Genesys Voice Platform

mpc Section

5/9/2025

mpc Section

- alarminterval
- amr_wb.maxptime
- amr_wb.ptime
- amr_wbpayload
- amr.enable_dtx
- amr.fmtp
- amr.maxptime
- amr.preferred_mode
- amr.ptime
- amrpayload
- amrwb.enable_dtx
- amrwb.fmtp
- amrwb.preferred_mode
- answerwithonecodec
- appendrejcodec
- asr.codec
- asr.preferredipinterface
- asr.srtp.cryptomethods
- asr.srtp.mode
- asr.srtp.sessionparamsoffer
- codec
- codecpref
- conference.allowloudestvideoecho
- cpa.busy
- cpa.carriermsg.0
- cpa.carriermsg.1
- cpa.carriermsg.2
- cpa.carriermsg.3
- cpa.carriermsg.4
- cpa.carriermsg.5
- cpa.carriermsg.6
- cpa.carriermsg.7
- cpa.carriermsg.8
- cpa.carriermsg.9
- cpa.carriermsg.readduration
- cpa.cm_enable_initial_tone_filter
- cpa.cm_initial_silence_suppression_level
- cpa.cm_match_percent
- cpa.custom1
- cpa.custom2
- cpa.custom3
- cpa.custom4
- cpa.enable_alternate_signals
- cpa.enable_carrier_messages
- cpa.enable_log_param
- cpa.enable_log_result
- cpa.fastbusy
- cpa.fax
- cpa.faxdur
- cpa.keptdur_before_statechange
- cpa.maxbeepdettime
- cpa.maxpostconntime
- cpa.maxpreconntime
- cpa.maxrings
- cpa.mintime_after_tone
- cpa.nframes_cm_detection
- cpa.no_ring_result
- cpa.postconnectresult.machine.list
- cpa.postconnsilduration
- cpa.postconnsilresult
- cpa.preconn_tones_det_mode
- cpa.preconnectresult.busy.list
- cpa.preconnectresult.custom1.list
- cpa.preconnectresult.custom2.list
- cpa.preconnectresult.custom3.list
- cpa.preconnectresult.custom4.list
- cpa.preconnectresult.fast_busy.list
- cpa.preconnectresult.sit_nocircuit.list
- cpa.preconnectresult.sit_operatorintercept.l
- cpa.preconnectresult.sit_reorder.list
- cpa.preconnectresult.sit_vacantcircuit.list
- cpa.priority_machine_machinegreetingdur
- cpa.priority_machine_maxvoicesigdur
- cpa.priority_machine_voicepausedur
- cpa.priority_normal_machinegreetingdur
- cpa.priority_normal_maxvoicesigdur
- cpa.priority_normal_voicepausedur
- cpa.priority_voice_machinegreetingdur
- cpa.priority_voice_maxvoicesigdur
- cpa.priority_voice_voicepausedur
- cpa.ringback
- cpa.sit_nocircuit
- cpa.sit_operatorintercept
- cpa.sit_reorder
- cpa.sit_vacantcircuit
- cpa.tone1.segment1.f1max
- cpa.tone1.segment1.f1min
- cpa.tone1.segment1.f2max
- cpa.tone1.segment1.f2min
- cpa.tone1.segment1.offtimemax

- cpa.tone1.segment1.offtimemin
- cpa.tone1.segment1.ontimemax
- cpa.tone1.segment1.ontimemin
- cpa.tone1.segment2.f1max
- cpa.tone1.segment2.f1min
- cpa.tone1.segment2.f2max
- cpa.tone1.segment2.f2min
- cpa.tone1.segment2.offtimemax
- cpa.tone1.segment2.offtimemin
- cpa.tone1.segment2.ontimemax
- cpa.tone1.segment2.ontimemin
- cpa.tone1.segment3.f1max
- cpa.tone1.segment3.f1min
- cpa.tone1.segment3.f2max
- cpa.tone1.segment3.f2min
- cpa.tone1.segment3.offtimemax
- cpa.tone1.segment3.offtimemin
- cpa.tone1.segment3.ontimemax
- cpa.tone1.segment3.ontimemin
- cpa.tone10.segment1.f1max
- cpa.tone10.segment1.f1min
- cpa.tone10.segment1.f2max
- cpa.tone10.segment1.f2min
- cpa.tone10.segment1.offtimemax
- cpa.tone10.segment1.offtimemin
- cpa.tone10.segment1.ontimemax
- cpa.tone10.segment1.ontimemin
- cpa.tone10.segment2.f1max
- cpa.tone10.segment2.f1min
- cpa.tone10.segment2.f2max
- cpa.tone10.segment2.f2min
- cpa.tone10.segment2.offtimemax
- cpa.tone10.segment2.offtimemin
- cpa.tone10.segment2.ontimemax
- cpa.tone10.segment2.ontimemin
- cpa.tone10.segment3.f1max
- cpa.tone10.segment3.f1min
- cpa.tone10.segment3.f2max
- cpa.tone10.segment3.f2min
- cpa.tone10.segment3.offtimemax
- cpa.tone10.segment3.offtimemin
- cpa.tone10.segment3.ontimemax
- cpa.tone10.segment3.ontimemin
- cpa.tone2.segment1.f1max
- cpa.tone2.segment1.f1min
- cpa.tone2.segment1.f2max
- cpa.tone2.segment1.f2min
- cpa.tone2.segment1.offtimemax
- cpa.tone2.segment1.offtimemin
- cpa.tone2.segment1.ontimemax
- cpa.tone2.segment1.ontimemin
- cpa.tone2.segment2.f1max
- cpa.tone2.segment2.f1min
- cpa.tone2.segment2.f2max
- cpa.tone2.segment2.f2min
- cpa.tone2.segment2.offtimemax
- cpa.tone2.segment2.offtimemin
- cpa.tone2.segment2.ontimemax
- cpa.tone2.segment2.ontimemin
- cpa.tone2.segment3.f1max
- cpa.tone2.segment3.f1min
- cpa.tone2.segment3.f2max
- cpa.tone2.segment3.f2min
- cpa.tone2.segment3.offtimemax
- cpa.tone2.segment3.offtimemin
- cpa.tone2.segment3.ontimemax
- cpa.tone2.segment3.ontimemin
- cpa.tone3.segment1.f1max
- cpa.tone3.segment1.f1min
- cpa.tone3.segment1.f2max
- cpa.tone3.segment1.f2min
- cpa.tone3.segment1.offtimemax
- cpa.tone3.segment1.offtimemin
- cpa.tone3.segment1.ontimemax
- cpa.tone3.segment1.ontimemin
- cpa.tone3.segment2.f1max
- cpa.tone3.segment2.f1min
- cpa.tone3.segment2.f2max
- cpa.tone3.segment2.f2min
- cpa.tone3.segment2.offtimemax
- cpa.tone3.segment2.offtimemin
- cpa.tone3.segment2.ontimemax
- cpa.tone3.segment2.ontimemin
- cpa.tone3.segment3.f1max
- cpa.tone3.segment3.f1min
- cpa.tone3.segment3.f2max
- cpa.tone3.segment3.f2min
- cpa.tone3.segment3.offtimemax
- cpa.tone3.segment3.offtimemin
- cpa.tone3.segment3.ontimemax
- cpa.tone3.segment3.ontimemin
- cpa.tone4.segment1.f1max
- cpa.tone4.segment1.f1min
- cpa.tone4.segment1.f2max
- cpa.tone4.segment1.f2min
- cpa.tone4.segment1.offtimemax
- cpa.tone4.segment1.offtimemin
- cpa.tone4.segment1.ontimemax
- cpa.tone4.segment1.ontimemin
- cpa.tone4.segment2.f1max
- cpa.tone4.segment2.f1min
- cpa.tone4.segment2.f2max
- cpa.tone4.segment2.f2min
- cpa.tone4.segment2.offtimemax
- cpa.tone4.segment2.offtimemin

- cpa.tone4.segment2.ontimemax
- cpa.tone4.segment2.ontimemin
- cpa.tone4.segment3.f1max
- cpa.tone4.segment3.f1min
- cpa.tone4.segment3.f2max
- cpa.tone4.segment3.f2min
- cpa.tone4.segment3.offtimemax
- cpa.tone4.segment3.offtimemin
- cpa.tone4.segment3.ontimemax
- cpa.tone4.segment3.ontimemin
- cpa.tone5.segment1.f1max
- cpa.tone5.segment1.f1min
- cpa.tone5.segment1.f2max
- cpa.tone5.segment1.f2min
- cpa.tone5.segment1.offtimemax
- cpa.tone5.segment1.offtimemin
- cpa.tone5.segment1.ontimemax
- cpa.tone5.segment1.ontimemin
- cpa.tone5.segment2.f1max
- cpa.tone5.segment2.f1min
- cpa.tone5.segment2.f2max
- cpa.tone5.segment2.f2min
- cpa.tone5.segment2.offtimemax
- cpa.tone5.segment2.offtimemin
- cpa.tone5.segment2.ontimemax
- cpa.tone5.segment2.ontimemin
- cpa.tone5.segment3.f1max
- cpa.tone5.segment3.f1min
- cpa.tone5.segment3.f2max
- cpa.tone5.segment3.f2min
- cpa.tone5.segment3.offtimemax
- cpa.tone5.segment3.offtimemin
- cpa.tone5.segment3.ontimemax
- cpa.tone5.segment3.ontimemin
- cpa.tone6.segment1.f1max
- cpa.tone6.segment1.f1min
- cpa.tone6.segment1.f2max
- cpa.tone6.segment1.f2min
- cpa.tone6.segment1.offtimemax
- cpa.tone6.segment1.offtimemin
- cpa.tone6.segment1.ontimemax
- cpa.tone6.segment1.ontimemin
- cpa.tone6.segment2.f1max
- cpa.tone6.segment2.f1min
- cpa.tone6.segment2.f2max
- cpa.tone6.segment2.f2min
- cpa.tone6.segment2.offtimemax
- cpa.tone6.segment2.offtimemin
- cpa.tone6.segment2.ontimemax
- cpa.tone6.segment2.ontimemin
- cpa.tone6.segment3.f1max
- cpa.tone6.segment3.f1min
- cpa.tone6.segment3.f2max
- cpa.tone6.segment3.f2min
- cpa.tone6.segment3.offtimemax
- cpa.tone6.segment3.offtimemin
- cpa.tone6.segment3.ontimemax
- cpa.tone6.segment3.ontimemin
- cpa.tone7.segment1.f1max
- cpa.tone7.segment1.f1min
- cpa.tone7.segment1.f2max
- cpa.tone7.segment1.f2min
- cpa.tone7.segment1.offtimemax
- cpa.tone7.segment1.offtimemin
- cpa.tone7.segment1.ontimemax
- cpa.tone7.segment1.ontimemin
- cpa.tone7.segment2.f1max
- cpa.tone7.segment2.f1min
- cpa.tone7.segment2.f2max
- cpa.tone7.segment2.f2min
- cpa.tone7.segment2.offtimemax
- cpa.tone7.segment2.offtimemin
- cpa.tone7.segment2.ontimemax
- cpa.tone7.segment2.ontimemin
- cpa.tone7.segment3.f1max
- cpa.tone7.segment3.f1min
- cpa.tone7.segment3.f2max
- cpa.tone7.segment3.f2min
- cpa.tone7.segment3.offtimemax
- cpa.tone7.segment3.offtimemin
- cpa.tone7.segment3.ontimemax
- cpa.tone7.segment3.ontimemin
- cpa.tone8.segment1.f1max
- cpa.tone8.segment1.f1min
- cpa.tone8.segment1.f2max
- cpa.tone8.segment1.f2min
- cpa.tone8.segment1.offtimemax
- cpa.tone8.segment1.offtimemin
- cpa.tone8.segment1.ontimemax
- cpa.tone8.segment1.ontimemin
- cpa.tone8.segment2.f1max
- cpa.tone8.segment2.f1min
- cpa.tone8.segment2.f2max
- cpa.tone8.segment2.f2min
- cpa.tone8.segment2.offtimemax
- cpa.tone8.segment2.offtimemin
- cpa.tone8.segment2.ontimemax
- cpa.tone8.segment2.ontimemin
- cpa.tone8.segment3.f1max
- cpa.tone8.segment3.f1min
- cpa.tone8.segment3.f2max
- cpa.tone8.segment3.f2min
- cpa.tone8.segment3.offtimemax
- cpa.tone8.segment3.offtimemin
- cpa.tone8.segment3.ontimemax

- cpa.tone8.segment3.ontimemin
- cpa.tone9.segment1.f1max
- cpa.tone9.segment1.f1min
- cpa.tone9.segment1.f2max
- cpa.tone9.segment1.f2min
- cpa.tone9.segment1.offtimemax
- cpa.tone9.segment1.offtimemin
- cpa.tone9.segment1.ontimemax
- cpa.tone9.segment1.ontimemin
- cpa.tone9.segment2.f1max
- cpa.tone9.segment2.f1min
- cpa.tone9.segment2.f2max
- cpa.tone9.segment2.f2min
- cpa.tone9.segment2.offtimemax
- cpa.tone9.segment2.offtimemin
- cpa.tone9.segment2.ontimemax
- cpa.tone9.segment2.ontimemin
- cpa.tone9.segment3.f1max
- cpa.tone9.segment3.f1min
- cpa.tone9.segment3.f2max
- cpa.tone9.segment3.f2min
- cpa.tone9.segment3.offtimemax
- cpa.tone9.segment3.offtimemin
- cpa.tone9.segment3.ontimemax
- cpa.tone9.segment3.ontimemin
- cpa.voice_level_db
- cpa.voice_range_db
- ctrleventpoollowthreshold
- ctrleventpoolthreshold
- default_audio_format
- dsp.g726littleendian
- dsp.g729a
- dtmf.detectedge
- dtmf.duration
- dtmf.gap

- dtmf.inband_amplitude
- dtmf.maxsilence
- dtmf.minduration
- dtmf.multidtmfonetimestamp
- dtmf.pauseduration
- dtmf.singlepacket
- fcr.defaultdtmfhandling
- fcr.gain
- font_paths_linux
- font_paths_win
- g722.maxptime
- g722.ptime
- g726_32.maxptime
- g726_32.ptime
- g729.fmtp
- g729.maxptime
- g729.ptime
- gsm.maxptime
- gsm.ptime
- h263_1998payload
- h263.fmtp
- h264.fmtp
- h264.in_band_param_sets_only
- h264payload
- health.maxprocessingtime
- health.waittime
- includeavpfinsdp
- maxmediathreads
- maxrecordencryptedfilesize
- maxrecordfilesize
- media.senddtmfdropaudio
- mediamgr.audiobuffersize
- mediamgr.autorecordformatselect
- mediamgr.CA_directory
- mediamgr.CA_file

- mediamgr.enableEODdoublecheck
- mediamgr.h263overrideTR
- mediamgr.hlsconsecutiveerrorsthreshold
- mediamgr.hlstotalerrorsthreshold
- mediamgr.ignore_cert_err
- mediamgr.isofilerecordheadersize
- mediamgr.maxcertificatecachesize
- mediamgr.maxcertificatelength
- mediamgr.maxcertificatesperprofile
- mediamgr.precacheprofileforcallrecording.0
- mediamgr.precacheprofileforcallrecording.1
- mediamgr.precacheprofileforcallrecording.2
- mediamgr.precacheprofileforcallrecording.3
- mediamgr.precacheprofileforcallrecording.4
- mediamgr.precacheprofileforcallrecording.5
- mediamgr.rec_iframe_delay_threshold
- mediamgr.recordmp3audiobuffer
- mediamgr.recordrtphinttrack
- mediamgr.recordwritetimeinterval
- mediamgr.rtsplowerbufferthreshold
- mediamgr.rtsppause
- mediamgr.rtspplayrange
- mediamgr.rtspupperbufferthreshold
- mediamgr.sharedhttpservers
- mediamgr.strictsamplingrate
- mediamgr.videobuffersize
- mediamgr.videofillingframeduration
- mediamgr.videofillingthreshold
- mixer.audiodelay_flush_all_threshold
- mixer.audiodelay_flush_silence_threshold
- mp3.bitrate
- mp3.compression_level_current_encoder
- mp3.interfrequency.encoding
- mp3.samplingrate
- mp3.use_current_encoder

- mp3.use_integer_transcoder
- mp3.use_integer_transcoder_current_request
- numdispatchthreads
- opus.apptype
- opus.bitrate
- opus.complexity
- opus.fmtp
- opus.packetloss
- opuspayload
- pcma.maxptime
- pcma.ptime
- pcmu.maxptime
- pcmu.ptime
- persistdympayfmtpair
- playcache.checkversiontime
- playcache.directory
- playcache.enable
- playcache.expiretime
- playcache.maxsize
- playremoteeodtimeout
- playremoteflushtimeout
- playsilencefill
- preferredipinterface
- record.allowsyncdiskwrite
- record.defaultdtmfhandling
- recordcachedir
- recordnumparallelpost
- recordpostbacklogthreshold
- recordpostinterval
- recordpostretrybackoff
- recordpostretrycount
- refframereqonconnjoin
- rru.beginsilence
- rru.endsilence
- rtcp.tos

- rtcp.tos.video
- rtcpfeedback.audio
- rtcpfeedback.video
- rtp.activetimeout
- rtp.audiobuffersize
- rtp.dejitter.delay
- rtp.dejitter.timeout
- rtp.dtmf.crlfenable
- rtp.dtmf.receive
- rtp.dtmf.send
- rtp.enablertcp
- rtp.filteroutduplicate
- rtp.fixedsocketthreads
- rtp.h264allowrfc3984stapa
- rtp.inputmode
- rtp.localaddr
- rtp.localaddrv6
- rtp.maxrtppacketsize
- rtp.multichantimeout
- rtp.overwritessrcandtimestamp
- rtp.packetseq
- rtp.portrange
- rtp.prefilltime
- rtp.request_iframe
- rtp.restrictsource
- rtp.rfc2429maxpacketsize
- rtp.sendmode
- rtp.senduponrecv
- rtp.source_buffer_video_data_size
- rtp.source_buffer_video_size
- rtp.statisticsinterval
- rtp.timeout
- rtp.tos
- rtp.tos.video
- rtp.video.udprecvbuffersize

- rtp.video.udpsendbuffersize
- rtp.videobuffersize
- rtp.vp8maxpacketsize
- rtsp.connection.portrange
- rtsp.localaddr
- rtsp.localaddrv6
- rtsp.rtp.localaddr
- rtsp.rtp.localaddrv6
- rtsp.rtp.portrange
- sdp.audiobandwidth
- sdp.connection
- sdp.map.origin.0
- sdp.map.origin.0.confgain
- sdp.map.origin.0.dtmftype
- sdp.map.origin.1
- sdp.map.origin.1.confgain
- sdp.map.origin.1.dtmftype
- sdp.map.origin.2
- sdp.map.origin.2.confgain
- sdp.map.origin.2.dtmftype
- sdp.map.origin.3
- sdp.map.origin.3.confgain
- sdp.map.origin.3.dtmftype
- sdp.map.origin.4
- sdp.map.origin.4.confgain
- sdp.map.origin.4.dtmftype
- sdp.map.origin.5
- sdp.map.origin.5.confgain
- sdp.map.origin.5.dtmftype
- sdp.map.origin.6
- sdp.map.origin.6.confgain
- sdp.map.origin.6.dtmftype
- sdp.map.origin.7
- sdp.map.origin.7.confgain
- sdp.map.origin.7.dtmftype

