



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

SIP Endpoint SDK Developer's Guide

Configuring SIP Endpoint SDK for OS X

Contents

- 1 Configuring SIP Endpoint SDK for OS X
 - 1.1 Using the Default Configuration File
 - 1.2 policy Domain
 - 1.3 proxies Domain
 - 1.4 system Domain
 - 1.5 Producing RTCP Extended Reports
 - 1.6 Working with Codec Priorities
 - 1.7 Audio Device Settings

Configuring SIP Endpoint SDK for OS X

Using the Default Configuration File

You can find the default configuration file in the following location:

```
<installation folder>/Bin/SipEndpoint.config
```

This file contains XML configuration details that affect how your SIP Endpoint SDK application behaves. The initial settings are the same as those specified for use with the QuickStart application included with your SIP Endpoint SDK release.

Configuration settings are separated into two containers: the **Basic Container** holds the connectivity details that are required to connect to your SIP Server, while the **Genesys Container** holds a variety of configuration settings.

Basic Container

The first Container ("Basic") holds the basic connectivity details that are required to connect to your SIP Server. This container has at least one connection (Connectivity) element with the following attributes:

```
<Connectivity user="DN" server="SERVER:PORT" protocol="TRANSPORT"/>
```

Make the following changes and save the updated configuration file before using the SIP Endpoint SDK:

- **user="DN"** — Supply a valid DN for the user attribute.
- **server="SERVER:PORT"** — Replace SERVER with the host name where your SIP Server is deployed, and PORT with the SIP port of the SIP Server 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.
- **protocol="TRANSPORT"** — Set the protocol attribute to reflect the protocol being used to communicate with SIP Server. Possible values are UDP, TCP, or TLS.

SRV Resolution

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

- SIP Endpoint SDK selects the SRV target based on the **priority** field only and does not consider the **weight** field of an SRV record.
- You can not 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 can not

use SRV records for the mailbox section or the **vq_report_collector** setting.

Genesys Container

The second Container ("Genesys") holds a number of configurable settings that are organized into domains and sections. These settings do not have to be changed, but can be customized to take full control over your SIP Endpoint SDK applications.

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
		rtp_inactivity_timeout
		rtp_port_min
		rtp_port_max
		signaling_qos
		sip_port_min
		sip_port_max
		sip_transaction_timeout
		vq_report_collector
		vq_report_publish
		vq_alarm_threshold
		answer_sdp_priority
		sip_port_binding
		video_max_bitrate
	video_qos	
	defer_device_release	
		session
auto_accept_video		
auto_answer		
auto_answer_delay		
dtmf_method		

Domain	Section	Setting
		dtmf_feedback
		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
		reject_session_when_busy
		number_sessions_for_busy
		sip_code_when_busy
rx_agc_mode		
	device	
		audio_in_device
		For more information, see Audio Device Settings
		audio_out_device
		capture_device
		ringer_device
		headset_name
		exclude_headset
		use_headset
		include_headset
codecs —See Working with Codec Priorities		
proxies		
	proxy	
		display_name
		domain
		password
		reg_interval
		reg_match_received_rport
		reg_timeout

Domain	Section	Setting
		mailbox
		server
		timeout
		transport
		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
		logger_type
	security	certificate
		certificate-key
		trusted-ca
		sec-protocol
		cipher-list
		ciphersuites
		tls-crl
		tls-target-name-check
		use_srtp

policy Domain

endpoint Section

audio_qos

Valid Values: Number

The integer value representing the DSCP bits to set for RTP audio packets. The value should be 4 * Preferred DSCP value.

include_os_version_in_user_agent_header

Valid Values: 0 or 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 or 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, empty Default Value: IPv4.

A value of IPv4 or an empty value means that the application selects an available local IPv4 address.

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

public_address

Valid Values: See description below.

Local IP address 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.
- A bare host name, for example, epsipmac2.

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

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—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.
- An adapter name or part of an adapter name prefixed with \$. For example, en0 or en1. The specified name must be different from the special values \$auto, \$ipv4, and \$host.
- \$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.

