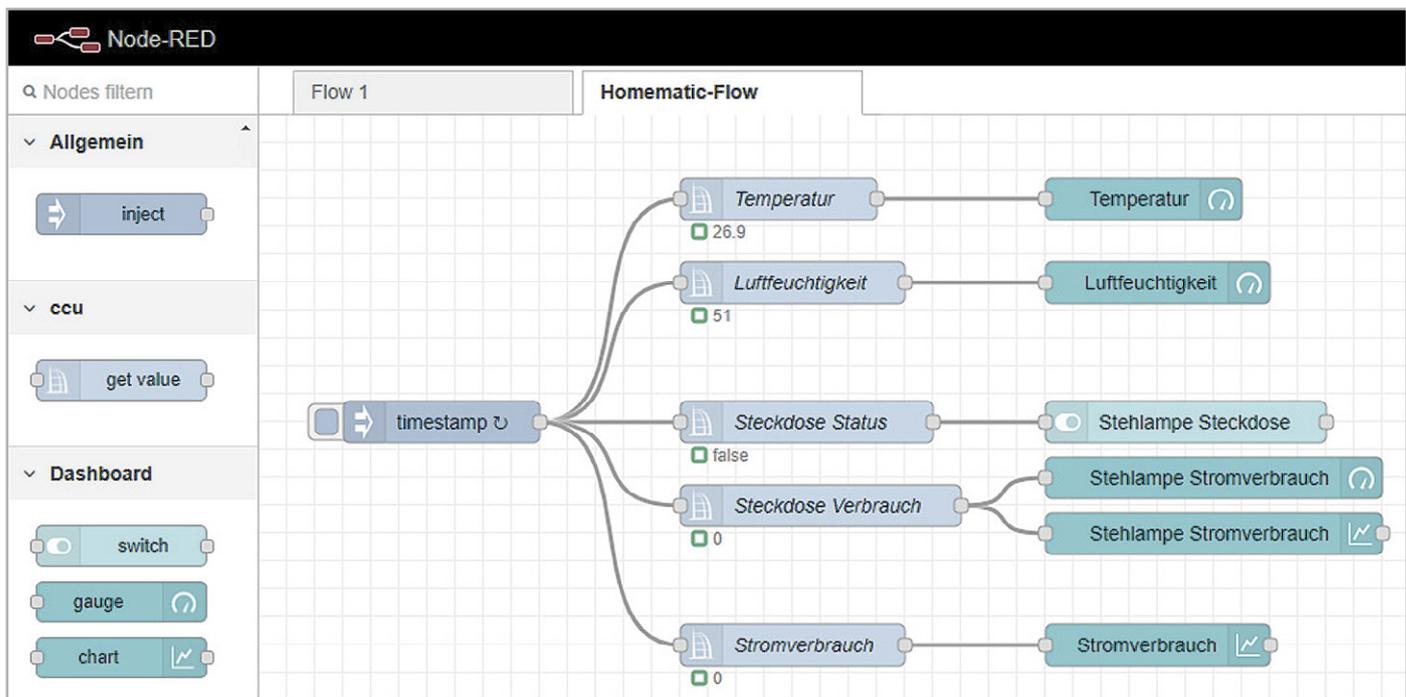
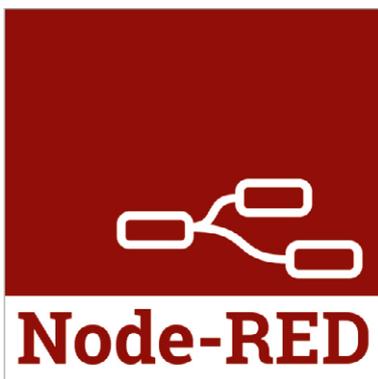


Vermittler im smarten Zuhause

Teil 4

Node-RED – ein grafisches Programmierungstool

Unser großer Vergleich von Vermittler-Software im Smart Home ist auf der Zielgeraden, und mit Node-RED besprechen wir nun das vierte kostenfreie Open-Source-Produkt. Smart Home ist in aller Munde; der Einsatz von Haussteuerungszentralen nimmt stetig zu. Es tummeln sich viele Broker-Systeme auf dem Markt, und es ist nicht leicht, sich für eines zu entscheiden. Eine Hilfestellung bietet diese mehrteilige Artikelserie: Nach dem Start im ELVjournal 3/2022 mit dem ioBroker [1], dem Teil 2 FHEM [2] und dem Teil 3 openHAB [3] ist nun die Software Node-RED an der Reihe.



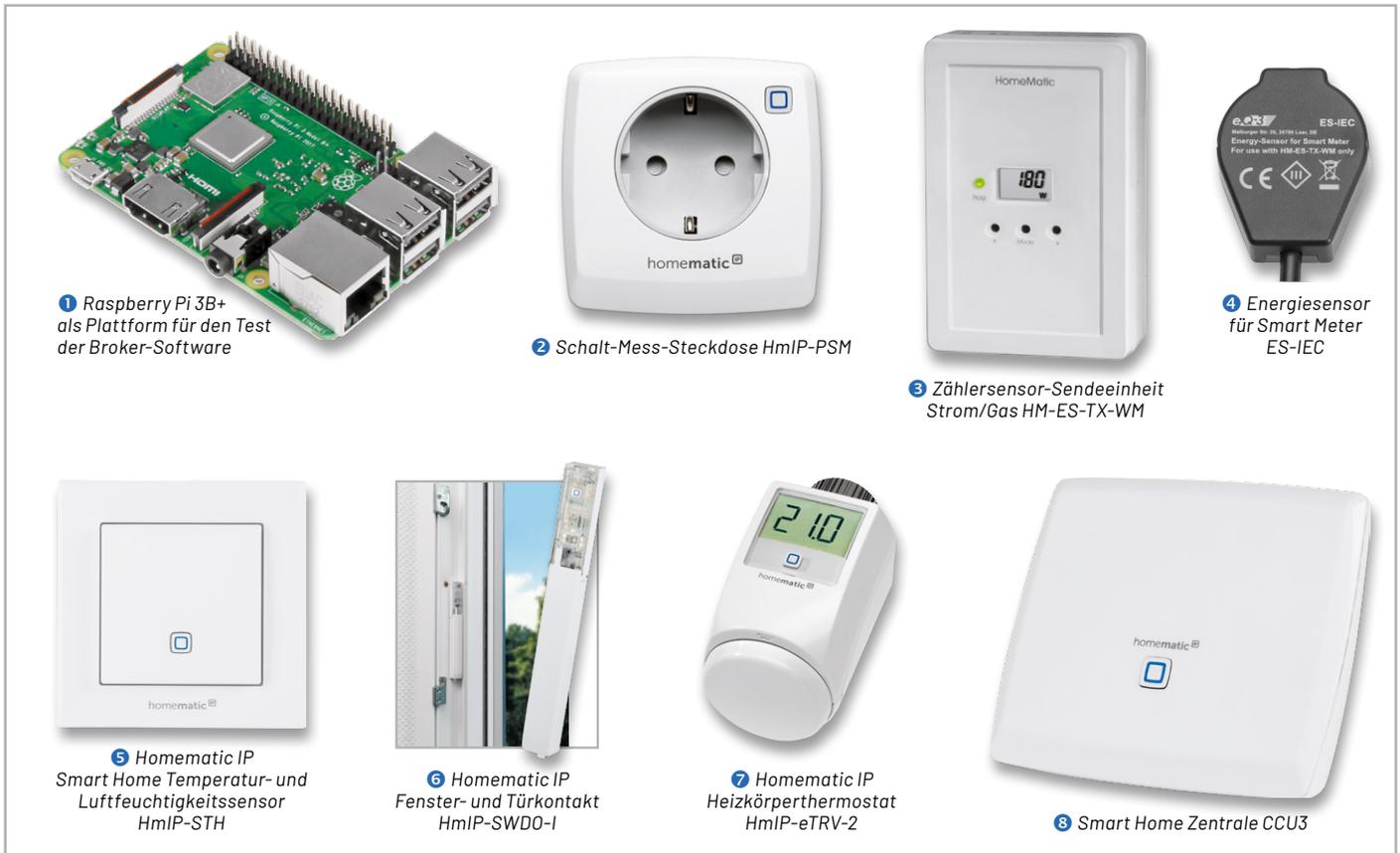


Bild 1: Überblick über alle Hardware-Komponenten für die Haussteuerungslösung

Grafisches Programmierwerkzeug

Node-RED bezeichnet sich als „Low-code programming for event-driven applications“ und ist ein von IBM entwickelter, grafischer Flow-Editor. In einem Webbrowser können JavaScript-Funktionen erstellt und gespeichert werden. Die Laufzeitumgebung baut auf Node.js auf, einer plattformübergreifenden Open-Source-JavaScript-Laufzeitumgebung.

