

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 425 564**

51 Int. Cl.:

H04L 29/06 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **08.02.2005 E 05704809 (2)**

97 Fecha y número de publicación de la concesión europea: **05.06.2013 EP 1754357**

54 Título: **Método para reforzar la tolerancia contra la reordenación de paquetes cuando se utiliza el principio de referencia segura en la compresión robusta de cabeceras**

30 Prioridad:

24.05.2004 SE 0401346

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

16.10.2013

73 Titular/es:

**TELEFONAKTIEBOLAGET LM ERICSSON (PUBL)
(100.0%)
PATENT UNIT
164 83 STOCKHOLM, SE**

72 Inventor/es:

**JONSSON, LARS-ERIK y
PELLETIER, GHYSLAIN**

74 Agente/Representante:

DE ELZABURU MÁRQUEZ, Alberto

ES 2 425 564 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Método para reforzar la tolerancia contra la reordenación de paquetes cuando se utiliza el principio de referencia segura en la compresión robusta de cabeceras

5 La presente invención se refiere a un método de compresión de cabeceras que usa un principio de referencia segura sobre canales que puede reordenar paquetes entre un compresor y un descompresor sin el riesgo de generar paquetes descomprimidos erróneamente.

10 Debido al tremendo éxito de Internet, ha llegado a ser una tarea desafiante hacer uso del Protocolo de Internet (IP) sobre todo tipo de enlaces. No obstante, debido al hecho de que las cabeceras de los protocolos IP son bastante grandes, no es siempre una tarea simple hacer esto una realidad para enlaces de banda estrecha, por ejemplo enlaces celulares. Como ejemplo, considerar datos de habla normales transportados por los protocolos (IP, UDP, RTP) usados para Voz sobre IP (VoIP), donde la cabecera puede representar alrededor del 70% del paquete provocando un uso muy ineficiente del enlace.

15 El término compresión de cabeceras (HC) comprende la técnica de minimizar el ancho de banda necesario para la información transportada en las cabeceras sobre una base por salto sobre enlaces punto a punto. Las técnicas en general tienen una historia de más de diez años dentro de la comunicad de Internet; existen varios protocolos usados comúnmente tales como RFC 1144, RFC 2507 y RFC 2508.

20 La compresión de cabeceras toma ventaja del hecho de que algunos campos en las cabeceras no están cambiando dentro de un flujo, o cambian con valores pequeños y/o predecibles. Los esquemas de compresión de cabeceras hacen uso de estas características y envían información estática solamente inicialmente, mientras que los campos que cambian se envían con sus valores absolutos o como diferencias de paquete a paquete. Se tiene que enviar información completamente aleatoria sin ninguna compresión en absoluto.

25 La compresión de cabeceras es de esta manera un componente importante para hacer los servicios IP sobre inalámbrico, tales como servicios de voz y vídeo, factibles económicamente. Las soluciones de compresión de cabeceras se han desarrollado por el Grupo de Trabajo de Compresión Robusta de Cabeceras (ROHC) del IETF para mejorar la eficiencia de tales servicios.

30 Cuando la compresión de cabeceras se usa sobre enlaces de reordenación, tales como túneles IP u otros circuitos virtuales de múltiples saltos, la reordenación de paquetes puede impactar generalmente un algoritmo de compresión de cabeceras al menos de tres formas diferentes:

- 1) los contextos del compresor y descompresor pueden salirse de la sincronización;
- 2) los paquetes se pueden descomprimir erróneamente por el descompresor (detectado);
- 3) los paquetes se pueden descomprimir erróneamente por el descompresor (no detectado).

35 Un problema abordado por la invención es que el principio de referencia segura no es robusto cuando la reordenación de paquetes ocurre entre el compresor y el descompresor. En particular, el uso del principio de referencia segura sobre enlaces que pueden reordenar paquetes puede conducir a paquetes que son descomprimidos erróneamente y entonces reenviados a las capas superiores (punto 3 anterior).

40 Señalar que esto no ocurre con el planteamiento optimista debido a que se usa información redundante (por ejemplo una suma de comprobación) para impedir errores de descompresión en todos los paquetes, y la descompresión defectuosa debida a la reordenación se puede detectar de esta manera por el descompresor. Los paquetes erróneos se pueden descartar entonces en lugar de ser reenviados a las capas superiores.

45 Según un aspecto de la presente invención hay proporcionado un método para un descompresor como se expone en la Reivindicación 1.

50 Según un segundo aspecto de la presente invención hay proporcionada una adaptación para un descompresor como se expone en la Reivindicación 10.

55 Realizaciones de la invención permiten un algoritmo de compresión de cabeceras usando el principio de referencia segura a ser usado sobre canales que pueden reordenar paquetes entre el compresor y el descompresor sin el riesgo de generar paquetes descomprimidos erróneamente. Esto se hace posible a partir de adaptar las propiedades de robustez del principio de referencia segura para tener en cuenta las características de la reordenación.

60 La robustez del algoritmo de compresión de cabeceras usando el principio de referencia segura (en compresión de cabeceras bidireccional) es dependiente del efecto acumulativo de las actualizaciones de contexto. Las modificaciones de la invención aseguran que la reordenación no puede afectar la descompresión basada en una referencia segura. Estas ideas son particularmente útiles para sistemas donde se usa la ROHC operando en modo

65

R.

