

The Media Inactive Call Detection feature enhances Cisco IOS behavior for disconnecting a call when an inactive condition is detected. The former behavior automatically disconnected inactive calls. The current feature provides more control for managing these calls.

Guide Contents
Troubleshooting Cisco IOS Voice Overview
Debug Command Output on Cisco IOS Voice Gateways
Filtering Troubleshooting Output
Cisco VoIP Internal Error Codes
Troubleshooting Cisco IOS Voice Telephony
Troubleshooting Cisco IOS Voice Protocols
Troubleshooting Cisco IOS Telephony Applications
Monitoring the Cisco IOS Voice Network
Cause Codes and Debug Values

Contents

- [1 Media Inactive Call Detection Overview](#)
- [2 Prerequisites for Media Inactive Call Detection](#)
- [3 Restrictions for Media Inactive Call Detection](#)
- [4 Information about Media Inactive Call Detection](#)
 - ◆ [4.1 Improved Functionality of Media Inactive Call Detection](#)
 - ◇ [4.1.1 Legacy Functionality](#)
 - ◇ [4.1.2 Current Functionality](#)
 - ◆ [4.2 Modifications to Information Tags and Internal Error Codes](#)
 - ◇ [4.2.1 evt feature report](#)
 - [4.2.1.1 Description](#)
 - [4.2.1.2 Syntax](#)
 - [4.2.1.3 Mode](#)
 - [4.2.1.4 Scope](#)
 - [4.2.1.5 Return Type](#)
 - [4.2.1.6 Direct Mapping](#)
 - [4.2.1.7 Event Names:](#)
 - [4.2.1.8 Example](#)
 - ◇ [4.2.2 evt feature type](#)
 - [4.2.2.1 Description](#)
 - [4.2.2.2 Syntax](#)
 - [4.2.2.3 Mode](#)
 - [4.2.2.4 Scope](#)
 - [4.2.2.5 Return Type](#)
 - [4.2.2.6 Direct Mapping](#)
 - [4.2.2.7 Event Names](#)
 - ◇ [4.2.3 evt feature param](#)
 - [4.2.3.1 Description](#)
 - [4.2.3.2 Syntax](#)
 - [4.2.3.3 Mode](#)
 - [4.2.3.4 Scope](#)
 - [4.2.3.5 Return Type](#)
 - [4.2.3.6 Direct Mapping](#)
 - [4.2.3.7 Event Parameter](#)
 - [4.2.3.8 Example](#)

◇ [4.2.4 media_timer_factor](#)

- [4.2.4.1 Description](#)
- [4.2.4.2 Syntax:](#)
- [4.2.4.3 Mode](#)
- [4.2.4.4 Scope](#)
- [4.2.4.5 Return Type](#)
- [4.2.4.6 Direct Mapping](#)
- [4.2.4.7 Timer factor:](#)
- [4.2.4.8 Example:](#)

◇ [4.2.5 media_inactivity_err](#)

- [5 Configuring Media Inactive Call Detection](#)
 - ◆ [5.1 SUMMARY STEPS](#)
 - ◆ [5.2 DETAILED STEPS](#)
- [6 Output Examples for Media Inactive Call Detection](#)
 - ◆ [6.1 show call active voice brief Command: Example](#)
 - ◆ [6.2 show running config Command: Example](#)
 - ◆ [6.3 Sample Tcl IVR script](#)

Media Inactive Call Detection Overview


The Media Inactive Call Detection feature detects inactive (silent) H.323 or SIP call-legs on Cisco IOS-based gateways, and reports this situation to the Tcl IVR 2.0 application (which can disconnect the call). When the Media Inactive Call Detection feature is enabled, Cisco IOS software does not automatically disconnect detected inactive calls. Inactivity is defined as no RTP/RTCP packets for a configurable length of time.

The command-line interface enables system administrators to do the following:

- Use additional filters to display one or more calls detected and reported as inactive
- Manually release a call by entering the called number, the calling number, or the call ID

The Media Inactive Call Detection feature enables a system administrator to view all detected inactive calls that have been reported to the Tcl script. An internal error code (IEC) is generated when an inactive call is requested to be cleared via the CLI command.

