

Configuring Redial Timers After Failed Callback Dial Attempts

Document ID: 4686

Introduction

Prerequisites

- Requirements
- Components Used
- Related Products
- Conventions

Configure

- Network Diagram
- Configurations

Verify

Troubleshoot

- Troubleshooting Commands

Related Information

Introduction

In most callback scenarios, the client router dials the callback server, then the two routers negotiate callback parameters. The server then disconnects the call and initiates the callback. However, during the interval between the initial call disconnect and the callback, the calling side might potentially place few consecutive outgoing calls to the server while it waits for the server to call back. This is normal dial-on-demand routing (DDR) behavior, since the client detects that the initial call failed, and does not know that a callback is in progress.

In order to prevent the client constantly dialing the callback server, use the **dialer redial** command on the calling side. This suppresses additional outgoing calls to the server, while waiting for the callback. The calls are suppressed until the pre-defined timer expires. For example, if the **dialer redial** interval time is 15 seconds, the client waits for 15 seconds before it initiates a redial. Within that time, the callback is completed and the client does not have to dial again.

Prerequisites

Requirements

This document assumes that you have the callback client and server configured for normal DDR. The client should be able to identify interesting traffic, dial the peer, connect, and pass traffic on the link. You can then configure callback per the example shown in this document.

For some additional DDR sample configurations and troubleshooting information, refer to these documents:

- Configuring BRI-to-BRI Dialup with DDR Dialer Maps
- Configuring ISDN DDR with Dialer Profiles
- Using the **show isdn status** Command for BRI Troubleshooting

This configuration has these Cisco IOS® software requirements:

- The **dialer redial** command was introduced in Cisco IOS Software Release 12.1(2). In Cisco IOS Software Release 12.2(2)T, the command was modified with additional options. For more information, refer to Redial Enhancements.

Components Used

The information in this document is based on these software and hardware versions:

- **Callback Client:** Cisco 2500 with one BRI interface. This router runs Cisco IOS Software Release 12.2(5).
- **Callback Server:** Cisco 2500 with one BRI interface. This router runs Cisco IOS Software Release 12.2(3).
- Two active BRI circuits.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Related Products

This configuration can be used with any ISDN caller ID callback scenario. Since this configuration is not platform dependent, it can be applied to routers that have a BRI interface.

The **dialer redial** command is not exclusively meant for callback scenarios. You can use it to restrict calls in any situation where the calling party repeatedly dials the other side. For more information, refer to Redial Enhancements.

Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

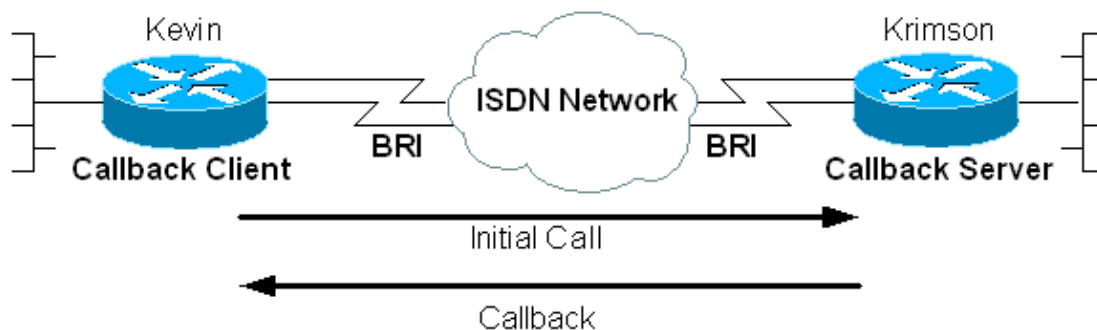
Configure

This section presents you with the information to configure the features this document describes.

Note: In order to find additional information on the commands this document uses, use the Command Lookup Tool (registered customers only).

Network Diagram

This document uses the network setup shown in this diagram.



Configurations

This document uses the configurations shown here.

Note: This configuration is not comprehensive. Only the relevant callback, interface, and dialer commands are included. For a more detailed example of caller ID based callback, refer to ISDN Authentication and Callback with Caller ID.

