XIQ API – Postman Runner Starters Guide

03 March 2021
Version 1.1

Table of Contents

Introduction

This document describes how to create an Environment in Postman to make use of the ExtremeCloudIQ Postman API collection.

Postman preparation

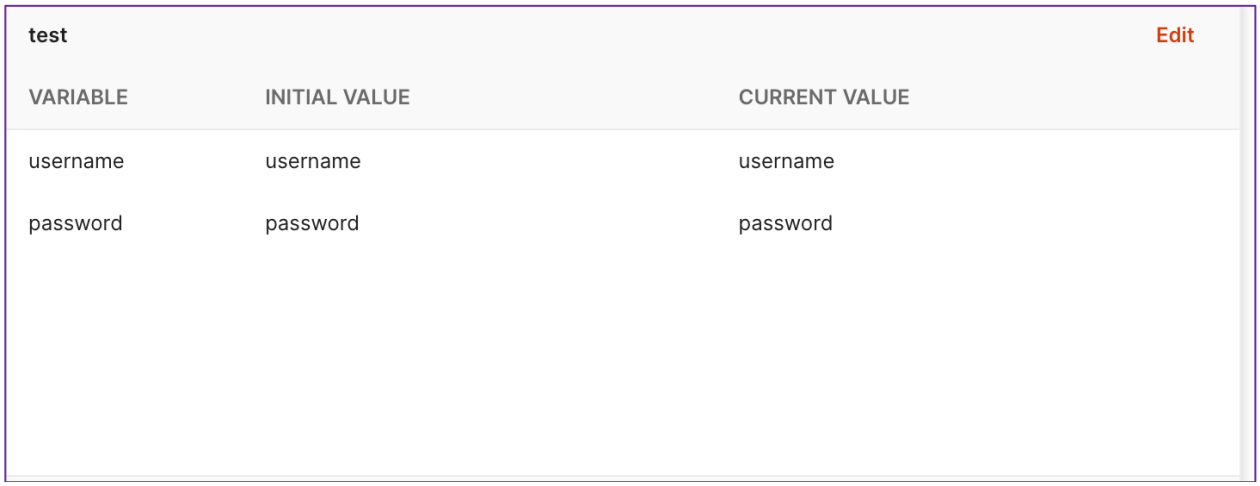
Download the postman test collection from de Extreme Networks Github page and import it into Postman.

To import Postman data, click Import. Select your file. Postman will automatically recognize Postman data, confirming the name, format, and what the file will import as. Click Import to bring your data into Postman.

Create environment

This Collection is using Postman Environment Variables. Create a local environment, create two variable entries one for username and the other for password, and then enter your credentials for the initial and current value and you will be able to use the whole Collection.

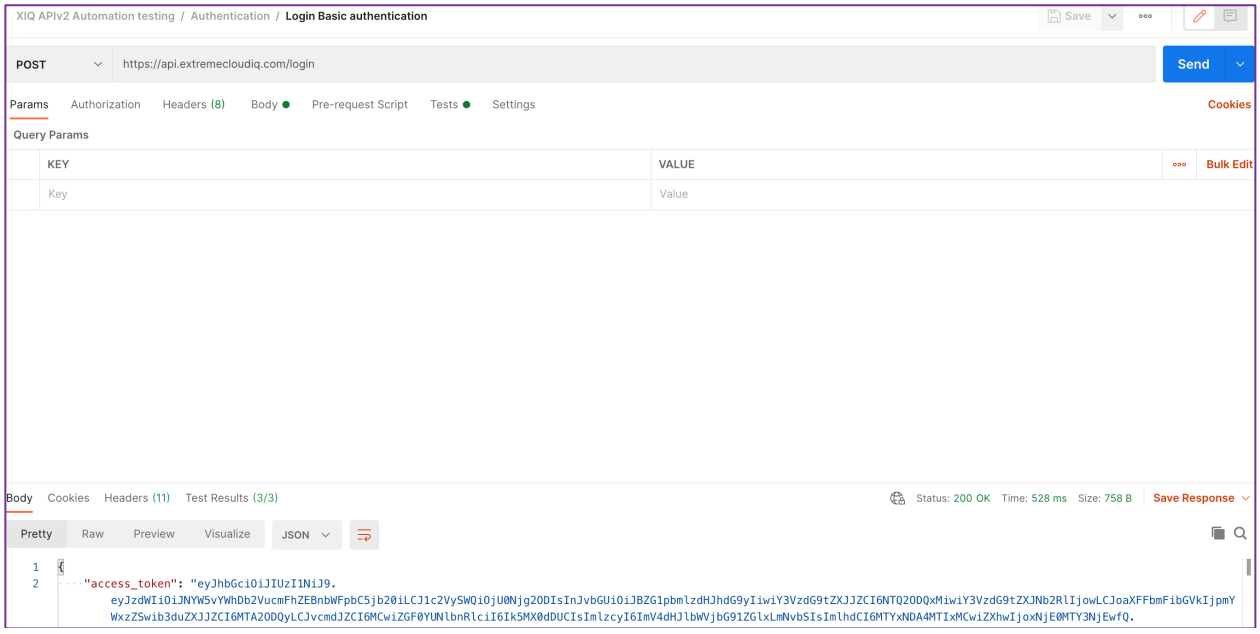
Click the “eye” icon in the upper right corner of the Postman app and select Manage Environments. Click the Add button to create a new environment.



