



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

## Genesys Engage Digital (eServices)

[settings](#)

5/1/2025

## settings

- allow-duplicated-kvp
- async-chat-enforce
- async-idle-alert
- async-idle-close
- async-idle-notice
- attach-session-statistics
- attach-stats-rep-events
- attach-stats-rep-place
- connection-down-report-scs
- flex-disconnect-timeout
- flex-no-agents-timeout
- flex-push-content
- flex-push-enabled
- flex-push-on-join
- flex-push-resend-attempts
- flex-push-resend-delay
- flex-push-timeout
- hide-attached-data
- identify-create-contact
- illegal-xml-char-mask
- ixn-resend-attempts
- ixn-resend-delay
- ixn-submittedby-name
- limit-average-interval
- limit-for-flex-users
- limit-for-reply-delay
- limit-for-sessions
- limits-control-enabled
- limits-reached-report-scs
- limits-restore-threshold
- log-suppress-ping-activity
- max-waiting-requests
- message-log-print-size
- report-silent-monitoring
- security-token-size
- send-typing-notice-back
- server-reply-timeout-ixn
- server-reply-timeout-ucs
- server-session-nickname
- session-password-enforce
- session-restoration-gap
- session-restoration-mode
- session-restore-do-purge
- session-restore-extend-by
- session-restore-push-send
- shutdown-notify-flex-push
- stop-abandoned-interaction
- transcript-auto-save
- transcript-auto-ucs-mode
- transcript-cleanup-action
- transcript-cleanup-apply
- transcript-cleanup-mask
- transcript-cleanup-typing
- transcript-content-control
- transcript-resend-attempts
- transcript-resend-delay
- transcript-save-notice
- transcript-save-on-error
- use-contact-server
- user-register-timeout
- web-user-max-messages
- xml-request-max-size

### allow-duplicated-kvp

**Default Value:** true

**Valid Values:** true, false

**Changes Take Effect:** Immediately

**Introduced:** 8.5.201.05

With a value of "false", Chat Server will return an error when any duplicated key-value pairs (for example, pairs with the same key name) in userdata or event attributes in chat protocol requests are provided. With a value of "true", Chat Server will ignore such conditions (and will use the value from the latest key-value pair).

**Attention:** While the backward compatibility default value is set to "true", it is recommended it be set to "false" for more controlled behavior.

## async-chat-enforce

**Default Value:** none

**Valid Values:** none, flex, basic, both

**Changes Take Effect:** Immediately

**Introduced:** 8.5.307.04

Specifies when Chat Server forces chat sessions to run in async mode.

- **none:** no forcing is applied.
- **flex:** chat sessions run in async mode, when started via flex chat protocol.
- **basic:** chat sessions run in async mode, when started via basic chat protocol.
- **both:** all chat sessions run in async mode.

## async-idle-alert

**Default Value:** 86400

**Valid Values:** Any integer from 1-1728000

**Changes Take Effect:** Immediately

**Introduced:** 8.5.202.09 (Restricted); 8.5.301.06 (Generally Available)

Specifies an inactivity alert timeout (in seconds) which is enabled only for async chat sessions. The inactivity timeout is set (or reset) for a session after any of the following activities: agent joined or left, any chat participant sent a message or a notice (as defined by "async-idle-notices"). If no qualifying activity is detected during this timeout, Chat Server:

- Sends the IDLE\_CONTROL\_ALERT notice with a message specified by the value of the "message-alert" option (in the section [inactivity-control]).
- Updates the interaction properties: GCTI\_Chat\_AsyncStatus=3 and GCTI\_Chat\_AsyncCheckAt={time to check interaction at}.
- Starts the timeout specified by the value of the "async-idle-close" option.

## async-idle-close

**Default Value:** 3600

**Valid Values:** Any integer from 1-1728000

**Changes Take Effect:** Immediately

**Introduced:** 8.5.202.09 (Restricted); 8.5.301.06 (Generally Available)

Specifies, in seconds, a timeout that starts after "async-idle-alert" (enabled only for async chat sessions). If any qualifying activity occurs, (see "async-idle-alert" for a description of what qualifies as activity), the timeout stops and "async-idle-alert" timer is reset. If no qualifying activity is detected during this timeout, Chat Server:

- Sends the IDLE\_CONTROL\_CLOSE notice with a message specified by the value of the "message-close" option (in the section [inactivity-control]).
- Closes the chat session.

