语音连接干线故障排除

目录

简介

语音连接中继线永久建立语音呼叫， VoIP、帧中继语音(VoFR)或者ATM语音(VoATM)。呼叫建立，当路由器出现，并且配置完成。当语音端口出现，语音端口自动地拨号假的电话号码指定在语音端口下并且发出呼叫到位置。语音端口完成呼叫对另一端通过对应的拨号对端。一旦此连接被建立，就路由器而言，语音呼叫在会话上和连接。

先决条件

要求

本文档没有任何特定的前提条件。

使用的组件

本文档不限于特定的软件和硬件版本。

本文档中的信息都是基于特定实验室环境中的设备创建的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的l网络实际，请确保您了解所有命令潜在影响，在您使用它前。

规则

有关文档规则的详细信息，请参阅 Cisco 技术提示规则。

问题

适合于到中继的常见问题是透明对路由器和非常难排除故障。在语音中继看到的常见问题表明，当呼叫在中继时发出，并且什么都听不到。这是其中一已知的问题用连接中继和是由许多不同的问题造成的。另一议题是没有适当地通过的双音多频音，并且信令从内部交换机(PBX)到PBX没有适当地传输。本文排除故障这些问题。

当语音卡车是上和活跃的时，信号在连接卡车不同运行。您通常发出在信号特征的语音端口下的任何命令不是相关和有用。语音中继变为信令conduit并且中继在VoIP链路间的信号。当您使用语音中继时， Pbx信令必须匹配端到端。就两台PBX机器而言，目标是使语音中继连接看起来相同与一条租用的T1线路与PBX，用完全透明的路由器，当一条清楚链路建立在两PBX之间在整个进程时。

当中继出现时，中继变为软件电缆，并且信号类型认为连接器类型。中继对使用的信号类型不关心。中继仍然出现，即使信号不配比在两端。只要在两端的PBX执行同样信令，中继正常运行。

解决方案

采取的方法，当您跟那排除故障连接中继问题不同时使用交换呼叫。要看到发生了什么确实，在中继验证后，您需要查找到Pbx信令。在您继续查看信令前，请验证中继上，并且数字信号处理器(DSP)进程语音数据包。

注意： 您很可能要禁用语音活动检测(VAD)为了排除故障。一旦验证中继正确地作用，您需要查看电话信令为了进一步排除故障。

如果中继建立，并且没人设法做呼叫，中继线保活信息反复传送在远程方框之间。这些Keepalive验证中继线连通性并且传播从端到端的信令信息。要验证这些Keepalive，请发出debug vpm signal命令。如果有许多中继，从**debug vpm的**输出发出命令，您能对单个端口限制输出，如果您**debug vpm port x命令**选项的问题，“x”是有问题的语音端口。当您查看所有端口时，这是从发出的**debug vpm signal命令的**输出：

```
21:18:12: [3/0:10(11)] send to dsp sig DCBA state 0x0
21:18:12: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
21:18:12: [3/0:12(13)] rcv from dsp SIG DCBA state 0x0
21:18:12: [3/0:20(21)] rcv from dsp SIG DCBA state 0x0
21:18:12: [3/0:12(13)] send to dsp SIG DCBA state 0x0
21:18:12: [3/0:20(21)] send to dsp SIG DCBA state 0x0
21:18:12: [3/0:0(1)] send to dsp SIG DCBA state 0x0
21:18:12: [3/0:3(4)] rcv from dsp SIG DCBA state 0x0
21:18:12: [3/0:9(10)] rcv from dsp SIG DCBA state 0x0
21:18:12: [3/0:3(4)] send to dsp SIG DCBA state 0x0
21:18:13: [3/0:9(10)] send to dsp SIG DCBA state 0x0
21:18:13: [3/0:19(20)] rcv from dsp SIG DCBA state 0x0
```

