

A man with short dark hair, wearing a purple button-down shirt under a black patterned jacket, stands with his hands raised in a dramatic, expressive gesture. He has a slightly open-mouthed, intense expression. The background is a dark, solid color.

arRESTed Development

Using REST APIs and Python

Mitch Davenport

Download Script: <https://github.com/mitch-davenport/cm-x-api-example>

API Automation

Define requirements

- Plot a device on a map

Find the API reference guide

- Find the relevant resources

Test the resources with Postman

- Find the true capabilities

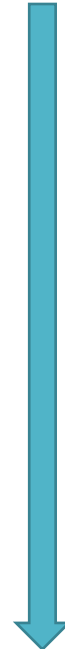
 - Some functionality is undocumented*

- Identify caveats

Create pseudocode

Convert pseudocode to Python

Low



High

Programming
Knowledge

Find the API Reference Guide

Search for CMX REST API

<https://yourCMXServer/apidocs/>

Find the relevant resources

Location

Client History API

Get client history by mac address

Configuration

Map Resources API

Get floor information

Get floor image



Location

Use the Location based REST APIs to find location specific details on visitors.

[View details »](#)



Configuration

Use the Configuration REST APIs to configure different aspects of MSE.

[View details »](#)

Postman?

Postman is a free client

Use it to test APIs

Craft web requests

How to use

Insert an API URL

Choose the type

Craft headers

The screenshot displays the Postman interface for a GET request titled "Client History By MAC". The URL is "https://10.151.50.63/api/location/v1/history/clients/dc:0c:5c:af:4e:89". The "Headers" tab is active, showing a table with two rows: "Authorization" (checked) and "Key".

KEY	VALUE
Authorization	Basic bmV0dG9vbHM6SGx1...
<input checked="" type="checkbox"/> Authorization	username:password
Key	Value

GET

This API returns client history by macaddress & date in format yyyy/mm/dd. /api/location/v1/history/clients/:macaddress

+ ...

Test with Postman

Client History API

Input

MAC address

Output

Map hierarchy

Campus

userCampus

Building

userBuilding

Floor

userFloor

Map Coordinates

X

userXFeet

Y

userYFeet

Unit is feet

Client History By MAC

GET https://10.151.50.63/api/location/v1/history/clients/dc:0c:5c:af:4e:89

Params Authorization Headers (2) Body Pre-request Script Tests

KEY	VALUE
Authorization	Basic bmV0dG9vbHM6SGx1...
<input checked="" type="checkbox"/> Authorization	username:password
Key	Value

