

## Capítulo 4

### CBT versión 2

Como se mencionó antes, CBT es un protocolo diseñado para operar en redes de área extensa, que en el futuro deberán soportar facilidades multicast<sup>19</sup> en sus routers y una diversidad y gran número de aplicaciones que harán uso de este tipo de transmisión.

Los principios de su diseño son escalabilidad y simplicidad, características que se ven reflejadas en la magnitud de la información de estado que debe mantenerse en los routers<sup>20</sup> y en la carga introducida en la red como consecuencia de la actividad del protocolo.

Como consecuencia de sus principios de diseño, presenta mayores demoras que los protocolos basados en la construcción de SPTs (Shortest Path Trees) y una mayor vulnerabilidad a fallas debido a la existencia de un único core por grupo<sup>21</sup>.

#### 4.1 Operación del protocolo

Inicialmente, cada router con capacidad CBT, conoce la dirección unicast de un nodo seleccionado como “core” o centro del árbol compartido a través del cual la información originada por los emisores será distribuida a los miembros del grupo multicast.

El mecanismo para la elección del nodo core, no está especificado en el protocolo, mientras que las alternativas para la distribución de la información respectiva a cores a los routers interesados puede realizarse manualmente (configuración estática) o a través de un procedimiento de bootstrapping, actualmente en desarrollo en el contexto de SM-PIM. Cualquiera de los routers puede ser seleccionado como core, no requiriendo capacidad especial. Un nodo core, puede serlo para uno o varios grupos, no siendo necesario que pertenezca a dichos grupos.

##### 4.1.1 Proceso de integración a un grupo

Inicialmente, un router que aún no integra el árbol de distribución de un grupo, recibe una solicitud de un host de su red local para integrarse al mismo. Esta solicitud será recibida a través de IGMP<sup>22</sup>. Como consecuencia de la solicitud recibida, la entidad CBT en el router determina, en base a las tablas de ruteo unicast, la interfaz de salida hacia el router seleccionado como core para el grupo. En el caso en que la interfaz no soporte transmisión multicast<sup>23</sup>, deberá tenerse en cuenta además la dirección unicast del próximo router camino al core.

Esta característica se debe a que en vínculos soportando transmisión multicast, las PDUs CBT se transmiten con la dirección multicast “all\_cbt\_routers” y con ttl igual a 1. Si en

---

<sup>19</sup> En la actualidad, el MBONE funciona encapsulando paquetes entre routers predeterminados que soportan multicast, pero un router en la Internet no necesariamente soporta actualmente transmisión multicast.

<sup>20</sup> Los términos “router” y “nodo” se utilizan en este capítulo de manera indistinta.

<sup>21</sup> Un nodo puede actuar como core para varios grupos, pero un grupo puede tener sólo un core.

<sup>22</sup> La solicitud podrá ser recibida también a través de una aplicación local en un nodo que actuando como router soporte funcionalidad a niveles superiores al de red.

<sup>23</sup> Cuando se menciona que la interfaz o el vínculo no soportan transmisión multicast, en realidad se hace referencia a una red que no provee tal capacidad, por ejemplo NBMA.

cambio, el vínculo no soporta transmisión multicast, las PDUs son enviadas a la dirección unicast del próximo router camino al core. Debe notarse aquí la independencia de CBT respecto del protocolo de ruteo unicast ya que sólo utiliza las tablas construidas por éste.

Una vez determinado el próximo router o interfaz hacia el nodo core, la entidad CBT envía una solicitud (JOIN-REQUEST) para integrarse al árbol de distribución.

Se almacena a su vez información transitoria respecto a la interfaz desde la que se originó el pedido y a la interfaz a la cual se solicitó el requerimiento:

<grupo, interfaz de llegada, interfaz de salida>

En la 3-upla generada, interfaz de llegada (downstream) o interfaz de salida (upstream) pueden ser direcciones unicast del próximo nodo camino al core en caso de que el vínculo no soporte multicast o referencias a interfaces locales en caso de que lo soporte <sup>24</sup>.

