

WxCC Agent DesktopでのWebRTCデータと統計情報の確認

内容

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[概要](#)

[クロム](#)

はじめに

このドキュメントでは、組み込みのブラウザツールを使用して使用可能なWebRTCコールデータと統計情報について説明します。

前提条件

要件

Webex Contact Center(WxCC)ソリューション、Chrome、Edge、Firefoxブラウザ、およびWebRTCプロジェクトに関するスキルと知識があることが推奨されます。

使用するコンポーネント

このドキュメントの内容は、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

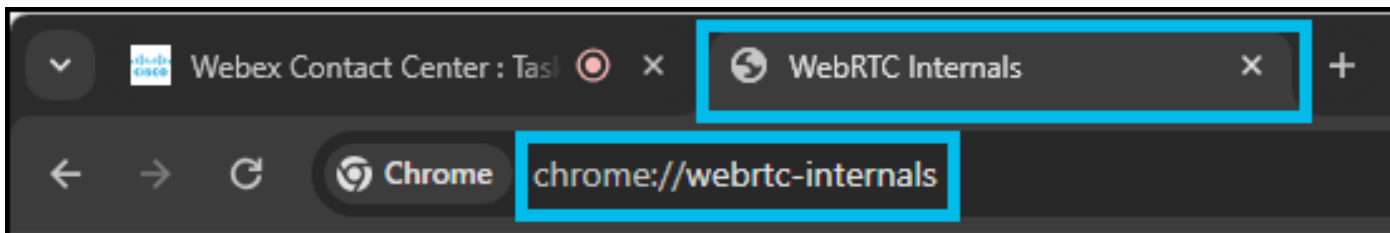
このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな(デフォルト)設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

概要

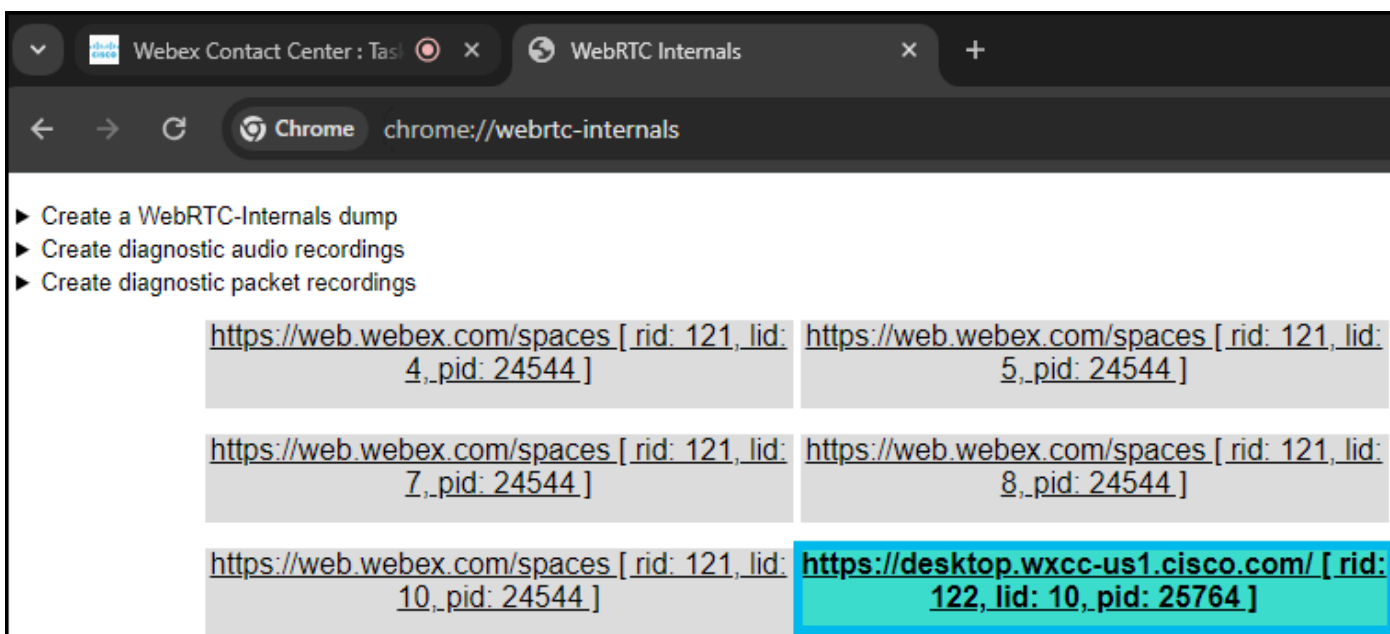
WebRTCをサポートするほとんどのWebブラウザには、WebRTC関連の問題のトラブルシューティングに役立つWebRTCツールが組み込まれています。

