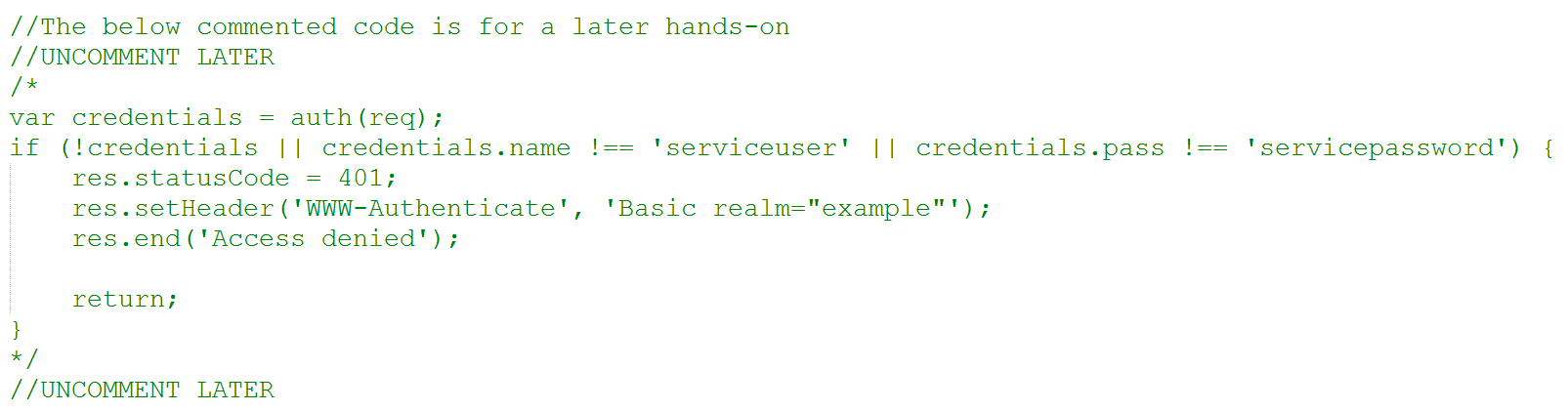
# Securing a Backend Service

## Configuring Basic Authentication

1. Open the server.js file of Product Assets service and uncomment the code within “UNCOMMENT LATER” section.

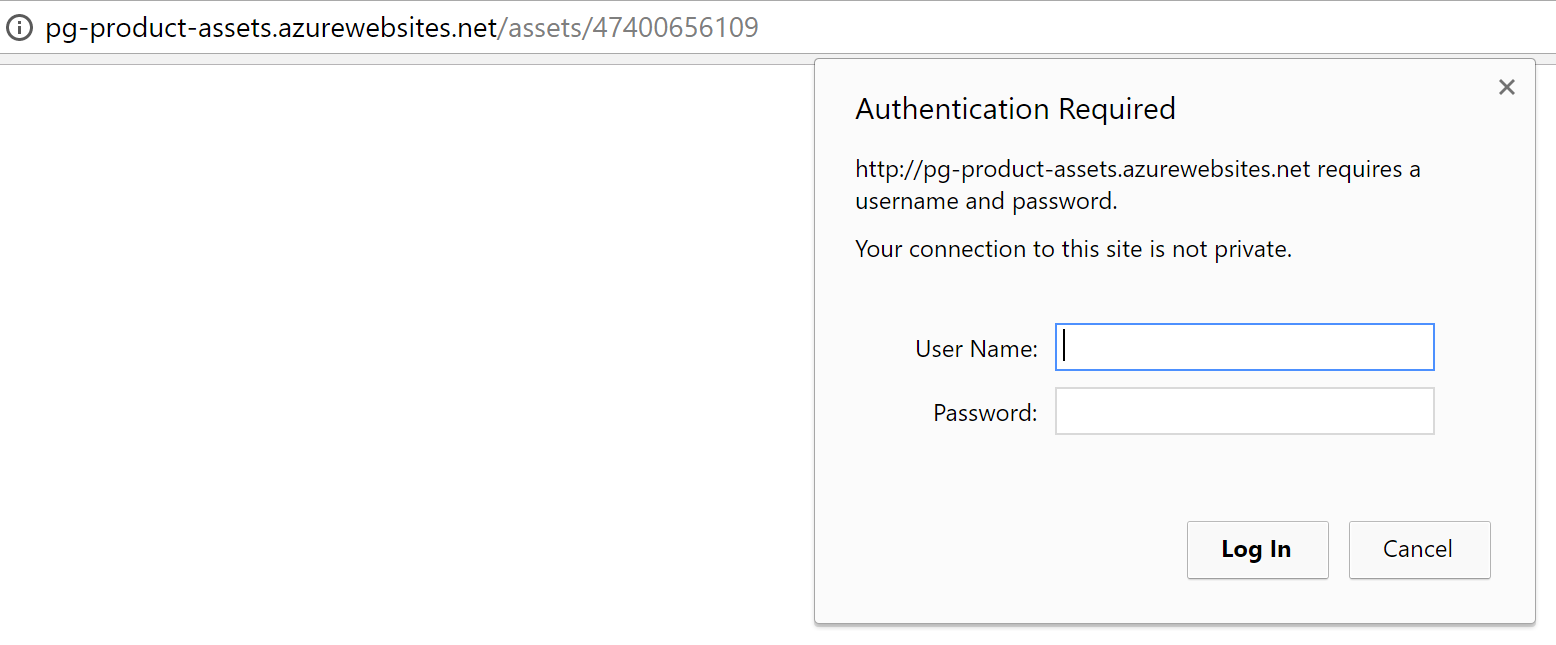


1. Understand the basic authentication code – not that the basic authentication username configured is “serviceuser” and password is “servicepassword”.
2. After the changes are done, run the following command to check-in the code.

