**Salesforce Data Cloud Functionalities**

**Data Cloud Streaming Ingestion API**

**Process Document**

**Overview:**

Ingestion API is a REST API and offers two interaction patterns: bulk and streaming.

The streaming pattern accepts incremental updates to a dataset as those changes are captured, while the bulk pattern accepts CSV files in cases where data syncs occur periodically.

The same data stream can accept data from the streaming and the bulk interaction.

**Steps to implement ingestion API through postman:**

1. **Create connected App on salesforce for authentication purpose:**

* A connected app needs to be configured for a developer to send data into Data Cloud using Ingestion API.
* A connected app is a framework that enables an external application to integrate with Salesforce using APIs and standard protocols, such as SAML, OAuth, and OpenID Connect.
* Connected apps use these protocols to authenticate, authorize, and provide single sign-on (SSO) for external apps.

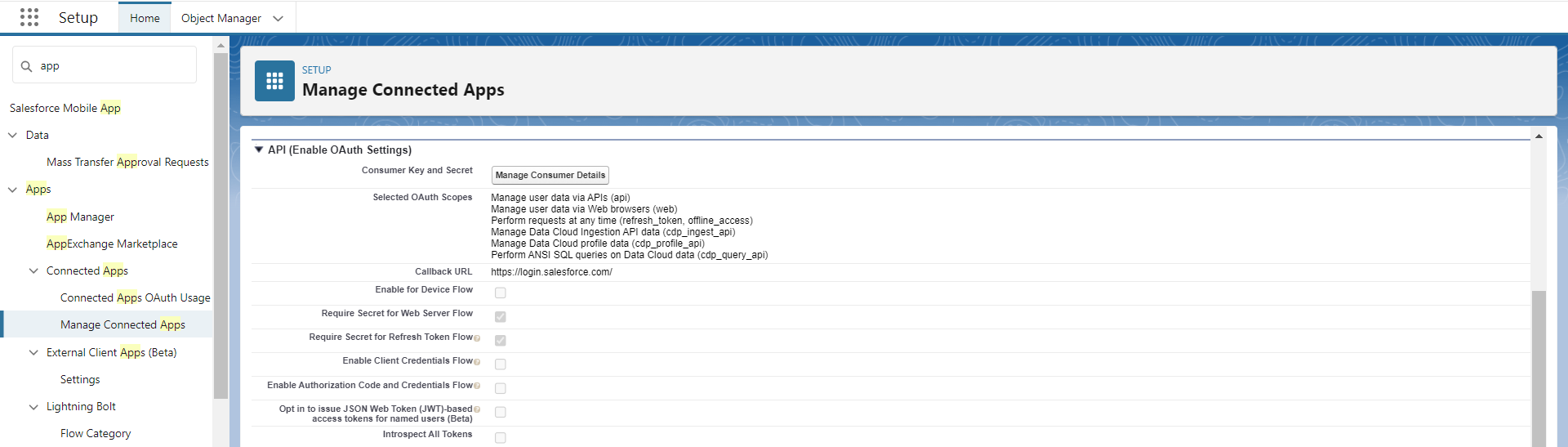
1. Go to Set up.
2. Search for App manager
3. Click on New Connected App
4. Give details like:

Connected App Name, Contact Email

1. Click on check box for Enable oauth settings
2. Give callback url as https://login.salesforce.com/
3. Select required oauth scopes like:

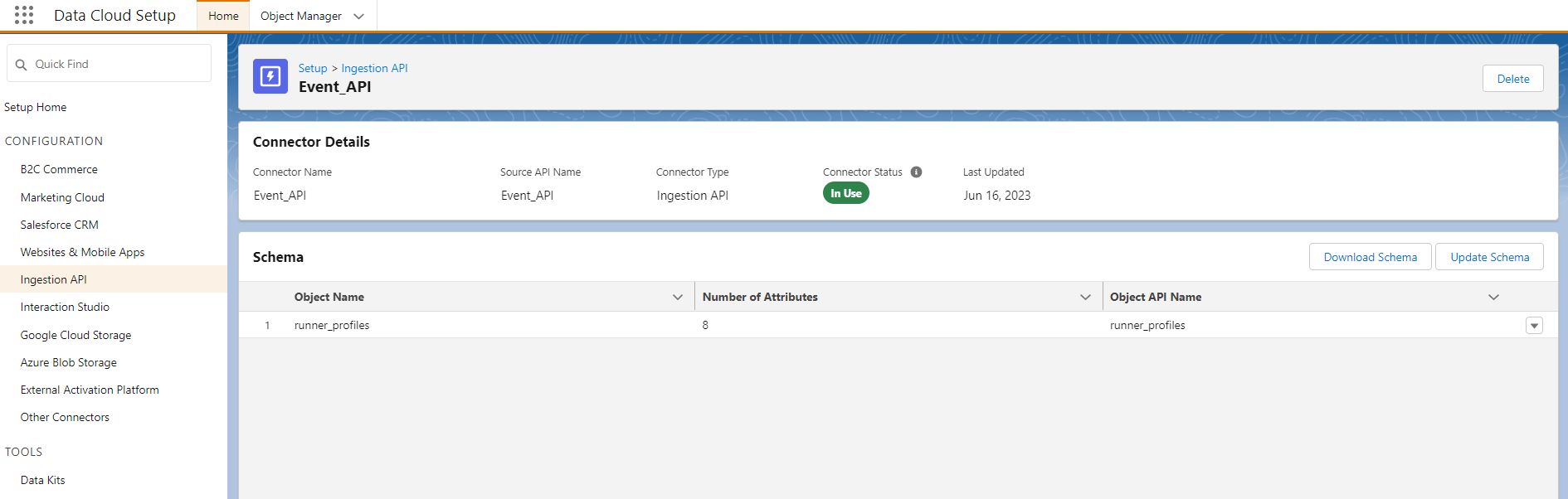
* Manage user data via APIs (api)
* Manage user data via Web browsers (web)
* Perform requests at any time (refresh\_token, offline\_access)
* Manage Data Cloud Ingestion API data (cdp\_ingest\_api)
* Manage Data Cloud profile data (cdp\_profile\_api)
* Perform ANSI SQL queries on Data Cloud data (cdp\_query\_api)

1. Check on Use Digital Signatures and upload **self signed digital certificate** obtained either from Open SSl or any certification Authority for authorization purpose.
2. Click Save.
3. Open Manage Connected Apps in the Setup and search further for the created connected app.
4. Click on Edit.
5. Under OAuth policies select “ All users may self-authorize” for permitted users and select “Relax IP restrictions” for IP relaxation.



1. **Create Ingestion API connector on Data Cloud:**
2. Go to Data Cloud Set Up
3. Click on Ingestion API
4. Click New
5. Give Connector Name
6. Upload JSON Schema to define the structure of data.(PFA sample **JSON file** in **yaml** format)





1. **Create Data Stream using Ingestion API connector**
2. Go to Data Streams Tab on Data cloud
3. Click new
4. Connect using Ingestion API connector
5. Use the configured connector
6. Give required details like Data Stream Name, Type of Data, Primary Key etc.
7. **Pushing Data from Postman using Streaming Ingestion API:**
8. To Set up the oauth 2.0 flow for getting access bearer token.
9. Create new collection on Postman
10. Under the collection Pass a post request to generate bearer access token using OAuth 2.0 JWT bearer token authorization.

* **What Is JWT Bearer Flow?**

1. Secure server-to-server integration without real time user involvement.
2. Client specifies user in a JSON web token (JWT) or [SAML](https://www.apexhours.com/salesforce-single-sign-on-flows-identity-flows/) format XML assertion and proves its own identity by appending a signature.
3. JWT Bearer token flow is Ideal for application which access SFDC only through API as there is no UI involved.
4. This flow uses a certificate to sign the JWT request and doesn’t require explicit user interaction.
5. However, this flow does require prior approval of the client app.

* **JWT Structure**

JSON Web Token consists of 3 parts:

1. Headers – Which contains the algorithm which will be used to sign the request {"alg":"RS256"}
2. Payload – This contains claims information which is an object containing information about user and additional data. Claims are set using parameters- {"Iss,aud,sub,exp"}
3. Signature – Signature consists of 3 parts and the structure is given below

<headerbase64encodedurl>.<claimsbase64encodedclaims>.<signature(uses algorithm like RS 256)>

* **How to Create JWT assertion :**

1. Firstly, a self signed digital certificate has to be generated along with the Private key either using open SSl or through any certifying authority online (eg. Getacert.com)
2. Once generated and provided with .crt file and private key we have to generate JWT assertion.
3. .crt file or the certificate needs to be uploaded to connected app in under use Digital signature.
4. For creating JWT assertion , we can use online website jwt.io to fill up the details and verify the signature.
5. At jwt.io, under algorithm select RS256.
6. Give the header details as follows:

{

"alg": "RS256",

"typ": "JWT"

