



TGH

Making Integrations Simpler



Creating API Plan using GraphQL Mutation

Author
Sowjanya Gurrala



Contents

Mutations in GraphQL API..... 2

What is a GraphQL API? 2

Mutations:..... 2

Syntax: 2

Input Variables: 2

Response: 2

Error Handling: 3

GraphQL API Management APIs: 3

Enabling API Metrics Data Collection..... 4

API Explorer: 5

Steps to create Plan..... 7

Conclusion: 12

References:..... 12

Mutations in GraphQL API

In this blog, you will learn how mutations of GraphQL APIs will help us to modify the server-side data in API Management. In order to showcase the concept of mutations we will create one API Plan using GraphQL mutation.

What is a GraphQL API?

- GraphQL is an open-source data query and manipulation language for APIs, as well as a runtime for fulfilling queries with existing data.
- GraphQL APIs are built based on GraphQL code.
- GraphQL is designed to address the issues of over-fetching and under-fetching of data, which are common challenges associated with traditional REST APIs.
- GraphQL allows you to make multiple resource requests in a single query call. Saves time and bandwidth by reducing the number of requests to the server.

Mutations:

- In GraphQL, mutations are operations used to modify or manipulate data on the server.
- While queries are used for reading the data, mutations are used for creating, updating or deleting data.
- Mutations allow clients to request changes to the data and receive a response to confirm the success or failure of those changes.

Here are some key components of mutations in GraphQL

Syntax:

Mutations have a syntax similar to queries but are defined using the mutation keyword. They include a set of fields that represent the data to be modified.

Input Variables:

Mutations often include input variables to pass data from the client to the server. This allows for dynamic and reusable mutation operations.

Response:

Like queries, mutations specify the fields they want in the response. The server returns the requested data after applying the mutation.

Error Handling:

GraphQL mutations include error handling. If a mutation encounters an error, the server returns an error object that includes details about what went wrong.

Overall mutations in GraphQL provide a flexible and efficient way to modify data on the server while allowing clients to precisely request the data they need in response to the mutation.

Now we will see the GraphQL API Management APIs in Boomi.

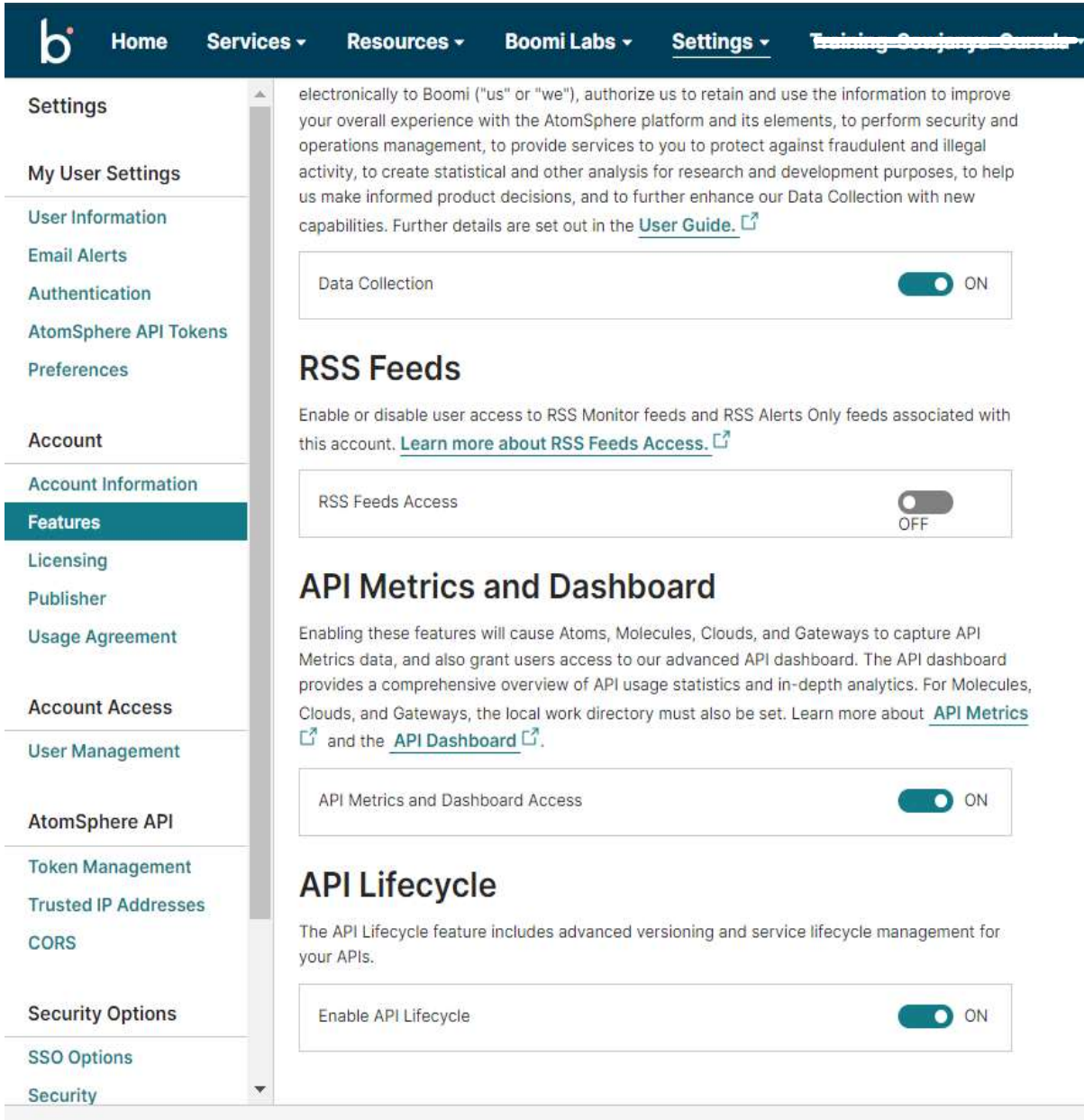
GraphQL API Management APIs:

