

技术支持服务请求的HSI数据收集

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[规则](#)

[标准必要信息](#)

[问题特定信息](#)

[相关电话的问题](#)

[他的应用程序Core dump](#)

[准备文件附件](#)

[信息详细信息](#)

[问题说明](#)

[一般信息](#)

[他的系统信息](#)

[他的当前配置](#)

[他的UNIX监听Trace](#)

[他的应用程序Trace](#)

[PGW思科刺探者\(PTC-MT\) Trace](#)

[Cisco PGW系统信息](#)

[PGW当前配置](#)

[PGW \(MDL\)呼叫跟踪](#)

[H.323终端系统信息](#)

[H.323终端debug命令的输出](#)

[他的内核文件](#)

[pstack和pmap命令输出](#)

[相关信息](#)

简介

当您打开与思科技术支持时的一服务请求，一些初步信息要求改善识别并且合格问题。一些此信息总是要求;其他信息要求取决于问题的本质。如果等待收集此信息，在您打开服务请求和工程师之后自讨苦吃，则不可避免的是将有解决方法的延迟。

因此，本文主要目标是识别需要的初步信息，根据问题种类，因此可以立即提供给工程师。本文辅目标是提供您一般使用指南跟随，当您收集思科技术支持的时信息，避免相同的数据的重复性测试和往事。

本文供支持根据思科H.323信令接口的Cisco用户使用(他的)系统的语音信令解决方案和Cisco PGW 2200 (以前呼叫SC2200和VSC3000或者思科电话控制器或者媒体网关控制器)。

先决条件

要求

本文档的读者应掌握以下这些主题的相关知识：

- [H.323信令接口指南](#)
- [Cisco媒介网关控制器](#)

使用的组件

本文档中的信息根据H.323信令接口(版本2.x或以上)和PGW媒体网关控制器(版本9)。

规则

在本文显示的命令也许看上去加前缀由这些提示符之一，给予应用环境征兆必须执行命令：

提示符	环境
%	UNIX csh-shell提示符。这是默认命令行界面(CLI)提示输入mgcusr UNIX帐户在登录以后。
#~#	UNIX根级别shell提示符。这是默认CLI提示输入root用户。发出su unix命令到那里。
mml>	人机语言(MML)应用程序提示符。发出mml命令从csh-shell提示符到那里。

有关文档规则的详细信息，请参阅 [Cisco 技术提示规则](#)。

标准必要信息

对于所有相关问题，应该立即总是提供此标准的信息给思科技术支持：

1. [问题说明](#)
2. [一般信息](#)
3. [他的系统信息](#)
4. [他的当前配置](#)

问题特定信息

根据问题的本质，其他信息也许是需要。本文考虑这些问题类型：

- [他相关电话的问题](#)
- [他的应用程序core dump](#)

相关电话的问题

如果问题通过介入一呼叫他的，应该提供此信息：

1. [标准必要信息](#)
2. [他的UNIX监听trace](#)
3. [他的应用程序trace](#)

由于他的严密相互作用与PGW，在PGW必须收集此信息：

1. [PGW思科刺探者\(PTC-MT\) trace](#)
2. [PGW系统信息](#)
3. [PGW当前配置](#)
4. [PGW \(MDL\)呼叫跟踪](#)

如果他的与Cisco IOS软件H.323终端谈，应该收集此信息：

1. [H.323终端系统信息](#)
2. [H.323终端debug命令output](#)

注意： 请求的跟踪，并且应该采集调试*同时和为一样的呼叫*这简化事件的相关性区别组件之间的。失败也许导致新，重复的申请信息从技术支持工程师，并且不可避免地请导致解决方法的延迟。

[他的应用程序Core dump](#)

如果他的应用程序遭受core dump，必须提供此信息：

1. [标准必要信息](#)
2. [他的内核文件](#)
3. [pstack和pmap命令输出](#)

[准备文件附件](#)

当您准备将提交的文件对服务请求或对技术支持工程师时，您应该设法压缩(和套件)预先文件。

要简单化此任务，请创建复制相关文件的一个临时子目录。例如，请发出`mkdir /var/tmp/ciscotac`命令做呼叫ciscotac的一个临时子目录。然后，请使用这些方法之一压缩文件：

- 一起被压缩的所有文件：

```
% cd /var/tmp/ciscotac

% tar cf - . | compress > ../files4tac.tar.Z
!--- This method creates one archive file in the parent directory !--- (so that tar fdoes
not archive its own archive) that contains !--- all of the files from /var/tmp/ciscotac. !---
- If you have gzip installed, you may replace compress with gzip.
```

