

Introducción a IGRP

Contenido

[Introducción](#)

[Objetivos del IGRP](#)

[El problema del ruteo](#)

[Resumen de IGRP](#)

[Comparación con RIP](#)

[Descripción detallada](#)

[Descripción completa](#)

[Características de estabilidad](#)

[Deshabilitar retenciones](#)

[Detalles del proceso de actualización](#)

[Ruteo de Paquetes](#)

[Recepción de actualizaciones de ruteo](#)

[Procesamiento periódico](#)

[Genere mensajes de actualización](#)

[Información de cálculo de métrica](#)

[Detalles de la implementación de IP](#)

[Solicitudes](#)

[Actualizaciones](#)

[Cómputos métricos](#)

[Información Relacionada](#)

Introducción

Este documento presenta el IGRP (Interior Gateway Routing Protocol). Tiene dos propósitos. Uno es presentar una introducción a la tecnología IGRP, para los que estén interesados en usarla, evaluarla y posiblemente implementarla. El otro es ofrecer una exposición más amplia de algunas ideas y conceptos interesantes que se incluyen en IGRP. Consulte Configuración de IGRP, Implementación del IGRP de Cisco y Comandos IGRP para obtener información sobre cómo configurar IGRP.

Objetivos del IGRP

El protocolo IGRP permite que varios gateways coordinen su routing. Las metas son las siguientes:

- Ruteo estable aún en redes muy grandes o complejas. No deben ocurrir bucles de routing, ni siquiera transitorios.
- Respuesta rápida a los cambios en la topología de la red

- Tara baja. Es decir, IGRP en sí no debe usar más banda ancha de la que necesita realmente para realizar su tarea.
- La división de tráfico entre varias rutas paralelas cuando son de conveniencia apenas similar.
- Tener en cuenta las tasas de errores y el nivel de tráfico en distintos trayectos.

La implementación actual de IGRP maneja el ruteo para TCP/IP. A pesar de esto, el diseño básico se creó para poder admitir una variedad de protocolos.

No hay una sola herramienta que resuelva todos los problemas de routing. Convencionalmente el problema de ruteo se desglosa en varias partes. Los protocolos como IGRP se denominan "protocolos de gateway internos" (internal gateway protocol, IGP). Su propósito es utilizarse dentro de un único grupo de redes, tanto bajo una sola administración como en administraciones muy coordinadas. Estos conjuntos de redes se encuentran conectados mediante "protocolos de gateway externa" (EGP). Un IGP está diseñado para hacer un seguimiento detallado de la topología de una red. A la hora de diseñar un IGP, se le da prioridad a la producción de rutas óptimas y a la rápida respuesta ante los cambios. Se espera que un EGP proteja a un sistema de redes contra errores o contra una distorsión intencional por parte de otros sistemas, el BGP es uno de estos protocolos de gateway exterior. La prioridad en la designación de un EGP está en la estabilidad y en los controles administrativos. A menudo es suficiente para que un EGP produzca una ruta razonable, en lugar de una ruta óptima.

IGRP tiene algunos aspectos similares a otros protocolos más antiguos como el Protocolo de información de ruteo de Xerox, el RIP de Berkeley y el Hello de Dave Mills. Difiere de estos protocolos principalmente en que están diseñados para redes más extensas y más complejas. Consulte la sección Comparación con RIP, para obtener una comparación más detallada con RIP, que es el protocolo más utilizado de la generación más antigua de protocolos.

Como estos protocolos anteriores, IGRP es un protocolo del vector distancia. Con un protocolo de este tipo, los gateways intercambian información de routing únicamente con los gateways adyacentes. Esta información de routing contiene un resumen de la información sobre el resto de la red. Se puede demostrar en términos matemáticos la medida en que todos los gateways en conjunto resuelven un problema de optimización en un algoritmo distribuido. Cada gateway sólo debe resolver parte del problema y sólo debe recibir una porción del total de los datos.

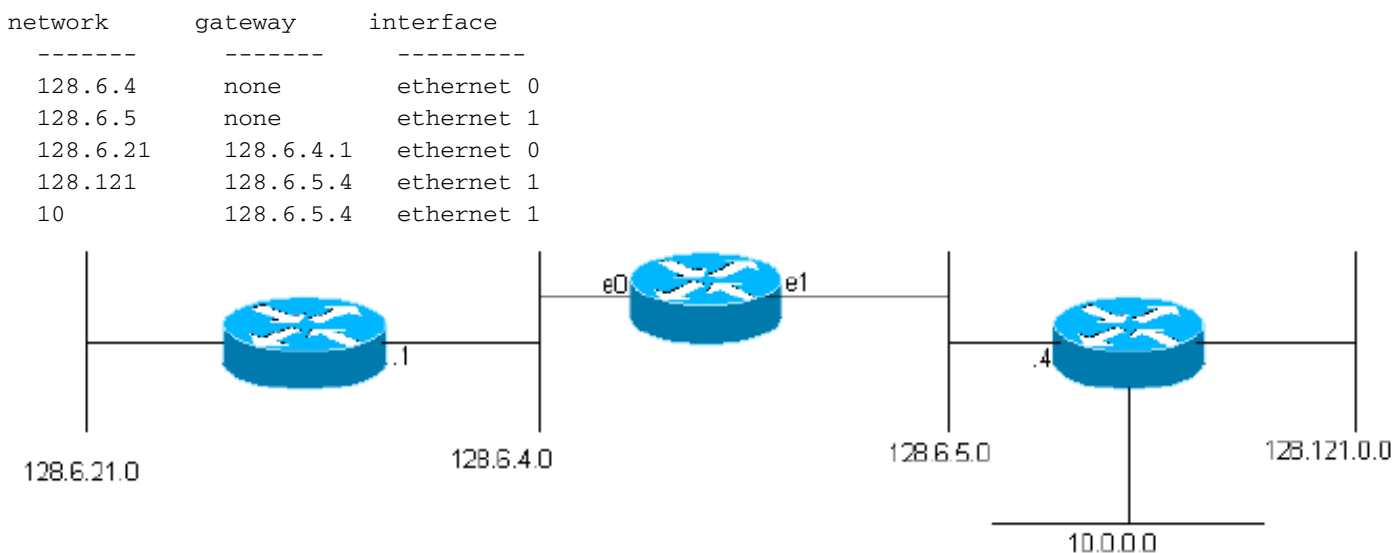
La alternativa principal para IGRP es IGRP mejorada (EIGRP) y una clase de algoritmos referidos como SPF (trayecto más corto primero). El protocolo OSPF emplea este concepto. Para obtener más información sobre el protocolo OSPF, consulte la [Guía de Diseño de OSPF](#). El protocolo OSPF se basa en una técnica de saturación, por la cual cada gateway se mantiene actualizado sobre el estado de cada interfaz en los otros gateways. Cada gateway soluciona independientemente el problema de optimización desde su punto de vista mediante los datos para toda la red. Cada método tiene sus ventajas. En algunas circunstancias, el SPF podrá responder a los cambios con mayor rapidez. Para evitar los loops de ruteo, IGRP debe ignorar la información nueva por algunos minutos después de que se producen ciertos tipos de cambios. Debido a que SPF recibe información directamente de cada gateway, puede evitar estos loops de ruteo. Por lo tanto puede actuar sobre información nueva en forma inmediata. Sin embargo, el SPF debe procesar considerablemente más datos que el IGRP, tanto en estructuras internas de datos como en mensajes entre gateways.

[El problema del ruteo](#)

IGRP está destinado para usarse en gateways que conectan varias redes. Suponemos que las redes utilizan tecnología basada en paquetes. En efecto, los gateways actúan como switches de

paquete. Cuando un sistema conectado a una red desea enviar un paquete a un sistema de una red diferente, dirige el paquete a una gateway. Si el destino está en una de las redes conectadas a la gateway, ésta enviará el paquete al destino. Si el destino está más alejado, el gateway reenviará el paquete a otro gateway que esté más cerca del destino. Los gateways utilizan tablas de routing a modo de ayuda para decidir qué hacer con los paquetes. El siguiente es un ejemplo simple de una tabla de routing. Las direcciones que se utilizan en los ejemplos son direcciones IP de Rutgers University. Tenga en cuenta que el problema básico de ruteo es similar para otros protocolos también, pero esta descripción supondrá que IGRP se usa para el ruteo de IP.)

Figure 1



(Como veremos más adelante, las tablas de routing IGRP reales tienen más información para cada gateway). Este gateway está conectado con dos redes Ethernet, llamadas 0 y 1. Se les asignaron los números de red IP (en realidad, números de subred) 128.6.4 y 128.6.5. Por lo tanto, los paquetes dirigidos a estas redes específicas se pueden enviar directamente al destino, simplemente mediante el uso de la interfaz Ethernet correspondiente. Hay dos gateways cercanas, 128.6.4.1 y 128.6.5.4. Los paquetes dirigidos a redes que no sean 128.6.4 ni 128.6.5 se reenviarán a alguno de los gateways. La tabla de routing indica el gateway que debe utilizarse para cada red. Por ejemplo, los paquetes direccionados a un host en la red 10 deberían ser reenviados a la gateway 128.6.5.4. Se espera que este gateway esté más cerca de la red 10; es decir, que la mejor ruta hacia la red 10 pase por este gateway. El objetivo principal de IGRP es permitir que los gateways generen y mantengan tablas de ruteo como esta.

