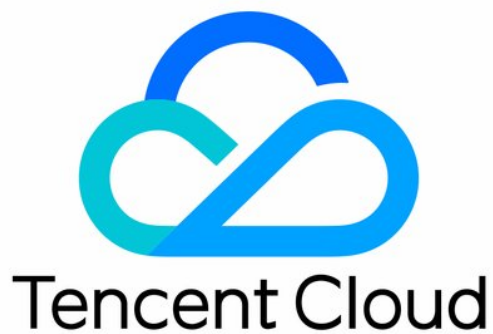


Chat

Client APIs

Product Documentation



Copyright Notice

©2013-2025 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by the Tencent corporate group, including its parent, subsidiaries and affiliated companies, as the case may be. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

Contents

Client APIs

JavaScript

Android

iOS & macOS

Swift

Flutter

Electron

Unity

React Native

C APIs

C++

Client APIs

JavaScript

Last updated : 2025-01-13 12:01:25

TencentCloudChat

`TencentCloudChat` is the namespace of the Chat JavaScript SDK and provides the static method `create()` for creating SDK instances, the event constant `EVENT`, and the type constant `TYPES`, and the signaling constant `TSignaling`.

Initialization

API	Description
create	Creates an SDK instance.

SDK Instance

Term	Description
Message	<code>Message</code> indicates the content to be sent and carries multiple attributes which specify whether you are the sender, the sender account, the message generation time, and so on.
Conversation	Two types of conversation are available: Client to Client (C2C): A one-to-one chat, involving only two participants. GROUP: A group chat, involving more than two participants.
<code>Profile</code>	<code>Profile</code> describes the basic information of a user, including the nickname, gender, personal signature, and profile photo URL.
<code>Friend</code>	<code>Friend</code> describes the basic information of a friend, including the remarks and friend list.
<code>FriendApplication</code>	<code>FriendApplication</code> describes the basic information of a friend request, including the friend source and remarks.
<code>FriendGroup</code>	<code>FriendGroup</code> describes the basic information of a friend group, including the friend group name and members.
<code>Group</code>	<code>Group</code> indicates a communication system for group chatting, including Work Group, Public Group, Meeting Group, Audio-Video Group, and Community Group.

GroupMember	GroupMember indicates the basic information of each group member, such as the ID, nickname, role, and the time of joining the group.
Group tip	A group tip is generated when an event such as adding or deleting group member occurs. The access side can configure whether to display group tips to group members.
Group system message	For example, when a user requests to join a group, the group admin receives a system message. After the admin accepts or rejects the request, the Chat SDK returns the result to the access side, which then displays the result to the user.
Message display on screen	The sent messages, including text segments and images, are displayed on the computer or phone screen.

Event

API	Description
on	Enables event listening.
off	Disables event listening.

Plugin registration

API	Description
registerPlugin	Registers a plugin.

Setting log level

API	Description
setLogLevel	Sets the log level. Logs below this level will not be printed.

SDK instance termination

API	Description
destroy	Terminates an SDK instance. The SDK will log out, disconnect the WebSocket persistent connection, and then release resources.

Login

API	Description
-----	-------------

login	Logs in to the Chat SDK using userID and userSig.
logout	Logs out to the Chat SDK.
isReady	Indicates the SDK enter the ready state or not.
getLoginUser	Get the userID of the logged-in user. If the user is not logged in, it returns an empty string ("").
getServerTime	Get the server time.

Message

API	Description
createTextMessage	Creates a text message.
createTextAtMessage	Creates a text message with the @ notification feature.
createImageMessage	Creates an image message.
createAudioMessage	Creates an audio message.
createVideoMessage	Creates a video message.
createCustomMessage	Creates a custom message.
createFaceMessage	Creates an emoji message.
createFileMessage	Creates a file message.
createLocationMessage	Creates a location message.
createMergerMessage	Creates a combined message.
downloadMergerMessage	Downloads a combined message.
createForwardMessage	Creates a forward message.
sendMessage	Sends a message.
revokeMessage	Recalls a message.
deleteMessage	Deletes a message.
translateText	Translates a message.
setMessageExtensions	Sets message extensions.

getMessageExtensions	Gets message extensions.
deleteMessageExtensions	Deletes message extensions.
addMessageReaction	Add message reactions. This interface allows for scenarios such as "emoji reactions" to be implemented.
removeMessageReaction	Remove message reactions.
getMessageReactions	Retrieve information of message reactions.
getAllUserListOfMessageReaction	Paginate and retrieve the user list for a specific message reaction.

Conversation

API	Description
modifyMessage	Modifies a message.
getMessageList	Gets the message list.
getMessageListHopping	Pulls the conversation message list by specified sequence or time range.
sendMessageReadReceipt	Sends message read receipts.
getMessageReadReceiptList	Pulls the list of message read receipts.
getGroupMessageReadMemberList	Pulls the list of members who have (or have not) read a group message.
findMessage	Queries local messages in a specified conversation by <code>messageID</code> .
setMessageRead	Marks a message as read.
getConversationList	Gets the conversation list.
getConversationProfile	Gets the conversation information.
deleteConversation	Deletes a conversation.
setConversationDraft	Set conversation draft.
clearHistoryMessage	Clears the chat history with a user from local storage and the cloud (without deleting the conversation).
pinConversation	Pins/Unpins a conversation to/from top.

setAllMessageRead	Marks the unread messages of all conversations as read.
setMessageRemindType	Sets the conversation message notification type. You can use this API to mute notifications or reject messages.
getTotalUnreadMessageCount	Gets the total unread message count of conversations.

Conversation group

API	Description
setConversationCustomData	Sets custom conversation data.
markConversation	Marks a conversation.
getConversationGroupList	Gets the list of conversation groups.
createConversationGroup	Creates a conversation group.
deleteConversationGroup	Deletes a conversation group.
renameConversationGroup	Renames a conversation group.
addConversationsToGroup	Adds a conversation to a conversation group.
deleteConversationsFromGroup	Deletes conversations from a conversation group.

Search

API	Description
searchCloudMessages	Search cloud messages.
searchCloudUsers	Search cloud users.
searchCloudGroups	Search cloud groups.
searchCloudGroupMembers	Search cloud group members.

Profile

API	Description
getMyProfile	Gets personal profile.
getUserProfile	Gets other user's profile.

updateMyProfile	Updates your personal profile.
getBlacklist	Gets your blacklist.
addToBlacklist	Adds a user to the blacklist.
removeFromBlacklist	Removes a user from the blacklist.

User status

API	Description
setSelfStatus	Sets one's own custom status.
getUserStatus	Queries a user's status.
subscribeUserStatus	Subscribes to a user's status.
unsubscribeUserStatus	Unsubscribes from a user's status.

Relationship Chain

API	Description
getFriendList	Gets the contacts in the SDK cache.
addFriend	Adds friends.
deleteFriend	Deletes friends.
checkFriend	Verifies friend relationship.
getFriendProfile	Gets the friend data and profile data of a specified user.
updateFriend	Updates the contacts data of friends.
getFriendApplicationList	Gets the friend request list in the SDK cache.
acceptFriendApplication	Accepts a friend request.
refuseFriendApplication	Rejects a friend request.
deleteFriendApplication	Deletes a friend request.
setFriendApplicationRead	Set a friend request as reads.
getFriendGroupList	Gets list of friend groups in the SDK cache.

createFriendGroup	Creates a friend group.
deleteFriendGroup	Deletes a friend group.
addToFriendGroup	Adds friends to a friend group.
removeFromFriendGroup	Removes friends from a friend group.
renameFriendGroup	Modifies the name of a friend group.

Group

API	Description
getGroupList	Gets the group list.
getGroupProfile	Gets the group profile.
createGroup	Creates a group.
dismissGroup	Disbands a group.
updateGroupProfile	Modifies the group profile.
joinGroup	Requests to join a group.
quitGroup	Quits a group.
searchGroupById	Searches for a group.
getGroupOnlineMemberCount	Gets the number of online users in an audio-video group.
changeGroupOwner	Transfers the group ownership.
getGroupApplicationList	Gets the list of requests to join a group.
handleGroupApplication	Processes requests to join the group.
initGroupAttributes	Initializes group attributes.
setGroupAttributes	Sets group attributes.
deleteGroupAttributes	Deletes group attributes.
getGroupAttributes	Gets group attributes.
setGroupCounters	Sets group counters.
increaseGroupCounter	Increases the specified group counters.

decreaseGroupCounter	Decreases the specified group counters.
getGroupCounters	Gets the specified group counters.

Group member

API	Description
getGroupMemberList	Gets the group member list.
getGroupMemberProfile	Gets a group member's profile.
addGroupMember	Adds a group member.
deleteGroupMember	Deletes a group member.
setGroupMemberMuteTime	Configures the muting period.
setGroupMemberRole	Modifies the group member's role.
setGroupMemberNameCard	Sets the name card for a group member.
setGroupMemberCustomField	Sets a custom field for a group member.
markGroupMemberList	Marks group members.

Topic

API	Description
getJoinedCommunityList	Gets the list of community groups the current user has joined.
createTopicInCommunity	Creates a topic in a community group.
deleteTopicFromCommunity	Deletes a topic in a community group.
updateTopicProfile	Updates a topic profile in a community group.
getTopicList	Gets the topic list in a community group.

Signaling

API	Description
addSignalingListener	Listens for a signaling event.
removeSignalingListener	Removes listener for a signaling event.

<code>invite</code>	Invites someone.
<code>inviteInGroup</code>	Invites members in a group.
<code>cancel</code>	The inviter cancels the invitation.
<code>accept</code>	The invitee accepts the invitation.
<code>reject</code>	The invitee rejects the invitation.
<code>getSignalingInfo</code>	Gets signaling info from a message.
<code>modifyInvitation</code>	Modifies a invitation.

Android

Last updated : 2025-01-23 16:13:12

Note:

Do not use APIs of new and old versions at the same time.

Initialization and Login APIs

To use the Tencent Cloud Chat service, you need to initialize the SDK and log in.

API	Description
initSDK	Initializes the SDK.
unInitSDK	Uninitializes the SDK.
addIMSDKListener	Adds the Chat listener.
removeIMSDKListener	Removes the Chat listener.
getVersion	Gets the version number.
getServerTime	Gets the server time.
login	Logs in.
logout	Logs out.
getLoginStatus	Gets the login status.
getLoginUser	Gets the UserID of the currently logged-in user.

Simple Message APIs

Use the following APIs for the sending and receiving of text and signaling (custom buffer) messages.

API	Description
addSimpleMsgListener	Sets an event listener for simple messages (text messages and custom messages). Do not use it and addAdvancedMsgListener at the same time.
removeSimpleMsgListener	Removes the event listener for simple messages (text messages and custom messages).

sendC2CTextMessage	Sends a one-to-one (C2C) text message.
sendC2CCustomMessage	Sends a one-to-one (C2C) custom (signaling) message.
sendGroupTextMessage	Sends a group text message.
sendGroupCustomMessage	Sends a group custom (signaling) message.

Signaling APIs

API	Description
addSignalingListener	Adds a signaling listener.
removeSignalingListener	Removes a signaling listener.
invite	Invites a user.
inviteInGroup	Invites certain users in the group.
cancel	Cancels an invitation.
accept	Accepts an invitation.
reject	Rejects an invitation.
getSignalingInfo	Gets the signaling information.
addInvitedSignaling	Adds invitation signaling (can be used for invitation signaling triggered by offline push messages for group invitations).
modifyInvitation	Modifies the invitation signaling.

Advanced Message APIs

If you need to send/receive rich media messages (such as image, video, and file messages) and use advanced features such as recalling messages, marking messages as read, and querying message history, use the following set of advanced message APIs. Do not use simple message APIs and advanced message APIs at the same time.

API	Description
addAdvancedMsgListener	Sets an event listener for advanced messages. Do not use it and addSimpleMsgListener at the same time.

removeAdvancedMsgListener	Removes the event listener for advanced messages.
createTextMessage	Creates a text message.
createCustomMessage	Creates a custom message.
createImageMessage	Creates an image message.
createSoundMessage	Creates an audio message.
createVideoMessage	Creates a video message.
createFileMessage	Creates a file message.
createLocationMessage	Creates a location message.
createFaceMessage	Creates an emoji message.
createMergerMessage	Creates a combined forward message.
createForwardMessage	Creates a single forward message.
createTargetedGroupMessage	Creates a targeted group message.
createAtSignedGroupMessage	Creates a group @ message.
sendMessage	Sends a message. The message object can be created using a <code>createXXXMessage</code> API.
setC2CReceiveMessageOpt	Sets the Mute Notifications option for one-to-one messages.
getC2CReceiveMessageOpt	Gets the Mute Notifications status for one-to-one messages.
setGroupReceiveMessageOpt	Sets the Mute Notifications option for group messages.
setAllReceiveMessageOpt	Sets the Mute Notifications status for global messages (daily repetition allowed).
setAllReceiveMessageOpt	Sets the Mute Notifications status for global messages.
getAllReceiveMessageOpt	Gets the Mute Notifications status for global messages.
getC2CHistoryMessageList	Gets one-to-one (C2C) message history.
getGroupHistoryMessageList	Gets the group message history.
getHistoryMessageList	Gets message history.
revokeMessage	Recalls a message. The message object can be created using a <code>createXXXMessage</code> API.

modifyMessage	Modifies a message.
markC2CMessageAsRead	Marks one-to-one (C2C) messages as read (a interface to be abandoned. Use the cleanConversationUnreadMessageCount interface).
markGroupMessageAsRead	Marks group messages as read (a interface to be abandoned. Use the cleanConversationUnreadMessageCount interface).
markAllMessageAsRead	Marks all conversations as read (a interface to be abandoned. Use the cleanConversationUnreadMessageCount interface).
deleteMessageFromLocalStorage	Deletes a message from the local storage.
deleteMessages	Deletes messages from local storage and the cloud.
clearC2CHistoryMessage	Clears chat history with a user from local storage and the cloud.
clearGroupHistoryMessage	Clears chat history of a group from local storage and the cloud.
insertGroupMessageToLocalStorage	Inserts a message in a group chat.
insertC2CMessageToLocalStorage	Inserts a message in a one-to-one chat.
findMessages	Finds local messages by <code>msgID</code> .
searchLocalMessages	Searches for local messages.
searchCloudMessages	Searchs for cloud messages.
sendMessageReadReceipts	Sends message read receipts.
getMessageReadReceipts	Gets message read receipts.
getGroupMessageReadMemberList	Gets the list of group members who have read group messages.
setMessageExtensions	Sets message extensions.
getMessageExtensions	Gets message extensions.
deleteMessageExtensions	Deletes message extensions.
addMessageReaction	Adds message reaction.
removeMessageReaction	Removes message reaction.
getMessageReactions	Gets message reactions.
getAllUserListOfMessageReaction	Gets all user list of message reaction.

pinGroupMessage	Sets group message pinning.
getPinnedGroupMessageList	Gets pinned group message list.

Group APIs

Tencent Cloud Chat SDK supports five preset group types, each of which pertains to different scenarios.

Work group (Work): users can join the group only after being invited by group members. This group type is the same as private group (Private) in earlier versions.