Interfaz de llegada, puede ser (y lo será en el caso que se está describiendo) una referencia a interfaz hoja (leaf), en caso en que la solicitud haya sido recibida por una entidad local o por IGMP en representación de un host de una red local al router. Esto significa que el router constituirá, una vez efectivamente integrado al árbol, una hoja del mismo y actuará de acuerdo a ello en ciertos procedimientos especiales.

La información generada y almacenada de la manera descrita se denomina transitoria ya que hasta este momento el nodo no pertenece al árbol de distribución, sino que debe esperar la respuesta al JOIN-REQUEST, un JOIN-ACK. El arribo de esta PDU confirma que se ha contactado al core o a un nodo ya perteneciente al árbol de distribución y que efectivamente se integra el árbol del grupo. Al recibir el JOIN-ACK, el nodo crea información permanente respecto de su pertenencia al árbol, eliminando la información transitoria.

La información transitoria tiene un tiempo máximo de vida, al cabo del cual, si no es recibido el JOIN-ACK, es eliminada. Los procedimientos de recuperación en este caso, dependen si el router es leaf o no. Sólo un router leaf es responsable por la recuperación, realizando una serie de reintentos de envío del JOIN-REQUEST, luego de los cuales desiste. Los próximos reintentos se realizarán en función de solicitudes recibidas de IGMP o de aplicaciones locales.

El procedimiento descrito, a través del cual se recibe una solicitud y se intenta la conexión al árbol a través del próximo router camino al core, se irá produciendo en los routers que reciban un JOIN-REQUEST (en este caso, no serán routers leaf).

Cuando la solicitud arriba a un nodo que ya integra el árbol de distribución o cumple funciones de core para el grupo, éste responde con un JOIN-ACK al nodo anterior, incorporando la interfaz a través de la cual fue recibida la solicitud (JOIN-REQUEST) a la lista de interfaces “child” para el grupo. Este procedimiento se repetirá router a router en camino inverso al recorrido por el JOIN-REQUEST, hasta que el JOIN-ACK llegue hasta el que originó la solicitud inicial.

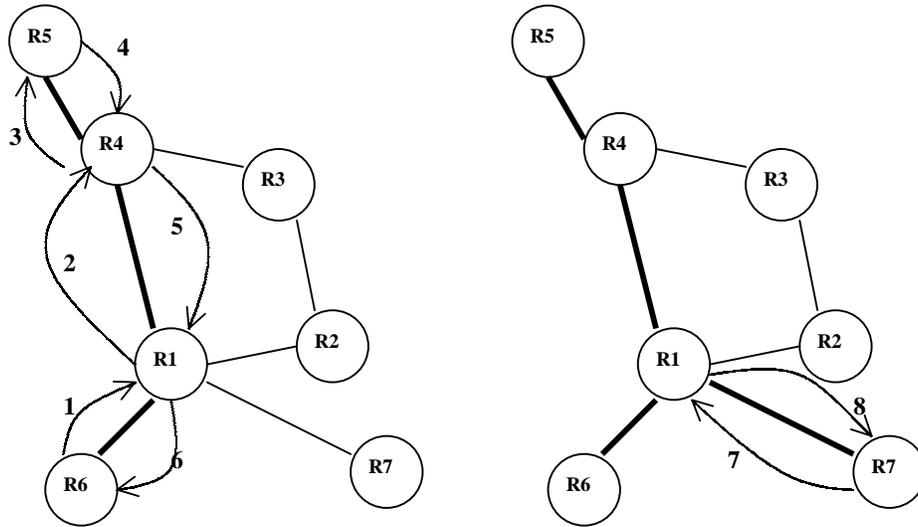
Como resultado del procedimiento descrito, se crean las entradas permanentes en los nodos, con la siguiente información:

< grupo, interfaz parent, { interfaz child, . . . , interfaz child } >

Debe tenerse en cuenta que la característica de parent o child de una interfaz sólo es utilizada por CBT (a diferencia de SM-PIM) para el mantenimiento del árbol de distribución y no para la distribución de los datos multicast, ya que el árbol creado por CBT es bidireccional.

