

Información importante sobre los Comandos de depuración

Contenido

[Introducción](#)

[prerrequisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Advertencias](#)

[Convenciones](#)

[Antes de la depuración](#)

[Cómo obtener las salidas de los depuradores](#)

[Otras tareas previas a la depuración](#)

[Para detener la depuración](#)

[Uso del comando debug ip packet](#)

[Depuraciones accionadas de manera condicional](#)

[Información Relacionada](#)

[Introducción](#)

Esta página proporciona algunas directrices generales sobre el uso de los debugs disponibles en las plataformas de Cisco IOS®, así como ejemplos del uso correcto del comando **debug ip packet** y el debugging condicional.

Nota: Este documento no explica cómo utilizar e interpretar los comandos y salidas **debug** específicos. Refiérase a la documentación apropiada de Referencia de Comandos Debug de Cisco para obtener información sobre los comandos **debug** específicos.

La salida de los **comandos debug privileged exec** proporciona la información de diagnóstico que incluyen una variedad de acontecimientos de funcionamiento entre redes referentes el estado del protocolo y a la actividad de la red en general.

[prerrequisitos](#)

[Requisitos](#)

Cisco recomienda que tenga conocimiento sobre estos temas:

- Conexión al router utilizando los puertos de consola, auxiliares y vty
- Problemas generales de la configuración del Cisco IOS
- Interpretación de las salidas de los debugs del Cisco IOS

Componentes Utilizados

Este documento no tiene restricciones específicas en cuanto a versiones de software y de hardware.

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si la red está funcionando, asegúrese de haber comprendido el impacto que puede tener cualquier comando.

Advertencias

Utilice los comandos de depuración con precaución. En general, se recomienda que estos comandos se utilicen sólo bajo la dirección del representante de soporte técnico de su router cuando se intenta resolver problemas específicos.

La habilitación de la depuración puede interrumpir el funcionamiento del router cuando las redes interconectadas experimentan condiciones de carga elevadas. Por lo tanto, si se habilita la registración, el servidor de acceso puede congelar intermitentemente tan pronto como el puerto de la consola consiga sobrecargado con los mensajes del registro.

Antes de que usted comience un **comando debug**, considere siempre la salida que este comando generará y la cantidad de tiempo esto puede tomar. Por ejemplo, si usted tiene un router con un Basic Rate Interface (BRI), el **debug ISDN q931** no dañará probablemente el sistema. Pero, hacer el mismo debug en un AS5800 con la configuración completa del e1 puede generar probablemente tanto la entrada que puede colgar y parar el responder.

Antes de hacer el debug de, mire su carga de la CPU con el **comando show processes cpu**. Verifique que usted tenga CPU suficiente disponible antes de que usted comience los debugs. Refiera a [resolver problemas CPU elevada la utilización en los routers Cisco](#) para más información sobre cómo manejar CPU elevada las cargas. Por ejemplo, si usted tiene un Cisco 7200 Router con una interfaz ATM que hace interligar entonces, dependiendo de la cantidad de subinterfaces configuradas, recomenzando el router pudo utilizar mucho su CPU. La causa aquí es que, para cada circuito virtual (VC), se necesita generar un paquete de unidad de datos del protocolo de puente (BPDU). Comenzar con las depuraciones en momentos tan críticos puede hacer que aumente drásticamente la utilización del CPU y resulte en un bloqueo o en una pérdida de conectividad de red.

Nota: Cuando los debugs se están ejecutando, usted no ve generalmente el prompt de router, especialmente cuando el debug es intensivo. Pero, en la mayoría de los casos, usted puede utilizar los **comandos no debug all or undebg all** para parar los debugs. Refiérase a la sección [Obtención de Salidas de Debug](#) para obtener más información sobre el uso seguro de los debugs

Convenciones

Consulte [Convenciones de Consejos TécnicosCisco](#) para obtener más información sobre las convenciones del documento.

Antes de la depuración

Además de los puntos mencionados arriba, asegúrese de comprender el impacto de las

depuraciones en la estabilidad de la plataforma. Debería considerar también a qué interfaz del router debe conectarse. Esta sección tiene algunas guías de consulta.