Public group (Public): Users can join a public group through requests, which need to be approved by the group owner or group admin.

Meeting group (Meeting): used together with [TRTC](#) to enable scenarios such as video conferencing and online education. Users can join and leave the group freely and view the message history before they join. Same as chat room (ChatRoom) in earlier versions.

Community: A user can join and leave a community freely. It is suitable for chat scenarios with a super large number of community members, such as knowledge sharing and game discussion. This feature is supported by a client with the SDK enhanced edition v5.8.1668 or later and the web SDK v2.17.0 or later. To use it, you need to purchase the [Pro edition](#)、[Pro Plus edition](#)、[Enterprise edition](#), and then enable it in [Console](#), path: Applications > Your App > Chat > Configuration > Group Configuration > Community.

Audio-video group (AVChatRoom): An audio-video group allows users to join and leave freely and is suitable for scenarios such as live streaming and chat rooms with on-screen comments. There is no limit on the number of group members.

API	Description
addGroupListener	Adds a group listener.
removeGroupListener	Removes a group listener.
createGroup	Creates a (simple) group
createGroup	Creates an (advanced) group. The group information and the initial group members can be set during group creation.
joinGroup	Joins a group.
quitGroup	Leaves a group.
dismissGroup	Disbands a group. Only the group owner and group admin can delete a group.
getJoinedGroupList	Gets the list of groups the current user has joined, excluding audio-video groups.

getGroupsInfo	Pulls the profiles of groups.
searchGroups	Search for local group profiles
searchCloudGroups	Search for cloud group profiles
setGroupInfo	Modifies a group profile.
initGroupAttributes	Initializes group attributes.
setGroupAttributes	Sets group attributes.
deleteGroupAttributes	Deletes group attributes.
getGroupAttributes	Gets group attributes.
getGroupOnlineMemberCount	Gets the number of online group members.
setGroupCounters	Sets group counters.
getGroupCounters	Gets group counters.
increaseGroupCounter	Increment group counters.
decreaseGroupCounter	Decrement group counters.
getGroupMemberList	Gets the group member list.
getGroupMembersInfo	Gets the profiles of specified group members.
searchGroupMembers	Search for local group members
searchCloudGroupMembers	Search for cloud group members
setGroupMemberInfo	Modifies the profile of a specified group member.
muteGroupMember	Mutes group members.
muteAllGroupMembers	Mute all group members. Only administrators or group owners can call this function.
kickGroupMember	Removes a member from a group.
setGroupMemberRole	Sets the role of a group member.
markGroupMemberList	Marks group members.
transferGroupOwner	Transfers the group ownership.

inviteUserToGroup	Invites users to a group.
getGroupApplicationList	Gets the list of requests to join a group.
acceptGroupApplication	Approves a request to join a group.
refuseGroupApplication	Rejects a request to join a group.
setGroupApplicationRead	Marks the request list as read.
getJoinedCommunityList	Gets the list of community groups the current user has joined.
createTopicInCommunity	Creates a topic.
deleteTopicFromCommunity	Deletes a topic.
setTopicInfo	Modifies topic information.
getTopicInfoList	Gets the list of topics.

Community Topic-Related APIs

If you need to create topics under a community, please use this set of interfaces. Communities are used to manage group members, and all topics under a community can not only share community members but also send and receive messages independently without interfering with each other.

API	Description
addCommunityListener	Adds a community listener.
removeCommunityListener	Removes a community listener.
createCommunity	Creates a community with topic support.
getJoinedCommunityList	Gets the list of communities with topic support that the current user has joined.
createTopicInCommunity	Creates a topic.
deleteTopicFromCommunity	Deletes a topic.
setTopicInfo	Modifies topic information.
getTopicInfoList	Gets a topic list.
createPermissionGroupInCommunity	Creates a community permission group.

deletePermissionGroupFromCommunity	Deletes a community permission group.
modifyPermissionGroupInfoInCommunity	Modifies a community permission group.
getJoinedPermissionGroupListInCommunity	Gets the list of community permission groups the current user has joined.
getPermissionGroupListInCommunity	Gets the list of community permission groups.
addCommunityMembersToPermissionGroup	Adds members to a community permission group.
removeCommunityMembersFromPermissionGroup	Removes members from a community permission group.
getCommunityMemberListInPermissionGroup	Gets the member list of a community permission group.
addTopicPermissionToPermissionGroup	Adds topic permissions to a permission group.
deleteTopicPermissionFromPermissionGroup	Removes topic permissions from a permission group.
modifyTopicPermissionInPermissionGroup	Modifies topic permissions in a permission group.
getTopicPermissionInPermissionGroup	Gets topic permissions in a permission group.

Conversation List APIs

The conversation list is the list a user sees on the first screen after login. It includes elements such as conversation node, conversation name, group name, last message, and unread count.

API	Description
addConversationListener	Adds a conversation listener.
removeConversationListener	Removes a conversation listener.
getConversationList	Gets the conversation list.
getConversationListByFilter	Gets the advanced conversation API to specify the conversation type, mark type, and group name.
getConversation	Gets a conversation.
getConversationList	Gets multiple conversations.
deleteConversation	Deletes a conversation.
deleteConversationList	Deletes a conversation list.

setConversationDraft	Sets a draft for a conversation.
setConversationCustomData	Sets custom conversation data.
pinConversation	Pins a conversation to the top.
markConversation	Marks a conversation.
getTotalUnreadMessageCount	Gets the total unread message count.
getUnreadMessageCountByFilter	Gets the total number of unread messages filtered by conversation.
subscribeUnreadMessageCountByFilter	Registers a listener for changes in the total number of unread messages in a specific conversation.
unsubscribeUnreadMessageCountByFilter	Unregisters a listener for changes in the total number of unread messages in a specific conversation.
cleanConversationUnreadMessageCount	Clears the unread message count for a conversation.
createConversationGroup	Creates a conversation group.
getConversationGroupList	Gets the list of conversation groups.
deleteConversationGroup	Deletes a conversation group.
renameConversationGroup	Renames a conversation group.
addConversationsToGroup	Adds a conversation to a conversation group.
deleteConversationsFromGroup	Deletes a conversation from a conversation group.

User Profile APIs

You can use the following APIs to query user profiles, modify your profile, and block messages from a specified user (that is, adding a specified user to the blocklist).

API	Description
getUsersInfo	Gets user profiles.
setSelfInfo	Modifies one's own user profile.
subscribeUserInfo	Subscribe to user profile.
unsubscribeUserInfo	Unsubscribe from user profile.

getUserStatus	Queries a user's status.
setSelfStatus	Sets one's own status.
subscribeUserStatus	Subscribes to a user's status.
unsubscribeUserStatus	Unsubscribes from a user's status.
searchUsers	Search cloud users
addToBlackList	Blocks messages from a user, which means adding the user to the blocklist.
deleteFromBlackList	Unblocks messages from a user, which means removing the user from the blocklist.
getBlackList	Gets the blocklist.

Offline Push APIs

We recommend you use the offline push service if you want your app to receive Chat service messages in real time when it runs in the background. As there is no unified push service in the Chinese mainland, you need to [configure Android offline push](#) for devices of different vendors separately.

API	Description
setOfflinePushConfig	Configures offline push.
doBackground	The APP can call this interface when it detects that the application has entered the background, which can be used to initialize the unread count for the desktop application badge.
doForeground	The APP can call this interface when it detects that the application has entered the foreground.

Friend Management APIs

Tencent Cloud Chat SDK does not check friend relationships by default when sending and receiving messages. To achieve the interactive experience of "add friends first, then send messages", you can log in to the [Console](#) to modify Relationship Check. When enabled, users can only send messages to their friends. When a user sends a message to a non-friend, SDK will report a 20009 error code. The configuration path is: Applications > Your App > Chat > Configuration > Login and Message > Relationship Check.

Use the following API to add or delete friends and manage friend lists.

--	--

API	Description
addFriendListener	Adds a contacts listener.
removeFriendListener	Removes a contacts listener.
getFriendList	Gets the contacts.
getFriendsInfo	Gets the profiles of specified friends.
setFriendInfo	Sets the profile of a specified friend.
searchFriends	Searches for friends.
addFriend	Adds a friend.
deleteFromFriendList	Deletes a friend.
checkFriend	Checks relationship with specified users.
getFriendApplicationList	Gets the list of friend requests.
acceptFriendApplication	Accepts a friend request.
refuseFriendApplication	Rejects a friend request.
deleteFriendApplication	Deletes a friend request.
setFriendApplicationRead	Marks a friend request as read.
createFriendGroup	Creates a friend list.
getFriendGroups	Gets the information of friend lists.
deleteFriendGroup	Deletes friend lists.
renameFriendGroup	Modifies the name of a friend list.
addFriendsToFriendGroup	Adds friends to a friend list.
deleteFriendsFromFriendGroup	Removes friends from a friend list.
subscribeOfficialAccount	Subscribes to official accounts.
unsubscribeOfficialAccount	Unsubscribes from official accounts.
getOfficialAccountsInfo	Gets the list of official accounts.
followUser	Follows users.

unfollowUser	Unfollows users.
getMyFollowingList	Gets my follow list.
getMyFollowersList	Gets my fan list.
getMutualFollowersList	Gets my mutual follow list.
getUserFollowInfo	Gets the follow/fan/mutual follow count information for a specified user.
checkFollowType	Checks the follow type of a specified user.

iOS & macOS

Last updated : 2025-01-23 16:13:12

Note:

Do not use APIs of new and old versions at the same time.

Initialization and Login APIs

To use the Tencent Cloud Chat service, you need to initialize the SDK and log in.

API	Description
initSDK	Initializes the SDK.
unInitSDK	Uninitializes the SDK
addIMSDKListener	Adds the Chat listener.
removeIMSDKListener	Removes the Chat listener.
getVersion	Gets the version number.
getServerTime	Gets the server time.
login	Logs in.
logout	Logs a user out
getLoginUser	Gets the currently logged-in user.
getLoginStatus	Gets the login status

Simple Message APIs

Use the following APIs for the sending and receiving of text and signaling (custom buffer) messages.

API	Description
addSimpleMsgListener	Sets an event listener for simple messages (text messages and custom messages).Do not use this API and addAdvancedMsgListener at the same time.
removeSimpleMsgListener	Removes the event listener for simple messages (text messages and custom

	messages)
sendC2CTextMessage	Sends a one-to-one chat (C2C) text message
sendC2CCustomMessage	Sends a one-to-one chat (C2C) custom (signaling) message
sendGroupTextMessage	Sends a group chat text message
sendGroupCustomMessage	Sends a group chat custom (signaling) message

Signaling APIs

API	Description
addSignalingListener	Adds a signaling listener.
removeSignalingListener	Removes a signaling listener.
invite	Invites a user.
inviteInGroup	Invites certain users in a group.
cancel	Cancels an invitation.
accept	Accepts an invitation.
reject	Rejects an invitation.
getSignalingInfo	Gets the signaling information.
addInvitedSignaling	Adds invitation signaling (can be used for invitation signaling triggered by offline push messages for group invitations).
modifyInvitation	Modifies the invitation signaling.

Advanced Message APIs

If you need to send/receive rich media messages (such as image, video, and file messages) and use advanced features such as recalling messages, marking messages as read, and querying message history, use the following set of advanced message APIs. Do not use simple message APIs and advanced message APIs at the same time.

API	Description
addAdvancedMsgListener	Sets an event listener for advanced messages. Do not use this API and

	addSimpleMsgListener at the same time.
removeAdvancedMsgListener	Removes the listener for advanced messages
createTextMessage	Creates a text message
createTextAtMessage	Creates an @ text message.
createCustomMessage	Creates a custom message
createImageMessage	Creates an image message
createSoundMessage	Creates a voice message.
createVideoMessage	Creates a video message
createFileMessage	Creates a file message.
createLocationMessage	Creates a location message
createFaceMessage	Creates an emoji message
createMergerMessage	Creates a combined forward message.
createForwardMessage	Creates a single forward message.
createTargetedGroupMessage	Creates a targeted group message.
createAtSignedGroupMessage	Creates a group @ message.
sendMessage	Sends a message. The message object can be created using a <code>createXXXMessage</code> API.
setC2CReceiveMessageOpt	Sets the Mute Notifications option for one-to-one messages.
getC2CReceiveMessageOpt	Gets the Mute Notifications status for one-to-one messages.
setGroupReceiveMessageOpt	Sets the Mute Notifications option for group messages.
getC2CHistoryMessageList	Gets the one-to-one chat (C2C) message history
getGroupHistoryMessageList	Gets the group chat message history
getHistoryMessageList	Gets the message history.
revokeMessage	Recalls a message. The message object can be created using a <code>createXXXMessage</code> API.
modifyMessage	Modifies a message.

markC2CMessageAsRead	Marks one-to-one chat (C2C) messages as read
markGroupMessageAsRead	Marks group messages as read
markAllMessageAsRead	Marks all conversations as read.
deleteMessageFromLocalStorage	Deletes a message from the local storage
deleteMessages	Deletes messages from local storage and the cloud.
clearC2CHistoryMessage	Clears chat history with a user from local storage and the cloud.
clearGroupHistoryMessage	Clears chat history of a group from local storage and the cloud.
insertGroupMessageToLocalStorage	Inserts a message in a group chat.
insertC2CMessageToLocalStorage	Inserts a message in a one-to-one chat.
findMessages	Finds local messages by <code>msgID</code> .
searchLocalMessages	Search for local messages.
searchCloudMessages	Search for cloud messages.
sendMessageReadReceipts	Sends message read receipts.
getMessageReadReceipts	Gets message read receipts.
getGroupMessageReadMemberList	Gets the list of group members who have read group messages.
setMessageExtensions	Sets message extensions.
getMessageExtensions	Gets message extensions.
deleteMessageExtensions	Deletes message extensions.
addMessageReaction	Adds message reaction.
removeMessageReaction	Removes message reaction.
getMessageReactions	Gets message reactions.
getAllUserListOfMessageReaction	Gets all user list of message reaction.
translateText	Translates a text message.
pinGroupMessage	Sets group message pinning.
getPinnedGroupMessageList	Gets pinned group message list.

Group APIs

Tencent Cloud Chat SDK supports five preset group types, each of which pertains to different scenarios.

Work group (Work): users can join the group only after being invited by group members. This group type is the same as private group (Private) in earlier versions.

Public group (Public): Users can join a public group through requests, which need to be approved by the group owner or group admin.

Meeting group (Meeting): used together with [TRTC](#) to enable scenarios such as video conferencing and online education. Users can join and leave the group freely and view the message history before they join. Same as chat room (ChatRoom) in earlier versions.

Community: A user can join and leave a community freely. It is suitable for chat scenarios with a super large number of community members, such as knowledge sharing and game discussion. This feature is supported by a client with the SDK enhanced edition v5.8.1668 or later and the web SDK v2.17.0 or later. To use it, you need to purchase the [Pro edition](#)、[Pro Plus edition](#)、[Enterprise edition](#), and then enable it in [Console](#), path: Applications > Your App > Chat > Configuration > Group Configuration > Community.