## async-idle-notices

**Default Value:** major

**Valid Values:** none, major, major2, typing, all

**Changes Take Effect:** Immediately

**Introduced:** 8.5.202.09 (Restricted); 8.5.301.06 (Generally Available)

**Modified:** 8.5.310.09

Specifies which notifications count as async chat session activity for the purpose of inactivity control.

- **all:** All user notifications are included (custom, push URL, typing on/off, update nickname).
- **major:** Only major notifications are included (push URL, file uploaded, file deleted).
- **major2:** Extends the major value with custom notifications.
- **none:** All notifications are excluded.
- **typing:** Only major and "typing on" notifications are included.

## attach-session-statistics

**Default Value:** none

**Valid Values:** all, fired, none

**Changes Take Effect:** Immediately

Enables calculating various statistics and attaching them to the interaction's user data at the end of the chat session. Possible values are:

- **none:** Do not attach anything.
- **all:** Attach all possible statistics (both encountered and non-encountered).
- **fired:** Attach only statistics that were encountered during the course of the chat session.

**Note:** Genesys recommends that you change the value of this option from the default value of "none" to "fired" if you deploy Historical Reporting for either Chat or Bot Gateway.

## attach-stats-rep-events

**Default Value:** 0

**Valid Values:** Any integer from 0-7

**Changes Take Effect:** Immediately

**Introduced:** 8.5.315.05

Enables the delivery of historical reporting data through custom reporting events. This type of delivery allows multiple updates through each chat session, versus only a single delivery at the end of a chat session through the userdata. Through this option, you can specify when the delivery of the reporting data must occur. The value of the option can be set to the sum of the following integers:

- **0:** The functionality is disabled (default value).
- **1:** Custom Reporting Event with reporting data is sent instead of a userdata update at the end of a chat session.
- **2:** Custom Reporting Event with reporting data is sent when async chat is placed on hold (when GCTI\_Chat\_AsyncStatus is set to -2).
- **4:** Custom Reporting Event with reporting data is sent when a message from the customer arrives during the chat session hold (when GCTI\_Chat\_AsyncStatus is set to 2).

For example, to force Chat Server to deliver reporting data for all possible cases, set the option to a value of 7, which is the sum of 1, 2 and 4. Attention: if this functionality is enabled, make sure that the option attach-stats-rep-place is set to the correct value.

## attach-stats-rep-place

**Default Value:** 0

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

**Introduced:** 8.5.315.05

If attach-stats-rep-events is set to a non-zero value, this option must contain the DBID (numeric value) of a configured place. This parameter is required for backward compatibility of GIM operations. Creating a simulated/fake place for this purpose is recommended. The same place can be shared between multiple instances of Chat Server.

## connection-down-report-scs

**Default Value:** true

**Valid Values:** true, false

**Changes Take Effect:** Immediately

With a value of 'false', Chat Server does not send a 'SERVICE UNAVAILABLE' notification to Solution Control Server when Chat Server detects disconnection from Interaction Server and/or UCS. This can prevent Chat Server from switching over when configured as primary/backup.

## flex-disconnect-timeout

**Default Value:** 45  
**Valid Values:** Any integer from 1-1728000  
**Changes Take Effect:** Immediately  
**Modified:** 8.5.301.06

Specifies timeout (in seconds) after which Chat Server disconnects an inactive web (flex) chat client. Inactivity is defined as the absence of protocol requests, not the absence of messages.

## flex-no-agents-timeout

**Default Value:** 0  
**Valid Values:** Any integer from 0-1728000  
**Changes Take Effect:** Immediately  
**Modified:** 8.5.308.06

Specifies timeout (in seconds) after which Chat Server disconnects a web (flex) chat participant if there are no agents in the chat session. After timeout expires, the timer fires (in other words, activates) only upon a subsequent request from a chat participant. Setting this option to "0" disables this functionality.

## flex-push-content

**Default Value:** session-id, user-id  
**Valid Values:** Any combination of: app-dbid, secure-key, session-id, user-id  
**Changes Take Effect:** Immediately  
**Introduced:** 8.5.311.06

Defines the content for flex-push notifications. It can include any combination of the following values: app-dbid, secure-key, session-id, user-id.

