



VERSION **4.2.8 - p**

**Ein großer Schritt nach vorn – und ein Blick in die Zukunft!**

Wir arbeiten intensiv am neuen, grundlegend überarbeiteten Evolution. Aufgrund mehrerer positiver Entwicklungen und laufender Verbesserungen wird die nächste Hauptversion etwas länger brauchen als ursprünglich geplant.

# INHALT

## ▪ Ein großer Schritt nach vorn – Version 4.2.8-p

- Warum Version 4.2.8-p wichtig ist
- Neuigkeiten auf einen Blick

### ◦ **Neuigkeiten und Verbesserungen in Version 4.2.8-p**

- 1. Neues Konfigurationspanel in Webmin
- 2. Intelligentere System-Updates
- 3. Verbesserte Verwaltung von Prozesspunkten
- 4. Der Statuspunkt wird nun automatisch hinzugefügt, wenn ein Prozesspunkt erstellt oder aktualisiert wird.
- 5. Konfiguration von Prozesspunkten
- 6. Bessere Visualisierung und Analytik
- 7. Verbesserte Integration von Smart Metering und DALI
- 8. Sicherheit, Benachrichtigungen und Automatisierung
- 9. Verbesserte Konnektivität und IoT-Integration
- 10. Flexiblere Vorlagen und Szenen
- 11. Besseres Logging und Systeminformationen
- 12. Evolution BMS – MQTT-Integration

## ▪ **Ausblick: Der nächste große Release ist Version 5.0.**

- LoRaWAN-Integration
- KNX Secure gemäß den Anforderungen des Cyber Resilience Act für sichere Kommunikation
- Erweiterbares Smart-Metering-Modul mit umfangreichen kundenspezifischen Anpassungsmöglichkeiten und einem sehr leistungsfähigen Berichtswerkzeug

## Ein großer Schritt nach vorn - VERSION 4.2.8-p

### Warum Version 4.2.8-p wichtig ist

Während wir die nächste Generation von Evolution vorbereiten, wollten wir nicht warten, bis du von den neuen Funktionen erfährst. Daher veröffentlichen wir **Evolution BMS 4.2.8-p als Pre-Release**. Dieses Update bringt Verbesserungen und neue Funktionen, die dein System sofort leistungsfähiger und einfacher machen. Gleichzeitig wird die Plattform für den nächsten großen Sprung vorbereitet.

### Neuigkeiten auf einen Blick

Mit Version 4.2.8-p wird Evolution BMS um einige Funktionen erweitert.

Version 4.2.8-p führt Evolution BMS ein gutes Stück weiter. **Updates** sind nun **intelligenter und zuverlässiger**, mit Außerdem gibt es vereinfachte Systembenachrichtigungen und eine cloudbasierte Firmwareverwaltung. Vereinfachten **Systembenachrichtigungen** und **Cloud-basierter Firmwareverwaltung**. Die **Verwaltung von...** **Prozesspunkten** wurde verfeinert, einschließlich der automatischen Erstellung von Statuspunkten und leichter Konfigurationsoptionen – das beschleunigt die Inbetriebnahme und reduziert Fehler.

