



# Troubleshooting and Monitoring Tools and Utilities

---

This chapter addresses the tools and utilities you use to configure, monitor, and troubleshoot Cisco CallManager.

This chapter contains the following topics:

- [Using Cisco CallManager Administration Details, page 2-1](#)
- [Using Microsoft Performance, page 2-2](#)
- [Using Microsoft Event Viewer, page 2-3](#)
- [Using SDI Trace, page 2-4](#)
- [Using SDL Trace, page 2-5](#)
- [Sniffer Trace, page 2-8](#)
- [Call Detail Records and Call Management Records, page 2-9](#)
- [Using Admin Serviceability Tool, page 2-11](#)

## Using Cisco CallManager Administration Details

Cisco CallManager Administration provides version information for the system, database, and other components.

### Procedure

- 
- Step 1** From the Cisco CallManager Administration window, click **Details**.
- Step 2** Write down the version you are using.
- 

Refer to the *Cisco CallManager Administration Guide* for more details.

## Using Microsoft Performance

You can use Windows 2000 Performance to collect and display system and device statistics for any Cisco CallManager installation. This administrative tool allows you to gain a full understanding of your system without studying the operation of each of its components.

You can use Performance to monitor a variety of system variables in real time. After adding the Cisco CallManager parameters, you can define the terms that will display statistics generated by the system. For example, you can monitor the number of calls in progress at any time, or the number of calls currently passing through a specific gateway. Performance shows both general and Cisco CallManager-specific status information in real-time.

## Opening Microsoft Performance

To open Performance on the server PC running Cisco CallManager:

Click **Start > Settings > Control Panel > Administration Tools > Performance**

## Customizing Performance

The Performance monitor must be customized to view the Cisco CallManager-related parameters that you want to monitor. Choose the object, counter, and instance you want to include.

Refer to the *Cisco CallManager Serviceability Administration Guide* for instructions on how to use objects and counters to customize Microsoft Performance for Cisco CallManager operations.

This document is available online at the following location:

[http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_callmg/3\\_1/service/index.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_callmg/3_1/service/index.htm)

## Using Microsoft Event Viewer

You can use Microsoft Event Viewer to display system, security, and application events (including Cisco CallManager) for the Windows NT Server. If a service (including TFTP) cannot read the database (where it gets the trace configuration), it will add errors to the Event Viewer.

### Opening Event Viewer

To open the Event Log on the server PC running Cisco CallManager:

Click **Start > Settings > Control Panel > Administrative Tools > Event Viewer**. The Event Viewer provides error logs for System, Security, and Applications. Cisco CallManager errors are logged under the Application log.

Refer to the Alarms chapter in the *Cisco CallManager Serviceability Administration Guide*, available online at the following location:

[http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_callmg/3\\_1/service/index.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_callmg/3_1/service/index.htm)

### Detailed Information about Events

You can double-click an event in the log to learn more information about the event.

# Using SDI Trace

You can use SDI traces to view local log files. To monitor the occurrence or the disposition of a request, you can trace the IP address, TCP handle, device name or time stamp. This device name could be tracked back to the building of the file, which shows the device pool and model. The device pool and model can be tracked back to the building of the configuration file prototype, which will list the network address of the Cisco CallManager(s) and the TCP connection port.

When observing SDI traces, notice that C++ class and routine names are included with most trace lines. Most routines associated with the serving of a particular request include the thread ID in a standard format.

SDI traces are explained in detail in the case studies in the appendices.

**Note**

---

Cisco CallManager Release 3.1 Serviceability supports the Trace interface. Refer to the following online location for information on Trace:

[http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_callmg/3\\_1/service/index.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_callmg/3_1/service/index.htm)

---

## SDI Trace Output

SDI traces generate files (for example, CCM000000000) that store traces of Cisco CallManager activities. These traces provide information about:

- Cisco CallManager initialization process
- registration process
- KeepAlive process
- call flow
- digit analysis
- related devices such as Cisco IP Phones, Gateways, Gatekeepers, and more.

This information can help you isolate problems when troubleshooting Cisco CallManager. To properly track the information you need—and only the information you need—it is important to understand how to set the options on the trace configuration interface.

If the trace is not configured properly, it will generate a large amount of information, making it very difficult to isolate problems. The following section explains how to properly configure a useful trace.