Die mit einer visuellen Programmierumgebung erstellten Programme („Flows“) werden im JSON-Format abgespeichert. Die JavaScript Object Notation (JSON) ist ein gut lesbares Textformat zum Datenaustausch zwischen Anwendungen. Im Jahr 2016 hat IBM Node-RED als Open-Source-Software an die JS-Foundation übertragen.

Die Flow-basierte Programmierung wurde in den 70er-Jahren von J. Paul Morrison erfunden und betrachtet die Elemente der Anwendung als Black Boxes. Eine am Datenstrom orientierte Programmierung geht von einem kontinuierlichen Datenstrom aus, der in Echtzeit verändert und ausgegeben wird. Die Elemente in Node-RED werden als Nodes (Knoten) bezeichnet, sie haben ein genau definiertes Verhalten und eine eindeutige Aufgabe. Ein Node bekommt Daten als Eingabe, verarbeitet diese Informationen und gibt sie an seinen Ausgang weiter. Der Datenfluss zwischen den Knoten wird als Flow bezeichnet. In einem Flow werden also unterschiedliche Eingabe-, Ausgabe- und Verarbeitungsnodes miteinander verbunden.

Auf den Internetseiten von Node-RED [4] finden Sie viele englischsprachige Informationen und Dokumentationen sowie Links zu einem Blog, zur

Dokumentation und zum Forum. Ein sehr aktuelles Buch zum Thema ist Ende 2021 von Udo Brandes erschienen: „Node-RED: Das umfassende Handbuch“ [5].

Die Hardware-Komponenten

In unserer ganzen Artikelserie wird für die verschiedenen Broker-Programme immer ein kostengünstiger Raspberry Pi 3B+ ① als Zentrale verwendet. Auf der Endgeräteseite nutzen wir Geräte aus dem Programm von Homematic/Homematic IP. Wir haben uns für folgende Homematic Komponenten entschieden (Bild 1):

- Schalt-Mess-Steckdose HmIP-PSM ②
- Zählersensor-Sendeeinheit Strom/Gas HM-ES-TX-WM ③
- Energiesensor für Smart Meter ES-IEC ④
- Temperatur- und Luftfeuchtigkeitssensor HmIP-STH ⑤
- Fenster- und Türkontakt HmIP-SWDO-I ⑥
- Heizkörperthermostat HmIP-eTRV ⑦

Eine ausführliche Beschreibung der einzelnen Geräte finden Sie in Teil 1 dieser Beitragsreihe. Den zugehörigen Fachbeitrag finden Sie als Download auf den Seiten des ELVjournals im Internet [3]. Alle Geräte sind an einer CCU3 ⑧ angelernt, und wir bleiben bei unserem sogenannten Schreibtischaufbau für die Vorstellung dieses und der weiteren Broker-Systeme. Bevor wir uns nun mit der Installation von Node-RED befassen, setzen wir noch voraus, dass sowohl die CCU3 als auch alle Geräte mit der aktuellen Firmware ausgestattet sind. Für unsere CCU3 nutzen wir die Version 3.61.7 und für die Geräte die jeweils im März 2022 aktuellen Firmware-Versionen.

Die Installation von Node-RED auf dem Raspberry Pi

Auf dem Raspberry Pi ist die OS-Version Raspberry Pi OS Lite (Release-Datum 28.01.2022 [6]) auf einer 16-GB-SD-Karte vorinstalliert. Über einen SD-Card-Writer (hier empfiehlt sich das Programm balena Etcher [7]) kann das Image auf die SD-Karte kopiert werden. Über einen SSH-Client (zum Beispiel über das Programm PuTTY [8]) greifen wir auf

```

pi@raspberrypi:~ $ sudo apt update
pi@raspberrypi:~ $ sudo apt full-upgrade

pi@raspberrypi:~ $ sudo shutdown -r now
oder
pi@raspberrypi:~ $ sudo reboot

```

Bild 2: Update/Upgrade-Befehle für den RaspberryPi

den Minirechner per Kommandozeile zu. Durch die Befehle `sudo apt update` und `sudo apt full-upgrade` werden nach der Grundinstallation vorab alle vorhandenen Pakete aktualisiert.

Nach einem Neustart über `sudo reboot` ist der Raspberry Pi auf dem aktuellen Stand (Bild 2).

Die Installation von Node-RED ist sehr einfach und weitgehend automatisiert. Verbinden Sie sich über einen SSH-Client mit dem Raspberry Pi – der Benutzername lautet „pi“ und das Start-Passwort „raspberry“.

Starten Sie die Installation mit diesem Befehl (Bild 3):

```
bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)
```

```

pi@raspberrypi:~ $ bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installer
s/master/deb/update-nodejs-and-nodered)

```

```

This script checks the version of node.js installed is 12 or greater. It will try to
install node 14 if none is found. It can optionally install node 12, 14 or 16 LTS for you.

```

```

If necessary it will then remove the old core of Node-RED, before then installing the latest
version. You can also optionally specify the version required.

```

```

It also tries to run 'npm rebuild' to refresh any extra nodes you have installed
that may have a native binary component. While this normally works ok, you need
to check that it succeeds for your combination of installed nodes.

```

```

To do all this it runs commands as root - please satisfy yourself that this will
not damage your Pi, or otherwise compromise your configuration.
If in doubt please backup your SD card first.

```

```

See the optional parameters by re-running this command with --help

```

```

Are you really sure you want to do this ? [y/N] ? y

```

```

Would you like to install the Pi-specific nodes ? [y/N] ? y

```

Bild 3: Installation von Node-RED nach dem ersten Start des Raspberry Pi

```

Running Node-RED update for user pi at /home/pi on raspbian

```

```

This can take 20-30 minutes on the slower Pi versions - please wait.

```

```

Stop Node-RED                ✓
Remove old version of Node-RED ✓
Remove old version of Node.js ✓
Install Node.js 14 LTS        ✓ v14.19.3   Npm 6.14.17
Clean npm cache               ✓
Install Node-RED core         ✓ 2.2.2
Move global nodes to local    -
Npm rebuild existing nodes    ✓
Install extra Pi nodes        -
Add shortcut commands         ✓
Update systemd script         ✓

```

```

Any errors will be logged to /var/log/nodered-install.log
All done.

```

```

You can now start Node-RED with the command node-red-start
or using the icon under Menu / Programming / Node-RED
Then point your browser to localhost:1880 or http://{your_pi_ip-address}:1880

```

```

Started : Sun 3 Jul 12:38:46 BST 2022
Finished: Sun 3 Jul 12:44:15 BST 2022

```

```

You may want to run node-red admin init
to configure your initial options and settings.

