

# Explorando dados e estatísticas do WebRTC no WxCC Agent Desktop

## Contents

---

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Overview](#)

[Cromo](#)

---

## Introdução

Este documento descreve os dados e estatísticas de chamadas WebRTC disponíveis usando ferramentas de navegador integradas.

## Pré-requisitos

### Requisitos

A Cisco recomenda que você tenha habilidades e conhecimento da solução Webex Contact Center (WxCC), navegadores Chrome, Edge, Firefox e projeto WebRTC.

### Componentes Utilizados

Este documento não se restringe a versões de software e hardware específicas.

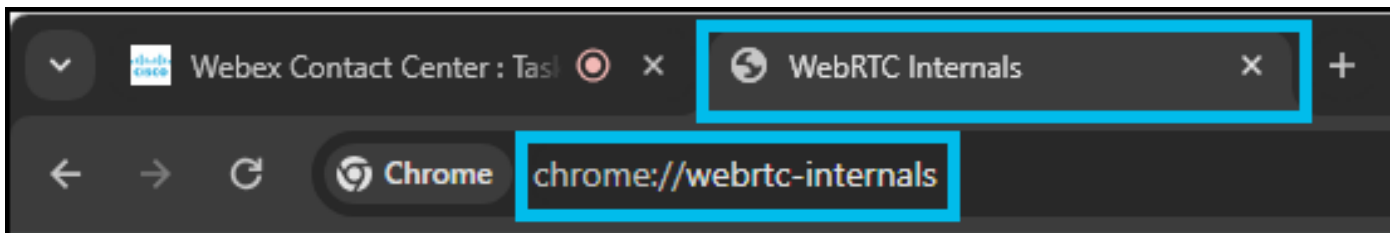
As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

## Overview

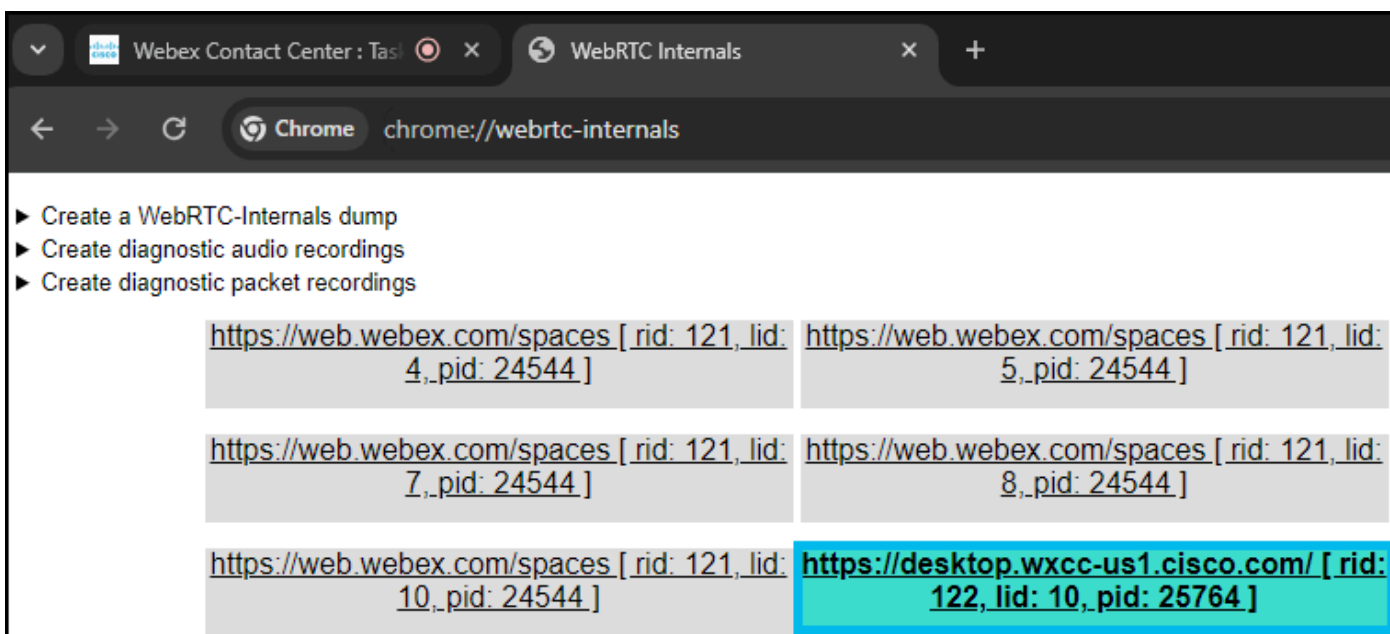
A maioria dos navegadores que suportam WebRTC têm ferramentas WebRTC integradas que podem ser úteis para solucionar problemas relacionados ao WebRTC.

