

# Configurar el servicio repetido MS entre un router y un PC de Windows

## Contenido

[Introducción](#)

[prerrequisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenciones](#)

[Teoría Precedente](#)

[Configurar](#)

[Resumen de la configuración](#)

[Diagrama de la red](#)

[Configuraciones](#)

[Configuración de cliente de Windows](#)

[Configuración del cliente de Windows 95 y 98](#)

[Configuración del cliente de Windows NT y 2000](#)

[Verificación](#)

[Troubleshooting](#)

[Comandos para resolución de problemas](#)

[Información Relacionada](#)

## Introducción

La implementación de la devolución de llamada de Microsoft no es obediente con el [RFC 1570](#) . Sin embargo, debido a la participación en el mercado grande del cliente del dial-up networking de Microsoft, Cisco ha implementado el Control Protocol de la devolución de llamada de Microsoft (MSCB) en el Software Release 11.3(2)T y Posterior de Cisco IOS®.

## prerrequisitos

### Requisitos

Antes de utilizar esta configuración, asegúrese de que cumple con los siguientes requisitos:

- Configure al servidor de acceso a la red (NAS) para validar las llamadas analógicas del cliente. El servicio repetido es una característica adicional del dial-in del módem. Por lo tanto, verifique si funciona este aspecto correctamente. Esto puede ayudarle a resolver problemas.
- El circuito T1/E1 debe ser capaz del dialout. Entre en contacto su compañía telefónica (compañía telefónica) para verificar esto.

## Componentes Utilizados

La información en este documento se basa en las versiones de Cisco IOS Software Release 11.3(2)T y Posterior.

Este escenario fue probado en un PC con el dial-up networking de Windows.

La información que se presenta en este documento se originó a partir de dispositivos dentro de 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 un comando antes de ejecutarlo.

## Convenciones

Para obtener más información sobre las convenciones del documento, consulte [Convenciones de Consejos Técnicos de Cisco](#).

## Teoría Precedente

El servicio repetido ejecuta en esta orden:

1. A usuario de la PC (cliente) conecta con el Cisco Access Server.
2. El proceso de devolución de llamada se negocia en la fase del (LCP) del Link Control Protocol del Point-to-Point Protocol (PPP).
3. Se realiza la autenticación PPP.
4. El Cisco IOS Software valida las reglas del servicio repetido para este usuario o línea y desconecta persona que llama para devolver la llamada.
5. El Cisco Access Server marca al cliente.

Hay cuatro tipos de MSCB:

1. Ningún servicio repetido.
2. Número de devolución de llamada especificada por el usuario.
3. número de devolución de llamada (preconfigurado) Servidor-especificado.
4. Lista de número de devolución de llamada preconfigurado.

La configuración predeterminada no es ningún servicio repetido (opción 1). Las opciones 2 o 3 pueden ser configuradas:

- Localmente (si no se utiliza a ningún servidor de AAA).
- En el TACACS+ o el perfil del usuario de RADIUS (si se utiliza el AAA).

Si se configura la opción 2, se indica al usuario que ingrese su número de devolución de llamada. Si se configura el option 3, el prompt ofrece solamente una opción, que es el número administrador-definido.

Cisco implementa solamente las funciones del servidor de devolución de llamada del MSCB y no de las funciones del cliente de servicio de devolución de llamadas. Esto significa que un router Cisco puede ser utilizado solamente como servidor MSCB y no como cliente MSCB. Además, la implementación de Cisco del MSCB requiere la autenticación ser realizada en el cliente.

## [Configurar](#)

En esta sección encontrará la información para configurar las funciones descritas en este documento.

### [Resumen de la configuración](#)

Para habilitar el MSCB, usted debe habilitar el **comando ppp callback accept** bajo interfaz de recepción (por ejemplo, grupo async). Además, porque se requiere la autenticación, usted debe autenticación del protocolo de autenticación (PAP) o del Challenge Handshake Authentication Protocol (CHAP) de la contraseña habilitada:

```
ppp authentication chap pap
```

Dos secuencias de comandos de conversación se crean automáticamente. Éstas son el **offhook** y las secuencias de comandos de conversación del **servicio repetido**:

```
chat-script offhook "" "ATH1" OK
chat-script callback ABORT ERROR ABORT BUSY ""
"ATZ" OK "ATDT \T" TIMEOUT60 CONNECT \c
```

Las secuencias de comandos de conversación también se aplican automáticamente a las líneas funcionando:

```
line 1 24
  script modem-off-hook offhook
  script callback callback
```

Un usuario debe **ser autorizado** a ser llamado detrás. Usted puede configurar esto localmente en el NAS o en el servidor de AAA externo (RADIUS o TACACS+), sobre la base de donde se salva la información del nombre de usuario y contraseña.

Esto es una configuración local para un usuario que se llame detrás en 5551212:

```
username callmeback callback-dialstring 5551212 password cisco
```

Esta configuración local es aplicable a los usuarios se permite que especificar su propio número de devolución de llamada:

```
username callmeback callback-dialstring "" password cisco
```

### [Diagrama de la red](#)

En este documento, se utiliza esta configuración de red:

### [Configuraciones](#)

Este documento usa esta configuración:

- isdn2-2 (router AS5200)