## flex-push-enabled

**Default Value:** true  
**Valid Values:** true, false  
**Changes Take Effect:** Immediately  
**Modified:** 8.5.308.06

Enables or disables new connection registrations for push-flex notifications. Push-flex notifications are used by the web API for CometD and push messaging on either mobile devices or HTTP server.

## flex-push-on-join

**Default Value:** false  
**Valid Values:** true, false  
**Changes Take Effect:** Immediately

**Introduced:** 8.5.315.05

Prohibits or allows a push-flex notification subscription during the chat session creation (via REST "Chat API Version 2").

## flex-push-resend-attempts

**Default Value:** 10

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

**Introduced:** 8.5.308.06

Specifies how many times Chat Server tries to resend a flex-push notification when the resend functionality is requested by a chat participant. Chat Server stops resending flex-push notification when either a delivery confirmation is being received or the total specified number of attempts is reached. Setting this option to "0" completely disables resending these flex-push notification (except for the first notice).

## flex-push-resend-delay

**Default Value:** 15

**Valid Values:** Any integer from 1-86400

**Changes Take Effect:** Immediately

**Introduced:** 8.5.308.06

Specifies how long, in seconds, Chat Server waits before resending flex-push notification when resend functionality is requested by a chat participant. Chat Server continues to resend flex-push notification until it reaches the total specified number of times, specified by flex-push-resend-attempts.

## flex-push-timeout

**Default Value:** 300

**Valid Values:** Any integer from 1-1728000

**Changes Take Effect:** Immediately

**Introduced:** 8.5.202.09 (Restricted); 8.5.301.06 (Generally Available)

**Modified:** 8.5.310.09

Specifies timeout (in seconds) after which Chat Server sends ping notification for push-flex connected client. If no request from the client is submitted within the next "flex-push-timeout", Chat Server activates flex-disconnect-timeout. Inactivity is defined as the absence of protocol requests, not the absence of messages

## hide-attached-data

**Default Value:** true

**Valid Values:** true, false

**Changes Take Effect:** Immediately

Prohibits or allows the printing of (attached) user data in the log output. A value of true prohibits printing.

## identify-create-contact

**Default Value:** -1

**Valid Values:** -1, 0, 1, 2, 3

**Changes Take Effect:** Immediately

If this option is set to '-1', Chat Server searches user data for 'IdentifyCreateContact' and sends the value of that parameter to UCS to determine whether to carry out contact identification. If this option is set to '0', Chat Server does not send the parameter 'IdentifyCreateContact' to UCS. For all other values (1-3) Chat Server sends those values to UCS (thus overwriting any value specified by 'IdentifyCreateContact' in the user data).

Note: UCS accepts the following values for 'IdentifyCreateContact':

- **1:** No attempt to identify the contact and no creation of a contact.
- **2:** Contact identification is attempted but if no contact is found a new one is not created.
- **3 (default):** Contact identification is attempted and if no contact is found a new one is created.

## illegal-xml-char-mask

**Default Value:** \*

**Valid Values:** Any character

**Changes Take Effect:** Immediately

Specifies the character that replaces illegal XML characters in incoming data from Chat Server clients. If more than one character is specified, the first one is used. If this option is empty, Chat Server does not replace illegal characters. Note that the presence of illegal characters could lead to failure of the request, followed by an error reply from Chat Server.

## ixn-resend-attempts

**Default Value:** 10

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

Specifies how many times Chat Server resends "Update" and "Stop" requests to Interaction Server. If a request fails due to a recoverable reason or because the timeout expires, Chat Server continues to send the request until it reaches the total specified number of attempts. Debugging purposes only: Set this option to "0" to completely disable sending these requests to Interaction Server.



## ixn-resend-delay

**Default Value:** 1

**Valid Values:** Any integer from 0-86400

**Changes Take Effect:** Immediately

Specifies how long, in seconds, Chat Server waits before resending "Update" and "Stop" requests to Interaction Server for the first time if the initial request failed due to a recoverable reason or because the timeout expired. For each subsequent resend, the current delay is multiplied by two. For example, if the current delay is four seconds and a resend occurs, the following delay is set to eight seconds. If there is another resend, the delay is set to 16 seconds. Set this option to "0" to disable requests resending.

