PACEdge User Manual
GFK-3178H
Mar 2024

PACEdge™
USER MANUAL

Contents

Warnings and Caution Notes as Used in this Publication

⚠ WARNING

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use.

In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.

⚠ CAUTION

Caution notices are used where equipment might be damaged if care is not taken.

Note: Notes merely call attention to information that is especially significant to understanding and operating the equipment.

These instructions do not purport to cover all details or variations in equipment nor to provide for every possible contingency to be met during installation, operation, and maintenance. The information is supplied for informational purposes only, and Emerson makes no warranty on the accuracy of the information included herein. Changes, modifications, and/or improvements to equipment and specifications are made periodically, and these changes may or may not be reflected herein. It is understood that Emerson may make changes, modifications, or improvements to the equipment referenced herein or to the document itself at any time. This document is intended for trained personnel familiar with the Emerson products referenced herein.

Emerson may have patents or pending patent applications covering the subject matter in this document. The furnishing of this document does not provide any license whatsoever to any of these patents.

Emerson provides the following document and the information included therein as-is and without warranty of any kind, expressed or implied, including but not limited to any implied, statutory warranty of merchantability or fitness for a particular purpose.

Section 1: Introduction

PACEdge processing maximizes the value of your data by improving operational reliability, safety, and energy consumption. PACEdge provides all aspects of edge processing and simplifies your IIoT application development, deployment, and administration. All components necessary in the IIoT application lifecycle are combined in one package to provide a unified interface to decrease your development time and increase your deployable footprint.

1.1 Revision History

Rev	Date	Description
H	Mar 2024	Adds support for PACEdge v2.4 release
G	Oct 2023	Updates related to the PACEdge v2.3 release
F	Mar 2023	Added cautions warning against the use of dollar signs ($) in passwords under Section 3.4 Initial Login.
E	Nov 2022	Updates related to the PACEdge v2.2 release
D	Dec 2021	Added details on Accessing the Connext OPC UA Server (Section 0)
C	Sep 2021	PACEdge 2.1 Initial release
B	Nov 2020	Added Section 7.2 Node-RED Dashboard Performance Indication
A	Sep 2020	PACEdge 2.0 Initial release

1.2 New Features in PACEdge 2.3

1.2.1 Basic Group Management Tool

As of PACEdge version v2.3, a basic version of the Group Management tool has been added. The feature is deigned to simplify the process of deploying, managing, and updating software on hundreds of the devices. In its initial release, the Group Management tool will support updates for Host Linux updates, PACEdge Application Updates, and Node-RED flow deployments. Users will have the ability to create groups of PACEdge devices and perform updates on all devices within a group. For more information about Group Management, please refer to Section 4.5,Group Manager.

1.2.2 Certificate Management Utility

As of PACEdge version v2.3, Certificate Management Utility has been added. Its primary purpose is to assist users in verifying that they are connecting to the PACEdge systems through their web browser. When functioning properly, this utility eliminates the security warning that appears when an untrusted device certificate is detected. Users have various options for implementing certificates:

- Trusting and installing the PACEdge CA certificate.
- Creating their own PKI and generate certificates for own PACEdge devices.
- Installing their own certificates that were generated by already existing PKI.

For more details about Certificate Management Utility and its use refer to Section 4.7, *Certificate Management.*

1.2.3 TimescaleDB Database

TimescaleDB is a type of PostgreSQL database that is specifically designed for managing time series data and has a user-friendly SQL interface. In the latest PACEdge v2.3 release, TimescaleDB has been included as a third database option, in addition to InfluxDB and MySQL. Going forward, TimescaleDB is the recommended database for both time series and relational data for PACEdge applications. Users employing MySQL or InfluxDB for their PACEdge applications are strongly encouraged to migrate to TimescaleDB at their earliest convenience.

1.2.4 phpMyAdmin

phpMyAdmin is a web-based management tool that offers a graphical interface for managing MySQL databases.

1.2.5 Python Container and Example (optional)

PACEdge v2.3 includes an optional Python container and Python-Node-RED application example. Refer to Section 5.11 *Python Container (optional),* for instructions on enabling the container and use of the example.

1.2.6 Applications Updated to a Newer Version

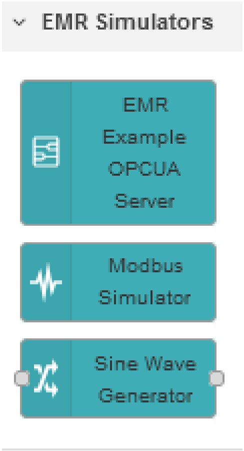
In PACEdge v2.3, all applications have been updated to a newer version. GFK-3198, PACEdge IPI lists the exact versions of each application, but the most important updates are as follows:

- Updated Movicon Connext © and WebHMI to Movicon.NExT™ v4.2.359
- Updated Node-Red to version 3.0.2. This includes functional updates, new nodes, improved editor and look and feel improvements.
- Updated Grafana® to version 9.5.2.

1.2.7 Functional Improvements in Node-RED

New examples are now available in the **Import->Examples->node-red-contrib-emerson** section, which now include an OPC-UA Server Simulator, an example Modbus TCP Simulator, and example Sine Wave generator. Upon installation, PACEdge-customized nodes will be added to the palette EMR Simulators category, which can be utilized to simulate external devices that generate data. A detailed description of how to use these examples flows and nodes is in *Section 5: PACEdge Usage Examples.*

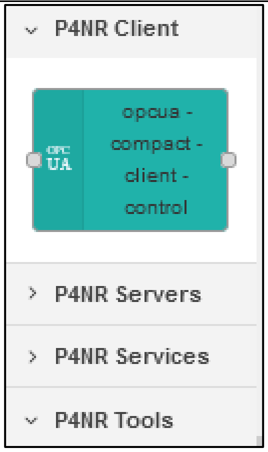
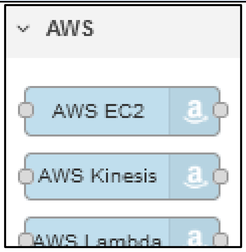
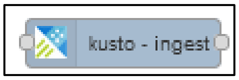
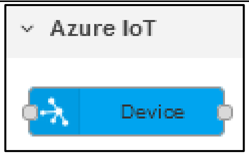
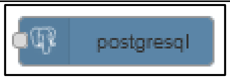
Figure 1: Data Source Simulators in Node-RED



.

Note: To have Emerson customized nodes added to the Node-RED node palette; import the corresponding example flow, as described in sections: *5.2, 5.3, 5.4, 5.8, 5.9, 5.10*

Emerson pre-configured data source simulators for Sine Wave, OPC-UA, Modbus TCP:

New Nodes	Node Icon
New OPC-UA node palette from Plus for Node-RED	 P4NR Client OPC UA opcua - compact - client - control P4NR Servers P4NR Services P4NR Tools
Cloud Nodes	
AWS Nodes	
Azure Kusto and IoT nodes	 
Postgresql Node to access new TimescaleDB database	

Section 2: Physical Connections and Configuration Settings

2.1 Network, Keyboard, Display

PACEdge is designed to operate on a headless computing device using web interfaces; however, if desired, it can be connected to a monitor, keyboard, and mouse using an RXi2-BP, RXi2-LP, or IPC 2010 Industrial PC.

To use the web interfaces, the user should be aware of the Ethernet port's assigned IP address or the device's host name. The IP address assignment can be easily checked using a keyboard and local display using Linux commands. Otherwise following options are available:

PACEdge Factory Installation					
Static IP on 1st Ethernet port: 192.168.3.100				DHCP assigned IP	
RXi2-LP	RXi2-BP	IPC 2010	CPL410/CPE400	RXi2-LP/ RXi2-BP/IPC 2010	CPL410/ CPE400
LAN port next to the USB port	ETH0 port	Port 2	ETH port	Any Ethernet port	ETH port

1. Follow the Ethernet Port and IP address guidance above.
2. Attach the Ethernet cable to an appropriate Ethernet port.

Optionally (RXi2-LP, RXi2-BP, IPC 2010), attach a USB keyboard to any USB ports.

Optionally (RXi2-LP, RXi2-BP), attach a display to the DP++ port.

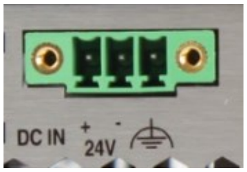
Optionally (IPC 2010), attach a custom USB-C to DP++ port cable. (Available as an accessory from Emerson. Refer to the IPC 2010 datasheet). **Note**: This is a special cable. Do not attempt to use off-the-shelf cables!

Note: DisplayPort (DP++) connection is well suited for conversion to HDMI and VGA. (Conversion adapters are not included).

2.2 Power

1. Connect a 24V DC +/-25% power to the power connector and power up the unit (Figure 2).
2. The corresponding plug must be a Phoenix Contact “1748367” or equivalent.
   RXi2-LP, RXi2-BP, IPC 2010, CPL410, and CPE400 can be powered with a DC power supply with 24V DC (+-25%) and at least delivering 2 A.

Figure 2: 24 V DC-IN



Signal Name	Pin (left to right)
Power + (24V DC)	1
Power - (24V DC)	2
FGND	3

Section 3: PACEdge Getting Started

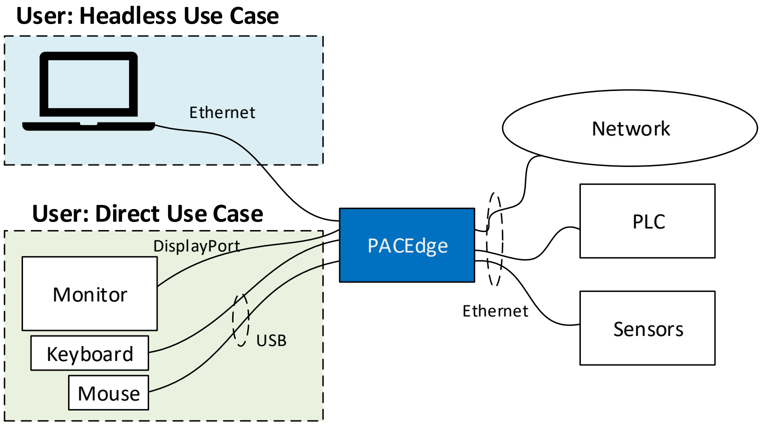
Note: PACEdge software comes pre-installed on Emerson Industrial PCs and Controllers.

3.1 PACEdge Usage Models

PACEdge software has two usage models:

1. **Direct Use Model**: Running on the Industrial PC., which has a directly attached Monitor, keyboard, and mouse.
   Note: this mode is not supported on CPL410 and CPE400 Controllers
2. **Headless Use Model**: Running on the Industrial PC, which is operating headless mode, the user accesses it remotely via Ethernet using a web interface.

Figure 3: Use Case Models



Note: for configuration, administrative tasks, or file transfer user can also use an SSH Client to access PACEdge. The same Linux (Cockpit) login credentials apply.

3.2 PACEdge in a Direct-Use Configuration

3.2.1 Getting Started

1. Connect the monitor to the RXi2-BP or RXi2-LP device using a DisplayPort cable.
   Note: If the monitor of choice has an HDMI or VGA input, use a standard off-the-shelf DP-HDMI or DP-VGA adapter.
   In case of IPC 2010, use special USB-C to DisplayPort cable (can be ordered as an accessory from Emerson)
2. Connect a keyboard and mouse to any of the USB ports.
3. Power up the PACEdge unit and wait until it boots.
4. The boot process will pause and ask for login details. Login as **admin** with password **edgestack**.

Note: The user will be asked to change the default password to a unique password at the first login.

5. Most interactions with PACEdge are done via a browser-based interface. Click on Activities->Show Applications and start the Firefox browser to get started.
6. Within the Firefox browser, go to **https://localhost** or, to use pre-installed PACEdge CA and certificate, go to: **https://hostname.local**
   a. Hostname depends on the HW being used. On RXi2-BP, the hostname is: pacedge-xxx, where xxx is a serial number that can be found on the device label. In case of RXi2-LP, CPL410, CPE400, xxx will be the last six MAC address letters. Easiest way to find them is to first time log in using IP address and then note what the hostname is.
7. Proceed to the section entitled 3.4, *Initial Login.*

3.3 PACEdge in Headless Configuration

In a Remote Headless configuration, the user interfaces with PACEdge via Ethernet using a remote device's (Panel PC, laptop) web browser.

3.3.1 Getting Started

1. Connect the Ethernet cable to the Ethernet port, depending on the hardware, labeled as follows:
   a. RXi2-BP: ETH0
   b. RXi2-LP: LAN (one next to COM port)
   c. IPC 2010: Port 2
   d. CPL410/CPE400: ETH
2. Setup the User PC Ethernet port IP address to be statically assigned as follows:
   a. **IPv4 Static IP: 192.168.3.10**
   (or similar in the same subnet)
   b. Netmask: 255.255.255.0
3. Power up the PACEdge unit and wait until it boots.
   Note: on CPL410/CPE400 wait until the GPOK LED is lit.
4. Open the browser of your choice and type in **192.168.3.100** or, if known ***hostname*.local,** in the address bar.

Note: Access via web browser uses pre-installed certificates and provides extra insurance that you are indeed connecting with the expected PACEdge device. The hostname is **pacedge-xxx**, where xxx is either serial number of the HW (RXi2-BP) or the last six digits of the MAC address. If not known, log in for first time using the static IP address, and take note of what the hostname is.

Note: All Ethernet ports are configured to get IP addresses assigned by the DHCP server. This dynamically assigned IP address can also be used to access PACEdge. On CPL410 and CPE400 first two IP addresses can be read from a built-in display by going into **Edge Settings->Network Config**.

5. Proceed to Section 3.4, Initial Login.

3.4 Initial Login

When using an IP address to access the device for the first time, a warning message will display when the user connects to PACEdge due to the certificate on the unit that is issued for hostname, not IP address. The message will state that the device's identity cannot be confirmed. To proceed, please click on **Advanced** and **Accept.**

Alternatively, the user can use the the ***hostname*.local** to access the device. The hostname will be "pacedge-xxx," where xxx is either the serial number of the device or the last six digits of the MAC address. For more details about certificate infrastructure please refer to Section 4.7, *Certificate Management*.

Note: PACEdge Certificate Authority (CA) certificate (can be downloaded from Emerson Customer Center or from PACEdge Cockpit utility) needs to be added to your browser's trusted certificate store in order for the units certificate to be trusted.

Next, please read and accept the Emerson End User License Agreement (EULA).

Once the EULA has been accepted, the user will be redirected to the Password Management screen. PACEdge software consists of multiple tools and applications, each with user management and password settings. To simplify user and password management tasks, PACEdge comes with pre-configured users: *admin*, *developer*, service, and *operators*, each having its password. Passwords need to be changed in two steps:

1. Use the Password Management page to change passwords for all PACEdge applications.
2. Use the link to Cockpit to change passwords for Cockpit/Linux.
   Once logging into Cockpit for the first time, it will automatically force the user to change the *admin* password. Once changed, please log out and log back into Cockpit as the *admin* user, go to the Accounts page and, as required, set passwords for the remaining users: *developer, service,* and *operators*.
   Note: The **admin** user password for Cockpit/Linux and other PACEdge applications can be different.
   Note: In Cockpit/Linux, per default, other user accounts are disabled. It is a good practice to only set passwords and enable user accounts as indeed required.
   Note: In Cockpit, do not set any passwords for the user *root*. For security purposes, this user is disabled, and setting a password for it would enable it, which is not recommended. Please consult the PACEdge Secure Deployment Guide (GFK-3197) for more details.

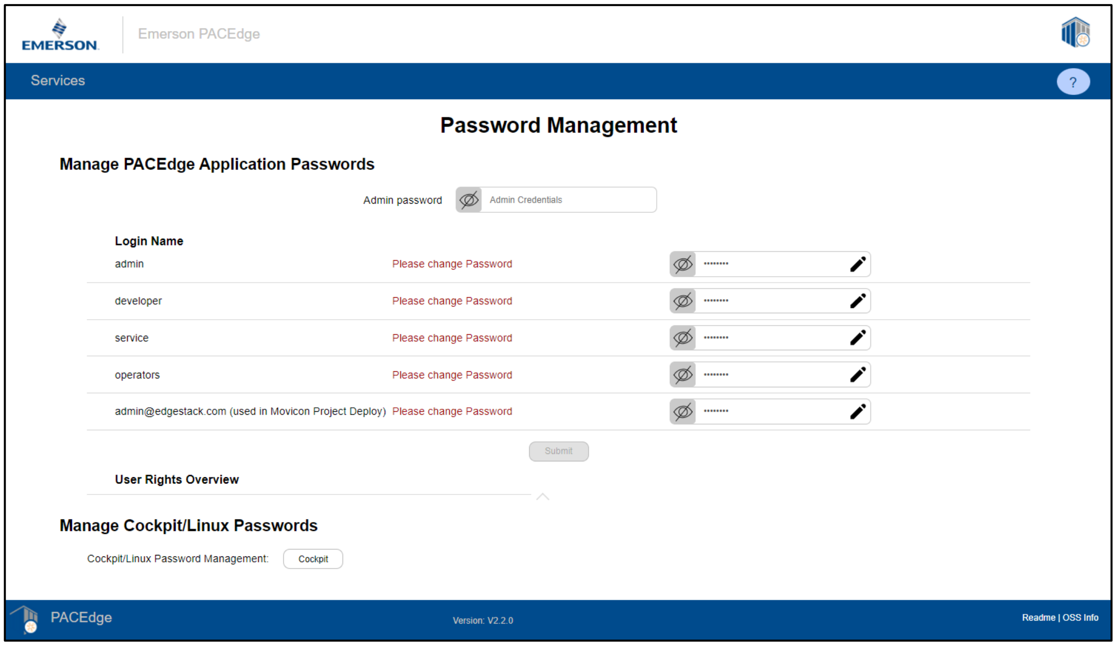
Note: When you click the Apply Changes button and then follow the Restart button, web services used for this connection will be restarted. As a result, depending on the browser used, you might get a message such as "This site can't be reached" or similar. This is normal behavior; manually enter https://ip_address/pw/ or https://*hostname*.local/pw/ in the browser address field to return to the Password Management page. You might need to refresh your browser to see the updated password state. The restart will take up to 60 seconds.

Note: For cybersecurity reasons and proper account setup, the user must change the default passwords at the first login.

Important

Please consult the PACEdge Secure Deployment Guide (GFK-3197) for recommended password changes and other cybersecurity-relevant settings. Detailed password change procedures can be found in 3.5, *PACEdge Users, Rights, and Passwords.*

Figure 4: Password Management Screen (Upper portion to change the password for all PACEdge Applications, a lower portion for Cockpit/Linux password change)



Password Management

Manage PACEdge Application Passwords

Admin password: Admin Credentials

Login Name		
admin	Please change Password	••••••••
developer	Please change Password	••••••••
service	Please change Password	••••••••
operators	Please change Password	••••••••
admin@edgestack.com (used in Movicon Project Deploy)	Please change Password	••••••••

User Rights Overview

Manage Cockpit/Linux Passwords

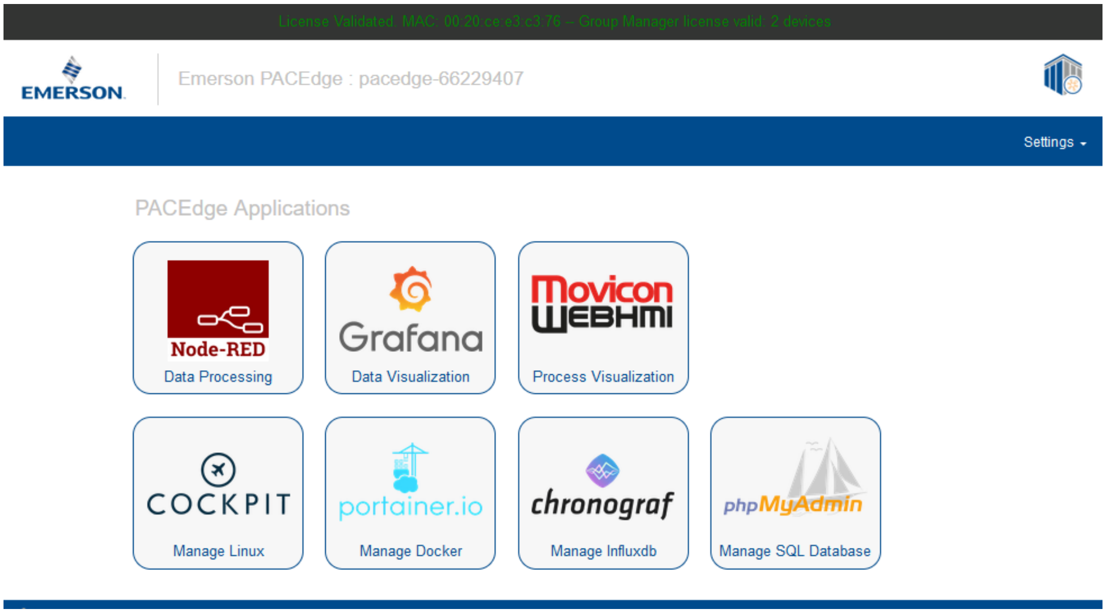
Cockpit/Linux Password Management: Cockpit

Note: A comment in red *Please change Password* will disappear once the password has been changed; however, it may require the user to refresh the browser.

For more detailed information about user roles, passwords, and the password management utility, please refer to Section 3.5, *PACEdge Users, Rights, and Passwords*.

Once the passwords have been changed, reboot the system, then click on the Emerson logo in the upper-left corner. The logo is a shortcut to the PACEdge Landing Page, which has shortcuts to Node-RED, Grafana, Cockpit, Portainer, and other applications. The user can return to the Password Management screen from the PACEdge landing page by clicking **Password Management** in the Settings drop-down list on the right side.

Figure 5: PACEdge Landing Page



License Validated. MAC: 00:20:ce:e3:c3:76 -- Group Manager license valid: 2 devices

Emerson PACEdge : pacedge-66229407

Note: A link to Movicon WebHMI is only active when an actual WebHMI project has been deployed on the PACEdge using Movicon.NExT Editor. When starting a new PACEdge unit, no WebHMI project is deployed; therefore, this shortcut is greyed out.

To explore PACEdge, click on Node-RED and log in with the following:

- **user**: admin
- **password**: (the password that was set on the Password Management page)

Start exploring example flows or create your own flows. Consult the Usage Example section in this document for details.

Important: Please consult the PACEdge Secure Deployment Guide (GFK-3197) for recommended password changes and other cybersecurity-relevant settings.

3.5 PACEdge Users, Rights, and Passwords

In PACEdge user roles and associated passwords are split into two groups:

- Linux/Cockpit users and passwords
- PACEdge Application users and passwords.
  (Password Management Utility, Node-RED, Grafana, Portainer, InfluxDB, MySQL, Chronograf)

Passwords for these two groups are independent of each other and have to be individually changed for each group. It is recommended to have different set of passwords for Linux/Cockpit and for PACEdge Applications, especially for the **admin** role.

The passwords of PACEdge shall be changed at the first login. Emerson strongly recommends using long (10 characters or more), complex passwords wherever passwords are used for authentication. Recommendations on password complexity and management can be found in

NIST Special Publication 800-63-3, Digital Identity Guidelines (https://pages.nist.gov/800-63-3/sp800-63b.html).

In PACEdge, each of the following applications has its users and user credential management interfaces:

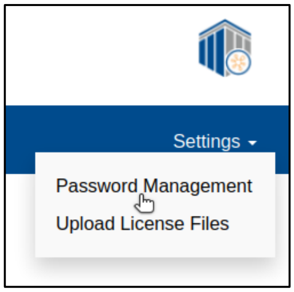
Table 1: Default Passwords

Functionality	Authenticated Subjects (user)	Default Passwords
SSH remote login	admin	Default password: “edgestack”
Cockpit/Linux	admin	Default password: “edgestack”
Chronograf	admin	Default password: “edgestack”
Grafana	admin	Default password: “edgestack”
Node-RED	admin	Default password: “edgestack”
Portainer	admin	Default password: “edgestack”
InfluxDB	admin	Default password: “edgestack”
MySQL	admin	Default password: “edgestack”
TimescaleDB	admin	Default password: “edgestack”
Connext/WebHMI	admin@edgestack.com	Default password: "Edgestack123!"

To secure PACEdge, it comes with pre-configured user roles, each role having different access rights and passwords. An automated Password Management utility helps administrators to configure user role passwords across a whole set of applications. This utility can be accessed from the PACEdge landing page.

Only the **admin** role can access this utility and change passwords for all the user roles.

Figure 6: Password Management

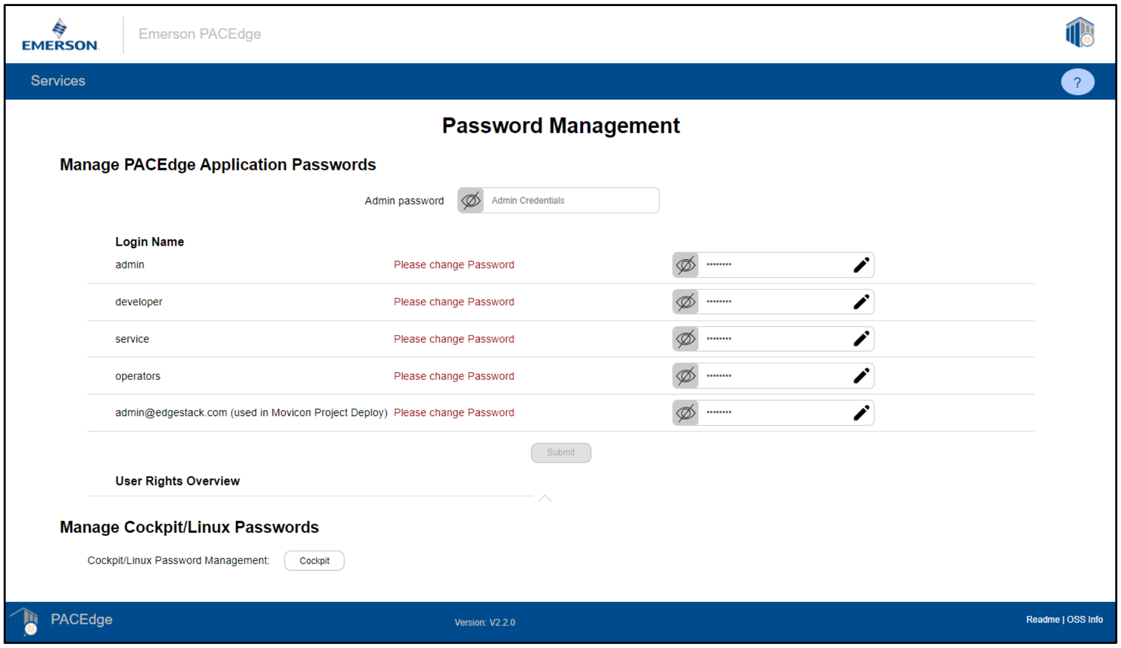


By default, all the login credentials in PACEdge are set to the following:

User Name: **admin**
Password: **edgestack**

Once on the Password Management page (Figure 7), the options to change passwords for each user role automatically across all PACEdge applications are on the top. On the bottom, the user can find the shortcut to change the passwords for the Cockpit/Linux.

Figure 7: Password Management Page



Password Management

Manage PACEdge Application Passwords

Admin password: Admin Credentials

Login Name		
admin	Please change Password	••••••••
developer	Please change Password	••••••••
service	Please change Password	••••••••
operators	Please change Password	••••••••
admin@edgestack.com (used in Movicon Project Deploy)	Please change Password	••••••••

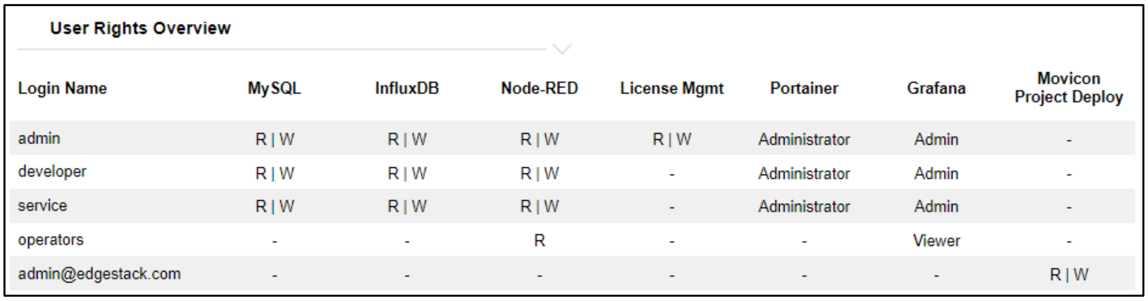
User Rights Overview

Manage Cockpit/Linux Passwords

Cockpit/Linux Password Management: Cockpit

Note the line in the utility called **User Right Overview**. If this line is expanded, a table is shown with all PACEdge applications on top and all users on the side, indicating each user's access rights for each application. In this table, R indicates Read, W indicates Write, and symbol (-) indicates no access rights. For Grafana, instead of R/W, the standard organization roles are used. To get more details about the Grafana roles, please visit https://grafana.com/docs/grafana/latest/permissions/organization_roles/. In Portainer, there is only one user role called Administrator. Notice a separate user **admin@edgestack.com.** This user is only for deploying the Movicon project onto the PACEdge device and can not be used to access any other PACEdge applications.

Figure 8 User Roles and Access Rights



User Rights Overview

Login Name	MySQL	InfluxDB	Node-RED	License Mgmt	Portainer	Grafana	Movicon Project Deploy
admin	R | W	R | W	R | W	R | W	Administrator	Admin	-
developer	R | W	R | W	R | W	-	Administrator	Admin	-
service	R | W	R | W	R | W	-	Administrator	Admin	-
operators	-	-	R	-	-	Viewer	-
admin@edgestack.com	-	-	-	-	-	-	R | W

Initially, the page will have notes urging the user to change passwords. After the password has been changed once, these notes will disappear; however, a browser refresh might be required to see the change reflected.

Important

It is highly recommended to change passwords by using provided automated utility instead of manually going into a specific application (such as Node-Red) and changing the password there. *If passwords between different applications get out of sync password management utility will fail.*

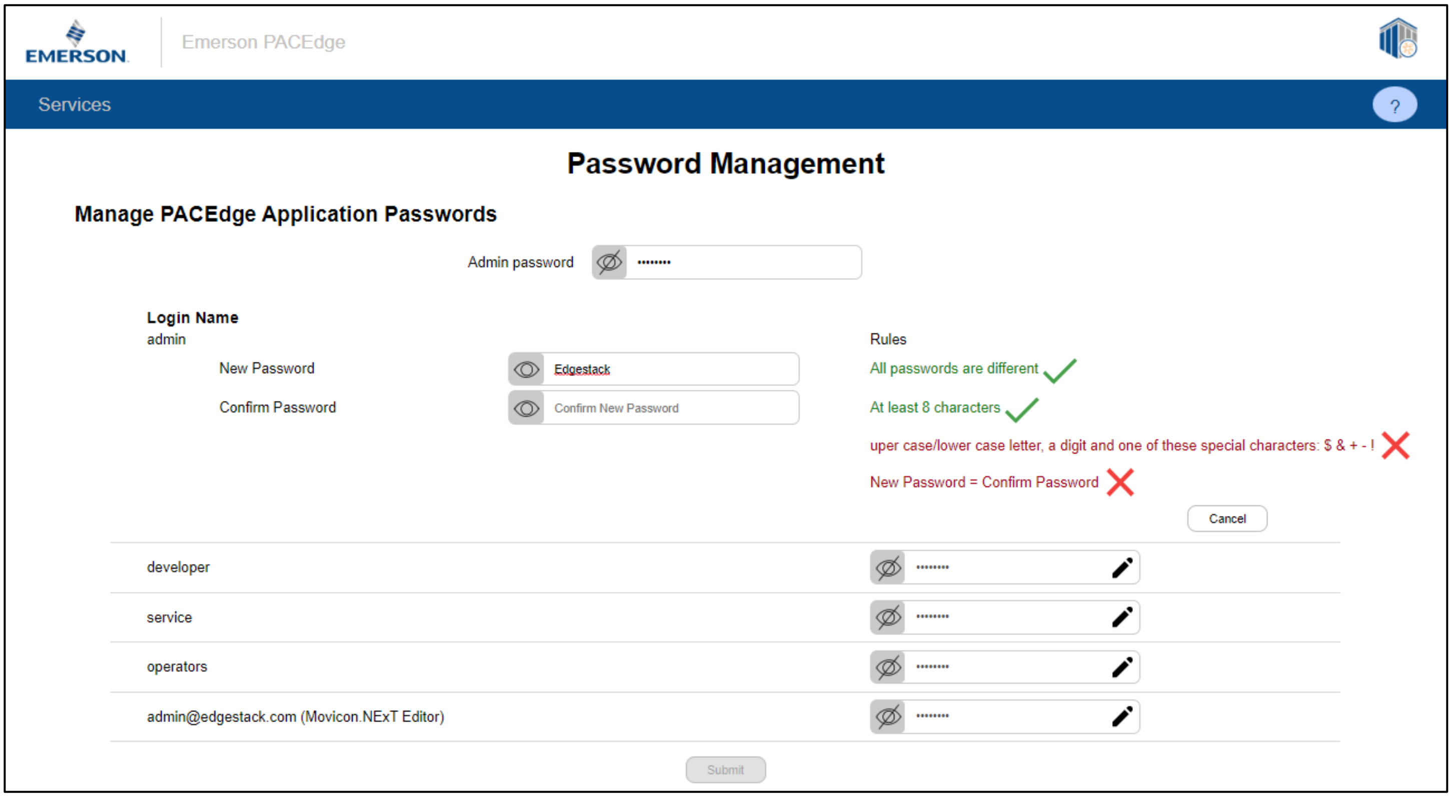
3.5.1 Changing Passwords via Automated Password Management Utility

The Password Management utility will automatically set passwords for all PACEdge applications and each user role. Furthermore, the utility will provide a link to change Cockpit/Linux passwords at the bottom of the page. Please remember to change both PACEdge Application passwords and Cockpit/Linux passwords.

To use the Password Management utility user needs to be logged in as an **admin**.

To change PACEdge Application passwords, please click on the pencil icon to open the entries, fill in all the fields and then click on *Submit* button at the bottom of this table. Notice that the *Submit* button will be greyed out and inactive if some password conditions are not met. These conditions are explicitly listed, initially in red, and dynamically change to green once satisfied, see (Figure 9).

Figure 9: Hints Dialog



Password Management

Manage PACEdge Application Passwords

Admin password ••••••••

Login Name
admin

New Password Edgestack
Confirm Password Confirm New Password

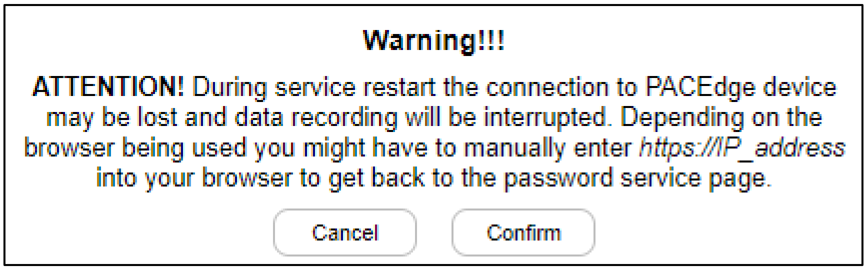
Rules
All passwords are different ✓
At least 8 characters ✓
uper case/lower case letter, a digit and one of these special characters: $ & + - ! ✗
New Password = Confirm Password ✗

developer	••••••••
service	••••••••
operators	••••••••
admin@edgestack.com (Movicon.NExT Editor)	••••••••

Note: The user could choose to change only a subset of all passwords, which might be practical later on, but the first time this utility is run, the user MUST set all passwords as this also sets up roles in databases and other utilities.

Once the *Submit* button is clicked, a dialogue box will warn that PACEdge Applications will be restarted during the password change, including the webserver that serves this page. Depending on the browser used, the user may receive a message indicating that the connection has been lost. In this case, the user should re-enter https://*ip_address* or https://*hostname*.local to return to the PACEdge landing page. The user can return to the Password Management application by going to Settings in the upper-right corner of the page.

Figure 10: Admin Password Change in Progress



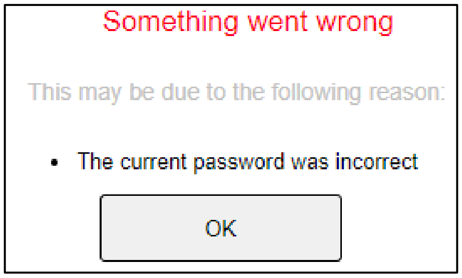
Warning!!!

ATTENTION! During service restart the connection to PACEdge device may be lost and data recording will be interrupted. Depending on the browser being used you might have to manually enter *https://IP_address* into your browser to get back to the password service page.

Once you click the Confirm button, some additional checks will be performed, and passwords will be changed if there is no error. In case of an error, an Error page will be shown (Figure 10). Following are most common reasons for this error:

- Current admin password is incorrect. The Password Management utility uses the admin password from the PACEdge Applications, not for Linux/Cockpit
- The admin role password in one of the PACEdge Applications (Node-RED, Grafana, Portainer, InfluxDB, MySQL) is different than in others. The most likely way for this to happen is when upgrading PACEdge from Version 2.1 to 2.2, where Portainer password was never changed and is still default **edgestack.** If this is the case, log into Portainer directly using **ip_address/portainer** (ex: **192.168.3.100/portainer**) as admin with the old password (**edgestack** if left at default) and change it to match admin password for the rest of the PACEdge applications.

Figure 11: Error When Entering Passwords



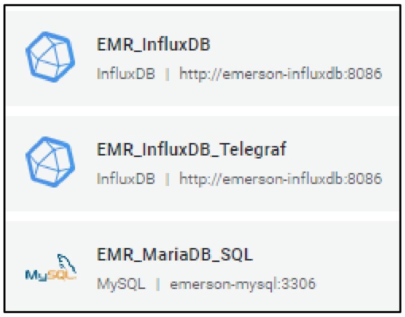
Something went wrong

This may be due to the following reason:

- The current password was incorrect

Note: The Password Management utility will change passwords for each user role in each PACEdge Application. However, in Node-RED flows, nodes accessing InfluxDB and MySQL applications still need to manually set/change the password (s).
Similarly, in Grafana, the Password Management utility will automatically change passwords for the three PACEdge default databases as shown in Figure 12. However, if the user has set up additional custom databases, those passwords need to be updated manually.

Figure 12: PACEdge default databases pre-configured in Grafana



EMR_InfluxDB
InfluxDB | http://emerson-influxdb:8086

EMR_InfluxDB_Telegraf
InfluxDB | http://emerson-influxdb:8086

EMR_MariaDB_SQL
MySQL | emerson-mysql:3306

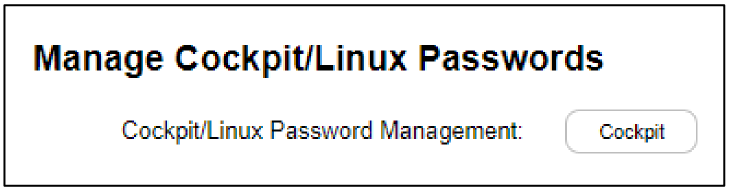
3.5.3 Changing Cockpit/Linux User Passwords

Please note that in addition to using the Password Management utility to change all passwords for the PACEdge Applications, the Cockpit/Linux user passwords must be managed separately.

Note: The automated Password Management utility, described in the previous chapter, does not change Cockpit/Linux user passwords. The default **admin** password is **edgestack.**

The easiest way to manage the Cockpit/Linux user passwords is to click on the Cockpit button, which is a link, on the bottom portion of the Password Management page, Figure 13: Cockpit/Linux Password Management Link

Figure 13: Cockpit/Linux Password Management Link

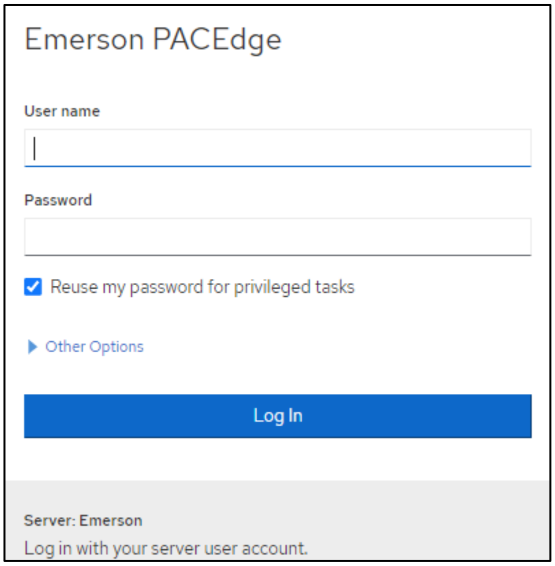


Manage Cockpit/Linux Passwords

Cockpit/Linux Password Management:

Once asked, log in with the **admin** credentials (Figure 14). The user will be prompted to change the default admin password if this is the first login.

Figure 14: Login to Cockpit/Linux



Emerson PACEdge

User name

Password

☑ Reuse my password for privileged tasks

Server: Emerson

Log in with your server user account.

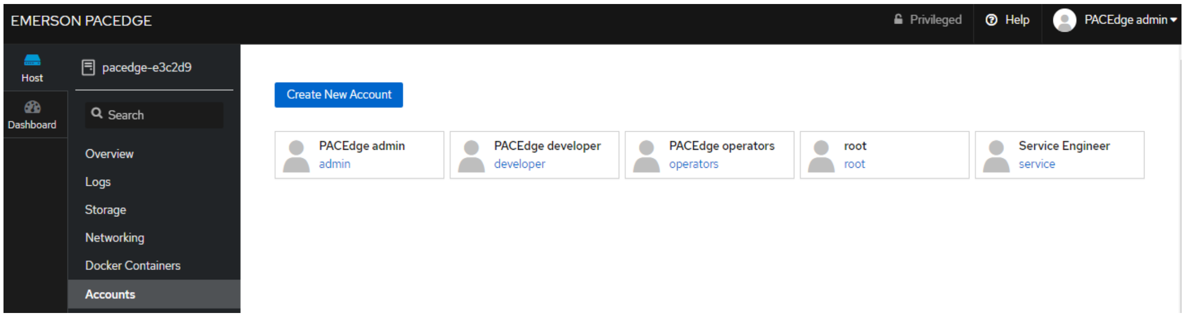
After changing the **admin** password, log out and log in to Cockpit again so that changes can take effect and the admin privileges are granted. The user can do this by accessing a drop-down menu in the screen's upper-right corner.

Once logged in as admin, user may set the passwords for the developer, service, and operators roles, Figure 15. Note that per default user roles other than admin are disabled and will remain so unless the password is set. Therefore, if additional user roles in Linux/Cockpit are not required, it is safer to keep them disabled and not set any password.

⚠ CAUTION

Do not set any password for the **root** user, as this would enable the **root** account and create a cybersecurity vulnerability.

Figure 15: User Profiles



PACEdge admin
admin

PACEdge developer
developer

PACEdge operators
operators

root
root

Service Engineer
service

3.6 PACEdge System Level Settings

PACEdge is Linux-based software but designed to be user-friendly with a graphical user interface (Cockpit) for all basic configuration tasks. To access Cockpit, open the browser, and type in the IP address of PACEdge. This will bring you to the PACEdge landing page, where you can click on the Cockpit icon.

3.6.1 System Configuration Changes via Cockpit

Cockpit supports most system configuration tasks, such as:

3.6.2 Changing Linux/Cockpit password refer to: Section 3.5.2

- Changing Cockpit/Linux User Passwords
- Adding /Removing users accounts
- Changing Hostname refer to: Section Changing Host Name under Section 3.6.1
- Modifying IP addresses, VLANs, enabling Firewall
- Managing Storage
- Browsing and moving files between PACEdge and remote computer
- Monitoring CPU, Memory, Network, Storage usage, and performance
- Analyzing system logs
- Monitoring system services
- Setting up the OPC UA Port Forwarding feature
- Shortcuts to other PACEdge applications
- Linux Terminal (for users that desire further customization)

Note: new version of Cockpit no longer has Docker Container management tab. From now on to manage Docker Containers a Portainer tool shall be used.

Changing Host Name

PACEdge supports mDNS/DNS-SD protocols, allowing devices to be discovered on the network using Hostname. By default, PACEdge systems are delivered with Hostname set to **pacedge-xxxxxxxx,** where xxxxxxxx is either an 8-digit serial number of the hardware device or the last 6-digit MAC address.

Note: starting with PACEdge version v2.3.0 devices are delivered with cyber security certificate issued using devices hostname, hence changing hostname to some other value will invalidate this certificate. Please refer to the section 4.7 Certificate Management for details.

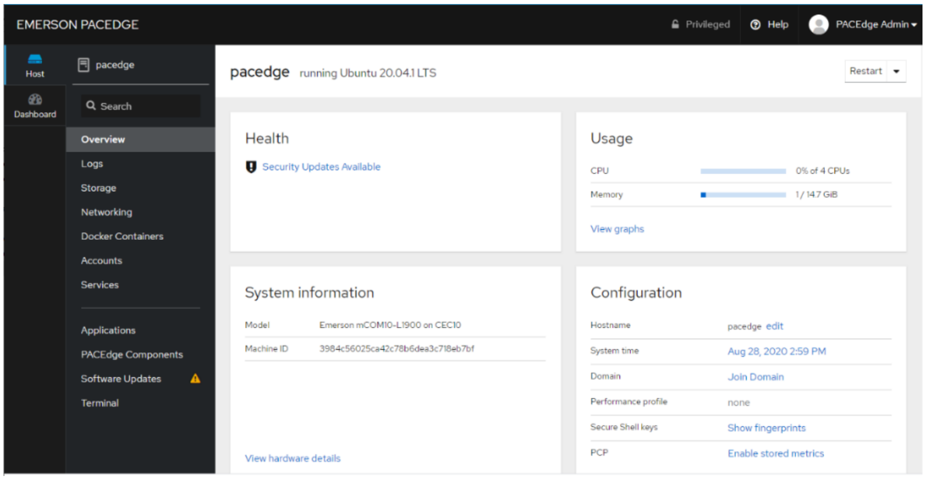
Note: In RXi2-LP, CPL410 and CPE400 serial number of the HW is not known to the Linux; hence last six digits on the Ethernet controller MAC address are used. To change the Hostname:

1. From the PACEdge landing page, open Cockpit and log in as an admin user.
2. In the Overview tab, in the Configuration section, click on the **edit** link that follows the Hostname.

Gracefully Restarting, Shutting Down the System

To gracefully shut down the system, log in to Cockpit and select the desired action in the Overview tab in the upper-right corner. Note that when selecting an action, there is also an option to select time delay.

Figure 16: Cockpit Screen



pacedge running Ubuntu 20.04.1 LTS

Health

Security Updates Available

Usage

CPU	0% of 4 CPUs
Memory	1 / 14.7 GiB

View graphs

System information

Model	Emerson mCOM10-L1900 on CEC10
Machine ID	3984c56025ca42c78b6dea3c718eb7bf

View hardware details

Configuration

Hostname	pacedge edit
System time	Aug 28, 2020 2:59 PM
Domain	Join Domain
Performance profile	none
Secure Shell keys	Show fingerprints
PCP	Enable stored metrics

For further Cockpit details and documentation, please consult online resources.

3.6.4 Physical – Logical Ethernet Port Mapping

The Linux Operating system enumerates Ethernet ports in a way that is not obvious which physical port corresponds to which logic port, as seen in Cockpit. The tables below will provide physical to logical Ethernet port mapping for different HW devices

Table 2 RXi2-BP Physical to Logical Ethernet Port Mapping

Physical Ethernet port (Label on device)	Logical Ethernet Port (seen in Cockpit)
ETH 0	enp1s0f0
ETH 1	enp10s0
ETH 2	enp2s0
ETH 3	enp7s0

Table 3 RXi2-LP Physical to Logical Ethernet Port Mapping

Physical Ethernet port (Label on device)	Logical Ethernet Port (seen in Cockpit)
ETH (upper, next to Serial)	enp2s0
ETH (lower, next to USB)	enp1s0

Table 4 IPC 2010 Physical to Logical Ethernet Port Mapping

Physical Ethernet port (Label on device)	Logical Ethernet Port (seen in Cockpit)
Port 1	eth1
Port 2	eth0

Table 5 CPL410/CPE400 Physical to Logical Ethernet Port Mapping

Physical Ethernet port (Label on device)	Logical Ethernet Port (seen in Cockpit)
ETH	enp1s0

3.7 PACEdge License File

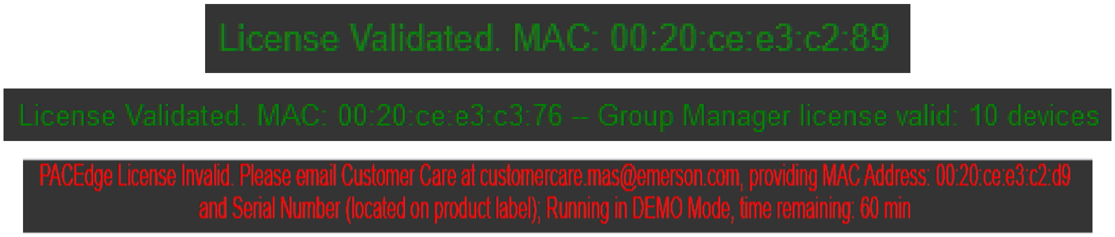
PACEdge is protected with the license file, which is locked to the physical device that PACEdge is running on. PACEdge license controls both what software package is enabled (PACEdge only, with Connext, with WebHMI) and also if the Group Manager feature for certain number of managed devices is enabled. PACEdge devices will come with software and a valid license pre-installed from the factory. However, if performing a restore to the factory default, as described in Section 8.2

PACEdge Software Restore/Recovery on RXi2-BP, RXi2-LP IPCs, on RXi2-BP, IPC 2010 and RXi2-LP a license will have to be manually installed. Note that on CPL410 and CPE400 the Factory Default image is integrated within the unit, hence if restored, the installation process will automatically restore the valid license, and the user does not have to take any action.

To provision the new license, follow the steps below:

1. Once the PACEdge device is booted, and the user accesses it via the landing page (using a browser), the license status will be indicated at the top of the page (one of the messages as shown below, message in green: OK, message in red: Error):

Figure 17: License Validation (green message: OK, red message: Error)



License Validated. MAC: 00:20:ce:e3:c2:89

License Validated. MAC: 00:20:ce:e3:c3:76 – Group Manager license valid: 10 devices

PACEdge License Invalid. Please email Customer Care at customercare.mas@emerson.com, providing MAC Address: 00:20:ce:e3:c2:d9 and Serial Number (located on product label); Running in DEMO Mode, time remaining: 60 min

2. If the message indicates **License Validated,** nothing is to be done; otherwise, copy the MAC Address shown in the message (for instance, 00:20:ce:e3:c2:d9) and get the serial number of the PACEdge device from the label on the unit. Contact Customer Care to obtain the valid license files. Contact details can be found in the *Technical Support* section located at the end of this document) **Note**: you will need two files: 1) license.json and 2) license.sig

3. Once both license files have been received from Customer Care, go into the PACEdge landing page, Settings, Upload License Files option, and upload both files:

Figure 18: Upload License Files Menu

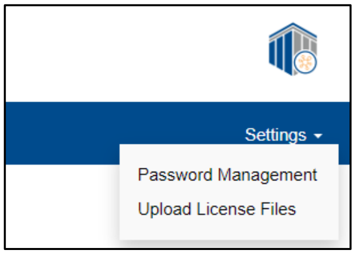
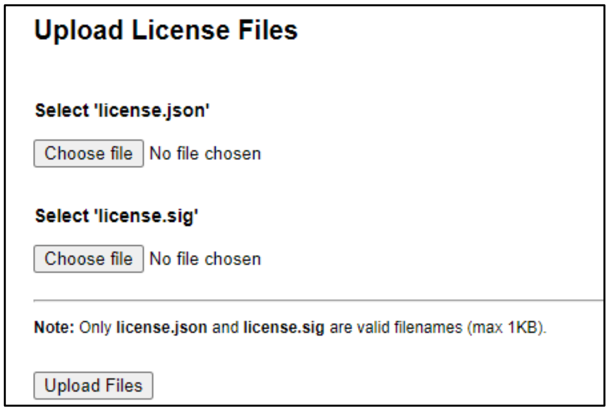


Figure 19: Browse for License Files



Upload License Files

Select 'license.json'

Select 'license.sig'

Note: Only **license.json** and **license.sig** are valid filenames (max 1KB).

Once the files have been uploaded, perform a graceful reboot of the PACEdge device via Cockpit. Information on how to gracefully reboot the system can be found in Section Gracefully Restarting, Shutting Down the System.

4. After reboot, wait a few seconds for all services to fully come up and refresh the browser. If successful, the **License Validated** message in step 1 should be visible.

3.7.1 Backing Up License Files

If desired, license files on the unit can be backed up to a computer that PACEdge is being accessed from and used later when for some reason license is lost. To do so, use the Navigator feature in Cockpit and download and save locally the following two files:

- /home/admin/pacedge/emerson-software/license/license.json
- /home/admin/pacedge/emerson-software/license/license.sig

Note: when using Navigator to download the files, right-click on the file and select the **Download** option. If the error is reported, log out of Cockpit and, log back in as an **admin** user, and repeat the download steps.

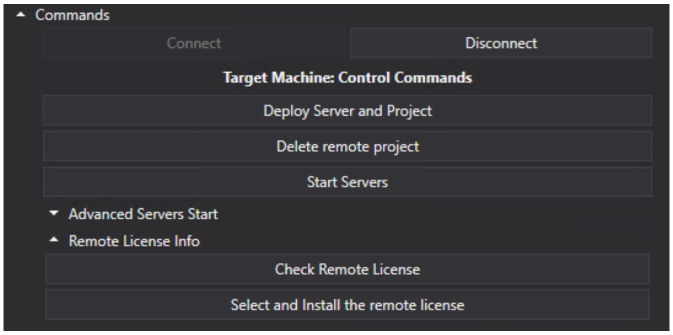
3.7.2 Licensed Connext and WebHMI Features

PACEdge license file enables certain Movicon features, such as Connext or WebHMI. To verify what Movicon features are enabled, please connect to the unit running PACEdge from your workstation where Movicon.NExT software is installed and do the following:

1. Follow the steps described in: Movicon.NExT to Connext/WebHMI on IPC to establish a connection to PACEdge using the Deploy Project window

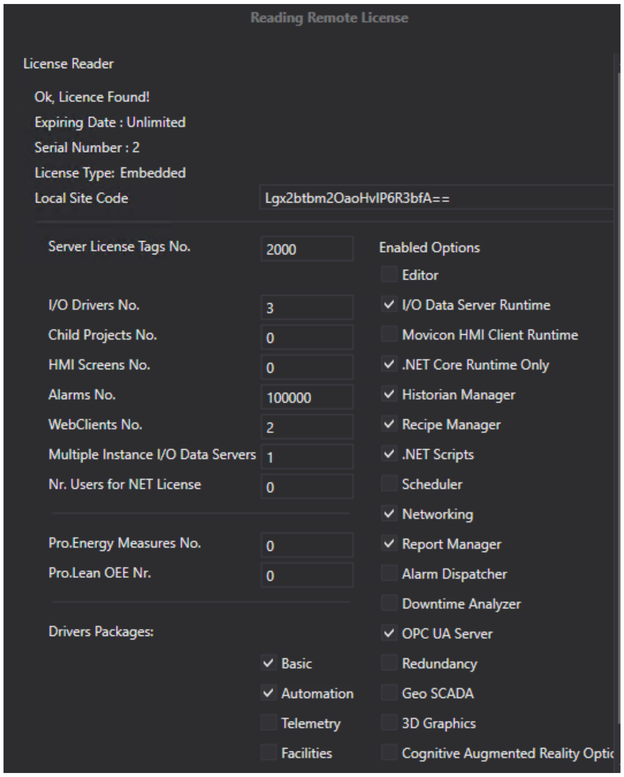
Go to the Remote License Info section and click on Check Remote License:

Figure 20: Connext and WebHMI License Check Dialogue



2. The picture below shows the enabled features for the Connext SKU:

Figure 21: License Enabled Features in Connext SKU



Reading Remote License

License Reader

Ok, Licence Found!
Expiring Date : Unlimited
Serial Number : 2
License Type: Embedded
Local Site Code Lgx2btbm2OaoHvIP6R3bfA==

Server License Tags No. 2000
I/O Drivers No. 3
Child Projects No. 0
HMI Screens No. 0
Alarms No. 100000
WebClients No. 2
Multiple Instance I/O Data Servers 1
Nr. Users for NET License 0

Pro.Energy Measures No. 0
Pro.Lean OEE Nr. 0

Drivers Packages:
☑ Basic
☑ Automation
☐ Telemetry
☐ Facilities

Enabled Options
☐ Editor
☑ I/O Data Server Runtime
☐ Movicon HMI Client Runtime
☑ .NET Core Runtime Only
☑ Historian Manager
☑ Recipe Manager
☑ .NET Scripts
☐ Scheduler
☑ Networking
☑ Report Manager
☐ Alarm Dispatcher
☐ Downtime Analyzer
☑ OPC UA Server
☐ Redundancy
☐ Geo SCADA
☐ 3D Graphics
☐ Cognitive Augmented Reality Optic

3.7.3 Licensed Group Manager Features

Group Manager feature can be licensed for a specific number of PACEdge devices that can be managed. How many devices can be managed is indicated at the top of the PACEdge Landing page.