Prerequisites for Media Inactive Call Detection

 **Note:** Before the following step is done, you should set the inactive timer. This can be done using the **timer receive-rtcp** command in the CLI or it can be set in the script itself using **infotag set media_timer_factor**.

The Media Inactive Call Detection feature requires Cisco IOS Release 12.3(4)T or later.

To enable the Media Inactive Call Detection feature, set the information tag **evt_feature_report** using **media_inactivity** type. For example:

```
infotag set evt_feature_report media_inactivity
```

Restrictions for Media Inactive Call Detection

- The Media Inactive Call Detection feature does not change the existing behavior for the default session application and Tcl IVR 1.0 or the existing Tcl IVR 2.0 script behavior that does not request the new feature.

- This feature does not support MGCP call legs.
- The Media Inactive Call Detection feature works in IP only. This feature does not include PSTN inactive call detection.
- This feature supports RTP/RTCP media inactivity detection and notification only on H.323 and SIP basic calls.

Information about Media Inactive Call Detection

This section provides information about the configuration of Media Inactive Call Detection.

- [Improved Functionality of Media Inactive Call Detection](#)
- [Modifications to Information Tags and Internal Error Codes](#)

Improved Functionality of Media Inactive Call Detection

Legacy Functionality

This functionality is an enhancement to the pre-existing Media Inactivity Timer feature, which enables gateways to monitor and disconnect VoIP calls if no RTCP packets are received within a configurable time period.

The Media Inactivity Timer feature requires the configuration of the Cisco IOS **ip rtcp report interval** command and the **timer receive-rtcp** command to enable detection of RTCP packets by the gateway. When these commands are configured, the gateway uses RTCP report detection, rather than RTP packet detection, to determine whether calls on the gateway are still active or should be disconnected.

If no RTCP packets are received in the resulting time period, the call is disconnected.

The **ip rtcp report interval** command configures the RTCP reporting interval in milliseconds in the range of 1 to 65535. The **timer receive-rtcp** command configures the multiplier in the range of 2 to 1000. These values can be adjusted depending on network traffic conditions. Under normal conditions, a value of **5000** for the **ip rtcp report interval** and a value of **5** for the **timer receive-rtcp** are typical.

Current Functionality

The Media Inactive Call Detection feature offers the following:

- The **show call active** command indicates that a call has no RTP or RTCP inactivity.
- The **clear call** command offers options so that an inactive call can be released using the called number or calling number. The **clear call** command has also been enhanced to configure the Q.850 release cause code to be used when the call is released.

Modifications to Information Tags and Internal Error Codes

This feature includes modifications to two information tags, the addition of two information tags, and an internal error code:

- [evt feature report](#)
- [evt feature type](#)
- [evt feature param](#)
- [media timer factor](#)
- [media inactivity err](#)

For more detailed information, refer to the [Cisco IOS Tcl IVR and VoiceXML Application Guide](#) and the [TCL IVR API Version 2.0 Programming Guide](#).

evt_feature_report

This existing tag has a new event name (**media_inactivity**) added for the Tcl script to request notification of media inactivity detection.

Description

To enable/disable certain feature events to be intercepted by the script

Syntax

Infotag set evt_feature_report [{"no_"}*event_names*]

Where *event_name* is a list of application event names that define what events should or should not be reported to application when call is active (connected). An event name with "no_" prefix means not to report it.

Mode

Write

Scope

ev_feature

Return Type

None

Direct Mapping

None

Event Names:

fax
modem
modem_phase
hookflash
onhook
offhook
media_inactivity

Example

The following example enables hookflash and disable fax and modem feature events to be received by the script:

```
infotag set evt_feature_report hookflash nofax no modem
```

The following example enables media_inactivity event to be received by the script:

```
infotag set evt_feature_report media_inactivity
```

evt_feature_type

This existing information tag adds two new event feature types representing media inactivity notification and media activity notification. Note that the script only needs to request for report type media_inactivity. However, after media inactivity is reported and the VoIP RTP starts receiving RTP/RTCP packets again, the event with type media_activity is automatically notified, indicating that the call is back alive.