クロム

WebRTCのコールデータおよび統計情報にアクセスするには、アクティブコール中にGoogle Chrome Webブラウザの新しいタブを開き、次のURLを使用する必要があります。
chrome://webrtc-internals にアクセスしてください。



WebRTC-internalsタブで、アクティブなWebRTCセッションを選択する必要があります(セッション名はdesktop.wxcc-us1.cisco.com。URLはWxCC領域によって異なります)。



次の情報が表示されます。

- 左側の(1)で、PeerConnection APIアクティビティのレコードを使用できます。RTCPeerConnectionは、WebRTC APIの中央インターフェイスです。ローカルピアとリモートピア間の接続を表し、接続の確立に必要なすべての機能とイベントを提供します(詳細については、<https://webrtc.org/getting-started/overview>を参照してください)。次のトレース(1)では、RTCPeerConnectionオブジェクトとのインタラクションおよび使用されているパラメータが表示されています。
- 右側の(2)では、現在のWebRTC接続に関してgetStats APIを介して収集されたメトリックを使用できます。
- 下の(3)は、現在のWebRTC接続に関してgetStats API (2)を利用して生成されたグラフを示しています。

https://desktop.wccx-us1.cisco.com/, { iceServers: [], iceTransportPolicy: all, bundlePolicy: max-compat, rtcMuxPolicy: require, iceCandidatePoolSize: 0 },

1
ICE connection state: new => completed
Connection state: new => connected
Signaling state: new => stable
ICE Candidate pair: 10.24.139.247:49870 <=>23.89.1.162:22744
▶ ICE candidate grid

Time	Event
4/17/2024, 10:13:52 AM	▶ setRemoteDescription (type: "offer", 2 sections)
4/17/2024, 10:13:52 AM	▶ setLocalDescription (type: "answer", 2 sections)
4/17/2024, 10:13:52 AM	▶ signalingstatechange
4/17/2024, 10:13:52 AM	▶ iceconnectionstatechange
4/17/2024, 10:13:52 AM	▶ connectionstatechange

2
Stats Tables

Filter statistics by type including

- ▶ media-playout (kind=audio, id=AP)
- ▶ certificate (id=CFE0 AA E8 8D CE FA A0 0E 63 4B 6C 16 C3 D0 AD C3 7B 2B A7 EE)
- ▶ codec (mimeType=audio/PCMU, payloadType=0, id=CIT01_0)
- ▶ codec (mimeType=audio/PCMU, payloadType=0, id=COT01_0)
- ▶ candidate-pair (state=succeeded, id=CPyDiSaDsy_T5aV8Gj8)
- ▶ local-candidate (candidateType=host, id=1toYJvKsp)
- ▶ local-candidate (candidateType=host, tcpType=active, id=I9OkJn/oQ)
- ▶ local-candidate (candidateType=host, tcpType=active, id=ILUW6L1VJ)
- ▶ inbound-rtcp (kind=audio, mid=0, ssrc=577774253, [codec]=PCMU (0), id=IT01A57774253)
- ▶ remote-candidate (candidateType=host, id=IT5aV8Gj8)
- ▶ local-candidate (candidateType=host, tcpType=active, id=IcCmyDGdn)
- ▶ local-candidate (candidateType=host, id=Id4Nah5bb)
- ▶ local-candidate (candidateType=host, tcpType=active, id=Ih75e1cck)
- ▶ local-candidate (candidateType=host, id=IhDILG2ss)
- ▶ local-candidate (candidateType=host, id=IyDiSaDsy)
- ▶ outbound-rtcp (kind=audio, mid=0, ssrc=3932673161, [codec]=PCMU (0), id=OT01A3932673161)
- ▶ peer-connection (id=P)
- ▶ remote-inbound-rtcp (kind=audio, ssrc=3932673161, id=RIA3932673161)
- ▶ remote-outbound-rtcp (kind=audio, ssrc=577774253, id=ROA57774253)
- ▶ media-source (kind=audio, id=SA1)
- ▶ transport (iceState=connected, dtlsState=connected, id=T01)

