Genesys Softphone Administrator's Guide

8/14/2025

Table of Contents

Search the table of all articles in this guide, listed in alphabetical order, to find the article you need.

Related documentation:

-

Genesys Softphone architecture

Contents

- Administrator

The architecture and features of Genesys Softphone when deployed in your environment.

Related documentation:

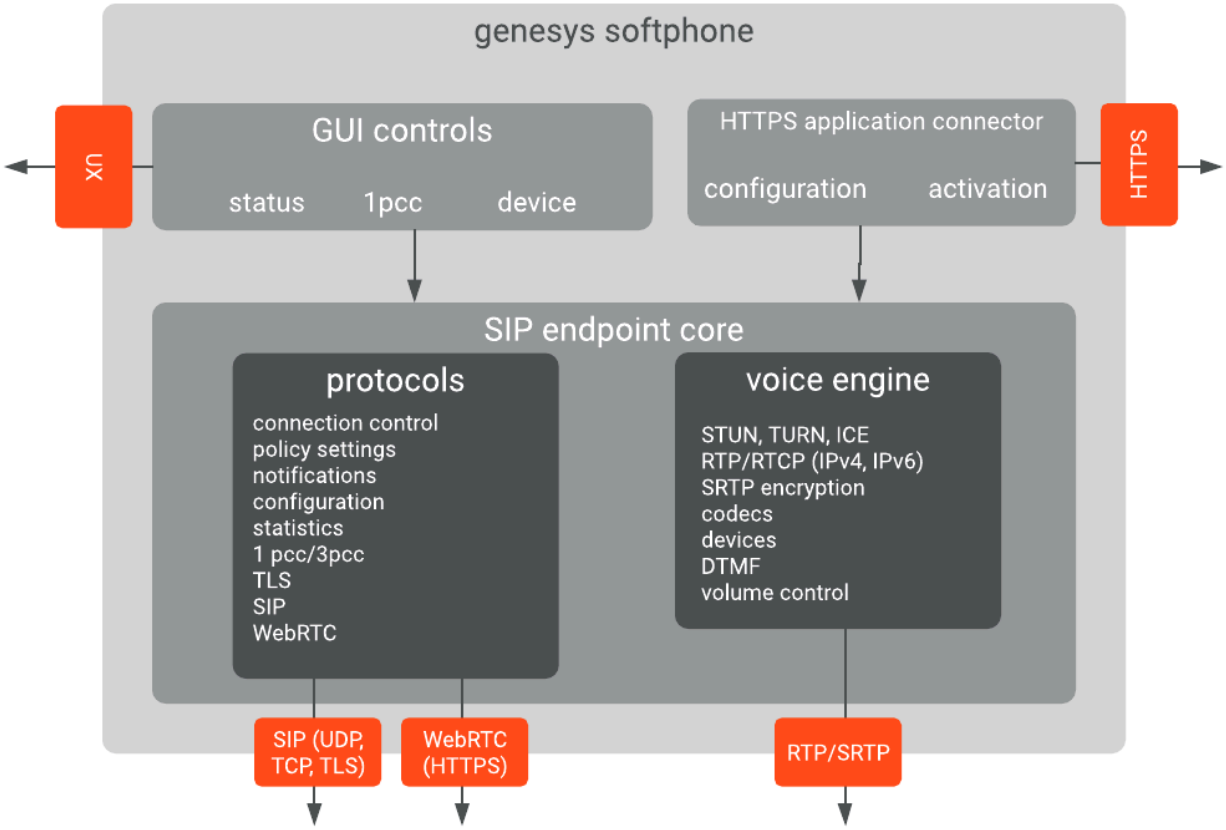
-

Architecture

Genesys Softphone embeds the Genesys SIP Endpoint Core Library to take advantage of the SIP-based third-party call control functionality.

Standard architecture

The following diagram illustrates the Genesys Softphone architecture when it is installed on a physical workstation as a standard executable gathering all product functionalities:



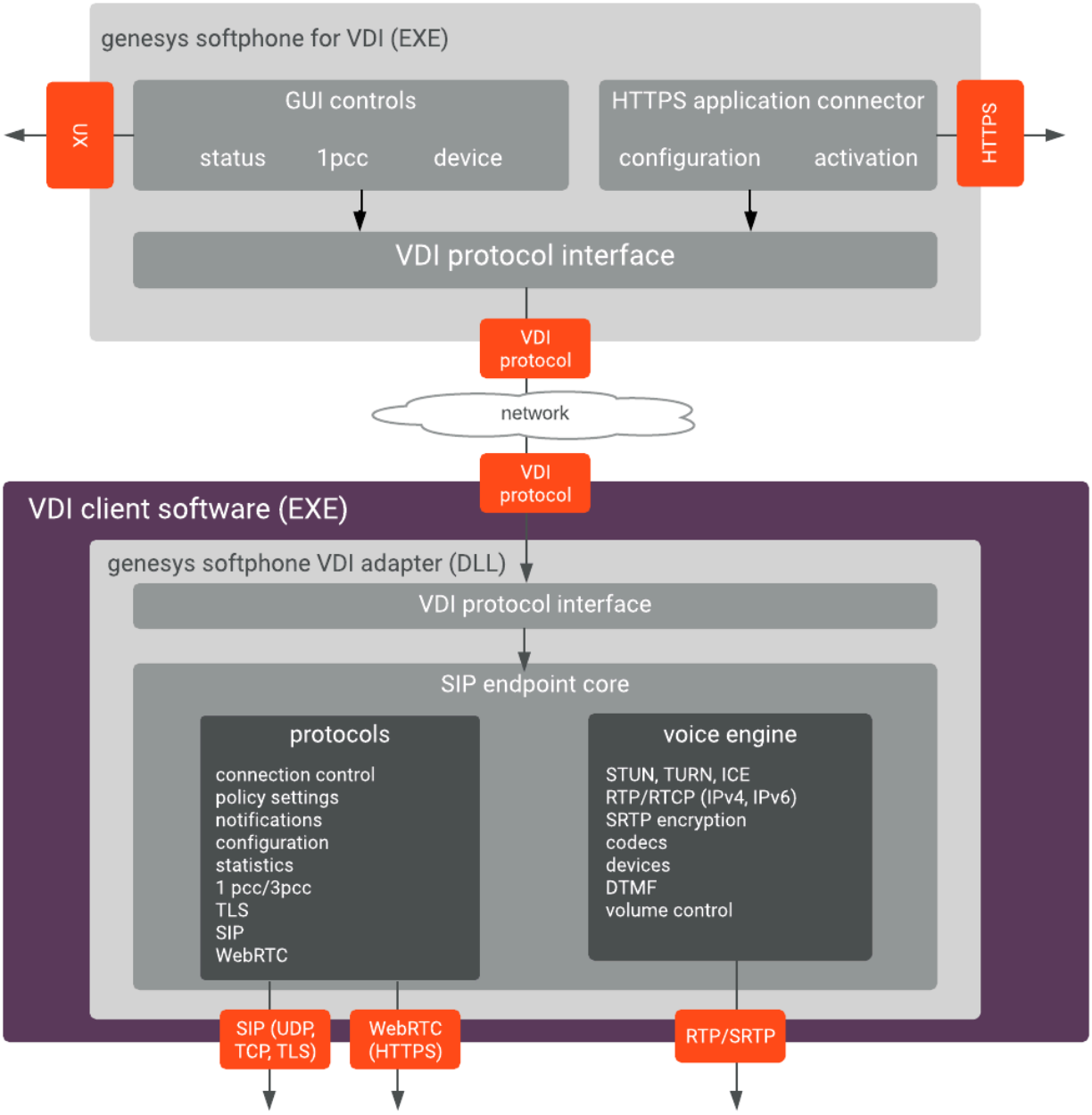
Architecture in VDI environments

Genesys Softphone supports the Citrix Virtual Desktop Infrastructure (VDI). When deployed in this kind of environment, the Genesys Softphone software is divided into two parts:

1. The application layer, running in the Virtualized system. This is the Genesys Softphone executable. The user interface runs here as well as connections to other applications, such as Agent Workspace. You install this through the Genesys Softphone installation package by selecting the Citrix installation option.
2. The Signaling Protocols, the Media Protocols, and the Audio Device management. These are off-loaded to the physical workstation to optimize call quality and ensure network and data center scalability. It is a plug-in (DLL) to the VDI Client run-time (Citrix Workspace app, previously known as Citrix Receiver). It is deployed by the Genesys Softphone VDI Adapter installation package.

The two Software parts communicate over the Citrix ICA proprietary protocol already established for standard Citrix operations; therefore, there is no need for any extra connectivity settings.

The following diagram illustrates the Genesys Softphone architecture in the Citrix VDI environment:



Features and functionality

Genesys Softphone media stack is based on Google's open source WebRTC Native Code package. Softphone includes an adaptive jitter buffer, Packet Loss Concealment (PLC), echo cancellation, and noise reduction. For more information refer to SDK for .NET.

The following are the standard features and functions of Genesys Softphone.

DTMF

The Genesys Softphone supports Dual-Tone Multi-Frequency (DTMF) signalling according to the RFC 2833 standard for third-party call control.

After receiving a NOTIFY with DTMF event, the Softphone Endpoint generates DTMF signals.

DTMF can be sent by using one of the three possible methods:

- InbandRTP
- RFC 2833
- SIP INFO message

Third-party call control

When the Genesys Softphone Endpoint has registered on the Genesys SIP Server, it will support the following third-party call control scenarios:

- Make a call
- Answer a call
- Hold and retrieve a call
- Single-step and two-step transfers
- Participate in a conference that is provided by the GVP
- Play DTMF signals.

SIP Voice

The Genesys Softphone supports the following codecs for SIP signaling:

- PCMU/8000 (G.711/mu-law)
- PCMA/8000 (G.711/A-law)
- G722/16000
- iLBC/8000 (iLBC — internet Low Bitrate Codec)
- iSAC/32000 ((iSAC/32kHz) — internet Speech Audio Codec)
- iSAC/16000
- G729/8000
- OPUS/48000/2

WebRTC Voice

The Genesys Softphone supports the following codecs for WebRTC signaling:

- OPUS
- G711

WebRTC and TLS in Windows 7

When Genesys Softphone is used for WebRTC communication, TLS 1.2 is used; however, Windows 7 does not support TLS 1.2 by default; therefore, you must enable TLS 1.2 in Windows 7 before you can use Genesys Softphone in WebRTC mode.

