

Descripción de los mecanismos de keepalive en el Cisco IOS

Contenido

[Introducción](#)

[Antecedentes](#)

[Mecanismos del keepalive de interfaz](#)

[Interfaces Ethernet](#)

[Interfaces en serie](#)

[Señales de mantenimiento de HDLC](#)

[Señales de mantenimiento de PPP](#)

[Interfaces de túnel GRE](#)

[Keepalives Crypto](#)

[Keepalives IKE](#)

[Keepalives NAT](#)

Introducción

Este documento describe los diversos mecanismos de keepalive en el [®] del Cisco IOS.

Antecedentes

Los mensajes de keepalive son enviados por un dispositivo de red vía una comprobación o un circuito virtual para todavía informar a otro dispositivo de red ese el circuito entre ellos las funciones. Para que el Keepalives trabaje hay dos factores esenciales:

- El intervalo de keepalive es el período de tiempo entre cada mensaje de keepalive que sea enviado por un dispositivo de red. Esto es siempre configurable.
- Las recomprobaciones del keepalive son la cantidad de veces que el dispositivo continúa enviando los paquetes de keepalive sin la respuesta antes de que el estado se cambie a “abajo”. Para algunos tipos de Keepalives esto es configurable, mientras que para otros hay un valor predeterminado que no puede ser cambiado.

Mecanismos del keepalive de interfaz

Interfaces Ethernet

En los medios de broadcast tales como un Ethernet, el Keepalives es levemente único. Puesto que hay muchos posibles vecinos en los Ethernets, el keepalive no se diseña para determinar si la trayectoria a cualquier un vecino particular en el alambre está disponible. Se diseña solamente para marcar que el sistema local tiene acceso de lectura y escritura al alambre de los Ethernets sí mismo. El router produce un paquete Ethernet consigo mismo como el MAC Address de origen y destino y código de tipo Ethernet especial de 0x9000. El hardware Ethernet envía este paquete sobre el alambre de los Ethernets y después recibe inmediatamente esta parte posterior del paquete otra vez. Esto marca el hardware de envío y de recepción en el adaptador Ethernet y la integridad básica del alambre.

Source MAC 00-00-0C-04-EF-04	Destination MAC 00-00-0C-04-EF-04	Protocol Type 9000	Data 0000 0100	Layer-2 Padding 0000 ... 0000
---------------------------------	--------------------------------------	-----------------------	-------------------	----------------------------------

Interfaces en serie

Las interfaces seriales pueden tener diversos tipos de encapsulaciones y cada tipo de encapsulación determina la clase de Keepalives que sea utilizado.

Ingrese el **comando keepalive** en el modo de configuración de la interfaz para fijar la frecuencia en la cual un router envía los paquetes ECHOREQ a su par:

- Para restablecer el sistema al intervalo del keepalive predeterminado de 10 segundos, ingrese el **comando keepalive** con la **ninguna** palabra clave.
- Para inhabilitar el Keepalives, ingrese el **comando disable del keepalive**.

Nota: **keepalive** El comando se aplica a las interfaces seriales que utilizan el link de datos de alto nivel Control (HDLC) o la encapsulación PPP. No se aplica a las interfaces seriales que utilizan la Encapsulación de Frame Relay.

Nota: Para el PPP y los tipos del encapsulado HDCL, un keepalive de cero inhabilita el Keepalives y está señalado en el **comando show running-config** hecho salir como **neutralización del keepalive**.

Señales de mantenimiento de HDLC

Otro mecanismo de keepalive bien conocido es señales de mantenimiento serial para el HDLC. Las señales de mantenimiento serial se envían hacia adelante y hacia atrás entre dos Routers y el Keepalives se reconocen. Con el uso de los números de secuencia de seguir cada keepalive, cada dispositivo puede confirmar si es par del HDLC recibió el keepalive que envió. Para el encapsulado HDCL, tres Keepalives ignorado hace la interfaz ser derribado.