Resumen de IGRP

Como se mencionó anteriormente, el protocolo IGRP permite que los gateways creen su propia tabla de routing mediante el intercambio de información con otros gateways. Una gateway comienza con entradas para todas las redes directamente conectadas con ella. Obtiene información sobre otras redes mediante el intercambio de actualizaciones de ruteo con gateways adyacentes. En el caso más simple, el gateway encontrará una ruta que será la mejor manera de llegar a cada red. Un trayecto se caracteriza por el siguiente gateway al que se deben enviar los paquetes, la interfaz de red que debe usarse y la información métrica. La información de métrica es una serie de números que indican la eficacia de la ruta. Esto le permite a la gateway comparar los trayectos que escuchó de varias gateways y decidir cuál utilizar. En algunos casos, puede ser conveniente dividir el tráfico en dos o más rutas. IGRP hará esto siempre que dos o más trayectos

sean igualmente buenos. El usuario también puede configurar la división del tráfico cuando las rutas tengan la misma eficacia. En este caso se enviará más tráfico a lo largo del trayecto con la mejor métrica. El motivo es que el tráfico puede dividirse entre una línea de 9600 BPS y una de 19200 BPS, y la línea de 19200 recibir aproximadamente el doble de tráfico que la línea de 9600 BPS.

Entre las métricas utilizadas por el protocolo IGRP se incluyen las siguientes:

- Tiempo de retardo topológico
- Ancho de banda del segmento de ancho de banda más angosto del trayecto
- Ocupación de canal de la ruta
- Confiabilidad del trayecto

Tiempo de demora topológico es el tiempo que se necesita para llegar al destino por esa ruta, si se supone que la red no está cargada. Por supuesto, existe un retardo adicional cuando la red está cargada. Sin embargo, la carga se contabiliza mediante el uso de la cifra de ocupación de canal y no al intentar medir las demoras reales. El ancho de banda del trayecto es simplemente el ancho de banda en bits por segundo del link más lento de la trayectoria. La ocupación de canal indica la cantidad de ancho de banda que se encuentra actualmente en uso. Este valor se mide y cambia con la carga. La confiabilidad indica el índice de errores actual. Es la fracción de paquetes que llegan al destino sin daños. Se mide.

Si bien no se usan como parte de la métrica, dos datos adicionales se transmiten junto con ella: conteo de saltos y MTU. El conteo de saltos es simplemente la cantidad de gateways que un paquete tendrá que atravesar para llegar a destino. MTU es el tamaño máximo de paquete que se puede enviar en todo el trayecto sin fragmentación. (Es decir, es la cantidad mínima de las MTU de todas las redes involucradas en el trayecto).

Sobre la base de la información de la métrica, se calcula una sola "métrica compuesta" para el trayecto. La métrica compuesta combina el efecto de los distintos componentes de la métrica en un solo número que representa la eficacia de esa ruta. La métrica compuesta es la que en realidad se utiliza para decidir cuál es la mejor ruta.

En forma periódica, cada gateway transmite toda su tabla de ruteo (con ciertas restricciones debido a la regla de horizonte dividido) hacia todas las gateways adyacentes. Cuando un gateway obtiene esta transmisión de otro gateway, compara la tabla con su tabla existente. Los destinos nuevos y las rutas nuevas se agregan a la tabla de routing del gateway. Las rutas en la transmisión se comparan con las rutas existentes. Si una ruta nueva es mejor, esta puede reemplazar la existente. La información en la difusión también se utiliza para actualizar la ocupación de canales y otra información sobre trayectos existentes. Este procedimiento general es similar al utilizado por todos los protocolos del vector de distancia. En la bibliografía de matemática, esto se conoce como el algoritmo de Bellman-Ford. Consulte [RFC 1058](#) para obtener un desarrollo detallado del procedimiento básico, que describe RIP, un protocolo de vector de distancia más antiguo.

En IGRP, el algoritmo Bellman-Ford general se modifica en tres aspectos importantes. En primer lugar, se utiliza un vector de métricas en vez de una métrica simple para caracterizar las rutas. En segundo lugar, en vez de elegir un solo trayecto con la medición más pequeña, el tráfico se divide en varios trayectos, cuyas mediciones se ajustan a un rango específico. En tercer lugar, se introducen varias funciones para brindar estabilidad en situaciones en las que la topología está cambiando.

Se selecciona el mejor trayecto sobre la base de una métrica compuesta:

$$[(K1 / Be) + (K2 * Dc)] r$$

Donde K1, K2 = constantes, Be = ancho de banda del trayecto descargado x (1 - ocupación del canal), Dc =(retardo topológico) y r = fiabilidad.

La ruta que contenga la menor métrica compuesta será la mejor ruta. Cuando hay varias rutas hacia el mismo destino, el gateway puede enviar los paquetes a través de más de una ruta. Esto se realiza de acuerdo con la métrica compuesta para cada trayecto de datos. Por ejemplo, si un trayecto tiene una métrica compuesta de 1 y otro trayecto tiene una métrica compuesta de 3, será enviado tanto como el triple de paquetes por el trayecto de datos que tiene la métrica compuesta de 1.

El uso de información de un vector de métricas tiene dos ventajas. La primera es que brinda la capacidad de admitir diversos tipos de servicio desde el mismo conjunto de datos. La segunda ventaja es una precisión mejorada. Cuando se utiliza una sola métrica, se lo suele tratar como si fuera una demora. Cada enlace en la ruta se agrega a la métrica total. Si hay un enlace con un ancho de banda reducido, suele representarse con una demora grande. Sin embargo, las limitaciones en el ancho de banda no se acumulan de la forma en que lo hacen las demoras. Como el ancho de banda se considera un componente aparte, se puede administrar sin inconvenientes. De manera similar, la carga puede ser administrada por un número de ocupación de canal separado.

El protocolo IGRP ofrece un sistema para interconectar redes de computadoras que pueden, con estabilidad, administrar una topología gráfica general, con bucles incluidos. El sistema mantiene información de las métricas de las rutas completas; es decir, conoce los parámetros de las rutas a todas las otras redes a las que se conecta cualquier gateway. El tráfico puede distribuirse sobre los trayectos paralelos y pueden computarse simultáneamente los parámetros de trayecto múltiple por toda la red.

Comparación con RIP

En esta sección, se comparan los protocolos IGRP y RIP. Esta comparación es útil ya que RIP se usa con frecuencia con propósitos similares a IGRP. Sin embargo, si se hace esto no es totalmente justo. El protocolo RIP no fue creado para cumplir con los mismos objetivos que el protocolo IGRP. El protocolo RIP se creó para ser usado en redes pequeñas con tecnología razonablemente uniforme. En dichas aplicaciones es generalmente adecuado.

La diferencia más básica entre IGRP y RIP es la estructura de sus métricas. Lamentablemente, este cambio no se puede retroadaptar al RIP. RIP requiere los algoritmos nuevos y las estructuras de datos presentes en IGRP.

RIP usa una métrica simple de "recuento de saltos" para describir la red. En el caso del IGRP, cada ruta se describe con una demora, un ancho de banda, etc.; en el caso del RIP, cada ruta se describe con un número del 1 al 15. Normalmente, este número se utiliza para representar la cantidad de gateways por los que pasa la ruta hasta llegar al destino. Esto significa que no se distingue entre una línea serial lenta y una Ethernet. En algunas implementaciones del RIP, es posible que el administrador del sistema especifique que un salto determinado se contabilice en más de una oportunidad. Las redes lentas pueden estar representadas por un gran conteo de saltos. Pero, debido a que el máximo es 15, esto no se puede hacer muchas veces. Por ejemplo, si una red Ethernet se representa con 1 y una línea de 56 KB se representa con 3, puede haber como máximo 5 líneas de 56 KB en una ruta; de lo contrario, se excederá el máximo de 15. De acuerdo con estudios realizados por Cisco, a fin de representar todo el rango de velocidades de

red disponibles, se necesita una métrica de 24 bits. Si la métrica máxima es demasiado pequeña, el administrador del sistema tendrá que tomar una decisión poco agradable: no puede distinguir entre rutas rápidas y lentas, o bien no puede aceptar su red completa dentro del límite. De hecho, algunas redes nacionales ahora son tan grandes que RIP no puede administrarlas, ni siquiera si cada salto se contabiliza solo una vez. El RIP (Protocolo de información de ruteo) simplemente no puede usarse para tales redes.

La respuesta obvia sería que se modifique el RIP para permitir una métrica más amplia. Lamentablemente, esto no funcionará. Al igual que todos los protocolos de vector de distancia, RIP tiene el problema de “contar hasta el infinito”. Esto se describe con más detalle en [RFC 1058](#). Cuando cambie la topología, se introducirán rutas falsas. Las métricas asociadas con estas rutas falsas aumentan lentamente hasta que llegan a 15, punto en el cual las rutas se eliminan. 15 es una cantidad máxima suficientemente pequeña que este proceso convergirá con bastante rapidez, suponiendo que se utilicen las actualizaciones activadas. Si el RIP se modificara y permitiera una métrica de 24 bits, los bucles permanecerían el tiempo suficiente para que la métrica se contabilice hasta 2^{24} . Esto no se puede tolerar. IGRP tiene funciones diseñadas para evitar la introducción de rutas falsas. Se analizan a continuación, en la sección 5.2. No es práctico administrar redes complejas sin introducir tales características o sin adoptar otro protocolo, como SPF.