- 单个被压缩的每个文件：

```
% cd /var/tmp/ciscotac

% compress *
!--- If you issue compress (or gzip) by itself, it compresses and !--- replace each file
individually, instead of creating a single !--- archive file. This is useful if the previous
method would result !--- in an archive file that is too large to upload. !--- For core dump
files, always use this method.
```

一旦归档文件或文件提供了给技术支持工程师，请从您的文件系统取消他们(和临时目录)。

[信息详细信息](#)

此部分提供特定和详细的过程关于将被采集的信息。

[问题说明](#)

提供那的逐步细节用户进行的操作，当问题出现时。详细信息应该也包括这些项目：

- 预料之中的行为
- 详细的观察行为
- 如果问题介入一呼叫：呼叫和被叫号码和在呼叫情形也许涉及的任何其他编号呼叫方向(请识别产生的和终止的呼叫信令协议。例如：对H.323的SS7。)
- 被看到的任何错误消息复制和插入的内容
- 问题是否是再现的？
- 什么是问题的频率？
- 行为是否更改基于呼叫方向、软件版本，组件使用的或者别的？换句话说，已知的同一个功能正确地工作，当您使用一组不同的变量时？

[一般信息](#)

提供此一般信息：

- 产品硬件与软件介入的*所有组件*的版本识别。
- 拓扑。这是图形或写入并且应该至少包括在呼叫路径和他们的IP地址涉及的所有组件。
- 网络部署状态：新的安装？这实验室(测验)环境？这生产网络？如果如此：什么时候是第一出现问题？什么最近更改做对组件介入？

[他的系统信息](#)

收集从这些MML命令的输出：

```
mml> rtrv-ne
```

```
mml> rtrv-ne-health
```

发出unix命令的此发现他的补丁程序级别：

```
% ls /opt/GoldWing/currentPM/bin/*main*
```

[他的当前配置](#)

此文本文件包含他的当前配置：

```
% ls /opt/GoldWing/currentPM/bin/*main*
```

或者，您能发出rtrv-config MML命令捕获当前配置。

[他的UNIX监听Trace](#)

来与Solaris捆绑在一起的标准的监听是包嗅探工具。

作为根，在您做一测试呼叫前，请发出此命令：

```
snoop -d interface -o fail.snoop
!--- interface is the relevant interface name and fail.snoop is !--- the file name of the trace
file that you want to write.
```

现在，请做测试呼叫。您应该看到数据包计数上升。普雷斯Ctrl+C，在您结束呼叫后。

提交fail.snoop文件对服务请求或给技术支持工程师(请参阅[PreparingFile附件](#))。

提示：发出/sbin/ifconfig -命令是否对接口名称是不确定的。

[他的应用程序Trace](#)

遵从此步骤捕获应用程序trace到文件。

1. 启用日志通过MML：

```
mm1> set-log:eisup:level=0xffff

mm1> set-log:callcontrol:level=0xffff

mm1> set-log:h323:level=0xffff

mm1> radlog::start

mm1> quit
```

2. 在测验之前倒空platform.log文件：

```
% cd /opt/GoldWing/currentPM/var/log

% log_erase
!--- This command purges the platform.log file. Source the !---
/opt/GoldWing/currentPM/local/setup.gw.csh file, if this !--- command is not recognized.
```

3. 做测试呼叫。

4. 当完成，请保存platform.log文件的复制并且禁用记录日志：

```
% cp platform.log fail.log

mm1> set-log:all:level=0x0000

mm1> radlog::stop

mm1> quit
```

5. 提交fail.log文件对服务请求或给技术支持工程师(请参阅[准备文件附件](#))。

[PGW思科刺探者\(PTC-MT\) Trace](#)

思科刺探者是思科内部包嗅探工具。PTC-MT是刺探者商业化的版本。

要获得在ASCII格式的一trace，请发出这些on命令PGW：

```
# cd snooper_directory

# ./snooper int interface_x ss7 nosnts mgcp noauep eisup detail hex > fail_interface_x.snooper &
!--- This command must be issued on one line. !--- Issue this command for every redundant
interface (interface_x) !--- in the PGW. fail_interface_x.snooper is the file name of the !---
ASCII trace file that you want to write.
```

或者，获得在二进制格式的trace，请发出这些命令：

```
# cd snooper_directory

# ./snooper int interface_x file fail_interface_x.snooper &
!--- Issue this command for every redundant interface (interface_x) !--- in the PGW.
fail_interface_x.snooper is the file name of the !--- binary trace file that you want to write.
```