Esto es particularmente aplicable a la mayoría de los perfiles de ROHC, incluyendo – pero no limitado a- los perfiles de compresión de cabeceras ROHC RTP (0x0001), UDP (0x0002), IP (0x0004), ESP (0x0003), UDP-Lite (0x0008), RTP/UDP-Lite (0x0007). Algunas de las soluciones propuestas también tienen la ventaja de no requerir ningún cambio de cualquiera de los estándares de ROHC.

Se describirán ahora realizaciones de la invención, a modo de ejemplo, con referencia a los dibujos anexos, en los que:

La Figura 1 ilustra la actualización de la Referencia Segura (SR), de entrega en orden.

La Figura 2 ilustra el problema con la Actualización de la Referencia Segura (SR), con reordenación según la última tecnología.

La Figura 3 ilustra la actualización de la Referencia Segura (SR), con reordenación según la invención.

Un glosario de abreviaturas usadas en esta especificación de patente se expone más adelante para facilitar una comprensión de la presente invención.

ROHC	Compresión Robusta de Cabeceras
Modo U	Modo unidireccional
FC	Contexto completo
VoIP	Voz sobre IP

La primera parte de lo siguiente describirá brevemente los rasgos y modos de operación relevantes, de compresión de cabeceras, para una mejor comprensión de la invención.

Compresión Robusta de Cabeceras (ROHC)

ROHC, como se define en la RFC 3095 [ROHC], es un marco extensible para el cual se pueden definir perfiles para la compresión de diversos protocolos. Para los servicios multimedia en tiempo real (por ejemplo voz, vídeo), los datos de aplicaciones se transportan extremo a extremo dentro de un flujo IP/UDP/RTP. La compresión de cabeceras de IP/UDP/RTP se define por el perfil de ROHC 0x0001 (ROHC RTP) y es aplicable para servicios de Voz sobre IP (VoIP) entre otros. El esquema de compresión de cabeceras ROHC RTP ha sido diseñado para comprimir eficientemente las cabeceras IP/UDP/RTP sobre una capa de enlace arbitraria. Excepto para negociación (ver también [ROHC-PPP]), los perfiles de ROHC solamente requieren que la alineación de tramas y detección de errores sea proporcionada por la capa de enlace, mientras que todas las demás funcionalidades se manejan por el esquema de ROHC en sí mismo.

Además del perfil de ROHC RTP, una serie de otros perfiles¹ de ROHC también han sido definidos para la compresión de:

- Cabeceras IP/UDP y cabeceras IP/ESP [ROHC];
- Cabeceras solamente IP [IP-ONLY];
- Cabeceras IP/UDP-Lite/RTP [ROHC-UDPLite].

Suposición de entrega en orden

Los perfiles de compresión de cabeceras definidos en la RFC 3095 [ROHC] fueron diseñados con *la suposición que el canal entre el compresor y el descompresor no reordenará los paquetes comprimidos de cabecera*; el canal se requiere para mantener la ordenación de paquetes para cada flujo comprimido. Esta suposición fue motivada debido a que los canales considerados inicialmente como candidatos potenciales para usar la ROHC garantizaban la entrega en orden de los paquetes; esta suposición fue útil para mejorar la eficiencia de compresión y la tolerancia frente a la pérdida de paquetes, objetivos que fueron clasificados los más altos en la lista de requisitos en el momento. El perfil para compresión de cabeceras IP solamente [IP-ONLY] y los perfiles para UDP-Lite son esencialmente extensiones a los perfiles encontrados en [ROHC]; por lo tanto, estos perfiles también heredan la misma suposición de entrega en orden.

Contexto de compresión de cabeceras

Un contexto de compresión contiene y mantiene información relevante acerca de paquetes pasados, y esta información se usa para comprimir y descomprimir paquetes posteriores. Tomado de la [ROHC]:

“El contexto del compresor es el estado que usa para comprimir una cabecera. El contexto del descompresor es el estado que usa para descomprimir una cabecera. Cualquiera de estos o los dos en combinación se conocen normalmente como “contexto”, cuando está claro que está previsto. El contexto contiene información relevante de las cabeceras previas en el flujo de paquetes, tal como campos estáticos y posibles valores de referencia para compresión y descompresión. Además, la información adicional que describe la secuencia de paquetes es también parte del contexto, por ejemplo información acerca de cómo el campo de Identificador IP

cambia y el interpaquetes típico aumenta en números de secuencias o sellos temporales.”

Máquinas de estado de compresión de cabeceras y sincronización de contexto

5 Uno puede normalmente realizar un esquema de compresión de cabeceras (tal como un Perfil de ROHC) como una máquina de estado y la tarea desafiante es para mantener los estados del compresor y descompresor, llamados contextos, consistentes uno con otro, mientras que se mantiene la sobrecarga de cabecera tan baja como sea posible. Hay una máquina de estado para el compresor, y una máquina de estado para el descompresor. La máquina de estado del compresor impacta directamente el nivel de eficiencia de compresión, ya que es una parte importante de la lógica que controla la elección del tipo de paquete comprimido a ser enviado; el propósito de la máquina de estado de descompresor es principalmente proporcionar la lógica para realimentación (si es aplicable) e identificar los tipos de paquetes para los que se puede intentar la descompresión.

