在语音和视频呼叫的Wireshark中破译RTP流以进行丢包分析

目录

[简介](#)
[问题](#)

简介

本文档介绍如何在Wireshark中解密用于语音和视频呼叫的丢包分析的实时流(RTP)流的过程。您可以使用Wireshark过滤器来分析在呼叫源和目标位置或靠近呼叫源和目标位置捕获的同步数据包。当怀疑网络丢失时，您必须排除音频和视频质量问题时，此功能非常有用。

问题

本示例使用以下呼叫流：

IP电话A（中心站点A）> 2960交换机>路由器>广域网路由器（中心站点）> IPWAN >广域网路由器（站点B）>路由器> 2960 > IP电话B

在此场景中，遇到的问题是，从IP电话A到IP电话B的视频呼叫会导致从中心站点A到分支站点B的视频质量不佳，其中中心站点质量良好，但分支站点有问题。

请参阅分支IP电话的流统计信息中的接收方丢失的数据包：

Streaming Statistics

Cisco IP Phone CP-8941(SEP00077ddfbe65)

Device Information
Network Setup
Network Statistics
Ethernet Information
Network
Device Logs
Console Logs
Core Dumps
Status Messages
Debug Display
Streaming Statistics
Stream 1
Stream 2

Remote Address	192.168.10.146/20568
Local Address	192.168.207.231/20808
Start Time	00:00:00
Stream Status	Not Ready
Host Name	SEP00077ddfbe65
Sender Packets	4745
Sender Octets	3144928
Sender Codec	H264
Sender Reports Sent	16
Sender Report Time Sent	11:19:34
Rcvr Lost Packets	199
Avg Jitter	40
Rcvr Codec	H264
Rcvr Reports Sent	1
Rcvr Report Time Sent	11:18:14
Rcvr Packets	4675
Rcvr Octets	3113320
MOS LQK	0.0000
Avg MOS LQK	0.0000
Min MOS LQK	0.0000
Max MOS LQK	0.0000
MOS LQK Version	0.9500
Cumulative Conceal Ratio	0.0000
Interval Conceal Ratio	0.0000
Max Conceal Ratio	0.0000
Conceal Secs	0
Severely Conceal Secs	0
Latency	389
Max Jitter	50
Sender Size	0 ms

解决方案

只有在分支机构端才会看到质量不佳，而且由于中心站点看到的图像良好，从中心到分支站点的数据流似乎正在网络中丢失数据包。

```
IP addressing scheme
Central IP phone: 192.168.10.146
Central Gateway: 192.168.10.253
```

```
Central WAN router: 192.168.10.254
Branch WAN router: 192.168.206.210
Branch Gateway: 192.168.206.253
Branch IP phone: 192.168.207.231
```

数据包捕获在中央和分支WAN路由器上进行，WAN会丢弃这些数据包。重点介绍从中央IP电话(192.168.10.146)到分支IP电话(192.168.207.231)的RTP流。 如果WAN丢弃从中央WAN路由器到分支WAN路由器的数据流上的数据包，则此数据流会丢失分支WAN路由器上的数据包。使用wireshark中的过滤器选项来隔离问题：

