show voice trace through shutdown (voice-port)

show voice trace

To display the call trace information about a specified port, use the **show voice trace** command in privileged EXEC mode.

show voice trace *interface-slot* [**detail**]

Syntax Description

interface-slot	Voice interface slot.
detail	(Optional) Displays detailed statistics of the specified port.

Command Default

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.

Usage Guidelines

Use the **show voice trace** command to display the call trace information about specified port. The field descriptions are self-explanatory.

Examples

The following is sample output from the **show voice trace** command:

```
Router# show voice trace 1/1/1 detail

1/1/1 Stack 0:
State Transitions: timestamp (state, event) -> (state, event) ...
96.732 (S_OPEN_PEND, E_DSP_INTERFACE_INFO) ->
96.732 (S_DOWN, E_HTSP_IF_INSERVICE) ->
97.092 (S_OPEN_PEND, E_HTSP_GO_UP) ->
Event Counts (zeros not shown): (event, count)
(E_HTSP_IF_INSERVICE, 1) :(E_HTSP_GO_UP, 1) :(E_DSP_INTERFACE_INFO, 1) :
State Counts (zeros not shown): (state, count)
(S_OPEN_PEND, 2) :(S_DOWN, 1) :
Stack 1:
State Transitions: timestamp (state, event) -> (state, event) ...
97.092 (DID_NULL, E_DSP_SIG_0100) ->
97.092 (DID_INIT, E_HTSP_INSERVE) ->
97.092 (DID_PENDING, E_DSP_SIG_0100) ->
Event Counts (zeros not shown): (event, count)
(E_HTSP_INIT, 1) :(E_HTSP_INSERVE, 1) :(E_DSP_SIG_0100, 2) :
State Counts (zeros not shown): (state, count)
(DID_NULL, 2) :(DID_INIT, 1) :(DID_PENDING, 1) :
```

show voice translation-profile

To display one or more translation profiles, use the **show voice translation-profile** command in privileged EXEC mode.

show voice translation-profile [{*name* | **sort** [{**ascending** | **descending**}]}]

Syntax Description

name	Name of the translation profile to display.
sort [**ascending** | **descending**	Display order of the translation profiles by *name*.

Command Default

Ascending order

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(11)T	This command was introduced.

Examples

The following sample output displays all the voice translation profiles in ascending order:

```
Router# show voice translation-profile sort ascending
Translation Profile: 1
 Rule for Calling number:
 Rule for Called number: 1
 Rule for Redirect number:
Translation Profile: 2
 Rule for Calling number:1
 Rule for Called number: 2
 Rule for Redirect number:
Translation Profile: 6
 Rule for Calling number:1
 Rule for Called number: 6
 Rule for Redirect number:2
```

The table below describes the fields shown in this output.

Table 1: show voice translation-profile Field Descriptions

Field	Description
Translation Profile	Name of the translation profile.
Rule for Called number	Number of the rule used for translating called numbers. If the field is blank, this translation profile does not have a rule assigned to that number type.
Rule for Calling number	Number of the rule used for translating calling numbers. If the field is blank, this translation profile does not have a rule assigned to that number type.
Rule for Redirect number	Number of the rule used for translating redirect numbers. If the field is blank, this translation profile does not have a rule assigned to that number type.

Related Commands

Command	Description
voice translation-profile	Initiates a voice translation-profile definition.
voice translation-rule	Initiates a voice translation-rule definition.

show voice translation-rule

To display one or more translation rules, use the **show voice translation-rule** command in privileged EXEC mode.

show voice translation-rule [{*number* | **sort** [{**ascending** | **descending**}]}]

Syntax Description

number	Number of the translation rule to display. Valid values are from 1 to 2147483647.
sort [**ascending** | **descending**	Display order of the translation rules by *number*.

Command Default

Ascending order

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(11)T	This command was introduced.

Usage Guidelines

Under each translation rule are numbered subrules.

Examples

The following sample output displays the translation rule number 6:

```
Router# show voice translation-rule 6
Translation-rule tag: 6
   Rule 1:
   Match pattern: 65088801..
   Replace pattern: 6508880101
   Match type: none   Replace type: none
   Match plan: none   Replace plan: none
```

The following sample output displays all the translation rules in ascending order:

```
Router# show voice translation-rule sort ascending
Translation-rule tag: 1
   Rule 3:
   Match pattern: 5108880...
   Replace pattern: 5108880101
   Match type: none   Replace type: none
   Match plan: none   Replace plan: none
   Rule 4:
   Match pattern: 510890....
   Replace pattern: 5108880101
   Match type: none   Replace type: none
   Match plan: none   Replace plan: none
Translation-rule tag: 2
   Rule 1:
   Match pattern: 51088802..
   Replace pattern: 5108880101
   Match type: none   Replace type: none
   Match plan: none   Replace plan: none
```

```
Rule 2:
Match pattern: 51088803..
Replace pattern: 5108880101
Match type: none   Replace type: none
Match plan: none   Replace plan: none
Rule 3:
Match pattern: 510889....
Replace pattern: 5108880101
Match type: none  Replace type: none
Match plan: none   Replace plan: none
Rule 4:
Match pattern: 510890....
Replace pattern: 5108880101
Match type: none   Replace type: none
Match plan: none   Replace plan: none
```

The table below describes the fields shown in this output.

Table 2: show voice translation-rule Field Descriptions

Field	Description
Translation-rule tag	Number of the translation rule.
Rule	Number of the rule defined within the translation rule.
Match pattern	SED-like expression used to match incoming call information.
Replace pattern	SED-like expression used to replace *match-pattern* in the call information.
Match type	Type of incoming calls to match.
Replace type	Type to replace Match type.
Match plan	Plan of incoming calls to match.
Replace plan	Plan to replace Match plan.

Related Commands

Command	Description
rule (voice translation-rule)	Defines the SED expressions for translating calls.
test voice translation-rule	Tests the rules in a translation-rule definition.
voice translation-rule	Initiates a voice translation-rule definition.
voice translation-profile	Initiates a voice translation-profile definition.

show voice trunk-conditioning signaling

To display the status of trunk-conditioning signaling and timing parameters for a voice port, use the **show voice trunk-conditioning signaling** command in user EXEC or privileged EXEC mode.

show voice trunk-conditioning signaling [{**summary***voice-port*}]

Syntax Description

summary	(Optional) Displays a summary of the status for all voice ports on the router or concentrator.
voice -port	(Optional) Displays a detailed report for a specified voice port.

Command Modes

User EXEC (>)
Privileged EXEC (#)

Command History

Release	Modification
12.0(3)XG	This command was introduced on the Cisco MC3810 as the **show voice permanent-call** command.
12.0(4)T	This command was integrated into Cisco IOS Release 12.0(4)T.
12.0(7)XK	This command was renamed **show voice trunk-conditioning signaling**.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(3)T	This command was implemented on the Cisco 2600 series and Cisco 3600 series.

Usage Guidelines

This command displays the trunk signaling status for analog and digital voice ports on the Cisco 2600 series and the Cisco 3600 series routers.

Examples

The following is sample output from the **show voice trunk-conditioning signaling summary** command:

```
Router# show voice trunk-conditioning signaling summary
2/0/0 is shutdown
2/0/1 is shutdown
3/0:0 8  is shutdown
3/0:1 1  is shutdown
3/0:2 2  is shutdown
3/0:3 3  is shutdown
3/0:5 5  is shutdown
3/0:6(6) :
 status :
3/0:7 7  is shutdown
3/1:0 8  is shutdown
3/1:1 1  is shutdown
3/1:3 3  is shutdown
3/1:5 5  is shutdown
3/1:7 7  is shutdown
```

The following is sample output from the **show voice trunk-conditioning signaling** command for voice port 3/0:6:

```
Router# show voice trunk-conditioning signaling 3/0:6
hardware-state ACTIVE signal type is NorthamericanCAS
status :
forced playout pattern = STOPPED
trunk_down_timer = 0, rx_ais_duration = 0, idle_timer = 0
```

The table below describes significant fields in these outputs.

Table 3: show voice trunk-conditioning signaling Field Descriptions

Field	Description
current timer	Time since last signaling packets were received.
forced playout pattern	Which forced playout pattern is sent to PBX: • 0 = no forced playout pattern is sent • 1 = receive IDLE playout pattern is sent • 2 = receive OOS playout pattern is sent
hardware-state	Hardware state based on received IDLE pattern: • IDLE = both sides are idle • ACTIVE = at least one side is active
signal type	Signaling type used by lower level driver: northamerica, melcas, transparent, or external.
idle timer	Time the hardware on both sides has been in idle state.
last-ABCD	Last received or transmitted signal bit pattern.
max inter-arrival time	Maximum interval between received signaling packets.
missing	Number of missed signal packets.
mode	Signaling packet generation frequency: • Fast mode = every 4 milliseconds • Slow mode = same frequency as keepalive timer
out of seq	Number of out-of-sequence signal packets.
playout depth	Number of packets in playout buffer.
prev-seq#	Sequence number of previous signaling packet.
refill count	Number of packets created to maintain nominal length of playout packet buffer.
rx_ais_duration	Time since receipt of AIS indicator.

Field	Description
seq#	Sequence number of signaling packet.
sig pkt cnt	Number of transmitted or received signaling packets.
signal path	Status of signaling path.
signaling playout history	Signaling bits received in last 60 milliseconds.
trunk_down_timer	Time since last signaling packets were received.
tx_oos_timer	Time since PBX started sending OOS signaling pattern defined by **signal pattern oos transmit**.
very late	Number of very late signaling packets.

Related Commands

Command	Description
show dial-peer voice	Displays the configuration for all VoIP and POTS dial peers configured on the router.
show voice dsp	Shows the current status of all DSP voice channels.
show voice port	Displays configuration information about a specific voice port.
show voice trunk-conditioning supervisory	Displays the status of trunk supervision and configuration parameters for voice ports.

show voice trunk-conditioning supervisory

To display the status of trunk supervision and configuration parameters for a voice port, use the **show voice trunk-conditioning supervisory** command in user EXEC or privileged EXEC mode.

show voice trunk-conditioning supervisory [{**summary***voice-port*}]

Syntax Description

summary	(Optional) Displays a summary of the status for all voice ports on the router or concentrator.
voice -port	(Optional) Detailed report for a specified voice port.

Command Modes

User EXEC (>)
Privileged EXEC (#)

Command History

Release	Modification
12.0(7)XK	This command was introduced on the Cisco 2600 series, Cisco 3600 series, and Cisco MC3810 platforms.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(3)T	This command was implemented on the Cisco 2600 series and Cisco 3600 series.
12.4(15)T10	The output of thiscommand was modified to report values configured by the **signal timing idle suppress-voice** command. The values for the **suppress-voice** and **resume-voice** keywords are shown as the "idle = *seconds*" and "idle_off = *milliseconds*" fields, respectively.

Usage Guidelines

This command displays the trunk supervision and configuration status for analog and digital voice ports.

Examples

The following is sample output for the **show voice trunk-conditioning supervisory** command:

```
Router# show voice trunk-conditioning supervisory 0/2/00/2/0 : state : TRUNK_SC_PENDING_START,
 voice : on, signal : off,active
 status: trunk disconn
 sequence oos : idle and oos
 pattern :rx_idle = 0101 rx_oos = 1111
 timeout timing : idle = 0, idle_off = 0, restart = 0, standby = 0, timeout = 0
 supp_all = 0, supp_voice = 0, keep_alive = 5
 timer: oos_ais_timer = 0, timer = 3
```

The following is sample output for the **show voice trunk-conditioning supervisory** command for voice port 0/2/0:

```
Router# show voice trunk-conditioning supervisory 3/0:6
0/2/0 : state : TRUNK_SC_PENDING_START, voice : on, signal : off,active
 status: trunk disconn
 sequence oos : idle and oos
 pattern :rx_idle = 0101 rx_oos = 1111
 timeout timing : idle = 0, idle_off = 0, restart = 0, standby = 0, timeout = 0
```

```
 supp_all = 0, supp_voice = 0, keep_alive = 5
 timer: oos_ais_timer = 0, timer = 3
```

The following shows a sample trunk conditioning setting for the **voice class permanent** command and sample output from the **s how voice trunk-conditioning supervisory** command that shows the values for the timeout timing field:

```
!
voice class permanent 1
  signal pattern idle transmit 0101
  signal pattern idle receive 0101
  signal pattern oos transmit 1111
  signal pattern oos receive 0101
  signal timing idle suppress-voice 10 resume-voice 150
!
Router# show voice trunk-conditioning supervisory

SLOW SCAN
0/0/0:0(1) : state : TRUNK_SC_CONNECT, voice : off , signal : on ,inactive
 status: rcv IDLE, trunk connected
 sequence oos : idle and oos
 pattern :rx_idle = 0101 rx_oos = 0101 tx_idle = 0101 tx_oos =  1111
 timeout timing : idle = 10, idle_off = 150, restart = 0, standby = 0, timeout = 30
 supp_all = 0, supp_voice = 0, keep_alive = 5
 timer: oos_ais_timer = 0, timer = 0
```

The table below describes the significant fields shown in the display.

Table 4: show voice trunk-conditioning supervisory Field Descriptions

Field	Description
idle	Timer setting (in seconds) configured by the **suppress-voice** option of the **signal timing idle suppress-voice** command.
idle_off	Timer setting (in milliseconds) configured by the **resume-voice** option of the **signal timing idle suppress-voice** command.
keep_alive	Signaling packets periodically sent to the far end, even if there is no signal change. These signaling packets function as keep alive messages.
active	Voice port configured as "connect trunk *xxxx* ."
oos_ais_timer	Time since the signaling packet with alarm indication signal (AIS) indicator was received.
pattern	4-bit signaling pattern.
restart	Restart timeout after far end is out-of-service (OOS).
rx-idle	Signaling bit pattern indicating that the far end is idle.
rx-oos	Signaling bit pattern sent to the PBX indicating that the network is OOS.
standby	Time before the inactive side goes back to standby after the far end goes OOS.
supp_all	Timeout before suppressing transmission of voice and signaling packets to the far end after detection of PBX OOS.

Field	Description
supp_voice	Timeout before suppressing transmission of voice packet to the far end after detection of PBX OOS.
timeout	Timeout for nonreceipt of keepalive packets before the far end is considered to be OOS.
timeout timing	Delay between the detection of incoming seizure and when the digital signal processor (DSP)-to-Cisco IOS interaction to open up the audio path is initiated.
TRUNK_SC_CONNECT	Trunk conditioning supervisory component status.

Related Commands

Command	Description
show dial-peer voice	Displays the configuration for all VoIP and POTS dial peers configured on the router.
show voice dsp	Displays the current status of all DSP voice channels.
show voice port	Displays configuration information about a specific voice port.
show voice trunk-conditioning signaling	Displays the status of trunk-conditioning signaling and timing parameters for a voice port.
voice-class permanent	Assigns a previously configured voice class for a Cisco trunk or FRF.11 trunk to a voice port.

Show voice vrf

To display the voice VRF configured at global configuration level and IP VRF associated with the bind interface configured under global sip services mode. use **show voice vrf** command in privileged EXEC mode.

show voice vrf

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS 15.6(2)T	This command was introduced.
Cisco IOS XE Denali 16.3.1	This command was integrated into Cisco IOS XE Denali 16.3.1.

Usage Guidelines

Use this command to display information associated with VRF.

Example

If voice vrf VRF1 is configured at global configuration level and sip bind is configured with interface that has vrf id VRF2, then the output is as follows:

```
Device# show voice vrf

==========VOICE VRF CONFIGURATION==========

Global voice vrf defined is: VRF1
Global sip bind for vrf is: VRF2
```

If voice vrf VRF1 is configured and sip bind is configured with interface that is not assigned any vrf id, then the output is as follows:

```
Device# show voice vrf

==========VOICE VRF CONFIGURATION==========

Global voice vrf defined is: VRF1
Global sip bind for vrf is: NA
```

If both voice vrf and sip bind at global level is not configured, then the output is as follows:

```
Device# show voice vrf

==========VOICE VRF CONFIGURATION==========

Global voice vrf defined is: NA
Global sip bind for vrf is: NA
```

show voice vtsp

To display information about the voice port configuration and Voice Telephony Service Provider (VTSP), use the **show voice vtsp** command in privileged EXEC mode.

show voice vtsp {**call** [{**dspstats** | **fsm** | **log** [*call-ID*] | **verbose**}] | **fork dsp-status**} [*call ID*]

Syntax Description

call	Displays the call control block information.
dspstats	(Optional) Displays the selective statistics of digital signal processor (DSP) voice channels.
fsm	(Optional) Displays information about the Finite State Machine Dump (FSM).
log *call-ID*	(Optional) Displays the call related logs. If a call ID is specified, this command displays the status of a specific call. The call ID value range is from 1 to 4294967295
verbose	(Optional) Displays the verbose output.
fork	Displays the media forking information.
dsp-status	Displays the status of media forking in the DSP.
call-ID	(Optional) Displays the status of the call. The value range is from 0x0 to 0xFFFFFFFF. >

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(24)T	This command was introduced in a release earlier than Cisco IOS Release 12.4(24)T.

Usage Guidelines

Use the **show voice vtsp** command to display information about the voice port configuration.

Examples

The following is sample output from the **show voice vtsp** command:

```
Router# show voice vtsp call dspstats 0x833

***DSP VOICE TX STATISTICS***
Tx Vox/Fax Pkts: 1337, Tx Sig Pkts: 0, Tx Comfort Pkts: 181
Tx Dur(ms): 46840, Tx Vox Dur(ms): 26740, Tx Fax Dur(ms): 0
        ***DSP VOICE RX STATISTICS***
Rx Vox/Fax Pkts: 1347, Rx Signal Pkts: 0, Rx Comfort Pkts: 180
Rx Dur(ms): 46840, Rx Vox Dur(ms): 23300, Rx Fax Dur(ms): 0
Rx Non-seq Pkts: 0, Rx Bad Hdr Pkts: 0
Rx Early Pkts: 0, Rx Late Pkts: 0
        ***DSP VOICE VP_DELAY STATISTICS***
Clk Offset(ms): 80, Rx Delay Est(ms): 50
Rx Delay Lo Water Mark(ms): 50, Rx Delay Hi Water Mark(ms): 70
        ***DSP VOICE VP_ERROR STATISTICS***
Predict Conceal(ms): 0, Interpolate Conceal(ms): 0
Silence Conceal(ms): 0, Retroact Mem Update(ms): 0
Buf Overflow Discard(ms): 0, Talkspurt Endpoint Detect Err: 0
```

```
        ***DSP LEVELS***
TDM Bus Levels(dBm0): Rx -68.5 from PBX/Phone, Tx -4.4 to PBX/Phone
TDM ACOM Levels(dBm0): +64.1, TDM ERL Level(dBm0): +10.0
TDM Bgd Levels(dBm0): -80.0, with activity being silence
        ***DSP VOICE ERROR STATISTICS***
Rx Pkt Drops(Invalid Header): 0, Tx Pkt Drops(HPI SAM Overflow): 0
        ***DSP VOICE GSMAMR-NB STATISTICS***
EncodingRate: 7 DecodingRate: 7
numEncodeChanges: 0 numDecodeChanges: 0
numCRCFail: 0 numFrameBadQuality: 0
numInvalidCMR: 0 numInvalidFrameType: 0
```

Related Commands

Command	Description
debug vtsp	Displays the state of the gateway and the call events.

show voip debug version

To display the current version of the Voice over IP debug structure, use the **show voip debug version**command in privileged EXEC mode.

show voip debug version

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.3(8)T	This command was introduced.

Examples

The following example shows output from the **show voip debug version** command:

```
Router# show voip debug version
voip debug version 1.0
```

The table below describes significant fields shown in the display.

Table 5: show voip debug version Field Descriptions

Field	Description
voip debug version 1.0	Shows the version of the debug structure.

Related Commands

Command	Description
show voip rtp connections	Displays RTP named event packets.

show voip fpi call-rate

To display the average call rates at the forwarding plane interface, use the **show voip fpi call-rate** command in privileged EXEC mode.

show voip fpi call-rate interval*seconds* **history** *seconds*

Syntax Description

interval	Displays the message rates at the FPI interface
seconds	The number of seconds for the interval. The range is from 1 to 300
history	Specifies how far back information is kept and displayed.
seconds	The number of seconds that will be displayed. The range is from 1 to 86400.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.9S	This command was introduced.

Usage Guidelines

This command displays the call-rate data that is collected on the forwarding plane interface when the **debug voip fpi call-rate** is enabled.

Example

The following shows the output for the **show voip fpi call-rate** command

```
Router# show voip fpi call-rate interval 1 history 1
------ ------ ------ ------ ------ ------ ------
Sec ADD MOD DEL EVT_UP EVT_DN CPU 5S
------ ------ ------ ------ ------ ------ ------
67 0 0 0 0 0 0
------ ------ ------ ------ ------ ------ ------
```

show voip fpi calls

To display call information for TDM and IVR calls in the Forwarding Plane Interface (FPI), use the **show voip fpi calls** command in privileged EXEC mode.

show voip fpi calls[{**all** | **confID** *identifier* | **callID** *identifier* | **correlator** *identifier*}]

Syntax Description

all	(Optional) Displays the detailed statistics for all calls in the FPI where the collection processes have been enabled.
confID *identifier*	(Optional) Displays detailed call information for a call at the application level.
callID *identifier*	(Optional) Displays detailed call information for a call based on the call ID.
correlator *identifier*	(Optional) Displays detailed call information for a call based on the correlator ID.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.9S	This command was introduced.
Cisco IOS XE Gibraltar 16.12.1a	The command output was enhanced to display the number of SRTP Rollover Counter (ROC) update events received from the data plane on per call basis.

