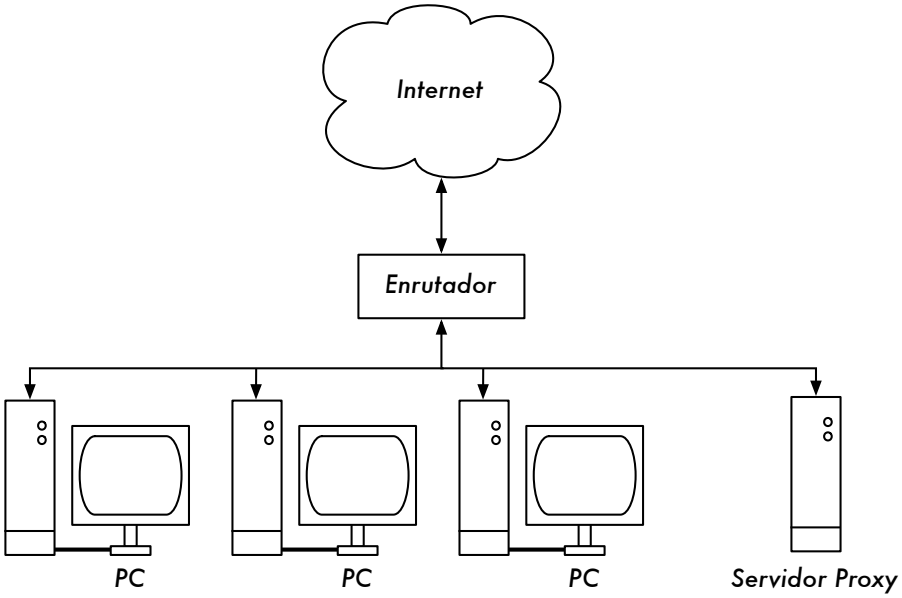

Figura 3.14: Esta red se basa en la confianza en que los usuarios van a configurar apropiadamente sus PCs para utilizar el servidor proxy.

En este caso el administrador generalmente utiliza una de las siguientes técnicas:

- **No divulgar la dirección de la pasarela por omisión (default gateway) a través de DCHP.** Esto puede funcionar por un tiempo, pero algunos usuarios que quieren eludir el proxy pueden encontrar o buscar la dirección de **la pasarela por omisión**. Una vez que esto pasa, se tiende a difundir cómo se elude el proxy.
- **Utilizar políticas de grupo o de dominio.** Esto es muy útil para configurar el servidor *proxy* adecuado para Internet Explorer en todas las computadoras del dominio, pero no es muy útil para evitar que el *proxy* sea eludido, porque se basa en el registro de un usuario en el dominio NT. Un

usuario con una computadora con Windows 95/98/ME puede cancelar su registro y luego eludir el *proxy*, y alguien que conoce la contraseña de un usuario local en su computadora con Windows NT/2000/XP puede registrarse localmente y hacer lo mismo.

- **Rogar y luchar con los usuarios.** Ésta nunca es una situación óptima para un administrador de red. La única forma de asegurarse que los *proxy* no van a ser eludidos es mediante la utilización del diseño de red adecuado, por medio de una de las tres técnicas descritas a continuación.

Cortafuego (*Firewall*)

Una de las maneras más confiable para asegurarse que las PC no van a eludir el *proxy* puede ser implementada utilizando un cortafuego.

El cortafuego puede configurarse para que solamente pueda pasar el servidor proxy, por ejemplo, para hacer solicitudes de HTTP a Internet. Todas las demás PC están bloqueadas, como se muestra en el siguiente diagrama.

