

Enin API - Getting Started

PDF Version ([getting_started.pdf](#))

Available APIs

The following API's are operated by Enin:

- **Analysis API** : Access to Enin data and analyses for direct integration into your business applications.
- **Datasets API** : For bulk download of Enin data and analyses.
 - Read the Datasets API Tutorial ([datasets_tutorial](#)) ([pdf \(datasets_tutorial/datasets_tutorial.pdf\)](#)) for a detailed introduction with examples.

Swagger Documentation

To access our swagger documentation you need one of the following:

- Bearer token from Auth0 using their Machine-to-Machine interface
 - See section API Authentication for how to generate a token.
- Basic HTTP authentication credentials

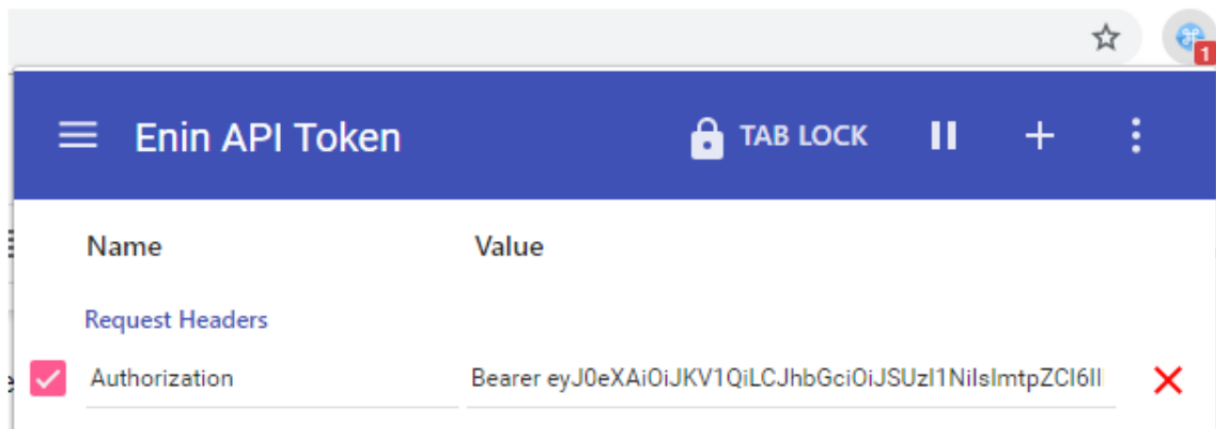
What authentication will work depends on what API you want to access:

- **Analysis API**
 - API root/docs: <https://api.enin.ai/analysis>
 - Authentication: App/M2M Token or Basic Auth
- **Datasets API**
 - API root/docs: <https://api.enin.ai/datasets>
 - Authentication: M2M Token or Basic Auth