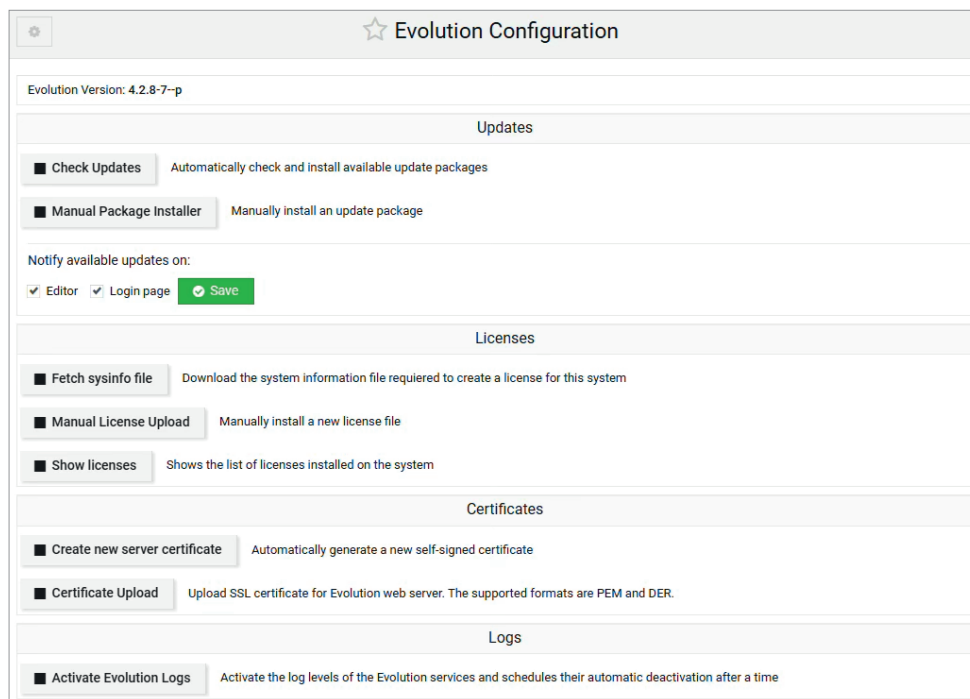
Die Prozesspunkte wurden verfeinert, einschließlich der automatischen Erstellung von Statuspunkten und einer erleichterten Handhabung. Das beschleunigt die Inbetriebnahme und reduziert Fehler.

Die **Visualisierung und Analytik** sind präziser, mit erweitertem **Logging**, besseren **Diagrammwerkzeugen** und einer Die Visualisierung und Analytik sind präziser und verfügen über erweiterte Logging-Funktionen, bessere Diagrammwerkzeuge und einer verbesserten Systemübersicht. Die Integration von Smart Metering und DALI ist intuitiver und flexibler. Gleichzeitig werden Sicherheit und Automatisierung durch neue Benachrichtigungs- und Alarmfunktionen gestärkt. verbesserten Systemübersicht. Die **Integration von Smart Metering und DALI** ist intuitiver und flexibler, während **Sicherheit** und **Automatisierung** durch neue **Benachrichtigungs- und Alarmfunktionen** gestärkt werden.

Auch die Konnektivität wird verbessert: Durch die Integration von Node-RED und MQTT werden IoT-Flows schneller und ermöglicht eine reibungslose, protokollübergreifende Automatisierung. Schließlich sind Vorlagen und Szenen flexibler. Integratoren haben so mehr Freiheit, Logik zu entwerfen, wiederzuverwenden und zu teilen.

## Neuigkeiten und Verbesserungen in Version 4.2.8-p

### 1. Neues Konfigurationspanel in Webmin

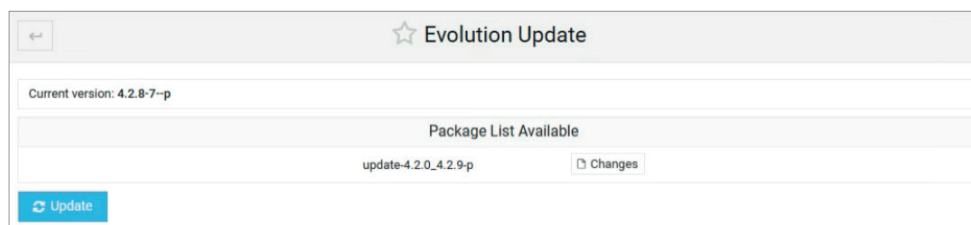


Das Server-Werkzeugpanel in Webmin wurde vollständig überarbeitet und integriert, um eine zentralere und flüssigere Administration zu ermöglichen. Wichtige Neuerungen:

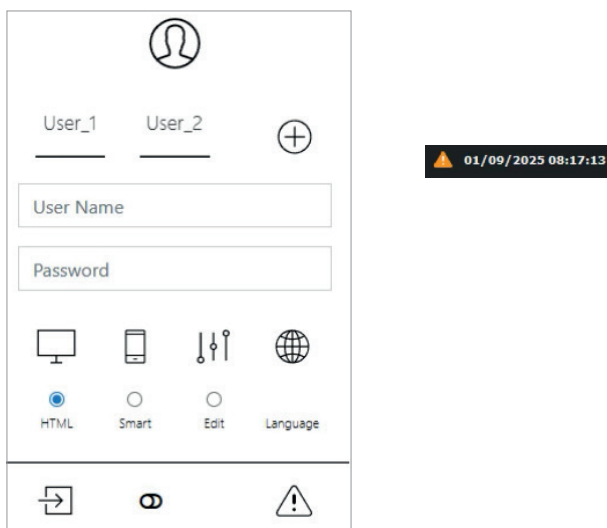
- **Vereinheitlichte Integration:** Alle Server-Admin-Werkzeuge sind in einem einzigen Panel vereint – kein verstreutes Menü mehr.
- **Smart Server Update:** Das neue Tool hilft dabei, den Server effizienter und sicherer aktuell zu halten.
- **Lizenzverwaltung:** Die Lizenzverwaltung bietet einen schnellen und klaren Überblick über alle installierten Lizenzen.
- **Installation von SSL-Zertifikaten:** eigenes Werkzeug für einfache Installation und Verwaltung von Sicherheitszertifikaten.
- **Steuerung der Systemlogs:** Funktion zum Aktivieren, Deaktivieren und Herunterladen von Logs direkt aus dem Panel. Die logs werden nach einer konfigurierbaren Zeit automatisch deaktiviert.
- **Update-Hinweise:** Du kannst festlegen, wo verfügbare Update-Hinweise angezeigt werden sollen – im Editor oder direkt in der Visualisierung (Visu).

## 2. Intelligenter System-Updates

- **Firmware-Aktualisierung aus der Cloud:** Halte dein System direkt aus der Cloud auf dem neuesten Stand.



- **Neue Update-Benachrichtigungen Die:** Hinweis-Icons im **Editor** und im **Login-Bildschirm** informieren sofort, wenn ein Update verfügbar ist.



## 3. Verbesserte Verwaltung von Prozesspunkten

- Der verbesserte **Startvorgang** sorgt für eine reibungslosere Initialisierung des Systems.
- Es gibt eine option zum **automatischen Senden** des **Standardwerts** oder des **zuletzt empfangenen Werts**, wenn der Status **unbekannt** ist.
- Die **Interpolation** von Prozesspunkten ist **standardmäßig deaktiviert**, um eine maximale Kontrolle zu gewährleisten.
- **Import und Export** von Seiten mit **Vorlagen** für Prozesspunkte ermöglichen eine schnellere Konfiguration und eine leichtere Migration zwischen Projekten.
- Der Import von **Gruppenadressen aus ESF- und KNXPROJ-Dateien** wurde verbessert und priorisiert die ersten. Die erste Adresse wird priorisiert, um Duplikate zu vermeiden.
- **Import** von **Statusgruppenadressen** über **Präfix-/Suffixabgleich** zulassen.

#### 4. Automatisches Hinzufügen des Statuspunkts beim Erstellen/Aktualisieren eines Prozesspunkts:

Import KNX files

1.) Select gateway KNXnet/IP: test

2.) KNX data: Charset UTF-8 Load data

3.) Options: ☐ Update of data and process points ☐ Apply advanced options Advanced

4.) Filter: Address

5.) Selection: Create process point Import

6.) Automatically Add Status Data Point When Creating or Updating a Process Point:

Status Text 1: Status Example: Light1 -> Status\_Light1

Status Text 2: FB Example: Light1 -> Light1\_FB

Example: Light1 -> Light\_FB\_1

Note. Spaces and special characters are ignored

Address	Name	Type	Subtype	Create process point	Import
0/0/1	Request date / time	1	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0/0/2	Date	18	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0/0/3	Time	1	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0/0/4	Date / time	1	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0/0/5	Disable screen save	1	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0/0/6	Switching	1	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0/0/7	Switching feedback	1	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0/0/8	Switching operating	1	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0/0/9	Forced object operat	1	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Previous Next Add Cancel

- Das Programm kann den zugehörigen **Statuspunkt erkennen** und **verknüpfen**.  
Die **Status-Gruppenadresse** muss einem **Namensmuster** folgen, das auf der **Sende-Gruppenadresse** basiert.