- kevin (callback client)
- krimson (callback server)

```

                                     kevin (callback client)
interface BRI0
  !-- Physical interface configuration.

  no ip address
  encapsulation ppp
  dialer rotary-group 1

  !-- BRI 0 is a member of rotary group 1.
  !-- The rotary group configuration is in interface Dialer 1.

  isdn switch-type basic-net3

  !-- Switch-type for this NAS. Obtain this information from the Telco.

  no cdp enable
  ppp authentication chap

  !-- Use PPP Challenge Handshake Authentication
  !-- Protocol (CHAP) authentication.

  !
  interface Dialer1

  !-- Configuration for rotary group 1.
  !-- The Dialer interface number (1) must exactly match the rotary group number
  !-- configured on the physical interfaces.

  ip address 10.9.8.2 255.255.255.0

  !-- The IP address for this dialer interface.
  !-- Note that the dialer map on the peer points to this address.

  encapsulation ppp
  dialer in-band

  !-- Enable V.25bis on this interface.

  dialer redial interval 15 attempts 2

  !-- Redial interval set to 15 seconds with two available attempts.
  !-- The router waits 15 seconds after a call fails, then dials again.

  dialer map ip 10.9.8.1 name krimson 6119

  !-- Dialer map for the peer.
  !-- The IP address of the remote router, authenticated username
  !-- and ISDN number to dial are specified.

  dialer-group 1
```

```
!--- Apply the interesting traffic definition from dialer-list 1.  
!--- Note: The specified dialer-group number must be the same as  
!--- the dialer-list number. In this example, it is defined as "1".
```

```
no cdp enable  
ppp authentication chap  
!  
dialer-list 1 protocol ip permit
```

```
!--- Define IP as interesting traffic.
```

krimson (callback server)

```
interface BRI0
```

```
no ip address  
encapsulation ppp  
no ip route-cache  
no ip mroute-cache  
no keepalive  
dialer rotary-group 1
```

```
!--- BRI 0 is a member of rotary group 1.  
!--- The rotary group configuration is in interface Dialer 1.
```

```
isdn switch-type basic-net3  
no fair-queue  
no cdp enable  
ppp authentication chap  
!
```

```
interface Dialer1
```

```
!--- Configuration for rotary group 1.  
!--- The Dialer interface number (1) must exactly match the rotary group number  
!--- configured on the physical interfaces.
```

```
ip address 10.9.8.1 255.255.255.0
```

```
!--- The IP address for this dialer interface.  
!--- Note that the dialer map on the peer points to this address.
```

```
encapsulation ppp  
no ip route-cache  
no ip mroute-cache  
load-interval 30  
no keepalive  
dialer in-band  
dialer enable-timeout 5
```

```
!--- The time (in seconds) to wait before you initiate callback  
!--- (default is 15 seconds). This permits the router to callback the client  
!--- 5 seconds after the initial call is disconnected.  
!--- Note: This timeout value is less than the dialer redial timer set  
!--- on the client. If you need to increase the enable-timeout,  
!--- you need to also raise the redial timer on the client.
```

```
dialer caller 6120 callback
```

```
!--- Permit calls from 6120 (the peer's ISDN number) and initiate  
!--- callback to the same number.
```

```
dialer map ip 10.9.8.2 name kevin 6120
```

```

!--- Dialer map for the peer.
!--- The remote router's IP address, authenticated username and
!--- ISDN number are specified.

dialer-group 1

!--- Apply the interesting traffic definition from dialer-list 1.
!--- Note: The specified dialer-group number must be the same
!--- as the dialer-list number.
!--- In this example, it is defined "1".

no fair-queue
no cdp enable
ppp authentication chap
!
dialer-list 1 protocol ip permit

!--- Define IP as interesting traffic.

```

Note: The command **dialer enable-timeout** on the callback server allows it to initiate a callback before the expiration of the default enable time of 15 seconds.

Verify

This section provides information you can use to confirm your configuration works properly.

Certain **show** commands are supported by the Output Interpreter Tool (registered customers only), which allows you to view an analysis of **show** command output.

- **show isdn active** Displays information about current calls. Provides information on both incoming and outgoing current ISDN calls.
- **show users** Displays information about the active lines on the router. You can also use the **show caller** command if your version of Cisco IOS software supports it.
- **show dialer** Displays general diagnostic information for interfaces configured for DDR.

Here is some **show** command output for successful calls. Pay attention to the sections in **bold** and the comments provided in the output. Compare the output you obtain with the results shown here.

