

## Table of contents

1	Kankun wifi plug 활용을 위해 OpenHAB2 + MQTT 설치하기(Ubuntu 12.04)	2
1.1	Zulu	2
1.1.1	Installation ( <a href="http://zulu.org/zulu-community/zuludocs">http://zulu.org/zulu-community/zuludocs</a> )	2
1.1.2	Additional setups	2
1.1.3	Choose JDK	2
1.2	OpenHAB2	2
1.2.1	설치( <a href="http://docs.openhab.org/installation/linux.html#package-repository-installation">http://docs.openhab.org/installation/linux.html#package-repository-installation</a> )	2
1.2.2	서비스 제어	2
1.2.3	초기 설정	2
1.3	MQTT	2
1.3.1	설치	3
1.3.2	기본 테스트	3
1.4	OpenHAB2 + MQTT기반 Switch 연동	3
1.4.1	/etc/openhab2/services/mqtt.cfg	3
1.4.2	/etc/openhab2/items/switch.items	3
1.4.3	/etc/openhab2/sitemaps/Test.sitemap	3
1.4.4	안드로이드폰에 OpenHAB 설치 설치	3
1.4.5	안드로이드에서 테스트	5
1.4.6	MQTT 메시지를 가지고 테스트(HW 스위치 시뮬레이션)	5
1.4.7	MQTT 사용자 인증 / 토픽 접근 제어	5
1.4.8	테스트 해보자.	5
1.4.9	테스트 해보자.	5
1.4.10	OpenHAB2의 mqtt.cfg 변경	5

## 1 Kankun wifi plug 활용을 위해 OpenHAB2 + MQTT 설치하기(Ubuntu 12.04)

이번 글은 우분투 12.04를 기준으로 OpenHAB2와 Mosquitto를 설치해서 MQTT 바인딩을 통해 가상의 스위치의 상태를 보고 제어하는 과정이다. 궁극적으로는 실제 센서나 원격제어 가능한 스위치에 사용할 수 있는 방법으로써 OpenHAB을 어떻게 사용할 수 있는지, MQTT를 가지고 어떻게 HW등과 연동하는지 알 수 있는 기회가 되었으면 한다. 추후 Kankun wifi plug를 위한 준비 과정이다. 참 길다 π.π

### 1.1 Zulu

다음은 OpenHAB2에서 권장하는 Open JDK계열의 Zulu를 설치하고 설정하는 과정이다.

#### 1.1.1 Installation (<http://zulu.org/zulu-community/zuludocs>)

```
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 0x219BD9C9
sudo apt-add-repository 'deb http://repos.azulsystems.com/ubuntu stable main'
sudo apt-get update
sudo apt-get install zulu-8
```

#### 1.1.2 Additional setups

cd ~ 한 후 vi .profile해서 다음을 추가해준다.

```
export JAVA_HOME=/usr/lib/jvm/zulu-8-amd64/
export PATH=$JAVA_HOME/bin:$PATH
```

그 후 source .profile해주면 된다.

#### 1.1.3 Choose JDK

혹시 이미 JDK가 설치되어있는 경우에는 sudo update-alternatives --config java해서 선택해줘야 한다.

## 1.2 OpenHAB2

### 1.2.1 설치(<http://docs.openhab.org/installation/linux.html#package-repository-installation>)

```
wget -qO - 'https://bintray.com/user/downloadSubjectPublicKey?username=openhab' | sudo apt-key add - echo 'deb
↪ http://dl.bintray.com/openhab/apt-repo2 stable main' | sudo tee /etc/apt/sources.list.d/openhab2.list
sudo apt-get update
sudo apt-get install openhab2 openhab2-addons
```

### 1.2.2 서비스 제어

- sudo update-rc.d openhab2 defaults
- sudo /etc/init.d/openhab2 start
- sudo /etc/init.d/openhab2 stop
- Restart 명령의 경우 서비스가 제대로 종료안되는 이슈가 있으므로 stop 후 start해야 함.

### 1.2.3 초기 설정

- sudo /etc/init.d/openhab2 start
- 웹브라우저에서 [http://openhabW\\_serverW\\_ip:8080](http://openhabW_serverW_ip:8080) 접속
- Standard package setup 클릭해서 UI 설치
- Paper UI 선택
- 우측 상단 메뉴 > Addons > Bindings tab에서 MQTT binding 선택 후 설치

## 1.3 MQTT

가벼운 메시징 프로토콜로서 센서 등과 같은 HW에서 사용하기 좋은 통신 방법이다. HTTP바인딩과 달리, 메시징 서버(=브로커)가 외부접근 가능하면, 방화벽 등의 상관없이 통신이 가능하여 선택하였다. 사실 로컬네트워크안에 라즈베리파이를 가지고 MQTT 서버(=브로커)를 운영해도 괜찮지만, 이미 가지고 있는 \$5 짜리 DigitalOcean 가상서버에 OpenHAB2와 MQTT서버를 운영함으로써, 방화벽 등의 문제도 피해갈 수 있으리라는 판단이었다. (사실 인터넷망이 공인 ip를 주지 않는 서비스라서 π.π)