IGRP hace algo más que simplemente incrementar el rango de métricas admitidas. Reestructura la métrica para describir el retardo, el ancho de banda, la confiabilidad y la carga. Tales consideraciones pueden representarse en una sola métrica como RIP. Sin embargo, el enfoque aplicado por IGRP es potencialmente más exacto. Por ejemplo, con una sola métrica, varios enlaces rápidos sucesivos parecerán ser equivalentes a un único enlace lento. Este podría ser el caso del tráfico interactivo, en el que la demora es la principal preocupación. Sin embargo, para transferencia masiva de datos, la principal preocupación es el ancho de banda y, en este caso, agregar métricas en forma conjunta no es el enfoque correcto. IGRP maneja la demora y el ancho de banda por separado, acumulando las demoras y tomando el mínimo de los anchos de banda. No es fácil ver cómo incorporar los efectos de confiabilidad y carga en una métrica de un solo componente.

En mi opinión, una de las principales ventajas del IGRP es la facilidad de la configuración. Puede representar directamente cantidades que tienen significado físico. Esto significa que se puede configurar automáticamente, en función del tipo de interfaz, la velocidad de línea, etc. Con una métrica de un solo componente, es más probable que la métrica tenga que ser “cocida” para incorporar efectos de varias cosas diferentes.

Otras innovaciones están más vinculadas con algoritmos y estructuras de datos que con el protocolo de ruteo. Por ejemplo, IGRP especifica algoritmos y estructuras de datos que soportan la división del tráfico entre varias rutas. Sin dudas, es posible diseñar una implementación de RIP que haga esto. Sin embargo, una vez que el ruteo está siendo reimplementado, no hay motivo para quedarse con RIP.

Hasta el momento, he descrito un protocolo “IGRP genérico”, una tecnología que podría admitir el routing de cualquier protocolo de red. No obstante, en esta sección cabe mencionar un poco más acerca de la implementación específica de TCP/IP. Ésa es la implementación que se comparará con RIP.

Los mensajes de actualización de RIP simplemente contienen instantáneas de la tabla de ruteo. Es decir, tienen una cantidad de destinos y valores de métrica y un poco más. La implementación IP del IGRP tiene estructuras adicionales. En primer lugar, el mensaje de actualización se identifica con un “número de sistema autónomo”. Esta terminología procede de la

tradición de Arpanet y, en ese ámbito, tiene un significado específico. Sin embargo, para la mayoría de las redes, significa que es posible ejecutar distintos sistemas de ruteo en la misma red. Esto es útil para los lugares en los que convergen las redes de varias organizaciones. Cada organización puede mantener su propio ruteo. Dado que cada actualización es etiquetada, es posible configurar que las gateways presten atención sólo a la correcta. Ciertas puertas de enlace están configuradas para recibir actualizaciones desde varios sistemas independientes. Éstas pasan información entre los sistemas de manera controlada. Tenga en cuenta que ésta no es una solución completa para los problemas de seguridad de ruteo. Todos los gateways se pueden configurar para que escuchen las actualizaciones de cualquier sistema autónomo. Sin embargo, todavía es una herramienta muy útil a la hora de implementar políticas de ruteo cuando hay un cierto grado de confianza entre los administradores de red.

La segunda función estructural acerca de los mensajes de actualización de IGRP afecta el modo en que IGRP maneja las rutas predeterminadas. La mayoría de los protocolos de ruteo poseen un concepto de ruta predeterminada. En general, no es práctico que en las actualizaciones de routing se enumeren todas las redes del mundo. Por lo general un conjunto de gateways necesitan información de ruteo detallada para las redes dentro de su organización. Todo el tráfico que tenga un destino afuera de su organización se puede enviar a uno de los gateways perimetrales. Esas gateways delimitadoras quizás tengan información más completa. La ruta al mejor gateway es una "ruta predeterminada". Es predeterminada porque se utiliza para llegar a cualquier destino que no se encuentra específicamente enumerado en las actualizaciones de routing interno. RIP, y algunos otros protocolos de routing, transmiten la información sobre la ruta predeterminada como si se tratara de una red real. IGRP aplica un enfoque diferente. En lugar de una sola entrada falsa para la ruta predeterminada, IGRP permite que las redes reales se marquen como candidatas para convertirse en una predeterminada. Esto se implementa colocando la información sobre esas redes en una sección exterior especial del mensaje de actualización. Sin embargo, también podría pensarse como la activación de un bit asociado con esas redes. Periódicamente IGRP busca todas las rutas predeterminadas de los candidatos y elige la de métrica inferior como la ruta predeterminada real.

Potencialmente, esta metodología de las rutas predeterminadas es de algún modo más flexible que aquella adoptada por la mayoría de las implementaciones RIP. Más comúnmente se pueden configurar los gateways RIP de modo que generen una ruta predeterminada con una determinada métrica especificada. La intención es que debería hacerse en las gateways de límite.

[Descripción detallada](#)

En esta sección, se proporciona una descripción detallada del protocolo IGRP.

[Descripción completa](#)

Al activar una gateway por primera vez, se inicializa su tabla de ruteo. Esto puede ser llevado a cabo por un operador desde una terminal de consola o mediante la lectura de la información en los archivos de configuración. Se provee una descripción de cada red conectada a la gateway, que incluye el retraso topológico a lo largo del link (por ejemplo, cuánto tiempo tarda un solo bit en atravesar el link) y el ancho de banda del link.

Figure 2

$$[(K1 / Be) + (K2 * Dc)] r$$

Donde r = confiabilidad fraccionaria (% de las transmisiones que se reciben correctamente en el siguiente salto); Dc = demora compuesta, Be = ancho de banda efectivo: ancho de banda sin carga x (1 - ocupación de canal); K1 y K2 = constantes.

Ecuación 2

En principio, el retraso compuesto, Dc, podría determinarse como se muestra a continuación:

$$Dc = Ds + Dcir + Dt$$

Donde Ds = retardo de conmutación, Dcir = retardo de circuito (retardo de propagación de 1 bit), y Dt = retardo de transmisión (retardo sin carga para un mensaje de 1500 bits).

Sin embargo, en la práctica, una cifra de demora estándar se utiliza para cada tipo de tecnología de red. Por ejemplo, existirá una cifra de retardo estándar para Ethernet y para las líneas seriales a cualquier velocidad en bits determinada.

Aquí se brinda un ejemplo de cómo podría verse la tabla de ruteo del gateway A en el caso del diagrama de red 6 presentado anteriormente. (Observe que no se muestran los componentes individuales del vector métrico por razones de simplicidad.)

Ejemplo de tabla de routing:

Red	Interfaz	Siguiente gateway	Métrico
1	NW 1	Ninguno	Conexión directa
2	NW 2	Ninguno	Conexión directa
3	NW 3	Ninguno	Conexión directa
4	NW 2	C	1270
	NW 3	B	1180
5	NW 2	C	1270
	NW 3	B	2130
6	NW 2	C	2040
	NW 3	B	1180

El proceso básico de creación de una tabla de ruteo mediante el intercambio de información con los vecinos se describe por medio del algoritmo de Bellman-Ford. El algoritmo se utilizó en protocolos anteriores, como RIP (RFC 1058). Para manejar redes más complejas, IGRP agrega tres funciones al algoritmo básico Bellman-Ford:

1. Para caracterizar trayectos, se usa un vector de métricas en vez de una métrica simple. Según la Ecuación 1, que se muestra más arriba, desde este vector se puede computar una métrica compuesta única. El uso de un vector permite que el gateway admita diferentes tipos de servicio mediante el uso de varios coeficientes diferentes en la ecuación 1. Además, permite una representación más exacta de las características de la red que una sola métrica.

2. En vez de elegir un solo trayecto con la medición más pequeña, el tráfico se divide en varios trayectos, cuyas mediciones se ajustan a un rango específico. Esto permite que varias rutas se usen en paralelo, lo que proporciona un ancho de banda efectivo mayor que cualquier ruta individual. El administrador de red especifica una V diferente. Todos los trayectos con la mínima métrica compuesta M se mantienen. Además, se conservan todos los trayectos cuya métrica es inferior a $V \times M$. El tráfico se distribuye entre varias rutas de manera inversamente proporcional a las métricas compuestas.
3. Existen algunos problemas con este concepto de varianza. Es difícil pensar en estrategias que usen valores de variación superiores a 1 y que no producen, a la vez, bucles de paquetes. En la Versión 8.2 de Cisco, no se implementa la característica de varianza. (No estoy seguro de la versión en que se eliminó la función). Como consecuencia, la variación se establece de forma permanente en 1.
4. Se han incorporado varias características a fin de proporcionar estabilidad en situaciones en las que la topología está cambiando. Estas funciones han sido creadas para evitar que se formen bucles de routing y "contar hasta el infinito", ambas características de intentos anteriores de utilizar algoritmos de tipo Ford para este tipo de aplicación. Las funciones de estabilidad primarias son "retenciones", "actualizaciones activadas", "horizontes divididos" y "envenenamiento". A continuación, se abordarán estas funciones en mayor detalle.