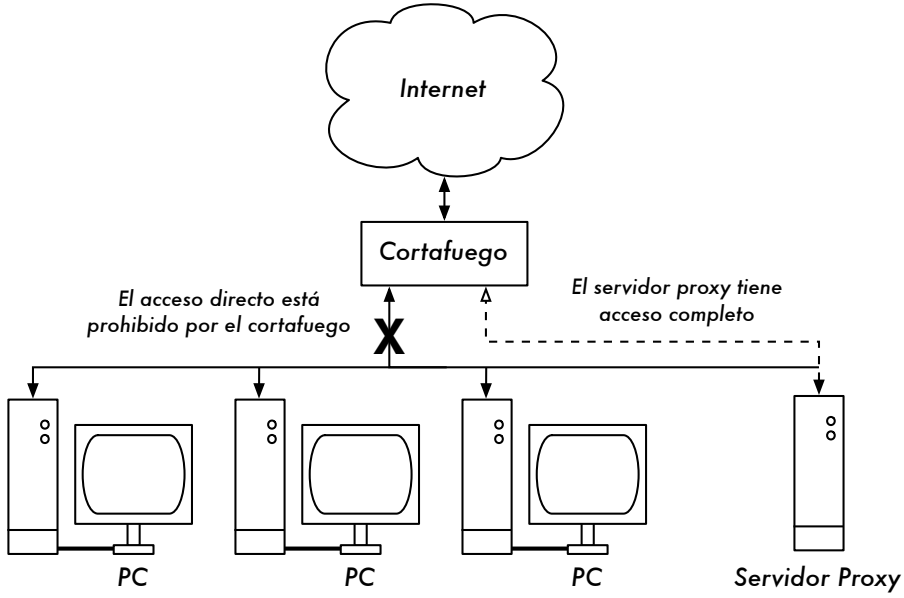

Figura 3.15: El cortafuego les impide a los PCs acceder a Internet directamente, pero les permite el acceso a través del servidor proxy.

Confiar en un cortafuego, como en el diagrama anterior, puede o no ser suficiente, dependiendo de cómo esté configurado. Si sólo bloquea el acceso desde la LAN del campus al puerto 80 en los servidores web, va a haber formas, para los usuarios inteligentes, de encontrar caminos que lo rodeen.

Aún más, van a ser capaces de utilizar protocolos sedientos de ancho de banda como Kazaa.

Dos tarjetas de red

Posiblemente, el método más confiable es el de instalar dos tarjetas de red en el servidor *proxy* y conectar la red del campus a Internet como se muestra en la siguiente figura. De esta forma, el diseño de red hace físicamente imposible alcanzar la Internet sin pasar a través del servidor *proxy*.

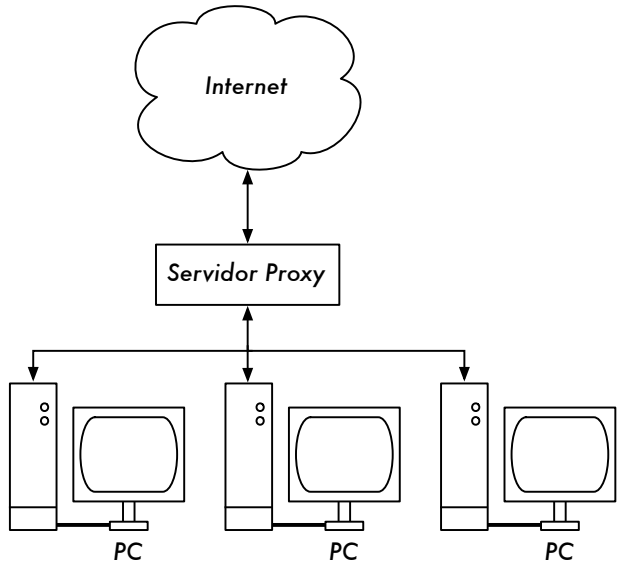

Figura 3.16: La única ruta hacia Internet es a través del proxy.

El servidor *proxy* en este diagrama no debe tener habilitado IP *forwarding*, a menos que los administradores conozcan exactamente qué es lo que quieren dejar pasar.

Una gran ventaja de este diseño es que puede utilizarse una técnica conocida como ***transparent proxying***. Utilizar *proxy* transparente significa que las solicitudes web de los usuarios son reenviadas automáticamente al servidor *proxy*, sin ninguna necesidad de configurar manualmente los navegadores web para que lo utilicen. Esto fuerza efectivamente a que todo el tráfico web sea almacenado localmente, lo que elimina muchas posibilidades de error de los usuarios, y va a trabajar incluso con dispositivos que no soportan el uso de un *proxy* manual. Para más detalles sobre cómo configurar un *proxy* transparente con Squid, diríjase a:

- *http://www.squid-cache.org/Doc/FAQ/FAQ-17.html*

- *http://en.tldp.org/HOWTO/mini/TransparentProxy-2.html*

Enrutamiento basado en políticas

Una forma de prevenir la circunvalación del *proxy* utilizando equipamiento Cisco es con una política de enrutamiento. El enrutador Cisco dirige transparentemente las solicitudes web al servidor *proxy*. Esta técnica es utilizada en la Universidad Makerere. La ventaja de este método es que, si el servidor *proxy* está caído, las políticas de enrutamiento pueden ser removidas temporalmente permitiéndoles a los clientes conectarse directamente a Internet.

