

# ThingPlug API for Python

Ver. 1.0

IoT 오픈 하우스

©2017 IoT 오픈 하우스 All Rights Reserved.

☞ for more information, visit our website at <https://lora.sktiot.com>

---

## 1. ThingPlug API for Python

아래 주소는 ThingPlug를 사용하기 위한 Python API를 제공하며, 해당 API를 사용하기 위한 예제를 포함하고 있다.

<https://github.com/kaizen8501/thingplugAPI>

### 1.1. 환경 설정

Step 1. 서드파티 모듈 설치하기

```
$ pip install -r requirements.txt
```

Step 2. Example 실행 하기

Example 폴더에는 ThingPlugAPI를 이용하여 ThingPlug의 기능을 사용하기 위한 예제들이 존재 하며 각각의 예제를 사용하기 위한 방법은 2. Examples를 참고 하기 바란다.

## 2. Examples

### 2.1. ThingPlug 계정의 uKey 가져 오기

login.py는 ThingPlug에 접속하기 위한 User ID와 User Password를 이용하여 uKey를 요청하는 예제 이다.

uKey란 ThingPlug의 Northbound (ThingPlug API to Application Server) API 접근 권한을 위한 Access Token으로 해당 uKey는 ThingPlug 홈페이지의 회원 가입 시 발급 가능 하다.

```
$ python login.py --help
usage: login.py [-h] -u USER_ID -p USER_PW [-th THINGPLUG_HOST]
               [-tp THINGPLUG_PORT]

ThingPlug Login Example

optional arguments:
  -h, --help            show this help message and exit
  -u USER_ID, --user_id USER_ID
                        ThingPlug User ID
  -p USER_PW, --user_pw USER_PW
                        ThingPlug User Password
  -th THINGPLUG_HOST, --thingplug_host THINGPLUG_HOST
                        ThingPlug Host IP(Default:onem2m.sktiot.com)
  -tp THINGPLUG_PORT, --thingplug_port THINGPLUG_PORT
                        ThingPlug Port(Default:9000)
```

```
$ python login.py -u USER_ID -p USER_PASSWORD
INFO:Login Success
ZG9qcUNscTA3RlhLZ.....
```

## IoT 오픈 하우스

테스트 망이 아닌 상용망(thingplugpf.sktiot.com)에 연결하기 위해서는 아래와 같이 THINGPLUG\_HOST와 THINGPLUG\_PORT를 파라미터로 입력하면 된다. THINGPLUG\_HOST와 THINGPLUG\_PORT를 별도로 입력하지 않으면, Default 값으로 onem2m.sktiot.com과 9000번 포트를 이용한다.

```
$ python login.py -u USER_ID -p USER_PASSWORD -th thingplugpf.sktiot.com -tp 9000
```

## 2.2. 등록된 Device List 가져오기

get\_device\_list.py는 ThingPlug의 계정에 등록되어 있는 Device들의 List를 요청하는 예제이다.

```
$ python get_device_list.py --help
usage: get_device_list.py [-h] -u USER_ID -p USER_PW [-th THINGPLUG_HOST]
                        [-tp THINGPLUG_PORT]
```

ThingPlug Login Example

optional arguments:

```
-h, --help            show this help message and exit
-u USER_ID, --user_id USER_ID
                        ThingPlug User ID
-p USER_PW, --user_pw USER_PW
                        ThingPlug User Password
-th THINGPLUG_HOST, --thingplug_host THINGPLUG_HOST
                        ThingPlug Host IP(Default:onem2m.sktiot.com)
-tp THINGPLUG_PORT, --thingplug_port THINGPLUG_PORT
                        ThingPlug Port(Default:9000)
```

아래와 같이 명령을 수행하면 내 계정에 등록되어 있는 Device의 개수와 Device ID를 확인할 수 있다.

```
$ python get_device_list.py -u USER_ID -p USER_PASSWORD
INFO:Login Success
INFO:0008DC000000
INFO:0008DC000001
INFO:0008DC000002
(True, u'3', [u'0008DC000000', u'0008DC000001', u'0008DC000002'])
```

-th(THINGPLUG\_HOST)와 -tp(THINGPLUG\_PORT) 옵션은 login.py와 동일 하다.

## 2.3. Device 주기 보고의 최신 값 가져오기

get\_latest\_data.py는 Device의 주기보고로 저장된 값 중에서 가장 최신에 저장된 값을 확인하기 위한 예제 이다.

```
$ python get_latest_data.py --help
usage: get_latest_data.py [-h] -u USER_ID -p USER_PW -ni NODE_ID -ct
CONTAINER
                        -ae APP_EUI [-th THINGPLUG_HOST]
                        [-tp THINGPLUG_PORT] [-pt PERIOD_TIME]
                        [-el ENABLE_LOG]
```

