

Understanding Call Tracker Outputs

[TAC Notice: What's Changing on TAC Web](#)

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

[Background Information](#)

[Benefits of Call Tracker](#)

[Call Tracker Configuration](#)

[Summary of Commands](#)

[Detailed Commands](#)

[Call Tracker Output](#)

[CALL_RECORD Parameters](#)

[MODEM_CALL_RECORD Parameters](#)

[MODEM_LINE_CALL_REC Parameters](#)

[MODEM_INFO_CALL_REC Parameters](#)

[MODEM_NEG_CALL_REC Parameters](#)

[Related SNMP MIBs](#)

[SNMP MIBs](#)

[CISCO-CALL-TRACKER-MIB](#)

[Cisco Support Community - Featured Conversations](#)

[Related Information](#)

Help us help you.



Please rate this document.

- Excellent
- Good
- Average
- Fair
- Poor



This document solved my problem.

- Yes
- No
- Just browsing



Suggestions for improvement:

(256 character limit)



Introduction

This document describes Call Tracker outputs. Call Tracker is a subsystem used to capture detailed data on the progress and status of calls, from the time the network access server receives a setup request or allocates a channel, until a call is rejected, terminated, or otherwise disconnected.

Prerequisites

Requirements

Before you configure Call Tracker and its associated features, you must complete these tasks on your network access

server:

- Configure ISDN and the modems. For more information, refer to [Configuring an Access Server with PRIs for Incoming Async and ISDN Calls](#).
- Ensure that calls can connect to the Network Access Server (NAS).
- Configure Simple Network Management Protocol (SNMP). For more information, refer to [Basic Dial NMS Implementation Guide](#).

Note: This task is required only if you use Call Tracker through SNMP.

Components Used

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

- Cisco IOS® Software Release 12.1(3)T and later
- Cisco AS5300, AS5350, AS5400, AS5800, and AS5850 platforms.

Note: Use the [Software Advisor](#) ([registered](#) customers only) to verify whether the Cisco IOS software version and platform you use supports this feature. Within the Software Advisor tool, search for the feature named *Call Tracker plus ISDN and AAA Enhancements*.

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

Conventions

Refer to the [Cisco Technical Tips Conventions](#) for information about document conventions.

Background Information

The data captured in the Call Tracker is maintained within the Call Tracker database tables and is accessible through Simple Network Management Protocol (SNMP), Command-Line Interface (CLI), or SYSLOG. Session information for all active calls and calls in the setup state is stored in an active table, while records for disconnected calls are moved to a history table. Call Tracker is notified of applicable call events by related subsystems such as ISDN, Point-to-Point Protocol (PPP), Content Switch Module (CSM), Modem, Exec, or TCP-Clear. SNMP traps are generated at the start of each call when an entry is created in the active table and at the end of each call when an entry is created in the history table. Call Record SYSLOGs are available through configurations that generate detailed information records for all call terminations. This information can be sent to SYSLOG servers for permanent storage and future analysis.

Here are some points to remember:

- The status and diagnostic data that is routinely collected from MICA modems is expanded to include new link statistics for active calls, such as the attempted transmit and receive rates, the maximum and minimum transmit

and receive rates, and locally and remotely issued retrains and speedshift counters. This connection data is polled from the modem at user-defined intervals and passed to Call Tracker.

- The TCP system has been enhanced to provide additional connection information to Call Tracker. Additional information includes:
 - The number and identity of the hosts to which a connection attempt was made before the connection was established, or the total failed attempts if no connection was made.
 - The reason an active session is disconnected, or the reason the network access server failed to connect to a host before it timed out.
 - The active session source and destination endpoints, which consist of the IP addresses and port numbers of the network access server and host.

For more information about Call Tracker, see [Call Tracker plus ISDN and AAA Enhancements for the Cisco AS5300 and Cisco AS5800](#).

Benefits of Call Tracker

This section lists the benefits of Call Tracker.

- Call Tracker provides more comprehensive and straightforward real-time monitoring of call activity.
- Call Tracker captures data for active and historical call sessions and allows external applications to access that data through SNMP, CLI, or SYSLOG.
- Call Tracker provides volume and usage statistics for call management decisions.
- Call Tracker improves upon and replaces the **modem call-record terse** feature because it provides more detailed output.

Note: Because they can generate similar SYSLOG output, do not enable Call Tracker and **modem call-record terse** at the same time. This action can result in duplicate entries for the same call.

Call Tracker Configuration

Summary of Commands

To configure Call Tracker, use these commands (in the order that they are listed):

1. **enable**
2. **configure terminal**
3. **calltracker enable**

4. **calltracker call-record**
5. **calltracker history max-size**
6. **calltracker history retain-mins**
7. **snmp-server packetsize byte-count**
8. **snmp-server queue-length**
9. **snmp-server enable traps calltracker**
10. **snmp-server host host community-string calltracker**
11. **calltracker timestamp msec** (Optional)
12. **modem link-info poll time or spe link-info poll modem** (Optional)
13. **exit**

Detailed Commands

	Command	Purpose
Step 1.	enable Example: Router> enable	Enters privileged EXEC mode or any other security level set by a system administrator. Enter your password if prompted.
Step 2.	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3.	calltracker enable Example: Router(config)# calltracker enable	Enables Call Tracker on the NAS.