Sitio web espejo (*mirror*)

Con el permiso del dueño o del administrador del sitio web, el sitio completo puede ser copiado durante la noche al servidor local, siempre que el mismo no sea demasiado grande. Esto es algo que se debe tener en cuenta para sitios web importantes, que son de interés particular para la organización, o que son muy populares entre los usuarios de la web. Si bien esto puede ser útil, tiene algunas fallas potenciales. Por ejemplo, si el sitio que es duplicado contiene programas CGI u otros contenidos dinámicos que requieren de interacción con el usuario, va a haber problemas. Un ejemplo es el sitio web que requiere que la gente se registre en línea para una conferencia. Si alguien se registra en línea en un servidor duplicado (y el programa de duplicado funciona bien), los organizadores del sitio no van a tener la información de que la persona se registró.

Debido a que un sitio duplicado puede infringir los derechos de copyright, esta técnica debe ser utilizada solamente con el permiso del sitio en cuestión. Si el sitio corre ***rsync***, puede ser duplicado utilizando rsync. Ésta es la forma más rápida y eficiente de mantener los contenidos del sitio sincronizados. Si el servidor web remoto no está corriendo rsync, se recomienda utilizar el software llamado ***wget***. Éste es parte de la mayoría de las versiones de Unix/Linux. Una versión de Windows puede encontrase en *http://xoomer.virgilio.it/hherold/*, o en el paquete de herramientas gratuito de Cygwin Unix (*http://www.cygwin.com/*).

Se puede utilizar un *script* que corra cada noche en un servidor web local y haga lo siguiente:

- Cambiar el directorio raíz del servidor web: por ejemplo, **`/var/www/`** en Unix, o **`C:\Inetpub\wwwroot`** en Windows.
- Duplicar el sitio web utilizando el siguiente comando:

```
wget --cache=off -m http://www.python.org
```

El sitio duplicado va a estar en el directorio **`www.python.org`**. El servidor web debe ser configurado para servir los contenidos de ese directorio como un host virtual basado en nombre. Ponga en marcha el servidor local DNS para falsificar una entrada para este sitio. Para que esto funcione, las PC clientes deben ser configuradas para usar el/los servidor(es) DNS local(es) como el DNS primario. (Esto es siempre aconsejable, porque el almacenamiento intermedio (*caching*) del servidor DNS acelera los tiempos de respuesta web).

Pre-poblar la memoria intermedia (*cache*) utilizando wget

En lugar de instalar un sitio web duplicado como se describió en la sección anterior, un mejor enfoque es el de poblar el *proxy* cache utilizando un proceso automatizado. Este método ha sido descrito por J. J. Eksteen y J. P. L. Cloete del CSIR en Pretoria, Sud África, en un artículo titulado **Mejorar el Acceso a la Red de Redes en Mozambique a Través del Uso de Servidores Proxy Reflejados y Almacenados** (**Enhancing International World Wide Web Access in Mozambique Through the Use of Mirroring and Caching Proxies**). En este artículo (disponible en línea en *http://www.isoc.org/inet97/ans97/cloet.htm*) los autores describen cómo trabaja el proceso:

> "*Un proceso automatizado recupera la página inicial del sitio y especifica el número de páginas extra (siguiendo recursivamente los enlaces HTML en las páginas recuperadas) a través del uso de un proxy. En lugar de copiar las páginas recuperadas en el disco local, el proceso de duplicación descarta las páginas recuperadas. Esto se hace para conservar los recursos del sistema así como para evitar posibles problemas de copyright. Mediante el uso del proxy como intermediario, se garantiza que las páginas recuperadas están en el cache del proxy como si un cliente hubiera accedido a esa página. Cuando un cliente accede a la página recuperada, le es brindada desde el cache y no desde el enlace internacional congestionado. Este proceso puede ser corrido en momentos de poco uso de la red, para maximizar la utilización del ancho de banda y no competir con otras actividades de acceso.*"

El siguiente comando (programado para correr en la noche, o una vez al día o a la semana) es todo lo que se necesita (debe repetirse para cada sitio que necesita ser pre-poblado).

```
wget --proxy-on --cache=off --delete after -m http://www.python.org
```

Explicación:

- **-m**: Duplica el sitio completo. wget comienza en *www.python.org* y sigue todos los hiperenlaces, es decir que descarga todas las subpáginas.

