

NICE WebRTC Redirection

Customer Readiness Executive Brief

Audience: Customer IT leadership adopting NICE WebRTC Redirection

Status: Draft

Last updated: September 24, 2026

This brief states the infrastructure requirements customers must meet to use NICE WebRTC Redirection. Detailed deployment steps, validation procedures, and diagnostics are in the [Customer Readiness and Validation Guide](#).

Who should read which document

Reader	Start here
IT leadership	This Executive Brief (owners, requirements, what “ready” means)
VDI / endpoint / browser policy	Receiver App Deployment Guide , ChromeOS Enterprise Deployment Guide , Citrix UCSDK Policy Guide , Omniissa Horizon Policy Guide
Network / security	Customer Readiness and Validation Guide (matchmaker, STUN, CXone UDP, TLS inspection, validation)

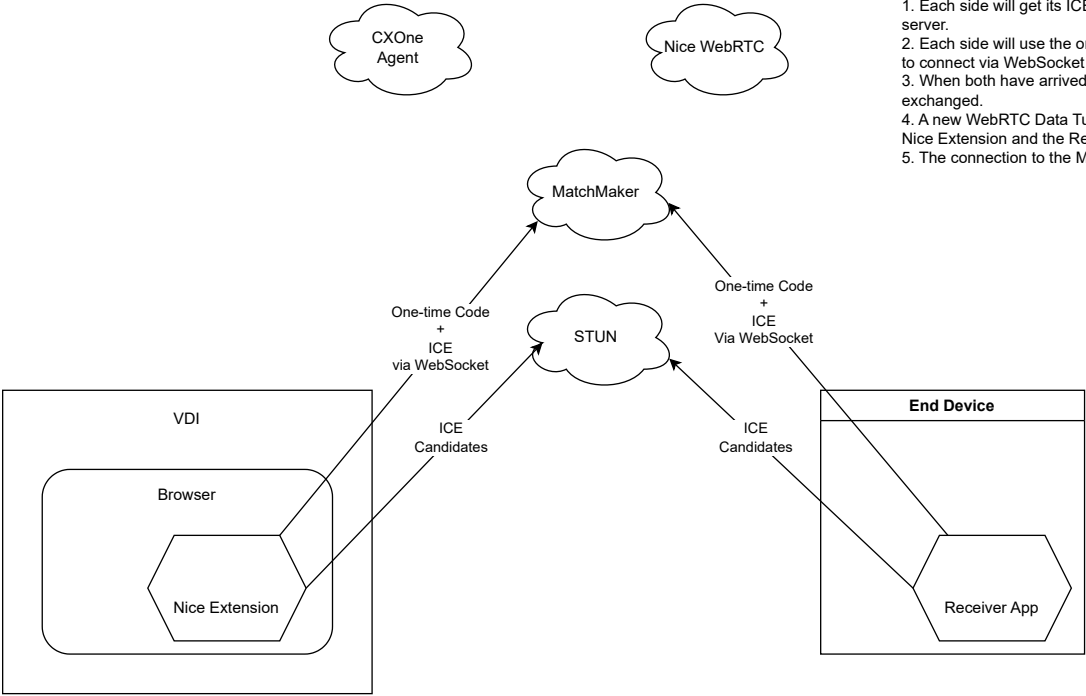
What you need to know

NICE WebRTC Redirection keeps CXone Agent in the hosted browser and moves voice media to the agent’s physical endpoint. Microphone, speaker, and the WebRTC media path run on that endpoint and connect directly to NICE CXone.

The NICE matchmaking service associates the hosted browser session with the endpoint Receiver. It is used for control and pairing only — voice media does not transit matchmaking.