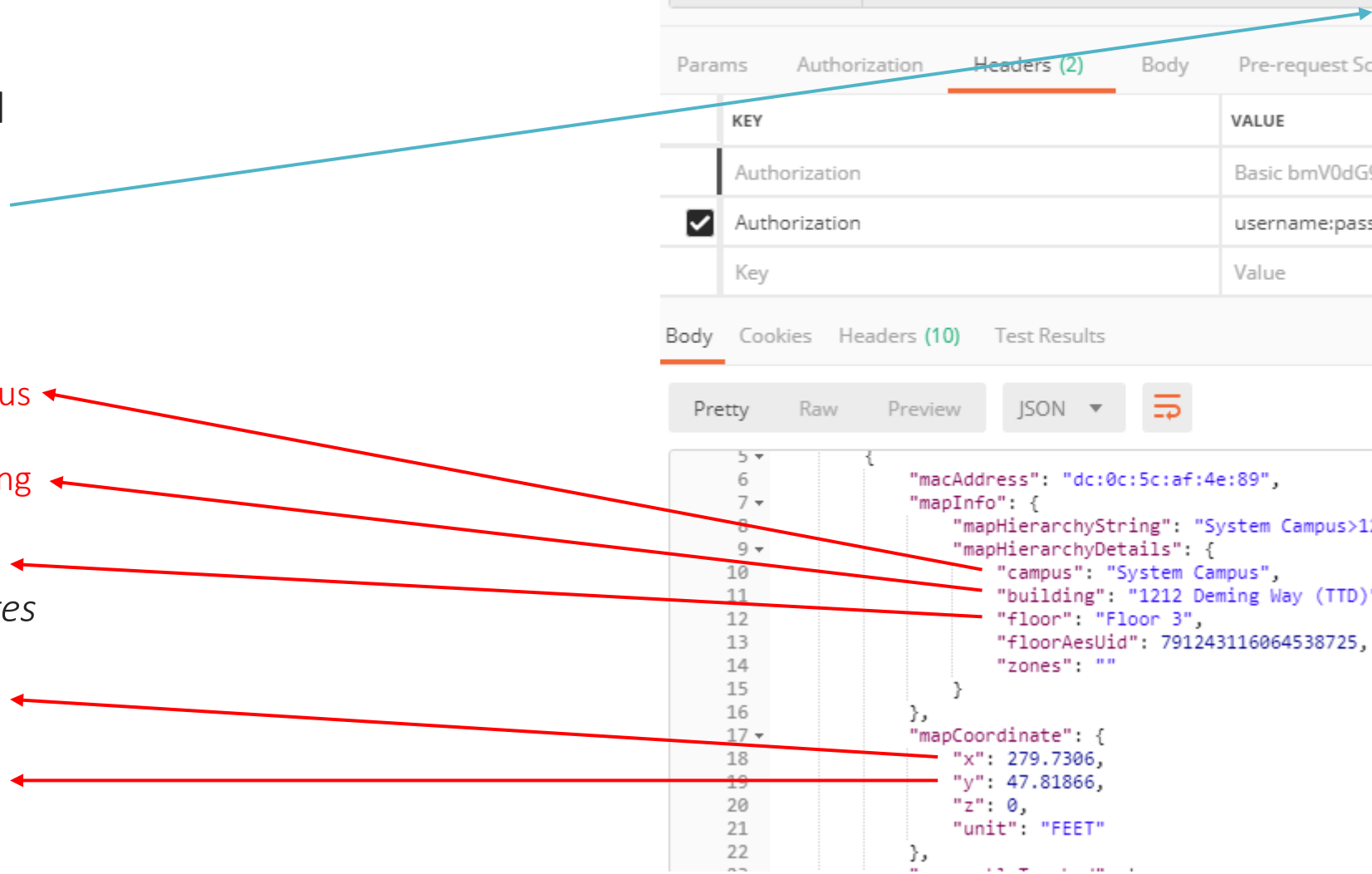
Body Cookies Headers (10) Test Results

Pretty Raw Preview JSON

```

5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
...
{
  "macAddress": "dc:0c:5c:af:4e:89",
  "mapInfo": {
    "mapHierarchyString": "System Campus>1212 Deming Way",
    "mapHierarchyDetails": {
      "campus": "System Campus",
      "building": "1212 Deming Way (TTD)",
      "floor": "Floor 3",
      "floorAesUid": 791243116064538725,
      "zones": ""
    }
  },
  "mapCoordinate": {
    "x": 279.7306,
    "y": 47.81866,
    "z": 0,
    "unit": "FEET"
  }
}

```



GET

Get floor inside specific building and specific campus /api/config/v1/maps/info/:campusName/:buildingName/:floorName

Test with Postman

Floor Information API

Input

userCampus

userBuilding

userFloor

Output

Map dimensions

Length

mapYFeet

Width

mapXFeet

Unit is feet

Image dimensions

Width

mapXPixels

Height

mapYPixels

Unit is pixels

Get Floor Information

Add a description

GET

https://10.151.50.63/api/config/v1/maps/info/:campusName/:buildingName/:floorName

Params

Authorization

Headers (2)

Body

Pre-request Script

Tests

KEY

VALUE

campusName

System Campus

buildingName

1212 Deming Way (TTD)

floorName

Floor 3

KEY

VALUE

Key

Value

Body

Cookies

Headers (10)

Test Results

Status: 200 OK

Time

Pretty

Raw

Preview

JSON



```

1 {
2   "aesUid": 791243116064538725,
3   "calibrationModelId": 791243116064538625,
4   "objectVersion": 0,
5   "name": "Floor 3",
6   "dimension": {
7     "length": 234,
8     "width": 320.5,
9     "height": 10,
10    "offsetX": 0,
11    "offsetY": 0,
12    "unit": "FEET"
13  },
14  "isOutdoor": false,
15  "floorNumber": 1,
16  "image": {
17    "imageName": "1546543052732.jpg",
18    "zoomLevel": 5,
19    "width": 4020,
20    "height": 2935,