Description

To return the feature type string when a feature event is received

Syntax

```
infotag get evt_feature_type
```

Mode

Read

Scope

ev_feature

Return Type

String

Direct Mapping

None

Event Names

fax
modem
modem_phase
hookflash
onhook
offhook
media_inactivity
media_activity

evt_feature_param

This is a new information tag added so that the Tcl application can pass along parameters related to the feature back to the script. The Media Inactive Call Detection feature uses this new tag to pass the information on whether RTCP packet has been received before the media inactive condition is met.

Description

To return a parameter related to a specific feature event

Syntax

infotag get evt_feature_param parameter_name

Mode

Read

Scope

ev_feature

Return Type

String

Direct Mapping

None

Event Parameter

media_inactivity_type-This parameter belongs to feature **media_inactivity**. The return string is:

- ◇ **no media received**-Media inactivity detected (no RTP or RTCP packets have been received for a configured amount of time). RTCP packet has been received before media inactivity condition is met.
- ◇ **no control info received**-Media inactivity detected (no RTP or RTCP packets have been received for a configured amount of time). No RTCP packet has been received before media inactivity condition is met.

Example

infotag get evt_feature_param media_inactivity_type

media_timer_factor

This new information tag gives the Tcl script the ability to overwrite the configured gateway **receive-rtcp** timer. This value is used to calculate the timeout value used to detect media inactivity.

Description

To set the timer receive-rtcp timer. This new value is used within the scope of the script. It does not change the gateway configuration.

Syntax:

infotag set media_timer_factor timer_factor

Mode

Write

Scope

None

Return Type

None

Direct Mapping


None

Timer factor:

An integer between 2 and 1000. This value is the multiple of RTCP report transmission interval-a value of 5 is recommended.

Example:

```
infotag set media_timer_factor 6
```

 **Note:** If the value specified is not between 2 and 1000, the script generates an error message.

media_inactivity_err

The internal error code used by the Tcl script for this feature is **media_inactivity_err** (common IEC error #8). This IEC is used to disconnect a call where media inactivity is detected and reported.

Configuring Media Inactive Call Detection

This feature is enabled using Tcl IVR 2.0.

This section describes the use of this feature to set parameters for determining call inactive status and then clearing specific calls. These new options are designed to work as filters.

SUMMARY STEPS

1. **enable**
2. Make a call using the Tcl IVR 2.0 script application. While the call is going, the commands in steps 3 and 4 can be used.
3. **show call active voice** [brief] [called-number number | calling-number number | compact | echo-canceller | id | media-inactive {called-number number | calling-number number}]
4. **clear call voice causecode** <1-127> {id <1-FFFF> | **media-inactive**}

DETAILED STEPS

	Command or Action	Purpose
1.	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
2.	Make a call using the TCL IVR script application	

		Starts a call. While the call is going, the commands in steps 3 and 4 can be used.
3.	<p>show call active voice [brief] [called-number number calling-number number compact echo-canceller id media-inactive {called-number number calling-number number}]</p> <p>Example:</p> <pre>Router# show call active voice media-inactive calling-number 4085551234 Router# show call active voice brief media-inactive called-number 4085554321</pre>	<p>Displays calls that have no RTP or RTCP activity.</p> <ul style="list-style-type: none"> • The brief keyword shows the brief version of active voice calls. • The called-number keyword shows only the call with the specified called number pattern. • The calling-number keyword shows only the call with the specified calling number pattern. • The compact keyword shows a compact version of the active voice calls. • The echo-canceller keyword shows echo canceller data for an active voice call. • The id keyword shows only the call with the specified ID. • The media-inactive keyword shows only the calls with media inactive being detected and notified. • The number argument is a sequence of digits representing a full, recognizable telephone number.