<p>Step 4.</p>	<p>calltracker call-record {terse verbose} [quiet]</p> <p>Example:</p> <pre>Router(config)# calltracker call- record verbose quiet</pre>	<p>The information provided can be gathered by SNMP and SYSLOG from the call history table of Call Tracker.</p> <p>The terse option generates a brief set of call records, which contains a subset of data stored within Call Tracker that is used primarily to manage calls.</p> <p>The verbose option generates a complete set of call records that contain all data stored within Call Tracker that is used primarily to debug calls.</p> <p>With the quiet option, the call record is sent only to the configured SYSLOG server and not to the console.</p>
<p>Step 5.</p>	<p>calltracker history max-size <i>number</i></p> <p>Example:</p> <pre>Router(config)#</pre>	<p>To configure the history buffer (the maximum number of call entries stored in the Call Tracker history table), use the calltracker history max-size <i>number</i> command.</p> <p><i>number</i> is the maximum number of call entries to store in the Call Tracker history table. The valid range is from zero through ten times the maximum DS0 supported on given platform. A value of 0 prevents any history from being saved.</p> <p>Because the reporting task is not a high priority process and because it must wait for available CPU, Call Tracker can take up to one minute to report after a call has disconnected. Therefore, you must configure the history buffer so that it is large enough to store the data that</p>

	<pre>calltracker history max-size 50</pre>	<p>will be reported.</p> <p>When you configure the buffer size, take into account the call length and type of call (ISDN is shorter than modem), and then determine the maximum number of calls that can be received over a one minute period. In addition, a higher call rate can occur when a configuration error or hardware failure occurs. Therefore, it is recommended that you use four times the number of ports on the platform.</p> <p>For more information, refer to Call Tracker plus ISDN and AAA Enhancements for the Cisco AS5300 and Cisco AS5800.</p>
<p>Step 6.</p>	<p>calltracker history retain- mins <i>minutes</i></p> <p>Example:</p> <pre>Router(config)# calltracker history retain- mins 5000</pre>	<p>Sets the number of minutes to store calls in the Call Tracker history table.</p> <p><i>minutes</i> is the length of time to store the calls. The valid range is from 0 through 26,000 minutes. A value of 0 prevents calls from being stored.</p>
<p>Step 7.</p>	<p>snmp-server packetsize <i>byte-count</i></p> <p>Example:</p> <pre>Router(config)# snmp-server packetsize 1024</pre>	<p>Establishes control over the largest Simple Network Management Protocol (SNMP) packet size permitted when the SNMP server receives a request or generates a reply.</p> <p><i>byte-count</i> is an integer from 484 to 8192. The default is 1500.</p>

<p>Step 8.</p>	<p>snmp-server queue-length <i>length</i></p> <p>Example:</p> <pre>Router(config)# snmp-server queue- length 50</pre>	<p>Defines the length of the message queue for each trap host. When a trap message is successfully transmitted, Cisco IOS software continues to empty the queue; however, it does not empty the queue faster than a rate of four trap messages per second.</p> <p>During device bootup, some traps can be dropped because of trap queue overflow on the device. If you think that traps are being dropped, you can increase the size of the trap queue (for example, to 100) to determine if traps can then be sent during bootup</p> <p><i>length</i> is an integer that specifies the number of trap events that can be held before the queue must be emptied. The default is 10.</p>
<p>Step 9.</p>	<p>snmp-server enable traps calltracker</p> <p>Example:</p> <pre>Router(config)# snmp-server enable traps</pre>	<p>SNMP notifications can be sent as traps or inform requests; this command enables both traps and inform requests.</p> <p>This command controls (enables or disables) Call Tracker CallSetup and CallTerminate notifications. CallSetup notifications are generated at the start of each call and when an entry is created in the active table (cctActiveTable). CallTerminate notifications are generated at the end of each call and when an entry is created in the history table (cctHistoryTable).</p>

<p>Step 10.</p>	<p>snmp-server host <i>host</i> <i>community-string</i> calltracker</p> <p>Example:</p> <pre>Router(config)# snmp-server host host community string calltracker</pre>	<p>Specifies the recipient of an Simple Network Management Protocol notification operation.</p> <p>SNMP notifications can be sent as traps or inform requests. Traps are unreliable because the receiver does not send acknowledgments when it receives traps. The sender cannot determine if the traps were received. However, an SNMP entity that receives an inform request acknowledges the message with a SNMP response protocol data unit (PDU). If the sender never receives the response, the inform request can be sent again. Therefore, informs are more likely to reach their intended destination.</p> <p>Compared to traps, informs consume more resources in the agent and in the network. Unlike traps, which are discarded as soon as they are sent, an inform request must be held in memory until a response is received or the request times out. Also, traps are sent only once; an inform may be retried several times. The retries increase traffic and contribute to a higher overhead on the network.</p> <p>If you do not enter an snmp-server host command, no notifications are sent. To configure the router to send SNMP notifications, you must enter at least one snmp-server host command. If you enter the command with no keywords, all trap types are enabled for the host.</p>
------------------------	--	---

