XIQ WiNG Migration Guide

by Tim Smith, SA – 011/11/2024 – v2.5

Overview:

This guide covers how to run the XIQ_wing_migrate.py script. This script can migrate the location hierarchy, rf-domains, and floors from WiNG to ExtremeCloud IQ (XIQ). This will save time by creating the location(s), building(s), and floor(s) from data within a WiNG Tech-Dump file. The script will also move APs to a floor in the associated building.

Target Audience: Semi-Technical

Definitions

XIQ	ExtremeCloud IQ
VIQ	Virtual ExtremeCloud IQ

Table of Contents

Prerequisites:

- ExtremeCloud IQ Public Cloud or Private Cloud instance (IQVA on-prem is not supported.)
- Knowledge of XIQ by onboarding access points; creating locations, buildings, and floors; naming and setting locations of access points; Uploading and assigning floorplans to floors
- RadSec Proxy requires TCP Port 2083 to be open on your internet firewall.
- One or more XIQ native access points or campus-based Wi-Fi systems (WiNG or IdentiFi)
- Download XIQ_Wing_location script files from GitHub.
    - https://github.com/timjsmith24/XIQ_Wing_location_migration
    - wing_migrate.py
    - app folder
        - Wing_importer.py
        - mapImportLogger.py
        - xiq_exporter.py
        - cc_map.csv
        - templates folder
            - wing_apconfig.textfsm
            - wing_rfdomain.textfsm
    - Wing Tech-dump file
        - (Versions tested)
            - 7.5.1.6-001R
            - 7.7.1.8-009R

Scripting Environment Preparation:

Information:

The XIQ_wing_migrate.py script requires, at minimum, Python 3.8 and tested up to Python 3.12. This script has only been tested with macOS and Windows but may be able to be executed from any device with Python and the needed modules installed. This script must be able to access the Wing Tech dump file and reach out to ExtremeCloud IQ.

The script, when run, will create a *map_importer.log* file in the app folder. This log file will show information about locations, buildings, and floors being created and the APs throughout the migration process. Any API errors experienced will also show up in the log file.

Device Choice:

This script has only been tested in MacOS, both Monterey and Ventura, and Windows, both versions 10 and 11. This script may be executed from any device that can run Python 3.8 or higher but is currently only supported once testing is completed. The device will need to be able to reach ExtremeCloud IQ. This can be done through a proxy. Proxy config is beyond the scope of this guide.

Python Installation:

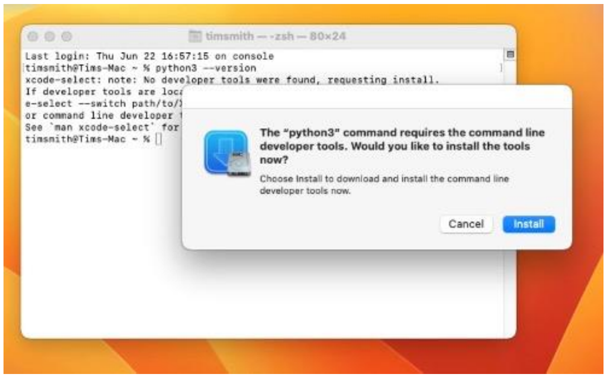
Depending on the device used, you may need to install Python or a different version of Python. The easiest way to check the version of Python is to open the terminal and type this command.

```
python3 --version
```

Below is an example of installing python3 for Mac OSX.

Mac OSX Ventura

- Open the terminal and enter *python3 --version*
  - This triggers the installation of Developer Tools
- Click Install
- Click Agree



```
Last login: Thu Jun 22 16:57:15 on console
timsmith@Tims-Mac ~ % python3 --version
xcode-select: note: No developer tools were found, requesting install.
If developer tools are loc
e-select --switch path/to/
or command line developer
See `man xcode-select` for
timsmith@Tims-Mac ~ %
```

The “python3” command requires the command line developer tools. Would you like to install the tools now?

Choose Install to download and install the command line developer tools now.

This will install Python version 3.9.6

- The Developer tools that installed python3 will also install pip3 in Ventura.
- Mac terminal will be used to install Python modules
- Running this command will check if pip is installed

```
pip3 --version
```

Required Modules:

The **requests, pandas,** and **textfsm** modules are required for the XIQ_wing_migration.py script. **>Note:** When pandas 3.0 comes out pyarrow will be required so pyarrow(15.0.0) has been added to the requirement.txt file.

