omnissa™

Horizon SDK for WebRTC Redirection Setup Guide

May 28, 2025

Contents

Introduction

This document provides Omnissa partners with step-by-step instructions on how to set up a Horizon environment for WebRTC Redirection of media streams in unified communication (UC) applications.

The Omnissa Horizon® SDK for WebRTC Redirection provides access to interfaces that WebRTC-based UC vendors can use to leverage the Media Optimization feature of Horizon Client and Horizon remote desktops. The SDK enables the offloading of audio, video, and screenshare content to the local client system.

Note: This document is a work-in-progress and its contents may change without notice. For the most up-to-date information about the Horizon SDK for WebRTC Redirection, visit kb.omnissa.com and search for “HorizonforWebRTCOptimization”.

Supported Features

The Horizon SDK for WebRTC Redirection supports the following features:

- Audio and video redirection in VDI desktops
- Audio and video redirection in RDSH desktops
- Audio and video redirection in remote applications
- Screenshare redirection to the client endpoint in VDI and RDSH desktops
- Multi-monitor screen sharing in VDI and RDSH desktops
- Mute/unmute
- PSTN calls (DTMF calling)
- Data channel

Known Issues and Limitations

The Horizon SDK for WebRTC Redirection does not support the following functionality:

- Linux and Mac clients
- Screenshare redirection when the UC application is used as a published application
- Give or take control of screenshares
- E911 functionality
- Background Blur

Design Considerations

Omnissa recommends that you consider the following design guidelines when implementing WebRTC media redirection for your UC application in a Horizon environment.

- Configure your application by default to detect whether WebRTC redirection is enabled on the Horizon desktop and if so, to enable WebRTC redirection at the application level. To detect whether WebRTC redirection is enabled on the desktop, have your application check the DWORD value in the registry: `Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Omnissa\Horizon\WebRTCRedir\electronAppEnabled`

- Design your application with a setting that allows the administrator to enable and turn off WebRTC redirection at the user level.

  Note: The administrator must have the ability to configure WebRTC redirection at the user level because not all users may be running a client platform supported by the Horizon SDK for WebRTC Redirection. Your application setting should be distinct from the existing WebRTC redirection setting in the Horizon GPO provided by Omnissa.

- To ensure optimal WebRTC performance, set up your application to load the Horizon SDK for WebRTC Redirection only if WebRTC redirection is enabled at the application level. Your application should not load the SDK if WebRTC redirection is not enabled at the application level.

- Have your application fall back to non-VDI behavior when the Horizon Client endpoint does not support WebRTC redirection. In the fallback case, WebRTC media is not redirected but the user can still use your application and access all other functionality within the Horizon desktop. For more information, see scenario 12 under Test Scenarios later in this document.

- Design your application with the capability to differentiate between a console session and a Horizon Client remote session. In a console session, WebRTC media should not be redirected but users should still be able to use your application. This capability addresses the case where Horizon Agent is running on a physical machine and users can connect to this machine using either the console or Horizon Client.

Installation and Configuration

This section describes the SDK installation and setup procedures.

System Requirements

This section describes the system requirements for using the Horizon SDK for WebRTC Redirection.

Note: System requirements for Real-Time Audio-Video (RTAV) are different and more substantial than when using the Horizon SDK for WebRTC Redirection. RTAV is used if Media Optimization is not turned on via GPO or if the endpoint does not support Media Optimization (fallback mode). For more information on using your application with RTAV, see the System Requirements for Real-Time Audio-Video article.

Omnissa Horizon Client Version - Minimum required version: 8.15.0 (Horizon 2503) for both agent and client
To support the latest features and interfaces of the SDK, ensure that you are running the latest release version of Horizon 8, which is available through Omnissa Customer Connect. - Omnissa Horizon Cloud Service on Microsoft Azure
To support the latest features and interfaces of the SDK, ensure that your Horizon Cloud pods are running the latest release version of the pod manifest. **Note**: If you are running an earlier release version of Horizon 8 or of the Horizon Cloud Service on Microsoft Azure pod manifest, some features and interfaces of the Horizon SDK for WebRTC Redirection are not supported.

Horizon 2503 release is the first release to support call reconnection (ICE restart) functionality. Earlier versions do not support this feature.

Client System
The SDK only supports Windows client endpoints at this time. Verify that each client system meets the following requirements:
- Windows 11 or Windows 10
- 2.4 GHz dual-core processor (minimum)
- Horizon Client for Windows 2206 or later

Note: For information about the operating systems supported for a specific version of Horizon Client, see the release notes for that client version.

Remote Desktop
Verify that the source virtual machine for each remote desktop meets the following requirements: - Equipped with 2 vCPUs, at minimum
- Running one of the following operating systems:
- Windows 11, Windows 10
- Windows 10 Enterprise (multi-session, for Azure only)
- Windows Server 2019, 2016, 2012 R2

UC Application
The Horizon SDK for WebRTC Redirection supports UC applications built on the Electron software framework.