Verify environment

Go to the authentication map and run the Login Basic Authentication request.
Everything is configured fine when you receive an access token from XIQ in the response body of your API request and you will be able to use the whole Collection.

In the Login Basic Authentication request is a pre-request script that resets the variables except the Username and Password. Run this request every time you start a new test run. This ensures that the variables used in other runs are set to default.

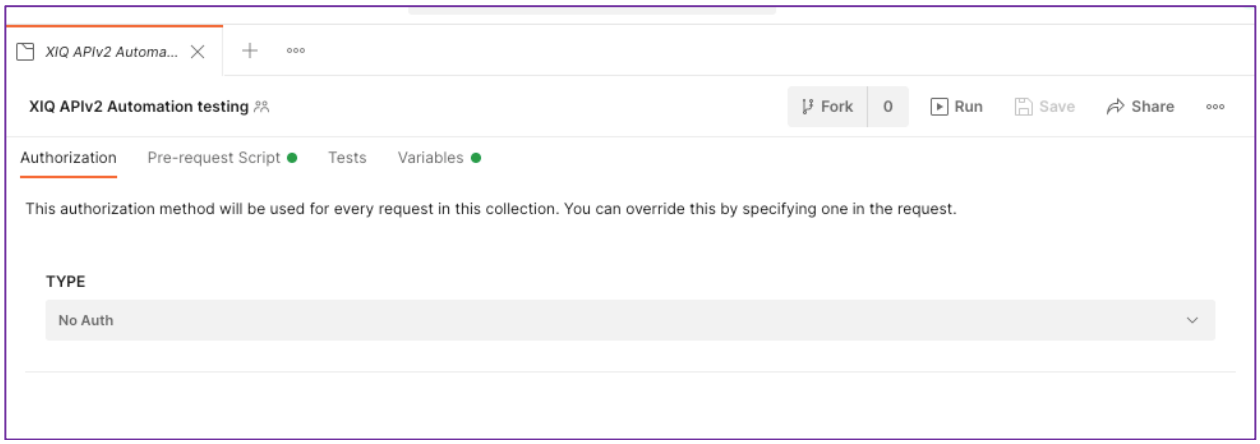


Collection Runner

The Collection Runner allows you to run sets of requests in a specified sequence. The Collection Runner will log your request test results, and the scripts can pass data between the requests.

Starting a collection run

To run a collection, open a collection and click **Run** on the overview tab.