Audio-video group (AVChatRoom): An audio-video group allows users to join and leave freely and is suitable for scenarios such as live streaming and chat rooms with on-screen comments. There is no limit on the number of group members.

API	Description
setGroupListener	Sets an event listener for groups. (This API is to be disused. Please use the <code>addGroupListener</code> and <code>removeGroupListener</code> APIs.)
addGroupListener	Adds an event listener for groups.
removeGroupListener	Removes an event listener for groups.
createGroup	(Simple API) Creates a group.
createGroup	(Advanced API) Creates a group. The group information and the initial group members can be set during group creation.
joinGroup	Joins a group
quitGroup	Leaves a group.
dismissGroup	Disbands a group. Only the group owner and group admin can disband a group.
getJoinedGroupList	Gets the list of groups the current user has joined, excluding audio-video groups.
getGroupsInfo	Pulls the profiles of groups

searchGroups	Search for local group profiles
searchCloudGroups	Search for cloud group profiles
setGroupInfo	Modifies the profile of a group
initGroupAttributes	Initializes group attributes.
setGroupAttributes	Sets group attributes.
deleteGroupAttributes	Deletes group attributes.
getGroupAttributes	Gets group attributes.
getGroupOnlineMemberCount	Gets the number of online group members.
getGroupMemberList	Gets the group member list
getGroupMembersInfo	Gets the profiles of specified group members
searchGroupMembers	Search for local group members
searchCloudGroupMembers	Search for cloud group members
setGroupMemberInfo	Modifies the profile of a specified group member
muteGroupMember	Mutes a group member.
inviteUserToGroup	Invites users to a group
kickGroupMember	Removes a member from a group.
setGroupMemberRole	Sets a role for a group member.
markGroupMemberList	Marks group members.
transferGroupOwner	Transfers the ownership of a group
getGroupApplicationList	Gets the list of applications to join a group
acceptGroupApplication	Accepts an application to join a group
refuseGroupApplication	Rejects a request to join a group.
setGroupApplicationRead	Marks the application list as read
getJoinedCommunityList	Gets the list of community groups the current user has joined.
createTopicInCommunity	Creates a topic.

deleteTopicFromCommunity	Deletes a topic.
setTopicInfo	Modifies topic information.
getTopicInfoList	Gets the message list.

Conversation List APIs

The conversation list is the list a user sees on the first screen after login. It includes elements such as conversation node, conversation name, group name, last message, and unread count.

API	Description
setConversationListener	Sets a conversation listener. (This API is to be disused. Please use the <code>addConversationListener</code> and <code>removeConversationListener</code> APIs.)
addConversationListener	Adds a conversation listener.
removeConversationListener	Removes a conversation listener.
getConversationList	Gets the conversation list
getConversation	Gets a specified conversation.
getConversationList	Gets multiple conversations.
getConversationListByFilter	Gets the advanced conversation API to specify the conversation type, mark type, and group name.
deleteConversation	Deletes a conversation
setConversationDraft	Sets a draft for a conversation
setConversationCustomData	Sets custom conversation data.
pinConversation	Pins a conversation to the top.
markConversation	Marks a conversation.
getTotalUnreadMessageCount	Gets the total unread message count.
createConversationGroup	Creates a conversation group.
getConversationGroupList	Gets the list of conversation groups.
deleteConversationGroup	Deletes a conversation group.

renameConversationGroup	Renames a conversation group.
addConversationsToGroup	Adds a conversation to a conversation group.
deleteConversationsFromGroup	Deletes a conversation from a conversation group.

User Profile APIs

You can use the following APIs to query user profiles, modify your profile, and block messages from a specified user (that is, adding a specified user to the blocklist).

API	Description
getUsersInfo	Gets users' profiles
setSelfInfo	Modifies one's own user profile
getUserStatus	Queries a user's status.
setSelfStatus	Sets one's own status.
subscribeUserStatus	Subscribes to a user's status.
unsubscribeUserStatus	Unsubscribes from a user's status.
searchUsers	Search cloud users
addToBlackList	Blocks messages from a specified user, which means adding the user to the blocklist.
deleteFromBlackList	Unblocks messages from a specified user, which means removing the user from the blocklist.
getBlackList	Gets the blocklist

Offline Push APIs

Use the offline push service if you want your app to receive Chat service messages in real time when it is in the background.

API	Description
setAPNSListener	Sets an APNs listener.

setAPNS	Configures APNs Push info.
setVOIP	Configures VoIP Push info.

Friend Management APIs

Tencent Cloud Chat SDK does not check friend relationships by default when sending and receiving messages. To achieve the interactive experience of "add friends first, then send messages", you can log in to the [Console](#) to modify Relationship Check. When enabled, users can only send messages to their friends. When a user sends a message to a non-friend, SDK will report a 20009 error code. The configuration path is: Applications > Your App > Chat > Configuration > Login and Message > Relationship Check.

Use the following API to add or delete friends and manage friend lists.

API	Description
setFriendListener	Sets a contacts and friend profile listener. (This API is to be disused. Please use the <code>addFriendListener</code> and <code>removeFriendListener</code> APIs.)
addFriendListener	Adds a contacts listener.
removeFriendListener	Removes a contacts listener.
getFriendList	Gets the contacts.
getFriendsInfo	Gets the profiles of specified friends
setFriendInfo	Sets the profile of a specified friend
searchFriends	Searches for friends.
addFriend	Adds a friend
deleteFromFriendList	Deletes a friend
checkFriend	Checks relationship with specified users.
getFriendApplicationList	Gets the list of friend requests
acceptFriendApplication	Accepts a friend request
refuseFriendApplication	Rejects a friend request
deleteFriendApplication	Deletes a friend request
setFriendApplicationRead	Marks a friend request as read.

<code>createFriendGroup</code>	Creates a friend list
<code>getFriendGroups</code>	Gets the information of friend lists
<code>deleteFriendGroup</code>	Deletes friend lists
<code>renameFriendGroup</code>	Modifies the name of a friend list
<code>addFriendsToFriendGroup</code>	Adds friends to a friend list
<code>deleteFriendsFromFriendGroup</code>	Removes friends from a friend list.

Swift

Last updated : 2025-01-23 16:13:12

Note:

Do not use APIs of new and old versions at the same time.

Initialization and Login APIs

To use Tencent Cloud IM services, you need to initialize the SDK and log in.

API	Description
initSDK	Initializes the SDK.
unInitSDK	Uninitializes the SDK.
addIMSDKListener	Add IM listener
removeIMSDKListener	Remove IM listener
getVersion	Gets the version number.
getServerTime	Gets the server time.
login	Logs in.
logout	Logs out.
getLoginUser	Gets the currently logged-in user.
getLoginStatus	Gets the login status.
getUserStatus	Gets user status information.
setSelfStatus	Sets user status for yourself.
subscribeUserStatus	Subscribes user status.
unsubscribeUserStatus	Unsubscribes user status.

Simple Message APIs

Use the following APIs for the sending and receiving of text and signaling (custom buffer) messages.

API	Description
addSimpleMsgListener	Sets an event listener for simple messages (text messages and custom messages). Do not use it and addAdvancedMsgListener at the same time.
removeSimpleMsgListener	Removes the event listener for simple messages (text messages and custom messages).
sendC2CTextMessage	Sends a one-to-one (C2C) text message.
sendC2CCustomMessage	Sends a one-to-one (C2C) custom (signaling) message.
sendGroupTextMessage	Sends a group text message.
sendGroupCustomMessage	Sends a group custom (signaling) message.

Signaling APIs

API	Description
addSignalingListener	Adds a signaling listener.
removeSignalingListener	Removes a signaling listener.
invite	Invites a user.
inviteInGroup	Invites certain users in the group.
cancel	Cancels an invitation.
accept	Accepts an invitation.
reject	Rejects an invitation.
getSignalingInfo	Gets the signaling information.
addInvitedSignaling	Adds invitation signaling (can be used for invitation signaling triggered by offline push messages for group invitations).
modifyInvitation	modify invitation.

Advanced Message APIs

If you need to send/receive rich media messages (images, videos, files, etc.) and use advanced features such as recalling messages, marking messages as read, and querying message history, use the following advanced message APIs. Do not use simple messages APIs and advanced message APIs at the same time.

API	Description
addAdvancedMsgListener	Sets an event listener for advanced messages. Do not use it and addSimpleMsgListener at the same time.
removeAdvancedMsgListener	Removes the listener for advanced messages.
createTextMessage	Creates a text message.
createTextAtMessage	Creates an @ text message.
createCustomMessage	Creates a custom message.
createImageMessage	Creates an image message.
createSoundMessage	Creates a voice message.
createVideoMessage	Creates a video message.
createFileMessage	Creates a file message.
createLocationMessage	Creates a location message.
createFaceMessage	Creates an emoji message.
createMergerMessage	Creates a combined forward message.
createForwardMessage	Creates a single forward message.
createTargetedGroupMessage	Creates a targeted group message
sendMessage	Sends a message. The message object can be created using a createXXXMessage API.
setC2CReceiveMessageOpt	Sets the Mute Notifications option for one-to-one messages.
getC2CReceiveMessageOpt	Gets the Mute Notifications status for one-to-one messages.
setGroupReceiveMessageOpt	Sets the Mute Notifications option for group messages.
getC2CHistoryMessageList	Gets one-to-one (C2C) message history.
getGroupHistoryMessageList	Gets group chat message history.
getHistoryMessageList	Gets message history.

revokeMessage	Recalls a message. The message object can be created using a createXXXMessage API.
modifyMessage	Modifies a message. The message object can be created using a createXXXMessage API.
markC2CMessageAsRead	Marks one-to-one (C2C) messages as read.
markGroupMessageAsRead	Marks group messages as read.
markAllMessageAsRead	Marks all messages as read.
deleteMessageFromLocalStorage	Deletes a message from local storage.
deleteMessages	Deletes messages from local storage and the cloud.
clearC2CHistoryMessage	Clears chat history with a user from local storage and the cloud.
clearGroupHistoryMessage	Clears chat history of a group from local storage and the cloud.
insertGroupMessageToLocalStorage	Inserts a message in a group chat.
insertC2CMessageToLocalStorage	Inserts a message in a one-to-one chat.
findMessages	Finds local messages by msgID.
searchLocalMessages	Search for local messages.
searchCloudMessages	Search for cloud messages.
sendMessageReadReceipts	Sends read receipts to the message sender for message which have been received.
getMessageReadReceipts	Gets read receipts for messages which have been sent by myself.
getGroupMessageReadMemberList	Gets group members profile who have read this message.
setMessageExtensions	Sets message extensions
getMessageExtensions	Gets message extensions
deleteMessageExtensions	Deletes message extensions
addMessageReaction	Adds message reaction.
removeMessageReaction	Removes message reaction.
getMessageReactions	Gets message reactions.
getAllUserListOfMessageReaction	Gets all user list of message reaction.

pinGroupMessage	Sets group message pinning.
getPinnedGroupMessageList	Gets pinned group message list.

Group APIs

Tencent Cloud Chat SDK supports five preset group types, each of which pertains to different scenarios.

Work group (Work): users can join the group only after being invited by group members. This group type is the same as private group (Private) in earlier versions.

Public group (Public): Users can join a public group through requests, which need to be approved by the group owner or group admin.

Meeting group (Meeting): used together with [TRTC](#) to enable scenarios such as video conferencing and online education. Users can join and leave the group freely and view the message history before they join. Same as chat room (ChatRoom) in earlier versions.

Community: A user can join and leave a community freely. It is suitable for chat scenarios with a super large number of community members, such as knowledge sharing and game discussion. This feature is supported by a client with the SDK enhanced edition v5.8.1668 or later and the web SDK v2.17.0 or later. To use it, you need to purchase the [Pro edition](#)、[Pro Plus edition](#)、[Enterprise edition](#), and then enable it in [Console](#), path: Applications > Your App > Chat > Configuration > Group Configuration > Community.

Audio-video group (AVChatRoom): An audio-video group allows users to join and leave freely and is suitable for scenarios such as live streaming and chat rooms with on-screen comments. There is no limit on the number of group members.

API	Description
addGroupListener	Adds an event listener for groups.
removeGroupListener	Removes event listener for groups.
createGroup	Creates a (simple) group.
createGroup	Creates an (advanced) group. The group information and the initial group members can be set during group creation.
joinGroup	Joins a group.
quitGroup	Quits a group.
dismissGroup	Deletes a group. Only the group owner and group admin can delete a group.
getJoinedGroupList	Gets the list of groups the current user has joined, excluding audio-video groups.

getGroupsInfo	Pulls profiles of groups.
searchGroups	Search for local group profiles.
searchCloudGroups	Search for cloud group profiles.
setGroupInfo	Modifies the profile of a group.
initGroupAttributes	Initializes group attributes.
setGroupAttributes	Sets group attributes.
deleteGroupAttributes	Deletes group attributes.
getGroupAttributes	Gets group attributes.
getGroupOnlineMemberCount	Gets the number of online group members.
getGroupMemberList	Gets group member list.
getGroupMembersInfo	Gets the profiles of specified group members.
searchGroupMembers	Search for local group members.
searchCloudGroupMembers	Search for cloud group members.
setGroupMemberInfo	Modifies the profile of a specified group member.
muteGroupMember	Mutes a group member.
inviteUserToGroup	Invites users to a group.
kickGroupMember	Removes a member from a group.
setGroupMemberRole	Sets the role for a group member.
markGroupMemberList	Mark group member list .
transferGroupOwner	Changes the group owner.
getGroupApplicationList	Gets the list of requests to join a group.
acceptGroupApplication	Accepts a request to join a group.
refuseGroupApplication	Rejects a request to join a group.
setGroupApplicationRead	Marks the request list as read.
getJoinedCommunityList	Gets the list of Community-groups that the current user has joined

createTopicInCommunity	Creates topic
deleteTopicFromCommunity	Deletes topic
setTopicInfo	Modifies topic information
getTopicInfoList	Gets topic list

Conversation List APIs

The conversation list is the list a user sees on the first screen after logging in to WeChat or QQ. It includes elements such as conversation node, conversation name, group name, last message, and unread count.

API	Description
addConversationListener	Adds a conversation listener.
removeConversationListener	Removes conversation listener.
getConversationList	Gets the conversation list.
getConversation	Gets a conversation.
getConversationList	Gets multiple conversations.
getConversationListByFilter	Gets multiple conversations (advanced API).
deleteConversation	Deletes a conversation.
setConversationDraft	Sets draft for a conversation.
setConversationCustomData	Sets custom data for a conversation.
pinConversation	Pins a conversation to the top.
markConversation	Marks a conversation to the top.
getTotalUnreadMessageCount	Gets the total unread message count.
createConversationGroup	Create conversation group.
getConversationGroupList	Get conversation group list.
deleteConversationGroup	Delete conversation group.
renameConversationGroup	Rename conversation group.

addConversationsToGroup	Add conversations to group.
deleteConversationsFromGroup	Delete conversations from group.

User Profile APIs

You can use the following APIs to query user profiles, modify your profile, and block messages from a specified user (i.e., adding a specified user to the blacklist).

