Configure Postman to Execute APIs on vManage

Contents

Introduction

System Requirements

Background Information

Configure Postman to Execute the APIs

- Step 1. Open Postman and create a new HTTP request.
- Step 2. Authenticate with your username and password credentials to vManage.
- Step 3. Request a token
- Step 4. Proceed to execute another API to vManage.
- Step 5. Close your session

Run API calls in an automated environment

- How to save token in a variable?
- How to clear SESSIONID cookie for new sessions?
- How to use Collection Runner

Introduction

This document describes how to execute Application Programming Interfaces (APIs) with Postman.

System Requirements

- Postman installed
- Access to vManage and username and password credentials

Note: If you do not have Postman, download it from https://www.postman.com/downloads/

Background Information

The primary or most-commonly-used HTTP verbs (or methods, as they are properly called) are POST, GET, PUT, PATCH, and DELETE.

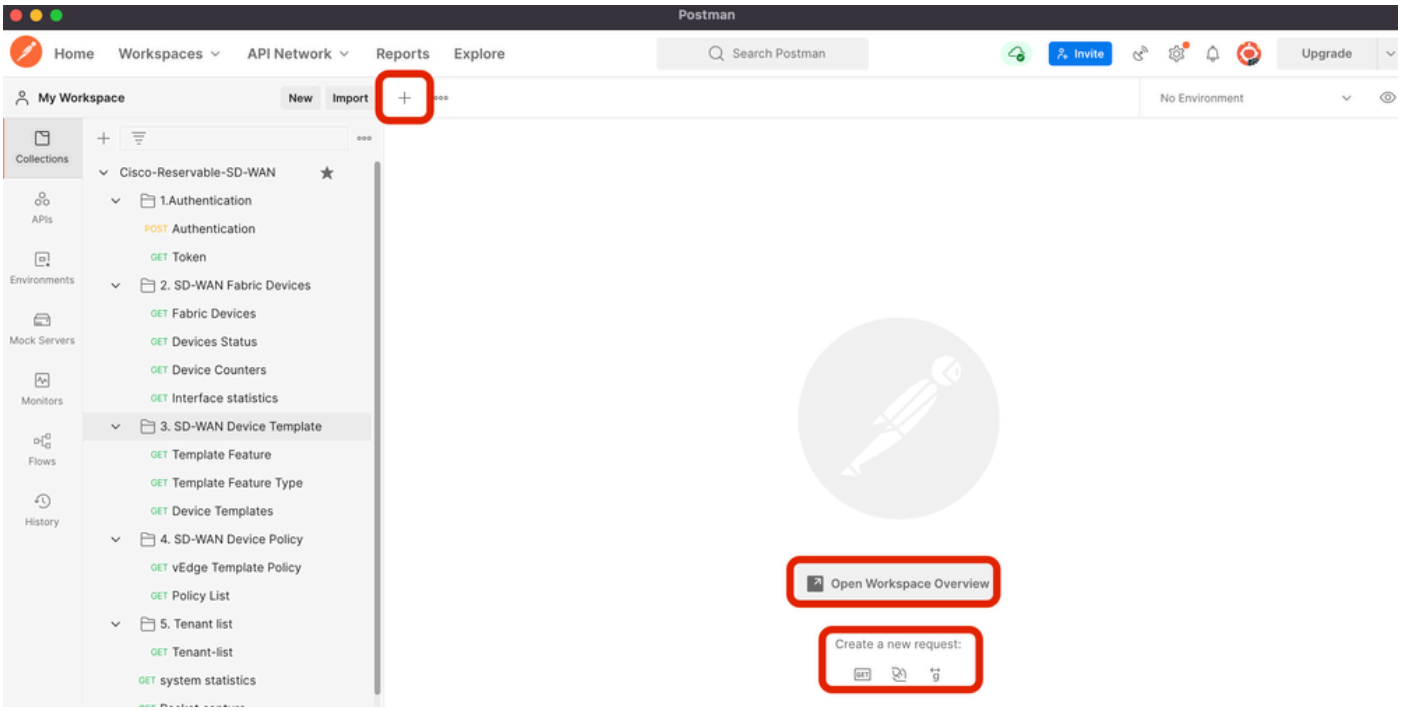
These correspond to create, read, update, and delete (or CRUD) operations, respectively.

There are a number of other verbs, too, but are utilized less frequently. Of those less-frequent methods, OPTIONS and HEAD are used more often than others.

Configure Postman to Execute the APIs

Step 1. Open Postman and create a new HTTP request.

You can create a new HTTP requests if you click in any of the highlighted options.



Create a new HTTP request.

Step 2. Authenticate with your username and password credentials to vManage.