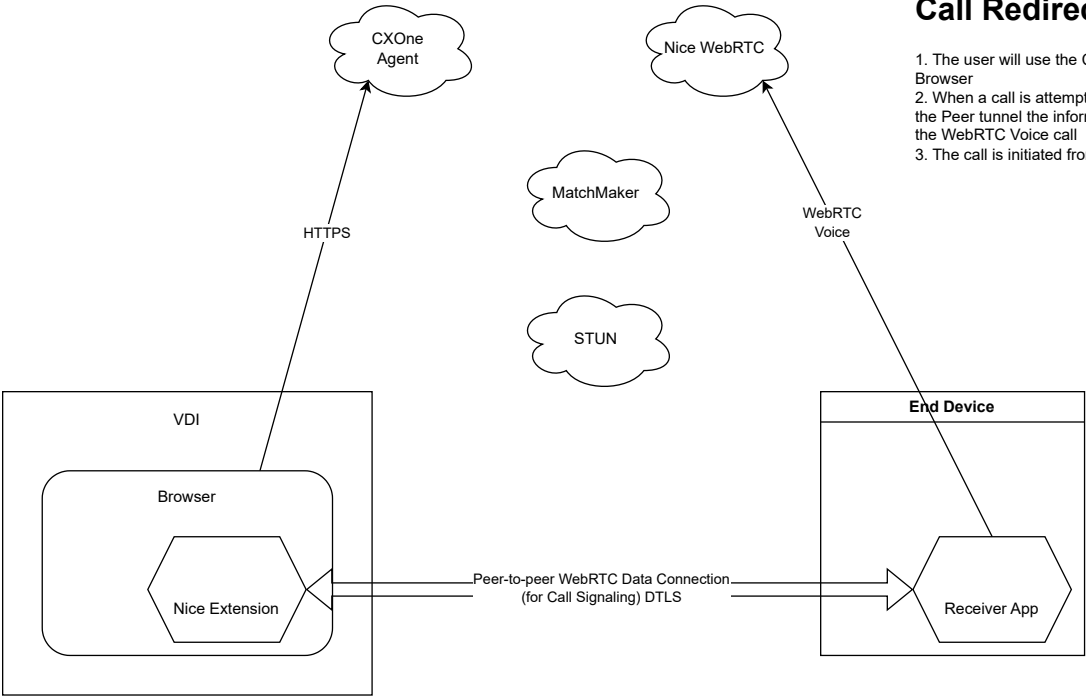
Matchmaking

1. Each side will get its ICE candidates from the STUN server.
2. Each side will use the one-time code (SPAKE 2 encrypted) to connect via WebSocket to the Matchmaker Service
3. When both have arrived, their ICE information is exchanged.
4. A new WebRTC Data Tunnel is established between the Nice Extension and the Receiver App
5. The connection to the Matchmaker service is then dropped



Call Redirection

1. The user will use the CXOne Agent web inside of the VDI Browser
2. When a call is attempted, the Nice Extension instead signals via the Peer tunnel the information that would be required to complete the WebRTC Voice call
3. The call is initiated from the Receiver App



Matchmaking

1. Each side obtains ICE candidates from the STUN service.
2. Each side uses the one-time code to connect over WebSocket to the matchmaking service.
3. When both have arrived, ICE information is exchanged.
4. A WebRTC data tunnel is established between the NICE browser extension and the Receiver.
5. The connection to the matchmaking service is then dropped.

Call redirection

1. The agent uses CXone Agent in the hosted browser.
2. When a call is attempted, the NICE extension signals the information required for the WebRTC voice call over the peer tunnel.
3. The call is initiated from the Receiver on the physical endpoint.

A working CXone call over ordinary hosted-desktop (VDI) audio does **not** mean redirection is active. Redirection requires the components, policies, and network paths below to be in place on the networks agents actually use.

If redirection cannot be established, CXone continues the call over hosted-desktop audio. Agents can keep working; the quality and network benefits of endpoint redirection are not present.

Infrastructure requirements

Each requirement below needs a named owner in the customer organization. NICE publishes the extension package, Receiver installers, update source, and matchmaking destinations used by this solution.

1. Supported endpoints and browsers

Agent physical endpoints must be one of:

Endpoint	Minimum version
Microsoft Windows	10 or 11, x64
Dell ThinOS	10.x, firmware 2508 or later
ChromeOS	Enterprise-managed (ChromeOS Enterprise Upgrade or equivalent)

The hosted browser where CXone Agent runs must be **Google Chrome** or **Microsoft Edge**.

A persistent hosted desktop is not required. In non-persistent environments, browser extension policy must apply on every new session.

HP ThinPro, IGEL, and macOS endpoints are outside the scope of this product documentation set.