## Configuring Traces

Traces are composed of user mask flags (also known as bits) and trace levels.

- 
- Step 1** Open Cisco CallManager Administration.
  - Step 2** Choose **Application > Cisco CallManager Serviceability**.
  - Step 3** Choose **Trace > Configuration**.
- 

Refer to the *Cisco CallManager Serviceability Administration Guide* for complete information about configuring Trace.

The *Cisco CallManager Serviceability Administration Guide* is located online at: [http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_callmg/3\\_1/service/index.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_callmg/3_1/service/index.htm)

## Using SDL Trace

Cisco engineers use SDL traces to find the cause of an error. You are not expected to understand the information contained in an SDL trace. However, while working with the Technical Assistance Center (TAC), you may be asked to enable the SDL trace and provide it to the TAC. SDL trace files can be saved to local directories, the Windows NT Event Viewer, and CiscoWorks 2000. To avoid any performance degradation on the server, be sure that after the trace has been captured, you turn off SDL tracing.

SDL trace provides a C interface to trace and alarms. Alarms are used to inform the administrator of unexpected events, such as being unable to access a file, database, Winsock, or being unable to allocate other operating system resources.

## Enabling SDL Trace

SDL traces are enabled in the **Service > Service Parameter** area in the Cisco CallManager Administration. Remember that these traces should be turned on only when requested by a TAC engineer.

Refer to the *Cisco CallManager Serviceability Administration Guide* for complete information about configuring Trace.

The *Cisco CallManager Serviceability Administration Guide* is located online at: [http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_callmg/3\\_1/service/index.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_callmg/3_1/service/index.htm)

Once SDL traces are enabled, collect the traces. If the traces are being sent to the local drive, then you can retrieve them in the Cisco\Trace subdirectory. Alternatively, the trace files can be sent to an event log or to CiscoWorks 2000.

SDL flag bits described in the following table are set in the **Service > Service Parameters** area in Cisco CallManager Administration. Following are two examples of desired values based on the particular problem.

- The recommended value for normal call debugging is SdlTraceTypeFlags=0x00000b04
- The recommended value for low level debugging or debugging gateways is SdlTraceTypeFlags=0x00004b05

The following table defines the SdlTraceTypeFlags Definitions.

SDLTraceTypeFlag	Value	Definition
traceLayer1	= 0x00000001	All Layer 1 trace on
TraceDetailLayer1	= 0x00000002	Detail Layer 1 trace on
TraceSdlLinkAdmin	= 0x00000004	Trace inter-Cisco CallManager links within a cluster
traceUnused	= 0x00000008	Not used
traceLayer2	= 0x00000010	All Layer 2 trace on
traceLayer2Interface	= 0x00000020	Layer 2 interface trace on
traceLayer2TCP	= 0x00000040	Layer 2 TCP trace on

SDLTraceTypeFlag	Value	Definition
TraceDetailLayer2	= 0x00000080	More detail dump of Layer 2 Frames
traceLayer3	= 0x00000100	All Layer 3 trace on
traceCc	= 0x00000200	All call control trace on
traceMiscPolls	= 0x00000400	Trace miscellaneous polls
traceMisc	= 0x00000800	Miscellaneous trace on (Database signals)
traceMsgtrans	= 0x00001000	Message Translation signals (TranslateIsdnToSdlReq, TranslateIsdnToSdlRes, TranslateSdlToIsdnReq, TranslateSdlToIsdnRes)
traceUuie	= 0x00002000	UUIE output trace on
traceGateway	= 0x00004000	Gateway signals

Data bits described in the following table are set in the **Service > Service Parameters** area in Cisco CallManager Administration. Following are two examples of desired values based on the particular problem.

- The recommended value for normal system debugging is SdlTraceDataFlags=0x110
- The recommended value when tracking problems with SDL links is 0x13D (non-compacted trace; if a compact trace is desired, bit 0x200 must be set. It can be set in combination with any other bits)

The following table shows the SDLTraceDataFlags Definitions.