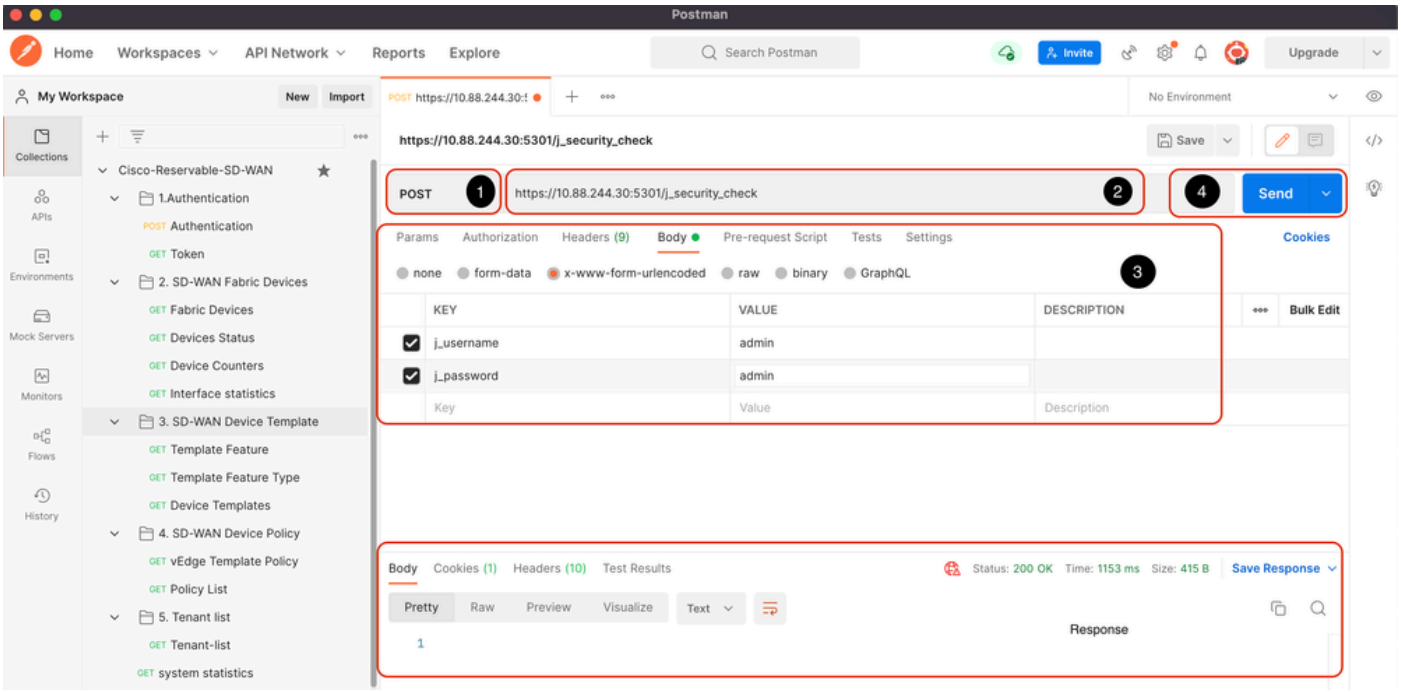
Create another HTTP request.

1. Select **POST** as your HTTP verb.
2. Add **https://<vmanage-ip>/j_security_check**next to POST.
3. Click **Body** and add as **KEY** parameters **j_username** and **j_password** and their values respectively.
4. Click **Send.**

Note: In this example vManage ip address is 10.88.244.30 and the port is 5301

Note: As username and password values, we use admin.

Fullfil the parameters in Postman.



vManage authentication.

