

Managing Voice Quality with Cisco Voice Manager (CVM) and Telemate

Document ID: 13940

Contents

Introduction

Prerequisites

- Requirements
- Components Used
- Conventions

Voice Quality Overview

- Measuring Voice Quality
- ITU G.113 Overview

Managing Voice Quality with CVM and Telemate

- Limitations
- Gateway Configuration
- CVM and Telemate Architecture
- Telemate Directory
- Reporting

Related Information

Introduction

This document describes the use of Cisco Voice Manager and Telemate to manage voice quality in a VoIP network. All content is based on a real world IP Telephony implementation. This document focuses on the application of the products and not the use of the products. You should already be familiar with CVM and Telemate and have access to the required product documentation. See Related Information for a list of related documentation.

When managing a large-scale VoIP network, you must have the necessary tools for objectively monitoring and reporting the voice quality in the network. Relying on user feedback alone is not feasible because it is subjective and incomplete. CVM, together with Telemate, can provide part of this function. It reports on voice quality by using the Impairment/Calculated Impairment Planning Factor (*Icpif*) calculated by an IOS gateway for each call. This allows the network manager to identify sites that suffer from poor voice quality and deal with them appropriately.

Once you identify problem sites, you may need other tools to troubleshoot possible network QoS issues. Two tools are the Internetwork Performance Monitor (IPM) and Cisco Service Assurance Agent (CSAA). These topics are discussed in another document posted on our web site.

Prerequisites

Requirements

Readers of this document should have knowledge of these topics:

- Cisco Voice Manager and Telemate

Components Used

This document is not restricted to specific software and hardware versions.

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

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

Voice Quality Overview

The following sections provide an overview of voice quality issues:

- Measuring Voice Quality
- ITU G.113 Overview

Measuring Voice Quality

The ITU standard G.113 specifies how to measure voice quality. This method dictates that you can determine the quality of voice calls by calculating the *Icpif*. IOS-based gateways calculate the *Icpif* value for every call and log it as part of the CDR record. In addition, it can send a Quality of Voice (QoV) trap via SNMP if the *Icpif* value of a call exceeds a preset value. This means that the gateways have built-in voice quality measurement abilities. All that is necessary is collect these measurements and analyze the data to identify any trends.

VoIP voice quality is mainly affected by network QoS. The call analysis will therefore focus on identifying voice quality problems on a per-site basis. If sites that have a large number of calls with poor voice quality can be identified, we can focus on any QoS issues in the network path to and from those sites.

ITU G.113 Overview

The following section is only a brief overview; consult the G.113 standard for more detailed information.

The general idea behind G.113 is to calculate an impairment factor for every piece of equipment along the voice path and then add them up to get the total impairment. There are different types of impairments (noise, delay, echo, etc) and the ITU divides them into five categories. Add them up to get the total impairment *Itot*:

$$Itot = Io + Iq + Idte + Idd + Ie$$

Each of these are defined as follows (using ITU terminology):

- *Io* impairments caused by non-optimum overall loudness rating and/or high circuit noise.
- *Iq* impairments caused by PCM type quantizing distortion.
- *Idte* impairments caused by talker echo.
- *Idd* speech communication difficulties caused by long one-way transmission times (delay).
- *Ie* impairments caused by special equipment, in particular non-waveform low-bit-rate codecs.

When Cisco IOS software calculates *Itot*, it ignores *Io* and *Iq* as being negligible and sets *Idte* to 0. The *Idd* value is derived from the following table, which comes from G.113:

Delay	<i>I_{dd}</i>
150	0
200	3
250	10
300	15
400	25
500	30
600	35
800	40

Normally *I_e* is a fixed value, depending only on the codec type. G.113 specifies the values for codecs typically used by Cisco gateways as shown in the following table:

Code	<i>I_e</i>
G.711	0
G.729/G.729a	10

However, because these codecs are used in a packet voice environment, the actual impairment depends on the packet loss. The higher the packet loss, the higher the impairment. Cisco engineering has measured voice quality with PSQM (ITU P.861) at discrete packet loss levels. The following table shows voice distortion values relative to packet loss levels for given codecs:

Packet Loss %	G.711	G.729/G.729a
0	0	10
1	8	15
2	12	20
3	18	25
4	22	30
5	26	34
6	28	38
7	30	40
8	32	42
9	34	44

As expected, G.729 is more susceptible to packet loss than G.711.