The **show isdn active** command displays information about the current call. This command can be used to verify that the callback is successfully completed. If callback is successful, **show isdn active** shows the call as incoming on the callback client and outgoing on the callback server.

```

kevin#show isdn active
-----
ISDN ACTIVE CALLS
-----
Call      Calling      Called      Remote   Seconds  Seconds  Seconds  Charges
Type     Number      Number      Name     Used     Left     Idle     Units/Currency
-----
In      6119        6120        krimson  12      107     12
-----

```

Note that the client (kevin) has received an incoming call (due to callback) and is connected to krimson (the callback server).

```

krimson#show isdn active
-----
ISDN ACTIVE CALLS
-----

```

Call Type	Calling Number	Called Number	Remote Name	Seconds Used	Seconds Left	Seconds Idle	Charges Units/Currency
Out		6120	kevin	17	102	17	0

Note: The server (krimson) has an outbound call and is connected to kevin.

Troubleshoot

When troubleshooting callback issues it is best to first verify that:

- The client can call and connect to the server using normal DDR.
- The server can call and connect to the client using normal DDR.

If these steps fail, the issue is not due to callback. Instead, it is due to a DDR or physical interface problem. Callback can then be configured on top of the DDR configuration. This troubleshooting section assumes the two points shown earlier have been tested and verified.

Here is troubleshooting information relevant to this configuration.

Symptom: The client does not dial the initial link.

- **Possible Cause:** The DDR configuration and interesting traffic definition is incorrect.
 - ◆ **Action:** Verify that the commands shown earlier are configured correctly on the client.
- **Possible Cause:** The ISDN interface or BRI circuit does not function correctly.
 - ◆ **Action:** Verify that the BRI circuit is active and functions. Refer to Using the **show isdn status** Command for BRI Troubleshooting.
- **Possible Cause:** A redial to the same destination is pending (from a previous attempt).
 - ◆ **Action:** Since new dial-out attempts are not initiated if a redial to the same destination is pending, clear the dial interface. For example, **clear interface BRI 0** or **clear interface Dialer 1**.

Symptom: The client repeatedly dials the callback server.

- **Possible Cause:** The dialer redial interval time is too low.
 - ◆ **Action:** Adjust the dialer redial interval time to a higher value. This causes the router to wait longer before it redials.
- **Possible Cause:** The dialer redial attempts is set high.
 - ◆ **Action:** Lower the number of redial attempts using the command **dialer redial interval time attempts number** . The router does not redial the destination after the specified number of attempts.

Symptom: The callback server does not callback the client.

- **Possible Cause:** The time to wait before initiating callback is too high.
 - ◆ **Action:** Lower the callback timer specified using the command **dialer enable-timeout time** . This value needs to be much lower than the **dialer redial** interval time.

- **Possible Cause:** The DDR configuration or BRI circuit does not function correctly.

- ◆ **Action:** Verify that the BRI circuit function. Refer to Using the **show isdn status** Command for BRI Troubleshooting for more information. Also refer to Configuring ISDN Caller ID Callback for DDR related issues.

Troubleshooting Commands

Certain **show** commands are supported by the Output Interpreter Tool (registered customers only) , which allows you to view an analysis of **show** command output.

Note: Before you issue **debug** commands, refer to Important Information on Debug Commands.

- **debug dialer [events | packets]** Displays DDR debugging information about the packets received on a dialer interface.
- **debug isdn event** Displays ISDN events that occur on the ISDN interface. The ISDN events that are displayed are Q.931 events (call setup and tear down of ISDN network connections).
- **debug isdn q931** Displays call setup and tear down of the ISDN network connection (Layer 3).
- **debug ppp negotiation** Displays information on PPP traffic and exchanges while it negotiates the PPP components that include Link Control Protocol (LCP), Authentication, and Network Control Protocol (NCP). A successful PPP negotiation first opens the LCP state, then authenticates, and finally negotiates NCP.
- **debug callback** Displays callback events. This needs to be used primarily on the callback server.

This debug output is obtained from the callback client.