<p>clear call voice causecode > {id >media-inactive}</p> <p>4. Example:</p> <pre>Router# clear call voice causecode id 112B</pre>	<p>Clears calls that show media inactive and can clear a specific call.</p> <ul style="list-style-type: none"> • The causecode keyword is a Q.850 disconnect cause code. The cause code can be specified as a number 1 through 127. • The id keyword can be used so that only the voice call with the specified ID is disconnected. • The 1-FFFF range is a call identifier as shown in brief format. • The media-inactive keyword clears only calls with media inactivity detected and notified.
--	---

Output Examples for Media Inactive Call Detection

The examples provided in this section include the following:

- [show call active voice brief Command: Example](#)
- [show running config Command: Example](#)
- [Sample Tcl IVR script](#)

show call active voice brief Command: Example

The existing **show call active voice brief** command has additional media inactive detection data in the IP call leg information.

```
<ID>: <start>hs.<index> +<connect> pid:<peer_id> <dir> <addr> <state>
dur hh:mm:ss tx:<packets>/<bytes> rx:<packets>/<bytes>
IP <ip>:<udp> rtt:<time>ms pl:<play>/<gap>ms lost:<lost>/<early>/<late>
delay:<last>/<min>/<max>ms <codec>
media inactive detected:<y/n> media cntrl rcvd:<y/n> timestamp:<time>
----- the above line is new -----
MODEMPASS <method> buf:<fills>/<drains> loss <overall%> <multipkt>/<corrected>
last <buf event time>s dur:<Min>/<Max>s
FR <protocol> [int dlci cid] vad:<y/n> dtmf:<y/n> seq:<y/n>
<codec> (payload size)
ATM <protocol> [int vpi/vci cid] vad:<y/n> dtmf:<y/n> seq:<y/n>
<codec> (payload size)
```

Cisco_IOS_Voice_Troubleshooting_and_Monitoring_--_Media_Inactive_Call_Detection

```
Tele <int>: tx:<tot>/<v>/<fax>ms <codec> noise:<l> acom:<l> i/o:<l>/<l> dBm
MODEMRELAY info:<rcvd>/<sent>/<resent> xid:<rcvd>/<sent> total:<rcvd>/<sent>/<drops>
  speeds(bps): local <rx>/<tx> remote <rx>/<tx>
Proxy <ip>:<audio udp>,<video udp>,<tcp0>,<tcp1>,<tcp2>,<tcp3> endpt: <type>/<manf>
bw: <req>/<act> codec: <audio>/<video>
  tx: <audio pkts>/<audio bytes>,<video pkts>/<video bytes>,<t120 pkts>/<t120 bytes>
  rx: <audio pkts>/<audio bytes>,<video pkts>/<video bytes>,<t120 pkts>/<t120 bytes>
Telephony call-legs: 1
SIP call-legs: 0
H323 call-legs: 1
Total call-legs: 2
11DF : 239062hs.1 +2 pid:1 Answer 4085254616 active
  dur 00:01:04 tx:2383/89775 rx:1187/23318
  Tele 3:D:13: tx:45900/2110/0ms g729r8 noise:-69 acom:45 i/0:-71/-29 dBm
11DF : 239684hs.1 +830 pid:1800877 Originate 18008770519 active
  dur 00:00:49 tx:1066/20965 rx:2378/47215
  IP 1.9.57.5:17750 rtt:5ms pl:38190/0ms lost:0/1/0 delay:50/50/70ms g729r8
  media inactive detected:y media cntrl rcv:y timestamp: 12595
  <----- the above line is new ----->
Telephony call-legs: 1
SIP call-legs: 0
H323 call-legs: 1
Total call-legs: 2
```