- sdp.map.origin.8
- sdp.map.origin.8.confgain
- sdp.map.origin.8.dtmftype
- sdp.map.origin.9
- sdp.map.origin.9.confgain
- sdp.map.origin.9.dtmftype
- sdp.videobandwidth
- srtp.cryptomethods
- srtp.maxerror
- srtp.mode
- srtp.sessionparams
- srtp.sessionparamsoffer
- telephone_event.maxptime
- telephone_event.ptime
- telephone_eventpayload
- tfcipayload
- tiasfraction

- transcoders
- transmitmultiplecodec
- tts.appendrejcodec
- tts.codec
- tts.preferredipinterface
- tts.srtp.cryptomethods
- tts.srtp.mode
- tts.srtp.sessionparamsoffer
- validatemediatimers
- videotranscoder.bitratecheckdelay
- videotranscoder.bitratechecktolerance
- videotranscoder.checkbitrate
- videotranscoder.checkframerate
- videotranscoder.frameratechecktolerance
- videotranscoder.h264.keyframeidrinterval
- videotranscoder.h264.keyframeinterval
- videotranscoder.h264.resolutions

- videotranscoder.maxbitrate
- videotranscoder.statsresetthreshold
- voipmetrics.enable
- vp8.adaptive
- vp8.defaultbitrate
- vp8.defaultframerateden
- vp8.defaultframeratenum
- vp8.defaultresolution
- vp8.maxkeyframeinterval
- vrmrecorder.codec
- vrmrecorder.enable
- vrmrecorder.preferredipinterface
- vrmrecorder.srtp.cryptomethods
- vrmrecorder.srtp.mode
- vrmrecorder.srtp.sessionparamsoffer
- wldebandconferences

> **Tip**
>
> The following descriptions were generated by dynamic query and include cached results that may be up to one day old. Click here to refresh the query.

alarminterval

Default Value: 900000
Valid Values: mpc.alarminterval must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

This parameter specifies the time interval to wait between sending failure alarm notifications. An initial alarm is generated on the first failure and if the failure continues we send another alarm notification with the updated failure count once the time interval specified by this parameter expires. The default value of this parameter is 15 minutes(900000 milli seconds), which means that we would send failure alarm notification for the first failure and then send another alarm notification after 15

minutes. This process is repeated until the root cause for the failure has been rectified.

amr_wb.maxptime

Default Value: 0
Valid Values: Choose between: 0, 20, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

amr_wb.ptime

Default Value: 20
Valid Values: Choose between: 0, 20, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will take precedence.

amr_wbpayload

Default Value: 112
Valid Values: A valid AMR-WB Payload can only be an integer from 96 to 127 inclusive
Changes Take Effect: At start/restart

Default payload type number to use for the AMR-WB codec

amr.enable_dtx

Default Value: 1

Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

This parameter controls whether the AMR transcoder will generate comfort noise frames when transcoding data to AMR format for which the voice activity detector indicates no speech.

amr.fmtp

Default Value: octet-align=0 | octet-align=1
Valid Values: <parameter=value>[|<parameter=value>]*
Changes Take Effect: At start/restart

Specifies the AMR SDP RTP payload configurations offered and accepted by the MCP. Set to one or more fmtp text values separated by the '|' character. The fmtp text is the same as would appear in the SDP negotiation (see RFC 4867). Each "|" separated value configures an AMR payload type. There are two fmtp parameters that can be set for each payload type, octet-align and mode-set.

Setting octet-align=0 or octet-align=1 disables or enables octet align mode for the payload.

Setting mode-set restricts the AMR modes for the payload. For example, setting "mode-set=0,1" only allows modes 0 and 1. If mode-set is not set, all modes are allowed unless the mode-set is restricted by the remote end. Valid modes are inclusively 0 to 7.

For example, setting this parameter to "octet-align=1" enables one payload type with octet aligned mode enabled and any mode allowed, and setting it to "octet-align=0 | octet-align=1" enables two payload types, one with bandwidth efficient mode enabled and any mode allowed, and one with octet aligned enabled and any mode allowed.

Note, the mode-set parameter can cause transcoding to be required. For example, if a prompt to be played is in AMR format mode 5, but only mode 0 is allowed in the payload, a transcoder will be invoked to transcode from AMR mode 5 to AMR mode 0.

Some AMR implementations may specify a fmtp options that are not actually activated for the payload. To work around this, the mpc.amr.fmtp can be set to "*". For this setting, all fmtp content in an SDP offer will be ignored and "octet-align=0" will be returned in the SDP answer. Similarly, an offer for this configuration will be set to "octet-align=0", and all fmtp content in the answer will be ignored.

amr.maxptime

Default Value: 0
Valid Values: Choose between: 0, 20, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or

is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

amr.preferred_mode

Default Value: 15
Valid Values: Choose between: 0, 1, 2, 3, 4, 5, 6, 7 or 15 (Disable)
Changes Take Effect: Immediately

Specifies the AMR Preferred Codec Mode. This is the value that the MCP sends to the remote end as the preferred mode for AMR data sent to the MCP.

amr.ptime

Default Value: 20
Valid Values: Choose between: 0, 20, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will take precedence.

amrpayload

Default Value: 105
Valid Values: A valid AMR Payload can only be an integer from 96 to 127 inclusive
Changes Take Effect: At start/restart

Default payload type number to use for the AMR codec

amrwb.enable_dtx

Default Value: 1
Valid Values: Choose between: 0 or 1

Changes Take Effect: At start/restart

This parameter controls whether the AMR-WB transcoder will generate comfort noise frames when transcoding data to AMR-WB format for which the voice activity detector indicates no speech.

amrwb.fmtp

Default Value: octet-align=0 | octet-align=1
Valid Values: <parameter=value>[|<parameter=value>]*
Changes Take Effect: At start/restart

Specifies the AMR-WB SDP RTP payload configurations offered and accepted by the MCP. Set to one or more fmtp text values separated by the '|' character. The fmtp text is the same as would appear in the SDP negotiation (see RFC 4867). Each "|" separated value configures an AMR-WB payload type. There are two fmtp parameters that can be set for each payload type, octet-align and mode-set.

Setting octet-align=0 or octet-align=1 disables or enables octet align mode for the payload.

Setting mode-set restricts the AMR-WB modes for the payload. For example, setting "mode-set=0,1" only allows modes 0 and 1. If mode-set is not set, all modes are allowed unless the mode-set is restricted by the remote end. Valid modes are inclusively 0 to 8.

For example, setting this parameter to "octet-align=1" enables one payload type with octet aligned mode enabled and any mode allowed, and setting it to "octet-align=0 | octet-align=1" enables two payload types, one with bandwidth efficient mode enabled and any mode allowed, and one with octet aligned enabled and any mode allowed.

Note, the mode-set parameter can cause transcoding to be required. For example, if a prompt to be played is in AMR-WB format mode 5, but only mode 0 is allowed in the payload, a transcoder will be invoked to transcode from AMR-WB mode 5 to AMR-WB mode 0.

Some AMR-WB implementations may specify a fmtp options that are not actually activated for the payload. To work around this, the mpc.amrwb.fmtp can be set to "*". For this setting, all fmtp content in an SDP offer will be ignored and "octet-align=0" will be returned in the SDP answer. Similarly, an offer for this configuration will be set to "octet-align=0", and all fmtp content in the answer will be ignored.

amrwb.preferred_mode

Default Value: 15
Valid Values: Choose between: 0, 1, 2, 3, 4, 5, 6, 7 or 15 (Disable)
Changes Take Effect: Immediately

Specifies the AMR-WB Preferred Codec Mode. This is the value that the MCP sends to the remote end as the preferred mode for AMR-WB data sent to the MCP.

answerwithonecodec

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately

When an SDP offer contains more than one codec per media line, this can be used to limit the codecs in the answer to one (the most preferred) when enabled. If disabled (the default), all the negotiated codecs will be returned in the answer.

appendrejcodec

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

When set to Enable, the MCP will advertise all supported codecs when generating an SDP answer or SDP offer. Even if codecs are rejected or not presented in the caller's SDP, the MCP will still support receiving these codecs. The MCP will not send for those SDPs unless a payload is presented by the caller.

asr.codec

Default Value: pcmu telephone-event
Valid Values: Choose between: "pcmu telephone-event", "pcma telephone-event", "g722 telephone-event", "g726 telephone-event", "g729 telephone-event", "gsm telephone-event", "amr telephone-event", "amr-wb telephone-event" or "tfci telephone-event".
Changes Take Effect: At start/restart

Specifies the MRCP ASR codec to be used.

asr.preferredipinterface

Default Value: V4
Valid Values: Choose between: V4 or V6
Changes Take Effect: At start/restart

Specifies the preferred IP interface to use (IPv4 or IPv6) for MRCP ASR when performing SDP

negotiation. In particular, this will be used to set the root connection attribute in SDP answers, and set the connection attribute in SDP offers.

asr.srtp.cryptomethods

Default Value: AES_CM_128_HMAC_SHA1_80
Valid Values: Any combination of: "AES_CM_128_HMAC_SHA1_80" and "AES_CM_128_HMAC_SHA1_32". Or "none".
Changes Take Effect: At start/restart

List of crypto suites corresponding to advertised capabilities offered by the MCP to the MRCP ASR server using SDP. See RFC4568 for the description of the suites. Methods can only contain alphanumeric characters.

asr.srtp.mode

Default Value: none
Valid Values: none: No SRTP supported: the MCP will ignore the crypto. offer: SRTP supported in outgoing SDP offers. If the other side ignores SRTP, the MCP will fall back to non SRTP mode. offer_strict: Same as offer, however if the other side doesn't use SRTP, negotiation will fail.
Changes Take Effect: At start/restart

Specifies the srtp mode for the MCP to use for MRCP ASR sessions.

asr.srtp.sessionparamsoffer

Default Value: none
Valid Values: Any combination of: "UNENCRYPTED_SRTP", "UNENCRYPTED_SRTCP" and "UNAUTHENTICATED_SRTP". Or "none".
Changes Take Effect: At start/restart

List of session parameters that the MCP will include in its SDP offers to the MRCP ASR server. See RFC4568 for their description. Note that RFC4568 doesn't allow unauthenticated srtcp.

codec

Default Value: pcmu pcma g722 opus g726 g729 gsm amr amr-wb h263 h263-1998 h264 vp8 telephone-event
Valid Values: Any combination of: pcmu, pcma, g722, opus, g726, g729, gsm, amr, amr-wb, tfci,

h263, h263-1998, h264, vp8 or telephone-event.
Changes Take Effect: Immediately/session

List of codec corresponding to advertised capabilities offered by the MCP using SDP. The offered codec list will control the codecs that are offered by the MCP to the remote party for media sent from the remote party to Genesys. The order of the codecs will determine the order in SDP offer presented by the MCP. telephone-event is mandatory if RFC2833 DTMF relay is required. If the telephone-event is not set then the mpc.rtp.dtmf.send and mpc.rtp.dtmf.receive will control the DTMF relay method.

>> Back to Top

codecpref

Default Value: r
Valid Values: Choose between: r or l
Changes Take Effect: Immediately/session

Specifies whether remote or local preferences will be used to interpret the accept codec list. If remote preferences are used, then the effective accept list will be the format list offered by the remote entity, filtered to include only those entries also on the locally configured list. If local preferences are used, then the local accept list will be used, but only including those capabilities offered by the remote entity. The "mpc.codecpref" parameter will be used to control this, and can be set to either Remote or Local; the default value will be Remote. r - Remote l - Local

>> Back to Top

conference.allowloudestvideoecho

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

Only applicable if the conference.video_output_algorithm is loudest or confrole where it falls back to the loudest. This parameter controls who the loudest speaker will see as the video. If true, the loudest speaker will see oneself. If false, the loudest spaker will see the previously selected video who was not oneself (if no previous, then no video).

>> Back to Top

cpa.busy

Default Value: na_busy
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's busy pattern. Default tone value is builtin North American busy (NA busy). Value "Disable" disables busy detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

The builtin North American busy has three segments. All segments are the same and are defined as follows:
Segment 1: f1=460-500(Hz), f2=600-640(Hz), ontime=360-640(ms), offtime=360-640(ms)
Segment 2: f1=460-500(Hz), f2=600-640(Hz), ontime=360-640(ms), offtime=360-640(ms)
Segment 3: f1=460-500(Hz), f2=600-640(Hz), ontime=360-640(ms), offtime=360-640(ms)

>> Back to Top

cpa.carriermsg.0

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://. Examples: cpa.carriermsg.0 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16 cpa.carriermsg.0 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.0 = http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages. The limitation of 100 messages is due to performance.

>> Back to Top

cpa.carriermsg.1

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://. Examples: cpa.carriermsg.1 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16 cpa.carriermsg.1 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.1 = http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages. The limitation of 100 messages is due to performance.

>> Back to Top

cpa.carriermsg.2

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://. Examples: cpa.carriermsg.2 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16

cpa.carriermsg.2 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.2 = http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages. The limitation of 100 messages is due to performance.

cpa.carriermsg.3

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://. Examples: cpa.carriermsg.3 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16 cpa.carriermsg.3 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.3 = http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages. The limitation of 100 messages is due to performance.

cpa.carriermsg.4

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://. Examples: cpa.carriermsg.4 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16 cpa.carriermsg.4 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.4 = http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages. The limitation of 100 messages is due to performance.

cpa.carriermsg.5

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://. Examples: cpa.carriermsg.5 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16 cpa.carriermsg.5 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.5 = http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages. The limitation of 100 messages is due to performance.

cpa.carriermsg.6

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://.
Examples: cpa.carriermsg.6 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16
cpa.carriermsg.6 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.6 =
http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages.
The limitation of 100 messages is due to performance.

cpa.carriermsg.7

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://.
Examples: cpa.carriermsg.7 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16
cpa.carriermsg.7 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.7 =
http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages.
The limitation of 100 messages is due to performance.

cpa.carriermsg.8

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://.
Examples: cpa.carriermsg.8 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16
cpa.carriermsg.8 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.8 =
http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages.
The limitation of 100 messages is due to performance.

cpa.carriermsg.9

Default Value:
Valid Values: Can be an empty string or a valid path to a carrier message file.
Changes Take Effect: Immediately

Specifies full path for the file with carrier message. Path requires a schema of file://, http:// or https://. Examples: cpa.carriermsg.9 = file://E:\cpatest\carrierMsg\cm_NotInUse_Operator1.pcm16 cpa.carriermsg.9 = file:///opt/msg/cm_NotInUse_Operator1.pcm16 cpa.carriermsg.9 = http://localhost/cm_NotInUse_Operator1.wav You can configure as many as 100 carrier messages. The limitation of 100 messages is due to performance.

cpa.carriermsg.readduration

Default Value: 60
Valid Values: The number must be a non-negative integer, greater than or equal to 60 and less than or equal to 300
Changes Take Effect: Immediately

Specifies the read duration. This is how many seconds we read from a carrier message file until we stop reading that file. This prevents MCP from reading carrier message files if they span far too long. Very long carrier message files can impact matching accuracy.

This parameter does not govern HTTP request timeouts.

A maximum read duration of 300 seconds for optimal matching accuracy and performance is enforced. A minimum of 60 seconds is required.

Do not rely on this parameter to precisely measure how long to read each file. Always ensure that the read duration is longer than your longest carrier message file.

After changing this parameter, the coefficients for all the carrier message will be recalculated again. Please give it some time to recalculate before attempting to make a call after changing this parameter.

cpa.cm_enable_initial_tone_filter

Default Value: true
Valid Values: Choose between: true or false
Changes Take Effect: At start/restart

Tone filter for suppressing generic single tone at the beginning of the incoming frames for carrier message detection. The suppression starts when 10 consecutive frames (400msec) of a given frequency is detected. There is no limit for tone duration, so both continuous and intermittent tone (like ring tones) are suppressed.

cpa.cm_initial_silence_suppression_level

Default Value: 64
Valid Values: A valid suppression level must be in the range 0-100 inclusive.
Changes Take Effect: At start/restart

Specifies the initial level needed for silence suppression in carrier messages detection. Common range for this parameter is 40 to 100. Increasing the value of this parameter results in high energy noise getting suppressed.

>> Back to Top

cpa.cm_match_percent

Default Value: 80
Valid Values: A valid matching percent should be in the range 40-80 inclusive.
Changes Take Effect: Immediately/session

Specifies matching percent for carrier messages detection. Common range for this parameter is 70 to 80. Increasing this value, the matching is favoured; decreasing the value, the matching is restricted.

>> Back to Top

cpa.custom1

Default Value: none
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's custom 1 pattern. Default tone value is "Disable" which disables custom 1 detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

>> Back to Top

cpa.custom2

Default Value: none
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's custom 2 pattern. Default tone value is "Disable" which disables custom 2 detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

cpa.custom3

Default Value: none
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's custom 3 pattern. Default tone value is "Disable" which disables custom 3 detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

cpa.custom4

Default Value: none
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's custom 4 pattern. Default tone value is "Disable" which disables custom 4 detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

cpa.enable_alternate_signals

Default Value: false
Valid Values: Choose between: true or false
Changes Take Effect: At start/restart

Enable CPA alternate signals checking. This parameter should be turned on when we have SIT tone coming right after ring tone, without any silence or with very small silence (less than 100 msec). This parameter also good for cases when voice is coming right after SIT tone with small silence in between.

cpa.enable_carrier_messages

Default Value: false
Valid Values: Choose between: true or false
Changes Take Effect: Immediately/session

Enables/ disables carrier messages detection.

cpa.enable_log_param

Default Value: false
Valid Values: Choose between: true or false
Changes Take Effect: At start/restart

Enable CPA configuration parameters logging

cpa.enable_log_result

Default Value: true
Valid Values: Choose between: true or false
Changes Take Effect: At start/restart

Enable CPA result logging

cpa.fastbusy

Default Value: na_fastbusy
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's fast busy pattern. Default tone value is builtin North American fast busy (NA fast busy). Value "Disable" disables fast busy detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

The builtin North American fast busy has three segments. All segments are the same and are defined as follows:
Segment 1: f1=460-500(Hz), f2=600-640(Hz), ontime=160-320(ms), offtime=180-320(ms)
Segment 2: f1=460-500(Hz), f2=600-640(Hz), ontime=160-320(ms), offtime=180-320(ms)
Segment 3: f1=460-500(Hz), f2=600-640(Hz), ontime=160-320(ms), offtime=180-320(ms)

cpa.fax

Default Value: standard_fax
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's fax pattern. Default tone value is builtin standard fax (Standard fax). Value "Disable" disables fax detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

The builtin standard fax has only one segment and is defined as follows:
Segment 1: f1=2090-2110(Hz), f2=0-0(Hz), ontime=cpa.faxdur(ms), offtime=n/a

cpa.faxdur

Default Value: 160
Valid Values: A valid fax duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies fax duration (msec) for CPA. Fax signal must be greater than or equal to this length to be detected.

cpa.keptdur_before_statechange

Default Value: 0
Valid Values: A valid duration kept upon state modification must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum most recent buffer duration (msec) to be kept when CPA is requested to move from one state to another.

cpa.maxbeepdettime

Default Value: 30000
Valid Values: A valid beepdetect time must be a numeric not less than 20
Changes Take Effect: At start/restart

Specifies maximum duration (msec) for CPA beep detection state before a timeout event is thrown.

cpa.maxpostconntime

Default Value: 20000
Valid Values: A valid postconnect time must be a numeric not less than 20
Changes Take Effect: At start/restart

Specifies maximum duration (msec) for CPA postconnect state before a timeout event is thrown.

cpa.maxpreconntime

Default Value: 30000
Valid Values: A valid preconnect time must be a numeric not less than 20
Changes Take Effect: At start/restart

Specifies maximum duration (msec) for CPA preconnect state before a timeout event is thrown.

cpa.maxrings

Default Value: 0
Valid Values: A valid maximum ringback must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum ringbacks before a no answer event is thrown by CPA. Zero disables ringback count.

cpa.mintime_after_tone

Default Value: 200
Valid Values: A valid integer number between 40 and 400, inclusive of both numbers.
Changes Take Effect: At start/restart

Specifies the minimum idle time to wait (in milli seconds) following a detected tone signal before confirming it. In some situations, a SIT tone could almost immediately be followed by a connect event, ending the preconnect state and the SIT detection. In such situations, a shorter value for this parameter, say 100 ms, may be needed to wrap up SIT detection before the connect event.

cpa.nframes_cm_detection

Default Value: 50
Valid Values: A valid number of frames must be in the range 25-100 inclusive.
Changes Take Effect: At start/restart

Specifies the number of frames needed for carrier messages detection. Increasing the number of frames improves precision of the matching process but delays time to result.

