Amazon S3 Events

Last Modified on 03/19/2020 5:49 pm EDT

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.

Supported Events and Resources

SAP Cloud Platform Open Connectors supports events via webhooks for Amazon S3.

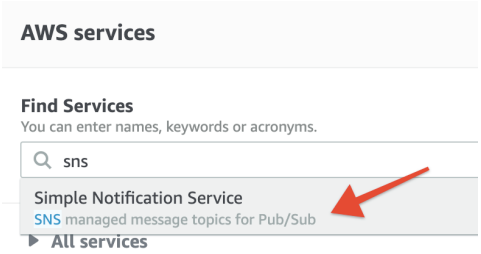
Webhooks

You can configure webhooks through the UI or through API in the JSON body of the `/instances` API call. First, you must set up webhooks on the Amazon website. SAP Cloud Platform Open Connectors uses Simple Notification Service (SNS) to enable notifications for Amazon S3.

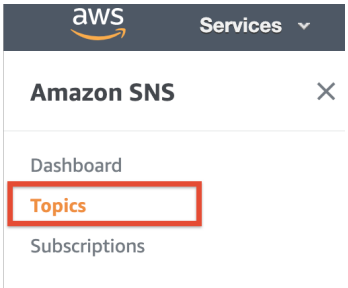
Configure Webhooks

Follow these steps to configure webhooks for Amazon S3

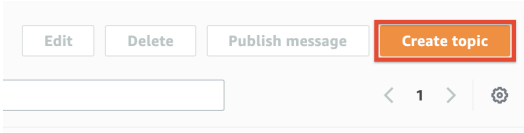
1. Via a web browser, go to https://aws.amazon.com/console/
2. SIgn in to Amazon Web Services using your credentials.
3. Once logged in, enter SNS in the search bar.



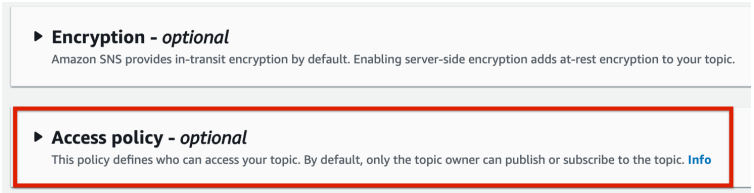
4. On the navigation panel to your left, click **Topics**.



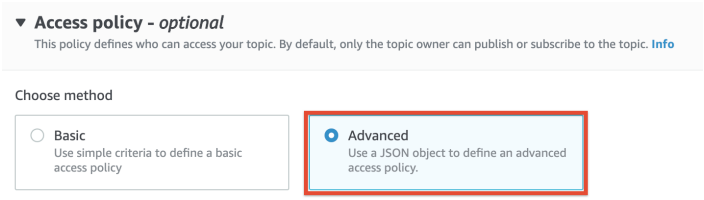
5. On the top right corner of the page, click **Create Topic.**



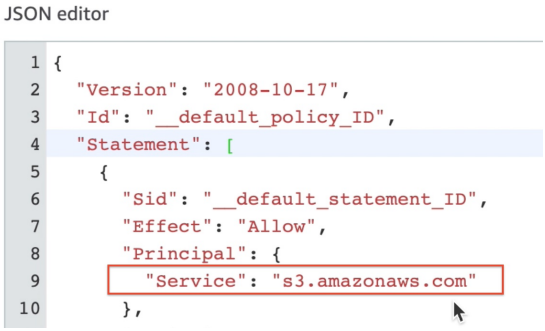
6. Enter a Topic name and Display Name (optional).
7. Click on **Access Policy.**



8. Choose the **Advanced** method to define your policy using a JSON object.



9. In the JSON editor, enter the **Service** i.e. Amazon S3's domain name.



10. Add the **Resource name** and the **Arn link** i.e. your S3 bucket name in the JSON.

11. Click on the **Create topic** button.

Use the Topic ARN and Topic Name on SAP Cloud Platform Open Connectors platform to enable webhooks for Amazon S3.

Configure Webhooks Through the UI

To configure webhooks through the UI, follow the same steps to authenticate a connector instance, and then turn on events. Fill in the two mandatory fields that appear -

- **SNS Topic Key** - Amazon S3 **Topic ARN** that you recorded while configuring Webhooks on the AWS Console.
- **SNS Topic Name** - Amazon **Topic Name** that you entered while creating a topic on the AWS Console.

For more information, see Authenticate an Connector Instance with Events (UI) or the connector-specific authentication topic.

Configure Webhooks Through API

To add webhooks when authenticating through the `/instances` API call, add the following to the `configuration` object in the JSON body. For more information about each parameter described here, see Parameters.

```
{
"event.notification.enabled": true,
"event.vendor.type": "webhooks",
"validate.instance": true,
"event.sns.topic.arn": "",
"event.sns.topic.name": ""
}
```

> **Note:** `event.notification.signature.key` is optional.

Example JSON with Webhooks

Instance JSON with webhooks events enabled:

```
{
  "element": {
    "key": "amazons3"
  },
  "configuration": {
    "filter.response.nulls": "true",
    "event.vendor.type": "webhooks",
    "event.notification.enabled": true,
    "validate.instance": "true",
    "event.sns.topic.arn": "xxxxxxxxxx",
    "event.sns.topic.name": "xxxxxxxxxx",
    "filemanagement.provider.access_key": "******",
    "filemanagement.provider.secret_key": "******",
    "filemanagement.provider.bucket_name": "xxxxxxxxxx",
    "filemanagement.provider.region_name": "xxxxxxxxxx"
  },
  "tags": [
    ""
  ],
  "name": ""
}
```


Parameters

API parameters are in `code formatting`.

Parameter	Description	Data Type
`key`	The connector key. amazons3	string
Name `name`	The name for the connector instance created during authentication.	string
Events Enabled `event.notification.enabled`	Optional. Identifies that events are enabled for the connector instance. Default: `false`	boolean
Event Type `event.vendor.type`	Optional. Identifies the type of events enabled for the instance, either `webhook` or `polling`.	string
Event Notification Callback URL `event.notification.callback.url`	For webhooks and polling. The URL where your app can receive events.	string

Parameter	For webhooks and polling. Description Optional	Data Type
Callback Notification Signature Key `event.notification.signature.key`	A user-defined key for added security to show that events have not been tampered with. This can be any custom value that you want passed to the callback handler listening at the provided Event Notification Callback URL.	string
Objects to Monitor for Changes `event.objects`	For webhooks and polling. Optional Comma separated list of objects to monitor for changes.	string
Configure Polling `event.poller.configuration`	Optional*. Configuration parameters for polling.	JSON object
tags	Optional. User-defined tags to further identify the instance.	string