For calls without media inactivity being detected, the display looks like the following:

```
...
11DF : 239062hs.1 +2 pid:1 Answer 4085254616 active
  dur 00:01:04 tx:2383/89775 rx:1187/23318
  Tele 3:D:13: tx:45900/2110/0ms g729r8 noise:-69 acom:45 i/0:-71/-29 dBm
11DF : 239684hs.1 +830 pid:1800877 Originate 18008770519 active
  dur 00:00:49 tx:1066/20965 rx:2378/47215
  IP 1.9.57.5:17750 rtt:5ms pl:38190/0ms lost:0/1/0 delay:50/50/70ms g729r8
  media inactive detected:y media cntrl rcv:n/a timestamp: n/a
  <----- the above line is new ----->
Telephony call-legs: 1
SIP call-legs: 0
H323 call-legs: 1
Total call-legs: 2
```

The long display form of active call records generated by the **show call active voice** command is also modified to add the media inactive detected information.

```
Router_5300# show call active voice
Telephony call-legs: 1
SIP call-legs: 0
H323 call-legs: 1
Total call-legs: 2
  GENERIC:
SetupTime=239062 ms
Index=1
PeerAddress=4085254616
PeerSubAddress=
PeerId=1
...
  TELE:
ConnectionId=[0xB21F398F 0x1DA111D4 0x800B85CB 0x3E43F332]
IncomingConnectionId=[0xB21F398F 0x1DA111D4 0x800B85CB 0x3E43F332]
TxDuration=51250 ms
VoiceTxDuration=2110 ms
```

Cisco_IOS_Voice_Troubleshooting_and_Monitoring_--_Media_Inactive_Call_Detection

```
...
  GENERIC:
SetupTime=239684 ms
Index=1
PeerAddress=18008770519
PeerSubAddress=
...
  VOIP:
ConnectionId[0xB21F398F 0x1DA111D4 0x800B85CB 0x3E43F332]
IncomingConnectionId[0xB21F398F 0x1DA111D4 0x800B85CB 0x3E43F332]
RemoteIPAddress=1.9.57.5
...
TranslatedRedirectCalledNumber=
TranslatedRedirectCalledOctet=0x7F
MediaInactiveDetected=yes <---- new
MediaInactiveTimestamp=12595 <---- new
MediaControlReceived=yes <---- new
Username=Telephony call-legs: 1
SIP call-legs: 0
H323 call-legs: 1
Total call-legs: 2
```

For calls where no media inactivity is detected and notified, the above three fields are displayed as follows:

```
TranslatedRedirectCalledNumber=
TranslatedRedirectCalledOctet=0x7F
MediaInactiveDetected=no <---- new
MediaInactiveTimestamp= <---- new
MediaControlReceived= <---- new
Username=Telephony call-legs: 1
SIP call-legs: 0
H323 call-legs: 1
Total call-legs: 2
```

show running config Command: Example

```
router# show running config
Building configuration...
This command has no effect on this line; use modem AT commands instead
This command has no effect on this line; use modem AT commands instead
Current configuration : 13850 bytes
!
version 12.2
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname "jc5400"
!
no boot startup-test
logging buffered 2000000 debugging
no logging console
enable secret 5 $1$afrj$LWwkVSLZ3cKak3OkHsAMt/
enable password lab
!
username 1111
username 2222 password 0 2222
username 123001 password 0 1001
username cisco
!
```

show running config Command: Example

Cisco_IOS_Voice_Troubleshooting_and_Monitoring_--_Media_Inactive_Call_Detection

```
!  
resource-pool disable  
clock timezone GMT -8  
tdm clock priority 1 6/0  
spe default-firmware spe-firmware-1  
aaa new-model  
!  
!  
aaa authentication login h323 local group radius  
aaa authentication login telnet none  
aaa authorization exec h323 local group radius  
aaa authorization exec telnet none  
aaa accounting connection h323 start-stop group radius  
aaa session-id common  
ip subnet-zero  
ip cef  
ip ftp username dump  
ip ftp password dump123  
no ip domain lookup  
ip host tftp-server1 10.1.1.211  
ip host rtsp-server1 10.1.1.211  
ip host radius-server1 10.1.1.211  
ip host gwip-server1 10.1.1.211  
ip host dump-server1 10.1.1.211  
!  
isdn switch-type primary-5ess  
isdn voice-call-failure 0  
!  
voice call carrier capacity active  
!  
!  
!  
voice cause-code  
!  
no voice hpi capture buffer  
no voice hpi capture destination  
!  
!  
ivr prompt memory 16384  
!  
fax interface-type fax-mail  
mta receive maximum-recipients 600  
!  
!  
!  
controller T1 6/0  
  framing esf  
  linecode b8zs  
  pri-group timeslots 1-24  
  no yellow generation  
  no yellow detection  
!  
controller T1 6/1  
  shutdown  
  framing sf  
  linecode ami  
  no yellow generation  
  no yellow detection  
!  
controller T1 6/2  
  shutdown  
  framing sf  
  linecode ami  
  no yellow generation
```

