# **Creating a New Action Using Adobe I/O Runtime to Trigger Multiple I/O Events**

## **Document Revision History**

| **SNo** | **Date** | **Version** | **Author** | **Changes** |
| --- | --- | --- | --- | --- |
| 1 | 17/02/2025 | 1.0 | Prakash | Created the document for new action creation using Adobe Runtime IO to trigger Multiple IO Events |
|  |  |  |  |  |
|  |  |  |  |  |

### 

### **Overview**

This guide outlines the process of creating and managing a new action using Adobe I/O Runtime. The API endpoint you create directly calls Adobe I/O Runtime, which then triggers multiple events based on the cmd parameter extracted from the XML. Depending on the value of cmd, different events will be triggered. **For example**, if cmd is 6, it will trigger the Parsed XML Event; if cmd is 5, it will trigger the Magento Notification Adobe Event. The event data will be sent to the webhook provided during event registration.

### **Source Code**

[Custom Adobe Events](https://drive.google.com/drive/folders/1liuetIWUYSUHXq-yVACeZqyQ7rl_OP1u?usp=sharing)

#### **Step 1: Log In**

Log in to Adobe I/O Runtime using the following command:

**>>** aio login

This step is necessary to authenticate your session with Adobe I/O Runtime and access its features. Logging in ensures that you have the appropriate permissions to create, manage, and invoke actions within your account.

Once logged in, follow the steps below to create and deploy a new action.

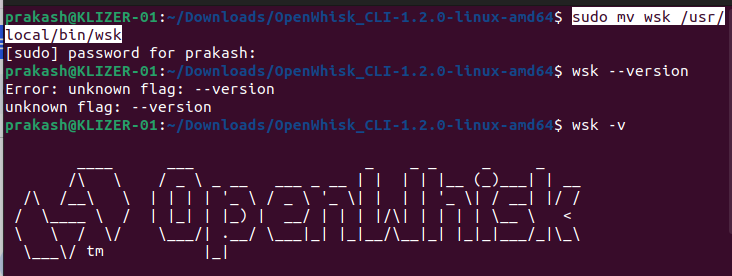
#### **Step 2: OpenWhisk Installation**

1. **Download OpenWhisk CLI**Download the OpenWhisk CLI file (OpenWhisk\_CLI-1.2.0-linux-amd64.tgz) from [this URL](https://github.com/apache/openwhisk-cli/releases).

**Extract the file and move it**Extract the file and run the following command to move it to /usr/local/bin:

**>>** sudo mv wsk /usr/local/bin/wsk

1. **Check the installation**To verify the installation, run:  
     
   **>>** wsk -v



#### **Step 3: Include External Packages**

If the action requires additional Node.js packages, follow these steps:

1. **Identify Required Packages**Identify the external Node.js packages your action depends on (e.g., axios for HTTP requests).

**Create a package.json File**In your action directory, initialize the package.json file:

**>>** npm init -y

Then, install the required packages:  
  
**>>** npm install axios

1. **Example package.json**After installing dependencies, your package.json might look like this:  
   {

"name": "descartes-action",

"version": "1.0.0",

"main": "index.js",

"dependencies": {

"axios": "^1.4.0"

}

}

1. **Zip the Files**Create a ZIP file of the action and its dependencies:  
     
   **>>** zip -r my-trackShipping.zip trackShipping.js actions/discartes-shipping/package.json actions/discartes-shipping node\_modules
2. **Check File Size**Ensure the ZIP file is under 50 MB:  
   **>>** du -sh my-action.zip
3. If the zip file exceeds 50MB, compress it or remove unnecessary packages.

**Deploy the Action**Deploy the zipped action with the following command:  
  
**>>** wsk action create TrackShipping my-trackShipping.zip --auth <AUTH\_TOKEN> --apihost https://adobeioruntime.net --kind nodejs:14

1. **Invoke the Action**  
   **>>** aio runtime action invoke TrackShipping
2. **Check Activation Details**To view the activation details:  
   **>>** aio runtime activation get <ACTIVATION\_ID>

#### **Step 4: Create the Event Provider and Metadata**

1. **Create New Event Provider**In the Adobe I/O console, create a new event provider as part of your integration.