1. 在wireshark中打开捕获。

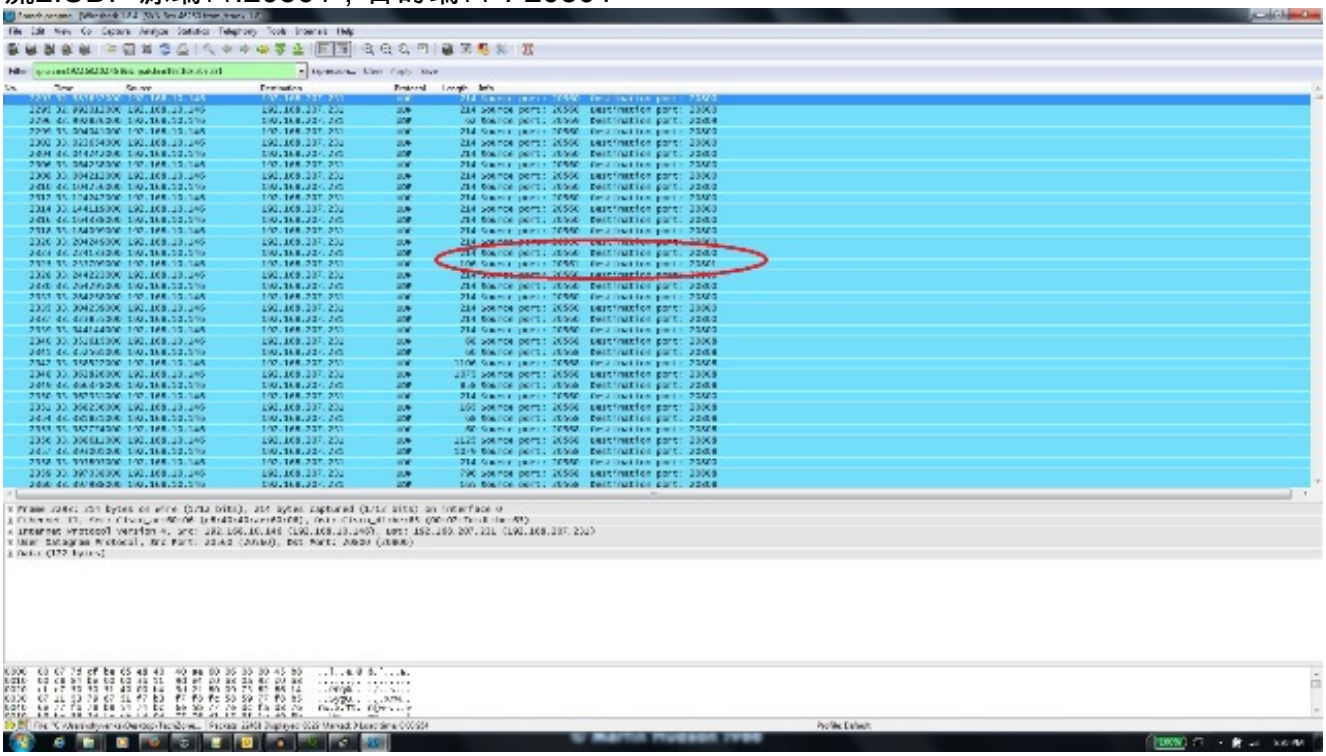
2. 使用过滤器ip.src==192.168.10.146 && ip.dst==192.168.207.231。这会过滤掉从中央IP电话到分支IP电话的所有UDP流。

3. 仅对分支机构端捕获执行分析，但请注意，对于中央捕获，您也必须执行这些步骤。

4. 在此屏幕截图中，UDP流在源IP地址和目的IP地址之间过滤，并包含两个UDP流（由UDP端口号区分）。 这是视频呼叫，因此有两个流：音频和视频。在本例中，两个流是：