SDLTraceDataFlag	Value	Definition
TraceSdlLinkState	= 0x001	Enable trace of SDL Link Initialization
TraceSdlLowLevel	= 0x002	Enable tracing of low-level SDL events, for example, fileOpen, socket events, and so on
TraceSdlLinkPoll	= 0x004	Enable tracing of SDL Link Poll message
TraceSdlLinkMsg	= 0x008	Enable tracing of SDL Link message
traceRawData	= 0x010	Enable raw signal data trace on all signals
TraceSdlTagMap	= 0x020	Enable tag mapping
traceCreate	= 0x100	Enable process create and stop traces
TraceNoPrettyPrint	= 0x200	Disable pretty printing of trace files



#### Tips

Be advised that information obtained from this interface could be very detailed, and therefore consume a large amount of disk space. For this reason, we advise you to turn on the trace file for a specific amount of time, then review the information and turn off the trace.

## Sniffer Trace

A Sniffer is a software application that monitors IP traffic on a network and provides information in the form of a trace. Sniffer traces provide information about the quantity and type of network traffic on your network. TCP/IP or UDP

packets are protocols utilized by Cisco CallManager and endpoint devices such as phones and gateways. Sniffer traces can also help you identify high levels of broadcast traffic that could result in voice audio problems or dropped calls.

Common Sniffer applications include Network Associates SnifferPro, Hewlett Packard Internet Advisor, and Acterna Domino. Domino offers sniffing hardware and software solutions and a network analyzer. If you want to use Domino, we recommend using the analysis software to evaluate a captured sniffer file (such as from the SnifferPro application).

## Sniffer Trace Applications

Use the following links to learn more about some available sniffer trace applications. Any sniffer application will work with Cisco CallManager.

- Network Associates SnifferPro: <http://www.sniffer.com/>
- Hewlett Packard Network Analyzer:  
[http://www.hp.com/rnd/products/top\\_tools/summary.htm](http://www.hp.com/rnd/products/top_tools/summary.htm)
- Acterna Domino Analyzer:  
<http://www.acterna.com/products/domino/index.html>
- Shomiti Surveyor Protocol Analyzer:  
<http://www.shomiti.com>

## Call Detail Records and Call Management Records

Call Detail Records (CDR) is a reporting option that logs every call made (or attempted) from any Cisco IP Phone. There are two kinds of CDRs:

- basic CDRs
- Diagnostic CDRs, also known as CMRs

Once enabled, you can open CDRs or CMRs in the SQL Server Enterprise Manager. CDR files are saved in a SQL database that can be exported to nearly any application, including Microsoft Access or Excel.

## Enabling or Disabling CDRs

CDR record creation is disabled by default when the system is installed. If you wish to have CDR data, you must enable CDRs in the **Service > Service Parameters** area of Cisco CallManager Administration. CDR processing can be enabled and disabled at any time while the system is in operation. You do not need to restart Cisco CallManager for the enabling or disabling of CDRs to take effect. The system will respond to all changes within a few seconds.

## Using CMRs

Diagnostic CDRs provide detailed call information, such as the number of packets sent, received, and lost, the amount of jitter and latency, and so on. This level of detail can provide explanations for some problems, such as one-way audio. For example, a one-way audio problem is indicated if a packet size of 10,000 is sent, but the received size is only 10.

CMR or diagnostic data is enabled separately from CDR data. CMR data will not be generated unless both CDRs and Call Diagnostics are enabled, but CDR data may be generated and logged without CMR data.

Perform the following procedure to enable CDRs.

**Caution**

---

Tracing voice connectivity requires that CDR logging be enabled on every Cisco CallManager installation in a cluster.

---

**Procedure**

- 
- Step 1** Open Cisco CallManager Administration.
  - Step 2** Choose **Service > Service Parameters**.
  - Step 3** Choose the IP address of your Cisco CallManager installation.
  - Step 4** Choose the Cisco CallManager service.
  - Step 5** From the list of Parameters, choose **CDREnabled**.
  - Step 6** Select **T** for True.

- Step 7** Click **Update**.  
Result: Call Detail Records will start logging immediately.
- 

## Using Admin Serviceability Tool

You can use the Admin Serviceability Tool (AST) in Cisco CallManager Serviceability to monitor the real-time behavior of the components in a Cisco CallManager cluster. The AST uses HTTP and TCP to monitor device status, system performance, and device discovery. It also connects directly to devices using HTTP for troubleshooting system problems.

The *Cisco CallManager Serviceability Administration Guide* is located online at:

[http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_callmg/3\\_1/service/index.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_callmg/3_1/service/index.htm)