Checking for existing Modules

You can check if the required modules are installed using the terminal. For each module, run the following command.

```
python3 -c “import requests”
```

```
python3 -c “import pandas”
```

```
python3 -c “import textfsm”
```

The module is not installed if a ‘*ModuleNotFoundError: No module named '<module name>*' error is returned.

Installing required modules

The required modules can be installed using pip3 using the downloaded requirements.txt file with the following command.

```
pip3 install -r requirements.txt
```

Or the modules can be installed individually using.

```
pip3 install requests==2.31.0
```

```
pip3 install pandas==2.2.1
```

```
pip3 install textfsm==1.1.3
```

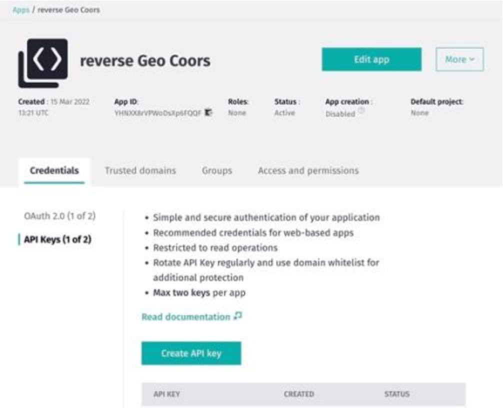
GeoCoordinates:

Information:

WiNG allows Geo Coordinates to be added to an rf-domain. Currently, XIQ doesn’t support Geo Coordinates; physical addresses can be used for buildings. If geo coordinates are configured in the rf-domain, the script can convert them to the physical addresses and assign that address to the building when creating it in XIQ. To do this, an API call must be made to a third-party location service at platform.here.com. To leverage this, an API token will need to be created. There is a free tier that allows up to 1000 requests per day.

Detailed instructions on creating the API token will be added in the full document, but for now, follow these steps.

1. Go to account.here.com and sign up for an account.
2. Once registered on platform.here.com, click the launcher on the top right of the screen and select Access Manager.
3. At the top select Apps, then 'Register a new app.'
4. Give the app a name like 'reverse geo coords' and select 'Register.'
5. Under credentials in the newly created app, select API Keys, then click 'Create API key.'
6. Copy the created API key and add it between the quotes on line 17 of the XIQ_wing_migrate.py script.



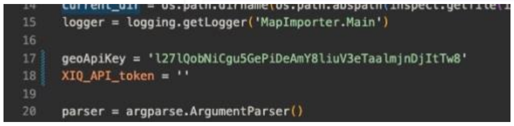
Apps / reverse Geo Coors

reverse Geo Coors

Created : 15 Mar 2022 13:21 UTC
App ID: YHNXX8rVPWoDsXp6FQQF
Roles: None
Status : Active
App creation : Disabled
Default project: None

- Simple and secure authentication of your application
- Recommended credentials for web-based apps
- Restricted to read operations
- Rotate API Key regularly and use domain whitelist for additional protection
- **Max two keys** per app

API KEY	CREATED	STATUS



```
14  current_dir = os.path.dirname(os.path.abspath(inspect.getfile(i
15  logger = logging.getLogger('MapImporter.Main')
16
17  geoApiKey = 'l27lQobNiCgu5GePiDeAmY8liuV3eTaalmjnDjItTw8'
18  XIQ_API_token = ''
19
20  parser = argparse.ArgumentParser()
```

If no API token exists or the geo-coordinates are not configured in the rf-domains, the buildings will be created with 'Unknown Address' in XIQ.

>Note: You can mute alerts by adding the --nogeolog flag.

Access to XIQ:

By default, the script, when run, will prompt the user for XIQ credentials. This User will have to have administrative access to the XIQ instance, as it will need to have the ability to create location elements. This user account must be a local account on the XIQ instance or have external access. To use an External XIQ account, a flag must be added when running the script. See the -external flag under the Arguments Section for more details.

Disclaimer: SAML accounts are currently not supported by our API.
The script does support hardcoding an API access token between the single quotes on line 18. When the token is added to the script, the script will bypass asking for the XIQ username and password and attempt to use the provided token.
For information on generating an API token, see the **Generating Specific Tokens** section in the A Guide to Getting Started with v2 APIs in XIQ Article. Note that the needed permissions for this script include locations, device, and device:list permissions. It is recommended to add the auth:r permission as well to view information about the token later, but this is not required for the script.

WiNG Tech Dump File:

Information:

The XIQ_Wing_migrate.py script will use the rf-domain information from the tech dump file to create the needed site, building, and floors in XIQ. The script will replicate the WiNG rf-domain System tree hierarchy in XIQ.