Usage Guidelines

Call ID is a unique identifier for each call leg. Each call has the following two call legs:

- In Leg—The call coming in to CUBE.
- Out Leg—The call going out of CUBE.

Use **callID** *identifier* option to view the call leg details at the application level.

Conf ID is a unique identifier for a call (including both in leg and out leg) at the application level. Use **confID** *identifier* option to know the bridging details between in leg and out leg.

Correlator ID is a unique identifier for a call's (including both in leg and out leg) media session. Use **correlator** *identifier* option to view the media session details of a call.

Example

The following are sample output from the **show voip fpi calls** command

```
Router#show voip fpi calls
Number of Calls : 2
---------- ---------- ---------- ----------- --------------- ---------------
    confID correlator    AcallID     BcallID           state           event
---------- ---------- ---------- ----------- --------------- ---------------
        20         20         87          88       ALLOCATED DETAIL_STAT_RSP
        21         21         89          90       ALLOCATED DETAIL_STAT_RSP
```

```
Router#show voip fpi calls confID 20
---------------------------------------------------------------------------
VoIP-FPI call entry details:
---------------------------------------------------------------------------
Call Type        :         z    IP_IP     confID          :              20
correlator       :               20     call_state       :       ALLOCATED
last_event       :  DETAIL_STAT_RSP     alloc_start_time :      2737426765
modify_start_time:                0     delete_start_time:               0
Media Type(SideA):              RTP     Media Type(SideB):             RTP
---------------------------------------------------------------------------
FPI State Machine Stats:
------------------------
create_req_call_entry_inserted              :         1
call_create_req_fsm_successful              :         1
call_provision_rsp_ok                       :         1
call_provision_rsp_fsm_successful           :         1
event_ind_media_up_to_app                   :         2
------------------------------------------------------------------------------------------
SIDE_A  RTP details   -   gccb=0x7FE69FA11C08
------------------------------------------------------------------------------------------
confID           :        20    fpi_user_data    :        20
callID           :        87    dstCallID        :        88    mainstcallID     :
87
srcport          :     16552    dstport          :     16580    DP add_sent      :
 1
dp_add_fail      :         0    dp_add_pending   :         0    dp_delete_sent   :
 0
dp_delete_waiting:         0    dp_delete_done   :         0    final_stats_pend :
 0
ha_create_sent   :         1    is_video         :         0    media_type       :
 0
is dspfarm xcode :        No    is conference    :        No    stream_type      :     VOICE
rtp_type         :  SENDRECV
------------------------------------------------------------------------------------------
SIDE_B  RTP details   -   gccb=0x7FE6A9B5A960
------------------------------------------------------------------------------------------
confID           :        20    fpi_user_data    :        20
callID           :        88    dstCallID        :        87    mainstcallID     :
88
srcport          :     16554    dstport          :     16400    DP add_sent      :
 1
dp_add_fail      :         0    dp_add_pending   :         0    dp_delete_sent   :
 0
dp_delete_waiting:         0    dp_delete_done   :         0    final_stats_pend :
 0
ha_create_sent   :         1    is_video         :         0    media_type       :
 0
is dspfarm xcode :        No    is conference    :        No    stream_type      :     VOICE
rtp_type         :  SENDRECV
------------------------------------------------------------------------------------------
Detailed Stats from DataPlane:
------------------------------
mgm_handle      : 20
-------------------------------------------------------------------
Call Present in :    FMAN RP   FMAN FP       CPP
                 ----------------------------------------
                         YES       YES       YES
-------------------------------------------------------------------
                      Field              sideA              sideB
-------------------------------------------------------------------
          dtmf_payload_type                  0                  0
```

```
   redundant_data_pyld_type                255                 255
                   tos_mask                  0                   0
                 dtmf_flags                  0                   0
                ucode_flags                  5                   5
                 local_port              16552               16554
             remote_port_tx              16580               16400
             remote_port_rx              16580               16400
                 session_id         0x30000050          0x30000052
  hairpin_prtnr_null(ucode)               NULL                NULL
       hairpin_prtnr_callid                  0                   0
         dsp_interface_null               NULL                NULL
-------------------------------------------------------------------

 DSP Resource Used : No


Router#show voip fpi calls callid 87
---------------------------------------------------------------------------
VoIP-FPI call entry details:
---------------------------------------------------------------------------
Call Type        :            IP_IP     confID           :              20
correlator       :               20     call_state       :       ALLOCATED
last_event       :  DETAIL_STAT_RSP     alloc_start_time :      2737426765
modify_start_time:                0     delete_start_time:               0
Media Type(SideA):              RTP     Media Type(SideB):             RTP
---------------------------------------------------------------------------
FPI State Machine Stats:
------------------------
create_req_call_entry_inserted              :         1
call_create_req_fsm_successful              :         1
call_provision_rsp_ok                       :         1
call_provision_rsp_fsm_successful           :         1
event_ind_media_up_to_app                   :         2
------------------------------------------------------------------------------------------
SIDE_A  RTP details    -    gccb=0x7FE69FA11C08
------------------------------------------------------------------------------------------
confID           :        20    fpi_user_data    :        20
callID           :        87    dstCallID        :        88    mainstcallID     :
87
srcport          :     16552    dstport          :     16580    DP add_sent      :
 1
dp_add_fail      :         0    dp_add_pending   :         0    dp_delete_sent   :
 0
dp_delete_waiting:         0    dp_delete_done   :         0    final_stats_pend :
 0
ha_create_sent   :         1    is_video         :         0    media_type       :
 0
is dspfarm xcode :        No   is conference    :        No    stream_type      :     VOICE
rtp_type         :  SENDRECV
------------------------------------------------------------------------------------------
SIDE_B  RTP details    -    gccb=0x7FE6A9B5A960
------------------------------------------------------------------------------------------
confID           :        20    fpi_user_data    :        20
callID           :        88    dstCallID        :        87    mainstcallID     :
88
srcport          :     16554    dstport          :     16400    DP add_sent      :
 1
dp_add_fail      :         0    dp_add_pending   :         0    dp_delete_sent   :
 0
dp_delete_waiting:         0    dp_delete_done   :         0    final_stats_pend :
 0
ha_create_sent   :         1    is_video         :         0    media_type       :
 0
```

```
is dspfarm xcode :        No    is conference    :        No    stream_type      :     VOICE
rtp_type         :  SENDRECV
------------------------------------------------------------------------------------------
Detailed Stats from DataPlane:
------------------------------
mgm_handle      : 20
-------------------------------------------------------------------
Call Present in :    FMAN RP   FMAN FP       CPP
                 ----------------------------------------
                         YES       YES       YES
-------------------------------------------------------------------
                      Field              sideA              sideB
-------------------------------------------------------------------
          dtmf_payload_type                  0                  0
   redundant_data_pyld_type                255                255
                   tos_mask                  0                  0
                 dtmf_flags                  0                  0
                ucode_flags                  5                  5
                 local_port              16552              16554
             remote_port_tx              16580              16400
             remote_port_rx              16580              16400
                 session_id         0x30000050         0x30000052
  hairpin_prtnr_null(ucode)               NULL               NULL
       hairpin_prtnr_callid                  0                  0
         dsp_interface_null               NULL               NULL
-------------------------------------------------------------------

 DSP Resource Used : No



Router#show voip fpi calls all

Number of Calls : 2
---------------------------------------------------------------------------
VoIP-FPI call entry details:
---------------------------------------------------------------------------
Call Type        :            IP_IP     confID           :              24
correlator       :               24     call_state       :       ALLOCATED
last_event       :  DETAIL_STAT_RSP     alloc_start_time :      2902404766
modify_start_time:                0     delete_start_time:               0
Media Type(SideA):              RTP     Media Type(SideB):             RTP
---------------------------------------------------------------------------
FPI State Machine Stats:
------------------------
create_req_call_entry_inserted             :         1
call_create_req_fsm_successful             :         1
call_provision_rsp_ok                      :         1
call_provision_rsp_fsm_successful          :         1
event_ind_media_up_to_app                  :         2
------------------------------------------------------------------------------------------
SIDE_A  RTP details   -   gccb=0x7FE69FA11C08
------------------------------------------------------------------------------------------
confID           :        24    fpi_user_data    :        24
callID           :        95    dstCallID        :        96    mainstcallID     :
95
srcport          :     16568    dstport          :     16580    DP add_sent      :
 1
dp_add_fail      :         0    dp_add_pending   :         0    dp_delete_sent   :
 0
dp_delete_waiting:         0    dp_delete_done   :         0    final_stats_pend :
 0
ha_create_sent   :         1    is_video         :         0    media_type       :
 0
```

```
is dspfarm xcode :        No    is conference    :        No    stream_type      :     VOICE
rtp_type         :  SENDRECV
------------------------------------------------------------------------------------------
SIDE_B  RTP details   -   gccb=0x7FE6A9B5A960
------------------------------------------------------------------------------------------
confID           :        24    fpi_user_data    :        24
callID           :        96    dstCallID        :        95    mainstcallID     :
96
srcport          :     16570    dstport          :     16400    DP add_sent      :
 1
dp_add_fail      :         0    dp_add_pending   :         0    dp_delete_sent   :
 0
dp_delete_waiting:         0    dp_delete_done   :         0    final_stats_pend :
 0
ha_create_sent   :         1    is_video         :         0    media_type       :
 0
is dspfarm xcode :        No    is conference    :        No    stream_type      :     VOICE
rtp_type         :  SENDRECV
------------------------------------------------------------------------------------------
Detailed Stats from DataPlane:
------------------------------
mgm_handle      : 24
-------------------------------------------------------------------
Call Present in :    FMAN RP   FMAN FP       CPP
                 ----------------------------------------
                         YES       YES       YES
-------------------------------------------------------------------
                      Field               sideA               sideB
-------------------------------------------------------------------
          dtmf_payload_type                   0                   0
   redundant_data_pyld_type                 255                 255
                   tos_mask                   0                   0
                 dtmf_flags                   0                   0
                ucode_flags                   5                   5
                 local_port               16568               16570
             remote_port_tx               16580               16400
             remote_port_rx               16580               16400
                 session_id          0x30000060          0x30000062
  hairpin_prtnr_null(ucode)                NULL                NULL
       hairpin_prtnr_callid                   0                   0
         dsp_interface_null                NULL                NULL
-------------------------------------------------------------------

 DSP Resource Used : No
------------------------------------------------------------
---------------------------------------------------------------------------
VoIP-FPI call entry details:
---------------------------------------------------------------------------
Call Type        :           IP_IP     confID           :              25
correlator       :              25     call_state       :       ALLOCATED
last_event       :  DETAIL_STAT_RSP    alloc_start_time :      2902505765
modify_start_time:               0     delete_start_time:               0
Media Type(SideA):             RTP     Media Type(SideB):             RTP
---------------------------------------------------------------------------
FPI State Machine Stats:
------------------------
create_req_call_entry_inserted             :        1
call_create_req_fsm_successful             :        1
call_provision_rsp_ok                      :        1
call_provision_rsp_fsm_successful          :        1
event_ind_media_up_to_app                  :        1
------------------------------------------------------------------------------------------
SIDE_A  RTP details   -   gccb=0x7FE6A9B9CFA8
------------------------------------------------------------------------------------------
```

```
confID           :        25    fpi_user_data    :        25
callID           :        97    dstCallID        :        98    mainstcallID     :
97
srcport          :     16572    dstport          :     16584    DP add_sent      :
 1
dp_add_fail      :         0    dp_add_pending   :         0    dp_delete_sent   :
 0
dp_delete_waiting:         0    dp_delete_done   :         0    final_stats_pend :
 0
ha_create_sent   :         0    is_video         :         0    media_type       :
 0
is dspfarm xcode :        No    is conference    :        No    stream_type      :     VOICE
rtp_type         :  SENDRECV
-------------------------------------------------------------------------------------------
SIDE_B  RTP details   -   gccb=0x7FE69FA132F8
-------------------------------------------------------------------------------------------
confID           :        25    fpi_user_data    :        25
callID           :        98    dstCallID        :        97    mainstcallID     :
98
srcport          :     16574    dstport          :     16404    DP add_sent      :
 1
dp_add_fail      :         0    dp_add_pending   :         0    dp_delete_sent   :
 0
dp_delete_waiting:         0    dp_delete_done   :         0    final_stats_pend :
 0
ha_create_sent   :         1    is_video         :         0    media_type       :
 0
is dspfarm xcode :        No    is conference    :        No    stream_type      :     VOICE
rtp_type         :  SENDRECV
-------------------------------------------------------------------------------------------
Detailed Stats from DataPlane:
------------------------------
mgm_handle      : 25
-------------------------------------------------------------------
Call Present in :    FMAN RP   FMAN FP       CPP
                 ----------------------------------------
                         YES       YES       YES
-------------------------------------------------------------------
                      Field              sideA              sideB
-------------------------------------------------------------------
          dtmf_payload_type                  0                  0
   redundant_data_pyld_type                255                255
                   tos_mask                  0                  0
                 dtmf_flags                  0                  0
                ucode_flags                  0                  5
                 local_port              16572              16574
             remote_port_tx              16584              16404
             remote_port_rx              16584              16404
                 session_id         0x30000064         0x30000066
  hairpin_prtnr_null(ucode)               NULL               NULL
       hairpin_prtnr_callid                  0                  0
         dsp_interface_null               NULL               NULL
-------------------------------------------------------------------

 DSP Resource Used : No
```

Determine information for a call based on correlator ID

To know the correlator ID, run `show call active voice compact` command and determine the active calls and its associated callID from the output. Make a note of the desired callID and enter the same while executing `show voip fpi calls callID xx` command. The command output displays the correlator ID associated with the desired call ID. Enter the correlator ID while executing `show voip fpi calls correlator ID` command to know the number of Rollover Counter (ROC) updates that have come from the data plane to the control plane for a specific call.

The following is a sample output of `show call active voice compact` command.

```
Router#show call active voice compact
<callID>  A/O FAX T<sec> Codec       type        Peer Address       IP R<ip>:<udp>
Total call-legs: 4
       212 ANS     T735   pass-throug VOIP        P5553001        1.2.111.4:8376   <<< CUBE
 leg1, the remote peer is 1.2.111.4:8376(5553001)
       213 ORG     T735   pass-throug VOIP        P5553101        1.2.111.6:18898  <<< CUBE
 leg2, the remote peer is 1.2.111.6:18898(5553101)
       214 ORG     T735   g711ulaw    VOIP        P      1.2.111.108:16904         <<<
software MTP leg1, the remote peer is 1.2.111.108:16904
       215 ORG     T735   g711ulaw    VOIP        P        1.2.111.4:8372          <<<
software MTP leg2, the remote peer is 1.2.111.4:8372
```

Select callID 214 from the output and enter it in the `show voip fpi calls callID xx` command as shown below:

```
Router#show voip fpi calls callID 214 | include correlator
correlator       :              102     call_state       :        ALLOCATED
```

Enter correlator ID as 102 while executing `show voip fpi calls correlator ID` command as shown below to know the number of Rollover Counter (ROC) updates that have come from the data plane to the control plane for a specific call.

```
Router#show voip fpi calls correlator 102 | inc event

last_event       :    GET_STATS_RSP     alloc_start_time :       1024243582

event_ind_srtp_roc_upd_to_app               :         4
```

show voip fpi rtts

To display maximum, minimum, average and histogram for round trip times for create, modify and delete requests from control plane to forwarding plane, use **show voip fpi rtts** command in privileged EXEC mode.

show voip fpi rtts

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.9S	This command was introduced.

Usage Guidelines

Use **show voip fpi rtts** command in privileged EXEC mode to display maximum, minimum, average and histogram for round trip times for create, modify, and delete requests from control plane to forwarding plane.

Example

```
Router#show voip fpi rtts
---------- ---------- ---------- ---------- ------------
   command      count  avg(msec)  max(msec) over_thrshld
---------- ---------- ---------- ---------- ------------
     ALLOC          1         38         38            0
    MODIFY          0          0          0            0
    DELETE          1          8          8            0
---------- ---------- ---------- ---------- ------------

HISTOGRAM
---------- ---------- ---------- ----------
     msecs      ALLOC     MODIFY     DELETE
---------- ---------- ---------- ----------
<= 10               0          0          1
<= 40               1          0          0
```

show voip fpi stats

To display the TDM and IVR statistics and error counters in the Forwarding Plane Interface (FPI), use the **show voip fpi stats** command in privileged EXEC mode.

show voip fpi stats [**fsm**]

Syntax Description

fsm	(Optional) Displays the finite state machine (FSM) events.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.9S	This command was introduced.
Cisco IOS XE Gibraltar 16.12.1a	The command output was enhanced to display the number of SRTP Rollover Counter (ROC) update events received from the data plane.