**>>** aio event provider create

**Follow the prompts**:

* Enter the label for the provider(e.g., Descrates shipping).
* Enter description about the provider.(e.g., A custom events provider, registered by your organization).
* Add a url that contains documentation about the provider. (Optional)

**Example output:**

Creating Event Provider... done

{

"\_links": {

"rel:eventmetadata": {

"href": "https://api.adobe.io/events/providers/32b34582-bd67-4817-b953-a0b878a5656b/eventmetadata"

},

"rel:update": {

"href": "https://api.adobe.io/events/1217276/4566206088345020180/4566206088345134504/providers/32b34582-bd67-4817-b953-a0b878a5656b"

},

"self": {

"href": "https://api.adobe.io/events/providers/32b34582-bd67-4817-b953-a0b878a5656b?eventmetadata=false"

}

},

"id": "32b34582-bd67-4817-b953-a0b878a5656b",

"label": "Descrates shipping",

"description": "A custom events provider, registered by your organization.",

"source": "urn:uuid:32b34582-bd67-4817-b953-a0b878a5656b",

"provider\_metadata": "3rd\_party\_custom\_events",

"instance\_id": "fc7fa863-91d1-4a49-8474-caea1e4ea302",

"event\_delivery\_format": "cloud\_events\_v1",

"publisher": "1CEF2816641486260A495FA8@AdobeOrg"

}

**To Get Event Metadata List**

**>>** aio event provider list

**2, Create New Event Metadata or Subscriber**Use the following command to create event metadata:  
  
**>>** aio event eventmetadata create 32b34582-bd67-4817-b953-a0b878a5656b (**Event Provider ID**)

* **Follow the prompts**:
  + Enter the label for the event metadata (e.g., Magento).
  + Enter the event code (e.g., com.descartes.magento).
  + Add a description about the event metadata.

**Example output**:  
{

"\_links": {

"rel:sample\_event": {

"href": "https://api.adobe.io/events/providers/32b34582-bd67-4817-b953-a0b878a5656b/eventmetadata/com.descartes.magento/sample\_event"

},

"rel:update": {

"href": "https://api.adobe.io/events/1217276/4566206088345020180/4566206088345134504/providers/32b34582-bd67-4817-b953-a0b878a5656b/eventmetadata/com.descartes.magento"

},

"self": {

"href": "https://api.adobe.io/events/providers/32b34582-bd67-4817-b953-a0b878a5656b/eventmetadata/com.descartes.magento"

}

},

"description": "Magento",

"label": "Magento",

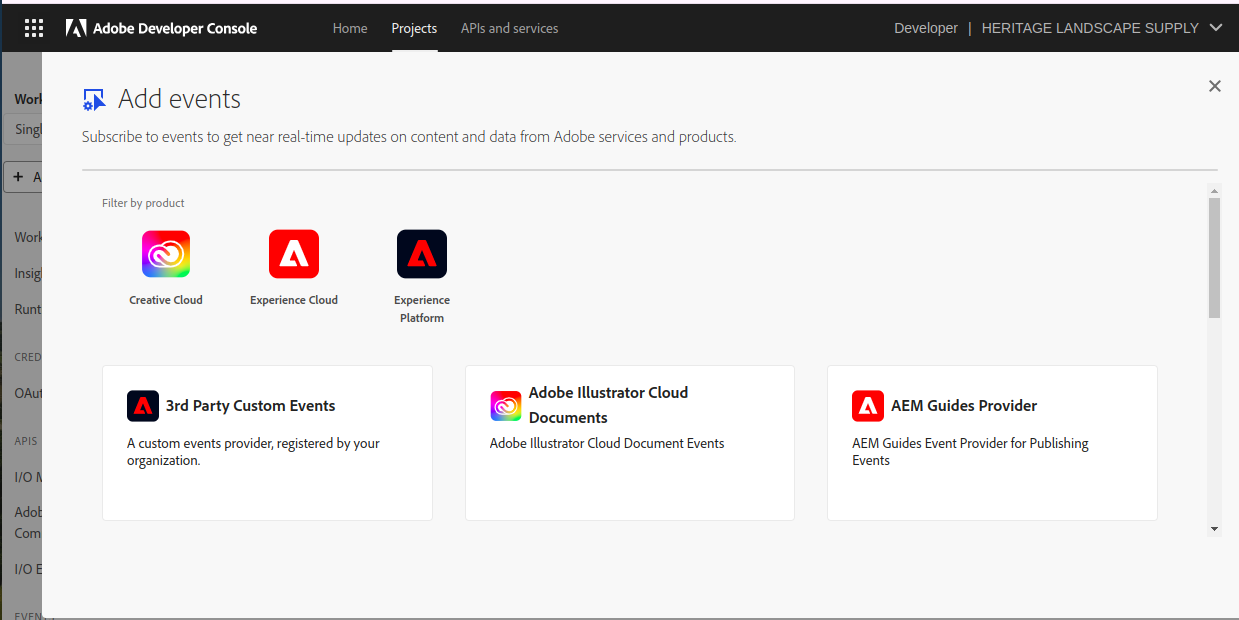
"event\_code": "com.descartes.magento"