#### Erforderliche Felder:

- Statustext 1 und Statustext 2 (optional):** Wenn einer oder beide in der Statusgruppenadresse erscheinen, entspricht der Rest des Statustextes genau der Sendegruppenadresse. Beispiel: Senden von `livingroom/light`, Status, `_fb_livingroom/light` ( `_fb_` = Statustext; der Rest `livingroom/light` = Sendegruppenadresse).

#### Hinweise:

- Leerzeichen** und **Sonderzeichen** werden ignoriert.
- Existiert nur eine **Status-Gruppenadresse** ohne passende Sende-Gruppenadresse, wird ein **neuer Prozesspunkt ohne Senden** erstellt.

**Beispiel 1 – Normalfall mit Sende- und Status-Gruppenadressen:**

- Sende-Gruppenadresse: Light1
- Statustext 1: Status
- Statustext 2: FB
- Passende Status-Gruppenadressen: Status\_light1, Light1\_FB, Light\_FB\_1
- **Erstellter Prozesspunkt:**
  - **Sende-Datenpunkt:** Gateway:1/1/1 (Flag **Senden** aktiviert)
  - **Status-Datenpunkt:** Gateway:1/1/101 (Flag **Lesen vom Bus** aktiviert)

**Beispiel 2 – Es existiert nur die Status-Gruppenadresse:**

- **Erstellter Prozesspunkt:**
  - **Sende-Datenpunkt:** Keiner
  - **Status-Datenpunkt:** Gateway:1/1/101 (Flag **Lesen vom Bus** aktiviert)

**5. Konfiguration von Prozesspunkten**

Process point

Id: 22353 Name: Light room - Active power

Name in modules:

General Advanced Processing Recording Visibility Value-Text

**Times and Delays**

Delay after reads: 0 ms

Delay after sending: 0 ms

Polling time: 0 ms

Read delay after sending: 0 ms

**Initial Value of Process Point at Startup**

☐ Do not initialize

☒ Default value 1 W

☐ Last received value 12 W

**Initial Startup Send Behavior**

☒ Don't send

☐ Send

**Runtime Behavior**

☐ Send only if value changes

☐ Read value on each use

When status is unknown

☒ Stay unknown

☐ Use initial value configuration

Save Cancel

**Zeiten und Verzögerungen:**

- Verzögerung nach Lesevorgängen (ms)
- Verzögerung nach Sendevorgängen (ms)
- Abfrageintervall (ms)
- Leseverzögerung nach dem Senden (ms)

**Initialwert beim Start:**

- Nicht initialisieren
- **Standardwert** (empfohlen)
- Zuletzt empfangener Wert

**Sendeverhalten beim Start:**

- Nicht senden
- Sofort senden

**Laufzeitverhalten:**

- **Nur bei Änderung** senden (optional)
- Wert bei **jeder Verwendung** lesen

**Umgang mit unbekanntem Status:**

- **Unbekannt** bleiben (Standard)
- **Initialwert-Einstellung** verwenden

**6. Bessere Visualisierung und Analytik**

- **Verbesserungen bei Diagrammen:**
  - Die **horizontale Achse** kann Werte eines **Prozesspunkts** verwenden.
  - Auswahl der **Zeiteinheit** für Betriebsstunden (**Sekunden, Minuten oder Stunden**).
  - Auswahl der Daten-Spalten, die exportiert werden sollen.
  - Export **vollständiger Reihen**, einschließlich **Vorlagen**.
  - Der export respektiert die **Regionseinstellungen** (Dezimaltrennzeichen).
- Erhöhung der **Zeichenanzahl** in **Alarm-E-Mails**.
- **Längere Aufbewahrung** von Systemlogs für historische Nachverfolgung.



## 7. Verbesserte Integration von Smart Metering und DALI

- **DALI-Berichte** können direkt mit dem **Export-Tool** ausgegeben werden.
- Umbenennen von **ECG** und **Gruppen** direkt im **DALI-Modul**.
- Einfachere Verwaltung und Navigation der **Favoriten** im **Smart Metering**.

## 8. Sicherheit, Benachrichtigungen und Automatisierung

- **Hysterese** für **Alarmschwellenwerte**.
- **E-Mail-Benachrichtigungen** können nun auch **exportierte Dateien** (bis zu **10 MB**) enthalten.
- Ein **E-Mail-Feld** wurde außerdem in den **Benutzerprofilen** ergänzt.

