

Raspberry: Setup Node Red

Node RED Einleitung

- “Flow”-basierte Programmierumgebung im Web Browser
- Nachrichten (JavaScript Objekte) fließen zwischen “Nodes”
 - Objekte haben meist “Payload”-Eigenschaft für Inhalt
 - In den Nodes können JavaScript-Funktionen erstellt werden, welche die Nachrichten nutzen können
- Einfaches Verbinden von Online Services, Hardware und APIs.
- Node Red basiert auf Node.JS und kann sowohl auf IoT-Devices als auch in der Cloud laufen.
- Flows können als JSON gespeichert, exportiert und importiert werden.
- Für Produktiv-Umgebung: Passwort für Editor setzen!
- <https://nodered.org/>
- <http://noderedguide.com/>

Zugriff Node RED Editor

1. Raspberry PI mit Strom versorgen
2. Raspberry PI und eigenen PC mit Netzwerk verbinden
3. Verbindung mit Node Red Editor am Raspberry PI
 - Feststellen der IP-Adresse des Raspberry Pis z.B. mit Fing (Android App), oder Angry IP Scanner (Linux)
 - Öffnen des Webbrowsers (Google Chrome)
 - Eingabe: `http://IP:1880`
 - Beispiel: <http://192.168.1.33:1880>
4. Zugriff auf Node Red Dashboard
 - Zuerst muss das Dashboard installiert werden, siehe weiterunten
 - Das Dashboard kann dann unter folgendem Link erreicht werden: <http://IP:1880/ui> wie z.B.: <http://192.168.1.33:1880/ui>

Empfehlung Security Node RED Editor

1. Passwort für Node RED Editor

- Infos: <https://nodered.org/docs/node-red-admin>
- Infos: <https://nodered.org/docs/security>

2. Installation Node-Red Admin

- `sudo npm install -g node-red-admin`

3. Passworthash generieren

- `node-red-admin hash-pw`
- Passwort eingeben
- In Datei

`.node-red/settings.js`
kopieren, Zeilen
auskommentieren
und username ggf.
anpassen

```
// Securing Node-RED
// -----
// To password protect the Node-RED editor and admin API, the following
// property can be used. See http://nodered.org/docs/security.html for details.
adminAuth: {
  type: "credentials",
  users: [{
    username: "user",
    password: "$2b$08$aNNLkjC2ncQAXjIqiC7p0.E4s/0dEsr7hMCLYh0.5rfgUN8rZGfWC",
    permissions: "*"
  }]
},
```

Node-Red Pakete installieren

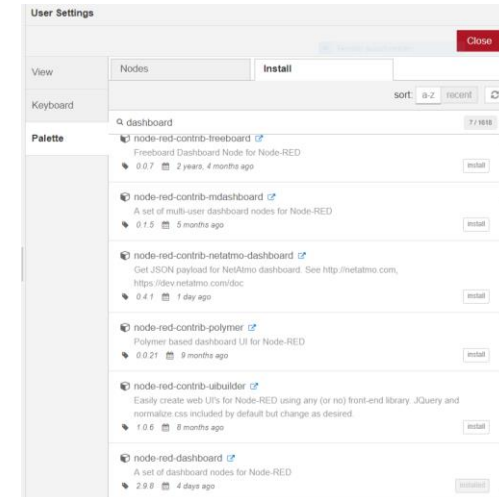
Installation von Dashboard und Serialport notwendig

1. Entweder Installation aus Konsole:

- `npm install node-red-dashboard`
- `npm install node-red-node-serialport`

2. Oder Installation aus Node-Red-Editor: Dashboard, Serialport

- a) Unter Einstellungen Manage Palette auswählen
- b) Auf Install klicken
- c) Name in Suchfeld eingeben `node-red-dashboard`
- d) Auf Install des Pakets klicken und warten!