```

GET Get floor image /api/config/v1/maps/image/:campusName/:buildingName/:floorName

Test with Postman

Floor Information API

Input

userCampus

userBuilding

userFloor

Output

Image file

mapImageFile

GET Get Floor Image

Get Floor Image

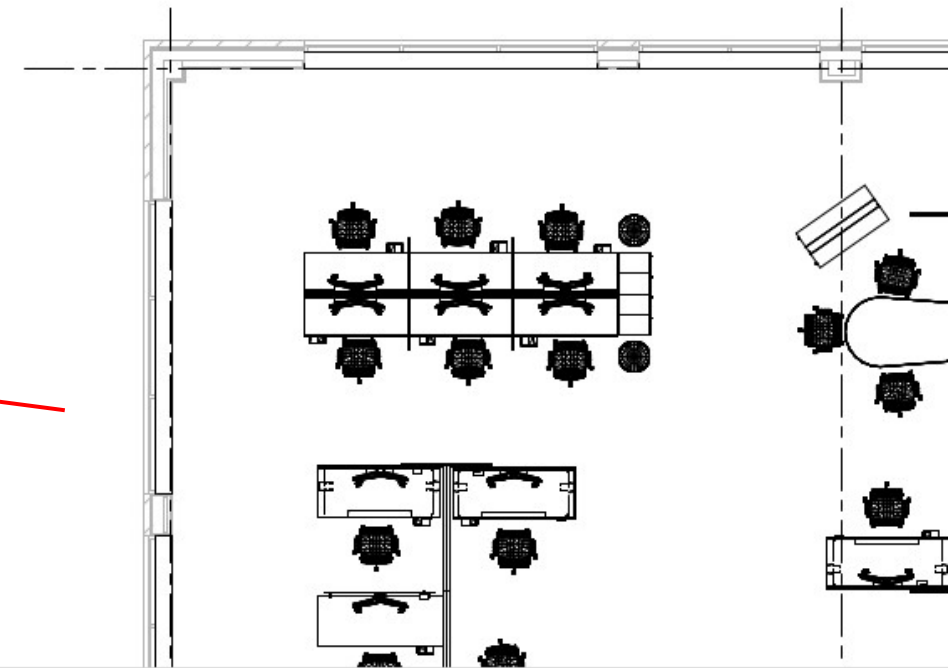
Add a description

GET https://10.151.50.63/api/config/v1/maps/image/:campusName/:buildingName/:floorName

KEY	VALUE
campusName	System Campus
buildingName	1212 Deming Way (TTD)
floorName	Floor 3
KEY	VALUE
Key	Value

Body Cookies Headers (10) Test Results

Status: 200 OK Time: 6



Create Pseudocode

Populate variables

Call client history API

Input: client MAC address

Call map info API

Input: *userCampus*, *userBuilding*, *userFloor*

Call map image API

Input: *userCampus*, *userBuilding*, *userFloor*

Convert users feet to pixels

Determine scale factor

$scaleFactor = mapXPixels / mapXFeet$

$userXPixels = userXFeet * scaleFactor$

$userYPixels = userYFeet * scaleFactor$

Draw location on map

Import *mapImagefile* into memory

Draw a red circle with radius 50 pixels centered on *userXPixels* and *userYPixels* on top of *mapImageFile*

Save the modified image to disk as user-location.png

Populated variables

Located Position

userXFeet

userYFeet

Current Location

userCampus

userBuilding

userFloor

Map X, Y in feet

mapXFeet

mapYFeet

Map X, Y in pixels

mapXPixels

mapYPixels

Map image file

mapImageFile

Convert Pseudocode to Python



Leverage existing modules

Install new modules with pip

`“pip install ModuleName”`

Make REST calls with “Requests”

Convert response to JSON

Manipulate images with “Pillow”

