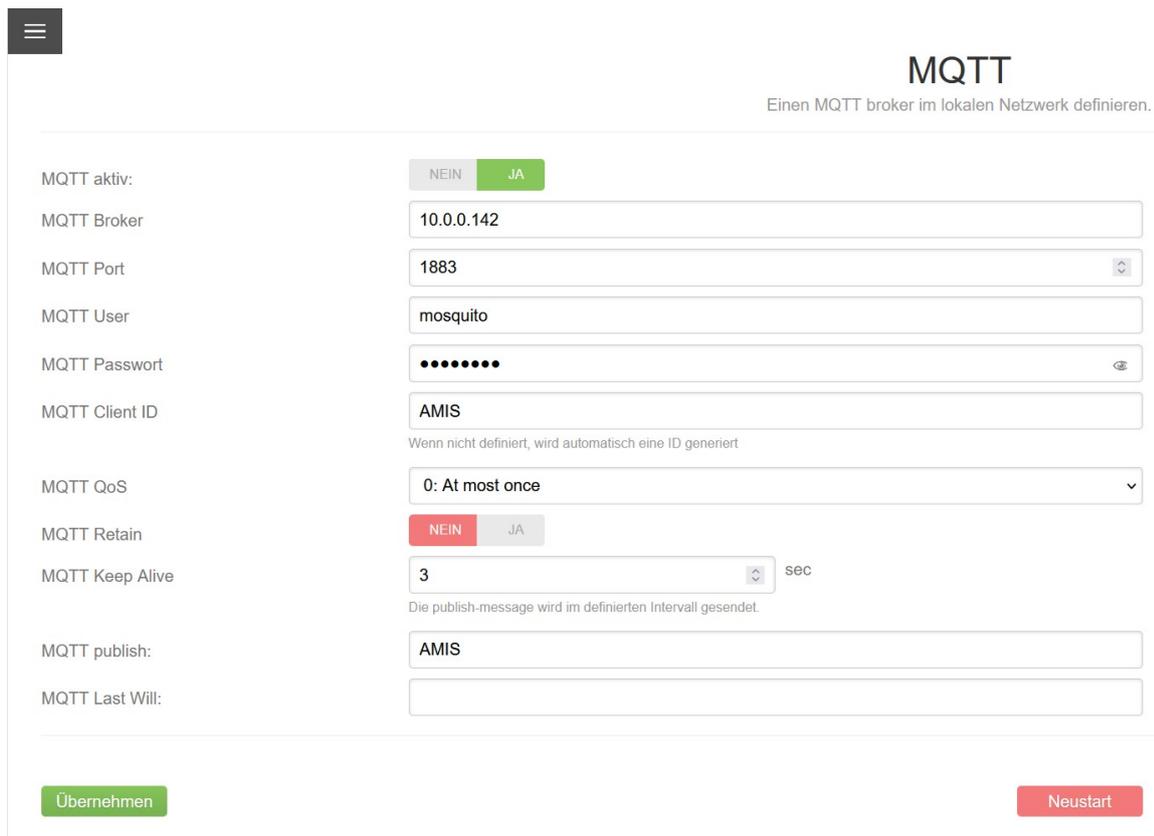


Konfiguration im Amis Webinterface:



MQTT

Einen MQTT broker im lokalen Netzwerk definieren.

MQTT aktiv: NEIN JA

MQTT Broker: 10.0.0.142

MQTT Port: 1883

MQTT User: mosquito

MQTT Passwort: ●●●●●●

MQTT Client ID: AMIS
Wenn nicht definiert, wird automatisch eine ID generiert

MQTT QoS: 0: At most once

MQTT Retain: NEIN JA

MQTT Keep Alive: 3 sec
Die publish-message wird im definierten Intervall gesendet.

MQTT publish: AMIS

MQTT Last Will:

1. iobroker: Paket "MQTT Broker/Client" installieren

2. iobroker: Paket als Server/broker konfigurieren und starten (User & Passwort kann leer bleiben)

*Alternative: wenn der MQTT-Server schon anderweitig läuft (z.B. auf einem Raspberry)
- Paket als „Client/Subscriber“ installieren und konfigurieren*

3. AMIS Leser: MQTT Client im Webinterface konfigurieren, Neustarten um zu iobroker zu verbinden (als Geräte name „AMIS“ verwenden) – Bild oben

4. iobroker: neue Einträge anlegen

4.1 Objekte -> „0_userdata.0“ anwählen -> „+“ -> „AMIS“ als Gerät anlegen

4.2 Objekte -> „AMIS“ anwählen -> „+“ -> folgende Datenpunkte Typ Zahl anlegen:
11280,170,180,270,280,370,381,470,481

5. iobroker: Paket „Script Engine“ installieren, um JavaScript nutzen zu können, um das JSON zu parsen und in Einzelwerte zu zerlegen (Update: Der "Skript Engine" Adapter heißt jetzt übrigens javascript (Scriptausführung))

6. iobroker: neuer Eintrag links „Skripte“ anwählen

7. Neues Skript anlegen mit angehängtem Code, anschließend Skript speichern und starten

8. Dann sollten unter „Objekte“ -> „0_userdata.0/AMIS“ -> die Werte je nach Intervall, der im AMIS Lesekopf konfiguriert ist, aktualisiert werden (Ich habe als „MQTT Keep Alive“ 3s Intervall eingestellt im Webinterface Lesekopf)

JavaScript Code

```
const idJson = 'mqtt.0.AMIS';
const id180 = '0_userdata.0.AMIS.180';
const id280 = '0_userdata.0.AMIS.280';
const id381 = '0_userdata.0.AMIS.381';
const id481 = '0_userdata.0.AMIS.481';
const id170 = '0_userdata.0.AMIS.170';
const id270 = '0_userdata.0.AMIS.270';
const id370 = '0_userdata.0.AMIS.370';
const id470 = '0_userdata.0.AMIS.470';
const id11280 = '0_userdata.0.AMIS.11280';

on(idJson, function(dp) {
  let obj = JSON.parse(dp.state.val);
  setState(id180, obj['1.8.0']/1000, true);
  setState(id280, obj['2.8.0']/1000, true);
  setState(id381, obj['3.8.1']/1000, true);
  setState(id481, obj['4.8.1']/1000, true);
  setState(id170, obj['1.7.0']/1000, true);
  setState(id270, obj['2.7.0']/1000, true);
  setState(id370, obj['3.7.0']/1000, true);
  setState(id470, obj['4.7.0']/1000, true);
  setState(id11280, obj['1.128.0'], true);
});
### Ende ###
```