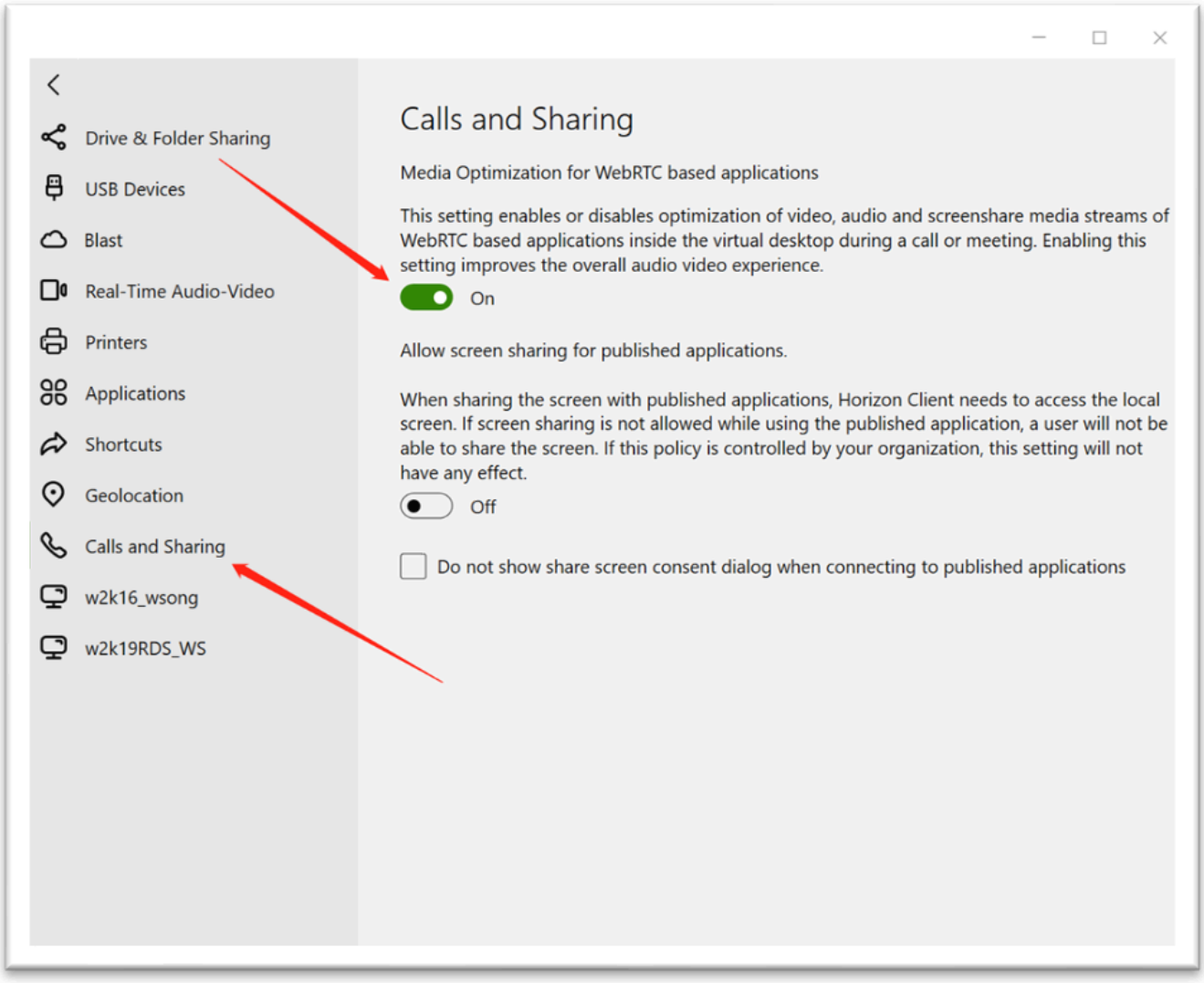
Note: The SDK only supports Electron-based UC applications running on Omnissa VDI deployments. It does not support CEF, PWA, or browser-based WebRTC applications.

To implement media optimization for browser-based WebRTC applications, use the Browser Content Redirection feature described in the Configuring Browser Redirection article.

Client Installation

The Horizon WebRTC Redirection option is installed and enabled in Horizon Client for Windows by default.

The user can turn off this option from the Horizon Client settings as shown in the following screenshot.



To verify that Horizon WebRTC Redirection is enabled on the Windows client endpoint:

1. On the Windows client, go to:
   `C:\Program Files\Omnissa\\Omnissa Horizon Client\`
2. Confirm the presence of `webrtc_sharedlib.dll` in the folder.

#- navigation - toc

Horizon Agent Installation and Configuration

The Horizon WebRTC Redirection feature is installed by default with Horizon Agent as part of the core feature set.

However, the feature is installed in deactivated state. You must manually activate or enable the feature using the Group Policy Management Editor (GPO editor).

To enable WebRTC Redirection on the agent machine using the GPO editor:

1. Download latest GPO bundle:
   - Go to Omnissa Customer Connect Downloads
   - Select **Horizon** and click **GO TO DOWNLOADS**.
   - Find **GPO Bundle** and click **DOWNLOAD NOW**:

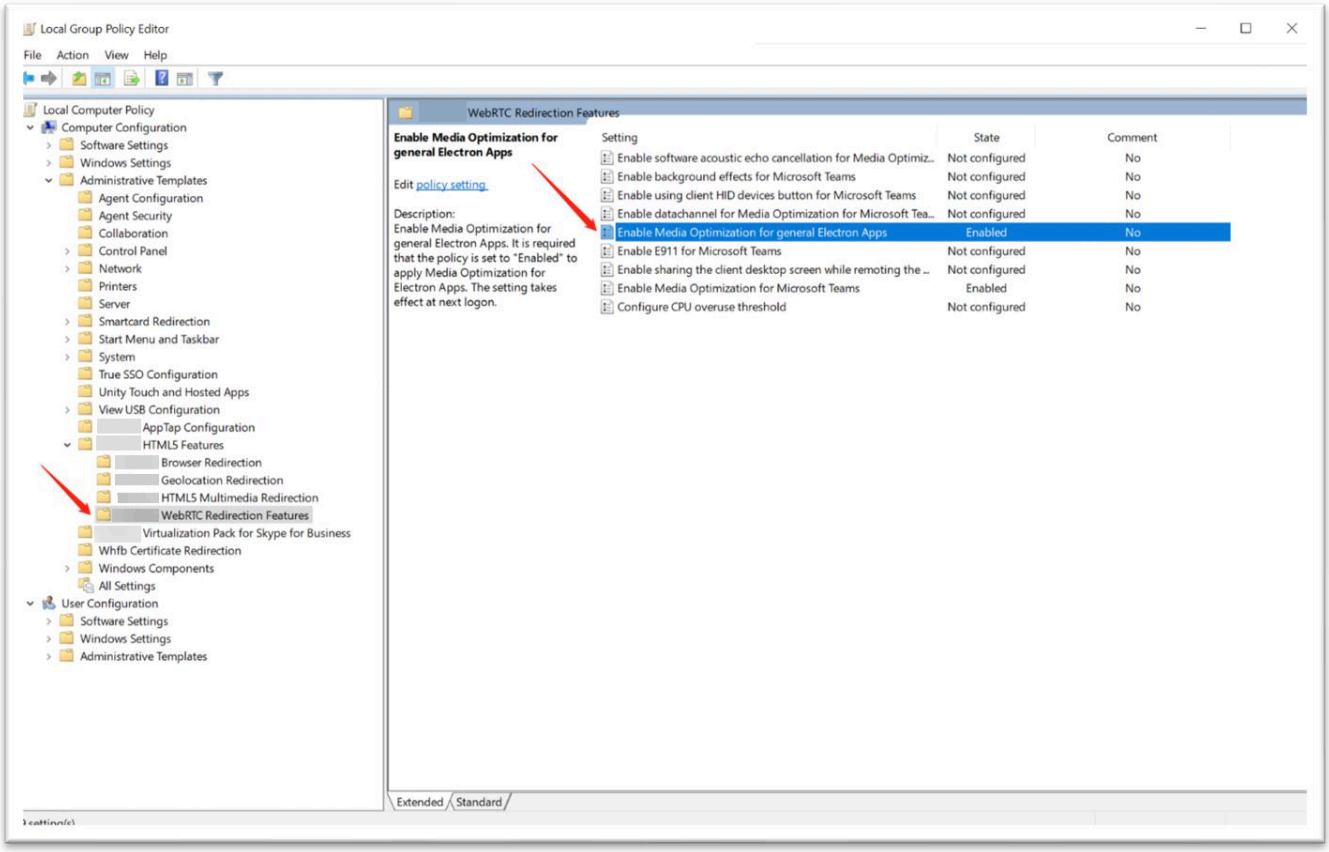
   `-Horizon-Extras-Bundle-YYMM-x.x.x-yyyyyyyy.zip` where YYMM is the marketing version, x.x.x is the internal version and yyyyyyyy is the build number.

2. Set up the Windows GPO policy to include Horizon ADMX template files.

3. Using the GPO editor, navigate to
   `Computer Configuration > Administrative Templates > View Agent Configuration > HTML5 Features > WebRTC Redirection Features`

4. Set **Enable Media Optimization for general Electron Apps** policy to the state of **Enabled**

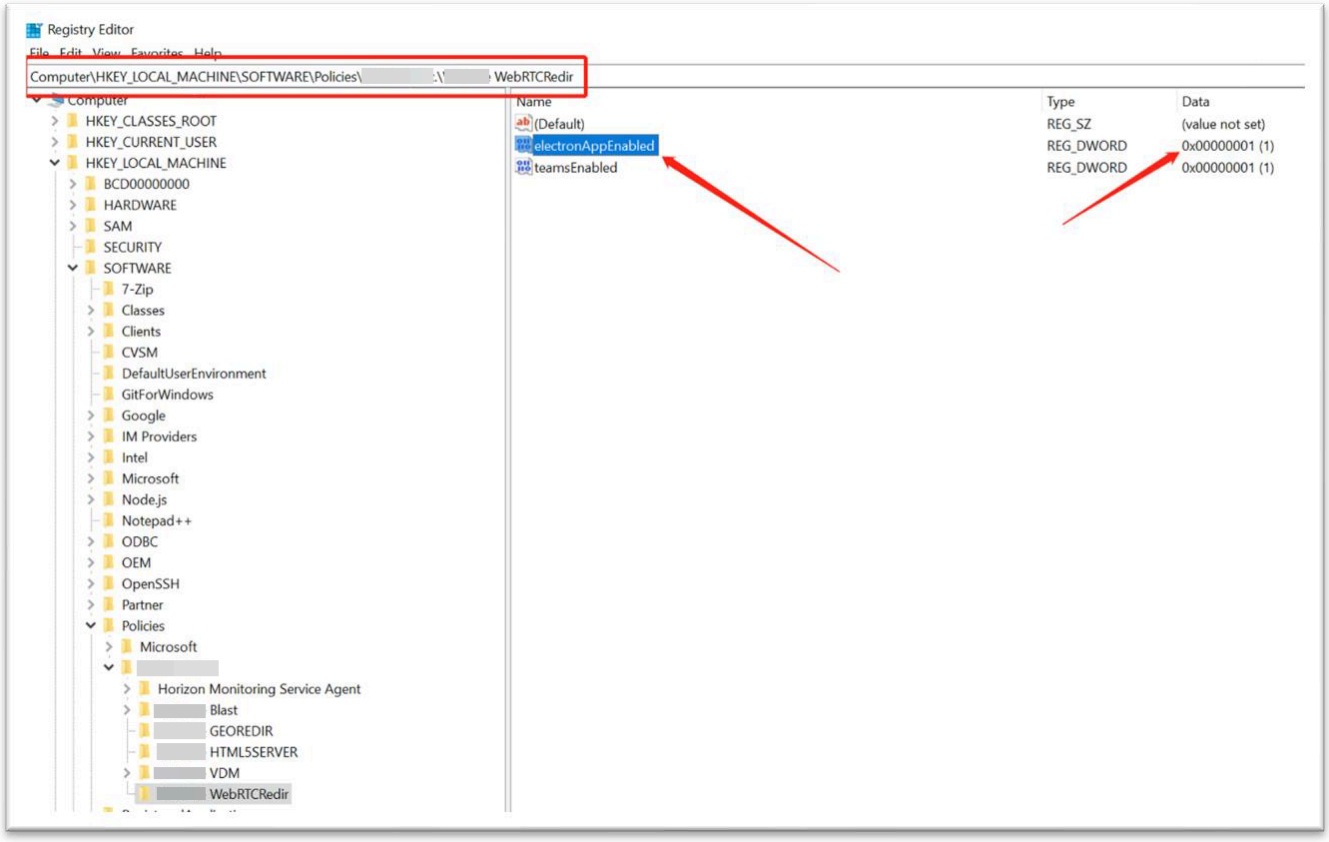
5. To make the group policy change take effect, log out from the agent machine and then log back in.