**git commit -a -m “added basic auth security”**

**git push productassets master**

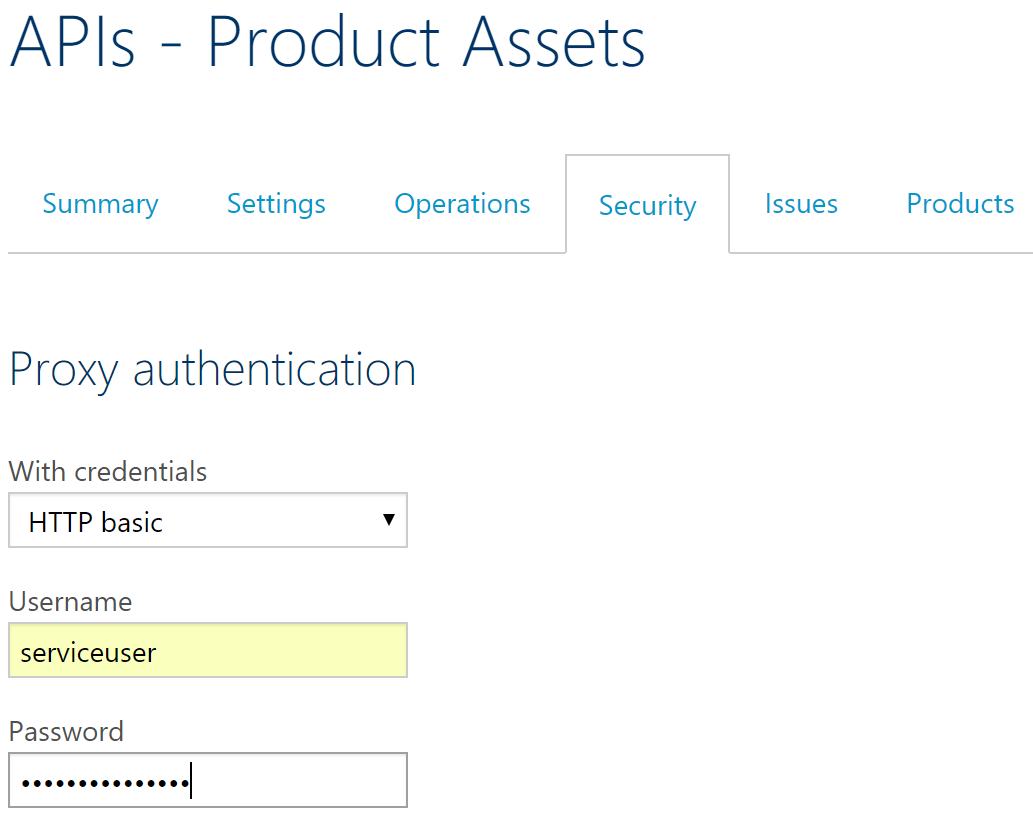
1. Wait for continuous deployment to Azure API App to complete
2. Access the Product Assets service – either the backend URL (<http://pg-product-assets.azurewebsites.net/assets/47400656109>) or the public URL configured in Azure API Management (<https://pgecommerce.azure-api.net/dam/assets/47400656109>). Note that there is an authentication dialog prompted now.