## ixn-submittedby-name

**Default Value:**

**Valid Values:** Any valid string value

**Changes Take Effect:** After restart

**Introduced:** 8.5.316.02

Specifies the value for the parameter SubmittedBy that Chat Server uses during the registration in Interaction Server. Using the same value for Chat Server applications forces Interaction Server to distribute EventProcessingStopped for all Chat Server instances instead of sending it only to the application which initially submitted the interaction.

## limit-average-interval

**Default Value:** 15

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

Specifies the interval, in seconds, for calculating the average time it takes Chat Server to process an incoming request. A value of "0" disables the calculation. This interval is used to calculate the value for the option "limit-for-reply-delay".

## limit-for-flex-users

**Default Value:** 1001

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

Defines the maximum number of flex users that Chat Server handles simultaneously.

## limit-for-reply-delay

**Default Value:** 2000

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

Specifies the maximum possible delay, in milliseconds, required for Chat Server to process an incoming request. The delay is calculated as an average value over an interval defined by the option "limit-average-interval".

## limit-for-sessions

**Default Value:** 1000

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

Defines the maximum number of chat sessions that Chat Server handles simultaneously.

## limits-control-enabled

**Default Value:** false

**Valid Values:** true, false

**Changes Take Effect:** Immediately

Enables (true) or disables (false) load restriction for Chat Server, controlled by options: limit-for-flex-users, limit-for-reply-delay, limit-for-sessions.

## limits-reached-report-scs

**Default Value:** true

**Valid Values:** true, false

**Changes Take Effect:** Immediately

With a value of "false" Chat Server does not send "SERVICE UNAVAILABLE" notifications to Solution Control Server when Chat Server reaches any of its configured load limits. This can prevent Chat Server from switching over when configured as primary/backup.

## limits-restore-threshold

**Default Value:** 80

**Valid Values:** Any integer from 0-100

**Changes Take Effect:** Immediately

Specifies the level (as percentage of maximum value) which metric must drop below in order to restore the service.

## log-suppress-ping-activity

**Default Value:** never

**Valid Values:** never, always, selective

**Changes Take Effect:** Immediately

Enables omission of flex protocol "ping" (i.e. empty Refresh request/replies) activity from logs.

- **never:** Do not suppress "ping" activity.
- **always:** Completely suppress "ping" activity.
- **selective:** Suppress "ping" activity except for non-empty replies (flex eventStatus).

## max-waiting-requests

**Default Value:** 500

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

Specifies the maximum number of sessions which reside in a waiting state after being created. A waiting state is defined as waiting for an agent to join the chat session. A value of 0 means that Chat Server will not create any new sessions.

## message-log-print-size

**Default Value:** 0

**Valid Values:** Any integer from 0-7000

**Changes Take Effect:** Immediately

Specifies the number of characters from the whole client message that the log prints, starting from the beginning of the message.

## report-silent-monitoring

**Default Value:** none

**Valid Values:** none, all, agents, clients

**Changes Take Effect:** Immediately

Specifies how Chat Server notifies chat session participants that the chat is being monitored. Possible

values:

- **none**: notification is disabled.
- **all**: notifies both agent and client participants.
- **agents**: notifies only agent participants.
- **clients**: notifies only client participants.

## security-token-size

**Default Value:** 20

**Valid Values:** Any integer from 1-1000

**Changes Take Effect:** Immediately

Specifies the number of characters for crypto-random generated security tokens used in chat protocols (the secureKey property).

## send-typing-notice-back

**Default Value:** true

**Valid Values:** true, false

**Changes Take Effect:** Immediately

Enables/disables sending typing notification event in transcript which is sent back to the client.

**Attention:** For backward compatibility value must be set to "true".

## server-reply-timeout-ixn

**Default Value:** 35

**Valid Values:** Any integer from 1-86400

**Changes Take Effect:** Immediately

Specifies how long (in seconds) Chat Server waits for a reply from Interaction Server. If no reply is received within the specified timeout, Chat Server considers the corresponding request to have failed.