# Cromo

Para acessar dados e estatísticas de chamadas do WebRTC, uma nova guia no navegador Google Chrome deve ser aberta durante uma chamada ativa e o seguinte URL deve ser usado: `chrome://webrtc-internals`.



Na guia WebRTC-internals, a sessão ativa do WebRTC deve ser selecionada (`desktop.wxcc-us1.cisco.com` em seu nome; A URL dependerá da região do WxCC).



As seguintes informações podem ser vistas:

- No lado esquerdo (1), um registro das atividades da API PeerConnection está disponível. O RTCPeerConnection é a interface central na API WebRTC. Ele representa a conexão entre o peer local e remoto e fornece todas as funções e eventos necessários para estabelecer a conexão (mais detalhes podem ser encontrados em <https://webrtc.org/getting-started/overview>). Esses rastreamentos (1) exibem interações com o objeto RTCPeerConnection e os parâmetros usados.
- No lado direito (2), as métricas coletadas através da API getStats sobre a conexão WebRTC atual estão disponíveis.
- Na parte inferior (3), os gráficos produzidos utilizando a API getStats (2) sobre a conexão WebRTC atual estão disponíveis.

https://desktop.wxcc-us1.cisco.com/, ( iceServers: [], iceTransportPolicy: all, bundlePolicy: max-compat, rtcpMuxPolicy: require, iceCandidatePoolSize: 0 ),

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

Filter statistics by type including

- ▶ media-playout (kind=audio, id=AP)
- ▶ certificate (id=CFE0AAE88DCEFAA00E634B6C16C3D0AD.C37B2BA7EE)
- ▶ 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)