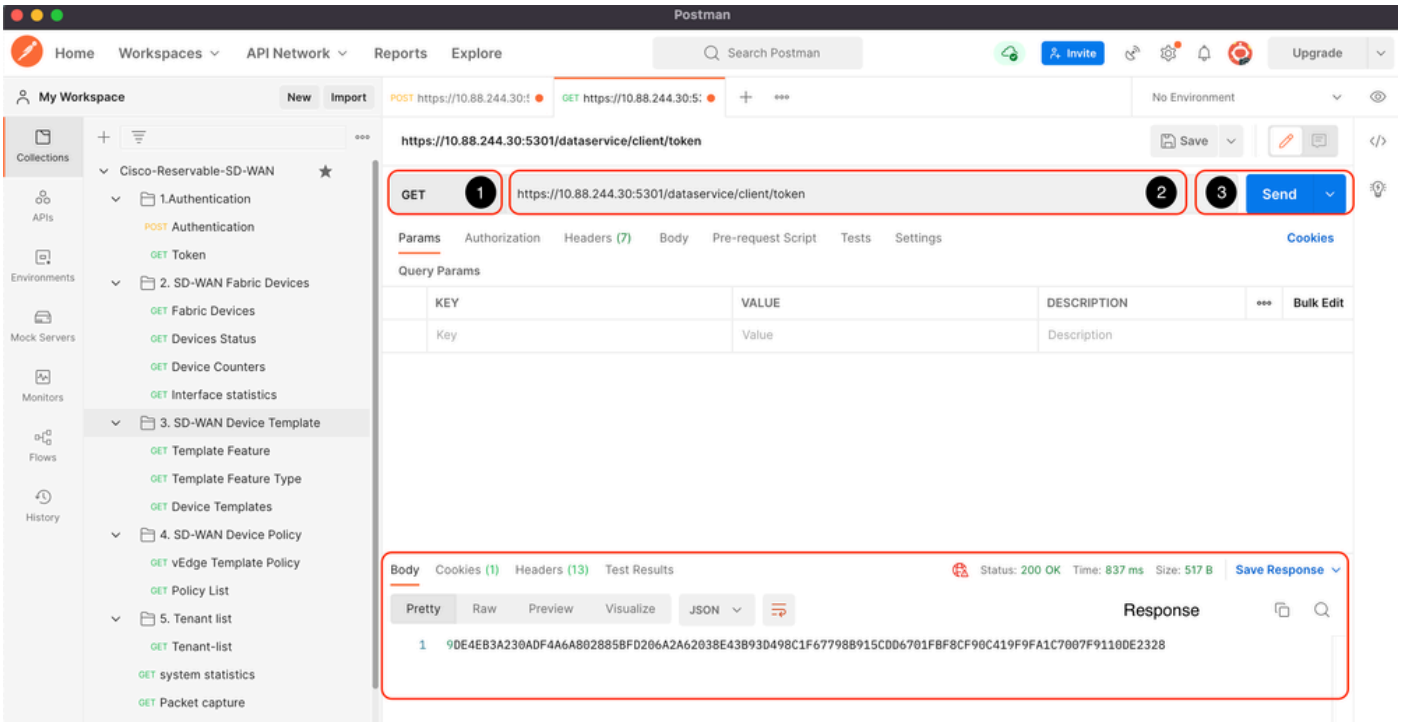
⚠ **Caution:** The response of this API call must be empty

Step 3. Request a token

1. Select **GET** as your HTTP verb.
2. Add the API call details next to GET **https://<vmanage-ip>/dataservice/client/token**
3. Click **Send**

✎ **Note:** Since vManage version 19.2.1, it has been made mandatory that a successfully logged in user needs to send X-XSRG-TOKEN or CSRF token for each POST/PUT/DELETE operation via API call.

Once the API call is executed, you get a response string in the body. Save that string. The image shown exemplifies the output In Postman.



Request a token for vManage

Warning: If you did not get a token as shown in the image please repeat the step.

Step 4. Proceed to execute another API to vManage.

This example entails a POST request

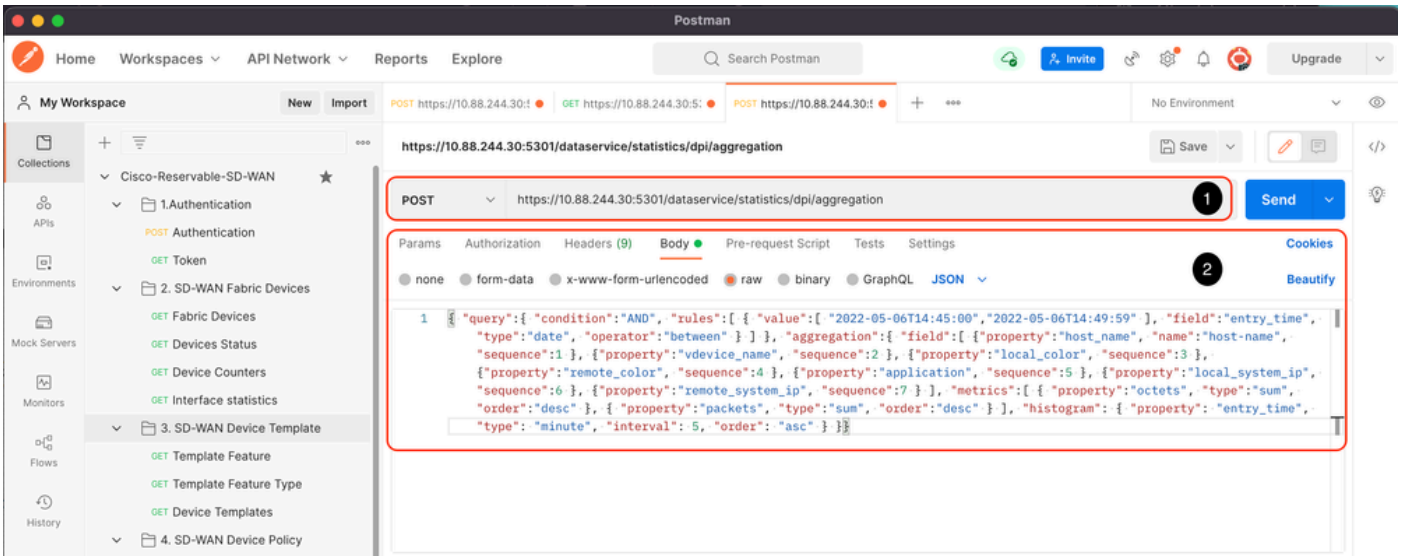
1. Select the API call to execute, in our case is **`https:///dataservice/statistics/dpi/aggregation`**