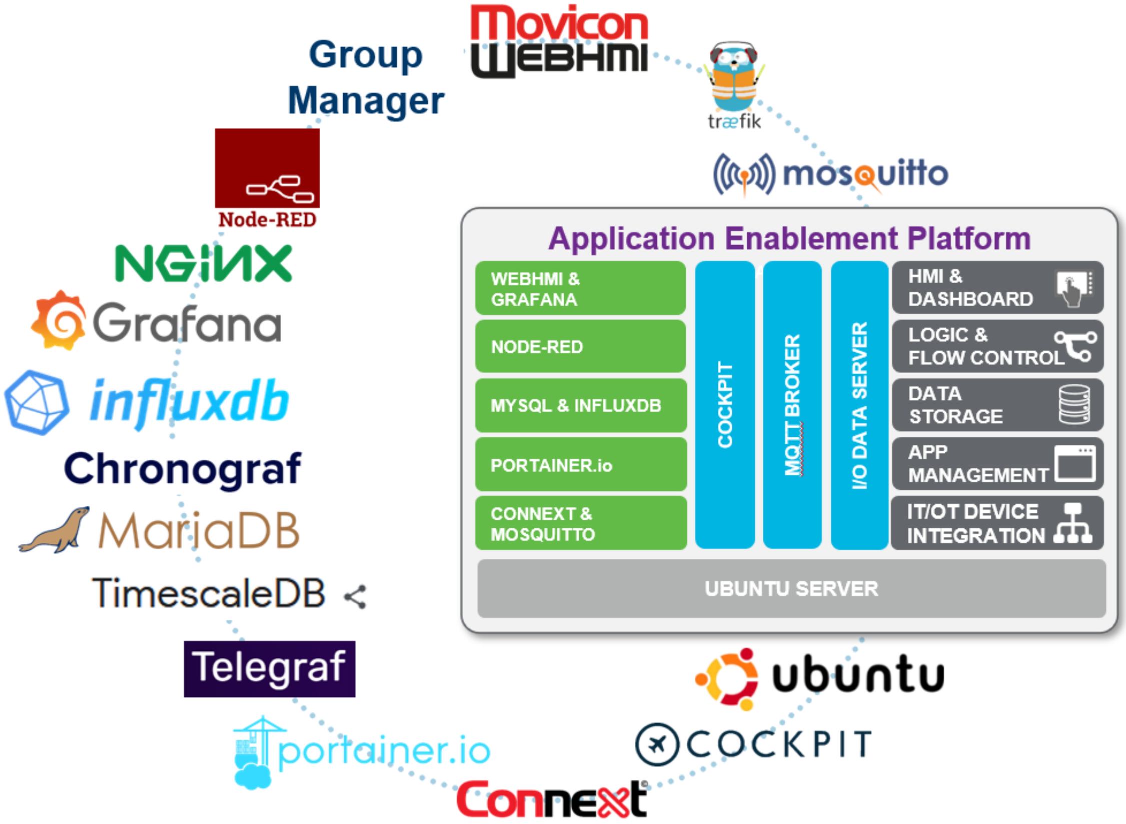
For more information, please refer to section: 4.6.6 Group Manager Licensing

Section 4: PACEdge Architecture Details

4.1 PACEdge Applications

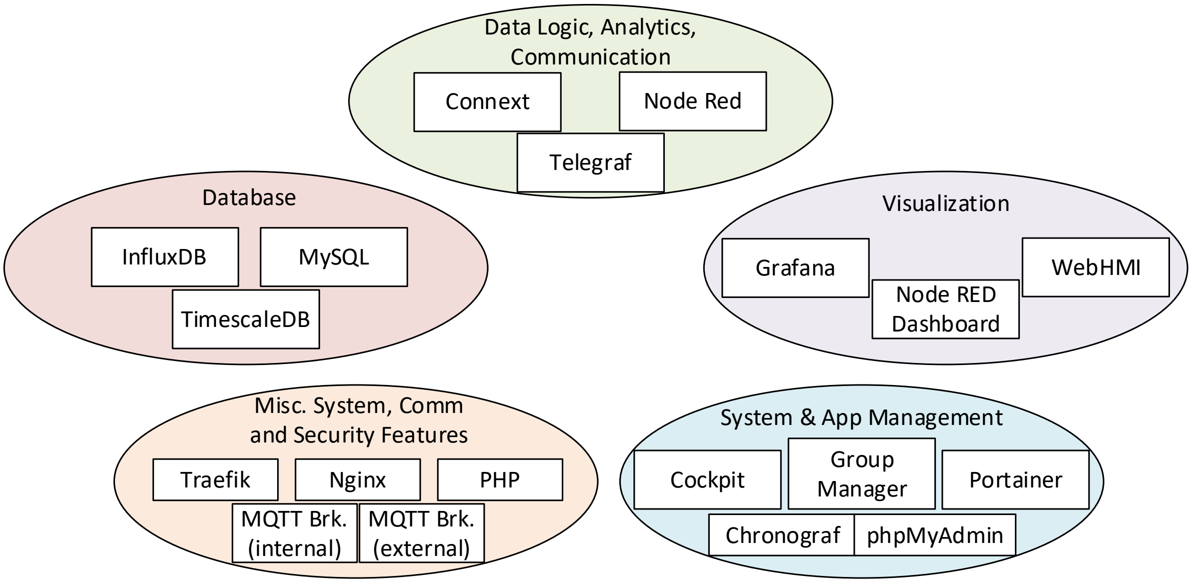
PACEdge entails several communications, data processing, data storage, and visualization applications that run on the Linux operating system. The following diagram gives an overview of the components that make up PACEdge:

Figure 22: Software Overview



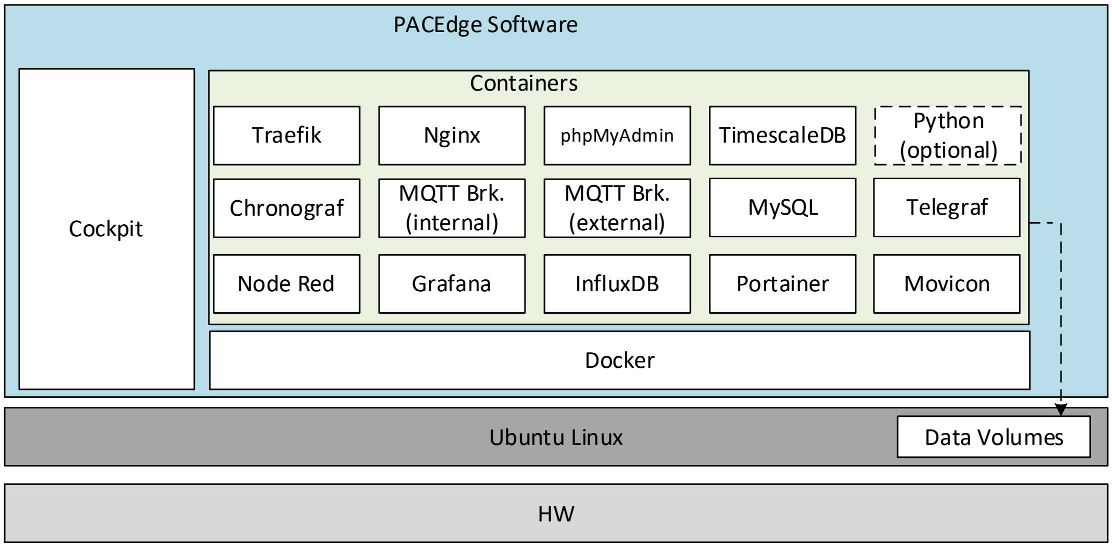
PACEdge tools and applications, based on the function that they perform, can be grouped into functional categories as follows:

Figure 23: PACEdge Application Categories



PACEdge was designed using Docker architecture, in which each application runs in its own Docker container (Figure 24). Since containers are designed to be easily replaceable, they do not retain an internal state between reboots unless specifically designed. With PACEdge, selected containers will map some of their data to data volumes on the host Linux system so that Node-RED, Grafana, and database changes can be saved between Container restarts and updates.

Figure 24: Containerized PACEdge Implementation



4.1.1 Application Development Components

Node-RED

Node-RED version: V3.0.2

Node-RED is the logic engine of the PACEdge. It provides a graphical way to wire together different APIs and services, enabling event-driven logic implementations. Node-RED is well known for its broad adoption in the software community and has many freely available nodes that can be easily installed. PACEdge comes with a large selection of pre-installed nodes, allowing users to easily send and receive data via MQTT, OPC-UA, ModubusTCP, and ModbusRTU interfaces. It also allows users to process, visualize data via the dashboard, store data in InfluxDB, MySQL and TimescaleDB databases, and send email alerts. Node-Red also has nodes for cloud connectivity.

Node-RED is implemented in a Docker Container and can be accessed via browser at

hostname.local/nodered/ or *p_address_of_PACEdge/nodered/*

For more details about Node-RED, please refer to examples later in this document, as well as to *www.nodered.org*

Connext and WebHMI

Supported Movicon.NExT version: V4.2

Connext is a data gateway, supporting a large number of field busses and proprietary communication protocols, allowing to receive data, internally share data with other applications, historize data and make it available to other software services via OPC UA Server

WebHMI, in addition to data gateway functionality, adds HMI visualization capability.

Grafana

Grafana version: v9.5.2

Grafana is a visualization tool that lets users view and analyze data and create alerts. Even though the Node-RED Dashboard already has its own data visualization, Grafana brings extra features and ways to scroll and zoom into specific portions of the graph. Grafana works with several databases, but in PACEdge, it is suggested to use TimescaleDB, InfluxDB and MySQL.

Grafana is implemented in a Docker Container and can be accessed via browser at *hostname.local/grafana/* or *p_address_of_PACEdge/grafana/*

For more details about Grafana, please refer to examples later on in this document, as well as to *www.grafana.com*

MQTT

MQTT version: V2.0.15

PACEdge implements two MQTT brokers: one is meant to be used for PACEdge internal communication between Docker containers, and the second one for external communication. Node-RED does have MQTT nodes pre-installed. MQTT brokers are implemented using Mosquitto open-source software.

For more details about MQTT, please refer to www.mqtt.org

Traefik

Traefik version: V2.10.0

Since most of the PACEdge applications are installed in their own Docker Containers and communicate with each other via network interfaces, Traefik is used as a front-end edge router to receive requests from the external world and route them to an appropriate service. Traefik enables accessing services such as Node-RED by simply adding /nodered at the end of the PACEdge IP address instead of the port number. In addition to simplifying application access and navigation Traefik also improves cybersecurity by reducing the number of open ports.

For more details about Traefik, please refer to docs.traefik.io/

Nginx and PHP

Nginx version: V2.3.0.4

NGNIX and PHP are services used to serve an initial landing page, which has important version information, end-user license agreement, password management, and license file management services, as well as shortcuts to services like Cockpit, Node-RED, Grafana, Portainer, Chronograf, WebHMI.

Telegraf

Telegraf version: V1.26.3

PACEdge implements Telegraf agent, which is plugin-based and can pull data, statistics, and logs from a variety of databases, underlying system resources (CPU, memory, disc, kernel, software logs, docker containers, etc.), and external devices, heavily focusing on IoT protocols, such as MQTT, AMQP, Cloud resources, etc. For the complete list of available plug-ins, please refer to: https://www.influxdata.com/products/integrations/?_integrations_dropdown=telegraf-plugins

By default, Telegraf is configured to gather system health statistics, listen to a specific MQTT topic on the internal MQTT Broker, and store resulting data in the InfluxDB database.

4.1.2 Database Components

MariaDB (MySQL)

MySQL version: MariaDB v10.5.16

MariaDB (MySQL) is a relational database. Node-RED has a pre-installed node to access this database. I/O Data Server in Connext can use MariaDB as a database. Grafana can pull, analyze, and help visualize data from MariaDB. MariaDB is implemented in a Docker Container.

For more information about MariaDB (MySQL), please refer to mariadb.org/

InfluxDB

InfluxDB version: v1.8.10

InfluxDB is a time-series database. Node-RED has nodes that enable the user to store and query data to and from InfluxDB, and Grafana connects to InfluxDB and retrieves data for visualization. Telegraf uses InfluxdDB to historize data it receives via input connectors

InfluxDB is implemented in a Docker Container and is expected to be managed by either Node-RED or Chronograf applications.

For more details about InfluxDB, please refer to *www.influxdata.com*

TimescaleDB

TimescaleDB version: v2.11.1

TimescaleDB is a type of PostgreSQL database that is specifically designed for managing time series data and has a user-friendly SQL interface.

For more information about TimescaleDB, please refer to *https://www.timescale.com/*

4.1.3 Administration Components

Cockpit Description

Cockpit version: V264

PACEdge is designed to offer the user a GUI experience. Even though it is based on a Linux operating system, all main system management tasks can be done via GUI, and a Cockpit is a tool that makes it happen. Cockpit provides system status and health information, resource (CPU, memory, storage, network) usage, network (IP address) management options, user management options, and different logs for troubleshooting.

Since Cockpit is meant to manage Linux operating system tasks, it runs on Linux as a native application and not in a Docker Container.

Cockpit can be accessed via browser at *hostname.local:9090/cockpites/* or *ip_address_of_PACEdge:9090/cockpites/*

For more details about Cockpit, please refer to *www.cockpit-project.org*

Portainer

Portainer version: V2.11.1

PACEdge is heavily utilizing Docker's container-based implementation, allowing users to add their own containers. Even though Cockpit already has Docker Container management features, a dedicated docker management tool Portainer adds additional functions and visualization options. Portainer allows users to monitor, start, and stop containers, check the container log file, configure restart policies, open ports, etc.

Portainer can be accessed via browser at *hostname.local/portainer/* or *p_address_of_PACEdge/portainer/*

For more details about Portainer, please refer to www.portainer.io/

Chronograf

Chronograf version: V1.10.1

Chronograf is a management interface for InfluxDB. With Chronograf, users can search and query data that is stored in InfluxDB, as well as perform database management tasks. Chronograf as well offers data visualization capabilities, similar to Grafana.

Chronograf is implemented in a Docker Container and can be accessed via a browser at *hostname.local/chronograf/* or *p_address_of_PACEdge/chronograf/*

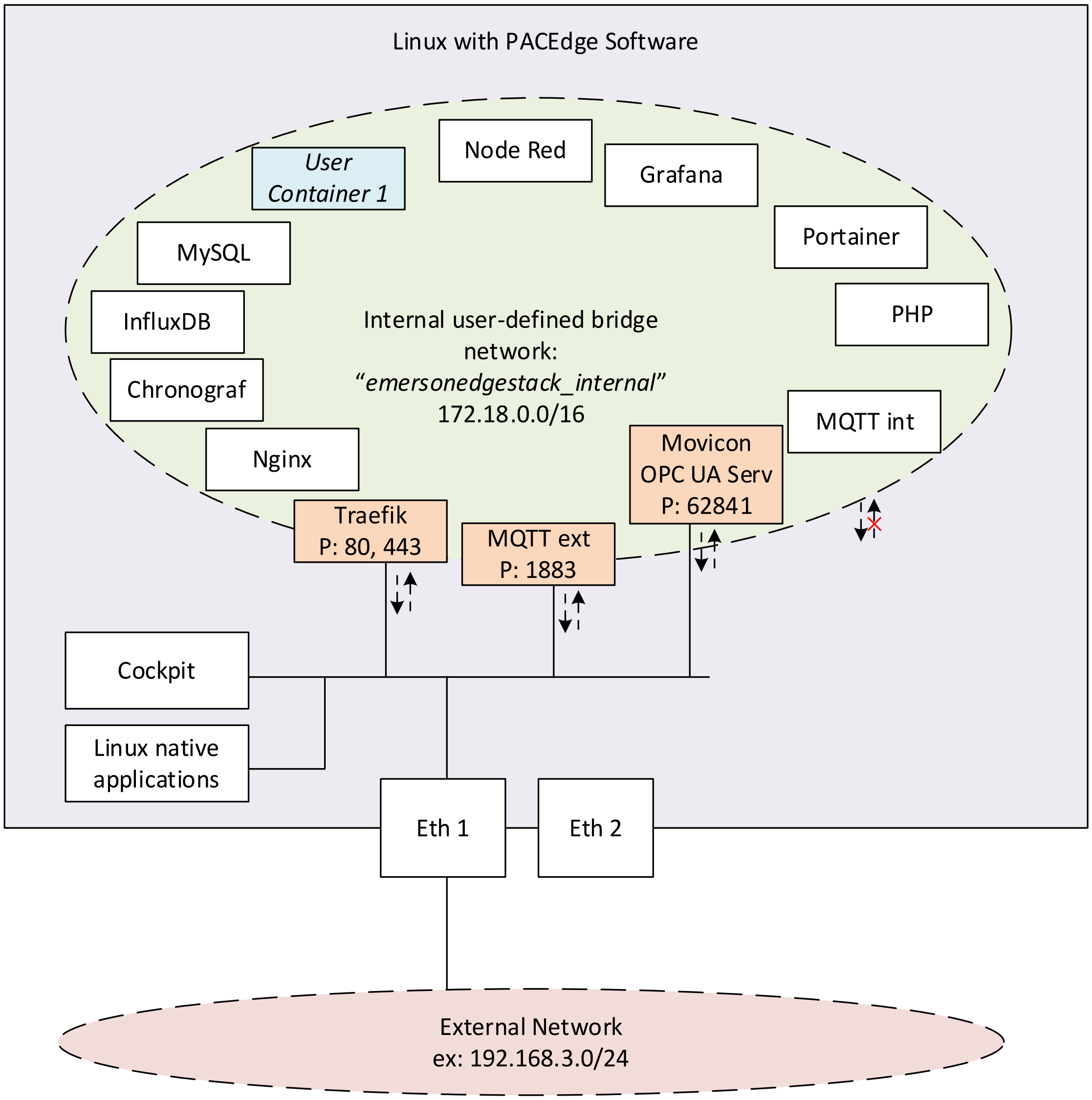
For more details about Chronograf, please refer to *www.influxdata.com/time-series-platform/chronograf/*

4.2 PACEdge Communications and Data Flow

Considering that most of the applications in PACEdge are implemented using Docker Containers, the main communication between them is implemented using network interfaces. For security and traffic segregation reasons, PACEdge uses an internal user-defined bridge network with IP subdomain address 172.18.0.0/16. Each Docker container is connected to this network and can access external services, such as the Internet, but are not directly accessible from outside. Some services, such as Traefik, external MQTT broker, and OPC UA server, have open ports (80, 443, 1883, 62841), enabling these services to be reached from the outside.

External Ethernet ports and IP addresses can be easily managed via Cockpit or other standard Linux tools.

Figure 25: PACEdge Network Communication Paths


Linux with PACEdge Software
User Container 1
Node Red
Grafana
MySQL
Portainer
InfluxDB
Internal user-defined bridge network: "emersonedgestack_internal" 172.18.0.0/16
PHP
Chronograf
MQTT int
Nginx
Movicon OPC UA Serv P: 62841
Traefik P: 80, 443
MQTT ext P: 1883
Cockpit
Linux native applications
Eth 1
Eth 2
External Network ex: 192.168.3.0/24

PACEdge Internal Communication Parameters

To access InfluxDB and MySQL from within the PACEdge, say from Node-RED flows or Connext, as well as for MQTT communication, both internal and external, certain communication parameters and authorization data need to be specified. Note that user passwords are the ones that have been set up at the beginning of PACEdge use.

InfluxDB Access Parameters

- Host: emerson-influxdb
- Port: 8086 (internal access only)
- Usernames: admin, developer, service
- Passwords: as changed by the user

Grafana communication parameters

Grafana is configured with the following users:

- admin: Full rights, can set up users and their passwords, add databases
- developer, service: can create a dashboard, but cannot set up databased
- operators: read-only access

To configure databases inside Grafana, log in as an admin user.

To set up the InfluxDB database:

- URL: http://emerson-influxdb:8086
- Database: data (or your own database name, need to match the InfluxDB database name in Node-RED)
- User: admin or developer
- Password: as changed by the user
- HTTP Method: GET

To set up MySQL database:

- Host: emerson-mysql:3306
- Database: data (or your own database name, need to match MySQL database name in Node-RED or Connext)
- User: admin or developer
- Password: as changed by the user

To set up TimescaleDB database:

- Choose PostgreSQL database type
- Host: emerson-timescale:5432
- Database: data (or your own database name, need to match Timescale database name in Node-RED)

- User: admin or developer
- Password: as changed by the user
- TLS/SSL Mode: disable

MySQL communication parameters

- Host: emerson-mysql
- Port: 3306 (internal access only)
- Username: admin, developer, service
- Password: as changed by the user

TimescaleDB communication parameters

- Host: emerson-timescale
- Port: 5432 (internal access only)
- Username: admin, developer, service
- Password: as changed by the user

Internal MQTT communication parameters

- Server: emerson-mqtt-internal-ipc
- Port: 1883
  PACEdge External Communication Parameters

PACEdge Landing page

To access the PACEdge landing page via browser:

- https://hostname.local or https://ip_address_of_PACEdge

Node-RED

- To access Node-RED via browser:
- https:// hostname.local/nodered/ or https:// ip_address_of_PACEdge /nodered/
- shortcut on the landing page
- Username: admin, developer, service, operators
- Password: as changed by the user

Grafana

To access Grafana via browser:

- https://hostname.local/grafana or https://ip_address_of_PACEdge /grafana/
- shortcut on the landing page
- Username: admin, developer, service, operators
- Password: as changed by the user

Cockpit

To access Cockpit via browser:

- https://hostname.local/cockpites or https:// ip_address_of_PACEdge:9090/cockpites/
- shortcut on the landing page
- Username: admin, developer, service
- Password: as changed by the user

Portainer

To access Portainer via browser:

- https://hostname.local/portainer or https:// ip_address_of_PACEdge /portainer/
- shortcut on the landing page
- Username: admin, developer, service
- Password: as changed by the user

WebHMI

To access WebHMI via browser:

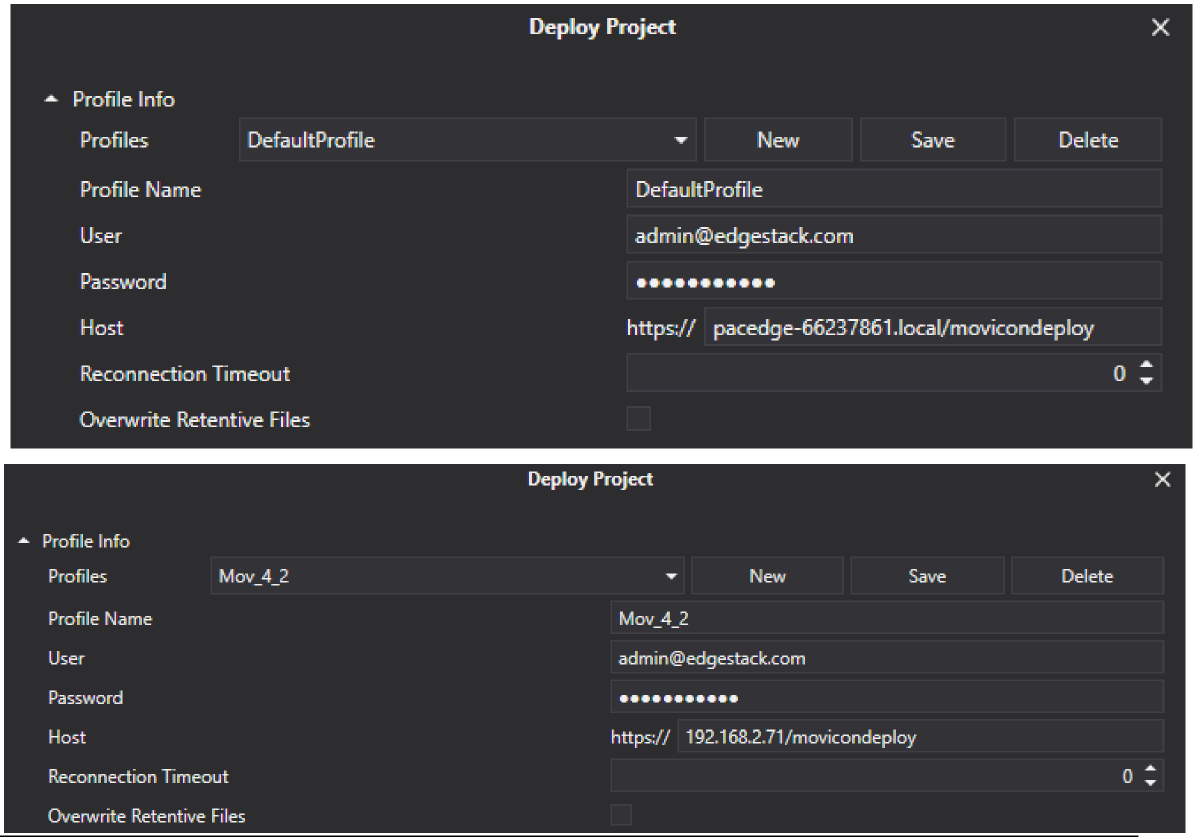
- https://hostname.local/webhmi/ or https:// ip_address_of_PACEdge/movicon/
- shortcut on the landing page

Movicon.NExT to Connext/WebHMI on IPC

Movicon.NExT software, running typically on Windows computers, is used to develop Connext and WebHMI applications. Once development is done, these applications must be uploaded to an industrial PC running PACEdge. To establish this upload, the following credentials need to be specified in Movicon.NExT:

- User: admin@edgestack.com
- Password: as changed by the user
- Host: xx.xx.xx.xx/movicondeploy
  (where xx.xx.xx.xx is either the *hostname*.local or an IP address of the remote industrial PC, such as 192.168.3.100)

Figure 26: List of credentials to deploy the Movicon project



Profiles: DefaultProfile
Profile Name: DefaultProfile
User: admin@edgestack.com
Password: ●●●●●●●●●●●
Host: https:// pacedge-66237861.local/movicondeploy
Reconnection Timeout: 0
Overwrite Retentive Files: □

Profiles: Mov_4_2
Profile Name: Mov_4_2
User: admin@edgestack.com
Password: ●●●●●●●●●●●
Host: https:// 192.168.2.71/movicondeploy
Reconnection Timeout: 0
Overwrite Retentive Files: □

Note: User name and password must match what is specified in /home/admin/pacedge/emerson-movicon/appsettings.json file. Starting with PACEdge release v2.2, please use the Password Management utility to change this password and do not modify this file directly.

External MQTT Communication

PACEdge has a second MQTT Broker, which is designated to be used for MQTT communication with external devices.

- When accessing (external) MQTT broker in the same PACEdge unit:
  - Server: emerson-mqtt
  - Port: 1883
- When accessing the MQTT broker in the other PACEdge unit, attached via Ethernet:
  - Server: xx.xx.xx.xx:1883
    (where xx.xx.xx.xx is an IP address of the remote unit, such as 192.168.2.90)
  - Port: 1883

By default, MQTT security (username, password, certificates, encryption) is disabled, but users can choose to enable it via MQTT Broker configuration settings.

For further details, please refer to: mosquitto.org

4.2.1 Accessing Connext OPC UA Server

Connext has an integrated OPC UA Server, which can be used by internal PACEdge applications, such as Node-RED, but also accessed externally by using OPC UA Clients and Browsers.

To access the Connext OPC UA server from outside the PACEdge device, specify either the hostname.local or the IP address of the PACEdge device and append the following port number **62841**. The example should look like this:

opc.tcp://pacedge-e3c228.local:**62841** or opc.tcp://192.168.3.100:**62841**

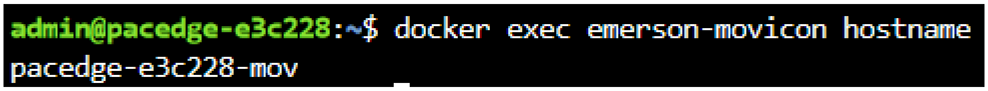
4.2.2 Accessing Connext OPC UA Server from Movicon.NExT Browser

When accessing the Connext OPC UA server on the PACEdge device from Movicon.NExT browser note that Movicon.NExT browser uses the Hostname instead of an IP address. The Movicon.NExT browser will check for a cybersecurity certificate on the target device, which is issued using the Hostname. Therefore, to pass the security check, the Hostname of the Connext container (not the hostname of the Linux OS), not the IP address, has to be used.

To find out the hostname of the Connext docker container, please follow the steps below:

1. Open Cockpit, go to the Terminal tab
2. Type command: **docker exec emerson-movicon hostname** and hit enter
3. Copy hostname shown

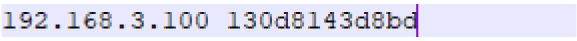
Figure 27: Finding the Hostname with Terminal



```
admin@pacedge-e3c228:~$ docker exec emerson-movicon hostname
pacedge-e3c228-mov
```

Note: Starting with PACEdge v2.2.0, main Linux and Movicon container hostnames have been changed to be unique**.** As a result, the Movicon hostname is no longer derived from the container ID and cannot be read using the Cockpit Docker tab.

4. If Movicon.NExT Editor runs on a Windows computer; the Hostname must be associated with the PACEdge device IP address. This association is established in the host's file:
   C:\Windows\System32\drivers\etc\hosts
   This file is protected and cannot be edited directly, therefore use the following procedure:
   a. Copy the host file to the Desktop folder.
   b. Open the host file with Notepad or another editor.
   c. Add a line with the PACEdge IP address, space, and Hostname of the container, such as:



```
192.168.3.100 130d8143d8bd
```

   d. Save the file and copy it back to the original location, overwriting the original file.
   e. Restart Windows.

4.2.3 Accessing OPC UA Server via Port Forwarding

Starting with version v2.2 PACEdge implements the OPC UA Port Forwarding feature. The OPC UA Port Forwarding feature enables temporary forwarding of OPC UA traffic from one Ethernet port to the other on the PACEdge device. This feature is very handy when using Movicon.NExT editor to browse OPC UA variables on the PLC, which is connected to PACEdge but is not directly accessible from the user PC where Movicon.NExT editor is running. This is the case on CPL410/CPE400 device, where the PLC's OPC UA server is accessible via an internal virtual Ethernet port (VNIC) or in RXi2-BP/RXi2-LP systems where one Ethernet port is used to connect to a PLC and another Ethernet port is used to connect to an engineering PC where Movicon.NExT is running.

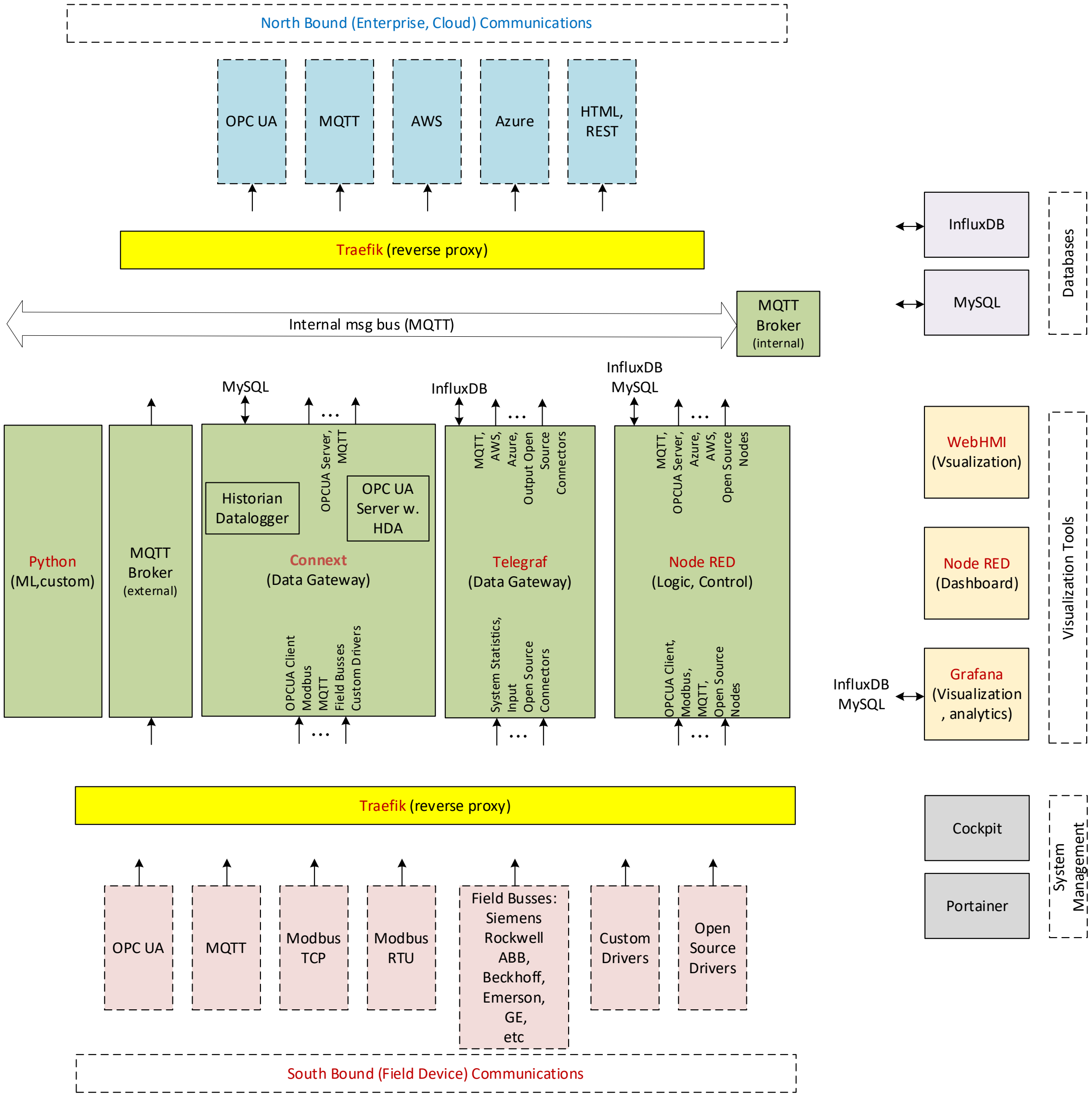
To use the OPC UA Port Forwarding feature, please follow the steps documented in the example: Section 5.6.2 - Example Walk Through, Enable OPC-UA Port Forwarding Feature.

Note: Example is based on CPL410 device, using internal VNIC. If using a different device and Ethernet port, please substitute IP Address and Port Number as required.

4.2.4 PACEdge Suggested Data Flows

In Edge applications, it is common to define data flow as “Southbound” (from field devices, sensors, and PLCs to the gateway) and “Northbound” (from the gateway to SCADA, Enterprise applications, and cloud applications). PACEdge implementation is ideally suited for such data flow scenarios. The following diagram shows different components and their communication capabilities:

Figure 28: PACEdge Communication Capabilities



4.2.5 Southbound Communication Capabilities

To communicate with different field devices, aka Southbound communication, PACEdge has the following capabilities:

Connext I/O Drivers

Connext data gateway component implements the so-called I/O Driver infrastructure, which is designed to communicate with many different field devices, supporting open as well as proprietary communication protocols, such as MQTT, OPC UA Client, Modbus, IEC 60870-5-104, IEC 61850 MMS, Siemens, Beckhoff, GE, Hilscher, Mitsubishi, Omron, Phoenix and many more. For the complete list, please refer to: https://www.progea.com/i-o-driver-list-movicon-next/?lang=en. Furthermore, I/O Driver infrastructure was designed to add new and custom drivers supporting efficient customization efficiently. Data that the driver is receiving is internally stored in the form of a data tag, which can be historized in MySQL, sent out via another driver (such as MQTT), or made available via the OPC UA Server.

Node-RED Nodes

Node-RED has a large open-source community developing and constantly adding new communication nodes. Node-RED supports OPC UA clients and servers, MQTT, Modbus TCP, Modbus RTU as well as a large number of proprietary protocols from Siemens, Rockwell, Beckhoff and others. Once data is received by Node-RED, it can easily be processed, stored in MySQL or InfluxDB databases, and sent out via another interface

Telegraf

PACEdge implements Telegraf agent, which is plugin based utility and can pull data, statistics, and logs from a variety of databases, underlying system resources (CPU, memory, disc, kernel, software logs, docker containers, etc.), and external devices, heavily focusing on IoT protocols, such as MQTT, AMQP, Cloud resources, etc. For the complete list of available plug-ins, please refer to: https://www.influxdata.com/products/integrations/?_integrations_dropdown=telegraf-plugins

For more details on how Telegraf is configured, please refer to an example later in this document: Section 5.7 - Configuring Telegraf.

Custom Driver

Given the open nature of PACEdge, users can also add their existing communication drivers, for example, written in Python, that run in the separate customer-specific Python docker container.

4.2.6 North Bound Communication Capabilities

To communicate with upper software layers, such as SCADA, enterprise systems, and Cloud, aka Northbound communication, PACEdge has the following capabilities:

OPC UA Server

PACEdge has different options to set up an OPC UA Server and make data available to other applications and systems.

Connext comes with an integrated high-performance OPC UA Server and is a recommended option. All data tags within the Connext environment are automatically published via OPC UA Server, which can be accessed at the address: opc.tcp://xx.xx.xx.xx:62841, where xx.xx.xx.xx is an IP address of the IPC.

Alternatively, Node-RED comes with pre-installed OPC UA Nodes, which include OPC UA servers.

MQTT

PACEdge comes with an MQTT Broker, which can be accessed outside the host IPC. MQTT Broker used is based on Mosquitto open-source implementation and supports Sparkplug B specification and security and encryption features. For access details, please refer to Section *External MQTT Communication.*

Cloud Connectivity

Cloud connectivity can be established using Node-RED nodes, which readily support AWS, Azure, and other Cloud providers.

Another option is to use Telegraf plug-ins.

Yet, another option is to use a custom docker container and install cloud agents or Python libraries

4.2.7 PACEdge Internal Communications and Data Flow

PACEdge can be seen as a Swiss army knife when solving a particular problem. Typically, there are more than one solution and implementation option. It is recommended that the user first understands what tools and options are available and then selects the most appropriate implementation to solve a particular problem.

MQTT – Internal Communication Bus

PACEdge has two MQTT Brokers, one of them is dedicated to internal communication only and not accessible from outside. MQTT protocol using an Internal MQTT Broker is a recommended internal data bus within the PACEdge. MQTT is fast, simple to set up, and has low overhead, saving CPU resources. Most agents and applications within PACEdge, such as Connext, Node-RED, Telegraf, and Python, readily support MQTT communication. For example, data received by one of the Connext drivers can be automatically published via MQTT, and Node-RED can subscribe to it. Alternatively, Node-RED can publish MQTT data, and Connext can receive it via the MQTT driver. Telegraf can also publish received data via MQTT, such as CPU utilization statistics. Starting with PACEdge v2.2.0, Telegraf is configured by default to subscribe to Internal MQTT broker, topics starting with emr_v1/…, parse received data, and store it in InfluxDB. This setup can store Connext data in InfluxDB by publishing it to MQTT internal broker with the appropriate topic.

OPC UA – Alternative for Internal Communication

OPC UA can also be used for internal data communication as an alternative to MQTT. For instance, Connext has an OPC UA server, and all data variables that Connext receives are published via OPC UA Server. Consequently, Node-RED can use the OPC UA Client and get data from Connext. Compared with MQTT, note that the OPC UA protocol is significantly heavier on CPU resources and more complicated to set up.

Sharing data via Databases

For not-so-real-time-critical data sharing, an excellent way to consider is storing data in the database and letting other applications extract data directly from it. For example, Connext has a Historian and Datalogger functionality, which can keep the data in a MySQL database. Consequently, Node-RED or Grafana can directly access the data with MySQL and perform further processing or offer visualization.

4.3 PACEdge Remote Access

PACEdge is designed to operate within the internal networks, protected by firewalls and other cyber security devices, the so-called on-premise use model. Nevertheless, there are use cases where remote access to the PACEdge system is required.

Following are a few remote access options that have been tested to work well with PACEdge

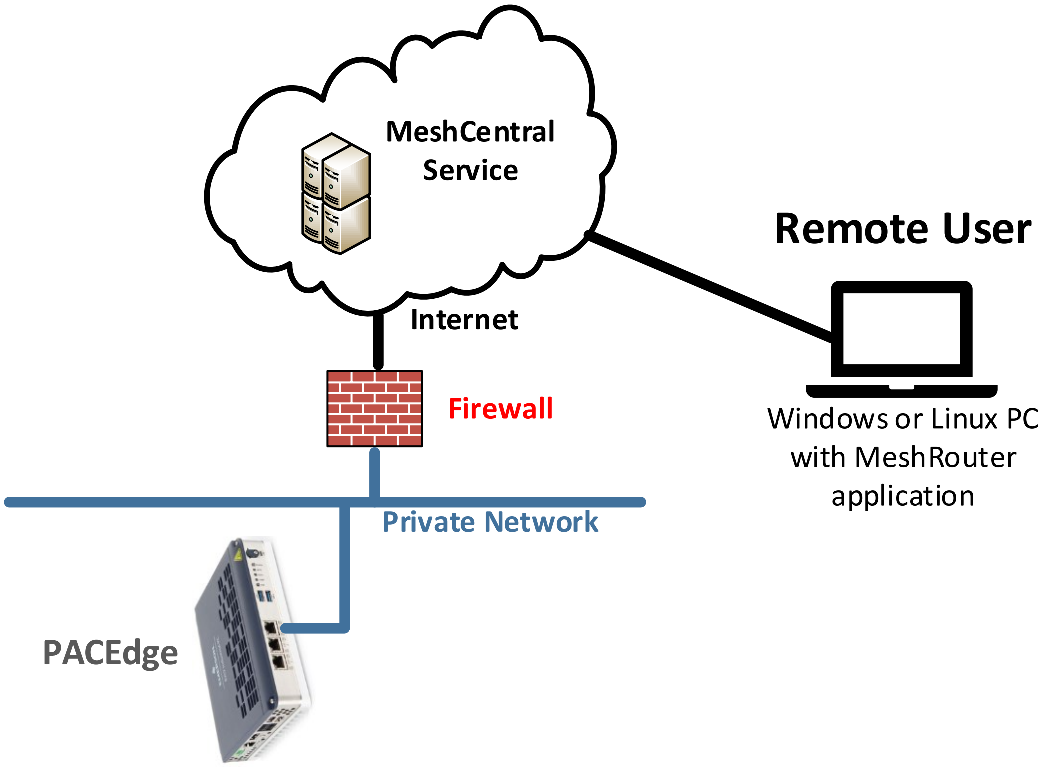
4.3.1 Remote Access using MeshCentral

MeshCentral is an open-source project based on the concept of having a MeshCentral Server hosted on the internet. MeshCentral site on Github has well documented procedure on how to setup your own server on AWS or Azure cloud. PACEdge devices, located on a private network, behind NAT and Firewall, would connect to this server. A remote user would log into the server and gain access to the desired device. Supported access type:

- Command line terminal,
- file browser, transfer,
- desktop view (RXi2-BP and RXi2-LP when the actual monitor is plugged into DisplayPort on the PACEdge device).

If an additional MeshCentral Router application is installed on the remote user's PC, web-based access to PACEdge applications (Node-RED, Grafana, Cockpit, etc.) is also supported.

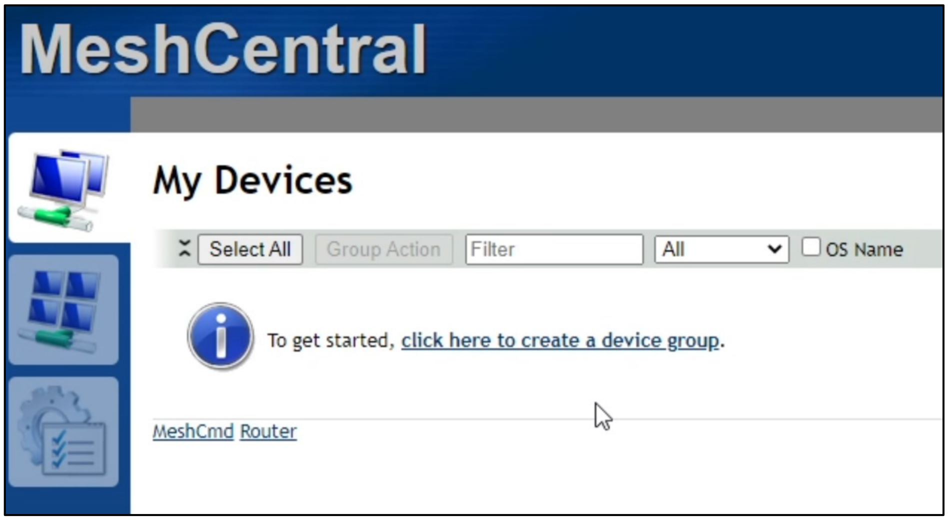
Figure 29 Remote Access using MeshCentral Software



4.3.2 Steps to setup MeshCentral: Create MeshCentral Account

1. Go to the MeshCentral website (https://github.com/Ylianst/MeshCentral), navigate to the docs folder, and check the MeshCentral Install Guide.
2. Setup your own MeshCentral server.
3. Log in to your account at MeshCentral. Once logged in, you will get the following view:

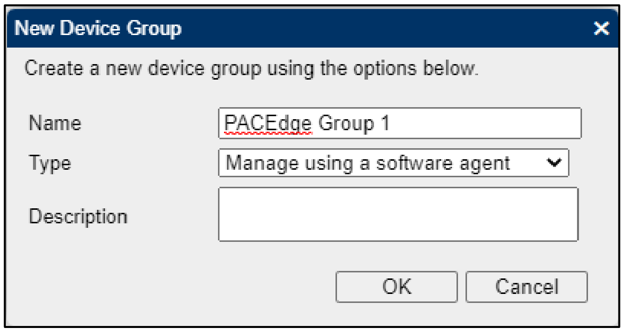
Figure 30: Mesh Control View



To get started, click here to create a device group.

4. Click on the link to create a device group, and give it a name. Leave the other values as default.

Figure 31: Device Group Name



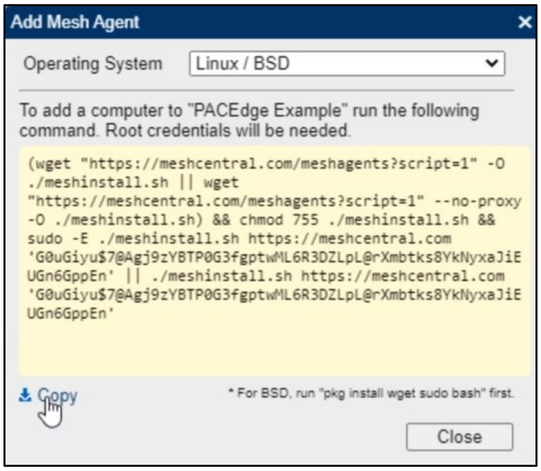
Create a new device group using the options below.

Name: PACEdge Group 1
Type: Manage using a software agent
Description:

5. Click **OK** to add the New Device Group.

6. Select **Linux/BSD** for the operating system and click on **Copy** to copy the link that gets automatically created.

Figure 32: Add Mesh Agent



Operating System: Linux / BSD

To add a computer to "PACEdge Example" run the following command. Root credentials will be needed.

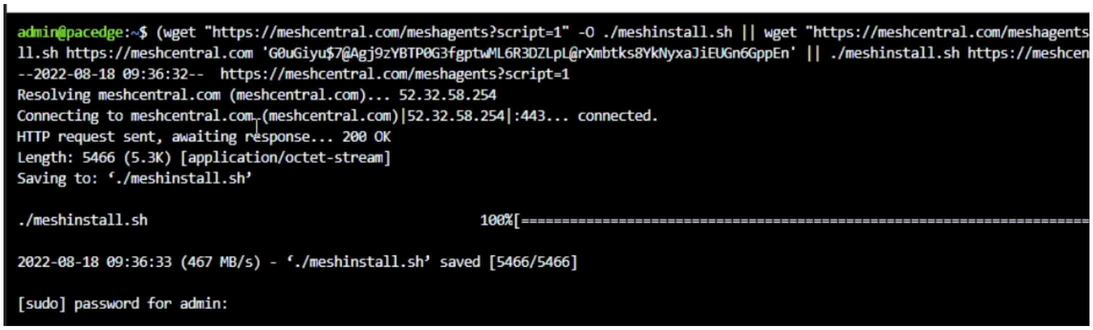
```
(wget "https://meshcentral.com/meshagents?script=1" -O
./meshinstall.sh || wget
"https://meshcentral.com/meshagents?script=1" --no-proxy
-O ./meshinstall.sh) && chmod 755 ./meshinstall.sh &&
sudo -E ./meshinstall.sh https://meshcentral.com
'G0uGiyu$7@Agj9zYBTP0G3fgptwML6R3DZLpL@rXmbtks8YkNyxaJiE
UGn6GppEn' || ./meshinstall.sh https://meshcentral.com
'G0uGiyu$7@Agj9zYBTP0G3fgptwML6R3DZLpL@rXmbtks8YkNyxaJiE
UGn6GppEn'
```

* For BSD, run "pkg install wget sudo bash" first.

Add PACEdge Device to MeshCentral Server

1. Using your web browser, connect to your PACEdge device, go to Cockpit, and log in as an administrator.
2. Go to Terminal, right-click into the Terminal window, and select Paste to paste the link you copied in the above step. Alternatively, you can use a CTRL+V shortcut to paste the link.
3. Hit the Enter key and enter the administrator password once requested.

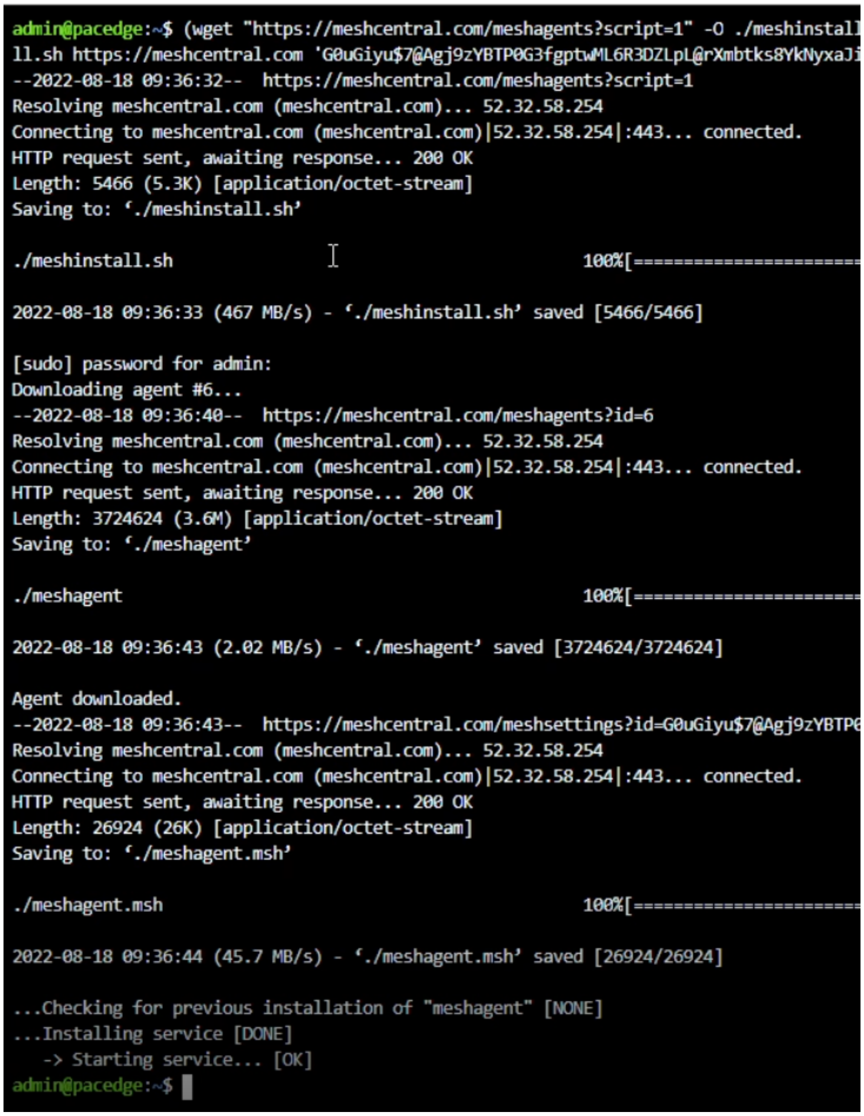
Figure 33: Enter credentials



```
admin@pacedge:~$ (wget "https://meshcentral.com/meshagents?script=1" -O ./meshinstall.sh || wget "https://meshcentral.com/meshagents
ll.sh https://meshcentral.com 'G0uGiyu$7@Agj9zYBTP0G3fgptwML6R3DZLpL@rXmbtks8YkNyxaJiEUGn6GppEn' || ./meshinstall.sh https://meshcen
--2022-08-18 09:36:32--  https://meshcentral.com/meshagents?script=1
Resolving meshcentral.com (meshcentral.com)... 52.32.58.254
Connecting to meshcentral.com (meshcentral.com)|52.32.58.254|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5466 (5.3K) [application/octet-stream]
Saving to: ‘./meshinstall.sh’

./meshinstall.sh                                    100%[====================================================================

2022-08-18 09:36:33 (467 MB/s) - ‘./meshinstall.sh’ saved [5466/5466]

[sudo] password for admin:
```

4. Once the installation of the MeshCentral agent is complete, a message similar to the one below will be shown:

Figure 34: Complete Message

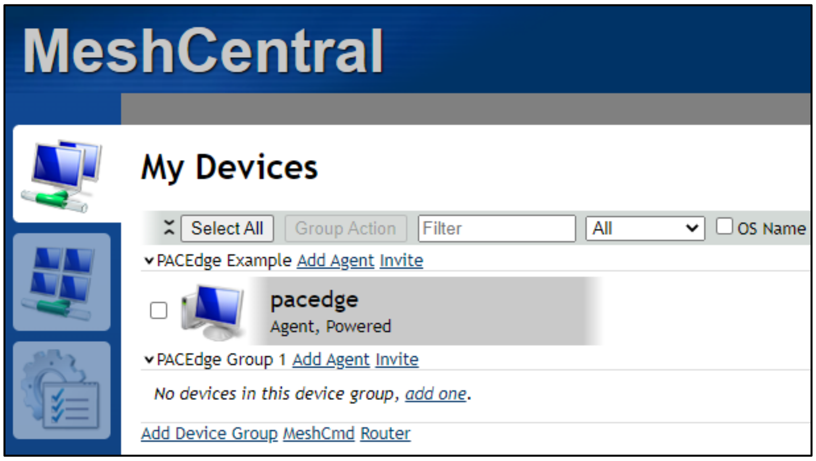


```
admin@pacedge:~$ (wget "https://meshcentral.com/meshagents?script=1" -O ./meshinstall
ll.sh https://meshcentral.com 'G0uGiyu$7@Agj9zYBTP0G3fgptwML6R3DZLpL@rXmbtks8YkNyxaJi
--2022-08-18 09:36:32--  https://meshcentral.com/meshagents?script=1
Resolving meshcentral.com (meshcentral.com)... 52.32.58.254
Connecting to meshcentral.com (meshcentral.com)|52.32.58.254|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5466 (5.3K) [application/octet-stream]
Saving to: ‘./meshinstall.sh’

./meshinstall.sh                        100%[======================

2022-08-18 09:36:33 (467 MB/s) - ‘./meshinstall.sh’ saved [5466/5466]

[sudo] password for admin:
Downloading agent #6...
--2022-08-18 09:36:40--  https://meshcentral.com/meshagents?id=6
Resolving meshcentral.com (meshcentral.com)... 52.32.58.254
Connecting to meshcentral.com (meshcentral.com)|52.32.58.254|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3724624 (3.6M) [application/octet-stream]
Saving to: ‘./meshagent’

./meshagent                             100%[======================

2022-08-18 09:36:43 (2.02 MB/s) - ‘./meshagent’ saved [3724624/3724624]

Agent downloaded.
--2022-08-18 09:36:43--  https://meshcentral.com/meshsettings?id=G0uGiyu$7@Agj9zYBTP6
Resolving meshcentral.com (meshcentral.com)... 52.32.58.254
Connecting to meshcentral.com (meshcentral.com)|52.32.58.254|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 26924 (26K) [application/octet-stream]
Saving to: ‘./meshagent.msh’

./meshagent.msh                         100%[======================

2022-08-18 09:36:44 (45.7 MB/s) - ‘./meshagent.msh’ saved [26924/26924]

...Checking for previous installation of "meshagent" [NONE]
...Installing service [DONE]
   -> Starting service... [OK]
admin@pacedge:~$ 
```

Test Remote Access to PACEdge

1. Log in to the MeshCentral Server and locate the PACEdge device.

Figure 35: PACEdge Device



PACEdge Example

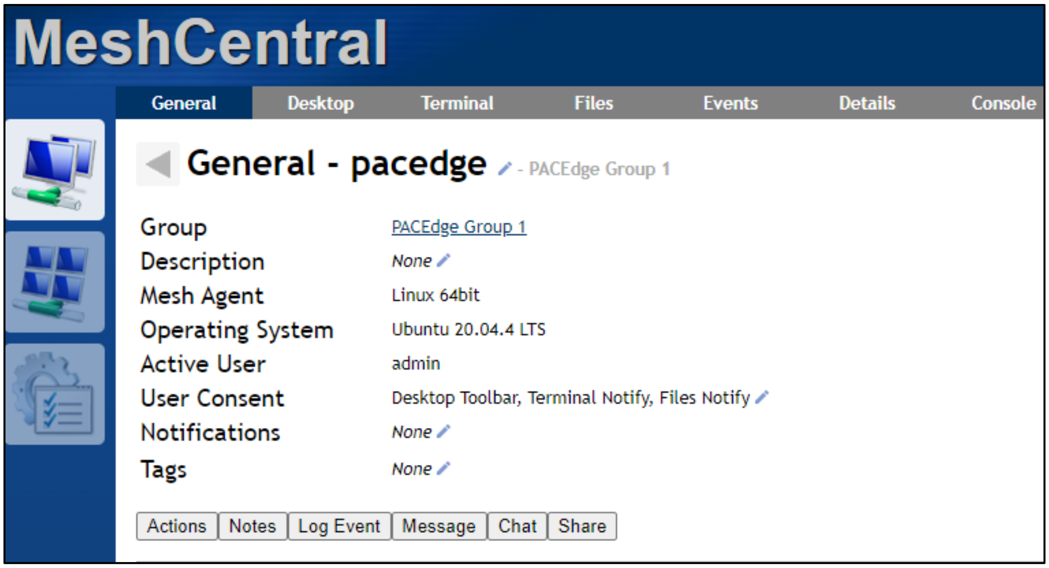
□ pacedge
Agent, Powered

PACEdge Group 1

No devices in this device group, add one.