## 9. Verbesserte Konnektivität und IoT-Integration

- **Mosquitto (MQTT)** wurde aktualisiert und verfügt nun über ein **Webmin-Panel**, mit dem sich der Dienst aktivieren bzw. deaktivieren lässt.
- **Digest-Authentifizierung** und bessere Verwaltung von **Verzögerungen** für **Netzwerkcameras**.
- Neuer **SRS-Dienst** zur Verbesserung des **IP-Kamera-Streamings**.

## 10. Flexiblere Vorlagen und Szenen

- **Vollständig anpassbare Vorlagen**: beliebiger Text, **Parameterübergabe**, Vererbung von Werten aus **Eltern-Vorlagen**.
- **Import und Export** kompletter Szenen im **XML-Format**.

## 11. Besseres Logging und Systeminformationen

- Das **Protokoll** speichert nun **An- und Abmeldungen**.
- **Benachrichtigungen** werden direkt im **Dashboard** angezeigt.

## 12. Evolution BMS – MQTT-Integration

### Einführung in MQTT

**MQTT (Message Queuing Telemetry Transport)** ist ein **leichtgewichtiges Publish/Subscribe-Nachrichtenprotokoll**, eine schnelle und effiziente Kommunikation zwischen Geräten entwickelt. Es wurde für eine schnelle und effiziente Kommunikation zwischen Geräten entwickelt.

Es wird in den Bereichen **IoT** und **Gebäudeautomation** weit eingesetzt, weil es:

- **minimalen Bandbreitenverbrauch** hat.

- auf **instabilen Netzen zuverlässig** funktioniert und
- eine reibungslose Kommunikation zwischen **vielen Geräten** ermöglicht.

In MQTT basiert die Kommunikation auf drei Hauptkomponenten:

- **Publishers** → senden Nachrichten an **Topics**.
- **Subscribers** → empfangen Nachrichten von **Topics**.
- **Broker** (vermittelnder Server) → verwaltet die Verteilung der Nachrichten zwischen Publishern und Subscribern.

### Node-RED und Evolution BMS

**Node-RED** ist ein leistungsstarkes **visuelles Programmierwerkzeug**, das in das **Evolution BMS** integriert ist. Es ermöglicht die Erstellung **flexibler Automatisierungs-Flows** zwischen Geräten, Diensten und Protokollen – und das ohne komplexe Programmierung.

Mit Evolution BMS kann Node-RED direkt mit den folgenden Protokollen interagieren:

- **KNX**
- **BACnet**
- **Modbus**
- **DALI**
- **MQTT**
- und weiteren unterstützten Protokollen

Die **Drag-&Drop-Oberfläche** vereinfacht **Integration und Automatisierung** und ist somit ideal, um **leistungsfähige Steuerlogiken** schnell zu erstellen.

## MQTT-Integration in Evolution BMS

**MQTT Configuration**

MQTT Service is stopped  
MQTT Service is disabled, so it does not start on boot

Start Stop Restart Start on boot Disable on boot Start now and on boot Disable now and on boot

**Evolution BMS MQTT Integration**

The Evolution BMS server includes a built-in MQTT broker, configurable from the Webmin panel. Once enabled, it automatically registers all MQTT topics used by publishers and subscribers (no manual topic setup required). In Node-RED, MQTT topics can be directly linked to Evolution process points for KNX integration (or any other Evolution protocol).

**Example 1:**

- A simulated temperature sensor injects 25 into the topic `mqtt/sensor1/temp/value`.
- This topic is connected to process point 13 and sent to the KNX bus.
- The subscriber node also outputs the received value (25%) in the debug panel.

**Example 2:**

- An Evolution input takes values from the KNX bus and publishes them to the topic `evolution/sending-values`.
- In this setup, it receives the value sent in Example 1 and displays it in debug.