Note: The client dials the server and the call is disconnected in preparation for callback. The server then dials the client and the call is established. Pay particular attention to the sections in **bold** font.

```

kevin#show debug
Dial on demand:
Dial on demand events debugging is on
PPP:
PPP authentication debugging is on
PPP protocol negotiation debugging is on
ISDN:
ISDN events debugging is on
ISDN Q931 packets debugging is on

kevin#ping 10.9.8.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.9.8.1, timeout is 2 seconds:
*Mar  1 01:09:22.372: BR0 DDR: rotor dialout [priority]
*Mar  1 01:09:22.376: BR0 DDR: Dialing cause ip (s=10.9.8.2, d=10.9.8.1).

!--- The dialing cause is an ICMP ping packet destined for 10.9.8.1.
!--- ICMP must be included in the interesting traffic definition
!--- (using the dialer-list command).

*Mar  1 01:09:24.368: BR0 DDR: rotor dialout [priority]
*Mar  1 01:09:24.372: BR0 DDR: Dialing cause ip (s=10.9.8.2, d=10.9.8.1)
*Mar  1 01:09:24.376: BR0 DDR: Attempting to dial 6119

!--- The number that is dialed.

*Mar  1 01:09:24.376: ISDN BR0: Outgoing call id = 0x8031, dsl 0
*Mar  1 01:09:24.380: ISDN BR0: Event: Call to 6119 at 64 Kb/s
*Mar  1 01:09:24.384: ISDN BR0: process_bri_call():
call id 0x8031,called_number 6119, speed 64, call type DATA
*Mar  1 01:09:24.388: CCBRI_Go Fr Host InPkgInfo (Len=20) :
```

```

*Mar 1 01:09:24.388: 1 0 1 80 31 0 4 2 88 90 18 1 83 70 5 80 36 31 31 39
*Mar 1 01:09:24.396:
*Mar 1 01:09:24.400: CC_CHAN_GetIdleChanbri: dsl 0
*Mar 1 01:09:24.400: Found idle channel B1
*Mar 1 01:09:24.416: ISDN BR0: TX -> SETUP pd = 8 callref = 0x31
*Mar 1 01:09:24.424: Bearer Capability i = 0x8890
*Mar 1 01:09:24.428: Channel ID i = 0x83
*Mar 1 01:09:24.436: Called Party Number i = 0x80, '6119',
Plan:Unknown, Type:Unknown
*Mar 1 01:09:24.500: ISDN BR0: RX <- CALL_PROC pd = 8 callref = 0xB1
*Mar 1 01:09:24.508: Channel ID i = 0x89
*Mar 1 01:09:24.524: CCBRI_Go Fr L3 pkt (Len=7) :
*Mar 1 01:09:24.524: 2 1 31 98 18 1 89
*Mar 1 01:09:24.528:
*Mar 1 01:09:24.532: ISDN BR0: LIF_EVENT: ces/callid 1/0x8031 HOST_PROCEEDING
*Mar 1 01:09:24.536: ISDN BR0: HOST_PROCEEDING
*Mar 1 01:09:24.536: ISDN BR0: HOST_MORE_INFO
*Mar 1 01:09:24.752: ISDN BR0: RX <- DISCONNECT pd = 8 callref = 0xB1
*Mar 1 01:09:24.760: Cause i = 0x8095 - Call rejected

!--- The call is rejected.
!--- The callback server initiates a callback based
!--- on the caller ID of the client.

*Mar 1 01:09:24.780: CCBRI_Go Fr L3 pkt (Len=8) :
*Mar 1 01:09:24.784: 45 1 31 96 8 2 80 95
*Mar 1 01:09:24.788:
*Mar 1 01:09:24.788: ISDN BR0: LIF_EVENT: ces/c.allid 1/0x8031HOST_DISCONNECT
*Mar 1 01:09:24.792: ISDN BR0: Event: Call to 6119 was hung up.
*Mar 1 01:09:24.796: ISDN BR0: process_disc_ack(): call id 0x8031, ces 1,
call type DATA cause 0x0
*Mar 1 01:09:24.808: ISDN: get_isdn_service_state():
idb 0x230B74 bchan 2is_isdn 1 Not a Pri
*Mar 1 01:09:24.812: BRI0: wait for isdn carrier timeout, call id=0x8031
*Mar 1 01:09:24.812: Dil DDR: Redial: Created on BR0 for ip (s=10.9.8.2,d=10.9.8.1)

!--- The client does not repeatedly dial the server.
!--- The client waits for the dialer redial timer (15 seconds)
!--- to expire before it attempts to dial again.

*Mar 1 01:09:24.816: CCBRI_Go Fr Host InPkgInfo (Len=10) :
*Mar 1 01:09:24.820: 6 0 1 80 31 0 8 2 0 8
*Mar 1 01:09:24.824:
*Mar 1 01:09:24.836: ISDN BR0: TX -> RELEASE pd = 8 callref = 0x31
*Mar 1 01:09:24.840: Cause i = 0x8095 - Call rejected
*Mar 1 01:09:24.888: ISDN BR0: RX <- RELEASE_COMP pd = 8 callref = 0xB1
*Mar 1 01:09:24.912: CCBRI_Go Fr L3 pkt (Len=4) :
*Mar 1 01:09:24.912: 5A 1 31 99
*Mar 1 01:09:24.916:
*Mar 1 01:09:24.920: ISDN BR0: LIF_EVENT: ces/callid 1/0x8031HOST_DISCONNECT_ACK
*Mar 1 01:09:24.924: ISDN: get_isdn_service_state(): idb 0x230B74 bchan 2
is_isdn 1 Not a Pri
*Mar 1 01:09:24.928: ISDN BR0: HOST_DISCONNECT_ACK: call type is DATA..
*Mar 1 01:09:29.784: ISDN BR0: RX <- SETUP pd = 8 callref = 0x1A

!--- Incoming SETUP message for the callback.

*Mar 1 01:09:29.788: Bearer Capability i = 0x8890
*Mar 1 01:09:29.796: Channel ID i = 0x89
*Mar 1 01:09:29.800: Calling Party Number i = 0xA1, '6119',Plan:ISDN,
Type:National
*Mar 1 01:09:29.816: Called Party Number i = 0xC1, '6120',Plan:ISDN,
Type:Subscriber(local)
*Mar 1 01:09:29.836: CCBRI_Go Fr L3 pkt (Len=25) :
*Mar 1 01:09:29.840: 5 1 9A 90 4 2 88 90 18 1 89 6C 5 A1 36 31 31 39 70 5 C1 36 31 32 30
*Mar 1 01:09:29.848:

```

```

*Mar 1 01:09:29.852: ISDN BR0: Incoming call id = 0x0029, dsl 0
*Mar 1 01:09:29.856: ISDN BR0: LIF_EVENT: ces/callid 1/0x29HOST_INCOMING_CALL
*Mar 1 01:09:29.860: ISDN BR0: HOST_INCOMING_CALL: (non-POTS) DATA
*Mar 1 01:09:29.864: ISDN BR0: HOST_INCOMING_CALL: (1) call_type = DATA
*Mar 1 01:09:29.868: ISDN BR0: HOST_INCOMING_CALL: voice_answer_data =
FALSE call type is DATA
*Mar 1 01:09:29.868: ISDN BR0: Event: Received a DATA call from 6119 on Blat 64 Kb/s
*Mar 1 01:09:29.872: ISDN BR0: Event: Accepting the call id 0x29
*Mar 1 01:09:29.876: ISDN BR0: RM returned call_type 0 resource type 0response 1
*Mar 1 01:09:29.880: CCBRI_Go Fr Host InPkgInfo (Len=9) :
*Mar 1 01:09:29.880: 7 0 1 0 29 3 18 1 89
*Mar 1 01:09:29.884:
*Mar 1 01:09:29.896: ISDN BR0: isdn_send_connect(): msg 4, call id 0x29,
ces 1 bchan 0, call type DATA
*Mar 1 01:09:29.900: %LINK-3-UPDOWN: Interface BRI0:1, changed state to up
*Mar 1 01:09:29.908: ISDN: get_isdn_service_state(): idb 0x230B74
bchan 2is_isdn 1 Not a Pri
*Mar 1 01:09:29.912: BR0:1 PPP: Treating connection as a callin
*Mar 1 01:09:29.916: BR0:1 PPP: Phase is ESTABLISHING, Passive Open [0sess, 0 load]
*Mar 1 01:09:29.916: BR0:1 LCP: State is Listen
*Mar 1 01:09:29.920: CCBRI_Go Fr Host InPkgInfo (Len=6) :
*Mar 1 01:09:29.924: 4 0 1 0 29 0
*Mar 1 01:09:29.924:
*Mar 1 01:09:29.936: ISDN BR0: TX -> CALL_PROC pd = 8 callref = 0x9A
*Mar 1 01:09:29.944: Channel ID i = 0x89
*Mar 1 01:09:29.972: ISDN BR0: TX -> CONNECT pd = 8 callref = 0x9A
*Mar 1 01:09:30.052: ISDN BR0: RX <- CONNECT_ACK pd = 8 callref = 0x1A
*Mar 1 01:09:30.060: Channel ID i = 0x89