Typical owner: Desktop / endpoint engineering

2. NICE browser extension (hosted session)

The NICE browser extension must be present in Chrome or Edge in the environment where CXone Agent runs.

Requirements:

- Force-install the published NICE extension by extension ID through enterprise browser policy (Group Policy, MDM, registry, or equivalent).
- Choose **one** distribution channel per browser profile: NICE enterprise update host **or** the NICE Chrome Web Store listing. Do not force-install both on the same profile.
- Apply the same policy on persistent and non-persistent hosted desktops so every session receives the extension.
- Allow the published NICE extension update source if store-only policy would otherwise block installation or updates.
- Ensure security policy does not block the approved extension package or update location.

Do not substitute, unpack, or redistribute alternate builds of the extension unless NICE instructs otherwise.

Typical owner: Browser / VDI policy

3. NICE WebRTC Receiver (physical endpoint)

The NICE WebRTC Receiver must be installed on each physical endpoint that will use the Receiver App path (as distinct from Citrix UCSDK or Omnissa Horizon paths).

Requirements:

- Deploy the published Receiver using the customer's existing software distribution (for example GPO, Intune, Dell Wyse Management Suite, or Chrome Enterprise).
- Keep the Receiver current when NICE releases updates.
- Allow the Receiver to execute under endpoint security / EDR / application control.
- Grant the Receiver access to the local microphone and audio-output devices (OS privacy settings and security software).

Typical owner: Endpoint management / security

4. Network and Security

Voice media leaves the **physical endpoint**, not the hosted desktop. Firewall and proxy rules that only cover the VDI / hosted browser path are not sufficient.

The following communications are required for the Receiver App path:

Source	Destination (FQDN / IP)	Protocol / Port	Purpose
Hosted browser	liaison.nice-receiver-app.com / 135.237.20.87	TCP 443 (HTTPS/WSS)	Session pairing and redirection control
Hosted browser	stun.nice-receiver-app.com (resolved A/AAAA)	UDP 19302	NAT discovery / ICE candidate gathering
Physical endpoint	liaison.nice-receiver-app.com / 135.237.20.87	TCP 443 (HTTPS/WSS)	Session pairing and redirection control
Physical endpoint	stun.nice-receiver-app.com (resolved A/AAAA)	UDP 19302	NAT discovery / ICE candidate gathering
Physical endpoint	This tenant's NICE CXone cluster	TCP 443	CXone / WebRTC control
Physical endpoint	This tenant's NICE CXone WebRTC infrastructure	UDP 1024–65535	Real-time voice media

NICE redirection service endpoints:

Service	FQDN	IP address	Port
Matchmaking	liaison.nice-receiver-app.com	135.237.20.87	TCP 443 (wss://liaison.nice-receiver-app.com/ws)
STUN	stun.nice-receiver-app.com	DNS A/AAAA via CNAME to stun.l.google.com (rotates)	UDP 19302
Documentation / downloads	docs.nice-receiver-app.com	Azure Static Web Apps A records (rotate)	TCP 443
Enterprise extension host	extensions.nice-receiver-app.com	Azure Static Web Apps A records (rotate)	TCP 443

Additional network facts:

- Allow STUN (stun.nice-receiver-app.com , UDP 19302) from the hosted browser (NICE extension) and from physical endpoints. Prefer FQDN-based allowlisting; STUN destination IPs change. There is **no TURN relay**. Networks that block STUN or direct WebRTC UDP will not get redirected media; the call may fall back to hosted-desktop audio.
- Use this tenant's **NICE CXone Connectivity Requirements** (FQDNs and IP ranges for the customer's own cluster, including FedRAMP where applicable). Do not reuse ranges from another environment.
- Review perimeter and endpoint firewalls, secure web gateways, VPN/SASE, explicit or transparent proxies, TLS inspection, DNS filtering, and UDP timeout / filtering policies.
- TLS inspection and proxies must not break certificate validation, HTTPS, WebSocket upgrade, or long-lived WSS sessions to the matchmaking service.
- Apply these rules on every network path agents use in production (office, home, VPN, branch), not only a lab segment.