- **--proxy-on**: Se asegura que wget haga uso del servidor *proxy*. Esto puede no necesitarse en aplicaciones donde se utiliza un servidor *proxy* transparente.
- **--cache=off**: Se asegura de que el contenido fresco es recuperado desde Internet, y no desde el servidor *proxy* local.
- **--delete after**: Borra la copia duplicada. El contenido duplicado permanece en el *cache* del *proxy* si hay suficiente espacio en el disco, y los parámetros del servidor *proxy* son aplicados correctamente.

Además, wget tiene muchas otras opciones; por ejemplo, proveer contraseñas para los sitios web que las requieren. Cuando utilizamos esta herramienta, Squid debe ser configurado con suficiente espacio en el disco para que contenga todos los sitios pre-poblados y más (para un uso normal de Squid que involucre otras páginas además de las pre-pobladas). Afortunadamente, el espacio de disco es cada vez más barato y su tamaño mucho más grande que nunca. Sin embargo, esta técnica puede ser utilizada solo con unos pocos sitios seleccionados. Estos sitios no deben ser muy grandes para que los procesos terminen antes de que las horas del día de trabajo comiencen, y se debe estar vigilando el espacio de disco disponible.

Jerarquías de memoria temporal (*cache*)

Cuando una organización tiene más de un servidor *proxy*, los mismos pueden compartir información *cache* entre ellos. Por ejemplo, si una página web está en el *cache* del servidor A, pero no en el cache del servidor B, un usuario conectado a través del servidor B puede acceder a la página web en el servidor A a través del servidor B. El ***Protocolo de Inter-Cache*** (Inter-Cache Protocol (***ICP***)) y el (Cache Array Routing Protocol (***CARP***)) pueden compartir información del *cache*. De éstos, el protocolo CARP es considerado el mejor. Squid soporta ambos protocolos, y el Servidor MS ISA soporta CARP. Para más información diríjase a: *http://squid-docs.sourceforge.net/latest/html/c2075.html*. El compartir información *cache* reduce el uso de ancho de banda en organizaciones donde se utiliza más de un *proxy*.

Especificaciones Proxy

En la red de un campus universitario, debería haber más de un servidor *proxy*, por razones de prestaciones y de redundancia. Con los discos actuales más baratos y más grandes, se pueden construir servidores *proxy* más poderosos, con 50 GB o más de espacio de disco asignado al *cache*. Las prestaciones del disco son importantes, por lo que los discos SCSI más rápidos se van a desempeñar mejor (aunque un *cache* basado en un IDE es mejor que nada). RAID (Redundant Array of Independent Disks) o el uso de espejos (mirror) no son recomendados.

Se aconseja dedicar un disco exclusivamente para el cache. Por ejemplo, un disco puede ser para el *cache*, y el segundo para el sistema operativo y la bitácora del *cache*. Squid está diseñado para utilizar toda la memoria RAM que puede conseguir porque es mucho más rápido cuando los datos son recuperados desde la memoria RAM que cuando vienen desde el disco duro. Para una red en un campus, la memoria RAM debe ser de 1GB o más: Además de la memoria requerida para el sistema operativo y otras aplicaciones, Squid requiere 10 MB de RAM por cada 1 GB de disco *cache*. Por lo tanto, si tenemos un espacio de disco de 50 GB asignados al *cache*, Squid va a requerir 500 MB de memoria extra.

La máquina también va a requerir 128 MB para Linux y 128 MB para X-windows. Otros 256 MB deben agregarse para otras aplicaciones, y para que todo pueda funcionar fácilmente. Nada mejora más el rendimiento de una computadora como la instalación de una gran cantidad de memoria, porque esto reduce la necesidad de utilizar el disco duro. La memoria es miles de veces más rápida que el disco duro. Los sistemas operativos modernos frecuentemente mantienen los datos accedidos en la memoria siempre que haya suficiente RAM disponible. Pero utilizan el archivo de la página del disco duro como un área de memoria extra cuando no tienen suficiente memoria RAM.

Almacenamiento intermedio (*cache*) y optimización de DNS

Los servidores DNS con sólo la función de *cache* no son autoridades de ningún dominio, solo almacenan los resultados de solicitudes pedidas por los clientes, tal como un servidor *proxy* que almacena páginas web populares por cierto tiempo. Las direcciones DNS son almacenadas hasta que su ***tiempo de vida*** (***TTL*** *por su sigla en inglés*) expira. Esto va a reducir la cantidad de tráfico DNS en su conexión a Internet, porque el *cache* DNS puede ser capaz de satisfacer muchas de las preguntas localmente. Por supuesto que las computadoras de los clientes deben ser configuradas para utilizar el nombre del servidor solo de *cache* como su servidor DNS. Cuando todos los clientes utilicen ese servidor DNS como su servidor principal, se poblará rápidamente el *cache* de direcciones IP a nombres, por lo tanto los nombres solicitados previamente pueden ser resueltos rápidamente. Los servidores DNS que son autoridades para un dominio también actúan como cache de la conversión nombres-direcciones de hosts de ese dominio.