Permita al **comando debug serial interface** para una conexión HDLC para permitir que el usuario vea el Keepalives se genera y se envía que:

Sample Output:

```
17:21:09.685: Serial0/0: HDLC myseq 0, mineseen 0*, yourseen 1, line up
```

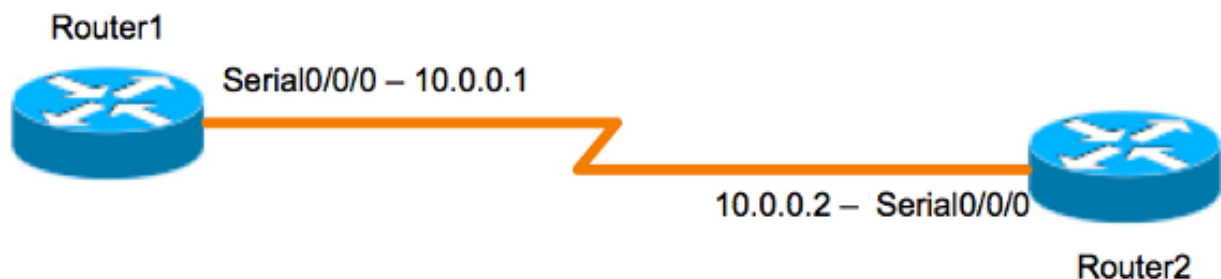
Las señales de mantenimiento de HDLC contienen tres pedazos para determinarlo trabajan:

- El “myseq” que es nuestro propio número que incrementa.
- El “mineseen” que es realmente un acuse de recibo del otro lado (incrementado) que lo dice cuenta con este número de nosotros.
- “Yourseen” que es nuestro acuse de recibo al otro lado.

Nota: Cuando la diferencia en los valores en el myseq y los campos del mineseen excede de tres en el router2, la línea va abajo y se reajusta la interfaz.

Puesto que las señales de mantenimiento de HDLC son Keepalives del tipo ECHOREQ, la frecuencia de keepalive es importante y se recomienda que hacen juego para arriba exactamente en los ambos lados. Si los temporizadores están fuera de sincronice, los números de secuencia comienzan a salir de la orden. Por ejemplo, si usted fija un lado a 10 segundos y el otro a 25 segundos, todavía permitirá que la interfaz permanezca para arriba mientras la diferencia en la frecuencia no sea suficiente hacer los números de secuencia estar apagados por una diferencia de tres.

Como un ejemplo de cómo las señales de mantenimiento de HDLC trabajan, el router1 y router2 están conectados directamente vía el Serial0/0 y el Serial2/0 respectivamente. Para ilustrar cómo el Keepalives fallado HDCL se utiliza para seguir los estados de la interfaz, el Serial0/0 será apagado en el router1.



Router 1

```
Router1#show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up (connected)
Hardware is HD64570
Internet address is 10.0.0.1/8
MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
[output is omitted]

17:21:09.685: Serial0/0: HDLC myseq 0, mineseen 0*, yourseen 1, line up
17:21:19.725: Serial0/0: HDLC myseq 1, mineseen 1*, yourseen 2, line up
17:21:29.753: Serial0/0: HDLC myseq 2, mineseen 2*, yourseen 3, line up
17:21:39.773: Serial0/0: HDLC myseq 3, mineseen 3*, yourseen 4, line up
17:21:49.805: Serial0/0: HDLC myseq 4, mineseen 4*, yourseen 5, line up
17:21:59.837: Serial0/0: HDLC myseq 5, mineseen 5*, yourseen 6, line up
17:22:09.865: Serial0/0: HDLC myseq 6, mineseen 6*, yourseen 7, line up
17:22:19.905: Serial0/0: HDLC myseq 7, mineseen 7*, yourseen 8, line up
17:22:29.945: Serial0/0: HDLC myseq 8, mineseen 8*, yourseen 9, line up
Router1 (config-if)#shut
17:22:39.965: Serial0/0: HDLC myseq 9, mineseen 9*, yourseen 10, line up
17:22:42.225: %LINK-5-CHANGED: Interface Serial0/0, changed state
to administratively down
```

```
17:22:43.245: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0,
changed state to down
```

