Events: What, Why, How?

SPM can graph not only performance and custom metrics, but also events. Such events may represent what is happening with a server or cluster, with an application (e.g., application or server restarts, deployments, alerts…), etc, as well as any sort of other event data that you want to correlate to metrics in SPM. Events are graphed in timeseries charts and these charts can be shown next to all SPM metrics charts. This makes it possible to easily correlate events and metrics. In addition to showing events as timeseries charts, a detailed listing of events can be seen and, of course, events can have tags and priority, and can be searched and filtered.

Events are also exposed via a REST API that let's you post, retrieve, and search your events. This REST API matches the Elasticsearch API, so you can use any Elasticsearch tool or client to post, get, and search events.

NOTE:

- To be able to use send Events to SPM, you need a Sematext account. If you don't already have it, you can create it here, it's free, no credit card needed. After you have Sematext account, create an SPM App to which Events will be sent.
- If you have already created some SPM Apps under your account in the past, you can send Events to any of them.
- If you just registered, you can create SPM Apps by following the steps after Sematext account registration, or by clicking directly here.

Event Fields

An event has the following set of fields, most of which are optional:

Field Name

Field Type

Required

Notes

timestamp

date

no

Represents time when event happened (if not specified, current time will be assumed). The format is dateOptionalTime e.g.: 2014-02-17T21:37:04+0100 or 2014-02-17T14:15:01.534471+02:00 or …

message

string

yes

Short description of event, e.g. “Elasticsearch node03 on host somehost06 restarted”. This is a default search field in SPM UI, so it is good to keep it concise, but search-friendly.

name

string

no

Event name, can be used as a short label for event, e.g. “Elasticsearch restart”.

tags

string array

no

Multivalued field. Each tag should be specified as a separate array element (e.g., “tags”:[“elasticsearch”, “restart”, “emergency fix”])

priority

string

no

You can use any values that make sense to you, like “high”, “very high” or 7. Note that sorting on this field will sort in lexicographical order.

creator

string

no

Person, application, or component that created an event. E.g., “John Smith”, “Elasticsearch”, “Some Batch Job”

data

string

no

Additional event data. It can be anything you may find useful to have along inside of event object. E.g., it could be stacktrace in case of “app_error” event, base64 encoded content of file generated during “user_registered” event, etc.

Adding Events

To post an event to your event stream use the following base endpoint:

```
http://event-receiver.sematext.com/APPLICATION_TOKEN/event
```

A single application token must be specified in the URL. Thus, to send multiple events associated with multiple applications, separate call to the API will need to be made for each application. You can add event type as a field in json message (e.g, **alert**, **app_restart**, **server_restart**, **reboot**, **deployment**…), but we suggest using a smaller number of distinct event types (1-10) to keep things manageable.

Example 1

Consider some SPM application whose token (your app tokens are at: https://apps.sematext.com/users-web/services.do) is **1111111-2222-3333-4444-555555555555**. To send a **server_restart** event call the Events API with token and event type:

http://event-receiver.sematext.com/1111111-2222-3333-4444-555555555555/event

with POST content in JSON format like this:

```
{
  "timestamp" : "2014-02-17T15:29:04+0100",
  "message": "Application MyApp on MyHost04 restarted",
  "type" : "server_restart"
}
```

To post the above event with curl use:

```
curl -XPOST "http://event-receiver.sematext.com/1111111-2222-3333-4444-555555555555/event" -d
{
  "timestamp" : "2014-02-17T15:29:04+0100",
  "message" : "Application MyApp on MyHost04 restarted",
  "type" : "server_restart"
}
'
```

Example 2

Same SPM Solr application, but we want to post **deployment** event with more event properties populated. In this case the HTTP endpoint would be:

http://event-receiver.sematext.com/1111111-2222-3333-4444-555555555555/event****

with HTTP POST content:

```
{
  "timestamp" : "2014-02-17T15:58:04+0100",
  "message": "Solr 4.6.1 version deployed on prodhost06",
```

```
  "name" : "Solr 4.6.1 deployment",
  "tags" : ["solr", "4.6.1", "deployment", "upgrade"],
  "priority" : "High",
  "creator" : "John Smith",
  "type" : "deployment"
}
```

or, again with curl:

```
curl -XPOST "http://event-receiver.sematext.com/1111111-2222-3333-4444-555555555555/event" -d
{
  "timestamp" : "2014-02-17T15:58:04+0100",
  "message" : "Solr 4.6.1 version deployed on prodhost06",
  "name" : "Solr 4.6.1 deployment",
  "tags" : ["solr", "4.6.1", "deployment", "upgrade"],
  "priority" : "High", "creator" : "John Smith",
  "type" : "deployment"
}
'
```

Searching Events in SPM

SPM user interface lets you to show events and metrics from a specific time period. Additionally, the event chart has a search box where you can further narrow down events to only those that match the input query.

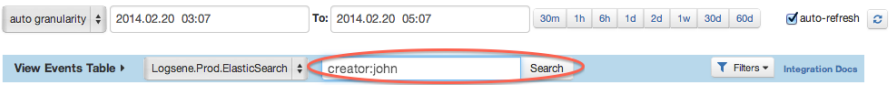

Figure 1:

The query syntax is specified by Elasticsearch's query string query, as described here.

You can search on any event field you included in the event when posting it.

Searching Events Programmatically

SPM exposes the Events Search HTTP API - as Elasticsearch search API - so events can be searched and retrieved programmatically/remotely, via HTTP, using curl or any other Elasticsearch client. The API endpoint is:

`http://event-receiver.sematext.com/APPLICATION_TOKEN`

Alternatively, you can also use the same endpoint which was used when adding events, where event type is specified, in which case the matching events will be limited to the type specified in the URI:

```
http://event-receiver.sematext.com/APPLICATION_TOKEN/event
```

The simplest way to run a query is using URI search, like this:

$ curl -XGET "http://event-receiver.sematext.com/1111111-2222-3333-4444-555555555555/_search?q=creator:john"

More query options are available when using request body search, e.g.:

```
curl -XGET "http://event-receiver.sematext.com/1111111-2222-3333-4444-555555555555/_search" -
  "query" : {
    "query_string" : {
      "query" : "MyHost04",
      "default_field" : "message"
    }
  }
'
```

This example shows how to use one of the simpler query types - query_string. To see which other query types are available, please check Elasticsearch docs.

Posting Events via HTTPS

You can use HTTPS instead of HTTP for all calls, in which case the endpoint becomes:

```
https://event-receiver.sematext.com/APPLICATION_TOKEN
```

Note: when using curl, you may experience **"SSL certificate problem"** errors. The reason is that curl doesn't bundle any CA certs any more, for more info see this. Regardless of curl errors, HTTPS communication should be functional.