## server-reply-timeout-ucs

**Default Value:** 35

**Valid Values:** Any integer from 1-86400

**Changes Take Effect:** Immediately

Specifies how long (in seconds) Chat Server waits for a reply from UCS. If no reply is received within the specified timeout, Chat Server considers the corresponded request to have failed.

## server-session-nickname

**Default Value:** system

**Valid Values:** Any string

**Changes Take Effect:** Immediately

A name to impersonate Chat Server in chat sessions when Chat Server performs specific actions on the chat session (such as inactivity control monitoring).

## session-password-enforce

**Default Value:** never

**Valid Values:** never, all, flex, clients, agents

**Changes Take Effect:** Immediately

**Modified:** 8.5.301.06

Specifies how Chat Server enforces the password when a participant joins the chat session. Possible values:

- **never:** session password verification is disabled.
- **all:** require password when any user joins a session.
- **flex:** require password when flex protocol user joins a session.
- **clients:** require password when a client (flex protocol user or basic protocol user of type "CLIENT") joins a session.
- **agents:** require password when an agent (basic protocol user of type "AGENT") joins a session.

## session-restoration-gap

**Default Value:** 100

**Valid Values:** Any integer from 1-1000000

**Changes Take Effect:** Immediately

Specifies the maximum gap between adjacent event IDs allowed when restoring a chat session from the UCS transcript (in HA mode). If the gap exceeds the specified value (that is, too many events are missing), Chat Server does not restore the chat session. This option allows for additional control of the integrity of chat transcripts.

## session-restoration-mode

**Default Value:** none

**Valid Values:** none, simple  
**Changes Take Effect:** Immediately

Allows ("simple") or prohibits ("none") Chat Server's processing of requests for session restoration (when Chat Server is working in high availability mode).

## session-restore-do-purge

**Default Value:** false  
**Valid Values:** true, false  
**Changes Take Effect:** Immediately  
**Introduced:** 8.5.312.10

Enables or disables sending purging requests to Chat Server which handles the chat session before it is restored in this instance.

## session-restore-extend-by

**Default Value:** none  
**Valid Values:** none, agent, esp, both  
**Changes Take Effect:** Immediately  
**Introduced:** 8.5.312.10  
**Modified:** 8.5.316.02

Specifies how Chat Server enables the session restoration from agent desktop and workflow:

- **none:** The functionality is disabled.
- **agent:** Allows Agent Desktop to start a session restoration.
- **esp:** Allows the Workflow to start a session restoration.
- **both:** Allows both Agent Desktop and Workflow to start a session restoration.

## session-restore-push-send

**Default Value:** false  
**Valid Values:** true, false  
**Changes Take Effect:** Immediately  
**Introduced:** 8.5.315.05

Enables or disables resuming push notifications after a session restoration from agent desktop and workflow as defined by the configuration option session-restore-extend-by. When enabled, flex-push-timeout (instead of flex-disconnect-timeout) is used after a session restoration and push notifications are resumed.

## shutdown-notify-flex-push

**Default Value:** 0

**Valid Values:** 0, 1, 2

**Changes Take Effect:** Immediately

**Introduced:** 8.5.315.05

If this option is set to 1 (for cluster and non-cluster mode) or 2 (cluster mode only) and the customer has subscribed to push notifications, Chat Server sends a push notification that the customer has left the chat session during the Chat Server shutdown. If set to 0, no push-notifications will be sent.

## stop-abandoned-interaction

**Default Value:** never

**Valid Values:** always, never, noagents, noanswer

**Changes Take Effect:** Immediately

**Modified:** 8.5.301.06

Specifies how Chat Server handles chat interactions (in Interaction Server) when the chat session is finished. Possible values:

- **always:** interactions are always stopped when the chat session is finished (not recommended value).
- **never:** interactions are never stopped by Chat Server (recommended value).
- **noagents:** interactions are stopped only if there are currently no agents or supervisors in the session (use value wisely).
- **noanswer:** interactions are stopped only if no agents ever joined the session - in other words, the session was never answered (use value wisely).