The following is a sample output from the **show voip fpi stats** command:

```
Router#show voip fpi stats
**************** VOIP FPI STATS ********************
----------- ---------- ---------- ---------- ----------
type        ReqSuccess    ReqFail RspSuccess    RspFail
----------- ---------- ---------- ---------- ----------
caps                 1          0          1          0
init                 1          0          1          0
params               1          0        N/A        N/A
config               0          0          0          0          0(skip)
deact                0          0          0          0     0(wrong state)
port add             2          0        N/A        N/A
port delete          0          0        N/A        N/A
*********************** ACTIVE *******************************
                          IDLE      ALLOCATING       ALLOCATED       MODIFYING
     CREATE_REQ              1               0               0               0
     MODIFY_REQ              0               0               1               0
     DELETE_REQ              0               0               0               0
  GET_STATS_REQ              0               0               0               0
    PROV_RSP_OK              0               1               0               1
  PROV_RSP_FAIL              0               0               0               0
     DELETE_RSP              0               0               0               0
  GET_STATS_RSP              0               0               0               0
  STATS_TMR_EXP              0               0               0               0
     TMR_EXPIRY              0               0               0               0
CREATE_STRM_REQ              0               0               0               0
MODIFY_STRM_REQ              0               0               0               0
DELETE_STRM_REQ              0               0               0               0
DETAIL_STAT_REQ              0               0               0               0
DETAIL_STAT_RSP              0               0               0               0
DT_STAT_TMR_EXP              0               0               0               0
                      DELETING  ALLOC_MOD_PEND MODIFY_MOD_PEND  DELETE_PENDING
     CREATE_REQ              0               0               0               0
     MODIFY_REQ              0               0               0               0
     DELETE_REQ              0               0               0               0
  GET_STATS_REQ              0               0               0               0
    PROV_RSP_OK              0               0               0               0
  PROV_RSP_FAIL              0               0               0               0
```

```
     DELETE_RSP             0               0               0               0
  GET_STATS_RSP             0               0               0               0
  STATS_TMR_EXP             0               0               0               0
     TMR_EXPIRY             0               0               0               0
CREATE_STRM_REQ             0               0               0               0
MODIFY_STRM_REQ             0               0               0               0
DELETE_STRM_REQ             0               0               0               0
DETAIL_STAT_REQ             0               0               0               0
DETAIL_STAT_RSP             0               0               0               0
DT_STAT_TMR_EXP             0               0               0               0
********************** END ACTIVE ****************************


*********************** STANDBY ******************************
                          IDLE      ALLOCATING       ALLOCATED       MODIFYING
     CREATE_REQ             0               0               0               0
     MODIFY_REQ             0               0               0               0
     DELETE_REQ             0               0               0               0
  GET_STATS_REQ             0               0               0               0
    PROV_RSP_OK             0               0               0               0
  PROV_RSP_FAIL             0               0               0               0
     DELETE_RSP             0               0               0               0
  GET_STATS_RSP             0               0               0               0
  STATS_TMR_EXP             0               0               0               0
     TMR_EXPIRY             0               0               0               0
CREATE_STRM_REQ             0               0               0               0
MODIFY_STRM_REQ             0               0               0               0
DELETE_STRM_REQ             0               0               0               0
DETAIL_STAT_REQ             0               0               0               0
DETAIL_STAT_RSP             0               0               0               0
DT_STAT_TMR_EXP             0               0               0               0
                      DELETING  ALLOC_MOD_PEND MODIFY_MOD_PEND  DELETE_PENDING
     CREATE_REQ             0               0               0               0
     MODIFY_REQ             0               0               0               0
     DELETE_REQ             0               0               0               0
  GET_STATS_REQ             0               0               0               0
    PROV_RSP_OK             0               0               0               0
  PROV_RSP_FAIL             0               0               0               0
     DELETE_RSP             0               0               0               0
  GET_STATS_RSP             0               0               0               0
  STATS_TMR_EXP             0               0               0               0
     TMR_EXPIRY             0               0               0               0
CREATE_STRM_REQ             0               0               0               0
MODIFY_STRM_REQ             0               0               0               0
DELETE_STRM_REQ             0               0               0               0
DETAIL_STAT_REQ             0               0               0               0
DETAIL_STAT_RSP             0               0               0               0
DT_STAT_TMR_EXP             0               0               0               0
********************** END STANDBY ***************************


Correlators in use:1

Corrupted table error (alloc):0

Corrupted table error (delete):0
----------- ---------- ---------- ---------- ----------
            gccb/rtpNL pr gccb NL no gccb sd badConfIds
----------- ---------- ---------- ---------- ----------
call create          0          0          0          0
            add sent T entry Fail entry insr fsm Succss
----------- ---------- ---------- ---------- ----------
                     0          0          1          1
```

```
            fsm failed ent delete       fail
----------- ---------- ---------- ---------- ----------
                     0          0          0

            entry !pre fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
call modify          0          0          1
            entry !pre entry del  fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
call delete          0          0          0          0

---------------- ---------- ---------- ---------- ----------
                 gccb/rtpNL pr gccb NL no gccb sd badConfIds
---------------- ---------- ---------- ---------- ----------
LPBK call create          0          0          0          0
                 add sent T entry Fail entry insr fsm Succss
---------------- ---------- ---------- ---------- ----------
                          0          0          0          0
                 fsm failed ent delete       fail
---------------- ---------- ---------- ---------- ----------
                          0          0          0

                 entry !pre fsm failed fsm Succss
---------------- ---------- ---------- ---------- ----------
LPBK call modify          0          0          0
                 entry !pre entry del  fsm failed fsm Succss
---------------- ---------- ---------- ---------- ----------
LPBK call delete          0          0          0          0
---------------- ---------- ---------- ---------- ----------

---------------- ---------- ---------- ---------- ----------
                 gccb/rtpNL pr gccb NL no gccb sd badConfIds
---------------- ---------- ---------- ---------- ----------
STRM call create          0          0          0          0
                 add sent T entry Fail entry insr fsm Succss
---------------- ---------- ---------- ---------- ----------
                          0          0          0          0
                 fsm failed ent delete       fail
---------------- ---------- ---------- ---------- ----------
                          0          0          0

                 entry !pre fsm failed fsm Succss
---------------- ---------- ---------- ---------- ----------
STRM call modify          0          0          0
                 entry !pre entry del  fsm failed fsm Succss
---------------- ---------- ---------- ---------- ----------
STRM call delete          0          0          0          0
---------------- ---------- ---------- ---------- ----------

             gccb !fnd entry !pre fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
call  stats          0          0          0          0
            fsm failed fsm Succss  entry del
----------- ---------- ---------- ---------- ----------
call  timer          0          0          0
            fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
stats timer          0          0
            entry !pre     rsp ok rsp failed
----------- ---------- ---------- ---------- ----------
provisn rsp          0          2          0
            fsm Succss fsm failed entry deld
----------- ---------- ---------- ---------- ----------
                     2          0          0
```

```
            entry !pre     rsp ok rsp failed fsm Succes
----------- ---------- ---------- ---------- ----------
 delete rsp          0          0          0          0
            fsm failed entry deld corr mismt inval gccb
----------- ---------- ---------- ---------- ----------
                     0          0          0          0
type        entry !pre     rsp ok rsp failed InvGCCB
----------- ---------- ---------- ---------- ----------
 stats  rsp          0          0          0          0
type        fsm Succss fsm failed corr mismt
----------- ---------- ---------- ---------- ----------
                     0          0          0
type       entry !pre mda DN App mda UP App srtp ROC upd lpbk mda DN lpbk mda UP Cor !match
 InvGCCB
----------- ---------- ---------- ---------- ------------ ----------- ----------- ----------
 ----------
 media evnt          0          0          2            1           0           0
 0          0

HA Stats

TDM-TDM Stats
            add sent T entry Fail entry insr fsm Succss
----------- ---------- ---------- ---------- ----------
tdm create           0          0          0          0
            fsm failed ent delete       fail
----------- ---------- ---------- ---------- ----------
                     0          0          0

            entry !pre fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
tdm modify         0          0          0
            entry !pre entry del  fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
tdm delete           0          0          0          0
            fsm failed fsm Succss  entry del
----------- ---------- ---------- ---------- ----------
 tdm  timer          0          0          0

            entry !pre     rsp ok rsp failed
----------- ---------- ---------- ---------- ----------
tdm prv rsp          0          0          0
            fsm Succss fsm failed entry deld
----------- ---------- ---------- ---------- ----------
                     0          0          0
            entry !pre     rsp ok rsp failed fsm Succes
----------- ---------- ---------- ---------- ----------
tdm del rsp          0          0          0          0
            fsm failed entry deld
----------- ---------- ----------
                     0          0

Single/Conferee Leg Stats
--------------- ---------- ---------- ---------- ----------
                gccb/rtpNL pr gccb NL no gccb sd badConfIds
--------------- ---------- ---------- ---------- ----------
singl/conf add           0          0          0          0
                add sent   entry Fail entry insr fsm Succss
--------------- ---------- ---------- ---------- ----------
singl/conf add         0          0          0          0
                fsm failed ent delete req_fail
--------------- ---------- ---------- ---------- ----------
singl/conf add         0          0          0
```

```
               entry !pre fsm failed fsm Succss
-------------- ---------- ---------- ---------- ----------
singl/conf mod          0          0          0
               entry !pre entry del  fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
singl/conf del          0          0          0          0

********************** ACTIVE ******************************
               FORK_SESS_IDLE FORK_SESS_ALLOCATING FORK_SESS_ALLOCATED FORK_SESS_MODIFYING
FORK_SESS_CREATE_REQ           0               0               0               0
FORK_SESS_MODIFY_REQ           0               0               0               0
FORK_SESS_DELETE_REQ           0               0               0               0
FORK_SESS_GET_STATS_REQ           0               0               0               0
FORK_SESS_PROV_RSP_OK           0               0               0               0
FORK_SESS_PROV_RSP_FAIL           0               0               0               0
FORK_SESS_DELETE_RSP           0               0               0               0
FORK_SESS_GET_STATS_RSP           0               0               0               0
FORK_SESS_STATS_TMR_EXP           0               0               0               0
FORK_SESS_TMR_EXPIRY           0               0               0               0
               FORK_SESS_DELETING FORK_SESS_DELETE_PENDING
FORK_SESS_CREATE_REQ           0               0
FORK_SESS_MODIFY_REQ           0               0
FORK_SESS_DELETE_REQ           0               0
FORK_SESS_GET_STATS_REQ           0               0
FORK_SESS_PROV_RSP_OK           0               0
FORK_SESS_PROV_RSP_FAIL           0               0
FORK_SESS_DELETE_RSP           0               0
FORK_SESS_GET_STATS_RSP           0               0
FORK_SESS_STATS_TMR_EXP           0               0
FORK_SESS_TMR_EXPIRY           0               0
********************* END ACTIVE ***************************


********************** STANDBY *****************************
               FORK_SESS_IDLE FORK_SESS_ALLOCATING FORK_SESS_ALLOCATED FORK_SESS_MODIFYING
FORK_SESS_CREATE_REQ           0               0               0               0
FORK_SESS_MODIFY_REQ           0               0               0               0
FORK_SESS_DELETE_REQ           0               0               0               0
FORK_SESS_GET_STATS_REQ           0               0               0               0
FORK_SESS_PROV_RSP_OK           0               0               0               0
FORK_SESS_PROV_RSP_FAIL           0               0               0               0
FORK_SESS_DELETE_RSP           0               0               0               0
FORK_SESS_GET_STATS_RSP           0               0               0               0
FORK_SESS_STATS_TMR_EXP           0               0               0               0
FORK_SESS_TMR_EXPIRY           0               0               0               0
               FORK_SESS_DELETING FORK_SESS_DELETE_PENDING
FORK_SESS_CREATE_REQ           0               0
FORK_SESS_MODIFY_REQ           0               0
FORK_SESS_DELETE_REQ           0               0
FORK_SESS_GET_STATS_REQ           0               0
FORK_SESS_PROV_RSP_OK           0               0
FORK_SESS_PROV_RSP_FAIL           0               0
FORK_SESS_DELETE_RSP           0               0
FORK_SESS_GET_STATS_RSP           0               0
FORK_SESS_STATS_TMR_EXP           0               0
FORK_SESS_TMR_EXPIRY           0               0
********************* END STANDBY **************************


Correlators in use:0

Corrupted table error (alloc):0
```

```
Corrupted table error (delete):0
----------- ---------- ---------- ---------- ----------
            gccb/rtpNL pr gccb NL no gccb sd badConfIds
----------- ---------- ---------- ---------- ----------
fork_sess create          0          0          0          0
            add sent T entry Fail entry insr fsm Succss
----------- ---------- ---------- ---------- ----------
                     0          0          0          0
            fsm failed ent delete       fail
----------- ---------- ---------- ---------- ----------
                     0          0          0

            entry !pre fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
Fork session modify          0          0          0
            entry !pre entry del  fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
fork_sess delete          0          0          0          0

---------------- ---------- ---------- ---------- ----------

             gccb !fnd entry !pre fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
fork_sess  stats          0          0          0          0
            fsm failed fsm Succss  entry del
----------- ---------- ---------- ---------- ----------
fork_sess  timer          0          0          0
            fsm failed fsm Succss
----------- ---------- ---------- ---------- ----------
stats timer          0          0
            entry !pre     rsp ok rsp failed
----------- ---------- ---------- ---------- ----------
provisn rsp          0          0          0
            fsm Succss fsm failed entry deld
----------- ---------- ---------- ---------- ----------
                     0          0          0
            entry !pre     rsp ok rsp failed fsm Succes
----------- ---------- ---------- ---------- ----------
 delete rsp          0          0          0          0
            fsm failed entry deld corr mismt inval gccb
----------- ---------- ---------- ---------- ----------
                     0          0          0          0
type        entry !pre     rsp ok rsp failed InvGCCB
----------- ---------- ---------- ---------- ----------
----------- ---------- ---------- ---------- ----------
 stats  rsp          0          0          0          0
type        fsm Succss fsm failed corr mismt
----------- ---------- ---------- ---------- ----------
                     0          0          0

media event rate:60 per 100msec, media timeout:50 secs
```

Cisco IOS XE Gibraltar 16.12.1a added information about SRTP Rollover Counter (ROC) in the command output. In the above sample output, **srtp ROC upd** represents the total number of ROC updates received from data plane to control plane.

show voip htsp

To display the voip and hybrid transport switching protocol (HTSP) connections active in the router, use the **show voip htsp**command in privileged EXEC mode.

show voip htsp info [**controller** [**T1** *slot-number*]]

Syntax Description

info	Displays htsp related information.
controller	(Optional) Displays information about controllers such as DS3,T1,and E1.
T1	(Optional) Displays information about T1 controller.
slot-number	(Optional) controller slot number.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.

Usage Guidelines

Use the**show voip htsp command to**display the voip and hybrid transport switching protocol (HTSP) connections active in the router.

Examples

The following is sample output from the **show voip htsp** command:

```
Router# show voip htsp
NOTE: '-' means Not Applicable for that signalling type
  SLOT/                                              TSP       TDM      TDM
 PORT/                                  TSP         BEAR     CONNECT   CROSS
 CHANNEL    HTSPINFO    VTSP_CDB        CDB         CHAN_T     DONE    CONNECT
=========  ==========  ==========  ==========  ==========  =======  =======
 02/00/01  0x677371E8  0x68905A48  0x67757AA4  0x677371E8     y        y
 02/00/02  0x67737780  0x00000000  0x00000000  0x00000000     n        n
 02/00/03  0x67737D18  0x68906548  0x67757584  0x67737D18     y        y
 02/00/04  0x677382B0  0x68904C88  0x677572F4  0x677382B0     y        y
 02/00/05  0x67738848  0x00000000  0x00000000  0x00000000     n        n
 02/00/06  0x67738DE0  0x00000000  0x00000000  0x00000000     n        n
 02/00/07  0x67739378  0x689054C8  0x67756B44  0x67739378     y        y
 02/00/08  0x67739910  0x68907888  0x677568B4  0x67739910     y        y
 02/00/09  0x67739EA8  0x00000000  0x00000000  0x00000000     n        n
 02/00/10  0x6773A440  0x00000000  0x00000000  0x00000000     n        n
 02/00/11  0x6773A9D8  0x68906D88  0x67756104  0x6773A9D8     y        y
 02/00/12  0x6773AF70  0x68908388  0x67755E74  0x6773AF70     y        y
 02/00/13  0x6773B508  0x00000000  0x00000000  0x00000000     n        n
 02/00/14  0x6773BAA0  0x00000000  0x00000000  0x00000000     n        n
 02/00/15  0x6773C038  0x689096C8  0x677556C4  0x6773C038     y        y
 02/00/17  0x6773C5D0  0x68909148  0x67755434  0x6773C5D0     y        y
 02/00/18  0x6773CB68  0x00000000  0x00000000  0x00000000     n        n
 02/00/19  0x6773D100  0x00000000  0x00000000  0x00000000     n        n
 02/00/20  0x6773D698  0x68905788  0x67754C84  0x6773D698     y        y
 02/00/21  0x6773DC30  0x68905D08  0x677549F4  0x6773DC30     y        y
```

```
02/00/22  0x6773E1C8  0x00000000  0x00000000  0x00000000     n        n
02/00/23  0x6773E760  0x00000000  0x00000000  0x00000000     n        n
02/00/24  0x6773ECF8  0x68906AC8  0x67754244  0x6773ECF8     y        y
02/00/25  0x6773F290  0x68907308  0x67753FB4  0x6773F290     y        y
02/00/26  0x6773F828  0x00000000  0x00000000  0x00000000     n        n
02/00/27  0x6773FDC0  0x00000000  0x00000000  0x00000000     n        n
02/00/28  0x67740358  0x689080C8  0x67753804  0x67740358     y        y
02/00/29  0x677408F0  0x68908908  0x67753574  0x677408F0     y        y
02/00/30  0x67740E88  0x00000000  0x00000000  0x00000000     n        n
02/00/31  0x67741420  0x68909408  0x67753054  0x67741420     y        y
02/02/01  0x67B88824  0x00000000  0x00000000      -          -        n
02/02/02  0x67B88DBC  0x00000000  0x00000000      -          -        n
02/02/03  0x67B89354  0x00000000  0x00000000      -          -        n
02/02/04  0x67B898EC  0x00000000  0x00000000      -          -        n
02/02/05  0x67B89E84  0x00000000  0x00000000      -          -        n
02/02/06  0x67B8A41C  0x00000000  0x00000000      -          -        n
02/02/07  0x67B8A9B4  0x00000000  0x00000000      -          -        n
02/02/08  0x67B8AF4C  0x00000000  0x00000000      -          -        n
02/02/09  0x67B8B4E4  0x00000000  0x00000000      -          -        n
```

Related Commands

Command	Description
debug voip vtsp	Displays information about the voice telephony service provider (VTSP).

show voip recmsp session

To display active recording Media Service Provider (MSP) session information, use the **show voip recmsp session** command in privileged EXEC mode.

show voip recmsp session [**detail call-id** *callid*]

Syntax Description

detail	(Optional) Displays detailed active session information.
call-id *callid*	(Optional) Specifies the recording MSP call ID. The range is from 0 to 65535.

Command Default

Displays brief information about recorded calls that have the anchor call ID, forked call ID, and MSP call ID.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.2(1)T	This command was introduced.

Usage Guidelines

Use the **show voip recmsp session** command to display MSP-related information about the recorder, for example, the way the recording MSP views the recording session.

The **show voip recmsp session detail call-id** *callid* command provides detailed information about each recording session. It provides details about the anchor leg and nonanchor leg. It also shows how the anchor and nonanchor streams are mapped to the forked leg Real-Time Transport Protocol (RTP) streams.

Examples

The following is sample output from the **show voip recmsp session detail call-id** command. Fields in the display are self-explanatory.

```
Router# show voip recmsp session detail call-id
 140
RECMSP active sessions:
Detailed Information
=========================
Recording MSP Leg Details:
Call ID: 143
GUID : 7C5946D38ECD
AnchorLeg Details:
Call ID: 141
Forking Stream type: voice-nearend
Participant: 708090
Non-anchor Leg Details:
Call ID: 140
Forking Stream type: voice-farend
Participant: 10000
Forked Leg Details:
Call ID: 145
Near End Stream CallID 145
Stream State ACTIVE
Far End stream CallID 146
```

```
Stream State ACTIVE
Found 1 active sessions
```

Related Commands

Command	Description
media-recording	Configures voice class recording parameters.

show voip rtp connections

To display Real-Time Transport Protocol (RTP) named event packets, use the **show voip rtp connections** command in privileged EXEC mode.

show voip rtp connections [**detail**]

Syntax Description

detail	(Optional) Displays the called-party and calling-party numbers associated with a call.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0	This command was introduced.
12.3(7)T	The **detail** keyword was added.
12.3(14)T	This command was implemented on the Cisco 2800 series and Cisco 3800 series.
12.4(2)T	This command was integrated into Cisco IOS Release 12.4(2)T.
12.4(22)T	Command output was updated to show IPv6 information.
Cisco IOS 15.6(2)T	The command output was enhanced to show VRF information.
Cisco IOS XE Gibraltar 16.12.1a	The command output was enhanced to show SRTP Rollover Counter (ROC) information.
Cisco IOS XE Gibraltar 16.12.2	The command output was updated to show SRTP Rollover Count (ROC) information based on Remote IP address and Port.

Usage Guidelines

This command displays information about RTP named event packets, such as caller ID number, IP address, and port for both the local and remote endpoints. The output from this command provides an overview of all the connections in the system, and this information can be used to narrow the criteria for debugging. The **debug voip rtp** command floods the console with voice packet information. You can use the **show voip rtp connections** command to get caller ID, remote IP address, or remote port identifiers that you can use to limit the output from the **debug voip rtp** command.

The **detail** keyword allows you to identify the phone or phones that have connected two RTP call legs to create VoIP-to-VoIP or VoIP-to-POTS hairpins. If the **detail** keyword is omitted, the output does not display calls that are connected by hairpin call routing.

Examples

The table below describes the significant fields shown in the examples. Each line of output under "VoIP RTP active connections" shows information for one call leg. A phone call normally consists of two call legs, one connected to the calling party and one connected to the called party. The router joins (or bridges) the two call legs to make a call. The **show voip rtp connections** command shows the RTP information for H.323 and Session Initiation Protocol (SIP) calls only; it does not directly show the POTS call legs. The information for the IP phone can be seen using the **show ephone offhook** command.

The following sample output shows an incoming H.323 call that is being directed to an IP phone attached to a Cisco Unified Communications Manager Express (Unified CME) system.

```
Router# show voip rtp connections
VoIP RTP active connections :
No. CallId  dstCallId  LocalRTP RmtRTP LocalIP          RemoteIP
1   21      22         16996    18174  10.4.204.37      10.4.204.24
Found 1 active RTP connections
```

The following sample output shows the same call as in the previous example, but using the **detail** keyword with the command. The sample output shows the called number (1509) and calling number (8108) on both call legs (21 and 22); the called and calling numbers are the same on both legs for a simple A-to-B call. Leg 21 is the H.323 segment of the and leg 22 is the POTS segment that goes to the IP phone.

```
Router#
show voip rtp connections detail
VoIP RTP active connections :
No. CallId  dstCallId  LocalRTP RmtRTP LocalIP          RemoteIP
1   21      22         16996    18174  10.4.204.37      10.4.204.24
  callId 21 (dir=1):called=1509 calling=8108 redirect=
     dest callId 22:called=1509 calling=8108 redirect=
   1 context 64FB3358 xmitFunc 6032E8B4
Found 1 active RTP connections
```

The following example shows the call from the previous example being transferred by extension 1509 to extension 1514. Notice that the dstCallId changed from 22 to 24, but the original call leg (21) for the transferred party is still present. This implies that H.450.2 capability was disabled for this particular call, because if H.450.2 was being used for the transfer, the transfer would have caused the incoming H.323 call leg to be replaced with a new call.

```
Router# show voip rtp connections

VoIP RTP active connections :
No. CallId  dstCallId  LocalRTP RmtRTP LocalIP         RemoteIP
1   21      24         16996    18174  10.4.204.37      10.4.204.24
Found 1 active RTP connections
```

The following example shows the detailed output for the same transfer as shown in the previous example. The original incoming call leg is still present (21) and still has the original called and calling numbers. The transferred call leg (24) shows 1509 (the transferring party) as the calling party and 1514 (the transfer destination) as the called party.

```
Router# show voip rtp connections detail
VoIP RTP active connections :
No. CallId  dstCallId  LocalRTP RmtRTP LocalIP         RemoteIP
1   21      24         16996    18174  10.4.204.37      10.4.204.24
  callId 21 (dir=1):called=1509 calling=8108 redirect=
     dest callId 24:called=1514 calling=1509 redirect=
   1 context 6466E810 xmitFunc 6032E8B4
Found 1 active RTP connections
```

The following sample output shows a cross-linked call with two H.323 call legs. The first line of output shows that the CallID for the first call leg is 7 and that this call leg is associated with another call leg that has a destination CallId of 8. The next line shows that the CallID for the leg is 8 and that it is associated with another call leg that has a destination CallId of 7. This cross-linkage between CallIds 7 and 8 shows that the first call leg is related to the second call leg (and vice versa). From this you can infer that the two call legs are actually part of the same phone call.