isdn2-2 (router AS5200)
Current configuration: ! version 11.3 service timestamps debug datetime msec

```

service password-encryption
no service udp-small-servers
no service tcp-small-servers
!
hostname isdn2-2
!
aaa new-model
aaa authentication login default none
aaa authentication login use-local local
aaa authentication ppp default local
aaa authorization network local
!--- Runs authorization for network-related service
requests (Example: PPP). !--- For an AAA server
implementation, replace "local" with TACACS+ or RADIUS
in !--- these statements. enable secret 5 <deleted> !
username callmeback callback-dialstring "" password 7
<deleted> !--- This is for mobile users. The client
specifies the callback number. !--- If a RADIUS server
is used, this information can be offloaded to the
server. ip domain-name cisco.com isdn switch-type
primary-5ess chat-script offhook "" "ATH1" OK chat-
script callback ABORT ERROR ABORT BUSY "" "ATZ" OK "ATDT
\T" TIMEOUT 60 CONNECT \c !--- The chat script
"callback" is used for the callback connection. clock
timezone PST -8 clock summer-time PDT recurring !!
controller T1 0 !--- Active T1 Primary Rate Interface
(PRI). framing esf clock source line secondary linecode
b8zs pri-group timeslots 1-24 ! controller T1 1 shutdown
! interface Ethernet0 ip address 172.16.25.52
255.255.255.240 ! interface Serial0 no ip address
shutdown ! interface Serial1 no ip address shutdown !
interface Serial0:23 !--- D-channel for T1 0. ip
unnumbered Ethernet0 encapsulation ppp dialer-group 1
isdn incoming-voice modem !--- Allows incoming ISDN
voice calls to be switched to the onboard modems. peer
default ip address pool default ! interface Group-Async1
ip unnumbered Ethernet0 ip tcp header-compression
passive encapsulation ppp async mode interactive peer
default ip address pool default no cdp enable ppp max-
bad-auth 3 ppp callback accept !--- Allows the group-
async to accept a callback request to a remote host. ppp
authentication chap !--- CHAP, PAP, or both must be
enabled for callback. group-range 1 12 ! router eigrp
202 network 172.16.0.0 distance 90 172.16.25.49 0.0.0.0
no auto-summary ! ip local pool default 172.16.25.59
172.16.25.62 !--- Default IP address pool for dial-in
clients. ip default-gateway 172.16.25.49 ip classless
dialer-list 1 protocol ip permit ! line con 0 line 1 6
autoselect during-login autoselect ppp script modem-off-
hook offhook script callback callback !--- Specifies a
chat script to issue AT commands to the modem during a
callback attempt. !--- The chat-scripts "offhook" and
"callback" were configured earlier. login authentication
use-local modem InOut transport input all line 7 12 !---
These modems are busied out and not used. autoselect
during-login autoselect ppp login authentication use-
local modem InOut modem busyout transport input all line
aux 0 exec-timeout 0 0 line vty 0 4 password 7 <deleted>
! end

```

## [Configuración de cliente de Windows](#)

## Configuración del cliente de Windows 95 y 98

Para Windows 95 y 98 PC, no hay configuración para devolución de llamada especial del cliente. El servidor de acceso maneja las características de devolución de llamada de la conexión. Windows 95 o los 98 PC visualiza un mensaje “para del servicio repetido que espera” para indicar que un servicio repetido está en curso.

## Configuración del cliente de Windows NT y 2000

Configure estas Plataformas para pedir el servicio repetido. Complete estos pasos para configurarlos:

1. Elija el **Start (Inicio) > Programs (Programas) > Accessories (Accesorios) > las comunicaciones > la red y las conexiones por línea telefónica.**
2. Elija **avanzado > las Preferencias de conexión telefónica manual** del menú.
3. Haga clic la lengüeta del **servicio repetido** para acceder el menú de las características de devolución de llamada tal y como se muestra en del [cuadro 1](#).
4. Configure sus opciones de devolución de llamada según las necesidades: Para no utilizar la función callback (devolución de llamada), haga clic en el botón No Callback (Sin devolución de llamada). Se le preguntará qué hacer cuando un servidor ofrece la devolución de una llamada, haga clic en el botón Ask Me During Dialing When The Server Offers. Para validar automáticamente las ofertas del servicio repetido, haga clic **siempre la llamada yo detrás en el botón abajo de los números**, y seleccione el dispositivo para utilizar de la lista. Para cambiar el número de teléfono del servicio repetido, seleccione el dispositivo y haga clic el **botón Edit**. Ingrese el número en el **campo de número telefónico** tal y como se muestra en del cuadro 1, y después haga clic la **AUTORIZACIÓN** en la llamada yo detrás en el cuadro de diálogo.
5. Haga clic el **campo de número telefónico**, y ingrese el número en la llamada yo detrás en el cuadro de diálogo (mostrado en el [cuadro 1](#)). Haga Click en OK cuando le acaban.
6. Cuando le acaban, haga clic la **AUTORIZACIÓN** en el cuadro de diálogo de las Preferencias de conexión telefónica manual. **Cuadro 1 – Acceda las características de devolución de llamada**

## Verificación

En esta sección encontrará información que puede utilizar para corroborar que su configuración esté funcionando correctamente.

La herramienta [Output Interpreter](#) (sólo para clientes [registrados](#)) permite utilizar algunos comandos “show” y ver un análisis del resultado de estos comandos.

- **muestre el active isdn** — información de las visualizaciones sobre los actuales entrantes y las llamadas ISDN salientes. Utilice este comando de verificar si el servicio repetido fue completado con éxito. Si el servicio repetido es acertado, **muestre que active isdn** muestra la llamada como saliente en el servidor de devolución de llamada.
- **usuarios de la demostración** — información de las visualizaciones sobre las líneas activas en el router. Usted puede también utilizar el **comando show caller** si su versión del Cisco IOS Software la soporta.

- **marcador de la demostración** — información de diagnóstico general de las demostraciones para las interfaces configuradas para el Dial-on-Demand Routing (DDR).

## Troubleshooting

En esta sección encontrará información que puede utilizar para solucionar problemas de configuración.

### Comandos para resolución de problemas

**Nota:** Antes de ejecutar un comando debug, consulte **Información Importante sobre Comandos Debug**.

Para más información sobre los **comandos debug**, vea la [referencia del comando Debug del Cisco IOS Release 12.0](#).

- **haga el debug de la autenticación aaa** — información de las visualizaciones sobre la autenticación AAA.
- **debug aaa authorization** — información de las visualizaciones sobre la autorización AAA.
- **debug callback** — eventos de devolución de llamada de las visualizaciones cuando el router utiliza un módem y un chat script a la devolución de llamada en un línea de la terminal.
- **módem del debug** — le permite para observar la actividad de línea del módem en un servidor de acceso.
- **debug ppp [paquete | negociación | error | autenticación]** — información de las visualizaciones sobre el tráfico e intercambios en una red interna que implementa el PPP. *paquete* — paquetes PPP de las visualizaciones que son enviados y recibidos. (Este comando muestra el vaciado de paquetes de bajo nivel). *negociación* — paquetes PPP de las visualizaciones transmitidos durante el inicio de PPP, cuando se negocian las opciones PPP. los errores del protocolo y la estadística de error de los *errores aparecen* se asociaron a la negociación y a la operación de la conexión PPP. *autenticación* — mensajes de protocolo de la autenticación de las visualizaciones, que incluyen los intercambios de la GRIETA y PAP.
- **charla del debug** — muestra el apretón de manos que ocurre entre el servidor de acceso y su módem interno mientras que el módem se da instrucciones para marcar hacia fuera. Un charla-script es los conjuntos de pares de la cadena esperar-enviar que definen el apretón de manos entre el equipo de terminal de datos (DTE) y los dispositivos de Equipo de comunicación de datos (DCE).
- **debug ISDN q931** — visualiza los mensajes y los debugs de configuración de la llamada y cierre del q.931 ISDN (canal D). En este escenario, la llamada del módem se lleva como servicio portador de voz sobre el Public Switched Telephone Network (PSTN).
- **debug modem csm** — le permite para resolver problemas los problemas del Call Switching Module (CS) en el Routers con los módems digitales internos. Con este comando puede realizar un seguimiento de la secuencia completa de la switching de las llamadas entrantes y salientes.

```
isdn2-2#show debug General OS: Modem control/process activation debugging is on AAA
Authentication debugging is on AAA Authorization debugging is on PPP: PPP protocol negotiation
debugging is on ISDN: ISDN Q931 packets debugging is on Chat Scripts: Chat scripts activity
debugging is on Modem Management: Modem Management Call Switching Module debugging is on isdn2-
2# !--- This is the initial call from the client. *Mar 1 01:24:48.643: ISDN Se0:23: RX <- SETUP
```