Tipos de paquetes y actualizaciones de contexto

15 Un paquete que proporciona los medios para que el descompresor verifique la descompresión con éxito es un *paquete de actualización de contexto*. Debido a que la descompresión se puede verificar, este tipo de paquete puede actualizar el contexto. Para ROHC, los tipos de paquete de actualización de contexto transportan un Código de Redundancia Cíclico (CRC) dentro de su formato; este es una suma de comprobación calculada sobre la cabecera no comprimida original. Este CRC se usa para verificar la descompresión con éxito de cada paquete; cuando es un éxito, el contexto se puede actualizar.

20 Un paquete que se basa en otros medios para garantizar la descompresión con éxito – es decir un formato de paquete no proporciona los medios para que el descompresor verifique la descompresión con éxito, y que solamente transporta la información necesaria para la descompresión en sí misma, es un paquete *autocontenido*. Estos paquetes no actualizan el contexto.

25 Modos de operación
Los perfiles de ROHC definidos en la RFC 3095, [IP-ONLY] y [ROHC-UDPLite] todos soportan tres modos de operación diferentes. En resumen, para un contexto específico, el modo controla las acciones y la lógica a realizar así como los tipos de paquetes a usar durante diferentes estados de la operación de compresión de cabeceras. Los tipos y formatos de paquetes que se permiten pueden variar de un modo a otro. El modo Unidireccional (modo U) se usa en el comienzo de cualquier compresión ROHC antes de que puede ocurrir cualquier transición a otros modos. El modo Optimista Bidireccional (modo O) aspira a maximizar la eficiencia de compresión y el uso escaso del canal de realimentación. El modo Fiable Bidireccional (modo R) aspira a maximizar la robustez frente a la propagación de pérdidas y la propagación de daños de contexto. Cada modo de operación tiene diferentes propiedades en términos de eficiencia de compresión, robustez y complejidad de procesamiento.

40 En el modo U, los paquetes se envían desde el compresor al descompresor solamente; este modo es utilizable de esta manera sobre enlaces donde un recorrido de retorno desde el descompresor al compresor es o bien no deseado o bien no está disponible, y se usan refrescos periódicos en este modo. El modo U es particularmente aplicable a canales de difusión o multidifusión.

45 El modo O es similar al modo U, con la diferencia de que se usa un canal de realimentación para enviar peticiones de recuperación de errores y (opcionalmente) reconocimientos de actualizaciones de contexto significativas desde el descompresor al compresor.

50 Señalar que la mayoría de los perfiles de ROHC, el modo U y el modo O son conocidos a menudo indistintamente usando el término *modo U/O*. Esto es porque el modo U y el modo O tienen características bastante similares, tales como un conjunto idéntico de formatos de paquetes para ambos modos así como una lógica similar para realizar actualizaciones de contexto. Esta lógica se denomina el *planteamiento optimista*, y proporciona robustez contra las pérdidas de paquetes para el procedimiento de actualización de contexto en el modo U/O. Ver también [ROHC, sección 5.3.1.1.1] para más detalles.

55 El modo R difiere significativamente de los otros dos modos. En particular, el modo R usa unos pocos tipos de paquetes diferentes solamente entendidos y útiles en este modo. No obstante, el modo R difiere principalmente haciendo un uso más extensivo del canal de realimentación y usa una lógica más estricta para realizar las actualizaciones de contexto. Esta lógica está basada en el *principio de referencia segura*, y proporciona robustez frente a pérdidas de paquetes para el procedimiento de actualización de contexto en el modo R. Ver también [ROHC, sección 5.5.1.2] para más detalles.

60 Principios de robustez en compresión robusta de cabeceras – planteamiento optimista

65 Un compresor de cabeceras puede usar el planteamiento optimista para reducir la sobrecarga de cabecera cuando se realizan actualizaciones de contexto. El compresor normalmente repite la misma actualización hasta que está bastante seguro de que el descompresor ha recibido con éxito la información. El número de paquetes consecutivos necesarios para obtener esta confianza está típicamente abierto a las implementaciones, y este número está relacionado normalmente con las características de pérdidas de paquetes del enlace donde se usa la compresión de

cabeceras. Todos los tipos de paquetes usados con el planteamiento optimista son de actualización de contexto.

Los principios de robustez en compresión robusta de cabeceras – principio de referencia segura

Un compresor de cabecera puede usar el principio de referencia segura para asegurar que la sincronización de contexto entre el compresor y el descompresor no puede ser perdida debido a pérdidas de paquetes. El compresor obtiene su confianza de que el descompresor ha actualizado con éxito el contexto a partir de un paquete de actualización de contexto basado en los reconocimientos recibidos por el descompresor. No obstante, la mayoría de los tipos de paquetes usados con el principio de referencia segura son autocontenidos y de esta manera no se supone que actualicen el contexto.

Compresión de cabeceras y reordenación entre el compresor y el descompresor

El grupo de trabajo (WG) de Transporte de Audio-Vídeo (AVT) del IETF está trabajando en compresión de cabeceras sobre múltiples saltos. Mientras que la compresión de cabeceras se prevé principalmente para abordar enlaces de baja velocidad donde el ancho de banda es escaso, ahorrando ancho de banda en las instalaciones de la troncal también es de importancia debido a los altos costes y la cantidad considerable de tráfico transportado dentro. La compresión de cabeceras se puede aplicar punto a punto entre cada nodo en la red troncal, no obstante esto requiere que los paquetes sean descomprimidos y vueltos a comprimir en cada nodo. El procesamiento se puede disminuir realizando compresión entre nodos no adyacentes en la red troncal, sobre un recorrido de múltiples saltos.