In an active system you can expect many lines of output that you would have to sort through to see which ones have this cross-linkage relationship. The lines showing two related call legs are not necessarily listed in adjacent order.

```
Router# show voip rtp connections
VoIP RTP active connections :
No. CallId   dstCallId          LocalRTP     RmtRTP         LocalIP        RemoteIP
1        7           8              16586      22346      172.27.82.2      172.29.82.2
2        8           7              17010      16590      172.27.82.2      192.168.1.29

Found 2 active RTP connections
```

The following example shows RTP information with IPv6 local and remote addresses:

```
Router# show voip rtp connections
VoIP RTP active connections :
No.  CallId   dstCallId  LocalRTP  RmtRTP   LocalIP                            RemoteIP
1     11      9          17424     18282    2001:DB8:C18:1:218:FEFF:FE71:2AB6
2001:DB8:C18:1:218:FEFF:FE71:2AB6
2     12      10         18282     17424    2001:DB8:C18:1:218:FEFF:FE71:2AB6
2001:DB8:C18:1:218:FEFF:FE71:2AB6
Found 2 active RTP connections
```

The following example shows RTP information with VRF details:

```
Router# show voip rtp connections
VoIP RTP Port Usage Information:
Max Ports Available: 23001, Ports Reserved: 101, Ports in Use: 2
Min Max Ports Ports Ports
Media-Address Range Port Port Available Reserved In-use
------------------------------------------------------------------------------
Global Media Pool 8000 48198 19999 101 2
------------------------------------------------------------------------------
VoIP RTP active connections :
No.    CallId  dstCallId    LocalRTP    RmtRTP     LocalIP     RemoteIP   MPSS    VRF
1      1        2             25000     16390    10.0.0.1     10.0.0.2     NO     VRF1
2      2        1             25002     16398    11.0.0.1     11.0.0.2     NO     VRF2
```

SRTP Rollover Counter (ROC) information is displayed in the “SSRC:ROC” format and is updated based on Remote IP address and Port.

The following example shows SRTP ROC information for an SRTP call:

```
Router#show voip rtp connections detail
VoIP RTP Port Usage Information:
Max Ports Available: 59794, Ports Reserved: 206, Ports in Use: 2
Port range configured
                                        Min   Max   Ports     Ports     Ports
Media-Address Range                     Port  Port  Available Reserved  In-use
------------------------------------------------------------------------------
Global Media Pool                       5500  65498 29897     103       2

IP Address Based Media Pool
------------------------------------------------------------------------------
8.39.15.21          8.39.15.21          5500  65498 29897     103       0
        Port-Range                      10000 20000
------------------------------------------------------------------------------
```

```
VoIP RTP active connections :
No. CallId     dstCallId  LocalRTP RmtRTP   LocalIP
RemoteIP                                  MPSS  VRF
1     323        324        5508     9256     10.64.86.90
10.65.105.60                              NO    NA
 callId 323 (dir=1): called=6010 calling=7776 redirect= loopback=NO confID=3 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                 peer callId 324: called=6010
 calling=7776 redirect= , confID:3
 , vrf = NA
   1 context 0x7F8FD8A428D0 xmitFunc 0x5605693121F0
2     324        323        5510     31826    10.64.86.90
10.64.88.52                               NO    NA
 callId 324 (dir=2): called=6010 calling=7776 redirect= loopback=NO confID=3 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                 peer callId 323: called=6010
 calling=7776 redirect= , confID:3
 , vrf = NA
   1 context 0x7F8FD8B11698 xmitFunc 0x5605693121F0
SRTP information for endpoints:
==========================================================
remote ip = 10.64.88.52, remote port=31826
 RX SRTP ROC Context (SSRC:ROC): 0xBF85C508:0x1
 TX SRTP ROC Context (SSRC:ROC): 0x1E4E1915:0x1
----------------------------------------------------------
Found 2 active RTP connections
```

In the above example, **0xBF85C508** is the Synchronization Source (SSRC) and **0x1** is the ROC. **RX SRTP ROC Context** is the crypto SRTP context for all the received streams for a media session. **TX SRTP ROC Context** is the crypto SRTP context for all the transmitted streams for a media session.

ROC increases after every RTP sequence number (maximum of 65535) roll over.

```
Router#show voip rtp connections detail
VoIP RTP Port Usage Information:
Max Ports Available: 59794, Ports Reserved: 206, Ports in Use: 2
Port range configured
                                        Min   Max   Ports     Ports     Ports
Media-Address Range                     Port  Port  Available Reserved  In-use
------------------------------------------------------------------------------
Global Media Pool                       5500  65498 29897     103       2

IP Address Based Media Pool
------------------------------------------------------------------------------
8.39.15.21          8.39.15.21          5500  65498 29897     103       0
        Port-Range                      10000 20000
------------------------------------------------------------------------------
VoIP RTP active connections :
No. CallId     dstCallId  LocalRTP RmtRTP   LocalIP
RemoteIP                                  MPSS  VRF
1     323        324        5508     9256     10.64.86.90
10.65.105.60                              NO    NA
 callId 323 (dir=1): called=6010 calling=7776 redirect= loopback=NO confID=3 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                 peer callId 324: called=6010
 calling=7776 redirect= , confID:3
 , vrf = NA
   1 context 0x7F8FD8A428D0 xmitFunc 0x5605693121F0
2     324        323        5510     31826    10.64.86.90
10.64.88.52                               NO    NA
 callId 324 (dir=2): called=6010 calling=7776 redirect= loopback=NO confID=3 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                 peer callId 323: called=6010
 calling=7776 redirect= , confID:3
 , vrf = NA
   1 context 0x7F8FD8B11698 xmitFunc 0x5605693121F0
SRTP information for endpoints:
```

```
==========================================================
remote ip = 10.64.88.52, remote port=31826
 RX SRTP ROC Context (SSRC:ROC): 0xBF85C508:0x2
 TX SRTP ROC Context (SSRC:ROC): 0x1E4E1915:0x2
----------------------------------------------------------
Found 2 active RTP connections
```

In the above example, **0xBF85C508** is the Synchronization Source (SSRC) and **0x2** is the ROC.

```
Router#show voip rtp connections detail
VoIP RTP Port Usage Information:
Max Ports Available: 59794, Ports Reserved: 206, Ports in Use: 2
Port range configured
                                        Min   Max   Ports     Ports     Ports
Media-Address Range                     Port  Port  Available Reserved  In-use
------------------------------------------------------------------------------
Global Media Pool                       5500  65498 29897     103       2

IP Address Based Media Pool
------------------------------------------------------------------------------
8.39.15.21          8.39.15.21          5500  65498 29897     103       0
        Port-Range                      10000 20000
------------------------------------------------------------------------------
VoIP RTP active connections :
No. CallId     dstCallId  LocalRTP RmtRTP   LocalIP
RemoteIP                                  MPSS  VRF
1     323        324        5508     9256     10.64.86.90
10.65.105.60                              NO    NA
 callId 323 (dir=1): called=6010 calling=7776 redirect= loopback=NO confID=3 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                 peer callId 324: called=6010
 calling=7776 redirect= , confID:3
 , vrf = NA
   1 context 0x7F8FD8A428D0 xmitFunc 0x5605693121F0
2     324        323        5510     31826    10.64.86.90
10.64.88.52                               NO    NA
 callId 324 (dir=2): called=6010 calling=7776 redirect= loopback=NO confID=3 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                 peer callId 323: called=6010
 calling=7776 redirect= , confID:3
 , vrf = NA
   1 context 0x7F8FD8B11698 xmitFunc 0x5605693121F0
SRTP information for endpoints:
==========================================================
remote ip = 10.64.88.52, remote port=31826
 RX SRTP ROC Context (SSRC:ROC): 0xBF85C508:0x1 0xF487C8FF:0x1 0xE127C8FF:0x1 0xC987C8FF:0x1
 0xD567C8FF:0x1
 TX SRTP ROC Context (SSRC:ROC): 0x1E4E1915:0x1 0xF487C8FF:0x1 0xE127C8FF:0x1 0xC987C8FF:0x1
 0xD567C8FF:0x1
----------------------------------------------------------
Found 2 active RTP connections
```

The above example shows ROC context on per SSRC basis when there are multiple SSRCs involved in a single media session (for example, during a video call).

In a High Availability scenario, the ROC updates are checkpointed and preserved across switchovers. An active router lists all the SSRCs that have received the ROC updates in the reverse chronological order. In the below example, **0xE502F046:0x2** has received ROC update most recently and **0x94A522FC:0x1** has received ROC update prior to it, and so on. If there are more than 5 SSRCs, then only the first five SSRCs (those which have received the ROC updates most recently) are considered for checkpointing.

The following example shows output from active router:

```
Device#show voip rtp connections detail
VoIP RTP Port Usage Information:
```

```
Max Ports Available: 59794, Ports Reserved: 206, Ports in Use: 2
Port range configured
                                        Min   Max   Ports     Ports     Ports
Media-Address Range                     Port  Port  Available Reserved  In-use
------------------------------------------------------------------------------
Global Media Pool                       5500  65498 29897     103       2

IP Address Based Media Pool
------------------------------------------------------------------------------
8.39.15.21          8.39.15.21          5500  65498 29897     103       0
        Port-Range                      10000 20000
------------------------------------------------------------------------------
VoIP RTP active connections :
No. CallId     dstCallId  LocalRTP RmtRTP   LocalIP                         RemoteIP
                                  MPSS  VRF
1     3          4          5500     29330    10.64.86.90
10.64.88.11                               NO    NA
 callId 3 (dir=0): called=6010 calling=7010 redirect= loopback=NO confID=1 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                   peer callId 4: called=6010
 calling=7010 redirect= , confID:1
 , vrf = NA
   1 context 0x7F378AC01E38 xmitFunc 0x55CD6A2182C0
2     4          3          5502     17580    10.64.86.90
10.64.88.52                               NO    NA
 callId 4 (dir=0): called=6010 calling=7010 redirect= loopback=NO confID=1 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                   peer callId 3: called=6010
 calling=7010 redirect= , confID:1
 , vrf = NA
   1 context 0x7F37D1CE7A38 xmitFunc 0x55CD6A2182C0
SRTP information for endpoints:
==========================================================
remote ip = 10.64.88.52, remote port=17580
 RX SRTP ROC Context (SSRC:ROC): 0xE502F046:0x2 0x94A522FC:0x1 0x79C19EC:0x1 0x8453A05E:0x8
 0xE27329A2:0x1 0xE08E9236:0x4 0xD8A97DA8:0x1 0xDCD0D1C7:0x1
 TX SRTP ROC Context (SSRC:ROC): 0xD22D83EE:0x2 0x8C9EFB1C:0x1 0x90A2D00C:0x1 0xD9C0D844:0x8
 0x54F9FA7D:0x1 0xDCA9E096:0x4 0x6D539A3B:0x1 0x5067FDE8:0x1
----------------------------------------------------------
Found 2 active RTP connections
```

The following example shows output from standby router:

```
Device#show voip rtp connections detail
VoIP RTP Port Usage Information:
Max Ports Available: 59794, Ports Reserved: 206, Ports in Use: 2
Port range configured
                                        Min   Max   Ports     Ports     Ports
Media-Address Range                     Port  Port  Available Reserved  In-use
------------------------------------------------------------------------------
Global Media Pool                       5500  65498 29897     103       2

IP Address Based Media Pool
------------------------------------------------------------------------------
8.39.15.21          8.39.15.21          5500  65498 29897     103       0
        Port-Range                      10000 20000
------------------------------------------------------------------------------
VoIP RTP active connections :
No. CallId     dstCallId  LocalRTP RmtRTP   LocalIP                         RemoteIP
                                  MPSS  VRF
1     3          4          5500     29330    10.64.86.90
10.64.88.11                               NO    NA
 callId 3 (dir=0): called=6010 calling=7010 redirect= loopback=NO confID=1 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                   peer callId 4: called=6010
 calling=7010 redirect= , confID:1
 , vrf = NA
2     4          3          5502     17580    10.64.86.90
```

```
10.64.88.52                              NO    NA
 callId 4 (dir=0): called=6010 calling=7010 redirect= loopback=NO confID=1 mode=3
rtp(tx:0/rx:0) rtcp(tx:0/rx:0) MPSS NO VRF NA                   peer callId 3: called=6010
 calling=7010 redirect= , confID:1
 , vrf = NA
SRTP information for endpoints:
==========================================================
remote ip = 10.64.88.52, remote port=17580
 RX SRTP ROC Context (SSRC:ROC): 0xE502F046:0x2 0x94A522FC:0x1 0x79C19EC:0x1 0x8453A05E:0x8
 0xE27329A2:0x1
 TX SRTP ROC Context (SSRC:ROC): 0xD22D83EE:0x2 0x8C9EFB1C:0x1 0x90A2D00C:0x1 0xD9C0D844:0x8
 0x54F9FA7D:0x1
----------------------------------------------------------
Found 2 active RTP connections
```

Table 6: show voip rtp connections Field Descriptions

Field	Description
No.	Identifier of an RTP connection in this output.
CallId	Internal call identifier of a telephony call leg (RTP connection).
dstCallId	Internal call identifier of a VoIP call leg.
LocalRTP	RTP port of the media stream for the local entity.
RmtRTP	RTP port of the media stream for the remote entity.
LocalIP	IPv4 or IPv6 address of the media stream for the local entity.
RemoteIP	IPv4 or IPv6 address of the media stream for the remote entity.
dir	0 indicates an outgoing call. 1 indicates an incoming call.
called	Extension that received the call.
calling	Extension that made the call.
redirect	Original called number if the incoming call was forwarded.
context	Internal memory address for the control block associated with the call.
xmitFunc	Internal memory address for the transmit function to which incoming RTP packets (on the H.323 and SIP side) are sent; the address for the function that delivers the packets to the ephone.
VRF	Virtual Routing and Forwarding (VRF) associated with the call.
SSRC:Index	SRTP Rollover Counter information.
RX SRTP ROC Context	The crypto SRTP context for all the received streams for a media session.
TX SRTP ROC Context	The crypto SRTP context for all the transmitted streams for a media session.

Related Commands

Command	Description
debug voip rtp	Enables debugging for RTP named event packets.
show ephone offhook	Displays information and packet counts for phones that are currently off hook.

show voip rtp forking

To display the Real-Time Transport Protocol (RTP) media-forking connections, use the **show voip rtp forking** command in privileged EXEC mode.

show voip rtp forking

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(24)T	This command was introduced in a release earlier than Cisco IOS Release 12.4(24)T.

Usage Guidelines

The **show voip rtp forking** command displays information about RTP named event packets, such as type of stream, IP address, and port for both the local and remote endpoints. The output from this command provides an overview of all the media-forking connections in the system, and this information can be used to narrow the criteria for debugging. The **debug voip rtp** command floods the console with voice packet information. You can use the **show voip rtp forking** command to display the remote IP address, or remote port identifiers that you can use to limit the output from the **debug voip rtp** command.

Examples

The following is sample output from the **show voip rtp forking** command:

```
Router# show voip rtp forking
VoIP RTP active forks :
 Fork 1
   stream type voice-only (0): count 1
     remote ip 9.13.36.101,  remote port 20590,  local port 17596
       codec g711alaw,  logical ssrc 0x60
       packets sent 237,  packets received 413
   stream type voice+dtmf (1): count 0
   stream type dtmf-only (2): count 0
   stream type voice-nearend (3): count 1
     remote ip 9.13.36.102,  remote port 18226,  local port 17434
       codec g729r8,  logical ssrc 0x103
       packets sent 39,  packets received 0
   stream type voice+dtmf-nearend (4): count 0
   stream type voice-farend (5): count 1
     remote ip 9.13.36.120,  remote port 16912,  local port 21098
       codec g729r8,  logical ssrc 0x105
       packets sent 39,  packets received 0
   stream type voice+dtmf-farend (6): count 0
   stream type video (7): count 0
```

The table below describes the significant fields shown in the display.

Table 7: show voip rtp forking Field Descriptions

Field	Description
stream type	Indicates the type of stream.

Field	Description
count	Number of packets in the specified type of stream.
remote ip	IPv4 or IPv6 address of the media stream for the remote entity.
remote port	RTP port of the media stream for the remote entity.
local port	RTP port of the media stream for the local entity.
codec	Codec supported on the specified channel.
logical ssrc	Indicates the logical synchronization source (SSRC) for the specified channel.
packets sent	Total number of packets sent from the channel.
packets received	Total number of packets received by the channel.

Related Commands

Command	Description
debug voip rtp	Enables debugging for RTP named event packets.

show voip rtp stats

To display the RTP statistics and error counters based on the configuration.

show voip rtp stats

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.9S	This command was introduced.
Cisco IOS XE Bengaluru 17.4.1a	Earlier, 'show voip rtp stats' command displayed details of ports that are allocated from the global port table only. From Cisco IOS XE Bengaluru 17.4.1a onwards, this command also displays details of ports that are allocated from the following port tables: • Media IP address based tables • Media VRF-based tables A unique identifier is generated and displayed for each table, which serves as a reference to 'clear voip rtp port' command.

The following examples display port allocations from multiple tables:

```
Router# show voip rtp stats
------------
RTP DP stats
------------
 DP:     add     add-pend   add-video     mod     mod-!rtp   del-leg1  del/dstroy
---- ---------- ---------- ---------- ---------- ---------- ---------- ----------
              4          0          0          0          0          0          0
 DP LPBK:    add     add-video     mod     mod-!rtp    del-leg1  del/dstroy
--------- ---------- ---------- ---------- ---------- ---------- ----------
                   0          0          0          0          0          0
 DP single leg:    add     mod     del/dstroy
---------------    -----   -----   -----------
                   0          0          0
 DP conf leg  :    add     mod     del/dstroy
---------------    -----   -----   -----------
                   0          0          0

dp_mod_dst_zero                  :         16
dp_mod_no_change                 :         32
dp_skip_mod_addnotdone           :          8


 VOIP RTP Max Media Loop Count : 6

VOIP RTP Stats Counters :
GCCB:Inserted =8        Removed =0
PORT:Allocated=8        Reserved=0        Released=0        Invalid Index=0        Port
Overwrite=0
```

```
SIPSPI:Leak(Avoided=0        Suspected=0       ) Lost Port Handle=0
RTSP:Leak(Avoided=0        Suspected=0       )

 VOIP RTP Error Counters :
        gccb null invalid callid (6) count  = 8
        gccb null for callid (7) count  = 18

 2 errortypes observed

                                        Min   Max   Ports     Ports     Ports
Media-Address Range                     Port  Port  Available Reserved  In-use
-------------------------------------------------------------------------------
Global Media Pool (ID :1)               10000 40000 14900     101       0

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------

                                        Min   Max   Ports     Ports     Ports
IP Address Based Media Pool (ID :4)     Port  Port  Available Reserved  In-use
-------------------------------------------------------------------------------------------
8.43.21.94          8.43.21.94          20000 30000 4900      101       4

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------
20000  Inserted     1        20000      N      N
20002  Inserted     4        20002      N      N
20004  Inserted     5        20004      N      N
20006  Inserted     8        20006      N      N

                                        Min   Max   Ports     Ports     Ports
IP Address Based Media Pool (ID :5)     Port  Port  Available Reserved  In-use
-------------------------------------------------------------------------------------------
10.65.125.167       10.65.125.167       25000 35000 5001      0         0

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------

                                        Min   Max   Ports     Ports     Ports
IP Address Based Media Pool (ID :6)     Port  Port  Available Reserved  In-use
-------------------------------------------------------------------------------------------
2001:DB8:85A3::8A2E:370:7334
2001:DB8:85A3::8A2E:370:8800            20000 30000 4900      101       0

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------

                                        Min   Max   Ports     Ports     Ports
VRF ID Based Media Pool (ID :2)         Port  Port  Available Reserved  In-use
-------------------------------------------------------------------------------------------
VRF1                                    14000 48000 16900     101       2

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------
14000  Inserted     6        14000      N      N
14002  Inserted     7        14002      N      N

                                        Min   Max   Ports     Ports     Ports
VRF ID Based Media Pool (ID :3)         Port  Port  Available Reserved  In-use
-------------------------------------------------------------------------------------------
VRF2                                    20000 48000 13900     101       2

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------
20000  Inserted     2        20000      N      N
```

```
20002  Inserted     3        20002      N      N

Total=513, GCCB(Inserted=8, Deleted=0, Null=0 Possible Leaked=0, Blocked=505)
------------------------------------------------------------------------------
```

The following example displays hung ports in a call scenario. There are two ways to determine the hung ports:

- Check the "Possible Leaked" value in the **show voip rtp stats** command output. The "Possible Leaked" value gives the total number of hung ports in all the port tables.
- Check the "Leak" flag value in each table. If it is "Y", then it is a hung port.

```
Router# show voip rtp stats
                                        Min   Max   Ports     Ports     Ports
Media-Address Range                     Port  Port  Available Reserved  In-use
------------------------------------------------------------------------------
Global Media Pool (ID :1)               8000  48198 19999     101       0

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------

                                        Min   Max   Ports     Ports     Ports
IP Address Based Media Pool (ID :2)     Port  Port  Available Reserved  In-use
------------------------------------------------------------------------------------------
8.43.21.94         8.43.21.94           10000 40000 14900     101       3

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------
10024    Null                           Y
10028    Null                           Y
10034    Null                           Y

Total=205, GCCB(Inserted=0, Deleted=0, Null=3, Possible Leaked=3, Blocked=202)
------------------------------------------------------------------------------


CSR#clear voip rtp port 2 10024,10028,10034
Any port(s) associated with an active call will not be cleared.[confirm]
Cleared port 10024
Cleared port 10028
Cleared port 10034
```

show voip rtp stats command after releasing hung ports. You can determine that there are no hung ports by performing the following:

- Check the "Possible Leaked" value in the **show voip rtp stats** command output. The "Possible Leaked" value should be zero.
- Check the "Leak" flag value in each port table. The values are removed from the tables.

```
                                        Min   Max   Ports     Ports     Ports
Media-Address Range                     Port  Port  Available Reserved  In-use
------------------------------------------------------------------------------
Global Media Pool (ID :1)               8000  48198 19999     101       0

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------
```

```
                                         Min   Max   Ports     Ports     Ports
IP Address Based Media Pool (ID :2)      Port  Port  Available Reserved  In-use
-----------------------------------------------------------------------------------------
8.43.21.94          8.43.21.94           10000 40000 14900     101       0

Port   GCCB Status  CallID   Src Port   Leak?  No call
------------------------------------------------------------

Total=202, GCCB(Inserted=0, Deleted=0, Null=0, Possible Leaked=0, Blocked=202)
-----------------------------------------------------------------------------
```

show voip stream-service callid

To display detailed information about a WebSocket call using the call ID that initiated the media forking request, use the **show voip stream-service callid** *callid* command in privileged EXEC mode.

show voip stream-service callid *callid*

Syntax Description

callid	The call control call-ID of a WebSocket call that initiated the media forking request.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1a	This command was introduced.