La división del tráfico (punto 2) plantea un riesgo bastante sutil. La variación V está diseñada para permitir que las gateways usen rutas paralelas con diferentes velocidades. Por ejemplo, puede haber una línea de 9600 BPS ejecutándose paralelamente con una línea de 19200 BPS para la redundancia. Si la variación V es 1, se usará solo la mejor ruta. Por lo tanto, la línea 9600 BPS no se utilizará si la línea 19200 BPS tiene una confiabilidad razonable. (Sin embargo, si varias rutas son iguales, la carga se distribuirá entre ellas). Al aumentar la variación, el tráfico se puede dividir entre la mejor ruta y otras rutas que tienen casi la misma eficacia. Con una variación suficientemente grande, el tráfico se dividirá entre dos líneas. El peligro es que en caso de una variación lo suficientemente grande, las rutas permitidas no sólo son más lentas sino que también tienen una "dirección equivocada". Por lo tanto, debe existir una regla adicional para evitar que el tráfico se envíe "en sentido ascendente": No se envía tráfico junto a aquellos trayectos cuya métrica compuesta remota (la métrica compuesta calculada en el siguiente salto) es mayor que la métrica compuesta calculada en el gateway. Por lo general, se recomienda a los administradores de sistemas que no fijen la variación sobre 1, excepto en situaciones específicas donde sea necesario usar trayectos paralelos. En este caso, la varianza es configurada cuidadosamente para proveer los resultados "correctos".

IGRP está diseñado para gestionar varios "tipos de servicios" y varios protocolos. Tipo de servicio es una especificación en un paquete de datos que modifica el modo en que se evaluarán las rutas. Por ejemplo, el protocolo TCP/IP le permite al paquete especificar la importancia relativa del ancho de banda alto, del retraso bajo o de la confiabilidad alta. Generalmente, las aplicaciones inactivas especificarán un retraso bajo, mientras que las aplicaciones de transferencia masiva especificarán un ancho de banda alto. Estos requisitos determinan los valores relativos de $K1$ y $K2$ que son apropiados para usarlos en Eq. 1. Cada combinación de especificaciones en el paquete que ha de admitirse se denomina "tipo de servicio". Para cada tipo de servicio debe elegirse un conjunto de parámetros $K1$ y $K2$. Se guarda una tabla de ruteo para cada tipo de servicio. Esto es así porque los trayectos son seleccionados y ordenados de acuerdo con la métrica compuesta definida por Eq. 1. Esto es diferente para cada tipo de servicio. La información de todas estas tablas de ruteo se combina para producir los mensajes de actualización de ruteo que intercambian las gateways, como lo describe la figura 7.

[Características de estabilidad](#)

Esta sección describe las retenciones, las actualizaciones activadas, el horizonte dividido y el envenenamiento. Estas funciones se han diseñado para evitar que los gateways elijan rutas erróneas. Como se describe en [RFC 1058](#), esto puede ocurrir cuando una ruta se vuelve inutilizable, debido a una falla de una gateway o una red. En principio, los gateways adyacentes detectan las fallas. Luego, envían actualizaciones de ruteo que indican que la antigua ruta no se puede utilizar. Sin embargo, es posible que las actualizaciones no lleguen en absoluto a determinadas partes de la red o se demoren en llegar a ciertos gateways. Una gateway que aún cree que la ruta vieja es buena puede continuar repartiendo esa información y, de esta forma, reintroduciendo a la ruta fallida dentro del sistema. Por último, esta información se propagará por la red y regresará al gateway que la volvió a insertar. El resultado es una ruta circular.

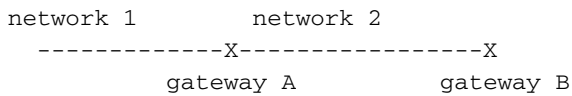
De hecho, existe cierta redundancia entre las contramedidas. Básicamente, las retenciones y las actualizaciones activadas deberían ser suficientes para evitar rutas erróneas en primer lugar. No obstante, en la práctica, diferentes errores de comunicación pueden hacer que resulten insuficientes. Las técnicas de horizonte dividido y envenenamiento de ruta han sido creadas para que no se formen bucles de routing en ningún caso.

Normalmente, las tablas de routing nuevas se envían a los gateways vecinos con regularidad (cada 90 segundos de manera predeterminada, aunque el administrador del sistema puede ajustar la frecuencia). Una actualización disparada es una nueva tabla de ruteo que se envía inmediatamente en respuesta a algún cambio. El cambio más importante es la eliminación de una ruta. Esto puede suceder porque ha transcurrido un tiempo de espera (probablemente, se ha desactivado una línea o un gateway vecino) o porque un mensaje de actualización del siguiente gateway en la ruta muestra que la ruta ya no se puede utilizar. Cuando una gateway G detecta que una ruta ya no se puede utilizar, genera inmediatamente una actualización. Esta actualización mostrará que la ruta es inutilizable. Considere lo que sucede cuando esta actualización llega a las gateways vecinas. Si la ruta del vecino apuntaba a G, el vecino debe quitar la ruta. Esto hace que el vecino active una actualización, etc. Por lo tanto, una falla desencadenará una oleada de mensajes de actualización. Esta oleada se propagará en toda la parte de la red en la que las rutas atravesaron el gateway o la red con falla.

Las actualizaciones disparadas serían suficientes si pudiéramos garantizar que la ola de actualizaciones ha alcanzado cada gateway apropiada de manera inmediata. Sin embargo, hay dos problemas. En primer lugar, algún enlace en la red puede descartar o dañar los paquetes que contienen el mensaje de actualización. Segundo, las actualizaciones provocadas no suceden de manera instantánea. Es posible que una gateway cuya actualización no haya sido activada emita una actualización normal en el tiempo incorrecto, haciendo que se vuelva a insertar la ruta incorrecta en un vecino cuya actualización ya ha sido activada. Las retenciones están diseñadas para solucionar estos problemas. La regla de retención indica que cuando se remueve una ruta, por algún tiempo no se aceptará ninguna ruta nueva para el mismo destino. Esto le otorga el tiempo necesario a las actualizaciones activadas para llegar a todos los otros gateways, para asegurarse de que cada nueva ruta que obtengamos no sea simplemente algún gateway reinsertando la antigua. El período de espera (holddown) debe ser lo suficientemente largo para permitir que la oleada de actualizaciones activadas atraviese toda la red. Además, debería incluir un par de ciclos de difusión para administrar paquetes caídos. Pensemos en qué sucede si una de las actualizaciones activadas se descarta o se daña. el gateway que envió la actualización enviará otra actualización cuando corresponda. Esto reiniciará la onda de las actualizaciones activadas en los vecinos que perdieron la onda inicial.

La combinación de actualizaciones activadas y tiempos de espera debe ser suficiente para deshacerse de rutas que caducaron y evitar que se vuelvan a insertar. Sin embargo, de todos modos, vale la pena adoptar algunas medidas de precaución adicionales. Permiten redes con mucha pérdida de información y redes que tienen particiones. Las precauciones adicionales que

requiere IGRP son horizonte dividido y envenenamiento de ruta. El horizonte dividido surge de la observación de que no tiene sentido hacer que una ruta vuelva en la dirección por la cual vino. Considere la siguiente situación:



El gateway A le indicará al gateway B que tiene una ruta a la red 1. Cuando el gateway B envía actualizaciones al gateway A, no hay ninguna razón para que mencione la red 1. Debido a que el gateway A está más cerca de la red 1, no hay ninguna razón para que considere dirigirse a través del gateway B. La regla de horizonte dividido establece que se debe generar un mensaje de actualización individual para cada vecino (en realidad, para cada red vecina). La actualización de un cierto vecino debería omitir rutas que apuntan a ese vecino. Con esta regla, se evita que se formen bucles entre gateways adyacentes. El ejemplo supone que A es una interfaz para fallas de la red 1. Sin la regla de horizonte dividido, el gateway B le indicaría al gateway A que puede llegar a la red 1. Como ya no tiene una ruta real, es posible que el gateway A tome esa ruta. En este caso, tanto el gateway A como el gateway B tienen rutas a la red 1. Pero A señalaría B y B señalaría A. Por supuesto, las actualizaciones activadas y las retenciones deberían evitar que esto sucediera. Sin embargo, debido a que no existe ninguna razón para enviar la información al lugar de dónde provino, de todos modos es recomendable dividir el horizonte. Además de su función en la prevención de loops, la división del horizonte mantiene el tamaño de los mensajes de actualización en un nivel bajo.

El horizonte dividido previene los loops entre gateways adyacentes. La técnica de envenenamiento de ruta ha sido creada para interrumpir bucles más grandes. La regla es que cuando una actualización muestra que la medición de una ruta existente se ha incrementado lo suficiente, hay un loop. La ruta deberá quitarse y colocarse en retención. Actualmente, la regla establece la eliminación de una ruta si la métrica compuesta aumenta en más de un factor de 1.1. No es seguro que cualquier aumento en la métrica compuesta active la eliminación de la ruta, ya que pueden ocurrir pequeños cambios en la métrica debido a cambios en la ocupación de canal o en la confiabilidad. Por lo tanto, el factor de 1.1 es sólo un heurístico. El valor exacto no es crítico. Esperamos que esta regla se necesite solo para interrumpir bucles de gran tamaño, ya que las actualizaciones activadas y los tiempos de espera se ocuparán de impedir los bucles pequeños.

