

# Assignment Three

## Purpose

The purpose of this assignment is to protect the proxies using Basic and OAuth Authorization.

For this assignment, you will need to create two proxies. One will be used for the main flow of requests and verify the correct credentials have been sent in. This proxy will have two URNs, one for each authorization type. In order to accept OAuth access tokens, you will need to create an additional proxy to create and revoke these tokens.

If invalid credentials are sent in, the flow should stop and a useful error should be returned to the user. If the correct credentials are sent in, the proxy should make a callout to Github and return information for a given user. The user will be passed in as a query parameter or header, your choice. This request can be done through Apigee's policies/flows or through a Node.js server, similar to the one in assignment two.

## Requirements

- Create two proxies on Apigee, one to accept the request that is routed to Github and another to create and revoke OAuth licenses.
  - The primary proxy should only accept GET requests.
  - The proxy should have two different URNs that only accept a given authorization credential.
    - \* `/oauth` should only accept OAuth Authorization access tokens.
    - \* `/basic` should only accept Basic Authorization headers.
- The response from Github should be some bit of information about a user, it can be their repos, name, followers, etc. There's a wide range to choose from, we're just looking for a successful response from Github.

The response from Github doesn't need to be manipulated in any way, just pass back to the user.
- If the user sends in an invalid authorization, return a helpful error message. Avoid using the default error response from Apigee.
- Include a small SoapUI project that can show all the requirements have been met.
  - This project should include valid requests, as well as requests that fail (sending an invalid Basic Authorization header, etc.).
- Answer the following question:
  - Basic Authorization is easily handled in the main flow, without the need for an additional proxy. Give one reason (there are many) OAuth tokens should not be granted in the main flow, assuming the user has sent in the correct credentials.

## Resources

- <https://developer.github.com/v3/>
  - Github API documentation. You will need to set up a free Github account to utilize their API.
- <http://apigee.com/docs/api-services/tutorials/create-your-first-api>
  - Tutorial from Apigee on creating an API. Useful if you don't wish to use Node.js as the target endpoint.
- <http://apigee.com/docs/api-services/tutorials/secure-calls-your-api-through-oauth-20-client>
  - Tutorial from Apigee on creating an OAuth proxy.