Usage Guidelines

The following are some of the details about the WebSocket call displayed by this show command:

- WebSocket ID
- Fork Session ID
- Call GUID
- Near-end Channel ID
- Far-end Channel ID
- Status
- TX/RX packets replicated
- TX/RX octets replicated
- TX/RX packets dropped
- TX/RX octets dropped

Examples

The following is a sample output for the command **show voip stream-service callid** *callid* for a call ID associated with a WebSocket connection.

```
router#Show voip stream-service callid 18
WebSocket id          : 11
Fork session id       : 2
Call GUID             : 3FBF760000010000001FF2691D0816AC
Near-end channel id   : 3
Far-end channel id    : 4
Status                : Active

TX packets replicated : 231
TX octets replicated  : 36960
TX packets dropped    : 0
TX octets dropped     : 0
RX packets replicated : 231
```

```
RX octets replicated  : 36960
RX packets dropped    : 0
RX octets dropped     : 0
```

The table describes significant fields shown in this output.

Table 8: Show voip stream-service callid <callid> Field Descriptions

WebSocket ID	The unique ID number associated with a WebSocket connection.
Fork Session ID	The ID number associated with the fork session of a WebSocket connection.
Call GUID	The unique ID for a WebSocket call.
Near-end Channel ID	The unique ID for the near-end (CVP side) channel of the forked call.
Far-end Channel ID	The unique ID for the far-end (CUBE side) channel of the forked call.
Status	The status of WebSocket forking: Active (media forking is in progress), Paused (media forking is on hold), Stopped (media forking is stopped).

Related Commands

Command	Description
show voip stream-service connection	Displays information about the active WebSocket connections in Unified Border Element.
show voip stream-service connection history	Displays information about all the closed WebSocket connections in Unified Border Element.
show voip stream-service server <ip:port>	Displays information about the WebSocket connection based on WebSocket server IP and port.

show voip stream-service connection

To display information about all the active WebSocket connections in CUBE, use the **show voip stream-service connection** command in privileged EXEC mode.

show voip stream-service connection

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1a	This command was introduced.

Usage Guidelines

Use this command to display the list of WebSocket connections in active state. The number of active calls and total calls is displayed for each of the active WebSocket connections. **Active Calls** are the count of active calls that uses this WebSocket connection for forking. **Total Calls** are the total number of calls that used this WebSocket connection for forking.

CUBE displays information on the **Remote Hostname** and port instead of remote IP address and port in the following scenarios:

- The JSON encoded MIME attachment in the SIP re-INVITE contains remote hostname instead of remote IP address.
- The CLI command **proxy** is configured under media profile configuration mode.

Examples

The following sample output displays active and total call information across the WebSocket IDs:

```
router#show voip stream-service connection
ID        Local IP:Port         Remote Hostname/IP:Port    Secure      Active Sessions
Total Sesions

68        10.65.125.186:30097   10.64.86.215:5022          No           10              10
66        10.65.125.186:41051   10.64.86.70:5067           Yes          1               1
30        10.65.126.206:46884   hdfwehdfgewjkgw...:8090    No           0               1

**Remote Hostname is truncated if it exceeds 15 characters.
-------------------------------------------------------------------------------------------
```

The table describes significant fields shown in this output.

Table 9: Show voip stream-service connection Field Descriptions

Local IP:Port	The IP address and port assigned to a WebSocket connection on CUBE.
Remote IP:Port	The IP address or hostname and corresponding port of the remote WebSocket server.
Active Calls	The total number of active calls on the WebSocket connection.
Total Calls	The total number of calls handled on this WebSocket connection.

Related Commands

Command	Description
show voip stream-service connection history	Displays information about all the closed WebSocket connections in CUBE.
show voip stream-service server <ip:port>	Displays information about the WebSocket connection based on WebSocket server IP and port.
show voip stream-service connection id <id>	Displays information about a WebSocket connection based on the WebSocket ID. Also, it displays all the forked call details.

show voip stream-service connection history

To display information about all the closed or stale WebSocket connections in Cisco Unified Border Element, use the **show voip stream-service connection history** command in privileged EXEC mode.

show voip stream-service connection history

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1a	This command was introduced.

Usage Guidelines

Use this command to display the list of all WebSocket connections in closed or stale state. The number of total calls and the reason for WebSocket connection disconnect is also displayed.

Note For this CLI command output, a maximum of 100 entries is supported per server.

Examples

```
router#show voip stream-service connection history
Id     Local IP:Port        Remote IP:Port      Secure  Total   Disconnect Cause
                                                        Sessions
16     10.65.125.186:41167  10.64.86.215:5022   No      5       WS_IDLE_TIMEOUT_CLOSURE

33     10.65.125.186:11079  10.64.86.215:5022   Yes     10      WS_IDLE_TIMEOUT_CLOSURE

48     10.65.125.186:38169  10.64.86.70:5067    No      1       WS_IDLE_TIMEOUT_CLOSURE

**Remote Hostname is truncated if it exceeds 15 characters
```

The table describes significant fields shown in this output.

Table 10: Show voip stream-service connection history Field Descriptions

Local ip:port	The IP address and port assigned to a WebSocket connection on Cisco Unified Border Element.
Remote ip:port	The IP address and port of the remote WebSocket server.
Total Calls	The total number of calls handled on this WebSocket connection.
Disconnect Cause	The reason for the termination of WebSocket connection.

Related Commands

Command	Description
show voip stream-service connection	Displays information about the active WebSocket connections in Unified Border Element.

Command	Description
show voip stream-service server <ip:port>	Displays information about the WebSocket connection based on WebSocket server IP and port.
show voip stream-service connection id <id>	Displays information about a WebSocket connection based on the WebSocket ID. Also, it displays all the forked call details.

show voip stream-service connection id

To display detailed information about a specific WebSocket connection in Cisco Unified Border Element, use the **show voip stream-service connection id** *id* command in privileged EXEC mode.

show voip stream-service connection id *id*

Syntax Description

id	The ID associated with a WebSocket connection.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1a	This command was introduced.
Cisco IOS XE Dublin 17.12.1a	Added support for GCM cipher suite negotiation.

Usage Guidelines

Use this command to display detailed information about a specific WebSocket connection. The information is provided based on the unique *id* associated with a WebSocket connection. The following are some of the details about the WebSocket connection displayed by the show command:

- WebSocket ID
- Total Call Count
- Active Call Count
- Server Address
- Local Address
- State
- Connection Timestamp
- Idle Timestamp
- Disconnect Cause
- Call Leg Details
- Secure
- TLS Version
- Cipher Suite
- Authorization Token

Examples

The following is a sample output for the command **show voip stream-service connection id** *id* for an active WebSocket connection.

```
router#show voip stream-service connection id 2
Id:2
```

```
Total call count:1
Active calls count:1
State: Active
Connected at: *Aug 21 20:34:43 UTC
Anchor leg cccallid        Data plane fork session id

2                          1
```

The following is a sample output for the command **show voip stream-service connection id** *id* for a disconnected WebSocket connection.

```
router#show voip stream-service connection id 16
Id:16
Total Calls:5
State: Disconnected
Connected at: *Aug 21 12:13:34 UTC
Disconnected at: *Aug 21 12:18:34 UTC
Disconnect Cause: WS_IDLE_TIMEOUT_CLOSURE
```

The following is a sample output for the command **show voip stream-service connection id** *id* for an idle WebSocket connection.

```
router#sh voip stream-service connection id 24
Id: 24
Total sessions: 1
Secure: No
Auth Token:
e2238f3a-e43c-3f54-a05a-dd2e4bd4631fe2238f3a-e43c-3f54-a05a-dd2e4bd4631fe2238f3a-e43c-3f54-a05a-dd2e...

Server Address: 8.43.24.49:2313
Local Address: 8.43.21.36:19631
Proxy : 8.43.24.189:8097
State: Disconnected
Connected at: *Oct 27 05:35:35 UTC
Disconnected at: *Oct 27 05:40:56 UTC
Disconnect Cause: WS_TCP_CONNECTION_CLOSURE
```

The following is a sample output for the command **show voip stream-service connection id** *id* for a secure WebSocket connection.

```
router#sh voip stream-service connection id 38
Id: 38
Total session count: 1
Active session count: 1
Secure: Yes
TLS Version: TLS1.2
Cipher Suite: AES128-SHA
Auth Token:
e2238f3a-e43c-3f54-a05a-dd2e4bd4631fe2238f3a-e43c-3f54-a05a-dd2e4bd4631fe2238f3a-e43c-3f54-a05a-dd2e...

Server Address: 8.43.24.49:2311
Local Address: 8.43.21.36:28469
Proxy : 8.43.24.189:8097
State: Active
Connected at: *Oct 27 05:42:27 UTC

Anchor leg cccallid        Data plane fork session id
       37                              3
```

The following is a sample output for the command **show voip stream-service connection id** *id* for a GCM specific cipher secure WebSocket connection:

```
router#show voip stream-service connection id 60
Id: 60
Total session count: 1
```

```
Active session count: 1
Secure: Yes
TLS Version: TLS1.2
Cipher Suite: ECDHE-RSA-AES256_GCM-SHA384
Auth Token: e2238f3a-e43c-3f54-a05a-dd2e4bd4631f
Server Address: 10.1.40.50:8051
Local Address: 10.2.10.10:52642
State: Active
Connected at: Feb 7 07:47:27 UTC

Anchor leg cccallid        Data plane fork session id
       58                              2
```

The table describes significant fields shown in this output.

Table 11: Show voip stream-service connection id <id> Field Descriptions

State	The current state of WebSocket connection (Active or Disconnected).
Id	The ID number associated with the WebSocket connection.
Active call count	The total number of active calls on the WebSocket connection.
Total call count	The total number of calls handled on this WebSocket connection.
Connected at	The timestamp at which the WebSocket connection was established.
Disconnected at	The timestamp when WebSocket connection was terminated. It's displayed only if the WebSocket connection is disconnected.
Disconnect Cause	The cause for termination of WebSocket connection.
Idle Since	The duration (in minutes) for which the WebSocket connection is idle.

Related Commands

Command	Description
show voip stream-service connection	Displays information about the active WebSocket connections in Unified Border Element.
show voip stream-service connection history	Displays information about all the closed WebSocket connections in Unified Border Element.
show voip stream-service server <ip:port>	Displays information about the WebSocket connection based on WebSocket server IP and port.

show voip stream-service server

To display information about all the WebSocket connections for a specific speech server ip and port, use the **show voip stream-service server** *ip:port* command in privileged EXEC mode.

show voip stream-service server *ip:port*

Syntax Description

ip:port	The IP address and port details associated with a speech server.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1a	This command was introduced.

Usage Guidelines

Use this command to display detailed information about the WebSocket connections specific to a speech server ip address and port. Detailed information about WebSocket connections such as WebSocket ID, state (active or disconnected), and total number of calls received on the WebSocket connection are displayed. If the WebSocket connection is in **Active** state, information is provided on the number of active calls. If the WebSocket connection is in **Disconnected** state, the disconnect cause is specified.

Examples

The following is a sample output for the command **show voip stream-service connection server** *ip:port* for an active WebSocket connection.

```
router#show voip stream-service server 10.64.86.70:5067
Total 2 connections found
ID        State           Secure   Total Calls  Active Session/Disconnect Cause
66        Active          Yes      1            1
48        Disconnected    No       1            WS_IDLE_TIMEOUT
```

The table describes significant fields shown in this output.

Table 12: Show voip stream-service connection id <id> Field Descriptions

State	The current state of WebSocket connection (Active or Disconnected).
Id	The ID number associated with the WebSocket connection.
Active Calls	The total number of active calls on the WebSocket connection.
Total Calls	The total number of calls handled on this WebSocket connection.
Disconnect Cause	The cause for termination of WebSocket connection.

Related Commands

Command	Description
show voip stream-service connection	Displays information about the active WebSocket connections in Unified Border Element.

Command	Description
show voip stream-service connection history	Displays information about all the closed WebSocket connections in Unified Border Element.
show voip stream-service connection id <id>	Displays information about a WebSocket connection based on the WebSocket ID. Also, it displays all the forked call details.

show voip stream-service statistics

To display statistical information about WebSocket connections in Cisco Unified Border Element, use the **show voip stream-service statistics** command in privileged EXEC mode.

show voip stream-service statistics

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1a	This command was introduced.

Usage Guidelines

The following are some of the statistical information about the WebSocket connection that is displayed by the command:

- Active connections
- Active forked sessions
- Total connections created
- Total forked sessions
- Connection closures
- Message statistics

Examples

The following is a sample output for the command **show voip stream-service statistics** .

```
router#show voip stream-service statistics
Active connections:          0
Active forked sessions:      3
Total connections created:   3
Total forked sessions:       8

Connection closures:
HTTP failures:               0
TCP failures:                0
TLS failures:                0
Remote WebSocket closures:   0
TCP connection closures:     1
Idle-timeouts:              1

Message statistics:
WS_CREATE_REQ:               3
WS_CREATE_RSP_OK:            3
WS_CREATE_RSP_FAIL:          0
WS_CLOSE_REQ:                1
WS_CLOSE_RSP:                1
WS_DOWN:                     1
WS_DOWN_ALL:                 1
```

The table describes fields that are shown in this output.

Table 13: Field Descriptions

Active connections	Number of active WebSocket connections.
Active forked sessions	Number of active forked sessions.
Total connections created	Total number of WebSocket connections created on CUBE.
Total forked sessions	Total number of forked sessions on CUBE.
Connection closures	Information about connection closures that are related to WebSockets on CUBE.
HTTP failures	Number of HTTP connection setup failures.
TCP failures	Number of TCP connection setup failures.
TLS failures	Number of Transport Layer Security (TLS) connection setup failures.
Remote WebSocket closures	Number of WebSocket connections that are closed remotely.
TCP connection closures	Number of TCP connection closures, including local and remote closures.
Idle-timeouts	Number of TCP connections closed by CUBE due to idle timeout.
Message statistics	Statistics about messages and responses for a WebSocket connection.
WS_CREATE_REQ	Count of requests for creating WebSocket connection.
WS_CREATE_RSP_OK	Count of responses for WebSocket connection requests that are successful.
WS_CREATE_RSP_FAIL	Count of responses for WebSocket connection requests that are unsuccessful.
WS_CLOSE_REQ	Count of requests for WebSocket connection closure.
WS_CLOSE_RSP	Count of responses for WebSocket connection closure.
WS_DOWN	Count of events in which WebSocket connection is down, including remote and local closures.
WS_DOWN_ALL	Count of all WebSocket down events during switchover. All WebSocket connections are closed during a Forwarding Plane (FP) Switchover. Each count represents one FP switchover.

Related Commands

Command	Description
clear voip stream-service statistics	Clears global WebSocket connections for CUBE.
show voip stream-service connection history	Displays information about all the closed WebSocket connections in CUBE.
show voip stream-service server <ip:port>	Displays information about the Server IP and port of a WebSocket connection.

show voip trace

To display the VoIP trace information for SIP calls received and sent on CUBE, use the **show voip trace** command in privileged EXEC mode.

show voip trace { **all** | **call-id** *identifier* | **correlator** *identifier* | **cover-buffers** | **session-id** *identifier* | **sip-call-id** *identifier* | **statistics** [**detail**] | **tenant** *identifier* }

Syntax Description

all	Displays all the traces for both active and disconnected calls.
call-id	Displays detailed call information for a call based on the CCAPI call ID.
correlator	Displays detailed call information for a call based on the VOIP FPI correlator ID.
cover-buffers	Displays the summary of cover buffers for all the buffers in the memory.
session-id	Displays detailed call information for a call based on the SIP session-ID.
sip-call-id	Displays detailed call information for a call based on the SIP call ID in a SIP INVITE message.
statistics	Displays the VoIP trace statistics for incoming and outgoing calls.
detail	(Optional) Displays the detailed VoIP trace statistics for incoming and outgoing calls.
tenant	Displays detailed call information for calls based on tenant tags.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.5.1a	The **show voip trace** command for SIP messages is enhanced to display the source and destination IP addresses.
Cisco IOS XE Cupertino 17.8.1a	The **show voip trace** command: • Enhanced to display the tenant tag. • Supports tenant based filtering.
Cisco IOS XE Dublin 17.12.1a	The **show voip trace** command for SIP messages is enhanced to display cause code in the cover buffer.

Usage Guidelines

Use **show voip trace** command to display the statistics, details of the memory expansion counters for message buffers to store additional logs, and information from trace buffers for all SIP call legs. When VoIP Trace is enabled (no shutdown), use **show voip trace** command to display information for all SIP call legs in the trace buffers.

Starting from Cisco IOS XE Dublin 17.12.1a, VoIP trace format is updated to include cause-code information display in the cover buffer.

Note

- If the count of cover buffers exceeds the threshold value 200, the router performance reduces for **show voip trace all** command. Use **show voip trace cover-buffers** and other filter commands instead of **show voip trace all**.
- To change the Timestamps displayed in the VoIP trace, configure the following:

```
router(config)# monitor event-trace timestamps datetime ?
  localtime      Use local time zone for timestamps
  msec           Include milliseconds in timestamp
  show-timezone  Add time zone information to timestamp
```

Examples

The following sample output displays voip trace information for cover-buffers. The tenant tag can be acquired using the **show voip trace cover-buffers**.

```
router# show voip trace cover-buffers
------------------ Cover Buffer ---------------
Search-key       = sipp:8765:121
  Timestamp      = Nov  9 04:47:39.427
  Buffer-Id      = 1
  CallID         = 121
  Peer-CallID    = 122
  Correlator     = 7
  Called-Number  = 8765
  Calling-Number = sipp
  SIP CallID     = 1-28575@8.41.17.71
  SIP Session ID = b91e516ba375585aae54b3f0abdd6f13
  GUID           = 87954DCE80A7
  Tenant         = 100
```

The following is a sample output of the **show voip trace cover-buffers** command displaying cause-code, which is supported from Cisco IOS XE Dublin 17.12.1a release:

```
Device# show voip trace cover-buffers
------------------ Cover Buffer ---------------
Search-key       = 808808:6666:4
  Timestamp      = Apr 27 09:54:54.491
  CallID         = 4
  Peer-CallID    = 5
  Correlator     = NA
  Called-Number  = 6666
  Calling-Number = 808808
  SIP CallID     = 1-18630@10.1.40.50
  SIP SessionID  =
  GUID           = 651A3C548005
  Tenant         = 0
  Cause-code     = recovery on timer expiry (102)
-----------------------------------------------
```

The following sample output displays a tenant based filtering for voip trace:

```
router# show voip trace ?
  all            Display all VoIP Traces
  call-id        Filter traces based on Internal Call Id
  correlator     Filter traces based on FPI Correlator
  cover-buffers  Display the summary of all cover buffers
  session-id     Filter traces based on SIP Session ID
  sip-call-id    Filter traces based on SIP Call Id
```

```
  statistics     Display statistics for VoIP Trace
  tenant         Filter traces based on tenant tags

vCUBE1# show voip trace tenant ?
 <0-10000>  Tenant tag to be matched, tag 0 indicates calls associated with no tenant (max
 of 10 tags)

vCUBE1# show voip trace tenant 1 ?
 <0-10000>  Tenant tag to be matched, tag 0 indicates calls associated with no tenant (max
 of 10 tags)
<cr>            <cr>

vCUBE1# show voip trace tenant  0 1 2 3 4 5 6 7 8 9 ?
<cr>            <cr>
```

Examples

The following sample output displays voip trace information for a call with call-id 121. The call id can be acquired using the **show voip trace cover-buffers**.