Refer to the following Microsoft document for the procedure to enable TLS 1.2: *Update to enable TLS 1.1 and TLS 1.2 as default secure protocols in WinHTTP in Windows*.

WebRTC and OAuth Support

WebRTC with OAuth is supported in Agent Workspace/Softphone connector mode only. If you use the standalone Softphone mode, you must migrate to connector mode. To enable this feature in Agent Workspace you must configure the value of the `sipendpoint.enable_webrtc_auth` option to `true`.

Virtual Desktop Infrastructure (VDI)

Softphone supports Virtual Desktop Infrastructure (VDI) to enable agents to use Softphone in a VDI environment.

Softphone can be deployed in a Citrix virtual environment.

- Prerequisites for installing Softphone in a Citrix VDI environment
- Installing the Genesys Softphone VDI Adapter

Localization

Starting from release 9.0.012.04, Genesys Softphone can be presented in various languages.

In Connector Mode, some agent applications like Workspace Web Edition can automatically align the language with the one selected in the controlling application. In other cases, the agent can select the language using the appropriate menu.

Deploying Genesys Softphone

Contents

- Administrator

How to deploy and configure the Genesys Softphone in your environment, including both standard and Virtual Desktop Infrastructure (VDI) installations.

Related documentation:

-

> **Important**
>
> You should receive access to the Genesys Softphone download when you purchase Softphone. Contact your Genesys representative if you did not receive access to the installation package.

Environment prerequisites

Ensure that your environment meets the prerequisites described in the following sections. This section covers the prerequisites for Genesys Softphone 9.0.1 and lower versions as well.

There is a prominent difference in using **Visual C++ Redistributable Packages for Visual Studio 2019** (required for Genesys Softphone 9.0.1) and **Visual C++ Redistributable Packages for Visual Studio 2013** (required for Genesys Softphone 9.0.0). These differences are noted in the respective procedures for user's convenience. Unless explicitly noted, all other prerequisites are required for installing Genesys Softphone in the environment you chose to deploy the software.

Supported operating systems

Refer to the Genesys Softphone and the Virtualization Platform Support topics in the *Genesys Supported Operating Environment Reference Manual* for a list of the latest supported operating systems.

Prerequisites for a full deployment of Genesys Softphone on a physical workstation

To work with Genesys Softphone, you must ensure that your system meets the software requirements established in the *Genesys Supported Operating Environment Reference Manual*, and meeting the following minimum software requirements:

- Visual C++ Redistributable Packages for Visual Studio 2019 (or above) is required for Genesys Softphone VDI Adapter 9.0.1:
  - The 64-bit version of the redistributable is installed by the Genesys Softphone 64-bit installer.

- Visual C++ Redistributable Packages for Visual Studio 2013 is required for Genesys Softphone VDI Adapter 9.0.0:
  - The 32-bit version of the redistributable is installed by Genesys Softphone 32-bit installer.
  - The 64-bit version of the redistributable is not installed by the Genesys Softphone 64-bit installer, so you must install `vcredist64.exe` from the above link when the 64-bit version of Genesys Softphone is installed.
- .NET Framework 4.0 or higher: This is used at installation time only when the Administrator installs Genesys Softphone with HTTPS connector based on a *self-signed certificate*.
- QoS requirement for voice, either one-to-one or on a conference connection capability, require the following:
  - ≤ 150 ms of one-way latency from mouth to ear (per the ITU G.114 standard)
  - ≤ 30 ms jitter
  - ≤ 1 percent packet loss
  - 17 to 106 kbps of guaranteed priority bandwidth per call (depending on the sampling rate, codec, and Layer 2 overhead)
  - 150 bps (plus Layer 2 overhead) per phone of guaranteed bandwidth for voice control traffic

> **Important**
>
> QoS policies are managed by the operating system. To configure a QoS policy in Windows, refer to Quality of Service (QoS) Policy in the Microsoft documentation.

- A headset or other microphone and speaker audio device that is supported by Windows OS installed on the workstation.

Prerequisites for deployment in a VDI environment

To work with Genesys Softphone in a VDI environment, you must ensure that your system meets the software requirements established in the *Genesys Supported Operating Environment Reference Manual*, as well as meeting the following minimum software requirements:

1. On the workstation running the VDI client:
   - For Citrix Workspace (formerly Citrix Receiver) for Windows (applicable to Genesys Softphone VDI Adapter 9.0.0 and 9.0.1):
     - Visual C++ Redistributable Packages for Visual Studio 2013 (32-bit version). The Genesys installation package 32-bit installs this redistributable package on the workstation where it is executed.
   - For VMWare Horizon on Windows,
     - Visual C++ Redistributable Packages for Visual Studio 2019 (or above) is required for Genesys Softphone VDI Adapter 9.0.1. The 64-bit package of the Genesys Softphone VDI Adapter installs this redistributable package on the workstation where it is executed.
     - Visual C++ Redistributable Packages for Visual Studio 2013 is required for Genesys Softphone version 9.0.0. You must install the `vcredist64.exe` from the link location before running the

Genesys Softphone VDI Adapter for VMWare Horizon.

- Quality of service (QoS) for voice, either one-to-one or on a conference connection capability, requires the following:
  - ≤ 150 ms of one-way latency from mouth to ear (per the ITU G.114 standard)
  - ≤ 30 ms jitter
  - ≤ 1 percent packet loss
  - 17 to 106 kbps of guaranteed priority bandwidth per call (depending on the sampling rate, codec, and Layer 2 overhead)
  - 150 bps (plus Layer 2 overhead) per phone of guaranteed bandwidth for voice control traffic

> **Important**
>
> QoS policies are managed by the operating system. To configure a QoS policy, refer to the documentation for your platform.

- A headset or other microphone and speaker audio device that is supported by the OS installed on either the client or the host.

2. On the workstation used to build deployments for running eLux systems:
   - Virtual Driver for Citrix shared object.
   - **libgsecurity** module.
   - Scout Enterprise ELIAS tool.
   - Windows workstation from which to run the Genesys Softphone VDI Adapter executable.
3. On the workstation used to build deployments for HP ThinPro systems:
   - The packaging tool recommended by your HP vendor.
4. On the VDI environment (Citrix Virtual Application/Desktop or VMWare Horizon server) that runs the application layer of the Genesys Softphone runtime:
   - Visual C++ Redistributable Packages for Visual Studio 2019 (or above) is required for Genesys Softphone version 9.0.1:
     - The 64-bit version of the redistributable is installed by the Genesys Softphone 64-bit installer.
   - Visual C++ Redistributable Packages for Visual Studio 2013) is required for Genesys Softphone version 9.0.0:
     - The 32-bit version of the redistributable is installed by Genesys Softphone 32-bit installer.
     - The 64-bit version of the redistributable is not installed by the Genesys Softphone 64-bit installer, so you must install `vcredist64.exe` from the above link when the 64-bit version of Genesys Softphone is installed.
   - .NET Framework 4.0 or higher: This framework is used only when the administrator installs Genesys Softphone with an HTTPS connector based on a *self-signed certificate*.
5. In the Citrix Virtual Application/Desktop settings:
   In Citrix Virtual Apps and Desktop LSTR 2203, the **virtual channel allow list** feature is enabled by

default. This means that in this Citrix release, and any older deployment where administrator explicitly enables this feature, only the Virtual Channels shipped by Citrix are authorized by default.
In this deployment, the administrator must explicitly authorize the Genesys Softphone Virtual Channel to make Genesys Softphone for VDI operate successfully. This can be done using the instructions on the Citrix Virtual Channel Security page. As a Citrix Administrator, you need to add a Virtual channel allow list policy and configure it with the value:
`GENESYS,\GenesysSoftphone_Citrix.exe`

Important

To use Workspace Web Edition and Genesys Screen Recording Service with Genesys Softphone in a VDI environment such as Citrix XenApp, you must configure the ***screen-recording-client-address*** option to point to the SRS Loopback address.

Installing Genesys Softphone for Windows

(For information on installing Genesys Softphone in a VDI environment sees Installing the Genesys Softphone VDI Adapter)

Tip

Beginning with Genesys Softphone 9.0.020.08, multiple installation packages are available from the download center, 32-bit and 64-bit. Ensure that you download the correct package for your environment.

To install Genesys Softphone, follow these steps:

1. Download the Genesys Softphone installation package.
2. Double-click the **setup.exe** file that is located in the **\windows** directory to open the **Genesys Installation Wizard**.
3. In the **Welcome to the Installation** window, click **Next**.
4. In the **Choose Destination Location** window, click **Next** to accept the default destination folder, or click **Browse** to select another destination location.
5. In the **Deployment Type** window, click **Standard** or **VDI: Citrix or VMware Horizon** (for Virtualization deployments only), and then click **Next**.
6. In the **Startup and Secure Connection options** window, you may choose one or more of the following options, and then click **Next**:
   - Auto Startup: Specifies that Genesys Softphone launches when Windows starts up. This means that agents do not have to manually launch Genesys Softphone before they launch Agent Workspace.

- Enable Dynamic Configuration Connector: Specifies that Agent Workspace is allowed to dynamically configure Genesys Softphone when it is launched.

  If you choose the Enable Dynamic Configuration Connector option, the **Dynamic Configuration Connector Parameters** window is displayed.

  a. Specify the Connector Port for Genesys Softphone. This port must be compliant with the value specified by the sipendpoint.uri option.

  b. Enable HTTPS secure connections (optional). If you choose a secure connection, you must choose the type of security certificate that you use:

  - Self-signed Certificate: In this mode, the IP creates a self-signed certificate, installs it in the Personal Certificate section of the workstation where **setup.exe** is executed and also installs it as a root certificate authority at Machine level in the workstation where **setup.exe** is executed.
  - Certificate Authorities from the Windows Certificate Store

> **Important**
>
> To properly install the self-signed certificate, .NET Framework 4.0 or higher is mandatory.

7. In the **Ready to Install** window, select **Install**. The wizard installs Genesys Softphone and all associated files in the selected directory and displays the **Installation Status** window. The installation might take several minutes.
8. In the **Installation Complete** window, select **Finish**.

> **Important**
>
> For more information about Genesys Softphone deployment for Agent Workspace, see Single sign on with Agent Workspace.