[Cómo obtener las salidas de los depuradores](#)

Los routers pueden mostrar resultados de depuración para diversas interfaces, dentro de los cuales se encuentran los puertos de consola, auxiliares y vty. Los routers también pueden registrar mensajes a un búfer interno, a un servidor syslog unix externo. Las instrucciones y advertencias para cada método se detallan a continuación:

Puerto de consola

Si usted está conectado en la consola, bajo configuraciones normales, ningún trabajo adicional necesita ser hecho. La salida de debug debería mostrarse automáticamente. Pero, asegúrese el **nivel de consola de registro** se fija como deseado y ese registro no se ha inhabilitado con el **comando no logging console**. Refiérase a [Uso de los Comandos Debug](#) para obtener más información.

Advertencia: Los debugs excesivos al puerto de consola de un router pueden hacer que se bloquee. Esto se debe a que el router da prioridad automáticamente a la salida de la consola por encima de otras funciones del router. Por lo tanto, si el router procesa una salida grande del depurador al puerto de la consola, puede bloquearse. Por lo tanto, si el resultado de la depuración es excesivo, utilice los puertos vty (telnet) o las memorias intermedias de registro para obtener sus depuraciones. Se proporciona más información a continuación:

Nota: De forma predeterminada, el registro está habilitado en el puerto de consola. Por lo tanto, el puerto de la consola siempre procesa el resultado de la depuración aunque se estuviese utilizando otro puerto u método (como Aux, vty o búfer) para capturar el resultado. Por lo tanto, Cisco recomienda que, bajo condiciones normales de funcionamiento, usted tiene el **comando no logging console** habilitado siempre y utiliza otros métodos para capturar los debugs. En situaciones en las que necesite usar la consola, vuelva a activar temporalmente el registro de consola.

Puerto auxiliar

Si usted está conectado vía un puerto auxiliar, teclee el **comando terminal monitor**. Verifique además que el comando no logging on no haya sido activado en el router.

Nota: Si utiliza el puerto auxiliar para monitorear el router, tenga presente que, cuando el router se reinicia, el puerto auxiliar no muestra la salida de la secuencia de inicio. Conecte con el puerto de la consola para ver la secuencia de arranque.

Puertos VTY

Si usted está conectado vía un puerto auxiliar o vía el telnet, teclee el **comando terminal monitor**. Verifique además que el comando no logging on no se haya utilizado.

Registro de mensajes en una memoria intermedia interna

El dispositivo de registro predeterminado es la consola; todos los mensajes se muestran en la

consola, a menos que se indique lo contrario.

Para registrar mensajes en una memoria intermedia interna, utilice el comando de configuración del router logging buffered. Ésta es la sintaxis completa de este comando:

```
logging buffered no logging buffered
```

El comando **logging buffered** copia los mensajes de registro en un buffer interno, en vez de escribirlos en la consola. El búfer es originalmente circular, por lo que los mensajes nuevos sobrescriben a los anteriores. Para mostrar los mensajes que se registran en el búfer, utilice el comando EXEC privilegiado show logging. El primer mensaje mostrado es el más antiguo del buffer. Puede especificar el tamaño del búfer así como el nivel de gravedad de los mensajes que ingresarán al registro.

Consejo: Asegúrese de que hay suficiente memoria disponible en el cuadro antes de ingresar el tamaño del buffer. Utilice el **comando show proc mem** del Cisco IOS para ver la memoria disponible.

El comando no logging buffered cancela el uso de la memoria intermedia y manda mensajes a la consola (predeterminada).

Mensajes de registro a un servidor UNIX Syslog.

Para registrar mensajes en el host del servidor Syslog, utilice el comando logging router configuration. La sintaxis completa de este comando es la siguiente:

```
logging <ip-address> no logging <ip-address>
```

El comando logging identifica un host del servidor syslog para recibir los mensajes de registro. El argumento < **ip-address**> es la dirección IP del host. Si se ejecuta este comando más de una vez, se crea una lista de servidores Syslog que reciben los mensajes de registro.

El comando no logging borra el servidor syslog con la dirección especificada de la lista de syslogs.

Para obtener más información sobre la configuración de un servidor Syslog, refiérase a [Uso de los Comandos Debug](#).