如果用*debug vpm port x命令*限制此，更加容易的调试解释，如此示例所显示， ：

```
21:21:08: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
21:21:12: [3/0:0(1)] send to dsp SIG DCBA state 0x0
21:21:13: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
21:21:17: [3/0:0(1)] send to dsp SIG DCBA state 0x0
21:21:18: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
21:21:22: [3/0:0(1)] send to dsp SIG DCBA state 0x0
21:21:23: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
21:21:27: [3/0:0(1)] send to dsp SIG DCBA state 0x0
21:21:28: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
21:21:32: [3/0:0(1)] send to dsp SIG DCBA state 0x0
```

Keepalive被发送并且接收每五秒。从DSP “发送对DSP”和“接收的”期限是从Cisco IOS观点。DSP的替代PBX能使更加可理解。这些是被看到的消息，当没有在中继时的活动。保活信息告诉电路的每个末端的路由器中继仍然是。当五这些消息连续时未命中，中继断开。如果中继在网络，经

常摆动其中一个原因是。要验证语音中继Keepalive是否被发送并且接收，请发出**debug vpm trunk-sc命令**。此调试不生成任何输出，直到中继线保活未命中。当Keepalive未命中时，这是**debug vpm trunk-sc命令**输出的示例：

```
22:22:38: 3/0:22(23): lost Keepalive
22:22:38: 3/0:22(23): TRUNK_SC state : TRUNK_SC_CONN_WO_CLASS, event TRUNK_RTC_LOST_KEEPALIVE
22:22:38: 3/0:22(23): trunk_rtc_set_AIS on
22:22:38: 3/0:22(23): trunk_rtc_gen_pattern : SIG pattern 0x0
22:22:38: 3/0:22(23): TRUNK_SC, TRUNK_SC_CONN_WO_CLASS ==> TRUNK_SC_CONN_DEFAULT_IDLE
22:22:39: 3/0:13(14): lost Keepalive 22:22:39: 3/0:13(14): TRUNK_SC state :
TRUNK_SC_CONN_WO_CLASS, event TRUNK_RTC_LOST_KEEPALIVE 22:22:39: 3/0:13(14): trunk_rtc_set_AIS
on 22:22:39: 3/0:13(14): trunk_rtc_gen_pattern : SIG pattern 0x0 22:22:39: 3/0:13(14): TRUNK_SC,
TRUNK_SC_CONN_WO_CLASS ==> TRUNK_SC_CONN_DEFAULT_IDLE
```

如果输出没有被看到，当debug vpm trunk-sc命令发出时，则Keepalive没有未命中。即使Keepalive未命中，中继坚持，直到五个连续的消息未命中。这意味着连接必须发生故障为25秒，在中继断开前。

连接中继的常见问题

有几个Bug关联与语音中继连接。如果看到异常的任何东西请检查这些Bug。当Cisco IOS软件12.2发布的时候，大多这些问题解决了并且集成。您能通过Bug查找意识自己这些是问题的原因与更旧的代码的。多数常见问题之一是获得PBX在中继连接正确地发信号。它似乎类似一个好想法减少中继和配置路由器，以便他们工作在每个末端，但是方法是确实妨碍达到预期目的，因为任何您当前更改变得需讨论中继一次设立。排除故障的最佳方法是功能的中继上和。

开始排除故障

查看基础设立是必要的这些正确地作用：

- 中继建立？发出**show voice call summary命令**，并且确保中继在S_CONNECTED状态。
- DSP处理数据包？发出**show voice dsp命令**验证此。如果看不到数据包由DSP处理，这是因为VAD启用并且是抑制数据包。关闭VAD，重新建立中继并且再查找。并且，请检查信息包计数器增加，当**show call active voice brief命令**发出时。此命令也显示VAD是否为呼叫登录问题启用。

如果中继连接到模拟端口在任何站点，验证PBX的操作在非中继的模式的是最佳的。要排除故障模拟E&M连接问题，参考了解和排除故障模拟E&M接口类型和布线。一旦一切正确验证并且作用，请提出中继并且查看通过在PBX之间的信令。

理想的方式排除故障语音中继连接问题将检查通过在PBX之间的信令。这是最佳有远程登录会话到有问题的每个的路由器，以便信令可以被观察作为它从一端通过到其他。本文使用E&M闪烁信令，因为是相当普遍的，并且闪烁定时必须被考虑到。

