

Zendesk Events

Last Modified on 01/02/2020 12:47 pm EST

SAP Cloud Platform Open Connectors supports events via polling or webhooks depending on the API provider. For more information about our Events framework, see [Events Overview](#).

Note: SAP Cloud Platform Open Connectors normalizes only the `objectId`, `objectType`, and `eventType` event data. Event data also contains raw data, which the provider returns. The raw data varies based on the endpoint.

Supported Events and Resources

SAP Cloud Platform Open Connectors supports polling events and webhooks for . After receiving an event, SAP Cloud Platform Open Connectors standardizes the payload and sends an event to the configured callback URL of your authenticated connector instance.

Polling

In order to enable polling, add these extra configurations to your instance JSON:

```
"event.notification.enabled": "true",  
"event.notification.callback.url": "",  
"event.poller.configuration": ""
```

instance JSON with polling events enabled:

```

{
  "element": {
    "key": "zendesk"
  },
  "providerData": {
    "code": "Code on Return the URL"
  },
  "configuration": {
    "oauth.api.key": "",
    "oauth.api.secret": "",
    "oauth.callback.url": "https://www.my_cool_app.com",
    "zendesk.subdomain": "",
    "event.notification.enabled": "true",
    "event.notification.callback.url": "",
    "event.poller.configuration": {
      "users": {
        "url": "/hubs/helpdesk/users",
        "idField": "id",
        "pageSize": 100,
        "datesConfiguration": {
          "updatedAtField": "updated_at",
          "updatedAtFormat": "yyyy-MM-dd'T'HH:mm:ss'Z'",
          "createdAtField": "created_at",
          "createdAtFormat": "yyyy-MM-dd'T'HH:mm:ss'Z'"
        }
      }
    }
  },
  "tags": [
    ""
  ],
  "name": ""
}

```

Webhooks

After you create an instance with webhooks enabled, your app will receive event notifications from Zendesk based on our default settings

Customization is an option based on your specific needs. See customization instructions below.

Webhook JSON

```

{
  "element": {
    "key": "zendesk"
  },
  "providerData": {
    "code": "Code on Return the URL"
  },
  "configuration": {
    "oauth.api.key": "",
    "oauth.api.secret": "",
    "oauth.callback.url": "https://www.my_cool_app.com",
    "zendesk.subdomain": "",
    "event.notification.enabled": "true",
    "event.notification.callback.url": ""
  },
  "tags": [
    ""
  ],
  "name": ""
}

```

NOTE: To begin all changes to tickets, your app will be notified.

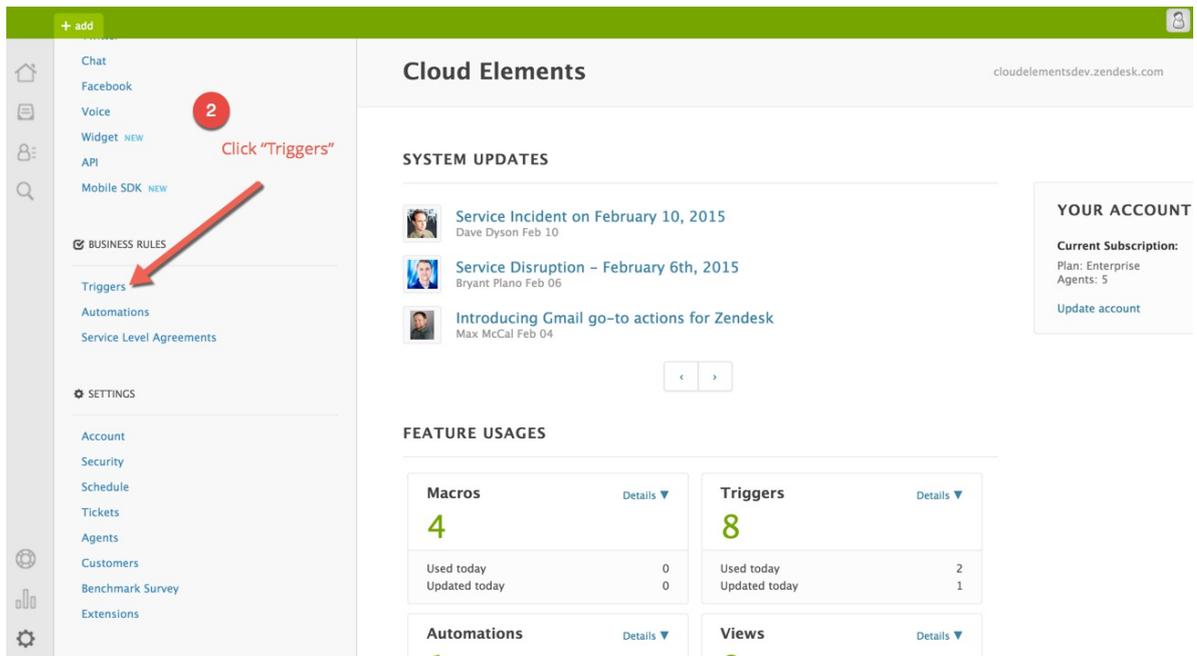
You have the option to limit that scope according to your needs.