Voice quality is all about human perception and expectation. The service level expectations of cell phone users are lower than those of fixed line users. We take this into account when calculating the *I_{cpif}* by reducing *I_{tot}* by the human expectation factor *A*. The formula for this is:

$$I_{cpif} = I_{tot} - A$$

G.113 also provides expectation factors for typical voice networks. See the following table:

Voice Network Access Method	Expect Factor A
Conventional fixed line PSTN	0
Local area wireless (cordless phone)	5
Wide area wireless (cell phone)	10
Satellite	20

G.113 also has a table that maps between *Icpif* value and voice quality. It is shown in the following table:

Voice Network Access Method	Expect Factor A
5	Very good
10	Good
20	Adequate
30	Limiting case
45	Exceptionally limiting case
55	Users likely to complain strongly

An *Icpif* value of zero for a call is a perfect score. This is should be our target for VoIP networks.

In a traditional voice network, the designer would calculate the total impairment budget.

For example, $I_o = 0$; $I_q = 0$; $I_{dte} = 0$; $I_{dd} = 3$; $I_e = 7$, which gives $I_{tot} = 10$.

If the user is accessing the network from a cordless phone, then the maximum expectation factor that can be subtracted is 5, so the end result is:

$$I_{cpif} = I_{tot} - A = 10 - 5 = 5$$

According to the previous table, the users are then likely to perceive the voice quality as being very good.

This document discusses a solution that uses the *Icpif* value for monitoring voice quality rather than using it for planning purposes.

Managing Voice Quality with CVM and Telemate

The following sections discuss how to manage voice quality with CVM and Telemate:

- Limitations
- Gateway Configuration
- CVM and Telemate Architecture
- Telemate Directory
- Reporting

Limitations

While the proposed solution does have some limitations, there appears to be no other scalable tools available.

Known limitations include:

- Only calls through a gateway are subject to quality control. You cannot measure calls from iPhone to iPhone. The gateway does not see these calls and CallManager does not currently support G.113.
- The *Icpif* calculation takes into account only packet loss and delay. Echo is not included in the *Icpif* calculations. Therefore, a call may suffer severe echo and still get a perfect *Icpif* score.
- Voice quality is only measured in the iPhone-to-gateway direction. The *Icpif* value in a packet voice network is likely to be asymmetric in the two directions. Any unidirectional network QoS problems in the gateway-to-iPhone direction will not be reflected in the *Icpif* value calculated by the gateway.
- Voice quality issues are generally more of an issue across a WAN. The solution discussed fits best in an environment with centralized gateways, as calls from iPhones at remote sites have to cross the WAN to access the gateways. If gateways are distributed (i.e., each remote site is serviced by a local gateway), then most gateway calls will not cross the WAN. VoIP calls across the WAN will be mainly iPhone-to-iPhone, and these are not visible to the gateway.

Gateway Configuration

As part of the proposed solution, all gateways need to be configured for CDR collection:

```
dial-control-mib max-size <max-number-of-cdr>
dial-control-mib retain-timer 600
```

All gateways must also have the QoV trap feature enabled. This feature is disabled by default:

```
Calibra#show dial-peer voice 99 | include QOV|Icpif
Expect factor = 0, Icpif = 20,
VAD = enabled, Poor QOV Trap = disabled,
```

This feature is enabled on a per VoIP dial-peer basis by adding the following:

```
dial-peer voice XYZ voip
snmp enable peer-trap poor-qov
icpif <threshold>
expect-factor 0
```

When a call completes, the gateway calculates the total impairment (*Itot*) for that call. It then subtracts the configured expect-factor from *Itot* to arrive at the actual *Icpif* value. If this number exceeds the *Icpif* threshold, then a QoV trap is sent. The call durations must be at least 10 seconds for the gateway to calculate the *Icpif* value for the call.