1

To verify that WebRTC Redirection GPO policy is enabled on the agent machine:

1. Go to:
   `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Omnissa\Horizon\WebRTCRedir`
2. Verify that the **electronAppEnabled** registry key is present in that folder.



#- navigation - toc

Verify the WebRTC Redirection Sample Application

As the final step of the Horizon WebRTC Redirection setup, verify that the WebRTC Redirection sample application is correctly configured.

The sample application is included with the Horizon SDK for WebRTC Redirection.

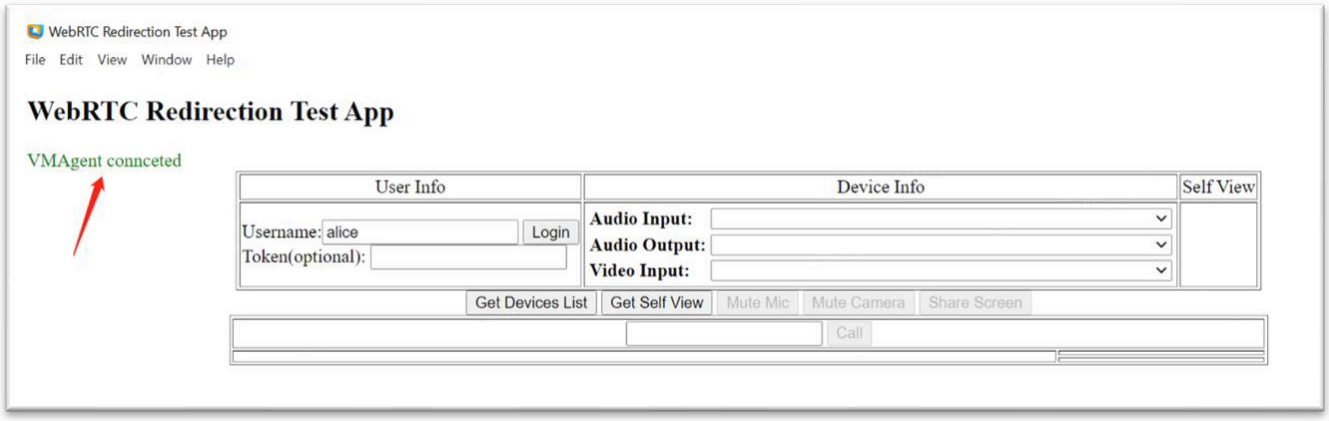
For detailed information, see:
- **“Setting Up the Sample Application”** in the *Omnissa Horizon SDK for WebRTC Redirection Programming Guide*
- Instructions in `webrtcredir\sample\README.txt` in the SDK zip file

To verify the test application setup:

1. **Launch** the WebRTC Redirection test application.
2. **Wait a few seconds** for the application to establish a connection to the agent machine.
3. Verify that a **green label** appears indicating **connected status**.

In connected status, the application can successfully call the Horizon WebRTC Redirection APIs, redirecting media streams to the client endpoint.

Example: Connected Status



If you see a **red “Not Connected”** label after launching the app, WebRTC Redirection was **not correctly enabled**.

In this status, the application falls back to calling the built-in Electron WebRTC APIs, which redirect WebRTC-based media streams using the non-optimized Real-Time Audio-Video feature. For more information, see the System Requirements for Real-Time Audio-Video.

Example: Not Connected Status

Installation and Configuration

This section describes the SDK installation and setup procedures.

System Requirements

This section describes the system requirements for using the Horizon SDK for WebRTC Redirection.

Note: System requirements for Real-Time Audio-Video (RTAV) are different and more substantial than when using the Horizon SDK for WebRTC Redirection. RTAV is used if Media Optimization is not turned on via GPO or if the endpoint does not support Media Optimization (fallback mode). For more information on using your application with RTAV, see the System Requirements for Real-Time Audio-Video article.

Omnissa Horizon Client Version - Minimum required version: 8.15.0 (Horizon 2503) for both agent and client
To support the latest features and interfaces of the SDK, ensure that you are running the latest release version of Horizon 8, which is available through Omnissa Customer Connect. - Omnissa Horizon Cloud Service on Microsoft Azure
To support the latest features and interfaces of the SDK, ensure that your Horizon Cloud pods are running the latest release version of the pod manifest. **Note**: If you are running an earlier release version of Horizon 8 or of the Horizon Cloud Service on Microsoft Azure pod manifest, some features and interfaces of the Horizon SDK for WebRTC Redirection are not supported.

Horizon 2503 release is the first release to support call reconnection (ICE restart) functionality. Earlier versions do not support this feature.

Client System
The SDK only supports Windows client endpoints at this time. Verify that each client system meets the following requirements:
- Windows 11 or Windows 10
- 2.4 GHz dual-core processor (minimum)
- Horizon Client for Windows 2206 or later

Note: For information about the operating systems supported for a specific version of Horizon Client, see the release notes for that client version.