API	Description
getUsersInfo	Gets users' profiles.
setSelfInfo	Modifies one's own user profile.
searchUsers	Search cloud users
addToBlackList	Blocks messages from a specified user, which means adding the user to the blacklist.
deleteFromBlackList	Unblocks messages from a specified user, which means removing the user from the blacklist.
getBlackList	Gets the blacklist.

Offline Push APIs

Use the offline push service if you want your app to receive IM messages in real time when it runs in the background. As there is currently no unified push service in the Chinese mainland, you need to [Offline Push\(iOS\)](#).

API	Description
setAPNSListener	Sets an APNs listener.
setAPNS	Configures APNS push.
setVOIP	Configures VOIP push.

Friend Management APIs

Tencent Cloud Chat SDK does not check friend relationships by default when sending and receiving messages. To achieve the interactive experience of "add friends first, then send messages", you can log in to the [Console](#) to modify Relationship Check. When enabled, users can only send messages to their friends. When a user sends a message to a non-friend, SDK will report a 20009 error code. The configuration path is: Applications > Your App > Chat > Configuration > Login and Message > Relationship Check.

Use the following API to add or delete friends and manage friend lists.

API	Description
addFriendListener	Adds a Contacts listener to receive friend list and blocklist change events.
removeFriendListener	Removes Contacts listener.
getFriendList	Gets the friend list.
getFriendsInfo	Gets the profiles of specified friends.
setFriendInfo	Sets the profile of a specified friend.
searchFriends	Searches for friends.
addFriend	Adds a friend.
deleteFromFriendList	Deletes a friend.
checkFriend	Checks relationship with specified users.
getFriendApplicationList	Gets the list of friend requests.
acceptFriendApplication	Accepts a friend request.
refuseFriendApplication	Rejects a friend request.
deleteFriendApplication	Deletes a friend request.
setFriendApplicationRead	Marks a friend request as read.
createFriendGroup	Creates a friend list.
getFriendGroupList	Gets the information of friend lists.
deleteFriendGroup	Deletes friend lists.
renameFriendGroup	Modifies the name of a friend list.
addFriendsToFriendGroup	Adds friends to a friend list.
deleteFriendsFromFriendGroup	Deletes friends from a friend list.

Flutter

Last updated : 2025-04-16 15:00:24

Initialization and Login APIs

To use the Tencent Cloud Chat service, you need to initialize the SDK and log in.

API	Description
initSDK	Initializes the SDK.
unInitSDK	Uninitializes the SDK.
login	Logs in.
logout	Logs out.
getLoginUser	Gets the UserID of the current login user.
getLoginStatus	Gets the login status.
getServerTime	Gets the current server time (not supported on web).
getVersion	Gets the version number.
getConversationManager	Conversation feature entry.
getFriendshipManager	Contacts feature entry.
getGroupManager	Advanced group feature entry.
getMessageManager	Advanced message feature entry.
getOfflinePushManager	Gets the version number.
getSignalingManager	Signaling entry.

Signaling APIs

API	Description
addSignalingListener	Adds a signaling listener.

<code>removeSignalingListener</code>	Removes a signaling listener.
<code>invite</code>	Invites a user.
<code>inviteInGroup</code>	Invites certain users in the group.
<code>cancel</code>	The inviter cancels the invitation.
<code>accept</code>	The invitee accepts the invitation.
<code>reject</code>	The invitee rejects the invitation.
<code>getSignalingInfo</code>	Gets signaling information.
<code>addInvitedSignaling</code>	Creates a signaling request.

Message Creation APIs

After a message is created, an `id` field will be returned. You can pass in the `id` field and related information to the message sending API (`sendMessage`) to send the message.

API	Description
<code>createTextMessage</code>	Creates a text message.
<code>createCustomMessage</code>	Creates a custom message.
<code>createImageMessage</code>	Creates an image message.
<code>createSoundMessage</code>	Creates an audio message.
<code>createVideoMessage</code>	Creates a video message.
<code>createTextAtMessage</code>	Creates an @ text message.
<code>createFileMessage</code>	Creates a file message.
<code>createLocationMessage</code>	Creates a location message.
<code>createFaceMessage</code>	Creates an emoji message.
<code>createMergerMessage</code>	Creates a combined message.
<code>createForwardMessage</code>	Creates a forward message.
<code>createTargetedGroupMessage</code>	Creates a targeted group message.

[appendMessage](#)

Adds a multi-element message.

Message Sending and Receiving APIs

If you need to send or receive rich media messages (such as image, video, and file messages) and use advanced features such as recalling messages, marking messages as read, and querying message history, we recommend the following advanced message APIs. (The original APIs for simple message sending and receiving used by v3.6.0 and earlier versions have been disused. Please use new APIs to create and send messages.)

API	Description
addAdvancedMsgListener	Sets an event listener for advanced messages.
removeAdvancedMsgListener	Removes the event listener for advanced messages.
getC2CHistoryMessageList	Gets historical one-to-one (C2C) messages.
getHistoryMessageList	Gets historical messages (advanced API).
getGroupHistoryMessageList	Gets historical group messages.
markC2CMessageAsRead	Marks one-to-one (C2C) messages as read.
markGroupMessageAsRead	Marks group messages as read.
markAllMessageAsRead	Marks all messages as read.
deleteMessageFromLocalStorage	Deletes a message from the local storage.
deleteMessages	Deletes local and roaming messages.
insertGroupMessageToLocalStorage	Adds a message to the message list of a group chat.
insertC2CMessageToLocalStorage	Adds a message to the message list of a one-to-one chat.
clearC2CHistoryMessage	Clears the chat history with a user from local storage and the cloud (without deleting the conversation).
clearGroupHistoryMessage	Clears the chat history of a group from local storage and the cloud (without deleting the conversation).
downloadMergerMessage	Gets the sub messages of a combined message.
setC2CReceiveMessageOpt	Sets the one-to-one message receiving option for a user (batch setting supported).

getC2CReceiveMessageOpt	Queries the one-to-one message receiving option of a user.
setGroupReceiveMessageOpt	Modifies the group message receiving option.
setLocalCustomData	Sets custom message data (saved locally, will not be sent to the peer end, and will become invalid after the app is uninstalled and reinstalled).
setLocalCustomInt	Sets custom message data and marks whether a voice or video message is played (such a message will be saved locally, not be sent to the peer end, and become invalid after the app is uninstalled and reinstalled).
revokeMessage	Recalls a message. By default, you can recall only messages that are sent within two minutes. You can customize the time limit for message recall via the console (Feature Configuration > Login and Message > Message Recall Settings).
modifyMessage	Modifies a message.
sendMessage	Sends a message.
searchLocalMessages	Searches for local messages.
sendMessageReadReceipts	Sends group message read receipts.
getMessageReadReceipts	Gets read receipts for messages sent by yourself.
getGroupMessageReadMemberList	Gets the list of members who have (or have not) read a message sent by yourself.

Group APIs

Tencent Cloud Chat SDK supports five preset group types, each of which pertains to different scenarios.

Work group (Work): users can join the group only after being invited by group members. This group type is the same as private group (Private) in earlier versions.

Public group (Public): Users can join a public group through requests, which need to be approved by the group owner or group admin.

Meeting group (Meeting): used together with [TRTC](#) to enable scenarios such as video conferencing and online education. Users can join and leave the group freely and view the message history before they join. Same as chat room (ChatRoom) in earlier versions.

Community: A user can join and leave a community freely. It is suitable for chat scenarios with a super large number of community members, such as knowledge sharing and game discussion. This feature is supported by a client with the

SDK enhanced edition v5.8.1668 or later and the web SDK v2.17.0 or later. To use it, you need to purchase the [Pro edition](#)、[Pro Plus edition](#)、[Enterprise edition](#), and then enable it in [Console](#), path: Applications > Your App > Chat > Configuration > Group Configuration > Community.

Audio-video group (AVChatRoom): An audio-video group allows users to join and leave freely and is suitable for scenarios such as live streaming and chat rooms with on-screen comments. There is no limit on the number of group members.

API	Description
addGroupListener	Adds an event listener for groups.
removeGroupListener	Removes an event listener for groups.
createGroup	Creates a group (advanced). The group information and the initial group members can be set during group creation.
joinGroup	Joins a group.
quitGroup	Leaves a group.
dismissGroup	Deletes a group. Only the group owner and group admin can delete the group.
getJoinedGroupList	Gets the list of groups the current user has joined, excluding audio-video groups.
getGroupsInfo	Gets the profiles of groups.
setGroupInfo	Modifies the profile of a group.
initGroupAttributes	Initializes group attributes. This will clear the existing group attribute list.
setGroupAttributes	Sets group attributes. If the group attributes already exist, their values are updated. Otherwise, the group attributes are added.
deleteGroupAttributes	Deletes specified group attributes. Passing in <code>null</code> for <code>keys</code> means clearing all group attributes.
getGroupAttributes	Gets specified group attributes. Passing in <code>null</code> for <code>keys</code> means getting all group attributes.
searchGroups	Searches for groups.
getGroupOnlineMemberCount	Gets the number of online users in a group. (This API is currently supported only by audio-video groups.)
getGroupMemberList	Gets the group member list.

getGroupMembersInfo	Gets the profiles of specified group members.
setGroupMemberInfo	Modifies the profile of a specified group member.
searchGroupMembers	Searches for group members.
muteGroupMember	Mutes a group member.
kickGroupMember	Removes members from a group.
setGroupMemberRole	Sets a role for a group member.
transferGroupOwner	Transfers the ownership of a group.
inviteUserToGroup	Invites users to a group.
getGroupApplicationList	Gets the list of requests to join a group.
acceptGroupApplication	Accepts a request to join a group.
refuseGroupApplication	Rejects a request to join a group.
setGroupApplicationRead	Marks the request list as read.
getJoinedCommunityList	Gets the list of community groups the current user has joined.
createTopicInCommunity	Creates a topic.
deleteTopicFromCommunity	Deletes a topic.
setTopicInfo	Sets topic attributes.
getTopicInfoList	Gets the topics of a group.

Conversation List APIs

The conversation list is the list a user sees on the first screen after login. It includes elements such as conversation node, conversation name, group name, last message, and unread count.

API	Description
addConversationListener	Adds a contacts listener.
removeConversationListener	Removes a contacts listener.
getConversationList	Gets the conversation list.

getConversationListByConversaionIds	Gets the list of specified conversations by conversation IDs.
pinConversation	Pins a conversation to the top.
getTotalUnreadMessageCount	Gets the total unread message count of a conversation.
getConversation	Gets a specified conversation.
deleteConversation	Deletes a conversation.
setConversationDraft	Sets a draft for a conversation.

User Profile APIs

You can use the following APIs to query user profiles, modify your profile, and block messages from a specified user (that is, adding a specified user to the blacklist).

API	Description
getUsersInfo	Gets user profiles.
getUserStatus	Gets users' online statuses.
setSelfInfo	Modifies one's own user profile.
setSelfStatus	Sets the status of the current logged-in user.
addToBlackList	Blocks messages from a user, which means adding the user to the blacklist.
deleteFromBlackList	Unblocks messages from a user, which means removing the user from the blacklist.
getBlackList	Gets the blacklist.

Offline Push APIs

We recommend you use the offline push service if you want your app to receive Chat service messages in real time when it runs in the background. As there is no unified push service in the Chinese mainland, you need to configure Android offline push for devices of different vendors separately.

API	Description
setAPNSListener	Sets the APNs offline push listener.

doBackground	Configures offline push.
doForeground	Configures offline push.
setOfflinePushConfig	Configures offline push.

Friend Management APIs

Tencent Cloud Chat SDK does not check friend relationships by default when sending and receiving messages. To achieve the interactive experience of "add friends first, then send messages", you can log in to the [Console](#) to modify Relationship Check. When enabled, users can only send messages to their friends. When a user sends a message to a non-friend, SDK will report a 20009 error code. The configuration path is: Applications > Your App > Chat > Configuration > Login and Message > Relationship Check.

Use the following API to add or delete friends and manage friend lists.

API	Description
addFriendListener	Adds a contacts listener.
removeFriendListener	Removes a contacts listener.
getFriendList	Gets the contacts.
getFriendsInfo	Gets the profiles of specified friends.
setFriendInfo	Sets the profile of a specified friend.
addFriend	Adds a friend.
deleteFromFriendList	Deletes friends.
checkFriend	Checks your relationship with a specified user.
getFriendApplicationList	Gets the list of friend requests.
acceptFriendApplication	Accepts a friend request.
refuseFriendApplication	Rejects a friend request.
deleteFriendApplication	Deletes a friend request.
setFriendApplicationRead	Marks a friend request as read.
createFriendGroup	Creates a friend list.
getFriendGroups	Gets the information of friend lists.

<code>deleteFriendGroup</code>	Deletes a friend list.
<code>renameFriendGroup</code>	Modifies the name of a friend list.
<code>addFriendsToFriendGroup</code>	Adds friends to a friend list.
<code>deleteFriendsFromFriendGroup</code>	Deletes friends from a friend list.
<code>searchFriends</code>	Searches for friends.

Electron

Last updated : 2024-05-20 15:43:47

Event Callback APIs

API	Description
TIMAddRecvNewMsgCallback	Adds the callback for receiving new messages.
TIMRemoveRecvNewMsgCallback	Deletes the callback for receiving new messages.
TIMSetMsgReadedReceiptCallback	Sets the callback for message read receipts.
TIMSetMsgRevokeCallback	Sets the callback for message recalls.
TIMSetMsgElemUploadProgressCallback	Sets the callback for the upload progress of message element files.
TIMSetGroupTipsEventCallback	Sets the callback for group system messages.
TIMSetConvEventCallback	Sets the callback for conversation events.
TIMSetNetworkStatusListenerCallback	Sets the callback for network connection status.
TIMSetKickedOfflineCallback	Sets the callback for notifications of forced logout.
TIMSetUserSigExpiredCallback	Sets the callback for user ticket expiration.
TIMSetOnAddFriendCallback	Sets the callback for adding friends.
TIMSetOnDeleteFriendCallback	Sets the callback for deleting friends.
TIMSetUpdateFriendProfileCallback	Sets the callback for updating friends' profiles.
TIMSetFriendAddRequestCallback	Sets the callback for friend requests.
TIMSetLogCallback	Sets the callback for logs.
TIMSetMsgUpdateCallback	Sets the callback for message update notifications returned after messages are modified in the cloud.

Chat SDK Initialization APIs

API	Description
TIMInit	Initializes the Chat SDK.
TIMUninit	Uninstalls the Chat SDK.
TIMGetSDKVersion	Gets Chat SDK version number.
TIMSetConfig	Sets extra user configurations.

Login and Logout APIs

API	Description
TIMLogin	Logs in.
TIMLogout	Logs out.

Conversation APIs

API	Description
TIMConvCreate	Creates a conversation.
TIMConvDelete	Deletes a conversation.
TIMConvGetConvList	Gets the conversation list of recent contacts.
TIMConvSetDraft	Sets the draft of a conversation.
TIMConvCancelDraft	Deletes the draft of a conversation.
TIMConvGetConvInfo	Gets a conversation list.
TIMConvGetTotalUnreadMessageCount	Gets the total unread message count of all conversations.
TIMConvPinConversation	Pins a conversation to the top.