- The Boomi API Management GraphQL APIs expose many GraphQL queries and mutations to retrieve and update data.
- It is a new implementation for API Management. You can make queries and mutations to the Gateway settings.
- The GraphQL API Management APIs provides programmatic access to functionality that is normally accessed through the API Management service.
- Programmatic access to GraphQL APIs is available through any applications like Altair, Postman, GraphQL Client.
- There is also a service provided in Boomi to execute GraphQL Queries and mutations i.e., API Explorer.
- To leverage the GraphQL APIs, we must first enable it for the API Gateway and runtime containers. When we enable API Metrics, it applies to all Gateways in your account.

Enabling API Metrics Data Collection

From the Boomi platform, go to **Settings>Account Information and setup**. From the left navigation select **Features**.

Switch the API Metrics and Dashboard Access to the ON position.



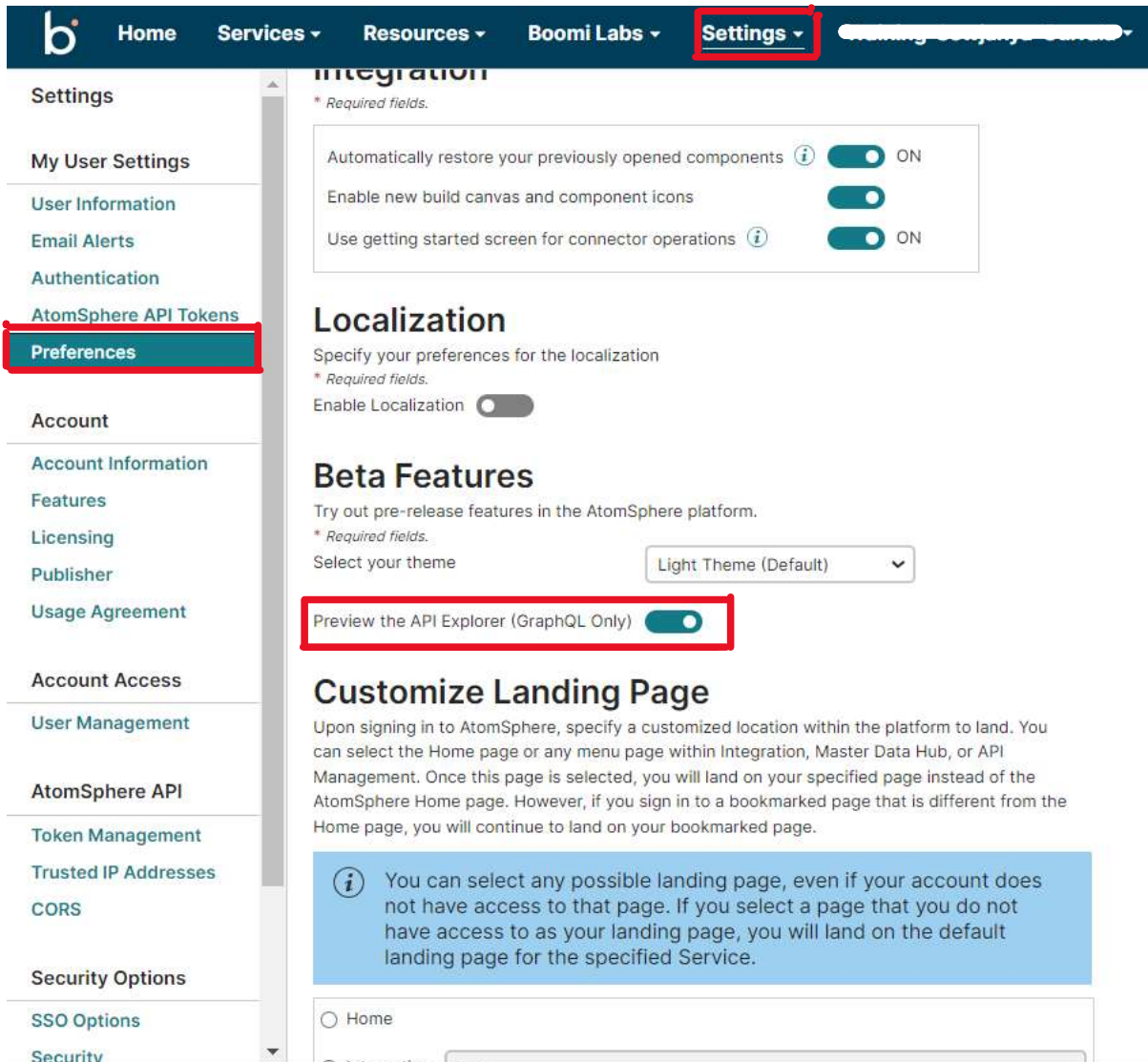
The screenshot shows the Boomi Settings interface. The left sidebar is expanded to the 'Features' section. The main content area displays the following settings:

- Data Collection:** Toggled ON.
- RSS Feeds:** Toggled OFF.
- API Metrics and Dashboard:** Toggled ON.
- API Lifecycle:** Toggled ON.



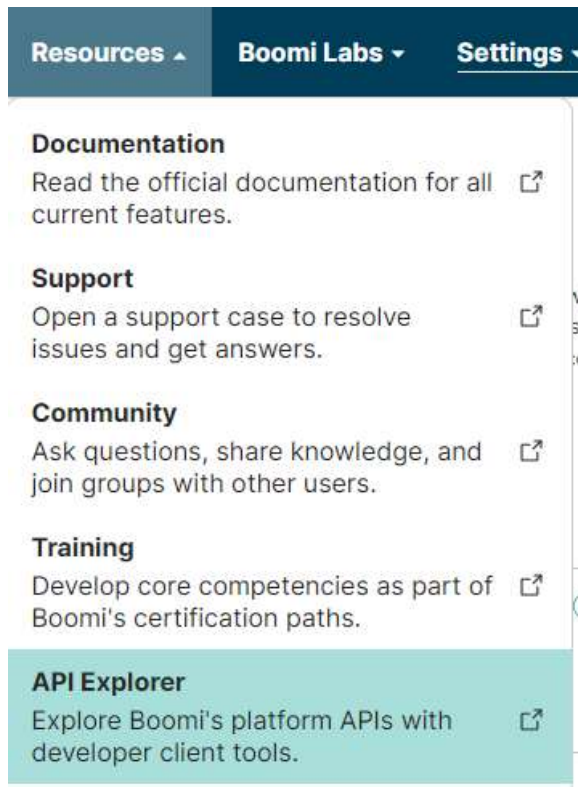
API Explorer:

- API Explorer is a resource provided by Boomi which allows us to interact with GraphQL API endpoints and execute GraphQL code i.e., queries and mutations within the platform instead of from third-party tools such as Postman and Altair.
- By default, API Explorer is disabled. We can enable it under **Settings > User Information > Preferences** tab.

