

Ecamm Remote API v4.7

Locating the service

```
searchForServicesOfType:@"_ecammliveremote._tcp."  
inDomain:@"local."
```

If your client is only intended to run on the same Mac as Ecamm, confirm that the discovered service is the local host so you do not accidentally control Ecamm on another Mac.

HTTP requests

All requests are HTTP **GET** requests.

Use URL-encoded query-string parameters.

Send these HTTP headers with every request:

User-Agent: [app name]/[app version]

EcammLive-UUID: A GUID string that should persist between sessions.

EcammLive-ClientName: Your app name, shown to the user in Ecamm.

EcammLive-DeviceName: Optional device name for remote devices.

EcammLive-ClientIcon: Optional URL of your square app icon.

Response shapes

- Most single-value getters return a one-item **JSON** array.
- Most list getters return an object shaped like `{"items":[...]}`.
- Dictionary getters return a **JSON** object keyed by UUID or path.
- Thumbnail getters return an object shaped like `{"icon":"data:image/...;base64,..."}.`
- Setters usually return a one-item **JSON** array such as `["Done"]` or `["Clicked it!"]`.
- If access is denied, setters return `["Access denied"]`. Most getters return `{}`.

- `getConnectionStatus` is the exception: it still returns status even before the client is approved.

Display info and status

`getInfo`

Parameters: none

Returns: a `JSON` object containing the current control state.

Note: avoid polling this more often than once every 2 seconds.

`getButtonLabel`

Parameters: none

Returns: a one-item `JSON` array containing the current Start button label.

`getPauseButtonLabel`

Parameters: none

Returns: a one-item `JSON` array containing the current Pause button label.

`getConnectionStatus`

Parameters: none

Returns: a one-item `JSON` array containing Allowed, Denied, Pending, or Not Found.

`setClickButton`

Parameters:

channel - destination UUID from `getChannels`. To use the selected scheduled live, send `channel=nothing`

title - optional title for a newly created live broadcast

desc - optional description for a newly created live broadcast

Action: clicks the Start or Record button.

`setClickPause`

Parameters: none

Action: clicks the pause recording button.

`setStartNewRecording`

Parameters: none

Action: starts a new recording while Ecamm is already streaming or recording.

setTakePhoto

Parameters:

countdown - optional; **true** waits three seconds before taking the photo

Action: saves a photo of the current Ecamm output.

Scenes

getSceneList

Parameters: none

Returns: {"items":[...]} with scene dictionaries.

getSceneDict

Parameters: none

Returns: a **JSON** object keyed by scene UUID.

getSceneImage

Parameters:

id - scene UUID

Returns: {"icon":"data:image/jpeg;base64,..."} for the scene thumbnail.

getCurrentScene

Parameters: none

Returns: a one-item **JSON** array containing the current scene UUID.

setScene

Parameters:

id - scene UUID

Action: switches to a scene.

setNext

Parameters: none

Action: switches to the next scene.

setPrev

Parameters: none

Action: switches to the previous scene.

setSceneList

Parameters:

action - **up**, **down**, or **pick**

Action: drives the scene list selection UI for control-surface integrations.

Mute and volume

Bus values used by **getMute**, **setMute**, **getVolume**, and **setVolume**:

mic
mic2
skype
systemaudio
interview
zoom
soundeffects
movie
guest_1, guest_2, guest_3, and so on

If bus is omitted for **getMute** or **setMute**, the main mic bus is used.

If bus is omitted for **getVolume** or **setVolume**, the main mic bus is used.

getMute

Parameters:

bus - optional bus name

Returns: a one-item **JSON** array containing **yes**, **no**, or **N/A**.

setMute

Parameters:

bus - optional bus name

Action: toggles mute on the requested bus.

getVolume

Parameters:

bus - optional bus name

Returns: a one-item **JSON** array containing a level from 0 to 100, or N/A.

setVolume

Parameters:

bus - optional bus name

volume - absolute level from 0 to 100

ticks - relative adjustment in hundredths; positive raises, negative lowers

Action: sets or nudges the level for the requested bus.