Remote Desktop
Verify that the source virtual machine for each remote desktop meets the following requirements: - Equipped with 2 vCPUs, at minimum
- Running one of the following operating systems:
- Windows 11, Windows 10
- Windows 10 Enterprise (multi-session, for Azure only)
- Windows Server 2019, 2016, 2012 R2

UC Application
The Horizon SDK for WebRTC Redirection supports UC applications built on the Electron software framework.

Note: The SDK only supports Electron-based UC applications running on Omnissa VDI deployments. It does not support CEF, PWA, or browser-based WebRTC applications.

To implement media optimization for browser-based WebRTC applications, use the Browser Content Redirection feature described in the Configuring Browser Redirection article.

Horizon Agent Installation and Configuration for browser-based WebRTC Redirection

The Horizon browser-based WebRTC Redirection feature is installed by default with Horizon Agent as part of the core feature set.

However, the feature is installed in deactivated state. You must manually activate or enable the feature using the Group Policy Management Editor (GPO editor).

Setup browser-based Webrtc Redirection for web application on agent

To enable Horizon WebRTC Redirection SDK Web App Support on the agent machine using the GPO editor:

1. Download latest GPO bundle:
   - Go to Omnissa Customer Connect Downloads
   - Select **Horizon** and click **GO TO DOWNLOADS**.
   - Find **GPO Bundle** and click **DOWNLOAD NOW**:

   `-Horizon-Extras-Bundle-YYMM-x.x.x-yyyyyyyy.zip` where YYMM is the marketing version, x.x.x is the internal version and yyyyyyyy is the build number.

2. Set up the Windows GPO policy to include Horizon ADMX template files. Using the GPO editor, navigate to
   `Computer Configuration > Administrative Templates > Omnissa Horizon Agent Configuration > Horizon HTML5 Features > Horizon WebRTC Redirection SDK Web App Support`

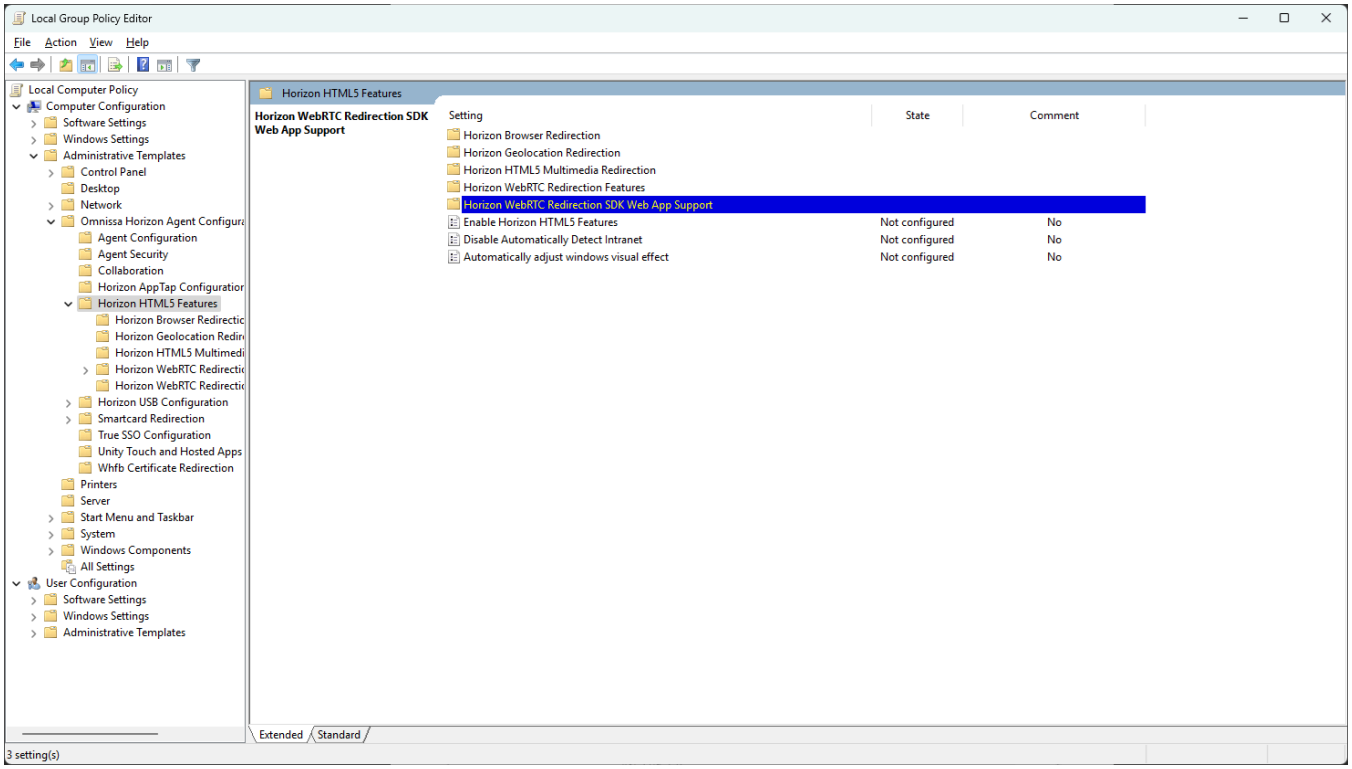
3. Set **Enable Media Optimization for web applications** policy to the state of **Enabled**