2. Click on the device to access the command line and file transfer access

Figure 36: File Transfer Access

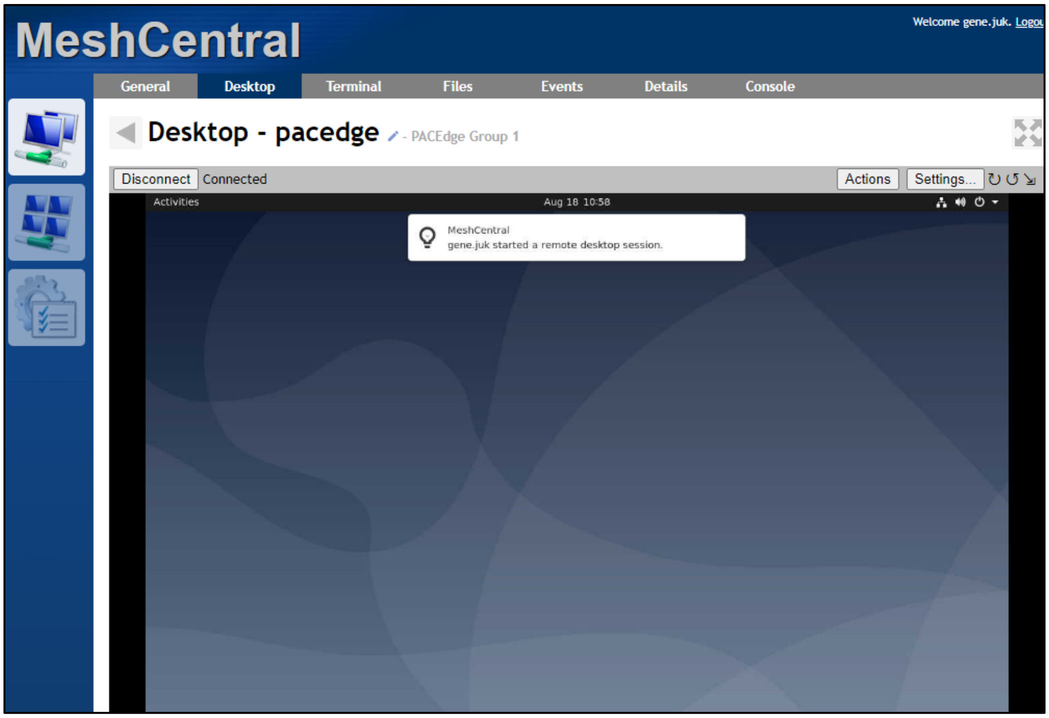


Group	PACEdge Group 1
Description	*None*
Mesh Agent	Linux 64bit
Operating System	Ubuntu 20.04.4 LTS
Active User	admin
User Consent	Desktop Toolbar, Terminal Notify, Files Notify
Notifications	*None*
Tags	*None*

NOTE: Depending on the PACEdge HW SKU and monitor being plugged into the PACEdge device via DisplayPort interface, the Desktop option might not be available

3. To access PACEdge on your desktop, click on the **Desktop** option and then click on the **Connect** button.
   Note: For this connection to work, a monitor must be plugged into the PACEdge device via the DisplayPort interface, and the user needs to be logged in to PACEdge via the local monitor, keyboard, mouse interface

Figure 37: Connected Device



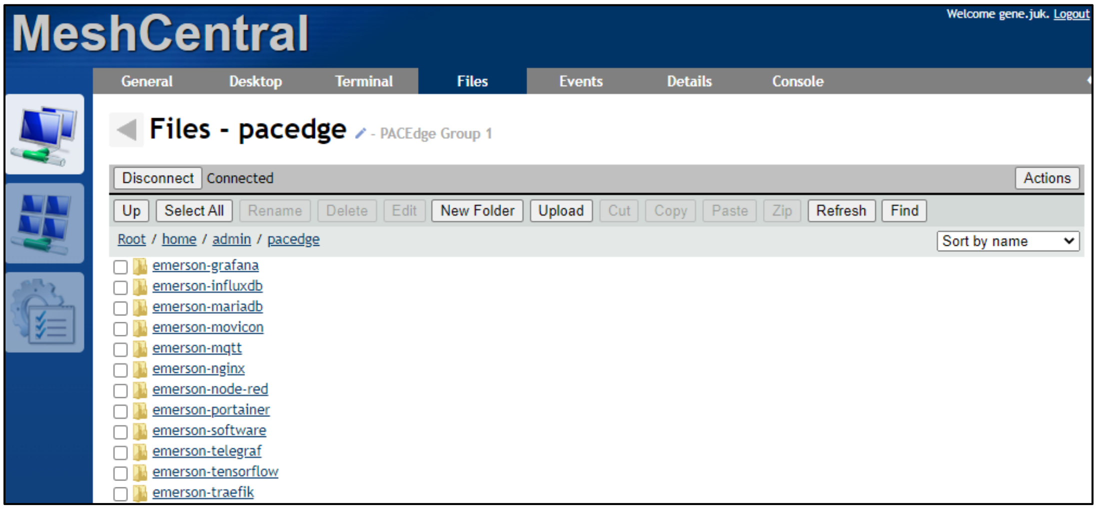
Connected

MeshCentral
gene.juk started a remote desktop session.

4. To access PACEdge's command line, click **Terminal** and then **Connect**.

5. To access PACEdge's file browser and transfer system, click on **Files** and then on **Connect**

Figure 38: Gaining file Browser/Transfer Access to PACEdge



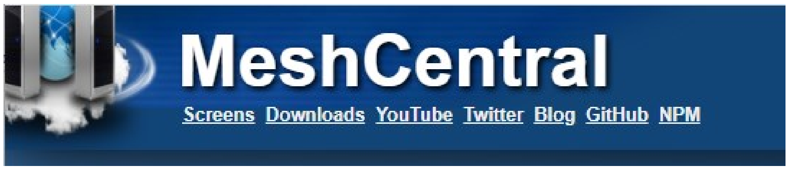
Root / home / admin / pacedge

- □ emerson-grafana
- □ emerson-influxdb
- □ emerson-mariadb
- □ emerson-movicon
- □ emerson-mqtt
- □ emerson-nginx
- □ emerson-node-red
- □ emerson-portainer
- □ emerson-software
- □ emerson-telegraf
- □ emerson-tensorflow
- □ emerson-traefik

Setup Remote Access to PACEdge Applications (Node-Red, Grafana, Cockpit)

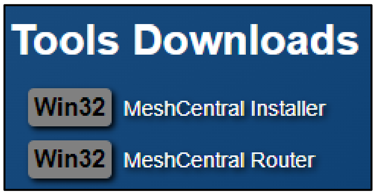
1. Go to https://meshcentral.com/downloads.html

Figure 39: Downloads Link



2. From the Tools Downloads section, click on the **Win32 MeshCentral Router** link (assuming you are using Windows based PC)

Figure 40: Tools Downloads



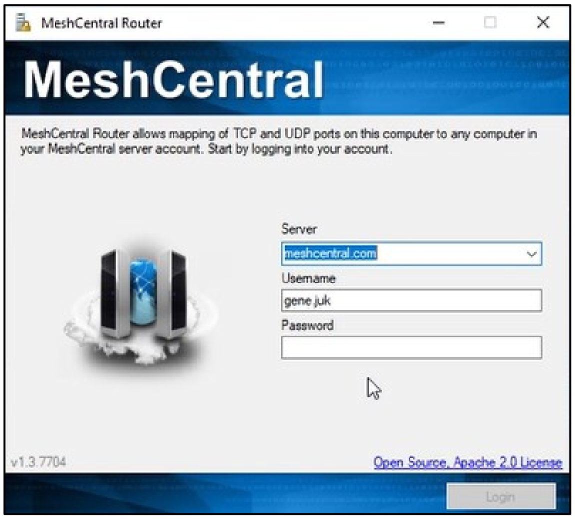
Tools Downloads

Win32 MeshCentral Installer

Win32 MeshCentral Router

3. Save MeshCentral Router locally on your PC and execute it
4. Log in with your MeshCentral credentials:

Figure 41: Log in



MeshCentral Router allows mapping of TCP and UDP ports on this computer to any computer in your MeshCentral server account. Start by logging into your account.

Server: meshcentral.com

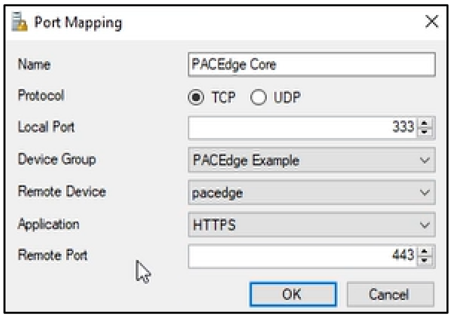
Username: gene juk

Password:

v1.3.7704

5. Once logged in, make sure you can see your PACEdge device under the **Devices** tab
6. Click on the **Mappings** tab
7. Click on **Add Map** button
8. Fill in details by choosing a port number that will be used locally (on your PC, use a large number 300-1000 range), selecting your remote PACEdge device, selecting HTTPS, and keeping the Remote Port number 443.

Figure 42: Port Mapping

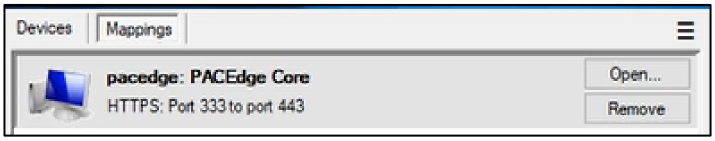


Port Mapping

Name: PACEdge Core
Protocol: ● TCP ○ UDP
Local Port: 333
Device Group: PACEdge Example
Remote Device: pacedge
Application: HTTPS
Remote Port: 443

9. Click the **OK** button and then click on the Open button in the dialogue below:

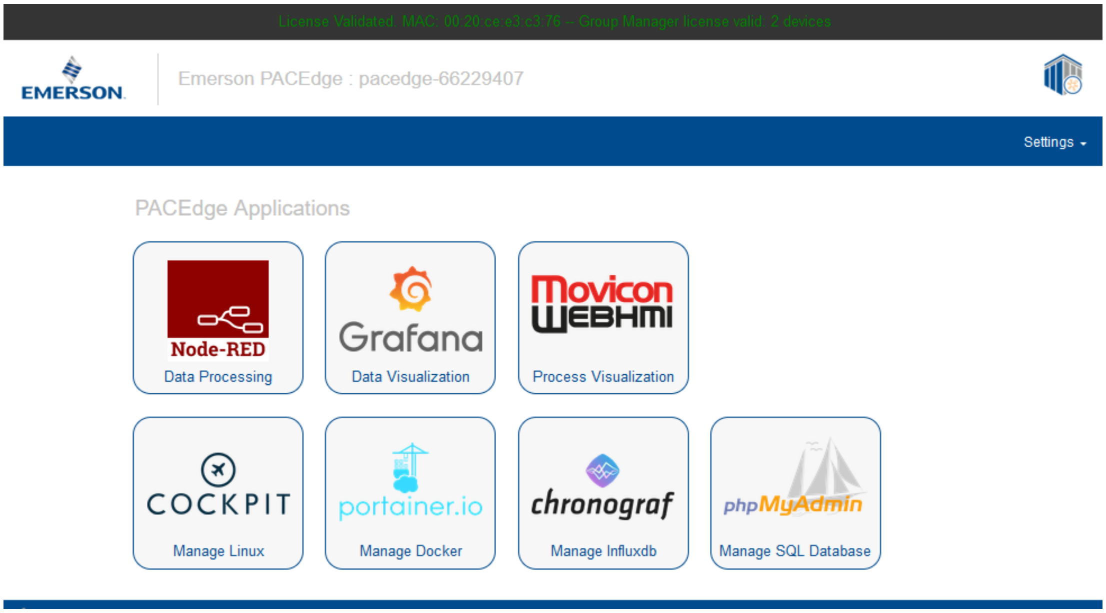
Figure 43: Mappings



pacedge: PACEdge Core
HTTPS: Port 333 to port 443

10. A new browser window will open, giving you access to the PACEdge landing page

Figure 44: PACEdge Landing Page



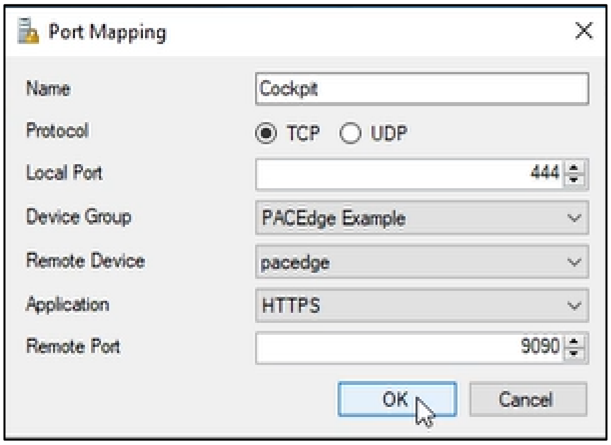
License Validated. MAC: 00:20:ce:e3:c3:76 -- Group Manager license valid: 2 devices

Emerson PACEdge : pacedge-66229407

11. Now you can access Node-RED, Grafana, Portainer, and Chronograf
Note: access to Cockpit via the link will not work because Cockpit requires using port number 9090. The following section will describe how to gain access to Cockpit.

12. To access Cockpit, create a second Port Mapping and make sure that Remote Port is manually set to **9090:**

Figure 45: Port Mapping



Port Mapping

Name: Cockpit

Protocol: ● TCP ○ UDP

Local Port: 444

Device Group: PACEdge Example

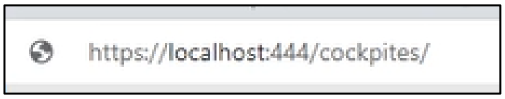
Remote Device: pacedge

Application: HTTPS

Remote Port: 9090

13. Click on the **Open** button for this new Port Mapping, and when you receive an error page “Not Found,” modify the address in the browser by adding /cockpites/ at the end:

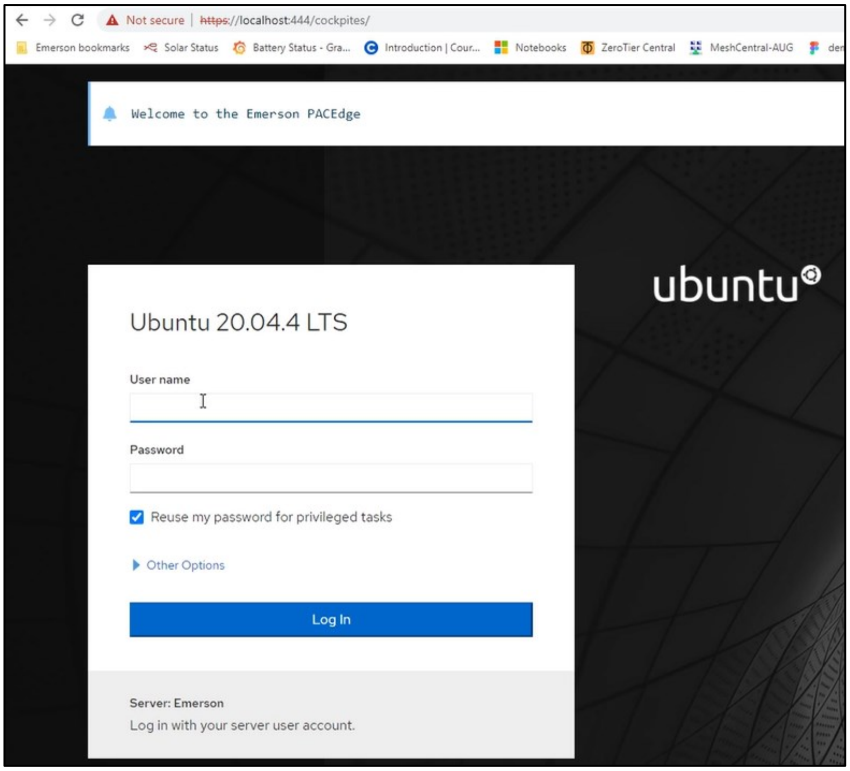
Figure 46: Localhost



https://localhost:444/cockpites/

14. Hit enter. You will have access to the Cockpit on your remote PACEdge device

Figure 47: Access Cockpit on Remote PACEdge Device



Welcome to the Emerson PACEdge

Ubuntu 20.04.4 LTS

User name

Password

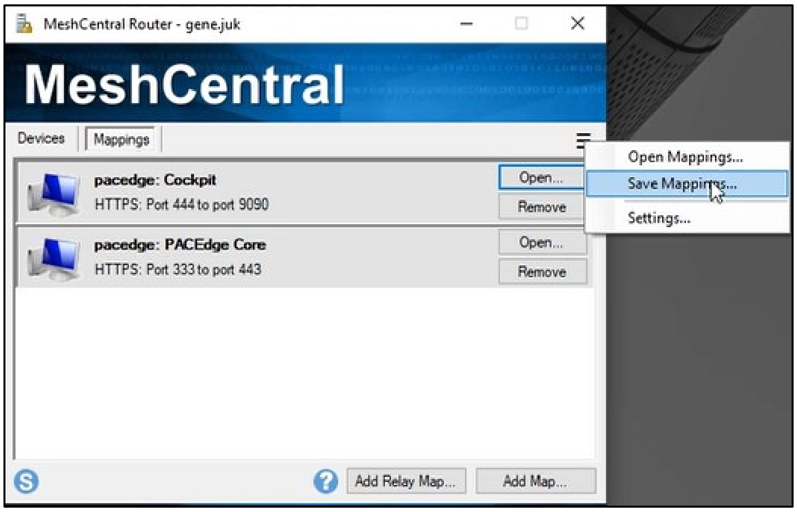
☑ Reuse my password for privileged tasks

Server: Emerson

Log in with your server user account.

15. To save the newly created Port Mappings, click on the three lines in the upper right corner and choose Save Mappings:

Figure 48: Save Mappings



pacedge: Cockpit
HTTPS: Port 444 to port 9090

pacedge: PACEdge Core
HTTPS: Port 333 to port 443

16. Next time MeshCentral Router is started, select Open Mappings to restore your Port Mapping.

4.3.3 Remote Access using ZeroTier

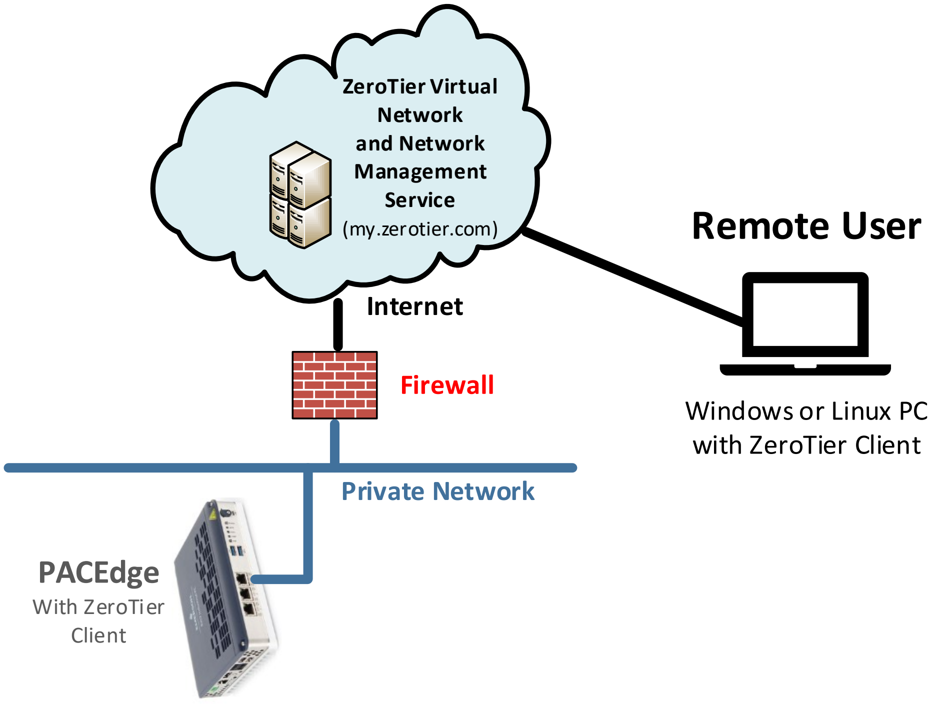
ZeroTier is a commercial service offering allowing the user to create a virtual network that spans different physical sites and devices behind NAT and firewalls. This can connect a user's PC to one or more remote PACEdge devices as if plugged into the same local physical network.

Although ZeroTier is a commercial offering, the Basic version, limited to 25 nodes, is free of charge and is well suited for testing.

To setup ZeroTier following steps need to be executed:

1. Create a ZeroTier account
2. Download and install ZeroTier onto your PC
3. ZeroTier is already pre-installed on PACEdge version 2.2 or later devices but deactivated by default.
4. Configure devices to join the same virtual network

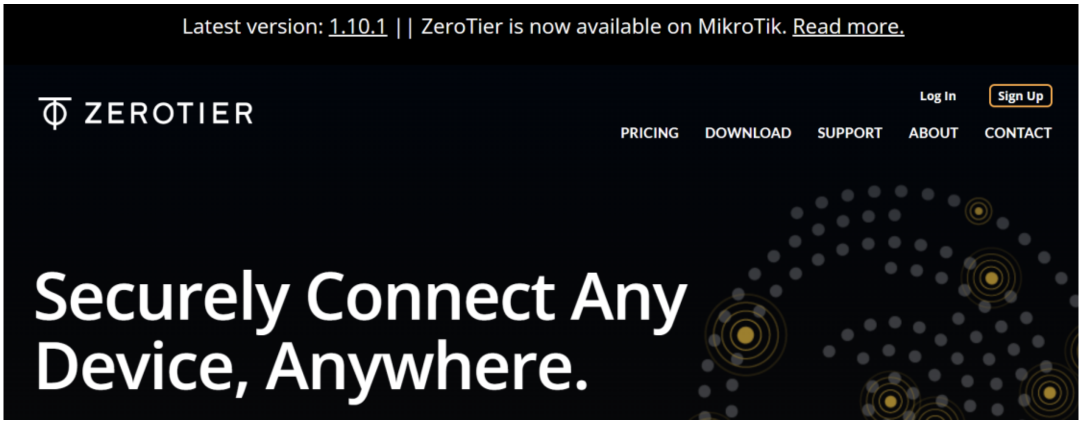
Figure 49 Remote Access using ZeroTier Software


ZeroTier Virtual Network and Network Management Service (my.zerotier.com)
Remote User
Internet
Firewall
Windows or Linux PC with ZeroTier Client
Private Network
PACEdge
With ZeroTier Client

Create ZeroTier account

1. Using a browser on your PC, go to www.zerotier.com and click on **Sign Up**

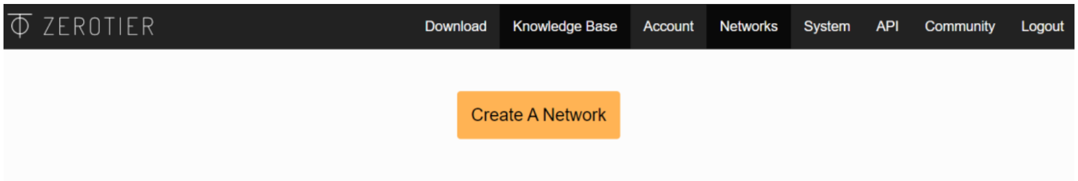
Figure 50: ZeroTier Account



Latest version: 1.10.1 | | ZeroTier is now available on MikroTik. Read more.

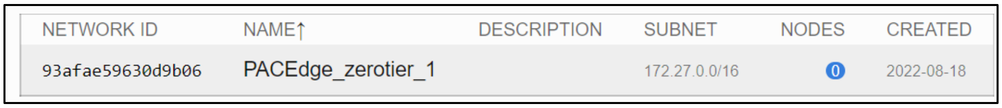
2. Once signed up, go to my.zerotier.com and log in
3. Click on Create a Network button

Figure 51: Create a Network



4. If you want to edit the network name or other properties, click on the created new network.
5. Copy the **Network ID** and keep it handy for the next steps

Figure 52: Network ID

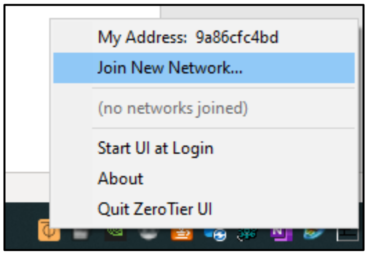


NETWORK ID	NAME↑	DESCRIPTION	SUBNET	NODES	CREATED
93afae59630d9b06	PACEdge_zerotier_1		172.27.0.0/16	0	2022-08-18

Install ZeroTier on PC

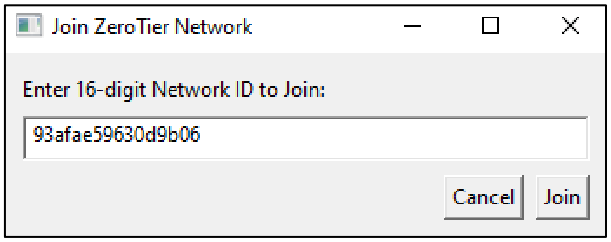
1. Go to www.zerotier.com/download/, select your operating system, download and install ZeroTier
2. Launch ZeroTier on your PC
Note: ZeroTier service might appear as one of the icons in the lower right corner of Windows. In such case, right-click on the ZeroTier icon and select Join New Network

Figure 53: Join New Network



3. Enter **Network ID** that was copied in the previous steps and click on **Join**

Figure 54: Enter Network ID



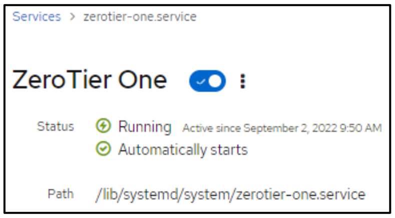
Enter 16-digit Network ID to Join: 93afae59630d9b06

Configure PACEdge Device to Join ZeroTier Network

1. By default, the ZeroTier service is installed in PACEdge but is disabled. To enable it, go to Cockpit, Services tab, scroll down to zerotier-one service, and click on it.

2. Click on the button to **Start and Enable**. The controller will turn blue, and the status will show Running

Figure 55: ZeroTier Enabled Toggle



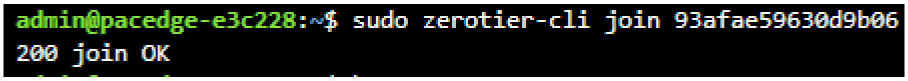
ZeroTier One

Status: Running Active since September 2, 2022 9:50 AM
Automatically starts

Path: /lib/systemd/system/zerotier-one.service

3. Copy **Network ID**, steps above, go to Cockpit, Terminal, and type the following:
*sudo zerotier-CLI join **Network_ID***
Where **Network_ID** is a 16-digit number from the steps above, enter the password if requested. The result will be: **join OK**

Figure 56: Join OK



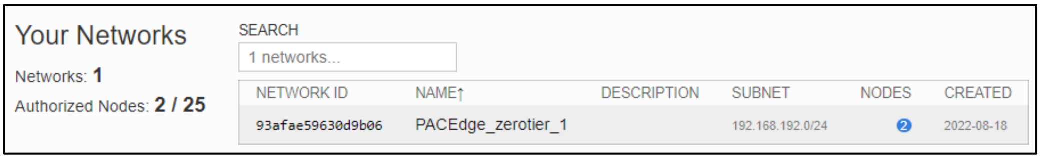
```
admin@pacedge-e3c228:~$ sudo zerotier-cli join 93afae59630d9b06
200 join OK
```

Configure and Start Virtual Network

1. Log into your account on my.zerotier.com site

2. Click on the network

Figure 57: Click on Network



Your Networks

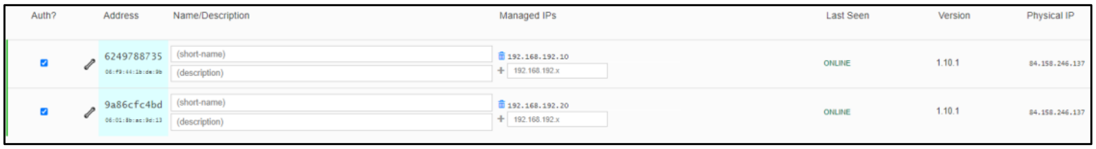
Networks: **1**

Authorized Nodes: **2 / 25**

NETWORK ID	NAME↑	DESCRIPTION	SUBNET	NODES	CREATED
93afae59630d9b06	PACEdge_zerotier_1		192.168.192.0/24	2	2022-08-18

3. Scroll down to where both devices (your PC and PACEdge) are listed with the option to Authenticate access to the network

Figure 58: Locate Devices



Auth?		Address	Name/Description	Managed IPs	Last Seen	Version	Physical IP
☑		6249788735 06:f9:44:1b:de:9b	(short-name) (description)	192.168.192.10 + 192.168.192.x	ONLINE	1.10.1	84.158.246.137
☑		9a86cfc4bd 06:01:8b:ac:9d:13	(short-name) (description)	192.168.192.20 + 192.168.192.x	ONLINE	1.10.1	84.158.246.137

4. Check **Auth?** Box for each device or optionally assign a specific IP address for each device. **Last Seen** status will be showing: **ONLINE**

Accessing PACEdge Remotely

1. From your PC, open a browser and type in the IP address assigned in the step above. In this particular example, it would be: 192.168.192.10
2. The PACEdge landing page will open with access to all PACEdge applications and Cockpit.

4.3.4 Remote Access using WireGuard

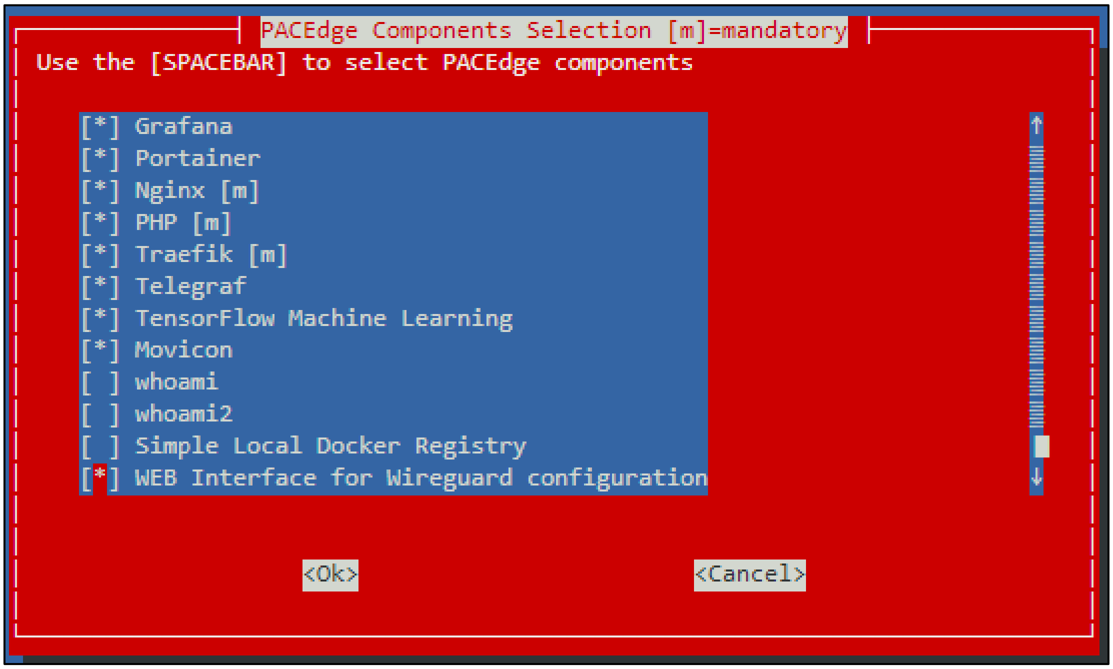
WireGuard is another VPN software that can establish a secure point-to-point connection between PACEdge and remote computers. WireGuard is a relatively simple and easy setup service, which does not require an external Server as a rendezvous point; however, it requires that the desired network port be opened in the exterior router or firewall, which gates access to the internal network where PACEdge is located.

PACEdge comes pre-installed with WireGuard service and an optional graphical WireGuard UI interface, which runs in its container. WireGuard UI container is disabled by default but can be enabled using the config-compose script. To enable the WireGuard UI container, please do the following:

1. Open Cockpit and go to Terminal
2. Type: cd /home/admin/pacedge
3. Type: ./config-compose.sh

Select **Yes** and hit **Enter** key to shut down PACEdge containers
In the list of available containers, using **arrow** keys, scroll down, and using the **space** key, select: **WEB Interface for Wireguard configuration**

Figure 59: Select WEB Interface for Wireguard Configuration



```
PACEdge Components Selection [m]=mandatory
Use the [SPACEBAR] to select PACEdge components

    [*] Grafana
    [*] Portainer
    [*] Nginx [m]
    [*] PHP [m]
    [*] Traefik [m]
    [*] Telegraf
    [*] TensorFlow Machine Learning
    [*] Movicon
    [ ] whoami
    [ ] whoami2
    [ ] Simple Local Docker Registry
    [*] WEB Interface for Wireguard configuration
```

4. Hit the **Tab** key to select OK and then enter; respond with enter key to start the PACEdge again.

5. Once all containers are up and running again, you can access WireGuard graphical configuration utility by entering the PACEdge IP address, followed by port number 5005, for example:
   http://192.168.3.100:5005
   user: admin
   password: edgestack

6. For details on how to configure WireGuard service on both ends, please consult online documentation at www.wireguard.com

7. **NOTE:** For Cybersecurity reasons and CPU resource optimization, once the WireGuard configuration is finished, please remove the WireGuard UI container by following the same steps as above, only this time removing the check mark in WEB Interface for Wireguard configuration.

4.3.5 Remote Access using OpenVPN

Starting with version 2.2.0, PACEdge comes with pre-installed OpenVPN software. OpenVPN software can be used to create secure point-to-point connections. PACEdge can be configured to act as both OpenVPN Server and a Client. If the user desires to use the OpenVPN service, it must be properly configured and started.
Note: if configured as Server, a network port in the router/firewall will have to be opened so that the external Client can reach PACEdge and establish a secure tunnel.

For OpenVPN configuration and use details, please consult the following:

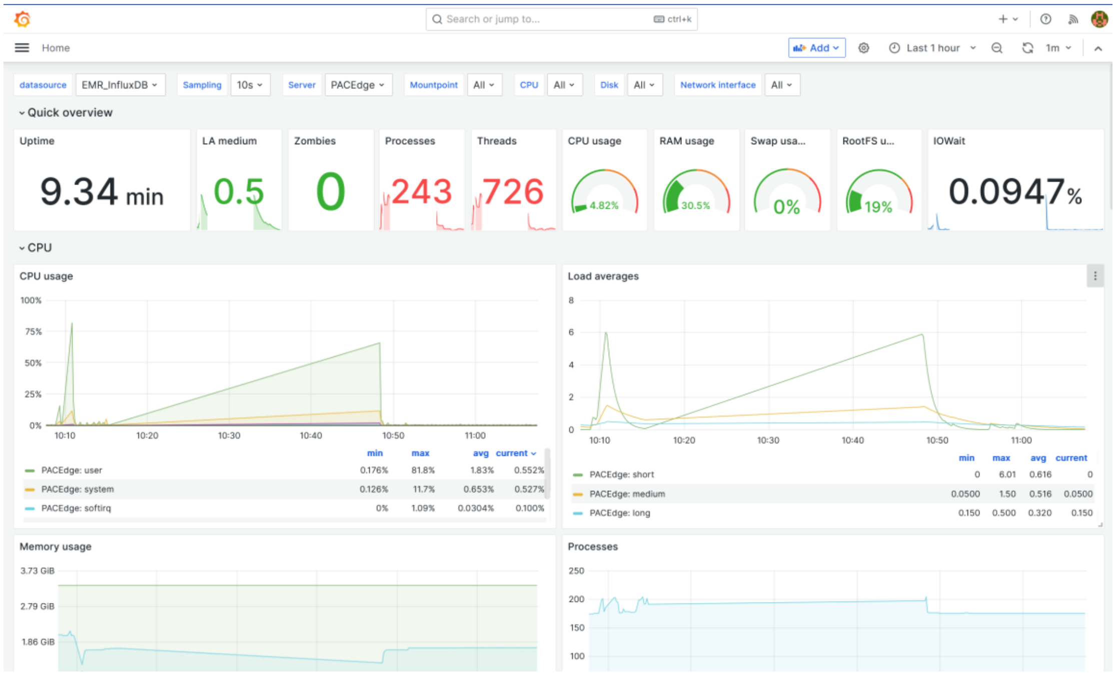
- github.com/OpenVPN/openvpn

openvpn.net (commercial offering based on OpenVPN)

4.4 PACEdge HW and SW Utilization and Statistics

PACEdge is configured to by default collect system statistics, such as CPU utilization, Memory usage, storage, network, kernel activities and store this data in InfluxDB database. Grafana has a pre-configured dashboard that allows to view and analyze these statistics. By default, InfluxDB is configured to have 7 day data retention period.

Figure 60: HW and SW Utilization and Statistics



datasource: EMR_InfluxDB · Sampling: 10s · Server: PACEdge · Mountpoint: All · CPU: All · Disk: All · Network interface: All · Last 1 hour · 1m

Quick overview

Uptime 9.34 min · LA medium 0.5 · Zombies 0 · Processes 243 · Threads 726 · CPU usage 4.82% · RAM usage 30.5% · Swap usa... 0% · RootFS u... 19% · IOWait 0.0947%

CPU

CPU usage

	min	max	avg	current
PACEdge: user	0.176%	81.8%	1.83%	0.552%
PACEdge: system	0.126%	11.7%	0.653%	0.527%
PACEdge: softirq	0%	1.09%	0.0304%	0.100%

Load averages

	min	max	avg	current
PACEdge: short	0	6.01	0.616	0
PACEdge: medium	0.0500	1.50	0.516	0.0500
PACEdge: long	0.150	0.500	0.320	0.150

4.6 Group Manager

PACEdge Group Manager is a feature that allows users to designate one PACEdge device as a Group Manager and then use it to update other PACEdge devices. As of PACEdge version 2.3.0 release Group Manager will support following features:

- Host Linux operating system update
- PACEdge Application (container) image updates
- PACEdge Application configuration updates
- Node-RED flow updates
- User interface: command line scripts

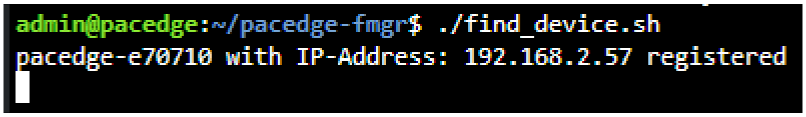
4.6.1 Configuring Device Groups

Before the Group Manager can be used, PACEdge devices need to be added to the Groups and provisioned with security keys so that Group Manager can later access them without providing the admin password for each device. The PACEdge device that is designated to be the Group Manager has a pre-configured ssh key, which needs to be securely transferred to each target device. This is done by using **device_onboarding** playbook, but before any playbook can be executed, the target device IP address needs to be added to the group file, called **hosts.ini**. Users can manually edit the **hosts.ini** file, or use an automated utility (**find_device.sh**) to find and add new devices to the hosts.ini file.
These are the steps to provision new PACEdge devices for Group Management:

Autodetect Target Devices

- On a Group Manager, go to Cockpit->Terminal, navigate to folder **pacedge-fmgr**:
  - **cd /home/admin/pacedge-fmgr/**
- This will start a utility which will wait for new devices to show up on the network (terminal window prompt will not return, leave it this way)
- Power one or more PACEdge devices that you want to manage (target devices). Make sure these devices are connected to the same network where Group Manager is connected. Once devices boot up, they will automatically broadcast a special message which Group Manager will receive and then automatically add each device to the Group (hosts.ini file):

Figure 61: Message when target device is detected

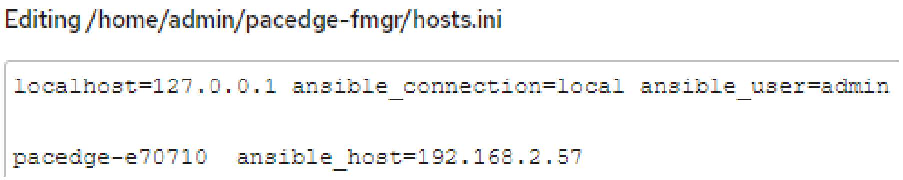


```
admin@pacedge:~/pacedge-fmgr$ ./find_device.sh
pacedge-e70710 with IP-Address: 192.168.2.57 registered
```

Modify Group File

1. Using Cockpit->Navigator, navigate to the following path: **/home/admin/pacedge-fmgr** and open the **hosts.ini** file:

Figure 62: hosts.ini file view with one target device

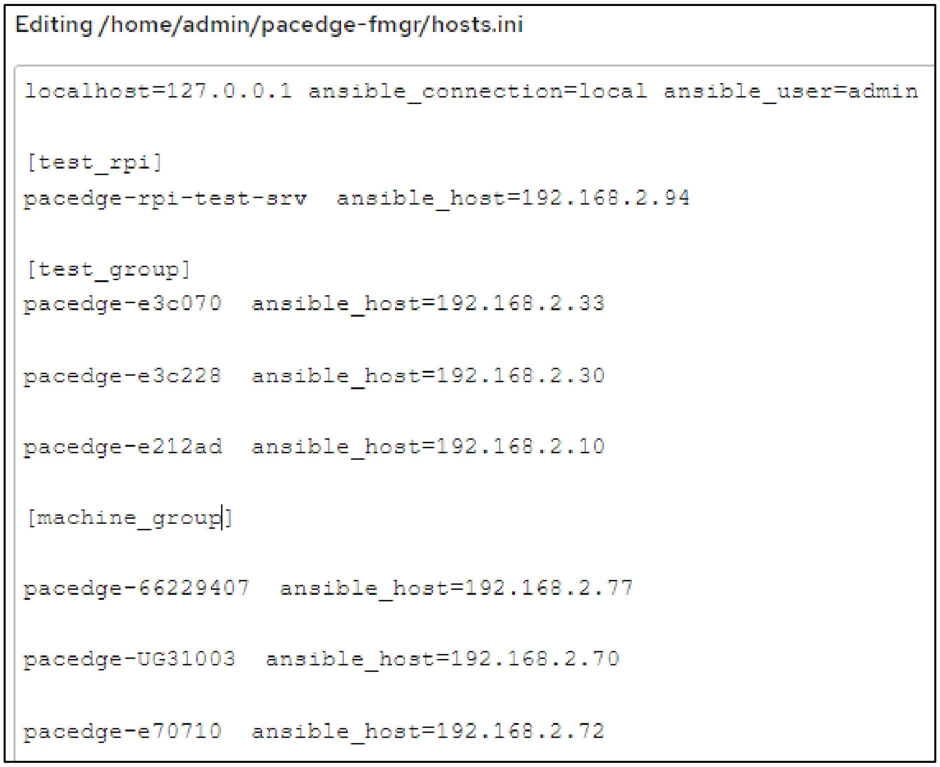


Editing /home/admin/pacedge-fmgr/hosts.ini

```
localhost=127.0.0.1 ansible_connection=local ansible_user=admin

pacedge-e70710  ansible_host=192.168.2.57
```

2. Edit **hosts.ini** file by giving your Group a name (to be entered in square brackets) or manually adding additional devices. You can also create multiple Groups. Afterwards, the software updates can be executed on a per-group basis.
   The following is an example of a hosts.ini file with multiple groups:

Figure 63: hosts.ini file view, multiple Groups



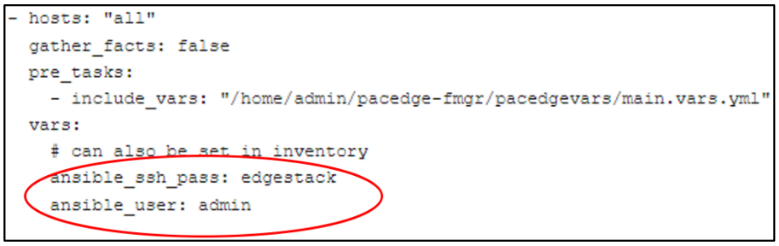
Editing /home/admin/pacedge-fmgr/hosts.ini

```
localhost=127.0.0.1 ansible_connection=local ansible_user=admin

[test_rpi]
pacedge-rpi-test-srv  ansible_host=192.168.2.94

[test_group]
pacedge-e3c070  ansible_host=192.168.2.33

pacedge-e3c228  ansible_host=192.168.2.30

pacedge-e212ad  ansible_host=192.168.2.10

[machine_group]

pacedge-66229407  ansible_host=192.168.2.77

pacedge-UG31003  ansible_host=192.168.2.70

pacedge-e70710  ansible_host=192.168.2.72
```

Provision Target Devices with Security Credentials

1. Using Cockpit->Navigator, open file: **/home/admin/pacedge-fmgr/playbooks/device_onboarding.yml** and change: **ansible_ssh_pass** value to your target devices Linux admin user password.

Figure 64: Setting admin password in device_onboarding.yml



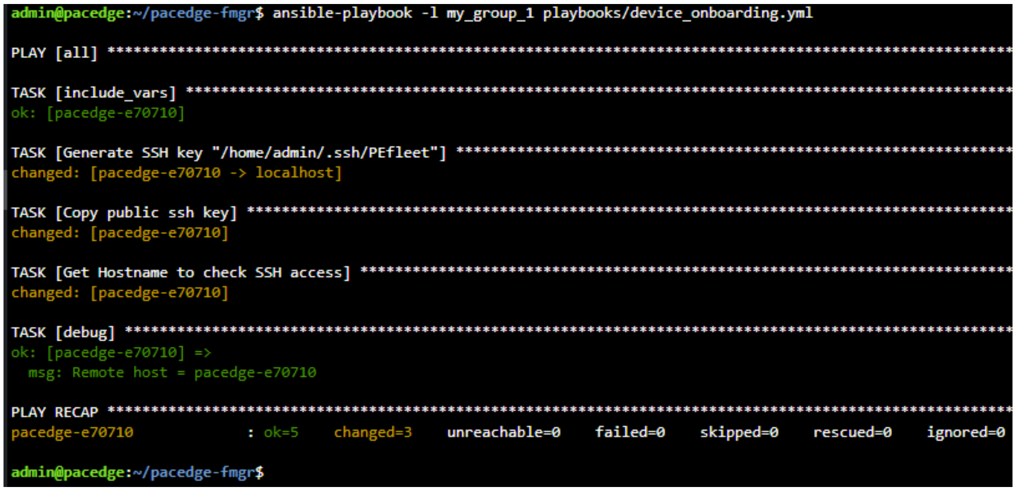
```
- hosts: "all"
  gather_facts: false
  pre_tasks:
    - include_vars: "/home/admin/pacedge-fmgr/pacedgevars/main.vars.yml"
  vars:
    # can also be set in inventory
    ansible_ssh_pass: edgestack
    ansible_user: admin
```

NOTE: This playbook is required only for the first provisioning step. For security purposes, please delete the actual password or change it to some other value after this playbook has been executed in the next step.

2. Provision target devices by setting up secure ssh key. In Cockpit->Terminal, while in **/home/admin/pacedge-fmgr/** folder, issue command:

    a. **ansible-playbook -l my_group_1 playbooks/device_onboarding.yml**
    replace my_group_1 with the name of the group that you have entered in the hosts.ini file.

Figure 65: Output of the device onboarding playbook



```
admin@pacedge:~/pacedge-fmgr$ ansible-playbook -l my_group_1 playbooks/device_onboarding.yml

PLAY [all] ******************************************************************************************************

TASK [include_vars] *********************************************************************************************
ok: [pacedge-e70710]

TASK [Generate SSH key "/home/admin/.ssh/PEfleet"] **************************************************************
changed: [pacedge-e70710 -> localhost]

TASK [Copy public ssh key] **************************************************************************************
changed: [pacedge-e70710]

TASK [Get Hostname to check SSH access] *************************************************************************
changed: [pacedge-e70710]

TASK [debug] ****************************************************************************************************
ok: [pacedge-e70710] =>
  msg: Remote host = pacedge-e70710

PLAY RECAP ******************************************************************************************************
pacedge-e70710             : ok=5    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

admin@pacedge:~/pacedge-fmgr$
```

4.6.2 Update Host Linux Operating System

Group Manager can be used for Host Linux OS updates. At the time of PACEdge version 2.3.0 release there is an option to upgrade older PACEdge devices version 2.2.x to version 2.3.0 using Group Manager. Such upgrade allows upgrading multiple devices remotely. For detailed procedure refer to section: 9.2.1 Upgrading via Group Manager.

4.6.3 Update PACEdge Applications

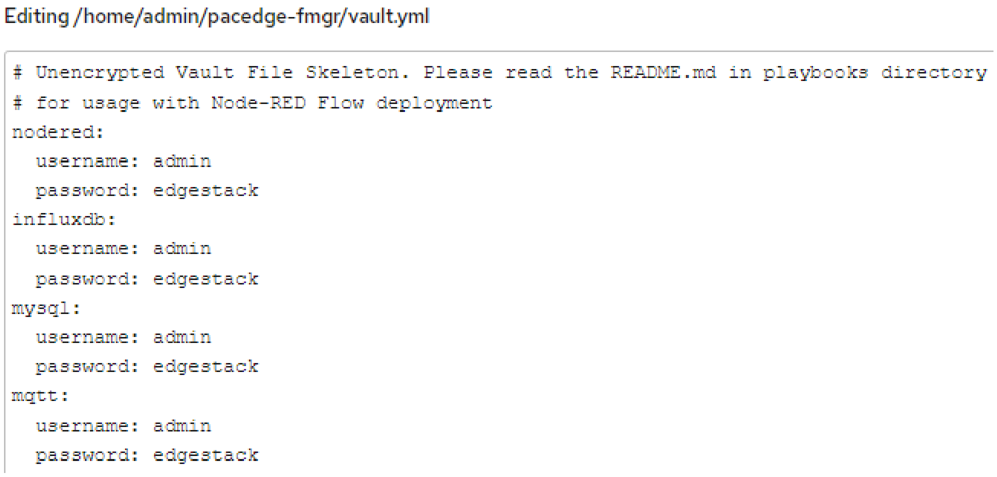
Group Manager can be used for PACEdge Application updates. At the time of PACEdge version 2.3.0 release there is an option to upgrade older PACEdge devices version 2.2.x to version 2.3.0 using Group Manager. Such upgrade allows upgrading multiple devices remotely. For detailed procedure refer to section: 9.2.1 Upgrading via Group Manager.

4.6.5 Update Node-RED flows

The Group Manager supports deploying Node-RED flows to target PACEdge devices. To do so:

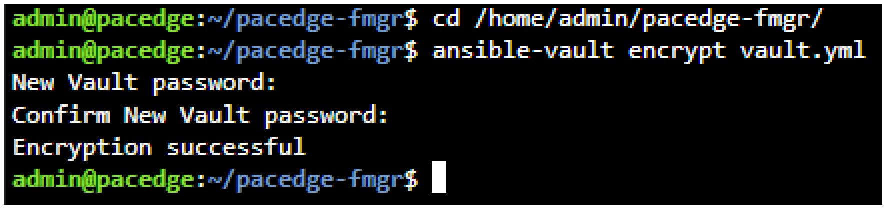
- Create a desired Node-RED flow and export it by using Export->Download option within Node-RED.
- Using Cockpit->Navigator, drag and drop the exported flow to folder: **/home/admin/pacedge-fmgr/repos/private/nodered-flows**
- If your Node-RED flow has nodes that require username and password authentication, such as InfluxDB, MySQL or MQTT nodes, then you need to configure username and password in an encrypted vault file. To do so, using Cockpit->Navigator open and edit file**: /home/admin/pacedge-fmgr/vault.yml.** Enter desired username and password values for each node and save the file.

Figure 66: View of Vault File



Editing /home/admin/pacedge-fmgr/vault.yml

```
# Unencrypted Vault File Skeleton. Please read the README.md in playbooks directory
# for usage with Node-RED Flow deployment
nodered:
  username: admin
  password: edgestack
influxdb:
  username: admin
  password: edgestack
mysql:
  username: admin
  password: edgestack
mqtt:
  username: admin
  password: edgestack
```

- Encrypt the vault file. Using Cockpit->Terminal, change into folder: /home/admin/pacedge-fmgr and then issue encrypt command as follows:
  - **cd /home/admin/pacedge-fmgr**
  - **ansible-vault encrypt vault.yml**
  - Note: you will be asked to secure the vault access with the password, enter it as requested
  - 

```
admin@pacedge:~/pacedge-fmgr$ cd /home/admin/pacedge-fmgr/
admin@pacedge:~/pacedge-fmgr$ ansible-vault encrypt vault.yml
New Vault password:
Confirm New Vault password:
Encryption successful
admin@pacedge:~/pacedge-fmgr$
```

- If you want to edit the vault file again, you first need to decrypt it by issuing:
  - **ansible-vault decrypt vault.yml**
- Deploy Node-RED flow to a Group. Using Cockpit->Terminal, while in folder: /home/admin/pacedge-fmgr issue command:

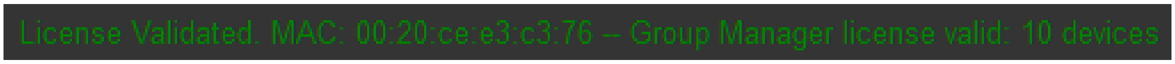
- ○ **ansible-playbook playbooks/deploy_flow.yml -l my_group_1 -e "flow=my_node_red_flow.json" --ask-vault-pass**
- ○ Note: replace my_group_1 with your group name
- ○ Note: replace my_node_red_flow.json with your Node-RED flow name that you have copied into **/home/admin/pacedge-fmgr/repos/private/nodered-flows/** folder earlier
- ○ Note: when executing command you will be asked for your vault password, please enter it.

- At this point, please go to your target device, refresh the browser and verify that Node-RED flow has been deployed. Note, this operation will replace your existing Node-RED flows with the new flow.

4.6.6 Group Manager Licensing

Group Manager feature requires a license, which limits how many target PACEdge devices can be updated. At the top of the PACEdge landing page the message will indicate how many target devices can be managed.

Figure 67: Group Manager status message, licensed for 10 devices

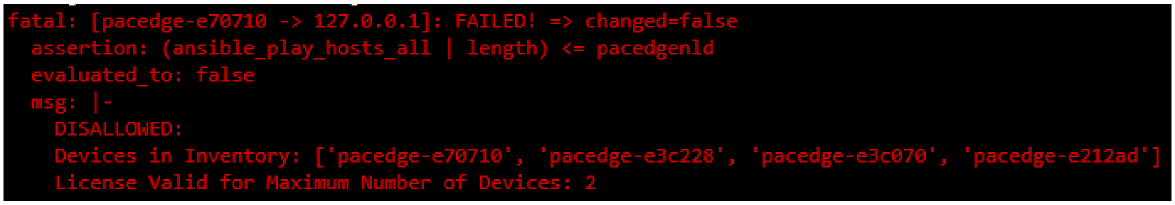


License Validated. MAC: 00:20:ce:e3:c3:76 -- Group Manager license valid: 10 devices

PACEdge software will periodically verify the number of devices configured in the Group Management lists and will compare that number to the number of devices that have been licensed.

If number of licensed devices is exceeded, the warning in red will be shown. To resolve the license violation, reduce the number of devices that are configured for group management or purchase additional licenses.

Figure 68: Error message indicating Group Manager license violation.



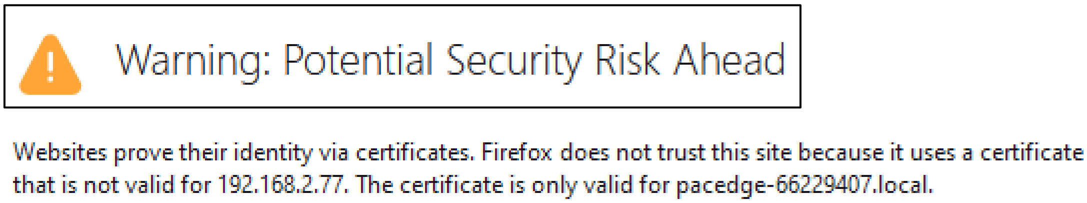
```
fatal: [pacedge-e70710 -> 127.0.0.1]: FAILED! => changed=false
  assertion: (ansible_play_hosts_all | length) <= pacedgenld
  evaluated_to: false
  msg: |-
    DISALLOWED:
    Devices in Inventory: ['pacedge-e70710', 'pacedge-e3c228', 'pacedge-e3c070', 'pacedge-e212ad']
    License Valid for Maximum Number of Devices: 2
```

Note: The number of devices in the management group verification is done once an hour. If you modify the device group configuration file, those changes will not be reflected immediately. To see the latest status, reboot the PACEdge device.

4.7 Certificate Management Utility

As of PACEdge v2.3, Certificate Management Utility has been added. The purpose of these certificates is not only to encrypt communication and prevent Man-In-The-Middle attacks, but also to ensure that your internet browser is indeed connecting to a PACEdge device and not to an attacker's device. When configured and used properly, a security warning message, as the one below, will longer be displayed.

Figure 69 Browser Security warning message example



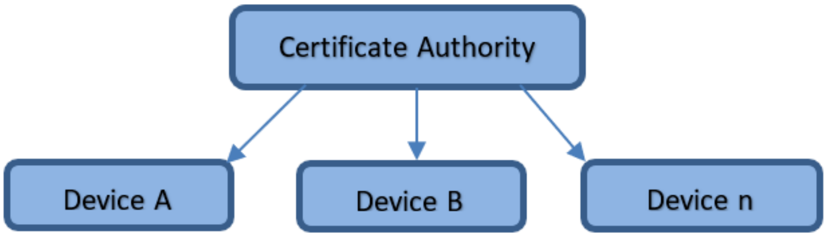
Warning: Potential Security Risk Ahead

Websites prove their identity via certificates. Firefox does not trust this site because it uses a certificate that is not valid for 192.168.2.77. The certificate is only valid for pacedge-66229407.local.