If a site group, site, building, or floor exists in XIQ with the same name used in the WiNG Tech dump, that location element may be used instead of creating a new element. The script may also prompt for a new name to be entered.

Buildings:

A building will be created in XIQ for each rf-domain with associated devices in the tech dump. If no devices are associated with the rf-domain, that building (rf-domain) will not be created in XIQ.

Site Groups and Sites:

Rf-domains can contain a tree-node, allowing for location hierarchy of the rf-domain. WiNG allows four elements to be added – Country, Region, City, and Campus. XIQ also allows 4 elements for location hierarchy. 3 optional Site Groups and a **required** site.
This script will leverage the information in the tree-node for the rf-domain if available. The script will take the last element of the tree-node as the site and the elements above as Site-Groups, if applicable.

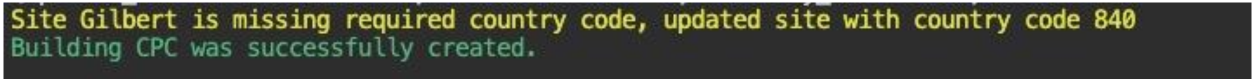
In XIQ, each location will need to have its own unique name. This was not the case with WiNG. The script will try to append the parent's name to the child to handle this. For example, if the same city name is used with different regions (or states) like Portland, Maine, and Portland, Oregon, the first city in the config will remain unchanged. The second city would become

Portland_Oregon. If this change does occur, there will be a message in the log about the change.

If any location already exists in XIQ with any of these names, the script will try to use that location even if the hierarchy above does not match. The script may also give you options to change the location's name. If a Site is found but the Site doesn't have a country code assigned (created before 23r7) the script will update the existing site with the country code found in the rf-domain (building) that will be created under the site. XIQ does not allow you to create a building under a site missing a country code.

>Note: Once Site groups, sites, and buildings are created in XIQ, you can move them around if needed.

XIQ now requires that each site has a country code assigned to it. If an existing site does not have a country code set, the script will update the site with the country code from the rf-domain before creating the building. XIQ will not allow buildings to be added to sites with missing country codes. If this occurs, a message will print stating what country code was set.



```
Site Gilbert is missing required country code, updated site with country code 840
Building CPC was successfully created.
```

>Note: If any an address was set for the site when created, the update will remove the address and leave it blank.

Floors:

Any floors created in an rf-domain will be added to XIQ. This config would be lines with *layout area xxxx floor xxx* in the rf-domain. The floor will also be created based on floors configured in any devices associated with the rf-domain. This config would be lines with *floor "xxxx"* in the device config. If neither of these configurations exists (floors not used in WiNG), then a **floor1** will be created, and all devices associated with the rf-domain will be placed there.

APs:

In XIQ, the APs will be assigned to the floor based on the rf-domain (building in XIQ) and the floor in the WiNG device config. The first floor pulled from the rf-domain config will be used if there is no floor in the device config. If no floors are configured in WiNG (neither in device config nor rf-domain config), the AP will be assigned the **floor1** floor created under the rf-domain.
Once all building and floors are created, the script will pull all XIQ devices that are not assigned a location. If any of these devices are found in the Tech Dump, the APs will be moved onto the floor as stated. If the device is not found in the Tech Dump, the script will print a note in the log that the device does not have a location set but was not found in the Tech Dump.

Running The Script:

To run the script, open the terminal to the location of the script and run the following:

```
Python3 XIQ_wing_migrate.py
```

You can also make the script executable by running

```
chmod +x XIQ_wing_migrate.py
```

Then you can run the script by typing

```
./XIQ_wing_migrate.py
```

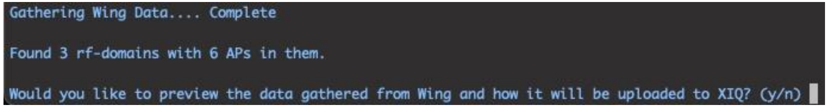
The script will ask you to enter a Wing tech-dump file. You can either enter the file's name, including the full path, or drag the file into the terminal in macOS and hit enter. You can also drag the file in Windows 11, but you will have to remove the added '&' at the beginning of the file name.

The script will collect the needed information from the Wing Tech dump file.

- **Note:** This process can take a few minutes, depending on the size of your config.

Once complete, the script will display how many rf-domains and APs were found.