>> Back to Top

cpa.no_ring_result

Default Value: 0
Valid Values: Choose between: 0, 2, 15, 16, 17 or 18.
Changes Take Effect: Immediately/session

This parameter specifies CPA result if no ringback is detected before postconnect event arrives. Default is none, that is, no CPA result returned. 0 - none 2 - busy 15 - human 16 - machine 17 - fax 18 - no_media

>> Back to Top

cpa.postconnectresult.machine.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result machine in postconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.postconnectresult.machine.list = 0 10 It means that if incoming message matches with carriermsg.0 or carriermsg.10, then CPA outputs result = machine

>> Back to Top

cpa.postconnsilduration

Default Value: 1000
Valid Values: A valid value must be in the range 200-3000 inclusive
Changes Take Effect: Immediately/session

Specifies the duration of silence that may trigger a configurable CPD result immediately after the postconnent event arrives. The default value is 1000 milliseconds. The range is between 200 and 3000 milliseconds. This parameter does not take effect until mpc.cpa.postconnsilresult is set to a value other than none (0).

cpa.postconnsilresult

Default Value: 0
Valid Values: Choose between: 0, 2, 15, 16, 17 or 18.
Changes Take Effect: Immediately/session

Specifies CPA result if a period of silence, which is configurable through mpc.cpa.postconnsilduration, is detected immediately after postconnect event arrives. The default option is none (0), that is, no CPA result returned. 0 - none 2 - busy 15 - human 16 - machine 17 - fax 18 - no_media

cpa.preconn_tones_det_mode

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Specifies the mode for the detection of preconnect tones. By default, the preconnect tones will only be detected in preconnect state. Optionally, the detection can be configured to operate in preconnect and postconnect state. 0 - Preconnect state only 1 - Preconnect and postconnect state NOTE: This option only applies to tones, carrier message detection (Voiceprint) during preconnect phase is not affected, that is, by setting this option to 1 (Preconnect and postconnect state) will not make MCP detect carrier messages at postconnect phase.

cpa.preconnectresult.busy.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result busy in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.busy.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result = busy.

cpa.preconnectresult.custom1.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.

Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result custom1 in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.custom1.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result = sit.custom1.

cpa.preconnectresult.custom2.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result custom2 in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.custom2.list = 1 3 5 It means that if incoming message matches with carriermsg.1, carriermsg.3 or carriermsg.5 then CPA outputs result = sit.custom2.

cpa.preconnectresult.custom3.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result custom3 in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.custom3.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result = sit.custom3.

cpa.preconnectresult.custom4.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result custom4 in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.custom4.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result = sit.custom4.

cpa.preconnectresult.fast_busy.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result fast busy in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.fast_busy.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result = busy.

cpa.preconnectresult.sit_nocircuit.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result sit_nocircuit in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.sit_nocircuit.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result = sit.nocircuit.

cpa.preconnectresult.sit_operatorintercept.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result sit_operatorintercept in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.sit_operatorintercept.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result = sit.operatorintercept.

cpa.preconnectresult.sit_reorder.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.

Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result sit_reorder in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.sit_reorder.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result defined = sit.reorder.

cpa.preconnectresult.sit_vacantcircuit.list

Default Value:
Valid Values: Can be an empty string or a space separated list of numbers ranging from 0 to 100.
Changes Take Effect: Immediately/session

Specifies list of carrier message files associated with result sit_vacantcircuit in preconnect mode. The list is a sequence of integer number equal to the index of carrier message previously specified. Example: cpa.preconnectresult.sit_vacantcircuit.list = 0 2 4 It means that if incoming message matches with carriermsg.0, carriermsg.2 or carriermsg.4 then CPA outputs result = sit.vacantcircuit.

cpa.priority_machine_machinegreetingdur

Default Value: 1500
Valid Values: A valid machine-favored machine greeting duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies machine-favored machine greeting duration (msec) for CPA.

cpa.priority_machine_maxvoicesigdur

Default Value: 600
Valid Values: A valid machine-favored maximum voice signal duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies machine-favored maximum voice signal duration (msec) for CPA.

cpa.priority_machine_voicepausedur

Default Value: 1100

Valid Values: A valid machine-favored voice pause duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies machine-favored voice pause duration (msec) for CPA.

cpa.priority_normal_machinegreetingdur

Default Value: 1800
Valid Values: A valid normal machine greeting duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies normal machine greeting duration (msec) for CPA.

cpa.priority_normal_maxvoicesigdur

Default Value: 800
Valid Values: A valid normal maximum voice signal duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies normal maximum voice signal duration (msec) for CPA.

cpa.priority_normal_voicepausedur

Default Value: 1000
Valid Values: A valid normal voice pause duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies normal voice pause duration (msec) for CPA.

cpa.priority_voice_machinegreetingdur

Default Value: 2000
Valid Values: A valid voice-favored machine greeting duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies voice-favored machine greeting duration (msec) for CPA.

cpa.priority_voice_maxvoicesigdur

Default Value: 1100
Valid Values: A valid voice-favored maximum voice signal duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies voice-favored maximum voice signal duration (msec) for CPA.

cpa.priority_voice_voicepausedur

Default Value: 850
Valid Values: A valid voice-favored voice pause duration must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies voice-favored voice pause duration (msec) for CPA.

cpa.ringback

Default Value: na_ringback
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Each of the result pattern types ringback, busy, fax, sit nocircuit, sit reorder, sit operatorintercept, sit vacantcircuit, custom1, custom2, custom3, and custom4, has a configuration parameter that sets the list of one or more tone definition names that are mapped to the type. Some pattern types have default tone settings. Specifying "Disable" for a pattern type will disable detection of the type.

The list of tone definition names consist of a set of builtin standard tones and a number of configured custom tones. The builtin tones include NA ringback (North American ringback), NA busy, NA fast busy, Standard fax, NA SIT no circuit, NA SIT vacant circuit, NA SIT operator intercept, NA SIT reorder. Configuration for up to 10 custom tones, named Custom tone 1 through Custom tone 10, each containing one to three segments is supported.

The configuration for each segment includes frequency range for one or two frequencies, on time range, off time range. Specifying zero to min and max for f1 and f2 implicitly disables the segment. Specifying f2 values implicitly enables a second frequency for the segment. Specifying segment2 values implicitly enables a second segment for the tone and so on for segment3. Any not set tone parameters will default to null/disabled values.

In this case, cpa.ringback then specifies tones for CPA's ringback type. Default tone value is builtin

North American ringback (NA ringback). Value "Disable" disables ringback detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

The builtin North American ringback is composed of three different ringback tones. Each of them has only one segment and is defined as follows:
NA RINGBACK 1: f1=420-445(Hz), f2=475-500(Hz), ontime=720-1280(ms), offtime=1440-2560(ms)
NA RINGBACK 2: f1=420-445(Hz), f2=475-500(Hz), ontime=720-1280(ms), offtime=2160-3840(ms)
NA RINGBACK 3: f1=420-445(Hz), f2=475-500(Hz), ontime=1440-2560(ms), offtime=2880-5120(ms)

>> Back to Top

cpa.sit_nocircuit

Default Value: na_sit_nocircuit
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's SIT no circuit pattern. Default tone value is builtin North American SIT no circuit (NA SIT nocircuit). Value "Disable" disables SIT no circuit detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

The builtin North American sit no circuit has three segments. Each segment is defined as follows:
Segment 1: f1=950-1020(Hz), f2=0-0(Hz), ontime=320-440(ms), offtime=0-60(ms)
Segment 2: f1=1400-1450(Hz), f2=0-0(Hz), ontime=320-440(ms), offtime=0-60(ms)
Segment 3: f1=1740-1850(Hz), f2=0-0(Hz), ontime=320-440(ms), offtime=0-100(ms)

>> Back to Top

cpa.sit_operatorintercept

Default Value: na_sit_operatorintercept
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's SIT operator intercept pattern. Default tone value is builtin North American SIT operator intercept (NA SIT operator intercept). Value "Disable" disables SIT operator intercept detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

The builtin North American sit operator intercept has three segments. Each segment is defined as follows:
Segment 1: f1=874-955(Hz), f2=0-0(Hz), ontime=160-300(ms), offtime=0-60(ms)
Segment 2: f1=1310-1430(Hz), f2=0-0(Hz), ontime=160-300(ms), offtime=0-60(ms)
Segment 3: f1=1740-1850(Hz), f2=0-0(Hz), ontime=320-440(ms), offtime=0-100(ms)

>> Back to Top

cpa.sit_reorder

Default Value: na_sit_reorder
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's SIT reorder pattern. Default tone value is builtin North American SIT reorder (NA SIT reorder). Value "Disable" disables SIT reorder detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

The builtin North American sit reorder has three segments. Each segment is defined as follows:
Segment 1: f1=874-955(Hz), f2=0-0(Hz), ontime=160-300(ms), offtime=0-60(ms)
Segment 2: f1=1400-1450(Hz), f2=0-0(Hz), ontime=320-440(ms), offtime=0-60(ms)
Segment 3: f1=1740-1850(Hz), f2=0-0(Hz), ontime=320-440(ms), offtime=0-100(ms)

>> Back to Top

cpa.sit_vacantcircuit

Default Value: na_sit_vacantcircuit
Valid Values: Any combination of: "na_ringback", "na_busy", "na_fastbusy", "na_sit_nocircuit", "na_sit_vacantcircuit", "na_sit_operatorintercept", "na_sit_reorder", "standard_fax", "tone1", "tone2", "tone3", "tone4", "tone5", "tone6", "tone7", "tone8", "tone9" and "tone10". Or "none".
Changes Take Effect: At start/restart

Specifies tones for CPA's SIT vacant circuit pattern. Default tone value is builtin North American SIT vacant circuit (NA SIT vacantcircuit). Value "Disable" disables SIT vacant circuit detection. Valid tones include builtins and Custom tone 1 to Custom tone 10.

The builtin North American sit vacant circuit has three segments. Each segment is defined as follows:
Segment 1: f1=950-1020(Hz), f2=0-0(Hz), ontime=320-440(ms), offtime=0-60(ms)
Segment 2: f1=1310-1430(Hz), f2=0-0(Hz), ontime=160-300(ms), offtime=0-60(ms)
Segment 3: f1=1740-1850(Hz), f2=0-0(Hz), ontime=320-440(ms), offtime=0-100(ms)

>> Back to Top

cpa.tone1.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 1 (Hz) for CPA.

>> Back to Top

cpa.tone1.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 1 (Hz) for CPA.

>> Back to Top

cpa.tone1.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 1 (Hz) for CPA.

>> Back to Top

cpa.tone1.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 1 (Hz) for CPA.

>> Back to Top

cpa.tone1.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 1 (msec) for CPA.

>> Back to Top

cpa.tone1.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 1 (msec) for CPA.

cpa.tone1.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 1 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone1.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 1 (msec) for CPA.

cpa.tone1.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 1 (Hz) for CPA.

cpa.tone1.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 1 (Hz) for CPA.

cpa.tone1.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 1 (Hz) for CPA.

cpa.tone1.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 1 (Hz) for CPA.

cpa.tone1.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 1 (msec) for CPA.

cpa.tone1.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 1 (msec) for CPA.

cpa.tone1.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 1 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone1.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 1 (msec) for CPA.

cpa.tone1.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 1 (Hz) for CPA.

cpa.tone1.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 1 (Hz) for CPA.

cpa.tone1.segment3.f2max

Default Value: 0

Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 1 (Hz) for CPA.

cpa.tone1.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 1 (Hz) for CPA.

cpa.tone1.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 1 (msec) for CPA.

cpa.tone1.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 1 (msec) for CPA.

cpa.tone1.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 1 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime

will be matched.

cpa.tone1.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 1 (msec) for CPA.

cpa.tone10.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 10 (Hz) for CPA.

cpa.tone10.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 10 (Hz) for CPA.

cpa.tone10.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 10 (Hz) for CPA.

cpa.tone10.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 10 (Hz) for CPA.

>> Back to Top

cpa.tone10.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 10 (msec) for CPA.

>> Back to Top

cpa.tone10.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 10 (msec) for CPA.

>> Back to Top

cpa.tone10.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 10 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

>> Back to Top

cpa.tone10.segment1.ontimemin

Default Value: 20

Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 10 (msec) for CPA.

cpa.tone10.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 10 (Hz) for CPA.

cpa.tone10.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 10 (Hz) for CPA.

cpa.tone10.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 10 (Hz) for CPA.

cpa.tone10.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 10 (Hz) for CPA.

cpa.tone10.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 10 (msec) for CPA.

cpa.tone10.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 10 (msec) for CPA.

cpa.tone10.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 10 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone10.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 10 (msec) for CPA.

cpa.tone10.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 10 (Hz) for CPA.

>> Back to Top

cpa.tone10.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 10 (Hz) for CPA.

>> Back to Top

cpa.tone10.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 10 (Hz) for CPA.

>> Back to Top

cpa.tone10.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 10 (Hz) for CPA.

>> Back to Top

cpa.tone10.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 10 (msec) for CPA.

cpa.tone10.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 10 (msec) for CPA.

cpa.tone10.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 10 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone10.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 10 (msec) for CPA.

cpa.tone2.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 2 (Hz) for CPA.

cpa.tone2.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 2 (Hz) for CPA.

cpa.tone2.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 2 (Hz) for CPA.

cpa.tone2.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 2 (Hz) for CPA.

cpa.tone2.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 2 (msec) for CPA.

cpa.tone2.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 2 (msec) for CPA.

cpa.tone2.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 2 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone2.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 2 (msec) for CPA.

cpa.tone2.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 2 (Hz) for CPA.

cpa.tone2.segment2.f1min

Default Value: 0

Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 2 (Hz) for CPA.

cpa.tone2.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 2 (Hz) for CPA.

cpa.tone2.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 2 (Hz) for CPA.

cpa.tone2.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 2 (msec) for CPA.

cpa.tone2.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 2 (msec) for CPA.

cpa.tone2.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 2 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone2.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 2 (msec) for CPA.

cpa.tone2.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 2 (Hz) for CPA.

cpa.tone2.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 2 (Hz) for CPA.

cpa.tone2.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 2 (Hz) for CPA.

>> Back to Top

cpa.tone2.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 2 (Hz) for CPA.

>> Back to Top

cpa.tone2.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 2 (msec) for CPA.

>> Back to Top

cpa.tone2.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 2 (msec) for CPA.

>> Back to Top

cpa.tone2.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 2 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone2.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 2 (msec) for CPA.

cpa.tone3.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 3 (Hz) for CPA.

cpa.tone3.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 3 (Hz) for CPA.

cpa.tone3.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 3 (Hz) for CPA.

cpa.tone3.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 3 (Hz) for CPA.

cpa.tone3.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 3 (msec) for CPA.

cpa.tone3.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 3 (msec) for CPA.

cpa.tone3.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 3 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone3.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 3 (msec) for CPA.

>> Back to Top

cpa.tone3.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 3 (Hz) for CPA.

>> Back to Top

cpa.tone3.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 3 (Hz) for CPA.

>> Back to Top

cpa.tone3.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 3 (Hz) for CPA.

>> Back to Top

cpa.tone3.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 3 (Hz) for CPA.

cpa.tone3.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 3 (msec) for CPA.

cpa.tone3.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 3 (msec) for CPA.

cpa.tone3.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 3 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone3.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 3 (msec) for CPA.

cpa.tone3.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 3 (Hz) for CPA.

cpa.tone3.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 3 (Hz) for CPA.

cpa.tone3.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 3 (Hz) for CPA.

cpa.tone3.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 3 (Hz) for CPA.

cpa.tone3.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 3 (msec) for CPA.

cpa.tone3.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 3 (msec) for CPA.

cpa.tone3.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 3 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone3.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 3 (msec) for CPA.

cpa.tone4.segment1.f1max

Default Value: 0

Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 4 (Hz) for CPA.

>> Back to Top

cpa.tone4.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 4 (Hz) for CPA.

>> Back to Top

cpa.tone4.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 4 (Hz) for CPA.

>> Back to Top

cpa.tone4.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 4 (Hz) for CPA.

>> Back to Top

cpa.tone4.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 4 (msec) for CPA.

cpa.tone4.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 4 (msec) for CPA.

cpa.tone4.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 4 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone4.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 4 (msec) for CPA.

cpa.tone4.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 4 (Hz) for CPA.

cpa.tone4.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 4 (Hz) for CPA.

>> Back to Top

cpa.tone4.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 4 (Hz) for CPA.

>> Back to Top

cpa.tone4.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 4 (Hz) for CPA.

>> Back to Top

cpa.tone4.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 4 (msec) for CPA.

>> Back to Top

cpa.tone4.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 4 (msec) for CPA.

cpa.tone4.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 4 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone4.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 4 (msec) for CPA.

cpa.tone4.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 4 (Hz) for CPA.

cpa.tone4.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 4 (Hz) for CPA.

cpa.tone4.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 4 (Hz) for CPA.

cpa.tone4.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 4 (Hz) for CPA.

cpa.tone4.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 4 (msec) for CPA.

cpa.tone4.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 4 (msec) for CPA.

cpa.tone4.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 4 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone4.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 4 (msec) for CPA.

cpa.tone5.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 5 (Hz) for CPA.

cpa.tone5.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 5 (Hz) for CPA.

cpa.tone5.segment1.f2max

Default Value: 0

Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 5 (Hz) for CPA.

cpa.tone5.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 5 (Hz) for CPA.

cpa.tone5.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 5 (msec) for CPA.

cpa.tone5.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 5 (msec) for CPA.

cpa.tone5.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 5 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime

will be matched.

cpa.tone5.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 5 (msec) for CPA.

cpa.tone5.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 5 (Hz) for CPA.

cpa.tone5.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 5 (Hz) for CPA.

cpa.tone5.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 5 (Hz) for CPA.

cpa.tone5.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 5 (Hz) for CPA.

cpa.tone5.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 5 (msec) for CPA.

cpa.tone5.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 5 (msec) for CPA.

cpa.tone5.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 5 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone5.segment2.ontimemin

Default Value: 20

Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 5 (msec) for CPA.

cpa.tone5.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 5 (Hz) for CPA.

cpa.tone5.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 5 (Hz) for CPA.

cpa.tone5.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 5 (Hz) for CPA.

cpa.tone5.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 5 (Hz) for CPA.

cpa.tone5.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 5 (msec) for CPA.

cpa.tone5.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 5 (msec) for CPA.

cpa.tone5.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 5 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone5.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 5 (msec) for CPA.

cpa.tone6.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 6 (Hz) for CPA.