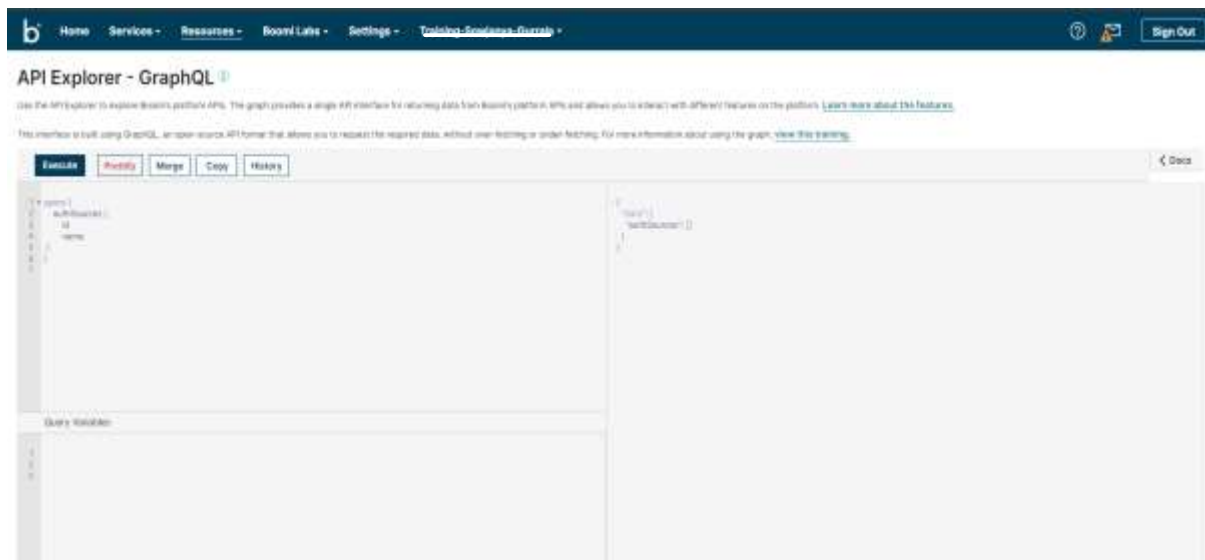


The screenshot shows the Boomi user settings interface. The top navigation bar includes 'Home', 'Services', 'Resources', 'Boomi Labs', and 'Settings'. The left sidebar lists various settings categories, with 'Preferences' highlighted in red. The main content area is divided into sections: 'Integration' (with a toggle for 'Automatically restore your previously opened components' set to ON), 'Localization' (with a toggle for 'Enable Localization' set to OFF), 'Beta Features' (with a dropdown for 'Select your theme' set to 'Light Theme (Default)' and a toggle for 'Preview the API Explorer (GraphQL Only)' set to ON, which is highlighted in red), and 'Customize Landing Page' (with a blue informational box stating that users can select any landing page, even if they don't have access to it, and will be redirected to the default landing page if they do not).

- After enabling, to launch API Explorer, select **Resources>API Explorer** from Boomi platform menu bar.



- The UI of API Explorer will look as shown below.



The API Explorer screen has different parts. From the top-left, you will see the Actions Buttons. Below these, you will find the Editor. From the Top-Right you will find the Docs button. Below is the Output pane. Finally, at the very bottom of the screen, you will see the Query Variables pane.

Now we will execute a mutation which will help us to create a plan in API Management.

Follow the below steps for the successful creation of plan using GraphQL mutation.