   流1:UDP 源端口:20560，目的端口：20800

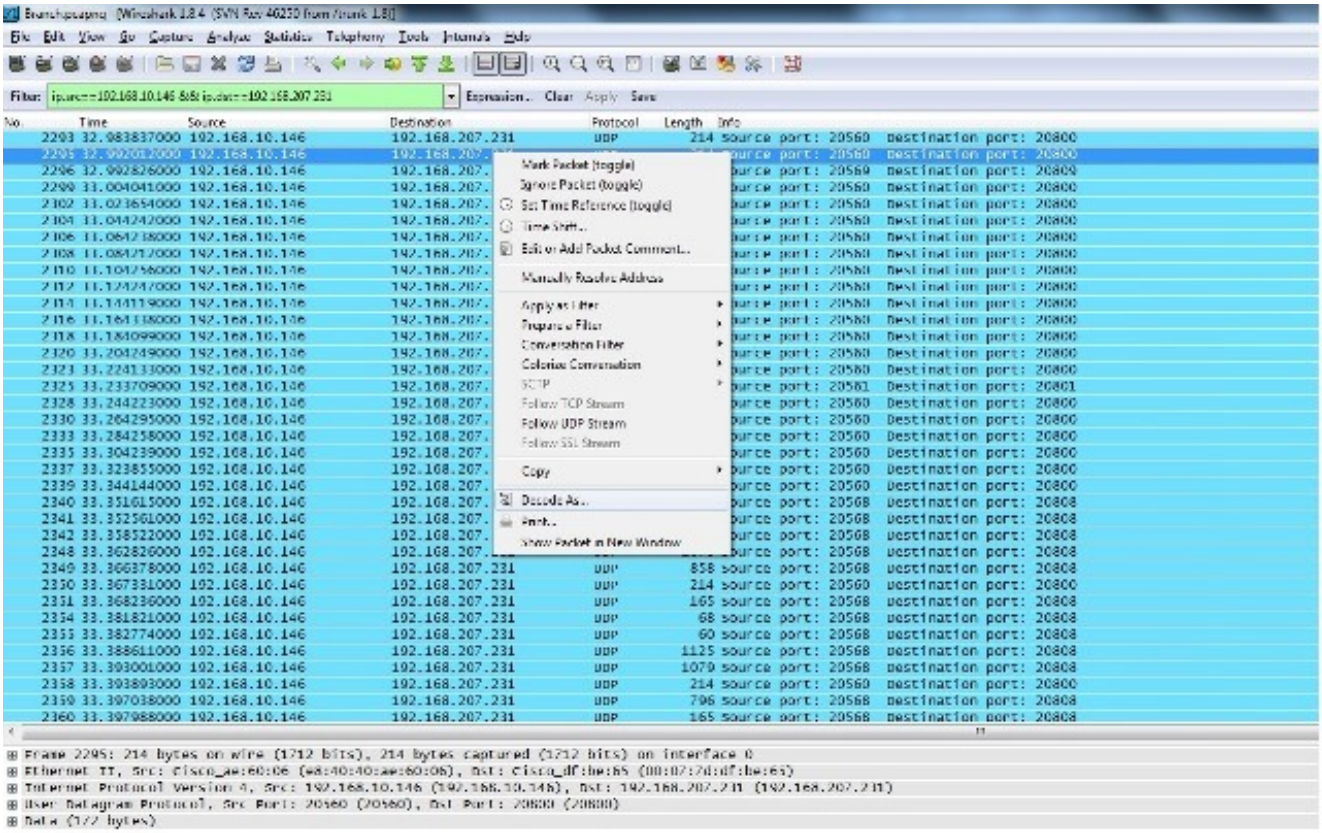
   流2:UDP 源端口:20561，目的端口：20801



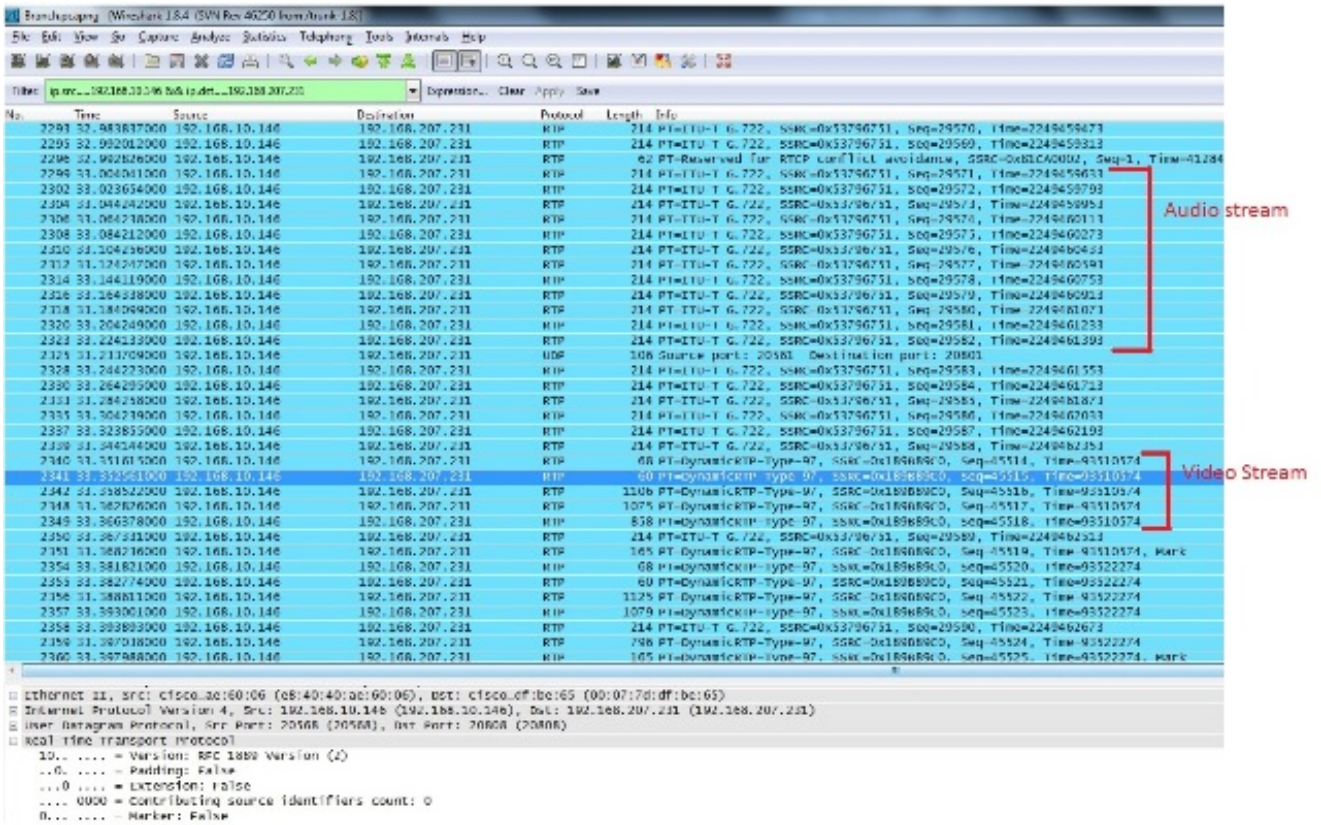
5. 从其中一个流中选择一个数据包，然后右键单击该数据包。