>> Back to Top

cpa.tone6.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 6 (Hz) for CPA.

>> Back to Top

cpa.tone6.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 6 (Hz) for CPA.

>> Back to Top

cpa.tone6.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 6 (Hz) for CPA.

>> Back to Top

cpa.tone6.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 6 (msec) for CPA.

cpa.tone6.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 6 (msec) for CPA.

cpa.tone6.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 6 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone6.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 6 (msec) for CPA.

cpa.tone6.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 6 (Hz) for CPA.

cpa.tone6.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 6 (Hz) for CPA.

cpa.tone6.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 6 (Hz) for CPA.

cpa.tone6.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 6 (Hz) for CPA.

cpa.tone6.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 6 (msec) for CPA.

cpa.tone6.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 6 (msec) for CPA.

>> Back to Top

cpa.tone6.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 6 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

>> Back to Top

cpa.tone6.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 6 (msec) for CPA.

>> Back to Top

cpa.tone6.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 6 (Hz) for CPA.

>> Back to Top

cpa.tone6.segment3.f1min

Default Value: 0

Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 6 (Hz) for CPA.

cpa.tone6.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 6 (Hz) for CPA.

cpa.tone6.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 6 (Hz) for CPA.

cpa.tone6.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 6 (msec) for CPA.

cpa.tone6.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 6 (msec) for CPA.

cpa.tone6.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 6 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone6.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 6 (msec) for CPA.

cpa.tone7.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 7 (Hz) for CPA.

cpa.tone7.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 7 (Hz) for CPA.

cpa.tone7.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 7 (Hz) for CPA.

cpa.tone7.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 7 (Hz) for CPA.

cpa.tone7.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 7 (msec) for CPA.

cpa.tone7.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 7 (msec) for CPA.

cpa.tone7.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 7 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone7.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 7 (msec) for CPA.

cpa.tone7.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 7 (Hz) for CPA.

cpa.tone7.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 7 (Hz) for CPA.

cpa.tone7.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 7 (Hz) for CPA.

cpa.tone7.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 7 (Hz) for CPA.

cpa.tone7.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 7 (msec) for CPA.

cpa.tone7.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 7 (msec) for CPA.

cpa.tone7.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 7 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone7.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 7 (msec) for CPA.

cpa.tone7.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 7 (Hz) for CPA.

cpa.tone7.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 7 (Hz) for CPA.

cpa.tone7.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 7 (Hz) for CPA.

cpa.tone7.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 7 (Hz) for CPA.

cpa.tone7.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 7 (msec) for CPA.

cpa.tone7.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 7 (msec) for CPA.

cpa.tone7.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 7 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone7.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 7 (msec) for CPA.

cpa.tone8.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 8 (Hz) for CPA.

cpa.tone8.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 8 (Hz) for CPA.

cpa.tone8.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 8 (Hz) for CPA.

cpa.tone8.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 8 (Hz) for CPA.

cpa.tone8.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 8 (msec) for CPA.

>> Back to Top

cpa.tone8.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 8 (msec) for CPA.

>> Back to Top

cpa.tone8.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 8 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

>> Back to Top

cpa.tone8.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 8 (msec) for CPA.

>> Back to Top

cpa.tone8.segment2.f1max

Default Value: 0

Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 8 (Hz) for CPA.

cpa.tone8.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 8 (Hz) for CPA.

cpa.tone8.segment2.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 8 (Hz) for CPA.

cpa.tone8.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 8 (Hz) for CPA.

cpa.tone8.segment2.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 8 (msec) for CPA.

cpa.tone8.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 8 (msec) for CPA.

cpa.tone8.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 8 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone8.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 8 (msec) for CPA.

cpa.tone8.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 8 (Hz) for CPA.

cpa.tone8.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 8 (Hz) for CPA.

>> Back to Top

cpa.tone8.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 8 (Hz) for CPA.

>> Back to Top

cpa.tone8.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 8 (Hz) for CPA.

>> Back to Top

cpa.tone8.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 8 (msec) for CPA.

>> Back to Top

cpa.tone8.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 8 (msec) for CPA.

cpa.tone8.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 8 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone8.segment3.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 8 (msec) for CPA.

cpa.tone9.segment1.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 1 of tone 9 (Hz) for CPA.

cpa.tone9.segment1.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 1 of tone 9 (Hz) for CPA.

cpa.tone9.segment1.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 1 of tone 9 (Hz) for CPA.

cpa.tone9.segment1.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 1 of tone 9 (Hz) for CPA.

cpa.tone9.segment1.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 1 of tone 9 (msec) for CPA.

cpa.tone9.segment1.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 1 of tone 9 (msec) for CPA.

cpa.tone9.segment1.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 1 of tone 9 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

cpa.tone9.segment1.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 1 of tone 9 (msec) for CPA.

cpa.tone9.segment2.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 2 of tone 9 (Hz) for CPA.

cpa.tone9.segment2.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 2 of tone 9 (Hz) for CPA.

cpa.tone9.segment2.f2max

Default Value: 0

Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 2 of tone 9 (Hz) for CPA.

cpa.tone9.segment2.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 2 of tone 9 (Hz) for CPA.

cpa.tone9.segment2.offtimemax

Default Value: 20
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 2 of tone 9 (msec) for CPA.

cpa.tone9.segment2.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 2 of tone 9 (msec) for CPA.

cpa.tone9.segment2.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 2 of tone 9 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime

will be matched.

cpa.tone9.segment2.ontimemin

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 2 of tone 9 (msec) for CPA.

cpa.tone9.segment3.f1max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 1 of segment 3 of tone 9 (Hz) for CPA.

cpa.tone9.segment3.f1min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 1 of segment 3 of tone 9 (Hz) for CPA.

cpa.tone9.segment3.f2max

Default Value: 0
Valid Values: A valid maximum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum frequency 2 of segment 3 of tone 9 (Hz) for CPA.

cpa.tone9.segment3.f2min

Default Value: 0
Valid Values: A valid minimum frequency must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum frequency 2 of segment 3 of tone 9 (Hz) for CPA.

>> Back to Top

cpa.tone9.segment3.offtimemax

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies maximum offtime of segment 3 of tone 9 (msec) for CPA.

>> Back to Top

cpa.tone9.segment3.offtimemin

Default Value: 0
Valid Values: A valid offtime must be a non-negative numeric
Changes Take Effect: At start/restart

Specifies minimum offtime of segment 3 of tone 9 (msec) for CPA.

>> Back to Top

cpa.tone9.segment3.ontimemax

Default Value: 20
Valid Values: A valid ontimemin must be a greater than or equal to 0 numeric
Changes Take Effect: At start/restart

Specifies maximum ontime of segment 3 of tone 9 (msec) for CPA. Setting the value to zero disables checking the ontime against a maximum value, so that any time greater than the minimum ontime will be matched.

>> Back to Top

cpa.tone9.segment3.ontimemin

Default Value: 20

Valid Values: A valid ontimemin must be a greater than or equal to 20 numeric
Changes Take Effect: At start/restart

Specifies minimum ontime of segment 3 of tone 9 (msec) for CPA.

cpa.voice_level_db

Default Value: 17.5
Valid Values: A valid voice level must be a number greater than or equal to one
Changes Take Effect: At start/restart

Specifies the active voice signal level (in dB relative to the maximum) for CPA. By default, this value is set to 70% of the default value of Voice range (dB); that is 17.5 dB. The valid range is 1 dB to Voice range (dB) inclusive. If the value is greater than Voice range (dB), it will be set to Voice range (dB).

cpa.voice_range_db

Default Value: 25
Valid Values: A valid voice range must be an integer greater than zero
Changes Take Effect: At start/restart

Specifies the minimum signal to noise ratio (or dynamic range) for the CPA signal (in dB). If the difference between the minimum and maximum signal level is less than this threshold, the entire signal is considered to be noise.

ctrleventpoollowthreshold

Default Value: 50
Valid Values: A valid value for the ctrleventpoollowthreshold is greater than or equal to 0 and less than or equal to 100.
Changes Take Effect: At start/restart

Once the mpc.ctrleventpoolthreshold value is reached for an individual event pool, the number of used events in that pool must drop below the low threshold value before calls can be accepted. Only enabled if mpc.ctrleventpoolthreshold is non-zero. The value is in percentage (%).

ctrleventpoolthreshold

Default Value: 75
Valid Values: A valid value for the ctrleventpoolthreshold is greater than or equal to 0 and less than or equal to 100.
Changes Take Effect: At start/restart

Specified a threshold for the percentage (%) of events in an individual media event pool can be reached before the MCP will start rejecting calls. Once reached, the number of used events must drop below the percentage specified in mpc.ctrleventpoollowthreshold before calls will be accepted again. When set to 0, this functionality is disabled.

>> Back to Top

default_audio_format

Default Value: ULAW
Valid Values: Choose between: ULAW or ALAW
Changes Take Effect: At start/restart

Default audio format used by the MCP

>> Back to Top

dsp.g726littleendian

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

This parameter specifies whether input/output of the G.726 data is in big-endian or in little-endian order. In addition to determining the order of the generated G.726 data, the incoming G.726 data order is assumed to be in the order specified by this parameter. 0 - Big-Endian 1 - Little-Endian

>> Back to Top

dsp.g729a

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately

Specifies whether to use G.729 Annex A for G.729 transcoding.

>> Back to Top

dtmf.detectedge

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Specifies whether a DTMF will be recognized on the "rising" edge (1st RFC2833 packet), or on the "falling" edge (when any of the following event happens: 1st RFC2833 packet with "E" bit set, upon receipt of a RTP packet that is not for the same DTMF event, or after the maximum silence gap timeout.

>> Back to Top

dtmf.duration

Default Value: 200
Valid Values: Possible values are integers from 10 to 1000 inclusive
Changes Take Effect: At start/restart

Specifies the duration (in milliseconds) of outgoing RFC2833 DTMF packets. Default value is 200.

>> Back to Top

dtmf.gap

Default Value: 100
Valid Values: Possible values are integers from 10 to 1000 inclusive.
Changes Take Effect: At start/restart

Specifies the gap between an outgoing RFC2833 DTMF packet and the following outgoing packet in milliseconds.

>> Back to Top

dtmf.inband_amplitude

Default Value: 15000
Valid Values: Possible values are integers from 0 to 32,767 inclusive.
Changes Take Effect: At start/restart

Specifies the amplitude for inband dtmf generator. The higher the the value, the greater the output amplitude in terms of dB. The default value of 15000 gives approximately -20dB while 3000 gives approximately -35dB for example.

>> Back to Top

dtmf.maxsilence

Default Value: 20
Valid Values: mpc.dtmf.maxsilence must be an integer that is at least equal to 0 and less than or equal to 120.
Changes Take Effect: At start/restart

The maximum silence permitted between same inband DTMF tones in ms to be considered no longer part of the same DTMF.

dtmf.minduration

Default Value: 0
Valid Values: mpc.dtmf.minduration must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the minimum duration of DTMF events in ms (either inband or RFC2833) required before a DTMF is detected by the dialog. Must be greater than or equal to 0ms.

dtmf.multidtmfonetimestamp

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Specifies whether the RFC2833 packets of multiple DTMFs will have the same timestamp. If it is true, the End bit is used to differentiate the DTMFs.

dtmf.pauseduration

Default Value: 200
Valid Values: Possible values are integers from 10 to 1000 inclusive
Changes Take Effect: At start/restart

Specifies the duration (in milliseconds) for the duration of a pause ('p') DTMF.

dtmf.singlepacket

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

When set to Disable, outgoing DTMF are represented by multiple (depends on mpc.dtmf.duration) RFC2833 packets followed by 3 RFC2833 packets with the End bit set. When set to Enable, outgoing DTMF are represented by a single RFC2833 packet with the End bit set.

fcr.defaultdtmfhandling

Default Value: as-is
Valid Values: as-is: (default) Record everything as-is from the RTP stream. Inband DTMFs will be recorded, but RFC2833 digits will not no-digits: Strip out all DTMF digits. This includes inband or RFC2833. NOTE: When telephone-event is negotiated on the call, if inband audio DTMFs are received, they will not be removed from the recording. all-digits: Record all DTMF digits, including inband, and generate audio for RFC2833 digits
Changes Take Effect: Immediately/session

Specifies the recording behavior for DTMFs in a Full Call Recording.

fcr.gain

Default Value: 0
Valid Values: The number must be an integer from -30 to 30 inclusive
Changes Take Effect: At start/restart

Gain (in dB from -30 to 30) applied to audio used in Full Call Recording (both input from the caller and output to the caller) from call participants.

font_paths_linux

Default Value: /usr/share/fonts/default/ghostscript
Valid Values: Please specify a valid '|' separated list of font folder paths.
Changes Take Effect: At start/restart

List of paths of font directories on a Linux MCP system, separated by the delimiter '|'. This information is used by the Video Text Overlay feature. All the font files are scanned at start-up, and their font name to file name mapping information is cached for fast look-up later. An empty value will disable

the feature. The special value "default" can be used for the builtin default value.

font_paths_win

Default Value: C:/Windows/Fonts
Valid Values: Please specify a valid '|' separated list of font folder paths.
Changes Take Effect: At start/restart

List of paths of font directories on a Windows MCP system, separated by the delimiter '|'. This information is used by the Video Text Overlay feature. All the font files are scanned at start-up, and their font name to file name mapping information is cached for fast look-up later. An empty value will disable the feature. The special value "default" can be used for the builtin default value.

g722.maxptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

g722.ptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will take precedence.

g726_32.maxptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40 or 60.
Changes Take Effect: Immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

g726_32.ptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40 or 60.
Changes Take Effect: Immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

g729.fmtp

Default Value:
Valid Values: Can be an empty string or choose between: "annexb=yes" or "annexb=no"
Changes Take Effect: At start/restart

Specifies the default G729 fmtp option string used by the platform when initiating SDP offer or answering SDP offer. When initiating SDP offer, if No fmtp is selected, fmtp attribute line for G729 will not be present in SDP offered by the platform. If annexb enabled is selected, fmtp attribute line for G729 will be present as "annexb=yes". If annexb disabled is selected, fmtp attribute line for G729 will be shown as "annexb=no". When answering to an SDP offer with a valid G729 fmtp line, this parameter has no effect and the platform will reply to the offer with the same fmtp line. When answering to an SDP offer without an fmtp line or with an invalid G729 fmtp line, the platform will reply to the offer using this configuration value with the same rule as initiating SDP offer.

>> Back to Top

g729.maxptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

g729.ptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

gsm.maxptime

Default Value: 0
Valid Values: Choose between: 0, 20, 40, 60, 80 or 100.
Changes Take Effect: immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

gsm.ptime

Default Value: 0

Valid Values: Choose between: 0, 20, 40, 60, 80 or 100.
Changes Take Effect: immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will take precedence.

h263_1998payload

Default Value: 99
Valid Values: A valid H263-1998 payload can only be an integer from 96 to 127 inclusive
Changes Take Effect: At start/restart

Default payload type number to use for the H263-1998 codec

h263.fmtp

Default Value:
Valid Values: <token>[;<token>...][|<token>[;<token>...]...]
Changes Take Effect: At start/restart

Specifies the SDP fmtp line offered and accepted by MCP for H263 and H263-1998. The value is in the format: <token>[;<token>...][|<token>[;<token>...]...] Note that the list of tokens are grouped by separating them by a vertical bar - each group of tokens represent a separate SDP fmtp line. When more than 1 fmtp's are specified, H263-1998 negotiation will use all of the fmtp's as the local fmtp (for example offer generation will generate 3 payloads for 3 fmtp's). For H263, only the first fmtp will be used as the local fmtp.

Following tokens are accepted: profile=<profile> level=<level> SQCIF=<mpi> QCIF=<mpi> CIF=<mpi> CIF4=<mpi> CIF16=<mpi> CUSTOM=<max width>,<max height>,<mpi> F=1 I=1 J=1 T=1 K=1 N=1 P=1

profile - Profile from 0 to 8 can be specified. level - Level from 10 to 70 are supported. SQCIF - Specifies 128 x 96 video resolution with fps @ 30000 / (1001 * mpi) QCIF - Specifies 176 x 144 video resolution with fps @ 30000 / (1001 * mpi) CIF - Specifies 352 x 288 video resolution with fps @ 30000 / (1001 * mpi) CIF4 - Specifies 704 x 480 video resolution with fps @ 30000 / (1001 * mpi) CIF16 - Specifies 1408 x 1152 video resolution with fps @ 30000 / (1001 * mpi) CUSTOM - Specifies custom video resolution with fps @ 30000 / (1001 * mpi) F=1 - Specifies that annex F (from ITU-T Recommendation H.263, January 2005) is supported. I=1 - Specifies that annex I (from ITU-T Recommendation H.263, January 2005) is supported. J=1 - Specifies that annex J (from ITU-T Recommendation H.263, January 2005) is supported. T=1 - Specifies that annex T (from ITU-T

Recommendation H.263, January 2005) is supported. K=1 - Specifies that annex K (from ITU-T Recommendation H.263, January 2005) is supported. Not supported when H263 transcoding is enabled. N=1 - Specifies that annex N (from ITU-T Recommendation H.263, January 2005) is supported. Not supported when H263 transcoding is enabled. P=1 - Specifies that annex P (from ITU-T Recommendation H.263, January 2005) is supported.

Please see RFC4629 for more details.

If H263 transcoding is enabled for the session, the fmtp containing the following tokens will not be offered and also not accepted by MCP: - profile value other than 0 - K=1 - N=1

If H263 transcoding is enabled and h263.fmtp is not specified, the following fmtp value will be set by default: profile=0;level=70|profile=0;level=70;F=1;I=1;J=1;T=1;P=1 If H263 transcoding is disabled and h263.fmtp is not specified, no fmtp will be generated in the offer and all fmtp's will be accepted by MCP. If H263 transcoding is disabled and h263.fmtp is not specified, fmtp negotiation will be done without the transcoding restrictions imposed on the allowed tokens.

>> Back to Top

h264.fmtp

Default Value: profile=b; level=3.1; packetization-mode=*;|profile=cb; level=3.1; packetization-mode=*;|profile=m; level=3.1; packetization-mode=*;
Valid Values: Please check the parameter description.
Changes Take Effect: At start/restart

Specifies the H264 SDP profile, level and packetization mode offered and accepted by the MCP. Set to one or more fmtp text values separated by the '|' character. One fmtp text is in the form of "profile=X; level=Y; packetization-mode=Z;".

X is an element of the set {*, b, cb, m, e, h, h10, h10i, h42, h42i, h44, h44i, c44i} specifying the H264 profile offered and accepted by the MCP. In the set, * is used as the wildcard to allow the MCP to offer and accept any valid profile. The rest of the set are profiles defined by H264 whose full name are (in respective order as the aforementioned set): {baseline, constrained baseline, main, extended, high, high 10, high 10 intra, high 4:2:2, high 4:2:2 intra, high 4:4:4, high 4:4:4 intra, cavlc 4:4:4 intra}. Invalid profile value will be replaced with * (wildcard).