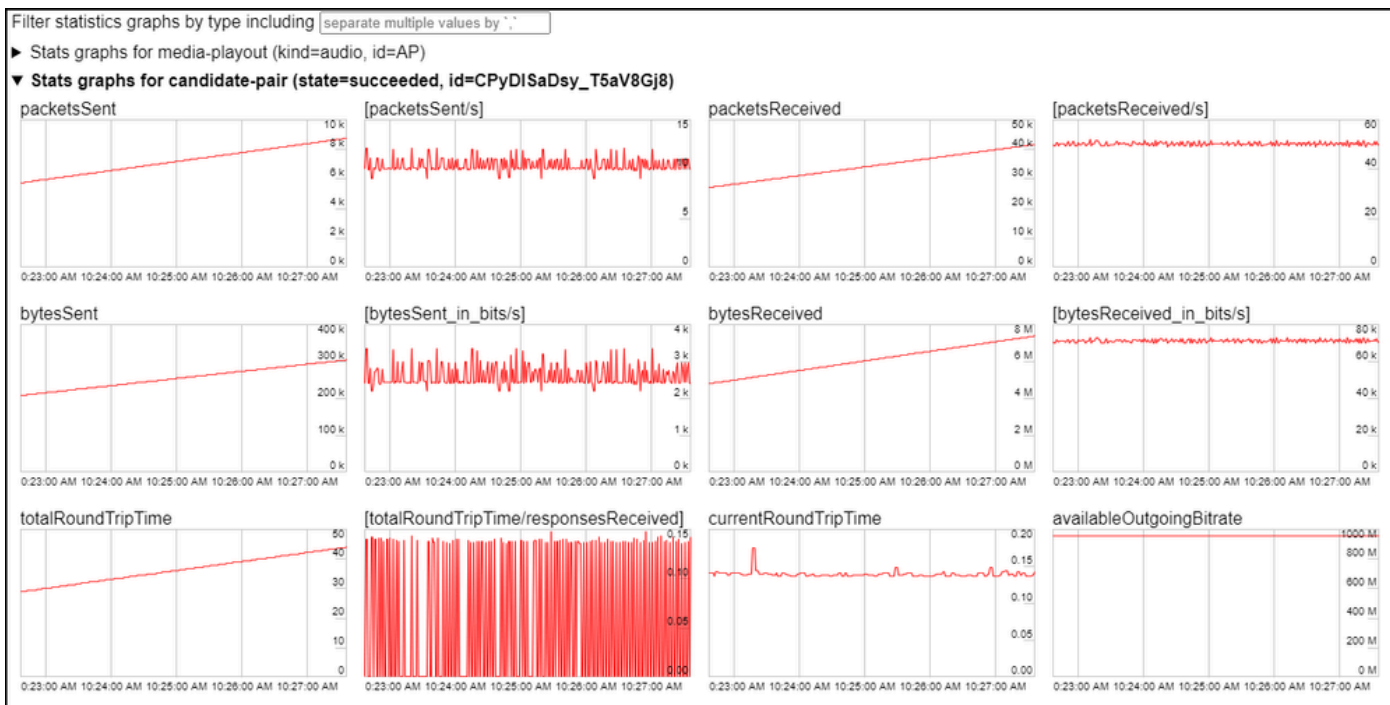
IPs e portas locais e remotos em uso para essa conexão WebRTC específica podem ser vistos sob rastreamentos das chamadas da API PeerConnection.