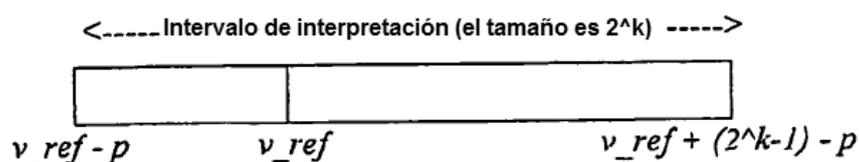
Por ejemplo, un recorrido de múltiples saltos puede ser una ruta Conmutada de Etiquetas Multiprotocolo (MPLS) en la red troncal, o un túnel IP. Una tasa de pérdida de paquetes más alta y la reordenación posible de paquetes caracterizan tales enlaces virtuales, abarcando sobre múltiples saltos. Esto puede resultar a partir de paquetes que se reencaminan o simplemente descartan en un nodo, debido a la congestión o fallo de un nodo. Los requisitos para compresión de cabeceras sobre múltiples saltos están en el [AVT-HC].

Codificación del Bit Menos Significativo (LBS)

El método de codificación del LSB se usa para codificar los campos de cabeceras cuyos valores están sujetos normalmente a cambios pequeños, tales como números de secuencia (SN), por ejemplo los SN de RTP o los SN creados en el descompresor cuando se comprimen protocolos que no tienen numeración de secuencia dentro de su formato de cabecera.

Los k bits menos significativos del valor del campo se envían en lugar del valor del campo entero, donde k es un entero positivo. Cuando se reciben esos bits, el descompresor deriva el valor original usando un valor recibido previamente v_{ref} .

Este método de codificación se garantiza para dar el resultado correcto si el compresor y el descompresor usan intervalos de interpretación en los cuales reside el valor original y en los cuales el valor original es el único valor que tiene los mismos bits lsb que aquéllos transmitidos.



El parámetro p se usa para cambiar el intervalo de interpretación con respecto a v_{ref} . Un método de codificación derivativo, llamado codificación de Ventana LSB, usa una ventana de v_{ref} candidato. Con el principio de referencia segura, los reconocimientos desde el descompresor permiten al compresor extraer los valores de v_{ref} que son más antiguos que el reconocido desde la ventana de deslizamiento. El LSB y W-LSB se describen en la [ROHC].

Revisión de robustez del principio de referencia segura

Una consecuencia del principio de referencia segura (donde no todos los paquetes son de actualización de contexto) es que solamente los valores reconocidos por el descompresor están incluidos como referencias en la ventana de deslizamiento de codificación (por ejemplo codificación del LSB o W-LSB). Este principio de robustez permite que los paquetes comprimidos sean enviados con un formato² que no incluye medios para que el descompresor verifique la correcta descompresión (usando por ejemplo una suma de comprobación sobre la cabecera no comprimida original), ya que la recepción de los bits decodificados LSB aplicados a la referencia segura es suficiente para una correcta descompresión. La descompresión enteramente se basa en el efecto acumulativo de actualizaciones previas a la referencia segura, y los datos comprimidos se basan en el valor actual de la referencia – que debe ser el mismo tanto para el compresor como el descompresor. Esto es adecuado cuando se garantiza una entrega en orden entre el compresor y el descompresor.

No obstante si puede ocurrir una reordenación, la consecuencia de este principio de robustez es que el

descompresor no tiene los medios para verificar la descompresión de los paquetes autocontenidos, es decir no suponía actualizar la referencia segura; estos paquetes normalmente cuentan para la mayoría de los paquetes intercambiados entre el compresor y el descompresor.

5 Como se mencionó anteriormente, algunos algoritmos de compresión de cabeceras (por ejemplo, la ROHC) pueden haber sido diseñados con la suposición de que el canal entre el compresor y el descompresor entrega paquetes al descompresor en el mismo orden que cuando salen del compresor. Esto significa que un compresor de última tecnología normalmente seleccionará el tipo de paquete más óptimo basado en las características de la cabecera a ser comprimida y basada en el contexto, *no basado en las características de reordenación posible del enlace*.

10 En particular, un compresor de cabeceras que opera usando el planteamiento de robustez del principio de referencia segura se espera que use el tipo de paquete autocontenido más óptimo (el más pequeño) (por ejemplo R-0 para modo R de ROHC) la mayoría del tiempo, el cual no suponía actualizar la referencia segura y de esta manera para el cual la descompresión no puede ser verificada. Otros tipos de paquetes autocontenidos ligeramente menos óptimos también pueden existir (por ejemplo R-1 para modo R de ROHC).

15 Esto significa que un descompresor de última tecnología *no tendrá la capacidad de manejar la reordenación y detectar si un paquete recibido fue descomprimido basado en la referencia equivocada en el contexto cuando se aplica el principio de referencia segura*. La sección 3 describe conceptos usados por implementaciones de ROHC de última tecnología.