```

```

pi@raspberrypi:~ $

```

Bild 4: Installationschritte von Node-RED

Bild 5: Konfigurations-Tool von Node-RED auf dem Raspberry Pi

```

pi@raspberrypi:~ $ node-red admin init
Node-RED Settings File initialisation
=====
This tool will help you create a Node-RED settings file.

▼ Settings file · /home/pi/.node-red/settings.js

User Security
=====
▼ Do you want to setup user security? · Yes
▼ Username · admin
▼ Password · ****
▼ User permissions · full access
▼ Add another user? · No

Projects
=====
The Projects feature allows you to version control your flow using a local git repository.

▼ Do you want to enable the Projects feature? · No

Flow File settings
=====
▼ Enter a name for your flows file · flows.json
▼ Provide a passphrase to encrypt your credentials file ·

Editor settings
=====
▼ Select a theme for the editor. To use any theme other than "default", you will need to inst
all @node-red-contrib-themes/theme-collection in your Node-RED user directory. · default
▼ Select the text editor component to use in the Node-RED Editor · monaco (default)

Node settings
=====
▼ Allow Function nodes to load external modules? (functionExternalModules) · Yes

Settings file written to /home/pi/.node-red/settings.js
pi@raspberrypi:~ $

```

Bild 6: Node-RED als Dienst auf dem Raspberry Pi installieren

```

pi@raspberrypi:~ $ sudo systemctl enable nodered.service
Created symlink /etc/systemd/system/multi-user.target.wants
/nodered.service → /lib/systemd/system/nodered.service.
pi@raspberrypi:~ $ sudo shutdown -r now

```

Der Installationsvorgang kann 20 bis 30 Minuten dauern, und es werden alle notwendigen Pakete (node.js, npm, node-red core) in den aktuellen Versionen nachinstalliert (Bild 4).

Danach wird das Konfigurations-Tool von Node-RED aufgerufen: „node-red admin init“ (Bild 5). In diesem Vorgang werden der/die User angelegt (zum Beispiel „admin“), das/die Passwort/e vergeben und weitere Settings vorgenommen.

Was nun noch fehlt, ist der Start von Node-RED auf dem Raspberry Pi. Er erfolgt mit dem Befehl

```
node-red start
```

oder kann mit der Befehlszeile

```
sudo systemctl enable nodered.service start
```

als Dienst aktiviert werden (Bild 6).

Jetzt startet Node-RED bei jedem Neustart des Raspberry Pi automatisch.

Anschließend können Sie die Node-RED-Oberfläche in einem Browser direkt aufrufen:

```
http://<ip-adresse>:1880
```

Sie werden zunächst aufgefordert, den Benutzernamen und das Passwort für die Weboberfläche einzugeben – unser Administrations-User heißt „admin“ (Bild 7).

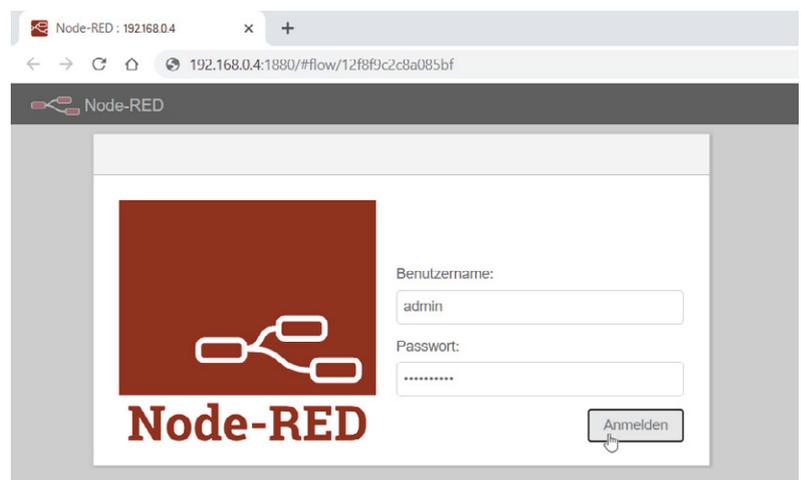


Bild 7: Anmeldemaske von Node-RED im Webbrowser

Danach startet ein neunteiliges Tutorial. Im Anschluss landen Sie auf einer noch leeren Startseite (Bild 8). Diese grafische Oberfläche besteht aus vier Bereichen:

- 1 Der Kopfleiste mit einem Übernahme-Button, einem User-Button und einem Menü-Button
- 2 Der nach Funktionen/Systemen gegliederten Palette mit den Nodes (Bild 9)
- 3 Dem Arbeitsbereich für die Flows
- 4 Der rechten Seitenleiste mit Info, Hilfe, Debug und Konfiguration

So funktioniert es

In Node-RED werden in einem Flow die Informationen von einem Knoten (Node) zu einem anderen Knoten als „msg“-Objekt weitergeleitet. Jeder Node hat eine festgelegte und eindeutige Aufgabe. Die Verknüpfungen (Wires) von einem Ausgang zum nächsten Eingang erfolgen mit gedrückter linker Maustaste von einem Punkt zum nächsten – überflüssige oder falsche Verbindungen lassen sich mit einem Mausklick markieren und über die ENTF-Taste schnell wieder löschen. Die Rückgängig-Funktion liegt wie bei Windows üblich auf der Tastenkombination STRG-Z. Für die Bearbeitung bzw. Konfiguration

eines Nodes wird durch einen Doppelklick mit der Maus ein Bearbeitungsfenster geöffnet.

Werden Daten an einen Node übermittelt, so kann er diese Daten verarbeiten und an einen oder mehrere Knoten weitergeben. Durch die Verbindung des Ausgangs eines Funktionsknotens mit dem Eingang eines Debug-Nodes können Sie im Debug-Fenster (Käfersymbol rechts oben) mitverfolgen, welche Werte zwischen den Knoten weitergegeben werden. Der Standard-Datenwert des jeweiligen Nodes wird in „msg.payload“ übermittelt.

Im Prinzip können Sie beliebig viele Debug-Nodes an andere Knoten anbinden, es empfiehlt sich jedoch, den Flow schrittweise zu testen und sich durch den Flow von links nach rechts durchzuarbeiten, denn bei zu vielen Debug-Knoten geht im Ausgabefenster sehr schnell der Überblick verloren.

CCU3 anbinden und Geräte hinzufügen

Um unsere Homematic Zentrale CCU3 mit Node-RED zu verbinden, muss zunächst eine Contribution (Palette) für Homematic installiert werden – diese Contribution heißt node-red-contrib-ccu und hat die Versionsnummer 3.4.2 (Bild 10).

Am schnellsten gelingt dies über den Menüpunkt „Palette verwalten“ und über das Suchwort „Homematic“ (Bild 10, 1). Während der Installation (Bild 10, 2) werden mehrere ccu-Nodes zur Palette hinzugefügt (Bild 10, 3) (s. Bild 9, rechts Spalte ccu).