Filter statistics graphs by type including

- ▶ Stats graphs for media-playout (kind=audio, id=AP)
- ▶ Stats graphs for candidate-pair (state=succeeded, id=CPyDiSaDsy_T5aV8Gj8)
- ▶ Stats graphs for inbound-rtcp (kind=audio, mid=0, ssrc=577774253, [codec]=PCMU (0), id=IT01A57774253)
- ▶ Stats graphs for outbound-rtcp (kind=audio, mid=0, ssrc=3932673161, [codec]=PCMU (0), id=OT01A3932673161)
- ▶ Stats graphs for peer-connection (id=P)
- ▶ Stats graphs for remote-inbound-rtcp (kind=audio, ssrc=3932673161, id=RIA3932673161)
- ▶ Stats graphs for remote-outbound-rtcp (kind=audio, ssrc=577774253, id=ROA57774253)
- ▶ Stats graphs for media-source (kind=audio, id=SA1)
- ▶ Stats graphs for transport (iceState=connected, dtlsState=connected, id=T01)

3

この特定のWebRTC接続で使用されているローカルおよびリモートのIPとポートは、PeerConnection APIコールのトレースで確認できます。

ICE connection state: new => completed

Connection state: new => connected

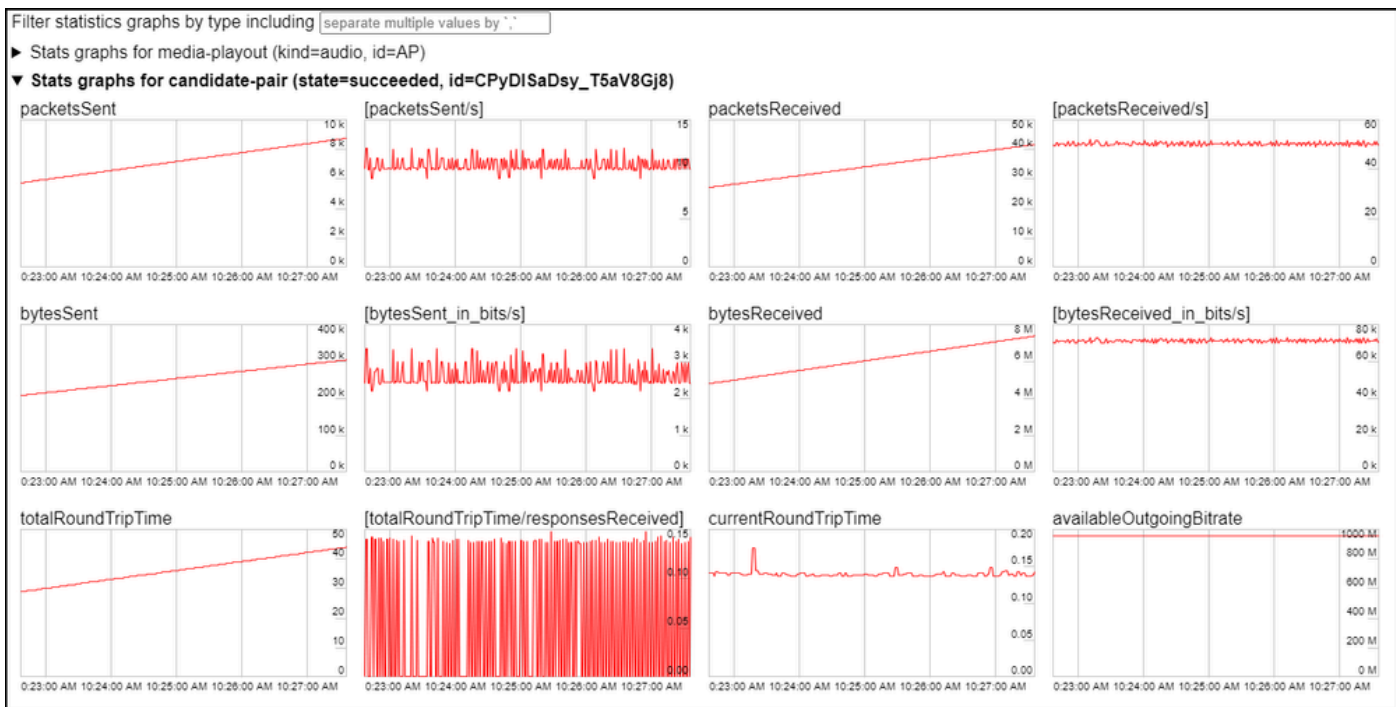
Signaling state: new => stable

ICE Candidate pair: 10.24.139.247:49870 <=>23.89.1.162:22744