[Deshabilitar retenciones](#)

A partir de la versión 8.2, el código de Cisco tienen una opción para inhabilitar la retención. La desventaja de las retenciones es que retrasan la adopción de una nueva ruta cuando falla una ruta antigua. Con los parámetros predeterminados, puede tardar varios minutos hasta que el router adopte una nueva ruta luego de una carga. Sin embargo, por los motivos explicados anteriormente, no es seguro simplemente quitar las retenciones. El resultado sería una cuenta hasta el infinito, como se describe en RFC 1058. Suponemos, aunque no podemos comprobarlo, que con una versión más sólida de la técnica de envenenamiento de ruta, los tiempos de espera dejarán de ser necesarios para detener la cuenta hasta el infinito. De esta forma, la inhabilitación de las retenciones da lugar a esta forma más fuerte de envenenamiento de ruta. Observe que las actualizaciones activadas y de división del horizonte siguen vigentes.

La manera más severa de envenenamiento de ruta está basada en un conteo de saltos. Si la cuenta de saltos para un trayecto aumenta, se elimina la ruta. Esto eliminará, obviamente, las rutas que sigan siendo válidas. Si algo más de la red cambia de manera que el trayecto ahora atraviesa una o más puertas de enlace, se incrementará el recuento de saltos. En este ejemplo, la ruta continúa siendo válida. Sin embargo, no existe manera alguna que sea completamente

segura de distinguir este caso de los loops de ruteo (cuenta a infinito) Por lo tanto, el enfoque más seguro es eliminar la ruta cada vez que aumenta el conteo de saltos. Si la ruta aún es legítima, será reinstalada por la actualización siguiente, y eso causará una actualización activada que reinstalará la ruta en otro lugar dentro del sistema.

En general, los algoritmos de vector de distancia¹ adoptan rutas nuevas con facilidad. El problema es borrar completamente los anteriores del sistema. De esta manera, una norma que es sumamente agresiva acerca de las rutas sospechosas que se eliminan, debería ser segura.

Detalles del proceso de actualización

El conjunto de procesos descrito en las Figuras 4 a 8 están diseñados para manejar un protocolo de red simple, por ejemplo, TCP/IP, DECnet o el protocolo ISO/OSI. Sin embargo, los detalles del protocolo se darán sólo para TCP/IP. Un único gateway puede procesar datos que obedecen a más de un protocolo. Dado que cada protocolo tiene diferentes estructuras de direcciones y formatos de paquetes, el código informático que se utiliza para implementar las figuras 4 a 8 será, por lo general, diferente para cada protocolo. El proceso que se describe en la figura 4 será el que más variará, como se describe en las notas detalladas de la figura 4. Los procesos que se describen en la figuras 5 a 8 tendrán la misma estructura general. La diferencia principal entre un protocolo y otro será el formato del paquete de actualización de ruteo, el cual debe ser diseñado para que sea compatible con un protocolo específico.

Observe que la definición de un destino puede variar de protocolo a protocolo. El método descrito puede utilizarse para el ruteo a hosts individuales, a redes o para esquemas de direcciones jerárquicas más complejos. El tipo de ruteo usado dependerá de la estructura de direccionamiento del protocolo. La implementación de TCP/IP actual sólo admite el ruteo a redes IP. Por lo tanto, “destino” en realidad significa número de subred o red IP. La información de subred sólo se mantiene para las redes conectadas.

Las figuras de la 4 a la 7 muestran el pseudocódigo para varias partes del proceso de ruteo utilizado por las gateways. Al inicio del programa, se ingresan parámetros y protocolos aceptables que describen cada interfaz.

El gateway solo administrará ciertos protocolos que se enumeran. Toda comunicación desde un sistema mediante un protocolo que no se encuentre en la lista será ignorada. Estas son las entradas de datos:

- Las redes a las cuales se conecta el gateway.
- Ancho de banda descargado de cada red.
- Retardo topológico de cada red.
- Confiabilidad de cada red.
- Ocupación de canal de cada red.
- MTU de cada red.

La función de métrica de cada ruta de datos luego se calcula de acuerdo con la ecuación 1. Tenga en cuenta que los primeros tres elementos son razonablemente permanentes. Son una función de la tecnología de red subyacente y no dependen de que la carga. Se las puede configurar desde un archivo de configuración o por entrada directa del operador. Tenga en cuenta que IGRP no utiliza el retraso medido. Tanto la teoría como la experiencia sugieren que es muy difícil para los protocolos que usan retraso medido mantener un ruteo estable. Existen dos parámetros medidos: confiabilidad y ocupación de canal. La confiabilidad se basa en las tasas de errores sobre los que informan por el hardware o firmware de las interfaces de red.

Además de estas entradas, el algoritmo de ruteo requiere un valor para varios parámetros de ruteo. Esto incluye valores del temporizador, variación y si se habilitan los tiempos de espera. Normalmente esto se especificaría mediante un archivo de configuración o un operador de entrada. (A partir de la versión Cisco 8.2, la varianza está configurada en 1 de manera permanente).

Una vez que se introduce la información inicial, las operaciones en el gateway son activadas por los eventos: la llegada de un paquete de datos en una de las interfaces de red o la caducidad de un temporizador. Los procesos descritos en las figuras 4 a 7 se accionan de la siguiente manera:

- Cuando llega un paquete, este se procesa según se muestra en la figura 4. Como resultado, el paquete se envía a otra interfaz, se descarta o se acepta para seguir procesándolo.
- Cuando el gateway acepta un paquete para su procesamiento posterior, se analiza de una forma propia del protocolo que no se describe en esta especificación. Si el paquete es una actualización del ruteo, se procesa de acuerdo con la Figura 5.
- En la figura 6, se muestra los eventos activados por un temporizador. El temporizador se establece para generar una interrupción una vez por segundo. Cuando ocurre una interrupción, se ejecuta el proceso mostrado en la Figura 6.
- En la figura 7, se muestra una subrutina de actualización de routing. Las llamadas a esta subrutina se muestran en las Figuras 5 y 6.
- Además, la Figura 8 muestra detalles de cálculos de métricas a los que hacen referencia las Figuras 5 y 7.

Hay cuatro constantes de tiempo fundamentales que controlan la propagación y la caducidad de las rutas. Un administrador del sistema puede establecer estas constantes de tiempo. Sin embargo, existen valores predeterminados. Dichas constantes de tiempo son:

- Tiempo de transmisión: todos los gateways transmiten las actualizaciones en todas las interfaces conectadas con la frecuencia aquí establecida. Lo predeterminado es una vez cada 90 segundos.
- Tiempo hasta determinar que la ruta perdió validez: si durante el tiempo aquí establecido no se recibe ninguna actualización para una ruta determinada, se considera que ha caducado. Debe ser varias veces el tiempo de transmisión para permitir que la red pueda descartar los paquetes que contienen una actualización. El valor predeterminado es el triple del tiempo de difusión.
- Tiempo de espera: cuando un destino se vuelve inalcanzable (o la métrica ha aumentado lo suficiente como para causar envenenamiento), el destino entra en el estado “en espera”. Durante este estado y el tiempo aquí establecido, no se aceptará ninguna ruta nueva hacia el mismo destino. El período de inactividad indica cuánto debe durar este estado. Debería ser varias veces el tiempo de transmisión. El valor predeterminado es el triple del tiempo de difusión más 10 segundos. (Tal como se describe en la [sección Deshabilitar retenciones, es posible deshabilitar las retenciones.](#))
- Tiempo de eliminación: si durante el tiempo aquí establecido no se recibe ninguna actualización para un destino determinado, la entrada correspondiente se elimina de la tabla de routing. Observe la diferencia entre el tiempo no válido y el tiempo de purga: Un trayecto se elimina y remueve luego del tiempo inválido. Si ya no quedan trayectos hacia un destino, el destino está ahora fuera de alcance. Sin embargo, la entrada de base de datos para el destino permanece. Debe permanecer para imponer el tiempo de retención. Luego del momento de purgar, la entrada de la base de datos se elimina de la tabla. Debe ser más largo que el tiempo no válido más el tiempo de retención. El tiempo predeterminado es de 7 veces

el tiempo de transmisión.

Estas figuras presuponen las siguientes estructuras de datos principales. Un conjunto aparte de estas estructuras de datos se mantienen para cada protocolo suportado por la gateway. Dentro de cada protocolo, se mantiene un conjunto distinto de estructuras de datos para cada tipo de servicio que se debe admitir.

Para cada destino conocido en el sistema, existe una lista (posiblemente nula) de trayectos hacia el destino, un vencimiento de retención y un último plazo de actualización. La fecha de la última actualización es la última vez en que se incluyó un trayecto para este destino en una actualización desde otra gateway. Tenga en cuenta que también existen tiempos de actualización para cada trayecto. Cuando se elimina el último trayecto hacia un destino, se desactiva ese destino, a menos que esté inhabilitada la opción de desactivar (consulte la sección Inhabilitar la función desactivar para obtener más información). El tiempo de vencimiento de retención indica el tiempo en el cual expira la retención. El hecho de que no sea cero indica que el destino está en espera. A fin de ahorrar tiempo de cálculo, también es una buena idea mantener una "mejor métrica" para cada destino. Esto es simplemente el mínimo de la métrica compuesta para todos los trayectos dirigidos al destino.