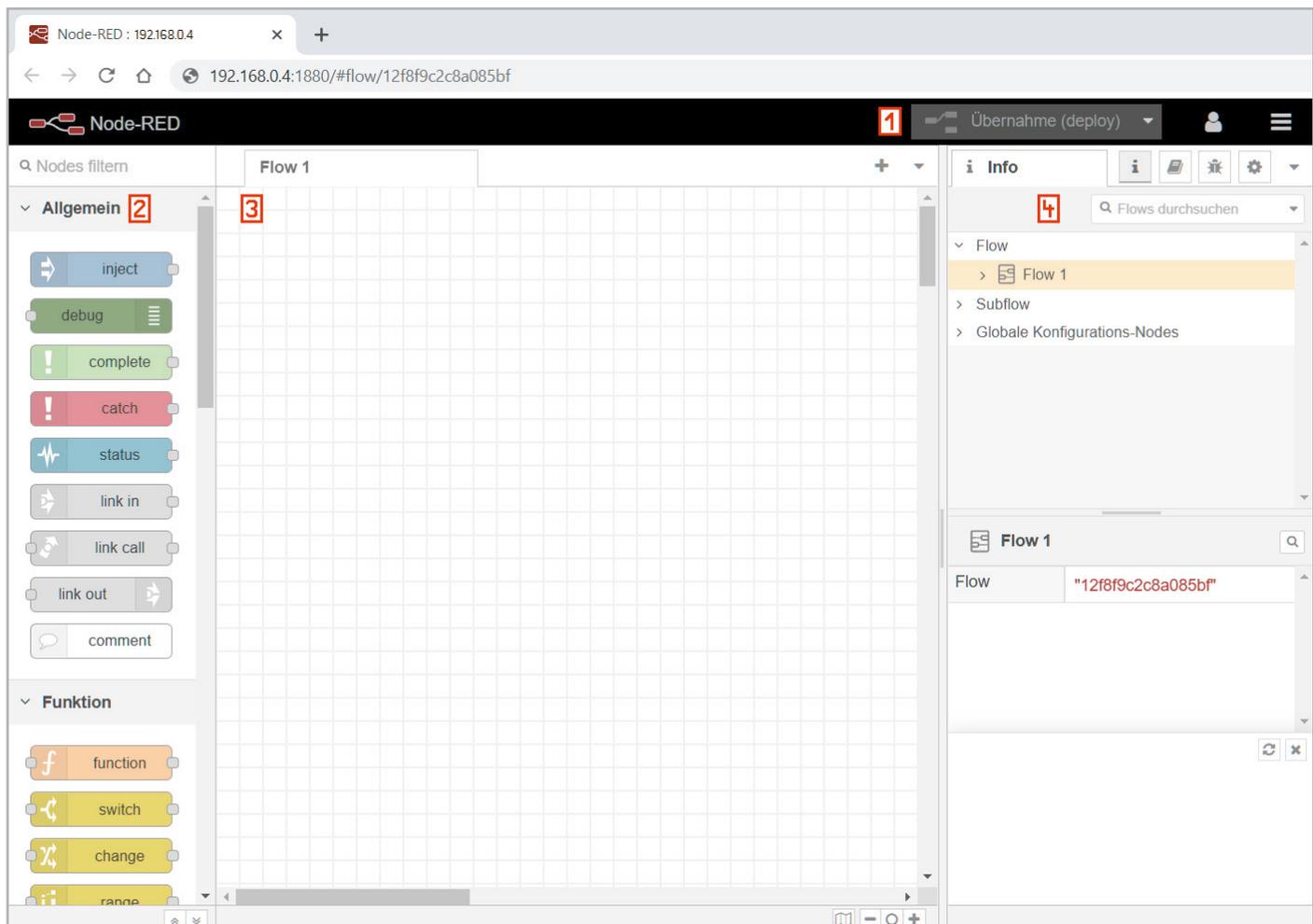


Bild 8: Node-RED-Startseite

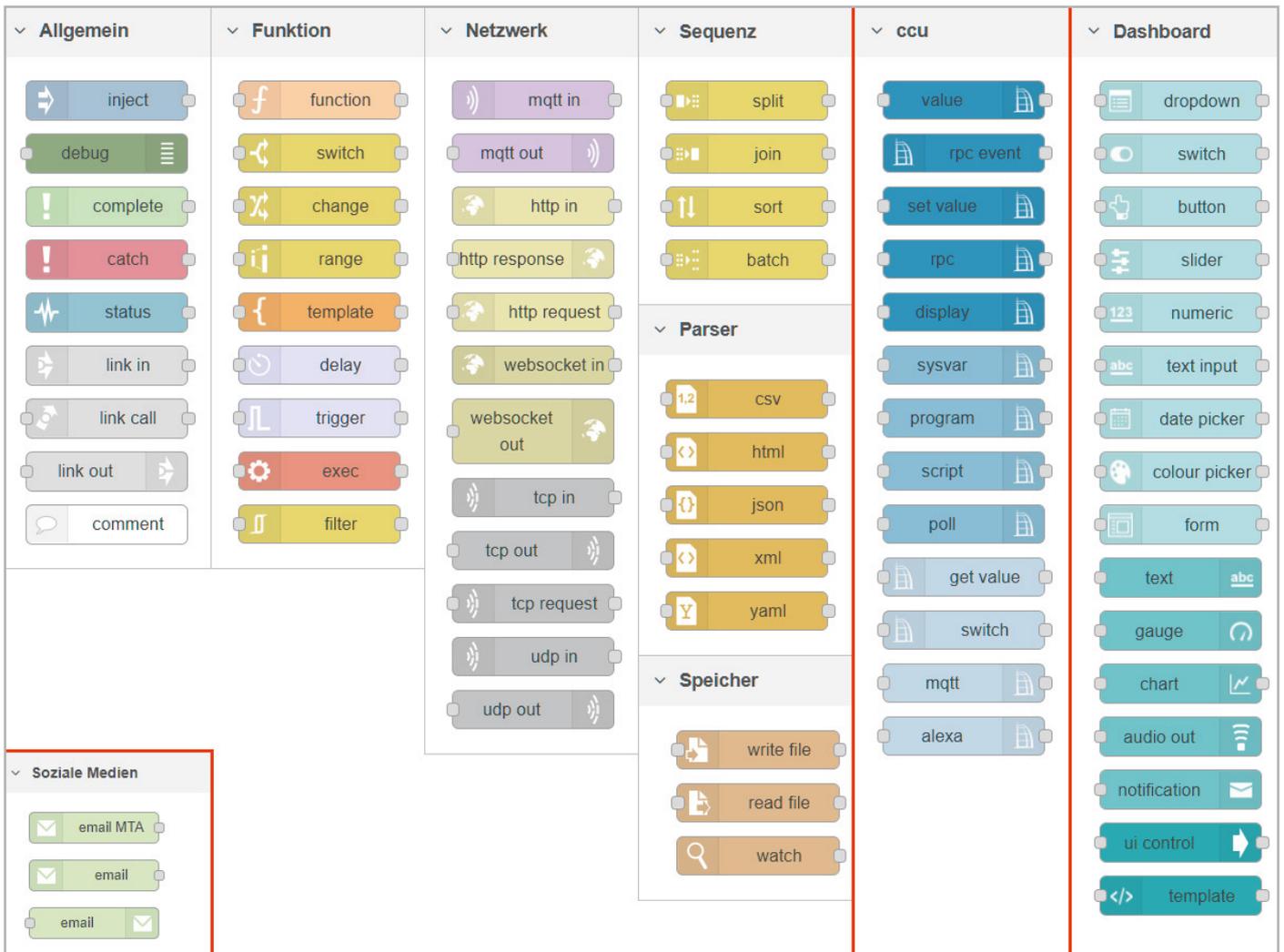


Bild 9: Standard-Nodes, Homematic/Homematic IP Nodes und Dashboard-Nodes in Node-RED