		<p>To enable multiple hosts, you must issue a separate snmp-server host command for each host. You can specify multiple notification types in the command for each host.</p> <p>When multiple snmp-server host commands are given for the same host, as well as the type of notification (trap or inform), each succeeding command overwrites the previous command. Only the last snmp-server host command is in effect. For example, if you enter an snmp-server host inform command for a host and then enter another snmp-server host inform command for the same host, the second command replaces the first.</p>
<p>Step 11.</p>	<p>calltracker timestamp msec (Optional)</p> <p>Example:</p> <pre>Router(config)# calltracker timestamp msec</pre>	<p>Displays the millisecond value of the call setup time in the Call Record (CDR) on the access server. If you do not execute this command, the call setup time is displayed in seconds.</p> <p>Note: You can use this command with Cisco IOS Releases 12.3(4) and 12.3(4) T only.</p>
	<p>modem link-info poll time seconds (Optional)</p> <p>or</p>	<p>Enables Call Tracker modem detail records.</p> <p>Optionally, you can use either the modem link-info poll time seconds command or the spe link-info poll modem seconds command. These commands set the polling interval at which link statistics for active calls are retrieved from the modem. The recommended poll time value is 320 seconds.</p>

Step 12.	spe link-info poll modem <i>seconds</i> (Optional) Example: <pre>Router(config)# modem link-info poll time 320</pre>	<p>To enable real-time call statistics from the MICA technologies modem to Call Tracker, you must use the modem link-info poll time command.</p> <p>Note: The modem link-info poll time command consumes a significant amount of memory, approximately 500 bytes for each MICA modem call. Use this command only if you require the specific data that it collects.</p>
Step 13.	exit Example: <pre>Router(config)# exit</pre>	<p>Exits the current mode.</p>

Call Tracker Output

The Call Tracker output is split between several records. This table lists and describes the Call Tracker output records.

Record Name	Description
CALL_RECORD	Generic data shared among all call categories. For a list of acceptable parameters, see CALL_RECORD Parameters .
MODEM_CALL_RECORD	Overall modem call information. For a list of acceptable parameters, see MODEM_CALL_RECORD Parameters .
MODEM_LINE_CALL_REC	Modem transport and physical layer information (for comprehensive debugging purposes). For a list of acceptable parameters, see MODEM_LINE_CALL_REC Parameters .

MODEM_INFO_CALL_REC	Modem status information (for comprehensive debugging purposes). For a list of acceptable parameters, see MODEM_INFO_CALL_REC Parameters .
MODEM_NEG_CALL_REC	Client and host negotiation information (for comprehensive debugging purposes). For a list of acceptable parameters, see MODEM_NEG_CALL_REC Parameters .

Note: Records that refer to the same call start with the same unique value in parameter **ct_hndl**.

CALL_RECORD Parameters

This table lists and describes the CALL_RECORD parameters.

Parameters	Description
ct_hndl	<p>Call Tracker Handle</p> <p>A unique number used by Call Tracker to handle active calls. The calls are assigned an identification (ID) number from 1 to 4,294,967,296. These IDs start with 1 and increment by 1. After 4,294,967,295 calls, the ID wraps, and the 4,294,967,296th call receives the next smallest available number that starts from 1. It is possible for the call history, syslog, and SNMP records to have the same ID number for different calls. This is because the number is only unique for active calls. Zero is not a valid value.</p>
	<p>Service Type</p> <p>Reports last known service type of call.</p> <ul style="list-style-type: none"> • none – No service associated with the call • other – Service active, but none of these: <ul style="list-style-type: none"> • slip – Serial line IP • ppp – PPP • mp – Multilink PPP (RFC 1990)

Service	<ul style="list-style-type: none"> ● tcpClear – Byte stream over TCP ● telnet – TELNET ● exec – Terminal server ● l2f – Virtual Private Data Network service (VPDN) that uses Layer 2 forwarding protocol ● l2tp – Virtual Private Data Network service (VPDN) that uses Layer 2 tunneling Protocol
Origin	<p>Indicates how the call was created.</p> <ul style="list-style-type: none"> ● originate – Dialout, the call was initiated locally, and the system sends the setup request. ● answer – Dialin, the call was initiated remotely, and the system receives the setup request.
Call Category	<p>Represents possible call categories or types.</p> <ul style="list-style-type: none"> ● none – No call category associated with call ● other – None of these: ● modem – Modem call ● isdn-sync – ISDN sync digital call Now mapped to syncData ● v110 – V110 Call ● v120 – V120 Call ● cas-digital – Channel Associated Signaling (CAS) 56k data call ● mgcpData – MGCP data call Now mapped to syncData