pd = 8 callref = 0x36 \*Mar 1 01:24:48.647: Bearer Capability i = 0x9090A2 \*Mar 1 01:24:48.651: Channel ID i = 0xA98393 \*Mar 1 01:24:48.651: Called Party Number i = 0xC1, '4084327528' \*Mar 1 01:24:48.663: ISDN Se0:23: Incoming call id = 0xA \*Mar 1 01:24:48.671: EVENT\_FROM\_ISDN::dchan\_idb=0x7F8EE0, call\_id=0xA, ces=0x1 bchan=0x12, event=0x1, cause=0x0 \*Mar 1 01:24:48.671: VDEV\_ALLOCATE: slot 0 and port 3 is allocated. \*Mar 1 01:24:48.675: EVENT\_FROM\_ISDN:(000A): DEV\_INCALL at slot 0 and port 3 \*Mar 1 01:24:48.675: CSM\_PROC\_IDLE: CSM\_EVENT\_ISDN\_CALL at slot 0, port 3 \*Mar 1 01:24:48.679: Fast Ringing On at modem slot 0, port 3 \*Mar 1 01:24:48.699: ISDN Se0:23: TX -> CALL\_PROC pd = 8 callref = 0x8036 \*Mar 1 01:24:48.703: Channel ID i = 0xA98393 \*Mar 1 01:24:48.735: ISDN Se0:23: TX -> ALERTING pd = 8 callref = 0x8036 \*Mar 1 01:24:49.699: Fast Ringing Off at modem slot 0, port 3 \*Mar 1 01:24:49.699: CSM\_PROC\_IC1\_RING: CSM\_EVENT\_MODEM\_OFFHOOK at slot 0, port 3 \*Mar 1 01:24:49.711: ISDN Se0:23: TX -> CONNECT pd = 8 callref = 0x8036 \*Mar 1 01:24:49.783: ISDN Se0:23: RX <- CONNECT\_ACK pd = 8 callref = 0x36 \*Mar 1 01:24:49.799: EVENT\_FROM\_ISDN::dchan\_idb=0x7F8EE0, call\_id=0xA, ces=0x1 bchan=0x12, event=0x4, cause=0x0 \*Mar 1 01:24:49.799: EVENT\_FROM\_ISDN:(000A): DEV\_CONNECTED at slot 0 and port 3 \*Mar 1 01:24:49.803: CSM\_PROC\_IC4\_WAIT\_FOR\_CARRIER:CSM\_EVENT\_ISDN\_CONNECTED at slot 0, port 3 *!--- Modem has established carrier.* \*Mar 1 01:25:11.123: **TTY4: DSR came up** \*Mar 1 01:25:11.127: tty4: Modem: IDLE->READY \*Mar 1 01:25:11.131: TTY4: EXEC creation \*Mar 1 01:25:11.135: AAA/AUTHEN: create\_user (0x7B009C) user='' ruser='' port='tty4' rem\_addr='async/4084327528' authen\_type=ASCII service=LOGIN priv=1 \*Mar 1 01:25:11.139: AAA/AUTHEN/START (3134998138): port='tty4' list='use-local' action=LOGIN service=LOGIN \*Mar 1 01:25:11.143: AAA/AUTHEN/START (3134998138): found list use-local \*Mar 1 01:25:11.143: AAA/AUTHEN/START (3134998138): Method=LOCAL *!--- Local AAA.* \*Mar 1 01:25:11.147: AAA/AUTHEN (3134998138): status = GETUSER \*Mar 1 01:25:13.951: TTY4: Autoselect(2) sample 7E \*Mar 1 01:25:13.955: TTY4: Autoselect(2) sample 7EFF \*Mar 1 01:25:13.959: TTY4: Autoselect(2) sample 7EFF7D \*Mar 1 01:25:13.959: TTY4: Autoselect(2) sample 7EFF7D23 \*Mar 1 01:25:13.963: TTY4 **Autoselect cmd: ppp negotiate** \*Mar 1 01:25:13.967: AAA/AUTHEN/ABORT: (3134998138) because Autoselected. \*Mar 1 01:25:13.967: AAA/AUTHEN: free\_user (0x7B009C) user='' ruser='' port='tty4' rem\_addr='async/4084327528' authen\_type=ASCII service=LOGIN priv=1 \*Mar 1 01:25:13.975: TTY4: EXEC creation *!--- PPP has been autoselected and begins negotiation.* %LINK-3-UPDOWN: Interface Async4, changed state to up \*Mar 1 01:25:16.611: As4 PPP: Treating connection as a dedicated line \*Mar 1 01:25:16.611: As4 **PPP: Phase is ESTABLISHING**, Active Open *!--- LCP negotiation begins.* \*Mar 1 01:25:16.615: As4 LCP: O CONFREQ [Closed] id 3 len 25 \*Mar 1 01:25:16.619: As4 LCP: ACCM 0x000A0000 (0x0206000A0000) \*Mar 1 01:25:16.623: As4 LCP: AuthProto CHAP (0x0305C22305) \*Mar 1 01:25:16.623: As4 LCP: MagicNumber 0x608D04A3 (0x0506608D04A3) \*Mar 1 01:25:16.627: As4 LCP: PFC (0x0702) \*Mar 1 01:25:16.627: As4 LCP: ACFC (0x0802) \*Mar 1 01:25:16.751: As4 LCP: I CONFACK [REQsent] id 3 len 25 \*Mar 1 01:25:16.755: As4 LCP: ACCM 0x000A0000 (0x0206000A0000) \*Mar 1 01:25:16.755: As4 LCP: AuthProto CHAP (0x0305C22305) \*Mar 1 01:25:16.759: As4 LCP: MagicNumber 0x608D04A3 (0x0506608D04A3) \*Mar 1 01:25:16.763: As4 LCP: PFC (0x0702) \*Mar 1 01:25:16.763: As4 LCP: ACFC (0x0802) \*Mar 1 01:25:17.003: As4 **LCP: I CONFREQ** [ACKrcvd] id 3 len 23 *!--- Incoming CONFREQ.* \*Mar 1 01:25:17.003: As4 LCP: ACCM 0x000A0000 (0x0206000A0000) \*Mar 1 01:25:17.007: As4 LCP: MagicNumber 0x004A4A09 (0x0506004A4A09) \*Mar 1 01:25:17.007: As4 LCP: PFC (0x0702) \*Mar 1 01:25:17.011: As4 LCP: ACFC (0x0802) \*Mar 1 01:25:17.011: As4 **LCP: Callback 6** (0x0D0306) *!--- Peer requests MS Callback (Option 6). !--- A PPP callback request uses Option 0.* \*Mar 1 01:25:17.015: As4 **LCP: O CONFACK** [ACKrcvd] id 3 len 23 \*Mar 1 01:25:17.015: As4 LCP: ACCM 0x000A0000 (0x0206000A0000) \*Mar 1 01:25:17.019: As4 LCP: MagicNumber 0x004A4A09 (0x0506004A4A09) \*Mar 1 01:25:17.023: As4 LCP: PFC (0x0702) \*Mar 1 01:25:17.023: As4 LCP: ACFC (0x0802) \*Mar 1 01:25:17.023: As4 LCP: **Callback 6** (0x0D0306) *!--- NAS CONFACKS all LCP parameters. !--- If the NAS refuses Callback (completely or just MS Callback), LCP may fail.* \*Mar 1 01:25:17.027: As4 LCP: State is Open *!--- Authentication begins.* \*Mar 1 01:25:20.095: As4 PPP: Phase is AUTHENTICATING, by this end \*Mar 1 01:25:20.099: As4 CHAP: O CHALLENGE id 4 len 28 from "isdn2-2" \*Mar 1 01:25:20.187: As4 CHAP: I RESPONSE id 4 len 26 from "callmeback" \*Mar 1 01:25:20.191: AAA/AUTHEN: create\_user (0x7ADEAC) user='callmeback' ruser='' port='Async4' rem\_addr='async/4084327528' authen\_type=CHAP service=PPP priv=1 \*Mar 1 01:25:20.195: AAA/AUTHEN/START (44582883): port='Async4' list='' action=LOGIN service=PPP \*Mar 1 01:25:20.199: AAA/AUTHEN/START (44582883): using "default" list \*Mar 1 01:25:20.199: AAA/AUTHEN/START (44582883): Method=LOCAL *!--- Authentication passes.* \*Mar 1 01:25:20.203: AAA/AUTHEN (44582883): **status = PASS** *!--- Check authorization for LCP. !--- With local AAA, this should pass. !--- For server-based AAA, this must be explicitly configured on the server.* \*Mar 1 01:25:20.207: AAA/AUTHOR/LCP As4: Authorize LCP \*Mar 1 01:25:20.207: AAA/AUTHOR/LCP: Async4: (3405067782): user='callmeback' \*Mar 1 01:25:20.211: AAA/AUTHOR/LCP: Async4: (3405067782): send AV service=ppp \*Mar 1 01:25:20.211: AAA/AUTHOR/LCP: Async4: (3405067782): send AV protocol=lcp \*Mar 1 01:25:20.215: AAA/AUTHOR/LCP: Async4 (3405067782): Method=LOCAL \*Mar 1 01:25:20.219: AAA/AUTHOR (3405067782): Post authorization status = PASS\_ADD \*Mar 1 01:25:20.223: AAA/AUTHOR/LCP As4: Processing AV service=ppp \*Mar 1 01:25:20.223: AAA/AUTHOR/LCP As4:

Processing AV protocol=lcp \*Mar 1 01:25:20.227: AAA/AUTHOR/LCP As4: Processing AV service=ppp  
\*Mar 1 01:25:20.227: AAA/AUTHOR/LCP As4: Processing AV protocol=lcp *!--- Callback-dialstring is null, so user is allowed to specify !--- their own callback number.* \*Mar 1 01:25:20.227:  
AAA/AUTHOR/LCP As4: **Processing AV callback-dialstring=** *!--- Authentication ACK is returned to client.* \*Mar 1 01:25:20.235: As4 **CHAP: O SUCCESS** id 4 len 4 *!--- Callback negotiation proceeds. Because callback-dialstring !--- is null, MCB debug says "Callback Number - Client ANY".* \*Mar 1 01:25:20.239: As4 **MCB: User callmeback Callback Number - Client ANY** *!--- The callback number of the client is requested. Client receives a dialog !--- box that prompts the user to type in the callback number. !--- Request is sent every two seconds. If the user is slow to type a response, !--- the call remains in this phase for a long time.* \*Mar 1 01:25:20.243: Async4 PPP: O MCB Request(1) id 20 len 9 \*Mar 1 01:25:20.243: Async4 MCB: O 1 14 0 9 2 5 0 1 0 \*Mar 1 01:25:20.247: As4 MCB: O **Request Id 20 Callback Type Client-Num delay 0** %LINEPROTO-5-UPDOWN: Line protocol on Interface Async4, changed state to up \*Mar 1 01:25:22.459: As4 MCB: **Timeout in state WAIT\_RESPONSE** \*Mar 1 01:25:22.463: Async4 PPP: O MCB Request(1) id 21 len 9 \*Mar 1 01:25:22.463: Async4 MCB: O 1 15 0 9 2 5 0 1 0 \*Mar 1 01:25:22.467: As4 MCB: O **Request Id 21 Callback Type Client-Num delay 0** \*Mar 1 01:25:24.499: As4 MCB: Timeout in state WAIT\_RESPONSE \*Mar 1 01:25:24.503: Async4 PPP: O MCB Request(1) id 22 len 9 \*Mar 1 01:25:24.503: Async4 MCB: O 1 16 0 9 2 5 0 1 0 \*Mar 1 01:25:24.507: As4 MCB: O **Request Id 22 Callback Type Client-Num delay 0** \*Mar 1 01:25:26.543: As4 MCB: Timeout in state WAIT\_RESPONSE \*Mar 1 01:25:26.547: Async4 PPP: O MCB Request(1) id 23 len 9 \*Mar 1 01:25:26.547: Async4 MCB: O 1 17 0 9 2 5 0 1 0 \*Mar 1 01:25:26.551: As4 MCB: O **Request Id 23 Callback Type Client-Num delay 0** \*Mar 1 01:25:28.583: As4 MCB: Timeout in state WAIT\_RESPONSE \*Mar 1 01:25:28.587: Async4 PPP: O MCB Request(1) id 24 len 9 \*Mar 1 01:25:28.587: Async4 MCB: O 1 18 0 9 2 5 0 1 0 \*Mar 1 01:25:28.591: As4 MCB: O **Request Id 24 Callback Type Client-Num delay 0** *!--- Client returned the callback number. Notice that the response !--- is for the initial request id 20.* \*Mar 1 01:25:29.763: Async4 PPP: I **MCB Response(2) id 20** len 17 \*Mar 1 01:25:29.767: Async4 MCB: I 2 14 0 11 2 D F 1 35 32 37 2D 39 36 35 31 0 \*Mar 1 01:25:29.767: As4 MCB: Received response *!--- Response is ignored because the id is 20. There have !--- been a few timeouts and id 24 (the last one sent) is expected.* \*Mar 1 01:25:29.771: As4 MCB: **Resp ignored. ID Expected 24, got id 20** \*Mar 1 01:25:30.623: As4 MCB: Timeout in state WAIT\_RESPONSE *!--- Send out new request (id 25).* \*Mar 1 01:25:30.627: Async4 PPP: O MCB Request(1) id 25 len 9 \*Mar 1 01:25:30.627: Async4 MCB: O 1 19 0 9 2 5 0 1 0 \*Mar 1 01:25:30.631: As4 MCB: O **Request Id 25 Callback Type Client-Num delay 0** *!--- Client has cached user response, and so the callback number is !--- returned right away.* \*Mar 1 01:25:30.715: Async4 PPP: I **MCB Response(2) id 25** len 17 \*Mar 1 01:25:30.719: Async4 MCB: I 2 19 0 11 2 D F 1 35 32 37 2D 39 36 35 31 0 \*Mar 1 01:25:30.723: As4 MCB: Received response *!--- Received client callback number is 527-9651.* \*Mar 1 01:25:30.723: As4 MCB: **Response CBK-Client-Num 2 13 15, addr 1-527-9651** *!--- Callback number acknowledged.* \*Mar 1 01:25:30.727: Async4 PPP: O **MCB Ack(3) id 26** len 17 \*Mar 1 01:25:30.731: Async4 MCB: O 3 1A 0 11 2 D F 1 35 32 37 2D 39 36 35 31 0 \*Mar 1 01:25:30.731: As4 MCB: O **Ack Id 26 Callback Type Client-Num delay 15** \*Mar 1 01:25:30.735: As4 MCB: **Negotiated MCB with peer** *!--- Client hangs up and begins to wait for callback. !--- This is indicated by an Incoming (I) TERMREQ.* \*Mar 1 01:25:30.815: As4 LCP: I **TERMREQ** [Open] id 5 len 4 \*Mar 1 01:25:30.815: As4 LCP: O **TERMAK** [Open] id 5 len 4 \*Mar 1 01:25:30.819: As4 MCB: Peer terminating the link \*Mar 1 01:25:30.819: As4 PPP: Phase is TERMINATING \*Mar 1 01:25:30.819: As4 MCB: Link terminated by peer, Callback Needed *!--- Initiate callback to client; sleeps for ten seconds.* \*Mar 1 01:25:30.823: As4 MCB: **Initiate Callback for callmeback at 527-9651** using Async \*Mar 1 01:25:30.827: As4 MCB: Async-callback in progress *!--- Drop modem and B-channel for initial call from client.* \*Mar 1 01:25:31.499: CSM\_PROC\_IC5\_OC6\_CONNECTED: CSM\_EVENT\_MODEM\_ONHOOK at slot 0, port 3 \*Mar 1 01:25:31.503: VDEV\_DEALLOCATE: slot 0 and port 3 is deallocated \*Mar 1 01:25:31.503: ISDN Se0:23: Event: Hangup call to call id 0xA %ISDN-6-DISCONNECT: **Interface Serial0:18 disconnected from unknown , call lasted 41 seconds** *!--- Call is completely disconnected.* \*Mar 1 01:25:31.523: ISDN Se0:23: TX -> DISCONNECT pd = 8 callref = 0x8036 \*Mar 1 01:25:31.523: Cause i = 0x8090 - Normal call clearing \*Mar 1 01:25:31.583: ISDN Se0:23: RX <- RELEASE pd = 8 callref = 0x36 \*Mar 1 01:25:31.655: ISDN Se0:23: TX -> RELEASE\_COMP pd = 8 callref = 0x8036 %LINEPROTO-5-UPDOWN: Line protocol on Interface Async4, changed state to down \*Mar 1 01:25:31.851: TTY4: Async Int reset: Dropping DTR \*Mar 1 01:25:33.695: As4 LCP: TIMEout: Time = 0x4E521C State = TERMSent \*Mar 1 01:25:33.699: As4 LCP: State is Closed \*Mar 1 01:25:33.699: As4 PPP: Phase is DOWN \*Mar 1 01:25:33.703: As4 PPP: Phase is ESTABLISHING, Passive Open \*Mar 1 01:25:33.707: As4 LCP: State is Listen %LINK-5-CHANGED: Interface Async4, changed state to reset \*Mar 1 01:25:33.879: As4 LCP: State is Closed \*Mar 1 01:25:33.879: As4 PPP: Phase is DOWN \*Mar 1 01:25:33.883: As4 IPCP: Remove route to 172.16.25.61 %LINK-3-UPDOWN: Interface Async4, changed state to down \*Mar 1 01:25:38.887: As4 LCP: State is Closed \*Mar 1 01:25:38.887: As4 PPP: Phase is DOWN *!--- Cleanup from previous call is finished.* \*Mar 1 01:25:40.863: CHAT4: **Matched chat script offhook to string offhook** \*Mar 1 01:25:40.867: CHAT4: Asserting DTR *!--- Modem goes offhook.* \*Mar 1 01:25:40.867: CHAT4: Chat script offhook started