Tip: If you wish to explore other API calls, please go to vManage url https://vmanage-ip:port/apidocs

---

2. Collect your API call body.

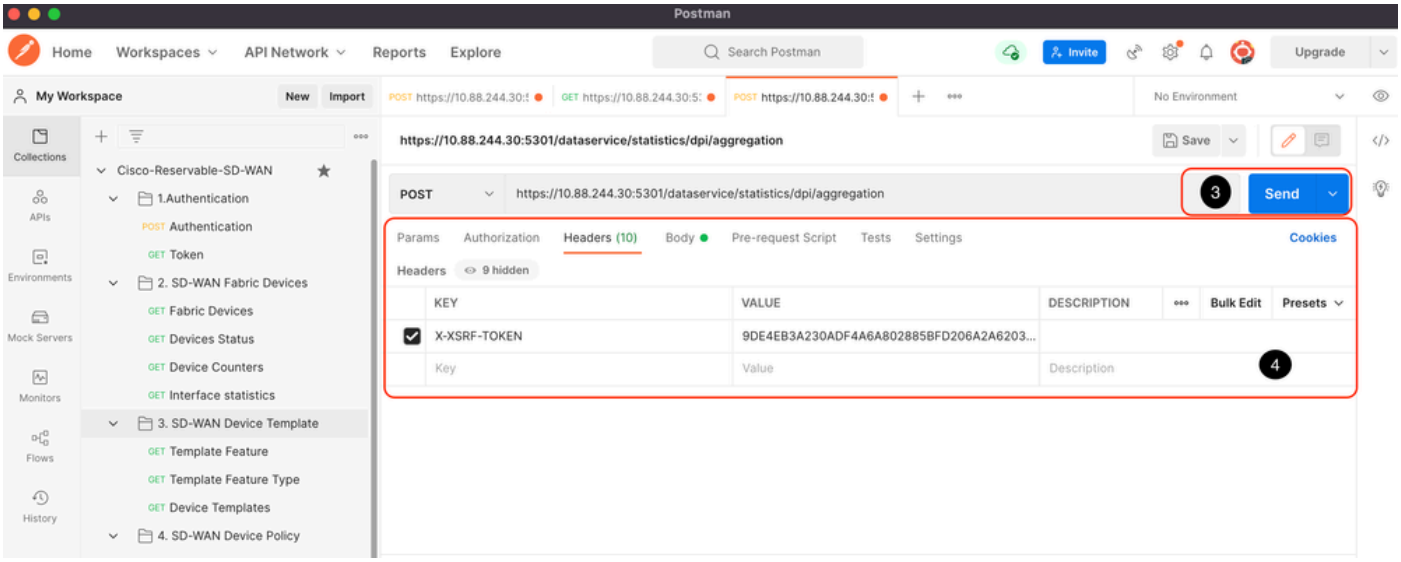
Note: This API call contains a body in JSON format



3. Click **Header** and add as **Key** the string **X-XSRF-TOKEN** as value.

4. Click **Send.**

The image shown displays how your API call must appear.



DPI aggregation API call.

Step 5. Close your session

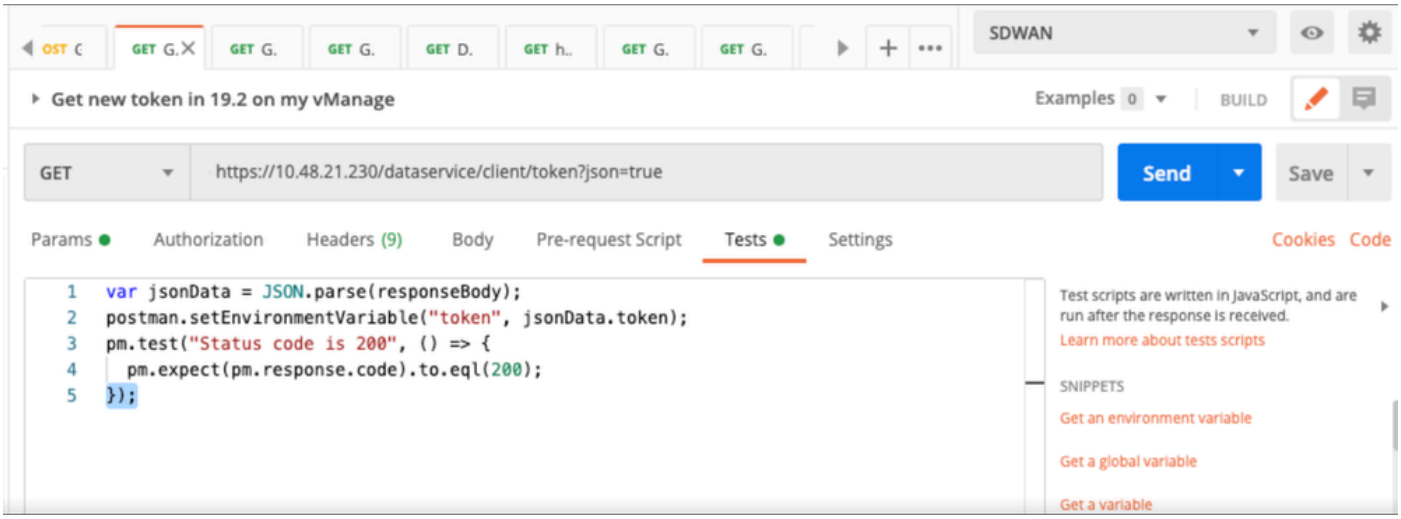
Once you have retrieved all the information needed from vManage and/or the devices, liberate resources of the vManage and eliminate the possibility for malicious users to use your session.

