

## Sample script for calling GetExtendedDevice API

Python script for ASP clients to make an authorized call to DAM GetExtendedDevice API. Consider this as reference and make proper change to this script

For example, put correct value for FILE\_PATH and BASE\_URL.

```
import math
import json
import jwt
import requests
import sys
import time

# Replace this with relative file path to the JSON file that contains
# private key.
FILE_PATH = "experimental/users/azarkabeer/key.json"
# Use one of the following depend on environment.
# UAT: "https://staging-hardwaresupplychain.sandbox.googleapis.com"
# PROD: "https://hardwaresupplychain.googleapis.com"
BASE_URL = "https://staging-hardwaresupplychain.sandbox.googleapis.com"

def generate_jwt():
    file_content = json.loads(open(FILE_PATH).read())
    iat = int(time.time())
    exp = iat + 60
    payload = {
        "iss": file_content['client_email'],
        "scope": "https://www.googleapis.com/auth/supplychain",
        "aud": "https://oauth2.googleapis.com/token",
        "iat": iat,
        "exp": exp
    }
    additional_headers = {
        "alg": "RS256",
        "typ": "JWT"
    }
```

```

    key = "\n".join([l.lstrip() for l in
file_content['private_key'].split("\n")])
    signed_jwt = str(jwt.encode(payload, key, headers=additional_headers,
algorithm='RS256'))
    return signed_jwt

def get_access_token(jwt):
    url = "https://oauth2.googleapis.com/token"
    data = {
        "grant_type": "urn:ietf:params:oauth:grant-type:jwt-bearer",
        "assertion": jwt
    }
    res = requests.post(url, data=data)
    return res.json().get("access_token")

def call_get_device(device_id, access_token):
    headers = {
        "Authorization": "Bearer " + access_token,
    }
    url = BASE_URL + "/v1/devices/" + device_id + "?view=DEVICE_AND_WARRANTY"
    res = requests.get(url, headers=headers)
    if res.reason == "Not Found":
        print("There is no device with device_id: " + device_id)
        return ""
    return res.text

def call_get_extended_device(device_id, access_token):
    headers = {
        "Authorization": "Bearer " + access_token,
    }
    url = BASE_URL + "/v1/devices/" + device_id + ":extendedDeviceInfo"
    params = dict(device_id=device_id, scopes="ITEM")
    res = requests.get(url, headers=headers, params=params)
    if res.reason == "Not Found":
        print("There is no device with device_id: " + device_id)
        return ""
    return res.text

if __name__ == "__main__":
    signed_jwt = generate_jwt()
    access_token = get_access_token(signed_jwt)
    device = call_get_extended_device(sys.argv[1], access_token)

```

```
print(device)
```

In the script, `call_get_device()` method is the one, team is using currently. The new method `call_get_extended_device()` is a sample implementation for calling the new `GetExtendedDevice` API. A sample response from the new API can be found below.

#### **Sample Response while testing the script**

```
azarkabeer@azarkabeer /google/src/cloud/azarkabeer/Dam2/google3 python3
experimental/users/azarkabeer/getDevices.py 02AA01AB511205F1
{
  "device": {
    "name": "devices/02AA01AB511205F1",
    "serialNumber": "02AA01AB511205F1",
    "itemNumber": "T200577",
    "rmaItemNumber": "T200577-R",
    "manufactureDate": {},
    "latestActivationDate": {
      "year": 2013,
      "month": 7,
      "day": 31
    },
    "firstActivationDate": {
      "year": 2013,
      "month": 7,
      "day": 31
    },
    "originalItemNumber": "T200577",
    "latestNonRmaItemNumber": "T200577"
  },
  "item": {
    "productKeyName": "NLT_2nd Gen_Stainless Steel",
```

```
"description": "Nest Learning Thermostat, US",
"deviceCondition": "PRISTINE",
"modelName": "T200577",
"color": "Stainless Steel",
"weight": {
  "unit": "GRAM",
  "value": 252
},
"height": {
  "unit": "MILLIMETER",
  "value": 32
},
"length": {
  "unit": "MILLIMETER",
  "value": 83
},
"width": {
  "unit": "MILLIMETER",
  "value": 83
},
"productAreaName": "Energy",
"productLineName": "Nest Learning Thermostat_2nd Gen",
"productType": "Thermostat",
"isBundle": "BUNDLE_ITEM_FALSE"
}
}
```

azarkabeer@azarkabeer /google/src/cloud/azarkabeer/Dam2/google3