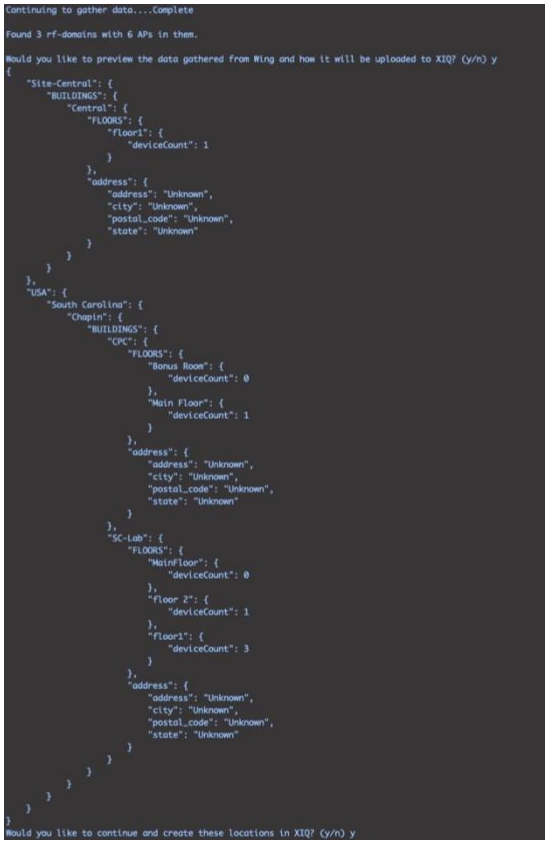
You will then be asked if you would like to preview the data gathered



```
Gathering Wing Data.... Complete

Found 3 rf-domains with 6 APs in them.

Would you like to preview the data gathered from Wing and how it will be uploaded to XIQ? (y/n)
```

.

If you choose to preview, the script will print a JSON-type file where you can see the discovered hierarchy. Here you can see how the data will be added to XIQ. Buildings will be added under a BUILDINGS object. Each building will be an object with a FLOORS object listing the floors associated with it as well as the address of the building. You will also see the device count for each floor.

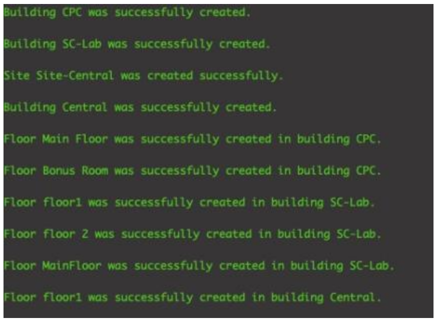


```
Continuing to gather data....Complete

Found 3 rf-domains with 6 APs in them.

Would you like to preview the data gathered from Wing and how it will be uploaded to XIQ? (y/n) y
{
    "Site-Central": {
        "BUILDINGS": {
            "Central": {
                "FLOORS": {
                    "floor1": {
                        "deviceCount": 1
                    }
                },
                "address": {
                    "address": "Unknown",
                    "city": "Unknown",
                    "postal_code": "Unknown",
                    "state": "Unknown"
                }
            }
        }
    },
    "USA": {
        "South Carolina": {
            "Chapin": {
                "BUILDINGS": {
                    "CPC": {
                        "FLOORS": {
                            "Bonus Room": {
                                "deviceCount": 0
                            },
                            "Main Floor": {
                                "deviceCount": 1
                            }
                        },
                        "address": {
                            "address": "Unknown",
                            "city": "Unknown",
                            "postal_code": "Unknown",
                            "state": "Unknown"
                        }
                    },
                    "SC-Lab": {
                        "FLOORS": {
                            "MainFloor": {
                                "deviceCount": 0
                            },
                            "floor 2": {
                                "deviceCount": 1
                            },
                            "floor1": {
                                "deviceCount": 3
                            }
                        },
                        "address": {
                            "address": "Unknown",
                            "city": "Unknown",
                            "postal_code": "Unknown",
                            "state": "Unknown"
                        }
                    }
                }
            }
        }
    }
}
Would you like to continue and create these locations in XIQ? (y/n) y
```

Once complete, the script will ask for your XIQ username and password.

- **Note:** The User account will need an administrator role to create the location, buildings, and floors.
- **Note:** If an API access token is added to the script, the script will bypass asking for your XIQ username and password and create the Location elements.