}

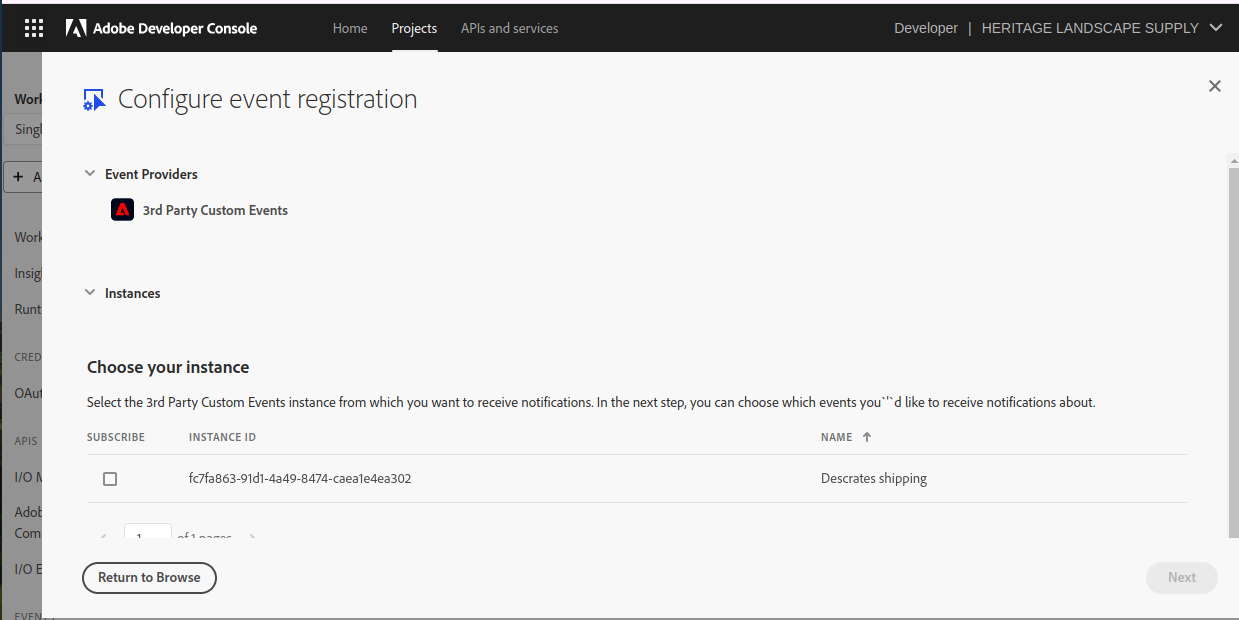
* **To Get Event Metadata List**  
  **>>** aio event eventmetadata list 32b34582-bd67-4817-b953-a0b878a5656b (**Event Provider ID**)