Installing Genesys Softphone in Silent mode for Windows

To install Genesys Softphone in Silent mode, use the Installation Wizard **Silent** arguments as follows:

1. Update the **genesys_silent.ini** file by making the following modifications:
   - Add the path to the Genesys Softphone directory. For example, **InstallPath=C:\GCTI\Genesys Softphone**.
   - Specify if Genesys Softphone is a physical workstation ("Std") or on a desktop/application virtualization environment (Citrix or VMWare Horizon) ("Citrix") by using the **DeploymentType=** parameter.
   - Specify whether Genesys Softphone starts automatically when Windows starts up by using the **Startup=** parameter.
   - Specify whether Agent Workspace can dynamically modify the Genesys Softphone configuration by using the **Connector=** parameter.

- If you are *deploying* Softphone for Agent Workspace dynamic configuration:
  - If the Connector is enabled, specify the Connector Port by using the **ConnectorPort=** parameter.
  - Specify whether the connector uses HTTPS secure connection by using the **HTTPS=** parameter.
  - If you are using a secure connection, specify the certificate type to be used by using the **CertificateType=** parameter.
  - If you assign the value **WindowsStore** to the **CertificateType** option, specify the certificate thumbprint by using the **CertThumbPrint=** parameter.
- If you are *upgrading* Genesys Softphone, specify:
  - **IPVersion=**
  - **IPBuildNumber=**

2. Execute the following command:
   `setup.exe /s /z"-s 'FullPathToGenesysSilentConfigurationFile' -sl 'FullPathToGenesysSilentResultFile'"` where:
   - `/s` specifies that the installation is running in InstallShield Silent Mode.
   - `/z` passes the Genesys Silent mode silent parameters to the installation.
   - `-s` specifies the full path to the silent configuration file. The is optional. If the parameter is not specified, the installation uses the **genesys_silent.ini** file in the same directory where the **setup.exe** is located.

     > **Important**
     > Enclose the value of the parameter by apostrophes (') if the parameter contains white symbols.

   - `-sl` specifies the full path to the installation results file. If the parameter is not specified, the installation creates the **genesys_install_result.log** file in the directory.

     > **Important**
     > Enclose the value of the parameter in apostrophes (') if the parameter contains white space characters.

   The **InstallShield setup.exe** installation starter requires that:
   - There is *no* space between the `/z` argument and quotation mark. For example, `/z"-s"` is valid, while `/z "-s"` is not valid.
   - There *is* a space between the -s,-sl parameters and quotation mark. For example, `/z"-s c:\temp\genesys_silent.ini"` is valid, while `/z "-sc:\temp\genesys_silent.ini"` is not valid. For example,
     `setup.exe /s /z"-s 'C:\8.5.000.05\windows\b1\ip\genesys_silent.ini' -sl 'C:\GSP\silent_setup.log'"`.
3. After executing this command, verify that Genesys Softphone is installed in the **C:**, and that the **silent_setup.log** file has been created in the **C:\GSP** directory.

Installing the Genesys Softphone VDI Adapter (Windows)

If you installed Genesys Softphone in a VDI environment, you must install the Genesys Softphone VDI Adapter on each workstation by following these steps:

1. From the Genesys Download Center, download the Genesys Softphone VDI Adapter package corresponding to your environment:
   - For Citrix, use the Genesys Softphone VDI Adapter 32-bit package
   - For VMWare Horizon, use the Genesys Softphone VDI Adapter 64-bit package
2. Double-click the **setup.exe** file that is located in the **\windows** directory to open the **Genesys Installation Wizard**.
3. In the **Welcome to the Installation** window, click **Next**.
4. In the **Select Operating System** window, select the option that is appropriate for your environment. For Citrix environments, select **Citrix support on Windows**. For VMware Horizon environments, select **VMware Horizon support on Windows**. Click **Next**.
5. If the **Choose Destination Location** window is presented (if installing on VMware Horizon), click **Next** to accept the default destination folder, or click **Browse** to select another destination location.
6. In the **Ready to Install** window, select **Install**. The wizard installs Genesys Softphone VDI Adapter and displays the **Installation Status** window.
7. In the **Installation Complete** window, select **Finish**.

Installing the Genesys Softphone VDI Adapter (eLux)

If you installed Genesys Softphone in a VDI environment, you must install the Genesys Softphone VDI Adapter on each eLux workstation by following these steps:

1. From the Genesys Download Center, download the Genesys Softphone VDI Adapter package 32-bit or 64-bit.
2. Double-click the **setup.exe** file located in the **\windows** directory to open the **Genesys Installation Wizard**.
3. In the **Welcome to the Installation** window, click **Next**.
4. In the **Select Operating System** window, select **eLux**, specify the destination to install the installation package, and click **Next**.
5. In the **Ready to Install** window, select **Install**. The wizard installs Genesys Softphone VDI Adapter and displays the **Installation Status** window.
6. In the **Installation Complete** window, select **Finish**.
   The installation package installs the following items:
   - a Virtual Driver for Citrix shared object
   - a **libgsecurity** module
   - a startup script to update the Citrix **module.ini** config file.

These files are packaged into an EPM/FPM pair, each with a separate signature file with four files for the VD package and three files with certificates used for signing:

- **genesys_vd-.UC_RP5-1.0.fpm**
- **genesysvd-.UC_RP5-1.0.epm**
- **genesys_vd-.UC_RP5-1.0.fpm.sig**
- **genesysvd-.UC_RP5-1.0.epm.sig**

- **0-VeriSign-RootCA.cer**: VeriSign Universal Root Certification Authority
- **1-Symantec-intermediate.cer**: Symantec Class 3 SHA256 Code Signing CA
- **2-Genesys-codesign.cer**: Genesys certificate used for signing packages

7. Import the package files to the existing container and add them to the client image using the Unicon Scout Enterprise ELIAS tool:
   1. Using the **Security / Manage certificates** menu option, import the certificates as trusted.
   2. If the client is configured with **signature check**, the VeriSign Root CA certificate must be installed on each client in the **/setup/cacerts** folder.
   3. To add packages to the container, in ELIAS select the **Container / Import Package** menu option, and then select the files with the **.epm** extension.
   4. To update the image definition file (IDF), open it in ELIAS, then add the new package by selecting **Genesys VD for Citrix,** in the right pane and press the **button.**
   5. Update the client workstation using the Scout Enterprise Console and perform these steps:
      - Check the firmware configuration of the relevant Thin Clients by selecting **Device configuration** and then choosing **Firmware**.
      - Update the device by selecting the **Commands / Update** option to initiate the update and force a device restart.

Installing the Genesys Softphone VDI Adapter (HP ThinPro)

> **Important**
>
> Beginning with version 9.0.1, Genesys Softphone VDI Adapter for HP ThinPro supports both Citrix and VMWare Horizon environments. Note that the 9.0.0 version supports only VMWare Horizon.

If you installed Genesys Softphone in a VDI environment, you must install the Genesys Softphone plugin for VMWare Horizon and Citrix Client on each HP ThinPro workstation by following these steps:

1. From the Genesys Download Center, download the Genesys Softphone VDI Adapter package 32-bit or 64-bit.

2. To open the **Genesys Installation Wizard**, double-click the **setup.exe** file located in the **\windows** directory.
3. In the **Welcome to the Installation** window, click **Next**.
4. In the **Select Operating System** window, select **HP ThinPro**, specify the destination to install the installation package, and click **Next**.
5. In the **Ready to Install** window, select **Install**. The wizard installs Genesys Softphone VDI Adapter and displays the **Installation Status** window.
6. In the **Installation Complete** window, select **Finish**.
   The installation package installs the following file for the Genesys Softphone plugin for VMWare Horizon and Citrix Client.

   This file is packaged into a DEB file:

   - **vdhgenesys__amd64.deb**
7. Deploy the Genesys Softphone plugin for VMWare Horizon and Citrix Client on ThinPro host:
   - Manual deployment:
     1. Copy the client-side package to any convenient location accessible by wget or scp.
     2. From the main menu, switch to **Administrator** on the ThinPro host.
     3. To copy the client-side package on ThinPro host, start **Xterm**.
     4. To install the client-side package, run the following commands:
        ```
        fsunlock

        dpkg -i vdhgenesys__amd64.deb

        fslock
        ```
   - Bulk deployment:
     Use the packaging tools and procedures provided by HP to build a new ThinPro OS package that contains the Genesys Softphone VDI Adapter Debian package, then distribute it to the HP ThinPro hardware boxes.

Installing the Genesys Softphone VDI Adapter in Silent mode

To install Genesys Softphone VDI Adapter in Silent mode, use the Installation Wizard **Silent** arguments as follows:

1. Update the **genesys_silent.ini** file by making the following modifications:
   - Specify if Genesys Softphone VDI Adapter should be installed for Windows (Citrix) ("citrix_windows", only on the 32-bit version), Windows (VMware Horizon) ("vmware_horizon", only on the 64-bit version), eLux ("citrix_elux_5"), or HP ThinPro OS (Citrix and VMware Horizon) ("vmware_pro") by using the **DeploymentType** parameter. For example, **DeploymentType=citrix_windows**.

     > **Important**

> Beginning with version 9.0.1, the Genesys Softphone VDI Adapter for HP ThinPro supports both Citrix and VMWare Horizon environments. The 9.0.0 version supports only VMWare Horizon.

- If installing on VMware Horizon, you can choose to install to a specific location by specifying the path with the **InstallPath** parameter. For example, **InstallPath=C:\GCTI\Genesys SoftphoneVDIAdapter**.
- If installing on eLux5, add the path to the Genesys Softphone VDI Adapter directory using the **InstallPath** parameter. For example, **InstallPath=C:\GCTI\Genesys SoftphoneVDIAdapter**.

2. If you are *upgrading* Genesys Softphone VDI Adapter, specify:
   - **IPVersion=**
   - **IPBuildNumber=**
3. Execute the following command:
   `setup.exe /s /z"-s 'FullPathToGenesysSilentConfigurationFile' -sl 'FullPathToGenesysSilentResultFile'"` where:
   - `/s` specifies that the installation is running in InstallShield Silent Mode.
   - `/z` passes the Genesys Silent mode silent parameters to the installation.
   - `-s` specifies the full path to the silent configuration file. The is optional. If the parameter is not specified, the installation uses the **genesys_silent.ini** file in the same directory where the **setup.exe** is located.