\*Mar 1 01:25:40.871: CHAT4: Sending string: ATH1 \*Mar 1 01:25:40.871: CHAT4: Expecting string: OK \*Mar 1 01:25:40.911: CSM\_PROC\_IDLE: CSM\_EVENT\_MODEM\_OFFHOOK at slot 0, port 3 \*Mar 1 01:25:40.963: CHAT4: Completed match for expect: OK \*Mar 1 01:25:40.967: CHAT4: **Chat script offhook finished, status = Success !--- Chat script "offhook" was successfully completed.** \*Mar 1 01:25:40.967: CHAT4: **Matched chat script callback to string callback !--- Chat script "callback" is initiated.** \*Mar 1 01:25:40.971: CHAT4: Asserting DTR \*Mar 1 01:25:40.975: CHAT4: Chat script callback started **!--- Reset modem to known state.** \*Mar 1 01:25:40.975: CHAT4: Sending string: ATZ \*Mar 1 01:25:40.979: CSM\_PROC\_OC1\_REQUEST\_DIGIT: CSM\_EVENT\_MODEM\_ONHOOK at slot 0, port 3 \*Mar 1 01:25:40.983: VDEV\_DEALLOCATE: slot 0 and port 3 is deallocated \*Mar 1 01:25:40.979: CHAT4: Expecting string: OK \*Mar 1 01:25:42.123: CHAT4: Completed match for expect: OK **!--- Dial the callback number of the client.** \*Mar 1 01:25:42.127: CHAT4: Sending string: **ATDT \T<527-9651>** \*Mar 1 01:25:42.131: CHAT4: Expecting string: CONNECT \*Mar 1 01:25:43.199: CSM\_PROC\_IDLE: CSM\_EVENT\_MODEM\_OFFHOOK at slot 0, port 3 **!--- Modem/ISDN needs to collect the digits from IOS before it makes the call.** \*Mar 1 01:25:43.327: DSX1\_MAIL\_FROM\_NEAT: DC\_READY\_RSP: mid = 5, slot = 2, unit = 1 \*Mar 1 01:25:43.331: CSM\_PROC\_OC1\_REQUEST\_DIGIT: CSM\_EVENT\_DIGIT\_COLLECT\_READY at slot 0, port 3 \*Mar 1 01:25:43.331: CSM\_PROC\_OC1\_REQUEST\_DIGIT: CSM\_EVENT\_ADDR\_INFO\_COLLECTED at slot 0, port 3 \*Mar 1 01:25:44.327: DSX1\_MAIL\_FROM\_NEAT: DC\_FIRST\_DIGIT\_RSP: mid = 5, slot = 2, unit = 1 \*Mar 1 01:25:44.331: CSM\_PROC\_OC2\_COLLECT\_1ST\_DIGIT: CSM\_EVENT\_GET\_1ST\_DIGIT at slot 0, port 3 \*Mar 1 01:25:47.331: DSX1\_MAIL\_FROM\_NEAT: DC\_ALL\_DIGIT\_RSP: mid = 5, slot = 2, unit = 1 \*Mar 1 01:25:47.331: CSM\_PROC\_OC3\_COLLECT\_ALL\_DIGIT: CSM\_EVENT\_GET\_ALL\_DIGITS at slot 0, port 3 \*Mar 1 01:25:47.335: CSM\_PROC\_OC3\_COLLECT\_ALL\_DIGIT: **called party num: (5279651) at slot 0, port 3 !--- Digits have been collected; ISDN call is made.** \*Mar 1 01:25:47.339: process\_pri\_call making a voice\_call. \*Mar 1 01:25:47.351: ISDN Se0:23: TX -> SETUP pd = 8 callref = 0x0005 \*Mar 1 01:25:47.355: **Bearer Capability i = 0x8090A2 !--- Bearer cap indicates call is an analog call.** \*Mar 1 01:25:47.355: Channel ID i = 0xE1808397 \*Mar 1 01:25:47.359: **Called Party Number i = 0xA1, '5279651'** \*Mar 1 01:25:47.431: ISDN Se0:23: RX <- CALL\_PROC pd = 8 callref = 0x8005 \*Mar 1 01:25:47.435: Channel ID i = 0xA98397 \*Mar 1 01:25:47.451: EVENT\_FROM\_ISDN:dchan\_idb=0x7F8EE0, call\_id=0xA005, ces=0x1 bchan=0x16, event=0x3, cause=0x0 \*Mar 1 01:25:47.451: EVENT\_FROM\_ISDN:(A005): DEV\_CALL\_PROC at slot 0 and port 3 \*Mar 1 01:25:47.455: CSM\_PROC\_OC4\_DIALING: CSM\_EVENT\_ISDN\_BCHAN\_ASSIGNED at slot 0, port 3 \*Mar 1 01:25:48.147: ISDN Se0:23: RX <- ALERTING pd = 8 callref = 0x8005 \*Mar 1 01:25:48.151: Progress Ind i = 0x8388 - In-band info or appropriate now available \*Mar 1 01:25:50.835: ISDN Se0:23: RX <- CONNECT pd = 8 callref = 0x8005 \*Mar 1 01:25:50.851: EVENT\_FROM\_ISDN:dchan\_idb=0x7F8EE0, call\_id=0xA005, ces=0x1 bchan=0x16, event=0x4, cause=0x \*Mar 1 01:25:50.855: EVENT\_FROM\_ISDN:(A005): DEV\_CONNECTED at slot 0 and port 3 \*Mar 1 01:25:50.859: CSM\_PROC\_OC5\_WAIT\_FOR\_CARRIER: CSM\_EVENT\_ISDN\_CONNECTED at slot 0, port 3 **!