### 1.3.1 설치

```
sudo apt-get install python-software-properties
sudo apt-add-repository ppa:mosquitto-dev/mosquitto-ppa
sudo apt-get update
sudo apt-get install mosquitto mosquitto-clients
```

### 1.3.2 기본 테스트

- 터미널 1: `mosquitto_sub -h localhost -t /test_topic`
- 터미널 2: `mosquitto_pub -h localhost -t /test_topic -m 'Message from terminal 2'`
- Topic = 채팅방이라고 생각하면 됨

## 1.4 OpenHAB2 + MQTT기반 Switch 연동

현재 시점으로 MQTT binding의 경우 OpenHAB2의 UI에서 설정이 안되므로 1.x처럼 파일을 직접 편집해야 함

### 1.4.1 /etc/openhab2/services/mqtt.cfg

```
# mymqtt = MQTT broker name
mymqtt.url=tcp://localhost:1883
mymqtt.clientId=openHAB
mymqtt.qos=2
mymqtt.retain=true
mymqtt.async=true
```

### 1.4.2 /etc/openhab2/items/switch.items

```
# mqttsw1 : OpenHAB2내부에서 사용되는 id
# mqtt : mqtt type binding
# mymqtt : broker name
# /testsw/1/command : 제어용 MQTT 채널
# /testsw/1/state: 상태확인용 MQTT 채널
```

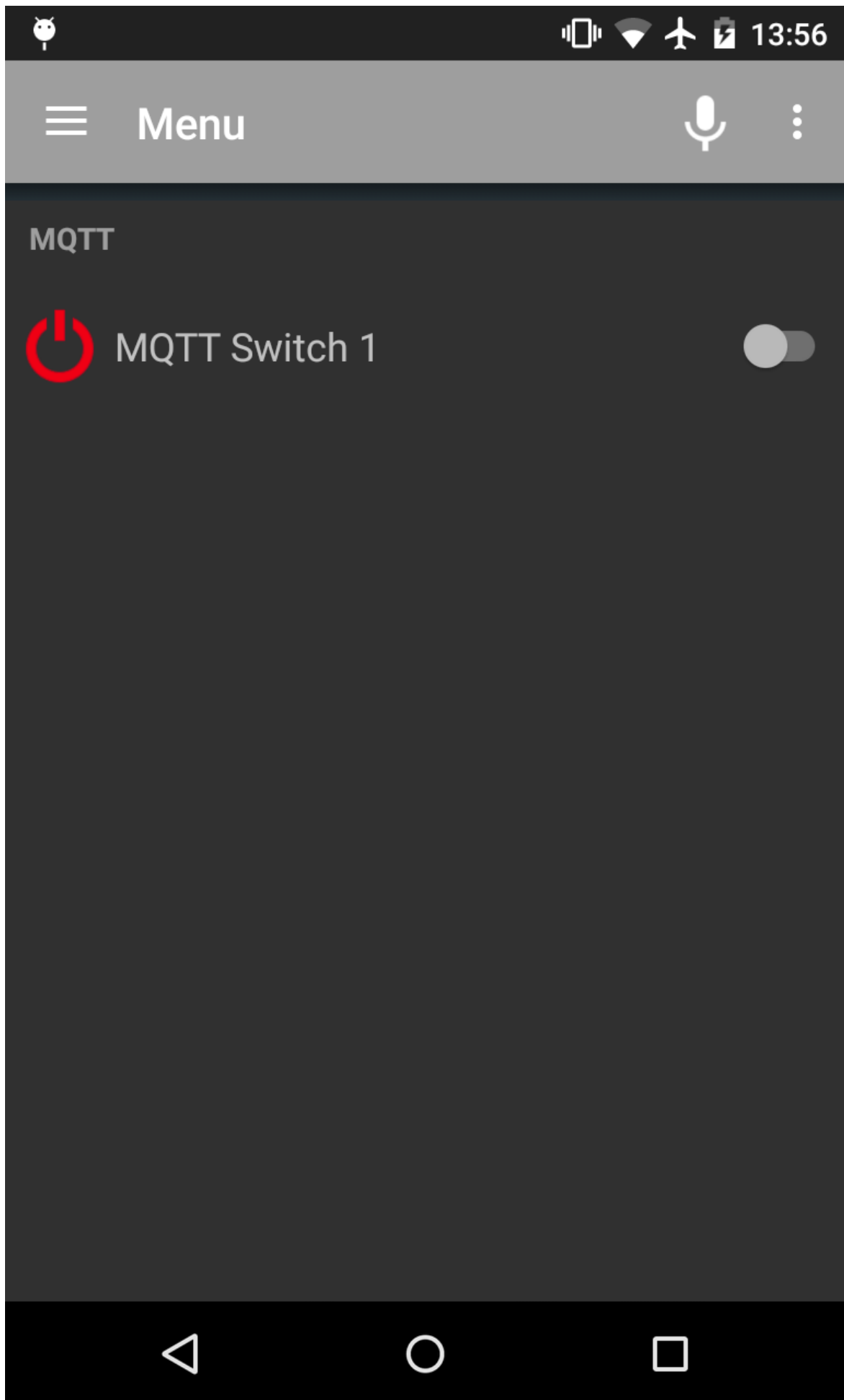
```
Switch mqttsw1 "Switch 1" (all) {mqtt=">[mymqtt:/testsw/1/command:command:ON:1],>[mymqtt:/testsw/1/command:command:OFF:0]
mymqtt:/testsw/1/state:state:ON:1],<[mymqtt:/testsw/1/state:state:OFF:0]}
```