show running config Command: Example

Cisco_IOS_Voice_Troubleshooting_and_Monitoring_--_Media_Inactive_Call_Detection

```
no yellow detection
!
controller T1 6/3
shutdown
framing sf
linecode ami
no yellow generation
no yellow detection
!
controller T1 6/4
shutdown
framing sf
linecode ami
no yellow generation
no yellow detection
!
controller T1 6/5
shutdown
framing sf
linecode ami
no yellow generation
no yellow detection
!
controller T1 6/6
shutdown
framing sf
linecode ami
no yellow generation
no yellow detection
!
controller T1 6/7
shutdown
framing sf
linecode ami
no yellow generation
no yellow detection
gw-accounting aaa
!
!
!
interface FastEthernet0/0
ip address 10.1.1.212 255.255.0.0
no ip redirects
no ip mroute-cache
duplex auto
speed auto
no cdp enable
!
interface FastEthernet0/1
no ip address
no ip redirects
no ip mroute-cache
shutdown
duplex auto
speed auto
no cdp enable
!
interface Serial0/0
no ip address
no ip mroute-cache
shutdown
clockrate 2000000
no cdp enable
!
```

show running config Command: Example

Cisco_IOS_Voice_Troubleshooting_and_Monitoring_--_Media_Inactive_Call_Detection

```
interface Serial6/0
  no ip address
  shutdown
!
interface Serial10/1
  no ip address
  no ip mroute-cache
  shutdown
  clockrate 2000000
  no cdp enable
!
interface Serial6/0:23
  no ip address
  isdn switch-type primary-5ess
  isdn incoming-voice modem
  isdn bchan-number-order ascending
  no keepalive
  no cdp enable
!
interface Group-Async0
  no ip address
  no ip mroute-cache
  group-range 1/00 1/107
!
ip classless
ip route 10.1.0.0 255.255.0.0 10.1.1.211
ip http server
!
ip pim bidir-enable
ip rtcp report interval 5000
!
!
no logging trap
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
!
!
radius-server host 10.1.1.211 auth-port 1645 acct-port 1646
radius-server key cisco
radius-server authorization permit missing Service-Type
radius-server vsa send accounting
radius-server vsa send authentication
!
call application voice testapp_JC tftp://10.1.1.211/Scripts/JC_app1.tcl
call rsvp-sync
!
voice-port 6/0:D
!
!
mgcp profile default
!
dial-peer cor custom
!
!
!
dial-peer voice 1000 pots
  application testapp_JC
  incoming called-number 5551001
  direct-inward-dial
  port 6/0:D
!
!
dial-peer voice 3000 voip
  destination-pattern 5551001
```

show running config Command: Example

Cisco_IOS_Voice_Troubleshooting_and_Monitoring_--_Media_Inactive_Call_Detection

```
session target ipv4:10.1.1.213
dtmf-relay h245-signal
codec g711ulaw
!
!
gateway
timer receive-rtcp 5
!
sip-ua
!
!
line con 0
exec-timeout 0 0
logging synchronous
line aux 0
logging synchronous
line vty 0 4
password lab
authorization exec telnet
login authentication telnet
line 1/00 1/107
no flush-at-activation
modem InOut
!
exception core-file jc5400_core
exception protocol ftp
exception dump 10.1.1.211
scheduler allocate 10000 400
end
```