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**

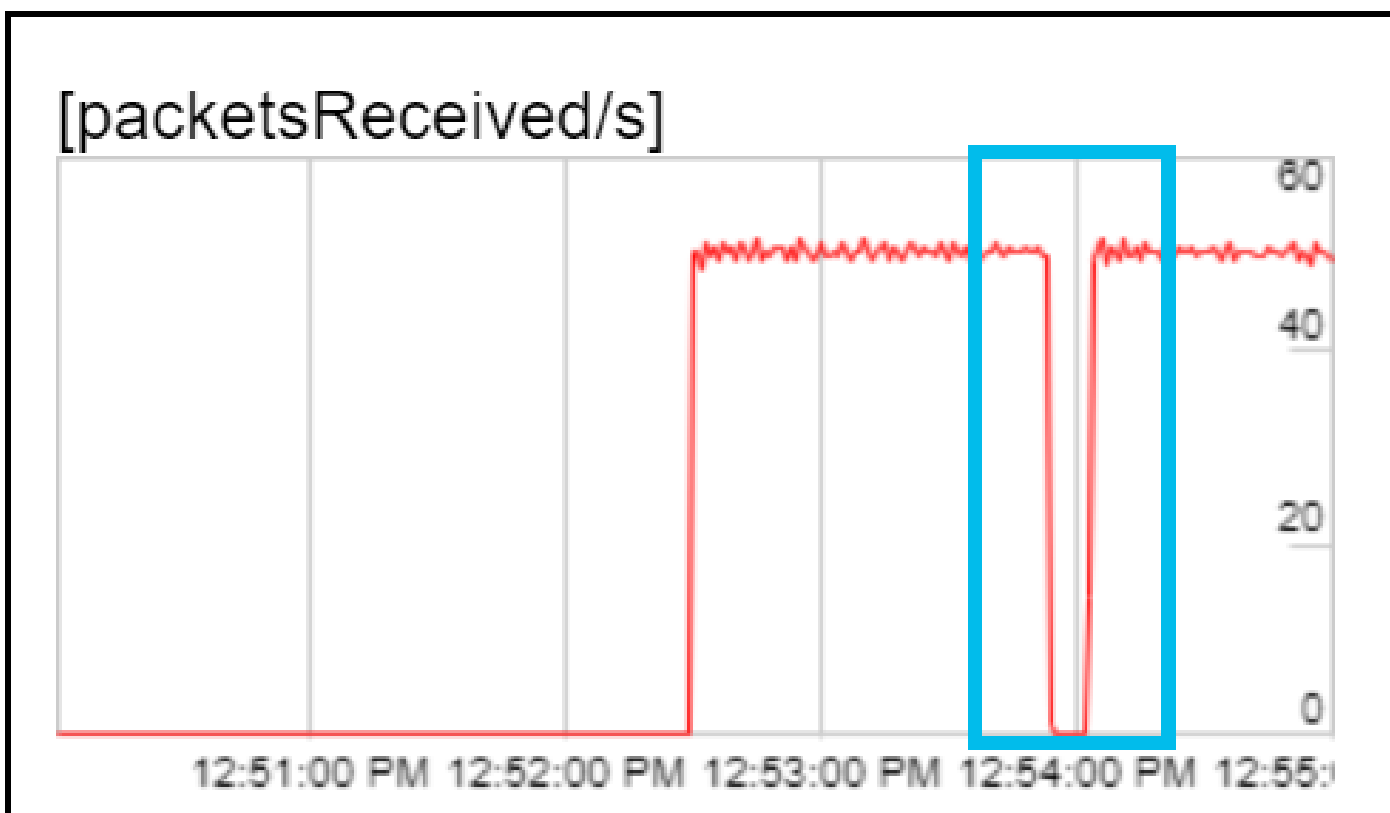
▶ ICE candidate grid

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

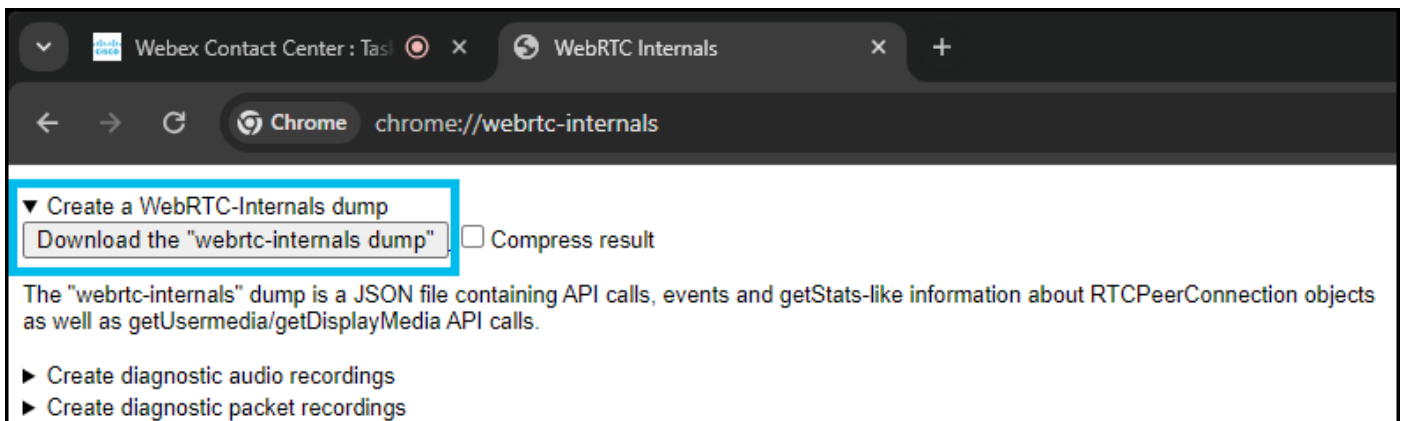
Várias estatísticas de chamadas ao vivo (ou seja, pacotes enviados, pacotes recebidos, latência, pacotes descartados e assim por diante.) gráficos relacionados podem ser vistos na parte inferior (texto em negrito).