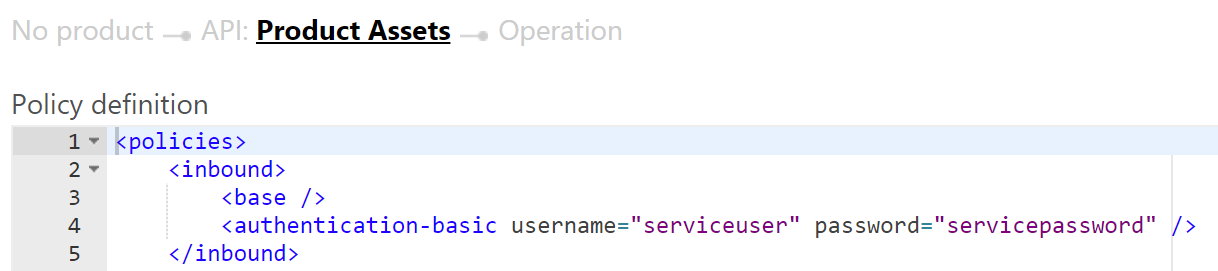
1. If credentials are not entered or incorrect credentials are entered, access to the service is denied.
2. Enter the right credentials (serviceuser / servicepassword) and the service returns results.

## Configuring Basic Authentication in API Management

1. Access the aggregated Products service (<https://pgecommerce.azure-api.net/products/47400656109>) and note that it throws a 404 error because of the error handling policy configured earlier.
2. Now go to Publisher Portal 🡪 APIs 🡪 Product Assets 🡪 Security and configure Proxy Authentication as shown below. This is for API Management to pass basic authentication credentials to the backend service.

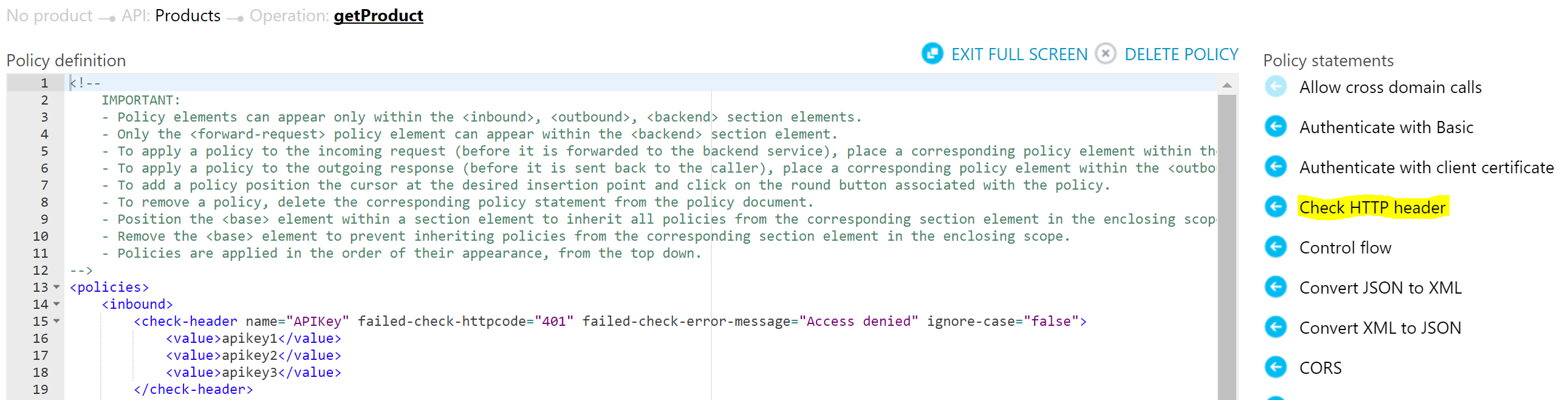


1. Access the aggregated Products service once again and note that it returns results without any prompt for credentials.
2. Also note that the above configuration as automatically added the “authentication-basic” policy to the inbound section of the Product Assets service.



# Securing A Service with API Key

1. Go to the policy editor of the **getProduct** operation of Products API.
2. Add the “Check HTTP Header” policy to the inbound section of the API and configure it as shown below.