Router 2

```
Router2#show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up (connected)
Hardware is HD64570
Internet address is 10.0.0.2/8
MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
[output is omitted]
```

```
17:21:04.929: Serial2/0: HDLC myseq 0, mineseen 0, yourseen 0, line up
17:21:14.941: Serial2/0: HDLC myseq 1, mineseen 1*, yourseen 1, line up
17:21:24.961: Serial2/0: HDLC myseq 2, mineseen 2*, yourseen 2, line up
17:21:34.981: Serial2/0: HDLC myseq 3, mineseen 3*, yourseen 3, line up
17:21:45.001: Serial2/0: HDLC myseq 4, mineseen 4*, yourseen 4, line up
17:21:55.021: Serial2/0: HDLC myseq 5, mineseen 5*, yourseen 5, line up
17:22:05.041: Serial2/0: HDLC myseq 6, mineseen 6*, yourseen 6, line up
17:22:15.061: Serial2/0: HDLC myseq 7, mineseen 7*, yourseen 7, line up
17:22:25.081: Serial2/0: HDLC myseq 8, mineseen 8*, yourseen 8, line up
17:22:35.101: Serial2/0: HDLC myseq 9, mineseen 9*, yourseen 9, line up
17:22:45.113: Serial2/0: HDLC myseq 10, mineseen 10*, yourseen 10, line up
17:22:55.133: Serial2/0: HDLC myseq 11, mineseen 10, yourseen 10, line up
17:23:05.153: HD(0): Reset from 0x203758
17:23:05.153: HD(0): Asserting DTR
17:23:05.153: HD(0): Asserting DTR and RTS
17:23:05.153: Serial2/0: HDLC myseq 12, mineseen 10, yourseen 10, line up
17:23:15.173: HD(0): Reset from 0x203758
17:23:15.173: HD(0): Asserting DTR
17:23:15.173: HD(0): Asserting DTR and RTS
17:23:15.173: Serial2/0: HDLC myseq 13, mineseen 10, yourseen 10, line down
17:23:16.201: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0,
changed state to down
Router2#
17:23:25.193: Serial2/0: HDLC myseq 14, mineseen 10, yourseen 10, line down
```

Señales de mantenimiento de PPP

Las señales de mantenimiento de PPP son un poco diferentes de las señales de mantenimiento de HDLC. A diferencia del HDLC, las señales de mantenimiento de PPP están más bien los ping. Los ambos lados pueden hacerse ping en su ocio. El movimiento negociado apropiado es responder SIEMPRE a este "ping". Tan para las señales de mantenimiento de PPP, la frecuencia o el valor del temporizador es solamente localmente relevante y no tiene ningún impacto en el otro lado. Incluso si un lado apaga el Keepalives, todavía RESPONDERÁ a esos pedidos de eco del lado que tiene un temporizador KEEPALIVE. Sin embargo, nunca iniciará ningunos sus los propio.