Let's look at an example, where the gateway configuration is as follows:

```
dial-peer voice XYZ voip
icpif 10
expect-factor 5
```

Assume that a call completes with an *Itot* value of 20. The gateway then subtracts an expect factor of 5 from this number, giving an *Icpif* value of 15. Because 15 is more than 10, the gateway generates a QoV SNMP trap.

Globally, it is necessary to enable QoV traps to be sent to CVM:

```
snmp-server enable traps voice poor-qov
snmp-server host 10.x.x.x public<----- CVM station
```

Beware that voice gateways generate linkup/linkdown SNMP traps each time a call is setup or torn down.

This can amount to an enormous number of traps on high density gateway. Make sure to disable these traps by adding the following command:

```
interface serial1/0:15no snmp trap link-status
```

CVM and Telemate Architecture

CVM and Telemate are completely separate applications. As the name implies, CVM is a Cisco developed product. Telemate, on the other hand, is a third party product that Cisco sells bundled with CVM.

CVM performs a variety of functions. The two functions that we will make use of are:

- Collecting Call Detail Records (CDR) from the gateways via SNMP.
- Receiving Quality of Voice (QoV) SNMP traps from the gateways.

After collecting this information, CVM formats the data and passes it onto Telemate via simple file sharing. Telemate then processes this data and stores it in a Microsoft SQL database. The end result is a database with a list of calls with their respective details, including the Icpif value. Various reports can then be run against the database, including QoV reports.

The Telemate QoV report that we are interested in is the "Packet Voice Calls with Quality of Service Traps" report. This report lists all calls for which the gateway generated a QoV trap. We are not interested in the individual calls; rather, we are interested in identifying the sites, if any, that have an above average percentage of calls with voice quality. To achieve this, Telemate needs to be able to categorize the calls by site. This is discussed in the following section.

Telemate Directory

By populating the Telemate directory with knowledge of which extensions reside at what sites, we can use Telemate to categorize calls by site.

The Telemate directory is a five-layer hierarchy, with the following levels:

- Level 1 – Company
- Level 2 – Division
- Level 3 – Department
- Level 4 – User
- Level 5 – Extension

You can associate multiple extensions with one user.

Ideally, we would like each call in the QoV report to be listed with the department name. We could then use the department name to represent a given site. This allows us to sort calls by department/site. But because extensions can be associated with users only, we have to achieve this in a slightly awkward manner. Basically we create one dummy user per site, and we make the name of this user the site name or site code. This dummy user is then assigned all extensions for that particular site. We can then sort the calls by user, which then becomes the equivalent to sorting them by site.

For the purpose of QoV reporting, we do not care about the top three levels of the directory hierarchy, and these can be assigned any arbitrary value.

For this implementation, there are 200 sites with 45,000 extensions assigned although not necessarily all in use. So the directory contains 200 dummy users and each dummy user is associated with the range of extensions for their site. Populating the directory manually would be an impossible task so we do this

semi-automatically by generating a CSV file with one line per extension, and we then use the Telemate import feature to import the file into the directory. Each line in this CSV file has the following format:

```
Company,Division,Department,User,Extension
```

Generating the CSV file itself is also done semi-automatically by running a Unix shell script. This script takes a seed file as input. This seed file lists the sites and the associated extension ranges. Each line in the seed file has this format:

```
site_name,extention_start,extension_end
```