Es gibt Vielzahl von Nodes <https://flows.nodered.org>

JavaScript Pakete installieren

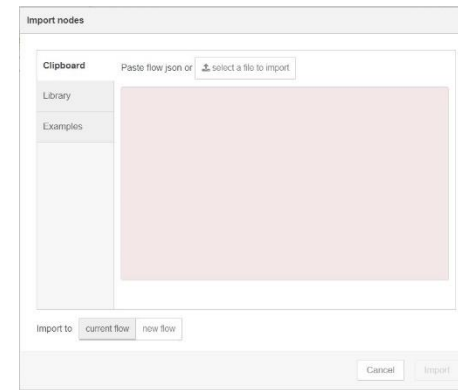
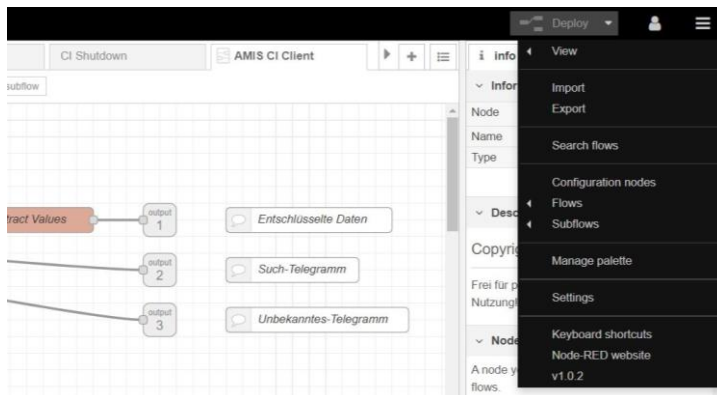
AES128 Library zur Entschlüsselung der Daten notwendig

1. JavaScript Paket installieren: `npm install aes-js`
 - Infos: <https://github.com/ricmoo/aes-js>
2. JavaScript Paket verfügbar machen:
 - Editieren von `.node-red/settings.js`
 - Unter `functionGlobalContext`: { folgendes einfügen:
`aesjs:require("aes-js")`

```
// The following property can be used to seed Global Context with predefined
// values. This allows extra node modules to be made available with the
// Function node.
// For example,
//   functionGlobalContext: { os:require('os') }
// can be accessed in a function block as:
//   global.get("os")
functionGlobalContext: {
  // os:require('os'),
  // jfive:require("johnny-five"),
  // j5board:require("johnny-five").Board({repl:false})
  aesjs:require("aes-js")
},
// `global.keys()` returns a list of all properties set in global context.
```

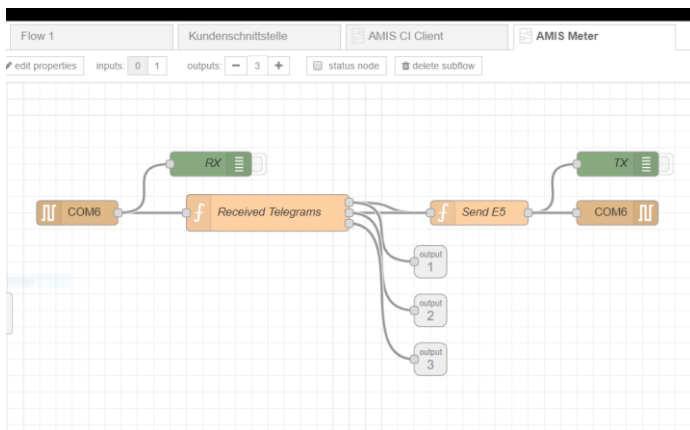
Importieren von Node RED Flows

- Im Node Red Editor im Menu (rechts oben) Import auswählen und auf select a file to import klicken.
 - 2019-11-14_RPI_CI_AMIS.json: Kommunikation mit AMIS-Zähler, Entschlüsselung und Darstellung im Dashboard.
 - Optional
 - 2019-11-14_RPI_CI_IP.json: Stellt USB-Port und IP-Adresse dar.
 - 2019-11-14_RPI_CI_Shutdown.json: Herunterfahren und Neustart
- Nach Import oder Änderungen auf Deploy klicken