### 1.4.3 /etc/openhab2/sitemaps/Test.sitemap

```
# Test : sitemap name, file name과 일치해야 함
# mqttsw1: OpenHAB2내부에서 사용되는 id
# "MQTT Switch 1": UI에서 표시되는 레이블
sitemap Test label="Menu"
{
  Frame label="MQTT"
  {
    Switch item=mqttsw1 label="MQTT Switch 1"
  }
}
```

### 1.4.4 안드로이드폰에 OpenHAB 설치 설치

- 실행 후 처음 보는 선택창에서 `_default` 선택(의미없음)
- 우측 상단 버튼(...) 눌러서 Setting 진입 후 `demo mode uncheck` 한 후 openHAB URL에 [http://openhabW\\_serverW\\_ip:8080](http://openhabW_serverW_ip:8080) 입력 후 이전으로 돌아가기 버튼 누르면 지정해준 서버에 연결함
- 우측 상단 버튼(...) 눌러서 Select default sitemap 선택해서 위에서 생성해준 Test를 선택하면 다음처럼 제어버튼 및 현재 상태를 볼 수 있음



#### 1.4.5 안드로이드에서 테스트

- OpenHAB서버에서 `tail -f /var/log/openhab2/events.log` 해서 상태변화를 감지함
- Toggle 스위치를 선택할 때마다 다음과 같은 메시지가 나오면서 상태가 변화됨을 확인

```
2017-02-27 19:58:24.608 [ItemCommandEvent ] - Item 'mqttsw1' received command OFF 2017-02-27
19:58:24.614 [ItemStateChangedEvent ] - mqttsw1 changed from ON to OFF
```

#### 1.4.6 MQTT 메시지를 가지고 테스트(HW 스위치 시뮬레이션)

- OpenHAB서버에서 `tail -f /var/log/openhab2/events.log` 해서 상태변화를 감지함
- `mosquitto_pub -h localhost -t /testsw/1/state -m 1` 실행 후 상태가 ON으로 변하는 지 확인(참고로 1, 0, '1', '0' 모두 가능함)

#### 1.4.7 MQTT 사용자 인증 / 토픽 접근 제어

- `sudo vi /etc/mosquitto/mosquitto.users` 해서 아이디:암호 형식으로 `user_1:password_1` 처럼 1줄에 1개씩 기재
- `sudo mosquitto_passwd -U /etc/mosquitto/mosquitto.users`
- `sudo vi /etc/mosquitto/mosquitto.conf` 해서 다음을 추가  
`allow_anonymous false password_file /etc/mosquitto/mosquitto.users`
- `sudo service mosquitto restart` 해서 재시작

#### 1.4.8 테스트 해보자.

- 터미널 1: `mosquitto_sub -h localhost -t /test_topic -u user_1 -P password_1`
- 터미널 2:
  - `mosquitto_pub -h localhost -t /test_topic -m "Message from terminal 2" -u user_1 -P password_1` > OK
  - `mosquitto_pub -h localhost -t /test_topic -m "Message from terminal 2"` > Connection refused
- `sudo vi /etc/mosquitto/mosquitto.acl` 해서 다음을 추가  
`user user_1 topic /testsw/1/command topic /testsw/1/state`

`sudo service mosquitto restart`

#### 1.4.9 테스트 해보자.

- 터미널 1: `mosquitto_sub -h localhost -t /testsw/1/state -u user_1 -P password_1`
- 터미널 2: `mosquitto_pub -h localhost -t /testsw/1/state -m "ON" -u user_1 -P password_1` > OK
- 터미널 1: `mosquitto_sub -h localhost -t /testsw/**2**/state -u user_1 -P password_1`
- 터미널 2: `mosquitto_pub -h localhost -t /testsw/**2**/state -m "ON" -u user_1 -P password_1` > 메시지가 보내지지 않음

#### 1.4.10 OpenHAB2의 mqtt.cfg 변경

설정한 아이디/암호를 `/etc/openhab2/services/mqtt.cfg`에 반영

```
mymqtt.user=user_1
mymqtt.pwd=password_1
```