Y is an element of the set {*, 1, 1b, 1.1, 1.2, 1.3, 2, 2.1, 2.2, 3, 3.1, 3.2, 4, 4.1, 4.2, 5, 5.1} specifying the H264 level of the corresponding profiles offered and accepted by the MCP. In the set, * is used as the wildcard to allow MCP to offer and accept any valid level for its corresponding profile. The rest of the set are levels defined by H264. Invalid level value will be replaced with * (wildcard).

Z is an element of the set {*, 0, 1} specifying the H264 packetization mode offered and accepted by the MCP. Similar to the others, * is used as the wildcard to allow the MCP to offer and accept any valid packetization mode. 0 refers to single NALU packetization mode while 1 demands non-interleaved packetization capability. The MCP does not support interleaved packetization mode, as well as, any value other than stated. Invalid packetization-mode value will be replaced with 0 (single NALU).

During SDP negotiation, each fmtp text value without wildcard will be translated to one H264 media fmtp line while those with wildcard will be translated to one or more H264 media fmtp line equivalent to the wildcard.
For example, profile=b; level=1.1; packetization-mode=* which is the first part of the default will be

translated to two H264 media fmtp lines of profile-level-id=42000B; packetization-mode=0; and profile-level-id=42000B; packetization-mode=1;. Note that profile "b" is equivalent to 66 in decimal or 42 in hexadecimal and level "1.1" is equivalent to level 11 in decimal or level 0B in hexadecimal. According to RFC3984, the use of profile-level-id and packetization-mode during capability exchange must be negotiated symmetrically except the level part can be downgraded. For example, the MCP is configured with h264.fmtp="profile=b; level=1; packetization-mode=1;" and it receives an offer with H264 media fmtp line of profile-level-id=420033; packetization-mode=1; which is equivalent to h264.fmtp="profile=b; level=5.1; packetization-mode=1". In this case, the profile and packetization-mode of the platform and the offer are symmetric but the level parts are not. This offer will be accepted by the MCP with level downgrade in its response, e.g. "profile-level-id=42000A; packetization-mode=1". The offerer when receiving the response will know that the answerer has accepted the offer but the level must be downgraded and the offerer will have to produce the H264 content accordingly when proceeding with this response.

Another example, the MCP is configured with h264.fmtp="profile=b; level=5.1; packetization-mode=1" and it receives an offer with H264 media fmtp line of profile-level-id-42000A; packetization-mode=1; which is equivalent to h264.fmtp="profile=b; level=1; packetization-mode=1". In this case, the profile and packetization-mode of the platform and the offer are symmetric but the level parts are not. This offer will still be accepted by the MCP without level downgrade in its response, e.g. "profile-level-id=42000A; packetization-mode=1;". The MCP accepts the offer because it is configured with a higher level which is capable of processing any lower level. The offerer will receive the response that the MCP has accepted the offer.

H264 transcoder does not support extended and advanced 4:4:4 profiles. Consequently, if H264 transcoding is enabled for the session, the fmtp containing one of the following profiles (whether enumerated by * or explicitly configured) will not be offered and also not accepted during the SDP negotiation: {e, h44, h44i, c44i}

>> Back to Top

h264.in_band_param_sets_only

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

This parameter specifies the utilization of sprop-parameter-sets attributes of the initial SDP offered to MCP. It does not; however, apply to any SDP offered by MCP as MCP does not send any sprop-parameter-sets in its SDP offering to another party.

When set to Disable (default), if any profile-level-id plus packetization-mode in the initial offered SDP is accepted by MCP without level downgrade, MCP will utilize the H264 parameter sets value provided by sprop-parameter-sets attributes if present and valid. Refer to H264 FMTP for more detail on how profile-level-id and packetization-mode are negotiated.

When set to Enable, MPC will not utilize any H264 parameter sets value provided by sprop-parameter-sets attributes in the offered SDP. Attribute in-band-parameter-sets=1 will be included in the response SDP for any H264 media line to explicitly declare that only parameter sets received via RTP will be utilized and that implicitly forces the offerer to include all parameter sets within RTP packets.

>> Back to Top

h264payload

Default Value: 113
Valid Values: A valid H264 payload can only be an integer from 96 to 127 inclusive
Changes Take Effect: At start/restart

Default payload type number to use for the H264 codec

health.maxprocessingtime

Default Value: 600000
Valid Values: mpc.health.maxprocessingtime must be an integer no less than 0.
Changes Take Effect: At start/restart

Specifies the maximum processimg time of media thread in milliseconds. If a media thread is processing a media object more than the maximum processing time, meaning the thread may fall into an inifinite loop or deadlock, MCP will be terminated. Default value is 600000. If it is set to 0, MCP will not check processing time at all.

health.waittime

Default Value: 0
Valid Values: mpc.health.waittime must be an integer no less than 0.
Changes Take Effect: At start/restart

Specifies the wait time in milliseconds for Health Thread to check the health of media threads periodically. It is reasonable to set health.waittime smaller than health.maxprocessingtime. Default value is 0, meaning MCP will not perform health check. Note: do not enable health check unless it is really necessary, e.g. MCP has to restart in order to recover from a stuck situation.

includeavpfinsdp

Default Value: none
Valid Values: Choose between: "none", "audio", "video" or "audioandvideo".
Changes Take Effect: Immediately/session

Sets if the MCP will include SAVPF / AVPF instead of SAVP / AVP in SDP. If set to "none" (default), SAVP / AVP will be used. If set to "audio", only audio will have SAVPF / AVPF. If set to "video", only video will have SAVPF / AVPF. If set to "audioandvideo", both audio and video will have SAVPF / AVPF.

maxmediathreads

Default Value: 16
Valid Values: mpc.maxmediathreads must be an integer greater than 0 and less than or equal to 100
Changes Take Effect: At start/restart

Specifies the maximum number of media threads that can be created within MPC. Default value is 16. It is highly recommended to use the default setting.

maxrecordencryptedfilesize

Default Value: 120000000
Valid Values: mpc.maxrecordencryptedfilesize must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

When encrypting of recorded media files is enabled, this parameter specifies the maximum file size. Encrypted recordings will stop when this limit (bytes) is exceeded. If encryption is enabled for a recording, the media is kept in memory and only written to disk at the end of the recording. This parameter limits the amount of memory that can be used for one recording. The default value of 120 MByte, is large enough to store 2 hours for 2 channels each at 64 Kbits/sec. Value of 0 disables this limit.

maxrecordfilesize

Default Value: 0
Valid Values: mpc.maxrecordfilesize is greater than or equal to 0 and less than or equal to 4000000000.
Changes Take Effect: Immediately/session

All recordings (regular and FCR) will stop when this limit (bytes) is exceeded. Note that the recorded file will usually exceed this limit by a few hundred bytes depending on the codec and the container chosen. Value of 0 disables this limit.

media.senddtmfdropaudio

Default Value: 1

Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Specifies the behavior of dtmf transmission when there are overlapping dtmf events and audio data. When the value is set to Enable, audio whose timestamp overlaps with dtmf will be dropped. When the value is set to Disable, audio timestamp shifting will be invoked to avoid overlapping.

mediamgr.audiobuffersize

Default Value: 102400
Valid Values: Possible values are integer values greater than or equal to 1024 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the audio buffer size for the non-TTS source (in bytes). Default value is 102400.

mediamgr.autorecordformatselect

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

This parameter can enable a mode where regular recording or FCR encoding format is determined based on the incoming content format and the formats supported by the container. Note that auto detection only applies for the cases where the recording MIME string specifies only the container and not the codec/videocodec parameter. For example, if recording MIME is specified as "video/x-avi" without the videocodec parameter and if the incoming stream is H264, the resulting recording file will be an AVI with the H264 format. If this parameter was disabled, specifying "video/x-avi" would always result in AVI with the H263 format being recorded (since H263 is the default format for AVI).

mediamgr.CA_directory

Default Value:
Valid Values: Can be an empty string or a valid path to the Certificate Authority folder.
Changes Take Effect: Immediately/session

Path to the Certificate Authority folder for MSML file-based call recording. This parameter has lower priority than CA_file and if file is set directory is not used.

mediamgr.CA_file

Default Value:
Valid Values: Can be an empty string or a valid path to the Certificate Authority certificate file.
Changes Take Effect: Immediately/session

Path to the Certificate Authority certificate file for MSML file-based call recording. This parameter has higher priority than CA_directory.

mediamgr.enableEODdoublecheck

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Enables END of DATA double check. This is needed for the files created by GVPi that are written in shorter temporary files.

mediamgr.h263overrideTR

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

This parameter is for video recording with H263 and H263-1998 video codec. Whenever video/audio out-of-sync happens in recorded files, enabling this parameter may solve the issue. By default the MCP uses an inherent property available in each H263 video sample called Temporal Reference to determine timing between each video sample in a recording session. Video/audio out-of-sync in recorded files, however, may occur if the Temporal References in the video frames are incorrect. Setting this parameter to Enable will allow the MCP to correct Temporal Reference and try to synchronize video and audio during recording sessions. Setting this parameter to Disable will keep the Temporal Reference intact.

mediamgr.hlsconsecutiveerrorsthreshold

Default Value: 5
Valid Values: Must be an integer of value equal to 0 to indicate no limit, or any value greater or equal to 1 to impose a limit.
Changes Take Effect: Immediately

Specifies the number of HTTP Live Streaming (HLS) segment fetching errors allowed to happen consecutively before giving up the fetching process. If the value is equal to 0, than MCP will try to fetch the segments until there are segments to be fetched, or until there is still content in the buffer to be played.

>> Back to Top

mediamgr.hlstotalerrorsthreshold

Default Value: 20
Valid Values: Must be an integer of value equal to 0 to indicate no limit, or any value greater or equal to 1 to impose a limit.
Changes Take Effect: Immediately

Specifies the total number of HTTP Live Streaming (HLS) segment errors allowed to happen before giving up the fetching process. This value must be equal or greater than the specified by the parameter mediamgr.hlsallowedconsecutiveerrors. If the value is equal to 0, than MCP will try to fetch the segments until there are segments to be fetched, or until there is still content in the buffer to be played. The following chart exemplifies how both thresholds work: | Segment# | Fetch result | Cons. Counter | Total Counter | | 1 | Good | 0 | 0 | | 2 | Fail | 1 | 1 | | 3 | Fail | 2 | 2 | | 4 | Good | 0 | 2 | | 5 | Fail | 1 | 3 |

>> Back to Top

mediamgr.ignore_cert_err

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

Specifies whether to ignore certificate verification errors and continue encrypting the media for MSML file-based call recording.

>> Back to Top

mediamgr.isofilerecordheadersize

Default Value: 55000
Valid Values: mpc.mediamgr.isofilerecordheadersize must be an integer that is greater than 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

The header of ISO file container grows as the content of the file grows. The MCP will reserve the header size to harddrive before recording media of an ongoing session. The MCP will actually record header at the end of the session and harddrive operations may be required if the reserved header size is not enough to accommodate the actual header size.

mediamgr.maxcertificatecachesize

Default Value: 2000000
Valid Values: mpc.mediamgr.maxcertificatecachesize must be an integer that is greater than or equal to 20000 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

This parameter sets the maximum size of the cached memory in bytes to store certificates for MSML recordings.

mediamgr.maxcertificatelength

Default Value: 5000
Valid Values: mpc.mediamgr.maxcertificatelength must be an integer that is greater than or equal to 1 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

This parameter sets the maximum length in characters of one certificate for MSML file-based call recording. If certificate will be longer than this length it will not be used for encryption of MSML recording.

mediamgr.maxcertificatesperprofile

Default Value: 10
Valid Values: mpc.mediamgr.maxcertificatesperprofile must be an integer that is greater than or equal to 1 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

This parameter sets the maximum number of certificates for MSML file-based call recording per IVR profile.

mediamgr.precacheprofileforcallrecording.0

Default Value:
Valid Values: Can be an empty string or a stringin the format:
"[TENANT_NAME].IVR_PROFILE_NAME".

Changes Take Effect: At start/restart

IVR profile information for pre-caching of encryption related parameters for call recording. If not specified, the encryption parameters will be cached when the profile is first accessed. The format of the profile information should strictly follow "[TENANT_NAME].IVR_PROFILE_NAME".

>> Back to Top

mediamgr.precacheprofileforcallrecording.1

Default Value:
Valid Values: Can be an empty string or a stringin the format: "[TENANT_NAME].IVR_PROFILE_NAME".
Changes Take Effect: At start/restart

IVR profile information for pre-caching of encryption related parameters for call recording. If not specified, the encryption parameters will be cached when the profile is first accessed. The format of the profile information should strictly follow "[TENANT_NAME].IVR_PROFILE_NAME".

>> Back to Top

mediamgr.precacheprofileforcallrecording.2

Default Value:
Valid Values: Can be an empty string or a stringin the format: "[TENANT_NAME].IVR_PROFILE_NAME".
Changes Take Effect: At start/restart

IVR profile information for pre-caching of encryption related parameters for call recording. If not specified, the encryption parameters will be cached when the profile is first accessed. The format of the profile information should strictly follow "[TENANT_NAME].IVR_PROFILE_NAME".

>> Back to Top

mediamgr.precacheprofileforcallrecording.3

Default Value:
Valid Values: Can be an empty string or a stringin the format: "[TENANT_NAME].IVR_PROFILE_NAME".
Changes Take Effect: At start/restart

IVR profile information for pre-caching of encryption related parameters for call recording. If not specified, the encryption parameters will be cached when the profile is first accessed. The format of the profile information should strictly follow "[TENANT_NAME].IVR_PROFILE_NAME".

>> Back to Top

mediamgr.precacheprofileforcallrecording.4

Default Value:
Valid Values: Can be an empty string or a stringin the format:
"[TENANT_NAME].IVR_PROFILE_NAME".
Changes Take Effect: At start/restart

IVR profile information for pre-caching of encryption related parameters for call recording. If not specified, the encryption parameters will be cached when the profile is first accessed. The format of the profile information should strictly follow "[TENANT_NAME].IVR_PROFILE_NAME".

>> Back to Top

mediamgr.precacheprofileforcallrecording.5

Default Value:
Valid Values: Can be an empty string or a stringin the format:
"[TENANT_NAME].IVR_PROFILE_NAME".
Changes Take Effect: At start/restart

IVR profile information for pre-caching of encryption related parameters for call recording. If not specified, the encryption parameters will be cached when the profile is first accessed. The format of the profile information should strictly follow "[TENANT_NAME].IVR_PROFILE_NAME". If additional profiles are wanted, new parameters with an incremental number at the end need to be added.

>> Back to Top

mediamgr.rec_iframe_delay_threshold

Default Value: 160
Valid Values: mpc.mediamgr.rec_iframe_delay_threshold must be an integer at least equal to -1 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

This parameter is for video recording with audio and video. When starting a recorder, a few initial video frames may be dropped as the first self-referencing intra frame is not received for some reasons. As a result, audio duration received prior to receiving the next self-referencing video frame may be too long and it makes audio and video get out-of-sync. This parameter limits how long in milliseconds the audio is allowed in this situation without having to do video filling. The value of -1 will disable this feature.

>> Back to Top

mediamgr.recordmp3audiobuffer

Default Value: 4000
Valid Values: mpc.mediamgr.recordmp3audiobuffer must be an integer that is greater than or equal

to 2000.
Changes Take Effect: At start/restart

Specifies the duration of the audio buffer in milliseconds for MP3 recording.

mediamgr.recordrtphinttrack

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

For ISO file container, recording a hint track for a media track into a recording file allows the file to be streamed when placed on a streaming server.

mediamgr.recordwritetimeinterval

Default Value: 1000
Valid Values: mpc.mediamgr.recordwritetimeinterval must be an integer equal to 0, or greater than or equal to 1000 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

This parameter sets the time period in milliseconds for the periodic writing of recording data to a file and must be an integer equal to 0, or greater than or equal to 1000. If time interval is set to 0 it means that time driven recording is disabled.

mediamgr.rtsplowerbufferthreshold

Default Value: 100
Valid Values: mpc.mediamgr.rtsplowerbufferthreshold must be an integer that is greater than 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

If mpc.media.rtsppause is set to 1, and RTP streaming has been paused. PLAY will be sent to resume RTP streaming if the packet buffer size has reduced to the lower threshold. Default value is 100 packets. The lower threshold must be smaller than the upper threshold. This value can be overridden using the RTSP URL parameter "vg-rtsp-lowerbufferthreshold".

mediamgr.rtsppause

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Select the option based on RTSP server support for PAUSE. This value can be overridden using the RTSP URL parameter "vg-rtspserver-pause". 0 - PAUSE is not supported by the RTSP server 1 - PAUSE is supported by the RTSP server

>> Back to Top

mediamgr.rtspplayrange

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Select the option based on RTSP server support for the Range parameter in a PLAY request. This value can be overridden using the RTSP URL parameter "vg-rtspserver-playrange". 0 - PLAY is not supported by the RTSP server 1 - PLAY is supported by the RTSP server

>> Back to Top

mediamgr.rtspupperbufferthreshold

Default Value: 200
Valid Values: mpc.mediamgr.rtspupperbufferthreshold must be an integer that is greater than 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

If mpc.media.rtsppause is set to 1, PAUSE will be sent to stop RTP streaming when the packet buffer size has reached the upper threshold. Default value is 200 packets. This value can be overridden using the RTSP URL parameter "vg-rtsp-upperbufferthreshold".