4.7.1 PKI and its use in PACEdge

Public Key Infrastructure (PKI) works along with Certificate Authority (CA), which signs certificates specific to devices. When the CA certificate is added to the trusted CA list in the broswer, all devices with certificates signed by that CA are deemed trusted. This is useful because each device has a unique certificate, which will change if device's hostname changes.

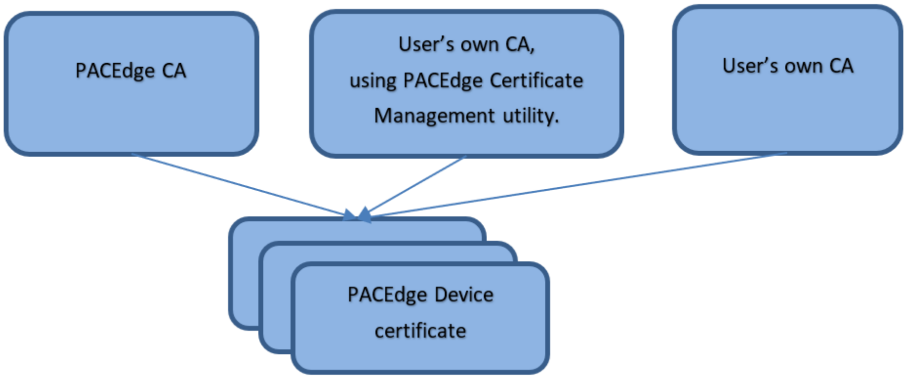
Figure 70: Figure 71 Certificate Authority trust chain



Note: To take advantage of the certificate when using browser, instead of using IP address, use the host name of the device, with .local appended. For example, "https://<hostname>.local"

There are multiple usage models of how the certificates can be used with the PACEdge devices:

Figure 71: Certificate Usage Options


PACEdge CA
User's own CA, using PACEdge Certificate Management utility.
User's own CA
PACEdge Device certificate

PACEdge Factory CA

Starting with version v2.3.0, all PACEdge devices will be shipped with device certificate that is signed by the PACEdge CA. Furthermore, the PACEdge CA Certificate will be available for download via the PACEdge certificate management utility or from the Emerson Customer Center website. Once this CA is added to the browser's trusted CA list, all PACEdge devices can be accessed using the device's hostname insuring device identity.

Note: If a pre-installed certificate on the device is lost, or a factory default recovery is exercised, this certificate will be lost. Emerson's second-level technical support can re-issue certificates, but users are recommended to isseu and manage their own CA certificates.

User's Own CA (using certificate management utility)

With the PACEdge Certificate Management utility, users can create their own CA and issue certificates for other PACEdge devices. By creating own CA users can insure that only their own PACEdge devices are trusted and is more secure than relying on the PACEdge factory CA. Instructions on how to setup own CA and issue own certificates is documented below.

User's Own CA (using existing infrastructure)

Some users will already have an existing PKI and will be able to issue certificates. In such cases, the PACEdge certificate management utility will allow users to upload those certificates onto a PACEdge device.

4.7.2 Services provided by Certificate Management

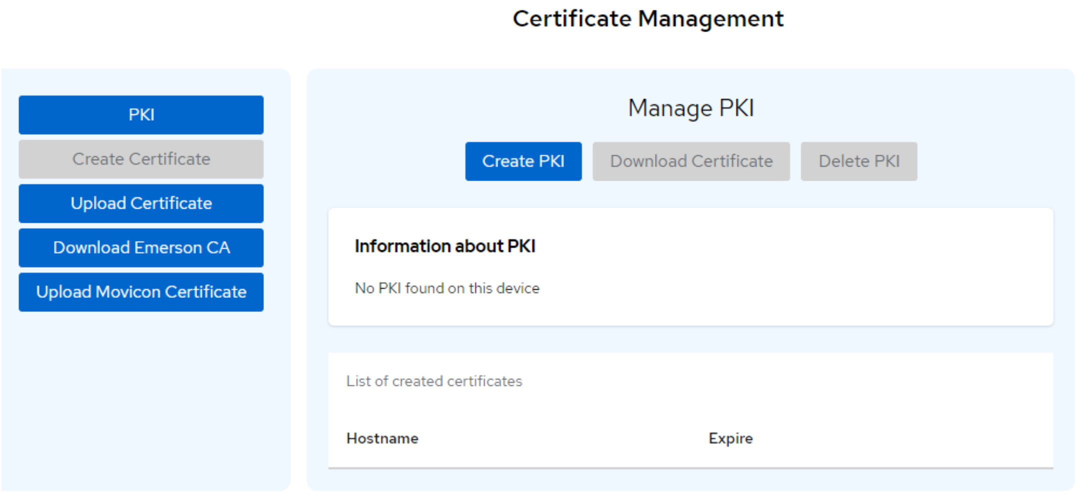
To perform certificate management functions the user needs to have administrative access rights. Administrative access can be requested by pressing the **Limited Access** button at the top of the Cockpit screen.

Figure 72: Limited Access button

The following services are provided by certificate management:

- Enable users to create their own PKI (CA).
- Use own CA to issue certificates for this and other devices and download a CA certificate (which should be installed in the browser).
- Upload previously created certificates (either by PACEdge certificate management or by other services) to this PACEdge device.
- Download PACEdge's own CA default certificate (used to sign PACEdge device certificates during production), which should be installed in the browser.
- Upload Movicon certificates, such as OPCUA certificates that come with OPCUA Servers.

Figure 73 Certificate management utility view



Certificate Management

Manage PKI

Information about PKI

No PKI found on this device

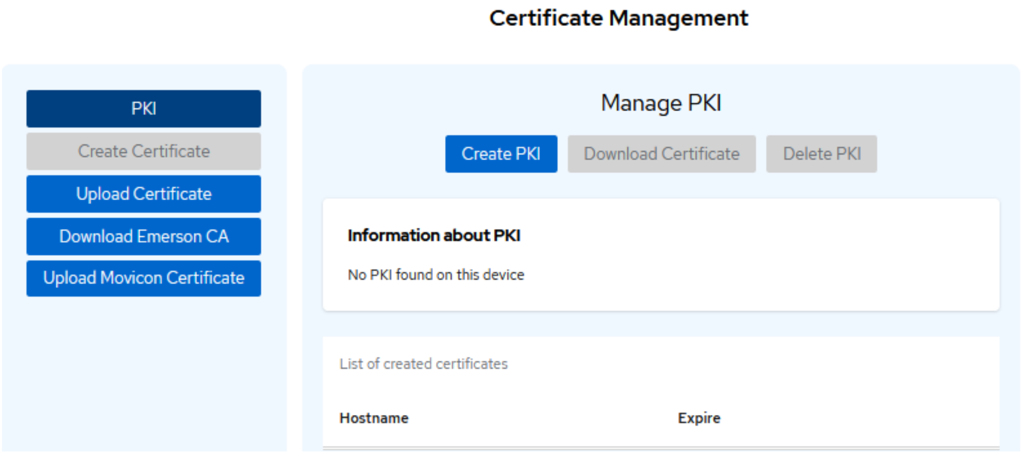
List of created certificates

Hostname	Expire

PKI tab

When accessing **PKI** tab, system will search for already existing PKI and, if found, will display details of it. Per default PACEdge units do not have own PKI, resulting in the screen below:

Figure 74 PKI tab view with no existing PKI



Certificate Management

Manage PKI

Information about PKI

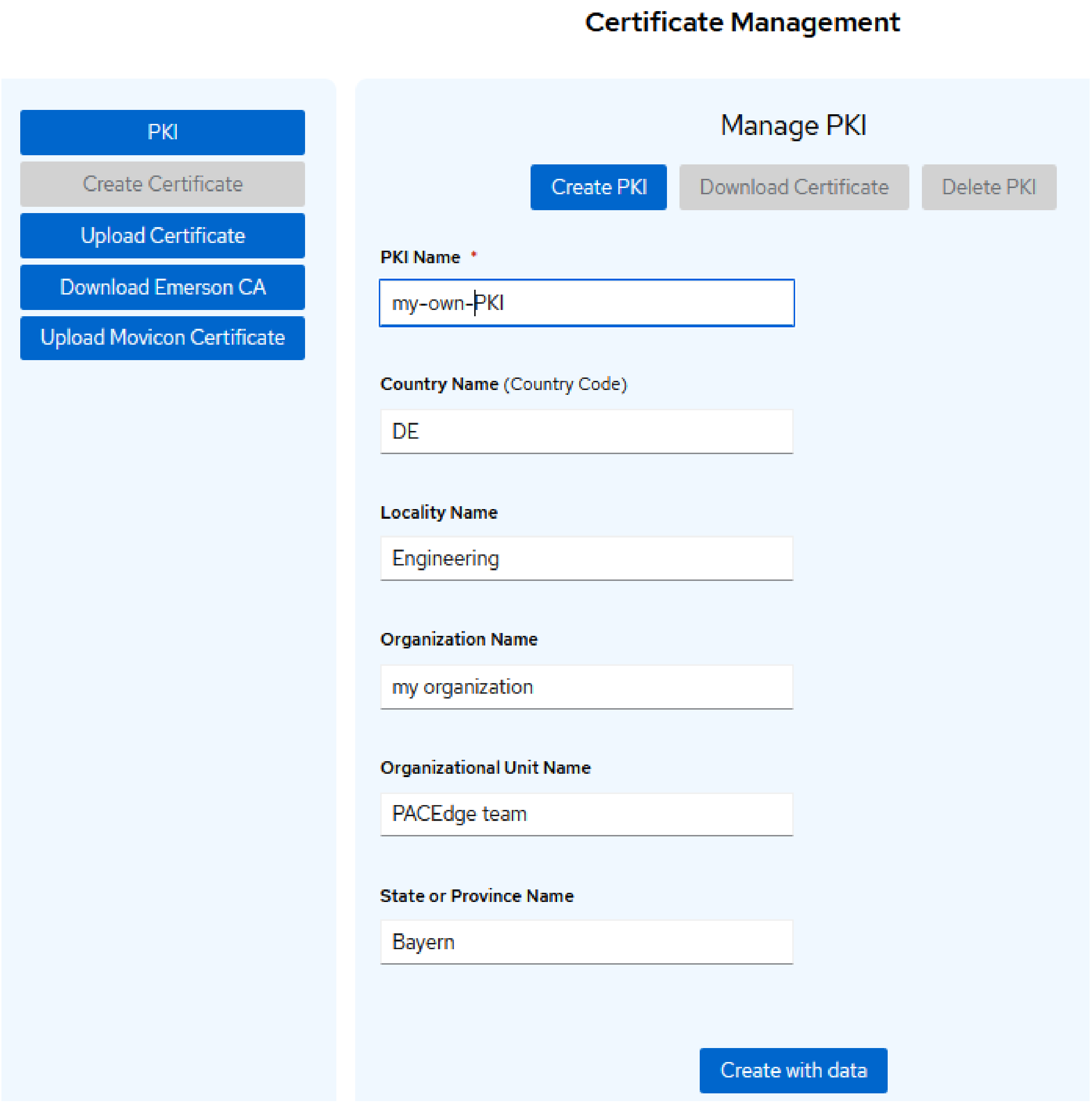
No PKI found on this device

List of created certificates

Hostname	Expire

In this tab, the user has the option to click on the **Create PKI** button, then fill in required information, such as:

Figure 75 Fill-in information to create own PKI

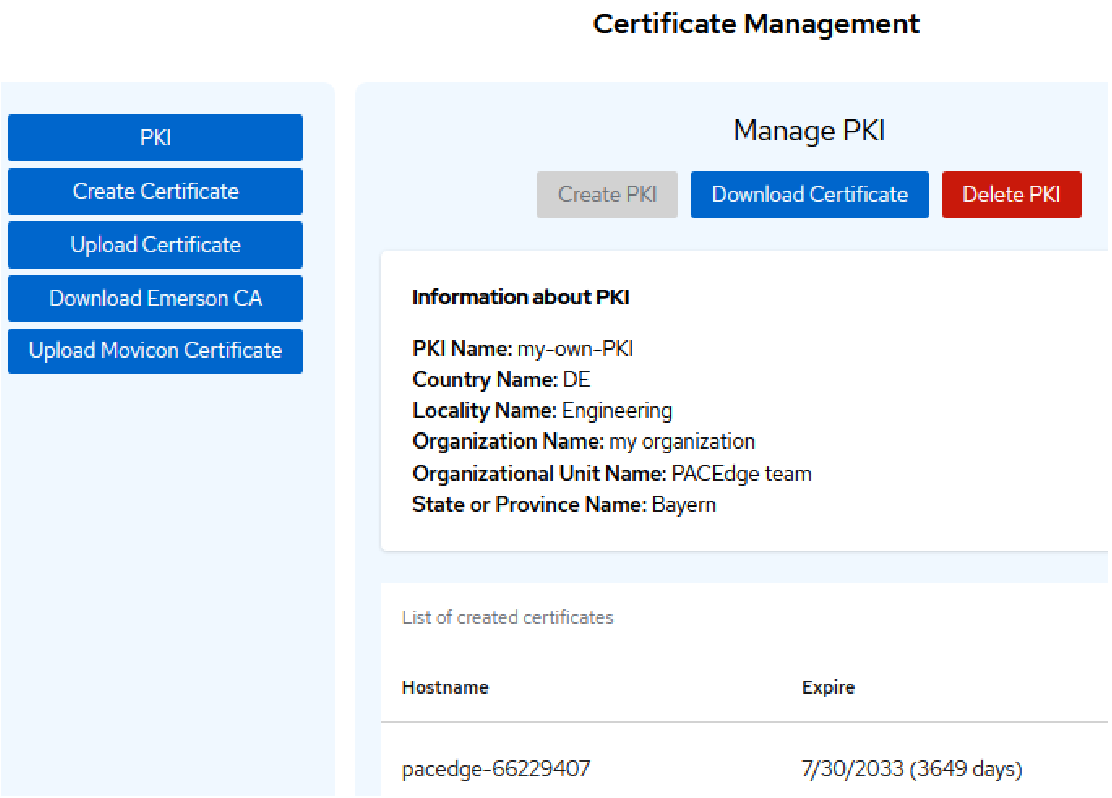


PKI Name *	my-own-PKI
Country Name (Country Code)	DE
Locality Name	Engineering
Organization Name	my organization
Organizational Unit Name	PACEdge team
State or Province Name	Bayern

Once the **Create with data** button has been clicked, the required PKI will be created, along with the CA certificate and device certificate. Since the device's certificate is being replaced, the PKI creation process will result in connection loss to the PACEdge device, which requires a restart to refresh the browser.

Now that your own PKI has been established, you will see following screen:

Figure 76: Enabled View Once its PKI has Been Created



Certificate Management

Manage PKI

Information about PKI

PKI Name: my-own-PKI
Country Name: DE
Locality Name: Engineering
Organization Name: my organization
Organizational Unit Name: PACEdge team
State or Province Name: Bayern

List of created certificates

Hostname	Expire
pacedge-66229407	7/30/2033 (3649 days)

The user can now use the **Download Certificate** button, which enables the user to download and then add the certificate to the browser. Going forward, all device certificates will be trusted automatically.

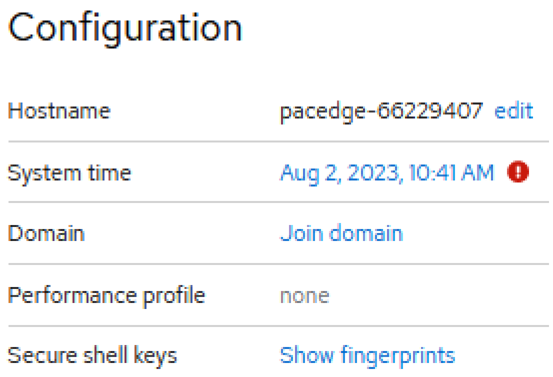
The user can also **Delete PKI**, along with the CA certificates.

Once completed, the user can go to **Create Certificate** tab and create certificates for other PACEdge devices as needed.

Create Certificate tab

In this tab, the user has the option to create one or more device certificates. For successful certificate creation, it is critical to provide an accurate device hostname. The PACEdge device hostname can be found by using Cockpit->Overview in Configuration tab. Note: At this point, the user can also change the hostname. If changing the hostname, please make sure that it is a unique name. It should also be noted that the associated certificate that was issued using the old hostname, will no longer be valid. Additionally, the user will need Administrative access rights to change the hostname.

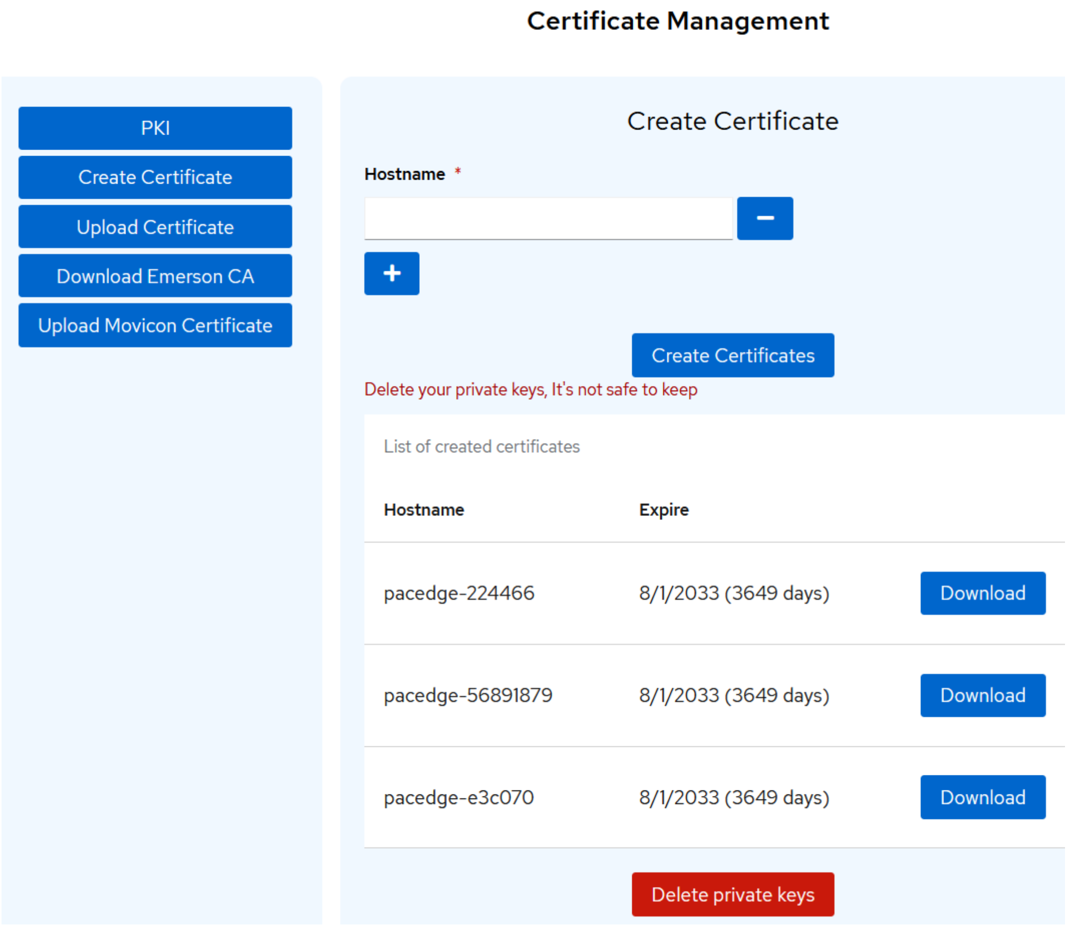
Figure 77 PACEdge Hostname in Cockpit



Configuration

Hostname	pacedge-66229407 edit
System time	Aug 2, 2023, 10:41 AM
Domain	Join domain
Performance profile	none
Secure shell keys	Show fingerprints

Figure 78 View when creating device certificates



Delete your private keys, It's not safe to keep

List of created certificates

Hostname	Expire	
pacedge-224466	8/1/2033 (3649 days)	Download
pacedge-56891879	8/1/2033 (3649 days)	Download
pacedge-e3c070	8/1/2033 (3649 days)	Download

Once the required certificates and associated keys have been created, they can be downloaded by pressing **Download** button. Next, these certificate/key files can be uploaded to required PACEdge device using the **Upload Certificate** tab.

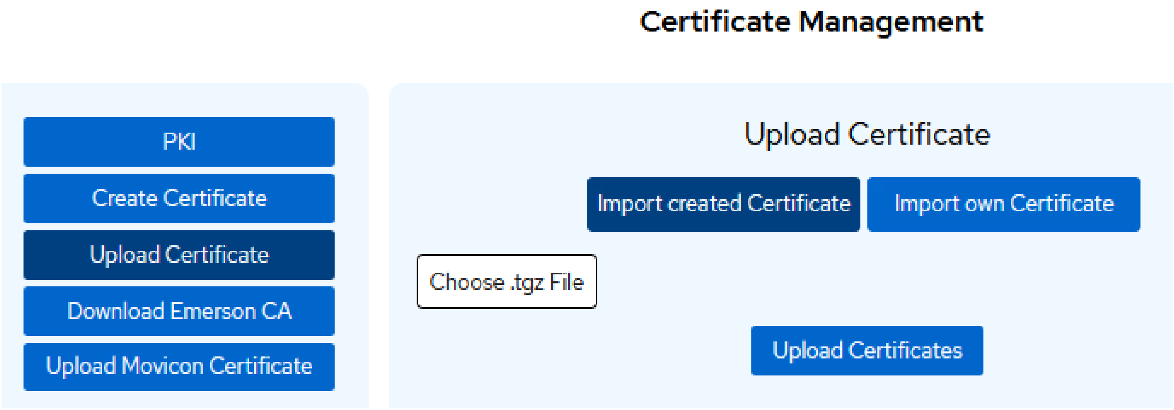
Note: Creating a certificate for the device will also create a key for it. Once the certificate and the key have been transferred to the intended device, storing their keys is not required and not desired (for security purposes.). Therefore, it is recommended to delete the key by pressing **Delete private keys** button.

Upload Certificate tab

Using the Upload Certificate tab, the user can upload certificate and private key files onto a PACEdge device. The following two main use cases are supported:

1. The User has created his own PKI using Certificate Management utility on another PACEdge device and generated a certificate and private key for this particular PACEdge device, which now need to be uploaded. In this case certificate and private key is packaged into .tgz archive. When clicking on **Choose .tgz File** button, the user is given the opportunity to select the required archive file on his computer. Clicking on **Upload Certificates** completes this operation.
2. The user has his own PKI and can generate certificates and private keys. Clicking on **Import own Certificate** button and then on **Choose .key File** and **Choose .crt File** buttons will allow the user to select the required files on his computer. Clicking on **Upload Certificates** completes this operation.

Figure 79 Upload Certificate tab view



Note: The user needs to have administrative access privileges to use these functions.

Note: Uploading the certificate will result in connection loss to PACEdge device, requiring the user to refresh the browser.

Download Emerson CA tab

This tab provides detailsinformation about the PACEdge Root CA , which signs the CAs used to sign the certificates for the units. These certificates come pre-installed. By clicking the **Download** button, the user can download the CA certificate onto their user’s computerand then install it onto their browser.

Upload Movicon Certificate tab

On this tab, users can upload the certificates necessary for certain field bus communications to function properly. One common example is the OPC-UA interface, which may require that the OPC-UA Client has an OPC-UA Server certificate to work. If using Movicon Connext, the server certificate can be uploaded to Connext by using this tab. To upload, click on the **Choose .der File** button, select the required file on your computer and then click on **the Upload Certificate** button.

4.7.3 Adding CA Certificates to Browsers

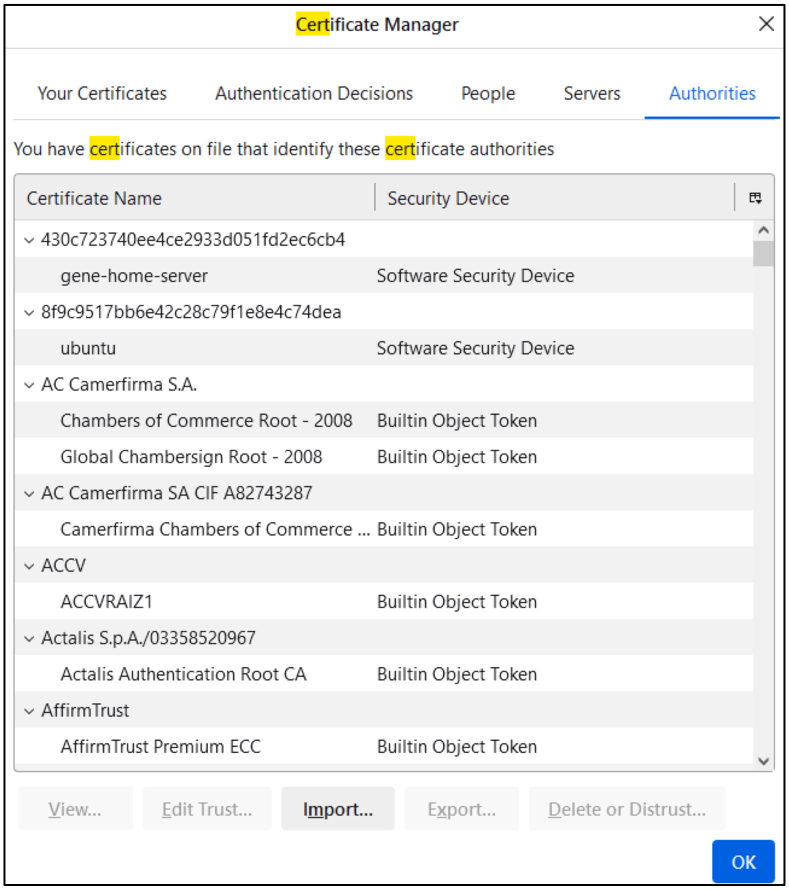
To be able to access PACEdge devices without a security warning, a new trusted authority has to be added to the browser

Firefox

To add a CA certificate to the Firefox browser:

1. Click on the menu icon button ≡ in the top- right corner and then click **Settings**.
2. Search for **cert**.
3. Click on **View Certificates** and go to the **Authorities** tab and click **Import**.
4. Choose the required certificate file on your computer.
5. Select **Trust for Websites** and click **OK**.

Figure 80 Firefox Certificate Manager view



You have certificates on file that identify these certificate authorities

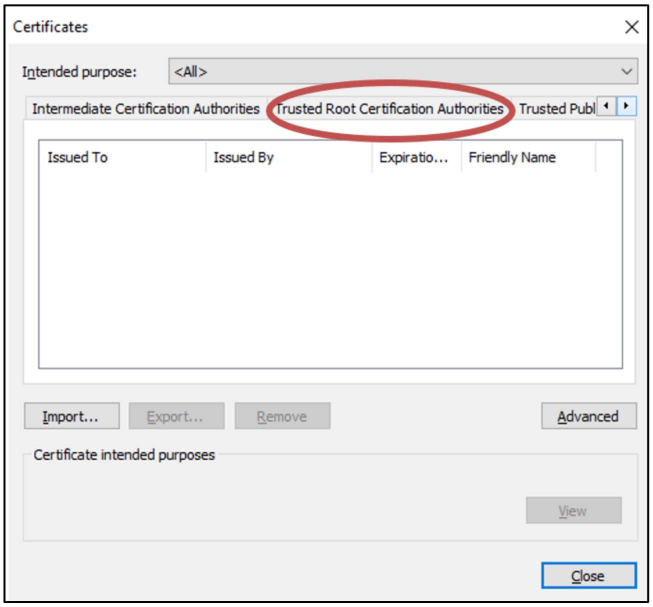
Certificate Name	Security Device
430c723740ee4ce2933d051fd2ec6cb4	
gene-home-server	Software Security Device
8f9c9517bb6e42c28c79f1e8e4c74dea	
ubuntu	Software Security Device
AC Camerfirma S.A.	
Chambers of Commerce Root - 2008	Builtin Object Token
Global Chambersign Root - 2008	Builtin Object Token
AC Camerfirma SA CIF A82743287	
Camerfirma Chambers of Commerce ...	Builtin Object Token
ACCV	
ACCVRAIZ1	Builtin Object Token
Actalis S.p.A./03358520967	
Actalis Authentication Root CA	Builtin Object Token
AffirmTrust	
AffirmTrust Premium ECC	Builtin Object Token

Chrome

To add a CA certificate to the Chrome browser:

1. In the top-right corner click on the three vertical dots button ⋮ abd select **Settings**
2. Click on **Privacy and Security**, then **Security**, **Manage device certificates Trusted Root Certification Authorities**, and finally **Import**
3. In the wizard that opens, click **Next**- and choose the Root Certificate file on your computer. Click **Next, Next**, and finally **Finish**.

Figure 81: Trusted Root Certification Authorities



Section 5: PACEdge Usage Examples

5.1 Node-RED Getting Started

5.1.1 Example Description

This example shows how to create a basic Node-RED flow that generates sine values and displays them via Node-RED Dashboard

5.1.2 Prerequisites

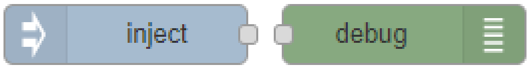
None

5.1.3 Example Walkthrough

Getting Started with Node-RED

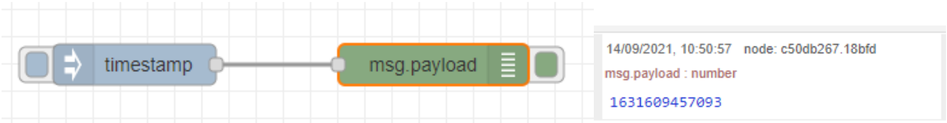
1. Open the Node-RED application from the list of Nodes on the left side.
2. Drag and drop the **inject** node and **debug** node into the middle of the page (flow editor).

Figure 82:Inject Node and Debug Node



3. Connect **inject** node to **debug** node by dragging the line between grey connection points.
4. Open the debug window by clicking on the bug symbol 🐞 in the upper-right corner.
5. Click on the red **deploy** button in the upper-right corner.
6. At this point, the flow has been deployed and is active. Clicking on the square icon in front of the **inject** node will generate a time stamp and will send it to **debug** node, which will display it in debug window, as shown below:

Figure 83: Timestamp



```
14/09/2021, 10:50:57   node: c50db267.18bfd
msg.payload : number
1631609457093
```

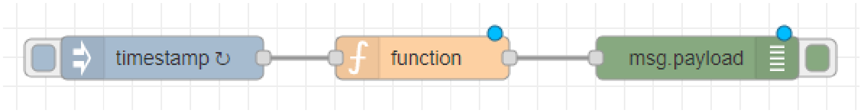
7. Double-click the **inject** node, and you will get a configuration dialogue where you can change the functionality of the **inject** node. For instance, you can send a string or JSON structure instead of sending the time stamp. For example, let us configure a repeat mode by selecting **interval** and setting it to every 1 sec. Click on **Done**.

8. Click on **Deploy** after making changes so that they take effect.
9. In the debug window, you will get a new timestamp every second

Adding a Function Node

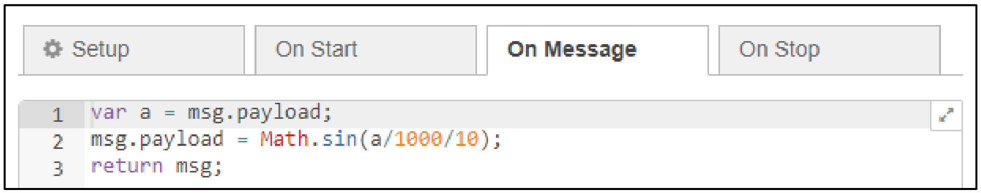
1. Drag and drop a function node and put it in between the **inject** and **debug** nodes. Connect all three nodes in a series. The message the inject node sends will go to the function node and then **debug** node.

Figure 84: Place the Function Node Between the Timestamp and Debug Nodes



2. Double-click on the function node and write the following JavaScript code:

Figure 85: JavaScript Code



```
var a = msg.payload;
msg.payload = Math.sin(a/1000/10);
return msg;
```

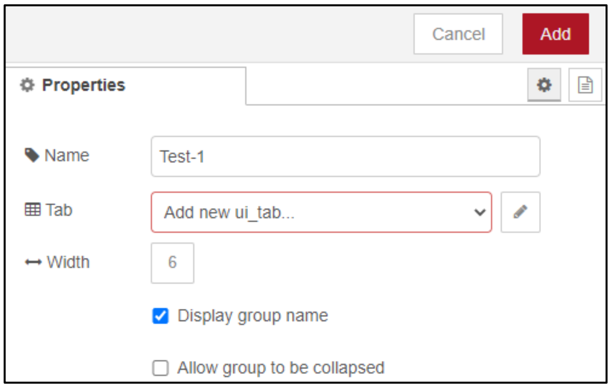
This code will first assign the received timestamp to **variable a**, then scale it down and calculate its Sine value. The result will be sent out to the next node.

3. Once deployed, you will see a value slowly changing between -1 and 1 in the debug window.

Adding Node-RED Dashboard

1. Drag and drop a **chart** node and connect it to the function node.
2. Double-click the chart node and make the following changes:
   a. In the Group, select **Add new ui_group** and click on the pen symbol to edit.
   b. In the Properties Window, provide a name (such as “Test-1”) in the Name field, and in the Tab field, select **Add new ui_tab** and click on the pen symbol to edit.

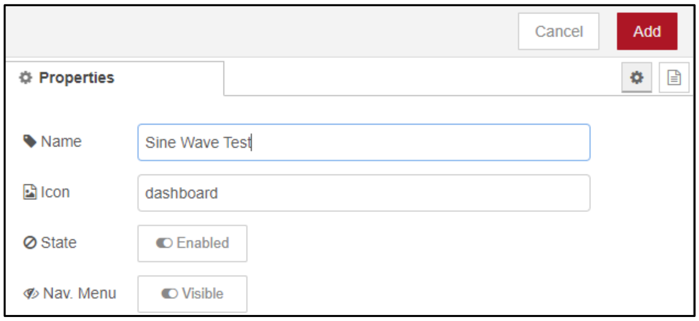
Figure 86: Properties



Name: Test-1
Tab: Add new ui_tab...
Width: 6
☑ Display group name
☐ Allow group to be collapsed

   c. In the new dialogue, give a name, such as **Sine Wave Test,** and click on the red **Add** button.

Figure 87: Sine Wave Test

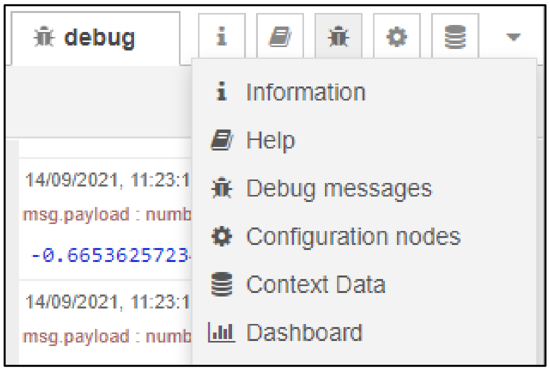


Name: Sine Wave Test
Icon: dashboard
State: Enabled
Nav. Menu: Visible

   d. Click on **Add** button again
3. Back in the Edit chart dialogue, in X-axis, select the **last 1 minute** so that the data for the previous 1 minute is visible.

4. Give the chart a name and click **Done.**
5. Click on **Deploy** to save the changes.
6. Open the Dashboard window by clicking on the drop-down symbol in the upper-right corner and select **Dashboard.**

Figure 88: Dashboard Menu

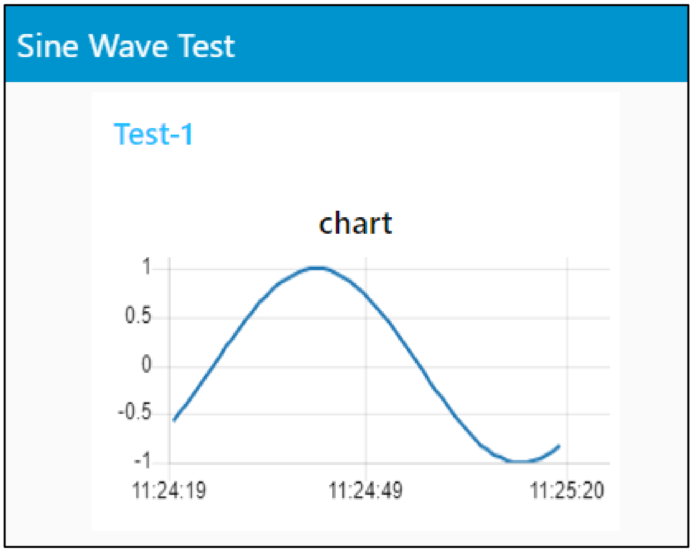


14/09/2021, 11:23:1
msg.payload : numb
-0.6653625723
14/09/2021, 11:23:1
msg.payload : numb

a. Click on the square with the arrow ⧉ symbol.

b. A new browser tab will open, and a sine wave will display:

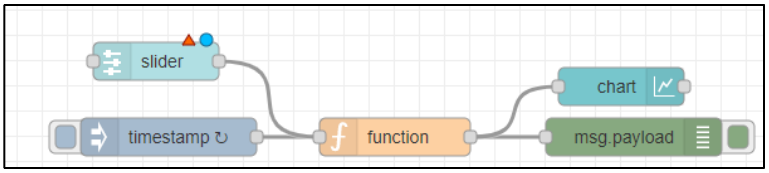
Figure 89: Sine Wave Test



Adding Control to the Node-RED Dashboard

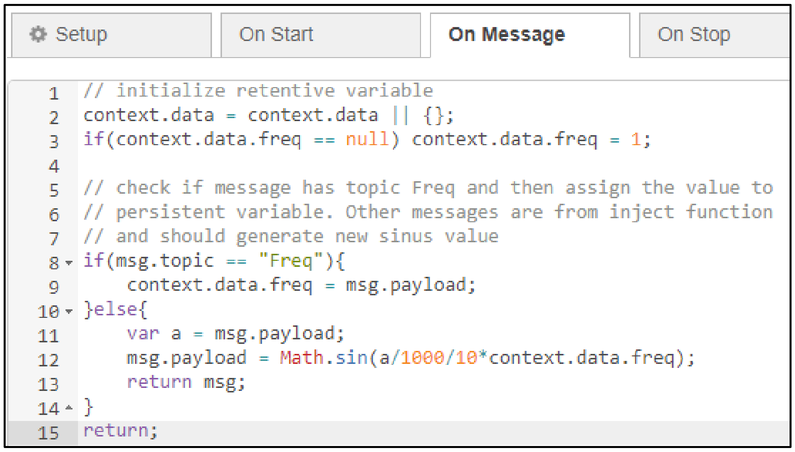
1. Add a slider control that can increase or decrease the sine wave frequency. To do so, drag and drop a **slider** node and connect it so that the output from the slider goes into the **function** node (in parallel with the **inject** node).

Figure 90: Slider Node



2. Double-click the **slider** node and select the group created earlier [Sine Wave Test]Test-1.
3. Note the Range. Set it to 1 to 10 in steps of 1.
4. In the Topic field, from the drop-down list, select String and then type in: **Freq.**
5. Click on **Done** to close the **slider** node.
6. Now we have a problem: two different messages are coming into the **function** node. We need to change the code to check for the message topic and take action accordingly. Change the code inside the function node as follows.

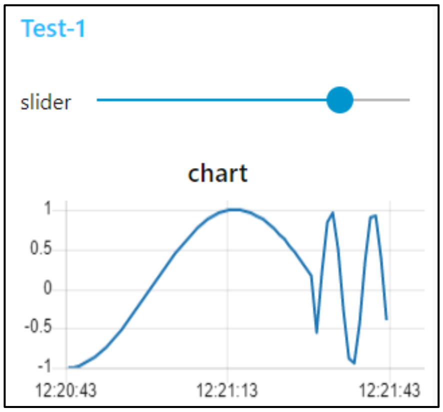
Figure 91: Function Node's Code Change



```
// initialize retentive variable
context.data = context.data || {};
if(context.data.freq == null) context.data.freq = 1;

// check if message has topic Freq and then assign the value to
// persistent variable. Other messages are from inject function
// and should generate new sinus value
if(msg.topic == "Freq"){
    context.data.freq = msg.payload;
}else{
    var a = msg.payload;
    msg.payload = Math.sin(a/1000/10*context.data.freq);
    return msg;
}
return;
```

7. Once deployed, check the dashboard and use the slider to change the multiplier inside the Sinus function. As a result, you will get a sine wave of different frequency

Figure 92: Changed Sinus Value



5.2 Using the InfluxDB Database

Starting with PACEdge version 2.2, new Emerson-specific Node-RED nodes have been introduced that simplify access to the InfluxDB database. These nodes are not a replacement for the generic InfluxDB nodes and have only a subset of functionality, but they are pre-configured and easier to use for basic tasks. The sections below will explain how to use both Emerson-customized InfluxDB nodes and standard InfluxDB nodes.

Note: to get Emerson customized InfluxDB nodes added to the Node-RED node palette, import an example **PacedgeInfluxDB**, as described below.

5.2.1 Using Emerson Customized InfluxDB Nodes

Example Description

This example shows how to access the InfluxDB database within the Node-RED flow using Emerson-customized Node-RED nodes. It shows how to install the nodes, create a new database, write to it, and read from it.

Prerequisites

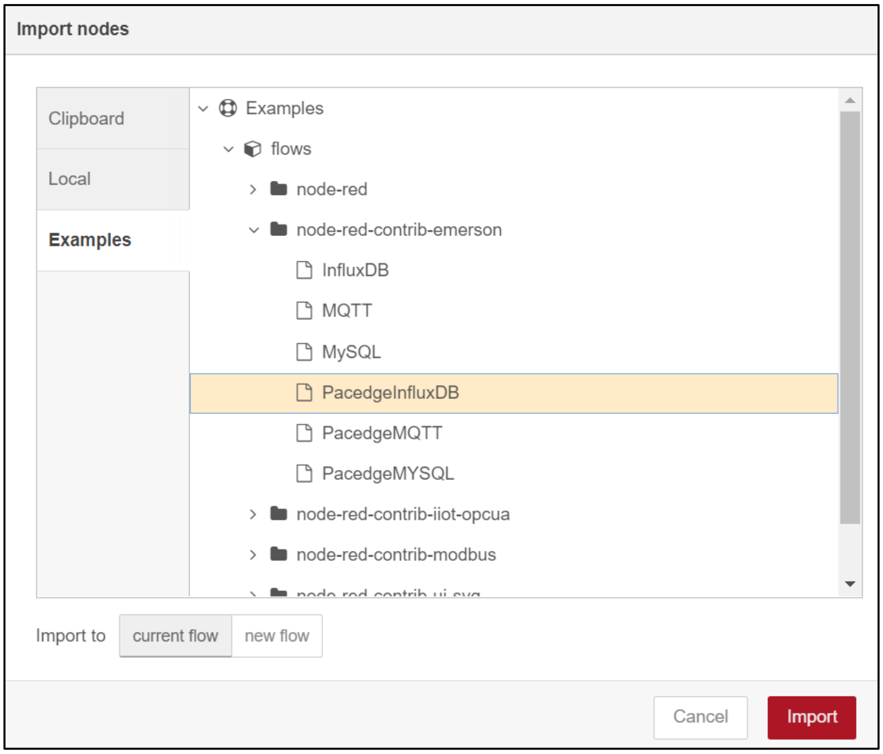
PACEdge version 2.2 or newer.

If new to Node-RED, completion of the 5.1 is highly recommended, as it shows some Node-RED usage basics and tips

Example Walkthrough

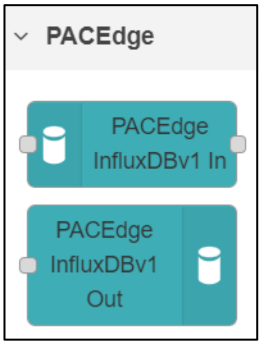
1. In the Node-RED, upper right corner, click on the three-line symbol ☰ and select Import.
2. In the Import Nodes dialog that opens, go to Examples, expand **node-red-contrib-emerson** and, select **PacedgeInfluxDB**, click on the Import button.

Figure 93: Select Import



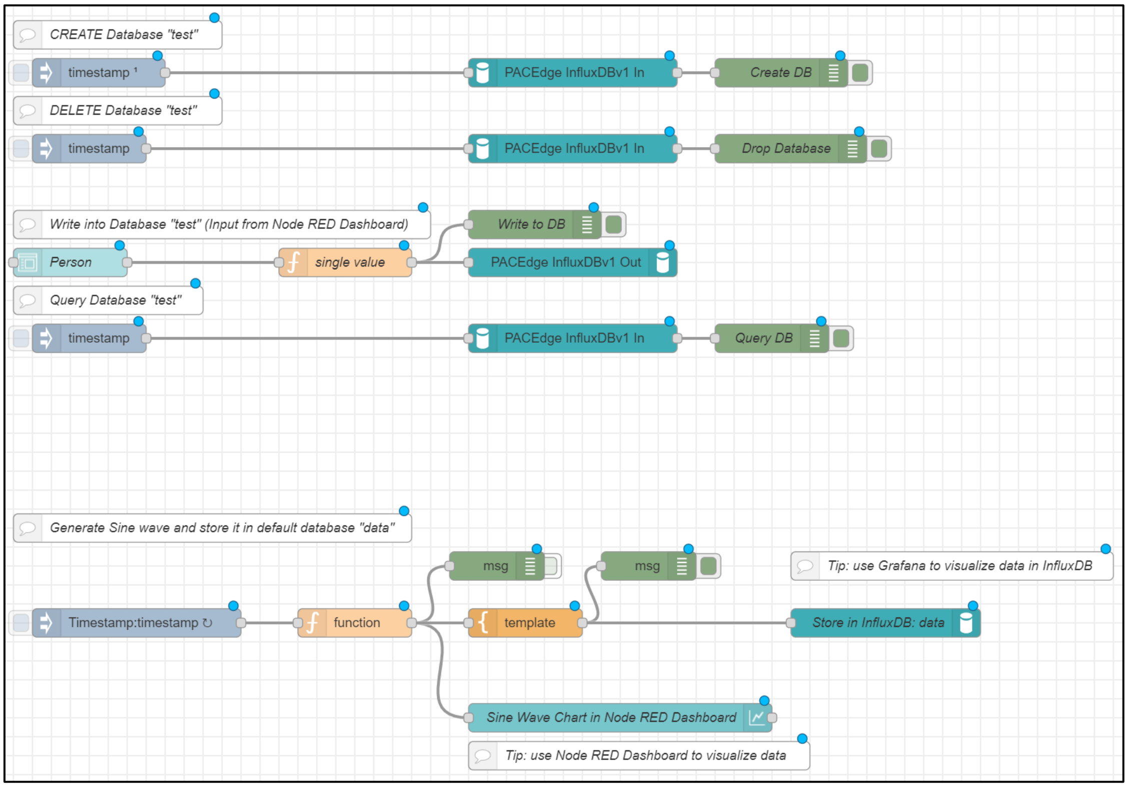
3. Now you have two new nodes in the node pallet on the left side.

Figure 94: Example Nodes



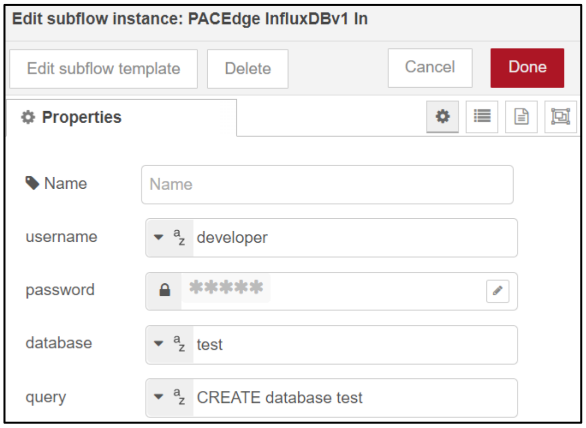
The following is an example flow showing how to write to InfluxDB and read from InfluxDB.

Figure 95: Example flow



4. For each instance of the PACEdge InfluxDB node, configure username and password. You can use **admin**, **developer,** or **service** users.

Figure 96: Configure Users



Edit subflow instance: PACEdge InfluxDBv1 In

Name	Name
username	developer
password	*****
database	test
query	CREATE database test

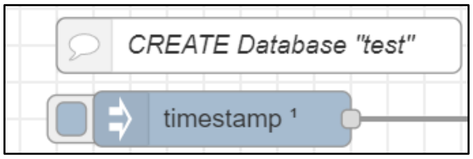
5. Once finished, click on the Deploy button Deploy

6. At this point, the example flow is active. It consists of two parts:

a. The bottom part generates and stores a new value every 200msec, representing the sine wave. Values will be stored in the InfluxDB database called **data**, which is present by default. You can observe it in Node-RED Dashboard and via Grafana.

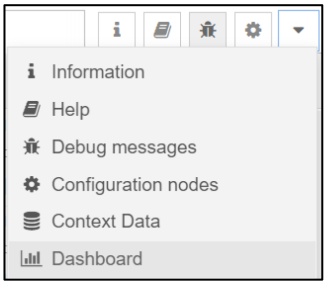
b. The upper part shows how to create a new database in InfluxDB called **test**, write data into it, query the data, and delete the database. Data entry will be done via Node-RED Dashboard, as explained below. Make sure to first create a test database by clicking on the button.

Figure 97: Create Database



7. Open Node-RED Dashboard by clicking on the triangle icon in the upper-right corner, then selecting Dashboard.

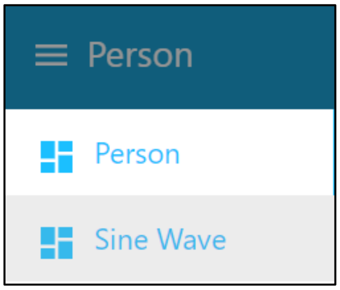
Figure 98: Dashboard



8. In the right-side pane that opens, click on the icon ↗ to open another browser tab that displays Node-RED Dashboard.

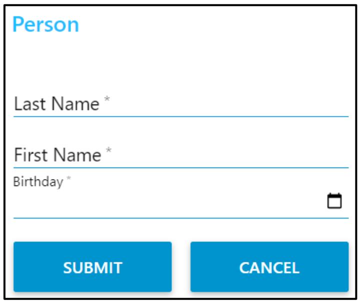
9. In the Node-RED Dashboard, there are two tabs accessible on the upper-left side of the window.

Figure 99: Person Tab



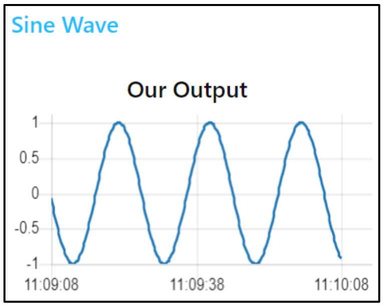
10. The **Person** tab allows the user you to enter data into the InfluxDB database called **test** (upper portion of the Node-RED flow).

Figure 100: Person Tab



11. Click on the **Sine Wave** tab to display the sine wave that is being generated and stored in the InfluxDB database called **data.**

Figure 101: Data



12. Once you enter one or more personal data using the **Person** tab, you can go back to the Node-RED flow browser tab and click on the button to Query database. Note that you must first create a database test; otherwise, you will get Error 404 Not Found**.** Note: In the Node-RED flow tab, please select Debug tab / so you can see query results and errors, if any.

5.2.2 Using Node-RED Standard InfluxDB Nodes

Example Description

This example shows how to access the InfluxDB database within the Node-RED flow using Node-RED standard InfluxDB nodes. It shows how to create a new database, write to it, and read from it.

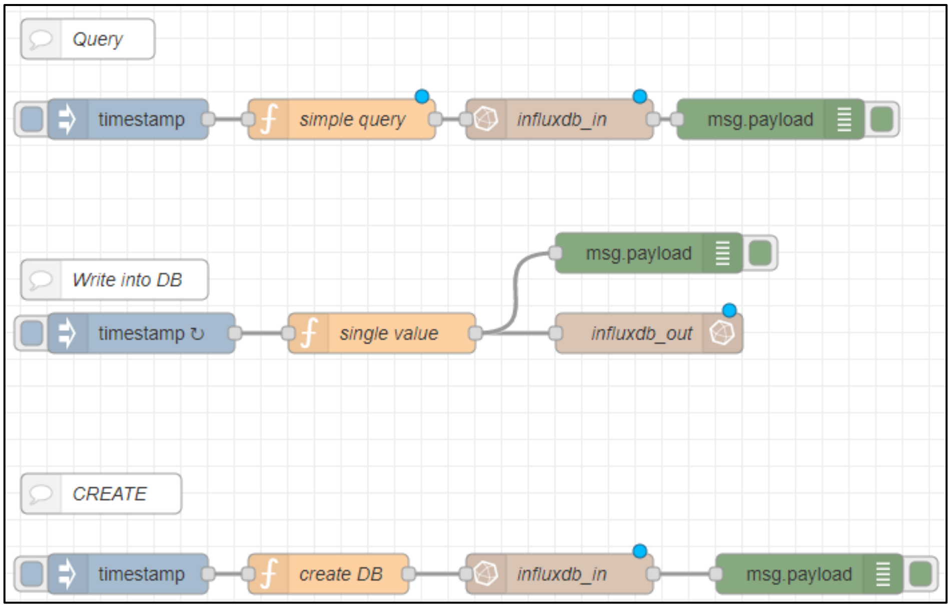
Prerequisites

If new to Node-RED, completion of the 5.1 is highly recommended, as it shows some Node-RED usage basics and tips

Example Walkthrough

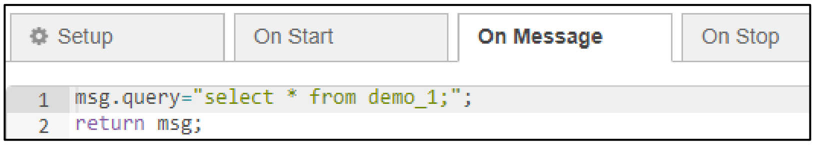
In the Node-RED flow window place **inject**, **function**, **influxdb_in**, **influxdb_out,** and **debug** nodes and interconnect them as shown. Note that the example below also uses **Comment** nodes to make it more readable.

Figure 102: Example Walkthrough



1. Double-click on the function node in the Query section and enter the following code. It will be used to query the database:

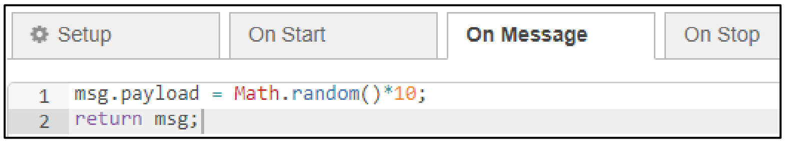
Figure 103: Query Section: Function Node's Code



```
msg.query="select * from demo_1;";
return msg;
```

2. Double-click on the function node in the Write into DB section and enter the following code. It will generate random values and send them to be stored in the database.

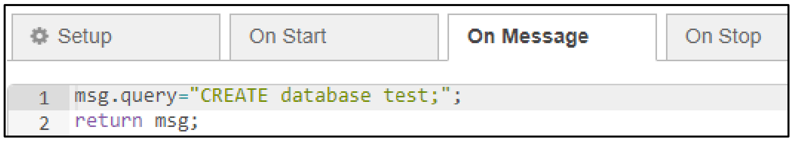
Figure 104: DB Section: Function Node's Code



```
msg.payload = Math.random()*10;
return msg;
```

4. Double-click on the function node in the Create section and enter the following code. It will be used to create a new database within InfluxDB.

Figure 105: Create Section: Function Node's Code



```
msg.query="CREATE database test;";
return msg;
```

5. Now, the first instance of the InfluxDB node (in the Query section) needs to be configured as follows:

   a. In the Server field, select **Add new influxdb** and click on the pen symbol to edit.

   b. In the Name, give the desired name, such as **my_influx**.

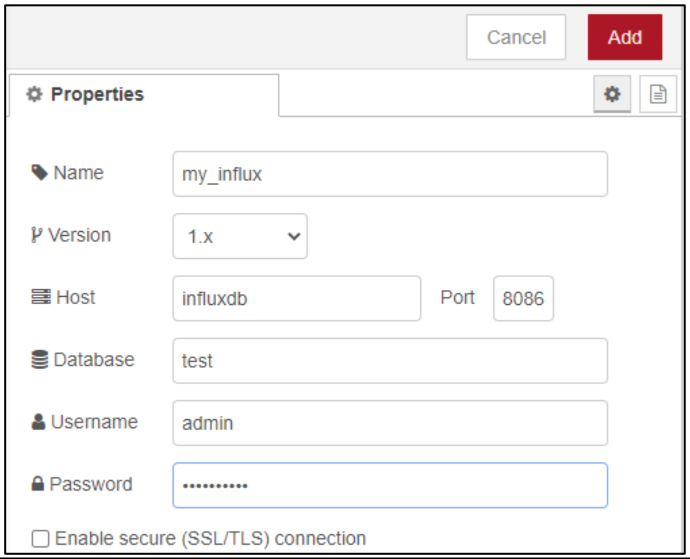
   c. In Host, enter: **influxdb** and keep the port number as 8086.

   d. In the Database, enter the **test**.

   e. In Username, enter either **admin** or **developer** or **service**.

   f. In the Password field, enter the password for the user that was entered above.

Figure 106: User Properties



Name: my_influx

Version: 1.x

Host: influxdb Port: 8086

Database: test

Username: admin

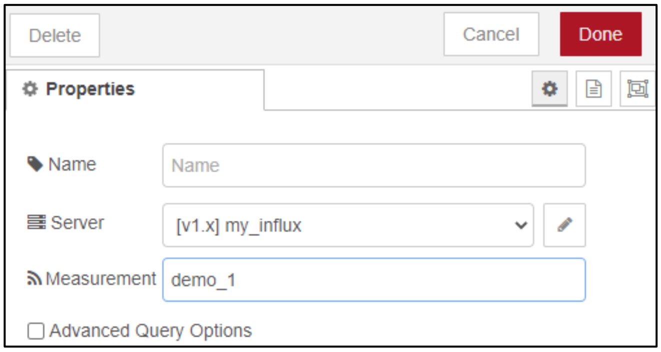
Password: ••••••••••

□ Enable secure (SSL/TLS) connection

   g. Click on **Add**, then on **Done.**

7. Double-click on the influx_out node in the Write into DB section. In the Server drop-down list, you should now see the newly configured database server with the name my_influx; select it since this is an output node that will write data into the database; we also need to provide the name for the Measurement. Please enter the desired name, such as **demo_1**. Click **Done** afterward.