这是从发起呼叫的路由器的输出连接对PBX：

```
May 22 19:39:03.582: [3/0:0(1)] rcv from dsp sig DCBA state 0x0
!--- It is in idle state. May 22 19:39:07.774: [3/0:0(1)] send to dsp SIG DCBA state 0x0 !---
ABCD bits=0000. May 22 19:39:08.586: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22
19:39:12.778: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:13.586: [3/0:0(1)] rcv from
dsp SIG DCBA state 0x0 May 22 19:39:17.777: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22
19:39:18.593: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:22.781: [3/0:0(1)] send to
dsp SIG DCBA state 0x0 May 22 19:39:23.593: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22
19:39:27.781: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:28.597: [3/0:0(1)] rcv from
dsp SIG DCBA state 0x0 May 22 19:39:32.785: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22
19:39:33.597: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:37.789: [3/0:0(1)] send to
```

dsp SIG DCBA state 0x0 May 22 19:39:38.601: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:39.777: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:39.797: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:39.817: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF *!--- Receives off-hook from PBX, and passes to remote end.* May 22 19:39:39.837: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:39.857: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:39.877: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:39.897: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:39.917: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:39.937: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:39.957: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:39.977: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:39.997: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.017: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.037: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.057: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.077: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.089: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.097: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.109: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.117: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF *!--- Receiving wink from remote side, and passes to PBX.* May 22 19:39:40.129: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.137: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.149: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.157: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.169: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.177: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.189: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.197: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.213: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.217: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.229: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.237: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.249: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.257: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.269: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.289: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.309: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.329: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.349: [3/0:0(1)] send to dsp SIG DCBA state 0x0 *!--- Wink ended from remote side, and passes to PBX.* May 22 19:39:40.369: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.389: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.409: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.429: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.449: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.469: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.493: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.509: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.529: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.549: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.569: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.589: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.613: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.629: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.649: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.669: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.689: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.709: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.729: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.749: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:40.769: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:45.773: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:50.081: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:50.101: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:50.121: [3/0:0(1)] send to dsp SIG DCBA state 0xF *!--- Wink ends, the remote end is now off-hook, the conversation happens.* May 22 19:39:50.141: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.161: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.181: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.197: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.221: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.241: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.261: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.261: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.281: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.301: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.321: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.341: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.361: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.381: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.401: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.421: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.441: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.461: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.481: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.501: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.521: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.541: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:50.561: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:55.265: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:55.561: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:00.269: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:40:00.565: [3/0:0(1)] send to

dsp SIG DCBA state 0xF May 22 19:40:05.268: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:40:05.564: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:10.272: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:40:10.568: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:15.276: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:40:15.572: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:19.676: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:19.696: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:19.716: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.736: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.756: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.776: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.796: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.796: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:40:19.816: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.816: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:40:19.836: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.836: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 *!--- Both side hung up, back to idle state.* May 22 19:40:19.856: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.856: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.876: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.876: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.896: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.896: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.916: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.916: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.936: [3/0:0(1)] send to dsp SIG DCBA state 0x0