--- ISDN call is connected.** \*Mar 1 01:25:50.867: ISDN Se0:23: **TX -> CONNECT\_ACK** pd = 8 callref = 0x0005 \*Mar 1 01:25:53.735: AAA/AUTHEN: free\_user (0x7ADEAC) user='callmeback' ruser='' port='Async4' rem\_addr='async/4084327528' authen\_type=CHAP service=PPP priv=1 **!--- Modems have established carrier.** \*Mar 1 01:26:13.487: CHAT4: Completed match for expect: CONNECT \*Mar 1 01:26:13.491: CHAT4: Sending string: \c \*Mar 1 01:26:13.491: CHAT4: Chat script callback finished, status = Success \*Mar 1 01:26:15.415: TTY4: **DSR came up** \*Mar 1 01:26:15.419: tty4: Modem: IDLE->READY \*Mar 1 01:26:15.439: TTY4: EXEC creation \*Mar 1 01:26:15.443: AAA/AUTHEN: create\_user (0x7ADEA4) user='' ruser='' port='tty4' rem\_addr='async/5279651' authen\_type=ASCII service=LOGIN priv=1 \*Mar 1 01:26:15.447: AAA/AUTHEN/START (2043462211): port='tty4' list='use-local' action=LOGIN service=LOGIN \*Mar 1 01:26:15.451: AAA/AUTHEN/START (2043462211): found list use-local \*Mar 1 01:26:15.451: AAA/AUTHEN/START (2043462211): Method=LOCAL \*Mar 1 01:26:15.455: AAA/AUTHEN (2043462211): status = GETUSER **!--- PPP negotiation begins again.** \*Mar 1 01:26:16.631: TTY4: Autoselect(2) sample 7E %LINK-3-UPDOWN: Interface Async4, changed state to up \*Mar 1 01:26:18.663: As4 PPP: Treating connection as a dedicated line \*Mar 1 01:26:18.663: As4 PPP: Phase is ESTABLISHING, Active Open \*Mar 1 01:26:18.667: As4 LCP: O CONFREQ [Closed] id 5 len 25 \*Mar 1 01:26:18.671: As4 LCP: ACCM 0x000A0000 (0x0206000A0000) \*Mar 1 01:26:18.675: As4 LCP: AuthProto CHAP (0x0305C22305) \*Mar 1 01:26:18.675: As4 LCP: MagicNumber 0x608DF70C (0x0506608DF70C) \*Mar 1 01:26:18.679: As4 LCP: PFC (0x0702) \*Mar 1 01:26:18.679: As4 LCP: ACFC (0x0802) \*Mar 1 01:26:18.779: As4 LCP: I CONFACK [REQsent] id 5 len 25 \*Mar 1 01:26:18.783: As4 LCP: ACCM 0x000A0000 (0x0206000A0000) \*Mar 1 01:26:18.787: As4 LCP: AuthProto CHAP (0x0305C22305) \*Mar 1 01:26:18.787: As4 LCP: MagicNumber 0x608DF70C (0x0506608DF70C) \*Mar 1 01:26:18.791: As4 LCP: PFC (0x0702) \*Mar 1 01:26:18.791: As4 LCP: ACFC (0x0802) \*Mar 1 01:26:19.707: As4 LCP: I CONFREQ [ACKrcvd] id 3 len 20 \*Mar 1 01:26:19.711: As4 LCP: ACCM 0x000A0000 (0x0206000A0000) \*Mar 1 01:26:19.711: As4 LCP: MagicNumber 0x004B3EF5 (0x0506004B3EF5) \*Mar 1 01:26:19.715: As4 LCP: PFC (0x0702) \*Mar 1 01:26:19.715: As4 LCP: ACFC (0x0802) \*Mar 1 01:26:19.719: As4 LCP: O CONFACK [ACKrcvd] id 3 len 20 \*Mar 1 01:26:19.723: As4 LCP: ACCM 0x000A0000 (0x0206000A0000) \*Mar 1 01:26:19.723: As4 LCP: MagicNumber 0x004B3EF5 (0x0506004B3EF5) \*Mar 1 01:26:19.727: As4 LCP: PFC (0x0702) \*Mar 1 01:26:19.727: As4 LCP: ACFC (0x0802) \*Mar 1 01:26:19.731: As4 LCP: State is Open **!---**