Figure 107: Demo_1



Name: Name

Server: [v1.x] my_influx

Measurement: demo_1

☐ Advanced Query Options

8. In the third **influxdb** node, select **my_influx** in the list of servers and click Done
9. Deploy the newly configured flow.
10. Now you can experiment by triggering actions when clicking on the trigger nodes. You will notice that if you try to read or write to the database, you will get an error stating that the database does not exist. Once you trigger the flow to create a database, you can write to and read from the database.
11. Note that the database is referenced by name in our flow tests. This name is entered in the InfluxDB nodes Database field, but it must also match in the function node create DB. Therefore, if you want to change the database name, you need to change it in the create database flow function node.
12. If you want to modify query parameters, edit the function node.

5.3 Using MySQL Database

Starting with PACEdge version 2.2, new Emerson-specific Node-RED nodes have been introduced that simplify access to MySQL database. These nodes are not a replacement for the generic MySQL nodes and have only a subset of functionality, but they are pre-configured and easier to use for basic tasks. The sections below will explain how to use both Emerson customized MySQL nodes and standard MySQL nodes.

Note: to get Emerson customized MySQL nodes added to the Node-RED node palette, import an example **PacedgeMYSQL**, as described below.

5.3.1 Using Emerson Customized MySQL Nodes

Example Description

This example shows how to access MySQL Database from Node-RED.

Prerequisites

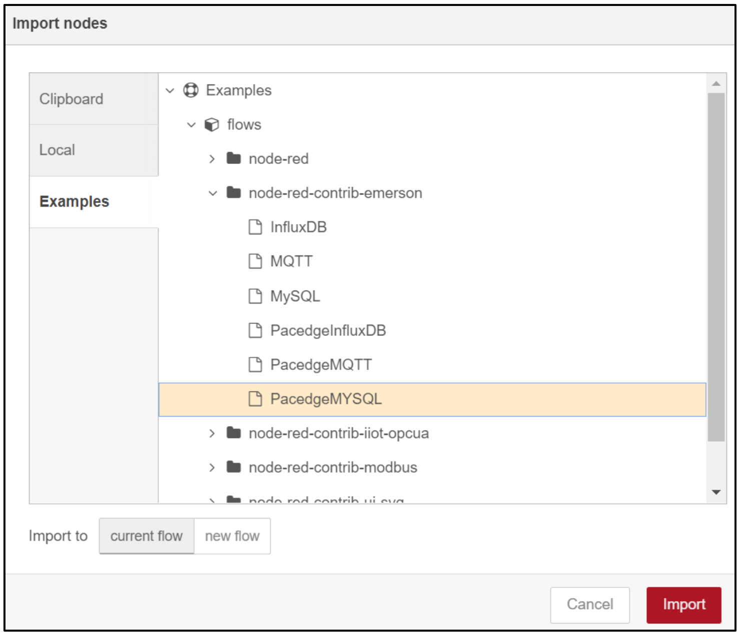
PACEdge version 2.2 or newer.

Completing the 5.1 Getting Started with Node-RED is highly recommended if new to Node-RED, as it shows some Node-RED usage basics and tips.

Example Walkthrough

1. In the Node-RED, upper right corner, click on the three-line symbol ☰ and select Import.
2. In the Import Nodes dialog that opens, go to Examples, expand **node-red-contrib-emerson** and, select **PacedgeMYSQL**, click on the **Import** button.

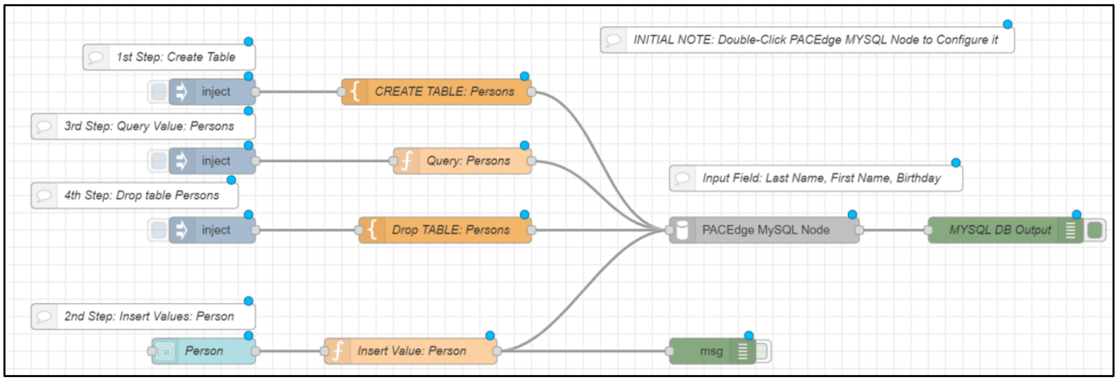
Figure 108: Import Nodes



3. Now you have one node in the node pallet on the left side PACEdge MySQL Node

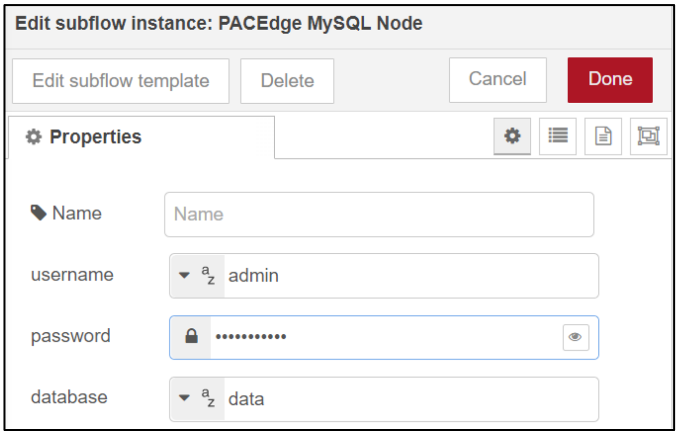
   And an example flow that shows how to write to MySQL and how to read from MySQL

Figure 109: MySQL Example


INITIAL NOTE: Double-Click PACEdge MYSQL Node to Configure it
1st Step: Create Table
inject
CREATE TABLE: Persons
3rd Step: Query Value: Persons
inject
Query: Persons
Input Field: Last Name, First Name, Birthday
4th Step: Drop table Persons
inject
Drop TABLE: Persons
PACEdge MySQL Node
MYSQL DB Output
2nd Step: Insert Values: Person
Person
Insert Value: Person
msg

4. Double click on PACEdge MySQL node and configure username and password. You can use **admin**, **developer,** or **service** users.

Figure 110: PACEdge MySQL node



Name: Name

username: admin

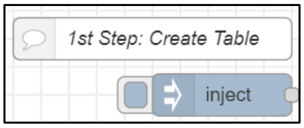
password: •••••••••••

database: data

5. Once finished, click on Deploy button Deploy

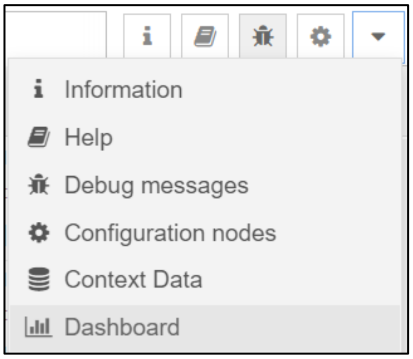
6. At this point, the example flow is active. In the flow, start by creating a new table in the MySQL database. To do so, click on Inject button

Figure 111: Inject Button



7. Try entering some data into the MySQL table. In the example, this is done by using Node-RED Dashboard. To open Dashboard, first, click on the triangle icon in the upper right corner, then select Dashboard

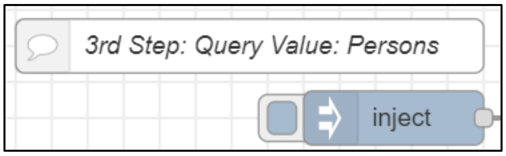
Figure 112: Node-Red Dashboard



8. In the right-side pane that opens, click on the icon ↗ to open another browser tab that displays Node-RED Dashboard

9. In the Dashboard, please enter one or more person details
Go back to the browser tab with Node-RED flow and click on the button to Query the database.

Figure 113: Query Database



Note: In the Node-RED flow tab, please select Debug tab / so you can see query results and errors, if any.

10. Feel free also to try deleting the MySQL table

5.3.2 Using Node-RED Standard MySQL Nodes

Example Description

This example shows how to access MySQL Database from Node-RED using standard MySQL nodes.

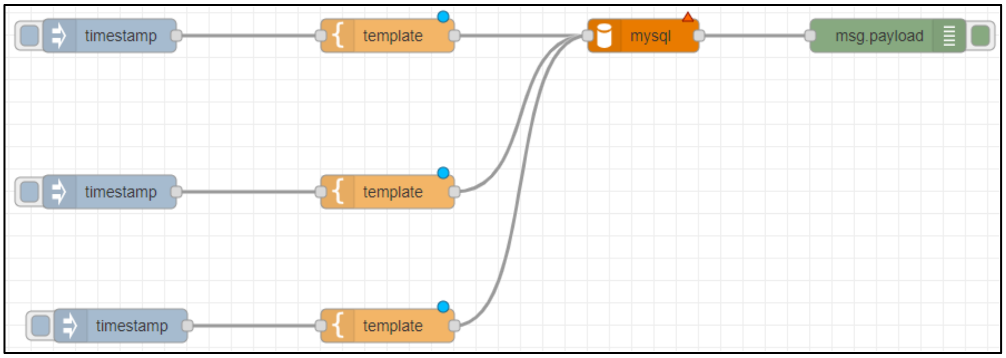
Prerequisites

Reading section 5.1, Getting Started with Node-RED, is highly recommended if new to Node-RED. It shows some Node-RED usage basics and tips.

Example Walkthrough

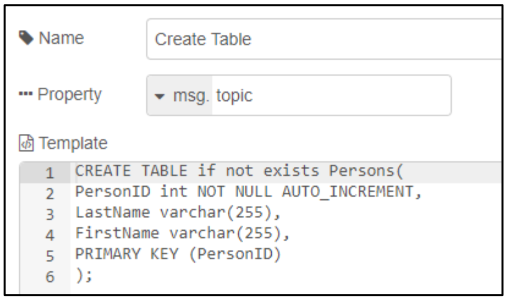
We will use a template note instead of a function node in this example. This is to show that there are alternative ways to implement the same functionality. Drag and drop inject, template, MySQL, and debug nodes and interconnect them as shown. In contrast to InfluxDB nodes, for MySQL, the same node is being used for both writing and querying the database, so it is more efficient to instantiate it only once and just send different commands.

Figure 114: Walkthrough Example



1. Double-click the first **template** node, name it **Create Table** in the Property field, type in msg.topic, and paste the following code into it:

Figure 115: Template Node



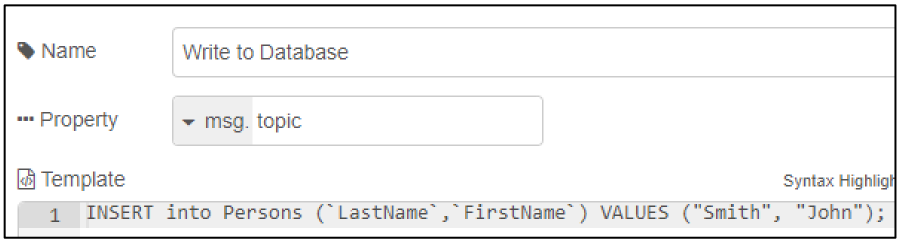
Name: Create Table

Property: msg. topic

```
CREATE TABLE if not exists Persons(
PersonID int NOT NULL AUTO_INCREMENT,
LastName varchar(255),
FirstName varchar(255),
PRIMARY KEY (PersonID)
);
```

3. Double-click the second **template** node, name it **Write to Database**, enter msg.topic, and fill in the following code.

Figure 116: Write to Database Node



Name: Write to Database

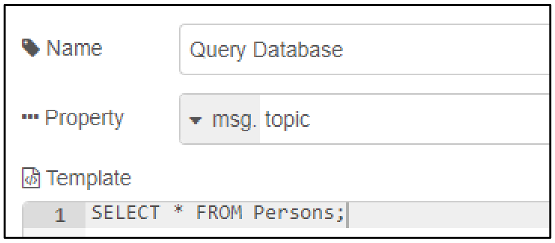
Property: msg. topic

Template

```
INSERT into Persons (`LastName`,`FirstName`) VALUES ("Smith", "John");
```

4. Double-click the third **template** node, name it **Write to Database**, enter msg.topic, and fill in the following code.

Figure 117: Write to Database Node



Name: Query Database

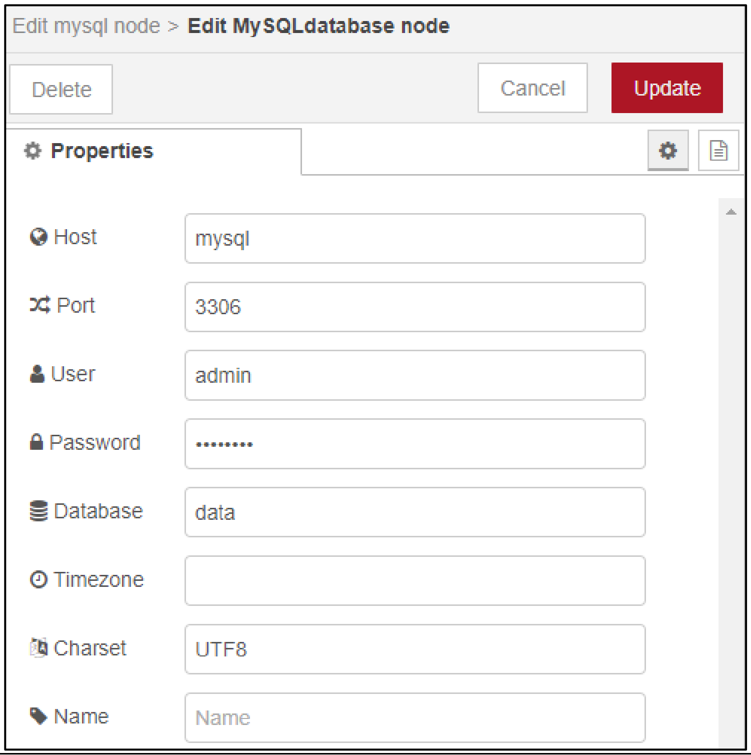
Property: msg. topic

Template

```
SELECT * FROM Persons;
```

5. Double-click the MySQL node in the Database, select Add new MySQL database and click on the pen icon to edit.
6. In the Host field, enter **mysql**. In the User field, enter either **admin** or **developer**. In the Password field, enter the associated password with the user you choose and the Database field value data. Click on **Update** and then on **Done**.

Figure 118: My SQLDatabase Node



Host	mysql
Port	3306
User	admin
Password	••••••••
Database	data
Timezone	
Charset	UTF8
Name	Name

7. Once all the edits are done, click on the **Deploy** button. If the MySQL node is properly configured, you will see a green dot with the label OK at the bottom of it.

8. Click on the first **inject** node to create a new table, then on the second **inject** node to enter data, and on the **third** inject node to read all entries from the table.

5.4 Using MQTT Communication

Starting with PACEdge version 2.2, new Emerson-specific Node-RED nodes have been introduced that simplify MQTT communication. These nodes are not a replacement for the generic MQTT nodes and have only a subset of functionality, but they are pre-configured and easier to use for basic tasks. Sections below will explain how to use both Emerson customized MQTT nodes as well as standard MQTT nodes

Note: to get Emerson customized MQTT nodes added to the Node-RED node palette, import an example **PacedgeMQTT**, as described below.

5.4.1 Using Emerson Customized MQTT Nodes

Example Description

This example shows how to use internal and external MQTT brokers and communication when using Emerson-customized MQTT nodes.

Prerequisites

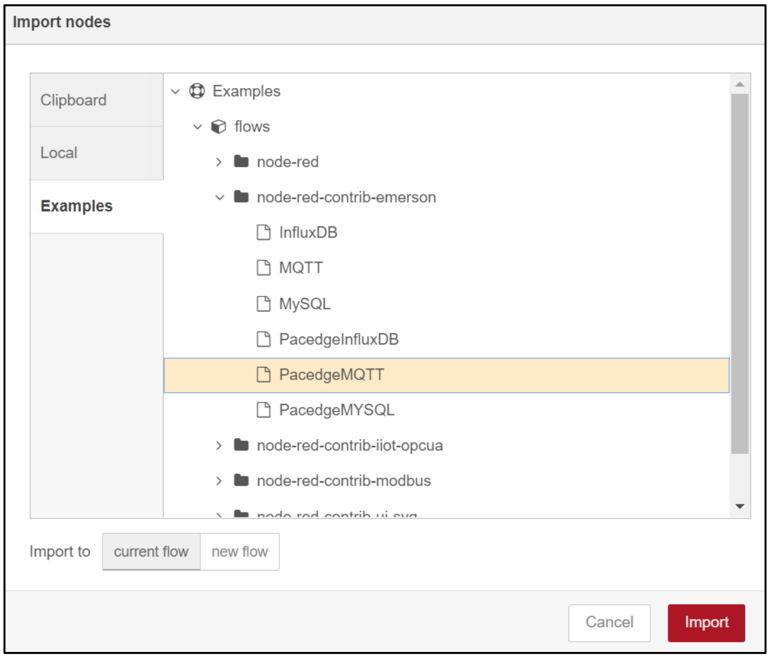
PACEdge version 2.2 or newer.

Completing section 5.1, *Getting Started with Node-RED* example, is highly recommended. It shows some Node-RED usage basics and tips.

Example Walk Through

1. In the Node-RED, upper right corner, click on the three-line symbol ☰ and select Import.
2. In the Import Nodes dialog that opens, go to Examples, expand **node-red-contrib-emerson** and, select **PacedgeMQTT**, click on the Import button.

Figure 119: PACEdge MQTT



Import nodes

- Clipboard
- Local
- **Examples**

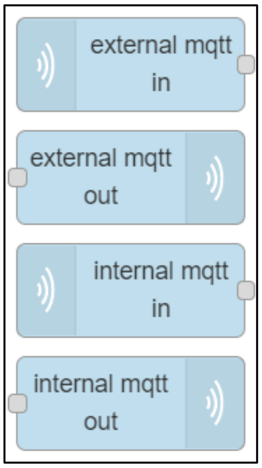
- Examples
  - flows
    - node-red
    - node-red-contrib-emerson
      - InfluxDB
      - MQTT
      - MySQL
      - PacedgeInfluxDB
      - PacedgeMQTT
      - PacedgeMYSQL
    - node-red-contrib-iiot-opcua
    - node-red-contrib-modbus
    - node-red-contrib-ui-svg

Import to current flow new flow

Cancel Import

3. Now you have four new nodes in the node pallet on the left side and an example flow that shows how to publish and subscribe to internal and external MQTT brokers in MySQL Database

Figure 120: Example Nodes

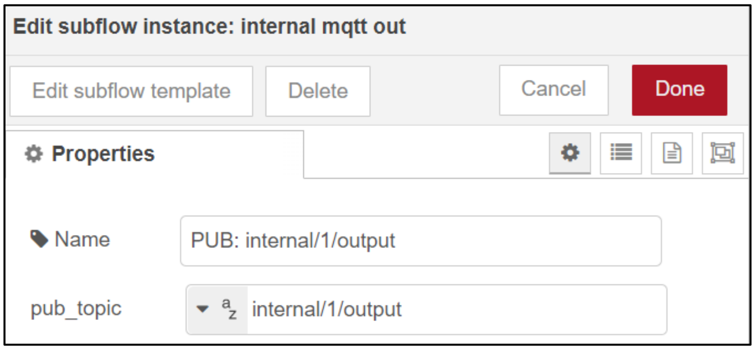


4. Click on Deploy button Deploy ▾

5. At this point, the example flow is active, and you can send a timestamp by clicking on Inject node button while observing the received message in the Debug window 🐞

6. Optionally, try changing the MQTT Topic that messages are being published to. To do so, first, click once on the MQTT PUB node (to select a segment) and then double-click to open the properties

Figure 121:pub_topic



Edit subflow instance: internal mqtt out

Name: PUB: internal/1/output

pub_topic: internal/1/output

Note: if you change the topic messages are being published to, you also need to change the topic you are subscribing to.

5.4.2 Using Node-RED Standard MQTT Nodes

Example Description

This example shows how to use internal and external MQTT brokers and communication when using Node-RED standard MQTT nodes.

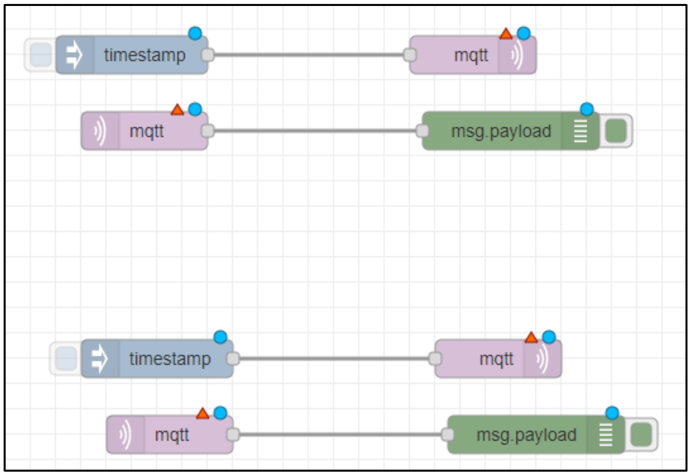
Prerequisites

Completing section 5.1, *Getting Started with Node-RED* example, is highly recommended. It shows some Node-RED usage basics and tips.

Example Walk Through

1. Drag and drop **inject**, **MQTT output**, **MQTT input,** and **debug** nodes and interconnect them as shown in the diagram.

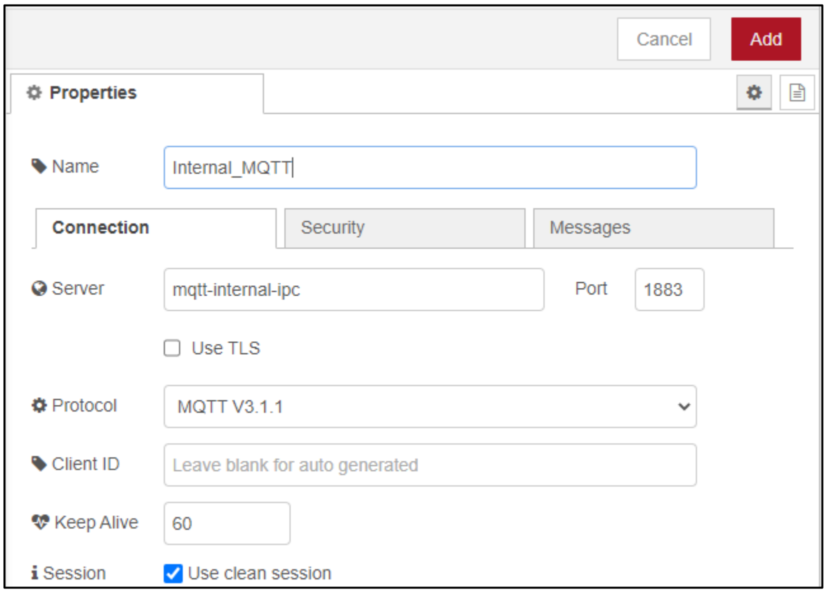
Figure 122: MQTT Nodes



2. We will use top nodes for internal MQTT communication. Double-click on the first MQTT output node, and in the Server window, select **Add new MQTT-broker** and click on the pen icon to edit.

3. Enter a name for this broker, such as **Internal_MQTT**, in the Server field, enter **mqtt-internal-ipc**, then click on **Add**.

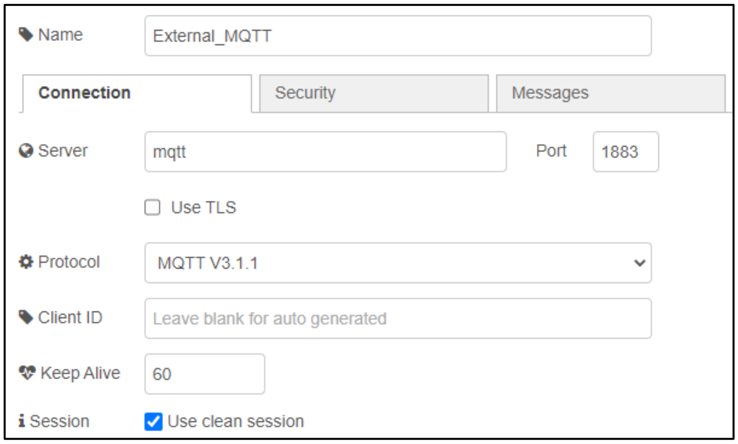
Figure 123: Name of Broker



Name: Internal_MQTT
Server: mqtt-internal-ipc
Port: 1883
□ Use TLS
Protocol: MQTT V3.1.1
Client ID: Leave blank for auto generated
Keep Alive: 60
Session: ☑ Use clean session

4. Enter the MQTT Topic name, **Internal_Test**, and click **Done** to close the window.
5. Then, double-click on the first MQTT input node and select the same server created above. Also, enter the same Topic as above, and click **Done**.
6. We will use bottom nodes for external MQTT communication. Double-click on the second **MQTT output** node, and in the Server window, select **Add new MQTT-broker** and click on the pen icon to edit.
7. Enter a name for this broker, such as **External_MQTT**, in the Server field, enter **mqtt**, then click on **Add**.

Figure 124: External MQTT Properties



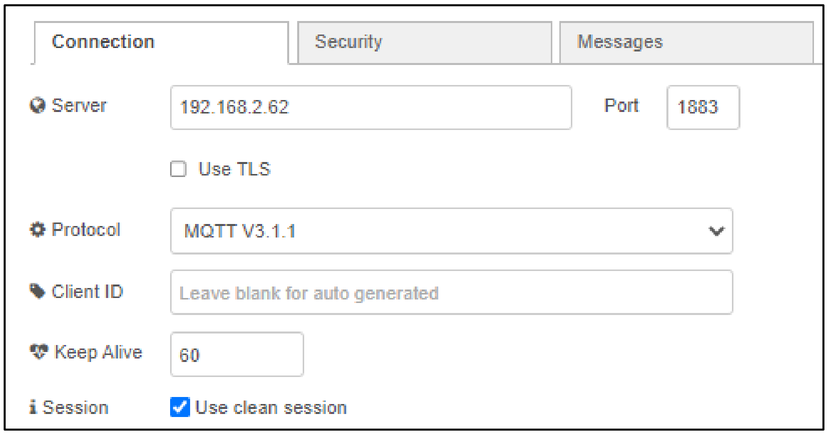
Name: External_MQTT
Server: mqtt
Port: 1883
□ Use TLS
Protocol: MQTT V3.1.1
Client ID: Leave blank for auto generated
Keep Alive: 60
Session: ☑ Use clean session

8. Enter the MQTT Topic name, **External_Test**, and click **Done** to close the window.

9. Double-click on the second MQTT input node and select the same external Server that was created above. Also, enter the same Topic as above, and click **Done**.
10. After all the edits are done, click on the **Deploy** button
11. When triggering each flow, clicking on the Inject node will show that a time stamp is generated, published to the respective MQTT Broker, and specified the topic. Then flow below, which connects to the same Broker and subscribes to the same topic, will receive it and print it in the debug window.

Note: The outside of this IPC internal MQTT Broker is not accessible, but an external broker is. If you have another device with MQTT, say another PACEdge system, then you can connect it to this setup by configuring the MQTT broker, where 192.168.2.62 is an IP address of the first PACEdge system.

Figure 125: Connection Settings



Server: 192.168.2.62 Port: 1883

□ Use TLS

Protocol: MQTT V3.1.1

Client ID: Leave blank for auto generated

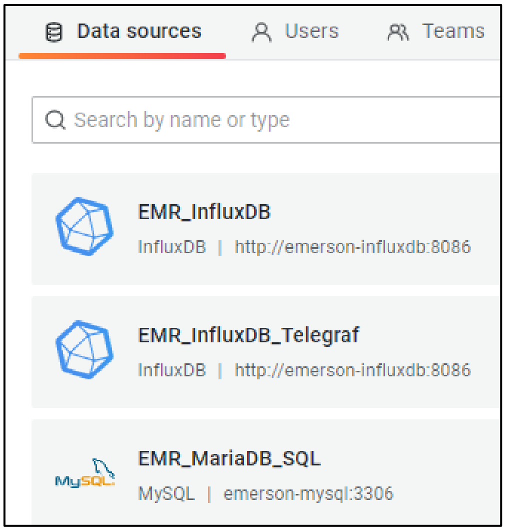
Keep Alive: 60

Session: ☑ Use clean session

5.5 Using Grafana

Grafana is a graphical application that lets users quickly visualize the data and perform different analytics. Grafana supports a large number of databases, including InfluxDB and MySQL. Starting with PACEdge version 2.2, Grafana comes pre-configured with default InfluxDB and MySQL databases **data,** named **EMR_InfluxDB** and **EMR_MariaDB_SQL**, which were used in the previous examples. In addition, by default, PACEdge is using Telegraf to collect system health data and to store it in InfluxDB database **telegraf_metrics**. This database is also pre-configured in Grafana under the name: **EMR_InfluxDB_Telegraf**.

Figure 126 Pre-configured Databased in Grafana



EMR_InfluxDB
InfluxDB | http://emerson-influxdb:8086

EMR_InfluxDB_Telegraf
InfluxDB | http://emerson-influxdb:8086

EMR_MariaDB_SQL
MySQL | emerson-mysql:3306

5.5.1 Example Description

This example will walk through using Grafana to analyze data in pre-configured databases and how to configure Grafana to use the new InfluxDB database.

5.5.2 Prerequisites

PACEdge version 2.2 or newer.

Data has been written into InfluxDB and MySQL databases using Node-RED examples in the previous section.

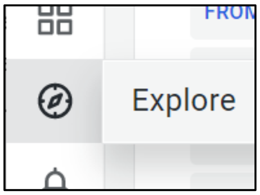
Completing section 5.1, *Getting Started with Node-RED* example, is highly recommended. It shows some Node-RED usage basics and tips.

5.5.3 Example Walk Through

Using Grafana to Analyze the pre-configured InfluxDB database

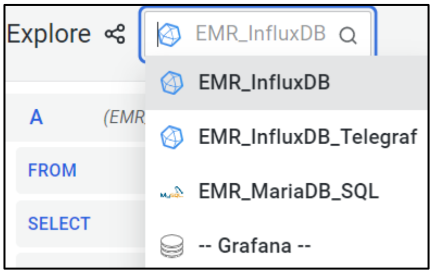
1. Open Grafana and log in as an admin user
2. Click on Explore icon in the upper left corner

Figure 127: Explore Button



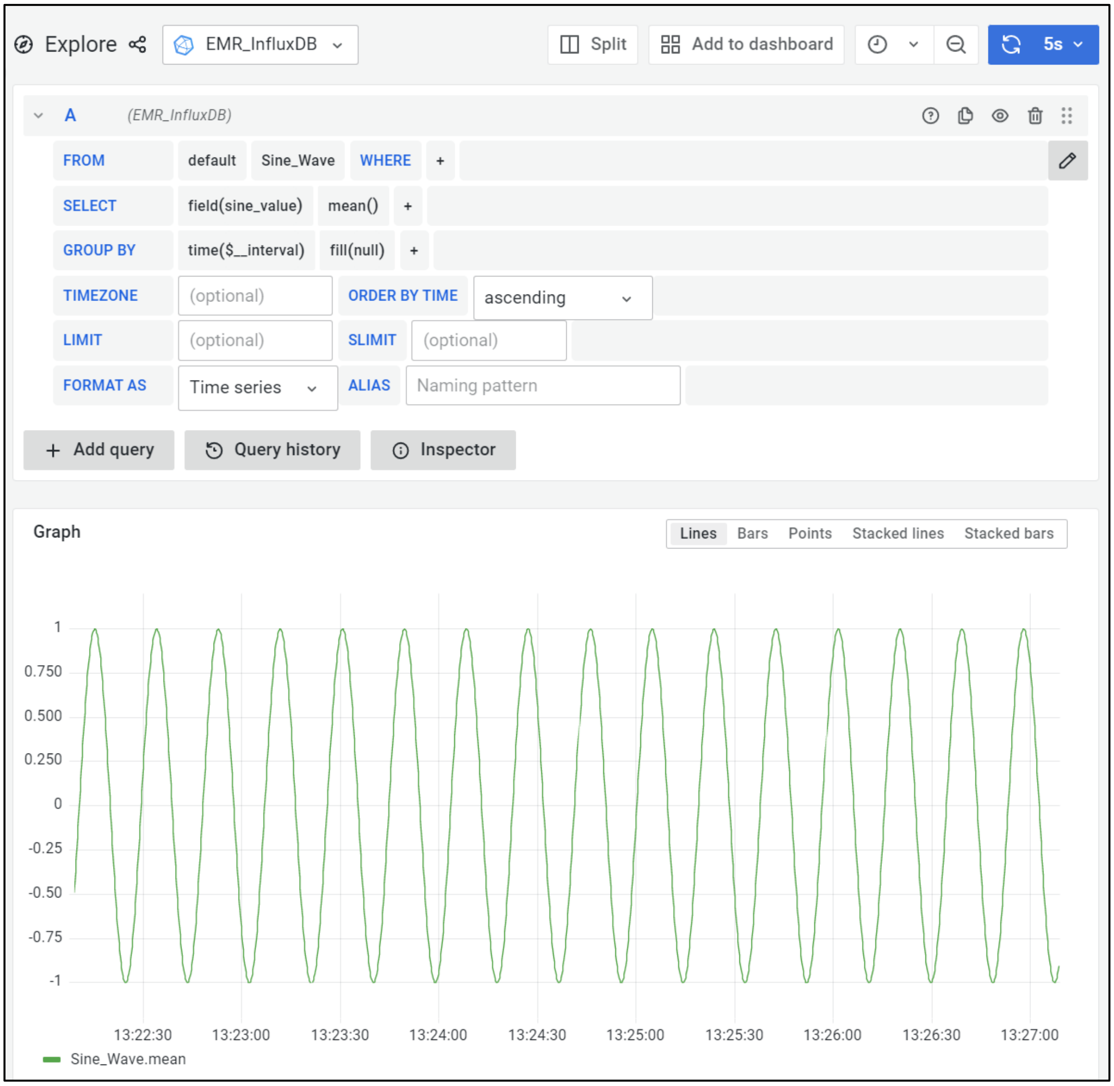
3. Select the EMR_InfluxDB database in the drop-down list

Figure 128: EMR_InfluxDB



4. In the Measurement, select **Sine_Wave**; in Field, select **sine_value**. All other values can be left at default. The measurement will show the sine wave generated by Node-RED and stored in InfluxDB.
   Note: Play around by changing the time to the last 5 minutes and setting the query to run every 5 sec.

Figure 129: EMR_InfluxDB



FROM default Sine_Wave WHERE +

SELECT field(sine_value) mean() +

GROUP BY time($__interval) fill(null) +

TIMEZONE (optional) ORDER BY TIME ascending

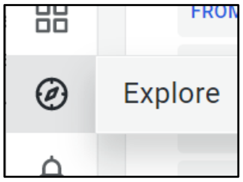
LIMIT (optional) SLIMIT (optional)

FORMAT AS Time series ALIAS Naming pattern

Using Grafana to Analyze pre-configured MySQL database

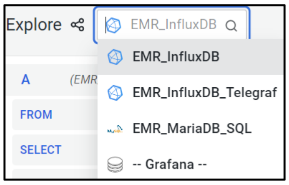
1. Open Grafana and log in as an admin user
2. Click on Explore icon in the upper left corner

Figure 130: Explore Button



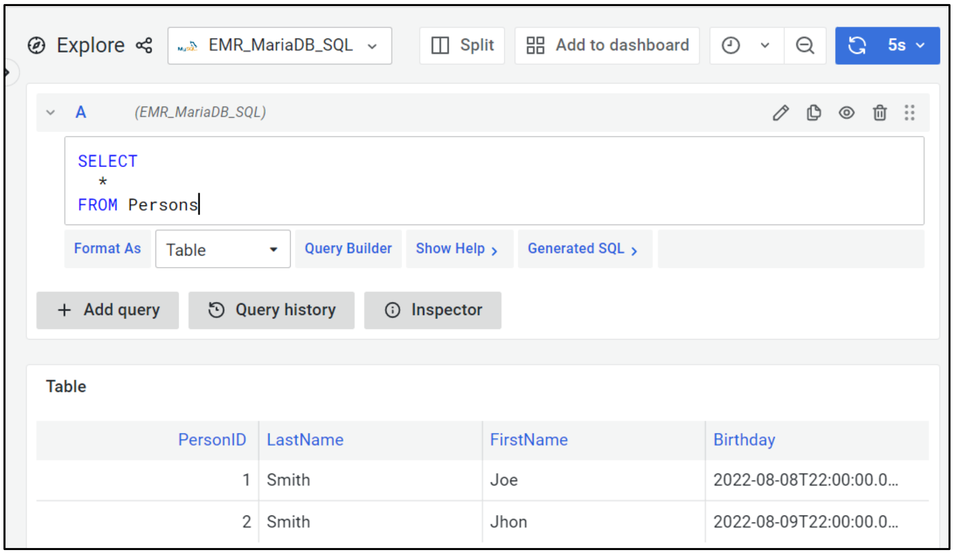
3. Select the EMR_MariaDB_SQL database in the drop-down list

Figure 131: EMR_InfluxDB



4. For this example, switch to text query mode by clicking on the pencil icon ✎ on the right side of the screen.
5. Modify query entry to show all values in the table called Persons and format output data as Table

Figure 132: Format Data as a Table



```
SELECT
  *
FROM Persons
```

Format As: Table

Table

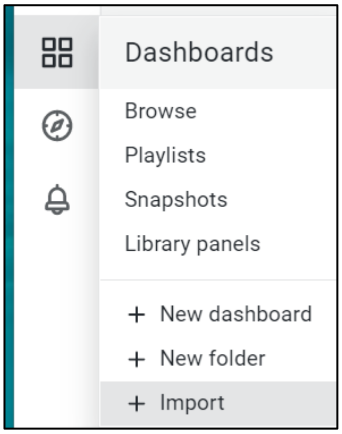
PersonID	LastName	FirstName	Birthday
1	Smith	Joe	2022-08-08T22:00:00.0…
2	Smith	Jhon	2022-08-09T22:00:00.0…

Install Open Community Grafana Dashboard

By default, PACEdge uses Telegraf to collect system health data and store it in InfluxDB. This data is stored in a database called telegraf_metrics under Grafana name EMR_InfluxDB_Telegraf. Few ready-to-use Grafana dashboards are available from the open community to visualize this data.

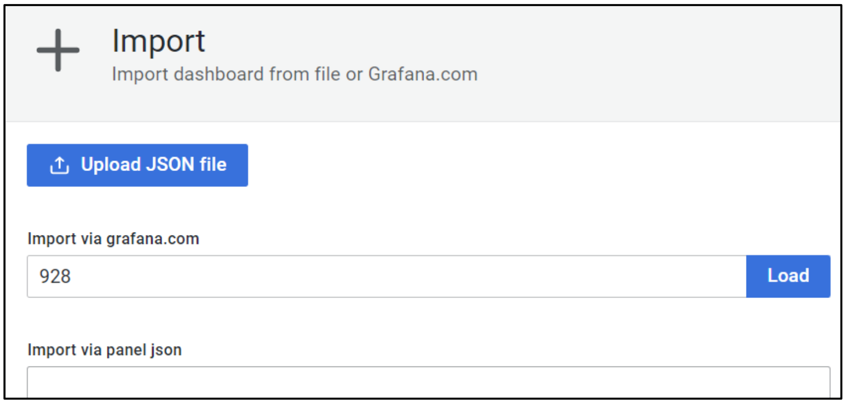
1. Open Grafana and log in as an admin user.
2. Hover the mouse over the Dashboards icon and select **+Import**

Figure 133: Dashboard



3. In the Import dialogue, enter number **928** into Import via the Grafana.com field and click the **Load** button.

Figure 134: Import Button



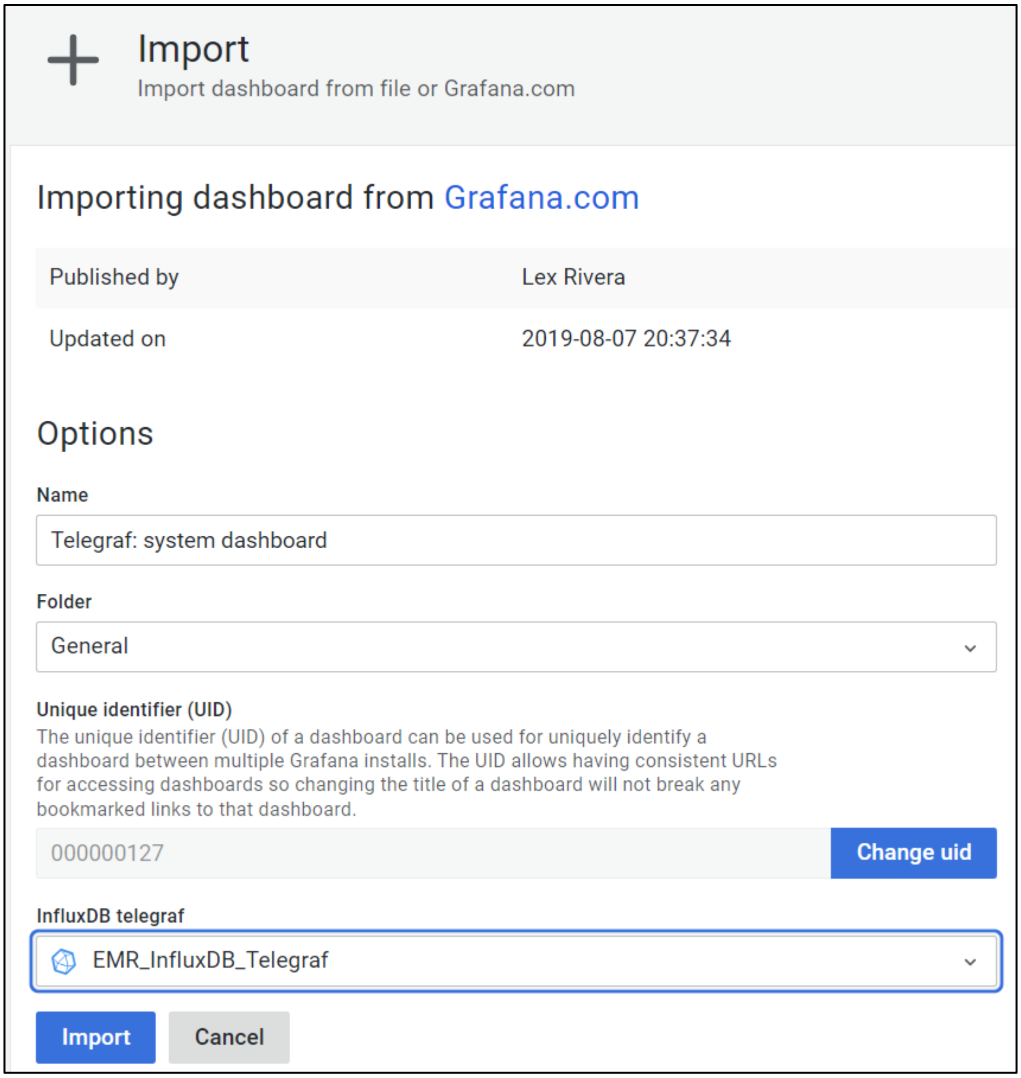
Import via grafana.com: 928

Import via panel json

Note: PACEdge device requires access to the Internet for this operation

4. Select **EMR_InfluxDB_Telegraf** database from the drop-down menu at the bottom and click Import.

Figure 135: Importing dashboard from Grafana



Import

Import dashboard from file or Grafana.com

Importing dashboard from Grafana.com

Published by	Lex Rivera
Updated on	2019-08-07 20:37:34

Options

Name

Telegraf: system dashboard

Folder

General

Unique identifier (UID)

The unique identifier (UID) of a dashboard can be used for uniquely identify a dashboard between multiple Grafana installs. The UID allows having consistent URLs for accessing dashboards so changing the title of a dashboard will not break any bookmarked links to that dashboard.

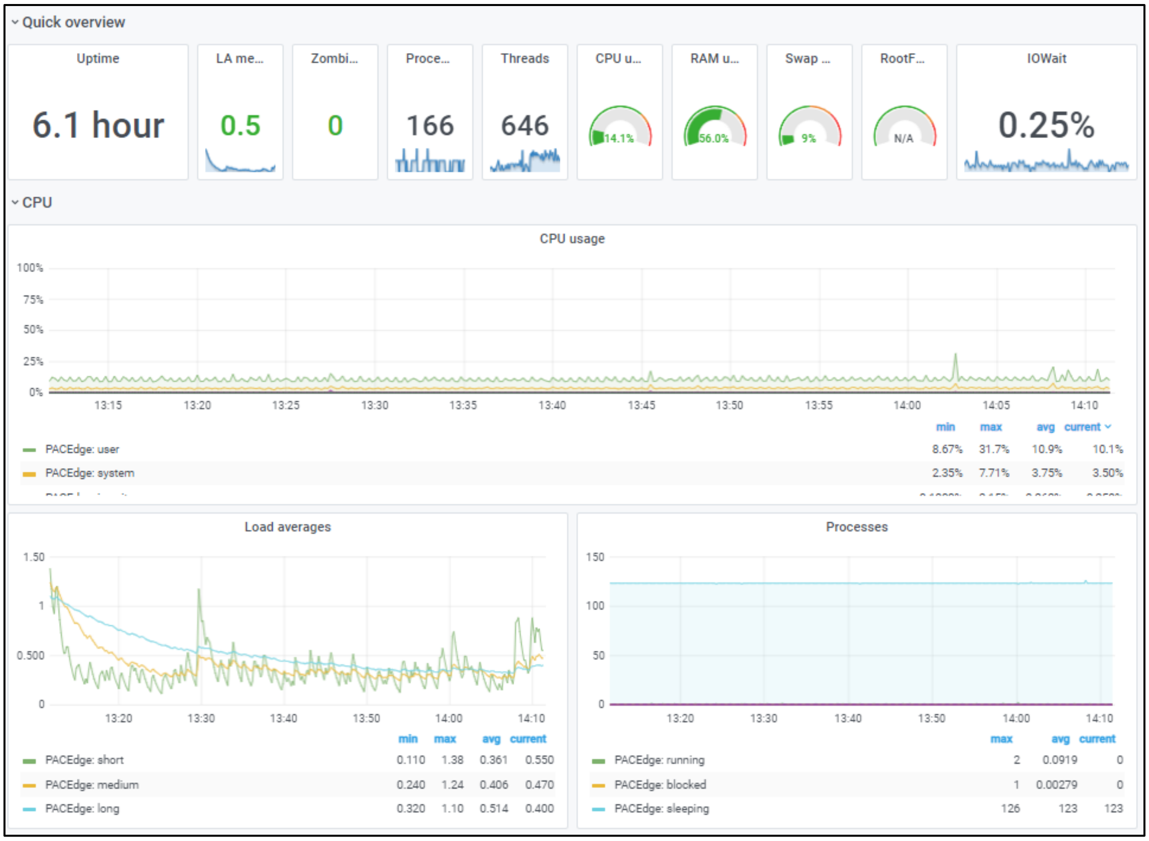
000000127

InfluxDB telegraf

EMR_InfluxDB_Telegraf

5. A dashboard showing the variety of PACEdge system functions will be shown

Figure 136: Dashboard



	min	max	avg	current
🟩 PACEdge: user	8.67%	31.7%	10.9%	10.1%
🟨 PACEdge: system	2.35%	7.71%	3.75%	3.50%

	min	max	avg	current
🟩 PACEdge: short	0.110	1.38	0.361	0.550
🟨 PACEdge: medium	0.240	1.24	0.406	0.470
🟦 PACEdge: long	0.320	1.10	0.514	0.400

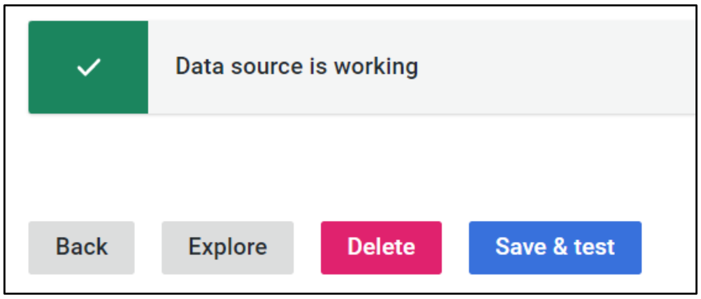
	max	avg	current
🟩 PACEdge: running	2	0.0919	0
🟨 PACEdge: blocked	1	0.00279	0
🟦 PACEdge: sleeping	126	123	123

Configuring Grafana with the new InfluxDB database

This example will demonstrate how to access the InfluxDB database test created in the Node-RED example above.
Note: This example will fail if a database test does not exist.

1. Open Grafana from the PACEdge landing page and log in as the admin user.
2. Move the cursor over the Configuration icon on the left side and select **Data Sources.**
3. Click on Add Data Source and select InfluxDB.
4. Enter values as shown, the admin password, and click on **Save & Test** at the bottom. You should get a message **Data Source is working.**

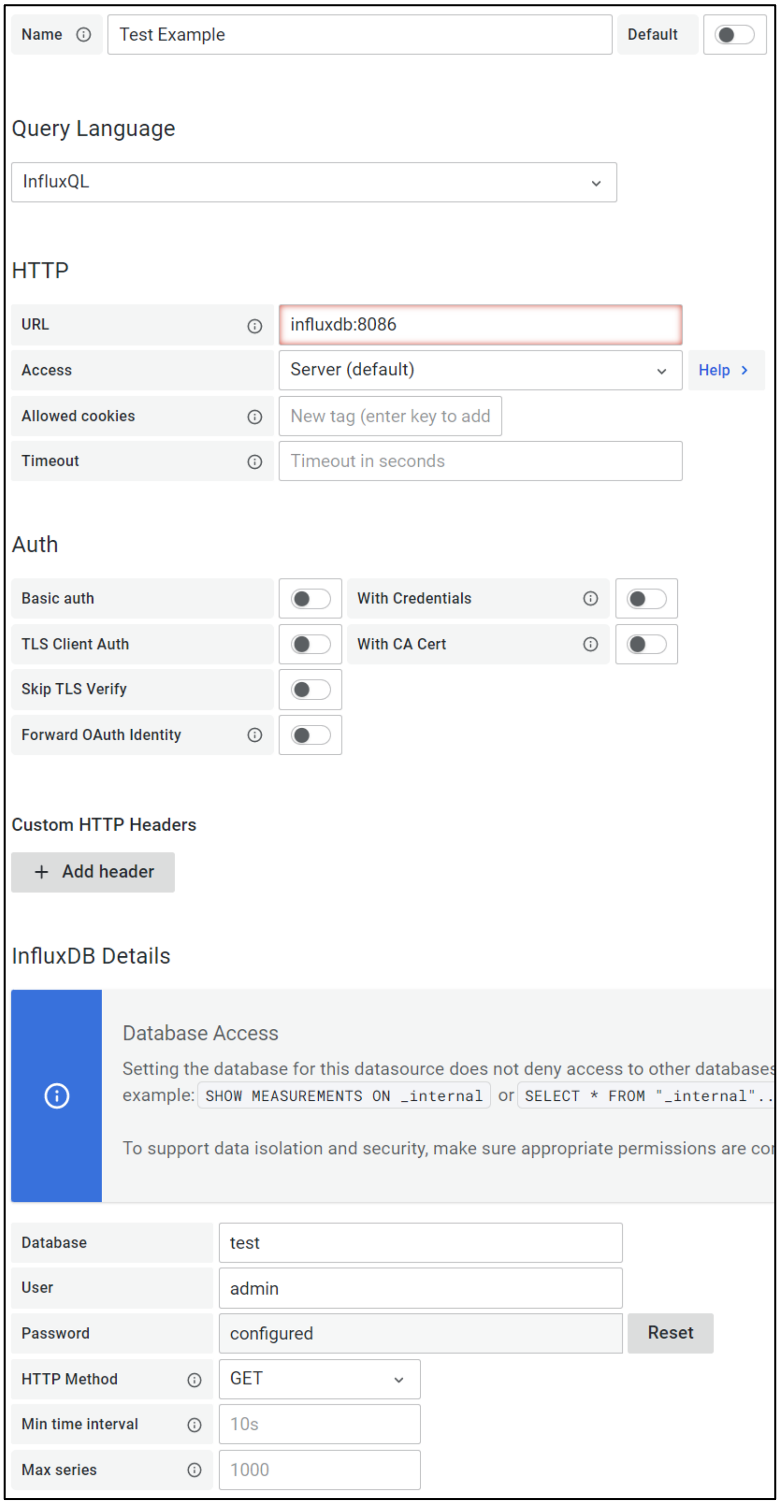
Figure 137: Data Source



Data source is working

Note: If the database in InfluxDB has not been created yet, then trying to add it in Grafana will fail. Make sure that a particular database already exists before adding it to Grafana.

Figure 138: Grafana Configuration Screen for InfluxDB



Name: Test Example

Default: ○

Query Language

InfluxQL

HTTP

URL: influxdb:8086

Access: Server (default)

Allowed cookies: New tag (enter key to add

Timeout: Timeout in seconds

Auth

Basic auth: ○

With Credentials: ○

TLS Client Auth: ○

With CA Cert: ○

Skip TLS Verify: ○

Forward OAuth Identity: ○

Custom HTTP Headers

InfluxDB Details

Database Access

Setting the database for this datasource does not deny access to other databases
example: `SHOW MEASUREMENTS ON _internal` or `SELECT * FROM "_internal"..`

To support data isolation and security, make sure appropriate permissions are co

Database: test

User: admin

Password: configured

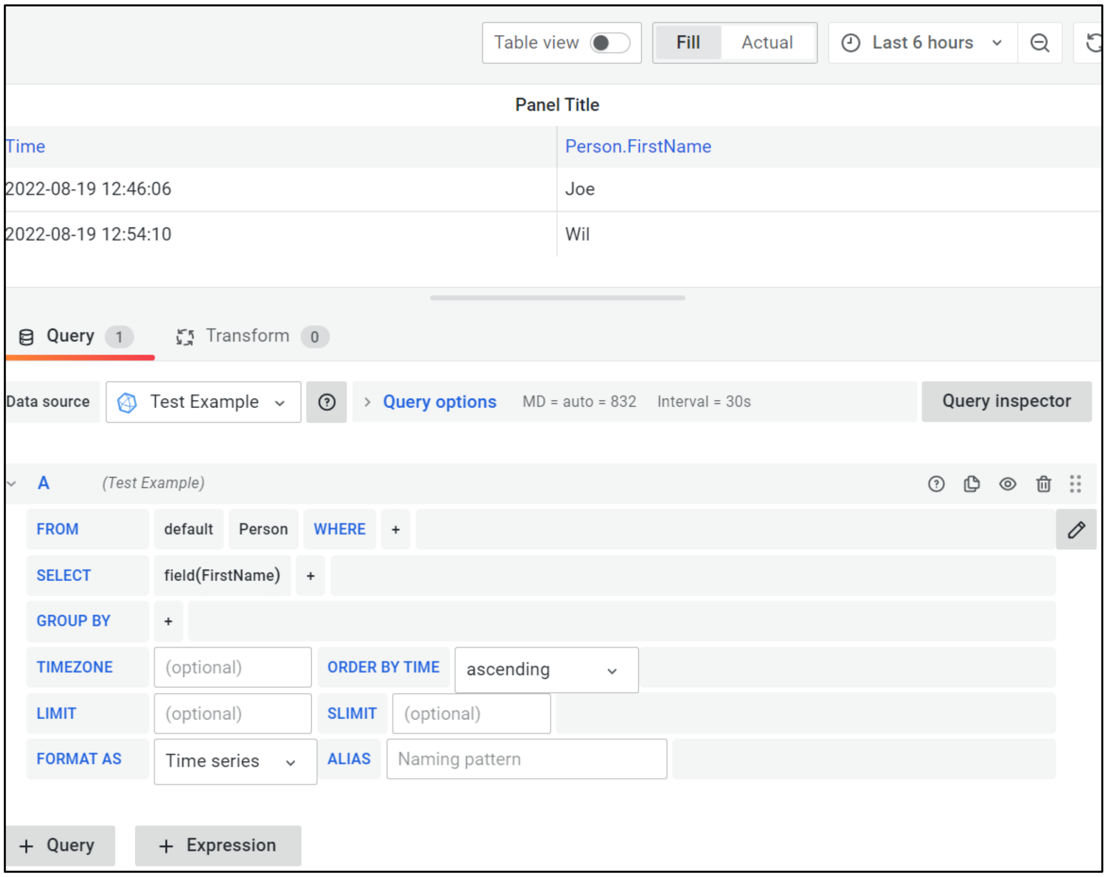
HTTP Method: GET

Min time interval: 10s

Max series: 1000

1. To create a new dashboard in Grafana, hover your mouse over the Dashboards symbol and click on **+New Dashboard**, then select the **Add a New Panel** option.
2. Select your configured InfluxDB database, **Test Example** in this case.
3. In line item FROM from the drop-down list, pick **Person**
4. In the SELECT line item, pick the field value **FirstName**
5. In the GROUP BY line, click on the word **Time** and select the option **Remove**
6. Click on the **Switch to table** option that will be shown.
7. At this point, you should see the first names of the persons you have entered in the Node-RED example.