If the value is specified as an explicit host name, the Contact header includes the host name 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.

rtp_inactivity_timeout

Valid Values: Integer between 5 and 150

Default Value: 150

Suggested Value: 30

Timeout interval in seconds for RTP inactivity.

rtp_port_min

Valid Values: Number

Default Value: 8000

The integer value representing the minimum value for an RTP port range. Must be within the valid port range of 8000 to 65535. If the minimum value is not specified or set to an invalid value, the default value of 8000 is used for rtp_port_min. 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: Number
Default Value: 9000

The integer value representing the maximum value for an RTP port range. Must be within the valid port range of 8000 to 65535. If the maximum value is not specified or set to an invalid value, the default maximum 9000 is used for `rtp_port_max`. Setting the maximum to a value that is less than the minimum is considered an error and results in a failure to initialize the endpoint.

signaling_qos

Valid Values: Number

The integer value representing the DSCP bits to set for SIP packets. Integer value should be configured to 4 * Preferred DSCP value.

sip_port_min

Valid Values: Number between 1 and 65535
Default Value: 5060

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 value is not specified or set to an invalid value, the default value of 5060 is used for `sip_port_min`. 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: Number between 1 and 65535
Default Value: 5080

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 maximum value is not specified or set to an invalid value, the default value of 5080 is used for `sip_port_max`. 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: Number between 1 and 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.

vq_report_collector

See [Producing RTCP Extended Reports](#)

vq_report_publish

See [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.

answer_sdp_priority

Valid Values: config or 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 or 1

Default Value: 0

- 0—opens 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.

Important

The `sip_port_binding` must be set to 0 for automatic IP address change detection to work properly. When

set to 1 for specific deployments, agents have to re-register connections manually after an IP address change.

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.

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 or 1

If set to 0, AGC (Automatic Gain Control) is disabled; if set to 1, it is enabled. Default: 1. 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 or 1

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

auto_answer_delay

Valid Values: Number in milliseconds

Default Value: 0

Time in milliseconds to wait before auto-answering (only applicable when auto_answer=1 and auto-answer is not blocked by missing headset).

callwait_tone_enabled

Valid Values: 0, 1

Default Value: 0

Specifies whether the call waiting tone is enabled for incoming calls, to be played when a new call arrives while user is on active call.

callwait_tone_file

Valid Values: Empty or the path to the call waiting tone 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 call waiting tone is enabled with the

callwait_tone_enabled option.

The call waiting tone file 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.

dtmf_method

Valid Values: Rfc2833, Info, or InbandRtp

Method to send DTMF

dtmf_feedback

Valid Values: 0 (disable feedback), 1 (enable feedback)

Default Value: 1

Controls whether local feedback tone is played when Endpoint sends DTMF to remote party.

echo_control

Valid Values: 0 or 1

If set to 1, echo control is enabled.

noise_suppression

Valid Values: 0 or 1

If set to 1, noise suppression is enabled.

dtx_mode

Valid Values: 0 or 1

If set to 1, DTX is activated.

reject_session_when_headset_na

Valid Values: 0 or 1

If the SDK is configured to use the headset setting (`policy.device.use_headset=1`) and the

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

`ringing_while_call_held`

Valid Values: 0 or 1

Default Value: 1

If set to 0, when another call is held, playing ringtone is suppressed and call wait tone is played instead (if configured). If set to 1 (default value), ringtone is played whenever a new call arrives and there are no other active calls; held calls are not considered active in this case.

`sip_code_when_headset_na`

Valid Values: Number

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, from 0 (conventional VAD) to 3 (aggressive high)

Sets the degree of bandwidth reduction.

`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 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 (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: An integer between 0 to 7.

Default Value: 3

- 0—event Ringing disabled.
- 1—event Ringing enabled.
- 2—play ringtone internally (event Ringing disabled).
- 3—play ringtone internally and enable event Ringing.
- 4—play ringtone through a separate ringer device.
- 5—play ringtone through ringer device and enable event Ringing.
- 6—play ringtone internally once for full duration (**ringing_timeout** is not used and the ringing does not stop when call is answered).
- 7—play ringtone once for full duration and enable event Ringing.

Important

If the ringtone is configured to play for full duration (values 6 or 7), then it will be played even in case of auto-answer with zero delay (configured either in SIP Endpoint SDK or desktop application), unless it is suppressed. This configuration setting assumes **ringing_file** option being configured with a short "beep" sound that would not interfere much with agent hearing the customer (as the ringtone will be played in parallel / mixed with remote audio stream).