Run API calls in an automated environment

Save cookies and variables to be used in subsequent API calls

How to save token in a variable?

Save the token in a variable for subsequent re-use.

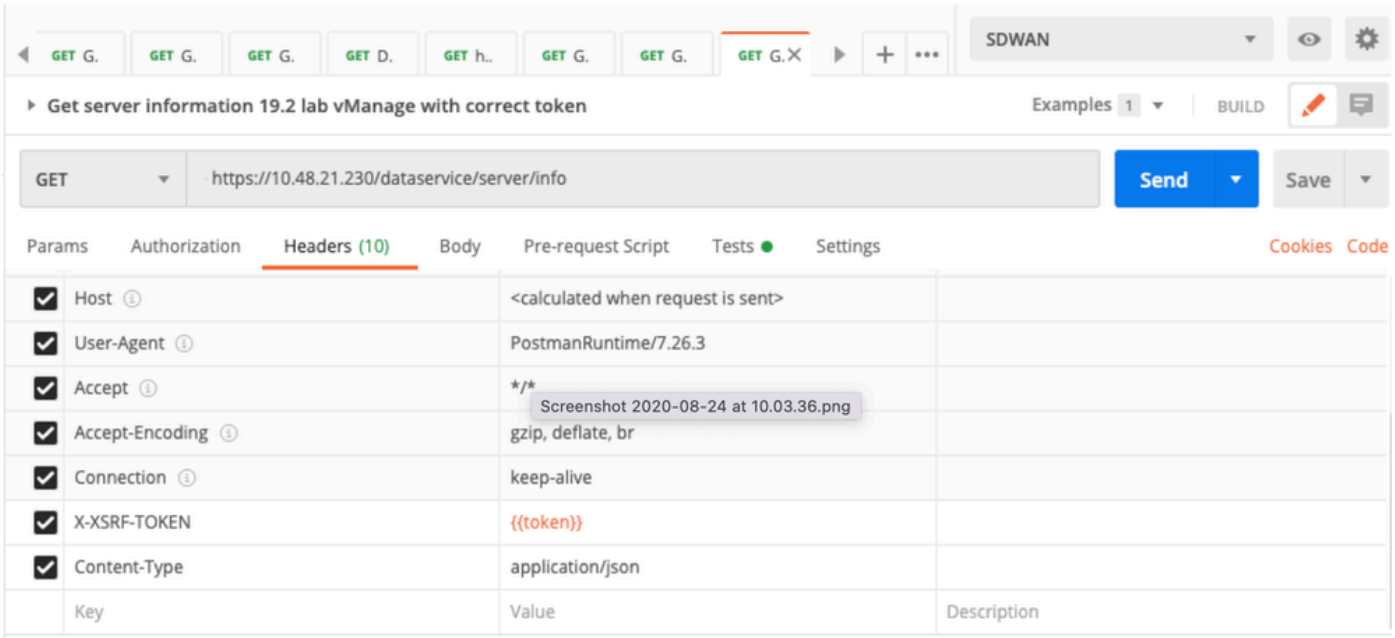


Save the token in a variable

When we request the token in JSON format, store it. Use the **Tests** tab and paste the shown lines.

```
var jsonData = JSON.parse(responseBody);
postman.setEnvironmentVariable("token", jsonData.token);
```

Afterwards, any API call can use token variable.

