

How to: Using EVAPI_Callback and EVAPI_CallbackEx

Overview

The evAPI callback functions are used to receive event notifications from the exacqVision server, such as motion detected by a camera or a configuration change made to a server.

Usage

To use the evAPI callback functionality, it is necessary to define and declare a callback function and then assign the function in the API. The format of the definition for the EVAPI_Callback and EVAPI_CallbackEx functions can be found in the API documentation under the Core API module.

To assign the callback function to be used by evAPI, it is necessary to call EVAPI_SetCallback or EVAPI_SetCallbackEx. If both callbacks are assigned, the API will default to using the EVAPI_Callback function and will not execute the EVAPI_CallbackEx function.

Parameters

int param - the event identifier

int value - the event state

PayloadType *payload - data associated with event

size_t length - the size of the payload parameter

PayloadType

u32 camera - camera ID or device ID associated with an event

i32 time - UTC time of the event

u16 ms - millisecond part of the event time

u8 type - 0 for audio, >=1 for video

u8 flags - 0x80 for motion, 0x40 for alarm

u32 trigger - trigger ID associated with an event

i32 time_end - UTC time of the end of the event

u16 ms_end - millisecond part of the event end time

ParamTypes

ParamType Discover - indicates a server was found.

param - Discover

value - N/A

payload - NULL

length - size of payload

ParamType Login - indicates there was a login attempt on the server.

param - Login

value - 0 for success, otherwise error

payload - NULL

length - size of payload

ParamType Config - indicates configuration information has been sent from the server.

param - Config

value - N/A

payload - NULL

length - size of payload

Notes: A ParamType Config is sent any time the configuration of the server changes, such as when a camera is added or removed from the server or a soft trigger is created.

ParamType Status - indicates status information has been sent from the server.

param - Status

value - N/A

payload - NULL

length - size of payload

ParamType Loss - indicates that there is no video signal detected on an analog camera input for the camera id in the payload.

param - Loss

value - 1 for video loss, 0 for no video loss

payload - camera

time

ms

length - size of payload

ParamType Motion - indicates that motion was detected on the camera that corresponds to the camera id in the payload.

param - Motion

value - 1 for motion, 0 for no motion

payload - camera

time

ms

length - size of payload

ParamType Connected - indicates the server no longer detects a previously detected IP camera with the camera id in the payload.

param - Connection

value - 1 for connected, 0 for not connected

payload - camera

time

ms

length - size of payload

ParamType Live - indicates that a live video frame corresponding to the camera id in the payload has been received and decompressed.

param - Live

value - N/A

payload - camera

Created On: 12th February 2021

KB Number: KB-00469-469-210212

time

ms

type

flags

length - size of payload

Notes: The type payload field indicates whether audio or video data is being streamed. The flags payload field indicates whether motion or an alarm associated with the camera has been triggered.

ParamType Stored - indicates that a stored video frame corresponding to the camera id in the payload has been received but NOT decompressed.

param - Stored

value - N/A

payload - camera

time

ms

type

flags

length - size of payload

Notes: The type payload field indicates whether audio or video data is being streamed. The flags payload field indicates whether motion or an alarm associated with the camera has been triggered.

ParamType SoftTrigger - indicates that a soft trigger has been set to either on or off on the server.

param - SoftTrigger

value - 1 for trigger on, 0 for trigger off

payload - time

ms

trigger

length - size of payload

ParamType InputTrigger - indicates that an input trigger has been set to either on or off on the server.

param - InputTrigger

value - 1 for trigger on, 0 for trigger off

payload - camera

time

ms

trigger

length - size of payload

Notes: The camera payload field represents the device ID associated with the input trigger.

ParamType EVAPI_DISPLAY - indicates that a frame is ready for display after a call to EVAPI_SetFrame. This means that the frame has been decompressed.

param - EVAPI_DISPLAY

value - N/A

payload - camera

time

Created On: 12th February 2021

KB Number: KB-00469-469-210212

ms

type

flags

length - size of payload

ParamType EVAPI_SEARCH_COMPLETE - indicates the server has completed and sent all the found video or meta data from a search.

param - EVAPI_SEARCH_COMPLETE

value - N/A

payload - camera

time

ms

type

flags

length - size of payload

ParamType EVAPI_TEMPERATURE - indicates the temperature status of a device on the server.

param - EVAPI_TEMPERATURE

value - 1 for alarm, 0 for no alarm

payload - camera

time

ms (optional)

length - size of payload

Notes: The camera payload field may also represent a device_id.