}

1. Give the payload details as follows:

{

"iss": "<Consumer Key to be obtained from connected app>",

"sub": "<org user name>",

"aud": "<org URL, eg, https://login.salesforce.com>",

"exp": "<current timestamp, can be obtained online from sites like unixtimestamp.com>

}

Where,

* **Iss** is the issuer which must contain the OAuth client\_id or the connected app for which you registered the certificate.
* **Aud** is the audience which identifies the authorization server as an intended audience. The authorization server must verify that it’s an intended audience for the token.
* **Sub** is the subject. If you’re implementing this flow for an Experience Cloud site, the subject must contain the user’s username.
* **Exp** is the date and time at which the token expires, expressed as the number of seconds from 1970-01-01T0:0:0Z measured in UTC. Salesforce allows a 3-minute buffer for clock skew.

1. Verify signature:

* Enter Certificate Details.
* Enter private Key Details.
* Once we see Signature Verified, we can use the encoded JWT assertion created.
* **Get Access Token:**

POST <https://login.salesforce.com/services/oauth2/token>

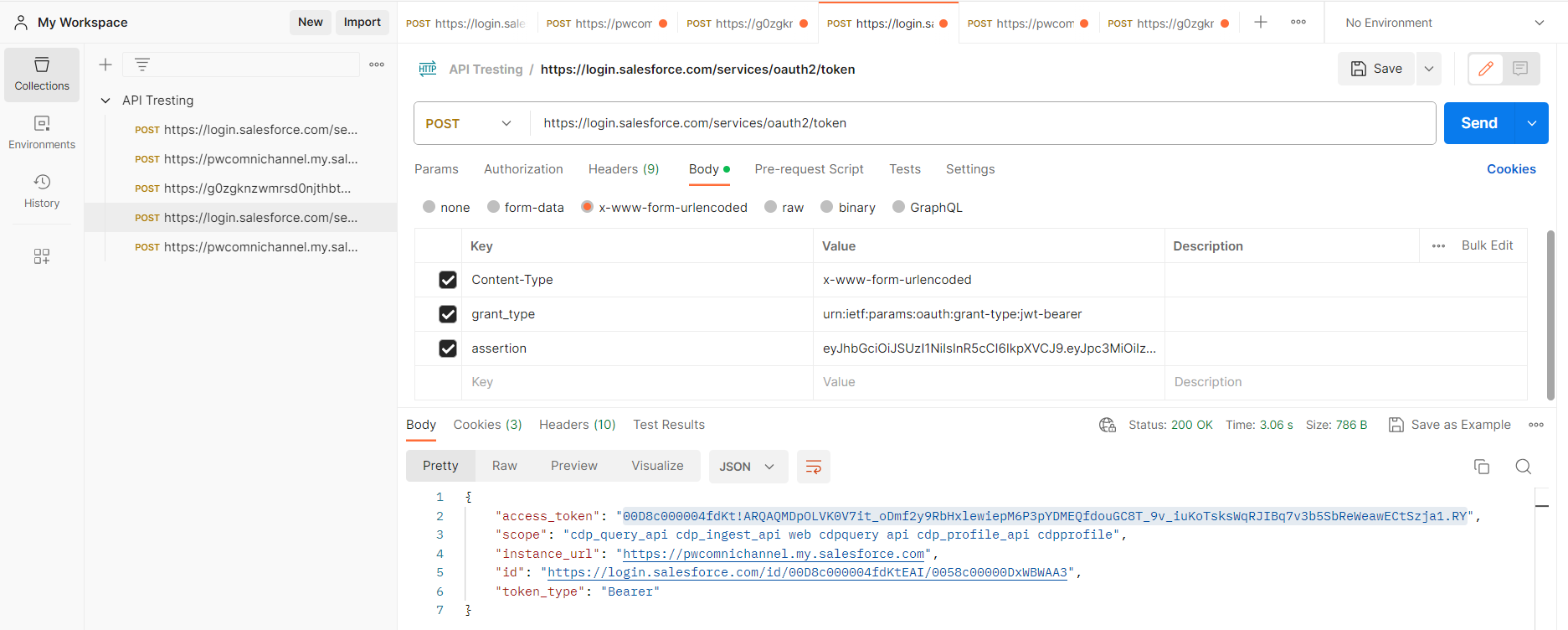
Under Body, use x-www-form-urlencoded

Give Key and value as follows:

* Content-Type = x-www-form-urlencoded
* grant\_type=urn:ietf:params:oauth:grant-type:jwt-bearer
* assertion= <obtained from above step>

Click on Send.