Access Swagger with Token Authentication

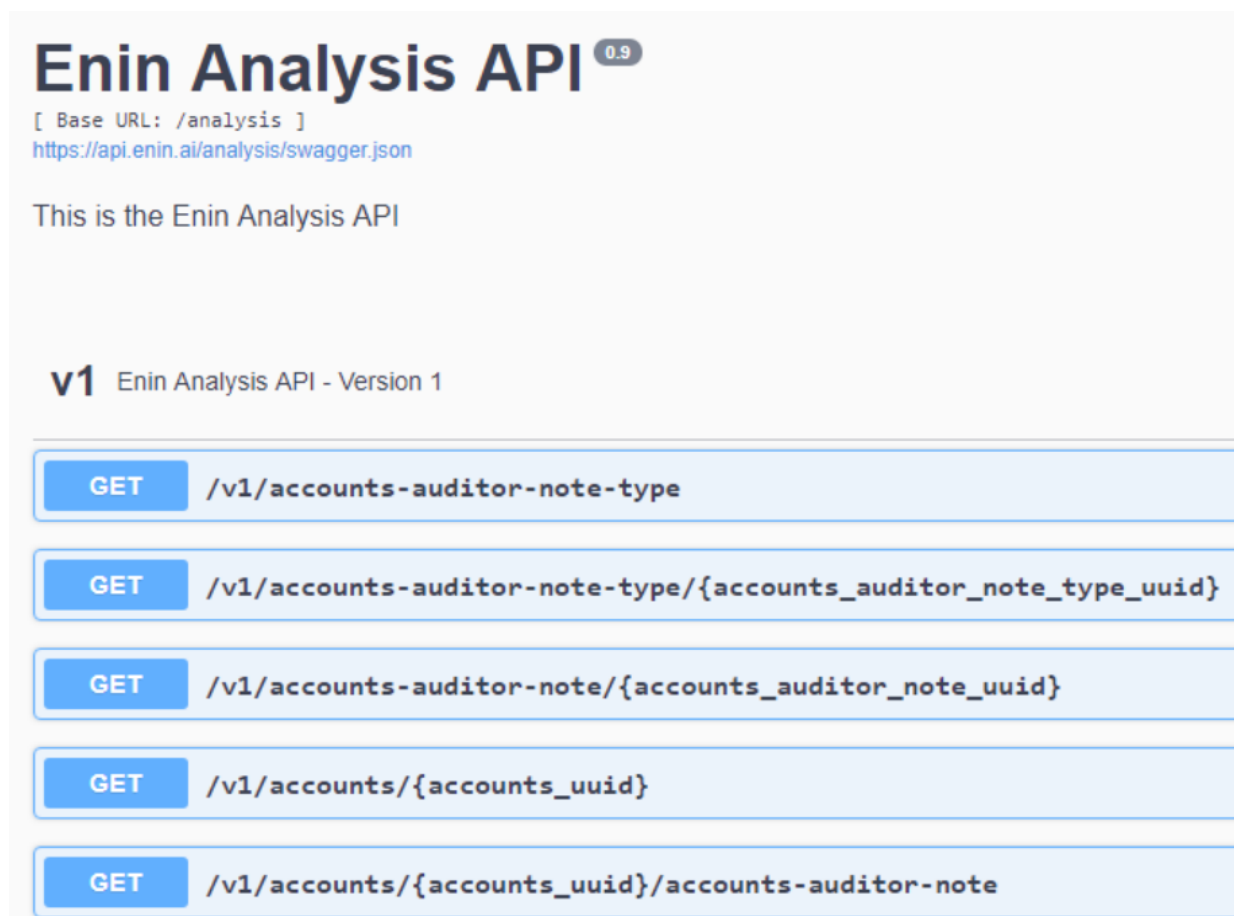
To use the swagger documentation with token based authentication you need to generate a token from Auth0. See the Token Authentication section for how to generate a Machine-to-Machine token.

When you have a token ready you should now use a tool to inject an HTTP "Authorization" header with "Bearer YOUR_TOKEN". You can use something like ModHeader (<https://chrome.google.com/webstore/detail/modheader/idgpnmonknjnojddfkgklijpfnfklij?hl=en>) or Postman to do this. If you choose to use ModHeader you will enter the token as follows:



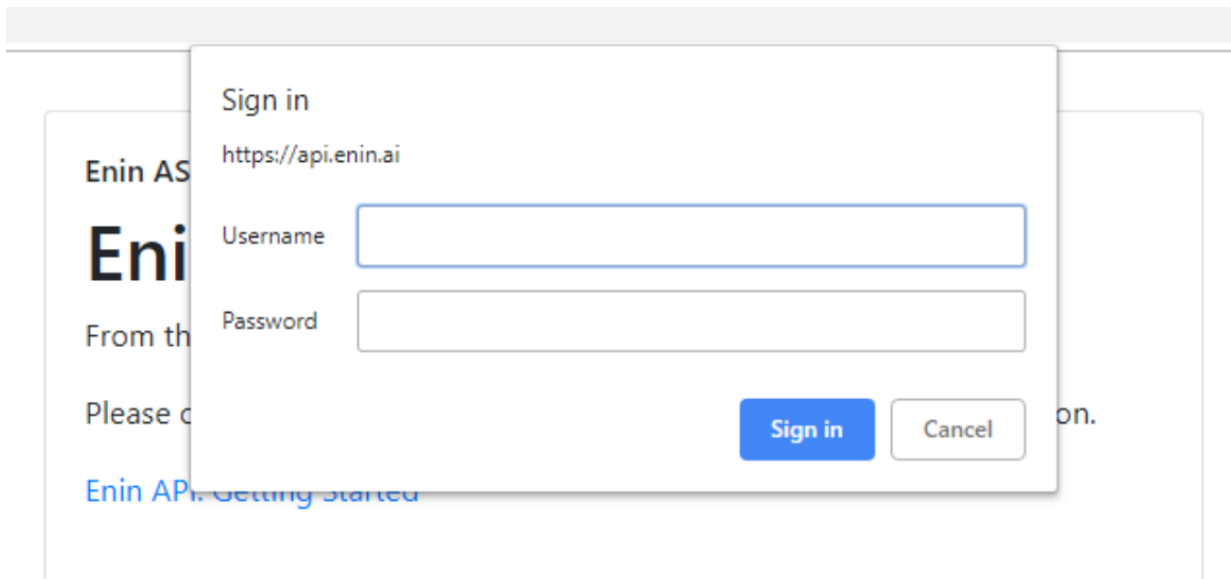
Consider locking to a particular tab you are working on. Because this setting will mess with any other authentication online (like google or facebook) if you forget to turn it off.