ParamType EVAPI_LICENSE - indicates exacq license information has been received from the server.

param - EVAPI_LICENSE

value - N/A

payload - NULL

length - size of payload

ParamType EVAPI_DEVICE_FAIL – indicates that a capture card on the server has failed to initialize.

param – EVAPI_DEVICE_FAIL

value – 1 for alarm, 0 for no alarm

payload – camera (device ID)

- time
- ms (optional)

length - size of payload

ParamType EVAPI_STORAGE_ALARM - indicates the status of a storage drive on the server.

param - EVAPI_STORAGE_ALARM

value - 1 for over threshold, 0 for below or equal to threshold

payload - camera

time

ms

length - size of payload

Notes: The drive number associated with the alarm is indicated by the camera field of the Payload.

ParamType EVAPI_DEVICE_BUTTON - indicates the status of the front panel push button on the exacq server.

param – EVAPI_DEVICE_BUTTON

value – 1 for alarm, 0 for no alarm

payload – camera (device ID)

- time

- ms (optional)

length - size of payload

ParamType EVAPI_LICENSE_PACKET - indicates that a license packet has been received after a call to EVAPI_LicenseRequest.

param – EVAPI_LICENSE_PACKET

value – N/A

payload – NULL

length - size of payload

ParamType EVAPI_CORE_THROTTLING – indicates the status of the server throttling video.

param – EVAPI_CORE_THROTTLING

value – 1 for alarm, 0 for no alarm

payload – time

length - size of payload

ParamType EVAPI_DEVICE_FAN_FAIL - indicates the status a video capture card fan.

param – EVAPI_DEVICE_FAN_FAIL

value – 1 for alarm, 0 for no alarm

payload – camera (device ID)

- time
- ms (optional)

length - size of payload

ParamType EVAPI_SYS_MGMT – indicates the status of storage device.

param – EVAPI_SYS_MGMT

value – 1 for alarm, 0 for no alarm

payload – camera (storage ID)

- time

length - size of payload

ParamType UpdateStatus – indicates the status of a remote server update.

param – UpdateStatus

value –

INSTALL_TIMEOUT	= -21,
INSTALL_ERROR	= -20,
WGET_PARSE_ERROR	= -17,
WGET_FILE_IO_ERROR	= -16,
WGET_NETWORK_ERROR	= -15,
WGET_SSL_ERROR	= -14,
WGET_AUTH_ERROR	= -13,

WGET_PROTOCOL_ERROR	= -12,
WGET_HOST_ERROR	= -11,
WGET_ERROR	= -10,
INVALID_PATH_URI	= -6,
INVALID_HOST_NAME	= -5,
CHECKSUM_FAILED	= -4,
INVALID_URI_LOCATION	= -3,
NO_BUNDLE_NAME	= -2,
UPDATE_FAILED	= -1,
NO_UPDATE	= 0,
BUNDLE_DOWNLOAD	= 1,
BUNDLE_INSTALL	= 2,
PENDING_UPDATE	= 3,
BUNDLE_FILE	= 4,
UPDATE_SUCCESS	= 13

payload – time

length - size of payload

ParamType EVAPI_META_DATA_SEARCH – returns video meta data from a EVAPI_MetaDataSearch.

param – EVAPI_META_DATA_SEARCH

value – 0

payload – camera

- time (beginning of meta data block)
- flags (0x80 for motion, 0x40 for alarm, 0x0 for free run)
- time_end (end of meta data block)

length - size of payload

ParamType EVAPI_SOFT_TRIGGER_SEARCH – returns results from a soft trigger search.

param – EVAPI_SOFT_TRIGGER_SEARCH

value – 0

payload –

- time
- time_ms
- trigger
- time_end
- ms_end

length - size of payload

ParamType EVAPI_INPUT_TRIGGER_SEARCH – returns results from an input trigger search.

param – EVAPI_INPUT_TRIGGER_SEARCH

value – 0

payload –

- time
- time_ms
- trigger
- time_end
- ms_end

length - size of payload

ParamType EVAPI_ARCHIVE_ALARM – indicates if the server has lost connection to an archive target .

param - Connection

Created On: 12th February 2021

KB Number: KB-00469-469-210212

value - 1 for connected, 0 for not connected

payload – camera (target ID)

- time

length - size of payload

Example

Function evCallback(parameters: int param, int value, PayloadType *payload, size_t length)

Switch on param

Case Discover

Do something

End Case

Case Login

Do something

End Case

....

Continue for other params

....

End Switch

End Function evCallback

Function Main

Initialize evAPI (EVAPI_INIT)

Set evAPI callback to evCallback (EVAPI_SetCallback)

....

Continue with other functions

....

End Function Main

Note: Samples of how to use the callback with C/C++ or C# are shipped with the SDK in the window and c-sharp samples.