ThingPlug Login Example

optional arguments:

```
-h, --help            show this help message and exit
-u USER_ID, --user_id USER_ID
                        ThingPlug User ID
-p USER_PW, --user_pw USER_PW
                        ThingPlug User Password
-ni NODE_ID, --node_id NODE_ID
                        ThingPlug Node ID
-ct CONTAINER, --container CONTAINER
                        ThingPlug Container Name
-ae APP_EUI, --app_eui APP_EUI
                        ThingPlug APP EUI
-th THINGPLUG_HOST, --thingplug_host THINGPLUG_HOST
                        ThingPlug Host IP(Default:onem2m.sktiot.com)
-tp THINGPLUG_PORT, --thingplug_port THINGPLUG_PORT
                        ThingPlug Port(Default:9000)
-pt PERIOD_TIME, --period_time PERIOD_TIME
                        Get Latest Data Period Time(Default:0,One Time)
-el ENABLE_LOG, --enable_log ENABLE_LOG
```

## IoT 오픈 하우스

---

아래와 같은 명령을 수행하면 해당 USER\_ID에 등록되어 있는 DEVICE\_ID의 최신 주기보고 값을 확인 할 수 있다. 출력되는 메시지 포맷은 아래와 같습니다. <현재 시간>,<최신 주기보고 값>,<lt(lastModifiedTime)>

```
python get_latest_data.py -u USER_ID -p PASSWORD -ni DEVICE_ID -ct CONTAINER
-aE APP_EUI
INFO:Login Success
2017-03-15 09:21:45.860000,1001,20170314133418042
```

주기적으로 DEVICE\_ID에 대한 최신 주기보고 값을 확인 하려면, 아래와 같이 -pt 옵션을 사용하면 된다. (단위:초)

```
python get_latest_data.py -u USER_ID -p PASSWORD -ni DEVICE_ID -ct CONTAINER
-aE APP_EUI -pt 60
INFO:Login Success
2017-03-15 09:56:16.998000,1001,20170314133418042
2017-03-15 09:57:17.311000,1001,20170314133418042
2017-03-15 09:58:17.475000,1001,20170314133418042
```

-el 옵션을 이용하면 주기적으로 출력되는 값을 파일로 저장 할 수 있다. 사용 방법은 아래와 같다.

```
python get_latest_data.py -u USER_ID -p PASSWORD -ni DEVICE_ID -ct CONTAINER
-aE APP_EUI -pt 60 -el 1
```

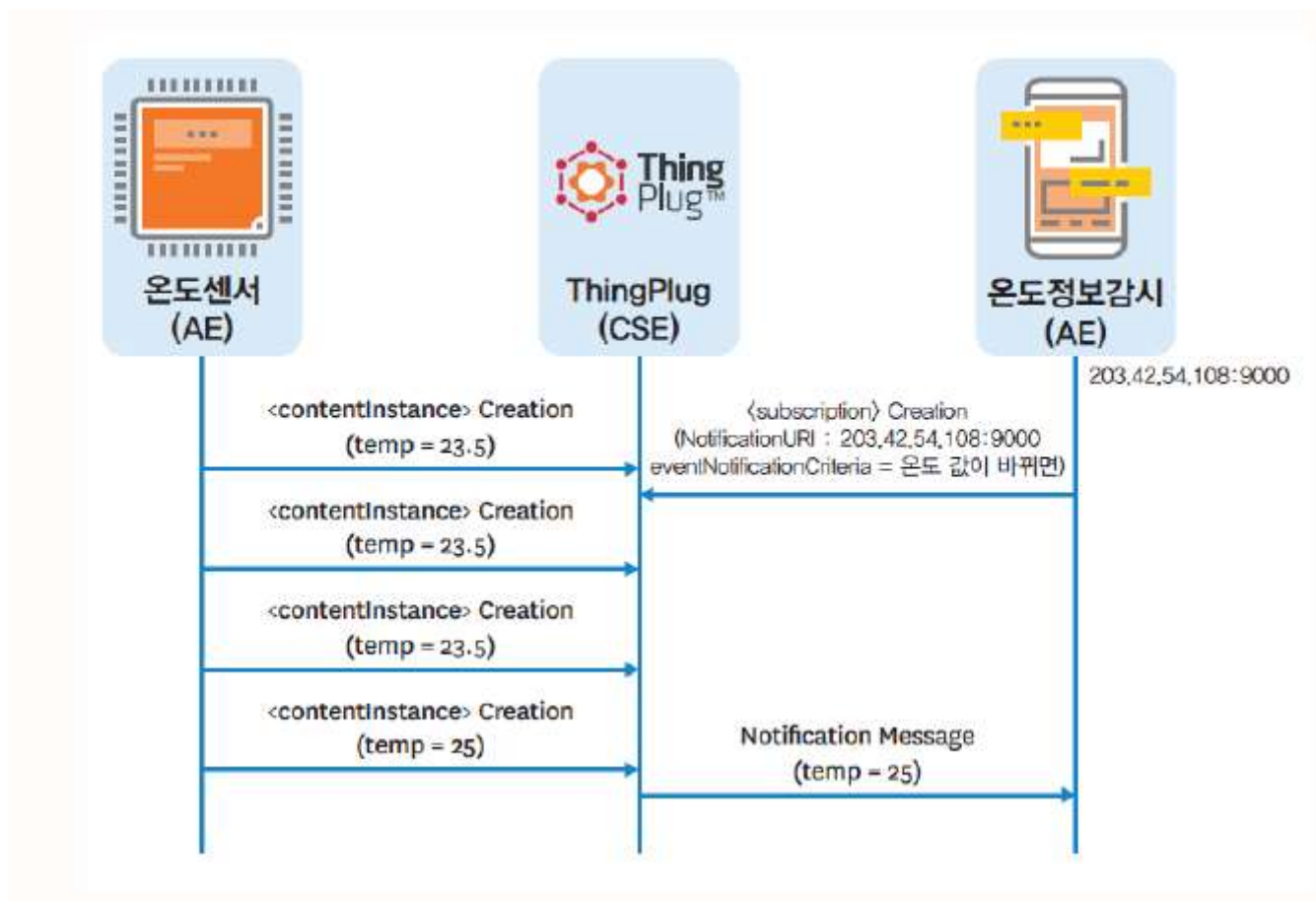
출력되는 메시지는 아래와 같다. 출력된 메시지는 get\_latest\_data.py 파일과 같은 경로의 get\_latest\_data.log 파일에 저장된다.

```
INFO:Login Success
2017-03-15 10:04:45.912000,1001,20170314133418042
```