6. 选**择Decode As...并**键入**RTP**。

7. 单击**Accept**和**Ok**以将流解码为RTP。

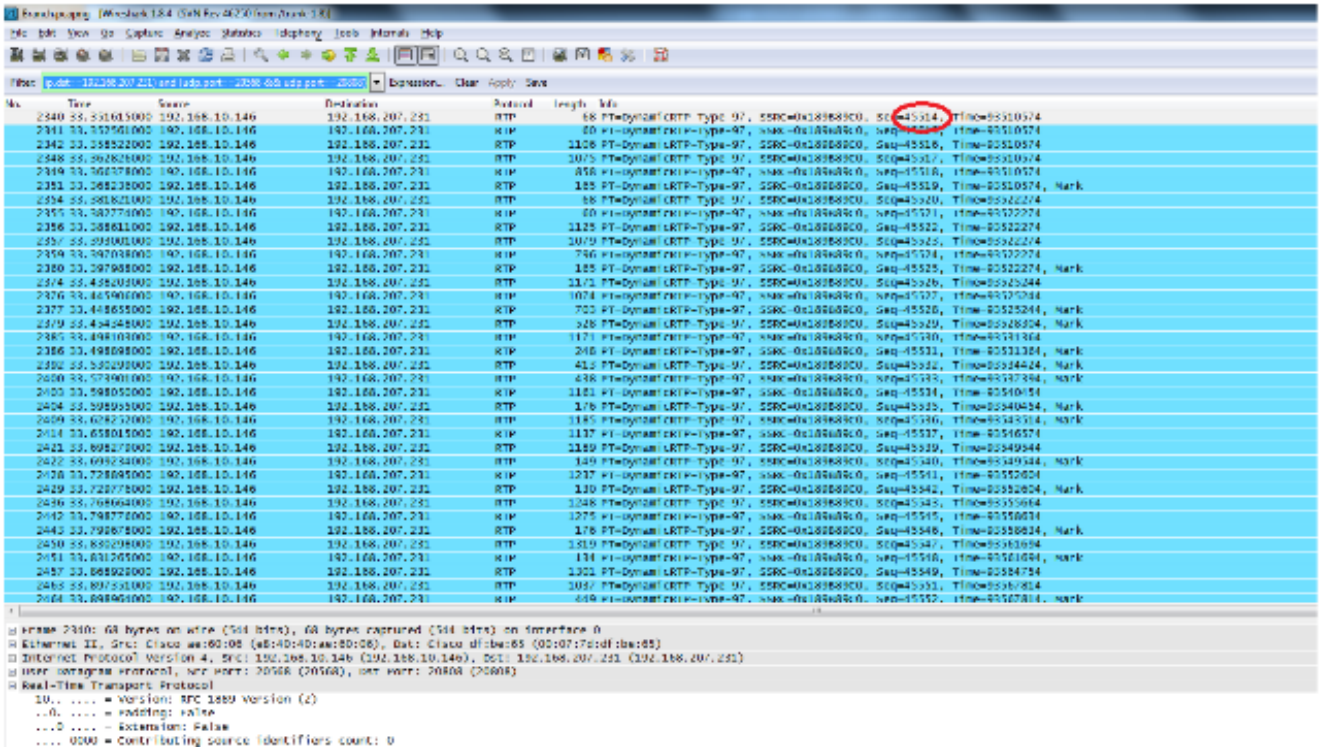


您的一个数据流被解码为RTP，另一个被解码为未解码的UDP。



8. 从未解码的流中选择数据包并将其解码为RTP。这会将音频和视频流解码为RTP。