Response will return Access token and instance URL



* **Exchange access token with Data Cloud:**

Run following Post request on Postman:

POST <Instance URL>/services/a360/token

Content-Type : x-www-form-urlencoded

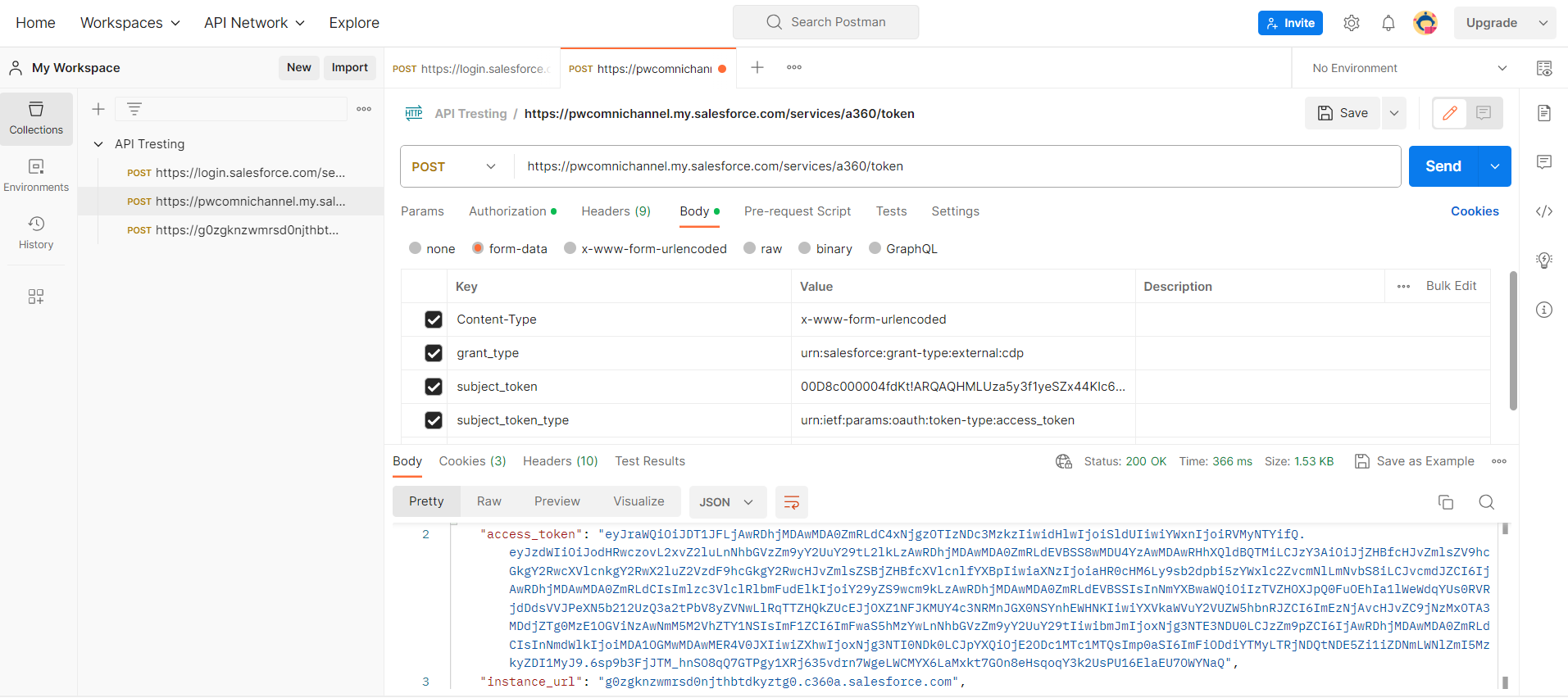
grant\_type=urn:salesforce:grant-type:external:cdp

&subject\_token=<CORE ACCESS TOKEN>

&subject\_token\_type=urn:ietf:params:oauth:token-type:access\_token

Where,

* CORE ACCESS TOKEN is access token obtained from previous request and Instance URL obtained from previous request.
* (Content-Type, grant\_type etc, will come under Params)
* Response will now return bearer access token for sending data through authorization request and Instance URL returned here will be used as Data cloud instance endpoint for sending data in next Post request.