>> Back to Top

mediamgr.sharedhttpservers

Default Value:
Valid Values: Can be an empty string or a list of space separated network addresses.
Changes Take Effect: At start/restart

Specifies the live HTTP server addresses (without port) delimited by space - address can be a hostname, IPv4, or IPv6 address.
e.g. genesyslab.com [fe80:0:0:0:200:f8ff:fe21:67cf] 192.168.0.101 dummyhost 192.168.0.102

3ffe:1900:4545:3:200:f8ff:fe21:67cf
If HTTP URL play request has streaming turned on and the URL address is one of the addresses specified by this configuration, the HTTP URL will be played in live HTTP streaming mode.
In live HTTP streaming mode, multiple sessions specifying the same URL will play from the same HTTP stream, and newly starting sessions will start playing from the currently arriving media.

mediamgr.strictsamplingrate

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

The sampling rate that is officially supported for audio is 8000 Hz and video is 90000 Hz. Some media files may indicate a different sampling rate than what supported and trying to play those files may result in bad media quality. If this parameter is set to Enable, media files indicating any sampling rate other than officially supported will not be played. If this parameter is set to Disable, media files indicating any sampling rate other than supported will still be attempted to play by MCP but without guarantee quality.

mediamgr.videobuffersize

Default Value: 256000
Valid Values: Possible values are integer values greater than or equal to 8000 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the video buffer size for non-TTS source (in bytes). This value should be sufficient for buffering up to 1 second of all supported modes of H263 and H264 video stream. Refer to the appropriate specification, e.g. ITU-T H.263 for H.263 and ITU-T H.264 for H.264, and GVP User Guide to determine the size to set to avoid overrunning the video buffer.

mediamgr.videofillingframeduration

Default Value: 1000
Valid Values: mpc.mediamgr.videofillingframeduration must be an integer that is greater than 32 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

This parameter is for video recording with audio and video. It is used when video filling happens, e.g. the total duration of video frames dropped exceeds mpc.mediamgr.rec_iframe_delay_threshold. When doing video filling, the duration of between the first and second recorded video frame will be stretched so that synchronization between audio and video is maintained. This parameter limits how

long the duration in milliseconds between two video frames can be stretched and guides the recorder to regenerate more than one video frame until it completes video filling duration. Note that the value of this parameter should not exceed that of mpc.mediamgr.maxdurationpervidframe. Half the value of mpc.mediamgr.maxdurationpervidframe is recommended and is set as the default.

mediamgr.videofillingthreshold

Default Value: 2000
Valid Values: mpc.mediamgr.videofillingthreshold must be a postive integer that is 0 or greater and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

This parameter is for video recording. It is used to decide when to trigger video filling; i.e., when the interval between the current and the last frame exceeds this threshold, video filling will be done on the next I-frame for this gap, with a new frame being created for each "videofillingframeduration". This parameter is in milliseconds, and this feature can be disabled by setting this to 0 or a large value.

mixer.audiodelay_flush_all_threshold

Default Value: 500
Valid Values: mpc.mixer.audiodelay_flush_all_threshold must be an integer that is at least equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the maximum difference between the current and expected packet time stamps when mixer flushes all buffered packets. The units are in milliseconds. Default value is 500. Setting to zero disables flushing.

mixer.audiodelay_flush_silence_threshold

Default Value: 100
Valid Values: mpc.mixer.audiodelay_flush_silence_threshold must be an integer that is at least equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the maximum difference between the current and expected silent packet time stamps when mixer flushes silent buffered packets. The units are in milliseconds. Default value is 100. Setting to zero disables flushing.

mp3.bitrate

Default Value: 16
Valid Values: Choose between: 8, 16, 24, 32, 48, 64, 96, 128, 160, 192, 256 or 320.
Changes Take Effect: Immediately/session

This specifies the MP3 encoding bitrate in Kbits/Second, which will determine the quality and size of a recorded MP3 file. Beware that bitrates above 160 kbps are not supported when sampling rate is less than 32 KHz, i.e., 16 KHz, as per MPEG 2 Layer 3 standard. 8 kbps bitrate is supported only for mono MP3 recording (for GIR only). Note that integer transcoding is not supported for bitrates lower than 32 kbps. For bitrates lower than 32 kbps, the sampling rate will be forced to 16 KHz, and MPEG 2 Layer 3 will be used.

>> Back to Top

mp3.compression_level_current_encoder

Default Value: 7
Valid Values: A number between 1 and 9 inclusive.
Changes Take Effect: Immediately/session

This parameter can be used to specify the quality level of the mp3 file from the current encoder, with 1 the highest and 9 the lowest quality. This parameter takes effect only when [mpc].mp3.use_current_encoder is true.

>> Back to Top

mp3.interfrequency.encoding

Default Value:
Valid Values: Can be an empty string or one of the following: 8 or 16.
Changes Take Effect: At start/restart

This can be used to specify the intermediate PCM16 format frequency as 8 or 16 KHz for MP3 encoding. By default, when this value is empty, the most suitable frequency based on the input format will be selected, except in the case of Call Recording, which would pick 8KHz by default.

>> Back to Top

mp3.samplingrate

Default Value: 32
Valid Values: Choose between: 16, 32 or 48.
Changes Take Effect: Immediately/session

This specifies the MP3 sampling rate used in encoding, in KHz. MPEG 1 Layer 3 standard is used for

encoding when the sampling rate is 32 KHz or higher, while MPEG 2 Layer 3 standard is used when it is lower. Beware that 16 KHz is not supported when MP3 integer transcoder option is enabled. 48 KHz is not supported during MSML call recording.

>> Back to Top

mp3.use_current_encoder

Default Value: false
Valid Values: Choose between: true or false
Changes Take Effect: Immediately/session

This parameter can be used to specify whether to use the current encoder for Stereo MP3 Encoding or not. If not, the Legacy version will be used.

>> Back to Top

mp3.use_integer_transcoder

Default Value: false
Valid Values: Choose between: true or false
Changes Take Effect: Immediately/session

This can be used to specify the type of MP3 transcoder. When set to false, a floating point implementation is used; else if set to true, an integer implementation is used.

>> Back to Top

mp3.use_integer_transcoder_current_encoder

Default Value: true
Valid Values: Choose between: true or false
Changes Take Effect: Immediately/session

This parameter can be used to specify the type of the current MP3 transcoder. When set to true, an integer implementation is used. If set to false, a floating point implementation is used. This parameter takes effect only when [mpc].mp3.use_current_encoder is true.

>> Back to Top

numdispatchthreads

Default Value: 1
Valid Values: mpc.numdispatchthreads must be an integer greater than 0 and less than or equal to 32
Changes Take Effect: At start/restart

Specifies the number of media dispatching threads.

opus.apptype

Default Value: voice
Valid Values: voice: VoIP Application that mostly uses speech. It may alter the output sound to make it more intelligible. audio: Application that uses music and speech. It provides lower coding delay.
Changes Take Effect: Immediately/session

Used to indicate the intended application type to the Opus encoder.

opus.bitrate

Default Value: 0
Valid Values: A number 0 for Opus default, or 6 to 510 for bitrate in kbps
Changes Take Effect: Immediately/session

This specifies the Opus maximum encoding bitrate in Kbits/second, or 0 to use Opus codec default (which is 12 kbps for NB codecs). Per RFC 6716, recommended values are 8-12 kbps for NB speech and 16-20 kbps for WB speech.

opus.complexity

Default Value: 10
Valid Values: A number between 0 and 10 inclusive.
Changes Take Effect: Immediately/session

Specifies Opus encoder's computational complexity. The range is 0 to 10, inclusive. The higher the value, the higher the complexity and the audio quality.

opus.fmtp

Default Value: useinbandfec=1
Valid Values: [useinbandfec=(0|1)] [; maxaveragebitrate=(6000-510000)]
Changes Take Effect: At start/restart

Specifies the default OPUS fmtp option string used by the platform when initiating SDP offer or

answering SDP offer. If the fmtp parameter is empty, the fmtp attribute line for OPUS will not be present in SDP offered by the platform.
The supported fmtp parameters are:
"useinbandfec" specifies whether MCP will take advantage of Opus in-band FEC or not. Opus in-band FEC helps mitigate errors when unstable transmissions occur. Possible values for "useinbandfec" are 1 and 0. If no value is specified, then in-band FEC will not be applied. For better error recovering results, the corresponding "maxaveragebitrate" should be set to a higher value, e.g. 48000.
"maxaveragebitrate" specifies the maximum average bitrate. Sending out this ftmp serves as a mandate to the remote end to not send a stream with average birate higher than the "maxaveragebitrate" value, as it might overload our network and/or receiver. Possible values for "maxaveragebitrate" are any integer in the range from 6000 to 510000 inclusive. If no value is specified, the bitrate will be automatically decided by MCP according to the audio source.
Unsupported fmtp parameters: usedtx; cbr; stereo; sprop-stereo; maxplaybackrate; sprop-maxcapturerate.

opus.packetloss

Default Value: 20
Valid Values: A number between 0 and 100 inclusive.
Changes Take Effect: Immediately/session

Used to indicate the expected packet loss percentage to the Opus encoder. The range is 0 to 100, inclusive. Higher values will give progressively more loss resistant behavior, at the expense of quality at a given bitrate in the lossless case, but greater quality under loss. The default value is 20, implying 20 percent packet loss is expected.

opuspayload

Default Value: 116
Valid Values: A valid Opus payload can only be an integer from 96 to 127 inclusive
Changes Take Effect: At start/restart

Default payload type number to use for the Opus codec

pcma.maxptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or

is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

pcma.ptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

pcmu.maxptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

pcmu.ptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will

take precedence.

persistdympayfmtpair

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately

With regards to RFC3264, a history of mappings from payload to format per m-line should be maintained for a session. This is to ensure a dynamic payload assigned to a particular format per m-line will not be reused for a different format when given a reINVITE and asked for an offer. When enabled, the aforementioned behavior will be performed. Note that a given reINVITE with an offer will be responded with regards to, if any, the given dynamic payloads, not the maintained payloads, due to backward-compatibility. When disabled, the legacy GVP method, where history is not remembered, will be used.

playcache.checkversiontime

Default Value: 300
Valid Values: The parameter must be set to integer of value 0 or greater.
Changes Take Effect: Immediately/session

This parameter sets the time period for checking that the source media of a cache entry has changed. This parameter does not apply to http://, https://, and file:// URL types, see note below.

When a entry in the cache is played, the source media will be checked for change if it has not been checked within this time period. If source media has been found to change the cache file(s) will be recreated using the changed media. Setting the value to zero will cause a check to be performed for every play. The value is set in seconds. The default value is 300.

For http://, https://, and file:// URL types, the checking of the source media for changes is handled by the fetching module so this parameter does not apply. For these URL types, if the media content provided by the fetching module changes it will be used on the next play. An exception to this is that for file:// URL types, this parameter does not apply if the prompt is played using the MSML <play> tag with precheck disabled.

playcache.directory

Default Value: $installationRoot$/cache/play
Valid Values: The parameter must be set to a directory path.
Changes Take Effect: At start/restart

This parameter sets the root directory of the play cache.

playcache.enable

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

This parameter enables the use of the play cache for media playing. When enabled, media played from http://, file://, rtsp://, and qtmf:// URL types will utilize the play cache. When transcoding is required to play the audio or video media from a source URL to an endpoint with particular audio and or video codecs settings, the play cache will save the transcoded media to audio and or video track file(s), and the media will be played from these track files the next time the URL is played to an endpoint with the same (or compatible) codecs settings. The play cache is enabled by default.

playcache.expiretime

Default Value: 24:00
Valid Values: The parameter must be set to format hours:minutes or hours, where hours and minutes are numeric values
Changes Take Effect: At start/restart

This parameter sets expire time for media URL entries in the play cache. If the amount of time since an entry for a media URL has been played exceeds this time, the entry will be deleted from the play cache. The format is hours:minutes or hours. Setting the value to zero disables deleting based on expire time. The default value is 24:00.

playcache.maxsize

Default Value: 500
Valid Values: The parameter must be set to integer greater than or equal to zero.
Changes Take Effect: At start/restart

This parameter sets the maximum disk space for the play cache. If the amount of disk space used by the play cache exceeds this value, cache entries will be deleted, starting with the least recently played. The value is set in MBytes. Setting the value to zero disables deleting based on disk spaced used. The default value is 500.

playremoteeodtimeout

Default Value: 10000
Valid Values: mpc.playremoteeodtimeout must be an integer greater than 0
Changes Take Effect: At start/restart

Specifies the duration in milliseconds to wait for remote buffer end of data callback. Playback by platform may involve data buffering locally and remotely, e.g. MCP as the local entity and PSTN-C as the remote one. After the local entity sends its very last packet to the remote, an EOD request will be issued to the remote entity where it will have to respond when its very last packet has been played. If there is no reply from the remote within this period, the local entity will issue a timeout on waiting and proceed.

>> Back to Top

playremoteflushtimeout

Default Value: 10000
Valid Values: mpc.playremoteflushtimeout must be an integer greater than 0
Changes Take Effect: At start/restart

Specifies the duration in milliseconds to wait for remote buffer flush callback. Playback by platform may involve data buffering locally and remotely, e.g. MCP as the local entity and PSTN-C as the remote one. After the local entity flushes its buffer, a flush request will be issued to the remote entity where it will have to respond. If there is no reply from the remote within this period, the local entity will issue a timeout on waiting and proceed.

>> Back to Top

playsilencefill

Default Value: 160
Valid Values: mpc.playsilencefill must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the amount of silence fill in milliseconds to add at the end of prompt play. Default is 160. Setting to zero disables play silence fill.

>> Back to Top

preferredipinterface

Default Value: V4
Valid Values: Choose between: V4 or V6
Changes Take Effect: Immediately

Specifies the preferred IP interface to use (IPv4 or IPv6) when performing SDP negotiation. In particular, this will be used to set the root connection attribute in SDP answers, and set the connection attribute in SDP offers.

record.allowsyncdiskwrite

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

Specifies whether synchronous write during recording is allowed.

record.defaultdtmfhandling

Default Value: as-is
Valid Values: as-is: (default) Record everything as-is from the RTP stream. Inband DTMFs will be recorded, but RFC2833 digits will not no-digits: Strip out all DTMF digits. This includes inband or RFC2833. NOTE: When telephone-event is negotiated on the call, if inband audio DTMFs are received, they will not be removed from the recording. all-digits: Record all DTMF digits, including inband, and generate audio for RFC2833 digits
Changes Take Effect: Immediately/session

Specifies the recording behavior for DTMFs in a Simple Recording.

recordcachedir

Default Value: $installationRoot$/cache/record
Valid Values: The parameter must be set to a directory path.
Changes Take Effect: At start/restart

This parameter sets the temporary recording cache directory in case of MSML call recording. Once the recording completes the the recording files are removed from the cache directory after successfully placing them at the final recording destination.

recordnumparallelpost

Default Value: 30
Valid Values: mpc.recordnumparallelpost must be an integer that is greater than or equal to 0 and

less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

While doing MSML call recording the recordings are added to a list if they need to be posted to Amazon S3, Call Recording API, HTTP/HTTPS or SpeechMiner. A separate posting thread wakes up from time to time and works on the list of recordings to be posted. This parameter specifies the max. number of active (post in progress) post attempts in the system at any given time. The default value is 30, which means 30 active record posts can exist in the system at a given time. Setting this parameter to 0 disables recordings from being posted/moved to final destination and the recording files remain in the record cache directory forever.

>> Back to Top

recordpostbacklogthreshold

Default Value: 25
Valid Values: mpc.recordpostbacklogthreshold must be an integer that is greater than or equal to 1 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

While doing MSML call recording the recordings are added to a list if they need to be posted to Amazon S3, Call Recording API, HTTP/HTTPS or SpeechMiner. A separate posting thread wakes up from time to time and works on the list of recordings to be posted. If due to any reason the posting thread lags behind, then we send an alarm notification as soon as the post backlog reaches or exceeds the threshold limit specified by this parameter. We send another alarm notification when the post backlog is below the configured threshold. The default value is 25, which means that we send an alarm when the post backlog reaches or exceeds 25 posts.

>> Back to Top

recordpostinterval

Default Value: 15000
Valid Values: mpc.recordpostinterval must be an integer that is greater than or equal to 1 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

While doing MSML call recording the recordings are added to a list if they need to be posted to Amazon S3, Call Recording API, HTTP/HTTPS or SpeechMiner. A separate record posting thread wakes up from time to time and works on the list of records to be posted. This parameter specifies the time gap between consecutive processing attempts by this thread. The unit for this parameter is milliseconds and the default value is 15 seconds (15000 milliseconds).

>> Back to Top

recordpostretrybackoff

Default Value: 120000

Valid Values: mpc.recordpostretrybackoff must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

While doing MSML call recording the recordings are added to a list if they need to be posted to Amazon S3, Call Recording API, HTTP/HTTPS or SpeechMiner. A separate posting thread wakes up from time to time and works on the list of recordings to be posted. This parameter specifies the backoff period between consecutive post retry attempts. This only applies to recording entries whose posting failed the first time and needs to be retried again. The unit for this parameter is milliseconds and the default value is 120 seconds (120000 milliseconds), which means that the posting thread would attempt post retries once every 2 minutes approximately. Setting this parameter to 0 causes all retry attempts to occur at the same time without any wait period.

>> Back to Top

recordpostretrycount

Default Value: 3
Valid Values: mpc.recordpostretrycount must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

While doing MSML call recording the recordings are added to a list if they need to be posted to Amazon S3, Call Recording API, HTTP/HTTPS or SpeechMiner. A separate posting thread wakes up from time to time and works on the list of recordings to be posted. This parameter specifies the max. number of retry attempts made for failed posting attempts. After the specified number of attempts the failing recording is removed from the list of recordings to be posted and is never retried again. The default value is 3, which means failed post attempts would be retried a maximum of 3 times. Setting this parameter to 0 disables any retry attempts for posts having recoverable errors.

>> Back to Top

refframereqonconnjoin

Default Value: true
Valid Values: Choose between: true or false
Changes Take Effect: Immediately/session

Enables requesting of intra-frames when there is a join or a bridge between connections / calls.

>> Back to Top

rru.beginsilence

Default Value: 1000
Valid Values: The value must be an integer from 0 to 10000 inclusive
Changes Take Effect: At start/restart

Specifies the amount of begin silence in milliseconds to insert for RRU. Default is 1000.

rru.endsilence

Default Value: 3000
Valid Values: The value must be an integer from 0 to 10000 inclusive
Changes Take Effect: At start/restart

Specifies the amount of end silence in milliseconds to insert for RRU. Default is 3000.

rtcp.tos

Default Value: 0
Valid Values: Possible values are integers from 0 to 255 inclusive.
Changes Take Effect: Immediately/session

Specifies the IP Differentiated Services Field (also known as ToS) to set in all outgoing audio RTCP packets. Note that this configuration does not work for Windows 2008. For Windows 2008, the setting needs to be configured at the OS level through the policy settings. Please refer to the GVP User's Guide.

rtcp.tos.video

Default Value: 0
Valid Values: Possible values are integers from 0 to 255 inclusive.
Changes Take Effect: Immediately/session

Specifies the IP Differentiated Services Field (also known as ToS) to set in all outgoing video RTCP packets. Note that this configuration does not work for Windows 2008. For Windows 2008, the setting needs to be configured at the OS level through the policy settings. Please refer to the GVP User's Guide.

rtcpfeedback.audio

Default Value: allow
Valid Values: Choose between: "force", "allow" or "disable".
Changes Take Effect: Immediately/session

Sets the behavior for RTCP feedback. If set to "allow" (default) the MCP will be enabled to send Generic NACK messages as per RFC 4585 Section 6.2.1 when the far end sends SAVPF or AVPF in SDP. If set to "force", the MCP will enable these messages independent of SDP. If set to "disable", the MCP will disable these messages independent of SDP. This configuration parameter covers audio.

>> Back to Top

rtcpfeedback.video

Default Value: allow
Valid Values: Choose between: "force", "allow" or "disable".
Changes Take Effect: Immediately/session

Sets the behavior for RTCP feedback. If set to "allow" (default) the MCP will be enabled to send Generic NACK messages as per RFC 4585 Section 6.2.1 and RTCP PLI as per RFC 4584 section 6.3.1, when the far end sends SAVPF or AVPF in SDP. If set to "force", the MCP will enable these messages independent of SDP. If set to "disable", the MCP will disable these messages independent of SDP. This configuration parameter covers video.

>> Back to Top

rtp.activetimeout

Default Value: 0
Valid Values: mpc.rtp.activetimeout should be an integer greater than 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

MPC will not send the first outgoing RTP packet until it received an incoming RTP packet or the RTP active timeout is reached. This value is time in milliseconds. Default to 0, in which RTP packets will be transmitted immediately.

>> Back to Top

rtp.audiobuffersize

Default Value: 50000
Valid Values: mpc.rtp.audiobuffersize must be an integer that is greater than 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the size of the buffer used for sending audio RTP data in bytes. If 0 is specified, buffer size will be initially set to 120000, and will be increased automatically if more space is needed.