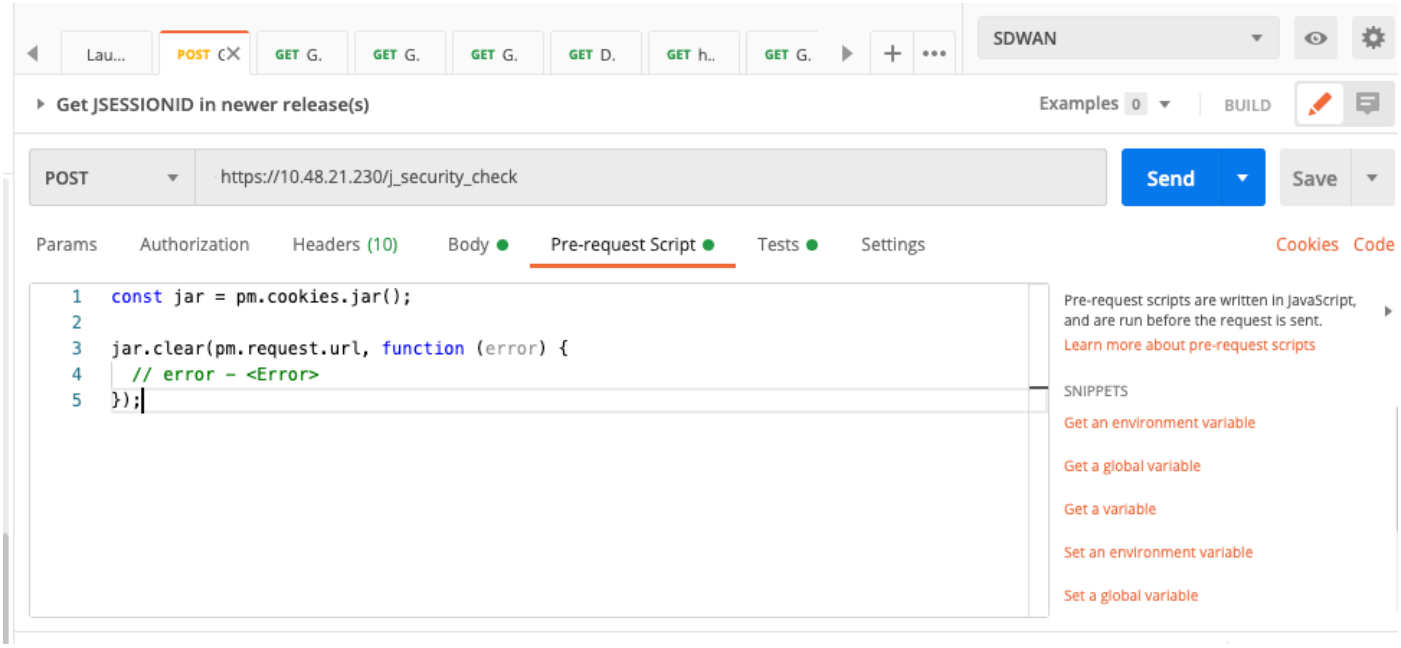


Use the token variable

How to clear SESSIONID cookie for new sessions?

Whenever you execute the API call to get out of, use JSESSIONID.

We cannot use any basic authentication like we did in earlier releases. Instead, we only provide credentials and save the ID in our cookie. Prior this, we can use a pre-test to clear all or specific cookies.