	<ul style="list-style-type: none"> • syncData – Sync digital data call for any call control • lapb-ta – LAPB or LAPB-TA call
DS0 slot/cntr/ chan	<p>Entry Slot/Port/DS0</p> <p>The DS0 link that contains the call. This may be a DS0 contained within a larger group of multiple DS0s within a single physical port.</p>
called	<p>Called Party Id</p> <p>The called telephone number for this call. For calls answered by the system, this corresponds to the Dialed Number Identification (DNIS). For calls originated by the system, this is the destination number. If not available this is a zero-length string.</p>
calling	<p>Calling Party ID</p> <p>The calling telephone number for this call. For calls answered by the system, this corresponds to the calling identification (CLID). For calls originated by the system, this is the number associated with the device. For the interworking call, this is the translated calling party number, if there is a translation rule for outgoing calls associated with the dial plan. If not available, this is a zero-length string.</p>
resource slot/port	<p>Resource Slot/Port</p> <p>Identification of the processing resource allocated to the call.</p>
userid	<p>Username ID</p> <p>The user login ID or zero-length string if unavailable. If this contains a non-zero length string, and cctHistoryUserValidationTime is zero, then the user failed validation</p>
ip	<p>IP Address</p> <p>The IP address assigned for this call, or 0.0.0.0 if not applicable or unavailable.</p>

mask	<p>IP Subnet Mask</p> <p>The IP subnet mask assigned for this call, or 0.0.0.0 if not applicable or unavailable.</p>
account id	<p>Accounting Session ID</p> <p>Accounting session identification assigned to this call by AAA. The session ID is sent by AAA to RADIUS as the Acct-Session-Id attribute or TACACS+ as the task_id. If no accounting session ID is assigned, the value is a null string.</p>
setup	<p>Setup Time</p> <p>Timestamp when the call was first made known to the system.</p>
conn	<p>Connection Time</p> <p>Time in seconds it took for the call to connect.</p>
phys	<p>Physical Layer Ready</p> <p>Time in seconds it took for the physical layer to achieve a steady state and the call is ready for higher protocol layers to begin. In the case of modem calls, the physical layer for the call achieves a steady state when the data rates, modulations, and error correcting protocols have been negotiated between the originating and answering modems. It also applies to digital calls that use adaptive rate technologies, such as V.110 and V.120.</p>
srvc	<p>Service Time</p> <p>The time it took to identify the service type.</p>
auth	<p>Authentication Time</p> <p>Time in seconds it took to validate the user identification that is associated with this call.</p>

init rx/tx b-rate	<p>Initial Receive/Transmit Bit Rate</p> <p>Initial receive and transmit data rate for this call. If the call is a synchronous digital call such as ISDN sync, this value is the data rate of the B-channel. If call is asynchronous, even if it uses a synchronous transmission medium such as ISDN, the value is the speed negotiated by the MICA or Nextport modem in bits per second. This value does not change, even if the data rate varies during the call. This value is zero until an initial data rate is determined.</p>
rx/tx chars	<p>Transmit/Receive Byte</p> <p>The number of bytes transmitted on the call. All the raw bytes are counted. This value includes any protocol headers that may or may not be present. Whether or not the protocol header is present depends on the value of service.</p>
time	<p>Time Connected</p> <p>The time in seconds the call is connected. This is the call duration in seconds from initial setup request to when the system initiates, detects, or is notified of call termination.</p>
disc subsys	<p>Disconnect Subsystem</p> <p>IOS subsystem that initiates, detects, or is notified of call termination.</p> <p>Subsystem types:</p> <ul style="list-style-type: none"> ● admin ● csm ● isdn mica ● none ● ppp ● rpm (Resource Pool Management) ● vpn (Virtual Private Network) ● vtsp (Voice Telephony)

	<p>Note: Although this information requires more knowledge of the Cisco IOS software than the average user possesses, it is useful to Cisco Technical Support personnel for troubleshooting connection issues.</p>
disc code	<p>Disconnect Cause Code</p> <p>Code that indicate the reason this call was terminated. For more information, refer to these documents:</p> <ul style="list-style-type: none"> • Interpreting NextPort Disconnect Reason Codes • MICA Modem States and Disconnect Reasons
disc text	<p>Disconnect Description</p> <p>Text that describes the disconnect reason provided. This may be a zero-length string if no text is available.</p> <p>For more information, refer to these documents:</p> <ul style="list-style-type: none"> • Interpreting NextPort Disconnect Reason Codes • MICA Modem States and Disconnect Reasons

Example

```
*Nov 16 18:30:26.097: %CALLTRKR-3-CALL_RECORD:
  ct_hndl=5, service=PPP, origin=Answer, category=Modem,
  DS0 slot/cntr/chan=0/0/22, called=71071, calling=6669999,
  resource slot/port=1/0, userid=maverick5200, ip=192.9.1.2,
  mask=255.255.255.0, account id=5, setup=10/16/1999 18:29:20,
  conn=0.10, phys=17.12, srvc=23.16, auth=23.16, init-rx/tx
  b-rate=31200/33600, rx/tx chars=246/161, time=53.50, disc
  subsys=ModemDrvr, disc code=0xA220, disc text= Rx (line to host)
  data flushing - not OK/EC condition - locally detected/received
  DISC frame -- normal LAPM termination
```

MODEM_CALL_RECORD Parameters

This table lists and describes the MODEM_CALL_RECORD parameters.