* **Run post request to send data to Data cloud using Ingestion API:**

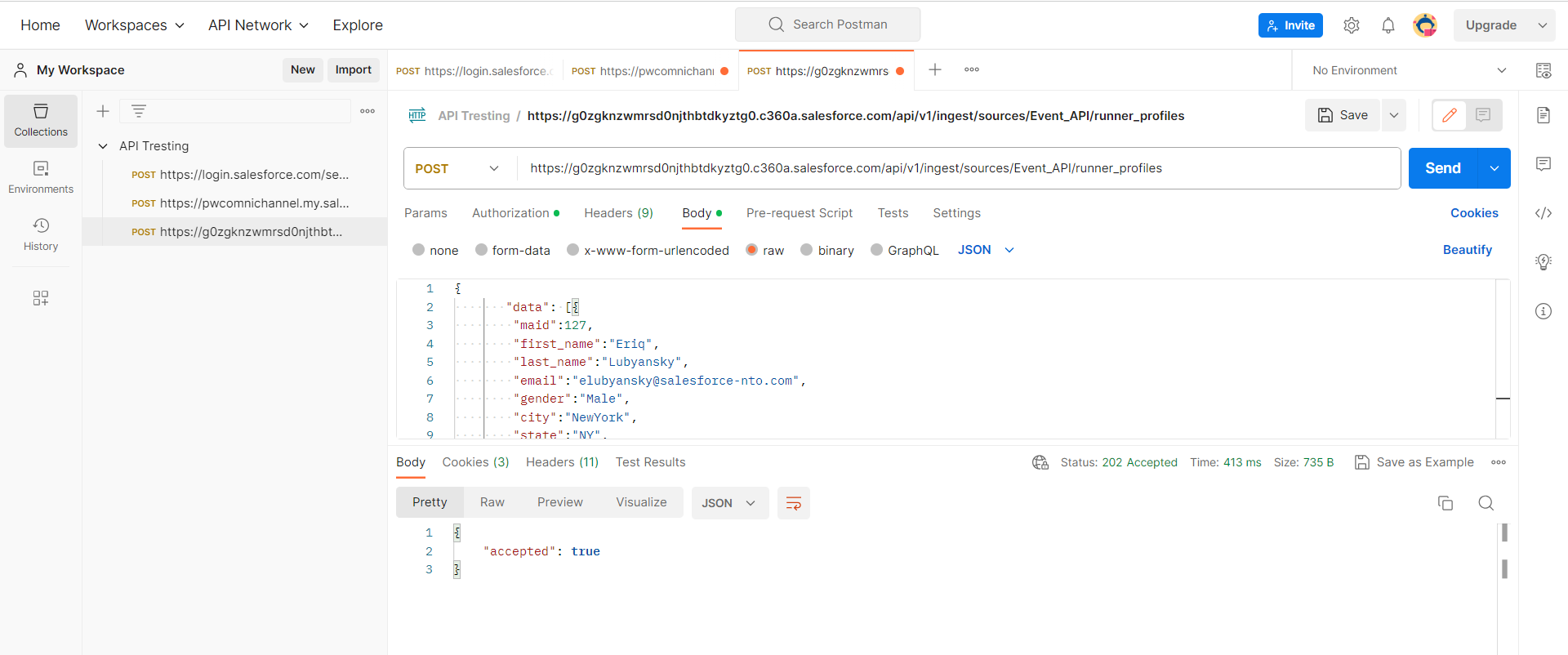
POST https://{{dne\_cdpOffcoreUrl}}/api/v1/ingest/sources/{name}/{object-name}

Eg: <https://g0zgknzwmrsd0njthbtdkyztg0.c360a.salesforce.com/api/v1/ingest/sources/Event_API/runner_profiles>

Where,

* {{dne\_cdpOffcoreUrl}} is CDP URL returned as instance URL in the response from Post request for Exchanging the tokens.
* {name} is ingestion API connector name create don data cloud
* {object-name} is object name into which data needs to be pushed in CDP.
* Make sure to enter access code as oauth bearer token under Authorization header obtained from previous Post request.
* Under Body header give the values of the data to be pushed in JSON format.

**NOTE:** records are pushed to CDP within 15-20 minutes.



**Reference Links:**

<https://developer.salesforce.com/docs/atlas.en-us.c360a_api.meta/c360a_api/c360a_getting_started_with_cdp.htm>

<https://developer.salesforce.com/docs/atlas.en-us.c360a_api.meta/c360a_api/c360a_api_insert_records.htm>

<https://help.salesforce.com/s/articleView?id=sf.remoteaccess_oauth_jwt_flow.htm&type=5>

<https://developer.salesforce.com/docs/atlas.en-us.sfdx_dev.meta/sfdx_dev/sfdx_dev_auth_key_and_cert.htm>

<https://www.apexhours.com/salesforce-oauth-2-0-jwt-bearer-flow/>