Um SvxLink mit Hilfe von MQTT steuern zu können, benötigen wir einen sogenannten Broker. Der Broker ist sozusagen der Vermittler zwischen Nachrichtensender (publisher) und -Empfänger (subscriber) und steht zwischen beiden. Er nimmt Daten der Publisher entgegen und verteilt Sie an die Subscriber. Wer einen eigenen MQTT-Broker betreiben möchte, kann diesen im lokalen Netzwerk oder direkt auf dem svx-client (DJSpot, DanielSpot, Relais...) installieren. Ein frei verfügbarer MQTT-Broker ist Mosquitto.

Mosquitto installieren

Im Raspberry-OS gehört Mosquitto zu den Respositories und kann direkt daraus installiert werden. Für Testzwecke ist es empfehlenswert, auch die MQTT-Clients zu installieren.

```
sudo apt install mosquitto mosquitto-clients
```

Konfiguration: Autostart

Für das automatische Ausführen von Mosquitto nach einem Systemneustart ist der Service in systemd zu aktivieren:

```
sudo systemctl enable mosquitto
```

Deaktivieren kann man dies ggf. durch:

```
sudo systemctl disable mosquitto
```

Mosquitto starten und stoppen

```
sudo systemctl start mosquitto
```

Mit folgendem Befehl beendet man den Mosquitto-Service:

```
sudo systemctl stop mosquitto
```

Hinweis: Wenn man Mosquitto beendet, werden alle Publisher- und Subscriber-Verbindungen beendet. Folglich müssen MQTT-Clients fehlertolerant sein und die Verbindung zum MQTT-Broker selbstständig aufbauen können.

Mosquitto lokal testen

Um zu testen, ob die Installation korrekt war, starten wir zunächst den MQTT-Subscriber-Client und teilen ihm mit, dass er auf das Topic „test“ reagieren soll.

```
mosquitto_sub -t test
```

Der Subscriber-Client kann nach dem Test mit STRG + C wieder beendet werden.

Nun benötigen wir ein zweites Terminal (Fenster/Reiter), von wo aus wir Daten (Payloads) an den Broker schicken können. Mit dem MQTT-Publisher-Client senden wir eine Nachricht (Payload) an den Broker mit dem Topic „test“. Der Übergabeparameter -t steht hierbei für „Topic“, -m für „Message“.

```
mosquitto_pub -t test -m "Hallo Welt"
```

Wenn im ersten Terminal die Nachricht „Hallo Welt“ angezeigt wird, dann war die MQTT-Kommunikation und damit auch die Installation erfolgreich.

Nun wissen wir zwar, dass die grundlegende Installation und Konfiguration des Brokers erfolgreich war, die eigentliche MQTT-Kommunikation muss allerdings noch eingerichtet werden. Das geht so:

Konfiguration: MQTT-Kommunikation im Netzwerk

Damit der MQTT-Broker Nachrichten annimmt und an einen oder mehrere Subscriber verteilt, muss die Kommunikation übers Netzwerk freigegeben bzw. konfiguriert werden. Prinzipiell ist eine Kommunikation **ohne** Authentifizierung möglich. Dies wird aber ausdrücklich NICHT empfohlen, da dann jeder Netzwerk-Client Nachrichten an den MQTT-Broker senden oder empfangen kann. Diesen Sicherheitsaspekt sollte man bedenken!

Besser ist es, die Kommunikation **mit** Authentifizierung einzurichten. Hierfür ist ein Benutzername und ein Passwort zu vergeben.

Konfiguration: Kommunikation ohne Authentifizierung

```
sudo nano /etc/mosquitto/conf.d/local.conf
```

In die noch leere Datei werden folgende Zeilen eingefügt:

```
listener 1883
allow_anonymous true
```

Listener <Port> spezifiziert den Port, auf dem der Broker lauschen soll. Allow_anonymous erlaubt die Kommunikation ohne Authentifizierung. Speichern und schließen mit Strg + O, Return, Strg + X.

Nun muss Mosquitto neu gestartet werden.

```
sudo systemctl restart mosquitto
```

Konfiguration: Kommunikation mit Authentifizierung mit Benutzername und Passwort

Für eine abgesicherte Kommunikation muss auf dem Broker ein Benutzername und ein Passwort erstellt werden. Folgender Befehl erzeugt einen Benutzer „mqttuser“. Das Passwort wird anschließend abgefragt. Notieren und nicht vergessen! Es wird ein Passwort-File „/etc/mosquitto/credentials“ erzeugt, welches das Passwort verschlüsselt speichert.

```
sudo mosquitto_passwd -c /etc/mosquitto/credentials mqttuser
```

Wenn die Datei schon existiert, können weitere Benutzer wie folgt hinzugefügt werden:

```
sudo mosquitto_passwd -b /etc/mosquitto/credentials USERNAME PASSWORD
```

Nun muss noch die Konfiguration von Mosquitto angepasst werden.

```
sudo nano /etc/mosquitto/conf.d/local.conf
```

Folgendes ist einzutragen:

```
listener 1883
allow_anonymous false
password_file /etc/mosquitto/credentials
```

Neustart von Mosquitto nicht vergessen!

Kommunikation übers Netzwerk prüfen

Um die Kommunikation zu testen, starten wir den MQTT-Subscriber-Client mit folgendem Kommando (ohne Authentifizierung am MQTT-Broker):

```
mosquitto_sub -t "#" -v
```

Oder mit Authentifizierung am MQTT-Broker :

```
mosquitto_sub -t "#" -v -u USER -P PASSWORD
```

Beenden mit STRG + C.

Inhalt modifiziert. Quelle: https://www.elektronik-kompendium.de/sites/raspberry-pi/2709041.htm

From:
././ - **Wiki FM-Funknetz**

Permanent link:
././doku.php?id=fm-funknetz:technik:mosquitto_einrichten

Last update: **22.04.2023 11:08**