Permita al **comando debug ppp packet** para una conexión PPP para permitir que el usuario vea las señales de mantenimiento de PPP se envían que:

```
Router2#show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up (connected)
Hardware is HD64570
Internet address is 10.0.0.2/8
MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
[output is omitted]
```

```
17:21:04.929: Serial2/0: HDLC myseq 0, mineseen 0, yourseen 0, line up
17:21:14.941: Serial2/0: HDLC myseq 1, mineseen 1*, yourseen 1, line up
17:21:24.961: Serial2/0: HDLC myseq 2, mineseen 2*, yourseen 2, line up
17:21:34.981: Serial2/0: HDLC myseq 3, mineseen 3*, yourseen 3, line up
17:21:45.001: Serial2/0: HDLC myseq 4, mineseen 4*, yourseen 4, line up
17:21:55.021: Serial2/0: HDLC myseq 5, mineseen 5*, yourseen 5, line up
17:22:05.041: Serial2/0: HDLC myseq 6, mineseen 6*, yourseen 6, line up
17:22:15.061: Serial2/0: HDLC myseq 7, mineseen 7*, yourseen 7, line up
17:22:25.081: Serial2/0: HDLC myseq 8, mineseen 8*, yourseen 8, line up
17:22:35.101: Serial2/0: HDLC myseq 9, mineseen 9*, yourseen 9, line up
17:22:45.113: Serial2/0: HDLC myseq 10, mineseen 10*, yourseen 10, line up
17:22:55.133: Serial2/0: HDLC myseq 11, mineseen 10, yourseen 10, line up
17:23:05.153: HD(0): Reset from 0x203758
17:23:05.153: HD(0): Asserting DTR
17:23:05.153: HD(0): Asserting DTR and RTS
17:23:05.153: Serial2/0: HDLC myseq 12, mineseen 10, yourseen 10, line up
17:23:15.173: HD(0): Reset from 0x203758
17:23:15.173: HD(0): Asserting DTR
17:23:15.173: HD(0): Asserting DTR and RTS
17:23:15.173: Serial2/0: HDLC myseq 13, mineseen 10, yourseen 10, line down
17:23:16.201: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0,
changed state to down
Router2#
17:23:25.193: Serial2/0: HDLC myseq 14, mineseen 10, yourseen 10, line down
```

y respuestas se reciben que:

```
Router2#show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up (connected)
Hardware is HD64570
Internet address is 10.0.0.2/8
MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
[output is omitted]
```

```
17:21:04.929: Serial2/0: HDLC myseq 0, mineseen 0, yourseen 0, line up
17:21:14.941: Serial2/0: HDLC myseq 1, mineseen 1*, yourseen 1, line up
17:21:24.961: Serial2/0: HDLC myseq 2, mineseen 2*, yourseen 2, line up
17:21:34.981: Serial2/0: HDLC myseq 3, mineseen 3*, yourseen 3, line up
17:21:45.001: Serial2/0: HDLC myseq 4, mineseen 4*, yourseen 4, line up
17:21:55.021: Serial2/0: HDLC myseq 5, mineseen 5*, yourseen 5, line up
17:22:05.041: Serial2/0: HDLC myseq 6, mineseen 6*, yourseen 6, line up
17:22:15.061: Serial2/0: HDLC myseq 7, mineseen 7*, yourseen 7, line up
17:22:25.081: Serial2/0: HDLC myseq 8, mineseen 8*, yourseen 8, line up
17:22:35.101: Serial2/0: HDLC myseq 9, mineseen 9*, yourseen 9, line up
17:22:45.113: Serial2/0: HDLC myseq 10, mineseen 10*, yourseen 10, line up
17:22:55.133: Serial2/0: HDLC myseq 11, mineseen 10, yourseen 10, line up
17:23:05.153: HD(0): Reset from 0x203758
17:23:05.153: HD(0): Asserting DTR
17:23:05.153: HD(0): Asserting DTR and RTS
17:23:05.153: Serial2/0: HDLC myseq 12, mineseen 10, yourseen 10, line up
17:23:15.173: HD(0): Reset from 0x203758
17:23:15.173: HD(0): Asserting DTR
17:23:15.173: HD(0): Asserting DTR and RTS
17:23:15.173: Serial2/0: HDLC myseq 13, mineseen 10, yourseen 10, line down
17:23:16.201: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0,
changed state to down
Router2#
17:23:25.193: Serial2/0: HDLC myseq 14, mineseen 10, yourseen 10, line down
```

Las señales de mantenimiento de PPP contienen tres pedazos:

- Número de ID - usado para identificar al cual el ECHOREQ el par responde.
- Tipo de keepalive - El ECHOREQ es Keepalives enviado por el dispositivo de origen y el ECHOREP es respuestas enviadas por el par.
- Números mágicos - las notificaciones incluyen los números mágicos del servidor y del cliente remoto. El par valida el número mágico en el paquete de pedido de eco LCP, y transmite el paquete de respuesta de eco correspondiente LCP que contiene el número mágico negociado por el router.