Steps to create Plan

Step 1:

Refer to the below link in Boomi documentation to understand the syntax of mutation which is used to create a Plan.

https://help.boomi.com/docs/Atomsphere/API%20Management/Topics/api-deployed_api_plan_category_d0596786-3535-44b7-9a20-366109708add

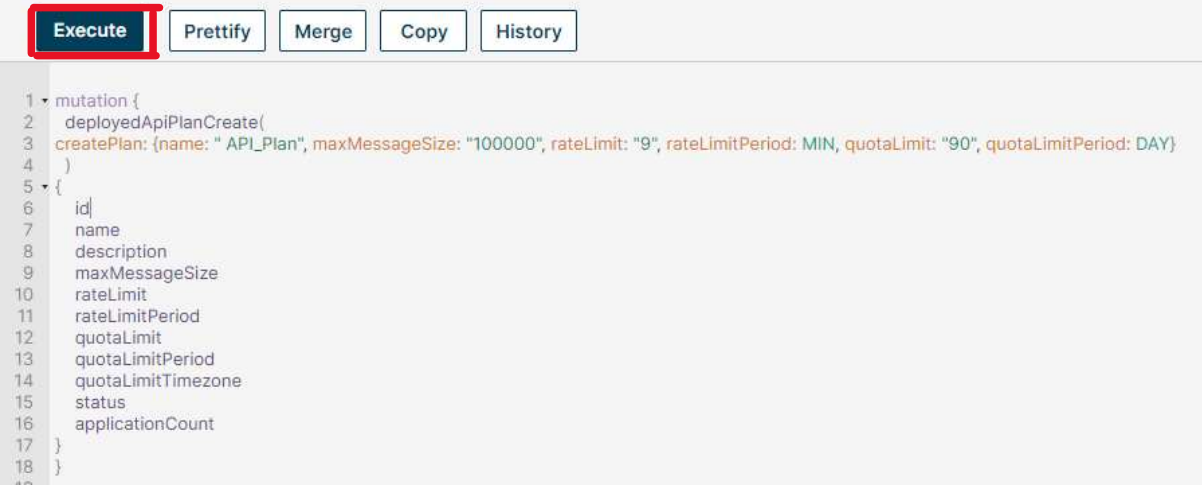
Step 2:

Create a mutation as shown below.

```
mutation {
  deployedApiPlanCreate(
    createPlan: { name: " API_Plan", maxMessageSize: "100000000000", rateLimit: "9",
rateLimitPeriod: MIN, quotaLimit: "90", quotaLimitPeriod: DAY}
  ){
    id
    name
    description
    maxMessageSize
    rateLimit
    rateLimitPeriod
    quotaLimit
    quotaLimitPeriod
    quotaLimitTimezone
    status
    applicationCount
  }
}
```


Step 3:


Copy and paste the query in the query pane and click on execute.



```
1 mutation {
2   deployedApiPlanCreate(
3     createPlan: {name: "API_Plan", maxMessageSize: "100000", rateLimit: "9", rateLimitPeriod: MIN, quotaLimit: "90", quotaLimitPeriod: DAY}
4   )
5   {
6     id
7     name
8     description
9     maxMessageSize
10    rateLimit
11    rateLimitPeriod
12    quotaLimit
13    quotaLimitPeriod
14    quotaLimitTimezone
15    status
16    applicationCount
17  }
18 }
```

Step 4:

The complete API Explorer will look as shown below. We can see the response in the output pane.



```
1 mutation {
2   deployedApiPlanCreate(
3     createPlan: {name: "API_Plan", maxMessageSize: "100000", rateLimit: "9", rateLimitPeriod: MIN, quotaLimit: "90", quotaLimitPeriod: DAY}
4   )
5   {
6     id
7     name
8     description
9     maxMessageSize
10    rateLimit
11    rateLimitPeriod
12    quotaLimit
13    quotaLimitPeriod
14    quotaLimitTimezone
15    status
16    applicationCount
17  }
18 }
```

```
{
  "data": {
    "deployedApiPlanCreate": {
      "id": "T84645-939-421-400-0246334647",
      "name": "API_Plan",
      "maxMessageSize": "100000",
      "rateLimit": "9",
      "rateLimitPeriod": "MIN",
      "quotaLimit": "90",
      "quotaLimitPeriod": "DAY",
      "quotaLimitTimezone": "UTC",
      "status": "ENABLED",
      "applicationCount": 1
    }
  }
}
```