#### **Step 5: Creating the Custom Adobe I/O Event**

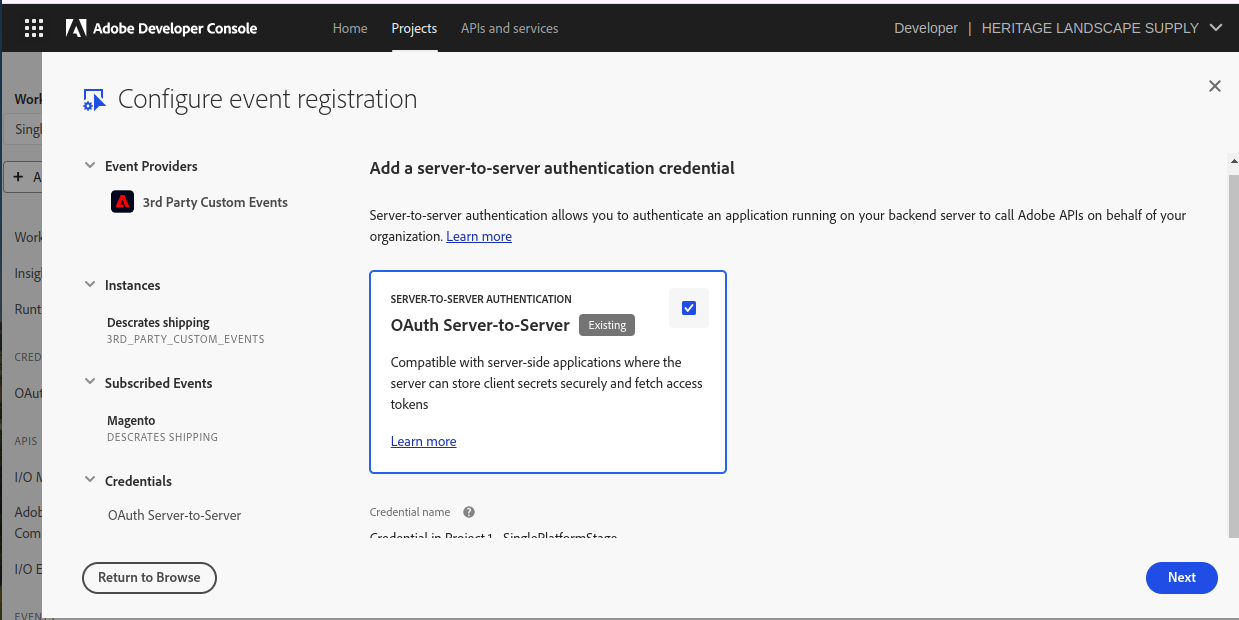
1. **Log In to the Adobe Developer Console**Log in, select the project, and click "Add Service" -> "Event".