## 2.4. Application Server에서 Push 메시지를 전달 받기

ThingPlug에서는 구독(Subscription) 및 통지(Notification)을 위한 API를 제공한다. 이는 마치 신문을 배달 받는 것과 같다. 어떤 신문을 구독하려면 신문에 연락해서 계약을 진행한다. 이런 행위를 구독이라고 한다. 이때 계약자는 배달 받을 주소와 배달 받는 신문의 종류 및 배달 주기 등을 전달하고, 계약에 따라 신문이 나오면 희망 배달지에 배달 즉, 통지되는 형태로 서비스가 진행된다. < ThingPlug\_API\_Document 참고 >

아래 그림과 같이, 온도정보를 감시하기 위한 장치에서 ThingPlug에게 온도센서 장치의 값을 구독 하겠다고 요청 할 수 있다. 구독 요청이 완료되면, ThingPlug에서는 온도센서 장치의 주기보고 값을 모니터링 하고 있다가 온도 값이 변경하면 구독 요청 시 등록한 NotificationURI에게 온도 값을 전달 하는 구조로 동작 한다.



### 2.4.1. Subscription\_mqtt

디바이스에서 송신한 데이터 값이 변경 되거나 특정 조건이 발생할 때 통지(Notification) 메시지 또는 푸시(Push) 메시지를 ThingPlug 서버가 보내도록 설정 할 수 있다. subscription\_mqtt.py는 통지 메시지를 MQTT로 수신 할 수 있도록 ThingPlug를 설정하는 예제이며, 아래와 같은 옵션을 이용 할 수 있다. 통지 메시지를 수신 하기 위한 방법은 MQTT, HTTP 두가지 방법이 있다. 단 HTTP 방식으로 메시지를 수신 할 경우, 장치의 IP 주소는

## IoT 오픈 하우스

공인 IP가 아닌 환경이거나 유동 IP를 사용하는 경우 별도의 설정이 필요한 제약이 있다.

```
python subscription_mqtt.py --help
```

```
usage: subscription_mqtt.py [-h] -u USER_ID -p USER_PW -ae APP_EUI
                             [-ni NODE_ID] [-ct CONTAINER] [-th THINGPLUG_HOST]
                             [-tp THINGPLUG_PORT] [-ci MQTT_CLIENT_ID]
```

ThingPlug Login Example

optional arguments:

```
-h, --help            show this help message and exit
-u USER_ID, --user_id USER_ID
                        ThingPlug User ID
-p USER_PW, --user_pw USER_PW
                        ThingPlug User Password
-ae APP_EUI, --app_eui APP_EUI
                        ThingPlug APP EUI
-ni NODE_ID, --node_id NODE_ID
                        ThingPlug Node ID
-ct CONTAINER, --container CONTAINER
                        ThingPlug Container Name(Default:LoRa)
-th THINGPLUG_HOST, --thingplug_host THINGPLUG_HOST
                        ThingPlug Host IP(Default:onem2m.sktiot.com)
-tp THINGPLUG_PORT, --thingplug_port THINGPLUG_PORT
                        ThingPlug Port(Default:9000)
-ci MQTT_CLIENT_ID, --mqtt_client_id MQTT_CLIENT_ID
                        ThingPlug MQTT Client ID(Default:bridge)
```

```
$ python subscription_mqtt.py -u USER_ID -p PASSWORD -ae APP_EUI -ct
CONTAINER
```

```
INFO:Login Success
```

```
INFO:000000123456789123456789
```

```
thingplug_00000037d02544ffffef04f10
```

```
INFO:registered subscription
```

```
INFO:subscription is deleted
```

```
INFO:subscription is created
```

```
INFO:mqtt connected
```

```
2017-03-15 12:34:01.040000,1123,2017-03-15T12:34:03+09:00
```

```
2017-03-15 12:34:31.466000,1125,2017-03-15T12:34:34+09:00
```