Parameter	Description
ct_hndl	<p>Call Tracker Handle</p> <p>A unique number used by Call Tracker to handle active calls. The calls are assigned an identification (ID) number from 1 to 4,294,967,296. These IDs start with 1 and increment by 1. After 4,294,967,295 calls, the ID wraps, and the 4,294,967,296th call receives the next smallest available number that starts from 1. It is possible for the call history, syslog, and SNMP records to have the same ID number for different calls. This is because the number is only unique for active calls. Zero is not a valid value.</p>
prot: last	<p>Error Correction Protocol: Last</p> <p>Reports last known error correction (EC) protocol in use.</p> <p>EC protocols:</p> <ul style="list-style-type: none">• normal (No EC present)• direct• mnp• lapmV42• syncMode• asyncMode (No EC present, same as normal)• ara1 (ARA 1.0)• ara2 (ARA 2.0)• other (EC protocol other than those identified)

<p>prot: attempt</p>	<p>Error Correction Protocol: Attempted</p> <p>Reports the error correction (EC) protocol first attempted. See <i>prot: last</i> for possible EC protocols.</p>
<p>comp: last</p>	<p>Compression Protocol: Last</p> <p>Reports the last compression protocol in use before the call terminated.</p> <p>Compression protocols include:</p> <ul style="list-style-type: none"> ● none (No Data Compression present) ● v42bisTx (V.42bis in transmit direction only) ● v42bisRx (V.42bis in receive direction only) ● v42bisBoth (V.42bis in receive and transmit directions) mnp5 ● v44Tx (V.44 in transmit direction only) ● v44Rx (V.44 in receive direction only) ● v44Both (V.44 in receive and transmit directions)
<p>comp: supp</p>	<p>Compression Protocol: Supported</p> <p>Compression protocol that could have been supported. See <i>comp: last</i> for possible compression protocols.</p>
	<p>Standard: Last</p> <p>This is the last modulation standard in use before call terminated.</p> <p>Modulation standards include:</p> <ul style="list-style-type: none"> ● other (Modulation other than those identified) ● bell103a

std: last

- bell212a
- v21
- v22
- v22bis
- v32
- v32bis
- vfc
- v34
- v17
- v29
- v33
- k56flex
- v23
- v32terbo
- v34plus
- v90
- v27ter
- v110

std: attempt

Standard: Attempted

Modulation standard attempted by client modem. See *std: last* for possible modulation standards.

std: init

Standard: Initial

First modulation standard attempted by client modem. See *std: last* for possible modulation standards.

std: snr	<p>Standard: Signal to Noise Ratio</p> <p>The ratio measurement of the desired signal to noise. This value can range from 0 to 70 dB and changes in 1 dB steps. Note that a 28.8-kbps connection demands an SNR of about 37 dB. Lower than this and the quality of the connection diminishes. A 33.6-kbps connection demands an SNR of 38 to 39 dB. Also note that a "clean" line has an SNR of about 41 dB.</p>
std: sq	<p>Standard: Signal Quality</p> <p>Measure of line quality for a given bit rate where 0 is the worst and 3 is steady state. If a 1 or 2 is present, the modem must shift down to a lower rate. Likewise, if the Sq value is 4 to 7, the modem speeds shift up to a higher rate. If the Sq value is high (for example, 7) and the bit rate is low, then there may be a problem at the remote end receiver.</p>
rx/tx: chars	<p>Received/Transmitted: Characters</p> <p>The number of bytes transmitted on the call. All the raw bytes are counted. This value includes any protocol headers that may or may not be present. Whether or not the protocol header is present depends on the value of service.</p>
ec: rx/tx	<p>Received/Transmitted: Error Correction Frames</p> <p>The number of EC frames received and transmitted.</p>
ec: rx bad	<p>Error Correction: Received Bad Frames</p> <p>The number of EC frames that had errors.</p>
rx/tx b-rate: last	<p>Receive/Transmit Bit-rate: Last</p> <p>The receive and transmit bit rate when the call terminated.</p>
rx/tx b-rate: low	<p>Receive/Transmit Bit-rate: Low</p> <p>The lowest receive and transmit bit rate encountered for the duration of the call.</p>

rx/tx b-rate: high	<p>Received/Transmit Bit-rate: High</p> <p>The highest receive and transmit bit rate encountered for the duration of the call.</p>
rx/tx b-rate: desired-client	<p>Receive/Transmit Bit-rate: Desired by Client</p> <p>Transmit and receive bit rate that the client wanted to maintain. It is possible that this is not always the bit rate that the host reports, as the host may not train up or down to accommodate.</p>
rx/tx b-rate: desired-host	<p>Receive/Transmit Bit-rate: Desired by Host</p> <p>Desired by host transmit and receive bit rate the host wanted to maintain.</p>
retr: local	<p>Retrains: Local</p> <p>Number of retrains initiated locally.</p>
retr: remote	<p>Retrains: Remote</p> <p>Number of retrains initiated by the remote modem</p>
retr: fail	<p>Retrains: Failed</p> <p>Number of retrains that failed.</p>
speedshift: local up/down	<p>Speed Shifts: Local Up/Down</p> <p>Number of speed up or down shifts initiated by the local modem.</p>
speedshift: remote up/down	<p>Speed Shifts: Remote Up/Down</p> <p>Number of speed up or down shifts initiated by the remote modem.</p>
speedshift: fail	<p>Speed Shifts: Failed</p> <p>Number of speed shifts that failed.</p>