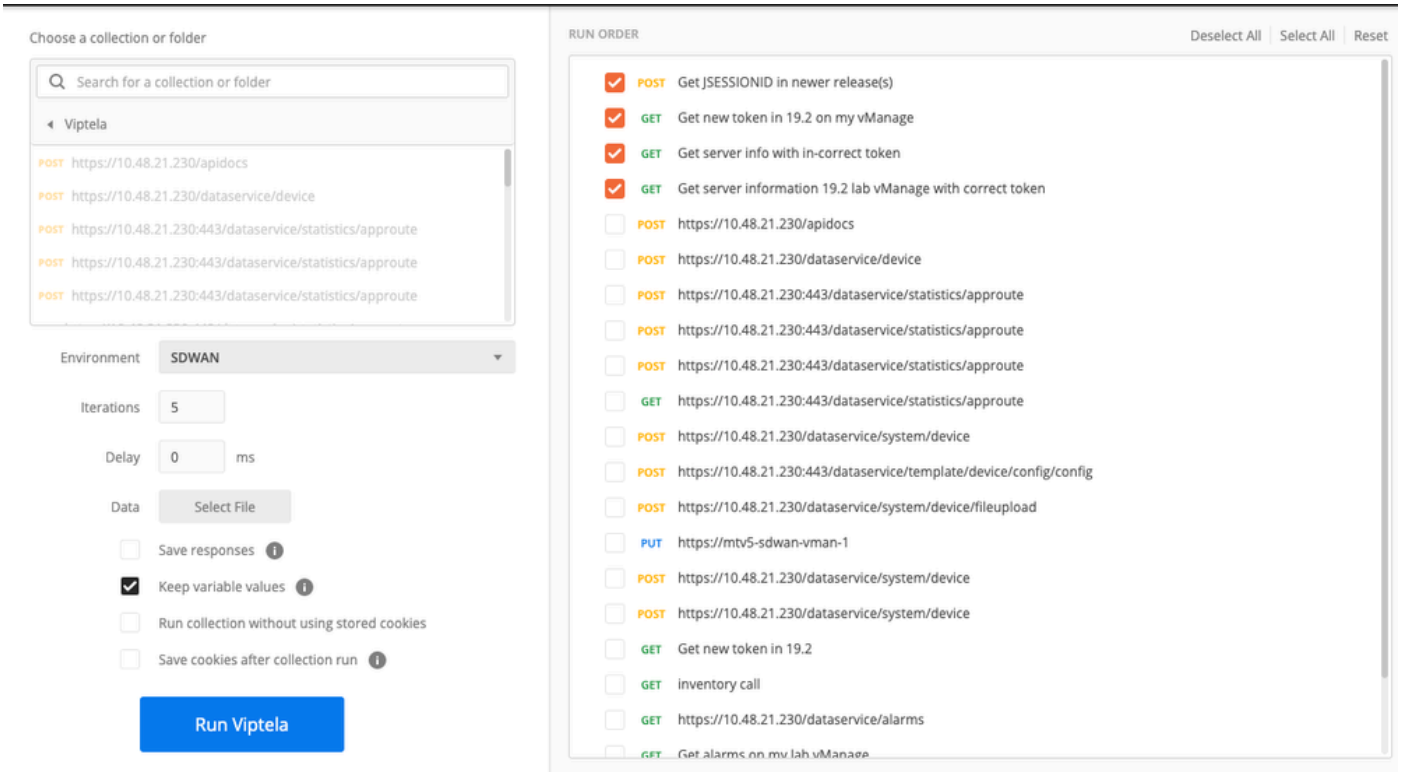
Clear cookies

This is via the code put in the Pre-request Script.

How to use Collection Runner

Now that we have some environment where we can run sessions and save data specific to each session, you can run a sequence of calls usung Collection Runner.

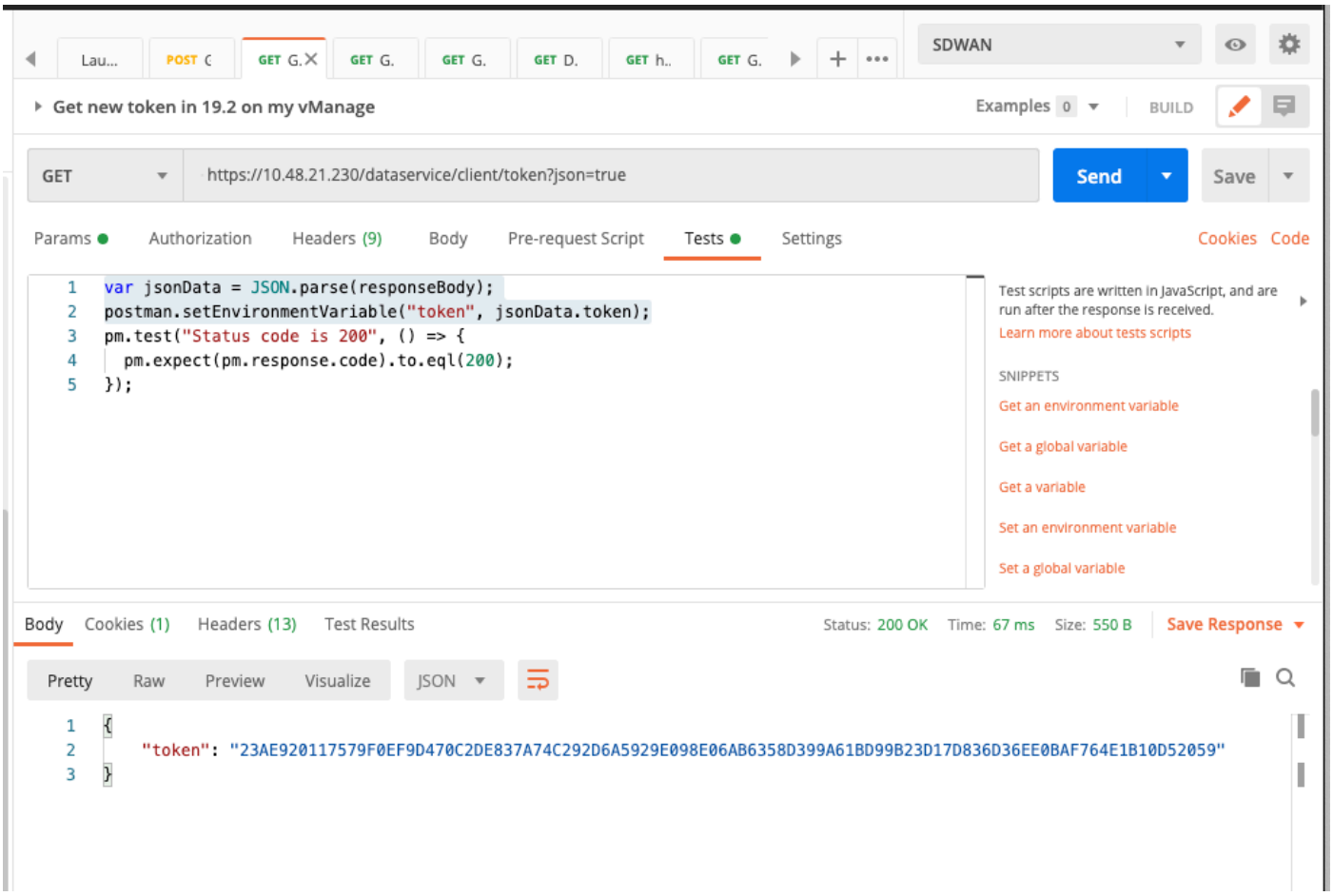
Select the order of events you want to repeat, select the repeat count so Postman can execute the API calls, the selected number of times times with results per run.