The shell script itself is very simple, and looks like this:

```
#----- Telemate script start -----
#!/bin/ksh

for i in `cat ./$1`
do (
    echo $i | awk 'BEGIN{FS=","}{for (j=($2+0);j<($3+0);++j) printf "Company,Division,Dept,'
) done
#----- Telemate script end -----
```

Assuming that the script itself is named 'make_dir' and that the seed file is called 'seedfile.csv', the import CSV telemate_dir.csv file is created by executing the following command at the Unix prompt:

```
unix$ make_dir seedfile.csv > telemate_dir.csv
```

The output file telemate_dir.csv is then imported into Telemate. Refer to the Telemate documentation for detailed instructions on how to do this.

Reporting

When running a Telemate report, you can select the output destination. For large reports, it is recommended that the file be produced in CSV format. You can then manipulate the report in Excel, where it would look like this:

Duration	Dialed #	Location	Date	Time	Site	Ext.
0:00:57	3-573-7783	10.200.16.33	10/05/2000	4:49:45PM	BLM	37569
0:00:57	3-573-7783	10.200.16.33	10/05/2000	4:49:45PM	BLM	37569
0:00:38	3-577-2958	10.200.16.33	10/05/2000	4:28:28PM	BLM	37576
0:00:38	3-577-2958	10.200.16.33	10/05/2000	4:28:28PM	BLM	37576
0:00:52	3-577-2985	10.200.16.33	10/05/2000	9:26:33PM	BLM	37593
0:01:19	3-577-1770	10.200.16.33	10/05/2000	7:26:05PM	BMC	34270
0:00:23	3-577-1770	10.200.16.33	10/05/2000	8:08:27PM	BMC	34270
0:00:23	3-577-1770	10.200.16.33	10/05/2000	8:08:27PM	BMC	34270
0:00:11	4-566-5302	10.132.16.33	10/05/2000	7:05:33PM	COR	42791
0:00:32	4-567-0417	10.132.16.33	10/05/2000	5:29:51PM	COR	42805
0:00:32	4-567-0417	10.132.16.33	10/05/2000	5:29:51PM	COR	42805

0:00:36	4-232-8545	10.132.16.33	10/05/2000	5:42:07PM	COR	42823
0:00:36	4-232-8545	10.132.16.33	10/05/2000	5:42:07PM	COR	42823
0:00:39	4-472-5011	10.132.16.33	10/05/2000	5:59:23PM	COR	46578
0:00:39	4-472-5011	10.132.16.33	10/05/2000	5:59:23PM	COR	46578
0:00:28	4-236-7687	10.132.16.33	10/05/2000	7:17:51PM	COR	46578
0:00:17	6-867-9766	10.132.16.35	10/05/2000	4:08:02PM	GIS	64197
0:00:17	6-867-9766	10.132.16.35	10/05/2000	4:08:02PM	GIS	64197
0:00:30	6-868-6889	10.132.16.35	10/05/2000	6:15:48PM	GIS	68549
0:00:30	6-868-6889	10.132.16.35	10/05/2000	6:15:48PM	GIS	68549
0:01:26	6-876-5223	10.132.16.35	10/05/2000	7:10:23PM	HAH	68369
0:01:26	6-876-5223	10.132.16.35	10/05/2000	7:10:23PM	HAH	68369
0:00:52	6-876-2223	10.132.16.35	10/05/2000	5:37:58PM	HAH	68397
0:01:05	4-477-5402	10.132.16.33	10/05/2000	4:23:20PM	JVL	47162
0:00:24	4-478-8848	10.132.16.33	10/05/2000	7:07:09PM	JVL	47168
0:00:24	4-478-8848	10.132.16.33	10/05/2000	7:07:09PM	JVL	47168
0:00:44	4-387-1333	10.132.16.33	10/05/2000	7:49:16PM	KIB	49252
0:00:44	4-387-1333	10.132.16.33	10/05/2000	7:49:16PM	KIB	49252
0:01:14	4-389-4299	10.132.16.33	10/05/2000	4:07:10PM	KIB	49254
0:01:14	4-389-4299	10.132.16.33	10/05/2000	4:07:10PM	KIB	49254
0:00:29	4-387-1337	10.132.16.33	10/05/2000	4:06:45PM	KIB	49256
0:00:29	4-387-1337	10.132.16.33	10/05/2000	4:06:45PM	KIB	49256
0:00:41	4-384-9269	10.132.16.33	10/05/2000	4:09:38PM	KIB	49261
0:00:41	4-384-9269	10.132.16.33	10/05/2000	4:09:38PM	KIB	49261
0:00:41	4-384-9269	10.132.16.33	10/05/2000	4:09:38PM	KIB	49261
0:00:17	4-387-1344	10.132.16.33	10/05/2000	4:33:04PM	KIB	49263
0:00:17	4-387-1344	10.132.16.33	10/05/2000	4:33:04PM	KIB	49263
0:00:31	6-367-5103	10.132.16.35	10/05/2000	8:44:46PM	LEV	64233
0:00:31	6-367-5103	10.132.16.35	10/05/2000	8:44:46PM	LEV	64233
0:00:30	6-368-9088	10.132.16.35	10/05/2000	4:11:06PM	LEV	64247
0:00:30	6-368-9088	10.132.16.35	10/05/2000	4:11:06PM	LEV	64247
0:00:38	4-570-2450	10.132.16.33	10/05/2000	4:08:26PM	LHT	43636
0:00:38	4-570-2450	10.132.16.33	10/05/2000	4:08:26PM	LHT	43636