注意：音频流采用G.722编解码器格式，Dynamic-RTP-97负载类型指示视频RTP流。

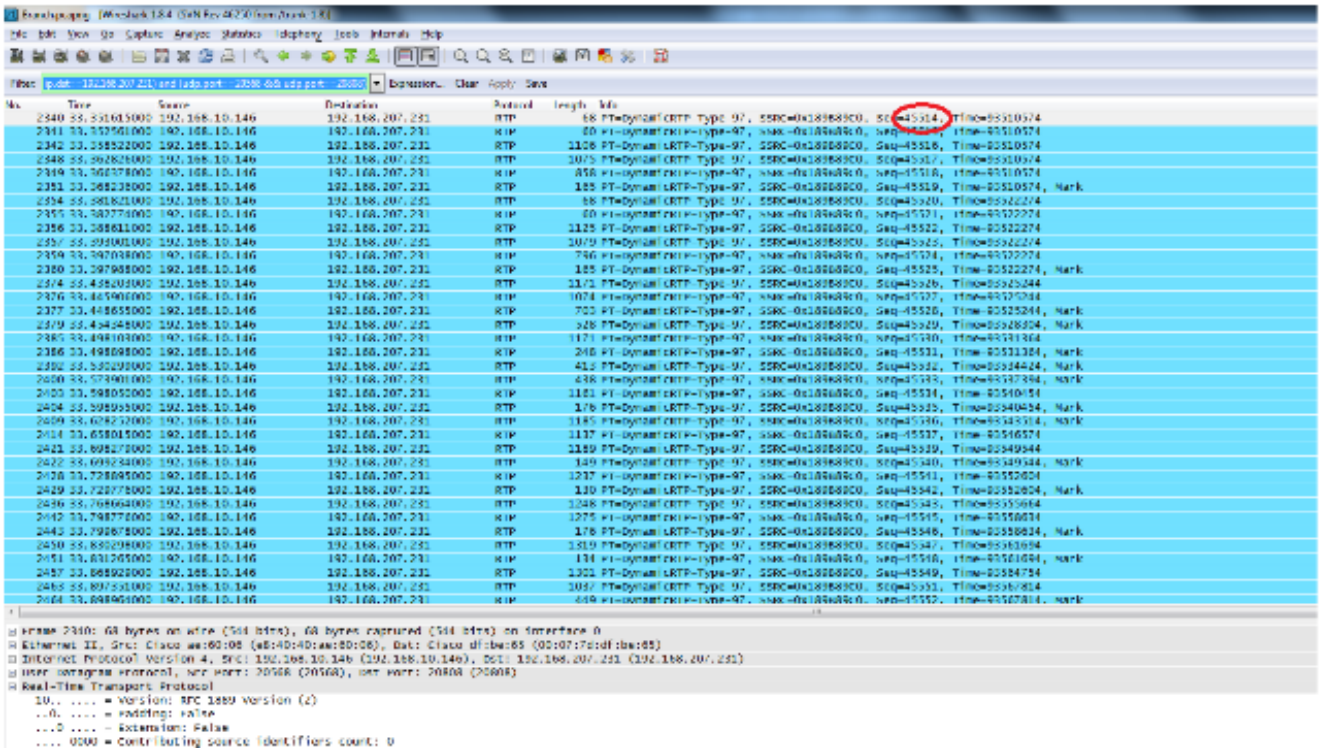


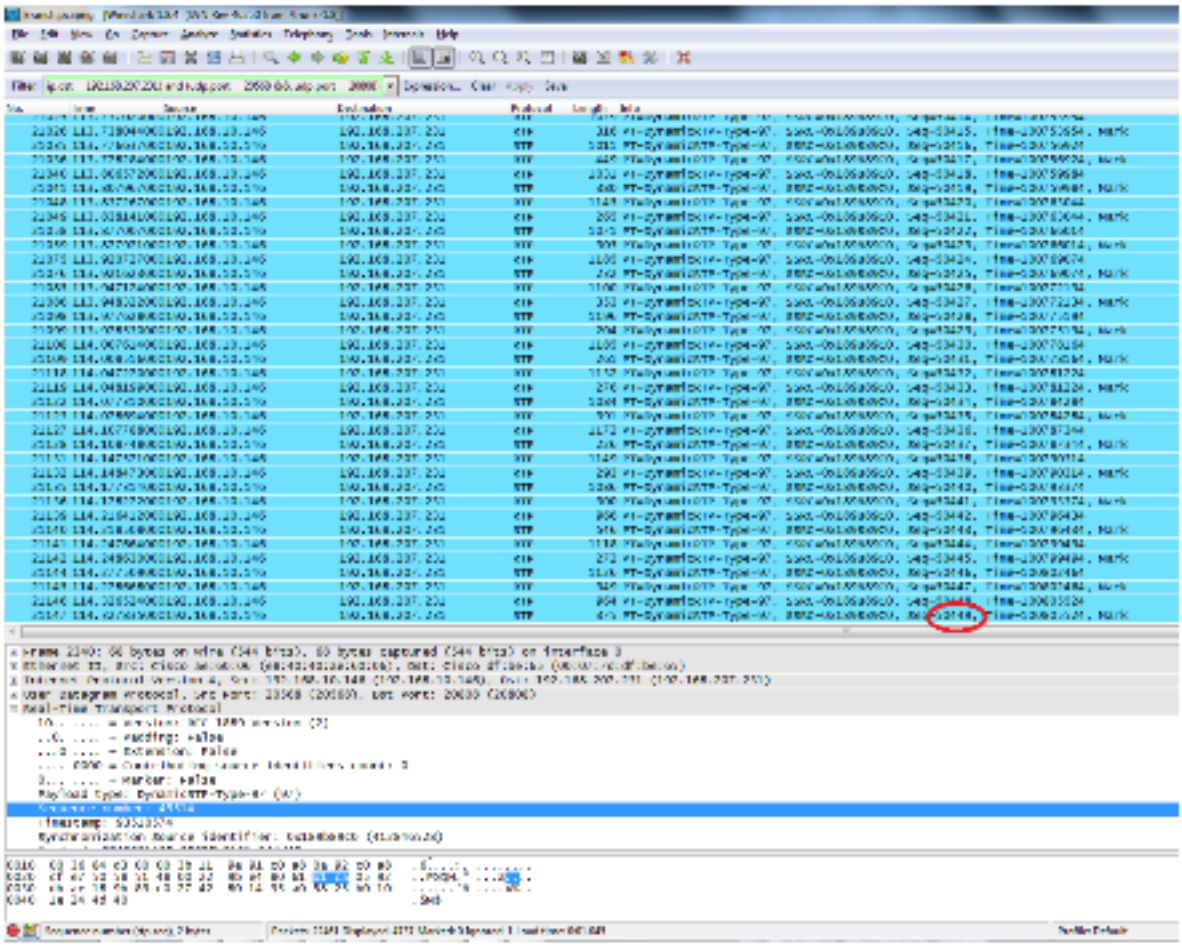
现在问题只出在视频质量上。关注视频RTP流，并使用此流的UDP端口号过滤掉其他流。

9. 在Wireshark实用程序的底部窗格中选择显示UDP端口信息的数据包之一，查看端口号。在上一屏幕截图中，从视频流中选择了其中一个数据包，您可以在底部窗格中看到源端口(20568)和目的端口(20808)信息。

提示：使用此过滤器：(ip.src==192.168.10.146 && ip.dst==192.168.207.231)&(udp.port eq 20568和udp.port eq 20808)。 您只会看到此屏幕截图中显示的视频RTP流。