此输出显示路由器终止呼叫。网络时间协议(NTP)是同步的。

May 22 19:39:03.582: [3/0:0(1)] send to dsp SIG DCBA state 0x0
May 22 19:39:07.774: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
!--- Idle state, both side on-hook. May 22 19:39:08.586: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:12.774: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:13.586: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:15.383: [1/0:0(1)] Signaling RTP packet has no particle *!--- You will see this message if you are running Cisco IOS !--- Software Release 12.2(1a) or later. It is not an error !--- message, it is a normal functioning state.* May 22 19:39:17.774: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:18.590: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:22.778: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:23.594: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:27.782: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:28.598: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:32.782: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:33.598: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:37.786: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:38.602: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:39.778: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:39.798: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:39:39.818: [3/0:0(1)] send to dsp SIG DCBA state 0xF *!--- Remote side off-hook, this is conveyed to the PBX.* May 22 19:39:39.838: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:39.858: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:39.878: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:39.898: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:39.918: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:39.938: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:39.958: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:39.978: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:39.998: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.018: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.038: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.058: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.078: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.090: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.098: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.110: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.118: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.130: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF *!--- Receive wink from PBX.* May 22 19:39:40.138: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.150: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.158: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.170: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.178: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.190: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.198: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.210: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.218: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.230: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.238: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.250: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.258: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:40.270: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.290: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.310: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.330: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:40.350: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 *!--- Wink ended, waiting for an answer.* May 22 19:39:40.370: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.390:

```
[3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.410: [3/0:0(1)] rcv from dsp SIG DCBA
state 0x0 May 22 19:39:40.430: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.450:
[3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.470: [3/0:0(1)] rcv from dsp SIG DCBA
state 0x0 May 22 19:39:40.490: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.510:
[3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.530: [3/0:0(1)] rcv from dsp SIG DCBA
state 0x0 May 22 19:39:40.550: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.570:
[3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.590: [3/0:0(1)] rcv from dsp SIG DCBA
state 0x0 May 22 19:39:40.610: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.630:
[3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.650: [3/0:0(1)] rcv from dsp SIG DCBA
state 0x0 May 22 19:39:40.670: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.690:
[3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.710: [3/0:0(1)] rcv from dsp SIG DCBA
state 0x0 May 22 19:39:40.730: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.750:
[3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:40.770: [3/0:0(1)] rcv from dsp SIG DCBA
state 0x0 May 22 19:39:45.262: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:39:45.770:
[3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:50.077: [3/0:0(1)] rcv from dsp SIG DCBA
state 0x0 May 22 19:39:50.097: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:39:50.117:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF !--- Receive off-hook from PBX. May 22 19:39:50.137:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.157: [3/0:0(1)] rcv from dsp SIG DCBA
state 0xF May 22 19:39:50.177: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.197:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.217: [3/0:0(1)] rcv from dsp SIG DCBA
state 0xF May 22 19:39:50.237: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.257:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.261: [3/0:0(1)] send to dsp SIG DCBA
state 0xF May 22 19:39:50.277: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.297:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.317: [3/0:0(1)] rcv from dsp SIG DCBA
state 0xF May 22 19:39:50.337: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.357:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.377: [3/0:0(1)] rcv from dsp SIG DCBA
state 0xF May 22 19:39:50.397: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.417:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.437: [3/0:0(1)] rcv from dsp SIG DCBA
state 0xF May 22 19:39:50.457: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.477:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.497: [3/0:0(1)] rcv from dsp SIG DCBA
state 0xF May 22 19:39:50.517: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.537:
[3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:39:50.557: [3/0:0(1)] rcv from dsp SIG DCBA
state 0xF !--- Both sides off-hook, the conversation happens. May 22 19:39:55.265: [3/0:0(1)]
send to dsp SIG DCBA state 0xF May 22 19:39:55.557: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF
May 22 19:40:00.269: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:00.561: [3/0:0(1)]
rcv from dsp SIG DCBA state 0xF May 22 19:40:05.269: [3/0:0(1)] send to dsp SIG DCBA state 0xF
May 22 19:40:05.561: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:40:10.273: [3/0:0(1)]
send to dsp SIG DCBA state 0xF May 22 19:40:10.565: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF
May 22 19:40:15.273: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:15.569: [3/0:0(1)]
rcv from dsp SIG DCBA state 0xF May 22 19:40:19.673: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF
May 22 19:40:19.693: [3/0:0(1)] rcv from dsp SIG DCBA state 0xF May 22 19:40:19.713: [3/0:0(1)]
rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.733: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
May 22 19:40:19.753: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.773: [3/0:0(1)]
rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.793: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0
May 22 19:40:19.797: [3/0:0(1)] send to dsp SIG DCBA state 0xF May 22 19:40:19.813: [3/0:0(1)]
rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.817: [3/0:0(1)] send to dsp SIG DCBA state 0xF
May 22 19:40:19.833: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.837: [3/0:0(1)]
send to dsp SIG DCBA state 0x0 !--- Both sides are back on-hook, back to idle. May 22
19:40:19.853: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22 19:40:19.857: [3/0:0(1)] send to
dsp SIG DCBA state 0x0 May 22 19:40:19.873: [3/0:0(1)] rcv from dsp SIG DCBA state 0x0 May 22
19:40:19.877: [3/0:0(1)] send to dsp SIG DCBA state 0x0 May 22 19:40:19.893: [3/0:0(1)] rcv from
dsp SIG DCBA state 0x0 May 22 19:40:19.897: [3/0:0(1)] send to dsp SIG DCBA state 0x0
```