     > **Important**
     > Enclose the value of the parameter by apostrophes (') if the parameter contains white symbols.

   - `-sl` specifies the full path to the installation results file. If the parameter is not specified, the installation creates the **genesys_install_result.log** file in the directory.

     > **Important**
     > Enclose the value of the parameter in apostrophes (') if the parameter contains white space characters.

   The **InstallShield setup.exe** installation starter requires that:
   - There is *no* space between the `/z` argument and quotation mark. For example, `/z"-s"` is valid, while `/z "-s"` is not valid.
   - There *is* a space between the -s,-sl parameters and quotation mark. For example, `/z"-s c:\temp\genesys_silent.ini"` is valid, while `/z "-sc:\temp\genesys_silent.ini"` is not valid. For example,
     `setup.exe /s /z"-s 'C:\9.0.007.03\windows\b1\ip\genesys_silent.ini' -sl 'C:\GSP\silent_setup.log'"`.
4. After executing this command, verify that Genesys Softphone VDI Adpater is installed in the expected

directory, and that the **silent_setup.log** file has been created in the **C:\GSP** directory.

Installing Genesys Softphone for macOS

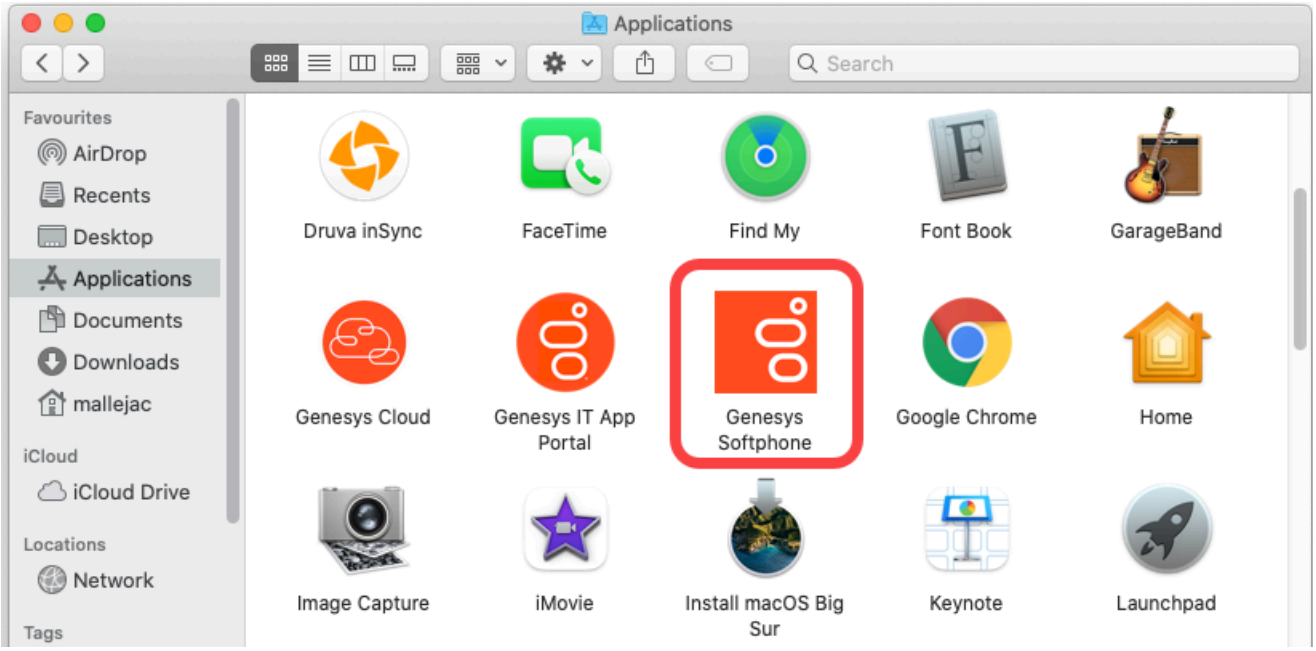
To install the Genesys Softphone for macOS:

1. Download the Genesys Softphone installation package.
2. Open a Terminal session. From the **/mac/bX/ip** directory path, run the **install.sh** script using administrator privileges:
   ```
   sudo ./install.sh
   ```

> ### Important
> If the installation fails due to insufficient macOS user privileges (for example, you receive an error message due to "unidentified developers"), you can perform the following workaround:
>
> - Open a Terminal session and run the following command: `sudo spctl --master-disable`. This allows applications from unidentified developers to be opened.
> - Execute the Genesys Softphone installation script.
> - When the installation is complete, run the command `sudo spctl --master-enable` to revert the system back to the default privilege settings.

3. At the **Launch Genesys Softphone on MacOS startup** prompt, enter **y** to enable Softphone to run automatically when the agent is opening the OS session or **n** for Softphone to be started manually by agent.
4. At the **Enable connector to allow dynamic configuration by Workspace Web Edition** prompt, enter **n** to select standalone mode or **y** to enable the connector.
   - If you are upgrading Genesys Softphone in standalone mode, you can then enter **n** to overwrite the existing configuration file (**Softphone.config**) or **y** to continue using the existing configuration file.
   - If you choose to enable the connector, confirm the default connector port number (`8000`), or enter the port number you want to use. You can then enter **y** to enable a secure connection (HTTPS) or **n** to use a non-secure connection (HTTP).
5. Enter **y** to accept the destination directory for the installation and continue.
6. After the installation process completes, the script displays messages to confirm the following:
   - The **Tuning** file attributes are automatically tuned.
   - If you enabled the connector with a secure connection (HTTPS), the RSA private key certificate is automatically created and installed.

You can launch Genesys Softphone from the **Applications** folder:



Installing Genesys Softphone in Silent mode for macOS

To install the Genesys Softphone in Silent mode:

1. Update the **genesys_silent.ini** file by making the following modifications:
   - Add the absolute path to the Genesys Softphone directory. For example, **InstallPath= /Applications/Genesys Softphone.app**.
   - Specify whether Genesys Softphone starts automatically when MacOS starts by setting the **AutoStart=** parameter.
   - Specify whether Workspace Web Edition can dynamically modify the Genesys Softphone configuration by setting the **EnableConnector=** parameter.
   - If you are *deploying* Genesys Softphone for Workspace Web Edition dynamic configuration:
     - If the Connector is enabled, specify the Connector Port by setting the **ConnectorPort=** parameter.
     - Specify whether the connector uses HTTPS secure connection by setting the **SecuredCommunication=** parameter.
   - Specify whether to keep the existing configuration file during upgrades by setting the **PreserveConfigFile=** parameter. If you enter no for this value, the configuration file is overwritten during the upgrade. (This parameter is ignored if the Connector is enabled.)
2. Enter the following command using administrator privileges:

```
sudo ./install.sh -s -fr ///mac/bX/ip/genesys_silent.ini -fl ///mac/bX/ip/
genesys_install_result.log
```

where

- is the path to the installation package.
- is the version of the installation package version you are installing. For example, `9.0.014.12`.

Configuring Genesys Softphone

Genesys Softphone installation includes a configuration file (**/Softphone.config>**) with configuration settings that are applied to the Softphone when it starts.

> **Important**
> You can make changes to the configuration file, but you must restart the Softphone before any of the changes take effect.

The configuration file is organized into *containers*. Each container is divided into *domains* that are further divided into *sections* that hold the *settings* for a group of parameters. The following configuration file examples describe the settings in each container:

For the description and valid values of each parameter, see Configuration Options Reference.

Basic container

The Basic container sets the Genesys Softphone user's DNs and the protocol used.

> **Important**
> If Single sign on is used with Agent Workspace, these parameters in configuration file are not taken in account.

Genesys container

The Genesys container sets the policy, endpoint, session, device, connector, codecs, proxy, mailbox, system, and security parameters.

Configuring the agent's DN

Set the following TServer section option for the DNs of the Place to which the agent is logging in:

- `sip-cti-control = talk,hold,dtmf`

> **Important**
> This option is mandatory to use third-party call control on the SIP device.

For information about configuring DN objects, see DNs in the Agent Setup documentation.

Configuring SIP Server

Genesys recommends setting the following SIP Server options:

- `dual-dialog-enabled=true` (default value)
- `make-call-rfc3725-flow=1` (allows for better and/or simpler codec negotiation)
- `ring-tone-on-make-call=true` (default value)
- `use-register-for-service-state=true`

For more information about these options, see the SIP Server Deployment Guide.

Suppressing the ringtone

The ringtone is generated for all incoming calls to Genesys Softphone. To suppress the ringtone for third-party call control for the originating DN, configure the following SIP Server option:

- `make-call-alert-info=`

or

- `make-call-alert-info=;service=3pcc`

> **Important**
> If at least one Genesys Softphone in the contact center is configured with the `ringing_enabled` option set to 1, the SIP Server `make-call-alert-info` option should be set to one of the values specified above.

Mass deployment on multiple workstations

Genesys Softphone can be distributed to end user workstations through Desktop Configuration Management solutions, such as Microsoft Endpoint Manager (formerly Microsoft SCCM), that enable mass deployments of desktop software.

Although Genesys Softphone is not shipped with a Microsoft Software Installer (MSI), you can repackage **Genesys Softphone setup** into an MSI using the silent installation mode. Next, use a repackaging tool, such as https://www.exemsi.com/ to repackage the MSI. Finally, deploy the resulting MSI package using your Desktop Configuration Manager.

Single sign on with Agent Workspace

Contents

- Administrator

How to configure Genesys Softphone for single sign-on with Agent Workspace.

Related documentation:

-

Genesys Softphone includes an HTTP/HTTPS connector to simplify using Genesys Softphone with Agent Workspace:

- Single sign-on—Agent Workspace controls the SIP settings for Softphone based on explicit Agent Workspace centralized options and agent login credentials (Place and DN).
- Simplified deployment—each agent workstation runs the same application and configuration files, avoiding workstation specific configuration.
- Password authentication—Agent Workspace passes the DN password as one of the parameters through the Genesys Softphone connector to allow the Softphone to securely login to SIP Server and avoid the need for MPLS.

Configuring Softphone for Agent Workspace

Softphone.config configuration file contains a **connector** section in the **policy** domain:

```
...
        ...
        ...
...
```

> **Important**
>
> You can enable Agent Workspace options for Softphone through Agent Setup.