Bind (named)

Bind es el programa estándar de facto utilizado para servicios de nombre en Internet. Cuando Bind está instalado y corriendo, va a actuar como un servidor cache (no se necesita más configuración). Bind puede ser instalado

desde un paquete como el Debian o un RPM. Instalarlo desde un paquete en general es el mejor método. En Debian, escriba

```
apt-get install bind9
```

Además de implementar *cache*, Bind también puede alojar zonas de autoridad, actuar como esclavo de zonas de autoridad, implementar *split horizon* (horizonte dividido), y todo lo demás que es posible con DNS.

dnsmasq

Un servidor DNS de cache alternativo es ***dnsmasq***. Está disponible para BSD y la mayoría de las distribuciones Linux, o desde *http://freshmeat.net/projects/dnsmasq/*. La gran ventaja de dnsmasq es la flexibilidad: actúa como un proxy DNS de *cache* y como una fuente autorizada para hosts y dominios, sin una configuración complicada de archivos de zona. Se pueden hacer actualizaciones a la zona de datos sin ni siquiera reiniciar el servicio. También actúa como servidor DHCP, e integra el servicio DNS con el de DHCP. Es liviano, estable y extremadamente flexible. Bind es, prácticamente, la mejor elección para redes muy grandes (mayores que un par de cientos de nodos), pero la simplicidad y flexibilidad de dnsmasq lo hacen atractivo para redes pequeñas y medianas.

Windows NT

Para instalar el servicio DNS en Windows NT4: seleccione Panel de Control ➜ Red ➜ Servicios ➜ Agregar ➜ Servidor DNS Microsoft. Inserte el CD de Windows NT4 CD cuando se le indique. Cómo configurar un servidor solo de memoria intermedia (*cache*) en NT se describe en el artículoKnowledge Base 167234. Una cita del artículo:

> "*Simplemente instale DNS y haga correr el Sistema Administrador de Nombres de Dominio (Domain Name System Manager). Dé un clic en DNS en el menú, seleccione Nuevo Servidor, y escriba la dirección IP de su computadora donde ha instalado DNS. Usted ahora tiene un servidor DNS solo de cache.*"

Windows 2000

Para instalar el servicio DNS: Inicio ➜Configuración ➜ Panel de Control ➜ Agregar o Quitar Programas. En Agregar o Quitar Componentes de Windows, seleccione Componentes ➜ Servicios de Red ➜ Detalles ➜ Sistema de Nombres de Dominios (DNS). Luego inicie el DNS MMC (Inicio ➜ Programas ➜ Herramientas Administrativas ➜ DNS) Desde el menú de Acción seleccione "Conectarse a la Computadora..." En la ventana de Selección de Computadora Destino, habilite "La siguiente computadora:" e ingrese el

nombre del servidor DNS que usted quiere almacenar. Si hay un . [punto] en el administrador DNS (aparece por omisión), significa que el servidor DNS piensa que es el servidor DNS raíz de Internet. Ciertamente no lo es. Para que todo funcione borre el . [punto].

DNS dividido y un servidor duplicado

El objetivo de un DNS dividido (también conocido como ***horizonte dividido***) es el de presentar una visión diferente de su dominio para el mundo interno y el externo. Hay más de una forma de dividir DNS; pero por razones de seguridad se recomienda que tenga dos servidores de contenidos DNS separados; el interno y el externo (cada uno con bases de datos diferentes).

Dividir el DNS permite a los clientes de la red del campus resolver las direcciones IP para el dominio del campus a direcciones locales RFC1918, mientras que el resto de Internet resuelve los mismos nombres a direcciones IP diferentes. Esto se logra teniendo dos zonas en dos servidores DNS diferentes para el mismo dominio.

Una de las zonas es utilizada para los clientes internos de la red y la otra para los usuarios en Internet. Por ejemplo, en la red siguiente el usuario dentro del campus de Makerere verá *http://www.makerere.ac.ug/* resuelto como 172.16.16.21, mientras que un usuario en otro dominio de Internet lo verá resuelto como 195.171.16.13.