Step 5:

This is the response we can see in the output pane. We can see the status of the executed query. Our plan was created with a unique id and the status as **“ENABLED”**. This represents that the query got executed successfully.

```
- {
-   "data": {
-     "deployedApiPlanCreate": {
-       "id": "18d6e345-909c-42d1-a700-35bc4057daa1",
-       "name": "API_Plan",
-       "description": null,
-       "maxMessageSize": 100000,
-       "rateLimit": 0,
-       "rateLimitPeriod": "MIN",
-       "quotaLimit": 90,
-       "quotaLimitPeriod": "DAY",
-       "quotaLimitTimezone": "UTC",
-       "status": "ENABLED",
-       "applicationCount": 0
-     }
-   }
- }
```

Now we have to confirm by checking in API Management > Configure Server > Plans

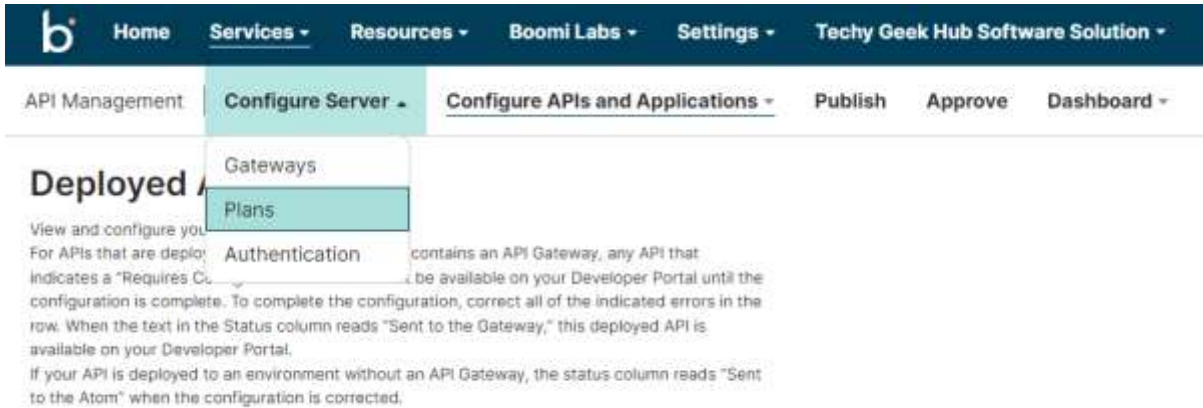
Step 6:

Go to Services and choose API Management.



Step 7:

Under Configure Server choose Plans.



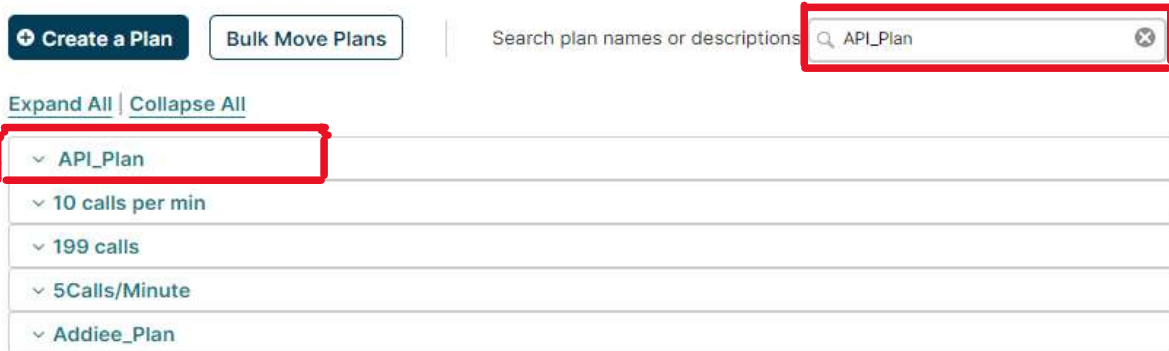
The screenshot shows the Techy Geek Hub software solution interface. The top navigation bar includes 'Home', 'Services', 'Resources', 'Boomi Labs', 'Settings', and 'Techy Geek Hub Software Solution'. Below this, a secondary navigation bar has 'API Management', 'Configure Server', 'Configure APIs and Applications', 'Publish', 'Approve', and 'Dashboard'. The 'Configure Server' menu is open, showing options for 'Gateways', 'Plans', and 'Authentication'. The 'Plans' option is highlighted. Below the navigation, there is a section titled 'Deployed' with a sub-header 'View and configure your APIs'. The text explains that for APIs with a 'Requires Configuration' status, they are not available on the Developer Portal until configuration is complete. It also notes that if an API is deployed to an environment without an API Gateway, the status column reads 'Sent to the Atom'.