Message APIs

API	Description
-----	-------------

TIMMsgBatchSend	Sends a message to multiple users at a time, but not to groups.
TIMMsgCancelSend	Cancels a message being sent based on <code>messageID</code> .
TIMMsgClearHistoryMessage	Clears the messages of a conversation.
TIMMsgDelete	Deletes the messages of a conversation.
TIMMsgDeleteMessageExtensions	Deletes message extensions.
TIMMsgDownloadElemToPath	Downloads message elements to a path.
TIMMsgDownloadMergerMessage	Downloads a combined message.
TIMMsgFindByMsgLocatorList	Locates a message of a conversation by using the message locator.
TIMMsgFindMessages	Queries local messages by <code>messageID</code> .
TIMMsgGetC2CReceiveMessageOpt	Queries the one-to-one message receiving option of a user.
TIMMsgGetMessageExtensions	Gets message extensions.
TIMMsgGetMsgList	Gets the message list of a specified conversation.
TIMMsgImportMsgList	Imports a message list to a conversation.
TIMMsgListDelete	Deletes the local and roaming messages of a conversation.
TIMMsgModifyMessage	Modifies a message.
TIMMsgReportReaded	Marks a message as read.
TIMMsgRevoke	Recalls a message.
TIMMsgSaveMsg	Saves a custom message.
TIMMsgSearchLocalMessages	Searches for local messages.
TIMMsgSendNewMsg	Sends a new message.
TIMMsgSendMessageV2	Sends a new message. (V2)
TIMMsgSetC2CReceiveMessageOpt	Sets the one-to-one message receiving option of a user.
TIMMsgSetGroupReceiveMessageOpt	Sets the group message receiving option.
TIMMsgSetMessageExtensions	Sets message extensions.
TIMMsgTranslateText	Translates text message.

TIMMsgConvertVoiceToText

Converts voice message to text message.

Group APIs

API	Description
TIMGroupCreate	Creates a group.
TIMGroupDelete	Disbands a group.
TIMGroupDeleteGroupAttributes	Deletes group attributes.
TIMGroupDeleteMember	Deletes a group member.
TIMGroupGetGroupAttributes	Gets specified group attributes.
TIMGroupGetGroupInfoList	Gets group information.
TIMGroupGetJoinedGroupList	Gets the list of groups that a user has joined.
TIMGroupGetMemberInfoList	Gets group member information.
TIMGroupGetOnlineMemberCount	Gets the number of online members of a group.
TIMGroupGetPendencyList	Gets the list of pending group requests.
TIMGroupHandlePendency	Handles pending group requests.
TIMGroupInitGroupAttributes	Initializes group attributes.
TIMGroupInviteMember	Invites someone to join a group.
TIMGroupJoin	Requests to join a group.
TIMGroupModifyGroupInfo	Modifies group information.
TIMGroupModifyMemberInfo	Modifies group member information.
TIMGroupQuit	Leaves a group.
TIMGroupReportPendencyReaded	Reports that a pending group request is read.
TIMGroupSearchGroupMembers	Searches for group members.
TIMGroupSearchGroups	Searches for groups.
TIMGroupSetGroupAttributes	Sets group attributes.

TIMMsgGetGroupMessageReadMemberList	Gets the list of members who have or have not read the group message.
TIMMsgGetMessageReadReceipts	Gets message read receipts.
TIMMsgSendMessageReadReceipts	Sends message read receipts.

User Profile APIs

API	Description
TIMProfileGetUserProfileList	Gets the profiles of specified users.
TIMProfileModifySelfUserProfile	Modifies one's own profile.

Relationship Chain APIs

API	Description
TIMFriendshipAddFriend	Adds friends.
TIMFriendshipAddToBlackList	Adds a user to the blacklist.
TIMFriendshipCheckFriendType	Checks the friend type (one-way or two-way).
TIMFriendshipCreateFriendGroup	Creates a friend list.
TIMFriendshipDeleteFriend	Deletes friends.
TIMFriendshipDeleteFriendGroup	Deletes a friend list.
TIMFriendshipDeleteFromBlackList	Removes a user from the blacklist.
TIMFriendshipDeletePendency	Deletes a pending friend request.
TIMFriendshipGetBlackList	Gets the blacklist.
TIMFriendshipGetFriendGroupList	Gets the information of a friend list.
TIMFriendshipGetFriendProfileList	Gets friend lists.
TIMFriendshipGetFriendsInfo	Gets friend information.
TIMFriendshipGetPendencyList	Gets the pending friend request list.

TIMFriendshipHandleFriendAddRequest	Handles a friend request.
TIMFriendshipModifyFriendGroup	Modifies a friend list.
TIMFriendshipModifyFriendProfile	Updates friends' profiles (such as remarks).
TIMFriendshipReportPendencyReaded	Reports that a pending friend request is read.
TIMFriendshipSearchFriends	Searches for friends.

Unity

Last updated : 2024-02-07 17:02:49

Note

For API details, see [here](#).

Conversation APIs

API	Description
ConvCancelDraft	Cancels a draft of a conversation.
ConvDelete	Deletes a conversation.
ConvGetConvInfo	Gets the information of conversations.
ConvGetConvList	Gets the conversation list.
ConvGetTotalUnreadMessageCount	Gets the total unread count of all conversations.
ConvPinConversation	Pins a conversation to the top.
ConvSetDraft	Sets a draft for a conversation.

Relationship Chain APIs

API	Description
FriendshipAddFriend	Adds a friend.
FriendshipAddToBlackList	Adds a user to the blacklist.
FriendshipCheckFriendType	Checks the friend type.
FriendshipCreateFriendGroup	Creates a friend list.
FriendshipDeleteFriend	Deletes a friend.
FriendshipDeleteFriendGroup	Deletes a friend list.
FriendshipDeleteFromBlackList	Removes a user from the blacklist.

FriendshipDeletePendency	Deletes a pending friend request.
FriendshipGetBlackList	Gets the blocklist.
FriendshipGetFriendGroupList	Gets the information of a friend list.
FriendshipGetFriendProfileList	Gets the profiles of all friends.
FriendshipGetFriendsInfo	Gets the information of friends.
FriendshipGetPendencyList	Gets the list of pending group requests.
FriendshipHandleFriendAddRequest	Handles a friend request.
FriendshipModifyFriendGroup	Modifies a friend list.
FriendshipModifyFriendProfile	Modifies the profile of a friend.
FriendshipReportPendencyReaded	Reports that a pending friend request is read.
FriendshipSearchFriends	Searches for friends.

Group APIs

API	Description
GroupCreate	Creates a group.
GroupCreateTopicInCommunity	Creates a topic.
GroupDelete	Disbands a group.
GroupDeleteGroupAttributes	Deletes custom group attributes.
GroupDeleteMember	Deletes a group member.
GroupDeleteTopicFromCommunity	Deletes a topic.
GroupGetGroupAttributes	Gets group attributes.
GroupGetGroupInfoList	Gets group information.
GroupGetJoinedCommunityList	Gets the list of community groups the current user has joined.
GroupGetJoinedGroupList	Gets the list of groups that a user has joined.
GroupGetMemberInfoList	Gets group member information.

GroupGetOnlineMemberCount	Gets the number of online group members.
GroupGetPendencyList	Gets the list of pending group requests.
GroupGetTopicInfoList	Gets the list of topics.
GroupHandlePendency	Handles pending group requests.
GroupInitGroupAttributes	Initializes custom group attributes.
GroupInviteMember	Invites a user to join a group.
GroupJoin	Joins a group.
GroupModifyGroupInfo	Modifies group information.
GroupModifyMemberInfo	Modifies group member information.
GroupQuit	Leaves a group.
GroupReportPendencyReaded	Reports that a pending group request is read.
GroupSearchGroupMembers	Searches for group members.
GroupSearchGroups	Searches for group profiles.
GroupSetGroupAttributes	Sets group attributes.
GroupSetTopicInfo	Modifies topic information.

Chat SDK Initialization APIs

API	Description
GetSDKVersion	Gets the version of the underlying library of the Chat SDK.
GetServerTime	Gets the server time (seconds).
Init	Initializes the Chat SDK.
SetConfig	Sets global configuration.
Uninit	Uninitializes the Chat SDK.

Login APIs

API	Description
GetLoginStatus	Gets the current login status.
GetLoginUserID	Gets the ID of the current login user.
Login	Logs in.
Logout	Logs out.

Message APIs

API	Description
GetMsgGroupMessageReadMemberList	Gets the list of members who have read the group message.
MsgBatchSend	Sends a message to multiple users at a time.
MsgCancelSend	Cancels the sending of a message.
MsgClearHistoryMessage	Clears historical messages.
MsgDelete	Deletes the messages of a conversation.
MsgDoBackground	This API can be called when the app detects that the app switches to the background.
MsgDoForeground	This API can be called when the app detects that the app switches to the foreground.
MsgDownloadElemToPath	Downloads a multimedia message.
MsgDownloadMergerMessage	Downloads a combined message.
MsgFindByMsgLocatorList	Locates a message by using a message locator.
MsgFindMessages	Locates local messages.
MsgGetC2CReceiveMessageOpt	Gets the message receiving option for a one-to-one chat.
MsgGetMessageReadReceipts	Gets message read receipts.
MsgGetMsgList	Gets the historical message list of a conversation.

MsgImportMsgList	Imports messages.
MsgListDelete	Deletes messages.
MsgMarkAllMessageAsRead	Marks all messages as read.
MsgModifyMessage	Modifies a message.
MsgReportReaded	Reports that a one-to-one message is read.
MsgRevoke	Recalls a message.
MsgSaveMsg	Saves a message.
MsgSearchLocalMessages	Searches for local messages.
MsgSendMessage	Sends a message.
MsgSendMessageReadReceipts	Sends message read receipts.
MsgSetC2CReceiveMessageOpt	Sets the one-to-one message receiving option.
MsgSetGroupReceiveMessageOpt	Sets the group message receiving option.
MsgSetLocalCustomData	Sets local message data.
MsgSetOfflinePushToken	Sets offline push configuration.

User Profile APIs

API	Description
GetUserStatus	Queries a user's status.
ProfileGetUserProfileList	Gets the profiles of specified users.
ProfileModifySelfUserProfile	Modifies one's own profile.
SetSelfStatus	Sets one's own status.
SubscribeUserStatus	Subscribes to a user's status.
UnsubscribeUserStatus	Unsubscribes from a user's status.

SDK Callback Registration APIs

API	Description
AddRecvNewMsgCallback	Adds the callback for receiving new messages.
RemoveRecvNewMsgCallback	Removes the callback for receiving new messages.
SetConvEventCallback	Sets the callback for conversation events.
SetConvTotalUnreadMessageCountChangedCallback	Sets the callback for a change in the total unread message count of conversations.
SetFriendAddRequestCallback	Sets the callback for friend requests.
SetFriendApplicationListDeletedCallback	Sets the callback for deleting a friend request.
SetFriendApplicationListReadCallback	Sets the callback for reading a friend request.
SetFriendBlackListAddedCallback	Sets the callback for adding a friend to the blocklist.
SetFriendBlackListDeletedCallback	Sets the callback for removing a friend from the blocklist.
SetGroupAttributeChangedCallback	Sets the callback for a group attribute change.
SetGroupTipsEventCallback	Sets the callback for group system messages.
SetGroupTopicChangedCallback	Sets the callback for topic updates.
SetGroupTopicCreatedCallback	Sets the callback for topic creation.
SetGroupTopicDeletedCallback	Sets the callback for topic deletion.
SetKickedOfflineCallback	Sets the callback for notifications of forced logout.
SetLogCallback	Sets the callback for logs.
SetMsgElemUploadProgressCallback	Sets the callback for the upload progress of message element files.
SetMsgReadedReceiptCallback	Sets the callback for message read receipts.
SetMsgRevokeCallback	Sets the callback for message recalls.
SetMsgUpdateCallback	Sets the callback for message update notifications returned after messages are modified in the cloud.
SetNetworkStatusListenerCallback	Sets the callback for network connection status.
SetOnAddFriendCallback	Sets the callback for adding friends.

SetOnDeleteFriendCallback	Sets the callback for deleting friends.
SetOnDeleteFriendCallback	Sets the callback for deleting friends.
SetSelfInfoUpdatedCallback	Sets the callback for updating the current user's profile.
SetUpdateFriendProfileCallback	Sets the callback for updating friends' profiles.
SetUserSigExpiredCallback	Sets the callback for user ticket expiration.
SetUserStatusChangedCallback	Sets the callback for user status update notifications.

React Native

Last updated : 2025-01-13 12:01:25

TencentCloudChat

`TencentCloudChat` is the namespace of the Chat JavaScript SDK and provides the static method `create()` for creating SDK instances, the event constant `EVENT` , and the type constant `TYPES` , and the signaling constant `TSignaling` .

Initialization

API	Description
<code>create</code>	Creates an SDK instance.

SDK Instance

Term	Description
<code>Message</code>	<code>Message</code> indicates the content to be sent and carries multiple attributes which specify whether you are the sender, the sender account, the message generation time, and so on.
<code>Conversation</code>	Two types of conversation are available: Client to Client (C2C): A one-to-one chat, involving only two participants. GROUP: A group chat, involving more than two participants.
<code>Profile</code>	<code>Profile</code> describes the basic information of a user, including the nickname, gender, personal signature, and profile photo URL.
<code>Friend</code>	<code>Friend</code> describes the basic information of a friend, including the remarks and friend list.
<code>FriendApplication</code>	<code>FriendApplication</code> describes the basic information of a friend request, including the friend source and remarks.
<code>FriendGroup</code>	<code>FriendGroup</code> describes the basic information of a friend group, including the friend group name and members.
<code>Group</code>	<code>Group</code> indicates a communication system for group chatting, including Work Group, Public Group, Meeting Group, Audio-Video Group, and Community Group.
<code>GroupMember</code>	<code>GroupMember</code> indicates the basic information of each group member, such as the ID, nickname, role, and the time of joining the group.

Group tip	A group tip is generated when an event such as adding or deleting group member occurs. The access side can configure whether to display group tips to group members.
Group system message	For example, when a user requests to join a group, the group admin receives a system message. After the admin accepts or rejects the request, the Chat SDK returns the result to the access side, which then displays the result to the user.
Message display on screen	The sent messages, including text segments and images, are displayed on the computer or phone screen.

Event

API	Description
on	Enables event listening.
off	Disables event listening.

Plugin registration

API	Description
registerPlugin	Registers a plugin.

Setting log level

API	Description
setLogLevel	Sets the log level. Logs below this level will not be printed.

SDK instance termination

API	Description
destroy	Terminates an SDK instance. The SDK will log out, disconnect the WebSocket persistent connection, and then release resources.

Login

API	Description
login	Logs in to the Chat SDK using userID and userSig.

logout	Logs out to the Chat SDK.
isReady	Indicates the SDK enter the ready state or not.
getLoginUser	Get the userID of the logged-in user. If the user is not logged in, it returns an empty string ("").
getServerTime	Get the server time.