Codec priority

Use the **enabled** section of the **codecs** domain in the **Softphone.config** configuration file to specify the order in which audio codecs are given priority.

> **Tip**
>
> For more details, refer to Working with Codec Priorities in the *SIP Endpoint SDK Developer's Guide 9.0.0NET*.

For example:

> **Warning**
>
> Any codec that is not explicitly included in the **enabled** section will not be used, even if the section for that codec is present in the configuration file.

To use the Genesys SIP Endpoint SDK 9.0 **enabled** section of the **codecs** domain, follow these guidelines:

- Codec names are *case-insensitive*. You can omit the clock rate portion of the section name unless needed to discriminate between two sections with the same name. The clock rate portion must be provided for **iSAC**.
- Specify codec parameters as a comma-separated list in parenthesis after an equals sign. You can use abbreviations such as "pt" for "payload_type".
- If there are codec conflicts, the value in the **enabled** section takes precedence over value in corresponding codec section, regardless of whether those values come from the configuration file. For example:

- If codec parameters are specified in-line (or a particular codec does not require any parameters, such as the PCMU and PCMA codecs), then a separate codec section is not necessary. In any case, codecs specified in the "enabled" section do not require presence of corresponding section to take effect.

Signing on with Agent Workspace

Before starting Agent Workspace, agents need to have Softphone running. Administrators can specify that Softphone starts automatically when the Windows user logs in or agents can startup Softphone.

User interface and call controls

When using Softphone with Agent Workspace, Softphone disables its default user interface. Instead, agents can use the Agent Workspace user interface for call controls, mute, and volume control. For information on the Agent Workspace user interface, see the Agent Workspace Help.

> **Important**
> In this mode, Genesys Softphone does not prevent the Agent Workspace application from claiming WCAG 2.1 levels A and AA when it supports this standard for its own UI.

Configuration options reference

Contents

- Administrator

Configuration settings by container and domain found in the Softphone.config file in the Genesys Softphone Installation Directory.

Related documentation:

-

For an example of the configuration file, see Configuring Genesys Softphone.

The **Softphone.config** file is installed, along with **genesys_softphone.exe**, by either the **Genesys Installation Wizard** or silently by command line. The contents of the **Softphone.config** file is generated by the choices specified in the wizard or by modifications made to the **genesys_silent.ini** file.

In the **Softphone.config** file, the following attributes of the **Connector** section are set by **setup.exe**: `protocol`, `port`, and `certificate_search_value`, while `enable_sessionid` and `auto_restart` are not. The default value of these attributes are designed to address most business deployments. However, if you want to adjust their values, follow these steps to make a custom deployment:

1. Install Genesys Softphone on an administrator's machine.
2. Edit the **Softphone.config** file to change the values of the attributes in the **Connector** section.
3. Repackage Genesys Softphone with the custom **Softphone.config** file through an IT-controlled installation.
4. Push the custom package to the agent workstations.

Basic Container

The first container ("Basic") holds the basic connectivity details that are required to connect to your SIP Server, optionally through a Session Border Controller (SBC). This container has at least one connection (Connectivity) element with the following attributes:

If you are using a configuration that supports Disaster Recovery and Geo-Redundancy, there may be multiple connection elements present with each specifying a separate possible connection.

You must make the following changes and save the updated configuration file before using Genesys Softphone:

- `user="DN"`: Supply a valid DN for the user attribute.
- `server="SERVER:PORT"`: Replace SERVER with the host name where your SIP Server or SBC is deployed, and PORT with the SIP port of the SIP Server or SBC host. The default SIP port value is 5060. For SRV

resolution, specify the SRV record without including the port number in the server's URI. Also see SRV Resolution below.

- `protocol="TRANSPORT"`: Set the protocol attribute to reflect the protocol being used to communicate with SIP Server or SBC. Possible values are `UDP`, `TCP`, or `TLS`.

SRV resolution

When using an SRV record for the **server** parameter, note the following:

- Do not specify the port in the server URI.
- Genesys Softphone does not take into account the **weight** field of an SRV record.
- You cannot combine IPv4 and IPv6 for a single FQDN.
- The maximum number of targets (SRV records) per service is 20.
- You can only specify SRV records in the **server** parameter of the **Connectivity** element. You cannot use SRV records for the mailbox section or the **vq_report_collector** setting.

WebRTC

You will have to make the following changes and save the updated configuration file before using the Genesys Softphone:

- `user="DN"`—Supply a valid DN for the user attribute.
- `server="WEBRTCGATEWAY_SERVER:WEBRTCGATEWAY_PORT?sip-proxy-address="SIPPROXY_SERVER:SIPPROXY_PORT"`—Replace WEBRTCGATEWAY_SERVER with the host name where the WebRTC Gateway is deployed, and PORT with the HTTPS port of the WebRTC Gateway. Also, replace SIPPROXY_SERVER and SIPPROXY_PORT (optional) with the connectivity parameters of the SIP Proxy that need to be contacted by the WebRTC Gateway to register this DN.
- `protocol="TRANSPORT"`—Set the protocol attribute to reflect the protocol being used to communicate with the WebRTC Gateway: HTTPS.

Important

Your environment can have up to six SIP URIs (Connectivity sections) that represent six endpoint connections with SIP Server.

Domain	Section	Setting	Default Value	Description
	Connectivity	user		The first user's DN extension as configured in the configuration database. Included in the SIP URI—for example, DN0@serverHostName0:port0>
		server		The SIP Server or

Domain	Section	Setting	Default Value	Description
				Proxy location for the first user. Included in the SIP URI—for example, serverHostName0:port0>
		protocol		The transport protocol for the first user. For example, UDP, TCP, or TLS.

Genesys Container

The second Container ("Genesys") holds a number of configurable settings that are organized into domains and sections. You don't have to change these settings but you can customize them.

An overview of the settings in this container and the valid values for these settings is provided here:

Domain	Section	Setting
policy		
	endpoint	
		audio_qos
		include_os_version_in_user_agent_header
		include_sdk_version_in_user_agent_header
		ip_versions
		public_address
		include_mac_address
		refer_to_proxy
		rtp_inactivity_timeout
		rtp_port_min
		rtp_port_max
		tcp_port_min
		tcp_port_max
		signaling_qos
		sip_port_min
		sip_port_max
		sip_transaction_timeout
		video_max_bitrate
		video_qos
		vq_report_collector
		vq_report_publish
		vq_alarm_threshold
		webrtc_audio_layer
		answer_sdp_priority

Domain	Section	Setting
		sip_port_binding
		defer_device_release
	session	
		agc_mode
		auto_accept_video
		auto_answer
		auto_answer_delay
		callwait_tone_enabled
		callwait_tone_file
		dtmf_feedback
		dtmf_method
		echo_control
		noise_suppression
		dtx_mode
		reject_session_when_headset_na
		sip_code_when_headset_na
		vad_level
		ringback_enabled
		ringback_file
		ringing_enabled
		ringing_timeout
		ringing_file
		ringing_while_call_held
		restart_audio_if_stuck
		reject_session_when_busy

Domain	Section	Setting
		number_sessions_for_busy
		sip_code_when_busy
		rx_agc_mode
	device	
		audio_in_device
		audio_out_device
		capture_device
		headset_name
		include_headset
		use_headset
codecs — See SIP Endpoint SDK for .NET 9.0.0NET—Working with Codec Priorities.		
proxies		
	proxy''	
		display_name
		domain
		password
		reg_interval
		reg_match_received_rport
		reg_timeout
		mailbox (sub-section of proxy)
		server
		timeout
		transport

Domain	Section	Setting
		user
		nat (sub-section of proxy)
		ice_enabled
		stun_server
		stun_server_port
		turn_password
		turn_relay_type
		turn_server
		turn_server_port
		turn_user_name
		system
	diagnostics	
		enable_logging
		log_file
		log_filter
		log_level
		log_options_provider
		log_options_endpoint
		logger_type
		log_segment
		log_expire
		log_time_convert
		log_time_format
	security	
		certificate

Domain	Section	Setting
		tls_enabled
		tls-target-name-check
		use_srtp
	media	
		ringing file

policy Domain

endpoint Section

audio_qos

Valid Values: Integer

Integer value representing the DSCP bits to set for RTP audio packets. **Note:** QoS is not supported for Windows Vista, Windows 7, or higher.

include_os_version_in_user_agent_header

Valid Values: `0`, `1`

Default Value: `1`

If set to `1`, the user agent field includes the OS version the client is currently running on.

include_sdk_version_in_user_agent_header

Valid Values: `0`, `1`

Default Value: `1`

If set to `1`, the user agent field includes the SDK version the client is currently running on.

ip_versions

Valid Values: `IPv4`, `IPv6`, `IPv4`, `IPv6`, `IPv6`, `IPv4`, or empty

Default Value: `IPv4`, `IPv6`

- IPv4—the application selects an available local IPv4 address; IPv6 addresses are ignored.
- IPv6—the application selects an available local IPv6 address; IPv4 addresses are ignored.
- IPv4, IPv6 or an empty—the application selects an IPv4 address if one exists. If not, an available IPv6 address is selected.
- IPv6, IPv4—the application selects an IPv6 address if one exists. If not, an available IPv4 address is selected.

Note: This parameter has no effect if the **public_address** option specifies an explicit IP address.

public_address

Valid Values: See description below

Default Value: Empty string which is fully equivalent to the `$auto` value

Local IP address or Fully Qualified Domain Name (FQDN) of the machine. This setting can be an explicit setting or a special value that the SDK uses to automatically obtain the public address.

Valid Values:

This setting may have one of the following explicit values:

- An IP address. For example, `192.168.16.123` for IPv4 or `FE80::0202:B3FF:FE1E:8329` for IPv6.
- A bare host name or fully qualified domain name (FQDN). For example, `epsipwin2` or `epsipwin2.us.example.com`.