Collection Runner

From the "library" of call, put them in a certain order to get a specfic flow/order to be executed.

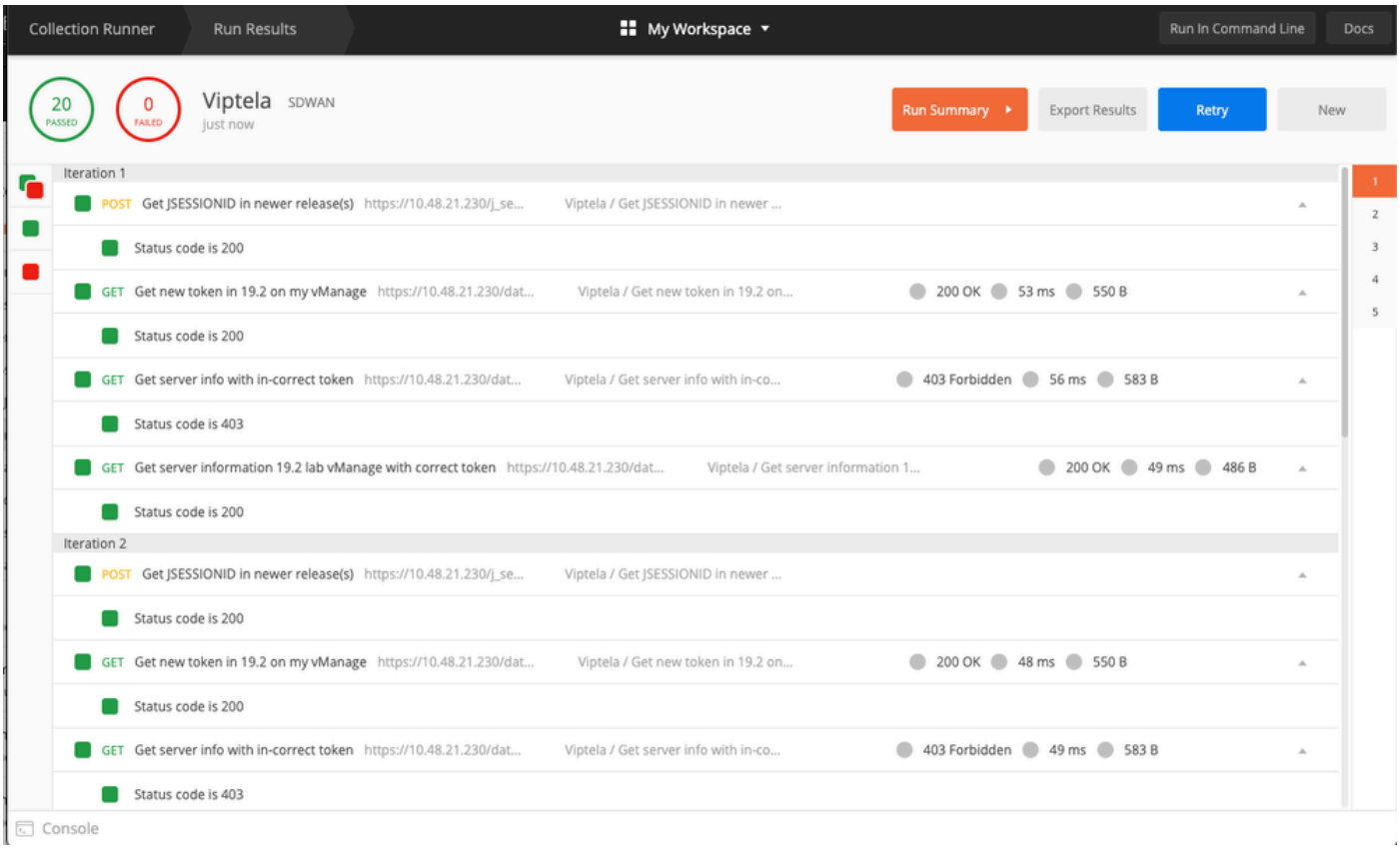
Put in a result check whether you get a 200 OK or other value as response and treat it as pass or fail.



Check response code

```
pm.test("Status code is 200", () => {
 pm.expect(pm.response.code).to.eql(200);
});
```

Then we can see passed or fail in our runs.



Automated run