**Note:** When the chat session is closing, it ignores the presence of bot participants when evaluating this option (in other words, bot participants are not considered "agents"). The legacy "true" and "false" values are also supported and converted to new values as following:

- "false" is interpreted as "never"
- "true" is interpreted as "noanswer"

By default, Chat Server does not stop any interaction after the chat session is finished. Instead, Chat Server updates the IsOnline interaction property to false and attaches reporting statistics. If your deployment includes Genesys Info Mart reporting, Genesys recommends that you set the value of this option to "noanswer", to enable accurate reporting of abandoned chats. In all other deployments, Genesys recommends that you set the value of this option to "never", and have the agent desktop and/or a workflow stop the interaction. Values other than "never" must be used cautiously, as stopping the interaction from Chat Server may lead to conflicts with updates from the agent desktop.

## transcript-auto-save

**Default Value:** 0

**Valid Values:** 0, 1, 2

**Changes Take Effect:** Immediately

If this option is set to 1 (recommended for HA) or 2, Chat Server sends the updated chat session transcript to Universal Contact Server after each submitted message. If set to 0, no UCS records are updated until the chat session ends. For the value 2, Chat Server also sends a special TRANSCRIPT\_SAVE\_DONE notification for basic clients (which can be used in custom agent desktops only).

## transcript-auto-ucs-mode

**Default Value:** 0

**Valid Values:** Any integer from 0-7

**Changes Take Effect:** Immediately

**Introduced:** 8.5.314.02

**Modified:** 8.5.315.05

Only when Cassandra HA is enabled and only for async chat, this option forces Chat Server to save the current content of the chat session transcript to UCS (along with saving it in Cassandra), when a qualified event is produced within the async chat session. The value of the option can be set to the sum of the following integers:

- **0:** The functionality is disabled (default value).
- **1:** Chat Transcript is sent to UCS when async chat is placed on hold (GCTI\_Chat\_AsyncStatus=-2).
- **2:** Chat Transcript is sent to UCS when agent desktop disconnects (GCTI\_Chat\_AsyncStatus=4).
- **4:** Chat Transcript is sent to UCS when agent transfers async chat (GCTI\_Chat\_AsyncStatus=5).

For example, to force Chat Server to save the chat session transcript to UCS when the chat is placed on hold, as well as when the agent transfers the chat, the value must be set to 5, which is the sum of 1 and 4.

## transcript-cleanup-action

**Default Value:** replace-digits

**Valid Values:** none, replace-all, replace-digits

**Changes Take Effect:** Immediately

**Modified:** 8.5.103

Specify how sensitive data is masked out. Possible values:

- **none:** no data is masked out, but Chat Server prints a notice in the log that sensitive data was found.
- **replace-all:** all symbols in the substring (matched by regular expression) are replaced.
- **replace-digits:** only digits in the substring (matched by regular expression) are replaced.

In release 8.5.103, this option was renamed **default-spec** and moved to the **[transcript-cleanup]** section.



## transcript-cleanup-apply

**Default Value:** never

**Valid Values:** always, history-final, never

**Changes Take Effect:** Immediately

**Modified:** 8.5.103

Enables/disables masking out sensitive data in chat session messages. Possible values:

- **always:** mask out sensitive data both in real-time and in transcript written to UCS.
- **history-final:** mask out sensitive data only in transcript written to UCS.
- **never:** disable this functionality.

In release 8.5.103, this option was renamed **apply-area** and moved to the **[transcript-cleanup]** section.

## transcript-cleanup-mask

**Default Value:** An asterisk (\*)

**Valid Values:** Any character

**Changes Take Effect:** Immediately

**Modified:** 8.5.103

Specifies the character that replaces sensitive data. If more than one character is specified, the first one is used.

In release 8.5.103, this option was renamed **default-rep-char** and moved to the **[transcript-cleanup]** section.

## transcript-cleanup-typing

**Default Value:** none

**Valid Values:** any, message, none, notice

**Changes Take Effect:** Immediately

**Discontinued:** 8.5.103 (replaced by typing-preview in the transcript-cleanup section)

Enables/disables masking out sensitive data in typing notifications only (by replacing all digits with the character specified by "transcript-cleanup-mask"). Possible values:

- **any:** mask out sensitive data both in message part and notice part of notification.
- **message:** mask out sensitive data only in message part of notification.
- **none:** disable this functionality.
- **notice:** mask out sensitive data only in notice part of notification.