Site Groups, sites, buildings, floors & floorplans



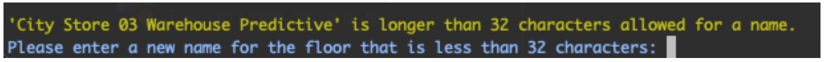
```
Building CPC was successfully created.

Building SC-Lab was successfully created.

Site Site-Central was created successfully.

Building Central was successfully created.

Floor Main Floor was successfully created in building CPC.

Floor Bonus Room was successfully created in building CPC.

Floor floor1 was successfully created in building SC-Lab.

Floor floor 2 was successfully created in building SC-Lab.

Floor MainFloor was successfully created in building SC-Lab.

Floor floor1 was successfully created in building Central.
```

Messages will appear in the terminal window as site groups, sites, buildings, and floors are created.
The script will first check if a building exists with the same name. If that building (rf-domain) exists in XIQ, the script will use that building and ignore the location hierarchy.
If the building does not exist, the script will check if a site with the same name exists. If so, the script will create the building (rf-domain) under the existing site, ignoring any other elements of the tree-node.
If the site is not found, the script will attempt to check Site Groups following the hierarchy; if they are not found, the script will attempt to create the site groups. Once completed, the script will proceed with creating the site and building.

- **Note**: XIQ requires each site group, site, and building to have a unique name. Floors within a building also must have their unique name.
- **Note:** Currently, the script does not collect the whole location tree. If it tries to create a site group or site and detects a duplicate name for a different element type, it will prompt for a new name for the element.

Once all buildings are found or created, the script will create all the floor elements.

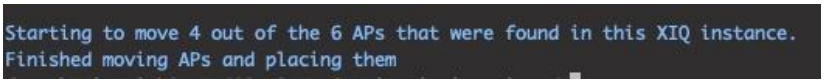
If a name is longer than 32 characters, the script will print the name with a message that it is longer than 32 characters and allow you to enter a new name.



```
'City Store 03 Warehouse Predictive' is longer than 32 characters allowed for a name.
Please enter a new name for the floor that is less than 32 characters:
```

Access Points

After creating all the locations, buildings, and floors, the script will move any WiNG APs found in the tech dump and XIQ to the correct floors. A message will display the count of APs found vs. those in the WiNG Tech dump. The messages will also inform that the script is moving and when it finishes.



```
Starting to move 4 out of the 6 APs that were found in this XIQ instance.
Finished moving APs and placing them
```

Arguments

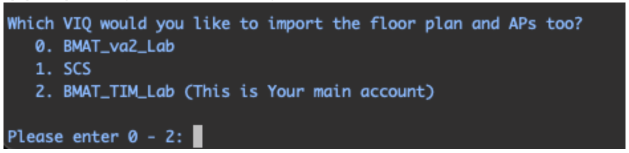
Three optional arguments can be added when running the script.

--external

This flag allows you to import the WiNG locations, buildings, and floors into a VIQ on which you are an external user.
To run the script on an externally managed VIQ in the terminal, you would run.

```
Python3 XIQ_wing_migrate.py --external
```

The script will start as usual, request a WiNG Tech-dump file, and then ask for your XIQ login. After logging in, you will be presented with a choice of which VIQ you want to import into.



```
Which VIQ would you like to import the floor plan and APs too?
   0. BMAT_va2_Lab
   1. SCS
   2. BMAT_TIM_Lab (This is Your main account)

Please enter 0 - 2:
```

--noaplog

This flag will suppress the log messages normally created when devices do not have the floor assigned to them in their config. If you typically do not assign a floor to the device, the log file could fill up with warning messages about APs not being set to a floor and the script assigning them to 'floor1'.

```
Python3 XIQ_wing_migrate.py --noaplog
```

--nogeolog

This flag will suppress the log messages, and the print screen that is normally created with the API for Geolocation is missing; the building addresses will be missing for the rf-domain.