Konfiguration Serialport und AES-Keys

- Setzen des Serialports
 - Nach Anschluss Lesekopf, Konfiguration der Serialport Nodes im Subflow AMIS Meter im Subflow AMIS-CI-Client
- Der AES-Key wird
 - entweder im Dashboard im Tab AMIS Kundenschnittstelle
 - oder im Node Red Editor im Subflow AMIS CI Client in der Funktion Set AES Key gesetzt



AMIS Kundenschnittstelle

Konfiguration Kundenschnittstelle

Status

Konfiguration AES-Key

AES-Key hier eingeben *

Entschlüsselung

Erfolg

BESTÄTIGEN

ABBRECHEN

IP-Display

CI AMIS

AMIS Zähler

Set AES Key

Edit function node

Properties

Name

Set AES Key

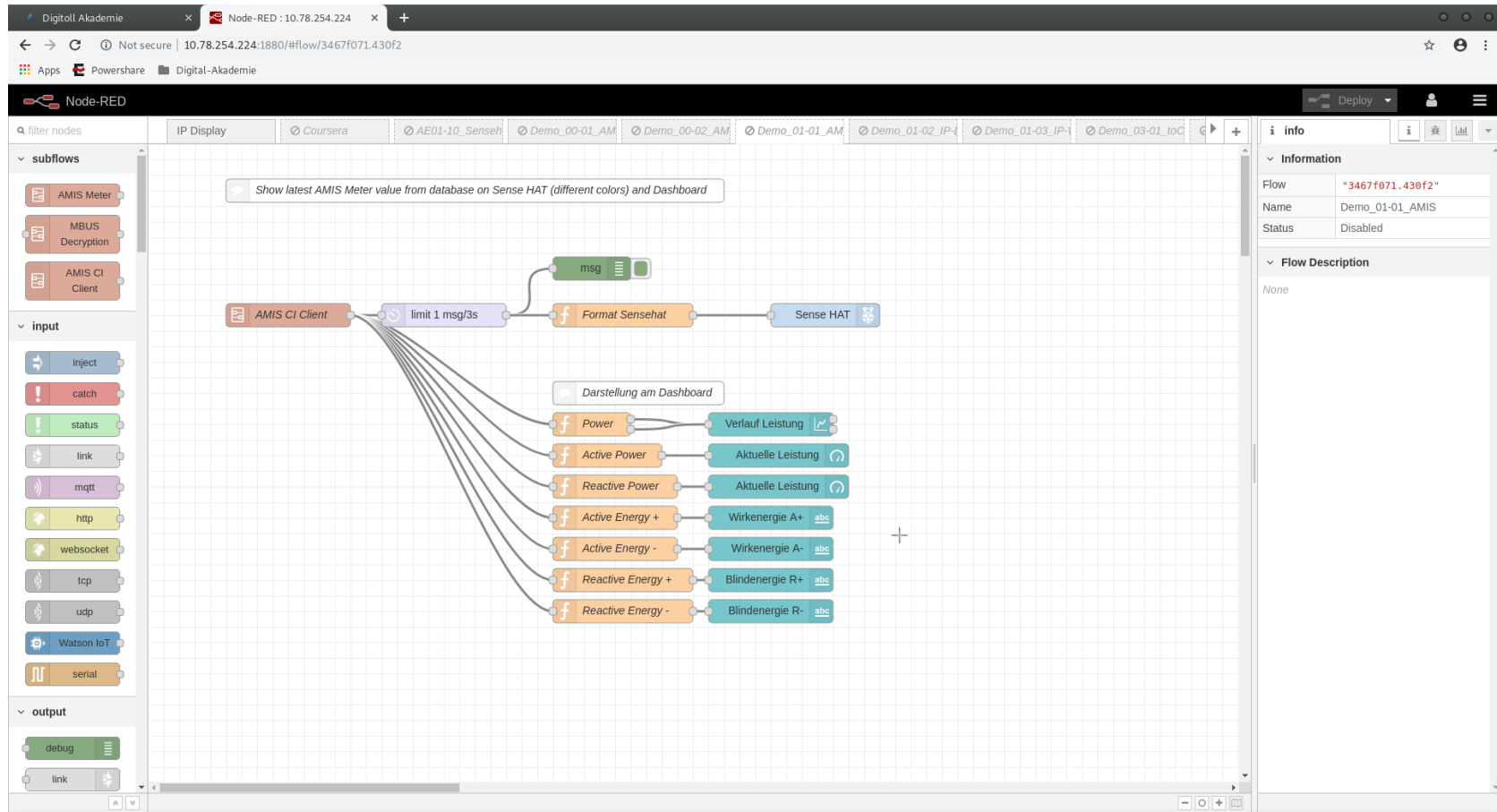
Function

```

1 //msg.aes_key_hex = "00112233445566778899AABCCDDEEFF";
2
3
4 msg.aes_key_hex = global.get('aes_key_hex') || "00112233445566778899AABCCDDEEFF";
5
6
7 //node.warn("AES-Key get: " + msg.aes_key_hex);
8
9 return msg;
10