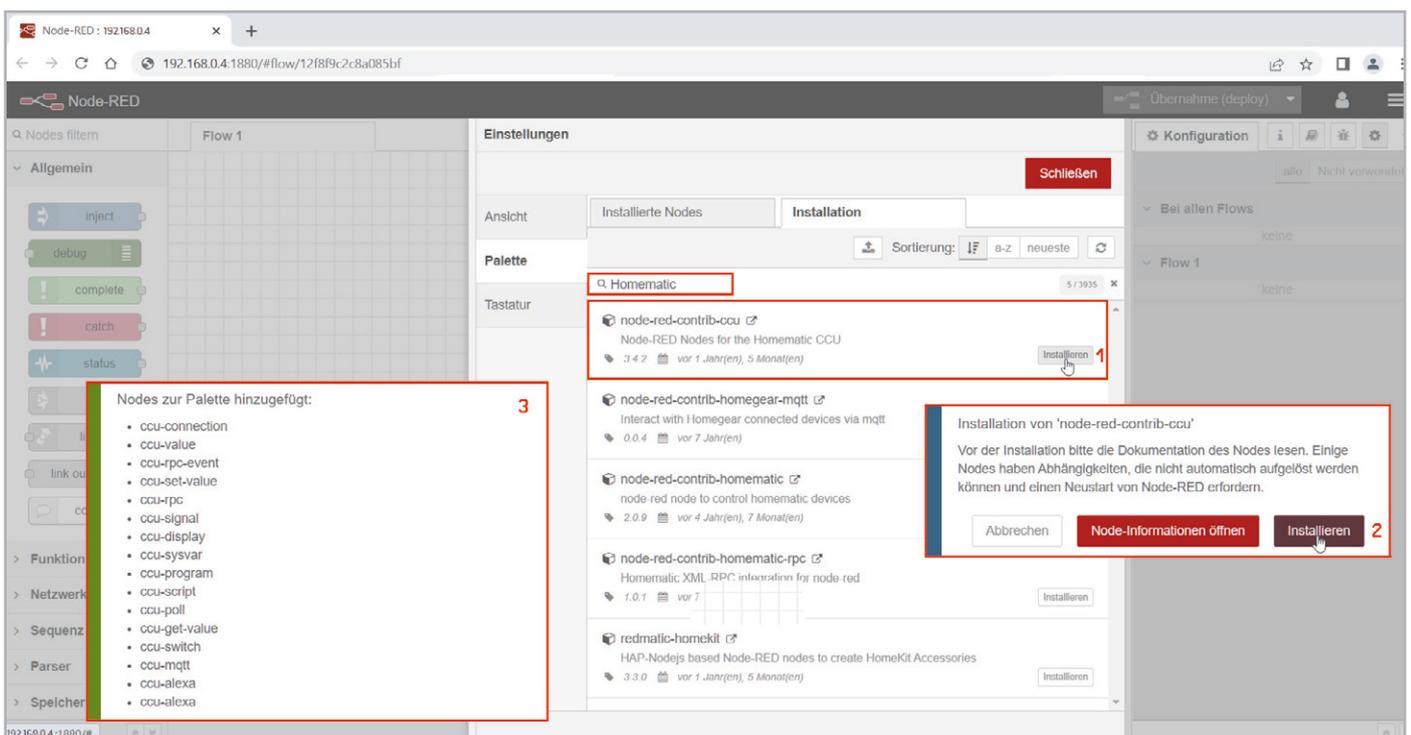


Bild 10: Installation der Homematic Contribution

Damit die CCU im Hausnetz gefunden wird, müssen Sie in der Konfiguration (Zahnrad-Symbol rechts oben) mehrere IP-Adressen, Ports und Einstellungen für die CCU-Connection ergänzen (Bild 11).

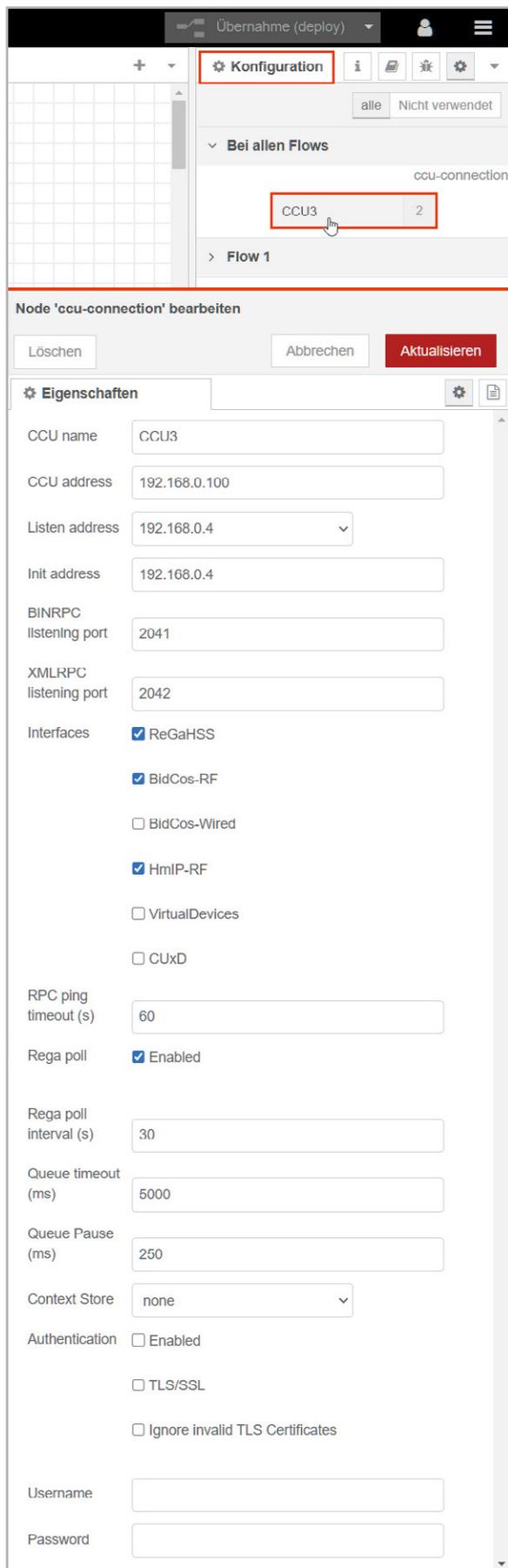


Bild 11: Konfiguration der CCU-Connection

Für unsere Testgeräte benötigen Sie die CCU-Interfaces ReGaHSS, BidCos-RF und HmIP-RF.

Nach Installation und Konfiguration der CCU-Contribution können Sie einen ersten Flow erstellen (Bild 12), um ein Gefühl für die Funktionsweise zu bekommen und Werte aus einem der Homematic Geräte auszulesen.

Platzieren Sie zunächst aus der Palette „ccu“ zwei Nodes „get value“ in den Arbeitsbereich von Flow 1. Danach brauchen Sie einen „Auslöser“ und ein Ergebnisfenster: Die benötigten Nodes aus der Palette „Allgemein“ heißen „inject“ und „debug“. Verbinden Sie mit der Maus die Nodes miteinander und beginnen Sie danach mit der Konfiguration aller Nodes.

Damit Werte in unserem Flow übermittelt werden, ist zunächst der inject-Node zu konfigurieren: Mit einem Doppelklick auf diesen Knoten öffnet sich das Bearbeiten-Fenster. Vergeben Sie einen Namen (z. B. „Test für Flow 1“), aktivieren Sie „Einmal injizieren...“ und setzen Sie das Wiederholungs-Intervall auf 30 Sekunden (Bild 12, links). Nach einem Klick auf „Fertig“ schließt sich das Fenster wieder, und es geht an die weiteren Nodes.

Die beiden Homematic Nodes („get value“) müssen mit der CCU verbunden werden, und sie nutzen dort das Interface „HmIP-RF“. Danach werden bei „Channel“ alle Homematic IP Geräte aufgelistet – unser Temperatur-/Luftfeuchtigkeitssensor heißt HmIP-STH Wohnzimmer, die gewünschten Werte werden im Kanal 1 angezeigt, und Temperatur und Luftfeuchtigkeit verstecken sich in den Datapoints ACTUAL_TEMPERATURE und HUMIDITY. Vergeben Sie einen sprechenden Namen für jeden Node und klicken Sie wieder auf „Fertig“ (Bild 12, rechts).

Am Ende übernehmen (deploy) Sie alle Eingaben – die blauen Punkte an den Nodes verschwinden, der Deploy-Button wechselt die Farbe, und Sie können Ihren ersten Flow mit einem Klick auf „Test für Flow 1“ ausprobieren (Bild 12, unten).

Der Debug-Node zeigt nun alle Werte, die ihn erreichen, in einem Debug-Fenster in der rechten Seitenleiste – die Aktivierung erfolgt über den Button mit dem Käfersymbol. Gerade bei der Entwicklung und dem Test von Flows sollten Sie reichlich davon Gebrauch machen. Durch die Konfiguration des Wiederholungsintervalls auf 30 Sekunden kommen nun regelmäßig Daten im Debug-Fenster an. Alle Werte und das Endergebnis sehen Sie in unserem Flow-1-Screenshot (Bild 12, unten rechts).

Die Verbindung zu weiteren Smart-Home-Systemen erfolgt auf ähnliche Weise über zusätzliche Module (Contributions) – die Auswahl ist hier riesig und erlaubt Ihnen die Anbindung nahezu aller gängigen Haussteuerungssysteme.

Visualisierung über ein Dashboard

Die graphische Darstellung in Node-RED wird als sogenanntes Dashboard realisiert und muss über eine eigene Contribution (z. B. node-red-dashboard) nachinstalliert werden (Bild 13).

