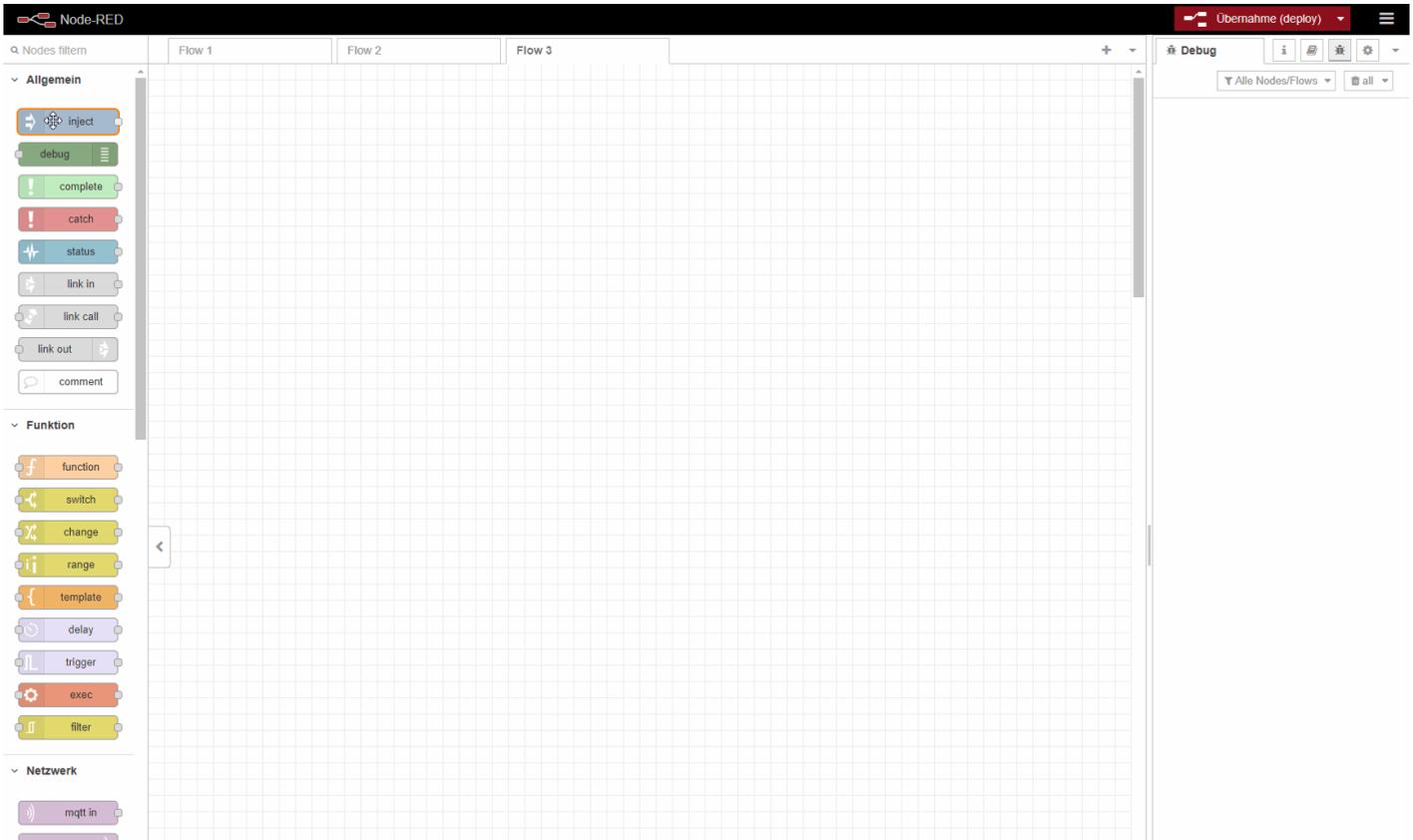


Node-Red Einführung

Diese Seite gilt als kleine Heranführung an Node-Red und dem Umgang der GHG-Notary.



1. Flow

In diesem Flow wird der TimeStamp als simulierter Stromzähler verwendet. Da die Ausgabe werde von Typ Integer und fortlaufend sind eignet er sich hervorragend.

Durch Debug kann man sich die Inhalte des payloads in der Debug-Leiste Rechts den Payload ausgeben lassen.

Mit dem [Metered-Notary](#) erstellen wir einen Beleg über den Strombezug basierend auf den Zählerstandsgang (Abfolge der Zählerstände eines Stromzählers). Der grüne Kasten unter dem Node zeigt uns, dass die Erstellung des Belegs erfolgreich war. Es kann eine vorhandene Wallet ausgewählt oder neu hinzugefügt werden.

2. Flow

In diesem Flow wird TimeStamp nur als Trigger/Inject genutzt. Dieser ermöglicht das manuelle Starten des Flows.

Der nächste Node **Certificates** holt sich alle Zertifikate, die wir in in Flow 1 in die Wallet abgelegt haben.

Mit einem weiteren Debug geben wir uns diese dann in dem Debug-Fenster aus.

In dem nächsten Schritt wird aufgezeigt, dass der **Present-Node** nur funktioniert, wenn diesem Zertifikate übergeben werden. Auch wenn die Wallet in dem Node angegeben wird, benötigt der Node die Übergabe von Zertifikaten, die dann für Dritte teilweise kenntlich gemacht werden.

Ähnlich funktioniert auch der **Presentations-Node**.

The screenshot shows the Node-RED interface with a flow in Flow 3. The flow starts with a 'Stromzähler' node connected to a 'Zertifikat' node (Certificate 0xb2552a9e42cb2...). The 'Zertifikat' node is connected to a 'Zählerstand' node. Below this, a 'timestamp' node is connected to a 'Certificates.TestWallet' node. The 'Certificates.TestWallet' node is connected to three parallel paths: 'debug 3', 'Present.TestWallet', and 'TestWallet'. Each path ends with a 'debug' node (debug 4 and debug 5). The Debug console on the right shows the output of the 'Present' node, which is an array of objects. The first object is a presentation with the following details:

```
presentation:
  "0x27f0cb3637758876efefc2ae5dbc7cd4cd06c2763dde77bef2ff66ebd6d59c7"
time: 1674557355288
owner:
  "0x56d1c6cd4f088879b2A9c0fd051f0025bce075"
issuer:
  "0xE7Fe0626D788e3F2e5ECD146F9b11daac1D8E447"
schema:
  "https://schema.corrently.io/tydids/gh8"
```

Mit dem Statistics-Node kann man sich von allen Zertifikaten in der angegebenen Wallet eine Zusammenfassung ausgeben lassen (Anzahl, sowie Summe der Treibhausgasemissionen, Stromverbrauch, Einsparungen, sowie deren öffentlichen Schlüssel/address der wallet)

Revision #1

Created 24 January 2023 10:53:17 by Christoph Dorus

Updated 24 January 2023 11:40:02 by Christoph Dorus