20 Última tecnología cuando se usa el principio de referencia segura
La Figura 1 muestra un ejemplo típico de un compresor (parte superior) y un descompresor (parte inferior) que operan usando el principio de referencia segura. Los paquetes comprimidos se intercambian con el tiempo (eje de Secuencia), y la ventana de deslizamiento de LSB de Referencia Segura (SR) se actualiza después de eventos específicos. Señalar que la ventana de deslizamiento puede contener más de un valor en ciertos momentos, pero *hay siempre solamente uno que es la referencia segura usada para la compresión y la descompresión de un campo específico*.

25 El objetivo de los iguales de compresión es permanecer siempre sincronizados con respecto a qué referencia se usa para la compresión/descompresión de un paquete particular. En particular, lo siguiente aplica y se refleja en la figura 1:

30 El *descompresor* solamente puede verificar la descompresión con éxito de los paquetes de actualización de contexto (paquetes que pueden actualizar la referencia segura).

35 El *descompresor* no puede verificar la descompresión con éxito de un paquete autocontenido (un paquete que no actualiza la referencia segura). El descompresor enteramente se basa en la suposición de entrega en orden, la cual a su vez proporciona la garantía de que los paquetes autocontenidos se reciben cuando la referencia segura activa es la misma referencia que fue usada cuando se comprime el paquete.

40 El *compresor* actualiza su ventana de deslizamiento de referencias seguras cuando se recibe un reconocimiento desde el descompresor. La(s) referencia(s) previa(s) (reconocida(s) y/o no reconocida(s)) se extraen de la ventana, y *solamente la última reconocida se mantiene como la referencia segura*.

45 El *descompresor* actualiza su ventana de deslizamiento de referencias seguras cuando se recibe un paquete para el cual los LSB son menos que los paquetes tempranos, indicando que se ha comprimido con la referencia que el descompresor ha reconocido previamente. *Solamente la última referencia para la cual fue enviado un reconocimiento se mantiene entonces como la referencia segura*.

50 Un algoritmo de compresión de cabeceras para el cual la robustez a las pérdidas de paquetes está basada en el principio de referencia segura, que es un principio acumulativo, no es robusto para cierto tipo de reordenación. Esto puede conducir a que algunos paquetes sean descomprimidos erróneamente y entonces reenviados a las capas superiores, de vuelta en la red o a la aplicación. Esto es posible debido a que los paquetes autocontenidos se pueden recibir por el descompresor fuera de servicio con respecto a otro paquete que actualiza una o más referencias del estado del descompresor usado para la descompresión de los campos de cabecera (por ejemplo el número de secuencia).

55 Los problemas surgen cuando un paquete autocontenido – un paquete que no proporciona los medios para verificar si su descompresión fue un éxito – se reordena de manera que el descompresor lo recibe después de una secuencia de paquetes que entre tanto actualizó la referencia segura. Esto significa que una reordenación que retarda un paquete autocontenido de manera que, entre tanto:

- 60 - el compresor ha recibido el reconocimiento para una nueva referencia segura;
- 65 - el compresor ha comenzado la compresión según esta nueva referencia;

- el descompresor ha recibido al menos un paquete comprimido con la nueva referencia segura antes de que el paquete reordenado se reciba.

Esto se muestra en la figura 2.

5 Por ejemplo, en el caso de un servicio de VoIP donde se genera una trama de habla en cada límite de 20 ms y envía individualmente, en el peor caso puede ser suficiente que el paquete de cabecera comprimida autocontenido se reordene con uno o dos paquete(s) adyacente(s), conduciendo a un retardo debido a la reordenación entre tan poco como 20-40 ms sobre el canal entre el compresor y el descompresor. En tal caso, pueden ocurrir artefactos en el habla resultante entregado al usuario; esto es porque los números de secuencia (y potencialmente otros campos comprimidos como función de los NS) del paquete reordenado son incorrectos, y la aplicación no puede detectar este error.

10 Para abordar los problemas debidos a la reordenación, podría ser diseñado un compresor de manera que solamente se envíen los paquetes de actualización de contexto en todos los momentos, incluso cuando se usa el principio de referencia segura. No obstante, este planteamiento aparentemente “directo” eliminaría alguna eficiencia de compresión y robustez al principio de referencia segura generando actualizaciones de contexto más frecuentes, y forzaría al descompresor en enviar una gran cantidad de (innecesaria) realimentación. Esto aumentaría la cantidad de datos intercambiados tanto en el canal directo como el inverso entre el compresor y el descompresor. Modificar los estándares existentes creando nuevos formatos de paquetes podría mitigar alguno de estos efectos, pero esto aún vencería en efecto las ideas básicas detrás del principio de referencia segura.

15 Cuando se aplica compresión de cabeceras sobre canales que pueden reordenar paquetes, es por lo tanto deseable asegurar que ningún paquete puede ser descomprimido erróneamente y propagado de vuelta en la red y a las aplicaciones. Esto es de esta manera útil para encontrar una solución para mejorar la tolerancia y robustez del principio de referencia segura frente a la reordenación, para impedir cualquier interrupción o artefactos en el servicio que podrían ser causados por el algoritmo de compresión de cabeceras. Esto debería ser hecho mientras que se minimiza la sobrecarga introducida por las modificaciones al algoritmo de compresión de cabeceras.