[Otras tareas previas a la depuración](#)

1. Configure su software de emulador de terminal (por ejemplo, HyperTerminal) de modo que pueda capturar la salida de debug a un archivo. Por ejemplo, en HyperTerminal, haga clic en **Transfer** y después en Capture Text, y elija las opciones adecuadas. Para obtener más información, refiérase a [Captura de la Salida de Texto desde Hyper Terminal](#). Para otro software de emulador de terminal, refiérase a la documentación del software.
2. Habilitar el sello de fecha/hora en milisegundos (mseg) utilizando el comando service timestamps:

```
router(config)#service timestamps debug datetime msec router(config)#service timestamps log datetime msec
```

Estos comandos añaden sellos de fecha/hora a los debugs en el formato MMM DD HH:

MILÍMETRO: SS, indicando la fecha y hora según el reloj del sistema. Si el reloj del sistema no ha sido configurado, un asterisco (*) precede la fecha y hora para indicar que es probable que estos datos no sean correctos.

Por lo general, es aconsejable configurar sellos de hora en milisegundos, ya que esto brinda un nivel alto de claridad al analizar los resultados de la depuración. Los sellos de fecha y hora en milisegundos proporcionan una mejor indicación de la sincronización de los diversos eventos de debug entre sí. Sin embargo, observe que, cuando el puerto de consola genera muchos mensajes, puede que éstos no se correlacionen con la temporización real del evento. Por ejemplo, si usted habilita el **debug x25 all** en un cuadro que tenga 200 VCs, y la salida se registra al buffer (usando los **comandos no logging console and logging buffered**), el grupo fecha/hora visualizado en la salida de los debugs (dentro del buffer) no pudo ser la hora exacta en que el paquete pasa a través de la interfaz. Por lo tanto, no utilice sellos de hora mseg para demostrar problemas de rendimiento sino para obtener información relativa acerca de cuándo tienen lugar los eventos.

Para detener la depuración

Para parar un debug, utilice los comandos **no debug all** o **undebug all**. Verifique que los debugs se han desactivado usando el comando **show debug**.

Recuerde que los comandos **no logging console** y **terminal** no monitor solamente evitan que se genere la salida desde el puerto de controla, auxiliar o vty respectivamente. No detienen la depuración y por lo tanto agotan los recursos de router.

Uso del comando debug ip packet

El comando **debug ip packet** produce información acerca de los paquetes que no son conmutados rápidamente por el router. Sin embargo, como genera un resultado para cada paquete, el resultado puede ser amplio y, por lo tanto, ocasionar el bloqueo del router. Por este motivo, sólo utilice el paquete IP de depuración bajo los más estrictos controles como se describe en esta sección.

La mejor forma de limitar la salida de **debug ip packet** es crear una lista de acceso vinculada a la depuración. Solamente los paquetes que coinciden con los criterios de la lista de acceso estarán sujetos al comando **debug ip packet**. Esta lista de acceso no necesita aplicarse en ninguna interfaz, sino que se aplica a la operación de debugging.

Antes de utilizar el paquete de depuración de ip, observe que el router está realizando la conmutación rápida predeterminada, o quizás la conmutación CEF, si fue configurado para hacerlo. Esto quiere decir que, una vez que esas técnicas están en su lugar, no se envía el paquete al procesador y, por lo tanto, la depuración no muestra nada. Para que funcione, necesita inhabilitar el fast switching en el router con el comando **no ip route-cache** (para los paquetes unicast) o **no ip mroute-cache** (para los paquetes multicast). Esto debería aplicarse en las interfaces en las que se supone que el tráfico fluye. Verifíquelo con el comando **show ip route**.

Advertencias:

- La inhabilitación de la conmutación rápida en un router que administra una gran cantidad de paquetes puede causar picos de uso de la CPU de modo que el equipo se cuelgue o pierda la conexión con sus pares.
- No inhabilite la opción de conmutación rápida en un router que esté ejecutando la Conmutación de protocolo de identificación múltiple (MPLS). MPLS se utiliza junto con CEF. Por lo tanto, desactivar la conmutación rápida en la interfaz puede tener efectos desastrosos.