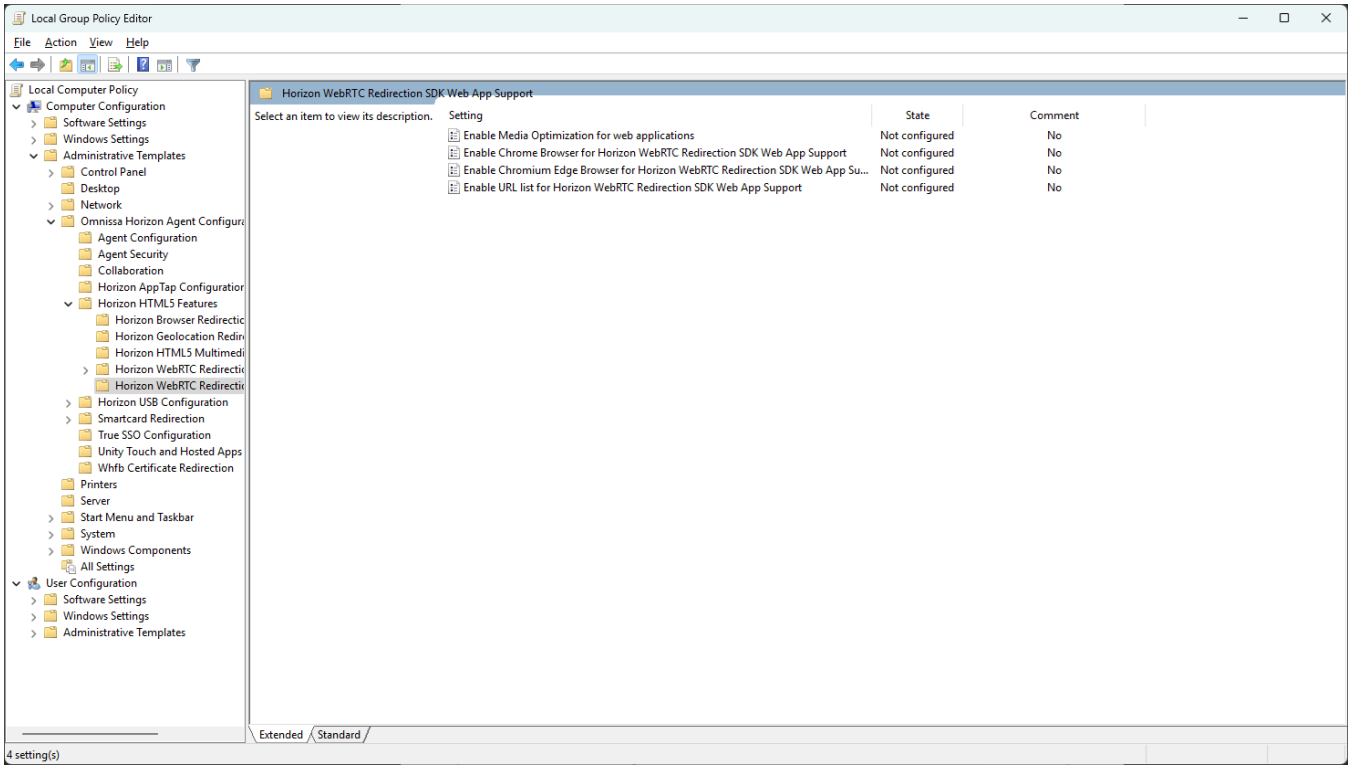
4. Set the policy to enable the Chrome/Edge browser. - To enable the Chrome browser, set **Enable Chrome Browser for Horizon WebRTC Redirection SDK Web App Support** policy to the state of **Enabled**

   - To enable the Edge browser, Set **Enable Chromium Edge Browser for Horizon WebRTC Redirection SDK Web App Support** policy to the state of **Enabled**