```

!--- Call is connected.

```

*Mar 1 01:09:30.076: CCBRI_Go Fr L3 pkt (Len=7) :
*Mar 1 01:09:30.076: F 1 9A 92 18 1 89
*Mar 1 01:09:30.080:
*Mar 1 01:09:30.084: ISDN BR0: LIF_EVENT: ces/callid 1/0x29 HOST_CONNECT
*Mar 1 01:09:30.084: ISDN BR0: Event: Connected to 6119 on Bl at 64 Kb/s
*Mar 1 01:09:31.920: BR0:1 LCP: TIMEOUT: State Listen

```

!--- LCP negotiation begins.

```

*Mar 1 01:09:31.924: BR0:1 LCP: O CONFREQ [Listen] id 20 len 15
*Mar 1 01:09:31.924: BR0:1 LCP: AuthProto CHAP (0x0305C22305)
*Mar 1 01:09:31.928: BR0:1 LCP: MagicNumber 0x004BDA2F (0x0506004BDA2F)
*Mar 1 01:09:31.964: BR0:1 LCP: I CONFACK [REQsent] id 20 len 15
*Mar 1 01:09:31.968: BR0:1 LCP: AuthProto CHAP (0x0305C22305)
*Mar 1 01:09:31.968: BR0:1 LCP: MagicNumber 0x004BDA2F (0x0506004BDA2F)
*Mar 1 01:09:32.116: BR0:1 LCP: I CONFREQ [ACKrcvd] id 7 len 15
*Mar 1 01:09:32.120: BR0:1 LCP: AuthProto CHAP (0x0305C22305)
*Mar 1 01:09:32.124: BR0:1 LCP: MagicNumber 0x5785162C (0x05065785162C)
*Mar 1 01:09:32.128: BR0:1 LCP: O CONFACK [ACKrcvd] id 7 len 15
*Mar 1 01:09:32.132: BR0:1 LCP: AuthProto CHAP (0x0305C22305)
*Mar 1 01:09:32.136: BR0:1 LCP: MagicNumber 0x5785162C (0x05065785162C)
*Mar 1 01:09:32.136: BR0:1 LCP: State is Open

```

!--- LCP negotiation is complete.

```

*Mar 1 01:09:32.140: BR0:1 PPP: Phase is AUTHENTICATING, by both [0 sess,0 load]
*Mar 1 01:09:32.144: BR0:1 CHAP: O CHALLENGE id 20 len 26 from "kevin"
*Mar 1 01:09:32.168: BR0:1 CHAP: I CHALLENGE id 19 len 28 from "krimson"
*Mar 1 01:09:32.172: BR0:1 CHAP: Waiting for peer to authenticate first
*Mar 1 01:09:32.184: BR0:1 CHAP: I RESPONSE id 20 len 28 from "krimson"
*Mar 1 01:09:32.188: BR0:1 CHAP: O SUCCESS id 20 len 4
*Mar 1 01:09:32.192: BR0:1 CHAP: Processing saved Challenge, id 19
*Mar 1 01:09:32.204: BR0:1 CHAP: O RESPONSE id 19 len 26 from "kevin"
*Mar 1 01:09:32.224: BR0:1 CHAP: I SUCCESS id 19 len 4

```

```

!--- CHAP authentication is successful.

*Mar 1 01:09:32.228: BR0:1 PPP: Phase is UP [0 sess, 0 load]
*Mar 1 01:09:32.232: BR0:1 IPCP: O CONFREQ [Closed] id 18 len 10
*Mar 1 01:09:32.236: BR0:1 IPCP:   Address 10.9.8.2 (0x03060A090802)
*Mar 1 01:09:32.244: BR0:1 IPCP: I CONFREQ [REQsent] id 30 len 10
*Mar 1 01:09:32.248: BR0:1 IPCP:   Address 10.9.8.1 (0x03060A090801)
*Mar 1 01:09:32.252: BR0:1 IPCP: O CONFACK [REQsent] id 30 len 10
*Mar 1 01:09:32.256: BR0:1 IPCP:   Address 10.9.8.1 (0x03060A090801)
*Mar 1 01:09:32.264: BR0:1 IPCP: I CONFACK [ACKsent] id 18 len 10
*Mar 1 01:09:32.268: BR0:1 IPCP:   Address 10.9.8.2 (0x03060A090802)
*Mar 1 01:09:32.272: BR0:1 IPCP: State is Open

```

```

!--- IPCP negotiation is complete.