Typical owner: Network / security

5. No competing WebRTC or media redirection

Only one WebRTC redirection or optimization technology may control the CXone WebRTC session.

Before go-live, identify and disable any other product that:

- Intercepts `RTCPeerConnection` or other WebRTC APIs
- Redirects browser multimedia or WebRTC media
- Virtualizes or replaces media-device APIs for the CXone application

Multiple interception mechanisms must not be enabled simultaneously for CXone. Choose one path: Citrix UCSDK, Omnissa Horizon, or the NICE Receiver App.

Typical owner: Collaboration / VDI / security

6. Audio devices on the physical endpoint

When redirection is active, headset and speaker selection happen on the physical endpoint.

Requirements:

- Intended headset (or other audio device) is connected to the physical endpoint and recognized by the OS.
- OS privacy controls allow microphone access.
- Endpoint security software does not block the Receiver from using audio devices.
- No other application exclusively locks the required devices.

Audio troubleshooting for redirected calls is performed on the physical endpoint, not inside the hosted desktop.

Typical owner: Desktop / operations

Readiness summary

A customer is ready to adopt NICE WebRTC Redirection when:

- Supported endpoint OS at the minimum versions above, and hosted Chrome or Edge, are in use
- The NICE browser extension is force-installed by policy on every hosted session (one distribution channel)
- The endpoint path for the chosen connection style is deployed (Receiver, Citrix Workspace App, or Horizon Client)

- Matchmaking (TCP 443) and STUN are open from the hosted browser and from physical endpoints where the Receiver path is used, and this tenant's CXone WebRTC paths (TCP 443 and UDP 1024–65535) are open from agent networks
 - Competing WebRTC / media-redirection products are not controlling the CXone session
 - Local microphone and audio output are available on the physical endpoint
-

Q&A

Business leaders

Why should we invest in WebRTC redirection?

Agent voice quality and latency improve because media is processed on the physical endpoint instead of being remoted through the hosted desktop. That reduces dependence on VDI audio remoting for real-time voice.

Does this change how agents use CXone?

No. Agents still work in CXone Agent in the hosted browser. Headset and speaker selection move to the physical endpoint when redirection is active.

What fails if we are not ready?

Calls can still complete using hosted-desktop audio (graceful fallback). You will not get the performance and network benefits of endpoint redirection until network, endpoint, and policy prerequisites are met.

Who owns readiness?

Name owners for browser policy, endpoint deployment, VDI/connection-style choice, network/firewall, and audio/operations. See the ownership notes in each requirement section above.

Must every site use the same connection style?

No. Choose one path per environment: NICE Receiver App, Citrix UCSDK, or Omnissa Horizon. Do not run multiple redirection technologies on the same CXone session.

IT administrators

Do we need new servers in our data center?

No. Matchmaking and STUN are NICE-hosted services. You deploy the browser extension (and Receiver or vendor client software) and open the documented outbound paths.

How is the browser extension installed?

By enterprise policy only — force-install one distribution channel (enterprise host or Chrome Web Store). Do not rely on users installing from the public store.

Can we keep our existing headset program?

Yes, if headsets connect to and are recognized by the physical endpoint OS. Audio troubleshooting for redirected calls is done on the endpoint, not inside the hosted desktop.

VDI administrators

Does this replace Citrix or Horizon?

No. CXone Agent still runs in the hosted session. Redirection only moves WebRTC media ownership to the endpoint (via Receiver, UCSDK/ HdxRtcEngine , or Horizon WebRTC Redirection).

How do we choose Receiver vs Citrix vs Horizon?

Use the path that matches your endpoint stack. Citrix and Horizon guides cover vendor-native offload; the Receiver App guide covers Windows, ChromeOS, and ThinOS where those vendor paths are not used.

Network administrators

What must be open for a pilot?

At minimum: matchmaking (`liaison.nice-receiver-app.com` / `135.237.20.87` , TCP 443), STUN (`stun.nice-receiver-app.com` , UDP 19302), and this tenant's CXone WebRTC HTTPS/UDP paths from agent networks. Full detail is in the [Customer Readiness and Validation Guide](#).

Does voice media go through NICE matchmaking?

No. Matchmaking is for pairing and control only. Voice media goes between the physical endpoint and NICE CXone.