This setting may have one of the following special values:

- $auto—The SDK selects the first valid IP address on the first network adapter that is active (status=up) and has the default gateway configured. IP family preference is specified by the policy.endpoint.ip_versions setting.
- $ipv4 or $ipv6—Same behavior as the $auto setting but the SDK restricts the address to a particular IP family.
- $host—The SDK retrieves the standard host name for the local computer using the gethostname system function.
- $fqdn—The SDK retrieves the fully qualified DNS name of the local computer. The SDK uses the GetComputerNameEx function with parameter ComputerNameDnsFullyQualified.
- An adapter name or part of an adapter name prefixed with `$`. For example, `$Local Area Connection 2` or `$Local`. The specified name must be different from the special values `$auto`, `$ipv4`, `$host`, and `$fqdn`.
  If the value is an explicit host name, FQDN, or $fqdn, the Contact header includes the host name or FQDN for the recipient of SIP messages (SIP Server or SIP proxy) to resolve on their own. For all other cases, including $host, the resolved IP address is used for Contact. The value in SDP is always the IP address.
- $net:subnet - The SDK will select the IP address matching the given network (from any local interface). The *subnet* is the full CIDR name as per RFC 4632. For example, `$net:192.168.0.0/16`.

include_mac_address

Valid Values: `0`, `1`

Default Value: `0`

If set to 1, the MAC address is included in the Contact header of the REGISTER message of the host's network interface in a format compatible with RFC 5626.

refer_to_proxy

Valid Values: `0`, `1`

Default Value: `0`

Specifies the destination of a referred INVITE.

- 0—Send the INVITE to the URL specified in the Refer-To header of the REFER message.
- 1—Send the INVITE to your configured SIP Proxy.

rtp_inactivity_timeout

Valid Values: 5-150

Default Value: 150

Suggested Value: 30

Timeout interval in seconds for RTP inactivity.

rtp_port_min

Valid Values: 9000-65535

The integer value representing the minimum value for an RTP port range. Must be within the valid port range of 9000 to 65535. If the minimum and maximum values are not specified or are set to an invalid value, the default minimum (9000) and maximum (minimum value + 999) are used. Setting the minimum to a value that is larger than the maximum is considered an error and will result in a failure to initialize the endpoint.

rtp_port_max

Valid Values: 9000-65535

The integer value representing the maximum value for an RTP port range. Must be within the valid port range of 9000 to 65535. If the minimum and maximum values are not specified or are set to an invalid value, the default minimum (9000) and maximum (minimum value + 999) are used. Setting the maximum to a value that is less than the minimum is considered an error and will result in a failure to initialize the endpoint.

tcp_port_min

Valid Values: 0-65535

The integer value representing the minimum value for a TCP client-side port range. Must be within the valid port range of 1 to 65535. If set to 0 (default) or if the configured range is not valid, SIP connections over TCP and TLS use ephemeral ports, assigned by the operating system.

tcp_port_max

Valid Values: 0-65535

The integer value representing the maximum value for a TCP client-side port range. Must be within the valid port range of 1 to 65535. If set to 0 (default) or if the configured range is not valid, SIP connections over TCP and TLS use ephemeral ports, assigned by the operating system.

If the value is non-zero and greater than the *tcp_port_min* value, this value specifies the maximum value for a TCP client-side SIP port range that will be used for all outgoing SIP connections over TCP and TLS transport.

signaling_qos

Valid Values: Integer

The integer value representing the DSCP bits to set for SIP packets. **Note:** QoS is not supported for Windows Vista, Windows 7, or higher.

sip_port_min

Valid Values: 1-65535

The integer value representing the minimum value for a SIP port range. Must be within the valid port range of 1 to 65535. If the minimum and maximum values are not specified or are set to an invalid value, the default minimum (5060) and maximum (minimum value + 6) are used. Setting the minimum to a value that is larger than the maximum is considered an error and will result in a failure to initialize the endpoint.

sip_port_max

Valid Values: 1-65535

The integer value representing the maximum value for a SIP port range. Must be within the valid port range of 1 to 65535. If the minimum and maximum values are not specified or are set to an invalid value, the default minimum (5060) and maximum (minimum value + 6) are used. Setting the maximum to a value that is less than the minimum is considered an error and will result in a failure to initialize the endpoint.

sip_transaction_timeout

Valid Values: 1-32000

Default Value: 4000

SIP transaction timeout value in milliseconds. Valid values are 1 through 32000, with a default value of 4000. The recommended value is 4000.

video_max_bitrate

Valid Values: Integer

Integer value representing the maximum video bitrate.

video_qos

Valid Values: Integer

The integer value representing the DSCP bits to set for RTP Video packets. **Note:** QoS is not supported for Windows Vista, Windows 7, or higher.

vq_report_collector

See SIP Endpoint SDK for .NET—Producing RTCP Extended Reports.

vq_report_publish

See SIP Endpoint SDK for .NET—Producing RTCP Extended Reports.

vq_alarm_threshold

Valid Values: `0` or a number from `1.0` to `5.0`

Default Value: `0`

Specifies the MOS threshold for generating Voice Quality Alarms. A `0` value disables the alarms. The recommended threshold value is `3.5`. Genesys recommends that you avoid using values above `4.2` as an MOS that high might not be obtainable with some codecs, even in perfect network conditions.

webrtc_audio_layer

Valid Values: `0`, `1`, `2`, `1000`, `2000`, `3000`

Default Value: `0`

Specifies which audio layer is used for WebRTC.

- `0` — The audio layer is defined by the `GCTI_AUDIO_LAYER` environment variable — Core audio is used if this environment variable is not specified.
- `1` — Wave audio layer is used.
- `2` — Core audio layer is used.
- `1000` — Instructs the audio layer to open the microphone channel when the endpoint starts up, using the audio layer type defined by option `0`, and to keep it open until the endpoint is terminated.
- `2000` — Opens the speaker channel for the life of the endpoint, using the audio layer type defined by option `0`. Eliminates any delay in opening the audio device when an incoming or outgoing call is connected, for example in environments where audio device startup is slow due to a required restart of the Windows MMCSS service.
- `3000` — Opens the microphone and speaker channels for the life of the endpoint, using the audio layer type defined by option `0`.

> **Important**
>
> Keeping the audio channels permanently open eliminates any delay in connecting audio device to the call works around any issues with device occasionally not starting (or stopping) properly, at the cost of very small performance penalty.

answer_sdp_priority

Valid Values: `config`, `offer`

Default Value: `config`

- `config`—the endpoint selects the first codec from the codec configuration listed in both the codec configuration and the SDP offer.
- `offer`—the endpoint selects the first codec in the SDP offer listed in both the codec configuration and the SDP offer.

sip_port_binding

Valid Values: `0`, `1`

Default Value: `0`

- `0`—open the SIP port to listen on any interface.
- `1`—the SIP port binds to the interface specified by the public_address setting and listens only on this IP address.

defer_device_release

Valid Values: Any integer

Default Value: `200`

If set to a non-zero value, releasing of audio devices will be deferred for a given time (in milliseconds) after the audio stream has been stopped, to avoid any potential service interruptions when the audio is going to be quickly restarted, and if audio device operations are too slow on the user workstation or have other problems with restart. A zero value disables the deferred device release.

session Section

agc_mode

Valid Values: `0`, `1`

Default Value: `1`

If set to 0, AGC (Automatic Gain Control) is disabled; if set to 1, it is enabled. Other values are reserved for future extensions. This configuration is applied at startup, after which time the **agc_mode** setting can be changed to 1 or 0 from the main sample application.

Note: It is not possible to apply different AGC settings for different channels in multi-channel scenarios.

auto_accept_video

Valid Values: `0`, `1`

This setting is only used in auto-answer scenarios when `auto_answer=1`.

If auto_accept_video is set to 1, both audio and video streams are accepted, otherwise incoming calls are answered as audio only, even if video is present in the offer.

auto_accept_video applies to a 3pcc answer when make-callrfc3275 is configured to 1 on the originating DN and a video codec is configured in the endpoint. auto_accept_video is not applied to a 3pcc answer when make-call-rfc3275 is configured to 2 on an originating DN, even if auto_accept_video is set to 1 and a video codec is configured in the endpoint.

auto_answer

Valid Values: `0`, `1`

If set to 1, all incoming calls should be answered automatically.

auto_answer_delay

Valid Values: Number in milliseconds

Default Value: `1500`

Time in milliseconds to wait before auto-answering. The recommended and default value is 1500 milliseconds.

callwait_tone_enabled

Valid Values: `0`, `1`

Default Value: `1`

Specifies whether the call waiting tone is enabled (`1`) or disabled (`0`). This configuration is applied at startup.

callwait_tone_file

Valid Values: Empty, or the path to the call waiting sound file. The path may be a file name in the current directory or the full path to the sound file.

Default Value: `callwait.wav`

Specifies the audio file that is played when the call waiting tone is enabled by the callwait_tone_enabled option.

Note: WebRTC does not support MP3 playback. The callwait file for built-in ringing should be a RIFF (little-endian) WAVE file using one of the following formats:

- kWavFormatPcm = 1, PCM, each sample of size bytes_per_sample
- kWavFormatALaw = 6, 8-bit ITU-T G.711 A-law
- kWavFormatMuLaw = 7, 8-bit ITU-T G.711 mu-law

Uncompressed PCM audio must 16-bit mono or stereo, and have a frequency of 8, 16, or 32 kHz.

dtmf_feedback

Valid Values: `0`, `1` (default). If set to `1`, DTMF feedback (audio tones played locally when sending DTMF signals to the remote side of the conversation) is enabled.

dtmf_method

Valid Values: `Rfc2833`, `Info`, `InbandRtp`

Method to send DTMF.

echo_control

Valid Values: `0`, `1`

If set to `1`, echo control is enabled.

noise_suppression

Valid Values: `0`, `1`

If set to `1`, noise suppresion is enabled.

dtx_mode

Valid Values: `0`, `1`

If set to `1`, DTX is activated.

reject_session_when_headset_na

Valid Values: `0`, `1`

If set to `1`, the SDK should reject the incoming session if a USB headset is not available.

sip_code_when_headset_na

Valid Values: SIP Error Code

Default Value: `480`

If a valid SIP error code is supplied, the SDK rejects the incoming session with the specified SIP error code if a USB headset is not available.

vad_level

Valid Values: `0`-`3`

Sets the degree of bandwidth reduction, from `0` for conventional VAD to `3` for aggressive high.

ringback_enabled

Valid Values: `0`, 1, 2, 3, 4, 6

Default Value: 2

Specifies whether the ringback tone is enabled for outgoing calls.