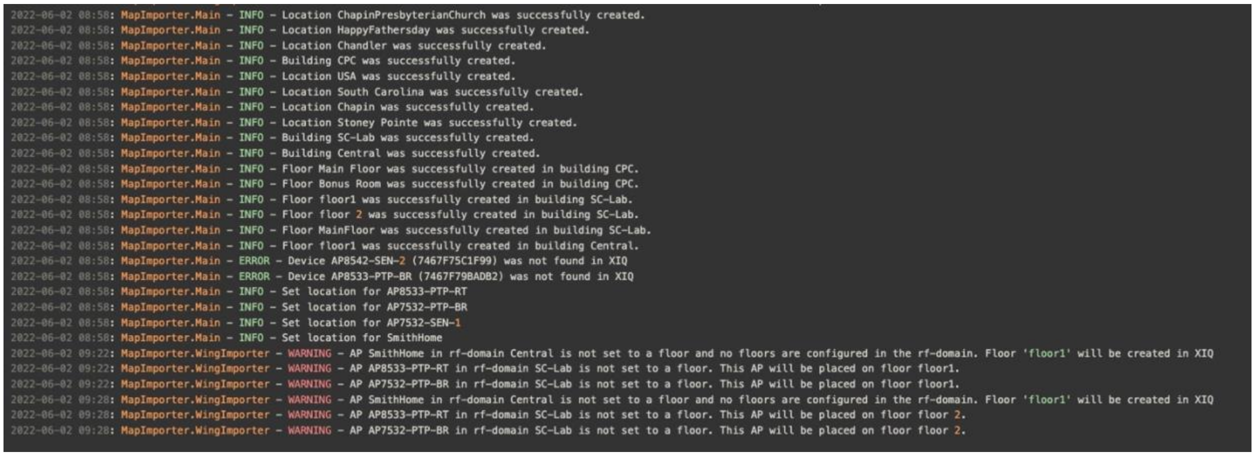
```
Python3 XIQ_wing_migrate.py --nogeolog
```

All combinations of flags can be added if needed.

```
Python3 XIQ_wing_migrate.py –noaplog --nogeolog
```

Log File

Upon running the script, a log file named map_importer.log will be created. This log file contains the same information printed on the screen. It is also an excellent place to check if any issues arise, as there is sometimes more information in the log file. The script will keep appending to the log file until it reaches 50MB. Then, the file will be rotated out. The script will allow for 5 files before deleting any old files.



```
2022-06-02 08:58: MapImporter.Main - INFO - Location ChapinPresbyterianChurch was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Location HappyFathersday was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Location Chandler was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Building CPC was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Location USA was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Location South Carolina was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Location Chapin was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Location Stoney Pointe was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Building SC-Lab was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Building Central was successfully created.
2022-06-02 08:58: MapImporter.Main - INFO - Floor Main Floor was successfully created in building CPC.
2022-06-02 08:58: MapImporter.Main - INFO - Floor Bonus Room was successfully created in building CPC.
2022-06-02 08:58: MapImporter.Main - INFO - Floor floor1 was successfully created in building SC-Lab.
2022-06-02 08:58: MapImporter.Main - INFO - Floor floor 2 was successfully created in building SC-Lab.
2022-06-02 08:58: MapImporter.Main - INFO - Floor MainFloor was successfully created in building SC-Lab.
2022-06-02 08:58: MapImporter.Main - INFO - Floor floor1 was successfully created in building Central.
2022-06-02 08:58: MapImporter.Main - ERROR - Device AP8542-SEN-2 (7467F75C1F99) was not found in XIQ
2022-06-02 08:58: MapImporter.Main - ERROR - Device AP8533-PTP-BR (7467F79BADB2) was not found in XIQ
2022-06-02 08:58: MapImporter.Main - INFO - Set location for AP8533-PTP-RT
2022-06-02 08:58: MapImporter.Main - INFO - Set location for AP7532-PTP-BR
2022-06-02 08:58: MapImporter.Main - INFO - Set location for AP7532-SEN-1
2022-06-02 08:58: MapImporter.Main - INFO - Set location for SmithHome
2022-06-02 09:22: MapImporter.WingImporter - WARNING - AP SmithHome in rf-domain Central is not set to a floor and no floors are configured in the rf-domain. Floor 'floor1' will be created in XIQ
2022-06-02 09:22: MapImporter.WingImporter - WARNING - AP AP8533-PTP-RT in rf-domain SC-Lab is not set to a floor. This AP will be placed on floor floor1.
2022-06-02 09:22: MapImporter.WingImporter - WARNING - AP AP7532-PTP-BR in rf-domain SC-Lab is not set to a floor. This AP will be placed on floor floor1.
2022-06-02 09:28: MapImporter.WingImporter - WARNING - AP SmithHome in rf-domain Central is not set to a floor and no floors are configured in the rf-domain. Floor 'floor1' will be created in XIQ
2022-06-02 09:28: MapImporter.WingImporter - WARNING - AP AP8533-PTP-RT in rf-domain SC-Lab is not set to a floor. This AP will be placed on floor floor 2.
2022-06-02 09:28: MapImporter.WingImporter - WARNING - AP AP7532-PTP-BR in rf-domain SC-Lab is not set to a floor. This AP will be placed on floor floor 2.
```