```

```

*Mar 1 01:09:32.276: BR0:1 DDR: dialer protocol up
*Mar 1 01:09:32.288: Di1 IPCP: Install route to 10.9.8.1
*Mar 1 01:09:33.228: %LINEPROTO-5-UPDOWN: Line protocol on
Interface BRI0:1, changed state to up
*Mar 1 01:09:35.908: %ISDN-6-CONNECT:
Interface BRI0:1 is now connected to 6119 krimson

```

```

!--- Call is connected and can now pass traffic.

```

```

kevin#

```

This **debug** was obtained from the callback server. Note that the server receives the initial call from the client, recognizes the call as eligible for callback, disconnects, and initiates a callback. You can correlate the messages in this debug to the client debugs.

```

krimson#show debug
Dial on demand:
Dial on demand events debugging is on
PPP:
PPP authentication debugging is on
PPP protocol negotiation debugging is on
ISDN:
ISDN events debugging is on
ISDN Q931 packets debugging is on
Callback:
Callback activity debugging is on
krimson#
*Mar 17 23:38:50.538: ISDN BR0: RX <- SETUP pd = 8 callref = 0x0B
*Mar 17 23:38:50.546:   Bearer Capability i = 0x8890
*Mar 17 23:38:50.550:   Channel ID i = 0x89
*Mar 17 23:38:50.558:   Calling Party Number i = 0xA1, '6120',
Plan:ISDN, Type:National

!--- Calling party information (caller ID) provided by the switch.
!--- The caller ID matches the number specified using the command
!--- dialer caller 6120 callback on the server.
!--- The call is rejected and the callback is initiated.

*Mar 17 23:38:50.570: Called Party Number i = 0xC1, '6119',
Plan:ISDN, Type:Subscriber(local)

!--- Called party information provided by the switch.

*Mar 17 23:38:50.590: CCBRI_Go Fr L3 pkt (Len=25) :
*Mar 17 23:38:50.590: 5 1 8B 90 4 2 88 90 18 1 89 6C 5 A1 36 31 32 30 70 5 C1 36 31 31 39
*Mar 17 23:38:50.602:
*Mar 17 23:38:50.602: ISDN BR0: Incoming call id = 0x004F, dsl 0
*Mar 17 23:38:50.606: ISDN BR0: LIF_EVENT: ces/callid 1/0x4F HOST_INCOMING_CALL
*Mar 17 23:38:50.610: ISDN BR0: HOST_INCOMING_CALL: (non-POTS) DATA
*Mar 17 23:38:50.614: ISDN BR0: HOST_INCOMING_CALL: (1) call_type = DATA

```

*Mar 17 23:38:50.614: ISDN BR0: HOST_INCOMING_CALL:
voice_answer_data =FALSE call type is DATA
*Mar 17 23:38:50.618: ISDN BR0: Event: Received a DATA call from 6120
on B1 at 64 Kb/s
*Mar 17 23:38:50.622: ISDN BR0: Event: Accepting the call id 0x4F
*Mar 17 23:38:50.626: **Di1 DDR: Caller id Callback server starting to kevin**

!--- DDR prepares for the callback.

*Mar 17 23:38:50.626: ISDN BR0: process_disconnect(): call id 0x4F,
call type is DATA, b_idb 0x221DA8, ces 1, cause Call rejected(0x15)
*Mar 17 23:38:50.634: ISDN: get_isdn_service_state(): idb 0x221DA8
bchan 2 is_isdn 1 Not a Pri
*Mar 17 23:38:50.638: ISDN BR0: HOST_INCOMING_CALL: DIALER ERROR 1:
bchan 0, call id 4F
*Mar 17 23:38:50.638: CCBRI_Go Fr Host InPkgInfo (Len=13) :
*Mar 17 23:38:50.642: 5 0 1 0 4F 3 8 1 95 8 2 80 95
*Mar 17 23:38:50.646:
*Mar 17 23:38:50.650: ISDN BR0: LIF_EVENT: ces/callid 1/0x4FHOST_DISCONNECT_ACK
*Mar 17 23:38:50.654: ISDN: get_isdn_service_state(): idb 0x221DA8
bchan 2 is_isdn 1 Not a Pri
*Mar 17 23:38:50.658: ISDN BR0: HOST_DISCONNECT_ACK: call type is DATA
*Mar 17 23:38:50.670: ISDN BR0: TX -> RELEASE_COMP pd = 8 callref = 0x8B
*Mar 17 23:38:50.674: **Cause i = 0x8095 - Call rejected**

!--- Call is "Rejected". Callback now begins.