The following sample output displays voip trace information for a IPv6 call:

```
router# show voip trace call-id 39
------------------ Cover Buffer ---------------
Search-key       = sipp:5678:39
  Timestamp      = *Dec 25 22:09:00.068
  Buffer-Id      = 1
  CallID         = 39
  Peer-CallID    = 40
  Correlator     = 16
  Called-Number  = 5678
  Calling-Number = sipp
  SIP CallID     = 1-8921@2001:420:54ff:13::312:71
  SIP Session ID = d921890ab3aa557891b6dd2888b0602b
  GUID           = 9FF305D88076
-----------------------------------------------
2232: *Dec 25 22:09:00.068: //39/9FF305D88076/CUBE_VT/SIP/Msg/ccsipDisplayMsg:
Received: SIP UDP message from [2001:420:54FF:13::312:71]:10000 to
[2001:420:54FF:13::652:23]:5060
INVITE sip:5678@[2001:420:54ff:13::652:23]:5060 SIP/2.0
Via: SIP/2.0/UDP [2001:420:54ff:13::312:71]:10000;branch=z9hG4bK-8921-1-0
From: sipp <sip:sipp@[2001:420:54ff:13::312:71]:10000>;tag=8921SIPpTag001
To: sut <sip:5678@[2001:420:54ff:13::652:23]:5060>
Call-ID: 1-8921@2001:420:54ff:13::312:71
CSeq: 1 INVITE
Contact: sip:sipp@[2001:420:54ff:13::312:71]:10000

Max-Forwards: 70
Subject: Performance Test
Content-Type: application/sdp
Content-Length:   161

v=0
o=user1 53655765 2353687637 IN IP6 [2001:420:54ff:13::312:71]
s=-
c=IN IP6 2001:420:54ff:13::312:71
t=0 0
m=audio 6001 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2234: *Dec 25 22:09:00.067: //39/9FF305D88076/CUBE_VT/SIP/FSM/SPI-State-Change: Current
State = STATE_NONE, Next State = STATE_IDLE, Current Sub-State = STATE_NONE, Next Sub-State
 = STATE_NONE
2235: *Dec 25 22:09:00.069: //39/9FF305D88076/CUBE_VT/SIP/MISC/Matched Dialpeer: Dir:Inbound,
 Peer-Tag:  3
```

```
2236: *Dec 25 22:09:00.069: //39/9FF305D88076/CUBE_VT/SIP/FSM/Offer-Answer: Event =
E_SIP_INVITE_SDP_RCVD, Current State = S_SIP_EARLY_DIALOG_IDLE, Next State =
S_SIP_EARLY_DIALOG_OFFER_RCVD
2237: *Dec 25 22:09:00.069: //39/9FF305D88076/CUBE_VT/SIP/FSM/IWF: Event =
E_SIP_IWF_EV_RCVD_SDP, Current State = S_SIP_IWF_SDP_IDLE, Next State =
S_SIP_IWF_SDP_RCVD_AWAIT_PEER_EVENT
2238: *Dec 25 22:09:00.070: //39/9FF305D88076/CUBE_VT/SIP/MISC/Media Stream Parameters:
Stream Type = voice-only, Stream State = STREAM_ADDING Negotiated Codec = g711ulaw, Negotiated
 DTMF Type = inband-voice, Stream Index =  1
2239: *Dec 25 22:09:00.071: //39/9FF305D88076/CUBE_VT/SIP/API:
cc_api_update_interface_cac_resource (0)
2240: *Dec 25 22:09:00.071: //39/9FF305D88076/CUBE_VT/SIP/API: voip_rtp_allocate_port (8020)
2241: *Dec 25 22:09:00.071: //39/9FF305D88076/CUBE_VT/SIP/MISC/Media Stream Parameters:
Stream Type = voice-only, Stream State = STREAM_ADDING Negotiated Codec = g711ulaw, Negotiated
 DTMF Type = inband-voice, Stream Index =  1
2242: *Dec 25 22:09:00.071: //39/9FF305D88076/CUBE_VT/SIP/API:
cc_api_call_setup_ind_with_callID (0)
2243: *Dec 25 22:09:00.072: //39/9FF305D88076/CUBE_VT/SIP/FSM/SPI-State-Change: Current
State = STATE_IDLE, Next State = STATE_RECD_INVITE, Current Sub-State = STATE_NONE, Next
Sub-State = STATE_NONE
2248: *Dec 25 22:09:00.073: //39/9FF305D88076/CUBE_VT/SIP/FSM/IWF: Event =
E_SIP_IWF_EV_SET_MODE, Current State = CNFSM_CONTAINER_STATE, Next State =
CNFSM_NO_STATE_CHANGE
2249: *Dec 25 22:09:00.074: //39/9FF305D88076/CUBE_VT/SIP/API: voip_rtp_create_session (0)
2250: *Dec 25 22:09:00.074: //39/9FF305D88076/CUBE_VT/SIP/API: voip_rtp_set_non_rtp_call
(0)
2251: *Dec 25 22:09:00.074: //39/9FF305D88076/CUBE_VT/SIP/API: voip_rtp_update_callinfo (0)
2252: *Dec 25 22:09:00.074: //39/9FF305D88076/CUBE_VT/SIP/FSM/Event-Action: Event =
SIPSPI_EV_CC_CALL_PROCEEDING, Current State = STATE_RECD_INVITE
2272: *Dec 25 22:09:00.077: //39/9FF305D88076/CUBE_VT/SIP/Msg/ccsipDisplayMsg:
Sent: SIP UDP message from [2001:420:54FF:13::652:23]:5060 to [2001:420:54FF:13::312:71]:10000

SIP/2.0 100 Trying
Via: SIP/2.0/UDP [2001:420:54ff:13::312:71]:10000;branch=z9hG4bK-8921-1-0
From: sipp <sip:sipp@[2001:420:54ff:13::312:71]:10000>;tag=8921SIPpTag001
To: sut <sip:5678@[2001:420:54ff:13::652:23]:5060>
Date: Fri, 25 Dec 2020 22:09:00 GMT
Call-ID: 1-8921@2001:420:54ff:13::312:71
CSeq: 1 INVITE
Allow-Events: telephone-event
Server: Cisco-SIPGateway/IOS-17.5.20201117.131853
Session-ID: 00000000000000000000000000000000;remote=e714644e7e385e90a1d75a34855ef73a
Content-Length: 0
```

The table describes significant fields that are shown in this output.

Table 14: Show voip trace Field Descriptions

Search-key	Displays the Search-key of the cover buffer. The Search-key value is the **Calling number:Called number:call ID**.
Timestamp	Displays the creation time of the cover buffer.
Buffer-Id	Displays the Buffer ID of the cover buffer.
CallID	Displays the Call ID of the respective call leg present in the cover buffer.
Peer-CallID	Displays the Peer Call ID of the respective call leg present in the cover buffer.
Correlator	Displays the Correlator ID of the respective call leg present in the cover buffer.

Called-Number	Displays the Called Number of the respective call leg present in the cover buffer.
Calling-Number	Displays the Calling Number of the respective call leg present in the cover buffer.
SIP CallID	Displays the SIP Call ID of the respective call leg present in the cover buffer.
SIP-Session ID	Displays the SIP Session ID of respective call leg present in the cover buffer.
GUID	Displays the GUID of the respective call leg present in the cover buffer.
Anchor Leg	Indicates whether the call leg in the buffer acts as the Anchor leg during recording.
Forked Leg	Indicates whether the call leg in the buffer acts as the Forked leg during recording.
Associated CallID's	Displays the Call IDs associated with forking.
tenant	Displays the tenant tag of the respective call leg present in the cover buffer.
cause-code	Starting from Cisco IOS XE Dublin 17.12.1a, displays the call success or call failure cause-code in the cover buffer for a call leg.

The following is sample output of show voip trace statistics after disabling voip trace:

```
router# show voip trace statistics
VoIP Trace Statistics
Tracing status              : DISABLED
router#
```

The following is sample output of show voip trace statistics after missing 50 call legs due to memory exhaustion:

```
router# show voip trace statistics
VoIP Trace Statistics
Tracing status              : ENABLED at Jun 15 10:01:24.911
Memory limit configured     : 10485760 bytes
Memory consumed             : 10039760 bytes (95%)
Total call legs dumped      : 3
Oldest trace dumped         : Jun 15 10:03:31.121, Search-key: sipp:799:200
Latest trace dumped         : Jun 15 10:25:03.616, Search-key: sipp:123:293
Total call legs captured    : 243
Total call legs available   : 116
Oldest trace available      : Jun 15 10:19:31.844, Search-key: sipp:799:125
Latest trace available      : Jun 15 10:25:03.616, Search-key: sipp:123:293
Total traces missed         : 50
router#
```

The table describes significant fields that are shown in this output.

Table 15: Show voip trace statistics Field Descriptions

Tracing status	Displays the timestamp, and the tracing status is enabled or disabled.
Memory limit configured	Displays the total memory-limit.
Memory consumed	Displays the current memory that is consumed by the buffers. The memory that is consumed is also displayed in percentage.
Total call legs dumped	Displays the total marked buffers that are dumped in the logging buffer.

Oldest trace dumped	Displays the timestamp and the search key of the first buffer dumped.
Latest trace dumped	Displays the timestamp and the search key of the newest buffer dumped.
Total call legs captured	Displays the total call legs that are captured after the trace is enabled.
Total call legs available	Displays the total call legs available in the history.
Oldest trace available	Displays the timestamp and the search key of the oldest buffer.
Latest trace available	Displays the timestamp and search key of the latest buffer.
Total traces missed	Displays the number of call legs missed due to memory-limit.

The following is a sample output of the **show voip trace cover-buffers**:

```
router# show voip trace cover-buffers
------------------ Cover Buffer ---------------
Search-key       = sipp:799:1
  Timestamp      = *Jun 25 14:55:35.318
  Buffer-Id      = 1
  CallID         = 1
  Peer-CallID    = 2
  Correlator     = NA
  Called-Number  = 799
  Calling-Number = sipp
  SIP CallID     = 1-630@10.64.86.70
  SIP Session ID =
  GUID           = C250D2778002
-----------------------------------------------
------------------ Cover Buffer ---------------
Search-key       = sipp:799:2
  Timestamp      = *Jun 25 14:55:35.338
  Buffer-Id      = 2
  CallID         = 2
  Peer-CallID    = 1
  Correlator     = NA
  Called-Number  = 799
  Calling-Number = sipp
  SIP CallID     = C254A2BD-B62A11EA-8008BF9C-3C4C9D37@8.43.21.71
  SIP Session ID =
  GUID           = C250D2778002
-----------------------------------------------
```

The following is sample output of the **show voip trace statistics detail**:

```
router# show voip trace statistics detail
VoIP Trace Statistics
Tracing status              : ENABLED at Jun 29 07:48:56.973
Memory limit configured     : 1048576000 bytes
Memory consumed             : 1000006016 bytes (95%)
Total call legs dumped      : 7298
Oldest trace dumped         : Jun 29 07:57:30.503, Search-key: 205521:405521:10043
Latest trace dumped         : Jun 29 09:41:44.251, Search-key: 218221:418221:69148
Total call legs captured    : 69148
Total call legs available   : 57851
Oldest trace available      : Jun 29 08:41:06.687, Search-key: 205521:405521:11043
Latest trace available      : Jun 29 10:13:21.091, Search-key: 218221:418221:69148
Total traces missed         : 0
```

```
Buffer Expansion Counters  :
============================================================
        Expansions      MSG     FSM     API     MISC
============================================================
            1           3517    0       0       0
            2           1441    0       0       0
            3           29      0       0       0
            4           629     0       0       0
            5           0       0       0       0
            6           0       0       0       0
            7           0       0       0       0
            8           0       0       0       0
            9           0       0       0       0
            10+         0       0       0       0
============================================================
```

The table describes significant fields that are shown in this output.

Table 16: Show detailed voip trace statistics detail Field Descriptions

Expansions	Displays the number of memory expansions that are performed to store the additional logs. For example, CUBE performed 1 message buffer expansion for storing SIP messages for 3517 cover buffers or 4 message buffer expansions for storing SIP messages for 629 cover buffers.
MSG	Displays the number of times the SIP message trace buffers have expanded.
FSM	Displays the number of times the Finite (Call) State Machine call trace buffers have expanded.
API	Displays the number of times the Functional call trace buffers have expanded.
Misc	Displays the number of times the Miscellaneous call trace buffers have expanded.

Examples

If you configure the CLI command **shutdown** in trace configuration sub-mode, the show command doesn't display trace information. The following is a sample show command output for the scenario:

```
router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#voice service voip
router(conf-voi-serv)#trace
router(conf-serv-trace)#shutdown
router(conf-serv-trace)#exit
router(conf-voi-serv)#exit
router(config)#end
router#show voip trace all | sec Cover Buffer
router#show voip trace all
                 No Data to Display !!

router#show voip trace call-id 7
                 No records for the filter specified !!

router#
```

Related Commands

Command	Description
trace	Enables the VoIP trace serviceability framework for SIP calls in CUBE.

Command	Description
shutdown (trace)	Disables the VoIP trace serviceability framework in CUBE.
memory-limit (trace)	Defines the memory limit for storing VoIP trace information.

show voip trunk group

To display the internal list of voip trunk groups, use the **show voip trunk group** command in user EXEC or privileged EXEC mode.

show voip trunk group

Syntax Description

This command has no arguments or keywords.

Command Default

Command Modes

User EXEC (>)
Privileged EXEC (#)

Command History

Release	Modification
15.2(2)T	This command was introduced.

Usage Guidelines

Use this command to display VOIP trunk groups.

Examples

The following example is a sample output from the **show voip trunk group** command.

```
Router# show voip trunk group

=====================================================
name: 1
protocol: cisco
ip: 1.3.45.2
xsvc: TRUE
```

Related Commands

Command	Description
voip trunk group	Specifies a VOIP trunk group.

show vrm active_calls

To display active-only voice calls either for a specific voice feature card (VFC) or for all VFCs, use the **show vrm active_calls**command in privileged EXEC mode.

show vrm active_calls {*dial-shelf-slot-number* | **all**}

Syntax Description

dial -shelf-slot-number	Slot number of the dial shelf. Range is from 0 to 13.
all	Displays list of all active calls for VFC slots.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(7)T	This command was introduced on the Cisco AS5800.

Usage Guidelines

Use this command to display active-only voice calls either for a specific VFC or for all VFCs. Each active call occupies a block of information describing the call. This information provides basically the same information as the **show vrm vdevice** command.

Examples

The following is sample output from this command specifying a dial-shelf slot number:

```
Router# show vrm active_calls 6
slot =  6 virtual voice dev (tag) =  61 channel id = 2
capabilities list map = 9FFF
last/current codec loaded/used = None
TDM timeslot = 241
Resource (vdev_common) status = 401 means :active others
tot ingress data =  24
tot ingress control  = 1308
tot ingress data drops  = 0
tot ingress control drops  = 0
tot egress data  = 22051
tot egress control  = 1304
tot egress data drops  = 0
tot egress control drops  = 0
slot =  6 virtual voice dev (tag) =  40 channel id = 2
capabilities list map = 9FFF
last/current codec loaded/used = None
TDM timeslot = 157
Resource (vdev_common) status = 401 means :active others
```

The table below describes significant fields shown in this output.

Table 17: show vrm active_calls Field Descriptions

Field	Description
slot	Slot where the voice card is installed.

Field	Description
virtual voice dev (tag)	ID number of the virtual voice device.
channel id	ID number of the channel associated with this virtual voice device.
capability list map	Bitmaps for the codec supported on that DSP channel. Values are the following: • CC_CAP_CODEC_G711U: 0x1 • CC_CAP_CODEC_G711A: 0x2 • CC_CAP_CODEC_G729IETF: 0x4 • CC_CAP_CODEC_G729a: 0x8 • CC_CAP_CODEC_G726r16: 0x10 • CC_CAP_CODEC_G726r24: 0x20 • CC_CAP_CODEC_G726r32: 0x40 • CC_CAP_CODEC_G728: 0x80 • CC_CAP_CODEC_G723r63: 0x100 • CC_CAP_CODEC_G723r53: 0x200 • CC_CAP_CODEC_GSM: 0x400 • CC_CAP_CODEC_G729b: 0x800 • CC_CAP_CODEC_G729ab: 0x1000 • CC_CAP_CODEC_G723ar63: 0x2000 • CC_CAP_CODEC_G723ar53: 0x4000 • CC_CAP_CODEC_G729: 0x8000
last/current codec loaded/used	Last codec loaded or used.
TDM time slot	Time-division-multiplexing time slot.
Resource (vdev_common) status	Current status of the VFC.
tot ingress data	Total amount of data (number of packets) sent from the PSTN side of the connection to the VoIP side of the connection.
tot ingress control	Total number of control packets sent from the PSTN side of the connection to the VoIP side of the connection.
tot ingress data drops	Total number of data packets dropped from the PSTN side of the connection to the VoIP side of the connection.
tot ingress control drops	Total number of control packets dropped from the PSTN side of the connection to the VoIP side of the connection.

Field	Description
tot egress data	Total amount of data (number of packets) sent from the VoIP side of the connection to the PSTN side of the connection.
tot egress control	Total number of control packets sent from the VoIP side of the connection to the PSTN side of the connection.
tot egress data drops	Total number of data packets dropped from the VoIP side of the connection to the PSTN side of the connection.
tot egress control drops	Total number of control packets dropped from the VoIP side of the connection to the PSTN side of the connection.

Related Commands

Command	Description
show vrm vdevices	Displays detailed information for a specific DSP or a brief summary display for all VFCs.

show vrm vdevices

To display detailed information for a specific digital signal processor (DSP) or summary information for all voice feature cards (VFCs), use the **show vrm vdevices**command in privileged EXEC mode.

show vrm vdevices {**vfc-slot-number voice-device-number** | **alarms** [**vfc-slot-number-for-alarms**] | **summary**}

Syntax Description

vfc -slot-number	Slot number of the VFC. Range is from 0 to 11.
voice -device-number	DSP number. Range is from 1 to 96.
alarms	DSP alarm statistics for all DSPs on all slots or specified slots.
vfc -slot-number-for-alarms	(Optional) Slots for which you need alarm information. If no slots are specified, alarm information for all slots is displayed.
summary	Synopsis of voice feature card DSP mappings, capabilities, and resource states.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(7)T	This command was introduced on the Cisco AS5800.
12.2(11)T	The **alarms** keyword and *vfc-slot-number-for-alarms* argument were added.

Usage Guidelines

Use this command to display detailed information for a specific DSP or a brief summary for all VFCs. The display provides information such as the number of channels, channels per DSP, bitmap of digital signal processor modules (DSPMs), DSP alarm statistics, and version numbers. This information is useful in monitoring the current state of your VFCs.

The display for a specific DSP provides information on the codec that each channel is using, if active, or on the codec that was last used and whether the channel is not currently sending cells. It also displays the state of the resource. In most cases, if there is an active call on that channel, the resource should be marked active. If the resource is marked as reset or bad, this may be an indication of a response loss for the VFC on a reset request. If this condition persists, you might experience a problem with the communication link between the router shelf and the VFC.

Examples

The following is sample output from this command specifying dial-shelf slot number and DSP number. In this particular example, the call is active so the statistics displayed are for this active call. If no calls are currently active on the device, the statistics would be for the previous (or last active) call.

```
Router# show vrm vdevices 6 1
slot =  6 virtual voice dev (tag) =  1 channel id = 1
capabilities list map = 9FFF
last/current codec loaded/used = None
TDM timeslot = 0
```

```
Resource (vdev_common) status = 401 means :active others
tot ingress data =  101
tot ingress control  = 1194
tot ingress data drops  = 0
tot ingress control drops  = 0
tot egress data  = 39722
tot egress control  = 1209
tot egress data drops  = 0
tot egress control drops  = 0
slot =  6 virtual voice dev (tag) =  1 channel id = 2
capabilities list map = 9FFF
last/current codec loaded/used = None
TDM timeslot = 1
Resource (vdev_common) status = 401 means :active others
tot ingress data =  21
tot ingress control  = 1167
tot ingress data drops  = 0
tot ingress control drops  = 0
tot egress data  = 19476
tot egress control  = 1163
tot egress data drops  = 0
tot egress control drops  = 0
```

The table below describes significant fields shown in this output.

Table 18: show vrm vdevices Field Descriptions

Field	Description
slot	Slot in which the voice card is installed.
virtual voice dev (tag)	ID number of the virtual voice device.
channel id	ID number of the channel that is associated with this virtual voice device.