Consideremos un escenario de ejemplo:

La Lista de acceso configurada en el router_122 es:

```
access-list 105 permit icmp host 10.10.10.2 host 13.1.1.1 access-list 105 permit icmp host 13.1.1.1 host 10.10.10.2
```

Esta lista de acceso permite cualquier paquete Internet Control Message Protocol (ICMP) del router del host_121 (con la dirección IP 10.10.10.2) al router del host_123 (con la dirección IP 13.1.1.1), así como en la dirección contraria. Es importante que permita los paquetes en ambas direcciones, de lo contrario, el router puede perder el paquete ICMP que está regresando.

Quite el Fast-Switching en solamente una interfaz en el router_122. Esto significa que usted puede ver solamente los debugs para los paquetes que son destinados para esa interfaz, según lo visto desde la perspectiva del IOS que intercepta el paquete. De los debugs, tales paquetes aparecen con el "d=". Puesto que usted todavía no ha dado vuelta apagado rápidamente a conmutar en la otra interfaz, el paquete de devolución no está conforme al **paquete del IP del debug**. Esta salida muestra cómo usted puede inhabilitar la transferencia rápida:

```
router_122(config)#interface virtual-template 1 router_122(config-if)#no ip route-cache
router_122(config-if)#end
```

Usted debe ahora activar el **paquete del IP del debug** con la lista de acceso definida anterior (lista de acceso 105).

```
router_122#debug ip packet detail 105 IP packet debugging is on (detailed) for access list 105
router_122# 00:10:01: IP: s=13.1.1.1 (Serial3/0), d=10.10.10.2 (Virtual-Access1), g=10.10.10.2,
len 100, forward 00:10:01: ICMP type=0, code=0 ! -- ICMP packet from 13.1.1.1 to 10.10.10.2. ! -
- This packet is displayed because it matches the ! -- source and destination requirements in
access list 105 00:10:01: IP: s=13.1.1.1 (Serial3/0), d=10.10.10.2 (Virtual-Access1),
g=10.10.10.2, len 100, forward 00:10:01: ICMP type=0, code=0 00:10:01: IP: s=13.1.1.1
(Serial3/0), d=10.10.10.2 (Virtual-Access1), g=10.10.10.2, len 100, forward 00:10:01: ICMP
type=0, code=0
```

Ahora, retiremos la conmutación rápida de la otra interfaz (en el router_122). Esto significa que todos los paquetes entre esas dos interfaces serán ahora conmutados por paquetes (lo cual es un requisito para el debug ip packet):

```
router_122(config)#interface serial 3/0 router_122(config-if)#no ip route-cache
router_122(config-if)#end router_122# 00:11:57: IP: s=10.10.10.2 (Virtual-Access1), d=13.1.1.1
(Serial3/0), g=172.16.1.6, len 100, forward 00:11:57: ICMP type=8, code=0 ! -- ICMP packet
(echo) from 10.10.10.2 to 13.1.1.1 00:11:57: IP: s=13.1.1.1 (Serial3/0), d=10.10.10.2 (Virtual-
Access1), g=10.10.10.2, len 100, forward 00:11:57: ICMP type=0, code=0 ! -- ICMP return packet
(echo-reply) from 13.1.1.1 to 10.10.10.2 00:11:57: IP: s=10.10.10.2 (Virtual-Access1),
d=13.1.1.1 (Serial3/0), g=172.16.1.6, len 100, forward 00:11:57: ICMP type=8, code=0 00:11:57:
IP: s=13.1.1.1 (Serial3/0), d=10.10.10.2 (Virtual-Access1), g=10.10.10.2, len 100, forward
00:11:57: ICMP type=0, code=0
```

Observe que la salida de debug ip packet no muestra ningún paquete que no coincida con los criterios de la lista de acceso. Para obtener más información sobre este procedimiento, refiérase a [Comprensión de los Comandos Ping y Traceroute](#).

Para obtener más información sobre cómo crear listas de acceso, refiérase a [Registro de Lista de Acceso IP Estándar](#).

[Depuraciones accionadas de manera condicional](#)

Cuando se habilita la función Conditionally Triggered Debugging, el router genera mensajes de

debugging para los paquetes que entran o salen del router en una interfaz especificada; el router no genera un resultado de la depuración para los paquetes que entran o salen por una interfaz diferente. Para obtener información con respecto a los usos de los debugs condicionales, refiérase a [Conditionally Triggered Debugging](#).

Mire una implementación sencilla de los debugs condicionales. Considere este escenario: el router mostrado abajo (traxbol) tiene dos interfaces (el serial0 y el serial 3) ambos encapsulado HDLC que se ejecuta.