Para la encapsulación PPP, cinco Keepalives ignorado hace la interfaz ser derribado

Interfaces de túnel GRE

El mecanismo de keepalive del túnel GRE es levemente diferente que para los Ethernetes o las interfaces seriales. Da la capacidad para que un lado origine y reciba los paquetes de keepalive a y desde un router remoto incluso si el router remoto no soporta las Señales de mantenimiento GRE. Puesto que el GRE es un mecanismo de tunelización del paquete para hacer un túnel el IP dentro del IP, un paquete del túnel IP GRE se puede construir dentro de otro paquete del túnel IP GRE. Para las Señales de mantenimiento GRE, las PRE-estructuras del remitente el paquete de la respuesta de keepalive dentro del paquete de pedidos del keepalive original de modo que las necesidades del extremo remoto solamente de hacer la descapsulación estándar GRE del encabezado IP externo GRE y entonces remiten el Paquete GRE interno IP. Este mecanismo hace la respuesta de keepalive remitir hacia fuera la interfaz física bastante que la interfaz del túnel. Para más detalles en el trabajo del Keepalives del túnel GRE, vea [cómo las Señales de mantenimiento GRE trabajan](#).

Keepalives Crypto

Keepalives IKE

El Keepalives del Internet Key Exchange (IKE) es un mecanismo usado para determinar si un par VPN puede ascendente y recibir el tráfico encriptado. El Keepalives crypto separado se requiere además de los keepalives de interfaz porque los pares VPN generalmente nunca están conectados de nuevo a la parte posterior, así que los keepalives de interfaz no proporcionan bastante información sobre el estado del par VPN.

En los dispositivos Cisco IOS, los keepalives IKE son habilitados por el uso de un método propietario llamado el Dead Peer Detection (DPD). Para permitir que el gateway envíe los DPD al par, ingrese este comando en el modo de configuración global:

```
crypto isakmp keepalive seconds [retry-seconds] [ periodic | on-demand ]
```

Para inhabilitar el Keepalives, utilice la forma del "no" de este comando. Para más información sobre lo que hace cada palabra clave en este comando, vea el [keepalive crypto del isakmp](#). Para más granularidad, el Keepalives se puede también configurar bajo perfil ISAKMP. Para más detalles, vea el [\[Cisco IOS IPsec\] de la descripción del perfil ISAKMP](#).

Keepalives NAT

En caso de los escenarios donde está un par VPN detrás de un Network Address Translation (NAT), el NAT-Traversal se utiliza para el cifrado. Sin embargo, durante los períodos inactivos es posible que la entrada de NAT en el dispositivo ascendente pudo medir el tiempo hacia fuera. Esto puede causar los problemas cuando usted trae para arriba el túnel y el NAT no es bidireccional. El Keepalives NAT se habilita para mantener asociar dinámico NAT vivo durante una conexión entre dos pares. El Keepalives NAT es paquetes UDP con un payload unencrypted de un byte. Aunque la implementación actual DPD sea similar al Keepalives NAT, hay una leve diferencia - el DPD se utiliza para detectar el estado de peer mientras que se envía el Keepalives NAT si la entidad del IPSec no envió ni recibió el paquete en un período especificado. El intervalo válido es entre 5 a 3600 segundos.

Consejo: Si se habilita el Keepalives NAT (a través del **comando `keepalive nacional del isamkp crypto`**), los usuarios deben asegurarse de que el valor ocioso sea más corto que el vencimiento de la asignación NAT de 20 segundos.

Para más información sobre esta característica, vea la [Transparencia NAT del IPSec](#).