Note: if ticks is supplied, Ecamm adjusts the current level instead of using volume.

Viewers

getViewers

Parameters: none

Returns: a one-item **JSON** array containing the current concurrent viewer count.

Cameras and source mode

getInputs

Parameters: none

Returns: {"items": [...]} with camera input dictionaries.

getDefaultCamera

Parameters: none

Returns: a one-item **JSON** array containing the default camera UUID.

setInput

Parameters:

id - camera UUID

Action: switches to a camera input.

getCurrentMode

Parameters: none

Returns: a one-item **JSON** array containing **cam**, **screen**, **video**, or **blank**.

setMode

Parameters:

mode - **cam**, **screen**, **video**, or **blank**

Action: switches the current source mode.

setPIP

Parameters: none

Action: toggles picture-in-picture visibility.

Video playback and PDFs

getVideoList

Parameters: none

Returns: {"items":[...]} with recently used **video** file dictionaries.

getVideoImage

Parameters:

id - file path to a **video**

Returns: {"icon":"data:image/jpg;base64,..."} for the **video** thumbnail.

setVideo

Parameters:

id - **video** file path or **video** identifier used by the current integrations

Action: starts **video** playback and switches to **video** mode when appropriate.

setPlayPause

Parameters: none

Action: toggles **play**/pause for the current **video** source.

setPrevSlide

Parameters: none

Action: moves to the previous page on the active PDF overlay. If **no** PDF overlay is active, Ecamm will use any available PDF overlay.

setNextSlide

Parameters: none

Action: moves to the next page on the active PDF overlay. If **no** PDF overlay is active, Ecamm will use any available PDF overlay.

Overlays

getOverlayList

Parameters:

filter - optional; use **animated** to return **animated** overlays only

Returns: {"items":[...]} with overlay dictionaries.

getOverlayDict

Parameters: none

Returns: a **JSON** object keyed by overlay UUID.

getOverlayImage

Parameters:

id - overlay UUID

play - optional; use **yes** to add the **play** badge used by **animated**-overlay controls

Returns: {"icon":"data:image/jpeg;base64,..."} for the overlay thumbnail.

setOverlay

Parameters:

id - overlay UUID

action - optional; use **play** to **play** an **animated** overlay instead of toggling visibility

pauseresume - optional; use **true** with action=**play** to toggle pause/resume behavior

Action: toggles overlay visibility or plays an **animated** overlay.

setOverlayText

Parameters:

id - overlay UUID

text - plain text string

Action: replaces the text contents of a text or fixed overlay while keeping its current styling.

setHideComment

Parameters: none

Action: hides the most recent comment overlay.

setShowComment

Parameters: none

Action: shows the most recent comment overlay.

setOverlayList

Parameters:

action - **up**, **down**, or **pick**

Action: drives the overlay list selection UI for control-surface integrations.

Sounds

getSoundList

Parameters: none

Returns: {"items":[...]} with sound dictionaries.

getSoundDict

Parameters: none

Returns: a **JSON** object keyed by sound UUID.

getSoundImage

Parameters:

id - optional sound UUID

Returns: {"icon":"data:image/png;base64,..."} for the sound thumbnail or group icon.

setSound

Parameters:

id - sound UUID, PRELOADED/... sound identifier, or file path

volume - optional level from 0 to 100

action - optional playback behavior; stop, loop, or restart

Action: plays a sound effect.

setSoundVolume

Parameters:

volume - level from 0 to 100

Action: sets the volume of the currently playing sound.

setSoundStop

Parameters: none

Action: stops the currently playing sound.

setSoundPause

Parameters: none

Action: pauses the currently playing sound.

setSoundList

Parameters:

action - **up**, **down**, or **pick**

Action: drives the sound list selection UI for control-surface integrations.

Audio filters

getAudioFilterList

Parameters: none

Returns: {"items": [...]} with audio filter dictionaries.

getAudioFilterDict

Parameters: none

Returns: a **JSON** object keyed by audio filter UUID.

setAudioFilter

Parameters:

id - audio filter UUID

Action: toggles an audio filter.