v90: stat	<p>V.90 Status</p> <p>Status of V90 before the call was terminated. Possible status values include:</p> <ul style="list-style-type: none">• no attempt• success• failure
-----------	--

v90: client	<p>V.90: Client</p> <p>Chipset used by the V.90 client modem.</p> <ul style="list-style-type: none">• n/a• Unknown• Rockwell• USR• Lucent• PCTel
-------------	---

v90: fail	<p>V.90 Failures</p> <p>V.90 failure.</p> <p>V.90 failures include:</p> <ul style="list-style-type: none">• none• clientNonPCM• clientFallback• serverV90Disabled
-----------	--

time(sec)	<p>Time (Seconds)</p> <p>How long the call lasted. This value is always returned regardless of the outcome of trainup or authentication.</p>
disc reason	<p>Disconnect Reason</p> <p>ASCII code supplied by the MICA or NextPort modem that disconnects the call.</p> <p>For more information, refer to these documents:</p> <ul style="list-style-type: none"> • Interpreting NextPort Disconnect Reason Codes • MICA Modem States and Disconnect Reasons

Example

```
*Nov 16 18:30:26.097: %CALLTRKR-3-MODEM_CALL_REC:
  ct_hdl=5, prot: last=LAP-M, attempt=LAP-M, comp: last=V.42bis-Both,
  supp= V.42bis-RX V.42bis-TX, std: last=V.34+, attempt=V.34+,
  init=V.34+,
  snr=38, sq=3, rx/tx: chars=246/161, ec: rx/tx=22/12, rx bad=46,
  rx/tx b-rate: last=33600/33600, low=31200/33600, high=33600/33600,
  desired-client=33600/33600, desired-host=33600/33600, retr: local=0,
  remote=0, fail=0, speedshift: local up/down=1/0, remote up/down=0/0,
  fail=0, v90: stat=No Attempt, client=(n/a), fail=None, time(sec)=52,
  disc reason=0xA220MODEM_LINE_CALL_REC Parameters
```

MODEM_LINE_CALL_REC Parameters

This table lists and describes the MODEM_LINE_CALL_REC parameters.

Parameter	Description

ct_hndl	<p>Call Tracker Handle</p> <p>A unique number used by Call Tracker to handle active calls. The calls are assigned an identification (ID) number from 1 to 4,294,967,296. These IDs start with 1 and increment by 1. After 4,294,967,295 calls, the ID wraps, and the 4,294,967,296th call receives the next smallest available number that starts from 1. It is possible for the call history, syslog, and SNMP records to have the same ID number for different calls. This is because the number is only unique for active calls. Zero is not a valid value.</p>
rx/tx level	<p>Receive/Transmit Level</p> <p>Receive/Transmit Level Power of the receive/transmit signal, ranges from 0 to -128 in dBm steps. Typically the range in the United States is about -22 dBm and in Europe is -12 dBm. A good range is from -12dBm to -24dBm. For more information, refer to:</p> <p>Understanding Transmit and Receive Levels on Modems</p>
phase-jit: freq	<p>Phase-jitter: Frequency</p> <p>Peak to peak differential (in hertz) between two signal points. Phase jitter that is not cancelled looks like "rocking" of the baseband quadrature amplitude modulation (QAM) constellation. The points look like arcs with longer arcs on the outer points.</p>
phase-jit: level	<p>Phase-jitter: Level</p> <p>Level amount of phase jitter measured and indicates how large the "rocking" is in degrees. On an oscilloscope, the constellation points would look like crescent moons. Values can range up to 15 degrees. The typical value is zero (that is, phase jitter is not normally present).</p>

far-end echo-level	<p>Far-end Echo-level</p> <p>Over long connections, an echo is produced by impedance mismatches at 2-wire-to-4-wire and 4-wire-to-2-wire hybrid circuitry. The far-end echo level (that portion of the sent analog signal that has bounced off of the remote modem analog front end) may range from 0 to -90 in dBm.</p>
freq offst	<p>Frequency Offset</p> <p>The difference (in hertz) between the expected RX carrier frequency and the actual RX carrier frequency.</p>
phase-roll	<p>Phase-roll</p> <p>Phase roll affects the echo signal coming back. A certain constellation pattern is sent from a modem and arrives at to the central office. Some echoed form of this signal/constellation pattern is sent back. However, the constellation shape may be rotated from 0 to 359 degrees. This rotation is called the phase roll.</p>
round-trip	<p>Round Trip Delay</p> <p>Total round trip propagation delay of the link (in milliseconds). This is important for proper echo cancellation. The amount that the delay varies on the network.</p>
d-pad	<p>Digital-pad</p> <p>Digital padding value.</p>
d-pad comp	<p>Digital-pad Compression</p> <p>This is an integer that represents the compression.</p> <ul style="list-style-type: none"> ● 0 = None ● 1 = V.42bis TX ● 2 = V.42bis RX ● 3 = V.42bis both ● 4 = MNP5 ● 5 = MH (FAX)