*Mar 17 23:38:55.622: **DDR: Callback timer expired**

*!--- The callback timer (5 seconds), specified using the
!--- dialer enable-timeout command has expired.*

*Mar 17 23:38:55.622: **Di1 DDR: beginning callback to kevin**

!--- Callback begins.

*Mar 17 23:38:55.626: BR0 DDR: rotor dialout [priority]
*Mar 17 23:38:55.626: BR0 DDR: Attempting to dial 6120
*Mar 17 23:38:55.630: ISDN BR0: Outgoing call id = 0x8074, dsl 0
*Mar 17 23:38:55.630: DDR: Freeing callback to kevin
*Mar 17 23:38:55.634: ISDN BR0: Event: Call to 6120 at 64 Kb/s
*Mar 17 23:38:55.634: ISDN BR0: process_bri_call(): call id 0x8074,
called_number 6120, speed 64, call type DATA
*Mar 17 23:38:55.638: CCBRI_Go Fr Host InPkgInfo (Len=20) :
*Mar 17 23:38:55.642: 1 0 1 80 74 0 4 2 88 90 18 1 83 70 5 80 36 31 32 30
*Mar 17 23:38:55.650:
*Mar 17 23:38:55.650: CC_CHAN_GetIdleChanbri: dsl 0
*Mar 17 23:38:55.654: Found idle channel B1
*Mar 17 23:38:55.666: ISDN BR0: TX -> SETUP pd = 8 callref = 0x74
*Mar 17 23:38:55.670: Bearer Capability i = 0x8890
*Mar 17 23:38:55.678: Channel ID i = 0x83
*Mar 17 23:38:55.682: Called Party Number i = 0x80, '6120',
Plan:Unknown, Type:Unknown
*Mar 17 23:38:55.758: ISDN BR0: RX <- CALL_PROC pd = 8 callref = 0xF4
*Mar 17 23:38:55.762: Channel ID i = 0x89
*Mar 17 23:38:55.778: CCBRI_Go Fr L3 pkt (Len=7) :
*Mar 17 23:38:55.778: 2 1 74 98 18 1 89
*Mar 17 23:38:55.782:
*Mar 17 23:38:55.786: ISDN BR0: LIF_EVENT: ces/callid 1/0x8074HOST_PROCEEDING
*Mar 17 23:38:55.786: ISDN BR0: HOST_PROCEEDING
*Mar 17 23:38:55.790: ISDN BR0: HOST_MORE_INFO
*Mar 17 23:38:56.062: ISDN BR0: **RX <- CONNECT** pd = 8 callref = 0xF4
*Mar 17 23:38:56.078: CCBRI_Go Fr L3 pkt (Len=4) :
*Mar 17 23:38:56.078: 7 1 74 91
...

*!--- The call is connected and LCP, authentication and IPCP negotiations are performed.
!--- The **debug** output is omitted here for brevity.*

```
*Mar 17 23:38:58.282: BR0:1 DDR: dialer protocol up
*Mar 17 23:38:58.290: Di1 IPCP: Install route to 10.9.8.2
*Mar 17 23:38:59.230: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:1,
changed state to up
*Mar 17 23:39:02.094: %ISDN-6-CONNECT: Interface BRI0:1 is now connected to 6120 kevin
```

!--- Call is connected and traffic can pass across the link.

Related Information

- **ISDN Authentication and Callback with Caller ID**
- **Configuring ISDN Caller ID Callback**
- **Redial Enhancements**
- **Access Technology Support**
- **Technical Support – Cisco Systems**

All contents are Copyright © 2006–2007 Cisco Systems, Inc. All rights reserved. Important Notices and Privacy Statement.

Updated: Jun 01, 2005

Document ID: 4686