Para cada trayecto a un destino están la dirección del próximo salto en el trayecto, la interfaz a ser utilizada, un vector de mediciones con las características del trayecto, el cual incluye el retraso topológico, el ancho de banda, la confiabilidad y la ocupación del canal. También se asocian otros datos con cada ruta, incluidos el conteo de saltos, la MTU, la fuente de información, la métrica compuesta remota y la métrica compuesta calculada a partir de estos números de acuerdo con la ecuación 1. También hay un tiempo de última actualización. La fuente de información indica de dónde vino la actualización más reciente para ese trayecto. En la práctica, es igual a la dirección del siguiente salto. El último plazo de actualización es simplemente el tiempo en el cual la actualización más reciente llegó para este trayecto. Se utiliza para dar por finalizados trayectos con el tiempo de espera agotado.

Observe que un mensaje de actualización IGRP consta de tres partes: interior, sistema (que significa "este sistema autónomo" pero no interior) y exterior. La sección interna es para rutas a subredes. No se incluye toda la información de subred. Se incluyen solo subredes de una red. Es la red asociada con la dirección a la cual se envía la actualización. Normalmente, las actualizaciones se difunden en cada interfaz, por lo tanto, se trata simplemente de la red por medio de la cual se envía la difusión. (Otros casos surgen a partir de las respuestas a una solicitud IGRP e IGRP punto a punto). Las redes principales (por ejemplo, no subredes) se colocan en la porción del sistema del mensaje de actualización, a menos que se marquen específicamente como exteriores.

Una red se marcará como exterior si se detectó de otro gateway y la información llegó a la porción exterior del mensaje de actualización. La implementación de Cisco también permite que el administrador del sistema declare redes específicas como exteriores. Las rutas exteriores se denominan también "candidatas predeterminadas". Son rutas que atraviesan o se dirigen a gateways que se consideran adecuados como predeterminados que deben utilizarse cuando no hay una ruta explícita a un destino. Por ejemplo, en Rutgers configuramos la gateway que conecta Rutgers a nuestra red regional para que marque la ruta a la estructura básica NSFnet como externa. La implementación de Cisco elige una ruta predeterminada al seleccionar la ruta exterior con la métrica menor.

Las siguientes secciones se incluyen con el fin de aclarar ciertas porciones de las figuras 4 a 8.

[Ruteo de Paquetes](#)

La Figura 4 describe el procesamiento integral de los paquetes de entrada. Esto se utiliza simplemente para clarificar la terminología. Obviamente, ésta no es una descripción completa de lo que hace una gateway IP.

Este proceso usa la lista de protocolos soportados y la información acerca de las interfaces ingresada cuando la gateway se inicializa. Los detalles del procesamiento de paquetes dependen del protocolo que utilice el paquete. Esto se determina en el Paso A. El paso A es la única parte de la Figura 4 que comparten todos los protocolos. Una vez que se conoce el tipo de protocolo, se utiliza la implementación de la figura 4 adecuada para el tipo de protocolo. El detalle del contenido de los paquetes se describe mediante las especificaciones del protocolo. Las especificaciones de un protocolo incluyen un procedimiento para determinar el destino de un paquete, un procedimiento para comparar el destino con las direcciones de la misma gateway para determinar si la gateway misma es el destino, un procedimiento para determinar si el paquete es una transmisión y un procedimiento para determinar si el destino es parte de una red especificada. Estos procedimientos se utilizan en los pasos B y C de la figura 4. La prueba en el paso D requiere una búsqueda de los destinos que figuran en la tabla de routing. La prueba se supera si existe una entrada en la tabla de routing para el destino y ese destino tiene asociada, como mínimo, una ruta utilizable. Tenga en cuenta que los datos de la ruta y el destino que se utilizan en este paso y en el siguiente se mantienen por separado para cada tipo de servicio admitido. Así, lo primero que hace este paso es determinar el tipo de servicio especificado por el paquete y seleccionar el conjunto de estructuras de datos correspondiente para usar en este y el siguiente paso.

Una ruta se puede utilizar con los fines de los pasos D y E si su métrica compuesta remota es menor que su métrica compuesta. Un trayecto cuya métrica remota compuesta es mayor que la métrica compuesta en un trayecto en que el próximo salto está más lejos del destino, según la medición. Esto se conoce como "ruta corriente arriba". Normalmente, se esperaría que el uso de mediciones prevenga el cierre de las rutas ascendentes. Es fácil entender que un trayecto ascendente nunca puede ser el mejor. Sin embargo, si se permite una variación importante, pueden utilizarse rutas que no sean la mejor. Algunas de ellas podrían ser corriente arriba.

El paso E computa la ruta a utilizar. No se consideran los trayectos cuya métrica compuesta remota no es menor que sus métricas compuestas. Si más de una ruta es aceptable, dichas rutas se utilizan en una forma ponderada de alternación por turnos (round-robin) La frecuencia con la cual un trayecto se usa es inversamente proporcional a su métrica compleja.

Recepción de actualizaciones de ruteo

En la figura 5, se describe el procesamiento de una actualización de routing recibida de un gateway vecino. Dichas actualizaciones constan de una lista de entradas, cada una de las cuales brinda información para un único destino. En una actualización de ruteo único, puede ocurrir que exista más de una entrada para el mismo destino con el fin de acomodar los tipos de servicios múltiples. Cada una de estas entradas se procesa de forma individual, como se describe en la figura 5. Si una entrada se encuentra en la sección exterior de la actualización, el indicador exterior se configurará para el destino si se lo agrega como resultado de este proceso.

El proceso completo descrito en la Figura 5 debe repetirse una vez para cada tipo de servicio admitido por el gateway, mediante la información del conjunto destino / trayecto asociado con ese tipo de servicio. Esto se muestra en el bucle de nivel más alejado, en la figura 5. La actualización de routing completa se debe procesar una vez para cada tipo de servicio. (Observe que la implementación actual de IGRP no es compatible con varios tipos de servicio de manera que el loop ultraperiférico en realidad no está implementado).

En el paso A se realizan pruebas de aceptabilidad básica en el trayecto. Esto debería incluir pruebas de razonabilidad para el destino. Los números de red imposibles ("Marcianos") deben rechazarse. (Consulte [RFC 1009](#) y [RFC 1122](#) para obtener más información.) Las actualizaciones también se rechazan si el destino al que hacen referencia está en espera, es decir, la fecha de caducidad de la espera es diferente de cero y posterior a la fecha actual.

En el paso B, se explora la tabla B para comprobar si esta entrada describe una ruta ya conocida. Una ruta en la tabla de routing se define por el destino con el que está asociada, el siguiente salto que figura como parte de la ruta, la interfaz de salida que se utilizará para la ruta y la fuente de información (la dirección de la cual provino la actualización —en la práctica, suele ser igual al siguiente salto). La entrada desde el paquete de actualización se describe como una ruta cuyo destino se enumera en la entrada, cuya interfaz de salida es la interfaz a la que llegó la actualización, y cuyo próximo salto y fuente de información será la dirección de la gateway que envió la actualización (la "fuente" S).

En los pasos H y T, está programado el proceso de actualización descrito en la Figura 7. Este proceso se ejecutará, en realidad, después de finalizar todo el proceso descrito en la figura 5. Es decir, el proceso de actualización que se describe en la figura 7 sucederá una sola vez, incluso si se activa varias veces durante el procesamiento descrito en la figura 5. Además, se deben tomar precauciones para evitar que las actualizaciones se ejecuten con demasiada frecuencia, si la red está cambiando rápidamente.

El paso K se realiza si el destino descrito por la entrada actual en el paquete de actualización ya existe en la tabla de routing. K compara la nueva métrica compuesta computada desde la información en el paquete de actualización con la mejor métrica compuesta para el destino. Observe que la mejor métrica compuesta no se vuelve a calcular en este momento, entonces, si el trayecto que se considera ya está en la tabla de ruteo, esta prueba puede comparar métricas nuevas y antiguas para el mismo trayecto.

El paso K se realiza para las rutas que son peores que la mejor métrica compuesta existente. Esto incluye tanto las rutas nuevas que son peores que las existentes y las rutas existentes cuya métrica compuesta ha aumentado. El paso L comprueba si el nuevo trayecto es aceptable. Tenga en cuenta que esta prueba implementa tanto la prueba de si una ruta nueva es lo suficientemente eficaz para mantenerla como el envenenamiento de ruta. Para que sea aceptable, el valor de retardo no debe ser el valor especial que indica un destino inalcanzable (para la implementación de la IP actual, todos ellos en un campo de bit 24) y la métrica compuesta (calculada como se especifica en la figura 8) debe ser admisible. Para determinar si la métrica compuesta es aceptable, compárela con las métricas compuestas del resto de las rutas hacia el destino. Use M como el valor mínimo. El nuevo trayecto es aceptable si es $< V \times M$, DONDE V ES LA VARIANCIÓN CONFIGURADA CUANDO SE INICIÓ LA GATEWAY. SI $V = 1$ (LO QUE SIEMPRE ES VERDADERO EN LA VERSIÓN 8.2 DE CISCO), ENTONCES UNA MÉTRICA PEOR QUE LA EXISTENTE NO ES ACEPTABLE. HAY UNA SOLA EXCEPCIÓN A ESTO: SI EL TRAYECTO TODAVÍA YA Existe Y ES EL ÚNICO HACIA EL DESTINO, SERÁ RETENIDO SI LA MEDICIÓN NO HA AUMENTADO MÁS DEL 10% (O EN AQUELLOS CASOS EN QUE LAS RETENCIONES SE HAN INHABILITADO, SI EL CONTEO DE SALTOS NO HA INCREMENTADO).