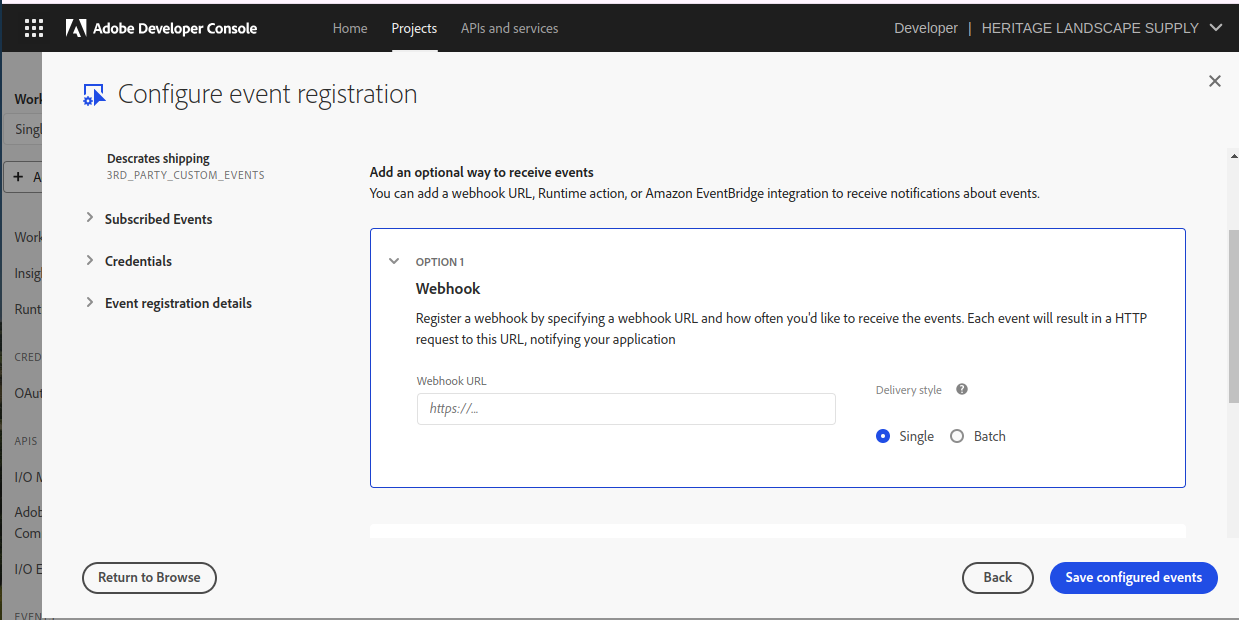
1. **Select 3rd Party Custom Events**



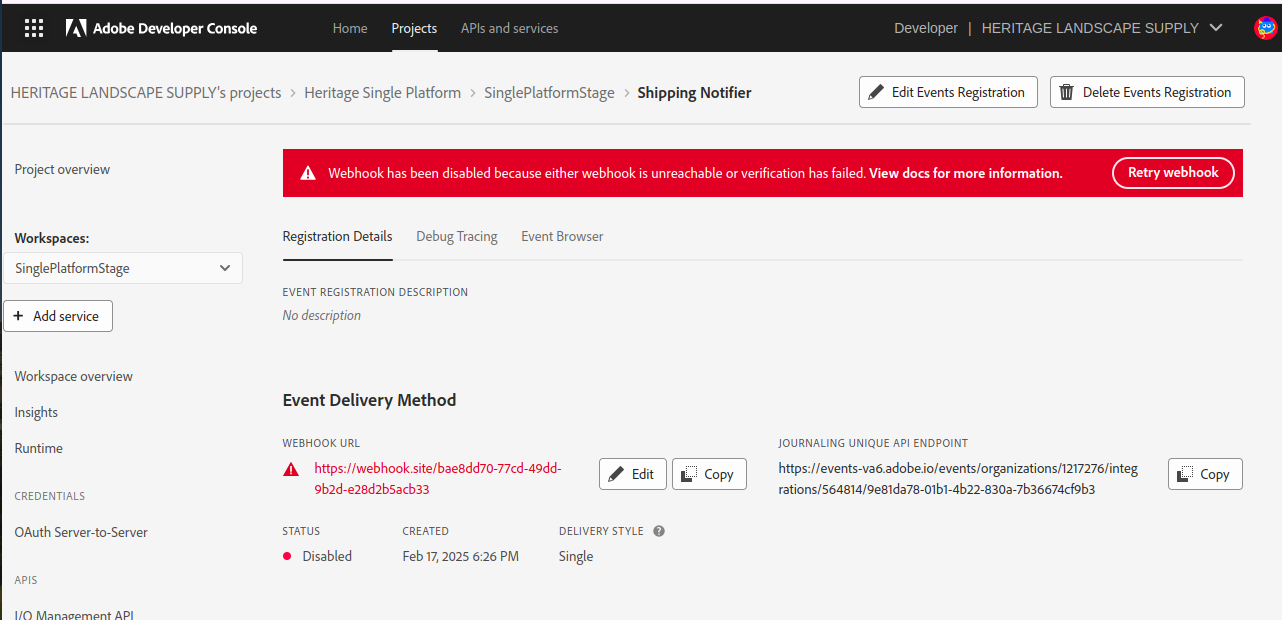
1. **Select "Event Provider"** and then choose **"Event Subscription"**.



1. **Provide the Webhook URL**Use<https://webhook.site> or your custom webhook URL.



1. **After Creating the Event**After successfully creating the event, you'll see the Journal API Endpoint, like the screenshot below:

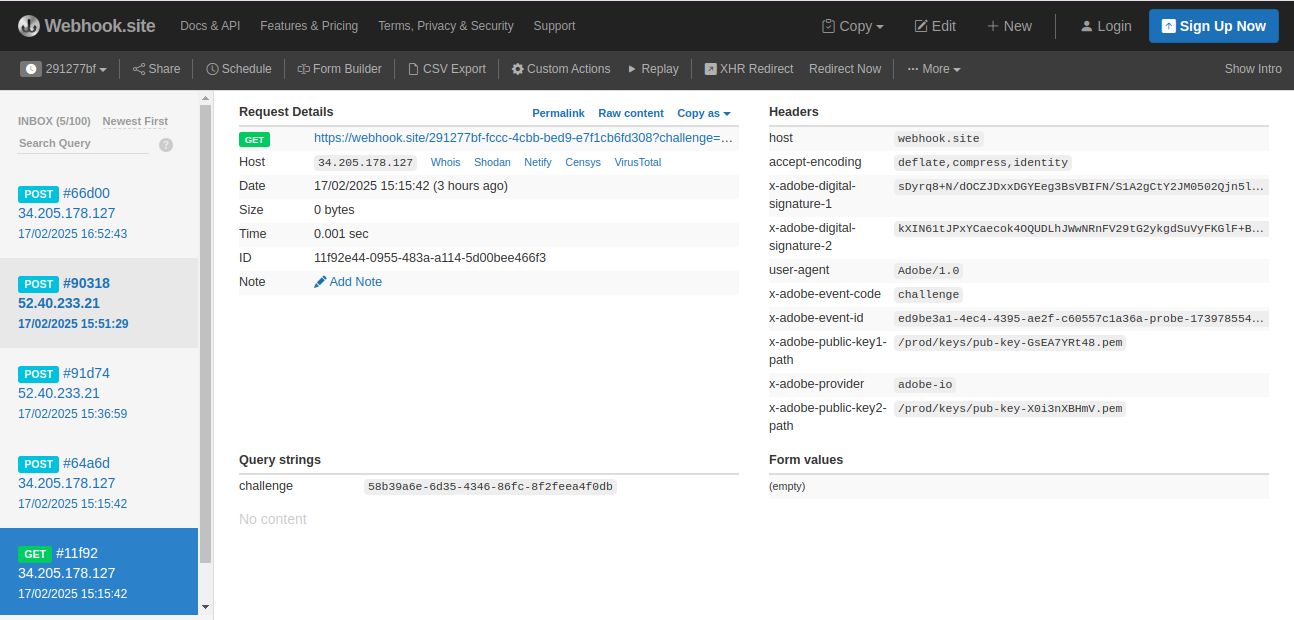


#### **Step 6: How to Activate the Webhook URL Manually**

**Steps to Get Challenge Code**After successfully registering the event, you will receive the Journal API URL:

<https://events-va6.adobe.io/events/organizations/1217276/integrations/564814/ed9be3a1-4ec4-4395-ae2f-c60557c1a36a>

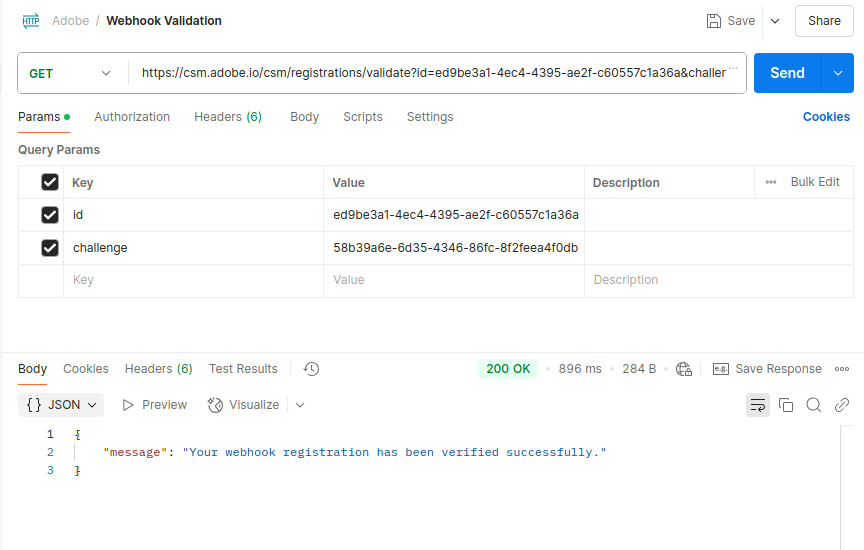
1. Extract the **Event ID** from the URL: ed9be3a1-4ec4-4395-ae2f-c60557c1a36a.
2. **Retrieve the Challenge Code**Go to your webhook.site or your custom site to retrieve the challenge code. Under the "**Query Strings**" tab, you'll find the challenge code.



**Activate Webhook URL Using Postman**

**URL**:  
<https://csm.adobe.io/csm/registrations/validate?id=ed9be3a1-4ec4-4395-ae2f-c60557c1a36a&challenge=58b39a6e-6d35-4346-86fc-8f2feea4f0db>

1. **Method**: GET  
   **Params**:
   * id: Event ID
   * challenge: Pass the challenge code here.
2. **Response**:  
   This will activate the webhook URL for your event provider.

****

#### **Step 7: Postman API to Trigger Multiple Events Using Adobe I/O Runtime**

**Overview**:  
This API directly calls Adobe I/O Runtime, which triggers multiple events based on the cmd parameter extracted from the XML. Different events are triggered based on the cmd value. **For instance**, if cmd is 6, it triggers the Parsed XML Event; if cmd is 5, it triggers the Magento Notification Adobe Event. The event data is sent to the webhook provided during event registration.

**URL**:

<https://adobeioruntime.net/api/v1/namespaces/1217276-testheritage-singleplatformstage/actions/TrackShipping?result=true&blocking=true>

**Method**: POST

**Authorization**:

* **Username**: 0a580553-46ad-4554-9155-3c115e4580ea
* **Password**: U3Ddkbw4bN3IIa2hD7G4A6bgPd8PVpVPqp1OnbPpODeyFCiuF8260sXVjJ5ex3g0

**Params**:

* result=true: To receive only the response.
* block=true: To receive both the activation ID and the response.