Reauthenticate the user. \*Mar 1 01:26:22.779: As4 PPP: Phase is AUTHENTICATING, by this end \*Mar 1 01:26:22.783: As4 CHAP: O CHALLENGE id 6 len 28 from "isdn2-2" \*Mar 1 01:26:22.887: As4 CHAP: I RESPONSE id 6 len 26 from "callmeback" \*Mar 1 01:26:22.895: AAA/AUTHEN: create\_user (0x8F1DAC) user='callmeback' ruser='' port='Async4' rem\_addr='async/5279651' authen\_type=CHAP service=PPP priv=1 \*Mar 1 01:26:22.899: AAA/AUTHEN/START (2174906802): port='Async4' list='' action=LOGIN service=PPP \*Mar 1 01:26:22.899: AAA/AUTHEN/START (2174906802): using "default" list \*Mar 1 01:26:22.903: AAA/AUTHEN/START (2174906802): Method=LOCAL \*Mar 1 01:26:22.903: AAA/AUTHEN (2174906802): status = PASS \*Mar 1 01:26:22.907: AAA/AUTHOR/LCP As4: Authorize LCP \*Mar 1 01:26:22.911: AAA/AUTHOR/LCP: Async4: (3262137315): user='callmeback' \*Mar 1 01:26:22.911: AAA/AUTHOR/LCP: Async4: (3262137315): send AV service=ppp \*Mar 1 01:26:22.915: AAA/AUTHOR/LCP: Async4: (3262137315): send AV protocol=lcp \*Mar 1 01:26:22.915: AAA/AUTHOR/LCP: Async4 (3262137315): Method=LOCAL \*Mar 1 01:26:22.923: AAA/AUTHOR (3262137315): Post authorization status =PASS\_ADD \*Mar 1 01:26:22.927: AAA/AUTHOR/LCP As4: Processing AV service=ppp \*Mar 1 01:26:22.927: AAA/AUTHOR/LCP As4: Processing AV protocol=lcp \*Mar 1 01:26:22.931: AAA/AUTHOR/LCP As4: Processing AV service=ppp \*Mar 1 01:26:22.931: AAA/AUTHOR/LCP As4: Processing AV protocol=lcp \*Mar 1 01:26:22.931: AAA/AUTHOR/LCP As4: Processing AV callback-dialstring= \*Mar 1 01:26:22.939: As4 CHAP: O SUCCESS id 6 len 4 \*Mar 1 01:26:22.943: As4 PPP: Phase is UP \*Mar 1 01:26:22.947: AAA/AUTHOR/FSM As4: (0): Can we start IPCP? \*Mar 1 01:26:22.947: AAA/AUTHOR/FSM: Async4: (345798021): user='callmeback' \*Mar 1 01:26:22.951: AAA/AUTHOR/FSM: Async4: (345798021): send AV service=ppp \*Mar 1 01:26:22.951: AAA/AUTHOR/FSM: Async4: (345798021): send AV protocol=ip \*Mar 1 01:26:22.955: AAA/AUTHOR/FSM: Async4 (345798021): Method=LOCAL \*Mar 1 01:26:22.955: AAA/AUTHOR (345798021): Post authorization status = PASS\_REPL !--- Negotiate IPCP. \*Mar 1 01:26:22.959: AAA/AUTHOR/FSM As4: We can start IPCP \*Mar 1 01:26:22.963: As4 IPCP: O CONFREQ [Closed] id 1 len 16 \*Mar 1 01:26:22.967: As4 IPCP: CompressType VJ 15 slots (0x0206002D0F00) \*Mar 1 01:26:22.967: As4 IPCP: Address 172.16.25.52 (0x0306AC101934) \*Mar 1 01:26:23.019: As4 IPCP: I CONFREQ [REQsent] id 1 len 40 \*Mar 1 01:26:23.023: As4 IPCP: CompressType VJ 15 slots CompressSlotID (0x0206002D0F01) \*Mar 1 01:26:23.027: As4 IPCP: Address 0.0.0.0 (0x030600000000) \*Mar 1 01:26:23.027: As4 IPCP: PrimaryDNS 0.0.0.0 (0x810600000000) \*Mar 1 01:26:23.031: As4 IPCP: PrimaryWINS 0.0.0.0 (0x820600000000) \*Mar 1 01:26:23.035: As4 IPCP: SecondaryDNS 0.0.0.0 (0x830600000000) \*Mar 1 01:26:23.035: As4 IPCP: SecondaryWINS 0.0.0.0 (0x840600000000) \*Mar 1 01:26:23.039: AAA/AUTHOR/IPCP As4: Start. Her address 0.0.0.0, we want 0.0.0.0 \*Mar 1 01:26:23.039: AAA/AUTHOR/IPCP As4: Processing AV service=ppp \*Mar 1 01:26:23.043: AAA/AUTHOR/IPCP As4: Processing AV protocol=ip \*Mar 1 01:26:23.043: AAA/AUTHOR/IPCP As4: Authorization succeeded \*Mar 1 01:26:23.047: AAA/AUTHOR/IPCP As4: Done. Her address 0.0.0.0, we want 0.0.0.0 \*Mar 1 01:26:23.047: As4 IPCP: Using pool 'default' \*Mar 1 01:26:23.051: As4 IPCP: Pool returned 172.16.25.60 \*Mar 1 01:26:23.051: As4 IPCP: O CONFREQ [REQsent] id 1 len 