Figure 139: Telegraf Metrics



Panel Title

Time	Person.FirstName
2022-08-19 12:46:06	Joe
2022-08-19 12:54:10	Wil

Data source: Test Example

Query options MD = auto = 832 Interval = 30s

A (Test Example)

FROM default Person WHERE +

SELECT field(FirstName) +

GROUP BY +

TIMEZONE (optional) ORDER BY TIME ascending

LIMIT (optional) SLIMIT (optional)

FORMAT AS Time series ALIAS Naming pattern

Configuring Grafana with MySQL database

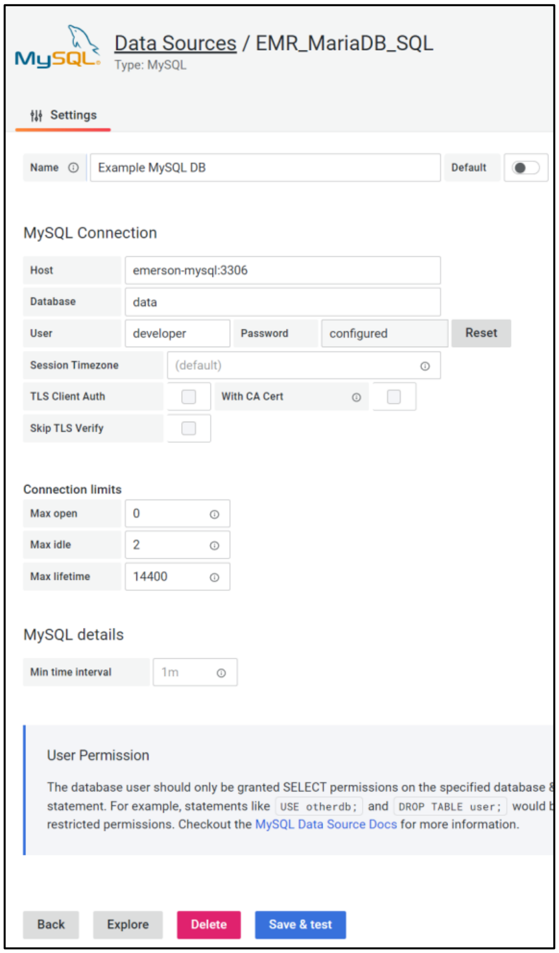
This example will show how to add a new MySQL database. For this purpose, we will use an already pre-configured database and add it under a different name.

1. From the PACEdge landing page, click on the **Grafana** link and log in as admin.
2. Move the cursor over the Configuration icon on the left side and select **Data Sources.**
3. Click on **Add Data Source** and select **MySQL.**

Note: if the database in MySQL has not been created yet, then trying to add it in Grafana will fail. Make sure that a particular database already exists before adding it to Grafana. The following steps assume that the database Movicon has been created in MySQL.

4. Enter values as shown and the admin or developer password, and click on Save & Test at the bottom. You should get a message **Database Connection OK**.

Figure 140: MySQL Connection



Data Sources / EMR_MariaDB_SQL
Type: MySQL

Name	Example MySQL DB	Default	

MySQL Connection

Host	emerson-mysql:3306			
Database	data			
User	developer	Password	configured	Reset
Session Timezone	(default)			
TLS Client Auth	□	With CA Cert	□	
Skip TLS Verify	□			

Connection limits

Max open	0
Max idle	2
Max lifetime	14400

MySQL details

Min time interval	1m

User Permission

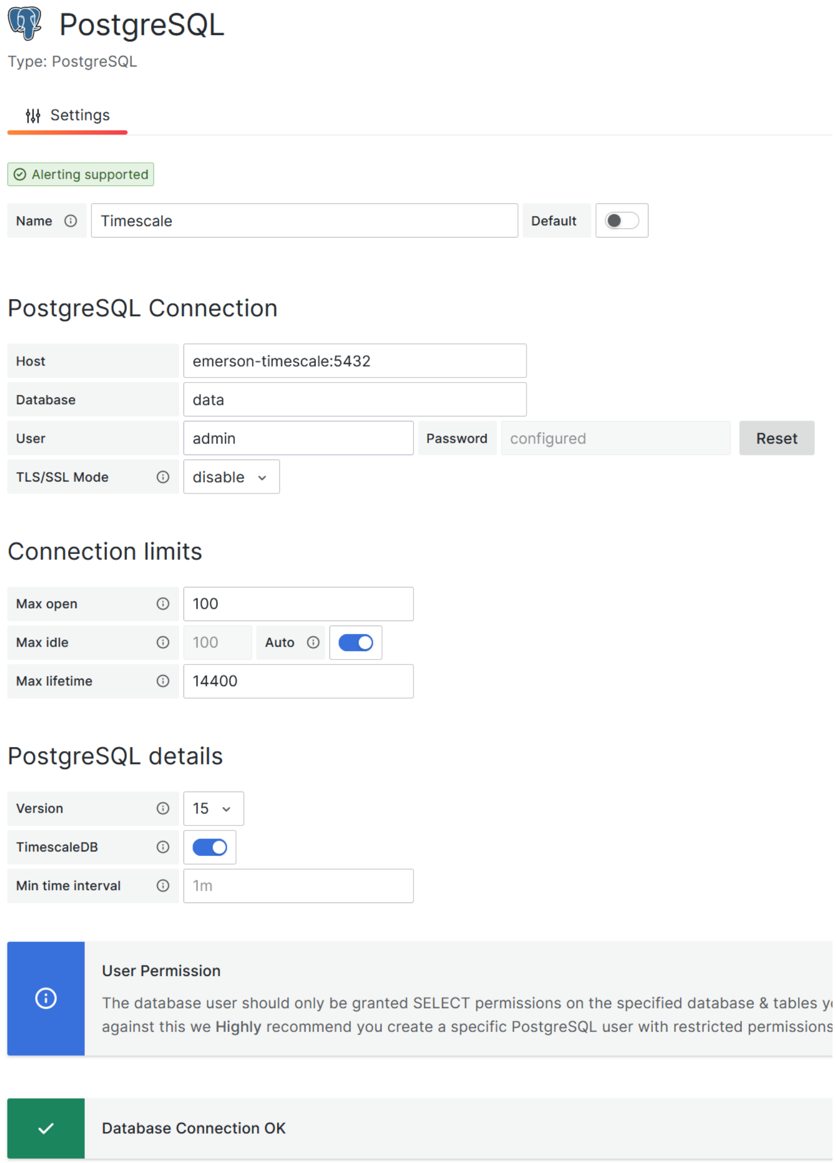
The database user should only be granted SELECT permissions on the specified database & statement. For example, statements like `USE otherdb;` and `DROP TABLE user;` would b restricted permissions. Checkout the MySQL Data Source Docs for more information.

Configuring Grafana with TimescaleDB database

To configure TimescaleDB as a datasource in Grafana, perform following steps:

1. Log into Grafana as admin user, on the left side of the screen, click on Adminitration and then select Data Sources.
2. Enter information as shown below, giving data base a desired name, but configuring Host exactly as shown. Put in admin or developer user and associated passwords, disable TLS (it is an internal connection, no need for it).

Figure 141 TimescaleDB Configuration Screen in Grafana



PostgreSQL
Type: PostgreSQL

Settings

Alerting supported

Name: Timescale
Default ○

PostgreSQL Connection

Host: emerson-timescale:5432
Database: data
User: admin
Password: configured
Reset
TLS/SSL Mode: disable

Connection limits

Max open: 100
Max idle: 100
Auto ●
Max lifetime: 14400

PostgreSQL details

Version: 15
TimescaleDB ●
Min time interval: 1m

User Permission
The database user should only be granted SELECT permissions on the specified database & tables y
against this we Highly recommend you create a specific PostgreSQL user with restricted permissions

Database Connection OK

3. Once data is typed in, click on the Save & test button and make sure you get a message “Database Connection OK”
4. Now you can explore the data and setup desired dashboards using TimescaleDB

5.6 Using Connext and WebHMI

This example shows how to instantiate the OPC UA client in Connext, get data from OPC UA Server on the PACSystems CPL410 Controller, and visualize this data via both WebHMI and Grafana.

5.6.1 Prerequisites

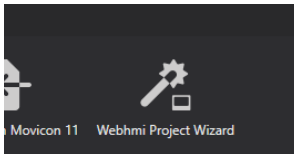
PACEdge version 2.2 or newer.

Basic knowledge of Movicon.NExT is required.

5.6.2 Example Walk Through

Movicon Connext and WebHMI functionality must first be created on the user workstation using Movicon.NExT software, typically running under Windows. Once required data connectivity, internal data tags, historian settings, and WebHMI screens are designed, a project can be uploaded to the unit running PACEdge. Note that when creating a new project with Movicon.NExT you need to select Webhmi Project type.

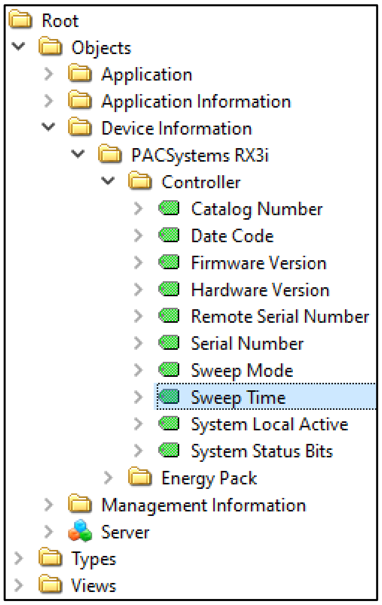
Figure 142: WebHMI Project



For detailed tutorials and documentation on creating Connext and WebHMI projects, please consult the Movicon documentation. The following sections will provide only fundamental guidance.

For example, let us assume we have a CPL410 and want to use OPC UA to get data from the CPL410 PACs OPC UA server. PACs controller typically has a system tag called Sweep Time, shown in the UA Expert figure below:

Figure 143: UA Expert view of CPL410 data tags

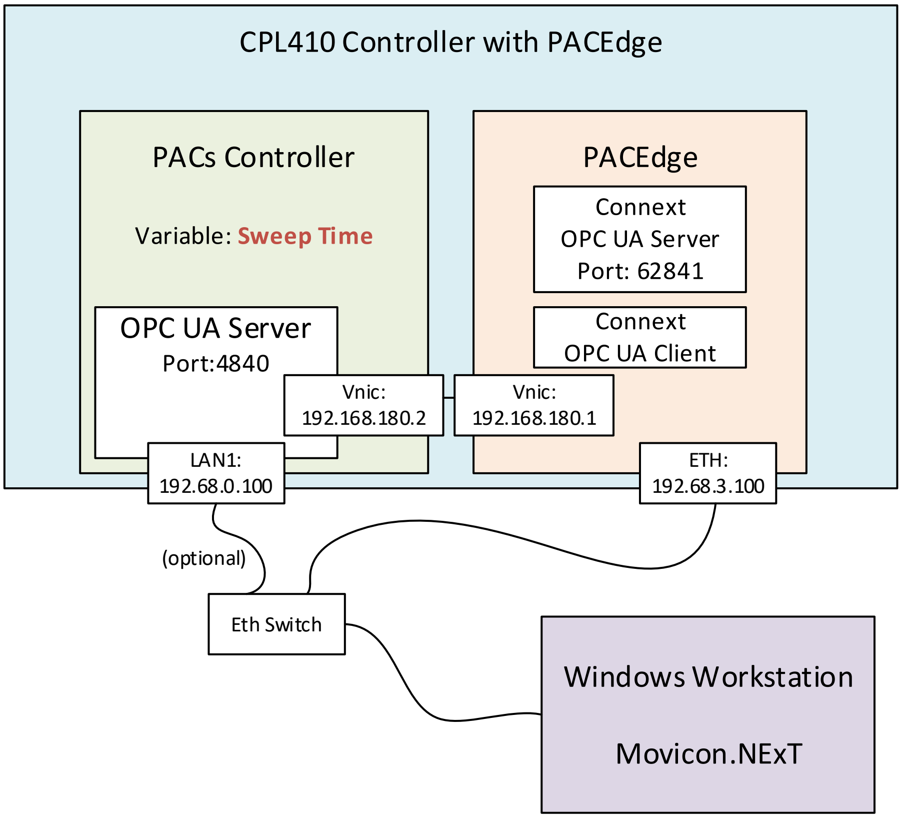


- Root
  - Objects
    - Application
    - Application Information
    - Device Information
      - PACSystems RX3i
        - Controller
          - Catalog Number
          - Date Code
          - Firmware Version
          - Hardware Version
          - Remote Serial Number
          - Serial Number
          - Sweep Mode
          - Sweep Time
          - System Local Active
          - System Status Bits
        - Energy Pack
    - Management Information
    - Server
  - Types
  - Views

In our example, we will read this tag using I/O Data Server in WebHMI, store it with Historian in the PACEdge MySQL database, and visualize it via both WebHMI and Grafana.

CPL410 is a hybrid device containing both PACs automation controller and PACEdge. A PACSystems controller has its own OPC UA server, which enables access to different data tags. This OPC UA server can be accessed by PACEdge using an internal virtual Ethernet port (VNIC) and an external Ethernet LAN1 port.

Figure 144: CPL410 Internal Block Diagram



There are different ways to configure OPC UA Client, but for this example, we will utilize the new OPC-UA Port Forwarding feature that was added in the PACEdge version 2.2.
The example will use the following steps:

- Use Movicon.NExT on the workstation to browse through OPC UA data tags on the PACs OPC UA Server using the new OPC-UA Port Forwarding feature via the internal VNIC port.
- Configure WebHMI project
- Test the WebHMI project locally on the workstation
- Transfer WebHMI project to target CPL410
- Test WebHMI project running on CPL410

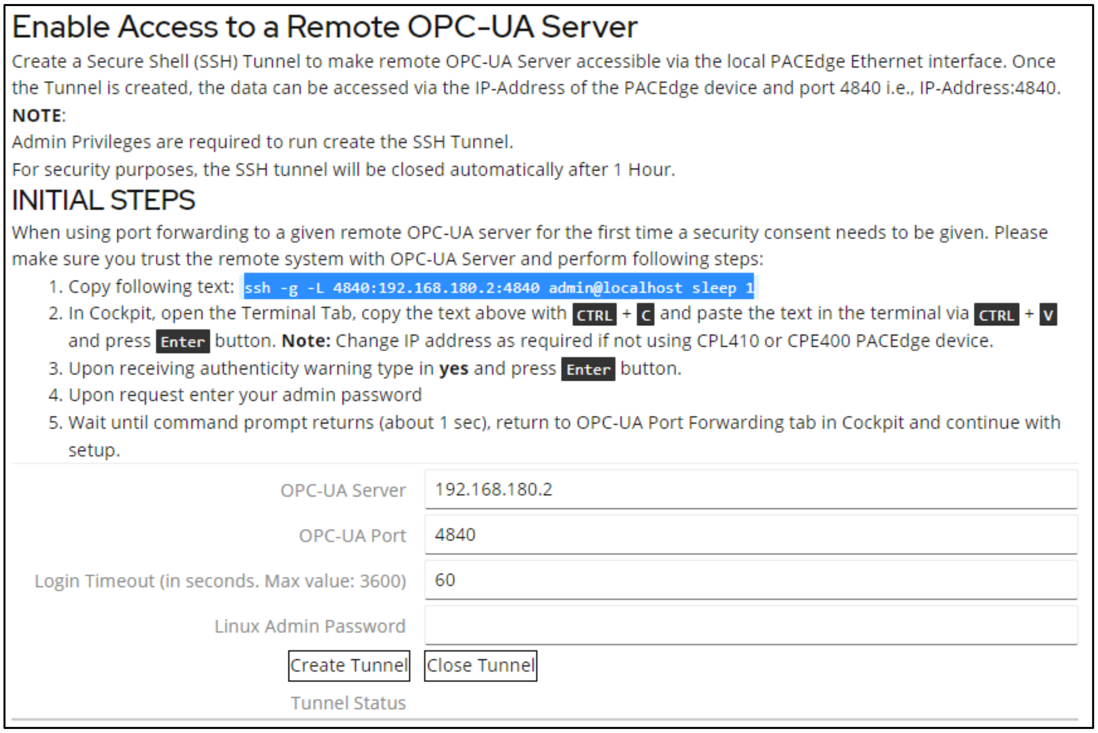
Example Walk Through

Enabling the OPC UA Port Forwarding Feature

The OPC UA Port Forwarding feature enables to temporarily forward port used for OPC-UA communication via internal VNIC to an external Ethernet port. This allows the user to configure the Movicon OPC-UA driver without needing to plug an Ethernet cable into the LAN1 port on the CPL410 or CPE400 controller. To enable OPC-UA Port Forwarding, follow these steps:

1. Log into Cockpit as an **admin** user
2. Open the OPC-UA Port Forwarding tab
3. If using this feature for the very first time, a manual step accepting the security implications of such port forwarding is required. To do so, highlight the text shown in the Cockpit and use the **CTRL+C** key combination to copy:

Figure 145: Enable Access to a Remote OPC UA Server



Enable Access to a Remote OPC-UA Server

Create a Secure Shell (SSH) Tunnel to make remote OPC-UA Server accessible via the local PACEdge Ethernet interface. Once the Tunnel is created, the data can be accessed via the IP-Address of the PACEdge device and port 4840 i.e., IP-Address:4840.

NOTE:

Admin Privileges are required to run create the SSH Tunnel.

For security purposes, the SSH tunnel will be closed automatically after 1 Hour.

INITIAL STEPS

When using port forwarding to a given remote OPC-UA server for the first time a security consent needs to be given. Please make sure you trust the remote system with OPC-UA Server and perform following steps:

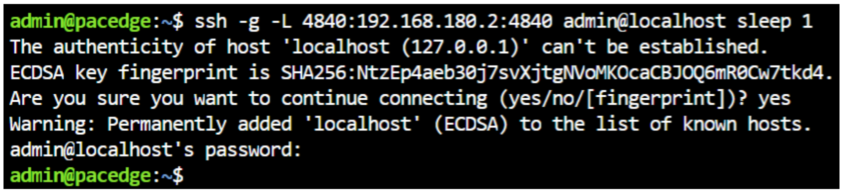
1. Copy following text: `ssh -g -L 4840:192.168.180.2:4840 admin@localhost sleep 1`
2. In Cockpit, open the Terminal Tab, copy the text above with `CTRL` + `C` and paste the text in the terminal via `CTRL` + `V` and press `Enter` button. **Note:** Change IP address as required if not using CPL410 or CPE400 PACEdge device.
3. Upon receiving authenticity warning type in **yes** and press `Enter` button.
4. Upon request enter your admin password
5. Wait until command prompt returns (about 1 sec), return to OPC-UA Port Forwarding tab in Cockpit and continue with setup.

OPC-UA Server	192.168.180.2
OPC-UA Port	4840
Login Timeout (in seconds. Max value: 3600)	60
Linux Admin Password	

4. Open Terminal in Cockpit and Paste this line by using the **CTRL+V** key combination, hit enter

6. A message stating that the authenticity of the localhost can not be established asking if you want to proceed with this operation. Enter yes.
   When asked, enter the admin password.

Figure 146: Enter Password



```
admin@pacedge:~$ ssh -g -L 4840:192.168.180.2:4840 admin@localhost sleep 1
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:NtzEp4aeb30j7svXjtgNVoMKOcaCBJOQ6mR0Cw7tkd4.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
admin@localhost's password:
admin@pacedge:~$
```

This operation set up a port forwarding for 1 sec and will close afterward. Wait a couple of seconds until the command prompt is back

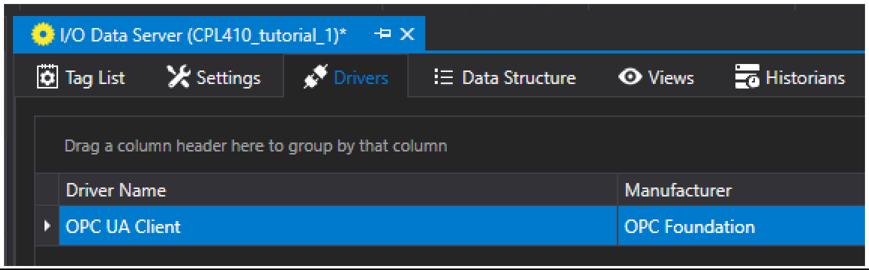
7. Return to the OPC-UA Port Forwarding tab in Cockpit, enter the admin password in the field, and click Create Tunnel. You will get the message: Tunnel Created. At this point, you can proceed with the next steps
   Note: port forwarding is default enabled for 60 seconds and for only one session. It will automatically get disabled afterward. This is done for security reasons.

Configuring I/O Data Server

Following are the steps to configure the OPC UA Client in I/O Data Server to read a variable from PACs Controller:

1. In Movicon.NExT use a wizard to create new WebHMI project
2. In Movicon.NExT, in the I/O Data Server section, double-click on **Drivers**.
3. In the main window, right-click, select **Add New Driver,** and select **OPC UA Client** from the list, click on the **>** button.
4. Keep the default settings in all steps and click on the **>** button to proceed.
5. At the last step click on **Finish**. You will see an OPC UA Client driver listed in the driver window.

Figure 147: OPC UA Client



I/O Data Server (CPL410_tutorial_1)*
Tag List
Settings
Drivers
Data Structure
Views
Historians
Drag a column header here to group by that column

Driver Name	Manufacturer
OPC UA Client	OPC Foundation

6. Right-click on the OPC UA Client entry and select **Import Tags**.

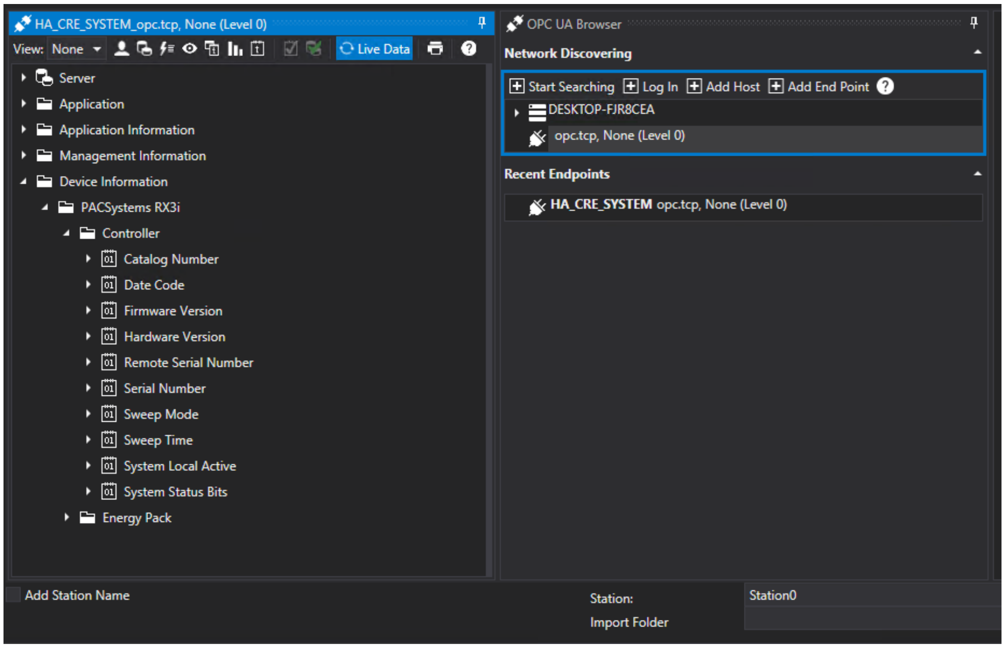
8. Select the **Browse Server** tab and then either:

    a. **Add End Point**: **opc.tcp://*ip_address_of_ETH_port*:4840**
    (where IP address is the address of ETH port, such as 192.168.3.100 if static default IP is being used)
    OR

    b. **Add End Point**: **opc.tcp://192.168.0.100:4840**

9. Double-click on the newly created server entry to connect to OPC UA Server, and browse for available tags.

Figure 148: Browse for Available Tags



Station: Station0

10. Click on **Sweep Time** and click **OK**. A new data tag will be added to the Connext Tags List.

Configuring WebHMI Screen

1. In Movicon.NExT, right-click on **Screens** and select **New** to configure the HMI Screen.
2. Choose the screen size, enter the screen's name and click **OK.**
3. On the right edge of the screen, click on **ToolBox**, then on **Displays**, and then drag a **display** to the HMI screen.
4. Double-click on the display. In the Tag field, select the **Sweep_Time** tag. Expand the Display Settings, and in the Precision Digits, enter **6**. Click on the checkmark in the upper-left corner to close the Properties dialogue.
5. At this point, we can test to see if our setup is working on the workstation locally. To do so, save the project and click on the **Start Runtime** button. Once HMI starts locally on the workstation, you will see controller sweep time being updated once a second.

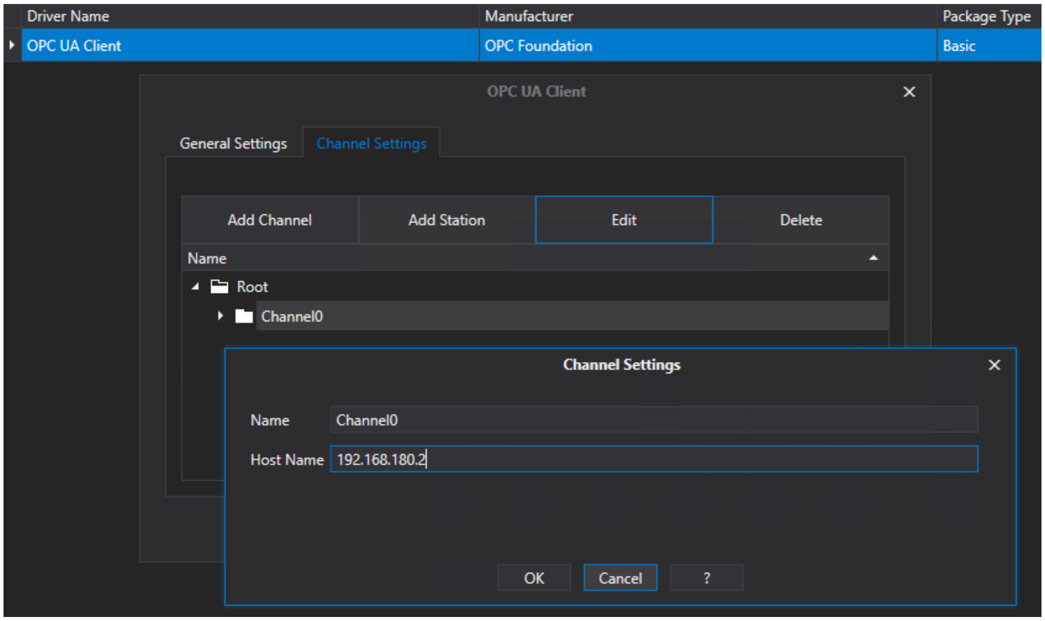
Note: To stop the HMI on the workstation, press the Alt+F4 keys.

Configure Connext/WebHMI Project for Deployment on CPL410 Controller

To transfer the WebHMI project from Movicon.NExT development tool to CPL410 controller, we need to change the ethernet interface to an internal VNIC inside the CPL410. Perform the following steps:

1. In Movicon.NExT I/O Data Server section, double-click on **Drivers**.
2. Double-click on **OPC UA Client**.
3. Select **Channel Settings**, select **Channel0**, and click **Edit**.
4. Change the Host Name to CPL410 VNIC IP address: 192.168.180.2.
5. Click **OK** to close all dialogues.

Figure 149: Host Name



Driver Name	Manufacturer	Package Type
OPC UA Client	OPC Foundation	Basic

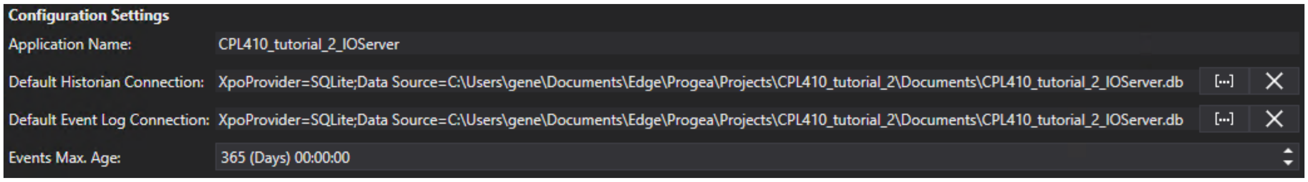
Name: Channel0

Host Name: 192.168.180.2

In addition, configure the Historian in I/O Data Server to use the PACEdge MySQL database. To do so:

1. In Movicon.NExT I/O Data Server section, double-click on **Settings**.
2. In Default Historian Connection, click on the three dots on the right side to access settings.

Figure 150: Configuration Settings



Configuration Settings

Application Name: CPL410_tutorial_2_IOServer

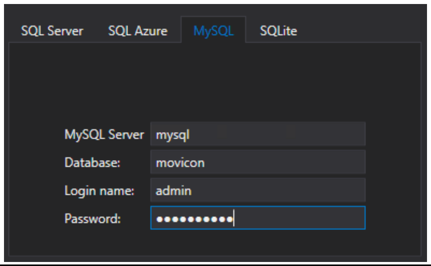
Default Historian Connection: XpoProvider=SQLite;Data Source=C:\Users\gene\Documents\Edge\Progea\Projects\CPL410_tutorial_2\Documents\CPL410_tutorial_2_IOServer.db

Default Event Log Connection: XpoProvider=SQLite;Data Source=C:\Users\gene\Documents\Edge\Progea\Projects\CPL410_tutorial_2\Documents\CPL410_tutorial_2_IOServer.db

Events Max. Age: 365 (Days) 00:00:00

3. Select the MySQL tab, enter **mysql** as the server name, choose a name for the database, login name as **admin**, and choose the admin password you have configured for the PACEdge. Click **OK**.

Figure 151: Password

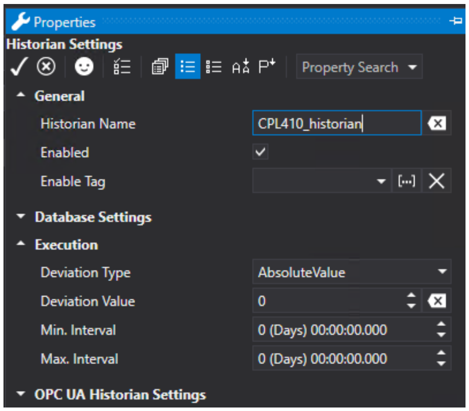


MySQL Server: mysql
Database: movicon
Login name: admin
Password: ••••••••••

4. At this point, you will get a message saying Unable to connect to the database, which is expected since you cannot connect to MySQL inside the PACEdge from outside. Click **OK** to choose this dialogue.

5. Double-click on **Historians in the I/O Data Server** section, then in the main window, right-click on **Historian** and choose **Add Historian Settings**. Keep default settings, give some name to the Historian instance and click on the checkmark in the upper-left corner.

Figure 152: Properties



Historian Settings

General
Historian Name: CPL410_historian
Enabled: ✓
Enable Tag:

Database Settings

Execution
Deviation Type: AbsoluteValue
Deviation Value: 0
Min. Interval: 0 (Days) 00:00:00.000
Max. Interval: 0 (Days) 00:00:00.000

OPC UA Historian Settings

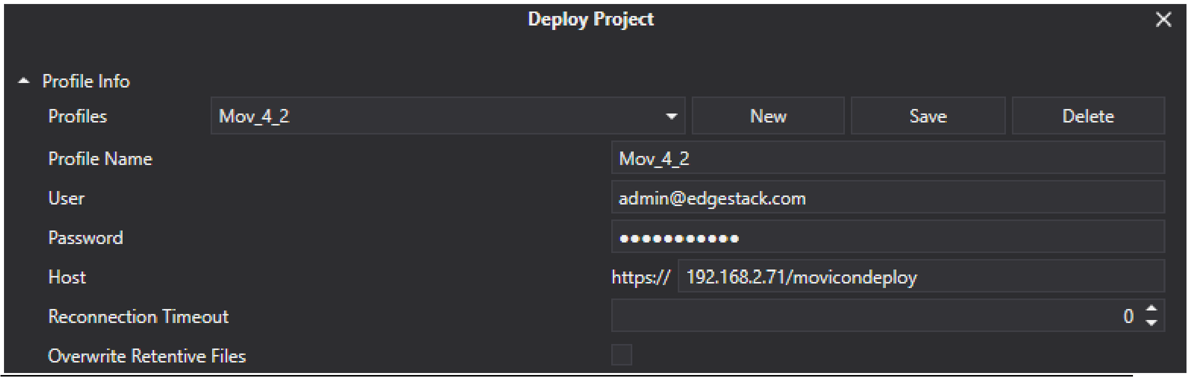
6. Right-click on the newly created historian, select Assign Tag, navigate to the **Sweep Time** variable, and click **OK.**

7. Save the Movicon.NExT project

Deploy Movicon Project to CPL410 PACEdge Controller

1. Click on the button **Create WebHMI**. Click on **Yes** to run WebHMI Server from remote.
2. In the Deploy Project window that opens, enter the username and password assigned using the Password Management utility.
   Note: username and password are stored in the appsettings.json file as described in section □ PACEdge External Communication Parameters

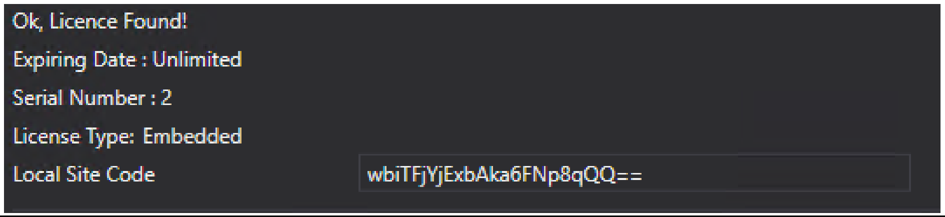
Figure 153: Connext Login



Profiles	Mov_4_2
Profile Name	Mov_4_2
User	admin@edgestack.com
Password	●●●●●●●●●●●
Host	https:// 192.168.2.71/movicondeploy
Reconnection Timeout	0
Overwrite Retentive Files	□

3. Click on the **Connect** button, respond with Yes to Deploy the Project, and upload I/O Data Server and WebHMI questions the first time you run.
4. Once everything is uploaded, expand Advanced Servers Start features and click on Start I/O Data Servers and Start WebHMI.
   Note: if you are making changes to the Movicon project and re-deploying the new version onto CPL410, make sure to stop both I/O Data Server and WebHMI and start them again so that changes take effect.
   Note: please be cautious when using the **Start Servers** button to start all servers (instead of starting each server individually, as described above). It was observed that this might lead to new instances of the servers being started without replacing the old versions.
5. Scroll down, expand **Remote Device** Info, and make sure both servers and one instance of each server are running.
   Note: if you are changing from one Movicon project to another, please make sure to stop I/O Data Server and WebHMI (from within the old project), then download the new project and start servers. Failure to do so will result in multiple instances of WebHMI and I/O Data Server running, leading to unpredictable behavior.
6. You can also check the license status from the same Deploy Project window. Expand Remote License Info and then click on Check Remote License. The CPL410 license will look like the following:

Figure 154: License Found



Ok, Licence Found!

Expiring Date : Unlimited

Serial Number : 2

License Type: Embedded

Local Site Code wbiTFjYjExbAka6FNp8qQQ==

At this point, go to the PACEdge landing page and refresh the browser. WebHMI Icon should become active, bringing you to the WebHMI screen.

Verifying OPC UA Servers

At this point, we have two OPC UA Servers active, one in CPL410 PACs Controller and one in PACEdge I/O Data Server. We can verify both by using an OPC UA Client on the workstation, such as UA Expert. In both cases, you will see the Sweep Time variable and can observe value changes:

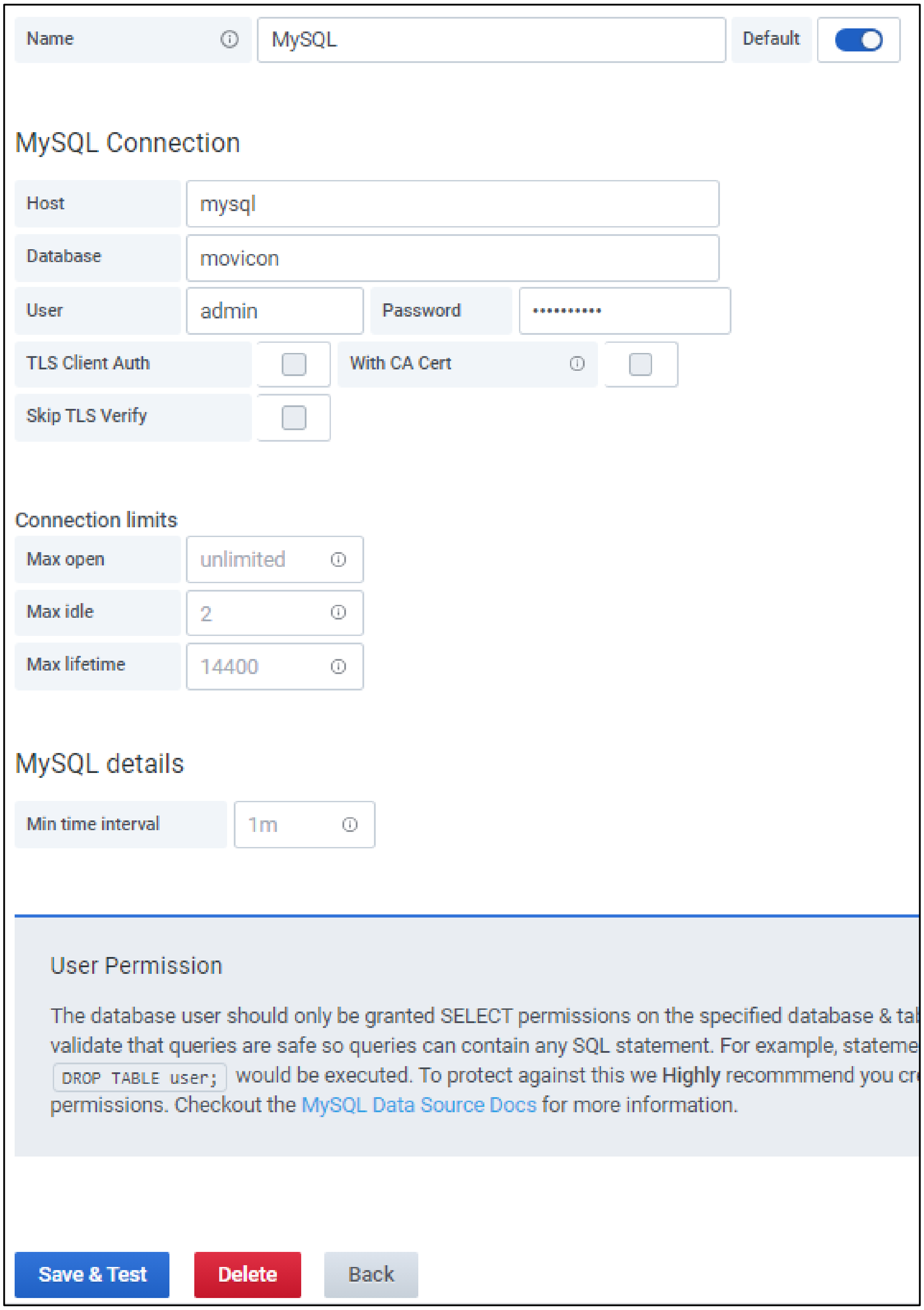
- To connect to OPC UA Server in PACs Controller, either:
    - Enable the OPC-UA Port Forwarding feature, the open UA Expert, and connect to opc.tcp://*ip_address_ETH_port*:4840, where *ip_address_ETH_port* is an IP address of ETH port
    - Or, ensure the Ethernet cable is plugged into the LAN1 port on the controller, and the workstation is configured to communicate on the 192.168.0.x subnet. Open UA Expert and connect to opc.tcp://192.168.0.100:4840
- To connect to OPC UA Server in PACEdge I/O Data Server, ensure the Ethernet cable is plugged into the ETH port on the controller and the workstation is configured to communicate on the subnet you have configured PACEdge for (by default: 192.168.3.x). Open UA Expert and connect to opc.tcp://192.168.3.100:62841

Visualizing via Grafana

To visualize the Sweep Time variable in Grafana:

1. From the PACEdge landing page, click on the **Grafana** link and log in as an admin.
2. Move the cursor over the Configuration icon on the left side and select **Data Sources.**
3. Click on **Add Data Source** and select **MySQL**.
4. Enter values as shown and the admin password, and click on Save & Test at the bottom. You should get a message **Database Connection OK.**

Figure 155: MySQL



Name ⓘ: MySQL
Default: ●

MySQL Connection

Host: mysql
Database: movicon
User: admin
Password: ••••••••••
TLS Client Auth: ☐
With CA Cert ⓘ: ☐
Skip TLS Verify: ☐

Connection limits

Max open: unlimited ⓘ
Max idle: 2 ⓘ
Max lifetime: 14400 ⓘ

MySQL details

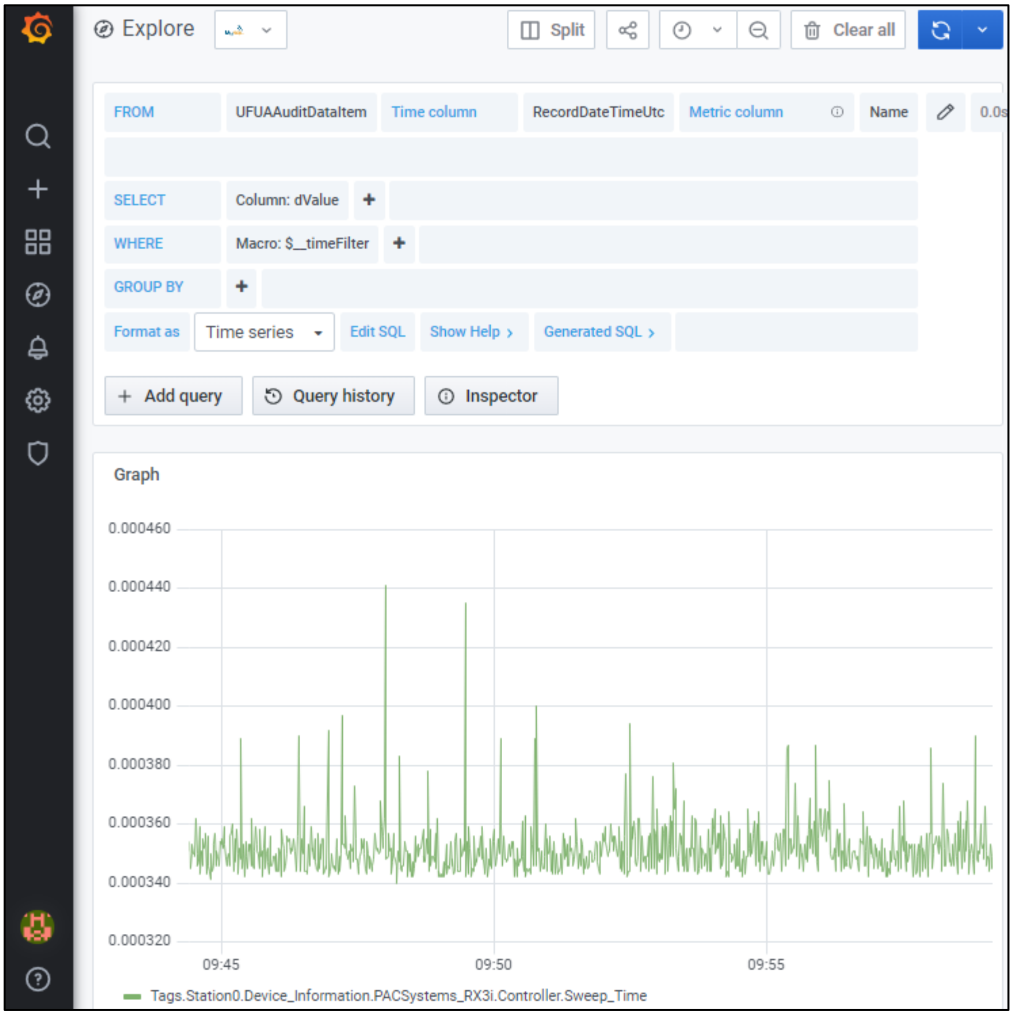
Min time interval: 1m ⓘ

User Permission

The database user should only be granted SELECT permissions on the specified database & tal
validate that queries are safe so queries can contain any SQL statement. For example, stateme
`DROP TABLE user;` would be executed. To protect against this we **Highly** recommmend you cr
permissions. Checkout the MySQL Data Source Docs for more information.

5. On the left sidebar, click on the **Explore** icon.
6. Make sure the newly configured MySQL database is selected, then:
    a. In FROM entry: make sure **UFUAAuditDataItem** is selected (default)
    b. In Time column entry: choose what time stamp should be used, can leave at default
    c. In Metric column entry: select **Name**
    d. In SELECT entry: select **dValue**
    e. You should see the historical values of the Sweep Time data tag.

Figure 156: Sweep Time Data



FROM UFUAAuditDataItem Time column RecordDateTimeUtc Metric column Name 0.0s

SELECT Column: dValue

WHERE Macro: $__timeFilter

GROUP BY

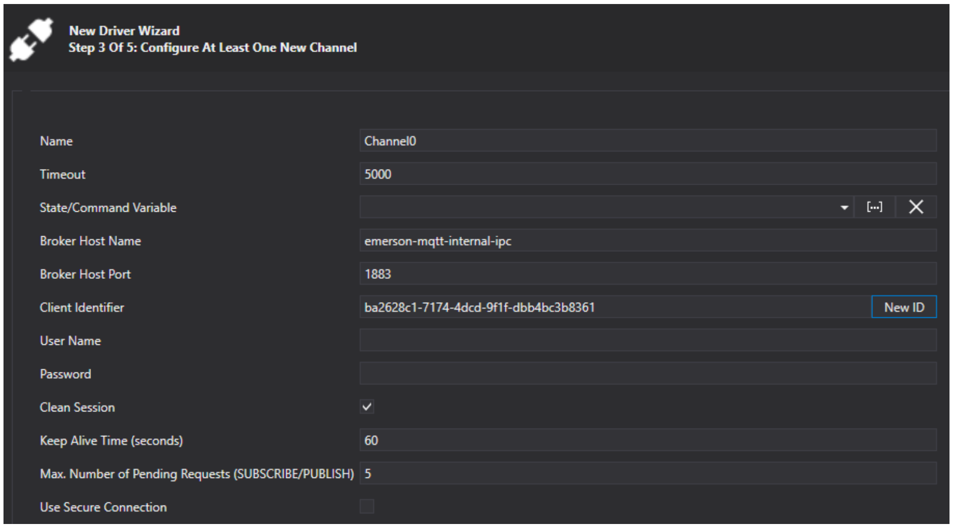
Format as Time series

Storing Connext Data in InfluxDB via Telegraf

Connext can historize and store data directly in MySQL database; however, MySQL is a relational database, not well suited to high volumes of time series data. Telegraf is already pre-configured to store data automatically into InfluxDB. The following steps show how to extend the Connext configuration to store data in InfluxDB.

1. In the Movicon project, add an MQTT driver by double-clicking on the **Drivers** (left side of the screen), then right-click in the main window, select **Add New Driver**, scroll the list and select **MQTT Client** driver.
    a. While configuring the MQTT driver, in step 3, enter Broker Host Name: **emerson-mqtt-internal-ipc**, then click on the **New ID** button to generate Client Identifier. Click on **>** to go to the next page.

Figure 157: MQTT Driver Configuration

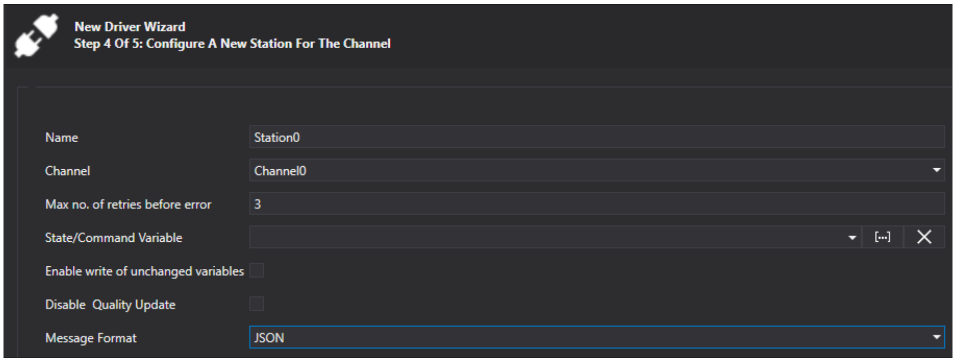


New Driver Wizard
Step 3 Of 5: Configure At Least One New Channel

Name	Channel0
Timeout	5000
State/Command Variable	
Broker Host Name	emerson-mqtt-internal-ipc
Broker Host Port	1883
Client Identifier	ba2628c1-7174-4dcd-9f1f-dbb4bc3b8361
User Name	
Password	
Clean Session	☑
Keep Alive Time (seconds)	60
Max. Number of Pending Requests (SUBSCRIBE/PUBLISH)	5
Use Secure Connection	□

   b. In step 4, select **JSON** for Message Format, then **>** and **Finish**.

Figure 158: Select JSON Message



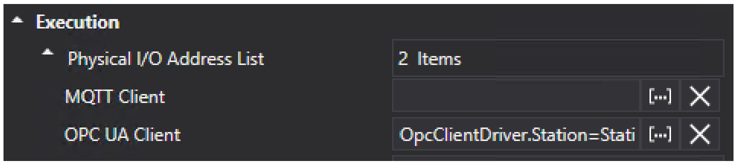
New Driver Wizard
Step 4 Of 5: Configure A New Station For The Channel

Name	Station0
Channel	Channel0
Max no. of retries before error	3
State/Command Variable	
Enable write of unchanged variables	□
Disable Quality Update	□
Message Format	JSON

2. In Movicon Editor, navigate to your tag **Sweep_Time** and double-click on it to open Properties

3. In tag properties, Execution section, expand Physical I/O Address List and click on … next to MQTT Client entry

Figure 159: Execution Section



Execution

Physical I/O Address List	2 Items
MQTT Client	
OPC UA Client	OpcClientDriver.Station=Stati

4. In the MQTT Client Dynamic Settings dialogue, enter the following:

   a. Topic Name: **emr_v1/Connext/Device_Information/PACSystems_RX3i/Controller/Sweep_Time**
   Note: it is mandatory that the topic starts with **emr_v1** and has at least 3, but no more than eight segments.

   b. In the Station field, select the newly created MQTT Client station; by default, it is **Station0**

   c. Put in Json Field for Tag value: **value**

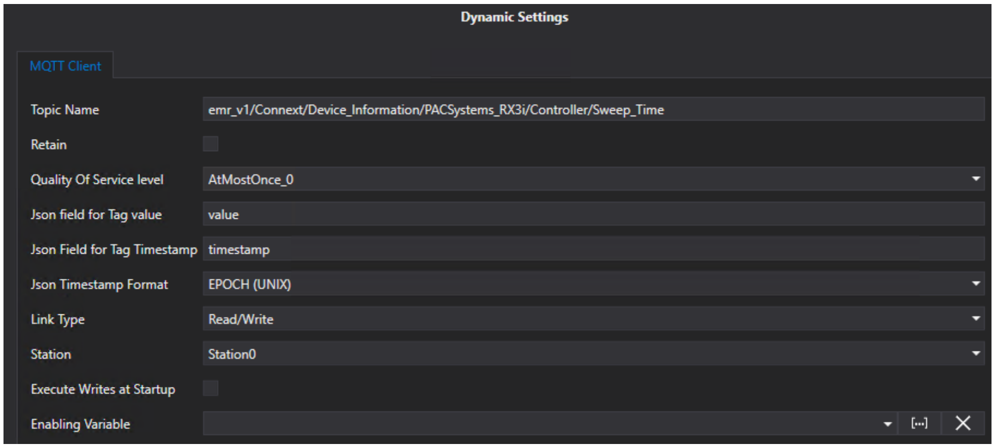
   d. Put in Json Field for Tag Timestamp: **timestamp**

   e. Select in Json Timestamp Format: **EPOCH(UNIX)**

   f. Select in Link Type: Read/Write

   g. Click OK and then check the mark to Accept Changes

Figure 160: Dynamic Settings of MQTT Client



Dynamic Settings

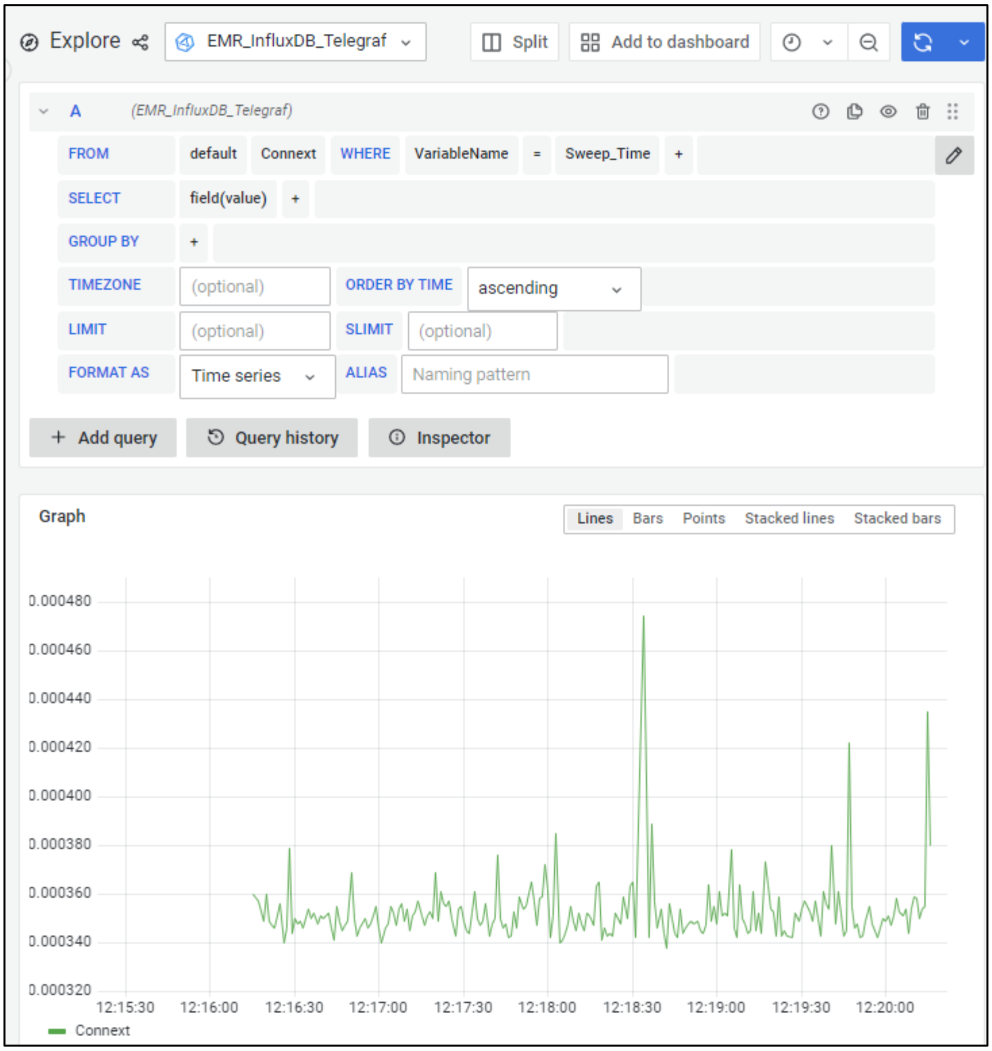
MQTT Client

Topic Name	emr_v1/Connext/Device_Information/PACSystems_RX3i/Controller/Sweep_Time
Retain	□
Quality Of Service level	AtMostOnce_0
Json field for Tag value	value
Json Field for Tag Timestamp	timestamp
Json Timestamp Format	EPOCH (UNIX)
Link Type	Read/Write
Station	Station0
Execute Writes at Startup	□
Enabling Variable	

5. Upload the newly modified project to the CPL410 PACEdge device by following the same steps as in

6. Deploy Movicon Project to CPL410 PACEdge Controller
7. Open Grafana and select **Explore** option on the left side of the window
8. Select the pre-configured **EMR_InfluxDB_Telegraf** database
9. In the FROM field, select **Connext** as your Measurement name
10. In the SELECT field, choose **value**. You will see the Sweep Time values graphically shown. Feel free to experiment with WHERE fields, where you can find the variable name and other information about the data tag that was configured in Connext

Figure 161: EMR-InfluxDB_Telegraf



A (EMR_InfluxDB_Telegraf)

FROM default Connext WHERE VariableName = Sweep_Time +

SELECT field(value) +

GROUP BY +

TIMEZONE (optional) ORDER BY TIME ascending

LIMIT (optional) SLIMIT (optional)

FORMAT AS Time series ALIAS Naming pattern

5.7 Configuring Telegraf to Log Data into InfluxDB

5.7.1 Example Description

PACEdge implements a Telegraf agent, which is plugin-based and can pull data, statistics, and logs from a variety of databases, underlying system resources (CPU, memory, disc, kernel, software logs, docker containers, etc.), and external devices, heavily focusing on IoT protocols, such as MQTT, AMQP, Cloud resources, etc. For the complete list of available plug-ins, please refer to: https://www.influxdata.com/products/integrations/?_integrations_dropdown=telegraf-plugins

Telegraf runs in his container and has a configuration file, which users can access and modify based on the requirements.

Staring with PACEdge v2.2.0, Telegraf comes pre-configured to:

- Pull system statistics and health information and store it in InfluxDB database **telegraf_metrics**, in Grafana mapped under the name: **EMR_InfluxDB_Telegraf database**
- Subscribe to Internal MQTT broker, all topics starting with: emr_v1/, parse the message and store data in InfluxDB database **telegraf_metrics**, in Grafana mapped under the name: **EMR_InfluxDB_Telegraf database**

5.7.2 Prerequisites

If new to Node-RED, completing 5.1 *Getting Started with Node-RED* and 5.4, *Using MQTT Communication* is highly recommended, as it shows some Node-RED usage basics and tips.