```


Node RED Beispiel 1: AMIS-Zähler



The screenshot shows the Node-RED web interface in a browser. The main workspace contains a flow with the following components:

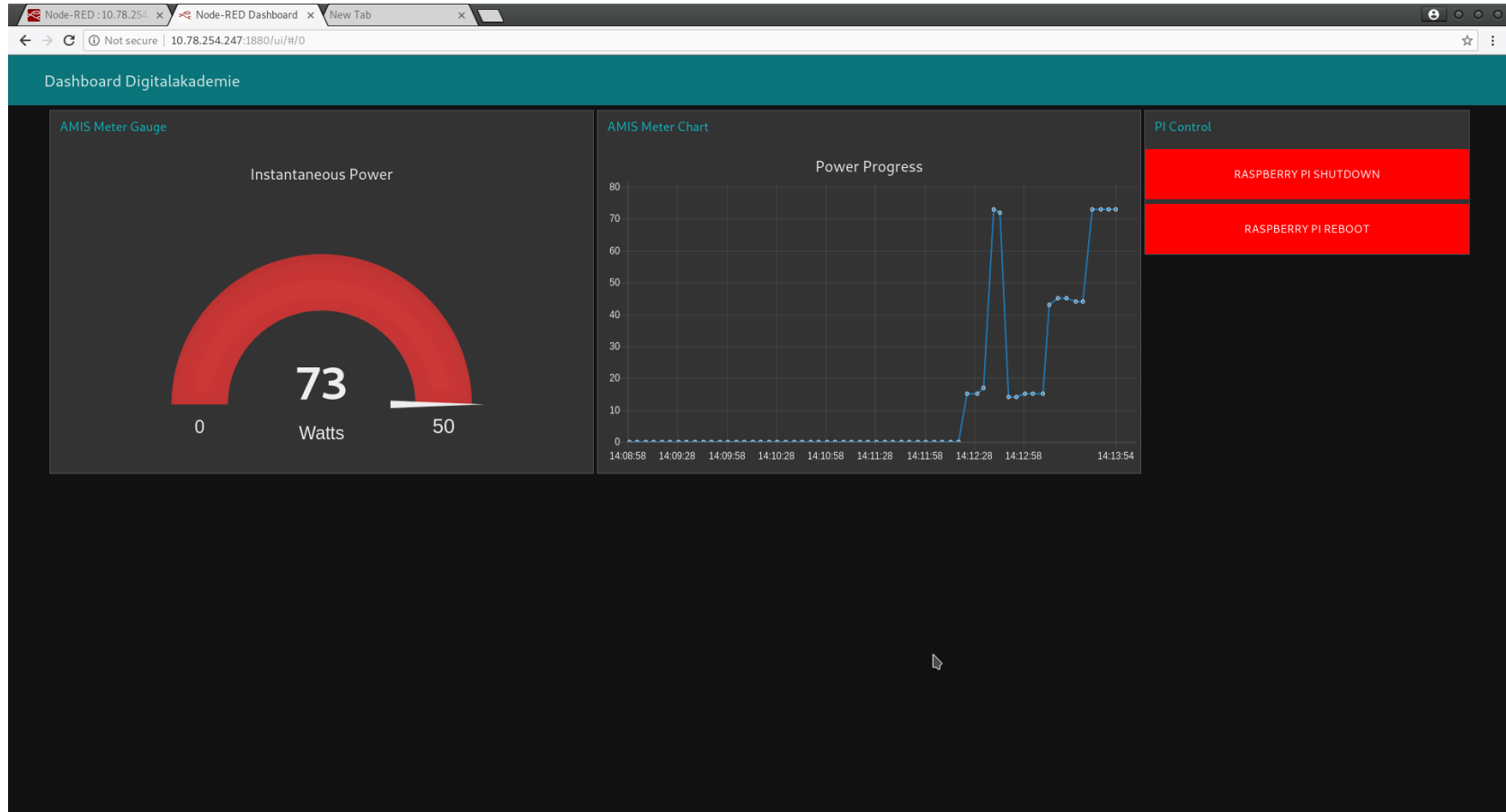
- AMIS CI Client** (input node) connected to **limit 1 msg/3s** (rate limiter).
- limit 1 msg/3s** connected to **Format Sensehat** (function node).
- Format Sensehat** connected to **Sense HAT** (output node).
- AMIS CI Client** also branches into six parallel paths, each leading to a function node: **Power**, **Active Power**, **Reactive Power**, **Active Energy +**, **Active Energy -**, and **Reactive Energy +**.
- Each function node is connected to a corresponding dashboard widget: **Verlauf Leistung**, **Aktuelle Leistung**, **Aktuelle Leistung**, **Wirkenergie A+**, **Wirkenergie A-**, **Blindenergie R+**, and **Blindenergie R-**.