	<ul style="list-style-type: none"> • 6 = MR (FAX) • 7 = MMR (FAX) • 8 = V.44 TX • 9 = V.44 RX • 10 = V.44 both • 0xFF (-1) = Data Compression not yet negotiated
rbs	<p>Robed Bit Signaling</p> <p>Actual RBS pattern observed by the modem. The 6 least significant bits (LSB) of the returned value indicate the periodic RBS pattern where a 1 denotes a PCM sample with a robbed-bit.</p>
const	<p>Constellation</p> <p>This is the number of points in the constellation.</p> <ul style="list-style-type: none"> • 0xFF = Invalid • 1 = 4 point • 2 = 16 point
rx/tx: sym-rate	<p>Receive/Transmit: Symbol-rate</p> <p>TX is symbol rate used to send samples to the line. RX is the symbol rate used to receive samples off of the line. The rates are synchronous with each other.</p>
rx/tx: carr-freq	<p>Receive/Transmit: Carrier-frequency</p> <p>For TX, carrier frequency used by the local DCE. For RX, carrier frequency used by the remote DCE.</p>

Example

```
*Nov 16 18:30:26.101: %CALLTRKR-3-MODEM_LINE_CALL_REC:
  ct_hndl=5, rx/tx lev1=-17/-16, phase-jit: freq=0, lev1=0,
  far-end echo-lev1=-71, freq offst=0, phase-roll=-98,
```


proj max rx b-rate: client	Projected Max Receive Bit-rate: client Projected maximum receive bit rate for the client.
rproj max rx b-rate: host	Projected Max Receive Bit-rate: Host Projected maximum receive bit rate for the host.
rx/tx: max neg I frame	Receive/Transmit: Maximum Negotiated I Frame. Transmit and receive maximum negotiated values for the frame.
rx/tx: neg window	Receive/Transmit: Negotiated Window Transmit and receive negotiation window.
T401 timeouts	T401 Timeouts Establish a connection to a client with V.42 EC enabled and pass data from the CSM. Query the statistic before the data is passed and again after the transfer was successful. The statistic should not increment.
tx window closures	Transmit Window Closures Establish a connection to a client and pass data from the CSM. The statistic only increments if the window closes and does not receive an ACK/NAK from the client modem. The expected result should indicate 0.
rx overruns	Received Overruns Total number of received overruns.
retrans frames	Retrain Frames Total retrain frames initiated.
v110: rx good	V.110: Received Good Number of v110 good frames received.
v110: rx bad	V.110: Received Bad Number of v110 bad frames received.

v110: tx	V.110: Transmitted Number of v110 frames transmitted.
v110: sync lost	v110: synchronization lost. Number of times v110 synchronization is lost.
ss7/cot	Signaling System 7 (SS7) and Continuity Test (COT) statistics.
v42bis size: dict	V.42bis Size: Dictionary Provides v42bis dictionary size.
test err	Test Error Self test error encountered.
reset	Reset DSP reset value.
v0 synch-loss	V.0 Synchronization Loss Establish a connection with a client and verify the query indicates 0. The counter should only increment V0 sync is lost in the received signal which will trigger a retrain.
Mail lost: host	Mail Lost: Host Number of host mail lost.
sp	SP Number of sp mail lost.
diag	Diagnostic Value for the portware diagnostic.

Example

```
*Nov 16 18:30:26.101: %CALLTRKR-3-MODEM_INFO_CALL_REC:
  ct_hndl=5, general info=0x0, rx/tx link-layer=264/182, NAKs=0/0,
  rx/tx ppp-slip=5/7, bad ppp-slip=0, proj max rx b-rate:
client=19200,
  host=24000, rx/tx: max neg I frame=128/128, neg window=15/15,
  T401 timeouts=1, tx window closures=0, rx overruns=0, retrans
frames=0,
```

```
v110: rx good=0, rx bad=0, tx=0, sync-lost=0, ss7/cot=0x00,
v42bis size: dict=1024, test err=0, reset=0, v0 synch-loss=0, mail
lost:
host=0, sp=0, diag=0x00000000000000000000000000000000
```

MODEM_NEG_CALL_REC Parameters

This table lists and describes the MODEM_NEG_CALL_REC parameters.