Publishing, comments, reactions, and profiles

getChannels

Parameters: none

Returns: {"items":[...] } with active streaming destination dictionaries.

setPublish

Parameters: none

Action: publishes the current Preview mode output.

getProfileList

Parameters: none

Returns: {"items":[...] } with profile dictionaries.

setProfile

Parameters:

id - profile UUID

Action: switches to a profile.

setComment

Parameters:

text - comment text

Action: posts a comment to the currently selected posting destination or previous destinations.

setAddMarker

Parameters:

text - optional marker text

dialogbox - optional; **true** shows the marker dialog instead of adding immediately

Action: adds a recording marker.

setReaction

Parameters:

reaction - one of Balloons, Confetti, Fireworks, Heart, Lasers, Rain, ThumbsDown, or ThumbsUp

Action: triggers a camera reaction effect on supported hardware.

getLiveDemo

Parameters: none

Returns: a one-item **JSON** array containing **yes** or **no**.

setLiveDemo

Parameters: none

Action: toggles Live Demo mode.

getHideShowUI

Parameters: none

Returns: a one-item **JSON** array containing **yes** or **no**.

setHideShowUI

Parameters: none

Action: toggles the Hide Main Window Controls feature.

Interview guest controls

setSolo

Parameters:

guest - guest index such as 1, 2, 3, or 4

Action: solos the requested guest in the interview area.

setAddLeft

Parameters:

guest - guest index such as 1, 2, 3, or 4

Action: adds the requested guest to the left side of the interview layout.

setAddRight

Parameters:

guest - guest index such as 1, 2, 3, or 4

Action: adds the requested guest to the right side of the interview layout.

Zoom controls

setZoomNew

Parameters:

pmi - optional; **true** uses the Personal Meeting ID, **false** starts a regular instant **meeting**

Action: starts a new Zoom **meeting**.

Note: in the raw HTTP API, omitting pmi behaves like **false**.

setZoomLeave

Parameters:

end - optional; **true** ends the **meeting** for everyone

Action: leaves the current Zoom **meeting**.

Note: in the raw HTTP API, omitting **end** behaves like **false**.

setZoomPanel

Parameters:

panel - **meeting**, **chat**, **participants**, or **sharing**

Action: shows or hides a Zoom **meeting** panel.

Note: the AppleScript bridge also accepts **share** and maps it to **sharing**.

setZoomVideo

Parameters: none

Action: starts or stops Zoom **video**.

setZoomAudio

Parameters: none

Action: mutes or unmutes Zoom microphone audio.

setZoomMuteAll

Parameters:

allowunmute - optional; **true** lets **participants** unmute themselves after the mute-all action

Action: mutes all Zoom **participants** except the host.

Note: omitting **allowunmute** behaves like **true**.

setZoomRaiseHand

Parameters: none

Action: raises or lowers the Zoom hand indicator.

setZoomFullScreen

Parameters: none

Action: toggles Zoom full **screen**.

setZoomCloudRecord

Parameters: none

Action: starts or stops Zoom cloud recording.

setZoomLocalRecord

Parameters: none

Action: starts or stops Zoom local recording.

setZoomPauseRecord

Parameters: none

Action: pauses or resumes Zoom recording.

setZoomPrevGallery

Parameters: none

Action: goes to the previous Zoom gallery page.

setZoomNextGallery

Parameters: none

Action: goes to the next Zoom gallery page.

setZoomGallerySpeaker

Parameters: none

Action: switches between Zoom gallery view and speaker view.

setSpotlightSelf

Parameters: none

Action: toggles Zoom spotlighting for the current user.

setZoomWaitingRoom

Parameters: none

Action: toggles the Zoom waiting room.

setZoomAdmitAll

Parameters: none

Action: admits all **participants** currently in the Zoom waiting room.

setZoomOriginalSound

Parameters: none

Action: toggles Zoom Original Sound.