On the right side, the 'info' panel shows the following details:

Information	
Flow	"3467f071.430f2"
Name	Demo_01-01_AMIS
Status	Disabled

The 'Flow Description' section is currently empty.

Node RED Beispiel 1: Dashboard AMIS-Zähler



Konfiguration Node RED Dashboard

- Ein Dashboard besteht aus Tabs und Groups
- Eine Dashboard-Node muss einer Group und einem Tab zugeordnet werden
- Wenn noch keine Tabs und Groups bestehen müssen diese erstellt werden.

The image displays the Node-RED configuration interface for a dashboard node. On the left, the 'Edit button node' window is open, showing the 'node properties' section. The 'Group' dropdown is set to 'PI Control [Dashboard Digitalakademie]'. The 'Background' is set to 'red'. Below this, the 'When clicked, send:' section is visible, with a payload of 'a_z' and a topic field. The 'Name' field contains 'Shutdown'. On the right, the 'dashboard' node's configuration is shown, including the 'Layout' tab and the 'Tabs & Links' section. The 'Tabs & Links' section shows a tree structure with 'Dashboard Digitalakademie' as the root, containing sub-items like 'AMIS Meter Chart', 'Environment Gauge', 'Environment Chart', 'PI Control', and 'Default'. The 'Dashboard Test' tab is also visible. On the far right, a preview of the dashboard is shown, featuring a dark theme with a 'Power Progress' section, an 'AMIS Meter Chart' area with the text 'Waiting for data from AMIS Meter', and a 'PI Control' section with two red buttons labeled 'RASPBERRY PI SHUTDOWN' and 'RASPBERRY PI REBOOT'.