1. Login to your Zendesk account and click “Settings”

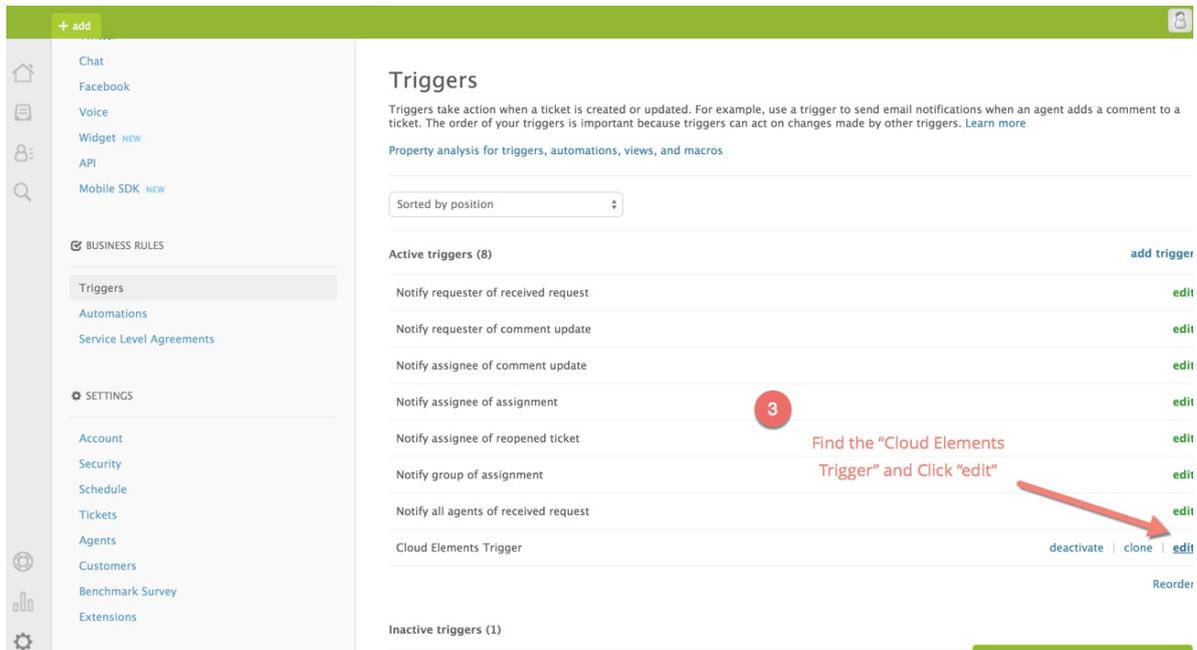
The screenshot shows the Zendesk dashboard interface. On the left sidebar, there is a gear icon representing settings. A red circle with the number '1' is placed over this icon, with a red arrow pointing to it and the text 'Click "Settings"'. The main dashboard area displays 'Open Tickets (current)' with 1 YOU and 6 GROUPS, and 'Ticket Statistics (this week)' with 0 GOOD, 0 BAD, and 0 SOLVED. Below this is a table of 'Tickets requiring your attention (4)'. The table has columns for ID, Subject, Requester, Requester updated, Group, and Assignee. The first row shows ticket #11 with subject 'Server 500 error on request' by John Smith, updated on Jan 09, in the Support group, assigned to Aaron.