Use the Excel "subtotals" feature to count the number of bad calls per user/site. Then create an Excel macro to semi-automate the subtotalling. See the following example:

Duration	Dialed #	Location	Date	Time	Site	Ext.
					5	

				BCM Count		
				BMC Count	3	
				COR Count	8	
				GIS Count	4	
				HAH Count	3	
				JVL Count	3	
				KIB Count	11	
				LEV Count	4	
				LHT Count	2	
				Grand Count	43	

The **Site** column now contains the number of bad calls to/from that site. The **Location** column in the report is the IP address of the other end of the VoIP leg and comes from the gateway CDR record. In a CallManager (CCM) environment, the signaling and media end points are two distinct IP addresses. The IP address listed is the signaling end-point (i.e., the CallManager). A DDTS (CSCds23283) has been submitted to request a knob that allows the CDR record to log the media IP address instead. This would allow bad calls to be sorted by subnet. This gives better granularity as there would typically be multiple subnets per site. If only some of these subnets are suffering QoV problems, then these can be identified.

We recommend that you set up the Telemate scheduler to automatically run the "Packet Voice Calls with Quality of Service Traps" report once a day. Completed reports can then be e-mailed to selected operational staff. These staff members then do a daily QoV audit for the past 24 hours. Reports should be archived for at least one month so that any deterioration in QoV can be correlated with any network changes performed around that time.

Note: Telemate version 4.7 or later is required for reporting to work properly with gateways operating in a CallManager environment. Earlier versions of Telemate assume that the local extensions are always on the POTS side of the gateway. In a CallManager environment, the local extensions (IPhones) are on the VoIP side of the gateway. As a result, earlier versions of Telemate get confused and the reports are of limited value.

Related Information

- **CiscoWorks 2000 Documentation**
- **CVM 2.0 Call History Report Technical Tip**
- **Telemate Technical Support Site** [☞](#)
- **Using Cisco Service Assurance Agent and Internetwork Performance Monitor to Manage Quality of Service in Voice over IP Networks**
- **Voice Technology Support**
- **Voice and Unified Communications Product Support**

- **Troubleshooting Cisco IP Telephony** 
 - **Technical Support – Cisco Systems**
-

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

© 2014 – 2015 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Feb 03, 2006

Document ID: 13940