注意： 此输出显示发生在语音中继的两边的信令哪些使用E&M闪烁信令。使用这些同样调试信令的其他类型能被看到。如果看到正确地建立的呼叫(如显示此处)，双向音频一定存在。如果查看**show voice dsp**或**show call active voice brief命令**输出，这可以验证。如果一切查找是细致的那里，并且获得音频问题(没有音频或单向)与模拟连接，再请检查这些连接。

确定什么呼叫是UP

因为它做的很少或无益查看**show call active voice**或**show voice call summary命令**输出为中继的呼叫，您需要简单方法确定哪些语音中继支持激活的呼叫。其中一个最简单的方法执行此是发出**show**

voice trunk-conditioning signaling命令与*包括*参数一道和使用*ABCD*作为包括的字符串，如此处所显示：

```
Phoenix#show voice trunk-conditioning signaling | include ABCD last-TX-ABCD=0000, last-RX-
ABCD=0000 last-TX-ABCD=0000, last-RX-ABCD=0000 last-TX-ABCD=0000, last-RX-ABCD=0000 last-TX-
ABCD=0000, last-RX-ABCD=0000 last-TX-ABCD=0000, last-RX-ABCD=0000 last-TX-ABCD=0000, last-RX-
ABCD=0000 last-TX-ABCD=0000, last-RX-ABCD=0000 last-TX-ABCD=1111, last-RX-ABCD=0000 !---
Timeslot 8. last-TX-ABCD=0000, last-RX-ABCD=0000 last-TX-ABCD=1111, last-RX-ABCD=1111 !---
Timeslot 10. last-TX-ABCD=0000, last-RX-ABCD=0000 last-TX-ABCD=0000, last-RX-ABCD=0000 last-TX-
ABCD=0000, last-RX-ABCD=0000 last-TX-ABCD=0000, last-RX-ABCD=0000 last-TX-ABCD=0000, last-RX-
ABCD=0000 last-TX-ABCD=0000, last-RX-ABCD=0000
```

注意： 此输出显示在时隙10和在时隙开始的另一呼叫的一活动呼叫八。您要做此相当长命令的一别名是否使用它很多。

DTMF排除故障

除摘机和挂机信令外，唯一路由器通过在PBX之间的其他事(除语音以外)是DTMF音。也有音频路径，因此这通常不是问题，但是有问题。问题发生在您如何执行在该路径的音频。使用低比特率编码为了保存带宽是有时更可取的。问题出来这些低比特率编码通过为人的语音写入的算法设计。除非客户使用g711编码， DTMF音不很好依照这些算法并且需要某个其他方法转达。答案在**dtmf-relay命令**在。此功能允许DSP在末端，开始音，认可DTMF音和从正常音频流分离它。基于它如何配置， DSP然后编码此音作为不同种实时协议(RTP)数据包或，当在链路间将分开发送的h245消息与音频流。这是在**传真中继**和**调制解调器中继**命令后的同一进程。

此功能提出Trunk故障排除的另一调试问题。如何验证什么位通过，如果没有呼叫建立，并且必须解压缩从信息包流的该信息在路由器之间？如何执行此取决于什么类型**dtmf-relay命令**被使用。