Über die Nodes in der Dashboard-Palette (Bild 9, rechte Spalte Dashboard) können Sie in Ihren Flows entsprechende Dashboard-Elemente mit Daten versorgen. Die Vorgehensweise ist immer wieder identisch: Nodes zu einem Flow hinzufügen, die Nodes miteinander verbinden, jeden Node einzeln konfigurieren und danach alles deployen. Die Gestaltungsmöglichkeiten sind sehr vielfältig und erfordern etwas Zeit zur Einarbeitung.

An dieser Stelle können wir leider nicht weiter auf Details eingehen, aber viele Nodes lassen sich sehr intuitiv konfigurieren – versuchen Sie es einfach. Zusätzliche Hilfestellungen bieten die Dokumentationen zu Node-RED [9] und zum node-red-dashboard [10]. Die Doku-Links zu den einzelnen Contributions sehen Sie in der Regel als kleines Pfeilsymbol rechts neben dem Contribution-Namen im Installations-Fenster.

Nach erfolgreichem Übernahme-Vorgang erreichen Sie das User-Interface (Dashboard) im Browser über die Adresse:

`http://<ip-adresse>:1880/ui/`

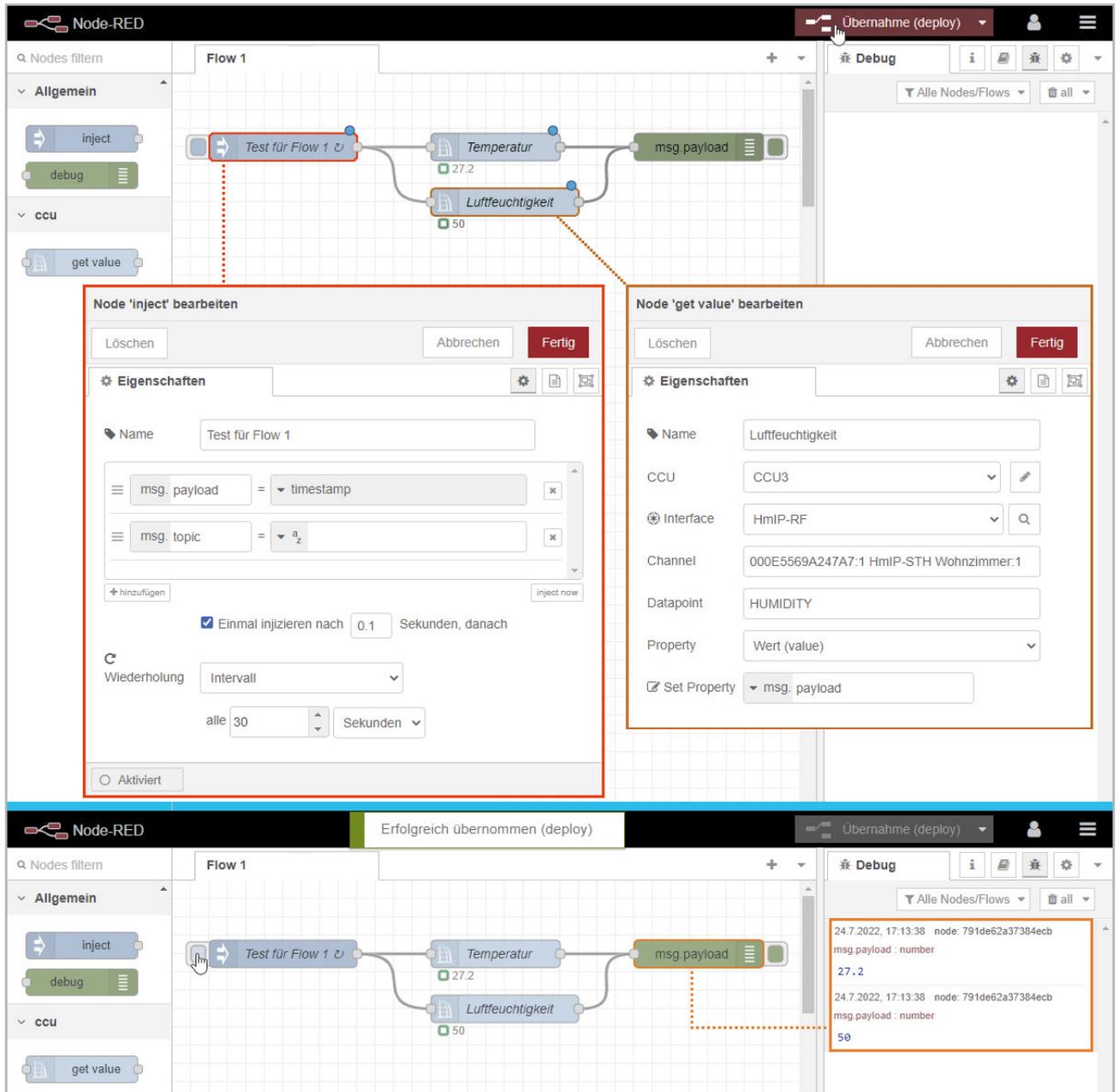


Bild 12: Ein erster Flow mit dem Luftfeuchtigkeitssensor HmIP-STH

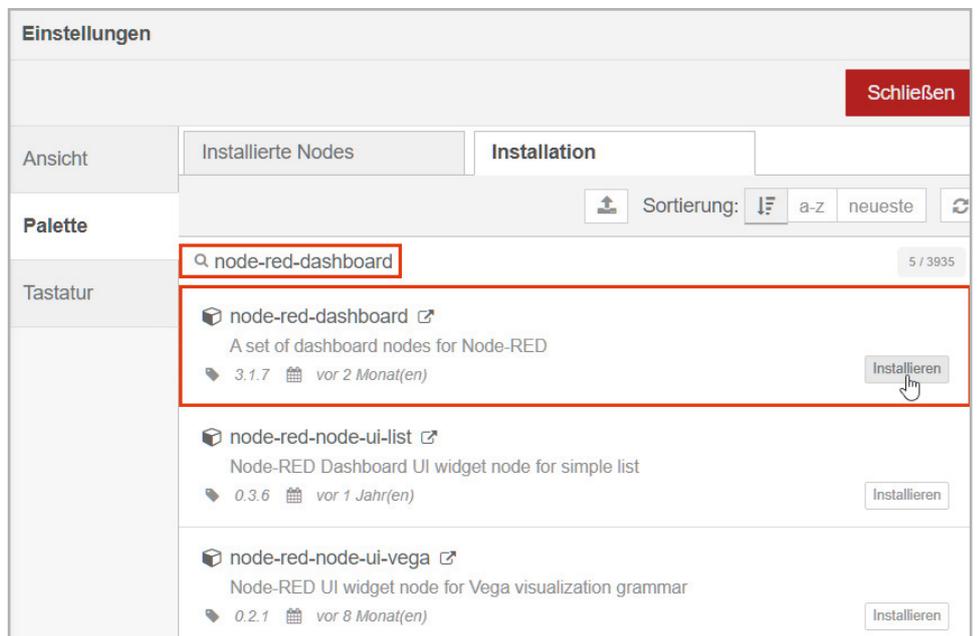


Bild 13: Installation des Dashboards

Ein Dashboard-Beispiel zeigt unser Homematic Flow für die Schalt-Mess-Steckdose HmlP-PSM an einer Stehlampe im Wohnzimmer und für die Werte des Temperatur- und Luftfeuchtigkeitssensors HmlP-STH (Bild 14 und 15).

Automatisierung

Natürlich besteht eine Automatisierungslösung nicht nur aus einem Dashboard zur Anzeige von Werten und zur Bedienung von Geräten. Richtig smart wird eine Hausautomation erst, wenn auf bestimmte Ereignisse mit entsprechenden Aktionen reagiert wird. Bei unserem Schreibtischaufbau wollen wir den Stromverbrauch unserer Stehlampen-Steck-

dose überwachen und bei Überschreiten der 60-W-Grenze eine E-Mail-Nachricht auslösen. Für den E-Mail-Versand ist wieder ein zusätzliches Modul für die Palette zu installieren (Bild 16) – die Konfiguration der E-Mail-Credentials erfolgt im Anschluss im Node „email“.