---

<sup>24</sup> En una implementación, se almacenará una dirección IP: la de la placa local hará referencia a la interfaz, mientras que la dirección IP del próximo nodo hará referencia al siguiente nodo camino al core.



**Fig 4.1. Construcción del árbol de distribución CBT:** En la parte izquierda se muestra la secuencia de PDUs intercambiada como consecuencia de un requerimiento de integración al grupo por parte del router R6. A la derecha se muestra la actividad que se produce como consecuencia de un requerimiento posterior de R7. El nodo core para el grupo es R5. En ambos casos los vínculos que integran el árbol de distribución están trazados con línea gruesa. (1,2,3,7:Join-request; 4,5,6,8:Join-Ack).

En la figura 4.1 se muestra el proceso de integración de un router a un grupo y el correspondiente conjunto de PDUs intercambiadas.

#### 4.1.2 Envío y distribución de datos

Una vez integrado el árbol de distribución, un router es capaz de recibir información multicast direccionada al grupo y entregarla a los hosts locales o aplicaciones interesadas.

Un router origen de los datos multicast (emisor) puede o no pertenecer al grupo. En caso de pertenecer, enviará el paquete por la totalidad de las interfaces correspondientes al grupo. De esta manera, la información fluirá por el árbol hacia todos los receptores. En el caso en que un router no miembro del grupo desee enviar información, sólo debe conocer la dirección unicast del core y encapsular el paquete multicast de manera de enviarlo unicast al core. Este desencapsulará el paquete y lo inyectará luego en todas las interfaces asociadas al grupo.

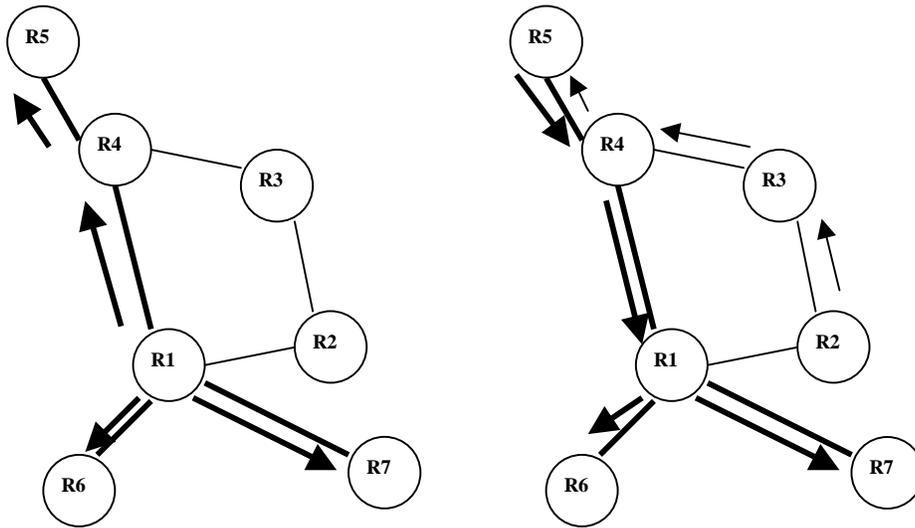
Un paquete que llega a un router, es simplemente enviado por todas las interfaces asociadas al grupo, excepto por la de arriba.

El la figura 4.2 se muestra la manera en que se distribuyen los datos.

#### 4.1.3 Poda del árbol

La poda del árbol de distribución se realiza en el sentido de las hojas hacia el core, y es iniciada por un router cuando la lista de interfaces child para un grupo se vuelve vacía. Esto puede deberse, en el caso de un router leaf, a que los hosts locales no responden a los queries IGMP, o en caso de routers no leaf, a requerimientos de poda de sus routers downstream.