Usted puede utilizar el comando normal debug serial interface para observar las señales de mantenimiento de HDLC recibidas en todas las interfaces. Usted puede observar el Keepalives en ambas interfaces.

```
traxbol#debug serial interface Serial network interface debugging is on traxbol# *Mar 8
09:42:34.851: Serial0: HDLC myseq 28, mineseen 28*, yourseen 41, line up ! -- HDLC keepalive on
interface Serial 0 *Mar 8 09:42:34.855: Serial3: HDLC myseq 26, mineseen 26*, yourseen 27, line
up ! -- HDLC keepalive on interface Serial 3 *Mar 8 09:42:44.851: Serial0: HDLC myseq 29,
mineseen 29*, yourseen 42, line up *Mar 8 09:42:44.855: Serial3: HDLC myseq 27, mineseen 27*,
yourseen 28, line up
```

Debugs condicionales para la interfaz seriales 3. del permiso. Esto significa que solamente se muestran los debugs para la interfaz serial 3. Utilice el comando **debug interface <interface_type interface_number>**.

```
traxbol#debug interface serial 3 Condition 1 set
```

Utilice el comando **show debug condition** para verificar que el debug del condicional es activo. Tome en cuenta que está activa una condición para la interfaz serial.

```
traxbol#show debug condition Condition 1: interface Se3 (1 flags triggered) Flags: Se3 traxbol#
```

Tenga en cuenta que sólo se muestran las depuraciones para la interfaz de serie 3.

```
*Mar 8 09:43:04.855: Serial3: HDLC myseq 29, mineseen 29*, yourseen 30, line up *Mar 8
09:43:14.855: Serial3: HDLC myseq 30, mineseen 30*, yourseen 31, line up
```

Utilice el comando **undebug interface <interface_type interface_number>** para quitar el debug condicional. Se recomienda que usted apaga los debugs (por ejemplo, usando el undebug todo) antes de que usted quite el activador condicional. Esto es así para evitar un aluvión de salidas de depuración cuando se elimina la condición.

```
traxbol#undebug interface serial 3 This condition is the last interface condition set. Removing
all conditions may cause a flood of debugging messages to result, unless specific debugging
flags are first removed. Proceed with removal? [yes/no]: y Condition 1 has been removed traxbol
```

Usted puede ahora observar que el debug para ambo interface serial 0 así como el serial 3 están visualizados.

```
*Mar 8 09:43:34.927: Serial3: HDLC myseq 32, mineseen 32*, yourseen 33, line up *Mar 8
09:43:44.923: Serial0: HDLC myseq 35, mineseen 35*, yourseen 48, line up
```

Advertencia: Algunas operaciones de debugging son condicionales en sí mismas. Un ejemplo es debugging atm. Con depuración ATM debe especificar explícitamente la interfaz para la cual las depuraciones se deben habilitar en vez de permitir las depuraciones en todas las interfaces atm y especificar una condición.

Esta sección muestra la manera correcta de limitar el paquete ATM que hace el debug de a una subinterfaz:

```
arielle-nrp2#debug atm packet interface atm 0/0/0.1 !--- Note that you explicitly specify the
sub-interface to be used for debugging ATM packets debugging is on Displaying packets on
```

```

interface ATM0/0/0.1 only arielle-nrp2# *Dec 21 10:16:51.891: ATM0/0/0.1(O): VCD:0x1 VPI:0x1
VCI:0x21 DM:0x100 SAP:AAAA CTL:03 OUI:0080C2 TYPE:0007 Length:0x278 *Dec 21 10:16:51.891: 0000
FFFF FFFF FFFF 0010 7BB9 BDC4 0800 4500 025C 01FE 0000 FF11 61C8 0A30 *Dec 21 10:16:51.891: 4B9B
FFFF FFFF 0044 0043 0248 0000 0101 0600 0015 23B7 0000 8000 0000 0000 *Dec 21 10:16:51.891: 0000
0000 0000 0000 0000 0000 0010 7BB9 BDC3 0000 0000 0000 0000 0000 0000 *Dec 21 10:16:51.891: 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 *Dec 21 10:16:51.891: 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 *Dec 21 10:16:51.891: 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 *Dec 21 10:16:51.895: 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 *Dec 21 10:16:51.895:
arielle-nrp2#

```

Si trata de habilitar la depuración ATM en todas las interfaces (con una condición aplicada), el router puede bloquearse si tiene un gran número de subinterfaces ATM. Se muestra un ejemplo del método incorrecto para depurar el atm.