Unser Beispiel-Flow „Steckdosen-Alarm“ (Bild 17) holt den Verbrauchswert alle 60 Sekunden aus dem Datenpunkt POWER im Kanal 6 des HmlP-PSM Wohnzimmer und übergibt ihn an den Funktionsknoten „function“. Dieser Node-Typ stellt das Grundgerüst für eigene Funktionen zur Verfügung und bietet über JavaScript-Code umfangreiche Steuerungsmöglichkeiten.

In unserem kleinen Beispiel wird Message-Payload in eine Variable „stromverbrauch“ eingelesen und in einer IF-Abfrage wird überprüft, ob der Wert von 60 (W) überschritten wird. Entsprechend werden daraufhin die Ausgabewerte an den E-Mail-Node definiert.

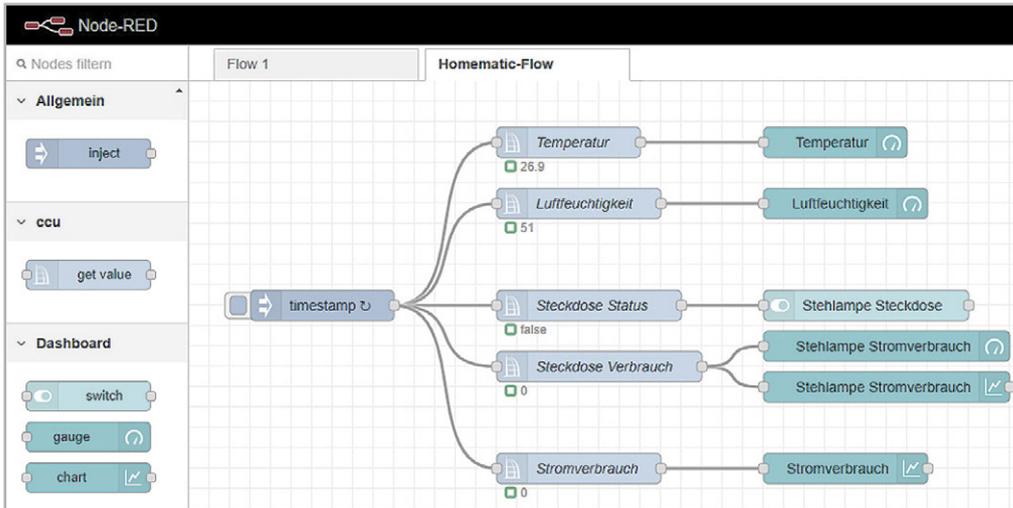


Bild 14: Homematic Flow zum Einlesen und Darstellen von Gerätewerten im Dashboard

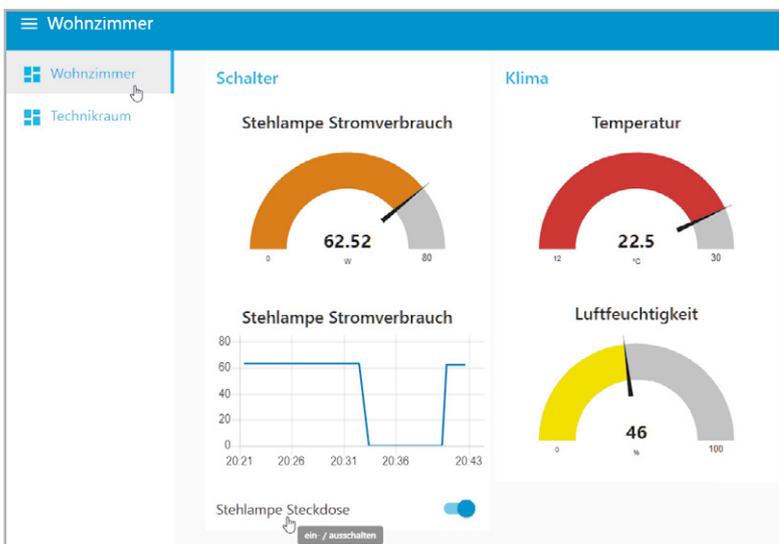


Bild 15: Dashboard-Beispiel in Node-RED

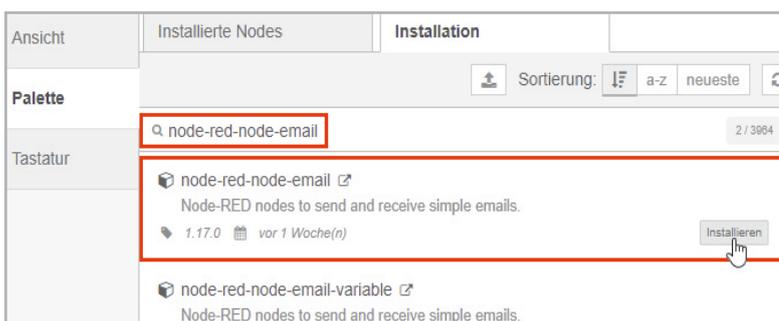


Bild 16: Installation der E-Mail-Contribution

Der nachfolgende E-Mail-Node erhält also nur einen Payload-Wert, falls die Bedingung erfüllt ist.

In den Eigenschaften des nachfolgenden E-Mail-versenden-Knotens müssen Sie zuerst die E-Mail-Credentials und den E-Mail-Empfänger eintragen:

- Empfänger-Adresse (An)
- Server-Adresse Ihres E-Mail-Accounts
- Port des E-Mail-Servers
- Ihren Benutzernamen und das zugehörige Passwort
- Klartext-Name des Empfängers für die Anzeige im Flow

Der E-Mail-Betreff und der Nachrichtentext verstecken sich in `msg.topic` und `msg.payload`.

Einen erfolgreichen E-Mail-Versand erkennen Sie kurz am Status „Sende“ und natürlich am Eintreffen der Nachricht in Ihrem Postfach.

Ein kleiner Tipp am Rande: Schalten Sie für erste Tests das Wiederholungsintervall des inject-Knotens ab und nutzen Sie einzelne Klicks auf den inject-Node. Denn sehr schnell sperrt der E-Mail-Server Ihren Account, wenn Sie ihn mit zu vielen Nachrichten fluten.

Fazit und Ausblick

Node-RED ist eine sehr gut gelungene, moderne Broker-Software und bringt einfach und schnell erste Erfolgserlebnisse. Wer sich von der englischen Dokumentation nicht abschrecken lässt, bekommt ein sehr mächtiges Werkzeug für das eigene Smart Home oder die eigene smarte Wohnung. Es macht sehr viel Spaß, sich mit Node-RED zu befassen und mit den Nodes zu jonglieren.

Nach Teil 4 unserer Broker-Serie werden wir in der nächsten Ausgabe auf IP-Symcon, eine kostenpflichtige Automatisierungslösung, eingehen. Die komplette Auflistung der jeweiligen Vor- und Nachteile aller vorgestellten Broker finden Sie zum direkten Vergleich ebenfalls in diesem fünften und letzten Beitrag unserer Artikelserie. **ELV**

The image displays the Node-RED interface for a 'Steckdosen-Alarm' (Outlet Alarm) flow. The flow consists of the following nodes: 'prüfe alle 60 Sekunden' (check every 60 seconds), 'Steckdose Verbrauch' (Outlet Consumption) with a value of 63.96, a 'function' node, and an 'email' node. Two panels provide detailed views of the 'function' and 'email' nodes.