6. Set **Enable URL list for Horizon WebRTC Redirection SDK Web App Support** policy. E.g https://*.mycompany.com/*

7. To make the group policy change take effect, log out from the agent machine and then log back in.

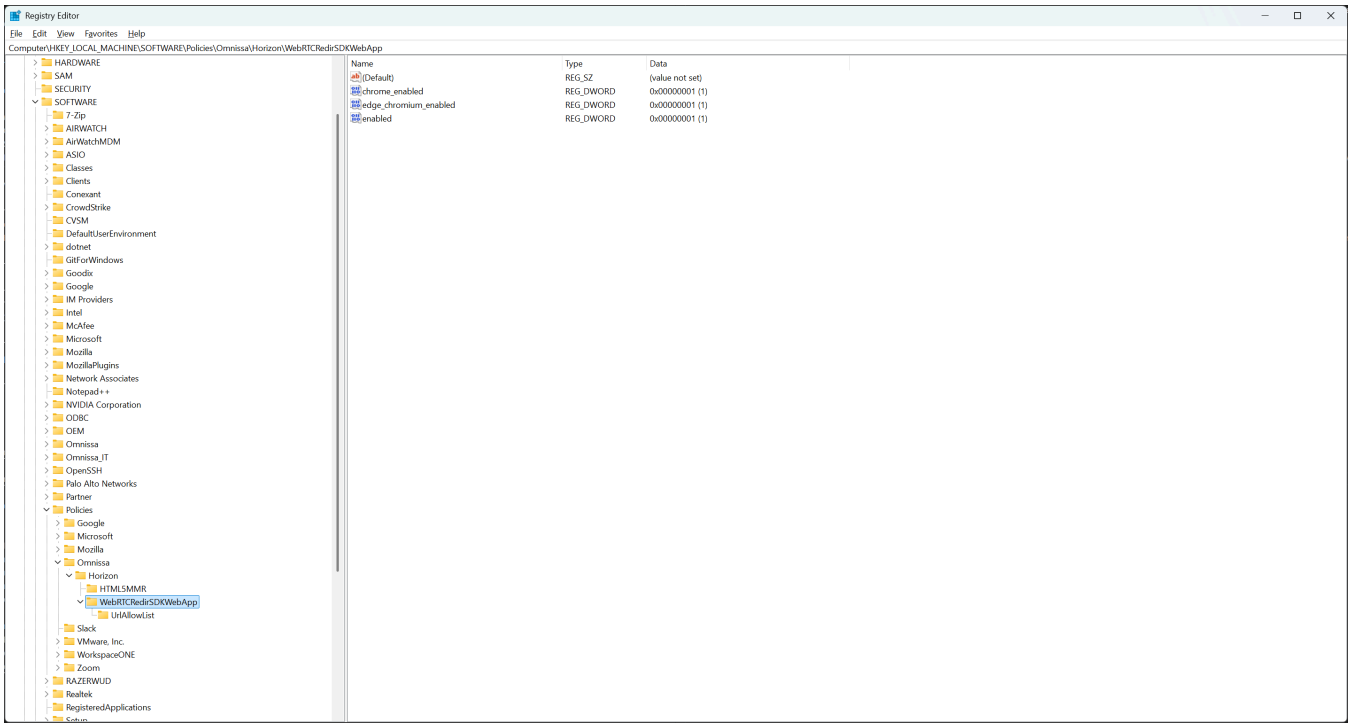




To verify that WebRTC Redirection GPO policy is enabled on the agent machine:

1. Go to: `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Omnissa\Horizon\WebRTCRedirSDKWebApp`
2. Verify that the **enabled**, **chrome_enabled**(or **edge_chromium_enabled**) registry key is present in that folder.

3. Verify that `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Omnissa\Horizon\WebRTCRedirSDKWebApp\UrlAllowList` registry key is present and has the URL list specified.



Omnissa browser Extension

- Horizon browser extension enables webapp support for the WebRTC SDK inside Horizon Agent and can be downloaded from the Chrome Store
- The extension is designed to be used by both Chrome and Edge browser.

Verify the browser-based WebRTC Redirection Sample Application

As the final step of the Horizon WebRTC Redirection for web app setup, verify that the WebRTC Redirection sample web application is correctly configured.

The sample web application is included with the Horizon SDK for WebRTC Redirection.

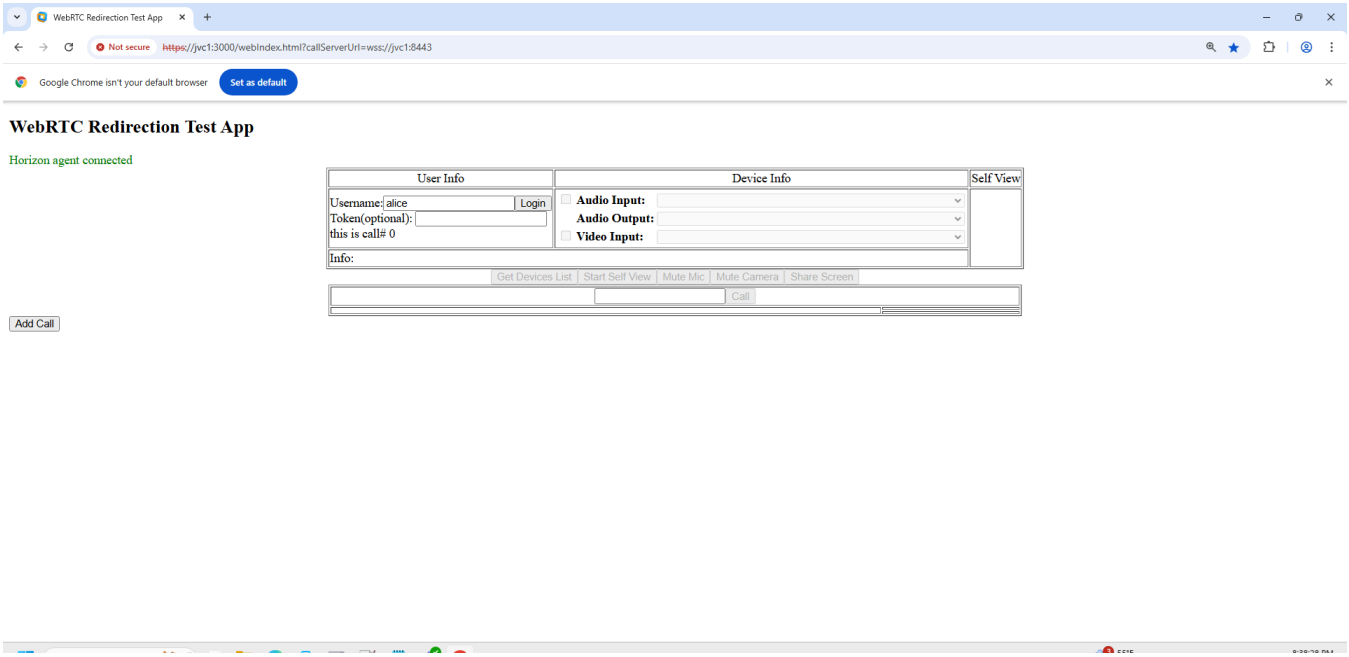
For detailed information, see:
- **“Setting Up the Sample Application”** in the *Omnissa Horizon SDK for WebRTC Redirection Programming Guide*
- Instructions in `sample\README.txt` in the SDK zip file

To verify the test application setup:

1. **Launch** Chrome browser and nagivate to the WebRTC Redirection sample web application URL.
2. **Wait a few seconds** for the application to establish a connection to the agent machine.
3. Verify that a **green label** appears indicating **connected status**.