El requerimiento de poda se realiza a través del envío de un QUIT-NOTIFICATION al nodo parent. Este envío no es confirmado por el parent, debido a lo cual la PDU se reenvía a intervalos regulares una cierta cantidad de veces. Al realizarse el primer envío, el router elimina la información del grupo almacenada localmente. Un router, al recibir un QUIT-NOTIFICATION por una interfaz child válida, eliminará la misma de su lista de child para el grupo<sup>25</sup>, lo que puede dar lugar a que dicha lista se convierta en vacía, lo que hará que se continúe el proceso de poda en dirección al core del grupo.



**Fig 4.2. Envío y distribución de paquetes multicast en CBT:** En la parte izquierda se muestra cómo fluye la información multicast originada por R1, quien la inyecta en todas las interfaces asociadas al grupo. A la derecha un router no miembro del grupo, R2, encapsula el paquete multicast y lo envía al core; éste lo inyecta en el árbol, en todas las interfaces. Los routers pertenecientes al árbol, copian un paquete recibido en todas las interfaces del grupo excepto en la de arriba. Las transmisiones unicast se muestran con línea fina, y las multicast con línea gruesa.

#### 4.1.4 Mantenimiento del árbol de distribución

En CBT versión 2, cada nodo child es el responsable de monitorear los vínculos con sus nodos parent<sup>26</sup>, para determinar si existe conectividad. En caso de detectarse pérdida de la conectividad, el nodo se desconecta de los árboles de distribución correspondientes<sup>27</sup>, iniciando un proceso similar hacia las hojas de los subárboles involucrados. De esta manera, un subárbol que resulte desconectado, se eliminará totalmente, y luego, los nodos leaf (interesados en la pertenencia al árbol) tratarán, independientemente, de restaurar la conexión. En ningún caso, un nodo mantiene el subárbol del cual es raíz e intenta reconectarse, debido a que este procedimiento podría dar lugar a la producción de ciclos en el árbol de distribución.

<sup>25</sup> La descripción es simplificada, ya que en casos de vínculos de acceso múltiple se toman otras acciones que serán descritas más adelante.

<sup>26</sup> En CBTv3 se incorpora un mecanismo a través del cual un nodo parent monitorea también el vínculo hacia sus nodos child.

<sup>27</sup> Aquellos árboles correspondientes a grupos cuya interfaz parent está asociada al vínculo.

El monitoreo de cada vínculo se realiza a través de PDUs ECHO-REQUEST, enviadas periódicamente por un nodo child, que son respondidas por el parent con un ECHO-REPLY. La recepción de un ECHO-REPLY por parte de un nodo, provoca el reinicio de los timers asociados a las entradas permanentes para cada grupo involucrado en el mensaje. Si el ECHO-REPLY no es recibido en término, se produce el timeout de la entrada que de esta manera es eliminada, provocando a su vez la eliminación del subárbol a través del envío de FLUSH-TREE a los nodos child para los grupos involucrados.

Un ECHO-REQUEST no contiene información respecto a grupos; un ECHO-REPLY, en cambio, contiene información explícita respecto de los grupos para los que el nodo emisor actúa como parent en el vínculo sobre el que se envía la PDU. Lo mismo ocurre con un FLUSH-TREE.

## **4.2 Operación en vínculos multiacceso**

La operación del protocolo se ve modificada en vínculos de acceso múltiple. Por un lado aparece el concepto de “designated router” (DR), que implica que uno de los routers conectados al vínculo será seleccionado para realizar las conexiones al árbol para todos los grupos, y por otro lado se modifica en algunos casos la operación descrita arriba para vínculos punto a punto.

La necesidad de un DR en un vínculo multiacceso está dada en el caso de que dicho vínculo conecte dos o más dominios de ruteo debido a que cada uno de ellos puede tener una visión diferente del ruteo unicast, lo cual podría dar lugar a la formación de ciclos en el árbol de distribución.