20 Lo siguiente describe una serie de realizaciones de la invención, basadas en los tres planteamientos diferentes anteriores y basadas en, pero no limitadas a, el comportamiento del compresor y del descompresor de la RFC-3095 [ROHC].

Lógica solamente de descompresor – usando la suma de comprobación de la capa de transporte

35 *Idea básica:* Si se usa la suma de comprobación de la capa de transporte, se puede usar por el descompresor como un medio (aunque débil) para detectar una descompresión errónea debida a la reordenación.

40 *Descripción:* El descompresor puede verificar que los paquetes autocontenidos que se podrían reordenar de manera que puedan ser descomprimidos erróneamente, es decir descomprimidos en base a la referencia segura incorrecta, se detectan como sigue: para una serie de paquetes autocontenidos equivalente a la reordenación estimada máxima posible después de una actualización de contexto (un ACK fue enviado para la nueva referencia), el descompresor verifica la exactitud del paquete descomprimido entero usando la suma de comprobación de la capa de transporte.

45 Señalar que esto no requiere ninguna modificación a los estándares. Señalar que esto es aplicable solamente cuando la suma de comprobación de la capa de transporte está habilitada, lo cual no siempre es el caso para IPv4.

Lógica solamente de descompresor: reconocimientos de retardo

50 *Idea básica:* El descompresor usa las propiedades de autorregulación de los métodos de codificación del LSB y W-LSB para determinar cuándo actualizar la referencia segura. El descompresor asegura que actualiza su referencia segura a un nuevo valor solamente cuando no pueden ser recibidos paquetes autocontenidos comprimidos usando una referencia segura más antigua después del primer paquete autocontenido comprimido usando la referencia actualizada.

55 *Descripción:* Después de que el descompresor recibe un paquete que puede actualizar la referencia segura, el descompresor puede asegurar que el compresor usará la referencia segura antigua hasta que los valores a ser comprimidos con esta referencia estén fuera del intervalo de interpretación que corresponde a paquetes de no actualización de contexto. En otras palabras, el descompresor puede asegurar que el compresor es forzado a enviar paquetes de actualización de contexto para un número de paquetes al menos igual a la reordenación máxima posible en el canal antes de la actualización de la referencia segura con un nuevo valor y enviar el reconocimiento correspondiente al descompresor.

60 Si el descompresor asegura que la referencia segura no es actualizada hasta que el compresor ha enviado un número de paquetes de actualización de contexto al menos igual a la reordenación máxima posible que pueden experimentar los paquetes en el canal entre el compresor y el descompresor, entonces no puede ocurrir ninguna descompresión errónea debida a la reordenación no detectada.

65

El descompresor puede lograr esto retardando el envío de un reconocimiento de una referencia segura actualizada por al menos la cantidad máxima de retardo de la reordenación posible que puede ocurrir. La idea general se muestra en la figura 3.

- 5 Señalar que esto se puede usar en combinación con la primera lógica alternativa. Señalar también que esto no requiere ninguna modificación a los estándares.

Realización posible usando ROHC

- 10 Tomando el ejemplo específico de ROHC en modo R, los tipos de paquetes R-0 y R-1^{*} cada uno tienen seis bits LSB en su formato; el tamaño resultante del intervalo de interpretación usando este tipo de paquetes es de esta manera 2⁶, es decir 64. Estos paquetes son autocontenidos y no actualizan la referencia segura. El tipo de paquetes UOR-2 tiene seis, nueve o 14 bits LSB en su formato (con extensiones); el tamaño máximo resultante del intervalo de interpretación usando este tipo de paquetes es de esta manera 2¹⁴, es decir 16.384.

- 15 Esto significa que un compresor ROHC enviaría normalmente el tipo de paquetes R-0 y R-1^{*} sin actualizar la referencia segura durante un máximo de 64 paquetes. Después de esto, el número de bits LSB dentro del formato de estos paquetes ya no es suficiente, y la referencia segura se debe actualizar o se debe usar un tipo de paquete diferente con más bits LSB para trasladar la información comprimida sin ambigüedades. Los paquetes como más bits LSB, tales como UOR-2, son de actualización de contexto.

- 20 Específicamente para la ROHC, la idea se puede resumir como sigue:

- 25 “Implementar el descompresor del modo R de ROHC de manera que el tipo de paquetes de R-0-CRC nunca es reconocido. Para otros tipos de paquetes de actualización de contexto, retardar los reconocimientos hasta que es recibido un número de paquetes de actualización de contexto consecutivos al menos igual al retardo máximo posible debido a la reordenación.”

Lógica compresor-descompresor – reconocimientos selectivos

- 30 *Idea básica: El compresor envía solamente paquetes de actualización de contexto. El descompresor actualiza su referencia segura y envía los reconocimientos correspondientes solamente para un tipo específico de paquete.*

- 35 *Descripción:* Los tipos de paquetes que se reconocen es un paquete que indica que el compresor debe comenzar usando una referencia segura actualizada para ser capaz de reanudar el uso del tipo de paquete más pequeño; alternativamente, es un paquete que actualiza información distinta de información secuencial (es decir distinta de los SN, y otros campos comprimidos como función de los SN tales como un Sello de tiempo y un IP-ID).

