Observation Event Monitoring API Reference

Overview

The Observation API provides a cursor-based pull endpoint for continuously retrieving event data from your account. It is designed for integration with SIEM platforms such as Microsoft Sentinel, Splunk, and Cribl.

The API uses a **token-based streaming model** rather than a date-range query model. A single opaque cursor replaces the need to track paging offsets, date windows, and gap detection. You store one value between calls and always pick up exactly where you left off.

Base URL: `https://{your-domain}/v1/events`

Authentication

All requests require a Bearer token in the `Authorization` header.

```
Authorization: Bearer obs_Ab3fK9x2...
```

API keys are prefixed with `obs_` and are generated through the administration interface. Keys are stored as SHA-256 hashes; the raw key is shown only once at creation time. If lost, generate a new key (the old key is immediately revoked).

Endpoint

GET /v1/events

Retrieve observation events using cursor-based pagination.

Query Parameters

Parameter	Type	Required	Default	Description
`cursor`	integer	No	0	Return events after this position. Omit or pass `0` for the first request.
`limit`	integer	No	1000	Maximum events per response. Range: 1-1000.

Request Headers

Header	Required	Description
`Authorization`	Yes	`Bearer <api_key>`

Response Headers

Header	Type	Description
`X-Next-Cursor`	integer	Cursor to use in the next request. `0` if no events were returned.
`X-Has-More`	string	`"true"` if more events may be available, `"false"` if you are caught up.
`Content-Type`	string	`application/x-ndjson`

`Retry-After`	integer	Seconds to wait before retrying (only on 429).

Response Body (200 OK)

The response body is **NDJSON** (Newline-Delimited JSON) -- one JSON object per line:

```
{"event_id": "ts-a1b2c3d4-e5f6-7890-abcd-ef1234567890", "type": "transfer_sent", "timestamp": "2025-01-
15T10:30:00", "data": {"sender": "alice@example.com", "recipient": "bob@example.com", "file_count": 3}}
{"event_id": "fd-b2c3d4e5-f6a7-8901-bcde-f12345678901", "type": "file_downloaded", "timestamp": "2025-01-
15T10:31:00", "data": {"file_name": "report.pdf", "downloaded_by": "bob@example.com"}}
```

Each event object contains:

Field	Type	Description
`event_id`	string	Unique event identifier. Prefixed by type: `ts-` (transfer_sent), `fd-` (file_downloaded), `ev-` (other).
`type`	string	Event type (e.g., `transfer_sent`, `file_downloaded`).
`timestamp`	string	ISO 8601 timestamp of when the event occurred.
`data`	object	Type-specific event payload.

Error Responses

401 Unauthorized -- Missing or invalid API key.

```
{"error": "unauthorized", "message": "Invalid API key"}
```

400 Bad Request -- Invalid parameter.

```
{"error": "bad_request", "message": "Invalid cursor"}
```

```
{"error": "bad_request", "message": "limit must not exceed 1000"}
```

429 Too Many Requests -- Rate limit exceeded. Includes `Retry-After` header.

```
{"error": "rate_limited", "message": "Too many requests"}
```

Rate Limiting

The API enforces a per-account rate limit. The default interval is **60 seconds** (one successful request per minute per account).

When rate-limited, the response includes a `Retry-After` header indicating the number of seconds to wait before the next request will be accepted.

How the Cursor Works

The cursor is an integer that represents your position in the event stream. It encodes both **paging** and **time progression** into a single value.

Key properties:

- **Monotonically increasing:** Each cursor is strictly greater than the previous one.
- **Stateless on the server:** The server does not track your position. You store the cursor yourself.
- **Gap-safe:** If events are deleted or gaps exist in the sequence, the cursor still returns correct results.
- **Time-ordered:** Events are stored and returned in chronological order, so advancing the cursor is equivalent to advancing forward in time.

The cursor IS your time filter. You do not need to specify `earliest` or `latest` times. The cursor automatically ensures you only receive events that occurred after your last retrieval.

Event Types

Type	ID Prefix	Description
`transfer_sent`	`ts-`	A file transfer was sent.
`file_downloaded`	`fd-`	A file was downloaded by a recipient.
Other types	`ev-`	Other observation events.

Pagination Behavior

The `limit` parameter controls the maximum number of events returned per request. The 1,000-event limit is **per page, not a total cap**.

- If the response returns exactly `limit` events, `X-Has-More` will be `"true"` -- call again with the new cursor to get the next page.
- If the response returns fewer than `limit` events, `X-Has-More` will be `"false"` -- you are caught up.
- If no events are available, the response body is empty, `X-Has-More` is `"false"`, and `X-Next-Cursor` echoes back your input cursor.