### **4.2.1 Elección del DR**

Cualquiera de los routers tiene la capacidad de actuar como DR, estando esta condición indicada por una variable (flag DR) asociada a cada una de las interfaces multiacceso de un router; por defecto su valor es FALSE, indicando que el router no es DR.

Cada router tiene almacenado un valor de preferencia por interfaz multiacceso, que puede variar entre 1 y 255, siendo 1 el valor que representa una mayor prioridad para la elección del router como DR, y 255 el que representa la menor prioridad. Por defecto, los routers asumen un valor de 255; pueden ser configurados con valores entre 1 y 254. El valor 0 se reserva para uso del router elegido como DR, independientemente de cual sea su valor de preferencia configurado.

La elección y mantenimiento como tal de un router para cumplir el rol de DR, se realiza de acuerdo al protocolo HELLO, como se describe a continuación.

Cada router tiene un timer para enviar periódicamente una PDU HELLO a la dirección multicast “all-cbt-routers”. Un router que recibe un HELLO con un valor de preferencia más bajo que el propio (o igual pero proveniente de un router con dirección más baja), reinicia el timer y de esta manera se abstiene de enviar el HELLO propio.

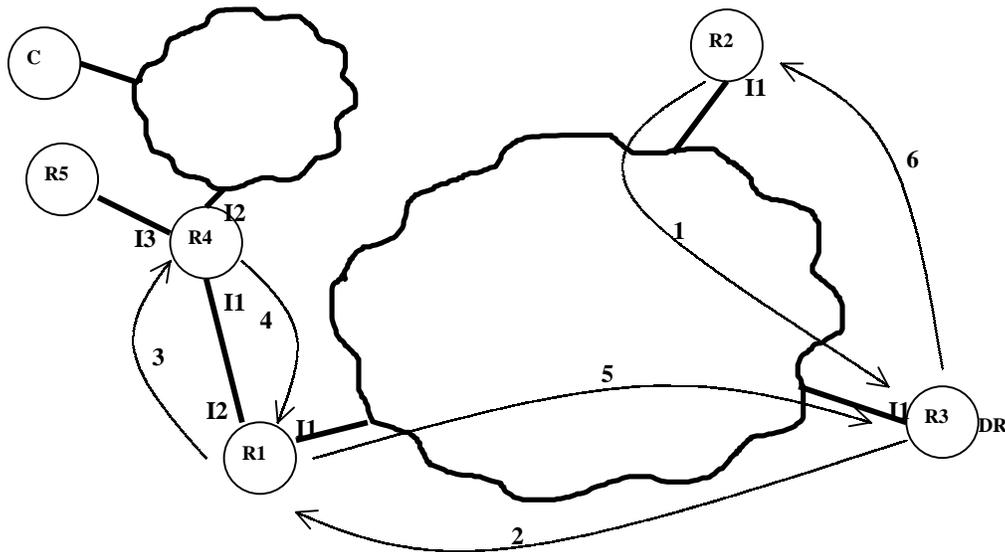
El router que ha anunciado un valor de preferencia más bajo, es el que se convierte en DR luego de un cierto tiempo, al no recibir HELLOs de los demás routers. Independientemente del valor de preferencia configurado, el DR anunciará un valor de preferencia 0.

Un router que recibe un HELLO con un valor de preferencia menor que el propio, o igual pero de un router con una dirección mayor, debe contestar enviando su valor de preferencia.

### **4.2.2 Consideraciones respecto a la operación del protocolo**

Como consecuencia de la existencia de un DR, un router podrá tener dos interfaces (desde el punto de vista del mantenimiento del árbol de distribución) sobre el mismo vínculo multiacceso<sup>28</sup>. De acuerdo a esto, un DR es el único router que procesa JOIN-REQUESTs multicast recibidos por el vínculo multiacceso.