El servidor DNS en el campus, como se ve en el diagrama anterior, tiene un archivo de zona para *makerere.ac.ug* y está configurado como la autoridad para ese dominio. Además, funciona como el servidor DNS cache para el campus de Makerere, y todas las computadoras en el campus están configuradas para utilizarlo como su servidor DNS.

Los registros DNS para el servidor DNS en el campus van a verse así:

```
makerere.ac.ug
www     CNAME   webserver.makerere.ac.ug
ftp     CNAME   ftpserver.makerere.ac.ug
mail    CNAME   exchange.makerere.ac.ug
mailserver      A       172.16.16.21
webserver       A       172.16.16.21
ftpserver       A       172.16.16.21
```

Pero hay otro servidor DNS en Internet que es en realidad la autoridad para el dominio *makerere.ac.ug*. Los registros DNS para esta zona externa van a verse así:

```
makerere.ac.ug
www     A 195.171.16.13
ftp     A 195.171.16.13
mail    A 16.132.33.21
MX mail.makerere.ac.ug
```

El DNS dividido no depende de la utilización de direcciones RFC 1918. Un ISP africano puede, por ejemplo, alojar sitios web en representación de una universidad pero también puede duplicar esos mismos sitios web en Europa. Siempre que los clientes de ese ISP acceden al sitio web, éste toma la dirección IP del ISP africano, y por lo tanto el tráfico permanece en el mismo país. Cuando visitantes de otros países acceden al sitio web, reciben la dirección IP del sitio web duplicado en el servidor en Europa. De esta forma los visitantes internacionales no congestionan la conexión VSAT del ISP cuando visitan el sitio web de la universidad. Esto se está convirtiendo en una solución atractiva, ya que el alojamiento web cerca del backbone de Internet se está haciendo muy económico.

Optimización del enlace a Internet

Como mencionamos anteriormente, se pueden alcanzar rendimientos superiores a 22Mbps mediante la utilización de equipamiento 802.11g estándar para redes inalámbricas. Este valor de ancho de banda probablemente sea al menos un orden de magnitud mayor que la que le ofrece su enlace a Internet, y es capaz de soportar cómodamente muchos usuarios simultáneos de Internet.

Pero si su conexión principal a Internet es a través de un enlace VSAT, se va a encontrar con algunos problemas de desempeño si utiliza los parámetros por omisión de TCP/IP. Optimizando su enlace VSAT, se pueden mejorar significativamente los tiempos de respuesta cuando se accede a hosts de Internet.

Factores TCP/IP en una conexión por satélite

Un VSAT es concebido a menudo como una tubería de datos ***larga y gruesa***. Este término se refiere a los factores que afectan el desempeño de TCP/IP en cualquier red que tenga un ancho de banda relativamente grande, pero mucha latencia. La mayoría de las conexiones a Internet en África y otras partes del mundo en desarrollo son vía VSAT. Por lo tanto, aún si una universidad tiene su conexión a través de un ISP, esta sección puede ser aplicable si la conexión del ISP es a través de VSAT. La alta latencia en las

redes por satélite se debe a la gran distancia del satélite y la velocidad constante de la luz. Esta distancia añade aproximadamente 520 ms al tiempo de ida y retorno de un paquete (RTT –round trip time– *por su sigla en inglés*), comparado con un RTT entre Europa y Estados Unidos de alrededor de 140 ms.