Step 8:

Search for the plan which you have created just now. We can see that our “API_Plan” got created.

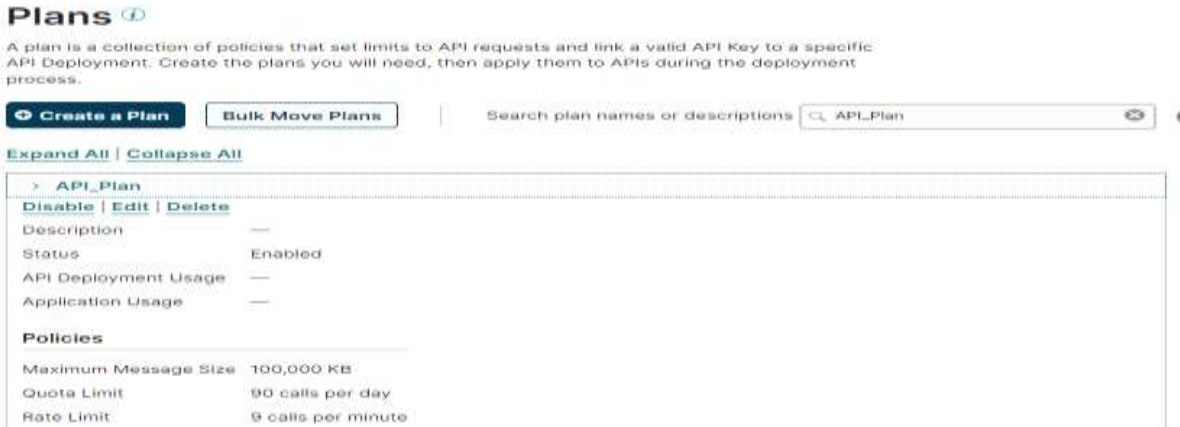
Plans ?

A plan is a collection of policies that set limits to API requests and link a valid API Key to a specific API Deployment. Create the plans you will need, then apply them to APIs during the deployment process.



The screenshot shows the 'Plans' page. At the top, there are buttons for 'Create a Plan' and 'Bulk Move Plans'. A search bar contains the text 'API_Plan'. Below the search bar, there are links for 'Expand All' and 'Collapse All'. A list of plans is displayed, with the first plan, 'API_Plan', highlighted by a red box. Other plans listed include '10 calls per min', '199 calls', '5Calls/Minute', and 'Addiee_Plan'.

Step 9: Expand the plan by clicking on the down arrow.