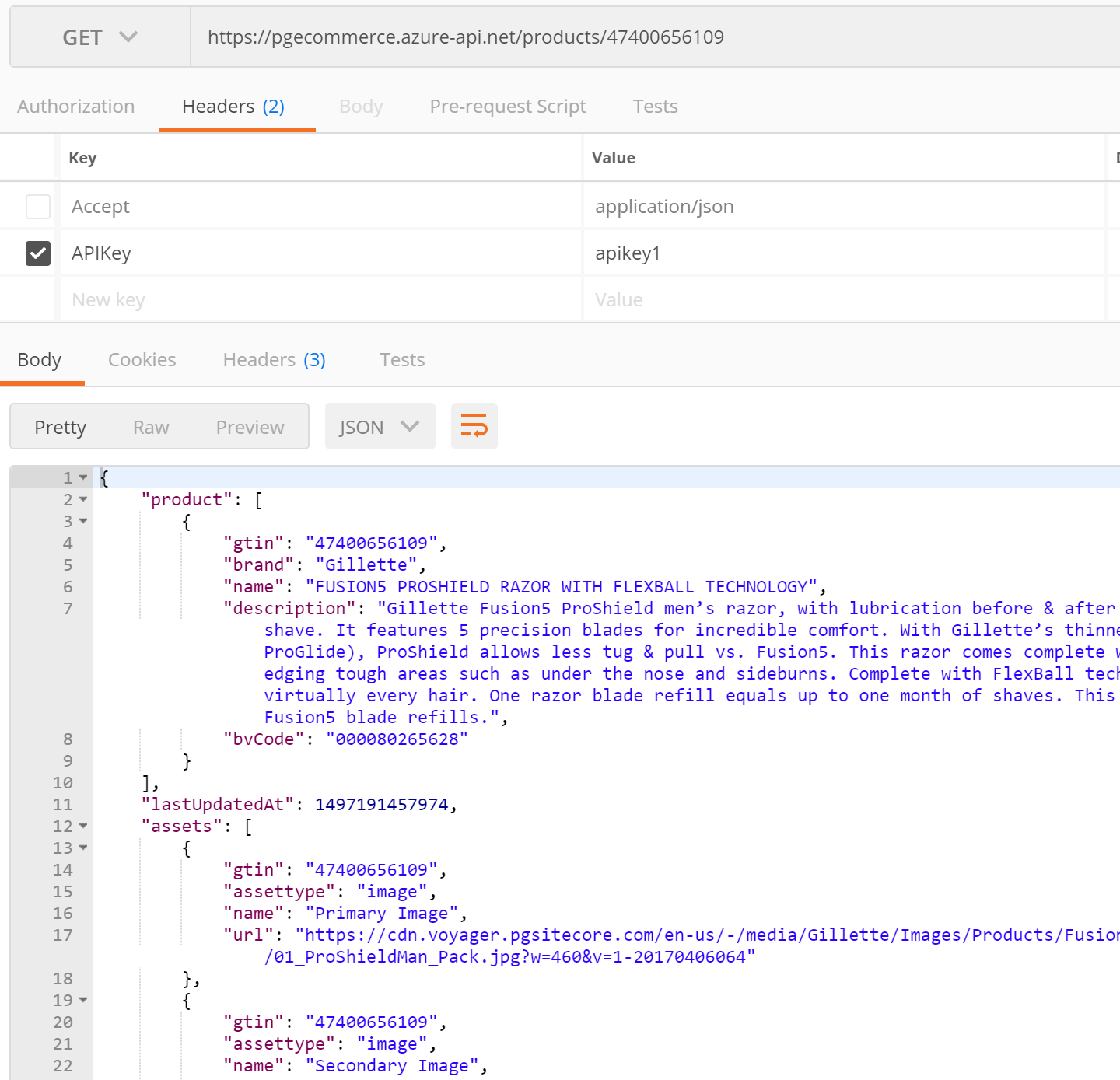
Alternatively copy the code in the attached file to the beginning of in the inbound section of the policy.



1. The above policy ensures that the service
2. Access the Products service and note that it returns a 404 error.

*Note: in ideal scenario, it would have throws a 401 Unauthorized error, but the error policy configured in earlier hands-on masks the 401 error with a 404 error.*

1. Open Postman and configure APIKey HTTP header with one of the values configure in the policy – as shown below. Access the service now and the service now returns results.



# References