📋 Report on RTP Payload Discrepancy Leading to Call Silence

🔲 1. Overview
This report examines an issue involving Real-time Transport Protocol (RTP) streams exchanged between the source IP 192.10.1.65 and the destination IP 208.93.9.179. The primary issue identified is the transmission of an oversized payload, which deviates from the expected standard, causing periods of silence during the call.

❗ 2. Issue Summary

The client was expected to send an RTP payload with a Payload size of 20 ms. However, the captured data shows that the payload size transmitted was 30 ms. This inconsistency directly contributed to silence in the call, as the Vega system was unable to properly process the larger-than-expected payload.

🔄 3. Call Flow Analysis
Below is a summary of the key events from the SIP and RTP flows:

The issue was compounded by the presence of multiple ongoing calls during the RTP capture. Here is a summary of the key events:

- SIP INVITE from 192.10.1.65 to 208.93.9.179 initiated the call.
- The session was established successfully with a 200 OK response, and RTP streams began flowing between the two endpoints.
- RTP streams with SSRC 0x19A85265 and SSRC 0x4D3C8D43 were exchanged between 192.10.1.65:10044 and 208.93.9.179:24342.
- The capture revealed two RTP streams operating on different ports (10040 and 10044), which were associated with different calls.
- The SIP INVITE for one of the calls was missing during the capture, as the packet capture began after the invite for the other call was sent, which led to the lack of invite data for the second call.

📊 4. RTP Analysis
A review of the RTP packets revealed the following key details:

Source Address	Source Port	Destination Address	Destination Port	Start Time	Duration (s)	Payload	Packets	Lost	Min Delta (ms)	Mean Delta (ms)	Lost %	Status	SSRC formatted
192.10.1.65	10044	208.93.9.179	24342	55.15717	38.651138	g711U	1285	2	29.776	30.102	0.16	Problem	0x19a8526
192.10.1.65	10040	208.93.9.179	42138	4.19609	32.870571	g711U	1094	1	29.224	30.074	0.09	Problem	0xc697f2be
208.93.9.179	42138	192.10.1.65	10040	4.18013	32.920705	g711U	1645	2	16.118	20.025	0.12	Problem	0xed910bd
208.93.9.179	24342	192.10.1.65	10044	55.21338	38.659701	g711U	1933	1	18.299	20.010	0.05	Problem	0x4d3c8d4

🥷 5. Root Cause Analysis
The root cause of the problem is the deviation in the RTP payload size. The client was expected to transmit a 20 ms payload but instead transmitted a 30 ms payload. The Vega system, which is configured to expect a 20 ms payload, could not handle the larger payload size. This caused momentary silences during playback as Vega had to buffer or drop excess data.

🔍 6. Technical Explanation
RTP payload size impacts the timing of audio packetization. When the payload size exceeds the expected duration, the receiving system's jitter buffer may overflow or be unable to decode the payload properly. This misalignment results in call silences or audio clipping. The following discrepancies were observed:

- **SSRC 0x19A85265**: Expected 20 ms, but received 30 ms, impacting 1285 packets.
- **SSRC 0x4D3C8D43**: Same discrepancy of 30 ms vs. 20 ms, affecting 1933 packets.

📉 7. Impact Analysis
The increased payload size affected the RTP streams' consistency and call quality. End users likely experienced silence or choppy audio during the call. The issue was further exacerbated by packet loss (up to 0.16%) on certain streams, compounding the impact on call quality.

💡 8. Recommendations

1. **Payload Configuration**: Ensure that the clients RTP payload size is correctly set to 20 ms.
2. **System Alerts**: Implement monitoring alerts to detect when RTP payload size exceeds the configured threshold.
3. **Vega Configuration**: Verify that the Vega system's jitter buffer can accommodate slight deviations in payload size, if possible.

☑️ 9. Conclusion
The analysis concludes that the silence observed in the RTP stream was due to a mismatch between the expected and actual RTP payload sizes. Corrective actions, including proper payload size configuration and enhanced monitoring, are recommended to prevent future occurrences of this issue.

📋 Files and Resources

Wireshark Capture (.cap)

- Review the Wireshark capture for detailed analysis:
  Cloudshark Wireshark Capture

🔍 Live Packet Analysis in MDX using CloudShark

Wireshark Capture (.cap)