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=<urn:alert:service:3pcc@genesys>`

or

- `make-call-alert-info=<file://null>;service=3pcc`

Important

If at least one application based on SIP Endpoint SDK 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 string or string to 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 be 16 bit mono or stereo and have a frequency of 8, 16, or 32 KHZ.

`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

use headset

`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. For more information, see [Audio Device Settings](#)

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.

ringer_device

Valid Values: A regex that matches the [ECMAScript](#) standard.

Name of the device dedicated to playing ringing tone (used when `ringing_enabled` set to 4 or 5)

headset_name

Valid Values: A regex that matches the [ECMAScript](#) standard.

The name of the headset model.

exclude_headset

Valid Values: A regex that matches the [ECMAScript](#) standard.

Default Value: Empty

When set, this will exclude devices whose name matches the regex pattern from being considered a valid headset for automatic device selection. Microphone and speaker parts of the excluded headset can still be selected manually (or even automatically, if no better device is found), but such a selection will not set "headset available" flag.

use_headset

Valid values: 0 or 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.

`include_headset`

Valid Values: Pair of device names or name parts with microphone and speaker names separated by colon, or comma-separated list of such pairs, for example: External Mic : Headphones

Default Value: Empty

When set, specifies the list of audio in/out devices to be considered as headset for automatic device selection, applicable to case when `use_headset = 1`. The names including delimiter character (quotes, colon or comma) must be enclosed in single or double quotes.

codecs Section

See [Working with Codec Priorities](#).

proxies Domain

proxy <n> Section

`display_name`

Valid Values: String

Proxy display name

`domain`

Valid Values: String containing any valid SIP domain

Default Value: Empty string

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 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. Password configured for DN object.

reg_interval

Valid Values: Integers greater than or equal to 0

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 by `regInterval`.

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 a mismatched IP address in the `received` parameter of a REGISTER response. This helps resolve the case where SIP Endpoint SDK for OS X has multiple network interfaces and obtains the wrong local IP address. A value of 0 (default) disables this feature and a value of 1 enables re-registration.

reg_timeout

Valid Values: Number

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 empty/null, then registration is disabled, putting the endpoint in standalone mode.

mailbox Section

Important

The **mailbox** section is a sub-section of the **proxy<n>** section.

password

Valid Values: String

Mailbox password

server

Valid Values: String

Proxy server address and port for this mailbox

timeout

Valid Values: Any positive integer

Default Value: 1800 seconds (30 minutes)

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, or tls

Transport protocol to use when communicating with server

user

Valid Values: String

Mailbox ID for this mailbox

nat Sub-section

Important

nat is a sub-section of the **proxy<n>** 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

Valid values of 0 or 1 disable or enable logging.

log_file

Valid values: String containing full path or relative path to the log file.

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: Number, 0-4

Valid values correspond to log levels:

- 0 = "Fatal"
- 1 = "Error"
- 2 = "Warning"
- 3 = "Info"
- 4 = "Debug".

log_options_provider

Valid Values: Comma-separated list of log setting for various low-level components, including:

- `gsip=N` - Genesys SIP library log level, default is 2
- `webrtc=(level)` - log level(s) for third-party WebRTC native code component. The default is error to include error messages and important WebRTC diagnostic, value `api` adds low-level API printouts.

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

Warning

This settings control low-level debug information used for troubleshooting. Please don't change the value unless instructed by Genesys Technical Support engineer.

`logger_type`

Valid Values: `external`

If set to `external`, an external logger is used.

`security Section`

Important

SIP Endpoint SDK no longer uses the **`tls_enabled`** setting.

`certificate`

See [Configuring certificate option for TLS](#).

`certificate-key`

See [Configuring certificate-key option for TLS](#).

`trusted-ca`

See [Configuring trusted-ca option for TLS](#).

`sec-protocol`

See [Configuring sec-protocol option for TLS](#).

cipher-list

See [Configuring cipher-list option for TLS](#).

ciphersuites

See [Configuring ciphersuites option for TLS](#).

tls-crl

See [Configuring tls-crl option for TLS](#).

tls-target-name-check

See [Configuring tls-target-name-check option for TLS](#).

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.

Producing RTCP Extended Reports

You can use SIP Endpoint SDK to produce RTCP Extended Reports (RFC 3611) and publish them according to RFC 6035 at the end of each call, using a collector address of your choice.