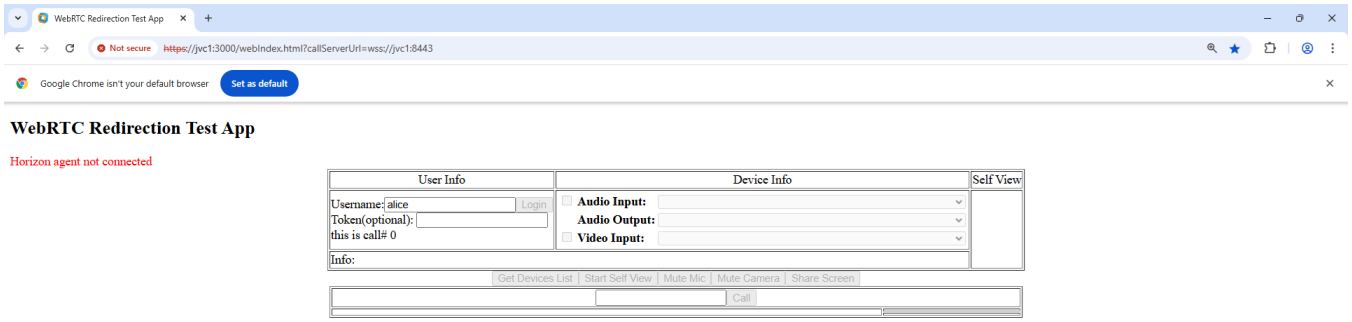
In connected status, the application can successfully call the Horizon WebRTC Redirection APIs, redirecting media streams to the client endpoint.

Example: Connected Status



If you see a **red “Not Connected”** label after launching the sample web application, WebRTC Redirection was **not correctly enabled**.

Example: Not Connected Status



Test Scenarios

Omnissa recommends running the following test scenarios to verify that your UC application works as expected with Media Optimization enabled through the Horizon SDK for WebRTC Redirection.

> **Note:** Scenario 12 tests fallback behavior when the client does not support Media Optimization but the remote desktop does. In this case, the app should still function using the Real-Time Audio-Video (RTAV) feature.

No.	Description	Details
1	One-to-one (1:1) calls	Start a person-to-person call with audio. Test across VDI and non-VDI users, including cross-org scenarios.
2	Group calls	Start a group call from a multiperson chat thread (if supported).
3	Meetings	Join a meeting from within the app or calendar invite. Verify functionality with both VDI and non-VDI participants.
4	Audio-to-video escalation	During an audio call or meeting, turn on your camera to escalate to video. Test muting/unmuting the camera and verify video transitions.
5	Screenshare	During a 1:1 call or meeting, share your desktop and confirm the receiver sees the shared screen.
6	Multi-monitor screenshare	Share screen 1, end it, then share screen 2. Confirm both screens display properly.
7	In-call device settings	While in a call or meeting, change your active device (mic, speaker, etc.). Verify the switch works in real time.
8	Mute/unmute using in-app controls	Use the app UI to mute/unmute. Verify audio is correctly suppressed and restored.
9	Escalation from 1:1 to group call	Add another participant to a 1:1 call and confirm the escalation works.
10	Hold/resume in meetings	Place a meeting on hold while accepting a 1:1 call. Then resume the meeting and confirm both actions succeed.
11	PSTN calling	Initiate a call to a PSTN number or add one to an existing call. Repeat for meetings.
12	Fallback case	Disable Media Optimization from Horizon Client settings. Reconnect to a remote desktop with Media Optimization enabled. Verify that the app continues to work using RTAV (non-optimized).

For more about fallback and RTAV behavior, see:
System Requirements for Real-Time Audio-Video

Collecting Logs for Troubleshooting

If you experience any issues on the client endpoint or agent machine, collect the necessary logs to assist your Omnissa support representative in troubleshooting.

Horizon Client Logs

Follow the appropriate procedure for your client platform to collect the client logs. Each procedure involves the following high-level tasks:
1. Disconnect from the problematic session.
2. Enable detailed logging at the **trace** level.
3. Reconnect to the session and reproduce the issue.
4. Collect the client logs.

To collect logs for Horizon Client for Windows:

1. Disconnect from the desktop session showing the problem.
2. Open **Registry Editor** as Administrator on the Horizon Client endpoint.
3. Add the following registry keys under:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Omnissa\Horizon\HTML5MMR
   ```

```
"html5mmr.log.noThrottle"=dword:00000001
```

```
HKEY_LOCAL_MACHINE\SOFTWARE\Omnissa\Horizon\HTML5MMR\WebRTCRedir
```

```
"html5mmr.log.webrtc.sharedlib.internal"=dword:00000001
"html5mmr.log.webrtc.sharedlib.network"=dword:00000001
"html5mmr.log.webrtc.sharedlib.media"=dword:00000001
"html5mmr.log.webrtc.sharedlib.signal"=dword:00000001
"html5mmr.log.webrtc.allowFullText"=dword:00000001
"html5mmr.log.webrtc.allowThrottle"=dword:00000000
"html5mmr.log.noThrottle"=dword:00000001
"html5mmr.log.webrtc.tracelevel"=dword:00000001
```

4. Reconnect to the remote desktop and reproduce the issue.
5. Collect the **DCT support logs** for Horizon Client for Windows as described in: Omnissa KB 1017939

Horizon Agent Logs

Use the following procedure to collect DCT support logs on the desktop machine.

To collect logs for Horizon Agent:

1. On the desktop machine exhibiting the problem issue, open `cmd.exe` as an administrator.
2. Navigate to the DCT log folder.

- For Horizon 8:

```
cd "C:\Program Files\Omnissa\Horizon\Agent\DCT"
```

- For Horizon Cloud on Microsoft Azure:

```
cd "C:\Program Files\Omnissa\Horizon Agents\Horizon Agent\DCT"
```

3. Run:

```
support.bat
```

4. The generated `.zip` folder will appear in the `hzn-sdct` folder on the desktop. Upload this zip for Omnissa Support.