>> Back to Top

rtp.dejitter.delay

Default Value: 0
Valid Values: rtp.dejitter.delay must be an integer that is greater than or equal to 0 and less than or equal to 10000.
Changes Take Effect: At start/restart

Specifies the total duration (in milliseconds) of RTP packets to buffer for the inter-arrival dejittering purpose. This will translate to an initial delay before the packets are dispatched internally for further processing. 0 disables the inter-arrival jitter removal functionality.

>> Back to Top

rtp.dejitter.timeout

Default Value: 200
Valid Values: rtp.dejitter.timeout must be an integer that is greater than or equal to 1 and less than or equal to 1000.
Changes Take Effect: At start/restart

Controls how long the buffered RTP packet will wait for the missing RTP (in milliseconds). If timeout occurs, the dispatch process is initiated regardless of the missing packet.

>> Back to Top

rtp.dtmf.crlfenable

Default Value: false
Valid Values: Choose between: true or false
Changes Take Effect: Immediately/session

If the flag is set to true CRLF will be added after Duration attribute

>> Back to Top

rtp.dtmf.receive

Default Value: SIPINFO INBAND
Valid Values: Any combination of: SIPINFO and INBAND. Or "none".
Changes Take Effect: Immediately/session

Specifies the allowable ways to receive DTMF in an RTP session if telephone-event (RFC2833) is not negotiated in SDP. Allowable values are to support DTMFs relayed over SIP INFO messages and/or over INBAND audio DTMFs. The default selection is to enable both, however inband DTMF will not be supported if telephone-event is negotiated. SIP INFO will be supported if selected here whether or not telephone-event is negotiated. If support neither is selected on it's own will result in neither being

supported. SIPINFO - Support DTMFs relayed over SIP INFO messages INBAND - Support inband audio DTMFs none - Support neither

rtp.dtmf.send

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: Immediately/session

Specifies the allowable ways to send DTMF in an RTP session if telephone-event (RFC2833) is not negotiated. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

rtp.enablertcp

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Specifies whether to transmit RTCP packets.

rtp.filteroutduplicate

Default Value: false
Valid Values: Choose between: true or false
Changes Take Effect: At start/restart

Enable to filter out duplicate RTP packets. By default it is not enabled. Enable it only when duplicate RTP packets may appear in some environments.

rtp.fixedsocketthreads

Default Value: 8
Valid Values: Possible values are integers from 0 to 100 inclusive.
Changes Take Effect: At start/restart

Specifies the fixed number of RTP socket threads. Fixed number of RTP socket threads specified by this parameter will be allocated at the start-up time, and no new RTP threads will be created during

run-time. If set to 0: 1 RTP thread will be created at start-up time, and new RTP threads will be created in proportion to the number of open sockets during run-time. Setting this parameter to 0 is not recommended. Instead this parameter should be set in proportion to the number of CPU cores in the system.

>> Back to Top

rtp.h264allowrfc3984stapa

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Specifies whether RFC3984 single-time aggregation packet type A (STAPA) can be used when non-interleaved packetization mode is negotiated. When non-interleaved mode is negotiated as the packet transport mode, single NAL unit, STAPA and fragmentation type A packets will be sent by default. Alternatively, this configuration, when set to Disable, can allow only single NAL unit and fragmentation type A packets to be used.

>> Back to Top

rtp.inputmode

Default Value: vad
Valid Values: Choose between: "continuous" or "vad"
Changes Take Effect: At start/restart

Specifies the input mode of incoming RTP streams: continuous - Continuous input; vad - VAD input.

>> Back to Top

rtp.localaddr

Default Value: $LocalIP$
Valid Values: Can be an empty string or a valid IPv4 address.
Changes Take Effect: At start/restart

mpc.rtp.localaddr provides configurability of the connection part (IPv4) of SDP messages sent by the MCP.
If this parameter is not specified, then the IP Address of the local system will be used.

>> Back to Top

rtp.localaddrv6

Default Value:
Valid Values: Can be an empty string or a valid IPv6 address.

Changes Take Effect: At start/restart

mpc.rtp.localaddr provides configurability of the connection part (IPv6) of SDP messages sent by the MCP.
If this parameter is not specified, then the IP Address of the local system will be used.

rtp.maxrtppacketsize

Default Value: 0
Valid Values: mpc.rtp.maxrtppacketsize must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the maximum size of RTP packets that are sent from the platform in bytes. This also controls the maximum size of the SRTP packets being sent. If 0 is specified, there is no limit - unless internal buffer specified by rtp.audiobuffersize or rtp.videobuffersize runs out of space. Note that for SRTP, huge packet sizes (above 80KB which is greater than the typical 1500 MTU limit) may cause SRTP encryption errors before being sent out. If the problem is encountered for very large video packets, it can be worked around by negotiating lower level/bitrate or by leveraging codec specific transport mechanisms (e.g. packetization-mode=1 SDP fmtp for H264).

rtp.multichantimeout

Default Value: 60000
Valid Values: mpc.rtp.multichantimeout must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the RTP timeout value in milliseconds for multichannel recordings. A RTP stream will be considered inactive if there has been no activity for the timeout period. A value of 0 disables this timeout. Default value is 60000.

rtp.overwritessrcandtimestamp

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Set to Enable to have the SSRC and timestamp of outgoing RTP packets overwritten during a media bridge. Setting to Disable would result in legacy VG Platform behavior. Setting to Enable is required for SRTP operations, and with certain devices that have problems with receiving inconsistent SSRC

and timestamp information.

rtp.packetseq

Default Value: 0
Valid Values: mpc.rtp.packetseq must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the sequence number for the first outgoing RTP packet. IF set to 0, the first sequence number will be randomly generated for each RTP stream. Default value is 0.

rtp.portrange

Default Value: 20000-45000
Valid Values: Possible values are the empty string or low-high, where low and high are integers from 1030 to 65535 inclusive
Changes Take Effect: At start/restart

Specifies the RTP ports to be used by MPC.

rtp.prefilltime

Default Value: 200
Valid Values: Possible values are integers from 100 to 1000 inclusive.
Changes Take Effect: At start/restart

Specifies the time (milliseconds) limit in which the pre-fill amount needs to be sent out by. If the value is M milliseconds and the pre-fill amount is N milliseconds, then the RTP packets will be sent out at (N/M) times the real-time rate for M milliseconds. When in faster than realtime and N/M results in less than double the real-time rate, N/M will be set to double the real-time rate. Possible values are 100 to 1000, inclusive. Default value is 200.

rtp.request_iframe

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

Allows/disallows requesting of video intra-frames. Intra-frames require more bandwith, but improve video quality.

rtp.restrictsource

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Specifies whether to allow dropping packets from other sources (filtering).

rtp.rfc2429maxpacketsize

Default Value: 1400
Valid Values: mpc.rtp.rfc2429maxpacketsize must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the maximum RTP packet size for RFC2429 packets in bytes. Specifies the maximum RTP packet size for H263 RFC2429 packets in bytes. Any packet that exceeds the limit will be broken down into smaller packets. This parameter is used to prevent the OS from sending fragmented UDP packets, which may not be supported by some devices.
Default value is 1400. Set to 0 to disable the limit.

rtp.sendmode

Default Value: vad
Valid Values: Choose between: "continuous" or "vad"
Changes Take Effect: At start/restart

Specifies the output mode of outgoing RTP streams: continuous - Continuous output; vad - VAD output.

rtp.senduponrecv

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

Specifies whether defer sending of RTP packets until valid RTP packets are received.

rtp.source_buffer_video_data_size

Default Value: 0
Valid Values: rtp.source_buffer_video_data_size must be an integer that is greater than or equal to 0 and less than or equal to 50000000.
Changes Take Effect: Immediately/session

Specifies the maximum amount of video media data (in bytes) that the RTP source buffer can contain. If set to zero, the value will be determined automatically based on the video bitrate.

If video conferencing with video_output_type set to mixed is to be utilized, the platform will buffer the last video IFrame and subsequent packets received for each conference participant. This parameter determines the amount of video data that can be buffered.

The default value zero is correct for most applications.

rtp.source_buffer_video_size

Default Value: 500
Valid Values: rtp.source_buffer_video_size must be an integer that is greater than or equal to 200 and less than or equal to 2000.
Changes Take Effect: Immediately/session

Specifies the maximum number of packets that the RTP source video buffer can contain. The default value 500 is correct for most applications.

If video conferencing with video_output_type set to mixed is to be utilized, the platform will buffer the last video IFrame and subsequent packets received for each conference participant. This parameter sets the number of packets that can be buffered. It should be set to the maximum expected input IFrame interval (in packet count) plus 20. For example, if a device will be connected that sends an IFrame every 600 packets, the value should be set to 620.

rtp.statisticsinterval

Default Value: 300000
Valid Values: Possible values are integers from 0 to 3600000 inclusive.
Changes Take Effect: At start/restart

Specifies the interval (in ms) at which statistics logging in the RTP layer will be logged. Setting this value to 0 will disable the statistics logging. If enabled, will log when an RTP connection is destroyed, regardless of interval.

rtp.timeout

Default Value: 60000
Valid Values: mpc.rtp.timeout must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the RTP timeout value in milliseconds. A RTP stream will be considered inactive if there has been no activity for the timeout period. A value of 0 disables this timeout. Default value is 60000.

rtp.tos

Default Value: 0
Valid Values: Possible values are integers from 0 to 255 inclusive.
Changes Take Effect: Immediately/session

Specifies the IP Differentiated Services Field (also known as ToS) to set in all outgoing audio RTP packets. Note that this configuration does not work for Windows 2008. For Windows 2008, the setting needs to be configured at the OS level through the policy settings. Please refer to the GVP User's Guide.

rtp.tos.video

Default Value: 0
Valid Values: Possible values are integers from 0 to 255 inclusive.
Changes Take Effect: Immediately/session

Specifies the IP Differentiated Services Field (also known as ToS) to set in all outgoing video RTP packets. Note that this configuration does not work for Windows 2008. For Windows 2008, the setting needs to be configured at the OS level through the policy settings. Please refer to the GVP User's Guide.

rtp.video.udprecvbuffersize

Default Value: 60480
Valid Values: mpc.rtp.udpbuffersize must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the size of the receive buffer for video RTP UDP sockets in bytes. Default value is 60480. If set to zero the buffer size will be set to the default for the operating system.

>> Back to Top

rtp.video.udpsendbuffersize

Default Value: 60480
Valid Values: rtp.video.udpsendbuffersize must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the size of the send buffer for video RTP UDP sockets in bytes. Default value is 60480. If set to zero the buffer size will be set to the default for the operating system.

>> Back to Top

rtp.videobuffersize

Default Value: 0
Valid Values: mpc.rtp.videobuffersize must be an integer that is greater than 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the size of the buffer used for sending RTP video data in bytes. This value should be sufficient for buffering up to 1 second of all supported modes of H263 and H264 video stream. Refer to the appropriate specification, e.g. ITU-T H.263 for H.263 and ITU-T H.264 for H.264, and GVP User Guide to determine the size to set to avoid overrunning the video buffer. If 0 is specified, buffer size will be initially set to 120000, and will be increased automatically if more space is needed.

>> Back to Top

rtp.vp8maxpacketsize

Default Value: 1400
Valid Values: mpc.rtp.vp8maxpacketsize must be an integer that is greater than or equal to 0 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the maximum RTP packet size for VP8 packets in bytes. Any VP8 packet that exceeds the limit will be broken down into smaller packets. This parameter is used to prevent the OS from sending fragmented UDP packets, which may not be supported by some devices.
Default value is 1400. Set to 0 to disable the limit.

>> Back to Top

rtsp.connection.portrange

Default Value: 14000-15999
Valid Values: Possible values are the empty string or low-high, where low and high are integers from 1030 to 65535 inclusive
Changes Take Effect:

Specifies the available ports to receive RTSP messages from RTSP servers.

>> Back to Top

rtsp.localaddr

Default Value:
Valid Values: Can be an empty string or a valid IPv4 address.
Changes Take Effect: At start/restart

Specifies the IPv4 interface to receive RTSP messages from RTSP servers. If this parameter is not specified, then the IP Address of the local system will be used. Note that most RTSP servers require this parameter and [mpc]rtsp.rtp.localaddr to be configured identically.

>> Back to Top

rtsp.localaddrv6

Default Value:
Valid Values: Can be an empty string or a valid IPv6 address.
Changes Take Effect: At start/restart

Specifies the IPv6 interface to receive RTSP messages from RTSP servers. If this parameter is not specified, then the IP Address of the local system will be used. Note that most RTSP servers require this parameter and [mpc]rtsp.rtp.localaddrv6 to be configured identically.

>> Back to Top

rtsp.rtp.localaddr

Default Value:
Valid Values: Can be an empty string or a valid IPv4 address.
Changes Take Effect: At start/restart

Specifies the interface to receive RTP media from RTSP servers. If this parameter is not specified, then the value in [mpc]rtp.localaddr will be used. Note that most RTSP servers require this parameter and [mpc]rtsp.localaddr to be configured identically.

rtsp.rtp.localaddrv6

Default Value:
Valid Values: Can be an empty string or a valid IPv6 address.
Changes Take Effect: At start/restart

Specifies the IPv6 interface to receive RTP media from RTSP servers. If this parameter is not specified, then the value in [mpc]rtp.localaddrv6 will be used. Note that most RTSP servers require this parameter and [mpc]rtsp.localaddrv6 to be configured identically.

rtsp.rtp.portrange

Default Value:
Valid Values: Possible values are the empty string or low-high, where low and high are integers from 1030 to 65535 inclusive
Changes Take Effect: At start/restart

Specifies the available ports to receive RTP media from RTSP servers. If this parameter is not specified, then the value in [mpc]rtp.portange will be used.

sdp.audiobandwidth

Default Value:
Valid Values: A valid bandwidth as specified in RFC4566.
Changes Take Effect: At start/restart

Specifies the value of the bandwidth attribute as specified in RFC4566. If not empty, this value will be added to the bandwidth attribute for the audio media description in offerring SDP

sdp.connection

Default Value:
Valid Values: IN (IP4|IP6) <connection-address>
Changes Take Effect: At start/restart

Specifies the connection value of outgoing SDP content for a call. Format is defined at RFC4566. It is indepedent of [mpc] rtp.localaddr, which sets the IP address on the platform to be used. This parameter should only be used if the you wish to send an IP address in SDP that is different from the IP address that will be used in [mpc] rtp.localaddr. Setting it to an empty string disables used of this parameter. Possible use cases include when the MCP has a private IP address and a public IP address, and the SDP needs to contain the public IP address.

Examples of possible values include:
- IN IP4 127.0.0.1
- IN IP6 ::1

sdp.map.origin.0

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.0.dtmftype and the confgain specified by mpc.sdp.map.origin.0.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.0 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

sdp.map.origin.0.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.0.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.0 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

sdp.map.origin.0.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.0 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

>> Back to Top

sdp.map.origin.1

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.1.dtmftype and the confgain specified by mpc.sdp.map.origin.1.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.1 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

>> Back to Top

sdp.map.origin.1.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.1.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.1 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

>> Back to Top

sdp.map.origin.1.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.1 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

>> Back to Top

sdp.map.origin.2

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.2.dtmftype and the confgain specified by mpc.sdp.map.origin.2.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.2 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

>> Back to Top

sdp.map.origin.2.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.2.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.2 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

>> Back to Top

sdp.map.origin.2.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.2 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

sdp.map.origin.3

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.3.dtmftype and the confgain specified by mpc.sdp.map.origin.3.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.3 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

sdp.map.origin.3.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.3.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.3 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

sdp.map.origin.3.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.3 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

sdp.map.origin.4

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.4.dtmftype and the confgain specified by mpc.sdp.map.origin.4.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.4 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

sdp.map.origin.4.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.4.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.4 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

sdp.map.origin.4.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.4 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

sdp.map.origin.5

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.5.dtmftype and the confgain specified by mpc.sdp.map.origin.5.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.5 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

sdp.map.origin.5.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.5.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.5 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

sdp.map.origin.5.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.5 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

>> Back to Top

sdp.map.origin.6

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.6.dtmftype and the confgain specified by mpc.sdp.map.origin.6.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.6 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

>> Back to Top

sdp.map.origin.6.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.6.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.6 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

>> Back to Top

sdp.map.origin.6.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.6 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

sdp.map.origin.7

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.7.dtmftype and the confgain specified by mpc.sdp.map.origin.7.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.7 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

sdp.map.origin.7.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.7.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.7 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

sdp.map.origin.7.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.7 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

sdp.map.origin.8

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.8.dtmftype and the confgain specified by mpc.sdp.map.origin.8.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.8 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

sdp.map.origin.8.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.8.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.8 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

sdp.map.origin.8.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.8 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

sdp.map.origin.9

Default Value:
Valid Values: A valid string must be of the format <FQDN or IP>/[session name content]
Changes Take Effect: At start/restart

Specifies the origin to match in the SDP. If origin specified by this parameter matches the SDP, the DTMF type specified by mpc.sdp.map.origin.9.dtmftype and the confgain specified by mpc.sdp.map.origin.9.confgain is used. Possible value is "<origin address>/<session name>". Where <origin address> is either a fully qualtified domain name or a dotted IP address. This value is matched against the address part of the "o=" line. Note, the value should be set to either the fully qualified domain name or a dotted IP address, depending on which form the end point sends in the SDP. If the end point may send either form then two mpc.sdp.map.origin.9 entries can be used, one set to the fully qualified domain name form and one set to the dotted IP address form. Where <session name> is the prefix or the entire content of the SDP "s=" (after the "s=" part) line to match. For example if the value is set to "192.168.0.1/phone-call", it will match on 192.168.0.1 in the address part of the "o=" line and require that the "s=" line start with "s=phone-call". If the <session name> is an empty string, it will match any "s=" line content. If both <origin address> and <session name> matches "o=" and "s=" respectively, then it is considered a match.

sdp.map.origin.9.confgain

Default Value: 100
Valid Values: mpc.sdp.map.origin.9.confgain must be an integer that is greater than or equal to 0 and less than or equal to 1000.
Changes Take Effect: At start/restart

Specifies the input gain factor to apply for the SDP matching connection when joining the conference. The connection's SDP must match the mpc.sdp.map.origin.9 configuration. The value is specified in percentage. 100 will denote no change. 30 will denote a new input volume of 30% of the original volume into the conference. 200 will denote a new input volume twice as high as the original volume.

sdp.map.origin.9.dtmftype

Default Value: INBAND
Valid Values: Choose between: SIPINFO or INBAND.
Changes Take Effect: At start/restart

Specifies the DTMF type to use when mpc.sdp.map.origin.9 matches. This is regardless of whether telephone-event is negotiated or not, and also overrides the mpc.rtp.dtmf.send configuration. SIPINFO - Support sending DTMFs over SIP INFO messages INBAND - Support sending inband audio DTMFs

>> Back to Top

sdp.videobandwidth

Default Value:
Valid Values: A valid bandwidth as specified in RFC4566.
Changes Take Effect: At start/restart

Specifies the value of the bandwidth attribute as specified in RFC4566. If not empty, this value will be added to the bandwidth attribute for the video media description in offerring SDP

>> Back to Top

srtp.cryptomethods

Default Value: AES_CM_128_HMAC_SHA1_80
Valid Values: Any combination of: "AES_CM_128_HMAC_SHA1_80" and "AES_CM_128_HMAC_SHA1_32". Or "none".
Changes Take Effect: At start/restart

List of crypto suites corresponding to advertised capabilities offered by the MCP using SDP. See RFC4568 for the description of the suites.