ID	Subject	Requester	Requester updated	Group	Assignee
#11	Server 500 error on request	John Smith	Jan 09	Support	Aaron
#53	Problem with software	Jane Smith	Jan 11	Support	-
#54	Problem with drip irrigation hosing	Mackey's Nursery and Garden Supply	Jan 15	Support	-
#56	Computer power supply overheating	Brianna Deschutes	Jan 17	Support	-

2. Scroll and find “Triggers” and click to select

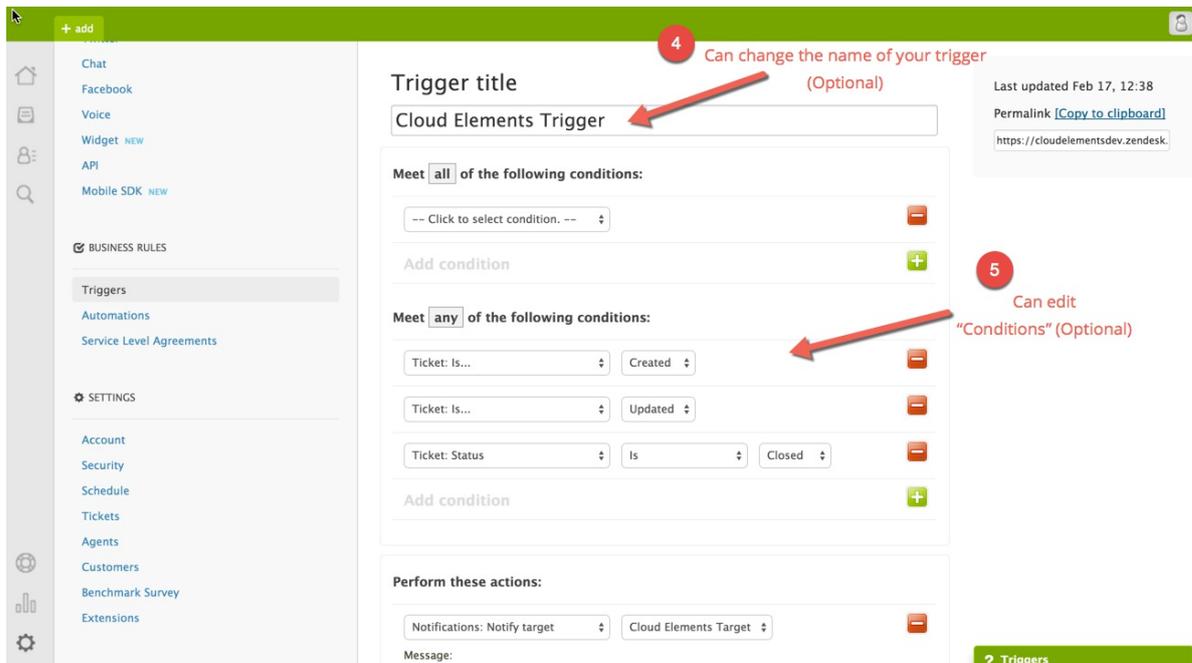


3. Find the Cloud Elements Trigger and click “edit”



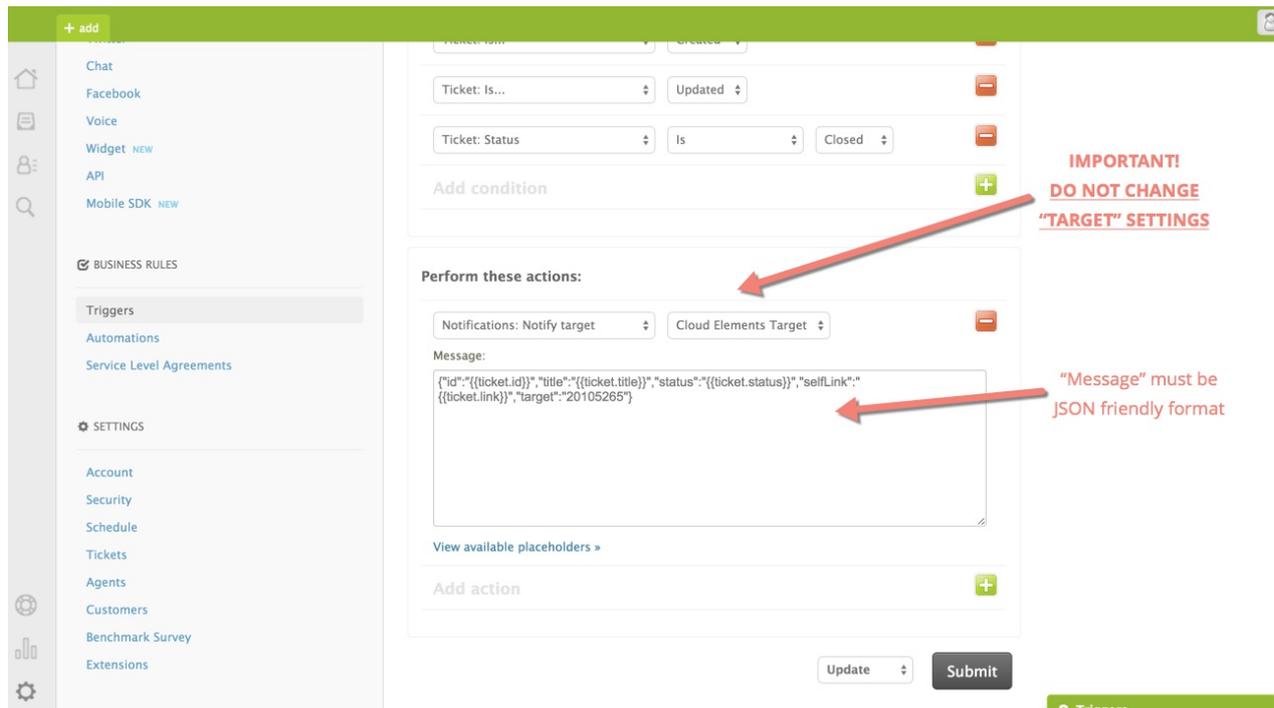
4. NOTE: the following steps are OPTIONAL. Can Change the name of the Trigger

5. Can Change the Conditions of the Trigger



IMPORTANT: Please do not remove the target field, events will not function if removed.

Events rely on the target remaining the same and Message field conforming to a JSON friendly format.



Feel free to add any of the Zendesk placeholders in your Message body, just remember to keep it JSON friendly.

- Click on the View available placeholders
- Add "Placeholders" to "Message" – Remember to keep in JSON friendly format.

6 Can Customize "Placeholders" by clicking "View available placeholders" (Optional)

Notifications: Notify target | Cloud Elements Target

Message:

```
{ "id": "{{ticket.id}}", "title": "{{ticket.title}}", "status": "{{ticket.status}}", "selfLink": "{{ticket.link}}", "target": "20105265" }
```

7 Add "Placeholders" to "Message" Remember to stay in JSON friendly format.

Available placeholders

Use placeholders to insert frequently used information into your text. Checkout the complete placeholder reference

- {{ticket.title}}**
Ticket subject.
- {{ticket.description}}**
Ticket description.
- {{ticket.link}}**
Full URL path to ticket
- {{ticket.id}}**
Unique ticket ID.
- {{ticket.external_id}}**
External ticket ID. Typically populated through the API.
- {{ticket.via}}**

? Triggers