Node 'function' bearbeiten:

```

1 var stromverbrauch = msg.payload;
2
3 if (stromverbrauch > 60 ) {
4
5     msg.payload = "hoher Stromverbrauch: " +
6                   stromverbrauch + " W";
7     msg.topic = "Verbrauchsalarm Wohnzimmer";
8
9 } else {
10    // do nothing
11 }
12 return msg;

```

Node 'email' bearbeiten:

- An: email@t-online.de
- Server: securesmtp.t-online.de
- Port: 465 Sichere Verbindung verwenden
- Benutzer-ID: thomas_mustermann@t-online.de
- Passwort:
- TLS: Überprüfen sie, ob das serverzertifikat gültig ist
- Name: Thomas Mustermann

Debug Console:

```

28.7.2022, 15:04:46 node: a2bf6bdcdb053a31
msg.payload: number
63.96
28.7.2022, 15:04:46 node: 9d6855cd296336e3
Verbrauchsalarm Wohnzimmer: msg.payload: string[25]
"hoher Stromverbrauch: 63.96 W"

```

Email Preview:

Verbrauchsalarm Wohnzimmer
 TM thomas_mustermann@t-online.de
 An email@t-online.de
 hoher Stromverbrauch: 63.96 W

Bild 17:
Automatisierungs-
Beispiel in Node-RED

i Weitere Infos

- [1] Vermittler im smarten Zuhause, Teil 1 – ioBroker – automate your life, ELVjournal 3/2022: Artikel-Nr. 252720
- [2] Vermittler im smarten Zuhause, Teil 2 – FHEM - Freundliche Hausautomation und Energie-Messung, ELVjournal 4/2022: Artikel-Nr. 252793
- [3] Vermittler im smarten Zuhause Teil 3 – openHAB – open Home Automation Bus, ELVjournal 5/2022: Artikel-Nr. 253020
- [4] Node-RED – Low-code programming for event-driven applications: <https://nodered.org/>
- [5] Udo Brandes: Node-RED: Das umfassende Handbuch, Rheinwerk Verlag GmbH, ISBN 978-3-8362-8559-9
- [6] Raspberry Pi – OS Download: <https://www.raspberrypi.com/software/>
- [7] balena Etcher SD-Card-Writer – Download: <https://www.balena.io/etcher/>
- [8] PuTTY SSH Client – Download: <https://www.putty.org/>
- [9] Node-RED-Dokumentation(englisch): <https://nodered.org/docs/>
- [10] Dokumentation zum Node-RED-dashboard: <https://flows.nodered.org/node/node-red-dashboard>
- [11] Node-RED Forum: <https://discourse.nodered.org/>

Alle Links finden Sie auch online unter: de.elv.com/elvjournal-links

Glossar

Contribution

Eine Contribution ist eine Sammlung von Nodes für Node-RED, die in einer Bibliothek (Node-RED nennt sie Palette) bereitgestellt werden und über den Palette-Manager installiert/verwaltet werden können.

Credentials

Unter Credentials versteht man einen Berechtigungsnachweis in Form von Login-Daten oder Anmeldeinformationen. Diese Angaben bestätigen einem System (z. B. einem E-Mail-Server) die Identität eines Benutzers. Der Nachweis geschieht in der Regel durch Benennen der Identität über einen Benutzernamen in Verbindung mit einem Passwort.

Dashboard

Das Dashboard ist die grafische Benutzeroberfläche (User Interface, UI) von Node-RED. Über besondere Dashboard- Nodes wird die Visualisierung realisiert.

Debug

Ein Debugger ist ein Werkzeug zum Diagnostizieren und Auffinden von Fehlern in Computersystemen oder Programmen. Debugging bezeichnet die Tätigkeit, Fehler zu diagnostizieren und aufzufinden. Node-RED stellt hierzu in der Palette „Allgemein“ einen debug-Node zur Verfügung.

Deploy/Übernahme

Unter Deployment versteht man die Prozesse zur Installation von Software. Bevor Änderungen in einem Flow ausgeführt werden, muss der Flow übernommen (deployed) werden.

Flow

Ein Flow ist der Datenstrom von einem Node zum anderen. In Node-RED repräsentiert ein Flow eine Registerkarte (Tab) im Arbeitsbereich und ist die Art und Weise, um die Knoten zu organisieren. In einem Flow werden unterschiedliche Eingabe-, Ausgabe- und Verarbeitungs-Nodes miteinander verbunden.

JSON

Das ist die Abkürzung für „JavaScript Object Notation“ und ist ein kompaktes, von Programmiersprachen unabhängiges Datenformat. In einer einfachen Textform erleichtert es den Datenaustausch zwischen Anwendungen.

Node/Knoten

Jeder Knoten hat eine festgelegte und eindeutige Aufgabe. Werden Daten an ihn übermittelt, so kann er diese Daten verarbeiten und an einen oder mehrere andere Knoten weitergeben. Die Weitergabe in einem Flow erfolgt in einem „msg“-Objekt (z. B. msg.payload).

Node.js

Node.js ist eine plattformübergreifende Open-Source-JavaScript-Laufzeitumgebung und kann JavaScript-Codes außerhalb eines Webbrowsers ausführen.

Palette

Die Palette zeigt die bereits in Node-RED installierten Contributions und ermöglicht über eine Suchfunktion das Hinzufügen weiterer Module.

Visualisierung

Unter Visualisierung (Sichtbarmachen) versteht man im Allgemeinen, dass abstrakte Daten und Zusammenhänge (z. B. Sensordaten und Texte) in eine graphische bzw. visuell erfassbare Form gebracht werden. Die (Standard-) Visualisierung in Node-RED ist das Dashboard.

Übersicht der Beiträge im ELVjournal zum Thema Broker-Software

ELVjournal Stand März 2022	Teil 1 ioBroker Automate your life	Teil 2 FHEM Freundliche Hausautomation und Energie-Messung	Teil 3 openHAB open Home Automation Bus
Ausgabe	3/2022	4/2022	5/2022
Download	https://www.iobroker.net/	https://fhem.de//fhem_DE.html	https://www.openhab.org/
Version(en)	v5.2.3 node.js V16.14.0 npm 8.3.1 js-controller 4.0.15	6.1	3.2.0 Java zulu 8.42.0.195
App	ioBroker (Android) ioBroker (iOS)	andFHEM (Android) FHEM-Remote (iOS)	openHAB (Android) openHAB (iOS)
Skripting	Rules, JavaScript, Typescript, Blockly	PERL in 99_myUtils	Rules, Groovy, JavaScript, JRuby, Jython
Homematic Anbindung	Adapter Homematic ReGaHSS 2x Adapter Homematic RPC	HMCCU	Homematic Binding
Server-Adresse	[IP-Adresse]:8080	[IP-Adresse]:8083	[IP-Adresse]:8080
Admin-Tool	[IP-Adresse]:8081	[IP-Adresse]:8083/fhem	[IP-Adresse]:8101
Visualisierung	Adapter VIS [IP-Adresse]:8082/vis/ Adapter jarvis [IP-Adresse]:8082/jarvis/	Dashboard Floorplan	HABPANEL
Dokumentation/Wiki	https://www.iobroker.net/#de/documentation	https://www.fhemwiki.de/wiki https://wiki.fhem.de/wiki/Hauptseite	https://www.openhab.org/docs
Forum	https://forum.iobroker.net	https://forum.fhem.de	https://openhabforum.de
ELVjournal Stand März 2022	Teil 4 Node-RED a flow-based programming tool	Teil 5 IP-Symcon Smart Home ohne Kompromisse	
Ausgabe	6/2022	1/2023	
Download	https://nodered.org/	https://www.symcon.de/	
Version(en)	v2.2.2 node.js V14.19.3 npm 6.14.17	6.3	
App	Termux mit node-RED (Android) Remote-RED (iOS)	IP-Symcon Mobile (Android) IP-Symcon Mobile (iOS)	
Skripting	Flows, JavaScript	PHP 7.4	
Homematic Anbindung	node-red-contrib-ccu node-red-contrib-homematic	Homematic Socket	
Server-Adresse	[IP-Adresse]:1880	[IP-Adresse]	
Admin-Tool	[IP-Adresse]:1880	[IP-Adresse]:3777/console	
Visualisierung	Dashboard [IP-Adresse]:1880/ui	Webfront [IP-Adresse]:3777	
Dokumentation/Wiki	https://nodered.org/docs	https://www.symcon.de/service/dokumentation	
Forum	https://discourse.nodered.org	https://community.symcon.de	