>> Back to Top

srtp.maxerror

Default Value: 5
Valid Values: mpc.srtp.maxerror must be an integer that is greater than or equal to 1 and less than or equal to the maximum integer as defined by the Genesys Administrator Help.
Changes Take Effect: At start/restart

Specifies the tolerance for SRTP errors in transmission and receiving of packets. When the number of errors is greater than this value, a call may be terminated.

srtp.mode

Default Value: none
Valid Values: none: No SRTP supported: the MCP will ignore the crypto. accept_only: SRTP supported for SDP offers sent to the MCP, no outgoing offers will add SRTP to m-lines that did not previously contain it. offer: SRTP supported for SDP offers received by the MCP and will be including in outgoing SDP offers. If the other side ignores SRTP, the MCP will fall back to non SRTP mode. If a previously negotiated m-line is used in a reoffer, or the far end requests an offer, and that m-line did not have SRTP negotiated, SRTP will NOT be added. If the far end reoffers and adds SRTP to a previously negotiated m-line, SRTP WILL be negotiated. offer_strict: Same as offer, however if the other side doesn't use SRTP, negotiation will fail. offer_selectable: Same as offer except - if an offer has two media lines that are the same except that one has crypto, only the one with crypto will be accepted. In its own offer, two media lines will be offered for each media type, one with crypto and the other without. If both media lines are accepted, all RTP will be sent and received only through the crypto line.
Changes Take Effect: At start/restart

Specifies the srtp mode for the MCP

srtp.sessionparams

Default Value: UNENCRYPTED_SRTP UNENCRYPTED_SRTCP UNAUTHENTICATED_SRTP
Valid Values: Any combination of: "UNENCRYPTED_SRTP", "UNENCRYPTED_SRTCP" and "UNAUTHENTICATED_SRTP". Or "none".
Changes Take Effect: At start/restart

List of session parameters that the MCP is willing to accept. See RFC4568 for their description. Note that RFC4568 doesn't allow unauthenticated srtcp.

srtp.sessionparamsoffer

Default Value:
Valid Values: Any combination of: "UNENCRYPTED_SRTP", "UNENCRYPTED_SRTCP" and "UNAUTHENTICATED_SRTP". Or "none".
Changes Take Effect: At start/restart

List of session parameters that the MCP will include in its SDP offers. See RFC4568 for their description. Note that RFC4568 doesn't allow unauthenticated srtcp.

telephone_event.maxptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP, or answering the SDP where the offer does not have the maxptime, the maxptime attribute will be set according to this configuration. If this configuration does not exist, or is disabled (0), the maxptime attribute will not be sent unless the SDP offer had the maxptime attribute. In the case where other codecs in the SDP also specify maxptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

telephone_event.ptime

Default Value: 0
Valid Values: Choose between: 0, 10, 20, 30, 40, 60, 80 or 100.
Changes Take Effect: Immediately

If the MCP is offering the SDP or answering the SDP where the offer does not have the ptime, the ptime attribute will be set according to this configuration. This configuration is also used as the transmission rate of this codec when the remote SDP does not specify the ptime attribute. Note that transmission rate will default to 20ms if this configuration is disabled. If disabled (0), ptime attribute will not be sent unless the SDP offer had the ptime attribute. In the case where the other codecs in the SDP also specify the configured ptime, the configuration of the codec listed before this codec will take precedence.

>> Back to Top

telephone_eventpayload

Default Value: 101
Valid Values: A valid telephone-event payload can only be an integer from 96 to 127 inclusive
Changes Take Effect: At start/restart

Default telephone-event payload to use by the MCP if none are specified

>> Back to Top

tfcipayload

Default Value: 96
Valid Values: A valid tfci payload can only be an integer from 96 to 127 inclusive
Changes Take Effect: At start/restart

Default payload type number to use for tfci

tiasfraction

Default Value: 100
Valid Values: tiasfraction is a percentage (out of 100) that must be an integer of value 0 or greater.
Changes Take Effect: At start/restart

When the TIAS bandwidth parameter is specified on incoming SDP, mpc.tiasfraction specifies the percentage of the TIAS bitrate that the MPC will try to achieve on the outbound media stream. If tiasfraction is 100 (default) then the MPC will try to limit the media bitrate to TIAS. In some cases it might go slightly over the TIAS limit (by perhaps one or two percent), so for safety it might be better to specify a tiasfraction value somewhat less than 100. It is possible to specify a tiasfraction greater than 100, but this is not recommended.

transcoders

Default Value: G722 GSM G726 G729 AMR AMR-WB MP3 OPUS H263 H264 VP8
Valid Values: Any combination of: G722, GSM, G726, G729, AMR, AMR-WB, MP3, OPUS, H263, H264 and VP8. Or "none".
Changes Take Effect: At start/restart

Specifies the list of transcoders to be used by MPC. Add H263 to allow video transcoding involving H263 codec: - H263 transcoding to/from H264 (note: H264 also has to be enabled). - H263 resolution downscaling. - H263 frame rate throttling. - H263 bit rate throttling. - H263 conference video mixing. - Textoverlay on H263 video Add H264 to allow video transcoding involving H264 codec: - H264 transcoding to/from H263 (note: H263 also has to be enabled). - H264 resolution downscaling. - H264 frame rate throttling. - H264 bit rate throttling. - H264 conference video mixing. - Textoverlay on H264 video Add VP8 to allow video transcoding involving VP8 codec Set to "none" in order to disable all transcoders.

transmitmultiplecodec

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

When media negotiation returns more than one supported codecs, this parameter specifies whether to allow transmission of all supported codecs, or restrict transmission to only one codec. If set to Enable, more than one codec can be transmitted. If set to Disable, only the codec at the top of the negotiated codec list will be transmitted. Note that for SIP devices that support multiple codecs, this parameter must be set to Disable for full call recording to work.

tts.appendrejcodec

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

When set to Enable, the MCP will advertise all supported codecs when generating an SDP offer to the MRCP TTS. Even if codecs are rejected or not presented in the caller's SDP, the MCP will still support receiving these codecs. The MCP will not send for those SDPs unless a payload is presented by the caller.

tts.codec

Default Value: pcmu
Valid Values: Any combination of: pcmu, pcma, g722, g726, g729, gsm, amr, amr-wb and tfci.
Changes Take Effect: Immediately

List of codec corresponding to advertised capabilities offered by the MCP to the MRCP TTS server using SDP. The offered codec list will control the codecs that are offered by the MCP to the remote party for media sent from the remote party to Genesys.

tts.preferredipinterface

Default Value: V4
Valid Values: Choose between: V4 or V6
Changes Take Effect: At start/restart

Specifies the preferred IP interface to use (IPv4 or IPv6) for MRCP TTS when performing SDP negotiation. In particular, this will be used to set the root connection attribute in SDP answers, and set the connection attribute in SDP offers.

tts.srtp.cryptomethods

Default Value: AES_CM_128_HMAC_SHA1_80
Valid Values: Any combination of: "AES_CM_128_HMAC_SHA1_80" and "AES_CM_128_HMAC_SHA1_32". Or "none".
Changes Take Effect: At start/restart

List of crypto suites corresponding to advertised capabilities offered by the MCP to the MRCP TTS server using SDP. See RFC4568 for the description of the suites.

tts.srtp.mode

Default Value: none
Valid Values: none: No SRTP supported: the MCP will ignore the crypto. offer: SRTP supported in outgoing SDP offers. If the other side ignores SRTP, the MCP will fall back to non SRTP mode. offer_strict: Same as offer, however if the other side doesn't use SRTP, negotiation will fail.
Changes Take Effect: At start/restart

Specifies the srtp mode for the MCP to use for MRCP TTS sessions

tts.srtp.sessionparamsoffer

Default Value:
Valid Values: Any combination of: "UNENCRYPTED_SRTP", "UNENCRYPTED_SRTCP" and "UNAUTHENTICATED_SRTP". Or "none".
Changes Take Effect: At start/restart

List of session parameters that the MCP will include in its SDP offers to the MRCP TTS server. See RFC4568 for their description. Note that RFC4568 doesn't allow unauthenticated srtcp.

validatemediatimers

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

When enabled, all media thread timers would be validated. This is useful for scenario where clock skew prevents media threads from waking up at regular intervals.

videotranscoder.bitratecheckdelay

Default Value: 10000
Valid Values: A valid value should be an integer greater or equal to 0
Changes Take Effect: At start/restart

This parameter specifies the bit rate check delay when bit rate check is enabled for video transcoding. Bit rate checking will start after the milliseconds specified by this parameter elapses.

videotranscoder.bitratechecktolerance

Default Value: 50
Valid Values: A valid value should be an integer greater or equal to 0
Changes Take Effect: At start/restart

This parameter specifies the bit rate check tolerance when bit rate check is enabled for video transcoding. Bit rate checking will allow the bit rate to go over the maximum by the percentage specified by this parameter.

videotranscoder.checkbitrate

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

When set to true and the transcoder for the incoming video format is enabled, video transcoding will be triggered when bit rate exceeds the maximum bit rate. When false, it will not.

videotranscoder.checkframerate

Default Value: 1
Valid Values: Choose between: 0 or 1
Changes Take Effect: Immediately/session

When set to true and the transcoder for the incoming video format is enabled, video transcoding will be triggered when frame rate exceeds the maximum frame rate. When false, it will not.

videotranscoder.frameratechecktolerance

Default Value: 50
Valid Values: A valid value should be an integer greater or equal to 0
Changes Take Effect: At start/restart

This parameter specifies the frame rate check tolerance when frame rate check is enabled for video

transcoding. Frame rate checking will allow the frame rate to go over the maximum by the percentage specified by this parameter.

videotranscoder.h264.keyframeidrinterval

Default Value: 1
Valid Values: A number between 0 and 2147483647 inclusive.
Changes Take Effect: At start/restart

This parameter specifies the IDR frame generation frequency of the H264 transcoder. IDR frame is one of types of i-frames in H264. For example, if 2 is specified, every other i-frame generated will be an IDR frame.

videotranscoder.h264.keyframeinterval

Default Value: 50
Valid Values: A number between 0 and 2147483647 inclusive.
Changes Take Effect: At start/restart

This parameter specifies the i-frame generation frequency of the H264 transcoder. For example, if 50 is specified, an i-frame will be generated per every 50 frames.

videotranscoder.h264.resolutions

Default Value: SQCIF QCIF QVGA CIF VGA 4CIF SVGA 720P
Valid Values: Any combination of: SQCIF, QCIF, QVGA, CIF, VGA, 4CIF, SVGA, 720P and custom resolution "WidthxHeight".
Changes Take Effect: At start/restart

This parameter specifies the list of H264 encodable resolutions when H264 transcoding is applied. If empty, defaults to "SQCIF QCIF QVGA CIF VGA 4CIF SVGA 720P". ITU-T H264 Recommendation document Table A.6 specifies the list of resolution/frame rate/bit rate limits per level. If the resolution needs to be downscaled because of the level requirement, the resolution closest downward in this list will be selected. Format: <resolution> <resolution> [<resolution> ...] Where <resolution> is <width>x<height> or one of the following keywords - SQCIF - Sub-QCIF resolution (128x96) QCIF - QCIF resolution (176x144) QVGA - QVGA resolution (320x240) CIF - CIF resolution (352x288) VGA - VGA resolution (640x480) 4CIF - 4CIF resolution (704x576) SVGA - SVGA resolution (800x600) 720P - 720P HD resolution (1280x720) When <width>x<height> syntax is used, the resolution must be less than or equal to 720P HD resolution. WARNING: MCP will fail to start if invalid resolution is specified.

videotranscoder.maxbitrate

Default Value: 500000
Valid Values: A valid value should be an integer greater or equal to 0
Changes Take Effect: Immediately/session

This parameter specifies maximum bit rate used for encoding when video transcoding is active. 0 indicates that there is no maximum imposed. If not 0, video transcoded output encoding bit rate is set to the minimum of the maximum allowed bitrate of the receiver and the value specified by this parameter.

>> Back to Top

videotranscoder.statsresetthreshold

Default Value: 60000
Valid Values: A number between 1000 and 2147483647 inclusive.
Changes Take Effect: At start/restart

This parameter specifies the accumulated duration (in milliseconds) threshold at which to trigger resetting of frame rate/bit rate statistics. For example, if set to 60000, the cumulative duration used for calculating frame rate/bit rate is reset approximately every 60 seconds. The statistics is used for checking and triggering video transcoding when frame rate/bit rate exceeds the required maximum. Lower value makes the transcoding trigger more sensitive to sudden bursts that exceeds the required maximum.

>> Back to Top

voipmetrics.enable

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

When enabled, MCP will collect several metrics defined in RFC 3611 for each audio session. The metrics can be divided into local and remote. For local metrics, MCP collects some of them by exchanging RTCP messages between itself and the remote party while some are calculated locally from ongoing activities. For remote metrics, the remote party, if supporting RFC 3611, will report MCP about its metrics periodically and MCP will record them whenever it receives an update.

At the end of each audio session, MCP users also have an option to publish the local and remote metrics to a destination. Refer to voipmetrics.* configuration under sip section for more information.

>> Back to Top

vp8.adaptive

Default Value: true
Valid Values: The value must be a boolean of either true or false.
Changes Take Effect: At start/restart

This parameter specifies the adaptivity of the VP8 encoder behaviour, and thus the determinism of its output. In adaptive mode (the default) the encoder will adapt to environmental conditions to provide the best overall user experience. Non-adaptive mode will result in the same output everytime, but will not adapt to environmental situations.

>> Back to Top

vp8.defaultbitrate

Default Value: 0
Valid Values: The bitrate value must be an integer of value 0 or greater.
Changes Take Effect: At start/restart

This parameter specifies the output bitrate in bits/sec for transcoding to the VP8 format. If set to 0 the default bitrate of the VP8 encoder will be used.
WARNING: MCP will fail to start if an invalid value is specified.

>> Back to Top

vp8.defaultframerateden

Default Value: 1001
Valid Values: The framerate numerator value must be 1 or greater.
Changes Take Effect: At start/restart

This parameter specifies the output framerate denominator for transcoding to the VP8 format. The framerate numerator and denominator values are combined to determine the framerate (e.g. 30000/1001 gives 29.97 frames/sec).
WARNING: MCP will fail to start if an invalid value is specified.

>> Back to Top

vp8.defaultframeratenum

Default Value: 30000
Valid Values: The framerate numerator value must be 1 or greater.
Changes Take Effect: At start/restart

This parameter specifies the output framerate numerator for transcoding to the VP8 format. The framerate numerator and denominator values are combined to determine the framerate (e.g. 30000/

1001 gives 29.97 frames/sec).
WARNING: MCP will fail to start if an invalid value is specified.

vp8.defaultresolution

Default Value: CIF
Valid Values: Choose between: SQCIF, QCIF, QVGA, CIF, VGA, 4CIF, SVGA, 720P or custom resolution "WidthxHeight".
Changes Take Effect: At start/restart

This parameter specifies the output resolution for transcoding to the VP8 format.

The value is specified by a keyword or width and height value as follows:
SQCIF - Sub-QCIF resolution (128x96)
QCIF - QCIF resolution (176x144)
QVGA - QVGA resolution (320x240)
CIF - CIF resolution (352x288)
VGA - VGA resolution (640x480)
4CIF - 4CIF resolution (704x576)
SVGA - SVGA resolution (800x600)
720P - 720P HD resolution (1280x720)
WidthxHeight - specifies a custom width and height

When WidthxHeight syntax is used, the resolution must be less than or equal to 720P HD resolution.
WARNING: MCP will fail to start if invalid resolution is specified.

vp8.maxkeyframeinterval

Default Value: 15
Valid Values: The value must be nonnegative.
Changes Take Effect: Immediately/session

This parameter, expressed as a number of frames, forces the encoder to code a keyframe if the last keyframe was vp8.maxkeyframeinterval frames ago. A value of 0 or 1 implies all frames will be keyframes.

vrmrecorder.codec

Default Value: pcmu pcma g722 opus g726 g729 gsm amr amr-wb h263 h263-1998 h264 vp8 telephone-event
Valid Values: Any combination of: pcmu, pcma, g722, opus, g726, g729, gsm, amr, amr-wb, h263,

h263-1998, h264, vp8, or telephone-event.
Changes Take Effect: At start/restart

Specifies a list of codecs supported by MCP for VRM recorder. This option is used to limit supported codecs on the RTP streams sent to the recorder. It is similar to the main "codec" parameter, except in this case, the codecs and order in the offer are dictated by the codecs negotiated on the inbound call legs. By restricting the codecs in this list, a streaming issue on recorder streams caused by lack of transcoding support could be avoided, where the recorder sends a SIP session refresh with an SDP that changes the codec priority order from the original MCP offer.

>> Back to Top

vrmrecorder.enable

Default Value: true
Valid Values: Choose between: true or false
Changes Take Effect: At start/restart

When set to true, the vrmrecorder for the CRQM feature will be enabled. When false, or not set, it will not be loaded.

>> Back to Top

vrmrecorder.preferredipinterface

Default Value: V4
Valid Values: Choose between: V4 or V6
Changes Take Effect: Immediately

Specifies the preferred IP interface to use (IPv4 or IPv6) when performing SDP negotiation (CRQM related). In particular, this will be used to set the root connection attribute in SDP answers, and set the connection attribute in SDP offers.

>> Back to Top

vrmrecorder.srtp.cryptomethods

Default Value: AES_CM_128_HMAC_SHA1_80
Valid Values: Any combination of: "AES_CM_128_HMAC_SHA1_80" and "AES_CM_128_HMAC_SHA1_32". Or "none".
Changes Take Effect: At start/restart

List of crypto suites corresponding to advertised capabilities offered by the MCP to a recording server using SDP. See RFC4568 for the description of the suites.

>> Back to Top

vrmrecorder.srtp.mode

Default Value: none
Valid Values: none: No SRTP supported: the MCP will ignore the crypto. offer: SRTP supported in outgoing SDP offers. If the other side ignores SRTP, the MCP will fall back to non SRTP mode. offer_strict: Same as offer, however if the other side doesn't use SRTP, negotiation will fail. offer_selectable: Same as offer except - two media lines will be offered for each media type, one with crypto and the other without. If both media lines are accepted, all RTP will be sent and received only through the crypto line.
Changes Take Effect: At start/restart

Specifies the srtp mode for the MCP to use for recording sessions

>> Back to Top

vrmrecorder.srtp.sessionparamsoffer

Default Value:
Valid Values: Any combination of: "UNENCRYPTED_SRTP", "UNENCRYPTED_SRTCP" and "UNAUTHENTICATED_SRTP". Or "none".
Changes Take Effect: At start/restart

List of session parameters that the MCP will include in its SDP offers to a recording server. See RFC4568 for their description. Note that RFC4568 doesn't allow unauthenticated srtcp.

>> Back to Top

widebandconferences

Default Value: 0
Valid Values: Choose between: 0 or 1
Changes Take Effect: At start/restart

When enabled and the conference's initial audio requirements are unknown, MCP will mix the conference's audio in wideband. This will generally increase conference sound quality at the cost of increased CPU usage.

>> Back to Top