Específicamente para la ROHC, la idea se puede resumir como sigue:

- 40 “Implementar el compresor de modo R de ROHC de manera que los tipos de paquetes R-0 y R-1^{*} no son usados (R-0-CRC puede ser usado en lugar de R-0, y UOR-2 en lugar de R-1^{*}). Implementar el descompresor del modo R de ROHC de manera que los paquetes R-0-CRC no son reconocidos nunca, y que otros paquetes son solamente reconocidos cuando el valor de SN descomprimido está fuera del intervalo de interpretación del LSB (W-LSB) para la referencia segura o si necesitan ser actualizados campos no secuenciales (es decir distintos de los SN, y otros campos comprimidos como función de los SN tales como un Sello de tiempo y un IP-ID).”

Lógica compresor-descompresor – intervalo de guarda de actualización de contexto

- 50 *Idea básica: Cuando se necesita actualizar el contexto para un cambio particular, el compresor envía un número X de paquetes de actualización de contexto consecutivos al menos igual al número máximo de reordenación de paquetes posible. El descompresor solamente reconoce el primer paquete de actualización de contexto de una secuencia (o asegura que el paquete reconocido fue seguido por X paquetes de actualización de contexto).*

- 55 *Descripción:* El descompresor puede asegurar que los paquetes que podrían ser reordenados de manera que puedan ser descomprimidos erróneamente, es decir basados en la referencia segura incorrecta, pueden ser detectados en el descompresor enviando un número de paquetes de actualización de contexto consecutivos al menos igual al número máximo de reordenación de paquetes posible; cuando se recibe un ACK para la referencia donde la cuenta comienza puede reanudar el envío de paquetes autocontenidos. El descompresor solamente reconoce la primera actualización de la secuencia (o asegura que el paquete reconocido fue seguido por un número suficiente de paquetes de actualización de contexto) y no los otros paquetes para esa secuencia (estos paquetes son salvaguardias frente a la reordenación).

Específicamente para la ROHC, la idea se puede resumir como sigue:

- 65 “Implementar el compresor de modo R de ROHC de manera que los tipos de paquetes R-0 y R-1^{*} no son usados para un intervalo al menos igual a la reordenación máxima posible que puede ocurrir a un paquete

5 entre el compresor y el descompresor cuando se realiza una actualización de contexto. Implementar el descompresor de modo R de ROHC de manera que se actualice la referencia segura y se envíe un reconocimiento para el primer paquete suponía actualizar un parámetro de contexto específico en una secuencia de paquetes de actualización de contexto (o el compresor asegura que el paquete reconocido fue seguido por un número suficiente de paquetes de actualización de contexto).”

10 Mientras que la invención ha sido descrita en conexión con lo que se considera actualmente que es la realización más práctica y preferida, se tiene que entender que la invención no tiene que estar limitada a la realización descrita, sino al contrario, se pretende que cubra diversas modificaciones y disposiciones equivalentes incluidas dentro del alcance de las reivindicaciones adjuntas. Especialmente, se debería señalar que incluso si los términos genéricos compresión de cabeceras, compresor de cabeceras y descompresor de cabeceras se usan para mostrar que la aplicabilidad de todas las ideas anteriores no está limitada a ningún esquema específico de compresión de cabeceras, aunque es ciertamente de más relevancia para [ROHC], [IP-ONLY] y [ROHC-UDPLite].

REIVINDICACIONES

- 5 1. Un método para un descompresor para mejorar el uso de un principio de referencia segura en un esquema de compresión de cabeceras sobre un canal que admite reordenación de paquetes entre un compresor y el descompresor, asegurando que no puede ocurrir una descompresión errónea de paquetes debido a la reordenación no detectada cuando se usa el principio de referencia segura, **caracterizado por** el paso de: verificar la exactitud de un número de paquetes de no actualización de contexto al menos igual a la reordenación máxima posible después de una actualización de contexto por medio de una suma de comprobación de la capa de transporte.
- 10 2. El método de la reivindicación 1, **caracterizado por** los pasos adicionales de:
- 15 detectar la descompresión errónea de paquetes de no actualización de contexto causados reordenando usando la suma de comprobación de la capa de transporte cuando está habilitado dicho principio de referencia segura, y aplicando la verificación de la suma de comprobación de la capa de transporte a un número de paquetes de no actualización de contexto al menos igual a la reordenación máxima posible después de que el contexto ha sido actualizado y se ha enviado el correspondiente reconocimiento.
- 20 3. El método de la reivindicación 1, **caracterizado por** los pasos adicionales de:
- evitar la descompresión errónea de paquetes de no actualización de contexto causada por dicha reordenación retardando los reconocimientos para actualizaciones de contexto con el propósito de forzar a dicho compresor en enviar paquetes de actualización de contexto para un número de paquetes al menos igual a la reordenación máxima posible que puede ocurrir en el canal entre dicho compresor y dicho descompresor.
- 25 4. El método de la reivindicación 3, **caracterizado por** los pasos adicionales de:
- 30 operar el descompresor en el modo R de ROCH, y nunca reconocer el tipo de paquete R-0-CRC, y retardar los reconocimientos para otros tipos de paquetes de actualización de contexto hasta que se han recibido un número de paquetes de actualización de contexto consecutivos al menos igual al retardo máximo posible debido a la reordenación.
- 35 5. El método de la reivindicación 1, **caracterizado por** los pasos adicionales de:
- realizar una actualización de contexto selectiva de dicho descompresor y enviar los reconocimientos correspondientes basados en semánticas particulares de un subconjunto específico de tipos de paquetes de actualización de contexto.
- 40 6. El método de la reivindicación 5, **caracterizado por** los pasos adicionales de:
- reconocer los paquetes que indican que el compresor debe comenzar usando una referencia segura actualizada para ser capaz de reanudar el uso del tipo de paquete más pequeño, o paquetes que actualizan información distinta de la información secuencial.
- 45 7. El método de la reivindicación 6, **caracterizado por** los pasos adicionales de:
- 50 operar el descompresor en el modo R de ROHC de modo que los paquetes R-0-CRC nunca son reconocidos, y que otros paquetes sean solamente reconocidos cuando el valor de SN descomprimido está fuera del intervalo de interpretación de LSB o W-LSB para la referencia segura o si necesitan ser actualizados los campos no secuenciales.
- 55 8. El método de la reivindicación 1, **caracterizado por** los pasos adicionales de:
- operar el descompresor en el modo R de ROHC de modo que la referencia segura se actualice y se envíe un reconocimiento para el primer paquete suponía actualizar un parámetro de contexto específico en una secuencia de paquetes de actualización de contexto, o el compresor asegura que el paquete reconocido fue seguido por un número de paquetes de actualización de contexto.
- 60 9. El método de cualquiera de las reivindicaciones 1 a 8, **caracterizado por** los pasos adicionales de: implementar dicho compresor y/o dicho descompresor según ROHC, IP-ONLY o ROHC-UDPLite.
- 65 10. Una disposición para un descompresor para mejorar el uso de un principio de referencia seguro en un esquema de compresión de cabeceras sobre un canal que admite una reordenación de paquetes entre un compresor y el descompresor, comprendiendo medios para asegurar que no puede ocurrir descompresión errónea de paquetes debido a la reordenación no detectada cuando se usa el principio de referencia segura, **caracterizada porque** dichos medios de aseguramiento están configurados para verificar la exactitud de un número de paquetes de no