Message

API	Description
createTextMessage	Creates a text message.
createTextAtMessage	Creates a text message with the @ notification feature.
createImageMessage	Creates an image message.
createAudioMessage	Creates an audio message.
createVideoMessage	Creates a video message.
createCustomMessage	Creates a custom message.
createFaceMessage	Creates an emoji message.
createFileMessage	Creates a file message.
createLocationMessage	Creates a location message.
createMergerMessage	Creates a combined message.
downloadMergerMessage	Downloads a combined message.
createForwardMessage	Creates a forward message.
sendMessage	Sends a message.
revokeMessage	Recalls a message.
deleteMessage	Deletes a message.
translateText	Translates a message.
setMessageExtensions	Sets message extensions.
getMessageExtensions	Gets message extensions.

deleteMessageExtensions	Deletes message extensions.
addMessageReaction	Add message reactions. This interface allows for scenarios such as "emoji reactions" to be implemented.
removeMessageReaction	Remove message reactions.
getMessageReactions	Retrieve information of message reactions.
getAllUserListOfMessageReaction	Paginate and retrieve the user list for a specific message reaction.

Conversation

API	Description
modifyMessage	Modifies a message.
getMessageList	Gets the message list.
getMessageListHopping	Pulls the conversation message list by specified sequence or time range.
sendMessageReadReceipt	Sends message read receipts.
getMessageReadReceiptList	Pulls the list of message read receipts.
getGroupMessageReadMemberList	Pulls the list of members who have (or have not) read a group message.
findMessage	Queries local messages in a specified conversation by <code>messageID</code> .
setMessageRead	Marks a message as read.
getConversationList	Gets the conversation list.
getConversationProfile	Gets the conversation information.
deleteConversation	Deletes a conversation.
setConversationDraft	Set conversation draft.
clearHistoryMessage	Clears the chat history with a user from local storage and the cloud (without deleting the conversation).
pinConversation	Pins/Unpins a conversation to/from top.
setAllMessageRead	Marks the unread messages of all conversations as read.

setMessageRemindType	Sets the conversation message notification type. You can use this API to mute notifications or reject messages.
getTotalUnreadMessageCount	Gets the total unread message count of conversations.

Conversation group

API	Description
setConversationCustomData	Sets custom conversation data.
markConversation	Marks a conversation.
getConversationGroupList	Gets the list of conversation groups.
createConversationGroup	Creates a conversation group.
deleteConversationGroup	Deletes a conversation group.
renameConversationGroup	Renames a conversation group.
addConversationsToGroup	Adds a conversation to a conversation group.
deleteConversationsFromGroup	Deletes conversations from a conversation group.

Search

API	Description
searchCloudMessages	Search cloud messages.
searchCloudUsers	Search cloud users.
searchCloudGroups	Search cloud groups.
searchCloudGroupMembers	Search cloud group members.

Profile

API	Description
getMyProfile	Gets personal profile.
getUserProfile	Gets other user's profile.
updateMyProfile	Updates your personal profile.

getBlacklist	Gets your blacklist.
addToBlacklist	Adds a user to the blacklist.
removeFromBlacklist	Removes a user from the blacklist.

User status

API	Description
setSelfStatus	Sets one's own custom status.
getUserStatus	Queries a user's status.
subscribeUserStatus	Subscribes to a user's status.
unsubscribeUserStatus	Unsubscribes from a user's status.

Relationship Chain

API	Description
getFriendList	Gets the contacts in the SDK cache.
addFriend	Adds friends.
deleteFriend	Deletes friends.
checkFriend	Verifies friend relationship.
getFriendProfile	Gets the friend data and profile data of a specified user.
updateFriend	Updates the contacts data of friends.
getFriendApplicationList	Gets the friend request list in the SDK cache.
acceptFriendApplication	Accepts a friend request.
refuseFriendApplication	Rejects a friend request.
deleteFriendApplication	Deletes a friend request.
setFriendApplicationRead	Set a friend request as reads.
getFriendGroupList	Gets list of friend groups in the SDK cache.
createFriendGroup	Creates a friend group.

deleteFriendGroup	Deletes a friend group.
addToFriendGroup	Adds friends to a friend group.
removeFromFriendGroup	Removes friends from a friend group.
renameFriendGroup	Modifies the name of a friend group.

Group

API	Description
getGroupList	Gets the group list.
getGroupProfile	Gets the group profile.
createGroup	Creates a group.
dismissGroup	Disbands a group.
updateGroupProfile	Modifies the group profile.
joinGroup	Requests to join a group.
quitGroup	Quits a group.
searchGroupById	Searches for a group.
getGroupOnlineMemberCount	Gets the number of online users in an audio-video group.
changeGroupOwner	Transfers the group ownership.
getGroupApplicationList	Gets the list of requests to join a group.
handleGroupApplication	Processes requests to join the group.
initGroupAttributes	Initializes group attributes.
setGroupAttributes	Sets group attributes.
deleteGroupAttributes	Deletes group attributes.
getGroupAttributes	Gets group attributes.
setGroupCounters	Sets group counters.
increaseGroupCounter	Increases the specified group counters.
decreaseGroupCounter	Decreases the specified group counters.

[getGroupCounters](#)

Gets the specified group counters.

Group member

API	Description
getGroupMemberList	Gets the group member list.
getGroupMemberProfile	Gets a group member's profile.
addGroupMember	Adds a group member.
deleteGroupMember	Deletes a group member.
setGroupMemberMuteTime	Configures the muting period.
setGroupMemberRole	Modifies the group member's role.
setGroupMemberNameCard	Sets the name card for a group member.
setGroupMemberCustomField	Sets a custom field for a group member.
markGroupMemberList	Marks group members.

Topic

API	Description
getJoinedCommunityList	Gets the list of community groups the current user has joined.
createTopicInCommunity	Creates a topic in a community group.
deleteTopicFromCommunity	Deletes a topic in a community group.
updateTopicProfile	Updates a topic profile in a community group.
getTopicList	Gets the topic list in a community group.

Signaling

API	Description
addSignalingListener	Listens for a signaling event.
removeSignalingListener	Removes listener for a signaling event.
invite	Invites someone.

<code>inviteInGroup</code>	Invites members in a group.
<code>cancel</code>	The inviter cancels the invitation.
<code>accept</code>	The invitee accepts the invitation.
<code>reject</code>	The invitee rejects the invitation.
<code>getSignalingInfo</code>	Gets signaling info from a message.
<code>modifyInvitation</code>	Modifies a invitation.

C APIs

Last updated : 2025-02-07 11:51:44

SDK Initialization APIs

To use the Tencent Cloud Chat service, you need to initialize the SDK and log in.

API	Description
<code>TIMInit</code>	Initializes the IM SDK.
<code>TIMUninit</code>	Uninstalls the IM SDK.
<code>TIMGetSDKVersion</code>	Gets IM SDK version number.
<code>TIMGetServerTime</code>	Gets the current server time.
<code>TIMSetConfig</code>	Sets extra user configurations.

Login and Logout APIs

API	Description
<code>TIMLogin</code>	Logs in.
<code>TIMLogout</code>	Logs out.
<code>TIMGetLoginUserID</code>	Gets the <code>userID</code> of the logged-in user.
<code>TIMGetLoginStatus</code>	Gets the login status.

Message APIs

API	Description
<code>TIMMsgSendMessage</code>	Sends a new message
<code>TIMMsgCancelSend</code>	Cancels a message based on the <code>messageID</code> .
<code>TIMMsgBatchSend</code>	Sends a message to multiple users at a time, but not

	to groups.
<code>TIMMsgDownloadElemToPath</code>	Downloads message elements (such as the image, video, audio, and file) to a path.
<code>TIMMsgDownloadMergerMessage</code>	Downloads a merged message.
<code>TIMMsgSetLocalCustomData</code>	Sets custom message data (saved locally, not sent to the peer, and invalidated upon app reinstallation).
<code>TIMMsgSetC2CReceiveMessageOpt</code>	Sets the one-to-one message receiving option for a user (batch setting supported).
<code>TIMMsgGetC2CReceiveMessageOpt</code>	Queries the one-to-one message receiving option for a user.
<code>TIMMsgSetGroupReceiveMessageOpt</code>	Sets the receiving option for group messages.
<code>TIMMsgSetAllReceiveMessageOpt</code>	Sets the global message receiving option for a logged-in user (supports setting do-not-disturb times for each day).
<code>TIMMsgSetAllReceiveMessageOpt2</code>	Sets the global message receiving option for a logged-in user.
<code>TIMMsgGetAllReceiveMessageOpt</code>	Gets the global message receiving option of a logged-in user.
<code>TIMMsgGetMsgList</code>	Gets the message list of a conversation.
<code>TIMMsgRevoke</code>	Recalls a message.
<code>TIMMsgModifyMessage</code>	Modifies a message.
<code>TIMMsgDelete</code>	Deletes the messages of a conversation.
<code>TIMMsgListDelete</code>	Deletes the local and roaming message lists of a conversation.
<code>TIMMsgClearHistoryMessage</code>	Clears the messages of a conversation.
<code>TIMMsgSaveMsg</code>	Saves a custom message.
<code>TIMMsgImportMsgList</code>	Imports a message list to a conversation.
<code>TIMMsgFindMessages</code>	Queries the list of local messages based on the <code>messageID</code> .

<code>TIMMsgFindByMsgLocatorList</code>	Locates a message of a conversation by using the message locator.
<code>TIMMsgSearchLocalMessages</code>	Searches for local messages.
<code>TIMMsgSearchCloudMessages</code>	Searches for cloud messages.
<code>TIMMsgSendMessageReadReceipts</code>	Sends message read receipts.
<code>TIMMsgGetMessageReadReceipts</code>	Gets message read receipts.
<code>TIMMsgGetGroupMessageReadMemberList</code>	Gets the list of group members who have read group messages.
<code>TIMMsgSetMessageExtensions</code>	Sets message extensions.
<code>TIMMsgGetMessageExtensions</code>	Gets message extensions.
<code>TIMMsgDeleteMessageExtensions</code>	Deletes message extensions.
<code>TIMMsgAddMessageReaction</code>	Adds message reaction.
<code>TIMMsgRemoveMessageReaction</code>	Deletes message reaction.
<code>TIMMsgGetMessageReactions</code>	Gets message reactions.
<code>TIMMsgGetAllUserListOfMessageReaction</code>	Gets the users who respond with the specified message by page.
<code>TIMMsgTranslateText</code>	Translates a text message.
<code>TIMPinGroupMessage</code>	Sets group message pinning.
<code>TIMGetPinnedMessageList</code>	Gets the pinned group message list.

Conversation APIs

The conversation list is the list a user sees on the first screen after login. It includes elements such as the conversation node, conversation name, group name, last message, and unread count.

API	Description
<code>TIMConvGetConvList</code>	Gets the conversation list.
<code>TIMConvGetConvInfo</code>	Queries a list of conversations.

<code>TIMConvGetConversationListByFilter</code>	An advanced API for getting the conversation list.
<code>TIMConvDelete</code>	Deletes a conversation.
<code>TIMConvDeleteConversationList</code>	Deletes the conversation list.
<code>TIMConvSetDraft</code>	Sets the draft for a conversation.
<code>TIMConvCancelDraft</code>	Deletes the draft for a conversation.
<code>TIMConvSetConversationCustomData</code>	Sets custom conversation data.
<code>TIMConvPinConversation</code>	Pins a conversation to the top.
<code>TIMConvMarkConversation</code>	Marks a conversation.
<code>TIMConvGetTotalUnreadMessageCount</code>	Gets the total unread message count of all conversations.
<code>TIMConvGetUnreadMessageCountByFilter</code>	Gets the total unread message count by filter.
<code>TIMConvSubscribeUnreadMessageCountByFilter</code>	Subscribes to notification of total unread message count for conversations specified by filter.
<code>TIMConvUnsubscribeUnreadMessageCountByFilter</code>	Unsubscribes from notification of total unread message count for conversations specified by filter.
<code>TIMConvCleanConversationUnreadMessageCount</code>	Cleans up the unread message count of the conversation.
<code>TIMConvCreateConversationGroup</code>	Creates a conversation group.
<code>TIMConvGetConversationGroupList</code>	Gets the list of conversation groups.
<code>TIMConvDeleteConversationGroup</code>	Deletes a conversation group.
<code>TIMConvRenameConversationGroup</code>	Renames a conversation group.
<code>TIMConvAddConversationsToGroup</code>	Adds a conversation to a conversation group.
<code>TIMConvDeleteConversationsFromGroup</code>	Deletes conversations from a group.

Group APIs

Tencent Cloud Chat SDK supports five preset group types, each of which pertains to different scenarios.

Work group (Work): users can join the group only after being invited by group members. This group type is the same as private group (Private) in earlier versions.

Public group (Public): Users can join a public group through requests, which need to be approved by the group owner or group admin.

Meeting group (Meeting): used together with [TRTC](#) to enable scenarios such as video conferencing and online education. Users can join and leave the group freely and view the message history before they join. Same as chat room (ChatRoom) in earlier versions.

Community: A user can join and leave a community freely. It is suitable for chat scenarios with a super large number of community members, such as knowledge sharing and game discussion. This feature is supported by a client with the SDK enhanced edition v5.8.1668 or later and the web SDK v2.17.0 or later. To use it, you need to purchase the [Pro edition](#)、[Pro Plus edition](#) or [Enterprise edition](#), and then enable it in [Console](#), path: Applications > Your App > Chat > Configuration > Group Configuration > Community.

Audio-video group (AVChatRoom): An audio-video group allows users to join and leave freely and is suitable for scenarios such as live streaming and chat rooms with on-screen comments. There is no limit on the number of group members.

API	Description
<code>TIMGroupCreate</code>	Creates a group.
<code>TIMGroupDelete</code>	Deletes or dissolves a group.
<code>TIMGroupJoin</code>	Requests to join a group.
<code>TIMGroupQuit</code>	Quits a group.
<code>TIMGroupGetJoinedGroupList</code>	Gets the list of groups that a user has joined.
<code>TIMGroupGetGroupInfoList</code>	Gets group information.
<code>TIMGroupSearchGroups</code>	Searches for local groups.
<code>TIMGroupSearchCloudGroups</code>	Searches for cloud groups.
<code>TIMGroupModifyGroupInfo</code>	Modifies group information.
<code>TIMGroupInitGroupAttributes</code>	Initializes the group attributes to clear the existing group attribute list.
<code>TIMGroupSetGroupAttributes</code>	Sets group attributes. If a group attribute already exists, its value will be updated; otherwise, it will be added.