- `0` — The ringback is not played when the INVITE dialog is not yet established. In scenarios where ringback is provided by Media Server, the ringback tone would be still present.
- 1 — The incoming media stream is played if provided by the Media gateway in a reliable provisional response with SDP.
- 2 — A local file is used for the ringback.
- 3 — The ringback is always played using either a local file or media provided by the gateway, if the provisional response is reliable.
- 4 — Same as 1, but the incoming media stream is played even if the provisional response from Media gateway is not reliable.
- 6 — The ringback is always played using either a local file or media provided by the gateway (regardless of whether the provisional response is reliable or not).

ringback_file

Valid Values: Empty or the path to the ringback sound file. The path can be a file in the current directory or the full path to the sound file.

Default Value: Empty

Specifies the audio file that is played when the ringing tone is enabled with the `ringing_enabled` option.

WebRTC does not support MP3 playback. The ringtone file for built-in ringback must be a RIFF (little-endian) WAVE file using one of the following formats:

- kWavFormatPcm = 1, PCM, each sample of size bytes_per_sample
- kWavFormatALaw = 6, 8-bit ITU-T G.711 A-law (8 KHz sampling rate)
- kWavFormatMuLaw = 7, 8-bit ITU-T G.711 mu-law (8 KHz sampling rate)

Uncompressed PCM audio must be 16-bit mono or stereo with a sampling rate of 8, 16, or 32 KHz.

ringing_enabled

Valid Values:

- `0`: None, disable ringtone.
- 1: (default) Play ringtone through system default device only. Configure media in system.media.ringing_file.
- 2: Play ringtone through communication device (headset) only. Configure media in policy.session.ringing_file.

- 3: Play ringtone through both devices at the same time (the combination of values 1 and 2).
- 4: Play ringtone through a separate ringer device, specified by policy.device.ringer_device.
- 5: Play ringtone through system default device and lay ringtone through a separate ringer device (the combination of values 1 and 4).
- 6: Play ringtone through the communication device (headset) once only for the full duration (policy.session.ringing_timeout is ignored, and ringing does not stop when the call is answered). Configure media in policy.session.ringing_file.
- 7: Play ringtone once for the full duration through both system default device and communication device (headset) (policy.session.ringing_timeout is ignored, and ringing does not stop when call is answered). Configure media in system.media.ringing_file and policy.session.ringing_file.

Default Value: 1

Specifies whether to enable the ringtone and on which device to play the media file. This option applies to both calls that are auto-answered by Softphone or by other applications such as Workspace Web Edition and by calls that are manually answered by an agent.

Suppressing the Ringtone

The ringtone is generated for all incoming calls to the Genesys SIP Endpoint SDK. To suppress the ringtone for third-party call control for the originating DN, configure the following SIP Server option:

- `make-call-alert-info=@genesys>`

or

- `make-call-alert-info=;service=3pcc`

> **Important**
>
> If at least one application based on SIP Endpoint in the contact center is configured with the `ringing_enabled` option set to a non-zero value, the SIP Server `make-call-alert-info` option should be set to one of the specified values.

ringing_timeout

Valid Values: Empty, `0`, or a positive number

Default Value: `0`

Specifies the duration, in seconds, of the ringing tone. If set to `0` or if the value is empty, the ringing time is unlimited.

ringing_file

Valid Values: Empty or the path to the ringing sound file. The path may be a file name in the current directory or the full path to the sound file.

Default Value: ringing.wav

Specifies the audio file that is played when the ringing tone is enabled with the `ringing_enabled` option.

Note that WebRTC does not support MP3 playback. The ringtone file for built-in ringing should be a RIFF (little-endian) WAVE file using one of the following formats:

- kWavFormatPcm = 1, PCM, each sample of size bytes_per_sample
- kWavFormatALaw = 6, 8-bit ITU-T G.711 A-law
- kWavFormatMuLaw = 7, 8-bit ITU-T G.711 mu-law

Uncompressed PCM audio must 16 bit mono or stereo and have a frequency of 8, 16, or 32 KHZ.

ringing_while_call_held

Valid Values: `0`, `1`

Default Value: `1`

Specifies whether to play ringtone or not when a call is held and a new call arrives.

- `0` - call wait tone (if configured) is played instead of ringtone.
- `1` - ringtone is played whenever a new call arrives and there are no other active calls; held calls are not considered active in this case.

restart_audio_if_stuck

Valid Values: Empty, `0`, `1`

Default Value: `0`

- `0` or Empty—disable auto restart for stuck audio
- `1`—enable auto restart for stuck audio

reject_session_when_busy

Valid Values: Empty, `0`, `1`

Default Value: `0`

- `0` or Empty—disable rejection of a session when busy
- `1`—enable rejection of a session when busy

number_sessions_for_busy

Valid Values: Positive integer

Default Value: 1

Sets the number of sessions before busy. Must be a positive integer.

sip_code_when_busy

Valid Values: Empty, 4xx, 5xx, 6xx

Default value: Empty

SIP error response code to use when busy. Can be set to any valid SIP error response code in the 4xx, 5xx, or 6xx range, for example, 486.

rx_agc_mode

Valid Values: `0`, `1`

Default value: `0`

When set to `1`, the SDK enables the receiving-side AGC allowing the volume of the received RTP stream to be adjusted automatically. When set to `0` (default), the feature is disabled.

device Section

audio_in_device

Valid Values: A regex that matches the ECMAScript standard.

Microphone device name.

audio_out_device

Valid Values: A regex that matches the ECMAScript standard.

Speaker device name.

capture_device

Valid Values: A regex that matches the ECMAScript standard.

Capture device name.

headset_name

Valid Values: A regex that matches the ECMAScript standard.

The name of the headset model.

include_headset

Valid Values: A pair of device names or name parts, with microphone and speaker names separated

by a colon, or a comma-separated list of name pairs. For example: `External Mic:Headphones`

If the names include delimiter characters such as quotes, colons, or comma, they must be enclosed in single or double quotes.

Specifies the list of audio in / out devices to be considered as a headset for automatic device selection. This option is applicable to the case when **use_headset** = "1"

ringer_device

Valid Values: A valid ringer device name: can be either the device proper name or a regular expression. This option is applicable when ringing_enabled = 4

use_headset

Valid Values: `0`, `1`

If set to `0`, the audio devices specified in audio_in_device and audio_out_device are used by the SDK.

If set to `1`, the SDK uses a headset as the preferred audio input and output device and the audio devices specified in audio_in_device and audio_out_device are ignored.

exclude_headset

Valid Values: Any valid regular expression.

Default Value: Empty

The name of a headset model or built-in audio device (example: `Realtek`). The specified device is excluded from being recognized or automatically selected as a valid headset. Note that components of an excluded device, such as a microphone or speaker, can still be selected manually (or automatically, if Softphone does not find a better device). However, even if a component of an excluded device is selected, Softphone does not recognize the excluded device as an available headset.

> **Important**
> This option is only available on Windows installations.

codecs Domain

See SIP Endpoint SDK for .NET 9.0.0NET—Working with Codec Priorities.

proxies Domain

Configure a proxy section for each connectivity line. For example, for three connectivity lines, configure sections for `proxy0`, `proxy1`, and `proxy2`.

When the proxy section does not exist in the configuration file for a particular connectivity line, the framework takes the configurations settings from the `proxy0` section. You can use this feature in use cases where the proxy sections are the same for all connectivity lines.

proxy Section

display_name

Valid Values: String

Proxy display name.

domain

Valid Values:Any valid SIP domain

Defalut Value: Empty

A SIP domain is an application layer configuration defining the management domain of a SIP proxy. The configured value should include hostport and may include uri-parameters as defined by RFC 3261. The scheme, userinfo, and transport URI parameters are included automatically.

If set to an empty string, SIP Endpoint SDK for .NET uses the parameters from the Connectivity section to construct the SIP domain value as it did in previous versions.

password

Valid Values: String

Proxy password.

reg_interval

Valid Values: Integer

Default Value: `0`

The period, in seconds, after which the endpoint starts a new registration cycle when a SIP proxy is down. Valid values are integers greater than or equal to `0`. If the setting is empty or negative, the default value is 0, which means no new registration cycle is allowed. If the setting is greater than 0, a new registration cycle is allowed and will start after the period specified.

reg_match_received_rport

Valid Values: `0` or `1`

Default Value: `0`

This setting controls whether or not SIP Endpoint SDK should re-register itself when receiving an IP

address (in the `received` parameter of a REGISTER response) that is different from the address supplied in the Contact header and does not match any local network interfaces. A value of `0` (default) disables this feature and a value of 1 enables re-registration.

Starting from 9.0.003, this setting is deprecated and is not recommended for use, unless suggested by Genesys Technical Support to fix specific problems. When the `received` parameter of a REGISTER response matches a local IP address, changing the IP address and re-registering is now done automatically.

reg_timeout

Valid Values: Number in seconds

The period, in seconds, after which registration should expire. A new REGISTER request will be sent before expiration. Valid values are integers greater than or equal to 0. If the setting is `0` or empty/null, then registration is disabled, putting the endpoint in standalone mode.

mailbox Sub-section

> **Important**
> **mailbox** is a sub-section of the **proxy** section.

password

Valid Values: String

Mailbox password.

server

Valid Values: String

Proxy server address and port for this mailbox.

timeout

Valid Values: Number in seconds

Default Value: `1800`

Subscription expiration timeout in seconds. If the setting is missing or set to `0`, the SDK uses a default timeout of `1800` seconds (30 minutes).

transport

Valid Values: `udp`, `tcp`, `tls`

Transport protocol to use when communicating with the server.

user

Valid Values: String

Mailbox ID for this mailbox.

nat Sub-section

> **Important**
> **nat** is a sub-section of the **proxy** section.

ice_enabled

Valid Values: Boolean

Enable or disable ICE.

stun_server

Valid Values: String

STUN server address. An empty or null value indicates this feature is not used.

stun_server_port

Valid Values: Valid port number

Default Value: 3478

STUN server port value.

turn_password

Valid Values: String

Password for TURN authentication.

> **Warning**
> Starting from 9.0.012.02, this setting is deprecated and is not recommended for use, unless suggested by Genesys Technical Support to fix specific problems. Use the GCTI_TURN_PASSWORD environment variable to set the password for TURN authentication.

turn_relay_type

Valid Values: `0`, `udp`, `1`, or `tcp`

Type of TURN relay.