5.7.3 Example Walk Through

Steps to access the Telegraf configuration file:

1. From the PACEdge landing page, open Cockpit and log in as an admin user.
2. Go to Navigator, navigate to /home/admin/pacedge/Emerson-telegraf/ folder, and open telegraf.conf file

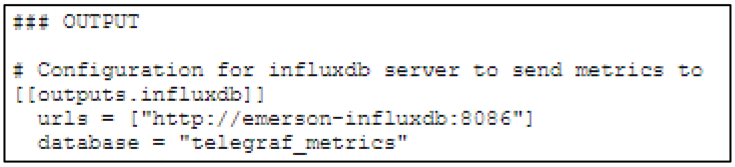
If making configuration changes, save the changed telegraf.conf file and restart the Telegraf container using either Cockpit-> Docker Containers or Portainer. Alternatively, the whole IPC can be rebooted.

Storing System Statistics in InfluxDB

By default, Telegraf is configured to pull the system it runs on statistics and dump them into InfluxDB every 10 seconds. Password to gain access to InfluxDB is now configured by the Password Management utility, so please make no password changes in this file.

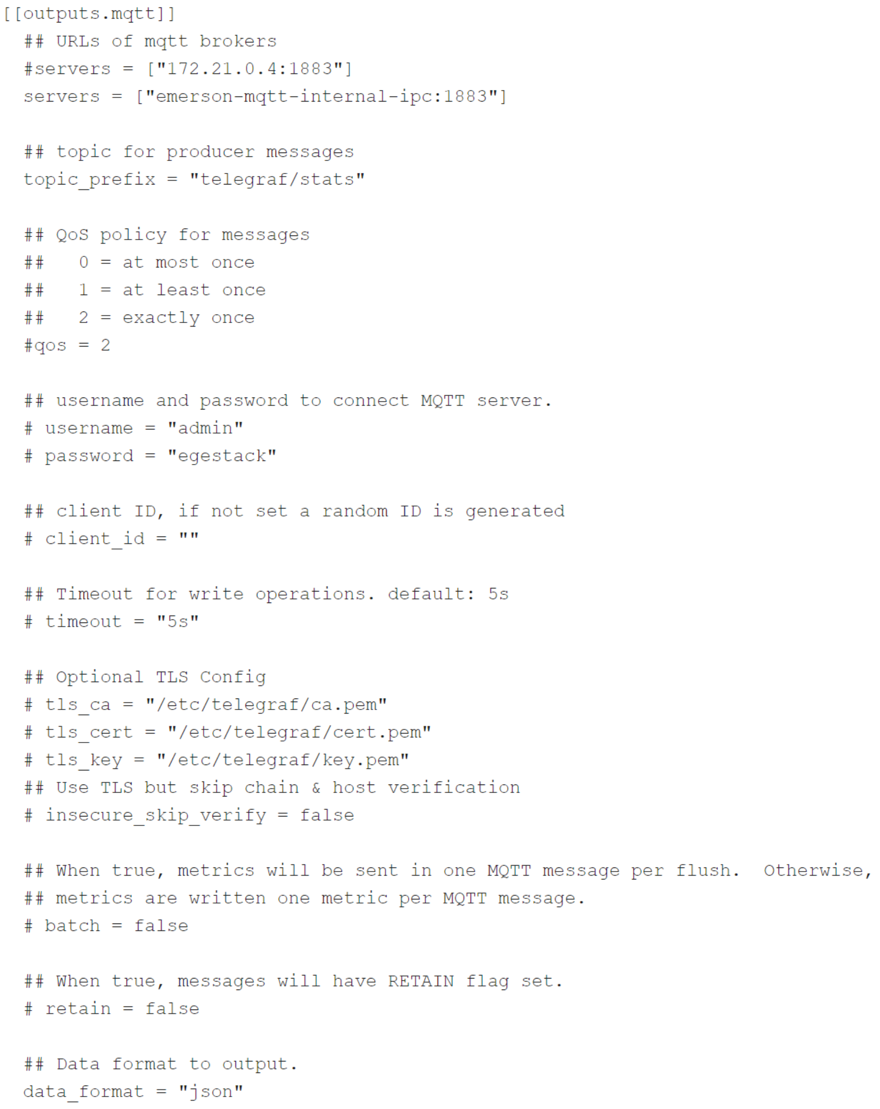
By default, Telegraf is configured to use InfluxDB database **telegraf_metrics** (which in Grafana is mapped to EMR_InfluxDB_Telegraf), where all data will be saved.

Figure 162: Telegraf Configuration File, database setting



```
### OUTPUT

# Configuration for influxdb server to send metrics to
[[outputs.influxdb]]
  urls = ["http://emerson-influxdb:8086"]
  database = "telegraf_metrics"
```

In addition to storing system's health data in InfluxDB, Telegraf can also be configured to publish gathered data to MQTT. Per default this functionality is disabled, but if desired, can be configured and enabled. To do so, in the telegraf.conf file, there is an **Output MQTT** section, which is currently commented out but can be enabled by the user. Once enabled, all data that is being stored in InfluxDB will also be published via MQTT.

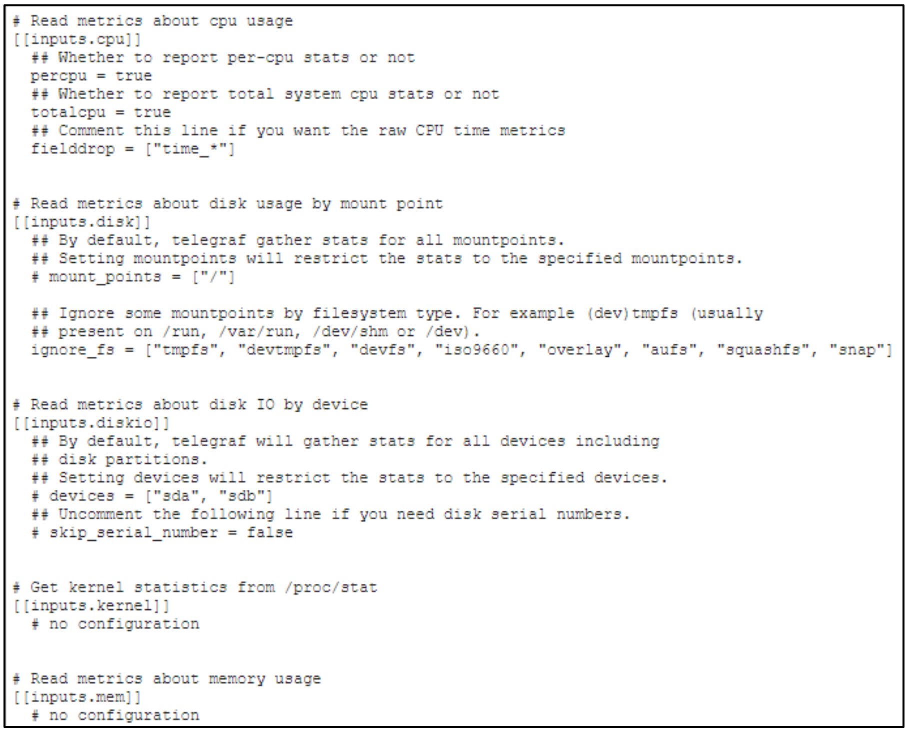


```
[[outputs.mqtt]]
  ## URLs of mqtt brokers
  #servers = ["172.21.0.4:1883"]
  servers = ["emerson-mqtt-internal-ipc:1883"]

  ## topic for producer messages
  topic_prefix = "telegraf/stats"

  ## QoS policy for messages
  ##   0 = at most once
  ##   1 = at least once
  ##   2 = exactly once
  #qos = 2

  ## username and password to connect MQTT server.
  # username = "admin"
  # password = "egestack"

  ## client ID, if not set a random ID is generated
  # client_id = ""

  ## Timeout for write operations. default: 5s
  # timeout = "5s"

  ## Optional TLS Config
  # tls_ca = "/etc/telegraf/ca.pem"
  # tls_cert = "/etc/telegraf/cert.pem"
  # tls_key = "/etc/telegraf/key.pem"
  ## Use TLS but skip chain & host verification
  # insecure_skip_verify = false

  ## When true, metrics will be sent in one MQTT message per flush.  Otherwise,
  ## metrics are written one metric per MQTT message.
  # batch = false

  ## When true, messages will have RETAIN flag set.
  # retain = false

  ## Data format to output.
  data_format = "json"
```

Figure 163 Telegraf configuration file enabling MQTT output (optional)

Further down in the config file is a section on what inputs should be captured and stored in InfluxDB.

Figure 164 Defining Inputs to be Captured and Stored in InfluxDB (partial list)



```
# Read metrics about cpu usage
[[inputs.cpu]]
  ## Whether to report per-cpu stats or not
  percpu = true
  ## Whether to report total system cpu stats or not
  totalcpu = true
  ## Comment this line if you want the raw CPU time metrics
  fielddrop = ["time_*"]


# Read metrics about disk usage by mount point
[[inputs.disk]]
  ## By default, telegraf gather stats for all mountpoints.
  ## Setting mountpoints will restrict the stats to the specified mountpoints.
  # mount_points = ["/"]

  ## Ignore some mountpoints by filesystem type. For example (dev)tmpfs (usually
  ## present on /run, /var/run, /dev/shm or /dev).
  ignore_fs = ["tmpfs", "devtmpfs", "devfs", "iso9660", "overlay", "aufs", "squashfs", "snap"]


# Read metrics about disk IO by device
[[inputs.diskio]]
  ## By default, telegraf will gather stats for all devices including
  ## disk partitions.
  ## Setting devices will restrict the stats to the specified devices.
  # devices = ["sda", "sdb"]
  ## Uncomment the following line if you need disk serial numbers.
  # skip_serial_number = false


# Get kernel statistics from /proc/stat
[[inputs.kernel]]
  # no configuration


# Read metrics about memory usage
[[inputs.mem]]
  # no configuration
```

By default, you can see that CPU, disk usage, and disk performance statistics are enabled. Users can manipulate these settings and tune what statistics they are interested in.

Storing Data via MQTT in InfluxDB

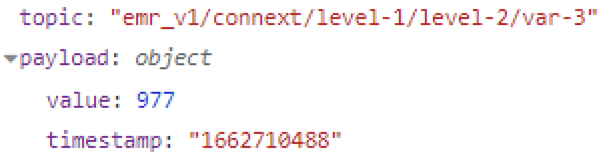
By default, Telegraf is configured to subscribe to the internal MQTT broker: **emerson-mqtt-internal-ipc**, topics starting with **emr_v1/**. Once a message with the matching topic is received, Telegraf will parse the rest of the topic, attempting to extract the InfluxDB Measurement name, Tags, and Fields. The following data parsing algorithm is applied:

- Measurement: next topic field after emr_v1/
- VariableName tag: last field in the topic
- Level-0 through Level-4 tags: fields in between Measurement and VariableName

MQTT message itself needs to be a JSON object, consisting of **value** and **timestamp**, where value is a number and timestamp is in seconds.

An example:

The following MQTT message would be parsed and stored in InfluxDB as follows:



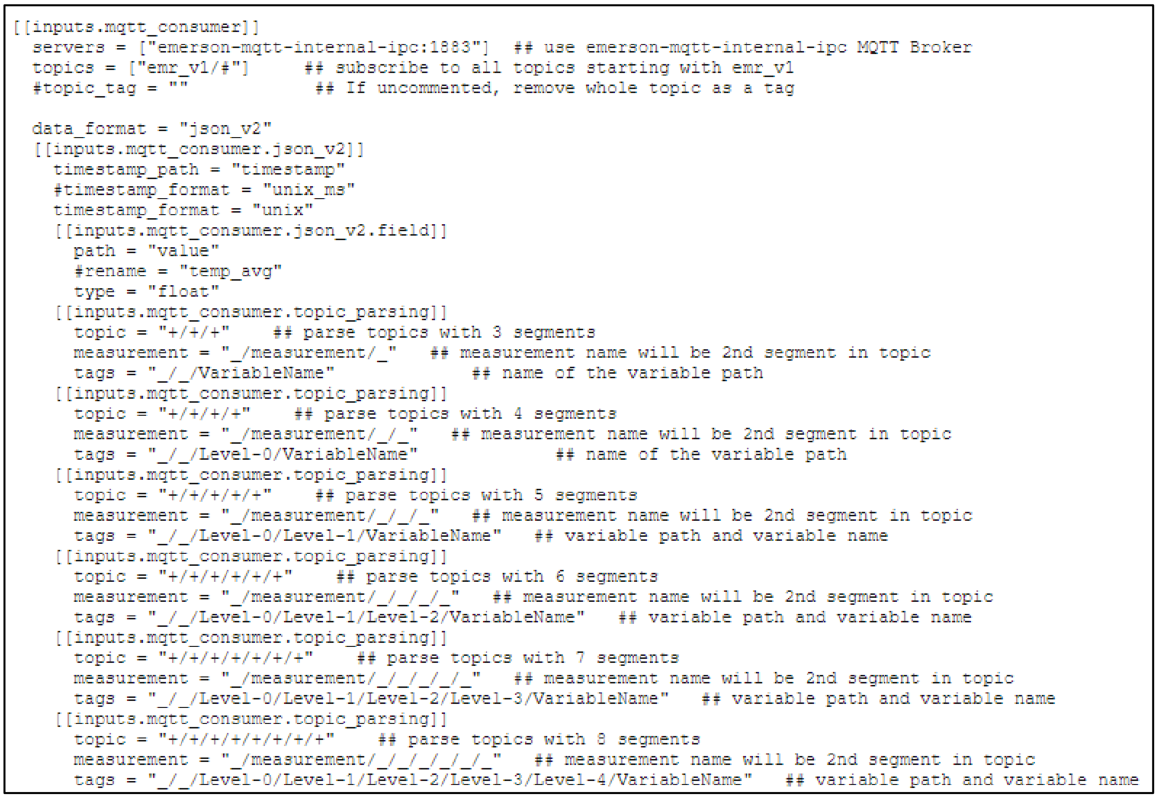
```
 topic: "emr_v1/connext/level-1/level-2/var-3"
▾payload: object
   value: 977
   timestamp: "1662710488"
```

- Measurement: **connext**
- Tag VariableName: **var-3**

- Tag Level-0: **level-1**
- Tag Level-1: **level-2**
- Timestamp: 1662710488
- Field: Value = **977**

Note that by default Telegraf is configured to parse MQTT topics that are a minimum of three and a maximum of eight segments long. This can be changed by editing the Telegraf configuration file

Figure 165 Telegraf Configuration File for MQTT to InfluxDB Logging



```
[[inputs.mqtt_consumer]]
  servers = ["emerson-mqtt-internal-ipc:1883"]  ## use emerson-mqtt-internal-ipc MQTT Broker
  topics = ["emr_v1/#"]     ## subscribe to all topics starting with emr_v1
  #topic_tag = ""            ## If uncommented, remove whole topic as a tag

  data_format = "json_v2"
  [[inputs.mqtt_consumer.json_v2]]
    timestamp_path = "timestamp"
    #timestamp_format = "unix_ms"
    timestamp_format = "unix"
    [[inputs.mqtt_consumer.json_v2.field]]
      path = "value"
      #rename = "temp_avg"
      type = "float"
    [[inputs.mqtt_consumer.topic_parsing]]
      topic = "+/+/+"    ## parse topics with 3 segments
      measurement = "_/measurement/_"   ## measurement name will be 2nd segment in topic
      tags = "_/_/VariableName"             ## name of the variable path
    [[inputs.mqtt_consumer.topic_parsing]]
      topic = "+/+/+/+"    ## parse topics with 4 segments
      measurement = "_/measurement/_/_"   ## measurement name will be 2nd segment in topic
      tags = "_/_/Level-0/VariableName"             ## name of the variable path
    [[inputs.mqtt_consumer.topic_parsing]]
      topic = "+/+/+/+/+"    ## parse topics with 5 segments
      measurement = "_/measurement/_/_/_"   ## measurement name will be 2nd segment in topic
      tags = "_/_/Level-0/Level-1/VariableName"   ## variable path and variable name
    [[inputs.mqtt_consumer.topic_parsing]]
      topic = "+/+/+/+/+/+"    ## parse topics with 6 segments
      measurement = "_/measurement/_/_/_/_"   ## measurement name will be 2nd segment in topic
      tags = "_/_/Level-0/Level-1/Level-2/VariableName"   ## variable path and variable name
    [[inputs.mqtt_consumer.topic_parsing]]
      topic = "+/+/+/+/+/+/+"    ## parse topics with 7 segments
      measurement = "_/measurement/_/_/_/_/_"   ## measurement name will be 2nd segment in topic
      tags = "_/_/Level-0/Level-1/Level-2/Level-3/VariableName"   ## variable path and variable name
    [[inputs.mqtt_consumer.topic_parsing]]
      topic = "+/+/+/+/+/+/+/+"    ## parse topics with 8 segments
      measurement = "_/measurement/_/_/_/_/_/_"   ## measurement name will be 2nd segment in topic
      tags = "_/_/Level-0/Level-1/Level-2/Level-3/Level-4/VariableName"   ## variable path and variable name
```

5.8 Sine Wave Simulator

Starting with PACEdge v2.3.0, Node-RED includes a Sine Wave Simulator example, which allows users to quickly generate time series data, store this data in InfluxDB database and visualize it in both Node-RED as well as Grafana dashboards.

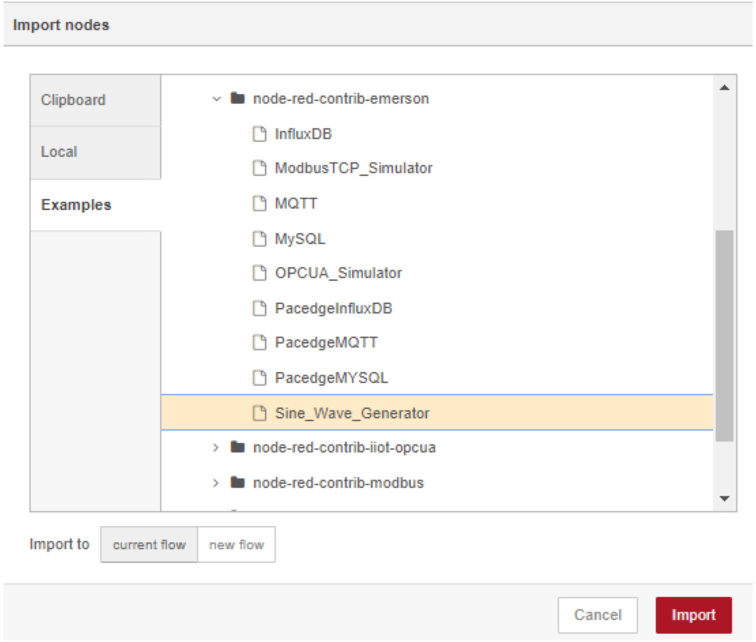
5.8.1 Prerequisites

PACEdge v2.3.0 or later

5.8.3 Example Walk Through

1. Open Node-RED, navigate to: Import->Examples->node-red-contrib-emerson, select **Sine_Wave_Generator** and click on Import button.

Figure 166 Sine Wave Simulator Import Dialogue



Import nodes

- Clipboard
- Local
- Examples

node-red-contrib-emerson
- InfluxDB
- ModbusTCP_Simulator
- MQTT
- MySQL
- OPCUA_Simulator
- PacedgeInfluxDB
- PacedgeMQTT
- PacedgeMYSQL
- Sine_Wave_Generator

node-red-contrib-iiot-opcua

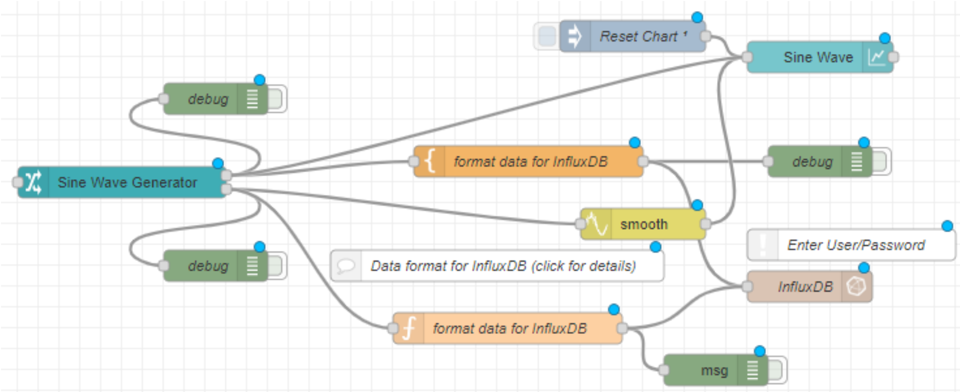
node-red-contrib-modbus

Import to: current flow | new flow

Cancel | Import

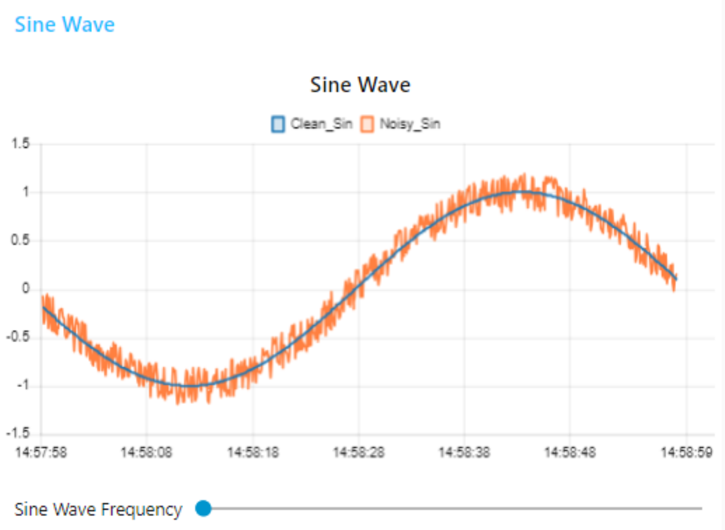
2. Resulting flow will be as shown below:

Figure 167 Sine Wave Simulator Node-RED flow



3. Double click on InfluxDB node, click on pencil next to the Server to open configuration dialogue. Enter username and password to log into InfluxDB. Click **Update**, then **Done**.
4. Click on **Deploy** to apply changes and start the flow.
5. Open Node-RED Dashboard (refer to section 5.1Node-RED Getting Started for details)

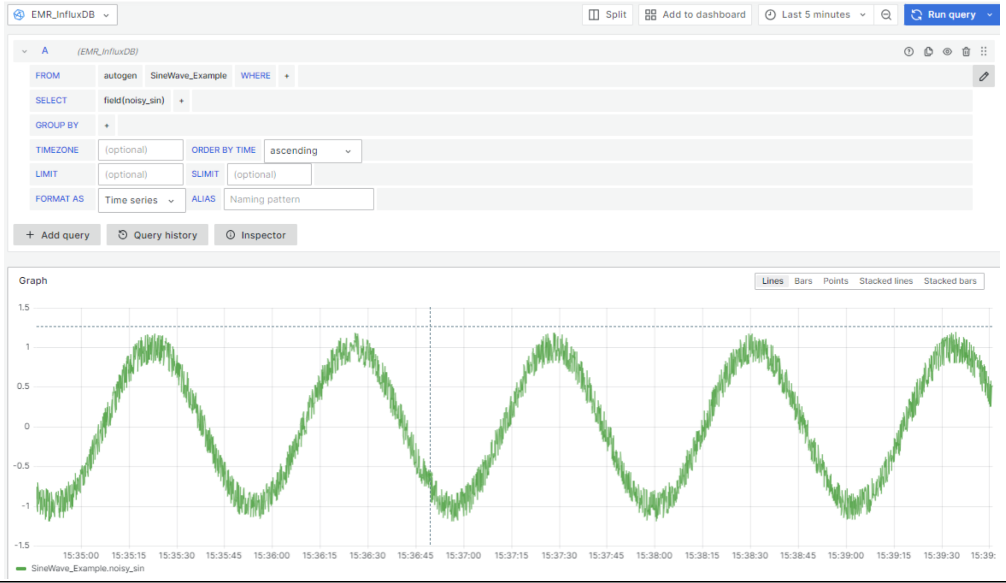
Figure 168 Node-RED Dasbhoard showing Sine Waves



6. Note, the blue line shows a clean sine wave and the orange line shows a sine wave with random noise applied to it. You can move the slider below and change the frequency of the sine wave.

7. To see the same data in Grafana, open Grafana, go to Explore, select EMR_InfluxDB database, for measurement select SineWave_Example, then choose a field (either clean_sin or noisy_sin) and observe the data

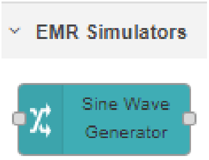
Figure 169 Noisy Sine Wave as seen in Grafana



FROM autogen SineWave_Example WHERE +
SELECT field(noisy_sin) +
GROUP BY +
TIMEZONE (optional) ORDER BY TIME ascending
LIMIT (optional) SLIMIT (optional)
FORMAT AS Time series ALIAS Naming pattern

8. Note, when installing this example in Node-RED a Sine Wave Simulator sublow gets added to the Node-RED node list on the left side of the screen. Feel free to use this simulator in other flows. To see node usage details, select the simulator node and open Help side bar on the right side of the screen.

Figure 170 Sine Wave Simulator Node-RED node



5.9 ModbusTCP Simulator

Starting with PACEdge v2.3.0, Node-RED includes a ModbusTCP Simulator example, which allows users to get familiar with ModbusTCP Node-RED nodes   These nodes can be used to connect to external ModbusTCP enabled devices and gather time series data, store this data in InfluxDB database, and visualize it in both Node-RED as well as Grafana dashboards.

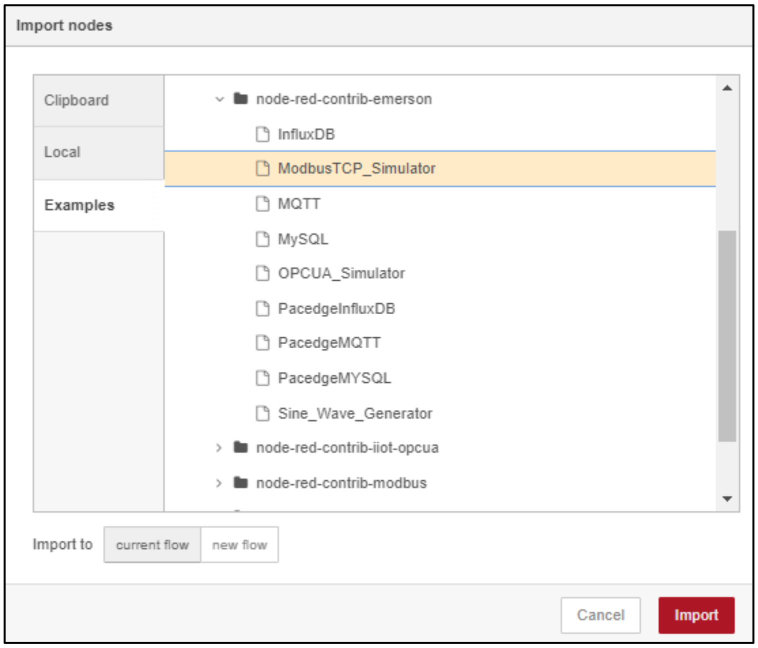
5.9.1 Prerequisites

PACEdge v2.3.0 or later

5.9.2 Example Walk Through

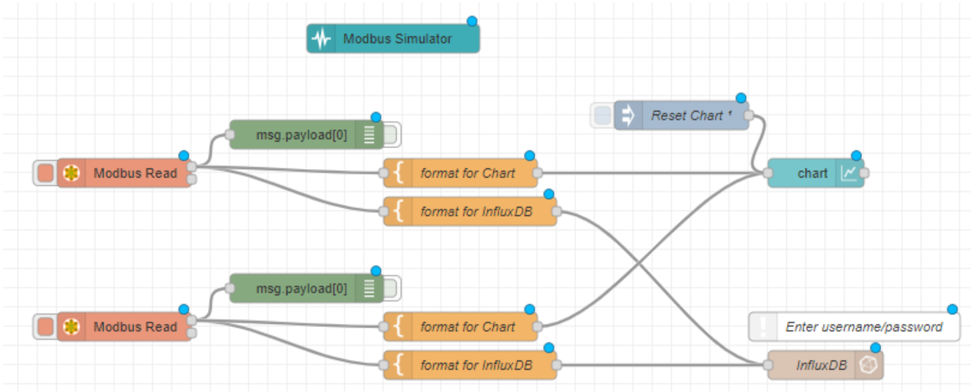
1. Open Node-RED, navigate to: Import->Examples->node-red-contrib-emerson, select **ModbusTCP_Simulator** and click on Import button.

Figure 171 ModbusTCP Simulator Import Dialogue



3. Resulting flow will be as shown below:

Figure 172 ModbusTCP Simulator Node-RED flow

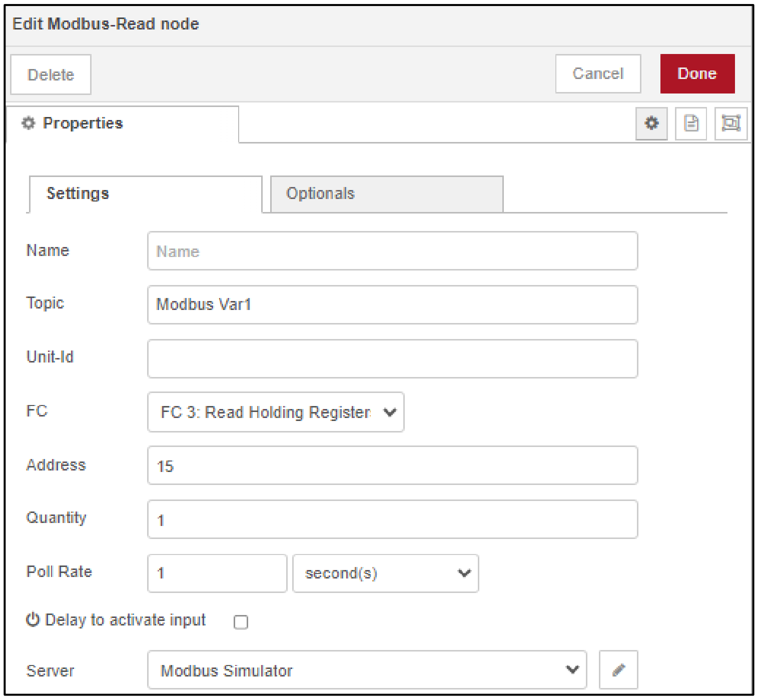


4. Double-click on InfluxDB node, click on pencil next to the Server to open configuration dialogue. Enter username and password to log into InfluxDB. Click **Update**, then **Done**.

5. Click on **Deploy** to apply changes and start the flow.

6. To see details of how ModbusTCP connection is setup, double click on two Modbus Read nodes and then on pencil icon next to the Server line

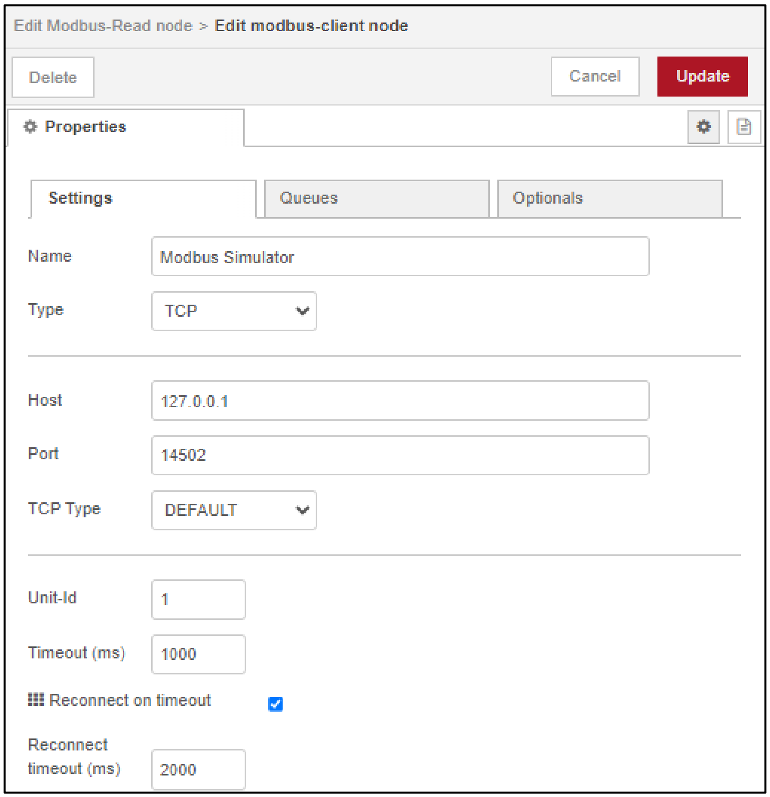
Figure 173 ModbusTCP Connection Settings



Edit Modbus-Read node

Name	Name
Topic	Modbus Var1
Unit-Id	
FC	FC 3: Read Holding Register
Address	15
Quantity	1
Poll Rate	1 second(s)
Delay to activate input	□
Server	Modbus Simulator

Figure 174 ModbusTCP Server Settings



Edit Modbus-Read node > **Edit modbus-client node**

Name: Modbus Simulator

Type: TCP

Host: 127.0.0.1

Port: 14502

TCP Type: DEFAULT

Unit-Id: 1

Timeout (ms): 1000

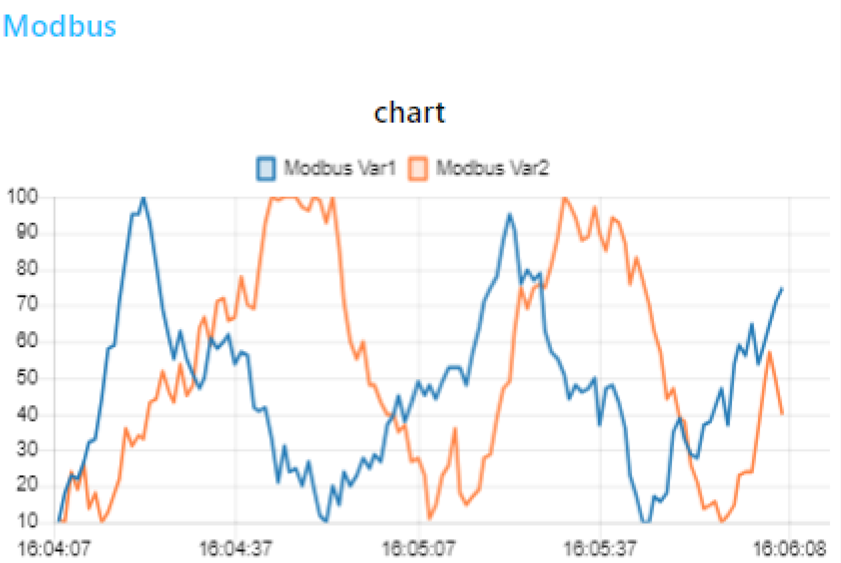
Reconnect on timeout ☑

Reconnect timeout (ms): 2000

a. To connect to your own ModbusTCP device, modify the Host IP address, Port number, as well as FC and Address settings.

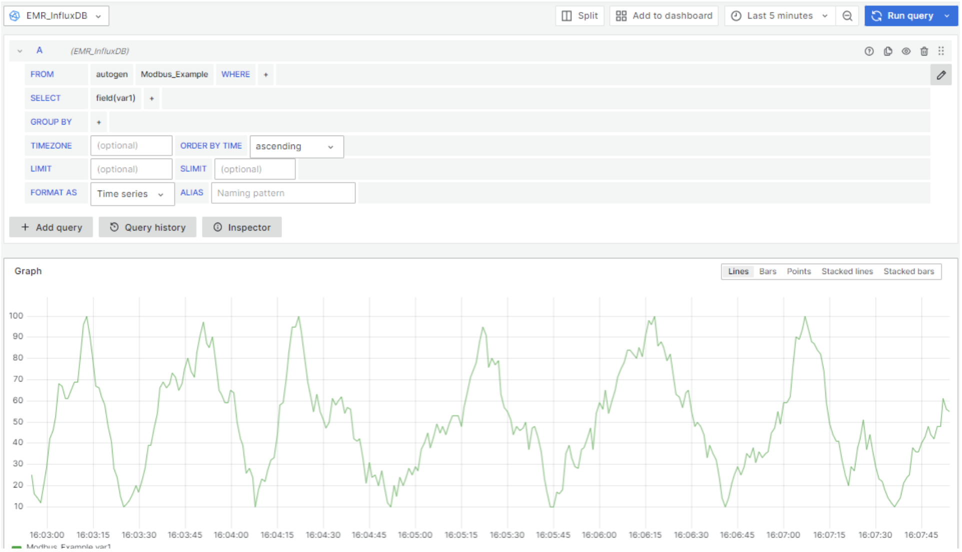
7. Open Node-RED Dashboard (refer to section 5.1Node-RED Getting Started for details)

Figure 175 Node-RED Dasbhoard showing two Modbus registers



8. To see the same data in Grafana, open Grafana, go to Explore, select EMR_InfluxDB database, for measurement select Modbus_Example, then choose a field (either var1 or var2) and observe the data

Figure 176 ModbusTCP register value as seen in Grafana



EMR_InfluxDB

A (EMR_InfluxDB)

FROM autogen Modbus_Example WHERE +

SELECT field(var1) +

GROUP BY +

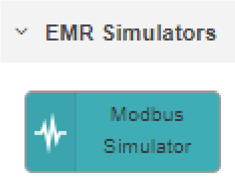
TIMEZONE (optional) ORDER BY TIME ascending

LIMIT (optional) SLIMIT (optional)

FORMAT AS Time series ALIAS Naming pattern

9. Note, when installing this example in Node-RED a Modbus Simulator sublow gets added to the Node-RED node list on the left side of the screen. Feel free to use this simulator in other flows. To see node usage details, select the simulator node and open Help side bar on the right side of the screen.

Figure 177 ModbusTCP Simulator Node-RED node



5.10 OPC-UA Simulator

Starting with PACEdge v2.3.0, Node-RED includes an OPC-UA Simulator example, which allows users to get familiar with OPC-UA Node-RED nodes. These nodes can be used to connect to external OPC-UA enabled devices and gather time series data, store this data in InfluxDB database, and visualize it in both Node-RED as well as Grafana dashboards.

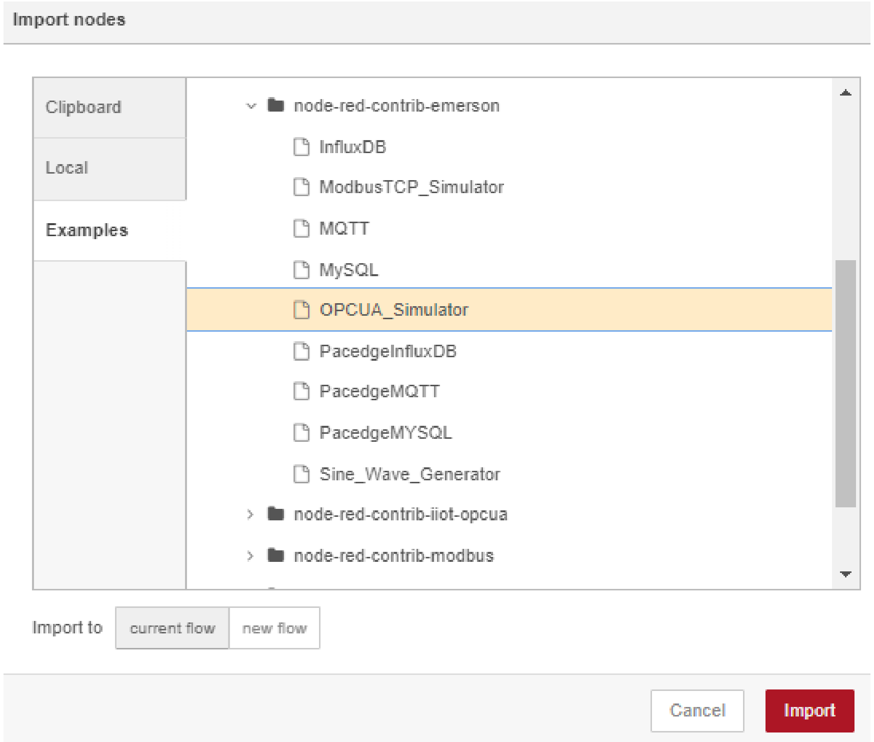
5.10.1 Prerequisites

PACEdge v2.3.0 or later

5.10.3 Example Walk Through

1. Open Node-RED, navigate to: Import->Examples->node-red-contrib-emerson, select **OPCUA_Simulator** and click on Import button.

Figure 178 Sine Wave Simulator Import Dialogue



Import nodes

Clipboard
Local
Examples

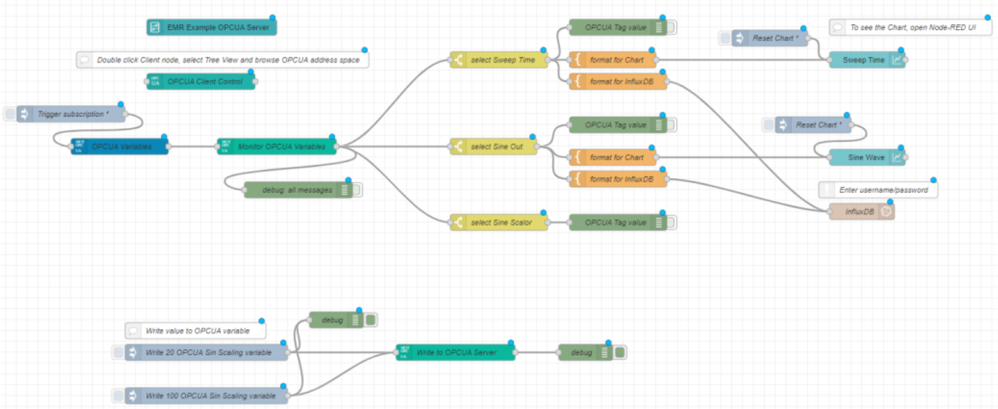
- node-red-contrib-emerson
  - InfluxDB
  - ModbusTCP_Simulator
  - MQTT
  - MySQL
  - OPCUA_Simulator
  - PacedgeInfluxDB
  - PacedgeMQTT
  - PacedgeMYSQL
  - Sine_Wave_Generator
- node-red-contrib-iiot-opcua
- node-red-contrib-modbus

Import to: current flow | new flow

Cancel | Import

2. Resulting flow will be as shown below:

Figure 179 OPC-UA Simulator Node-RED flow



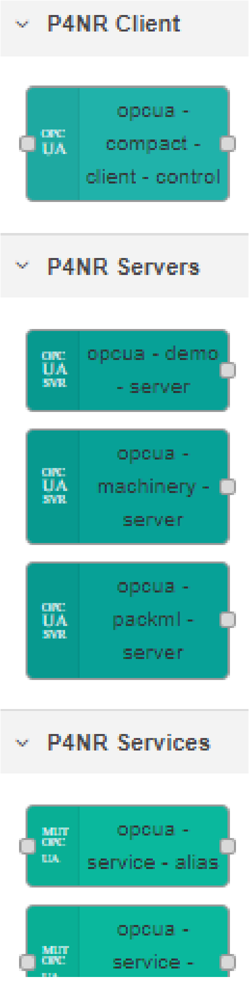
3. Double click on InfluxDB node, click on pencil next to the Server to open configuration dialogue. Enter username and password to log into InfluxDB. Click **Update**, then **Done**.

4. Click on **Deploy** to apply changes and start the flow.

6. To see OPC-UA connectivity details, explore the following nodes.

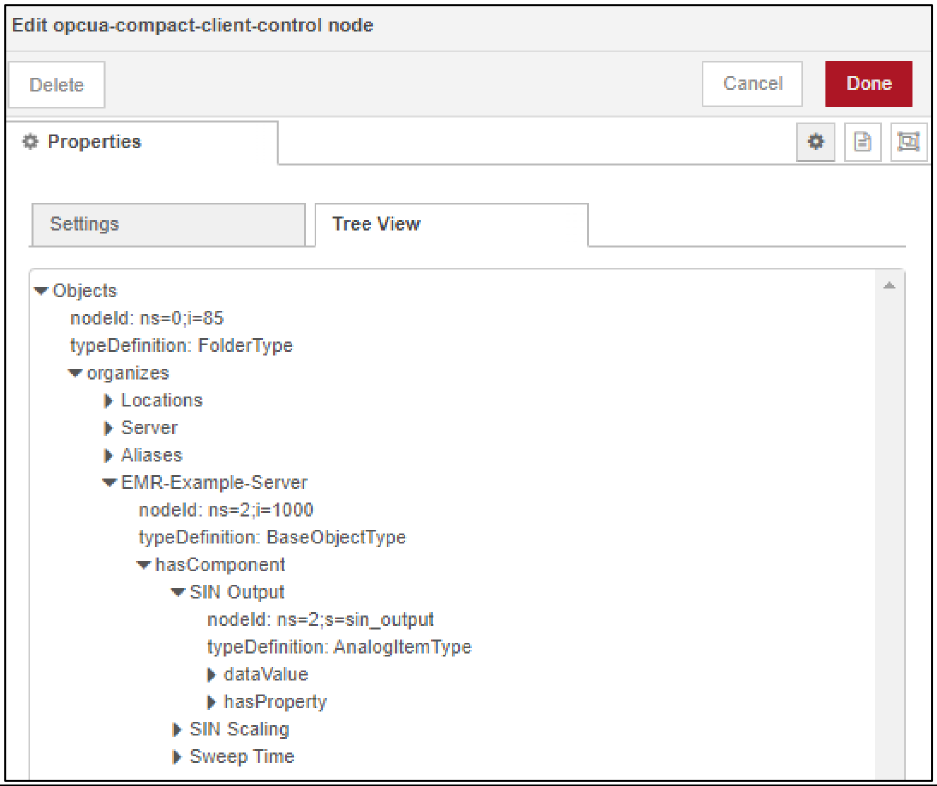
Note: The example uses a new OPC-UA node palette from Plus For Node-RED (P4NR). P4NR nodes are already pre-installed and can be seen in the node palette. Highlighting one of the nodes in the flow and then opening Help bar on the right side provides more usage information.

Figure 180 P4NR OPC-UA Node-RED nodes



8. To browse OPC-UA Server address space, double click on OPCUA Client Control node, then select Tree View and explore the address space:

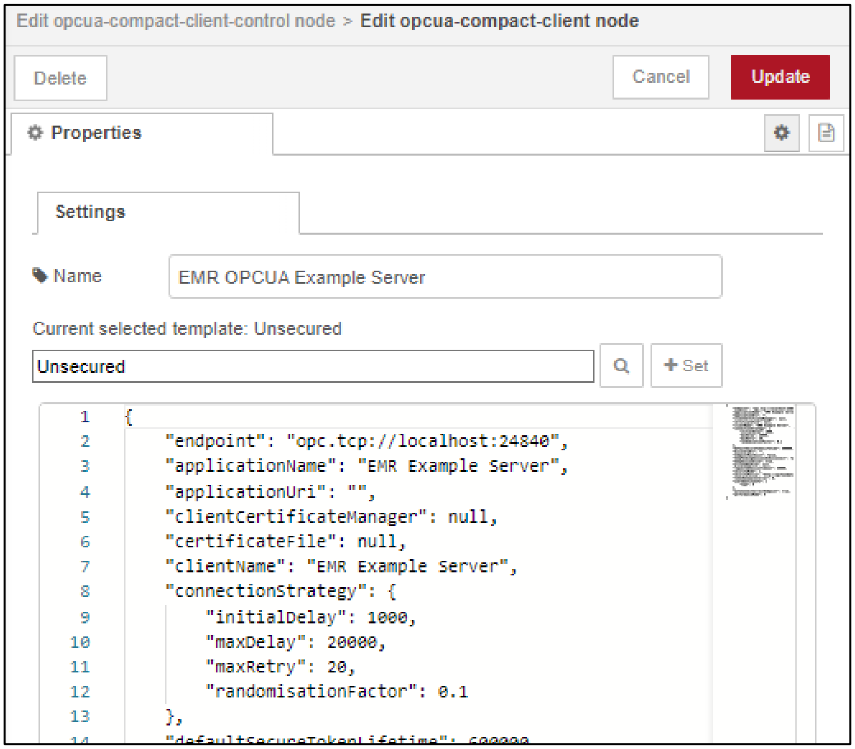
Figure 181 OPC-UA Address Space Browser View



```
▼ Objects
    nodeId: ns=0;i=85
    typeDefinition: FolderType
  ▼ organizes
     ▶ Locations
     ▶ Server
     ▶ Aliases
     ▼ EMR-Example-Server
          nodeId: ns=2;i=1000
          typeDefinition: BaseObjectType
        ▼ hasComponent
           ▼ SIN Output
                nodeId: ns=2;s=sin_output
                typeDefinition: AnalogItemType
              ▶ dataValue
              ▶ hasProperty
           ▶ SIN Scaling
           ▶ Sweep Time
```

10. To see the OPC-UA Server settings, click on the pencil icon next to the Client Node line. Note the "endpoint" line that defines OPC-UA server IP address and port number. You can change settings in this line to connect to your OPC-UA server of choice.

Figure 182 OPC-UA Server Connection Settings


Edit opcua-compact-client-control node > Edit opcua-compact-client node
Delete
Cancel
Update
Properties
Settings
Set

Name: EMR OPCUA Example Server

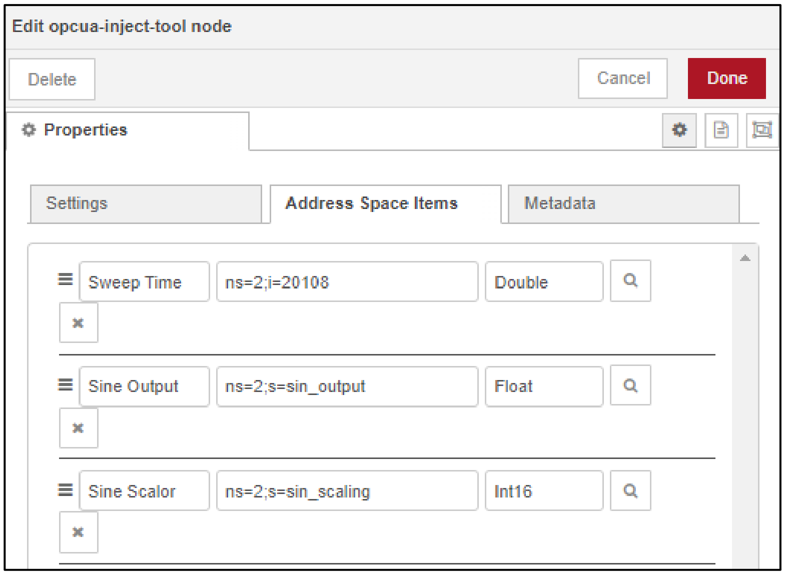
Current selected template: Unsecured

Unsecured

```
{
    "endpoint": "opc.tcp://localhost:24840",
    "applicationName": "EMR Example Server",
    "applicationUri": "",
    "clientCertificateManager": null,
    "certificateFile": null,
    "clientName": "EMR Example Server",
    "connectionStrategy": {
        "initialDelay": 1000,
        "maxDelay": 20000,
        "maxRetry": 20,
        "randomisationFactor": 0.1
    },
    "defaultSecureTokenLifetime": 600000
```

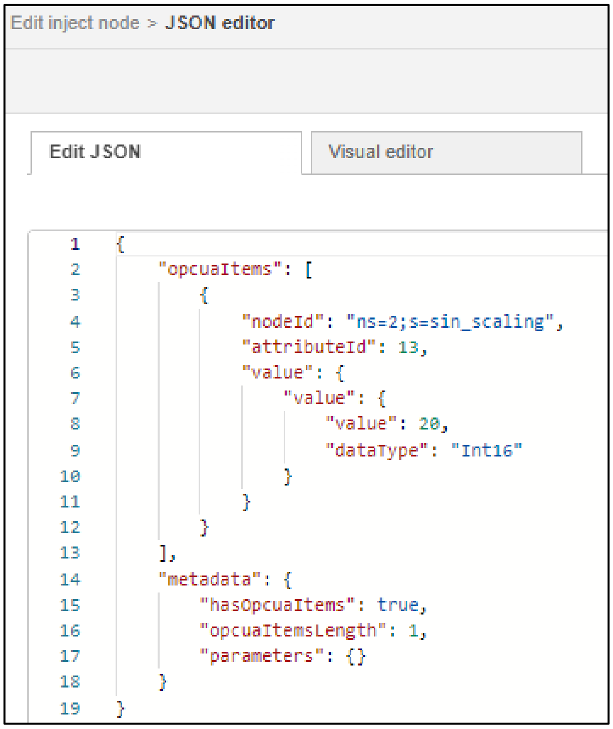
11. To see the OPC-UA variable settings, double click on blue OPCUA Variables icon, then click on Address Space Items

Figure 183 OPC-UA Variable Settings



Sweep Time	ns=2;i=20108	Double
Sine Output	ns=2;s=sin_output	Float
Sine Scalor	ns=2;s=sin_scaling	Int16

12. The bottom portion of the example flow contains a node to write a value to an OPC-UA variable. Click on the square button to the left of the Write… node in order to perform a write. Double click one of the Write xx OPCUA Sin Scaling variable nodes to see data object that is being sent to Write to OPCUA Server node.

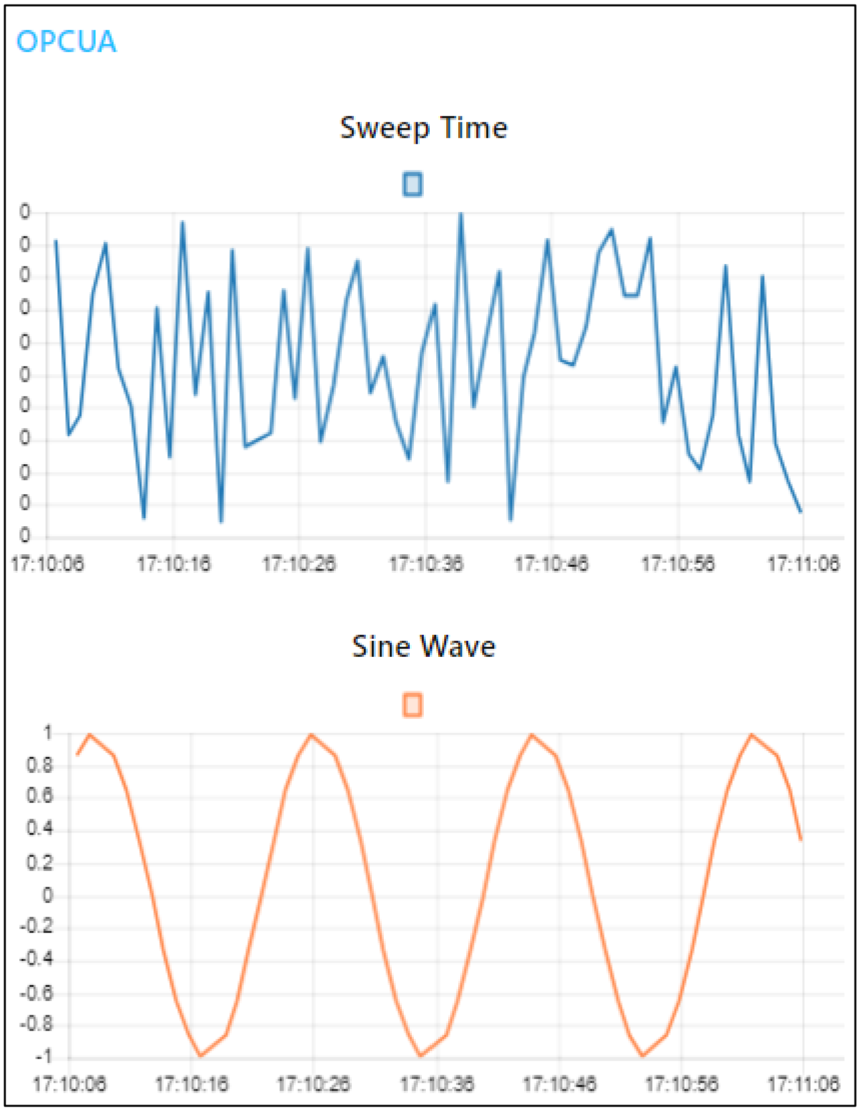


```
{
    "opcuaItems": [
        {
            "nodeId": "ns=2;s=sin_scaling",
            "attributeId": 13,
            "value": {
                "value": {
                    "value": 20,
                    "dataType": "Int16"
                }
            }
        }
    ],
    "metadata": {
        "hasOpcuaItems": true,
        "opcuaItemsLength": 1,
        "parameters": {}
    }
}
```

a. You can change the “value” field in order to write your value of choice.

14. Open Node-RED Dashboard (refer to section 5.1Node-RED Getting Started for details)

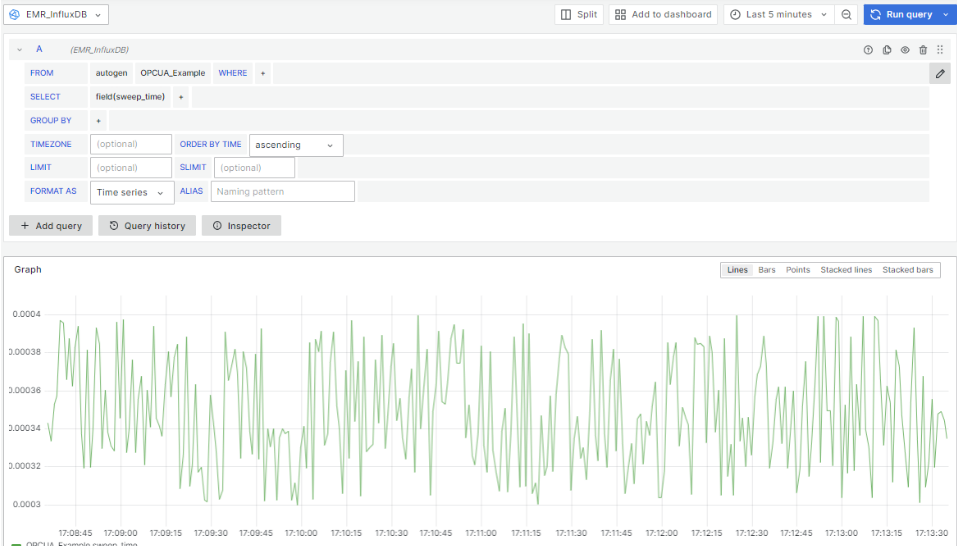
Figure 184 Node-RED Dasbhoard showing OPC-UA Simulator Data



15. Note the blue line shows simulated PLC sweep time and the orange line shows simulated Sine Wave data. Writing a different sine wave frequency factor into OPC-UA server will result in the sine wave frequency changing.