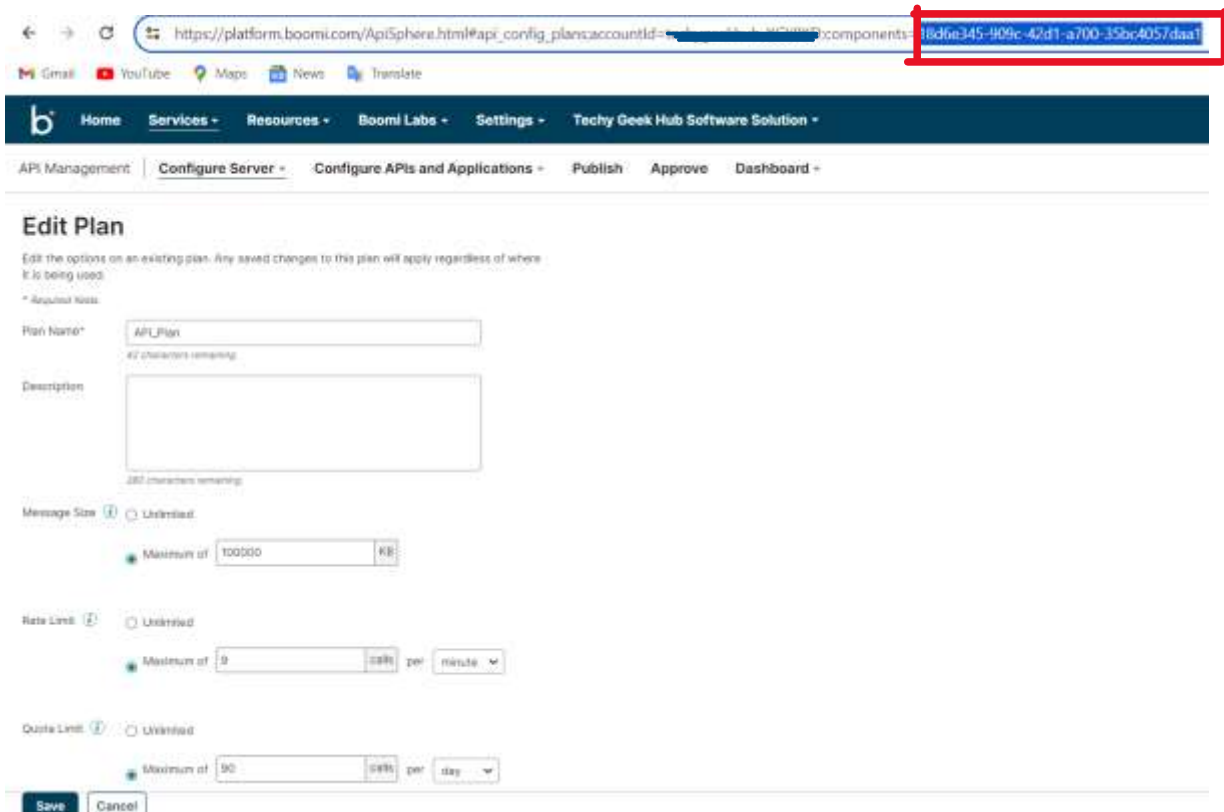


We can see all the plan details which we have configured in the query.

Step 10:

Click on Edit. We can see the plan Id in the browser search bar whose value is exactly same as the Id value which we got in the query response.

(For reference compare both the values)



©TGH Software Solutions Pvt. Ltd.

No part of this document may be copied, reproduced, republished, uploaded, posted, publicly displayed, encoded, translated, transmitted or distributed in any way to any other computer, server, website or other medium for publication or distribution, without TGH's prior written consent



Conclusion:

This is how we can create Plan using GraphQL mutation.

For more information, please find the below reference links.

References:

- <https://help.boomi.com/>
- https://help.boomi.com/docs/atomsphere/api%20management/topics/api-graphql_api_management_apis_18f1a55a-b3d7-4b9e-ab0a-162fc4a67686/
- https://help.boomi.com/docs/Atomsphere/API%20Management/Topics/api-deployed_api_plan_category_d0596786-3535-44b7-9a20-366109708add
- <https://graphql.org/>
- https://help.boomi.com/docs/Atomsphere/API%20Management/Topics/api-deployed_api_plan_category_d0596786-3535-44b7-9a20-366109708add#mutation-syntax



TGH

Making Integrations Simpler

boomi
Partner



TGH Software Solutions Pvt. Ltd.

www.techygeekhub.com

At TGH, we specialize in driving digital transformation through seamless Integration Technologies.

Operating as an INTEGRATION FACTORY, we serve as a one-stop shop for all your integration needs. Our expert team is well-versed in enterprise software and legacy system integration, along with leading iPaaS technologies like Boomi, MuleSoft, Workato, OIC, and more.

We're committed to enhancing business processes and solving problems through our integration expertise.



Email address

connect@techygeekhub.com



Phone number

+ 011-40071137
+ 91-8810610395



Our offices

Noida Office

iThum
Plot No -40, Tower A,
Office No: 712,
Sector-62, Noida,
Uttar Pradesh, 201301

Hyderabad Office

Plot no: 6/3, 5th Floor,
Techno Pearl Building,
HUDA Techno Enclave,
HITEC City, Hyderabad,
Telangana 500081