<code>TIMGroupDeleteGroupAttributes</code>	Deletes group attributes.
<code>TIMGroupGetGroupAttributes</code>	Gets a specified group attribute. If `NULL` is passed in for `json_keys`, all the group attributes will be obtained.
<code>TIMGroupGetOnlineMemberCount</code>	Gets the number of online members in a group.
<code>TIMGroupSetGroupCounters</code>	Sets group counters.
<code>TIMGroupGetGroupCounters</code>	Gets group counters.
<code>TIMGroupIncreaseGroupCounter</code>	Increases the group counter.
<code>TIMGroupDecreaseGroupCounter</code>	Decreases the group counter.
<code>TIMGroupGetJoinedCommunityList</code>	Gets the list of communities the current user has joined.
<code>TIMGroupCreateTopicInCommunity</code>	Creates a topic.
<code>TIMGroupDeleteTopicFromCommunity</code>	Deletes a topic.
<code>TIMGroupSetTopicInfo</code>	Modifies topic information.
<code>TIMGroupGetTopicInfoList</code>	Gets the list of topics.
<code>TIMGroupGetMemberInfoList</code>	Gets group member information.
<code>TIMGroupSearchGroupMembers</code>	Searches for local group members.
<code>TIMGroupSearchCloudGroupMembers</code>	Searches for cloud group members.
<code>TIMGroupModifyMemberInfo</code>	Modifies group member information.
<code>TIMGroupInviteMember</code>	Invites someone to join a group.
<code>TIMGroupDeleteMember</code>	Removes a member from a group.
<code>TIMGroupMarkGroupMemberList</code>	Marks group members.
<code>TIMGroupGetPendencyList</code>	Gets the pending group request list.(Pending group requests are those that are not handled, for example, invitations or requests to join a group.)
<code>TIMGroupHandlePendency</code>	Handles pending group requests.
<code>TIMGroupReportPendencyReaded</code>	Reports that a pending group request is read.

Community Topic APIs

API	Description
<code>TIMCommunityCreate</code>	Creates a community that supports a topic.
<code>TIMCommunityGetJoinedCommunityList</code>	Gets the list of communities the current user has joined.
<code>TIMCommunityCreateTopicInCommunity</code>	Creates a topic.
<code>TIMCommunityDeleteTopicFromCommunity</code>	Deletes a topic.
<code>TIMCommunitySetTopicInfo</code>	Modifies topic information.
<code>TIMCommunityGetTopicInfoList</code>	Gets the list of topics.
<code>TIMCommunityCreatePermissionGroupInCommunity</code>	Creates a community permission group.
<code>TIMCommunityDeletePermissionGroupFromCommunity</code>	Deletes a community permission group.
<code>TIMCommunityModifyPermissionGroupInfoInCommunity</code>	Modifies a community permission group.
<code>TIMCommunityGetJoinedPermissionGroupListInCommunity</code>	Gets the list of community permission groups a user has joined.
<code>TIMCommunityGetPermissionGroupListInCommunity</code>	Gets a community permission group list.
<code>TIMCommunityAddCommunityMembersToPermissionGroup</code>	Adds members to a community permission group.
<code>TIMCommunityRemoveCommunityMembersFromPermissionGroup</code>	Removes members from a community permission group.
<code>TIMCommunityGetCommunityMemberListInPermissionGroup</code>	Gets the list of community permission group members.
<code>TIMCommunityAddTopicPermissionToPermissionGroup</code>	Adds topic permission to the permission group.
<code>TIMCommunityDeleteTopicPermissionFromPermissionGroup</code>	Deletes topic permission from

	the permission group.
<code>TIMCommunityModifyTopicPermissionInPermissionGroup</code>	Modifies the topic permission in the permission group.
<code>TIMCommunityGetTopicPermissionInPermissionGroup</code>	Gets the topic permission in the permission group.

User Profile APIs

API	Description
<code>TIMProfileGetUserProfileList</code>	Gets users' profiles.
<code>TIMProfileModifySelfUserProfile</code>	Modifies one's own profile.
<code>TIMSubscribeUserInfo</code>	Subscribes to a stranger's profile.
<code>TIMUnsubscribeUserInfo</code>	Unsubscribes from a stranger's profile.
<code>TIMSearchUsers</code>	Search for cloud users

User Status APIs

API	Description
<code>TIMGetUserStatus</code>	Gets user status information.
<code>TIMSetSelfStatus</code>	Sets one's own status.
<code>TIMSubscribeUserStatus</code>	Subscribes to a user's status.
<code>TIMUnsubscribeUserStatus</code>	Unsubscribes from a user's status.

Friend Management APIs

By default, Tencent Cloud Chat does not check your relationship with a user when receiving and sending messages. You can enable **Check Relationship for One-to-One Messages** on **Feature Configuration > Login and Message > Relationship Check** in the [Chat console](#) and use the following APIs to add/delete friends and manage your friends.

API	Description
<code>TIMFriendshipGetFriendProfileList</code>	Gets the contacts.
<code>TIMFriendshipGetFriendsInfo</code>	Gets information of friends.
<code>TIMFriendshipModifyFriendProfile</code>	Updates the profile of a friend (such as remarks).
<code>TIMFriendshipSearchFriends</code>	Searches for a friend.
<code>TIMFriendshipAddFriend</code>	Adds a friend.
<code>TIMFriendshipDeleteFriend</code>	Deletes a friend.
<code>TIMFriendshipCheckFriendType</code>	Checks the friend type (one-way or two-way).
<code>TIMFriendshipGetPendencyList</code>	Gets the pending friend request list.
<code>TIMFriendshipHandleFriendAddRequest</code>	Handles friend requests.
<code>TIMFriendshipReportPendencyReaded</code>	Reports that a pending friend request is read.
<code>TIMFriendshipDeletePendency</code>	Deletes a pending friend request.
<code>TIMFriendshipGetBlackList</code>	Gets the blacklist.
<code>TIMFriendshipAddToBlackList</code>	Adds a user to the blacklist.
<code>TIMFriendshipDeleteFromBlackList</code>	Removes a user from the blacklist.
<code>TIMFriendshipCreateFriendGroup</code>	Creates a friend group.
<code>TIMFriendshipGetFriendGroupList</code>	Gets the information of a friend group.
<code>TIMFriendshipDeleteFriendGroup</code>	Deletes a friend group.
<code>TIMFriendshipModifyFriendGroup</code>	Modifies a friend group.

Official Account APIs

Official accounts can send broadcast messages and one-to-one (C2C) messages to subscribers.

API	Description
<code>TIMSubscribeOfficialAccount</code>	Subscribes to an official account.

<code>TIMUnsubscribeOfficialAccount</code>	Unsubscribes from an official account.
<code>TIMGetOfficialAccountsInfo</code>	Gets a list of official accounts.

Follow Feature APIs

The follow feature can help establish and maintain simple connections between users, facilitating interaction and communication among them.

API	Description
<code>TIMFollowUser</code>	Follows users.
<code>TIMUnfollowUser</code>	Unfollows users.
<code>TIMGetMyFollowingList</code>	Gets my following user list.
<code>TIMGetMyFollowersList</code>	Gets my follower list.
<code>TIMGetMutualFollowersList</code>	Gets my mutual following user list.
<code>TIMGetUserFollowInfo</code>	Gets the information of followings/followers/mutual followings of a specified user.
<code>TIMCheckFollowType</code>	Checks the follow type of a specified user.

Offline Push APIs

API	Description
<code>TIMMsgSetOfflinePushToken</code>	Sets offline push configuration information (for iOS and Android platforms only).
<code>TIMMsgDoBackground</code>	Specifies the initial unread count on the desktop badge (for iOS and Android platforms only). This API can be called when the app detects that the app switches to the background.
<code>TIMMsgDoForeground</code>	Called when the app detects that the app switches to the foreground (for iOS and Android platforms only).

Signaling APIs

API	Description
<code>TIMSignalingInvite</code>	Invites a user.
<code>TIMSignalingInviteInGroup</code>	Invites certain users in the group.
<code>TIMSignalingCancel</code>	Cancels an invitation.
<code>TIMSignalingAccept</code>	Accepts an invitation.
<code>TIMSignalingReject</code>	Rejects an invitation.
<code>TIMGetSignalingInfo</code>	Gets the signaling information.
<code>TIMSignalingModifyInvitation</code>	Modifies the invitation signaling.

Event Callback APIs

IM SDK initialization and login callback APIs

API	Description
<code>TIMSetNetworkStatusListenerCallback</code>	Sets the callback for network connection status.
<code>TIMSetKickedOfflineCallback</code>	Sets the callback for forced logout.
<code>TIMSetUserSigExpiredCallback</code>	Sets the callback for user ticket expiration.
<code>TIMSetLogCallback</code>	Sets the callback for logs.

Message callback APIs

API	Description
<code>TIMAddRecvNewMsgCallback</code>	Adds the callback for receiving new messages.
<code>TIMRemoveRecvNewMsgCallback</code>	Deletes the callback for receiving new messages.
<code>TIMSetMsgElemUploadProgressCallback</code>	Sets the callback for the upload progress of message element files.
<code>TIMSetMsgReadedReceiptCallback</code>	Sets the callback for message read receipts.

<code>TIMSetMsgRevokeCallback</code>	Sets the callback for message recalls.
<code>TIMSetMsgUpdateCallback</code>	Sets the callback for message update notifications returned after messages are modified in the cloud.
<code>TIMSetMsgExtensionsChangedCallback</code>	Sets the callback for message extension changes.
<code>TIMSetMsgExtensionsDeletedCallback</code>	Sets the callback for message extension deletion.
<code>TIMSetMsgReactionsChangedCallback</code>	Sets the callback for message reaction changes.
<code>TIMSetMsgAllMessageReceiveOptionCallback</code>	Sets the callback for the global message receiving option.

Conversation callback APIs

API	Description
<code>TIMSetConvEventCallback</code>	Sets the callback for conversation events.
<code>TIMSetConvTotalUnreadMessageCountChangedCallback</code>	Sets the callback for changes in the unread message count of a conversation.
<code>TIMSetConvUnreadMessageCountChangedByFilterCallback</code>	Sets the callback for changes in the unread message count of filtered conversations.
<code>TIMSetConvConversationGroupCreatedCallback</code>	Sets the callback for conversation group creation.
<code>TIMSetConvConversationGroupDeletedCallback</code>	Sets the callback for conversation group deletion.
<code>TIMSetConvConversationGroupNameChangedCallback</code>	Sets the callback for a conversation group name change.
<code>TIMSetConvConversationsAddedToGroupCallback</code>	Sets the callback for adding a conversation to a conversation group.
<code>TIMSetConvConversationsDeletedFromGroupCallback</code>	Sets the callback for deleting a conversation from a conversation group.

Group callback APIs

API	Description
<code>TIMSetGroupTipsEventCallback</code>	Sets the callback for group system messages.
<code>TIMSetGroupAttributeChangedCallback</code>	Sets the callback for a group attribute change.
<code>TIMSetGroupCounterChangedCallback</code>	Sets the callback for a group counter change.
<code>TIMSetGroupTopicCreatedCallback</code>	Sets the callback for group topic creation.
<code>TIMSetGroupTopicDeletedCallback</code>	Sets the callback for group topic deletion.
<code>TIMSetGroupTopicChangedCallback</code>	Sets the callback for group topic changes.

Community topic callback APIs

API	Description
<code>TIMSetCommunityCreateTopicCallback</code>	Sets the callback for community topic creation.
<code>TIMSetCommunityDeleteTopicCallback</code>	Sets the callback for community topic deletion.
<code>TIMSetCommunityChangeTopicInfoCallback</code>	Sets the callback for community topic changes.
<code>TIMSetCommunityReceiveTopicRESTCustomDataCallback</code>	Sets the callback for the topic custom system messages pushed by REST API.
<code>TIMSetCommunityCreatePermissionGroupCallback</code>	Sets the callback for permission group creation.
<code>TIMSetCommunityDeletePermissionGroupCallback</code>	Sets the callback for permission group deletion.
<code>TIMSetCommunityChangePermissionGroupInfoCallback</code>	Sets the callback for permission group information changes.
<code>TIMSetCommunityAddMembersToPermissionGroupCallback</code>	Sets the callback for adding a member to the permission group.

<code>TIMSetCommunityRemoveMembersFromPermissionGroupCallback</code>	Sets the callback for removing a member from the permission group.
<code>TIMSetCommunityAddTopicPermissionCallback</code>	Sets the callback for adding a topic permission to the permission group.
<code>TIMSetCommunityDeleteTopicPermissionCallback</code>	Sets the callback for deleting a topic permission from the permission group.
<code>TIMSetCommunityModifyTopicPermissionCallback</code>	Sets the callback for modifying a topic permission in the permission group.

User profile and status callback APIs

API	Description
<code>TIMSetSelfInfoUpdatedCallback</code>	Sets the callback for one's own profile changes.
<code>TIMSetUserStatusChangedCallback</code>	Sets the callback for user status changes.
<code>TIMSetUserInfoChangedCallback</code>	Sets the callback for subscriber profile changes.

Friend callback APIs

API	Description
<code>TIMSetOnAddFriendCallback</code>	Sets the callback for adding friends.
<code>TIMSetOnDeleteFriendCallback</code>	Sets the callback for deleting friends.
<code>TIMSetUpdateFriendProfileCallback</code>	Sets the callback for updating friends' profiles.
<code>TIMSetFriendAddRequestCallback</code>	Sets the callback for friend requests.
<code>TIMSetFriendApplicationListDeletedCallback</code>	Sets the callback for deleting a friend request.
<code>TIMSetFriendApplicationListReadCallback</code>	Sets the callback for reading a friend request.
<code>TIMSetFriendBlackListAddedCallback</code>	Sets the callback for adding a friend to the blocklist.
<code>TIMSetFriendBlackListDeletedCallback</code>	Sets the callback for removing a friend from the

	blocklist.
<code>TIMSetFriendGroupCreatedCallback</code>	Sets the callback for friend group creation.
<code>TIMSetFriendGroupDeletedCallback</code>	Sets the callback for friend group deletion.
<code>TIMSetFriendGroupNameChangedCallback</code>	Sets the callback for a friend group name change.
<code>TIMSetFriendsAddedToGroupCallback</code>	Sets the callback for adding a friend to a friend group.
<code>TIMSetFriendsDeletedFromGroupCallback</code>	Sets the callback for deleting a friend from a friend group.

Official account callback APIs

API	Description
<code>TIMSetOfficialAccountSubscribedCallback</code>	Sets the callback for subscribing to an official account.
<code>TIMSetOfficialAccountUnsubscribedCallback</code>	Sets the callback for unsubscribing from an official account.
<code>TIMSetOfficialAccountDeletedCallback</code>	Sets the callback for deleting an official account.
<code>TIMSetOfficialAccountInfoChangedCallback</code>	Sets the callback for subscribed official account information changes.

Follow feature callback APIs

API	Description
<code>TIMSetMyFollowingListChangedCallback</code>	Sets the callback for my following list changes.
<code>TIMSetMyFollowersListChangedCallback</code>	Sets the callback for my followers list changes.
<code>TIMSetMutualFollowersListChangedCallback</code>	Sets the callback for mutual following list changes.

Signaling callback APIs

API	Description
<code>TIMSetSignalingReceiveNewInvitationCallback</code>	Sets the callback for receiving a new invitation.

<code>TIMSetSignalingInvitationCancelledCallback</code>	Sets the callback for canceling an invitation.
<code>TIMSetSignalingInviteeAcceptedCallback</code>	Sets the callback for accepting an invitation by invitees.
<code>TIMSetSignalingInviteeRejectedCallback</code>	Sets the callback for rejecting an invitation by invitees.
<code>TIMSetSignalingInvitationTimeoutCallback</code>	Sets the callback for an invitation timeout.
<code>TIMSetSignalingInvitationModifiedCallback</code>	Sets the callback for modifying an invitation.