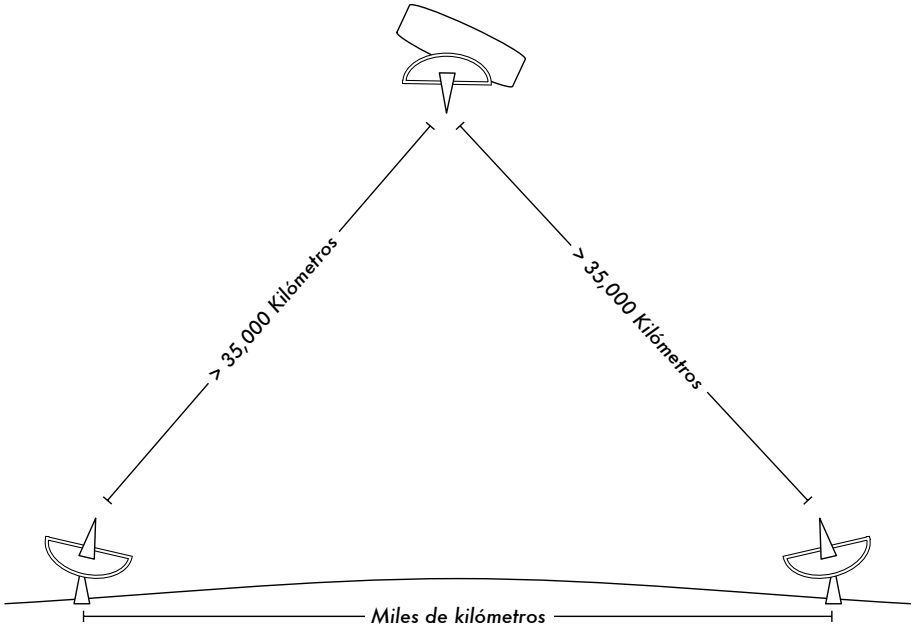

Figura 3.17: Debido a la velocidad de la luz y las largas distancias involucradas, la confirmación de recepción de un paquete ping puede tomar más de 520ms en un enlace VSAT.

Los factores que impactan más significativamente el rendimiento de TCP/IP son tiempos de propagación largos, **grandes productos de ancho de banda por retardo y errores de transmisión**.

Generalmente en una red satelital se deben utilizar sistemas operativos que soportan las implementaciones TCP/IP modernas. Estas implementaciones soportan las extensiones RFC 1323:

- La opción de ***escalado de ventana*** para soportar ventanas TCP de gran tamaño (mayores que 64KB).
- ***Recepción selectiva*** (***SACK*** *por su sigla en inglés*) para permitir una recuperación más rápida de los errores de transmisión.
- Matasellos (*Timestamps*) para calcular los valores de RTT y la expiración del tiempo de retransmisión para el enlace en uso.

Tiempos de ida y vuelta largos (RTT)

Los enlaces por satélite tienen un promedio de RTT de alrededor de 520ms hasta el primer salto. TCP utiliza el mecanismo de comienzo lento al inicio de la conexión para encontrar los parámetros de TCP/IP apropiados para la misma. El tiempo perdido en la etapa de comienzo lento es proporcional al RTT, y para los enlaces por satélite significa que TCP se encuentra en el modo de comienzo lento por más tiempo de lo que debiera. Esto disminuye drásticamente el rendimiento de las conexiones TCP de corta duración. Esto puede verse cuando descargar un sitio web pequeño sorprendentemente toma mucho tiempo, mientras que cuando se transfiere un archivo grande se obtienen velocidades de datos aceptables luego de un rato.

Además cuando se pierden paquetes, TCP entra en la fase de control de congestión y, debido al alto RTT permanece en esta fase por largo tiempo, reduciendo así el rendimiento de las conexiones TCP, sean de larga o corta duración.

Producto ancho de banda-retardo elevado

La cantidad de datos en tránsito en un enlace en un momento dado es el producto del ancho de banda por el RTT. Debido a la gran latencia del enlace satelital, este producto es grande. TCP/IP le permite a los hosts remotos enviar cierta cantidad de datos previamente sin esperar la confirmación (*acknowledgment*). Normalmente en una conexión TCP/IP se requiere una confirmación (ACK) para cada transmisión. Sin embargo el host remoto siempre puede enviar cierta cantidad de datos sin confirmación, lo que es importante para lograr una buena tasa de transferencia en conexiones con productos ancho de banda-retardo de propagación elevados. Esta cantidad de datos es denominada ***tamaño de la ventana TCP***. En las implementaciones TCP/IP modernas el tamaño de la ventana generalmente es de 64KB.

En las redes satelitales, el valor del producto ancho de banda-retardo es importante. Para utilizar el enlace en toda su capacidad, el tamaño de la ventana de la conexión debe ser igual al producto del ancho de banda-retardo. Si el tamaño de ventana máximo permitido es de 64KB, teóricamente el máximo rendimiento que se puede conseguir vía satélite es (tamaño de la ventana) / RTT, o 64KB / 520 ms. Esto da una tasa de transferencia de datos máxima de 123kB/s, correspondiente a 984 kbps, aunque la capacidad del enlace sea mucho mayor.

Cada encabezado de segmento TCP contiene un campo llamado ventana anunciada, que especifica cuantos bytes de datos adicionales está preparado para aceptar el receptor. La ***ventana anunciada*** es el tamaño actual de la memoria de almacenamiento intermedio del receptor. El emisor no está autorizado a enviar más bytes que la ventana anunciada. Para maximizar el

rendimiento, las memorias de almacenamiento intermedio del emisor y el receptor deben ser al menos iguales al producto ancho de banda-retardo. El tamaño de la memoria de almacenamiento intermedio en la mayoría de las implementaciones modernas de TCP/IP tiene un valor máximo de 64KB.