El paso V se realiza cuando la nueva información para un trayecto indica que la métrica compuesta disminuirá. Se comparan las métricas compuestas de todas las rutas hacia el destino D. En esta comparación, se utiliza la nueva métrica compuesta P en lugar de la que aparece en la tabla de ruteo. Se calcula la métrica compuesta mínima M. Luego todos los trayectos D se vuelven a examinar. Si la métrica compuesta para cualquier trayecto $> M \times V$, se elimina ese trayecto. V es la variación, ingresada cuando la gateway fue inicializada. (A partir de la versión Cisco 8.2, la varianza está configurada en 1 de manera permanente).

Procesamiento periódico

El proceso que se describe en la figura 6 se activa una vez por segundo. Examina varios temporizadores en la tabla de ruteo, para comprobar si alguno ha caducado. Estos temporizadores se describen más arriba.

En el Paso U, se activa el proceso descrito en la Figura 7.

Los pasos R y S son necesarios dado que la métrica compuesta almacenada en la tabla de ruteo depende de la ocupación del canal, que cambia en el tiempo, en función de las mediciones. La ocupación del canal se recalcula en forma periódica por medio de un promedio fluctuante de tráfico medido a través de la interfaz. Si el nuevo valor calculado difiere del existente, todas las métricas combinadas que conciernen a esa interfaz deben ser ajustadas. Se examinan todos los trayectos que se presentan en la tabla de ruteo. La métrica compuesta de cualquier ruta cuyo próximo salto use la interfaz "I" es recalculada. Esto se realiza de acuerdo con la Ecuación 1, usando como ocupación del canal el máximo del valor almacenado en la tabla de ruteo como parte de la métrica del trayecto y también, la ocupación del canal recientemente calculada de la interfaz.

Genere mensajes de actualización

La Figura 7 describe cómo el puerto de enlace genera mensajes de actualización para enviar a los otros puertos de enlace. Se genera un mensaje por separado para cada interfaz de red conectada al gateway. Ese mensaje es entonces enviado a otras gateways a las que se puede acceder a través de la interfaz (Paso J). Esto se realiza por lo general, enviando el mensaje como una transmisión. Sin embargo, si la tecnología o el protocolo de red no permiten la transmisión, puede que sea necesario enviar el mensaje a cada gateway individualmente.

En general, el mensaje se crea agregando una entrada para cada destino en la tabla de routing, en el paso G. Tenga en cuenta que se deben usar los datos de la ruta/el destino asociados con cada tipo de servicio. En el peor de los casos, se agrega una nueva entrada a la actualización para cada destino para cada tipo de servicio. Sin embargo, antes de agregar una entrada en el mensaje de actualización en el paso G, se escanean las entradas que ya se agregaron. Si la entrada nueva ya está presente en el mensaje de actualización, no se vuelve a agregar. Una entrada nueva duplica una existente cuando los destinos y los gateways de siguiente salto son los mismos.

A fin de brindar simplicidad, el seudocódigo omite una cosa —los mensajes de actualización de IGRP tienen tres partes: interior, sistema y exterior—, lo que significa que hay, en realidad, tres bucles a través de los destinos. El primero incluye solo subredes de la red a la cual se envía la actualización. El segundo incluye todas las redes principales (por ejemplo, no subredes) que no se han marcado como exteriores. El tercero incluye todas las redes principales que se han marcado como exteriores.

El paso E implementa la prueba de división del horizonte. En el caso normal, esta prueba falla con las rutas cuyo mejor trayecto sale de la misma interfaz por la que se está enviando la actualización. Sin embargo, si se envía la actualización a un destino específico (por ejemplo, en respuesta a un pedido de IGRP desde otra gateway o como parte de un "IGRP punto a punto"), la división del horizonte falla sólo si el mejor trayecto provenía originalmente de ese destino (su "fuente de información" es la misma que el destino) y su interfaz de salida es la misma que aquella de donde vino el pedido.

Información de cálculo de métrica

La Figura 8 describe cómo se procesa la información métrica a partir de los mensajes de actualización recibidos por la gateway, y cómo se genera para los mensajes de actualización enviados por la gateway. Observe que la entrada se basa en una ruta específica al destino. Si hay más de una ruta hacia el destino, se elige una ruta cuya métrica compuesta es mínima. Si más de un trayecto posee la métrica compuesta mínima, se utiliza una regla para desempatar arbitraria. (Para la mayoría de los protocolos, esto se basa en la dirección del próximo hop gateway.)

Figura 4: Procesamiento de paquetes entrantes

Data packet arrives using interface I

A Determine protocol used by packet

If protocol is not supported
then discard packet

B If destination address matches any of gateway's addresses
or the broadcast address
then process packet in protocol-specific way

C If destination is on a directly-connected network
then send packet direct to the destination, using
the encapsulation appropriate to the protocol and link type

D If there are no paths to the destination in the routing
table, or all paths are upstream
then send protocol-specific error message and discard the packet

E Choose the next path to use. If there are more than
one, alternate round-robin with frequency proportional
to inverse of composite metric.

Get next hop from path chosen in previous step.

Send packet to next hop, using encapsulation appropriate
to protocol and data link type.

Figura 5: Procesamiento de actualizaciones de routing entrantes

Routing update arrives from source S

For each type of service supported by gateway
Use routing data associated with this type of service

For each destination D shown in update

A If D is unacceptable or in holddown
then ignore this entry and continue loop with next destination D

B Compute metrics for path P to D via S (see Fig 8)

If destination D is not already in the routing table
then Begin

Add path P to the routing table, setting last
update times for P and D to current time.

```

H      Trigger an update

      Set composite metric for D and P to new composite
      metric computed in step B.

      End

Else begin (dest. D is already in routing table)

K      Compare the new composite metric for P with best
      existing metric for D.

      New > old:

L      If D is shown as unreachable in the update,
      or holddowns are enabled and
      the new composite metric >
      (the existing metric for D) * V
      [use 1.1 instead of V if V = 1,
      as it is as of Cisco release 8.2]
O      or holddowns are disabled and
      P has a new hop count > old hop count
      then Begin

      Remove P from routing table if present

      If P was the last route to D
      then Unless holddowns are disabled
      Set holddown time for D to
      current time + holddown time
T      and Trigger an update

      End

      else Begin

      Compute new best composite metric for D

      Put the new metric information into the
      entry for P in the routing table

      Add path P to the routing table if it
      was not present.

      Set last update times for P and D to
      current time.

      End

      New <= OLD:

V      Set composite metric for D and P to new
      composite metric computed in step B.

      If any other paths to D are now outside the
      variance, remove them.

      Put the new metric information into the
      entry for P in the routing table

      Set last update times for P and D to
      current time.

End

```

End of for

End of for

Figura 6: Procesamiento periódico

Process is activated by regular clock, e.g. once per second

For each path P in the routing table (except directly connected interfaces)

If current time < P'S LAST UPDATE TIME + INVALID TIME
THEN CONTINUE WITH THE NEXT PATH P

Remove P from routing table

If P was the last route to D
then Set metric for D to inaccessible
Unless holddowns are disabled,
Start holddown timer for D and
Trigger an update

else Recompute the best metric for D

End of for

For each destination D in the routing table

If D's metric is inaccessible
then Begin

Clear all paths to D

If current time >= D's last update time + flush time
then Remove entry for D

End

End of for

For each network interface I attached to the gateway

R Recompute channel occupancy and error rate

S If channel occupancy or error rate has changed,
 then recompute metrics

End of for

At intervals of broadcast time

U Trigger update

Figura 7: Generación de actualización

Process is caused by "trigger update"

For each network interface I attached to the gateway

Create empty update message

```

For each type of service S supported

    Use path/destination data for S

    For each destination D

E        If any paths to D have a next hop reached through I
          then continue with the next destination

          If any paths to D with minimal composite metric are
          already in the update message
          then continue with the next destination

G        Create an entry for D in the update message, using
          metric information from a path with minimal
          composite metric (see Fig. 8)

          End of for

    End of for

J        If there are any entries in the update message
          then send it out interface I

    End of for

```

Figura 8: Detalles sobre cálculos de métricas

En esta sección, se describe el procedimiento para calcular métricas y conteos de saltos de una actualización de routing que llega. La entrada para esta función es la entrada para un destino específico en un paquete de actualización de routing. El resultado es un vector de métricas que se puede utilizar para calcular la métrica compuesta y un conteo de saltos. Si se agrega este trayecto a la tabla de ruteo, se ingresa todo el vector de métrica en la tabla. Los parámetros de interfaz utilizados en las siguientes definiciones son los que se configuraron al inicializar la gateway, para la interfaz a la cual llegó la actualización de ruteo, excepto que la ocupación del canal y la confiabilidad están basadas en un promedio fluctuante de tráfico medido a través de la interfaz.