28 \*Mar 1 01:26:23.055: As4 IPCP: PrimaryDNS 0.0.0.0 (0x810600000000) \*Mar 1 01:26:23.059: As4 IPCP: PrimaryWINS 0.0.0.0 (0x820600000000) \*Mar 1 01:26:23.059: As4 IPCP: SecondaryDNS 0.0.0.0 (0x830600000000) \*Mar 1 01:26:23.063: As4 IPCP: SecondaryWINS 0.0.0.0 (0x840600000000) \*Mar 1 01:26:23.067: As4 IPCP: I CONFREQ [REQsent] id 1 len 16 \*Mar 1 01:26:23.067: As4 IPCP: CompressType VJ 15 slots (0x0206002D0F00) \*Mar 1 01:26:23.071: As4 IPCP: Address 172.16.25.52 (0x0306AC101934) \*Mar 1 01:26:23.139: As4 IPCP: I CONFREQ [ACKrcvd] id 2 len 16 \*Mar 1 01:26:23.139: As4 IPCP: CompressType VJ 15 slots CompressSlotID (0x0206002D0F01) \*Mar 1 01:26:23.143: As4 IPCP: Address 0.0.0.0 (0x030600000000) \*Mar 1 01:26:23.147: AAA/AUTHOR/IPCP As4: Start. Her address 0.0.0.0, we want 172.16.25.60 \*Mar 1 01:26:23.147: AAA/AUTHOR/IPCP As4: Processing AV service=ppp \*Mar 1 01:26:23.151: AAA/AUTHOR/IPCP As4: Processing AV protocol=ip \*Mar 1 01:26:23.151: AAA/AUTHOR/IPCP As4: Authorization succeeded \*Mar 1 01:26:23.151: AAA/AUTHOR/IPCP As4: Done. Her address 0.0.0.0, we want 172.16.25.60 \*Mar 1 01:26:23.155: As4 IPCP: O CONFREQ [ACKrcvd] id 2 len 10 \*Mar 1 01:26:23.159: As4 IPCP: Address 172.16.25.60 (0x0306AC10193C) \*Mar 1 01:26:23.255: As4 IPCP: I CONFREQ [ACKrcvd] id 3 len 16 \*Mar 1 01:26:23.259: As4 IPCP: CompressType VJ 15 slots CompressSlotID (0x0206002D0F01) \*Mar 1 01:26:23.263: As4 IPCP: Address 172.16.25.60 (0x0306AC10193C) \*Mar 1 01:26:23.263: AAA/AUTHOR/IPCP As4: Start. Her address 172.16.25.60, we want 172.16.25.60 \*Mar 1 01:26:23.267: AAA/AUTHOR/IPCP: Async4: (3819567164): user='callmeback' \*Mar 1 01:26:23.271: AAA/AUTHOR/IPCP: Async4: (3819567164): send AV service=ppp \*Mar 1 01:26:23.271: AAA/AUTHOR/IPCP: Async4: (3819567164): send AV protocol=ip \*Mar 1 01:26:23.275: AAA/AUTHOR/IPCP: Async4: (3819567164): send AV addr\*172.16.25.60 \*Mar 1 01:26:23.275: AAA/AUTHOR/IPCP: Async4 (3819567164): Method=LOCAL \*Mar 1 01:26:23.279: AAA/AUTHOR (3819567164): Post authorization status = PASS\_REPL \*Mar 1 01:26:23.283: AAA/AUTHOR/IPCP As4: Reject 172.16.25.60, using 172.16.25.60 \*Mar 1 01:26:23.287: AAA/AUTHOR/IPCP As4: Processing AV service=ppp \*Mar 1 01:26:23.291: AAA/AUTHOR/IPCP As4: Processing AV protocol=ip \*Mar 1 01:26:23.291: AAA/AUTHOR/IPCP As4: Processing AV addr\*172.16.25.60 \*Mar 1 01:26:23.295: AAA/AUTHOR/IPCP As4: Authorization succeeded \*Mar 1 01:26:23.295: AAA/AUTHOR/IPCP As4: Done. Her address 172.16.25.60, we want 172.16.25.60 \*Mar 1 01:26:23.299: As4 IPCP: O CONFREQ [ACKrcvd] id 3 len 16 \*Mar 1 01:26:23.303:

As4 IPCP: CompressType VJ 15 slots CompressSlotID (0x0206002D0F01) \*Mar 1 01:26:23.303: As4 IPCP: Address 172.16.25.60 (0x0306AC10193C) \*Mar 1 01:26:23.307: As4 IPCP: State is Open \*Mar 1 01:26:23.323: As4 IPCP: Install route to 172.16.25.60 %LINEPROTO-5-UPDOWN: Line protocol on Interface Async4, changed state to up !--- Client is connected.

## [Información Relacionada](#)

- [Configurar la devolución de llamada asíncrona](#)
- [Devolución de llamadas de PPP sobre ISDN](#)
- [Configuración de la devolución de llamada PPP para DDR](#)
- [El configurar devolución de llamada PPP con TACACS+](#)
- [Configuración del tono de respuesta PPP con RADIUS](#)
- [Páginas de Soporte de Productos de Acceso](#)
- [Páginas de soporte de la tecnología de marcación](#)
- [Soporte Técnico - Cisco Systems](#)