

GETTING STARTED

^

Introduction

Authentication with API keys

Authentication with OAuth

Specifying Filter Parameters

Timezone Offsets

Fields

Updating Specific Fields

Pagination

Webhooks

Rate Limits

HTTP Response Codes

API Clients

RESOURCES

^

Advanced Filtering

Leads

Contacts

Activities

Call

Created

Email

EmailThread

LeadStatusChange

Meeting

Note

OpportunityStatusChange

SMS

TaskCompleted

Opportunities

Tasks

Memberships

Users

Organizations

Roles

Custom Fields

Lead Custom Fields

Contact Custom Fields

Opportunity Custom Fields

Activity Custom Fields

Shared Custom Fields

Custom Field Schemas

Lead Statuses

Opportunity Statuses

Pipelines

Groups

Reporting

Email Templates

SMS Templates

Connected Accounts

Send As

Sequences

Event Log

Most actions in Close that change an object are logged in the event log. For example, creating a lead, sending an email, or deleting a note will create an event log entry. The event log API allows you to access these events, up to 30 days back in history.

View the list of logged actions

Note: For privacy reasons, certain fields are only visible to non-admins for one hour.

Event format

An event object consists of the following fields:

- `date_created`: Event creation date
- `date_updated`: Even update date. Note that existing events can be updated if multiple actions to the same object happen in a short time frame. In this case, an event's `date_updated` may change, but the `date_created` will stay the same.
- `organization_id`: The organization ID
- `user_id`: The ID of the user who generated this event, or `null` if the event wasn't directly generated by a user action.
- `request_id`: A string with the unique request identifier. Multiple events from the same request can be associated by looking at the request identifier.
- `api_key_id`: The API key ID used if the event was generated by a request made through the Close API using an API key, otherwise `null`.
- `object_type`: Type of the object. Examples are `lead`, `activity.email`, `status_change.opportunity`. For a full list, see [List of events](#).
- `object_id`: The ID of the object that is affected.
- `lead_id`: The ID of the corresponding lead, if any, or `null`. In case of a lead event, this is equivalent to `object_id`.
- `action`: The type of the event. The most common types are:
 - `created`: An object was created.
 - `updated`: An object was updated.
 - `deleted`: An object was deleted.

Certain object types can generate other actions (`merged`, `completed`, `sent`, etc.). For a full list, see [List of events](#).

- `changed_fields`: For update events, contains a list of fields that have changed.
- `data`: Contains the payload of the affected object as if you fetched it via API (`null` for delete events). For non-admins, this field is only returned for events less than one hour old.
- `previous_data`: For update events, this contains a dictionary of fields that have changed and their previous values. For delete events, this contains all the object attributes. For non-admins, this field is only returned for events less than one hour old.
- `meta`: Contains additional helpful information for certain event types.
 - `bulk_action_id`: For events caused by bulk actions, contains the bulk action ID.
 - `merge_source_lead_id`: For lead merge events, contains the ID of the source lead (that is being merged into the destination lead).
 - `merge_destination_lead_id`: For lead merge events, contains the ID of the destination lead (that the source lead merges into).
 - `request_method`: For events that are associated with an HTTP request, contains the request method of the request (e.g. `POST`).
 - `request_path`: For events that are associated with an HTTP request, contains the request path of the request (e.g. `/api/v1/lead/`).

Event consolidation

Multiple `updated` events for the same object may be consolidated into one event. In this case, an existing event is updated: The updated event will retain the original `date_created` and have a new `date_updated`. The updated event's `data` will contain the updated state of the object, and the attributes in `previous_data` are merged (e.g. if a lead's name changes from A to B, and then from B to C, the updated event would contain `{"data": {"name": "C"}, "previous_data": {"name": "A"}}`).