

# ONSTAK VA CLOUD DATA DOCUMENTATION

This document elaborates how to get data from VA platform cloud.

## Step1:

open VA code directory in local system and activate anaconda environment

## Step2:

Run this command `python get_va_cloud_data.py -help`

There is an optional argument naming “-cfg”. This argument take value as name of json file like “config.json”. The default name is “config.json” in the code. If you have json file with some other names like “config\_v2.json” or “config\_v1.json” you have to pass -cfg argument with your spacificed name of json file. e.g

`python get_va_cloud_data.py -cfg “config_v2.json”`

if your json file name is already “config.json” there is no need to pass -cfg argument.

## Step3:

There are two positional argument {camera, model}. To get information about cameras you have to pass camera as argument. To get information about deployed models you have to pass model argument like

`python get_va_cloud_data.py camera`

`python get_va_cloud_data.py model`

## Camera

In Camera there are four sub argemnts

1. -camera\_list
2. -camera\_info
3. -rtsp
4. -camera\_csv

### 1. -camera\_list

camera\_list shows the list of all cameras names

`python get_va_cloud_data.py camera -camera_list`

```
Thermal Basement Camera  
test  
IOT_CAM_2  
CISCO CAM MQTT  
Cisco Cam RTSP  
Cisco Meraki DL  
Cisco Meraki  
Data Center  
Developers Space  
Basement  
IoT Hall  
Ground Floor
```

## 2. -camera\_info

this argument take a value as name of the camera and show complete json of that camera. You pass the argument as

```
python get_va_cloud_data.py camera -camera_info "Developers Space"
```

e.g

```
{'id': 6, 'name': 'Developers Space', 'location': '34.0110647,-118.4107829', 'description': '', 'rtspHost': '10.39.110.246', 'rtspPort': 554, 'rtspPath': '/ch1/main/av_stream.h264', 'rtspUsername': 'dl-team', 'rtspPassword': 'dl@Onstak123', 'image': '6_2.jpg', 'macAddress': None, 'serialKey': None, 'networkId': None, 'ciscoMerakiCameraName': '', 'thermalPort': None, 'createdBy': 'akhter.ali', 'updatedBy': 'amir.nadeem', 'createdAt': '2021-11-01T15:13:43.000Z', 'updatedAt': '2022-09-26T08:32:45.000Z', 'cameraStreamType': 1, 'cameraRTSPInputType': 1, 'make': None, 'model': None, 'accessMethod': None, 'integrationDetailId': None, 'tenantId': 2, 'DLModels': [], 'integration_detail': None}
```

## 3. -rtsp

This argument create rtsp of the camera. You to pass camera name as a value

```
python get_va_cloud_data.py camera -rtsp "Developers Space"
```

## 4. -camera\_csv

This argument create a complete csv of all camera available on tenant with their rtps

```
python get_va_cloud_data.py camera -camera_csv
```

# Model

you can get model argument help as

```
python get_va_cloud_data.py model --help
```

In model there are three sub argument

1. -model\_list
2. -mode\_info
3. -model\_csv

## 1. -model\_list

This will list all the deployed models on the tenant

```
python get_va_cloud_data.py model -model_list
```

just like

```
Mask Detection
Social Distance
Crowd
Intrusion
Facial Recognition
Age & Gender
Emotions
Occupancy
NLP
Vehicles
Weapon
Pose Detection
License Plates Recognition
People Count
Person Fall Detection
```

## 2. -mode\_info

This will model name as value to display complete json of that model

```
python get_va_cloud_data.py model -model_info "Occupancy"
```

```
{'id': 8, 'name': 'Occupancy', 'dlObjects': [{'name': 'Person', 'attributes': [{'key': 'occupancydetected', 'values': ['true', 'false', 'number']}]}], 'description': 'Video analytics occupancy detection', 'image': '1654251858511-792990250.jpg', 'shortDescription': None, 'displayName': 'Occupancy Detection', 'createdAt': '2020-07-01T12:00:00.000Z', 'updatedAt': '2022-06-03T10:24:18.000Z'}
```

## 3. -model\_csv

This argument will create and save csv of all the models available on the tenant.

```
python get_va_cloud_data.py model -model_csv
```