Draw circle

Use `ImageDraw.Draw.ellipse`

Resize image

`Image.thumbnail`



Convert Pseudocode to Python



```
def callClientHistoryAPI (macAddressToLocate):  
    return userXFeet, userYFeet, userCampus, userBuilding, userFloor  
  
def callFloorInformationAPI (userCampus, userBuilding, userFloor):  
    return mapXFeet, mapYFeet, mapXPixels, mapYPixels  
  
def callFloorImageAPI (userCampus, userBuilding, userFloor):  
    return mapImageFile  
  
def drawClientLocationOnImage (mapImageFile, userXFeet, userYFeet, scaleFactor):  
    return mapImageFileWithClient  
  
if __name__ == '__main__':  
    callClientHistoryAPI (macAddressToLocate)  
    callFloorInformationAPI (userCampus, userBuilding, userFloor)  
    callFloorImageAPI (userCampus, userBuilding, userFloor)  
    drawClientLocationOnImage (mapImageFile, userXFeet, userYFeet, scaleFactor)
```

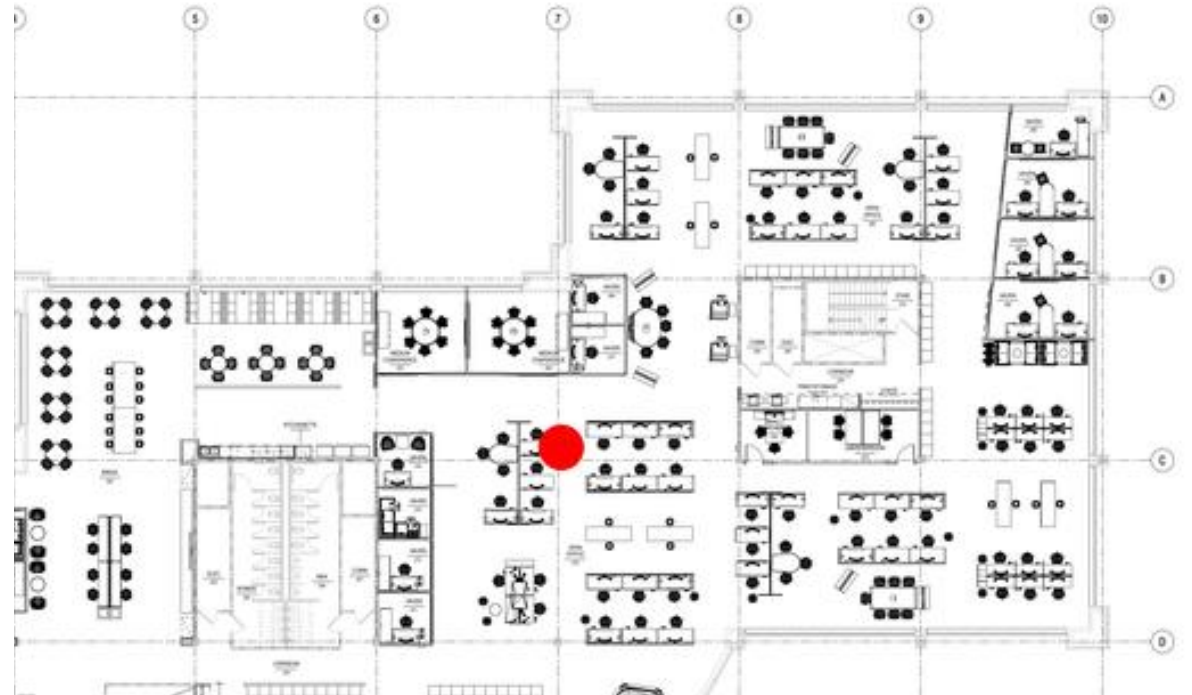
Extend Functionality

Plot client history over time

Plot locations of clients with poor signal

Plot the AP locations on the map

Make accessible via a webpage



Links

Python Script

<https://github.com/mitch-davenport/cmx-api-example>

Postman

www.getpostman.com

Python

www.python.org

Requests

www.python-requests.org

Pillow

www.python-pillow.org

API Uses for WLAN Professionals

Create Bots

Jabber, Slack, Twitter bots that relay information or perform commands

Wireless Client Locations

Allow helpdesk to locate wireless clients

Custom Dashboards

Integrate data from multiple API sources and different vendors into a single pane

Create Your Own APIs

Allow users to reset POE on a port simply by going to a URL

Example: www.network.net/resetpoeport.php?hostname=my-sw-1&interface=Gi3/0/48

Thank You!