注意：写下此流的第一个和最后一个RTP序列号。





第一个RTP序列号45514，最后一个是50449，用于已过滤的视频RTP流。

10. 确保第一个和最后一个RTP序列号数据包同时存在于两个捕获中。例如，中心和分支捕获），并注意流的SSRC在两个捕获上将相同。
11. 优化过滤器以仅匹配第一个和最后一个RTP流之间的数据包。

    序列号用于细化数据流，以防捕获不同时进行，但两者之间有轻微延迟。

    注意：分支站点可能会在45514后启动一些序列号。

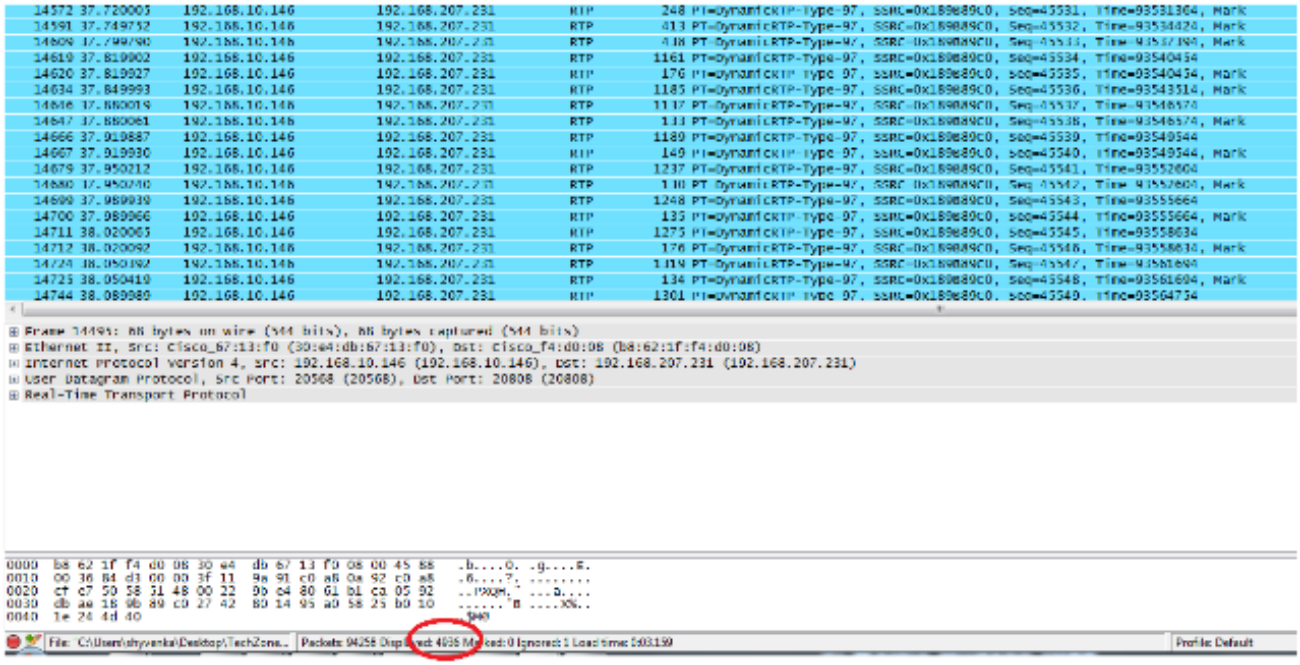
12. 选择开始和结束序列号。这些数据包在捕获和优化过滤器中都存在，以仅显示开始和结束RTP序列号之间的数据包。此项的过滤器为：

```
(ip.src==192.168.10.146 && ip.dst==192.168.207.231) && (udp.port eq 20568
and udp.port eq 20808) && ( rtp.seq>=44514 && rtp.seq<=50449 )
```