如此示例所显示， dtmf-relay cisco-rtp命令，使用一种所有权Cisco有效载荷类型，因此您必须查看下来DSP发现此。您能与**debug vpm port x/x一道**发出**debug vpm signal命令：y.z命令**(对有问题的端口限制输出)发现位通过对DSP在始发端。此输出显示在始发端，不在终止端。

```
*Mar  1 00:22:39.592: htsp_digit_ready: digit = 31
*Mar  1 00:22:39.592: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:40.021: htsp_digit_ready: digit = 32
*Mar  1 00:22:40.021: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:40.562: htsp_digit_ready: digit = 33
*Mar  1 00:22:40.562: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:40.810: [1/0:1(2)] rcv from dsp SIG DCBA state 0xF
*Mar  1 00:22:41.131: htsp_digit_ready: digit = 34
*Mar  1 00:22:41.131: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:41.499:  [1/0:1(2)] Signaling RTP packet has no partical
*Mar  1 00:22:41.499: [1/0:1(2)] send to dsp SIG DCBA state 0xF
*Mar  1 00:22:41.672: htsp_digit_ready: digit = 35
*Mar  1 00:22:41.672: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:42.192: htsp_digit_ready: digit = 36
*Mar  1 00:22:42.192: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:42.789: htsp_digit_ready: digit = 37
*Mar  1 00:22:42.789: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:43.350: htsp_digit_ready: digit = 38
*Mar  1 00:22:43.350: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:44.079: htsp_digit_ready: digit = 39
*Mar  1 00:22:44.079: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:45.249: htsp_digit_ready: digit = 30
*Mar  1 00:22:45.249: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:45.810: [1/0:1(2)] rcv from dsp SIG DCBA state 0xF
*Mar  1 00:22:46.007: htsp_digit_ready: digit = 2A
*Mar  1 00:22:46.011: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:46.572: [1/0:1(2)] Signaling RTP packet has no partical
```

```
*Mar  1 00:22:46.572: [1/0:1(2)] send to dsp SIG DCBA state 0xF
*Mar  1 00:22:46.628: htsp_digit_ready: digit = 23
*Mar  1 00:22:46.628: [1/0:1(2), S_TRUNKED, E_VTSP_DIGIT]
*Mar  1 00:22:50.815: [1/0:1(2)] rcv from dsp SIG DCBA state 0xF
all digits 0-9 are represented by 30-39, * = 2A and # = 23.
```

您能验证什么位从始发端发送用dtmf-relay h245-alphanumeric命令。dtmf-relay h245-alphanumeric命令使用h.245的字母数字部分表达音。如此示例所显示，当debug h245 asn1命令启用时，位能容易地被看到在产生和中继的终止端：

始发端：

```
*Mar  1 00:34:17.749: H245 MSC OUTGOING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric : "1"

*Mar  1 00:34:17.749: H245 MSC OUTGOING ENCODE BUFFER::= 6D 400131
*Mar  1 00:34:17.753:
*Mar  1 00:34:18.350: H245 MSC OUTGOING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric : "2"

*Mar  1 00:34:18.350: H245 MSC OUTGOING ENCODE BUFFER::= 6D 400132
*Mar  1 00:34:18.350:
*Mar  1 00:34:18.838: H245 MSC OUTGOING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric : "3"

*Mar  1 00:34:18.838: H245 MSC OUTGOING ENCODE BUFFER::= 6D 400133
```

终止端：

```
*Mar  1 17:45:16.424: H245 MSC INCOMING ENCODE BUFFER::= 6D 400131
*Mar  1 17:45:16.424:
*Mar  1 17:45:16.424: H245 MSC INCOMING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric : "1"

*Mar  1 17:45:17.025: H245 MSC INCOMING ENCODE BUFFER::= 6D 400132
*Mar  1 17:45:17.025:
*Mar  1 17:45:17.025: H245 MSC INCOMING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric : "2"

*Mar  1 17:45:17.514: H245 MSC INCOMING ENCODE BUFFER::= 6D 400133
*Mar  1 17:45:17.514:
*Mar  1 17:45:17.514: H245 MSC INCOMING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric : "3"
```

当调试象dtmf-relay h245-alphanumeric命令使用时， dtmf-relay h245-signal命令是非常类似的，并且能被看到同样。总之，排除故障连接中继用dtmf-relay命令是相当困难没有被提及的调试。

相关信息

- 透明 CCS 的配置与故障排除
- 语音技术支持
- 语音和 IP 通信产品支持
- Cisco IP 电话故障排除
- 技术支持 - Cisco Systems