- `0` or `udp` for TURN over UDP.
- `1` or `tcp` for TURN over TCP.

turn_server

Valid Values: String

TURN server address. An empty or null value indicates this feature is not used.

turn_server_port

Valid Values:Valid port number

Default Value: 3478

TURN server port value.

turn_user_name

Valid Values: String

User ID for TURN authorization

> **Warning**
> Starting from 9.0.012.02, this setting is deprecated and is not recommended for use, unless suggested by Genesys Technical Support to fix specific problems. Use the GCTI_TURN_USERNAME environment variable to set the username for TURN authentication.

system Domain

diagnostics Section

enable_logging

Valid Values: `0` or `1`

Default Value: 1

Disable or enable logging.

log_file

Valid Values: String

Log file name, for example, `SipEndpoint.log`.

log_filter

Valid Values: *Empty*, `dtmf`

Default Value: *Empty*

Specifies the list of log filters to be applied to hide sensitive data from the endpoint log. Currently the only supported filter is `dtmf`, which hides all occurrences of DTMF data from the log (by replacing entered digits with 'x').

log_level

Valid Values: `0`-`4`

Default Value: 3

Log levels: `0` = "Fatal"; 1 = "Error"; 2 = "Warning"; 3 = "Info"; 4 = "Debug"

log_options_provider

Valid Values: Valid values for webrtc, `warning`, `state`, `api`, `debug`, `info`, `error`, `critical`

Example value: `gsip=2, webrtc=(error,critical)`

log_options_endpoint

Valid Values: `0`-`4`, same as **log_level**

Default Value: 2

Log levels: `0` = "Fatal"; 1 = "Error"; 2 = "Warning"; 3 = "Info"; 4 = "Debug"

5 = Logging disabled.

This setting should not be set higher than log_level setting.

logger_type

Valid Values: file

If set to `file` the log data will be printed to the file specified by the **log_file** value.

log_segment

Valid Values: `false`, number, or number in KB,MB, or hr

Default Value: `10 MB`

- false: No segmentation is allowed
- or KB: Size in kilobytes
- MB: Size in megabytes
- hr: Number of hours for segment to stay open

Specifies the segmentation limit for a log file. If the current log segment exceeds the size set by this option, the file is closed and a new one is created. This option is ignored if log output is not configured to be sent to a logfile.

log_expire

Valid Values: `false`, number, number `file`, number day

Deafult Value: `10` (store 10 log fragments and purge the rest)

- false: No expiration; all generated segments are stored.
- or file: Sets the maximum number of log files to store. Specify a number from 1—1000.
- day: Sets the maximum number of days before log files are deleted. Specify a number from 1—100.

Determines whether log files expire. If they do, sets the measurement for determining when they expire, along with the maximum number of files (segments) or days before the files are removed. This option is ignored if log output is not configured to be sent to a log file.

log_time_convert

Valid Values: `local`, `utc`

Default Value: `local`

- `local`: The time of log record generation is expressed as a local time, based on the time zone and any seasonal adjustments. Time zone information of the application's host computer is used.
- `utc`: The time of log record generation is expressed as Coordinated Universal Time (UTC).

Specifies the system in which an application calculates the log record time when generating a log file. The time is converted from the time in seconds since the Epoch (00:00:00 UTC, January 1, 1970).

log_time_format

Valid Values: `time`, `locale`, `ISO8601`

Default Value: time

- `time`: The time string is formatted according to the HH:MM:SS.sss (hours, minutes, seconds, and milliseconds) format.
- `locale`: The time string is formatted according to the system's locale.
- `ISO8601`: The date in the time string is formatted according to the ISO 8601 format. Fractional seconds are given in milliseconds.

Specifies how to represent, in a log file, the time when an application generates log records. A log record's time field in the ISO 8601 format looks like this: 2001-07-24T04:58:10.123.

security Section

> **Important**
>
> SIP Endpoint SDK no longer uses the **tls_enabled** setting.

certificate

Valid Values: String

Thumbprint value of the Public endpoint certificate file which is used as a client-side certificate for outgoing TLS connections and server-side certificate for incoming TLS connections. For example, `78 44 34 36 7a c2 22 48 bd 5c 76 6b 00 84 5d 66 83 f5 85 d5`

This option replaces the **cert_file** option from previous versions. For backwards compatibility, the SDK accepts **certificate** or **cert_file**.

tls-target-name-check

Valid Values: `no`, `host`

Default Value: `host`

Specifies if the Common Name in the subject field and/or the Subject Alternate Names of the server's certificate will be compared to the target host name (option value `host`). If they are not identical, the connection fails. If the option is set to `no`, a comparison is not made, and the connection is allowed.

use_srtp

Valid Values: `optional`, `allowed`, `disabled`, `off`, `elective`, `both`, `enabled`, `force`, `mandatory`

Indicates whether to use SRTP:

- `optional` or `allowed`—do not send secure offers, but accept them
- `disabled` or `off`—do not send secure offers and reject incoming secure offers
- `elective` or `both`—send both secure and non-secure offers and accept either
- `enabled`—send secure offers, accept both secure and non-secure offers
- `force` or `mandatory`—send secure offers, reject incoming non-secure offers

Adding either `',UNENCRYPTED_SRTCP'` (long form) or `',UEC'` (short form) to any value (for example, "*enabled,UEC*""), would result in the **UNENCRYPTED_SRTCP** parameter being added to that offer. When this parameter is negotiated, RTCP packets are not encrypted, but are still authenticated.

media Section

ringing_file

Valid Values: Empty, String file name

Default Value: `ringing.mp3`

The Ringing sound file name in the current directory or the full local path to the ringing sound file.

Audio device settings

Contents

- Administrator

How to set up your audio devices, such as headsets, to work with Genesys Softphone.

Related documentation:

-

Genesys Softphone uses the following criteria to select its audio input and output devices:

- Basic Settings—the basic settings for audio input and output devices
- Selection Rules—the rules used to select an audio device, auto-answer a call, and reject a call
- Combinations of settings—different combinations of settings affect audio device selection, auto-answer, and call rejection

Basic settings

Headsets and other audio input devices are configured by using the following parameters:

- `headset_name`
- `audio_in_device`
- `audio_out_device`

If none of the audio devices that are accessible to the endpoint match the device names in the configuration file, the Genesys Softphone will pick up the first available devices from the WebRTC lists for audio devices.

> **Tip**
>
> The `headset_name`, `audio_in_device`, and `audio_out_device` options support both device proper names and regular expressions.

Selection rules

The following rules are used to select an audio device, auto-answer a call, and reject a call.

Audio device selection

The procedure for audio device selection is applied on startup and every time any changes are made to device presence (such as when a new device is plugged in or an existing device is removed):

1. The first device in the applicable list that is present in the system is selected when possible. This device (or devices) will either be specified by `headset_name` or by `audio_in_device` and `audio_out_device`, depending on whether `use_headset` has been enabled.
2. If none of the configured devices are present (or if the configuration list is empty), then the Genesys Softphone will select the audio devices using the priority that has been provided by WebRTC, based on the order of the available devices in its device list.

Auto-answer

In cases where either of the following conditions is met, the auto-answer functionality is blocked (a policy of `should answer` returns `unknown`, although a manual answer is still possible):

- `use_headset` is set to 1, and none of the devices listed in the `headset_name` settings is currently present (but session rejection is not applicable, that is, `reject_session_when_headset_na` has been set to 0)
- The Genesys Softphone was unable to find any usable microphone or speaker device (applicable to cases when `use_headset` is set to 0)

Finally, when `auto_answer` is set to 1 and the auto-answer functionality is not blocked (and the call was not already rejected), the Genesys Softphone answers the incoming call automatically (the `should answer` policy returns `true`).

Rejecting a call

For backward compatibility with previous releases, a call can only be rejected when both of the following conditions are met (a policy of `should answer` returns `false`):

- Both `use_headset` and `reject_session_when_headset_na` are set to 1
- None of the devices listed in the `headset_name` settings is currently present

When these conditions are met using Genesys Softphone with SIP, an incoming call is rejected with a SIP response code as configured in the `sip_code_when_headset_na` setting. If the setting is missing or the value does not belong to the valid range of 400 to 699, then the default of 480 (`Temporarily Unavailable`) is used.

In addition, when these conditions are met, the Genesys Softphone will refuse to initiate any new calls, that is, it will reject outgoing calls.

Note that the availability of a fallback device (selected by Step 2 in the Audio Device Selection section) does not affect call rejection.

Combinations of settings

The following combinations of settings affect audio device selection, auto-answer, and call rejection in the ways described below.

use_headset=1

Headset is Available The Genesys Softphone considers a headset to be available if a headset was found by name in the list of `headset_name` entries. (The highest priority device in the list is selected). Outgoing calls can be initiated.	`auto_answer=1`	Incoming calls are answered automatically.
	`auto_answer=0`	Incoming calls are answered manually.
Headset is Not Available The Genesys Softphone decides that no headset is available if a headset was not found by name in the list of `headset_name` entries. An audio device is still assigned, if possible (that is, if any supported devices are present in the system), using the first available audio input and output devices from the list compiled by WebRTC.	No auto-answer is possible in this sub-case, so the `auto_answer` setting is not used	`reject_session_when_headset_na=1` • Incoming calls are automatically rejected • Outgoing calls are blocked
		`reject_session_when_headset_na=0` • Incoming calls can be answered manually—it is assumed that the agent will plug the headset in (or use an available non-headset device, if applicable) before answering the call • Outgoing calls can be initiated—it is the agent's responsibility to ensure that the appropriate audio devices are available before the call is answered by the remote side

use_headset=0

Audio devices are configured using the names from the `audio_in_device` and `audio_out_device` settings. The Genesys Softphone selects the highest-priority input and output devices from that list or, if no valid devices are found in that list, from the first available devices in the list compiled by WebRTC. Outgoing calls can be initiated.

Both microphone and speaker are available	`auto_answer=1`	Incoming calls are answered automatically.
	`auto_answer=0`	Incoming calls are answered manually.
Either microphone or speaker is not available	No auto-answer is possible in this sub-case, so the `auto_answer` setting is not used	Auto-rejection is not applicable, so the `reject_session_when_headset_na` setting is not used

• Incoming calls can be answered manually—it is assumed that the agent will plug in the headset (or use an available non-headset device, if applicable) before answering the call • Outgoing calls can be initiated—it is the agent's responsibility to ensure that the appropriate audio devices are available before the call is answered by the remote side		