Un router que recibe un QUIT-NOTIFICATION multicast por una interfaz correspondiente a un vínculo multiacceso, no debe eliminar inmediatamente la interfaz child, dado que puede haber otros routers child en la misma interfaz<sup>29</sup>. En cambio, activa un timer (cache\_deletion\_timer) para eliminar la información referente a la interfaz luego de cumplido el timeout. Esta espera da tiempo a otros routers child sobre el mismo vínculo, para producir el envío de un JOIN-REQUEST para el grupo y lograr de esta manera que el parent no elimine la interfaz.



**Fig 4.3. Secuencia de PDUs intercambiadas en un vínculo multiacceso: actuando R3 como designated router para el establecimiento del árbol de distribución del grupo G teniendo como core al nodo C**

Para posibilitar un uso eficiente de la red respecto al envío del JOIN-REQUEST, un router que recibe un QUIT-NOTIFICATION por una interfaz parent multiacceso, activa un timer con un tiempo al azar entre 0 y un tiempo máximo prefijado, para realizar el envío. Si antes del vencimiento, el router detecta un JOIN-REQUEST para el mismo grupo, proveniente de un router que ha elegido un tiempo menor, cancela el envío pendiente.

En la figura 4.3 se muestra el intercambio de PDUs en el caso de un vínculo multiacceso con un DR (R3) que no es el mejor camino al core (C).

La notación utilizada para indicar las tablas en cada nodo se muestra más abajo. El primer elemento (G) indica el grupo, el segundo elemento la interfaz parent, compuesta de una

<sup>28</sup> Un DR tendrá una interfaz child multicast sobre el vínculo multiacceso y una interfaz parent unicast, con la dirección del próximo nodo camino al core, en caso de que el nodo (DR) no constituya el mejor camino al core (para la métrica de ruteo unicast) entre los routers del vínculo multiacceso.

<sup>29</sup> El router parent sólo tiene información respecto de la interfaz pero no respecto a los routers en particular.

referencia a la interfaz física y entre paréntesis una indicación del nodo adyacente (\* indica que no existe referencia a un nodo específico –envío multicast-) y el tercer elemento el conjunto de interfaces child, para las cuales se utiliza la misma notación que para la interfaz parent.

Inicialmente, R4 pertenece al árbol de distribución del grupo (G), como consecuencia de un requerimiento de R5.

Su tabla es: <G, I2 (\*), { I3 (\*)}>

Se produce a continuación la solicitud del router R2 para integrarse al grupo G. R2 obtiene como mejor interfaz hacia el core C, su interfaz I1 correspondiente al vínculo multiacceso. Como consecuencia se produce el intercambio de PDUs en el orden especificado en la figura:

|                             |                         |
|-----------------------------|-------------------------|
| 1 JOIN-REQUEST multicast    | 4 JOIN-ACK multicast    |
| 2 JOIN-REQUEST unicast a R1 | 5 JOIN-ACK unicast a R3 |
| 3 JOIN-REQUEST multicast    | 6 JOIN-ACK multicast    |

Como consecuencia de no ser el DR el mejor router al core en el vínculo multiacceso, se producen dos PDUs adicionales. Debe tenerse en cuenta que el overhead es mínimo, ya que no se produce en la distribución de los datos.

Las tablas en cada router contendrán:

R1: <G, I2 (\*), I1 (R3)>  
R2: <G, I1 (\*), {LEAF}>  
R3: <G, I1 (R1), { I1 (\*)}>  
R4: <G, I2 (\*), { I1 (\*), I3 (\*)}>

### 4.3 Resumen

En este capítulo se presentó una descripción simple de las características más importantes de CBTv2: integración a un grupo y construcción del árbol de distribución, mantenimiento y poda del árbol y envío de datos. Posteriormente se describieron características adicionales que surgen al considerar su operación en vínculos multiacceso. Esta descripción junto con la documentación del protocolo[6][7][8] constituyen la base para la especificación realizada en el capítulo 5.