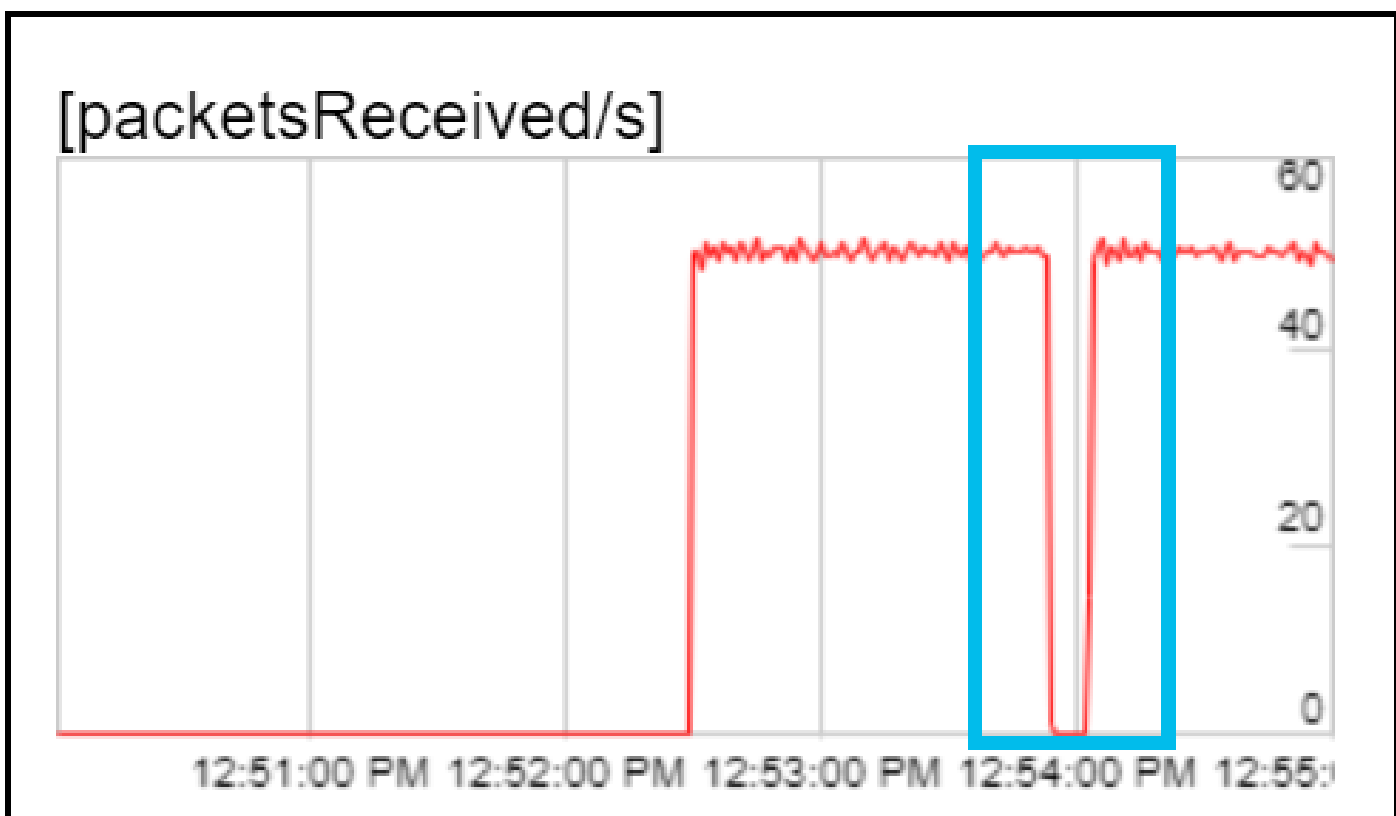
▶ ICE candidate grid

Time	Event
4/17/2024, 10:13:52 AM	▶ setRemoteDescription (type: "offer", 2 sections)
4/17/2024, 10:13:52 AM	▶ setLocalDescription (type: "answer", 2 sections)
4/17/2024, 10:13:52 AM	▶ signalingstatechange
4/17/2024, 10:13:52 AM	▶ iceconnectionstatechange
4/17/2024, 10:13:52 AM	▶ connectionstatechange

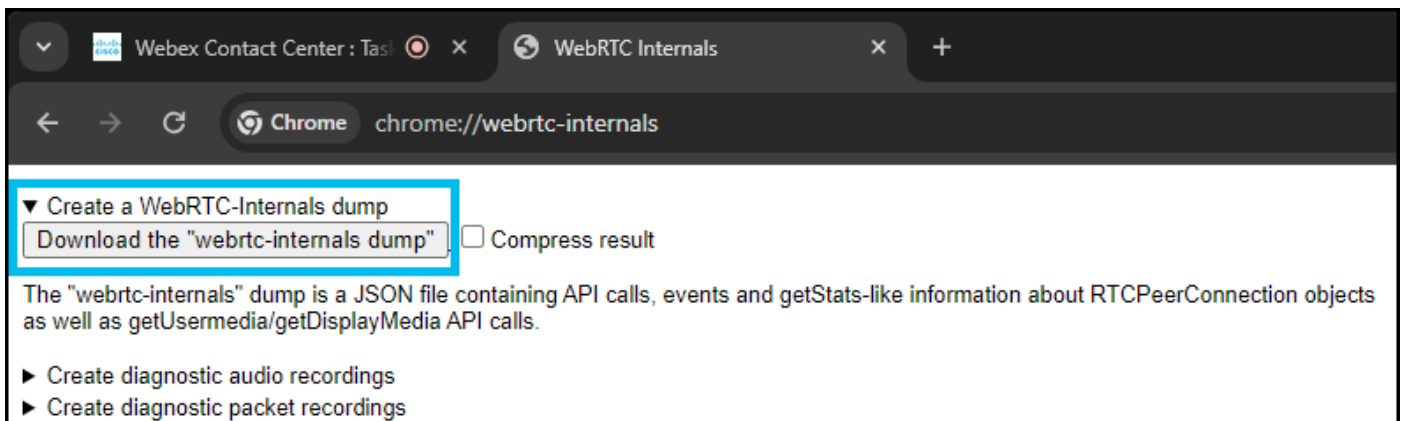
送信されたパケット、受信されたパケット、遅延、ドロップされたパケットなど、さまざまなライブコールの統計情報が下部に表示されます (太字のテキスト)。