You should now have access to the swagger documentation. It should look something like the following:



Access Swagger with Basic Authentication

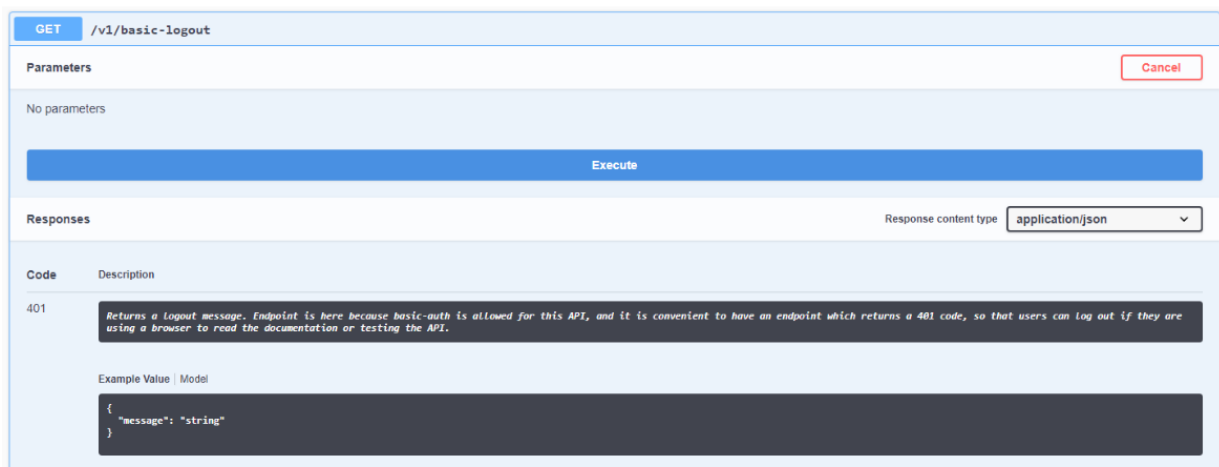
To use basic authentication to use the swagger documentation simply provide your Basic Auth Client ID and Client Secret as username and password in your browser:



To log out from Basic Authentication visit the appropriate `basic-logout` endpoint:

- <https://api.enin.ai/analysis/v1/basic-logout>
- <https://api.enin.ai/datasets/v1/basic-logout>

You can also simply run the endpoint in the swagger documentation:

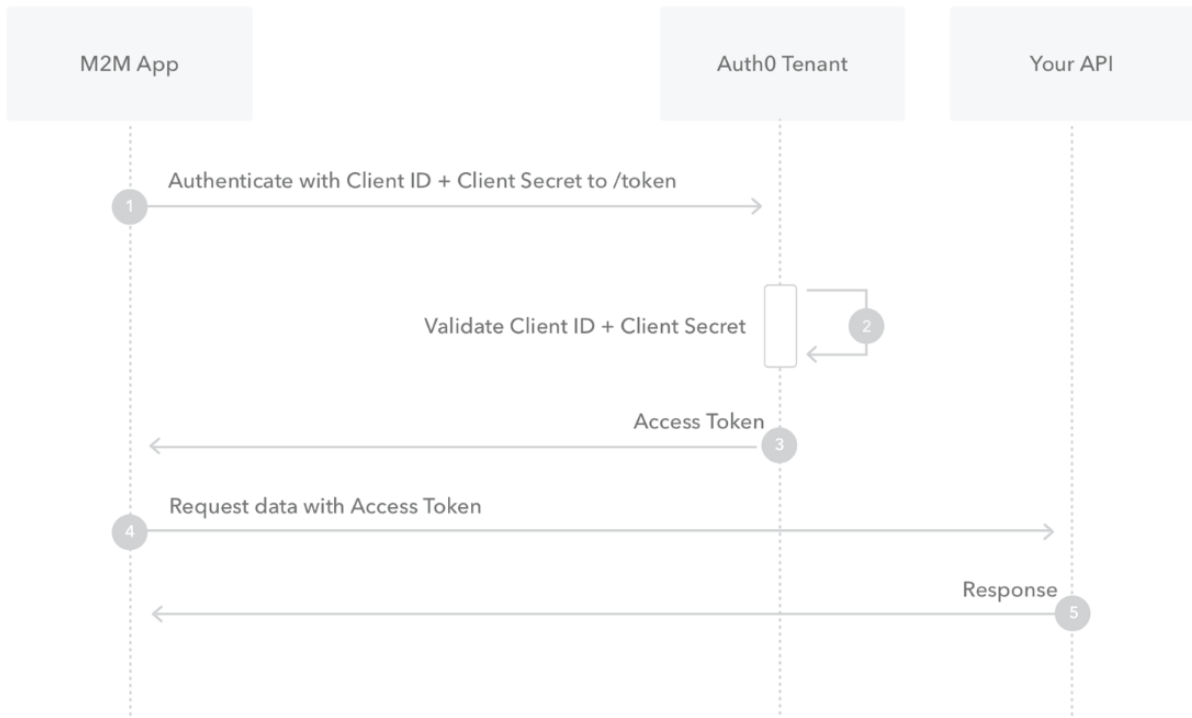


API Authentication

Each of Enin's APIs support either token based authentication only, or both token and basic authentication.

Token Authentication

Token authentication is recommended for production applications. It works by sending your Client ID and Client Secret to <https://login.enin.ai> using a HTTP POST request. You will get a token as a response which can be use to access <https://api.enin.ai>.



- How token authentication works (<https://auth0.com/docs/flows/concepts/m2m-flow>)
- Guide to retrieve and use token for API access (<https://auth0.com/docs/flows/guides/m2m-flow/call-api-using-m2m-flow>)

Python Example

```

import requests
from pprint import pprint

auth = requests.post(
    "https://login.enin.ai/oauth/token",
    headers={
        "content-type": "application/json",
    },
    json={
        "grant_type": "client_credentials",
        "client_id": "YOUR_TOKEN_CLIENT_ID",
        "client_secret": "YOUR_TOKEN_CLIENT_SECRET",
        "audience": "https://api.enin.ai/analysis",
    },
).json()

print("Auth Token")
pprint(auth)

# TODO: Store/cache access token for next call!
# Do not ask for a new token for every call to API.

data = requests.get(
    "https://api.enin.ai/analysis/v1/company/N0917540640",
    headers={
        "accept": "application/json",
        "authorization": "Bearer " + auth['access_token'],
    }
).json()

pprint(data)
  
```

R Example

```
# install.packages("httr")
require("httr")
# install.packages("jsonlite")
require("jsonlite")

auth <- POST(
  "https://login.enin.ai/oauth/token",
  body=list(
    grant_type = "client_credentials",
    client_id = "YOUR_TOKEN_CLIENT_ID",
    client_secret = "YOUR_TOKEN_CLIENT_SECRET",
    audience = "https://api.enin.ai/analysis"
  ),
  encode = "json"
)
auth <- content(auth, "text", encoding='utf8')
auth <- fromJSON(auth)

print("Auth Token")
print(auth)

# TODO: Store/cache access token for next call!
# Do not ask for a new token for every call to API.

data <- GET(
  "https://api.enin.ai/analysis/v1/company/N0917540640",
  add_headers(authorization = paste("Bearer", auth$access_token))
)
data <- content(data, "text", encoding='utf8')
data <- fromJSON(data)

print("\nEnin API Status")
print(data)
```

Basic Authentication

Basic authentication is available for quick integration, but discouraged in favor token authentication.

Python Example

```
import requests
from pprint import pprint

data = requests.get(
  "https://api.enin.ai/analysis/v1/company/N0917540640",
  headers={
    "accept": "application/json",
  },
  auth=(
    'YOUR_BASIC_CLIENT_ID',
    'YOUR_BASIC_CLIENT_SECRET',
  ),
).json()

print("Example Data")
pprint(data)
```

R Example

```
# install.packages("httr")
require("httr")
# install.packages("jsonlite")
require("jsonlite")

data <- GET(
  "https://api.enin.ai/analysis/v1/company/N0917540640",
  authenticate(
    'YOUR_BASIC_CLIENT_ID',
    'YOUR_BASIC_CLIENT_SECRET',
    type = "basic"
  )
)
data <- content(data, "text", encoding='utf8')
data <- fromJSON(data)

print("\nExample Data")
print(data)
```