**Example 3:**

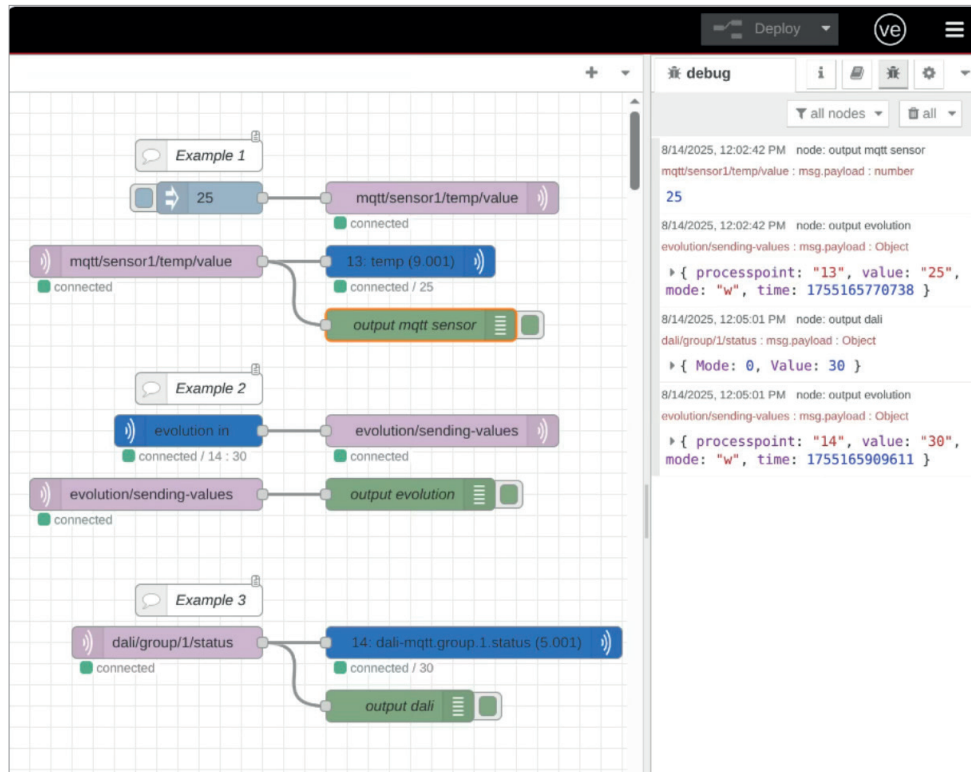
- The subscriber node listens to the topic `dali/group1/status` from a IPAS DalControl ed4 Pro device.
- In this setup, it receives the value from the dali device and sends it to the process point 14.

Der **Evolution-BMS-Server** verfügt über einen **integrierten MQTT-Broker**, der vollständig über das **Webmin-Panel** konfiguriert werden kann.

### Schlüsselmerkmale:

- **Integrierter MQTT-Broker** → kein externer Broker erforderlich.
- **Automatische Topic-Erfassung** → alle von Publishern und Subscribern verwendeten Topics werden automatisch registriert.
- **Integration mit Node-RED** → MQTT-Topics lassen sich direkt mit **Prozesspunkten** in Evolution verknüpfen, um sie mit **KNX** oder anderen unterstützten Protokollen zu integrieren.

## Beispiele



## Beispiel 1: MQTT-Sensor → Evolution → KNX

1. Ein simuliertes Temperatursignal veröffentlicht den Wert **25** im Topic `mqtt/sensor1/temp/value`.
2. `mqtt/sensor1/temp/value`.
3. Dieses Topic wird in **Node-RED** mit **Prozesspunkt 13** verbunden.
4. **Evolution** sendet den Wert auf den **KNX-Bus**.
5. Der **Subscriber-Knoten** zeigt den empfangenen Wert (**25°C**) außerdem im Debug-Panel an.

## Beispiel 2: KNX → Evolution → MQTT

1. Ein **Eingang** in Evolution erhält Werte vom **KNX-Bus**.
2. Diese werden im Topic
3. `evolution/sending-values` veröffentlicht.
4. In dieser Konfiguration empfängt er den im **Beispiel 1** gesendeten Wert und zeigt ihn im Debug-Panel an.

### Beispiel 3: DALI → Evolution → KNX

1. Der **Subscriber-Knoten** hört auf das Topic
2. dali/group1/1/status.
3. Die Werte stammen von einem **IPAS DaliControl e64 Pro**.
4. Evolution sendet diese Werte an **Prozesspunkt 14** zur Nutzung in **KNX** oder anderen Protokollen.