一旦捕获测试呼叫，请勿忘记结束刺探者进程。

提交fail_interface_x.snooper文件或文件对服务请求或给技术支持工程师(请参阅[PreparingFile附件](#))。

注意：

- 确保seedfile.txt文件正确地配置!
- 如果选择收集一二进制trace，请勿忘记转发seedfile.txt文件。
- 如果使用PTC-MT而不是刺探者，请用ptcmt替换上一个命令的刺探者并且用nomtm替换nosnts。
- 如果不安排刺探者或PTC-MT安装，请使用UNIX监听工具。

[Cisco PGW系统信息](#)

收集从这些MML命令的输出：

```
mml> rtrv-ne
```

```
mml> rtrv-ne-health
```

此unix shell命令显示已安装MGC补丁程序：

```
% pkginfo -i | grep CSC
```

[PGW当前配置](#)

此mml命令导出PGW的当前配置：

```
mml> prov-exp:all:dirname="directory-name"
```

起因于该命令的所有文件在/opt/CiscoMGC/etc/cust_specific/目录名称目录存储。

提交文件对服务请求或给技术支持工程师(请参阅[准备文件附件](#))。

PGW (MDL)呼叫跟踪

应该运行MDL呼叫跟踪一样简要地尽可能。这尽可能低保持在系统性能的潜在影响;并且它对仅相关呼叫也限制呼叫数量在trace的，尽量。因为做复杂化的它找出相关呼叫，在trace的多个呼叫不是理想。

1. 开始呼叫跟踪：

```
mml> sta-sc-trc:sig-path:confirm  
!--- sig-path is the call's incoming signaling path.
```

注意：发出rtrv-dest : all命令确定sig-path。

2. 做测试呼叫。

3. 终止呼叫跟踪：

```
mml> stp-sc-trc:all  
!--- Note the BTR file name that is displayed at this point. mml> quit
```

4. 开始get_trc.sh脚本：

```
% cd /opt/CiscoMGC/var/trace  
  
% get_trc.sh filename.btr  
!--- filename.btr is the file name that was displayed when you !--- stopped the trace.
```

5. 如果多个呼叫是存在trace，首先请导航对相关呼叫ID通过发出N、P或者id命令。

6. 发出C命令写出此呼叫trace，在ASCII格式，到TRC文件。

7. 提交TRC文件对服务请求或给技术支持工程师(请参阅[准备文件附件](#))。

欲了解更详细的信息关于此步骤，参考PGW文档的[跟踪部分](#)。

H.323终端系统信息

从特权模式的内部，请收集这些命令输出：

```
show version
```

```
show running-config
```

如果设备是非Cisco的IOS，请设法获得相似的信息。

H.323终端debug命令的输出

如果系统负载允许它，请收集调试指令下列表的输出—有故障呼叫的。

注意：在您发出这些调试指令前，请确保您安排毫秒时间戳和顺序编号启用在配置里：

```
service timestamps debug datetime msec
service timestamps log datetime msec
service sequence-numbers
```

```
debug cch323 session
```

```
debug cch323 h225
```

```
debug cch323 h245
```

```
debug h225 asn1
```

```
debug h225 asn1 errors
```

```
debug h225 events
```

```
debug h225 q931
```

```
debug h245 asn1
```

```
debug h245 asn1 errors
```

```
debug h245 events
```

如果设备是非Cisco的IOS，请设法获得相似的调试信息。

[他的内核文件](#)

如果他的应用程序失败，core dump写入到有此文件名格式的一个文件：

```
/opt/GoldWing/currentPM/bin/core_timestamp !--- timestamp is in the form YYYYMMDDhhmmss.
```

必须分开压缩内核文件或文件，在您提交他们对服务请求或给技术支持工程师前(请参阅[准备文件附件](#))。

[pstack和pmap命令输出](#)

发出他的内核文件的这些UNIX命令：

```
# cd /opt/GoldWing/currentPM/bin
# ls -l core_*
# pstack core_file > core_file.proc
```

```
# pmap core_file >> core_file.proc  
!--- core_file is the core dump file name that you retrieved !--- with the ls -l core_* command.
```

提交 *core_file.proc* 文件对服务请求或给技术支持工程师(请参阅[PreparingFile附件](#))。

相关信息

- [Cisco PGW 2200 Softswitch技术说明](#)
- [语音技术支持](#)
- [语音和统一通信产品支持](#)
- [Cisco IP 电话故障排除](#)
- [技术支持 - Cisco Systems](#)