opEvents REST API Reference

General

opEvents versions 2.0 and newer provide a REST-style API for limited event management, e.g. creation and retrieval of single events, and lookup of events.
This version also comes with a simple client (both source and compiled), `/usr/local/omk/bin/create_remote_event.pl` (and `.exe`).

Authentication

Authentication is required to access any of the methods listed below.

POST to /login to Authenticate

`POST /omk/opEvents/login`

You must authenticate to opEvents first. Two parameters must be supplied as post-body, username and password.

Request

Parameter	Description
username	The username to authenticate with
password	The password for the user

Successful Response

A cookie is created and sent with the response. This must be saved and used with subsequent requests.

Event Management

POST to /events for Event Creation

`POST /omk/opEvents/events`

The body of the request must be a valid JSON document, containing the desired event properties. See opEvents Normalised Event Properties for a description of all the properties. Some properties (e.g. `date`, `time`) can be omitted and will be filled in automatically. As an absolute minimum, a `node` or a `host` property, and an `event` property must be present. If (and only if!) `node` is not present, then opEvents looks up `host` and attempts to find the canonical node for the hostname or IP address from the `host` property - this heuristic is described in more detail here. The resulting `node` must be known to opEvents and must not be disabled for opEvents.

Request Example for Node Down

```
{
"node": "test-node",
"event": "Node Down",
"level": "Major",
"priority": 6,
"state": "down",
"stateful": "Node",
"details" : "Ping failed"
}
```

Request Example for Node Up

```
{
"node": "test-node",
"event": "Node Up",
"level": "Normal",
"priority": 1,
"state": "up",
"stateful": "Node",
"details" : ""
}
```

Successful Response

HTTP Status	HTTP Headers	Body	Description
201	Location	JSON object with `success` and `id` properties	The success property is set to 1 and only if the request was successful. The id property is the new event's ID (but see the Limitations section below) The `Location` header contains the complete URL for retrieving the newly created event.

Unsuccessful Response

HTTP Status	Body	Description
401	N/A	You are not authenticated.
404	N/A	You are authenticated but not authorised to create events in opEvents.
400	JSON object with `success` being `0` and an `error` property.	The `error` property contains an explanation of what went wrong with your request, e.g. why the parameters were considered invalid.

GET of /events for Event Listing

`GET /omk/opEvents/events`

If your GET call provides an `Accept` header indicating JSON, or if you use `/events.json` as URI, then opEvents will look for matching events and return their properties in the form of a JSON object, an array of events. Extra query parameters can be used to narrow down the listing or search for particular events only; without parameters you will get all events of the last two hours.

Optional Query Parameters

Parameter	Description
`o_start`, `o_end`	Start and end of the period you are interested in. Takes all standard opEvents Date/Time formats and UNIX seconds. Note, you should pass o_summarise=1 to make sure UNIX time is not rounded.
`o_node_name`	Name of the node you are interested in. You can use "`regex:<regular expression>`" or a plain text string.

`ev_event_name`	Name of the event you are interested in. You can use "`regex:<regular expression>`" or a plain text string.
`ev_event_type`	Type of the event, i.e. what source it came from. e.g: `nmis_eventlog`, `api`.
`ev_event_element`	Element in question. Not present for all events. Regex or plain text string.
`ev_event_details`	Details that were supplied with the event. Not present for all events. Regex or plain text string.

Successful Response

HTTP Status	Body	Description
200	JSON array of objects	Each array element is a JSON object with the raw properties of the event in question.

Unsuccessful Response

HTTP Status	Body	Description
401	N/A	You are not authenticated.
404	N/A	You are authenticated but not authorised to create events in opEvents.
200	Empty JSON array	Your request was valid, but there were no matching events.

GET of /events/<eventid> for Event Retrieval

If your GET call provides an `Accept` header indicating JSON, or if you use `/events/<eventid>.json` as URI, then the event will be looked up and all properties will be returned in the form of a JSON object.

Successful Response

HTTP Status	Body
200	JSON object with all known event properties.

Unsuccessful Response

HTTP Status	Body	Description
401	N/A	You are not authenticated.
404	N/A	You are authenticated but not authorised to view events in opEvents.
404	JSON object with an `error` property.	The `error` property contains an explanation of what went wrong with your request, e.g. if you request a non-existent event.

Limitations

- Events created by the REST API are subject to (stateful) deduplication, which is performed *asynchronously* and *after* the API call returns.
  If your newly created event is eliminated by the deduplication process, then the event ID returned by the creation API call will point to a non-existent event.