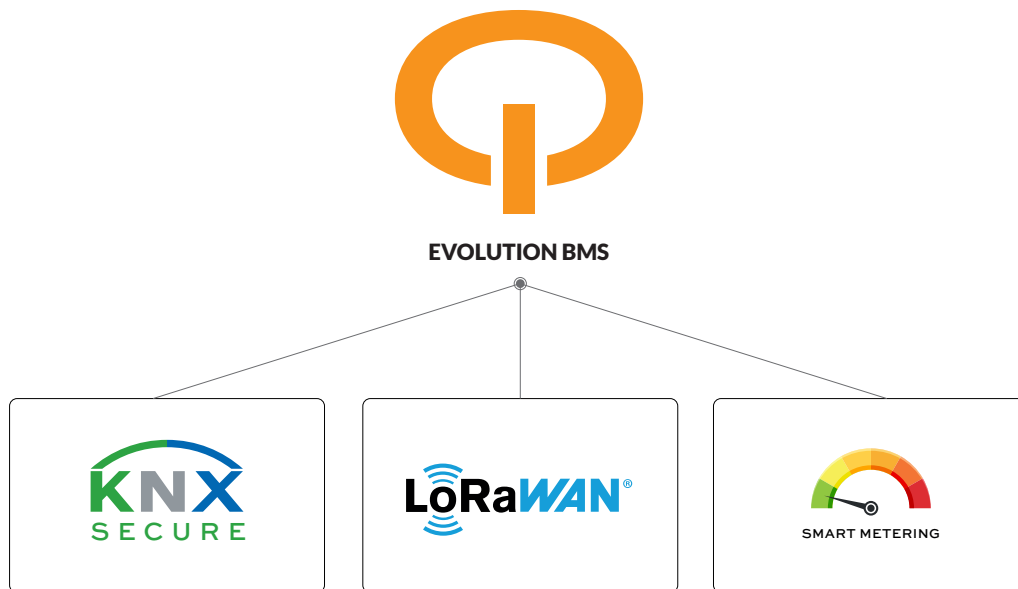
### Zusammenfassung und Best Practices

- Nutze den **integrierten MQTT-Broker** von Evolution für eine nahtlose Integration.
- Nutze **Node-RED**, um **flexible Automatisierungs-Flows** zwischen verschiedenen Protokollen zu erstellen.
- Verwende **aussagekräftige Topic-Namen** für eine bessere Struktur.
- Aktiviere **TLS-Verschlüsselung** und **Authentifizierung**, wenn du den MQTT-Broker nach außen verfügbar machst.

### Warum sollte auf Evolution BMS 4.2.8-p aktualisiert werden?

Bei diesem Release liegt der Fokus auf **intelligenteren Updates**, einer **besseren Datenvisualisierung** und **verbesserten IoT-Integration** sowie eine optimierte Benutzerfreundlichkeit für Integratoren und Endanwender. Mit optimierten Konfigurationstools und leistungsfähigeren Analysen bleibt Evolution BMS eine der **vielseitigsten und skalierbarsten Gebäudemanagementlösungen** auf dem Markt. Aktualisieren Sie noch heute und seien Sie bereit für die revolutionäre nächste Version!

## Ausblick – Nächster großer Release Version 5.0



- **LoRaWAN-Integration.**
- **Implementierung von KNX Secure**, konform zu den Anforderungen des **Cyber Resilience Act** für sichere Kommunikation.
- **Erweitertes Smart-Metering-Modul** mit umfassenden kundenspezifischen Anpassungen und einem **leistungsfähigen Berichtswerkzeug**.

Diese Erweiterungen machen den nächsten Release **deutlich leistungsfähiger und zukunftssicher**, sodass **Evolution** weiterhin neue Maßstäbe im Gebäudemanagement setzen wird. Die nächste Hauptversion wird eine **vollständig neu gestaltete Benutzeroberfläche** enthalten, die modern, intuitiv ist und über **wiederverwendbare, responsive Widgets** verfügt, die flexibel in **Dashboards** eingesetzt werden können.

Darüber hinaus wird das erweiterte Smart-Metering-Berichtswerkzeug die wichtigsten Widgets direkt im neuen Panel integrieren, um eine noch stärkere Analytik zu erreichen.