Important

The publish message is sent to the specified collector address only if the `vq_report_collector` parameter is configured with the

user@server:port;transport=udp format. For example, collector@127.0.0.1:5160;transport=udp.

Settings:

```
<domain name="policy">
<section name="endpoint">
  ...
<!--
Valid values for Voice Quality (VQ) report publish setting (vq_report_publish):
0--VQ report is not published
1--VQ report is published to the collector at the end of the call--
  see the vq_report_collector setting information below
-->
<setting name="vq_report_publish" value="0"/>
<!--
Valid values for Voice Quality (VQ) report collector setting (vq_report_collector):
NULL or Empty--The VQ report is published to the proxy described in the
Connectivity section
FQDN or IP address along with port and transport--
  collector@SipServer.genesyslab.com:5060;transport=udp
-->
<setting name="vq_report_collector"
value="collector@SipServer.genesyslab.com:5060;transport=udp"/>
</section>
```

Endpoint:

The `vq_report_publish` and `vq_report_collector` settings can be read from the Endpoint Policy by using the following methods:

```
GetEndpointPolicy(EndpointPolicyQuery.VqReportCollector);
GetEndpointPolicy(EndpointPolicyQuery.VqReportPublish);
```

Working with Codec Priorities

Codecs are listed by name in the codecs domain of the configuration file. The **enabled** section explicitly defines the enabled codecs and their priorities. If the **enabled** section is not included or if the audio and video settings have empty values, then the priority is defined as the same order as listed in the codec setting. To disable a codec, remove the codec name from the list.

Important

Media connections between two SIP endpoints (or between SIP Endpoint and Media Server/Gateway) can be successful only when they have a codec that they both support. In tightly controlled environments, it may be possible to limit the number of enabled codecs to only one preferred codec, but it is highly recommended to include the fallback codec(s) like universally supported PCMU and PCMA to the end of the priority list.

The audio setting can include a list of comma-separated audio codec names in the order of their priorities. The video setting can include a list of comma-separated video codec names in the order of their priorities.

Codec parameters can be specified inline as a comma-separated list enclosed in parenthesis after an equal (=) sign.

```
<setting name="audio" value="g729=(fmtpt='annexb=no'),opus=(pt=125),pcmu,pcma"/>
```

The `maxpkt` optional video parameter specifies the maximum RTP packet size for video codecs (including the RTP header but not counting the UDP and IP overheads). The valid values for this parameter are integers from 1024 to 1472, and the default value is 1442. (Any number outside the valid range would be adjusted to the closest allowable value.)

```
<setting name="video" value="vp8,maxpkt=1280"/>
```

The setting `payload_type` is an integer. It should be specified for codecs with dynamic payload type, such as iSAC, iLBC, OPUS, and the valid values are between 96 and 127.

The setting `fmtpt` is a string with valid values as defined by RFC3555 for codec *g729* and RFC7587 for codec *Opus*.

Important

While any parameter defined by RFC7587 can be specified as the value of the `fmtpt` for the Opus codec, SIP Endpoint SDK considers only the `maxaveragebitrate` parameter, which specifies the maximum average bit rate of a session in bits per second (bit/s). Bit rate values can be from 6000 to 64000 (any value outside the range will be adjusted to the minimum/maximum allowed). Recommended bit-rate values:

- from 6000 to 8000 to save bandwidth, with voice quality comparable to G.729 codec.
- 10000 for narrow-band quality (comparable to G.711).
- 20000 for wide-band quality.

The codec *ulpfec/90000* supports the *vp8* codec with forward error correction.

Example

```
<domain name="codecs">
  <section name="enabled">
    <setting name="audio" value="g729,pcmu,pcma"/>
    <setting name="video" value="h264,maxpkt=1200"/>
  </section>
  <section name="G729/8000">
    <setting name="fmtpt" value="annexb=no"/>
  </section>
  <section name="PCMU/8000"/>
  <section name="PCMA/8000"/>
  <section name="G722/16000"/>
  <section name="iLBC/8000">
    <setting name="payload_type" value="102"/>
  </section>
</domain>
```

```
<section name="iSAC/32000">
  <setting name="payload_type" value="104"/>
</section>
<section name="iSAC/16000">
  <setting name="payload_type" value="103"/>
</section>
<section name="g729/8000">
  <setting name="fmt" value="annexb=yes"/>
</section>
<section name="opus/48000/2">
  <setting name="payload_type" value="120"/>
  <setting name="fmt" value="maxaveragebitrate=20000"/>
</section>
</section>
<section name="vp8">
  <setting name="payload_type" value="100"/>
</section>
<section name="vp9">
  <setting name="payload_type" value="101"/>
</section>
<section name="h264">
  <setting name="payload_type" value="108"/>
  <setting name="fmt" value="profile-level-id=420028"/>
</section>
<section name="ulpfec/90000">
  <setting name="payload_type" value="97"/>
</section>
</domain>
```