Sample Tcl IVR script

The following is a sample script that is provided for reference purposes only.

```
silence_detect_demo.tcl
#-----
# Copyright (c) 2003 by cisco Systems, Inc.
# All rights reserved.
#-----
#
# This tcl demo script monitors Media Inactive Call. The Media Inactive Call
# Detection feature detects inactive (silent)H.323 or Sip call-legs on Cisco
# IOS based gateways, and reports this situation to the TCL IVR 2.0 application
# and TCL IVR application checks for the events and logs those events.
#
# This script is designed to place a call to the dnis if DID is configured.
# Otherwise, output dial-tone and collects digits from the caller against
# the dial-plan. If an inactive condition 'ev_feature' is detected then script
# begins to log the inactivity events. However, if the VOIP RTP starts receiving
# RTP/RTCP packets again, then event with media_activity type will be automatically
# notified, indicating that the call is back alive.
#
#-----
# Example Script
#-----
proc init { } {
    global param
    global timerFactor
    set param(interruptPrompt) true
    set param(abortKey) *
    set param(terminationKey) #
    set timerFactor 6
}
```

Cisco_IOS_Voice_Troubleshooting_and_Monitoring_--_Media_Inactive_Call_Detection

```
proc act_Setup { } {
    global dest
    if { [infotag get leg_isdid] } {
        set dest [infotag get leg_dnis]
        leg proceeding leg_incoming
        leg setup $dest callInfo leg_incoming
        fsm setstate PLACECALL
    } else {
        leg setupack leg_incoming
        playtone leg_incoming tn_dial
        set param(dialPlan) true
        leg collectdigits leg_incoming param
    }
}
proc act_GotDest { } {
    global dest
    set status [infotag get evt_status]
    if { $status == "cd_004" } {
        set dest [infotag get evt_dcdigits]
        leg proceeding leg_incoming
        leg setup $dest callInfo leg_incoming
    } else {
        call close
    }
}
proc act_CallSetupDone { } {
    infotag set evt_feature_report media_inactivity
    infotag set media_timer_factor $timerFactor
    set status [infotag get evt_status]
    if { $status == "ls_000" } {
    } else {
        call close
    }
}
proc act_EvFeatureReceived { } {
    global timerFactor
    set featureType [infotag get evt_feature_type]
    if { $featureType == "media_inactivity" } {
        log -s "media inactivity or silence is detected"
        set inactivity_type [infotag get evt_feature_param media_inactivity_type]
        if { $inactivity_type == "no media received" } {
            log -s "media inactivity, RTCP packet was previously received"
        }
        if { $inactivity_type == "no control info received" } {
            log -s "media inactivity, no RTCP packet was previously received "
        }
    } elseif { $featureType == "media_activity" } {
        log -s "media activity detected and call is back alive, VOIP RTP starts receiving
RTP/RTCP packets"
    } else {
        log -s "other Events have been detected"
    }
}
proc act_Cleanup { } {
    call close
}
init
#-----
#   State Machine
#-----
set fsm(any_state,ev_disconnected)      "act_Cleanup      same_state"
set fsm(CALL_INIT,ev_setup_indication)  "act_Setup        GETDEST"
set fsm(GETDEST,ev_collectdigits_done)  "act_GotDest      PLACECALL"
set fsm(PLACECALL,ev_setup_done)        "act_CallSetupDone CALLACTIVE"
```


Cisco_IOS_Voice_Troubleshooting_and_Monitoring_--_Media_Inactive_Call_Detection

```
set fsm(CALLACTIVE,ev_feature) "act_EvFeatureReceived same_state"  
set fsm(CALLACTIVE,ev_disconnected) "act_Cleanup CALLDISCONNECT"  
set fsm(CALLDISCONNECT,ev_disconnected) "act_Cleanup same_state"  
set fsm(CALLDISCONNECT,ev_disconnect_done) "act_Cleanup same_state"  
fsm define fsm CALL_INIT  
# End the application
```