In release 8.5.103, this option was renamed **typing-preview** and moved to the **[transcript-cleanup]** section.

## transcript-content-control

**Default Value:** 0

**Valid Values:** Any integer from 0-3

**Changes Take Effect:** Immediately

**Introduced:** 8.5.315.05

Enables or disables different chat session transcript content features. The value of the option can be set to the sum of the following integers:

- **0:** The functionality is disabled (default value).
- **1:** Messages and notices with '{"content-control":{"exclude-store":"true"}}' in eventAttributes are not saved into the chat transcript (in UCS and/or Cassandra).
- **2:** Messages and notices with '{"content-control":{"expire-sec":"<seconds>"}}' in eventAttributes are removed from the chat transcript after specified number of <seconds>.

## transcript-resend-attempts

**Default Value:** 10

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

**Modified:** 8.5.301.06

Specifies how many times Chat Server tries to send "Update" and "Close" requests to UCS. If one request fails (with a recoverable reason), or a timeout expires, Chat Server continues to send requests until it reaches the total specified number of times. Setting this option to "0" completely disables sending these requests to UCS and should be used only for debug purposes.

**Note:** When Cassandra is deployed, the resending of the "Update" request is configured by "retry-attempts" in Cassandra RAP.

## transcript-resend-delay

**Default Value:** 1

**Valid Values:** Any integer from 0-86400

**Changes Take Effect:** Immediately

**Modified:** 8.5.301.06

Specifies how long, in seconds, Chat Server waits before resending "Update" and "Close" requests to UCS for the first time if the initial request failed due to a recoverable reason or because the timeout expired. For each subsequent resend, the current delay is multiplied by two. For example, if the current delay is four seconds and a resend occurs, the following delay is set to eight seconds. If another resend occurs, the delay is set to 16 seconds. Set this option to "0" to disable resending requests.

**Note:** When Cassandra is deployed, the resending of the "Update" request is configured by "resent-delay" in Cassandra RAP.

## transcript-save-notices

**Default Value:** selective

**Valid Values:** all, none, selective, selective2

**Changes Take Effect:** Immediately

Specifies which notifications are saved in the transcript in the UCS database.

- **all:** All notifications are saved in the transcript.
- **none:** No notifications are saved in the transcript.
- **selective:** All notifications except typing notifications (USER\_TYPING\_ON and USER\_TYPING\_OFF) and idle control notifications (IDLE\_CONTROL\_ALERT and IDLE\_CONTROL\_CLOSE) are saved in the transcript.
- **selective2:** All notifications except typing notifications (USER\_TYPING\_ON and USER\_TYPING\_OFF) are saved in the transcript.

## transcript-save-on-error

**Default Value:** continue

**Valid Values:** continue, close

**Changes Take Effect:** Immediately

Specifies how to process an unrecoverable error from Universal Contact Server in the course of a chat session (when updating the transcript). The possible actions are to continue, or to close the chat session.

## use-contact-server

**Default Value:** true

**Valid Values:** true, false

**Changes Take Effect:** At start/restart

If this option is set to true, Chat Server creates and updates records in the Universal Contact Server database for each session that is created. If set to false, Chat Server does not interact with Universal Contact Server.

## user-register-timeout

**Default Value:** 25

**Valid Values:** Any integer from 1-86400

**Changes Take Effect:** Immediately

Specifies the timeout, in seconds, during which the chat client must send a registration/login request before it is disconnected. This option is intended for debugging Web Applications. It has almost no practical use in a regular environment.

## web-user-max-messages

**Default Value:** 100

**Valid Values:** Any integer from 0-1000000

**Changes Take Effect:** Immediately

Sets a limit on the number of messages a web user can submit during a session. This limit prevents the transcript from growing too large, thus reducing the over-consumption of RAM as well as Universal Contact Server database space.

## xml-request-max-size

**Default Value:** 32768

**Valid Values:** Any integer from 200-2147482624

**Changes Take Effect:** Immediately

**Modified:** 8.5.308.06

Sets a limit on the size (in bytes) of incoming XML packets. This limit helps prevent Chat Server from hanging during an XML parsing of overly large amounts of incoming data. The actual payload size (in symbols) could be estimated roughly as the XML packet size multiplied by 0.45.