Por exemplo - a ausência de pacotes de entrada causaria a ausência de áudio:

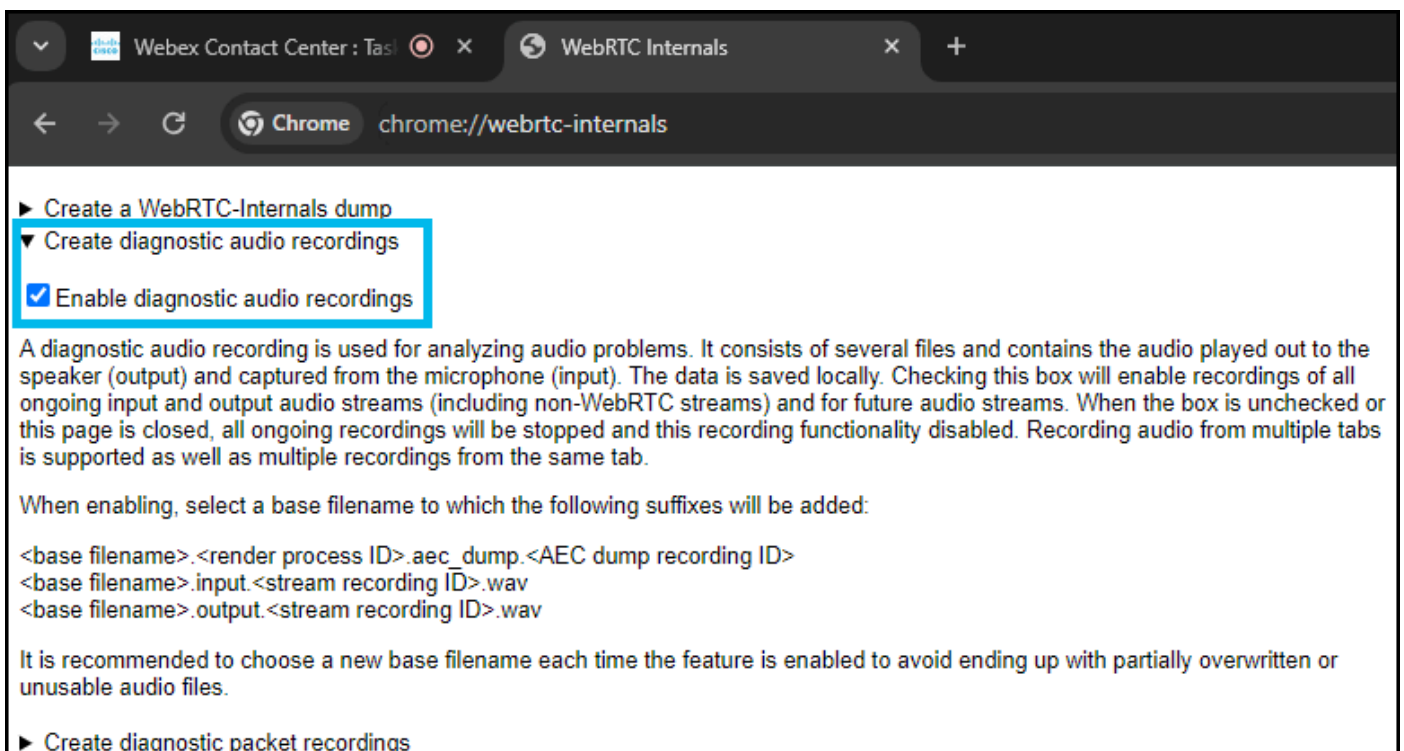


O despejo de log do WebRTC pode ser recuperado clicando em 'Criar um despejo do WebRTC-Internals' e clicando em 'Baixar a opção "despejo do webrtc-internals"'.



Note: O arquivo de despejo de log do WebRTC pode ser analisado e analisado usando ferramentas específicas, ou seja, <https://fippo.github.io/webrtc-dump-importer/>

O áudio WebRTC também pode ser capturado para fins de diagnóstico.



## Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.