Para soslayar el problema de versiones de TCP/IP que no exceden el tamaño de la ventana de 64KB, se puede utilizar una técnica conocida como suplantación de confirmación (***TCP acknowledgment spoofing***) (vea más adelante Mejora del Rendimiento del Proxy).

Errores de transmisión

En las implementaciones de TCP/IP más viejas, siempre se consideraba que la pérdida de paquetes era causada por la congestión (en lugar de errores de enlace). Cuando esto sucede TCP adopta una defensiva contra la congestión, requiriendo tres confirmaciones duplicadas (ACK), o ejecutando un inicio lento (*slow start*) en el caso de que el tiempo de espera haya expirado.

Debido al alto valor de RTT, una vez que esta fase de control de la congestión ha comenzado, toma un largo rato para que el enlace satelital TCP/IP vuelva al nivel de rendimiento anterior. Por consiguiente, los errores en un enlace satelital tienen un efecto más serio en las prestaciones de TCP que sobre los enlaces de latencia baja. Para solucionar esta limitación, se han desarrollado mecanismos como la ***Confirmación Selectiva*** (***SACK*** por su sigla en inglés). SACK especifica exactamente aquellos paquetes que se han recibido permitiendo que el emisor retransmita solamente aquellos segmentos que se perdieron debido a errores de enlace.

El artículo sobre detalles de implementación de TCP/IP en Windows 2000 afirma:

> "*Windows 2000 introduce soporte para una importante característica de desempeño conocida como Confirmación Selectiva (SACK). SACK es especialmente importante para conexiones que utilizan ventanas TCP de gran tamaño.*"

SACK ha sido una característica estándar desde hace algún tiempo en Linux y BSD. Asegúrese de que tanto su enrutador Internet como el ISP del sitio remoto soporten SACK.

Implicaciones para las universidades

Si un sitio tiene una conexión a Internet de 512 kbps, las configuraciones por omisión de TCP/IP son suficientes, porque una ventana de 64 KB puede cubrir hasta 984 kbps. Pero si la universidad tiene más de 984 Kbps, es probable que en algunos casos no se obtenga todo el ancho de banda disponible

del enlace debido a los factores de "tubería de datos larga y gruesa" discutidos anteriormente. Lo que estos factores implican realmente es que impiden que una computadora tome todo el ancho de banda. Esto no es malo durante el día, porque mucha gente está usando el ancho de banda. Pero si por ejemplo, se programan grandes descargas para la noche, el administrador puede querer hacer uso de todo el ancho de banda, y los factores de "tubería de datos larga y gruesa" pueden ser un obstáculo. Esto puede transformarse en algo crítico si una cantidad significativa de su tráfico de red se enruta a través de un túnel único o una conexión VPN hasta el otro extremo del enlace VSAT.

Los administradores pueden considerar tomar algunas medidas para asegurarse de que están aprovechando la totalidad del ancho de banda disponible, afinando las configuraciones de TCP/IP. Si una universidad ha implementado una red donde el tráfico tiene necesariamente que pasar a través de un *proxy* (impuesto por el diseño de red), entonces las únicas computadoras que pueden realizar conexiones directas a Internet serán los servidores *proxy* y de correo electrónico.

Para más información, vea: *http://www.psc.edu/networking/perf_tune.html*.

Proxy que mejora las prestaciones (PEP- Performance enhancing Proxy)

La idea de PEP se describe en la RFC 3135 (vea *http://www.ietf.org/rfc/rfc3135*), y podría ser un servidor Proxy con un disco cache grande que tiene extensiones RFC 1323, entre otras características. Una computadora portátil tiene una sesión TCP con PEP en el ISP. Ese PEP, y el que está en el proveedor de satélite se comunican utilizando diferentes sesiones TCP, inclusive, su propio protocolo privado. El PEP del proveedor de satélite toma los archivos desde el servidor web. De esta forma, la sesión TCP se divide y por lo tanto se evitan las características del enlace que afectan las prestaciones del protocolo (los factores de tubería larga y gruesa), utilizando por ejemplo suplantación de confirmaciones TCP (*TCP ACK spoofing*). Adicionalmente, PEP reactúa como proxy y realiza captura previa (*pre-fetching*) para acelerar todavía más el acceso a la web.

Este sistema puede ser construido desde cero utilizando por ejemplo Squid, o adquiriendo soluciones ofrecidas por varios vendedores.