Field	Description
capabilities list map	Bitmaps for the codec supported on that DSP channel. Values are as follows: • CC_CAP_CODEC_G711U: 0x1 • CC_CAP_CODEC_G711A: 0x2 • CC_CAP_CODEC_G729IETF: 0x4 • CC_CAP_CODEC_G729a: 0x8 • CC_CAP_CODEC_G726r16: 0x10 • CC_CAP_CODEC_G726r24: 0x20 • CC_CAP_CODEC_G726r32: 0x40 • CC_CAP_CODEC_G728: 0x80 • CC_CAP_CODEC_G723r63: 0x100 • CC_CAP_CODEC_G723r53: 0x200 • CC_CAP_CODEC_GSM: 0x400 • CC_CAP_CODEC_G729b: 0x800 • CC_CAP_CODEC_G729ab: 0x1000 • CC_CAP_CODEC_G723ar63: 0x2000 • CC_CAP_CODEC_G723ar53: 0x4000 • CC_CAP_CODEC_G729: 0x8000 • CC_CAP_CODEC_GSMEFR: 0x40000 • CC_CAP_CODEC_T38FAX: 0x10000
last/current codec loaded/used	Last codec loaded or used.
TDM timeslot	Time-division-multiplexing time slot.

Field	Description
Resource (vdev_common) status	Current status of the VFC. Values are as follows: • FREE = 0x0000 • ACTIVE_CALL = 0x0001 • BUSYOUT_REQ = 0x0002 • BAD = 0x0004 • BACK2BACK_TEST = 0x0008 • RESET = 0x0010 • DOWNLOAD_FILE = 0x0020 • DOWNLOAD_FAIL = 0x0040 • SHUTDOWN = 0x0080 • BUSY = 0x0100 • OIR = 0x0200 • HASLOCK = 0x0400 /* vdev_pool has locked port */ • DOWNLOAD_REQ = 0x0800 • RECOVERY_REQ = 0x1000 • NEGOTIATED = 0x2000 • OOS = 0x4000
tot ingress data	Total amount of data (number of packets) sent from the public switched telephone network (PSTN) side of the connection to the VoIP side of the connection.
tot ingress control	Total number of control packets sent from the PSTN side of the connection to the VoIP side of the connection.
tot ingress data drops	Total number of data packets dropped from the PSTN side of the connection to the VoIP side of the connection.
tot ingress control drops	Total number of control packets dropped from the PSTN side of the connection to the VoIP side of the connection.
tot egress data	Total amount of data (number of packets) sent from the VoIP side of the connection to the PSTN side of the connection.
tot egress control	Total number of control packets sent from the VoIP side of the connection to the PSTN side of the connection.
tot egress data drops	Total number of data packets dropped from the VoIP side of the connection to the PSTN side of the connection.

Field	Description
tot egress control drops	Total number of control packets dropped from the VoIP side of the connection to the PSTN side of the connection.

The following sample output displays alarm statistics for slot 6 of the DSP.

```
Router# show vrm vdevices alarms 6
----------------------ALARM STATISTICS FOR SLOT 6 ------------------------
TAG Mod DSP Chn OperStat  AlmCnt   AlmTime     AlmCause     AlmText
------------------------------------------------------------------------
1   1   1   1   READY CD  0        0           1
            2   READY CD  0        0           1
2   1   2   1   READY CD  0        0           1
            2   READY CD  0        0           1
3   1   3   1   READY CD  0        0           1
            2   READY CD  0        0           1
4   1   4   1   READY CD  0        0           1
            2   READY CD  0        0           1
5   1   5   1   READY CD  0        0           1
            2   READY CD  0        0           1
6   1   6   1   READY CD  0        0           1
            2   READY CD  0        0           1
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
7   2   1   1   READY CD  0        0           1
            2   READY CD  0        0           1
8   2   2   1   READY CD  0        0           1
            2   READY CD  0        0           1
9   2   3   1   READY CD  0        0           1
            2   READY CD  0        0           1
10  2   4   1   READY CD  0        0           1
!
94  16  4   1   READY CD  0        0           1
            2   READY CD  0        0           1
95  16  5   1   READY CD  0        0           1
            2   READY CD  0        0           1
96  16  6   1   READY CD  0        0           1
            2   READY CD  0        0           1
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

The table below describes significant fields shown in this output.

Table 19: show vrm vdevices alarms Field Descriptions

Field	Description
TAG	Logical tag number.
Mod	DSP module number.
DSP	DSP number within the module.
Chn	Channel number for the DSP within the module.
OperStat	Operational status of the channel.
AlmCnt	Alarm count since bootup on that channel.
AlmTime	Time at which last alarm message was received.

Field	Description
AlmCause	Cause of last alarm message received.
AlmText	Text message corresponding to the last alarm message.
Possible Values for the Operational Status of the Channel (OperStat)	
RESET	RESET state.
DOWN	DOWN state.
READY CR	CORE READY state.
READY CD	CODEC READY state.
IDLE V	VOICE IDLE state.
IDLE FAX	FAX IDLE state.
READY V	VOICE READY state.
READY FX	FAX READY state.
READY D	DTMF READY state.
UNKNOWN	UNKNOWN state.

The following is sample output from this command specifying a summary list. In the "Voice Device Mapping" area, the "C_Ac" column indicates the number of active calls for a specific DSP. If there are any nonzero numbers under the "C_Rst" and/or "C_Bad" column, a reset request was sent, but it was lost; this could mean a faulty DSP.

```
Router# show vrm vdevices summary
***********************************************************
******summary of voice devices for all voice cards*********
***********************************************************
slot = 6 major ver = 0 minor ver = 1 core type used = 2
number of modules = 16 number of voice devices (DSPs) = 96
chans per vdevice = 2 tot chans = 192 tot active calls = 178
module presense bit map = FFFF tdm mode = 1 num_of_tdm_timeslots = 384
auto recovery is on
number of default voice file (core type images) = 2
file 0 maj ver = 0 min ver = 0 core_type = 1
trough size = 2880 slop value = 0 built-in codec bitmap = 0
loadable codec bitmap = 0 fax codec bitmap = 0
file 1 maj ver = 3 min ver = 1 core_type = 2
trough size = 2880 slop value = 1440 built-in codec bitmap = 40B
loadable codec bitmap = BFC fax codec bitmap = 7E
-------------------Voice Device Mapping------------------------
Logical Device (Tag)  Module#  DSP#  C_Ac  C_Busy  C_Rst  C_Bad
---------------------------------------------------------------
1                     1        1     2     0       0      0
2                     1        2     2     0       0      0
3                     1        3     2     0       0      0
4                     1        4     2     0       0      0
5                     1        5     2     0       0      0
```

```
6                     1        6      2      0        0       0
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
7                     2        1      2      0        0       0
8                     2        2      2      0        0       0
9                     2        3      2      0        0       0
10                    2        4      1      0        0       0
11                    2        5      2      0        0       0
12                    2        6      1      0        0       0
.
.
.
91                    16       1      2      0        0       0
92                    16       2      2      0        0       0
93                    16       3      1      0        0       0
94                    16       4      2      0        0       0
95                    16       5      2      0        0       0
96                    16       6      2      0        0       0
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Total active call channels = 178
Total busied out channels = 0
Total channels in reset = 0
Total bad channels = 0
Note :Channels could be in multiple states
```

The table below describes significant fields shown in this output.

Table 20: show vrm vdevices summary Field Descriptions

Field	Description
slot	Slot number in which the VFC is installed.
major ver	Major version of firmware running on the VFC.
minor ver	Minor version of firmware running on the VFC.
core type used	Type of DSPware in use. Values are as follows: • 1 = UBL (boot loader) • 2 = high complexity core • 3 = medium complexity core • 4 = low complexity core • 255 = invalid
number of modules	Number of modules on the VFC. Maximum number is 16.
number of voice devices (DSP)s	Number of possible DSPs. Maximum number is 96.
chans per vdevice	Number of channels (meaning calls) that each DSP can handle.
tot chans	Total number of channels.
tot active calls	Total number of active calls on this VFC.
module presense bit map	Indicates a 16-bit bitmap, each bit representing a module.

Field	Description
tdm mode	Time-division-multiplex bus mode. Values are as follows: • 0 = VFC is in classic mode. • 1 = VFC is in plus mode. This field should always be 1.
num_of_tdm_timeslots	Total number of calls that can be handled by the VFC.
auto recovery	Whether auto recovery is enabled. When autorecovery is enabled, the VRM tries to recover a DSP by resetting it if, for some reason, the DSP stops responding.
number of default voice file (core type images)	Number of DSPware files in use.
number of default voice file (maj ver)	Major version of the DSPware in use.
min ver	Minor version of the DSPware in use.
core_type	Type of DSPware in use. Values are as follows: • 1 = boot loader • 2 = high complexity core • 3 = medium complexity core • 4 = low complexity core
trough size	Indirect representation of the complexity of the DSPware in use. **Note** Effective with Cisco IOS Release 12.1(5)XM, this value is no longer displayed.
slop value	Indirect representation of the complexity of the DSPware in use. **Note** Effective with Cisco IOS Release 12.1(5)XM, this value is no longer displayed.

Field	Description
built-in codec bitmap	Bitmap of the codec built into the DSP firmware. Values are as follows: • CC_CAP_CODEC_G711U: 0x0001 • CC_CAP_CODEC_G711A: 0x0002 • CC_CAP_CODEC_G729IETF: 0x0004 • CC_CAP_CODEC_G729a: 0x0008 • CC_CAP_CODEC_G726r16: 0x0010 • CC_CAP_CODEC_G726r24: 0x0020 • CC_CAP_CODEC_G726r32: 0x0040 • CC_CAP_CODEC_G728: 0x0080 • CC_CAP_CODEC_G723r63: 0x0100 • CC_CAP_CODEC_G723r53: 0x0200 • CC_CAP_CODEC_GSM: 0x0400 • CC_CAP_CODEC_G729b: 0x0800 • CC_CAP_CODEC_G729ab: 0x1000 • CC_CAP_CODEC_G723ar63: 0x2000 • CC_CAP_CODEC_G723ar53: 0x4000 • CC_CAP_CODEC_G729: 0x8000 • CC_CAP_CODEC_GSMEFR: 0x40000 • CC_CAP_CODEC_T38FAX: 0x10000

Field	Description
loadable codec bitmap	Loadable codec bitmap for the loadable codecs. Values are as follows: • CC_CAP_CODEC_G711U: 0x0001 • CC_CAP_CODEC_G711A: 0x0002 • CC_CAP_CODEC_G729IETF: 0x0004 • CC_CAP_CODEC_G729a: 0x0008 • CC_CAP_CODEC_G726r16: 0x0010 • CC_CAP_CODEC_G726r24: 0x0020 • CC_CAP_CODEC_G726r32: 0x0040 • CC_CAP_CODEC_G728: 0x0080 • CC_CAP_CODEC_G723r63: 0x0100 • CC_CAP_CODEC_G723r53: 0x0200 • CC_CAP_CODEC_GSM: 0x0400 • CC_CAP_CODEC_G729b: 0x0800 • CC_CAP_CODEC_G729: = 0x1000 • CC_CAP_CODEC_G723ar63: 0x2000 • CC_CAP_CODEC_G723ar53: 0x4000 • CC_CAP_CODEC_G729: 0x8000 • CC_CAP_CODEC_GSMEFR: 0x40000 • CC_CAP_CODEC_T38FAX: 0x10000
fax codec bitmap	Fax codec bitmap. Values are as follows: • FAX_NONE = 0x1 • FAX_VOICE = 0x2 • FAX_144 = 0x80 • FAX_120 = 0x40 • FAX_96 = 0x20 • FAX_72 = 0x10 • FAX_48 = 0x08 • FAX_24 = 0x04
Logical Device (Tag)	Tag number or DSP number on the VFC.
Module#	Number identifying the module associated with a specific logical device.

Field	Description
DSP#	Number identifying the DSP on the VFC.
C_Ac	Number of active calls on the identified DSP.
C_Busy	Number of busied-out channels associated with the identified DSP.
C_Rst	Number of channels in the reset state associated with the identified DSP.
C_Bad	Number of defective ("bad") channels associated with the identified DSP.
Total active call channels	Total number of active calls.
Total busied out channels	Total number of busied-out channels.
Total channels in reset	Total number of channels in the reset state.
Total bad channels	Total number of defective channels.

Related Commands

Command	Description
show vrm active_calls	Displays active-only voice calls either for a specific VFC or for all VFCs.

show vsp

To display cumulative information about voice streaming processing (VSP) sessions, use the **show vsp** command in privileged EXEC mode.

show vsp {**all** | **debug** | **session** | **statistics**}

Syntax Description

all	Displays all available information on VSP sessions, including the information specified by the other keywords listed in this table.
debug	Displays the type of debugging information that is enabled by using the **debug vsp** command.
session	Displays cumulative statistics about active VSP sessions.
statistics	Displays statistics about active VSP sessions, including memory statistics.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(11)T	This command was introduced on the Cisco 3640, Cisco 3660, Cisco AS5300, Cisco AS5350, and Cisco AS5400.

Usage Guidelines

Use the **clear vsp statistics** command to reset the counters to 0 for the **show vsp** command.

Examples

The following is sample output from the **show vsp debug** command:

```
Router# show vsp debug
         VSP:<1>[0x62291660](0x62291660) debug_flag=0x7FF
```

The following is sample output from the **show vsp session** command:

```
Router# show vsp session
VSP_STATS:Session Statistics -
        sessions total=0; max_active=0, current=0
        session_duration last=0; max=0, min=0 ms
        pre_stream_wait last=0; max=0, min=0 ms
        stream_duration last=0; max=0, min=0 ms
        post_stream_wait last=0; max=0, min=0 ms
        stream_size last=0; max=0, min=0 bytes
        streaming_rate last=0; max=0, min=0 bytes/sec
        total_packet_count last=0; max=0, min=0 packets
        drop_packet_count last=0; max=0, min=0 packets
        particle_packet_count last=0; max=0, min=0 packets
```

The following is sample output from the **show vsp statistics** command:

```
Router# show vsp statistics
VSP_STATS:Session Statistics -
        sessions total=0; max_active=0, current=0
        session_duration last=0; max=0, min=0 ms
```

```
        pre_stream_wait last=0; max=0, min=0 ms
        stream_duration last=0; max=0, min=0 ms
        post_stream_wait last=0; max=0, min=0 ms
        stream_size last=0; max=0, min=0 bytes
        streaming_rate last=0; max=0, min=0 bytes/sec
        total_packet_count last=0; max=0, min=0 packets
        drop_packet_count last=0; max=0, min=0 packets
        particle_packet_count last=0; max=0, min=0 packets
VSP_STATS: Format Statistics -
        au_format_count=20
        wav_format_count=3
        other_format_count=0
VSP_STATS: Codec Statistics -
        codec_g729_count=4
        codec_g726_count=10
        codec_g711_count=0
        codec_g728_count=2
        codec_g723_count=5
        codec_gsm_count=2
        codec_other_count=0
VSP_STATS: Media Statistics -
        ram_count=23
        http_count=0
        smtp_count=0
        rtsp_count=0
        other_count=0
VSP_STATS:RTP Statistics -
        ts_gap_samples max=76800, min=80 samples
        [Unexpected SSRC Change (USC)]
                usc_count last=0; total=0, max=0, min=0
        [Out of sequence packet (OOSP)]
                oosp_count last=0; total=0, max=0, min=0
        [Unexpected timestamp gap (UTG)]
                max_utg_count last=0; total=0, max=0, min=0
        [Comfort Noise (CN)]
                max_cn_count last=4; total=70, max=8, min=4
        [Unexpected payload type or size (UPTS)]
                upt_count last=0; total=0, max=0, min=0; last_type=0
                ups_count last=0; total=198, max=61, min=0; last_size=2 bytes
        [Data exceeds limit (DEL)]
                del_count last=0; total=2, max=1, min=0
        [Silence exceeds timeout (SET)]
                set_count last=0; total=0, max=0, min=0
VSP_STATS:Packet Statistics -
        [Silence patching total (SPT)]
                spt_count last=296; total=7230, max=889, min=290
        [Concealment patching total (CPT)]
                cpt_count last=0; total=34, max=18, min=0
        [Normal patching total (NPT)]
                npt_count last=171; total=4249, max=453, min=106
```

The table below describes the fields shown in this output.

Table 21: show vsp statistics Field Descriptions

Field	Description
Session Statistics	
sessions total; max_active, current	Total number of VSP sessions since router startup or since the **clear vsp statistics** command was used. The active value should always be 0.

Field	Description
session_duration last; max, min	Duration of the last (most recent) session, and of the longest and shortest sessions in msecs.
pre_stream_wait last; max, min	Msecs that elapsed before the arrival of the first packet. Values are shown for last session, and for the session with the longest and shortest waits.
stream_duration last; max, min	Msecs between first packet arrival and last packet flush. Values are shown for last session, and for the session with the longest and shortest durations.
post_stream_wait last; max, min	Msecs between last packet flush and close of session.
stream_size last; max, min	Data streaming size.
streaming_rate last; max, min	Data streaming rate.
total_packet_count last; max, min	Total packets processed.
drop_packet_count last; max, min	Total packets dropped. The difference between the total packet count and packets dropped is the number of packets that have been accepted.
particle_packet_count last; max, min	Total particle packets processed.
Format Statistics	
au_format_count	Number of VSP sessions that used audio files in .au format.
wav_format_count	Number of VSP sessions that used audio files in .wav format.
other_format_count	Number of VSP sessions that used audio files of an unknown format.
Codec Statistics	
codec_g729_count	Number of VSP sessions that used the G.729 codec.
codec_g726_count	Number of VSP sessions that used the G.726 codec.
codec_g711_count	Number of VSP sessions that used the G.711 codec.
codec_g728_count	Number of VSP sessions that used the G.728 codec.
codec_g723_count	Number of VSP sessions that used the G.723 codec.
codec_gsm_count	Number of VSP sessions that used the GSM codec.
codec_other_count	Number of VSP sessions that used an unknown codec.
Media Statistics	
ram_count	Total number of RAM recordings and playouts.

Field	Description
http_count	Total number of HTTP recordings and playouts.
smtp_count	Total number of SMTP recordings.
rtsp_count	Total number of RTSP recordings and playouts.
other_count	Should always be 0.
RTP Statistics	
ts_gap_samples max min	Permissible timestamp gap in samples.
[Unexpected SSRC Change (USC)]	
usc_count last; total, max, min	Number of times that the source of the streaming has changed.
[Out of sequence packet (OOSP)]	
oosp_count last; total, max, min	Number of out-of-sequence packets.
[Unexpected timestamp gap (UTG)]	
max_utg_count last; total, max, min	Number of packets with an unexpected timestamp gap.
[Unexpected payload type or size (UPTS)]	
upt_count last; total, max, min; last_type	Number of comfort noise packets.
ups_count last; total, max, min; last_size	Number of packets with unexpected nonvoice payload sizes.
[Data exceeds limit (DEL)]	
del_count last; total, max, min	Number of times that the total recording size is larger than the preset recording size.
[Silence exceeds timeout (SET)]	
set_count last; total, max, min	Number of times that the timestamp gap is larger than the preset timeout value.
Packet Statistics	
[Silence patching total (SPT)]	
spt_count last; total, max, min	Number of silence packets that have been inserted during recording.
[Concealment patching total (CPT)]	
cpt_count last; total, max, min	Number of concealment packets that have been inserted during recording.
[Normal patching total (NPT)]	

Field	Description
npt_count last; total, max, min	Number of normal packets that have been patched during recording.

Related Commands

Command	Description
clear vsp statistics	Clears the statistics for VSP sessions.

show wsapi

To display information on the Cisco Unified Communication IOS services, including registration, statistics, and route information, use the **show wsapi** command in user EXEC or privileged EXEC mode.

show wsapi{**http-client** | **http-server** | **registration** | **registration**{**all** | **xcc** | **xcdr** | **xsvc**} | **svcc route**}

Syntax Description

http-client	Displays the statistics that have been collected on the http client interface.
http-server	Displays the statistics that have been collected on the http server interface.
registration	Displays the currently registered applications on the WSAPI subsystem.
all	Displays all registered applications.
xcc	Displays the applications that are registered to the XCC provider.
xcdr	Displays the applications that are registered to the XCDR provider.
xsvc	Displays the applications that are registered to the XSVC provider.
xsvc route	Displays the internal route information in the XSVC provider.

Command Modes

User EXEC
Privileged EXEC

Command History

Release	Modification
15.2(2)T	This command was introduced.

Usage Guidelines

Use this command to display information on the Cisco Unified Communication IOS services.

Examples

The following example shows a sample output from the **show wsapi http-client** command.

```
Router# show wsapi http-client

WSAPI Outgoing Notify/Solicit Message Statistics
=========================================
wsapi_show_httpc_callback_context_invalid: 0
wsapi_show_httpc_callback_context_error: 0
wsapi_show_httpc_callback_no_reg: 5
wsapi_show_httpc_callback_notify_OK: 85
wsapi_show_httpc_callback_notify_error: 0
wsapi_show_httpc_callback_client_error: 0
wsapi_show_httpc_callback_error: 7
wsapi_show_httpc_callback_client_error: 0
wsapi_show_httpc_callback_decode_error: 28
wsapi_show_httpc_callback_no_txID: 0
wsapi_show_httpc_callback_OK: 655
wsapi_show_httpc_create_msg_error: 0
```

```
wsapi_show_httpc_context_active: 0
wsapi_tx_context_freeq depth: 4
```

The following example shows a sample output from the **show wsapi http-server** command.