- Demora = demora del paquete + demora de interfaz topológica
- Ancho de banda = $\text{máx}(\text{ancho de banda de paquete}, \text{ancho de banda de interfaz})$
- Confiabilidad = $\text{mín}(\text{confiabilidad desde el paquete}, \text{confiabilidad de la interfaz})$
- Ocupación de canales = $\text{máx}(\text{ocupación de canales del paquete}, \text{ocupación de canales de la interfaz})$ (Se utiliza el ancho de banda máximo porque la métrica de ancho de banda se almacena de forma inversa. Conceptualmente, queremos el ancho de banda mínimo). Tenga en cuenta que la ocupación de canal original del paquete debe guardarse, ya que será necesaria para volver a calcular la ocupación de canal efectiva cada vez que cambie la ocupación de canal de la interfaz.

Los valores siguientes no forman parte del vector métrico, pero también se mantienen en la tabla de ruteo como características del trayecto:

- Conteo de saltos = conteo de saltos del paquete.
- MTU = $\text{mín}(\text{MTU de paquete}, \text{MTU de interfaz})$
- Métrica compuesta remota = se calcula a partir de la ecuación 1 utilizando los valores de métrica del paquete. Es decir, los componentes de la métrica son los del paquete y no están actualizados como se muestra arriba. Naturalmente, este valor debe calcularse antes de que se realicen los ajustes que se muestran más arriba.

- Métrica compuesta = calculada desde la ecuación 1 utilizando los valores métricos calculados según se describe en esta sección.

Esta parte restante de esta sección describe el procedimiento para calcular métricas y conteo de saltos para el envío de actualizaciones de ruteo.

Esta función determina la información de medición y el recuento de saltos que se introducirán en un paquete de actualización saliente. Se basa en una ruta específica hacia un destino, si hay alguna ruta utilizable. Si no existen trayectos, o los trayectos son todos ascendentes, el destino se denomina inaccesible.

If destination is inaccessible, this is indicated by using a specific value in the delay field. This value is chosen to be larger than the largest valid delay. For the IP implementation this is all ones in a 24-bit field.

If destination is directly reachable through one of the interfaces, use the delay, bandwidth, reliability, and channel occupancy of the interface. Set hop count to 0.

Otherwise, use the vector of metrics associated with the path in the routing table. Add one to the hop count from the path in the routing table.

Detalles de la implementación de IP

Esta sección describe brevemente los formatos de paquete que utiliza Cisco IGRP. IGRP se envía mediante el uso de datagramas IP con el protocolo IP 9 (IGP). El paquete comienza con un encabezado. Comienza inmediatamente después del encabezado IP.

```
unsigned version: 4; /* protocol version number */
  unsigned opcode: 4; /* opcode */
  uchar edition; /* edition number */
  ushort asystem; /* autonomous system number */
  ushort ninterior; /* number of subnets in local net */
  ushort nsystem; /* number of networks in AS */
  ushort nexterior; /* number of networks outside AS */
  ushort checksum; /* checksum of IGRP header and data */
```

Para los mensajes de actualización, la información de ruteo va inmediatamente después del encabezado.

El número de versión es actualmente 1. Los paquetes con otros números de versión se ignoran.

El opcode puede ser 1 = update o 2 = request

Esto indica el tipo de mensaje. El formato de los dos tipos de mensajes se brindará a continuación.

La edición es un número de serie que se incrementa toda vez que se produce un cambio en la tabla de ruteo. (Esto se realiza en condiciones en las que el seudocódigo anterior indica activar una actualización de routing). El número de edición permite que los gateways eviten el procesamiento de actualizaciones que contienen información que ya han visto. (Esto no está actualmente implementado. Es decir, el número de edición es generado correctamente, pero se ignora en la entrada. Dado que es posible que se pierdan paquetes, no queda claro si el número de edición es suficiente para evitar duplicar el proceso. Deberá verificar que todos los paquetes

asociados con la edición hayan sido procesados.

Asystem es el número de sistema autónomo. En la implementación de Cisco, un gateway puede participar en más de un sistema autónomo. Cada uno de esos sistemas ejecuta su propio protocolo IGRP. Conceptualmente, hay tablas de ruteo completamente separadas para cada sistema autónomo. Las rutas que llegan vía IGRP desde un sistema autónomo se envían solamente en actualizaciones para ese AS. Este campo permite que el gateway seleccione el conjunto de tablas de ruteo que utilizará para el procesamiento de este mensaje. Si la gateway recibe un mensaje IGRP de un AS para el cual no está configurado, se ignora. De hecho, la implementación de Cisco permite que la información se "fugue" de un AS a otro. Sin embargo, considero que eso es una herramienta administrativa y no una parte del protocolo.

Ninterior, nsystem y nexterior indican el número de entradas en cada una de las tres secciones de mensajes de actualización. Estas secciones se describieron anteriormente. No hay ninguna otra demarcación entre las secciones. Se toma a las primeras n entradas interiores como interiores, las próximas n entradas del sistema como del sistema y la n exterior final como exterior.

La suma de comprobación es una suma de comprobación de IP, computada usando el mismo algoritmo de suma de comprobación como un suma de comprobación de UDP. La suma de comprobación se calcula en el encabezado IGRP y en toda información de ruteo que le sigue. El campo checksum se establece en cero al calcular el valor checksum. El valor checksum no incluye el encabezado IP; tampoco hay ningún encabezado virtual, como en UDP y TCP.

Solicitudes

Una solicitud IGRP solicita al destinatario que envíe su tabla de routing. El mensaje de la solicitud tiene solo un encabezado. Se utilizan solamente los campos de versión, código de operación y sistema autónomo. Todos los otros campos se establecen en cero. Se espera que el receptor envíe un mensaje de actualización normal IGRP al solicitante.

Actualizaciones

Un mensaje de actualización IGRP contiene un encabezado, seguido inmediatamente por las entradas de routing. Se incluirá la cantidad de entradas de ruteo que quepa en un datagrama de 1500 bytes (incluido el encabezado IP). Con las declaraciones de la estructura actual, se permiten hasta 104 entradas. Si se necesitan más entradas, se envían diversos mensajes de actualización. Debido a que los mensajes de actualización son simplemente procesados entrada por entrada, no tiene ninguna ventaja utilizar un mensaje único fragmentado a varios independientes.

Esta es la estructura de una entrada de routing:

```
uchar number[3];          /* 3 significant octets of IP address */
  uchar delay[3];          /* delay, in tens of microseconds */
  uchar bandwidth[3];     /* bandwidth, in units of 1 Kbit/sec */
  uchar mtu[2];           /* MTU, in octets */
  uchar reliability;      /* percent packets successfully tx/rx */
  uchar load;             /* percent of channel occupied */
  uchar hopcount;        /* hop count */
```

Los campos definidos como uchar[2] y uchar[3] son meramente números enteros binarios de 16 y 24 bits, en un orden de red IP normal.

Number define el destino que se describe. Es una dirección IP. Para ahorrar espacio, se

proporcionan únicamente los 3 primeros bytes de la dirección IP, salvo en la sección interior. En la sección interior, se dan los 3 últimos bytes. Para las rutas de sistema y externas, no son posibles las subredes, por lo que el byte de orden bajo es siempre cero. Las rutas interiores son siempre subredes de una red conocida, por lo que se suministra el primer byte de ese número de red.

El retraso es en unidades de 10 microsegundos. Esto le proporciona un rango de 10 microsegundos a 168 segundos, que parece ser suficiente. Un retardo de todos unos indica que la red es inalcanzable.

El ancho de banda es ancho de banda inverso en bits por segundo ampliado por un factor de $1.0e10$. El rango comprende desde una línea de 1200 BPS a 10 Gbps. (Esto es, si el ancho de banda es N Kbps, el número utilizado es $10000000 / N$).

La MTU es en bytes.

La confiabilidad se indica como una fracción de 255. Es decir, 255 es 100%.

La carga está dada como una fracción de 255.

El conteo de saltos es un conteo simple.

Debido a las unidades extrañas utilizadas para el ancho de banda y retardo, algunos ejemplos parecen correctos. Estos son los valores predeterminados que se utilizan para varios medios comunes.

	Delay	Bandwidth
Satellite	200,000 (2 sec)	20 (500 Mbit)
Ethernet	100 (1 ms)	1,000
1.544 Mbit	2000 (20 ms)	6,476
64 Kbit	2000	156,250
56 Kbit	2000	178,571
10 Kbit	2000	1,000,000
1 Kbit	2000	10,000,000

Cómputos métricos

Aquí tiene una descripción de cómo se computa la métrica compuesta en la versión 8.0(3) de Cisco.

```
metric = [K1*bandwidth + (K2*bandwidth)/(256 - load) + K3*delay] *  
         [K5/(reliability + K4)]
```

If K5 == 0, the reliability term is not included.

The default version of IGRP has K1 == K3 == 1, K2 == K4 == K5 == 0

Información Relacionada

- [Página de Soporte de IP Routing](#)
- [Página de soporte de IGRP](#)
- [Soporte Técnico - Cisco Systems](#)