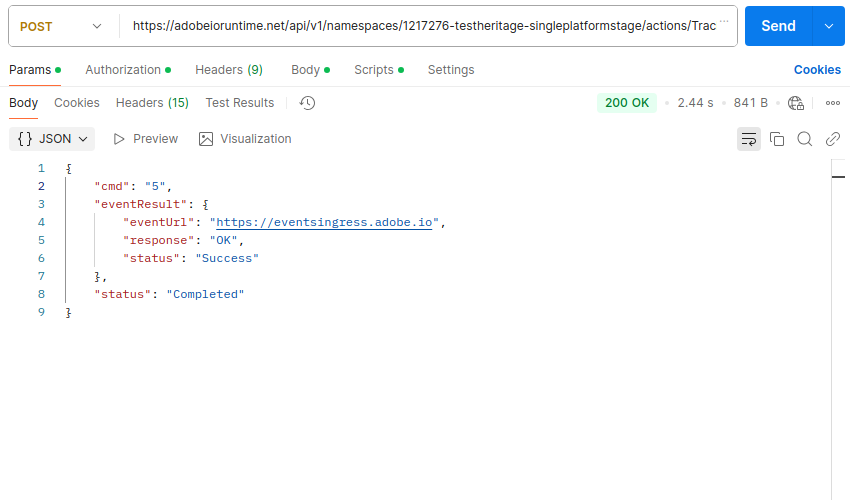
**Payload** (Raw JSON):