```
Router# show wsapi http-server

WSAPI Incoming Request Message Statistics
=========================================
wsapi_show_https_urlhook: 23
wsapi_show_https_post_action: 23
wsapi_show_https_post_action_fail: 0
wsapi_show_https_xml_fault: 0
wsapi_show_https_post_action_done: 23
wsapi_show_https_service_timeout: 0
wsapi_show_https_send_error: 0
wsapi_show_https_invalid_context: 0
wsapi_show_https_data_active: 0
wsapi_https_data_q depth: 1
wsapi_show_https_internal_service_error: 0
wsapi_show_https_service_unavailable_503: 0
wsapi_show_https_not_found_404: 0
wsapi_show_https_registration_success: 9
wsapi_show_https_not_registered: 0
wsapi_show_https_registration_auth_fail: 1
wsapi_show_https_registration_fail: 0
wsapi_show_https_un_registered: 0
```

The following example shows a sample output from the **show wsapi registration** command.

```
Router# show wsapi registration

Provider XCC
=====================================================
registration
id: 4FA11CC:XCC:myapp:5
appUrl:http://sj22lab-as2:8090/xcc
appName: myapp
provUrl: http://10.1.1.1:8090/cisco_xcc
prober state: STEADY
connEventsFilter:
CREATED|AUTHORIZE_CALL|ADDRESS_ANALYZE|REDIRECTED|ALERTING|CONNECTED|TRANSFERRED|CALL_DELIVERY|DISCONNECTED|HANDOFF_JOIN|HANDOFF_LEAVE
mediaEventsFilter:
DTMF|MEDIA_ACTIVITY|MODE_CHANGE||TONE_DIAL|TONE_OUT_OF_SERVICE|TONE_RINGBACK|TONE_SECOND_DIAL
blockingEventTimeoutSec: 1
blockingTimeoutHandle: CONTINUE_PROCESSING

Provider XSVC
=====================================================
registration index: 2
id: 4FA0F8C:XSVC:myapp:3
appUrl:http://sj22lab-as2:8090/xsvc
appName: myapp
provUrl: http://10.1.1.1:8090/cisco_xsvc
prober state: STEADY
route filter:
event filter: off

Provider XCDR
=====================================================
registration index: 1
id: 4FA10A0:XCDR:myapp:1
appUrl:http://sj22lab-as2:8090/xcdr
```

```
appName: myapp
provUrl: http://10.1.1.1:8090/cisco_xcdr
prober state: STEADY
cdr format: COMPACT
event filter: off
```

The following example shows a sample output from the **show wsapi xsvc route** command.

```
Router# show wsapi xsvc route

Route SANJOSE_SIP
=====================================================
Type: VOIP
Description: OUT
Filter:
Trunk:
Trunk Name: 1.3.45.2
Trunk Type: SIPV2
Trunk Status: UP
Route SANJOSE_PRI
=====================================================
Type: PSTN
Description: IN
Filter:
Trunk:
Trunk Name: Se0/1/0:23
Trunk Type: ISDN PRI
Trunk Status: UP
Total channels 2
Channel bitmap 0x01FFFFFE 1-24
Link bitmap 0x00000006
Alarm 0x00000001
Time elapsed 516
Interval 92
CurrentData
0 Line Code Violations, 0 Path Code Violations
0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
TotalData
49 Line Code Violations, 7 Path Code Violations,
0 Slip Secs, 1 Fr Loss Secs, 1 Line Err Secs, 0 Degraded Mins,
0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 2 Unavail Secs
Trunk Name: Se0/1/1:23
Trunk Type: ISDN PRI
Trunk Status: UP
Total channels 2
Channel bitmap 0x01FFFFFE 1-24
Link bitmap 0x00000006
Alarm 0x00000001
Time elapsed 516
Interval 92
CurrentData
0 Line Code Violations, 0 Path Code Violations
0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
TotalData
42 Line Code Violations, 4 Path Code Violations,
0 Slip Secs, 1 Fr Loss Secs, 1 Line Err Secs, 0 Degraded Mins,
0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 2 Unavail Secs
```

Related Commands

Command	Description
provider	Enables a Cisco Unified Communicatoins IOS service provider.

show xcsp port

To display the status of a router port under the control of the external control service provider (XCSP) subsystem, use the **show xcsp port**command in privileged EXEC mode.

show **xcsp** **port** *slot-num* *port-num*

Syntax Description

slot -num	Slot number of the interface card. Values are as follows: • Cisco AS5350: From 0 to 3. • Cisco AS5400: From 0 to 7. • Cisco AS5850: From 0 to 5 and from 8 to 13. Slots 6 and 7 are reserved for the route switch controller (RSC).
port -num	Port number of the interface card. Values are as follows: • Cisco AS5350: For T1/E1, from 0 to 7. For T3, from 1 to 28. • Cisco AS5400: For T1/E1, from 0 to 7. For T3, from 1 to 28. • Cisco AS5850: For T1/E1, from 0 to 23. For T3, from 1 to 28.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XB	This command was introduced.
12.2(11)T	The command was integrated into Cisco IOS Release 12.2(11)T and implemented on the Cisco AS5850.

Examples

The following is sample output from this command:

```
Router# show xcsp port 1 0
Slot 1 configured
Number of ports configured=1 slot state= Up
====================================================
Port 0 State= Up type = 5850 24 port T1
Channel states
  0  Idle
  1  Idle
  2  Idle
  3  Idle
  4  Idle
  .
  .
  .
  22  Idle
  23  Idle
```

The table below describes significant fields in this output.

Note To get the field description output, you must enter the *slot-num* and *port-num* arguments for the **show xcsp port** command.

Table 22: show xcsp port Field Descriptions

Field	Descriptions
Port	Port number. Range is from 1 to 28.
State	Port state; can be Up or Down.
type	T1 or E1 ports on the AS5400: 8. T1 or E1 ports on the AS5850: 24. T3 ports on the AS5400 and AS5850: 28.
Channel states	Channel states. Values are as follows: • Blocked • Connection in progress • Cot Check In Progress • Cot Check Pending • Down • Idle • In Release in progress • In Use • Invalid • Loopback • Not Present • Out of Service • Out Release in progress • Playing Tone • Shutdown

Related Commands

Command	Description
show xcsp slot	Displays the status of XCSP slots.

show xcsp slot

To display the status of a router slot under the control of the external control service provider (XCSP) subsystem, use the **show xcsp slot** command in privileged EXEC mode.

show xcsp slot *slot-num*

Syntax Description

slot -num	The slot number of the T1 or E1 interface card. Values are as follows: • Cisco AS5350: From 0 to 3. • Cisco AS5400: From 1 to 7. • Cisco AS5850: From 0 to 5 and from 8 to 13. Slots 6 and 7 are reserved for the route switch controller (RSC).

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XB	This command was introduced.
12.2(11)T	The command was integrated into Cisco IOS Release 12.2(11)T and implemented on the Cisco AS5850.

Examples

The following is sample output from this command:

```
Router# show xcsp slot 1
Slot 1 configured
Number of ports configured=1 slot state= Up
```

The table below describes significant fields shown in this output.

Table 23: show xcsp slot Field Descriptions

Field	Description
slot state	Slot state; can be either Up or Down.

Related Commands

Command	Description
show xcsp port	Displays the status of XCSP ports.

shut

To shut down a set of digital signal processors (DSPs) on the Cisco 7200 series router, use the **shut** command in DSP configuration mode. To put DSPs back in service, use the **no** form of this command.

shut *number*
no shut *number*

Syntax Description

number	Number of DSPs to be shut down.

Command Default

No shut

Command Modes

DSP configuration

Command History

Release	Modification
12.0(5)XE	This command was introduced on the Cisco 7200 series.
12.1(1)T	This command was modified to add information about DSP groups.

Usage Guidelines

This command applies to VoIP on the Cisco 7200 series routers.

Examples

The following example shuts down two sets of DSPs:

```
shut 2
```

shutdown (Annex G neighbor)

To disable the service relationships requirement for border elements, use the **shutdown** command in config-nxg-neigh-srvc mode. To enable the service relationship for border elements, use the **no**form of this command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

The Annex G neighbor is shut down.

Command Modes

Annex G neighbor service (config-nxg-neigh-svc)

Command History

Release	Modification
12.2(11)T	This command was introduced.

Usage Guidelines

The **no shutdown** command verifies that a domain name has been configured and ensures that the border element has been configured to reject messages from unknown "stranger" border elements.

Examples

The following example enables the border element:

Router(config-nxg-neigh-srvc)# **no shutdown**

Related Commands

Command	Description
access -policy	Requires that a neighbor be explicitly configured.
inbound ttl	Sets the inbound time-to-live value.
outbound retry -interval	Defines the retry period for attempting to establish the outbound relationship between border elements.
retry interval	Defines the time between delivery attempts.
retry window	Defines the total time that a border element attempts delivery.

shutdown (Annex G)

To shut down the Annex G border element (BE), use the **shutdown** command in Annex G configuration mode. To reinstate the Annex G BE, use the no form of this command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

The Annex G border element is not shut down.

Command Modes

Annex G configuration (config-annexg)

Command History

Release	Modification
12.2(2)XA	This command was introduced.
12.2(4)T	This command was integrated into Cisco IOS Release 12.2(4)T. This command was not supported on the Cisco AS5300, Cisco AS5350, and Cisco AS5400 in this release.
12.2(2)XB1	This command was implemented on the Cisco AS5850.
12.2(11)T	This command was integrated into Cisco IOS Release 12.2(11)T.

Usage Guidelines

While the Annex G BE is in shutdown state, all Annex G messages received from neighbors are ignored and the colocated gatekeeper does not use the Annex G BE for address resolution.

Examples

The following example shuts the BE down:

```
Router(config)# call-router h323-annexg be20
Router(config-annexg)# shutdown
```

Related Commands

Command	Description
call -router	Enables the Annex G border element configuration commands.
show call -router status	Displays the Annex G BE status.

shutdown (dial-peer)

To change the administrative state of the selected dial peer from up to down, use the **shutdown** command in dial-peer configuration mode. To change the administrative state of this dial peer from down to up, use the **no** form of this command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

No shutdown

Command Modes

Dial-peer configuration (config-dial-peer)

Command History

Release	Modification
11.3(1)T	This command was introduced on the Cisco 3600 series.
12.1(1)	This command was modified for store-and-forward fax.

Usage Guidelines

When a dial peer is shut down, you cannot initiate calls to that peer.

This command applies to both on-ramp and off-ramp store-and-forward fax functions.

Examples

The following example changes the administrative state of voice telephony (plain old telephone service [POTS]) dial peer 10 to down:

```
dial-peer voice 10 pots
shutdown
```

The following example changes the administrative state of voice telephony (POTS) dial peer 10 to up:

```
dial-peer voice 10 pots
no shutdown
```

Related Commands

Command	Description
dial -peer voice	Enters dial-peer configuration mode, defines the type of dial peer, and defines the dial-peer tag number.

shutdown (DSP Farm profile)

To disable the digital signal processor (DSP) farm profile, use the **shutdown** command in DSP farm profile configuration mode. To allocate DSP farm resources and associate with the application, use the **no** form of this command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

DSP farm profile configuration (config-dspfarm-profile)

Command History

Release	Modification
12.3(8)T	This command was introduced.

Usage Guidelines

It is essential that the profile be disabled by using the **shutdown** command before a DSP farm profile is updated.

Examples

The following example allocates DSP farm resources and associates with the application:

```
Router(config-dspfarm-profile)#
 no shutdown
```

Related Commands

Command	Description
codec (dspfarm-profile)	Specifies the codecs supported by a DSP farm profile.
description (dspfarm-profile)	Includes a specific description about the DSP farm profile.
dspfarm profile	Enters the DSP farm profile configuration mode and defines a profile for DSP farm services.
maximum sessions (dspfarm-profile)	Specifies the maximum number of sessions that need to be supported by the profile.

shutdown (gatekeeper)

To disable the gatekeeper, use the **shutdown** command in gatekeeper configuration mode. To enable the gatekeeper, use the **no** form of this command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled (shut down)

Command Modes

Gatekeeper configuration (config-gk)

Command History

Release	Modification
11.3(2)NA	This command was introduced on the Cisco 2500 series and Cisco 3600 series.
12.0(3)T	The command was integrated into Cisco IOS Release 12.0(3)T and implemented on the Cisco MC3810.

Usage Guidelines

The gatekeeper does not have to be enabled before you can use the other gatekeeper configuration commands. In fact, it is recommended that you complete the gatekeeper configuration before bringing up the gatekeeper because some characteristics may be difficult to alter while the gatekeeper is running, as there may be active registrations or calls.

The no shutdown command enables the gatekeeper, but it does not make the gatekeeper operational. The two exceptions to this are as follows:

- If no local zones are configured, a **no shutdown** command places the gatekeeper in INACTIVE mode waiting for a local zone definition.
- If local zones are defined to use an HSRP virtual address, and the HSRP interface is in STANDBY mode, the gatekeeper goes into HSRP STANDBY mode. Only when the HSRP interface is ACTIVE does the gatekeeper go into the operational UP mode.

Examples

The following command disables a gatekeeper:

```
shutdown
```

Related Commands

Command	Description
shutdown (gateway)	Shuts down all VoIP call service on a gateway.

shutdown (gateway)

To shut down all VoIP call service on a gateway, use the **shutdown**commandin voice service configuration mode. To enable VoIP call service, use the **no** form of this command.

shutdown [**forced**]
no shutdown

Syntax Description

forced	(Optional) Forces the gateway to immediately terminate all in-progress calls.

Command Default

Call service is enabled

Command Modes

Voice service configuration (config-voi-serv)

Command History

Release	Modification
12.3(1)	This command was introduced.
Cisco IOS XE Amsterdam 17.2.1r	Introduced support for YANG models.

Examples

The following example shows VoIP call service being shut down on a Cisco gateway:

```
voice service voip
shutdown
```

The following example shows VoIP call service being enabled on a Cisco gateway:

```
voice service voip
no shutdown
```

Related Commands

Command	Description
shutdown (gatekeeper)	Disables the gatekeeper.

shutdown (mediacard)

To disable a selected media card, use the **shutdown** command in mediacard configuration mode. To enable a selected media card, use the **no** form of this command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Media card configuration

Command History

Release	Modification
12.3(8)XY	This command was introduced on the Communication Media Module.
12.3(14)T	This command was integrated into Cisco IOS Release 12.3(14)T.
12.4(3)	This command was integrated into Cisco IOS Release 12.4(3).

Usage Guidelines

Use the **no shutdown** command at the end of media card configuration. If there are any active connections when you disable the media card, the Digital Signal Processor Resource Manager (DSPRM) displays a warning message indicating that the DSP resources allocated on other media cards for some of the resource pool in this media card will be removed or that there are active connections available in this resource pool and prompts you for a response. Profiles that use resources on this card must be brought up separately after using this command.

Examples

The following example shows how to enable a media card:

```
no shutdown
```

Related Commands

Command	Description
resource-pool	Creates a DSP resource pool on the selected media card.

shutdown (auto-config application)

To disable an auto-configuration application for download, use the **shutdown** command in auto-config application configuration mode. To enable an auto-configuration application for download, use the **no**form of this command.

shutdown
no shutdown

Syntax Description

This command has no keywords or arguments.

Command Default

Disabled

Command Modes

Auto-config application configuration (auto-config-app)

Command History

Release	Modification
12.3(8)XY	This command was introduced on the Communication Media Module.
12.3(14)T	This command was integrated into Cisco IOS Release 12.3(14)T.

Examples

The following example shows the **shutdown** command used to enable an auto-configuration application for download:

```
Router(auto-config-app)# no shutdown
```

Related Commands

Command	Description
auto-config	Enables auto-configuration or enters auto-config application configuration mode for the SCCP application.
show auto-config	Displays the current status of auto-configuration applications.

shutdown (RLM)

To shut down all of the links under the RLM group, use the **shutdown** command in RLM configuration mode. RLM does not try to reestablish those links until the command is negated. To disable this function, use the **no** form of this command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

RLM configuration

Command History

Release	Modification
11.3(7)	This command was introduced.

Related Commands

Command	Description
clear interface	Resets the hardware logic on an interface.
clear rlm group	Clears all RLM group time stamps to zero.
interface	Defines the IP addresses of the server, configures an interface type, and enters interface configuration mode.
link (RLM)	Specifies the link preference.
protocol rlm port	Reconfigures the port number for the basic RLM connection for the whole rlm-group.
retry keepalive	Allows consecutive keepalive failures a certain amount of time before the link is declared down.
server (RLM)	Defines the IP addresses of the server.
show rlm group statistics	Displays the network latency of the RLM group.
show rlm group status	Displays the status of the RLM group.
show rlm group timer	Displays the current RLM group timer values.
timer	Overwrites the default setting of timeout values.

shutdown (settlement)

To deactivate the settlement provider, use the shutdown command in settlement configuration mode. To activate a settlement provider, use the no **shutdown** command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

The default status of a settlement provider is deactivated. The settlement provider is down.

Command Modes

Settlement configuration

Command History

Release	Modification
12.0(4)XH1	This command was introduced on the Cisco 2500 series, Cisco 3600 series, and Cisco AS5300.
12.1(1)T	This command was integrated into Cisco IOS Release 12.1(1)T.

Usage Guidelines

Use this command at the end of the configuration of a settlement server to bring up the provider. This command activates the provider. Otherwise, transactions do not go through the provider to be audited and charged. Use the shutdown command to deactivate the provider.

Examples

The following example enables a settlement server:

```
settlement 0
 no shutdown
```

The following example disables a settlement server:

```
settlement 0
 shutdown
```

Related Commands

Command	Description
connection -timeout	Configures the time that a connection is maintained after completing a communication exchange.
customer -id	Identifies a carrier or ISP with a settlement provider.
device -id	Specifies a gateway associated with a settlement provider.
encryption	Sets the encryption method to be negotiated with the provider.
max -connection	Sets the maximum number of simultaneous connections to be used for communication with a settlement provider.
response -timeout	Configures the maximum time to wait for a response from a server.

Command	Description
retry -delay	Sets the time between attempts to connect with the settlement provider.
session -timeout	Sets the interval for closing the connection when there is no input or output traffic.
settlement	Enters settlement configuration mode and specifies the attributes specific to a settlement provider.
type	Configures an SAA-RTR operation type.

shutdown (trace)

To disable the VoIP Trace framework in CUBE, use the **shutdown** command in trace configuration mode. To re-enable VoIP tracing, use the **no** form of this command.

Syntax Description

shutdown	Disables the VoIP Trace framework.
[no] shutdown	Enables the VoIP Trace framework.

Command Default

VoIP Trace is enabled by default.

Command Modes

Trace configuration mode (conf-serv-trace)

Command History

Release	Modification
Cisco IOS XE Amsterdam 17.3.2 Cisco IOS XE Bengaluru 17.4.1a	This command was introduced for Cisco Unified Border Element.

Usage Guidelines

VoIP Trace is disabled using the command **shutdown** under the **trace** configuration mode as follows:

```
router (config)#voice service voip
router(conf-voi-serv)#trace
router(conf-serv-trace)#?
Voip Trace submode commands:
default      Set a command to its defaults
exit         Exit from voice service voip trace mode
no           Negate a command or set its defaults
shutdown     Shut Voip Trace debugging
memory-limit Set limit based on memory used
router(conf-serv-trace)#shutdown
```

To re-enable VoIP Trace, configure the CLI command **no shutdown** under the **trace** configuration mode as follows:

```
router (config)#voice service voip
router(conf-voi-serv)#trace
router(conf-serv-trace)#no ?
exit     Exit from voice service voip trace mode
shutdown Shut Voip Trace debugging
router(conf-serv-trace)# no shutdown
```

If you configure **shutdown** :

- Tracing for active calls is stopped.
- All existing traces in memory are deleted.

Only new calls received after VoIP Trace is enabled are monitored.

Examples

The following is a sample of CLI command **shutdown** configured under trace configuration sub-mode:

```
router (config)#voice service voip
router(conf-voi-serv)#trace
router(conf-serv-trace)#shutdown
```

Related Commands

Command	Description
memory-limit (trace)	Defines the memory limit for storing VoIP Trace information.
trace	Enables the VoIP Trace serviceability framework in CUBE.
show voip trace	Displays the VoIP Trace information for SIP legs on a call received on CUBE

shutdown (voice-port)

To take the voice ports for a specific voice interface card offline, use the **shutdown** command in voice-port configuration mode. To put the ports back in service, use the **no** form of this command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Command Default

Shutdown

Command Modes

Voice-port configuration (config-voiceport)

Command History

Release	Modification
11.3(1)T	This command was introduced on the Cisco 3600 series.
12.4(22)T	Support for IPv6 was added.

Usage Guidelines

When you use this command, all ports on the voice interface card are disabled. When you use the **no**form of the command, all ports on the voice interface card become enabled. A telephone connected to an interface hears silence when a port is shut down.

Examples

The following example takes voice port 1/1/0 offline:

```
voice-port 1/1/0
shutdown
```

Related Commands

Command	Description
shutdown (port)	Disables a port.