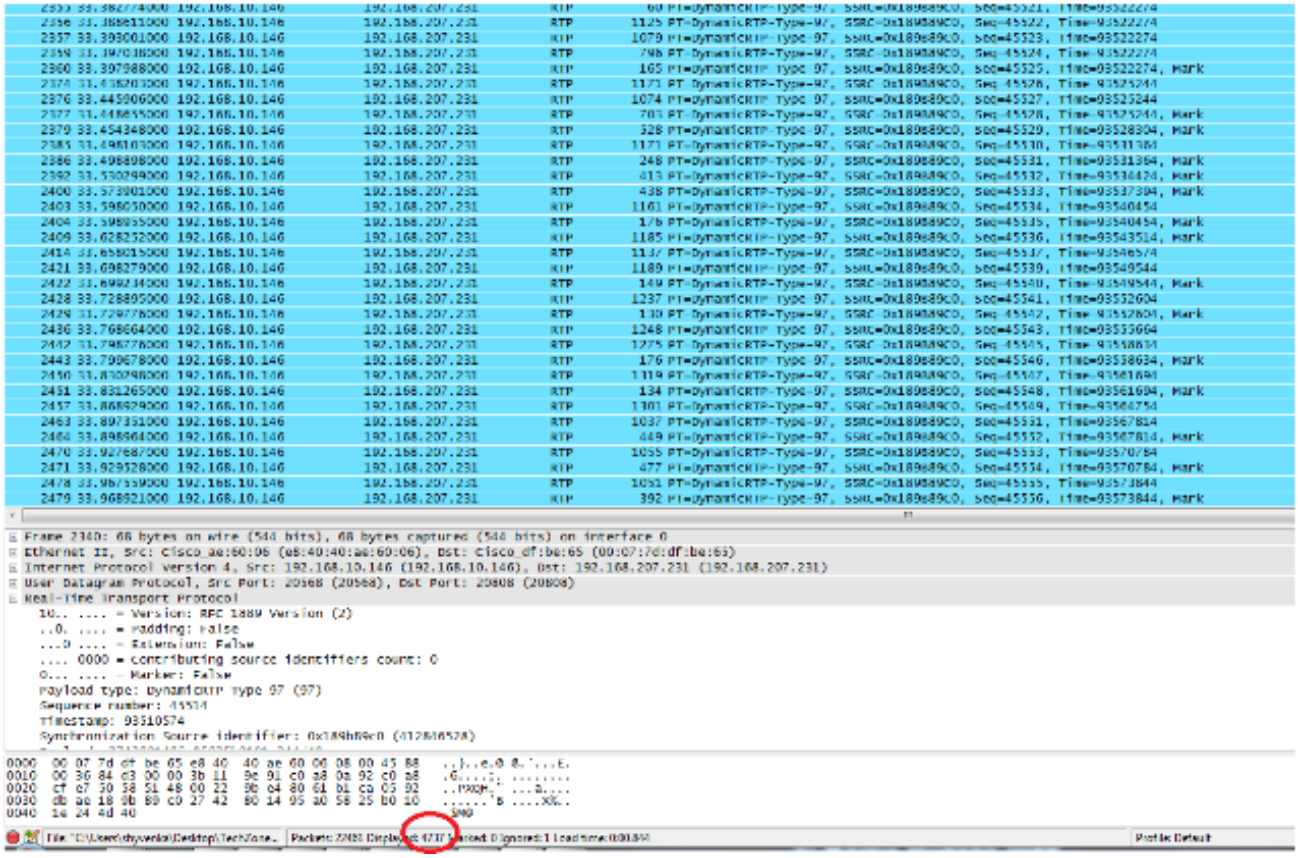
    当同时捕获时，两个捕获的开始或结束时不会丢失任何数据包。如果您看到其中一个捕获在开始/结束时不包含几个数据包，请使用两个数据包中丢失的捕获的第一个序列号或捕获中的最后一个序列号来细化两个捕获的过滤器。观察在同一序列号（RTP序列号范围）之间两点捕获的数据包。

    应用过滤器时，您会在中心站点和分支站点看到以下内容：

中心站点:



分支站点:



注意Wireshark实用程序底部窗格中两个捕获的过滤数据包计数。**Displayed** 计数指示符合所需过滤条件的数据包数。

中心站点有4,936个数据包，这些数据包在开始(45514)和结束(50449)RTP序列号之间符合所需的过滤条件，而在分支站点只有4,737个数据包。这表示丢失了199个数据包。请注意，这199个数据包与“Rcvr Lost Pkts”计数199匹配，该计数在本文档开头显示的分支机构端IP电话

的流统计信息中可见。

这确认了所有接收器丢失数据包实际上是广域网上丢弃的网络丢失。这就是处理涉及可疑网络丢弃的音频/视频质量问题时隔离网络中丢包点的方式。