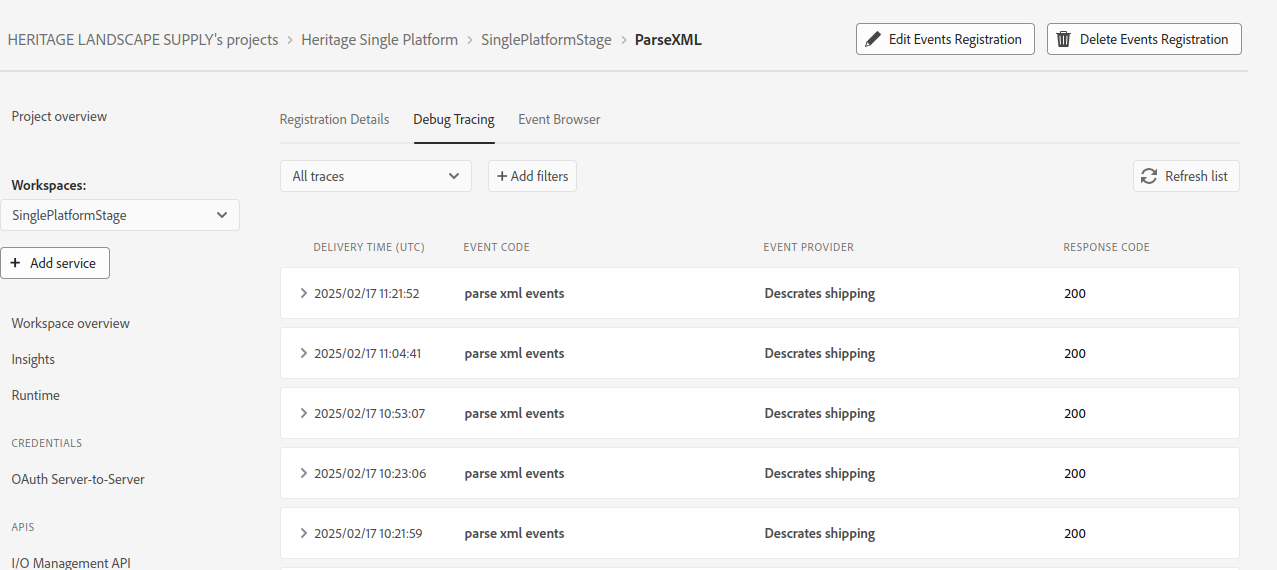
{

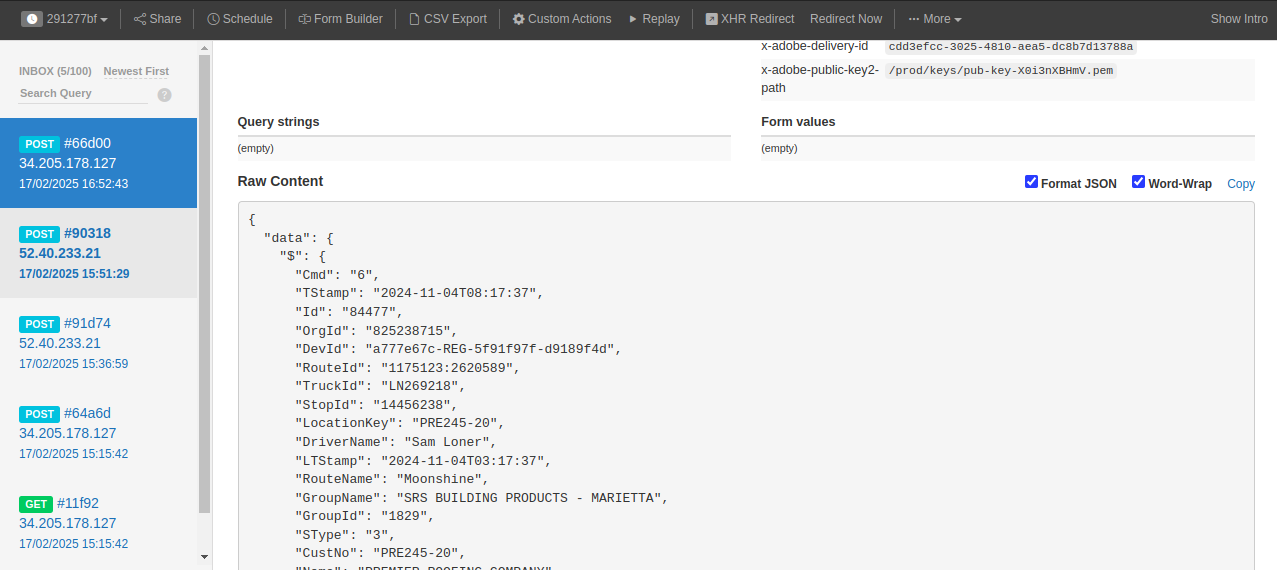
"\_\_ow\_body": "<?xml version='1.0' encoding='utf-8'?><MLW Cmd='6' TStamp='2024-11-04T08:17:37' Id='84477' OrgId='825238715' DevId='a777e67c-REG-5f91f97f-d9189f4d' RouteId='1175123:2620589' TruckId='LN269218' StopId='14456238' LocationKey='PRE245-20' DriverName='Sam Loner' LTStamp='2024-11-04T03:17:37' RouteName='Moonshine' GroupName='SRS BUILDING PRODUCTS - MARIETTA' GroupId='1829' SType='3' CustNo='PRE245-20' Name='PREMIER ROOFING COMPANY' LocationAddress='55 SPIVEY LN' LocationCity='SWAINSBORO' LocationState='GA' LocationPostalCode='30401'><GPS Altitude='355.06502091418952' Latitude='33.981149902660228' Longitude='-84.5433675785606' NoOfSat='7' Speed='0.098615773664189582' SatTStamp='2024-11-04T08:17:37' Direction='0' FixQuality='A' /><Job Id='29927687' JobId='29927687' JType='3' Status='2' Name='WHSE' DispId='0040422976-001'><Extension Name='SHIPVIA' Value='GROUND DROP' /><Extension Name='ORDERTYPE' Value='SO' /><Extension Name='DISPACTHID' Value='14212661' /><Extension Name='BRANCH' Value='0131' /><Extension Name='CUSTOMERPO' Value='2488801-1-ATL' /><Item ItemId='41790499' Status='1' Key='' Name='' LoadedOnCart='False' /></Job><Job Id='29927870' JobId='29927870' JType='3' Status='2' Name='WHSE' DispId='40422976'><Extension Name='SHIPVIA' Value='GROUND DROP' /><Extension Name='ORDERTYPE' Value='SO' /><Extension Name='DISPACTHID' Value='14212670' /><Extension Name='BRANCH' Value='0131' /><Extension Name='CUSTOMERPO' Value='2488801-1-ATL' /><Item ItemId='41790932' Status='1' Key='' Name='' LoadedOnCart='False' /></Job><Extension Name='STOPTYPE' Value='Customer' /><Extension Name='CUSTOMERTYPE' Value='GROUND DROP' /><Extension Name='SHIPCODE' Value='4' /><Extension Name='SHIPCODE' Value='4' /><Extension Name='ADDRESS2' Value='WESLEY DEFREITAS' /></MLW>"

}

**Response**:  
The response will include the processed event data.

****





### **Conclusion**

This document provides the full process for creating an Adobe I/O Runtime action that triggers multiple events. Ensure you follow each step carefully and test the system to confirm that the events are triggered correctly based on the cmd parameter from the XML.