たとえば、着信パケットが欠落すると、音声が欠落します。

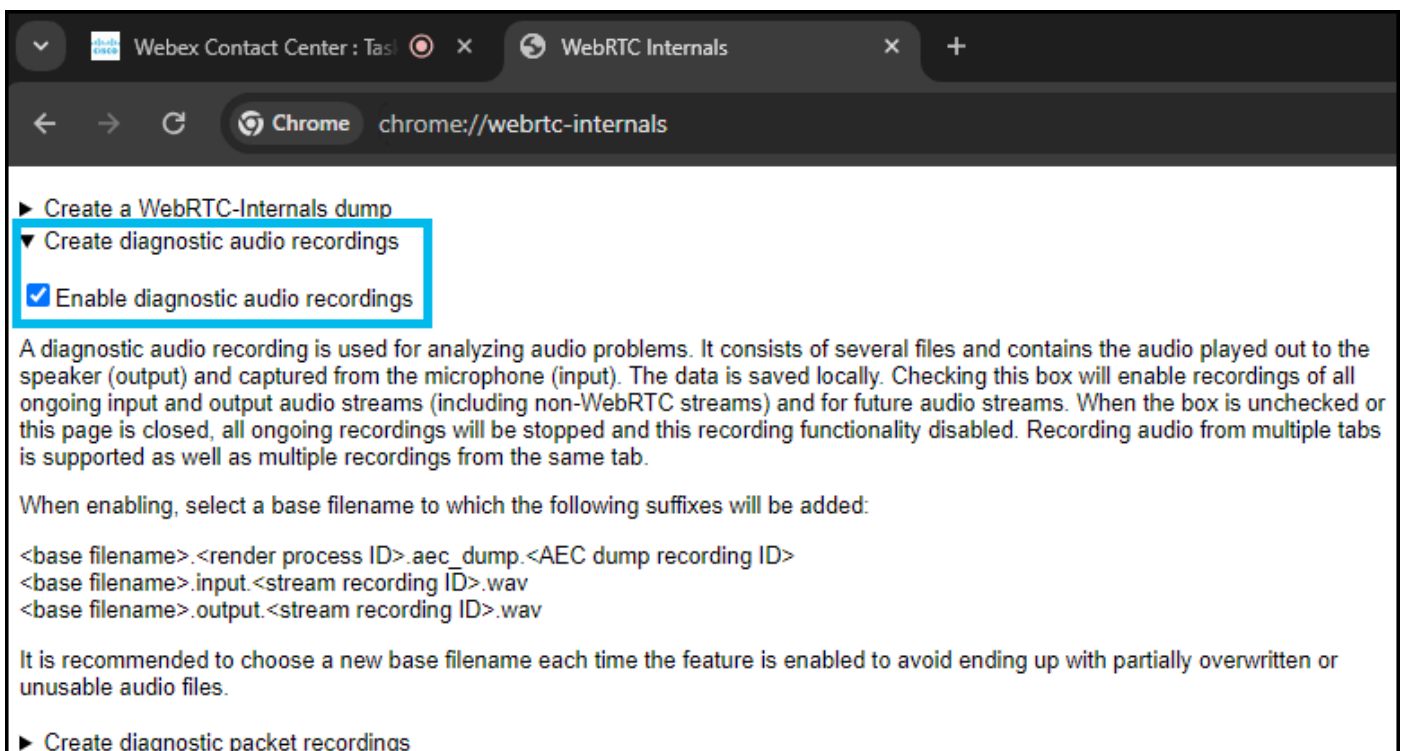


WebRTCのログダンプを取得するには、「WebRTC-Internalsダンプを作成」をクリックし、「webrtc-internalsダンプをダウンロード」オプションをクリックします。



注：WebRTCログダンプファイルは、特定のツール、つまり<https://fippo.github.io/webrtc-dump-importer/>を使用して解析および分析できます。

WebRTCの音声は、診断目的でキャプチャすることもできます。



翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。