C++

Last updated : 2025-01-23 16:13:12

Initialization and Login APIs

To use the Tencent Cloud Chat service, you need to initialize the SDK and log in.

API	Description
InitSDK	Initializes the IM SDK.
UnInitSDK	Uninstalls the IM SDK.
GetVersion	Gets the version number.
GetServerTime	Gets the server time.
Login	Logs a user in.
Logout	Logs a user out.
GetLoginStatus	Gets the login status.
GetLoginUser	Gets the UserID of the current login user.

Simple Message APIs

Use the following APIs for the sending and receiving of text and signaling (custom buffer) messages.

API	Description
AddSimpleMsgListener	Sets an event listener for simple messages (text messages and custom messages). Do not use it and AddAdvancedMsgListener at the same time.
RemoveSimpleMsgListener	Removes the event listener for simple messages (text messages and custom messages).
SendC2CTextMessage	Sends a one-to-one (C2C) text message.
SendC2CCustomMessage	Sends a one-to-one (C2C) custom (signaling) message.
SendGroupTextMessage	Sends a group text message.

[SendGroupCustomMessage](#)

Sends a group custom (signaling) message.

Signaling APIs

API	Description
AddSignalingListener	Adds a signaling listener.
RemoveSignalingListener	Removes a signaling listener.
Invite	Invites a user.
InviteInGroup	Invites certain users in the group.
Cancel	The inviter cancels the invitation.
Accept	The invitee accepts the invitation.
Reject	The invitee rejects the invitation.
GetSignalingInfo	Gets the signaling information.
AddInvitedSignaling	Adds invitation signaling (can be used for invitation signaling triggered by offline push messages for group invitations).
ModifyInvitation	Modifies the invitation signaling.

Advanced Message APIs

If you need to send/receive rich media messages (such as image, video, and file messages) and use advanced features such as recalling messages, marking messages as read, and querying message history, use the following set of advanced message APIs. Do not use simple message APIs and advanced message APIs at the same time.

API	Description
AddAdvancedMsgListener	Sets an event listener for advanced messages. Do not use it and AddSimpleMsgListener at the same time.
RemoveAdvancedMsgListener	Removes an event listener for advanced messages.
CreateTextMessage	Creates a text message.
CreateTextAtMessage	Creates an @ text message.

CreateCustomMessage	Creates a custom message.
CreateCustomMessage	Creates a custom message (supports setting offline push information).
CreateImageMessage	Creates an image message.
CreateSoundMessage	Creates an audio message.
CreateVideoMessage	Creates a video message.
CreateFileMessage	Creates a file message.
CreateLocationMessage	Creates a location message.
CreateFaceMessage	Creates an emoji message.
CreateMergerMessage	Creates a combined forward message.
CreateForwardMessage	Creates a single forward message.
CreateTargetedGroupMessage	Creates a targeted group message.
CreateAtSignedGroupMessage	Creates an @ group message.
SendMessage	Sends a message. The message object can be created using a <code>CreateXXXMessage</code> API.
SetC2CReceiveMessageOpt	Sets the Mute Notifications option for one-to-one messages.
GetC2CReceiveMessageOpt	Gets the Mute Notifications status for one-to-one messages.
SetGroupReceiveMessageOpt	Sets the Mute Notifications option for group messages.
SetAllReceiveMessageOpt	Sets the global message receiving option (supports setting do-not-disturb times for each day).
SetAllReceiveMessageOpt	Sets the global message receiving option.
GetAllReceiveMessageOpt	Gets the global message receiving option of a logged-in user.
GetHistoryMessageList	Gets message history.
RevokeMessage	Recalls a message. The message object can be created using a <code>createXXXMessage</code> API.
ModifyMessage	Modifies a message. The message object can be created using a <code>createXXXMessage</code> API.

MarkC2CMessageAsRead	Marks one-to-one (C2C) messages as read. (It is deprecated. Please call API CleanConversationUnreadMessageCount instead).
MarkGroupMessageAsRead	Marks group messages as read. (It is deprecated. Please call API CleanConversationUnreadMessageCount instead).
MarkAllMessageAsRead	Marks all messages as read. (It is deprecated. Please call API CleanConversationUnreadMessageCount instead).
DeleteMessages	Deletes messages from local storage and the cloud.
ClearC2CHistoryMessage	Clears chat history with a user from local storage and the cloud.
ClearGroupHistoryMessage	Clears chat history of a group from local storage and the cloud.
InsertGroupMessageToLocalStorage	Inserts a message in a group chat.
InsertC2CMessageToLocalStorage	Inserts a message in a one-to-one chat.
FindMessages	Finds local messages by <code>msgID</code> .
SearchLocalMessages	Searches for local messages.
SearchCloudMessages	Searches for cloud messages.
SendMessageReadReceipts	Sends message read receipts.
GetMessageReadReceipts	Gets message read receipts.
GetGroupMessageReadMemberList	Gets the list of group members who have read group messages.
setMessageExtensions	Sets message extensions.
getMessageExtensions	Gets message extensions.
deleteMessageExtensions	Deletes message extensions.
AddMessageReaction	Adds message reaction.
RemoveMessageReaction	Deletes message reaction.
GetMessageReactions	Gets message reactions.
GetAllUserListOfMessageReaction	Gets all users of message reactions by page.
pinGroupMessage	Sets group message pinning.
getPinnedGroupMessageList	Gets pinned group message list.

Group APIs

Tencent Cloud Chat SDK supports the following preset group types, each of which pertains to different scenarios.

Work group (Work): Users can join the group only after being invited by group members. This group is similar to ordinary WeChat groups.

Public group (Public): Similar to QQ groups, users can join a public group through requests, which need to be approved by the group owner or group admin.

Meeting group (Meeting): Used together with [TRTC](#) to enable scenarios such as video conferencing and online education. Users can join and leave the group freely and view the message history before they join.

Community: Community supports up to 100,000 members and a user can join and leave a community freely without approval. It is suitable for chat scenarios with a super large number of community members, such as knowledge sharing and game discussion. This feature is supported by a client with the SDK edition V5.8 or later. To use it, you need to purchase the [Pro edition](#)、[Pro Plus edition](#)、[Enterprise edition](#), and then enable it via [console](#)> **Feature Configuration** > **Group Configuration** > **Group Feature Configuration** > **Community**.

Audio-video group (AVChatRoom): An audio-video group allows users to join and leave freely and is suitable for scenarios such as live streaming and chat rooms with on-screen comments. There is no limit on the number of group members.

API	Description
AddGroupListener	Adds an event listener for groups.
RemoveGroupListener	Removes an event listener for groups.
CreateGroup	Creates a simple group.
CreateGroup	Creates an advanced group. The group information and the initial group members can be set during group creation.
JoinGroup	Joins a group.
QuitGroup	Leaves a group.
DismissGroup	Disbands a group. Only the group owner and group admin can disband a group.
GetJoinedGroupList	Gets the list of groups the current user has joined, excluding audio-video groups.
GetGroupsInfo	Pulls the profiles of groups.
SearchGroups	Search for local group profiles.
SearchCloudGroups	Search for cloud group profiles.

SetGroupInfo	Modifies the profile of a group.
InitGroupAttributes	Initializes group attributes.
SetGroupAttributes	Sets group attributes.
DeleteGroupAttributes	Deletes group attributes.
GetGroupAttributes	Gets group attributes.
GetGroupOnlineMemberCount	Gets the number of online group members.
GetGroupMemberList	Gets the group member list.
GetGroupMembersInfo	Gets the profiles of specified group members.
SearchGroupMembers	Search for local group members.
SearchCloudGroupMembers	Search for cloud group members.
SetGroupMemberInfo	Modifies the profile of a specified group member.
MuteGroupMember	Mutes a group member.
MuteAllGroupMembers	Mutes all group members. It can be called only by the administrator or group owner.
KickGroupMember	Removes a member from a group.
KickGroupMember	Removes a member from a group (support setting a duration for prohibiting group joining).
SetGroupMemberRole	Sets a role for a group member.
markGroupMemberList	Marks group members.
TransferGroupOwner	Transfers the group ownership.
InviteUserToGroup	Invites users to a group.
GetGroupApplicationList	Gets the list of requests to join a group.
AcceptGroupApplication	Approves a request to join a group.
RefuseGroupApplication	Rejects a request to join a group.
SetGroupApplicationRead	Marks the request list as read.
GetJoinedCommunityList	Gets the list of communities the current user has joined.

CreateTopicInCommunity	Creates a topic.
DeleteTopicFromCommunity	Deletes a topic.
SetTopicInfo	Modifies topic information.
GetTopicInfoList	Gets the list of topics.
SetGroupCounters	Sets group counters.
GetGroupCounters	Gets group counters.
IncreaseGroupCounter	Increases the group counter.
DecreaseGroupCounter	Decreases the group counter.

Community Topic APIs

Community is used to manage group members. All topics under a community can not only share community members, but also send and receive messages independently without interfering with each other.

API	Description
AddCommunityListener	Adds an event listener for communities.
RemoveCommunityListener	Removes an event listener for communities.
CreateCommunity	Creates a community that supports a topic.
GetJoinedCommunityList	Gets the list of communities the current user has joined.
CreateTopicInCommunity	Creates a topic.
DeleteTopicFromCommunity	Deletes a topic.
SetTopicInfo	Modifies topic information.
GetTopicInfoList	Gets the list of topics.
CreatePermissionGroupInCommunity	Creates a community permission group.
DeletePermissionGroupFromCommunity	Deletes a community permission group.
ModifyPermissionGroupInfoInCommunity	Modifies a community permission group.
GetJoinedPermissionGroupListInCommunity	Gets the list of community permission groups a user has joined.

GetPermissionGroupListInCommunity	Gets a community permission group list.
AddCommunityMembersToPermissionGroup	Adds members to a community permission group.
RemoveCommunityMembersFromPermissionGroup	Removes members from a community permission group.
GetCommunityMemberListInPermissionGroup	Gets the list of community permission group members.
AddTopicPermissionToPermissionGroup	Adds topic permission to the permission group.
DeleteTopicPermissionFromPermissionGroup	Deletes topic permission from the permission group.
ModifyTopicPermissionInPermissionGroup	Modifies the topic permission in the permission group.
GetTopicPermissionInPermissionGroup	Gets the topic permission in the permission group.

Conversation List APIs

The conversation list is the list a user sees on the first screen after login. It includes elements such as conversation node, conversation name, group name, last message, and unread count.

API	Description
AddConversationListener	Adds a conversation listener.
RemoveConversationListener	Removes a conversation listener.
GetConversationList	Gets the conversation list.
GetConversation	Gets a conversation.
GetConversationList	Gets multiple conversations.
GetConversationListByFilter	Gets multiple conversations (support filtering based on conversation type, mark type, group name, etc.).
getConversationListByFilter	Gets the advanced conversation API to specify the conversation type, mark type, and group name.
DeleteConversation	Deletes a conversation.
SetConversationDraft	Sets a draft for a conversation.
SetConversationCustomData	Sets custom conversation data.
PinConversation	Pins a conversation to the top.

MarkConversation	Marks a conversation.
GetTotalUnreadMessageCount	Gets the total unread message count.
CreateConversationGroup	Creates a conversation group.
GetConversationGroupList	Gets the list of conversation groups.
DeleteConversationGroup	Deletes a conversation group.
RenameConversationGroup	Renames a conversation group.
AddConversationsToGroup	Adds a conversation to a conversation group.
DeleteConversationsFromGroup	Deletes a conversation from a conversation group.
GetUnreadMessageCountByFilter	Gets the total unread message count of conversations by filter.
SubscribeUnreadMessageCountByFilter	Subscribes to notification of total unread message count for conversations specified by filter.
UnsubscribeUnreadMessageCountByFilter	Unsubscribes from notification of total unread message count for conversations specified by filter.

User Profile APIs

You can use the following APIs to query user profiles, modify your profile, and block messages from a specified user (that is, adding a specified user to the blocklist).

API	Description
GetUsersInfo	Gets user profiles.
SetSelfInfo	Modifies one's own user profile.
SubscribeUserInfo	Subscribes to a user's profile.
UnsubscribeUserInfo	Unsubscribes from a user's profile.
GetUserStatus	Queries a user's status.
SetSelfStatus	Sets one's own status.
SubscribeUserStatus	Subscribes to a user's status.
UnsubscribeUserStatus	Unsubscribes from a user's status.

SearchUsers	Search cloud users.
AddToBlackList	Blocks messages from a user, which means adding the user to the blocklist.
DeleteFromBlackList	Unblocks messages from a user, which means removing the user from the blocklist.
GetBlackList	Gets the blocklist.

Friend Management APIs

Tencent Cloud Chat SDK does not check friend relationships by default when sending and receiving messages. To achieve the interactive experience of "add friends first, then send messages", you can log in to the [Console](#) to modify Relationship Check. When enabled, users can only send messages to their friends. When a user sends a message to a non-friend, SDK will report a 20009 error code. The configuration path is: Applications > Your App > Chat > Configuration > Login and Message > Relationship Check.

Use the following API to add or delete friends and manage friend lists.

API	Description
AddFriendListener	Adds a contacts listener.
RemoveFriendListener	Removes a contacts listener.
GetFriendList	Gets the contacts.
GetFriendsInfo	Gets the profiles of specified friends.
SetFriendInfo	Sets the profile of a specified friend.
SearchFriends	Searches for friends.
AddFriend	Adds a friend.
DeleteFromFriendList	Deletes a friend.
CheckFriend	Checks relationship with specified users.
GetFriendApplicationList	Gets the list of friend requests.
AcceptFriendApplication	Accepts a friend request.
RefuseFriendApplication	Rejects a friend request.
DeleteFriendApplication	Deletes a friend request.

SetFriendApplicationRead	Marks a friend request as read.
CreateFriendGroup	Creates a friend list.
GetFriendGroups	Gets the information of friend lists.
DeleteFriendGroup	Deletes friend lists.
RenameFriendGroup	Modifies the name of a friend list.
AddFriendsToFriendGroup	Adds friends to a friend list.
DeleteFriendsFromFriendGroup	Deletes friends from a friend list.

Official Account APIs

Official accounts can send broadcast messages and one-to-one (C2C) messages to subscribers.

API	Description
SubscribeOfficialAccount	Subscribes to an official account.
UnsubscribeOfficialAccount	Unsubscribes from an official account.
GetOfficialAccountsInfo	Gets a list of official accounts.

Follow Feature APIs

The follow feature can help establish and maintain simple connections between users, facilitating interaction and communication among them.

API	Description
FollowUser	Follows users.
UnfollowUser	Unfollows users.
GetMyFollowingList	Gets my following user list.
GetMyFollowersList	Gets my follower list.
GetMutualFollowersList	Gets my mutual following user list.
GetUserFollowInfo	Gets the information of followings/followers/mutual followings of a

	specified user.
CheckFollowType	Checks the follow type of a specified user.