Parameter	Description
ct_hndl	<p>Call Tracker Handle</p> <p>A unique number used by Call Tracker to handle active calls. The calls are assigned an identification (ID) number from 1 to 4,294,967,296. These IDs start with 1 and increment by 1. After 4,294,967,295 calls, the ID wraps, and the 4,294,967,296th call receives the next smallest available number that starts from 1. It is possible for the call history, syslog, and SNMP records to have the same ID number for different calls. This is because the number is only unique for active calls. Zero is not a valid value.</p>
v8bis cap	<p>V.8bis Capabilities.</p> <p>Capabilities list received during V.8bis represented in hex. Refer to ITU-T V.8bis for more information about these bits.</p>
v8bis mod_sl	<p>V.8 Bis Mode Select</p> <p>Mode selected during V.8bis represented in hex. Refer to ITU-T V.8bis for more information about these bits.</p>
v8 jnt-menu	<p>V.8 Joint-menu</p> <p>Joint menu exchanged during V.8 represented in hex. Refer to ITU-T V.8 for more information about these bits.</p>
v8 call-menu	<p>V.8 Call-Menu</p> <p>Call menu exchange V.8 Call-Menu during V.8 represented in hex. Refer to ITU-T V.8 for more information about these bits.</p>
v90 train	<p>V.90 Train</p> <p>Representation of V.90 Train in Hex.</p>

CISCO-CALL-TRACKER-MIB

This table lists and describes the traps that are sent when a call is received by the host and Call Tracker is configured to send SNMP traps to a host.

Name	Description
1.3.6.1.4.1.9.9.9991.1.2.3.1.2	The object ID (OID) of the trap.
.x	The ct_hndl assigned to the call.
=	
Timeticks: (119447) 0:19:54.47	The uptime of the router when the call arrived.

Example

```
Mar 12 06:27:00
  localhost
  snmptrapd[28977]:
  172.22.35.14:
  1.3.6.1.4.1.9.9.9991.1.2.3.1.2.1 = Timeticks: (119447) 0:19:54.47
```

This trap came from the host 172.22.35.14 and the **ct_hndl** assigned to the call is 1. With the **ct_hndl**, it is possible to poll further information from the active table as described in the SNMP section. The uptime of the host when the call arrived was Timeticks: (119447) 0:19:54.47.

This table lists and describes the traps that are sent when a call is released by or released from the system and Call Tracker is configured to send SNMP traps to a host.

Name	Description
1.3.6.1.4.1.9.9.9991.1.3.8.1.2	The OID of the trap
.x	The ct_hndl assigned to the call when it was active.
=	
Gauge: 1	The entry assigned to the call in the history table.

Example

```
Mar 12 06:27:21
  localhost
  snmptrapd[28977]:
  172.22.35.14:
```

1.3.6.1.4.1.9.9.9991.1.3.8.1.2.1 = Gauge: 1

The trap in this example came from the host 172.22.35.14. The original **ct_hndl** number in this case is 1, and the entry in the history table (value returned) is 1. These numbers must always be the same, but this cannot be guaranteed. You can use the number returned to get any further information about the call from the history table as described in the SNMP section.

Cisco Support Community - Featured Conversations

[Cisco Support Community](#) is a forum for you to ask and answer questions, share suggestions, and collaborate with your peers. Below are just some of the most recent and relevant conversations happening right now.

Want to see more? Join us by clicking [here](#)

•

[Understanding Call-Tracker output](#) **mkoopsen3 Replies** 7 years, 4 months ago

•

[Call Tracker Output](#) **tamer1 Reply** 7 years, 3 months ago

•

[User Tracker](https://supportforums.cisco.com/people/kjmattakat%40bremer.com) **https://supportforums.cisco.com/people/kjmattakat%40bremer.com8 Replies** 2 years, 10 months ago

•

[Syslog outputs](#) **j.parry1 Reply** 7 years, 8 months ago

•

[Understanding Call per second](#) **sm-menzozi3 Replies** 7 years, 5 months ago

•

[Understanding call flow in CER](#) **jason-calbert2 Replies** 3 years, 4 months ago

•

[Understanding IPCC, CAI Manager](#) **wilson_123423 Replies** 2 years, 6 months ago

•

[Understanding IPCC, Call Manager](#) **wilson_12343 Replies** 2 years, 7 months ago

•

[Understanding IPCC, Call Manager](#) **wilson_12342 Replies** 2 years, 7 months ago

•

[understanding CDR and call recording](https://supportforums.cisco.com/people/meetnajmi%40rediffmail.com) **https://supportforums.cisco.com/people/meetnajmi%40rediffmail.com2 Replies** 1

Related Information

- [Call Tracker plus ISDN and AAA Enhancements for the Cisco AS5300 and Cisco AS5800](#)
 - [Basic Dial NMS Implementation Guide](#)
 - [Cisco MIB Navigator](#)
 - [MICA Modem States and Disconnect Reasons](#)
 - [Interpreting NextPort Disconnect Reason Codes](#)
 - [Cisco IOS SNMP Traps Supported and How to Configure Them](#)
 - [Technical Support and Documentation - Cisco Systems](#)
-

Home	How to Buy	Login	Profile	Feedback	Site Map	Help
----------------------	----------------------------	-----------------------	-------------------------	--------------------------	--------------------------	----------------------

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2009 - 2010 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)