En este caso, puede ver que se aplica una condición, pero también que ésta no tiene ningún efecto. Usted puede todavía ver el paquete de la otra interfaz. En este escenario de laboratorio usted tiene solamente dos interfaces y mismo pocos tráfico. Si el número de interfaces es alto, después la salida de los debugs para todas las interfaces es extremadamente alta y puede hacer al router colgar.

```

arielle-nrp2#show debugging condition Condition 1: interface AT0/0/0.1 (1 flags triggered)
Flags: AT0/0/0.1 ! -- A condition for a specific interface. arielle-nrp2#debug atm packet ATM
packets debugging is on Displaying all ATM packets arielle-nrp2# *Dec 21 10:22:06.727:
ATM0/0/0.2(O): ! -- You see debugs from interface ATM0/0/0.2, even though the condition ! --
specified ONLY AT0/0/0.1 VCD:0x2 VPI:0x5 VCI:0x37 DM:0x100 SAP:AAAA CTL:03 OUI:0080C2 TYPE:000E
Length:0x2F *Dec 21 10:22:06.727: 0000 0000 0180 0000 107B B9BD C400 0000 0080 0000 107B B9BD
C480 0800 0014 *Dec 21 10:22:06.727: 0002 000F 0000 *Dec 21 10:22:06.727: un a *Dec 21
10:22:08.727: ATM0/0/0.2(O): VCD:0x2 VPI:0x5 VCI:0x37 DM:0x100 SAP:AAAA CTL:03 OUI:0080C2
TYPE:000E Length:0x2F *Dec 21 10:22:08.727: 0000 0000 0180 0000 107B B9BD C400 0000 0080 0000
107B B9BD C480 0800 0014 *Dec 21 10:22:08.727: 0002 000F 0000 *Dec 21 10:22:08.727: 11 *Dec 21
10:22:10.727: ATM0/0/0.2(O): VCD:0x2 VPI:0x5 VCI:0x37 DM:0x100 SAP:AAAA CTL:03 OUI:0080C2
TYPE:000E Length:0x2F *Dec 21 10:22:10.727: 0000 0000 0080 0000 107B B9BD C400 0000 0080 0000
107B B9BD C480 0800 0014 *Dec 21 10:22:10.727: 0002 000F 0000 *Dec 21 10:22:10.727: *Dec 21
10:22:12.727: ATM0/0/0.2(O): VCD:0x2 VPI:0x5 VCI:0x37 DM:0x100 SAP:AAAA CTL:03 OUI:0080C2
TYPE:000E Length:0x2F *Dec 21 10:22:12.727: 0000 0000 0080 0000 107B B9BD C400 0000 0080 0000
107B B9BD C480 0800 0014 *Dec 21 10:22:12.727: 0002 000F 0000 *Dec 21 10:22:12.727: *Dec 21
10:22:13.931: ATM0/0/0.1(O): !--- You also see debugs for interface ATM0/0/0.1 as you wanted.
VCD:0x1 VPI:0x1 VCI:0x21 DM:0x100 SAP:AAAA CTL:03 OUI:0080C2 TYPE:0007 Length:0x278 *Dec 21
10:22:13.931: 0000 FFFF FFFF FFFF 0010 7BB9 BDC4 0800 4500 025C 027F 0000 FF11 6147 0A30 *Dec 21
10:22:13.931: 4B9B FFFF FFFF 0044 0043 0248 0000 0101 0600 001A 4481 0000 8000 0000 0000 *Dec 21
10:22:13.931: 0000 0000 0000 0000 0000 0000 0010 7BB9 BDC3 0000 0000 0000 0000 0000 *Dec 21
10:22:13.931: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 *Dec 21
10:22:13.931: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 *Dec 21
10:22:13.931: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 *Dec 21
10:22:13.935: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

```

[Información Relacionada](#)

- [Soporte de Tecnología de Discado y Acceso](#)
- [Soporte Técnico y Documentación - Cisco Systems](#)