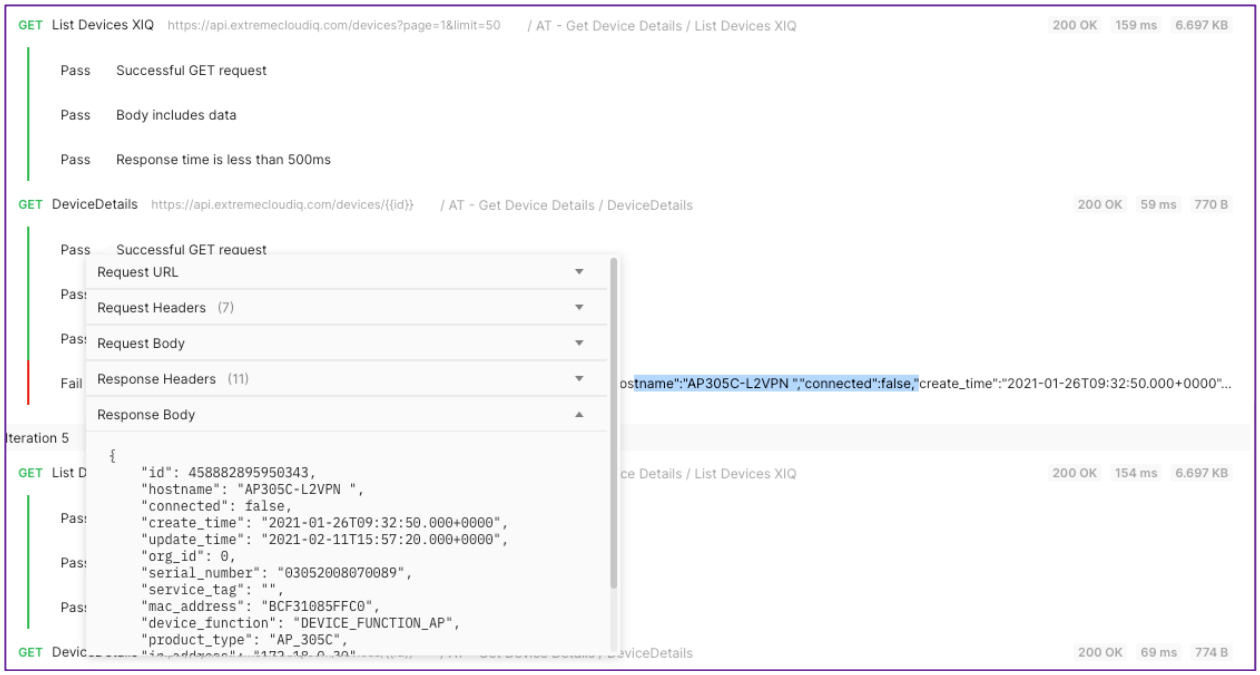


You can optionally choose config parameters for your collection run:

- The number of iterations for your collection run.
- An interval delay (milliseconds) between each request.
- A data file for the collection run.
- Saving responses to the log will allow you to review them but can affect performance.
- Persisting variables will mean that any variables updated by the run will remain changed after it. If you do not persist variables, changes will not be saved after the run completes.
- If your requests use cookies, you can optionally disable them for a collection run.
- Saving cookies means that any values affected by requests during the run will remain after it completes.

Postman will display your request executions and test results in Realtime, but you can export the results in a JSON file was well.

Click a request name to access more data on what happened when it executed.



Available test series

Get device details (xAPIv2)

List device overview -> Show device details for specific device

This workflow will list all the devices in ExtremeCloudIQ and it will show the details for the devices.
Change the number of iterations to the number of devices in ExtremeCloudIQ and you will see the device details of all these devices in the output of the test results.

This run will test the following items:

- Successful get request
- Data in response body
- Response time
- Serial number of device per run

Change admin status (xAPIv2)

List device overview -> show device details for specific device -> Change status to unmanaged -> Show device details to verify -> Change status to managed -> Show device details to verify

This workflow will help you to unmanage or manage your devices.
This run will test the following items:

- Successful get request
- Data in response body
- Response time
- Verifiy the device status (Managed or unmanaged)
- Serial number of device per run

Get SSID details (xAPIv2)

List network policies -> List SSID for a specific network policy

This workflow will list all the Network policies in ExtremeCloudIQ and it show you the SSID configured per Network policy.
Change the number of iterations to the number of network policies in ExtremeCloudIQ and you will see the SSID details of all these Network policies in the output of the test results.

This run will test the following items:

- Successful get request
- Data in response body
- Response time
- Network policy name

Change Network policy (xAPIv2)

List device overview -> List network policies -> Get Network policy for 1 device -> Assign network policy to device -> Get network policy for 1 device to verify

This workflow will help you to change the network policy for 1 or multiple devices. You need to import a CSV file with deviceIDs and NetworkIDs to be able to run this workflow

This run will test the following items:

- Successful get request
- Data in response body
- Response time
- Network policy name

Push CLI commands (xAPIv2)

This workflow will list all the devices in ExtremeCloudIQ and it will show the details for the devices. It will send the CLI command defined in the request body to the device.

Change the number of iterations to the number of devices in ExtremeCloudIQ and you will see the details of all these devices in the output of the test results and the CLI command will be send to all these devices.

This run will test the following items:

- Successful get request
- Data in response body
- Response time

Onboard device and assign NP(xAPv2)

Onboard devices -> List devices -> List Network Policies -> assign Network policy to multiple devices -> List assigned devices for Network policy.

This workflow will help you to onboard devices and to update the network policy for 1 or multiple devices. You need to import a CSV file with deviceIDs and NetworkIDs to be able to run this workflow

This run will test the following items:

- Successful get request
- Data in response body
- Response time

Private Client Group – port assignment (xAPIv2)

List all key based PCG -> get key based PCG users -> Get port assignment details of AP150W -> assign ports on AP150W -> Get port assignment details of AP150W

This workflow will help you to assign users to the ports of a PCG enabled AP150W.

This run will test the following items:

- Successful get request
- Data in response body
- Response time
- Assigned users to ETH1, ETH2 and ETH3

Onboard and configure switches (xAPIv1 and xAPIv2)

Onboard device -> Get device details for one device ->Change hostname for device -> Get port types -> Put switch port details -> Get switch port details -> Push configuration and firmware to device -> Get configuration deployment status -> List assigned devices for network policy

This workflow will help you to onboard switches, make some device specific configurations and to update the network policy for 1 or multiple devices. You need to import a CSV file with deviceIDs and NetworkIDs to be able to run this workflow

This run will test the following items:

- Successful get request
- Data in response body
- Response time
- Serial number of device per run