Configuring Capture Devices

In order to define the list of preferred capture devices and their priorities, the devices should be added to the device policy section in the **SipEndpoint.config** file:

```
<domain name="policy">
  <section name="device">
    <!--
      The priority of a device depends on its position in this section.
      The higher the position, the higher device priority.
      To disable a device it should be commented out or removed from the file.
    -->
    . . .
    <!-- Capture -->
    <setting name="capture_device" value="CaptureDeviceName-HighPriority"/>
    <setting name="capture_device" value="CaptureDeviceName-LowPriority"/>
  </section>
</domain>
```

Audio Device Settings

You can configure a headset and other audio input devices using the following parameters in the configuration file:

- use_headset
- headset_name
- exclude_headset
- audio_in_device
- audio_out_device

If the SDK cannot access any of the configured audio devices, the SDK uses the default system audio devices.

If use_headset is set to 1, the SDK searches a headset with name matching configured headset_name but not matching exclude_headset value. If the SDK does not find the headset, it uses the system defaults.

If use_headset is set to 0, the SDK skips the search for a headset and searches for audio in and audio out devices with names configured by audio_in_device and audio_out_device. If the devices are not found, the SDK uses the system defaults.

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)

Important

A device is selected when the device name matches or partially-matches the regular expression configured in the parameters.

Auto-answer and Call Rejection

The auto-answer and call rejection features depend on the use_headset, auto_answer, and reject_session_when_headset_na configuration settings and whether or not audio devices are available. The following tables describe auto-answer and call rejection behavior:

Auto-answer and Call Rejection when use_headset=1

<p><i><u>Headset is Available</u></i></p> <p>The SDK considers a headset available if the headset is found by name.</p> <p>Outgoing calls can be initiated.</p>	auto_answer=1	Incoming calls are answered automatically:
	auto_answer=0	Incoming calls are answered manually and the user explicitly

		selects whether or not video streams should be accepted (using the <code>has_video</code> parameter supplied in the <code>gs_session_info</code> argument)
<p><i>Headset is Not Available</i></p> <p>The SDK decides that no headset is available if a headset was not found by name.</p> <p>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 or the system defaults.</p>	<p>No auto-answer is possible in this sub-case, so the <code>auto_answer</code> setting is not used</p>	<p><code>reject_session_when_headset_na=1</code></p> <ul style="list-style-type: none"> • Incoming calls are automatically rejected • Outgoing calls are blocked
		<p><code>reject_session_when_headset_na=0</code></p> <ul style="list-style-type: none"> • 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

Auto-answer and Call Rejection when `use_headset=0`

Audio devices are configured using the `audio_in_device` and `audio_out_device` settings. If no valid `audio_in_device` and `audio_out_device` devices are found, the SDK selects the system defaults. Outgoing calls can be initiated.

<p><i>Both microphone and speaker are available</i></p>	<p><code>auto_answer=1</code></p>	<p>Incoming calls are answered automatically:</p> <ul style="list-style-type: none"> • As audio if <code>auto_accept_video=0</code> • As audio with video if the call has video and <code>auto_accept_video=1</code>
	<p><code>auto_answer=0</code></p>	<p>Incoming calls are answered manually and the user explicitly selects whether or not video streams should be accepted (using the <code>has_video</code> parameter supplied in the <code>gs_session_info</code> argument)</p>
<p><i>Either microphone or speaker is not available</i></p>	<p>No auto-answer is possible in this sub-case, so the <code>auto_answer</code> setting is not used</p>	<p>Auto-rejection is not applicable, so the <code>reject_session_when_headset_na</code></p>

<ul style="list-style-type: none">• 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		setting is not used
--	--	---------------------

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, an incoming call is rejected with a SIP response code as configured in the `sip_code_when_headset_na` setting.