actualización de contexto al menos igual a la reordenación máxima posible después de una actualización de contexto por medio de una suma de comprobación de la capa de transporte.

5 **11.** La disposición de la reivindicación 10, **caracterizada porque** dichos medios de aseguramiento están configurados para detectar una descompresión errónea de paquetes de no actualización de contexto causados reordenando por medio de la suma de comprobación de la capa de transporte cuando dicho principio de referencia segura está habilitado, y para aplicar la verificación de la suma de comprobación de la capa de transporte a un número de paquetes de no actualización de contexto al menos igual a la reordenación máxima posible después de que el contexto ha sido actualizado y el reconocimiento correspondiente ha sido enviado.

10 **12.** La disposición de la reivindicación 10, **caracterizada porque** dichos medios de aseguramiento están configurados para evitar una descompresión errónea de paquetes de no actualización de contexto causada por dicha reordenación retardando los reconocimientos para actualizaciones de contexto con el propósito de forzar a dicho compresor en enviar paquetes de actualización de contexto para un número de paquetes al menos igual a la reordenación máxima posible que puede ocurrir en el canal entre dicho compresor y dicho descompresor.

15 **13.** La disposición de la reivindicación 10, **caracterizada porque** dicho descompresor está configurado para realizar una actualización de contexto selectiva y para enviar los reconocimientos correspondientes basados en semánticas particulares de un subconjunto específico de tipos de paquetes de actualización de contexto.

20

Fig 1

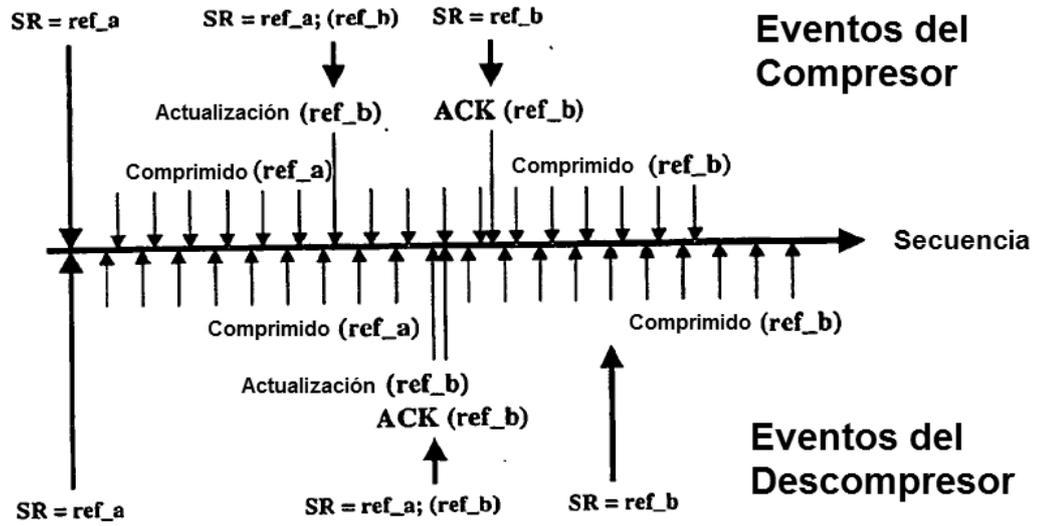


Fig 2