16. To see the same data in Grafana, open Grafana, go to Explore, select EMR_InfluxDB database, for measurement select OPCUA_Example, then choose a field (either sine_wave or sweep_time) and observe the data

Figure 185 Sweep Time data as seen in Grafana



EMR_InfluxDB

A (EMR_InfluxDB)

FROM autogen OPCUA_Example WHERE +

SELECT field(sweep_time) +

GROUP BY +

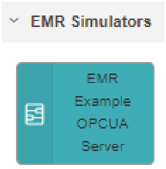
TIMEZONE (optional) ORDER BY TIME ascending

LIMIT (optional) SLIMIT (optional)

FORMAT AS Time series ALIAS Naming pattern

17. Note, when installing this example in Node-RED an OPC-UA Server Simulator sublow gets added to the Node-RED node list on the left side of the screen. Feel free to use this simulator in other flows.

Figure 186 OPC-UA Server Simulator Node-RED node



5.11 Python Container (optional)

Starting with v2.3.0, PACEdge includes an optional Python container. In a few simple steps, user can enable a Python container, which comes with an example application that demonstrates how Node-RED can be used to trigger python scripts based on tag values or some advanced logic. Note that to activate a Python container PACEdge device needs to have Internet access. This is required only during activation.

5.11.1 Prerequisites

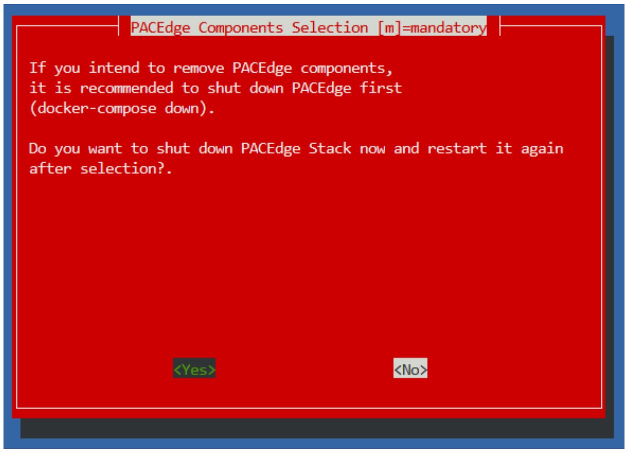
PACEdge v2.3.0 or later and Internet access during Python container activation.

5.11.2 Example Walk Through

1. Please make sure that PACEdge device has Internet connectivity.
2. Open Cockpit and navigate to Terminal, issue following commands:
    a. cd pacedge
    b. ./createcomposefile.sh -i

3. In the dialogue that asks if you want to shut down PACEdge stack, make sure Yes is highlighted and hit Enter key

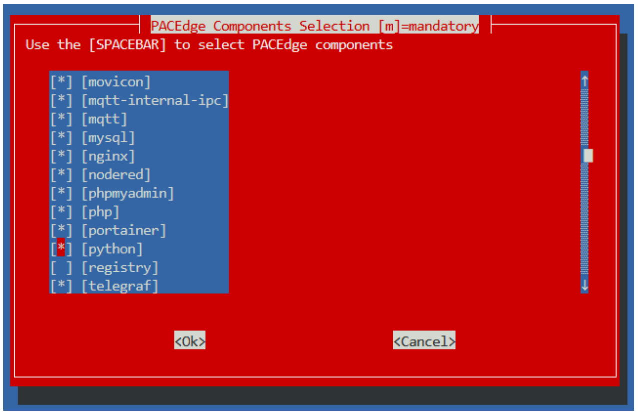
Figure 187 Dialogue asking to shut down PACEdge stack



```
If you intend to remove PACEdge components,
it is recommended to shut down PACEdge first
(docker-compose down).

Do you want to shut down PACEdge Stack now and restart it again
after selection?.
```

4. Scroll down till option python, use space bar to place a star, then Tab key to highlight OK and hit Enter key

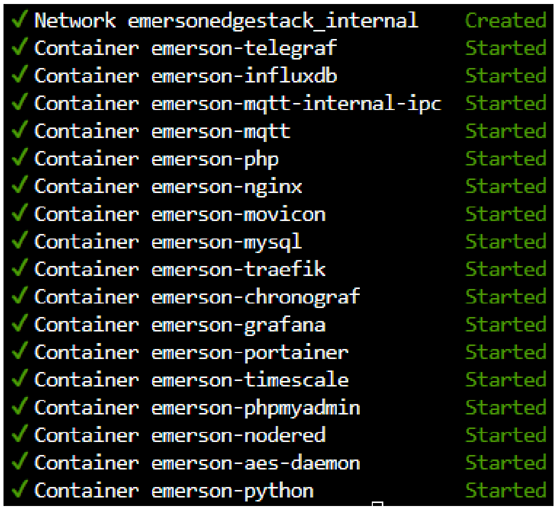
Figure 188 Dialogue to select Python container



```
Use the [SPACEBAR] to select PACEdge components

   [*] [movicon]
   [*] [mqtt-internal-ipc]
   [*] [mqtt]
   [*] [mysql]
   [*] [nginx]
   [*] [nodered]
   [*] [phpmyadmin]
   [*] [php]
   [*] [portainer]
   [*] [python]
   [ ] [registry]
   [*] [telegraf]
```

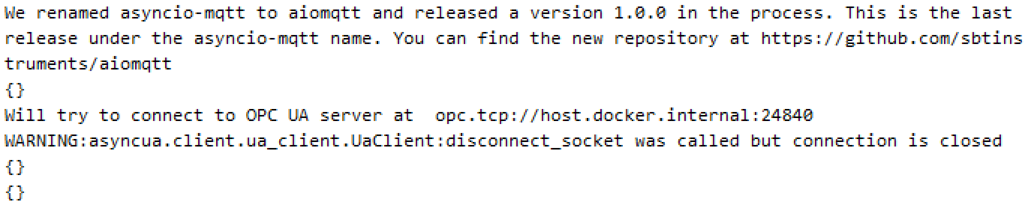
5. It will take few minutes to download all required components and bring back up the PACEdge stack. The final message will look like this (note emerson-python container in the list):

Figure 189 View of PACEdge Stack being brought up



```
✓ Network emersonedgestack_internal    Created
✓ Container emerson-telegraf           Started
✓ Container emerson-influxdb           Started
✓ Container emerson-mqtt-internal-ipc  Started
✓ Container emerson-mqtt               Started
✓ Container emerson-php                Started
✓ Container emerson-nginx              Started
✓ Container emerson-movicon            Started
✓ Container emerson-mysql              Started
✓ Container emerson-traefik            Started
✓ Container emerson-chronograf         Started
✓ Container emerson-grafana            Started
✓ Container emerson-portainer          Started
✓ Container emerson-timescale          Started
✓ Container emerson-phpmyadmin         Started
✓ Container emerson-nodered            Started
✓ Container emerson-aes-daemon         Started
✓ Container emerson-python             Started
```

6. Open Portainer, navigate to emerson-python container and check the logs. It will look like following:



```
We renamed asyncio-mqtt to aiomqtt and released a version 1.0.0 in the process. This is the last
release under the asyncio-mqtt name. You can find the new repository at https://github.com/sbtins
truments/aiomqtt
{}
Will try to connect to OPC UA server at  opc.tcp://host.docker.internal:24840
WARNING:asyncua.client.ua_client.UaClient:disconnect_socket was called but connection is closed
{}
{}
```

Note: per default, once Python container has been activated, every time it starts it will automatically execute **python_dispatcher_v2.py** python file, in directory: **/home/admin/pacedge/emerson-python/user_scripts**. This file is part of the example and is trying to connect to the OPCUA Server Simulator in Node-RED, which requires further steps to get activated and port 24840 to be opened. Please reference a more advanced example in PACEdge Knowledgebase portal.

7. Go back to Navigator in Cockpit, navigate to folder: **/home/admin/pacedge/emerson-python/user_scripts**, rename original **python_dispatcher_v2.py** file to some other name, then create new file with following line of python code in it:
print("Hello from Python")
and save the file under the name **python_dispatcher_v2.py**.

   a. Go back to Portainer, emerson-python container and click on Restart

   b. Now go to Logs and you will see message Hello from Python being printed

   c. At this point you can start creating your own python application and as long at it is saved under the name **python_dispatcher_v2.py** in the folder mentioned above, it will be automatically started everytime PACEdge stack is brought up.

   d. For a more advanced example on how to use Python alongside Node-RED to gather data via OPC-UA, create advanced triggering conditions in Node-RED, trigger python scripts, communicate data back and forth between Node-RED and Python, please search the PACEdge Knowledgebase portal.

Section 6: PACEdge Performance Indication

PACEdge software is based on flexible and easy-to-use components, enabling fast application development. This section provides performance expectations and guides the user in selecting a suitable PACEdge hardware option and software implementation choices.

The subsections below will document the OPC UA Client in Node-RED performance, Node-RED Dashboard performance, Node-RED Modbus TCP, and Connext Historian using MySQL performance. It is expected that the Connext OPC UA Client and Modbus TCP client will result in higher performance as in Node-RED.

Note: Interacting with PACEdge via the locally (DisplayPort) attached monitor and local browser puts extra workload on the PACEdge unit compared to the remotely accessing PACEdge via Ethernet. Due to the lower computation performance of RXi2-LP, it is not recommended to use RXi2-LP in this mode.

In the test results below, “Loc” indicates that PACEdge is being used with a locally attached Monitor, keyboard, mouse, and browser inside the PACEdge, whereas “Rem” indicates that PACEdge is being accessed remotely via Ethernet.

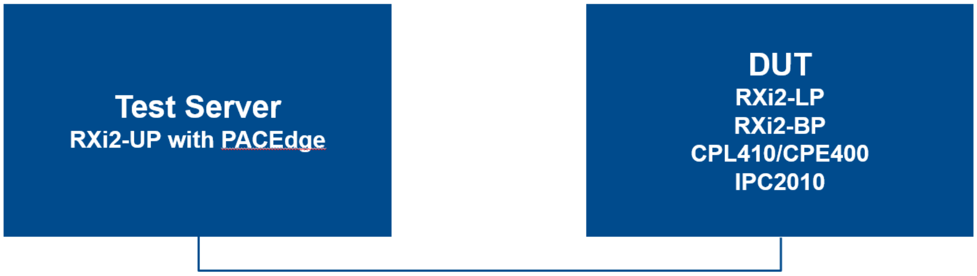
6.1 OPC UA Performance Indication

THE OPC UA Performance test was done using two PACEdge devices: one configured to act as OPC UA Server (highest performance IPC, RXi2-UP) and the other as an OPC UA Client. The server was set up to generate a large number of variables and to have them updated at the specified rate. The client was using two ways of getting variable updates:

1. Using single variable read operation:
   a. The client sends 100 single-variable read requests
   b. The server responds with 100 replies
   c. Once all replies are received, the Client repeats the sequence.
2. Using batch read request:
   d. The client sends one read request containing 100 variables
   e. The server responds with 10 replies, each containing 10 variable updates
   f. Once all replies are received, the Client repeats the sequence

In addition to performing OPC UA read transactions, PACEdge stores all variables in the InfluxDB database and uses one graph and one dial to show performance via the Node-RED dashboard and the Grafana dashboard.

Figure 190: Performance Indication



The following table provides test results of the Node-RED OPC UA Client performance test:

Table 6 OPC UA Variable Read Performance (variables/sec)

OPC UA Client Functions	Units	LP (Rem)	BP (4-core) (Rem)	BP (4-core) (Loc & Rem)	CPL410/ CPE400 (Rem)	IPC 2010 (Rem)
Test: OPC UA **Single** Var Read InfluxDB Node-RED Dashboard Grafana Dashboard	**Var/sec**	**100**	**1000**	**1000**	**90**	**150**
Test: OPC UA **Batch** Var Read InfluxDB Node-RED Dashboard Grafana Dashboard	**Var/sec**	**450**	**3100**	**2900**	**350**	**450**

6.2 Node-RED Dashboard Performance Indication

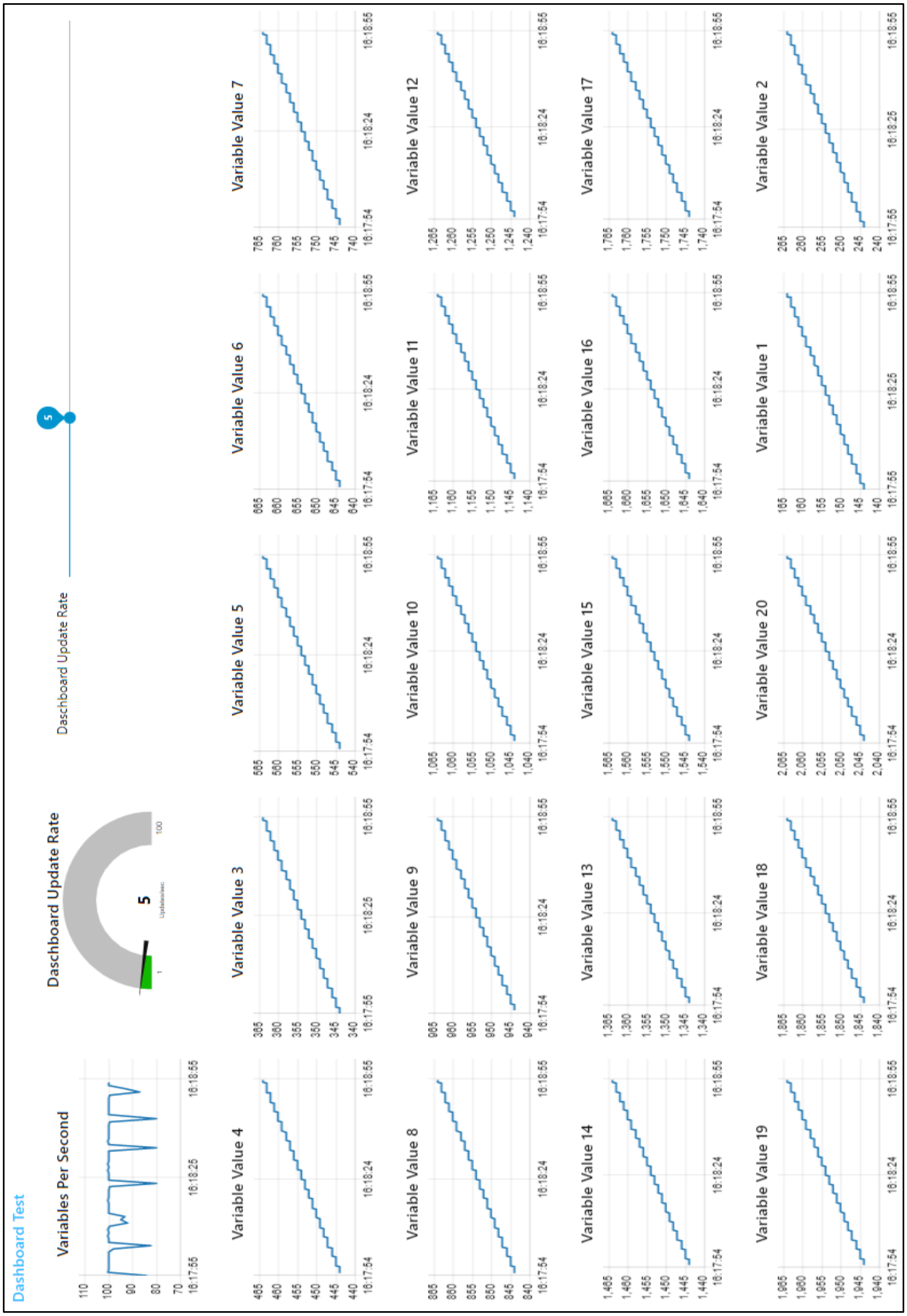
For this test, PACEdge with OPC UA Client reads 20 variables from OPC UA Server, stores them in the InfluxDB, and displays them in 20 Node-RED Dashboard graphs. The read and graph update rate is increased to the point where further increases result in noticeable control latency and lag in graphs being updated. In addition, Grafana is being used to display CPU, Memory, and Disk utilization statistics.

Note: in this particular test, a heavy load is being placed on a computer where the web browser is running, which might become a performance limiting factor.

Table 7 Node-RED Dashboard Performance (variable updates per second)

Functions	Units	LP (Rem)	BP (4-core) (Rem)	BP (4-core) (Loc & Rem)	CPL410 (Rem)	IPC 2010 (Rem)
OPC UA Client Node-RED DashboardGrafana	**Dashboard Update Rate: (Var/sec)**	**80**	**100**	**80**	**50**	**80**

Figure 191: Dashboard Update Rate



6.3 Node-RED Modbus TCP Performance Indication

In this test, the PACEdge device with Modbus TCP (Server) reads 1000 variables from Modbus TCP (Client). The server issues 10 FC3 read requests for 100 variables each. After all, responses are received, the read cycle is repeated. All received variables are stored in InfluxDB; the performance graph is displayed in Node-RED Dashboard; IPC load is being captured via Telegraf and displayed via Grafana

Table 8 Node-RED Modbus TCP Read Performance

Functions	Units	LP (Rem)	BP (4-core) (Rem)	BP (4-core) (Loc & Rem)	CPL410 (Rem)	IPC 2010 (Rem)
Modbus TCP Server InfluxDB Node-RED Dashboard Telegraf Grafana	Variable Read: (Var/sec)	8K	40K	40K	4K	8K

6.4 Node-RED Connext Historian Performance Indication

In this test, the PACEdge device uses the Modbus TCP Client driver within the Connext data gateway to read variables and store data via Historian inside the MySQL database. The test measures how much data Historian can store in MySQL. In parallel, Telegraf collects the IPC load, and Grafana displays it.

Client Functions	Parameter	LP (Rem)	BP_1605 (Rem)	BP_1605 (Loc & Rem)	CPL410 (Rem)	IPC 2010 (Rem)
Connext Modbus TCP driver MySQL Telegraf Grafana	Variable store rate: (Var/sec)	50	750	750	50	50

Section 7: Saving and Restoring User Data

PACEdge software consists of several applications, such as Node-RED, InfluxDB, Grafana, MySQL, Connext, WebHMI, etc. When a user creates his specific flows and stores data in the databases, all this user-specific data will be stored in the Linux file system. This data can be periodically copied and stored in other locations, such as a USB storage device, for backup purposes.

User data location is at /home/admin/docker/volumes/.

The following are the most critical applications/folders:

- emersonedgestack_grafana-storage
- emersonedgestack_influxdb
- emersonedgestack_mysql
- emersonedgestack_nodered
- emersonedgestack_portainer
- emersonedgestack_movicon
- emersonedgestack_timescaledb

Note: The data stored/restored is in the**/home/admin/docker/volumes/*application-name*/_data/** folder. When restoring data, copy back folders into this location.
Note: access requires admin user rights.

Note: Cockpit/Navigator can be used to copy files and folders using the graphical user interface

Alternatively, Node-RED flows and Grafana dashboards can be stored (exported) and later installed (imported) from each application natively. Connext and WebHMI projects are, by definition, developed on engineering workstations using Movicon.NExT software can be restored by performing a new upload to the PACEdge device.

7.1 Exporting and Importing Flows in Node-RED

7.1.1 Export Flow

While in Node-RED application:

1. Click on the menu icon on the right side of the screen ≡.
2. Select either the **current flow** or **all flows** option.
3. Click on the **Download** button.
4. Select the location (on your client system, not PACEdge) where to store the flow.json file.

7.1.3 Import Flow

While in Node-RED application:

1. Click on the menu icon on the right side of the screen ≡.
2. Click on select a file to import.
3. Navigate to your client system's desired flow xxx.json file (not PACEdge).
4. Click on **Open**.
5. Click on **Import**.

7.2 Exporting and Importing Dashboards in Grafana

7.2.1 Export Dashboard

While in the Grafana application:

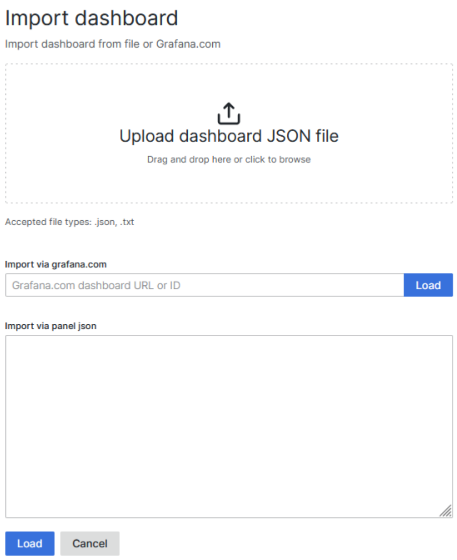
1. Navigate to your desired Dashboard.
2. Click on the **Share Dashboard** icon .
3. Click on **Export.**
4. Click on **Save to file** and select the location to store it on your client system (not PACEdge).

7.2.2 Import Dashboard

While in the Grafana application:

1. Click on the hamburger icon in the upper left corner and then click on **Dashboards**
2. Click to open a drop down list under **New** (right side of the screen) and select **Import**

Figure 192: Import Dashboard Menu



Import dashboard

Import dashboard from file or Grafana.com

Upload dashboard JSON file

Drag and drop here or click to browse

Accepted file types: .json, .txt

4. Click on Upload dashboard JSON File.
5. Select your desired Grafana view file xxx.json and click on **Load**.
6. If asked, select the required database from the drop-down menu
   Note: If there is no valid InfluxDB database there, then most likely Grafana was not configured yet to connect with database. Please follow the steps in configuring Grafana in this manual first to configure required database.
7. Click on the **Import** button.

7.3 Importing Projects in Connext/WebHMI

Connext and WebHMI projects are first created on engineering workstations using Movicon.NExT software and then loaded into the PACEdge device. Therefore there is no need to export projects from PACEdge. Projects can be loaded from the engineering workstation onto the PACEdge device when recovery is needed.

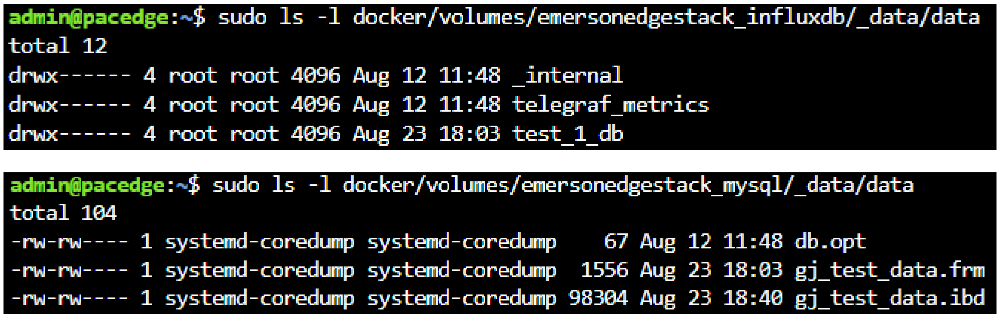
7.4 Backing Up and Restoring InfluxDB and MySQL Databases

InfluxDB and MySQL databases are stored in the main Linux file system at the following path:

- /home/admin/docker/volumes/emersonedgestack_influxdb/_data/
- /home/admin/docker/volumes/emersonedgestack_mysql/_data/

As an example, see the figure below:

Figure 193: Example paths



```
admin@pacedge:~$ sudo ls -l docker/volumes/emersonedgestack_influxdb/_data/data
total 12
drwx------ 4 root root 4096 Aug 12 11:48 _internal
drwx------ 4 root root 4096 Aug 12 11:48 telegraf_metrics
drwx------ 4 root root 4096 Aug 23 18:03 test_1_db
```

```
admin@pacedge:~$ sudo ls -l docker/volumes/emersonedgestack_mysql/_data/data
total 104
-rw-rw---- 1 systemd-coredump systemd-coredump    67 Aug 12 11:48 db.opt
-rw-rw---- 1 systemd-coredump systemd-coredump  1556 Aug 23 18:03 gj_test_data.frm
-rw-rw---- 1 systemd-coredump systemd-coredump 98304 Aug 23 18:40 gj_test_data.ibd
```

When a backup is desired, the user can copy all folders from **…./_data/** folder and restore them to the same location later. Cockpit Navigator can be conveniently used for the copy operations.

Note: Access to these folders requires admin privileges.

7.5 Saving License Files

PACEdge devices come with pre-installed software and valid factory license files. Under normal circumstances, the user does not have to worry about keeping a copy of the valid license file. Still, if restoring to PACEdge Factory Default needs to be done on RXi2-BP, RXi2-LP and IPC 2010 Industrial PCs, then a valid license file will be required afterward to fully activate the PACEdge software. In such a situation, the user can always request a new license file from the Customer Care representative, as described in section 3.7, *PACEdge License File*. Still, an alternative would be to make a copy of the license file before performing a restore operation. The easiest way to copy files is using the Navigator application in Cockpit. To do so:

1. Open Navigator
2. Navigate to: /home/admin/pacedge/emerson-software/license/ folder
3. Right-click on the **license.json** file and choose the Download option, specify the location to store the file on your local PC

   NOTE: If the download fails with the error: **Failed – Network error**, log out from Cockpit and log in again as the admin user.

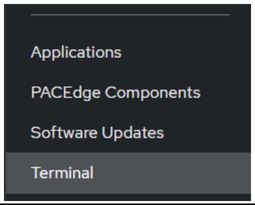
4. Repeat the steps for **license.sig** file in the same folder.

Alternatively, one can store license files on a USB storage device directly attached to the PACEdge device by following the steps below:

Mount the USB storage device by following the procedure in **10.1 Mounting USB** .

1. Within the Cockpit, go to the Terminal screen and enter the following commands:

Figure 194: Open Cockpit's Terminal



   a. **sudo cp pacedge/emerson-software/license/license.json /mnt/usb**
   b. Enter the admin password (if requested).
   c. **sudo cp pacedge/emerson-software/license/license.sig /mnt/usb**

2. Next, unmount the USB storage device by following: 10.2 Un-Mounting .
3. You may remove the USB storage device and save the files: 1) license.json and 2) license.sig on your Windows workstation.

Section 8: PACEdge Software Backup/Restore/Recovery

With PACEdge software, users have the ability to install additional software packages and customized configurations to their liking. During the development and validation phases, it's common for users to want to either back up their current setup or revert back to the original PACEdge state.

The backup procedure allows the user to create a copy of PACEdge software with own customizations, flows, views, databases, Connext, and WebHMI projects, including Linux operating system. This step might be useful for creating an operational backup or duplicating the existing setup onto several additional systems.

Users can use the restore procedure to restore PACEdge either to its Factory Default state or to a previously saved backup. Restoring to the Factory Default state is also possible during a PACEdge upgrade to the latest version, where it is acceptable to overwrite any user-specific data on the old unit. Additionally, the backup/restore procedure can also be used to create a golden setup image, which can then be restored to multiple other PACEdge devices. This process even allows for migration between IPC models, for example, from an RXi2-LP to an RXi2-BP or vice versa.

Note: Restoring the device will result in the erasure of all current data, including databases and license files. To avoid losing important information, please ensure the backup of critical data and license files using the procedures documented in Section 7: *Saving and Restoring User Data.*

Note: This procedure shows how to recover PACEdge software installation back to Factory Default; however, if the user has entered UEFI setting menus and modified UEFI settings, these will remain as is (applicable to RXi2-BP and RXi2-LP hardware). To restore UEFI settings to Factory Defaults, enter the UEFI Settings menu and choose the recovery option on the Save & Exit page.

Based on the hardware type, please refer to the applicable sections bellow.

8.1 PACEdge Software Backup on RXi2-BP, RXi2-LP IPCs

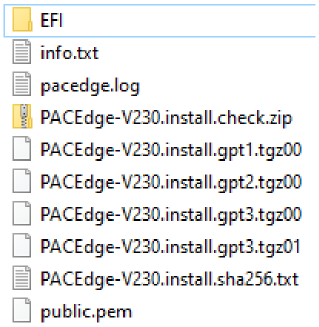
From the Emerson Software Downloads site, download **PACEdge v2.3.0 Backup Restore Install RXi2-BP RXi2-LP** (file name: **PEv230BRIUtilBP.zip**). Then follow the steps below to perform a backup:

1. Get an empty, min 32 GB size USB storage device. Make sure it is empty, as conflicts might arise.

Note: if using old USB storage devices, one might experience a problem in which a device may not boot from the USB storage device. This is typically due to the file system changes and boot record configuration on the USB storage device. The workaround, in this case, is either: 1) use a brand new USB storage device, 2) download from the Internet one of Linux installation ISO images, use a Windows utility such as Rufus, to make a bootable USB storage device, then delete all the files from the USB storage device and proceed with next steps.

2. Copy all files to the root directory of the USB storage device. The directory structure should look like this:

Figure 195: Files at the Root Directory of the USB storage device

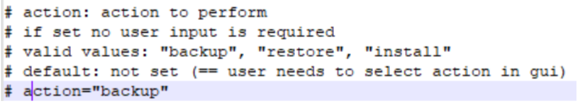


```
EFI
info.txt
pacedge.log
PACEdge-V230.install.check.zip
PACEdge-V230.install.gpt1.tgz00
PACEdge-V230.install.gpt2.tgz00
PACEdge-V230.install.gpt3.tgz00
PACEdge-V230.install.gpt3.tgz01
PACEdge-V230.install.sha256.txt
public.pem
```

Note: Double-check if the automatic PACEdge Factory Image install is disabled; otherwise, it will automatically overwrite all your data and install a Factory Default PACEdge image. To check:

a. In Notepad or a similar editor, open file: EFI\BOOT\unattendedinstall

b. Look at the top of the file for a line **action=” install”** or **action=” backup.”** Make sure this line is commented out with **#** in front of it. This way utility will stop and ask you what operation should be executed.

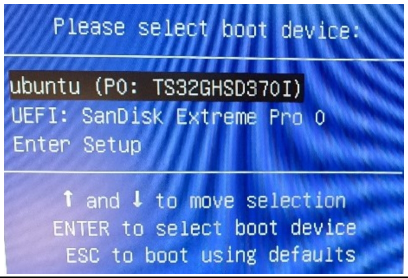


```
# action: action to perform
# if set no user input is required
# valid values: "backup", "restore", "install"
# default: not set (== user needs to select action in gui)
# action="backup"
```

c. Make the necessary changes and save the modified file.

3. Plug the USB storage device into any of the USB ports, power up the IPC, and keep pressing the F7 button to get into the boot selection menu; select the USB storage device to boot from (in the figure below, this would be the second line item: UEFI: SanDisk Extreme Pro 0.)

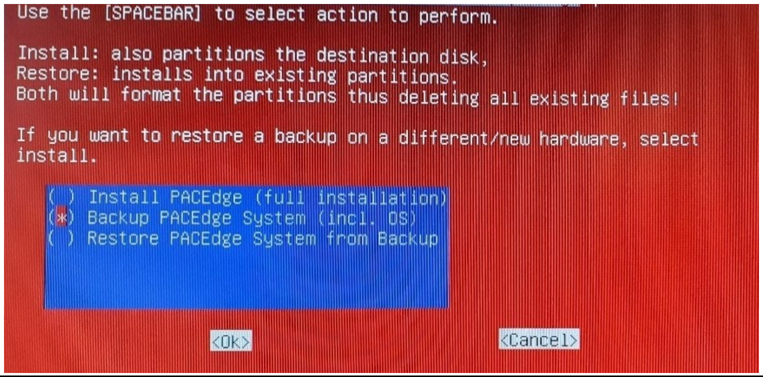
Figure 196: Select the Boot Device



```
Please select boot device:

ubuntu (P0: TS32GHSD370I)
UEFI: SanDisk Extreme Pro 0
Enter Setup

↑ and ↓ to move selection
ENTER to select boot device
ESC to boot using defaults
```

4. Wait until the following dialog appears. Use arrow keys to move the cursor and the space key to select the **Backup** option. Then use the Tab key to move the cursor to the **OK** button and hit enter.

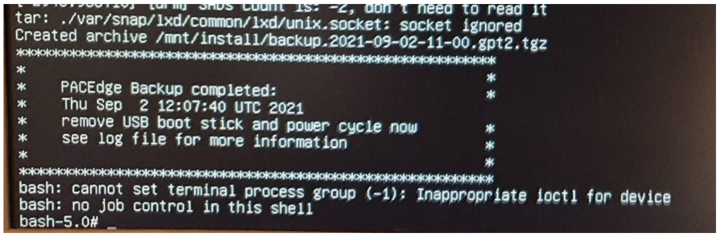
Figure 197: Select the Backup Option



```
Use the [SPACEBAR] to select action to perform.

Install: also partitions the destination disk,
Restore: installs into existing partitions.
Both will format the partitions thus deleting all existing files!

If you want to restore a backup on a different/new hardware, select
install.

   ( ) Install PACEdge (full installation)
   (*) Backup PACEdge System (incl. OS)
   ( ) Restore PACEdge System from Backup
```

5. Wait for the completion message as shown below. Depending on the hardware type, creating a backup might take 10 to 60 minutes.

Figure 198: Backup Completed Screen



```
SADS count is: -2, don't need to read it
tar: ./var/snap/lxd/common/lxd/unix.socket: socket ignored
Created archive /mnt/install/backup.2021-09-02-11-00.gpt2.tgz
*************************************************************
*                                                           *
*    PACEdge Backup completed:                              *
*    Thu Sep  2 12:07:40 UTC 2021                           *
*    remove USB boot stick and power cycle now              *
*    see log file for more information                      *
*                                                           *
*************************************************************
bash: cannot set terminal process group (-1): Inappropriate ioctl for device
bash: no job control in this shell
bash-5.0# _
```

s

Note: After the backup, on the USB storage device, you will see the second set of installation files with word **backup** in them.

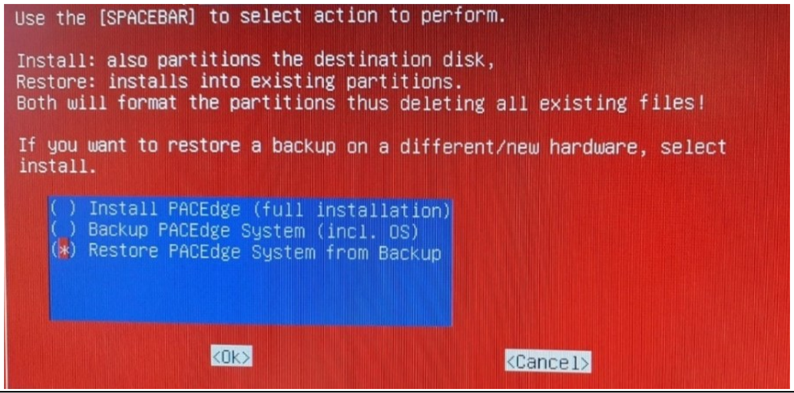
8.3 PACEdge Software Restore/Recovery on RXi2-BP, RXi2-LP IPCs

To restore PACEdge software, either to your previously made backup or to the Factory Default (Recovery), please perform the following steps:

1. Perform steps 1-5 document in Section 8.1 *PACEdge Software Backup on RXi2-BP, RXi2-LP IPCs.*
2. Wait until the following dialog appears, and then use the arrow keys to move the cursor and the space key to select the applicable option:
   a. Choose **Restore PACEdge** option when restoring previously backed up version on the same hardware unit.
   b. Choose **Install PACEdge (full installation)** option when recovering to a different HW device or performing recovery to a Factory Default (can also be used for the fresh upgrade to the latest PACEdge version).

   Then use the Tab key to move the cursor to the **OK** button and hit enter.

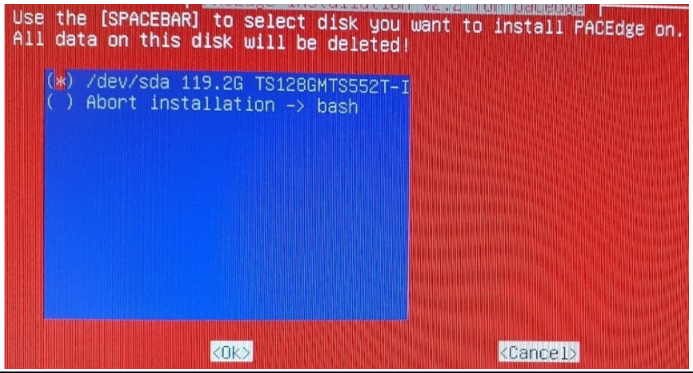
Figure 199: Select either Restore or Install option



```
Use the [SPACEBAR] to select action to perform.

Install: also partitions the destination disk,
Restore: installs into existing partitions.
Both will format the partitions thus deleting all existing files!

If you want to restore a backup on a different/new hardware, select
install.

    ( ) Install PACEdge (full installation)
    ( ) Backup PACEdge System (incl. OS)
    (*) Restore PACEdge System from Backup

              <Ok>                          <Cancel>
```

3. If asked for a disk partition, use the arrows, space, and Tab keys to select the first partition listed and click **OK** to proceed.

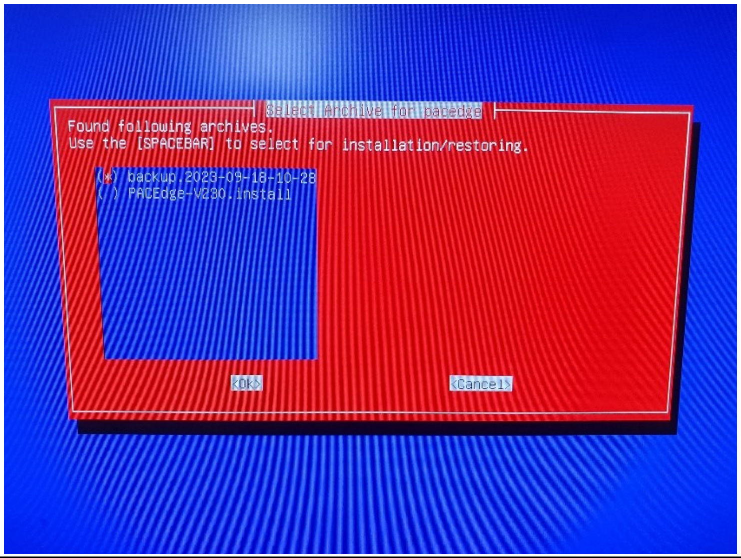
Figure 200: Select the Disk Partition (If Applicable)



```
Use the [SPACEBAR] to select disk you want to install PACEdge on.
All data on this disk will be deleted!

    (*) /dev/sda 119.2G TS128GMTS552T-I
    ( ) Abort installation -> bash

              <Ok>                        <Cancel>
```

4. Use arrows, Space, and Tab keys to select either a factory default image (**PACEdge-V230.install**) or one of your backups, then choose **OK** and hit enter.

Figure 201: Select the Backup or Factory Default (Install) Image

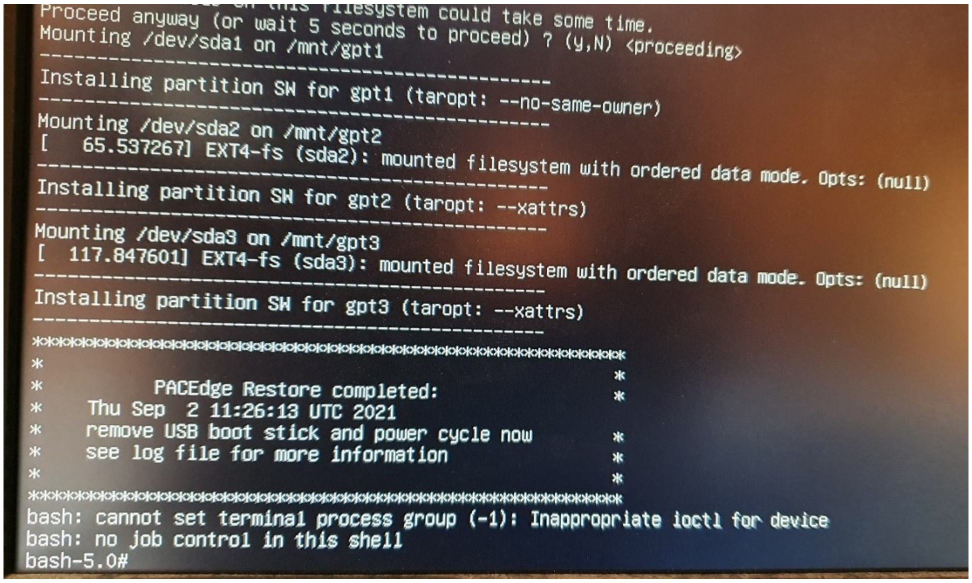


```
Select Archive for pacedge
Found following archives.
Use the [SPACEBAR] to select for installation/restoring.

(*) backup.2023-09-18-10-28
( ) PACEdge-V230.install
```

5. Wait until the restore procedure completes, as indicated by a message.
   Note: during the installation process, please ignore any questions that might come up on the screen, letting them time out and continue. No user intervention is required.

Note: Depending on the HW type, this step may take anywhere from a couple of minutes to 10-20 minutes.

Figure 202: PACEdge Restore Completed



```
                 on this filesystem could take some time.
Proceed anyway (or wait 5 seconds to proceed) ? (y,N) <proceeding>
Mounting /dev/sda1 on /mnt/gpt1
------------------------------------------------------------
Installing partition SW for gpt1 (taropt: --no-same-owner)
------------------------------------------------------------
Mounting /dev/sda2 on /mnt/gpt2
[   65.537267] EXT4-fs (sda2): mounted filesystem with ordered data mode. Opts: (null)
------------------------------------------------------------
Installing partition SW for gpt2 (taropt: --xattrs)
------------------------------------------------------------
Mounting /dev/sda3 on /mnt/gpt3
[  117.847601] EXT4-fs (sda3): mounted filesystem with ordered data mode. Opts: (null)
------------------------------------------------------------
Installing partition SW for gpt3 (taropt: --xattrs)
------------------------------------------------------------
************************************************************
*                                                          *
*          PACEdge Restore completed:                      *
*   Thu Sep  2 11:26:13 UTC 2021                           *
*   remove USB boot stick and power cycle now              *
*   see log file for more information                      *
*                                                          *
************************************************************
bash: cannot set terminal process group (-1): Inappropriate ioctl for device
bash: no job control in this shell
bash-5.0#
```

Note: If you have backed up a PACEdge device with a valid license and then restored it on the same physical unit, the license will also be restored and remain valid. If restore is done on a different physical unit, or a Factory Default Recovery is done, the license will be invalid and require a new license file to be manually installed, as described in 3.7, *PACEdge License File*.

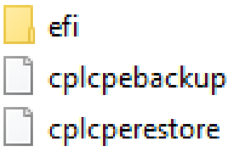
NOTE: After PACEdge restore, you might have to Upload again Movicon project (if used) by using Movicon.NExT Editor.

8.4 PACEdge Software Backup on CPL410, CPE400 Controllers

To perform PACEdge software backup, from the Emerson Software Downloads site, download **PACEdge 2.3 Backup Restore CPL410 CPE400** (file name: **PEv230BRUtilCPL.zip**). Please use the following procedure to perform a backup:

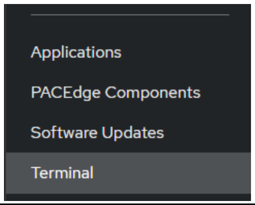
1. Copy the files to an empty USB storage device. The following are the required files:

Figure 203: Copy Files to USB storage device



2. Insert USB-Stick into CPL/CPE, USB1 port.
3. Mount USB storage device in Linux OS as described in 10.1. Mounting USB
4. Now, within the Cockpit, go to the Terminal screen and enter the following commands:

Figure 204: Cockpit's Terminal



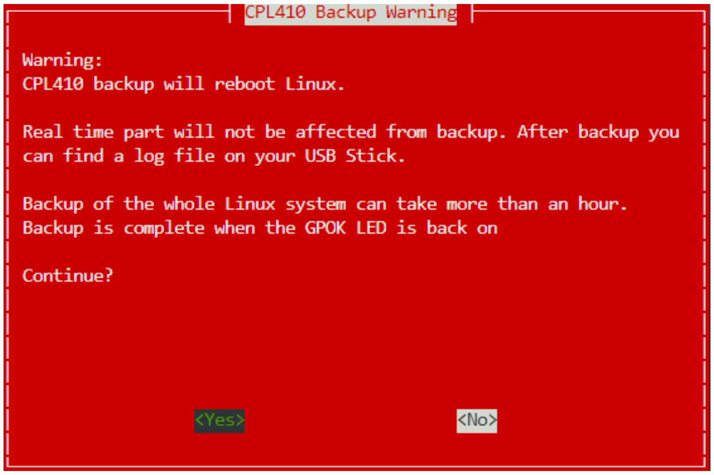
a. **cd /mnt/usb/**

b. **sudo ./cplcpebackup**

c. Enter the admin password if asked for.

6. Using the arrow keys, select **Yes** and press **enter**.

Figure 205: Select the Backup



```
CPL410 Backup Warning

Warning:
CPL410 backup will reboot Linux.

Real time part will not be affected from backup. After backup you
can find a log file on your USB Stick.

Backup of the whole Linux system can take more than an hour.
Backup is complete when the GPOK LED is back on

Continue?
```

7. When the backup starts, GPOK LED will turn off. It will remain off until the backup is complete, the PACEdge side of the controller is rebooted, and the GPOK LED is ON. It might take close to 2h to perform a backup. You can observe SSD LED, and as long as it periodically flashes backup process is in progress.

Before removing the USB storage device, please properly un-mount it by the following procedure in 10.2 Un-Mounting

8. At this point, you can remove the USB storage device and inspect the content on Windows PC. It will have several large archives with names starting with backup.

8.5 PACEdge Software Restore/Recovery on CPL410, CPE400 Controllers

8.5.1 CPL410, CPE400 PACEdge Recovery to Factory Default

CPL410 and CPE400 have a pre-installed Factory Default image. If Factory Default recovery is needed, please follow the steps below. Switching to the Factory Default image will delete all existing data, including Node-Red flows, Grafana views, and database content. License file will be preserved.

1. Using CPL410/CPE400 built-in display trigger Reset to Factory Default process:
    a. Press the DISP button until the arrow points at the menu entry: **Edge Settings**, then press SEL.
    b. Press the DISP button until the **Commands** option is selected, then press SEL.
    c. Press the DISP button until the **Factory Reset** option is selected, then press SEL.
    d. Press the DISP button until the **OK** option is selected, then press SEL.
2. At this point, GPOK LED will go OFF, and you need to reboot a complete unit by removing and applying power again. After the unit is up again, check the update status via display:
    a. Press the DISP button until the **Edge Settings** option is selected, then press SEL.
    b. Check that the message **Resetting GP** is shown. Also, a blinking SSD LED will indicate the recovery is in progress.
3. Wait until GPOK LED turns ON. This can take up to 20 minutes.
4. At this point, Factory Default restore is done. PACEdge is functional, and the License file is valid.

8.5.2 CPL410, CPE400 Restore of PACEdge Backup Image

If a PACEdge backup image was created, it could be later restored on the same HW family unit (meaning CPL410 to CPL410 or CPE400 to CPE400). If restoring to a physically different unit, new valid licenses will be required afterward to activate the PACEdge.

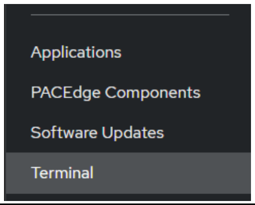
Note that the backup restore operation will overwrite all existing data, including Node-Red flows, Grafana views, and database content.

To perform PACEdge software restore from the backup, use the same USB storage device that was used to create a backup. Please use the following procedure to perform a restore

1. Insert USB-Stick into CPL410/CPE400, USB1 port.
2. Mount USB storage device in Linux OS as described in 10.1, *Mounting USB .*

Now, within the Cockpit, go to the Terminal screen and enter the following commands:

Figure 206: Cockpit's Terminal



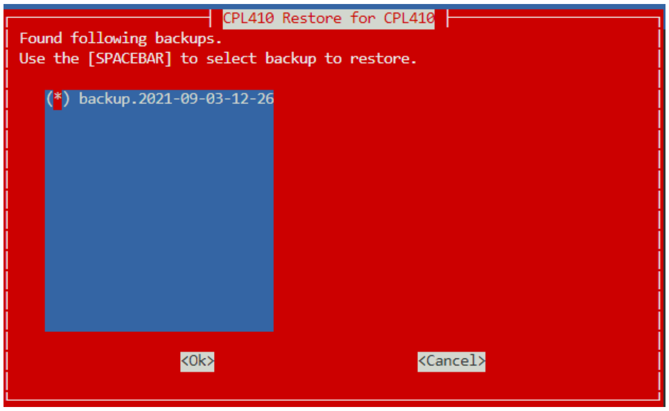
a. **cd /mnt/usb/**

b. **sudo ./cplcperestore**

c. Enter the admin password (if applicable).

3. Using the arrow keys, select **Yes** and press **enter.**
4. Using arrow keys and space keys, mark the backup image you would like to restore, then with the Tab key, select OK and **enter**.

Figure 207: Select the Backup File



```
CPL410 Restore for CPL410
Found following backups.
Use the [SPACEBAR] to select backup to restore.

  (*) backup.2021-09-03-12-26

          <Ok>                    <Cancel>
```

5. When restore starts, GPOK LED will turn off. It will remain off until the backup is complete, the PACEdge side of the controller is rebooted, and the GPOK LED is ON. Restore can take up to 1h to complete. Observe SSD LED on the front panel, if it periodically flashes the Restore operation is in progress.

Before removing the USB storage device, please properly un-mount it by the following procedure in **10.2 Un-Mounting** .

6. Remove the USB storage device.

NOTE: After PACEdge restore, you might have to Upload again Movicon project (if used) by using Movicon.NExT Editor.

8.7 PACEdge Software Backup on IPC 2010

To perform Backup/Restore operations on the IPC 2010, an off-the-shelf serial cable will be required. Cable pinout is defined in 10.3, *Serial RS232 Cable for IPC 2010*.

From the Emerson Software Downloads site, download **PACEdge v2.3.0 Backup Restore Install IPC 2010** (file name: **PEv230BRIUtilIPC 2010.zip**). Then follow the steps below to perform a backup:

1. Insert a USB storage device, with a minimum of 32 GB of storage, into the USB port. Ensure the storage device is empty, as otherwise, conflicts may arise..

   Note: If using an old USB storage device, one might experience a problem in which a device may not boot from the USB storagedevice. This is typically due to the file system changes and boot record configuration on the USB storage device. The workaround, in this case, is either: 1) use a brand new USB storage device, 2) download from the Internet one of Linux installation ISO images, use a Windows utility such as Rufus, to make a bootable USB storage device, then delete all the files from the USB storage device and proceed with next steps.

2. Copy all files to the root directory of the USB storage device. The directory structure should look like this:

Figure 208: Files at the Root Directory of the USB storage device

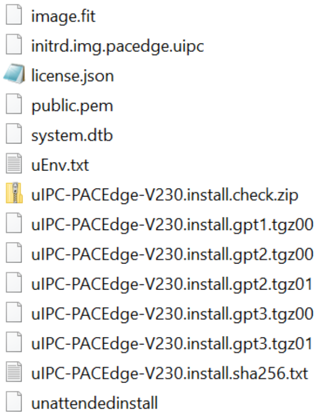
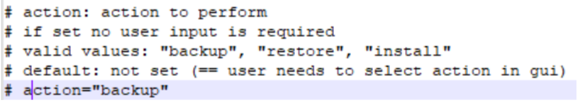


image.fit
initrd.img.pacedge.uipc
license.json
public.pem
system.dtb
uEnv.txt
uIPC-PACEdge-V230.install.check.zip
uIPC-PACEdge-V230.install.gpt1.tgz00
uIPC-PACEdge-V230.install.gpt2.tgz00
uIPC-PACEdge-V230.install.gpt2.tgz01
uIPC-PACEdge-V230.install.gpt3.tgz00
uIPC-PACEdge-V230.install.gpt3.tgz01
uIPC-PACEdge-V230.install.sha256.txt
unattendedinstall

Note: Double-check if the automatic PACEdge Factory Image install is disabled; otherwise, it will automatically overwrite all your data and install a Factory Default PACEdge image. To check:

a. In Notepad or a similar editor, open file: **unattendedinstall**

b. Look at the top of the file for a line **action=" install"** or **action=" backup."** Make sure this line is commented out with **#** in front of it. This way utility will stop and ask you what operation should be executed.

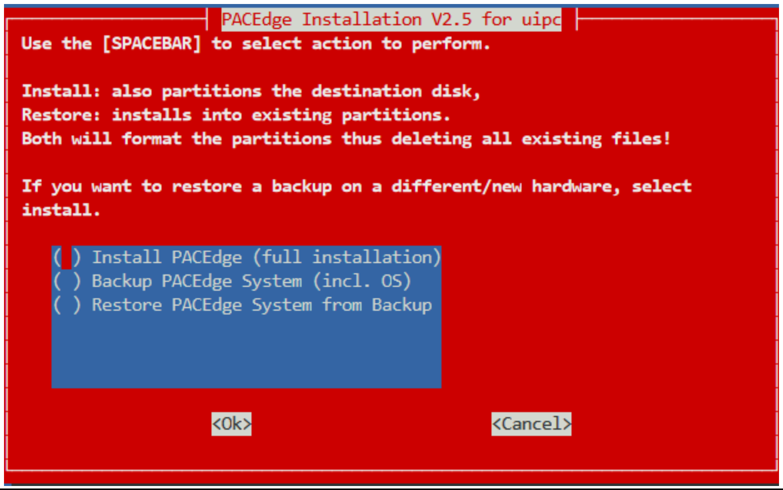


```
# action: action to perform
# if set no user input is required
# valid values: "backup", "restore", "install"
# default: not set (== user needs to select action in gui)
# action="backup"
```

c. Make changes as necessary and save the modified file.

3. Connect the serial cable between IPC 2010 RS232 connector and your PC Serial port. For cable details refer to: 10.3 Serial RS232 Cable for IPC 2010.
4. Plug the USB storage device into any of the USB ports, power up the IPC, and wait until the following screen appears:

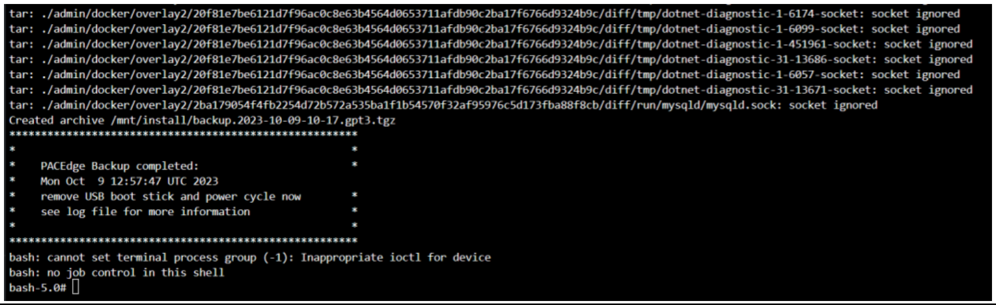
Figure 209: IPC 2010 Backup/Restore Utility Screen 1



```
PACEdge Installation V2.5 for uipc
Use the [SPACEBAR] to select action to perform.

Install: also partitions the destination disk,
Restore: installs into existing partitions.
Both will format the partitions thus deleting all existing files!

If you want to restore a backup on a different/new hardware, select
install.

   ( ) Install PACEdge (full installation)
   ( ) Backup PACEdge System (incl. OS)
   ( ) Restore PACEdge System from Backup

            <Ok>                     <Cancel>
```

5. Using space bar mark the entry: Backup PACEdge, then using Tab key highlight (turns black) OK button and hit Enter key.
6. Wait until backup is complete (can take 2-3 hours) as indicated by the screen:

Figure 210 IPC 2010 Backup Completion Message



```
tar: ./admin/docker/overlay2/20f81e7be6121d7f96ac0c8e63b4564d0653711afdb90c2ba17f6766d9324b9c/diff/tmp/dotnet-diagnostic-1-6174-socket: socket ignored
tar: ./admin/docker/overlay2/20f81e7be6121d7f96ac0c8e63b4564d0653711afdb90c2ba17f6766d9324b9c/diff/tmp/dotnet-diagnostic-1-6099-socket: socket ignored
tar: ./admin/docker/overlay2/20f81e7be6121d7f96ac0c8e63b4564d0653711afdb90c2ba17f6766d9324b9c/diff/tmp/dotnet-diagnostic-1-451961-socket: socket ignored
tar: ./admin/docker/overlay2/20f81e7be6121d7f96ac0c8e63b4564d0653711afdb90c2ba17f6766d9324b9c/diff/tmp/dotnet-diagnostic-31-13686-socket: socket ignored
tar: ./admin/docker/overlay2/20f81e7be6121d7f96ac0c8e63b4564d0653711afdb90c2ba17f6766d9324b9c/diff/tmp/dotnet-diagnostic-1-6057-socket: socket ignored
tar: ./admin/docker/overlay2/20f81e7be6121d7f96ac0c8e63b4564d0653711afdb90c2ba17f6766d9324b9c/diff/tmp/dotnet-diagnostic-31-13671-socket: socket ignored
tar: ./admin/docker/overlay2/2ba179054f4fb2254d72b572a535ba1f1b54570f32af95976c5d173fba88f8cb/diff/run/mysqld/mysqld.sock: socket ignored
Created archive /mnt/install/backup.2023-10-09-10-17.gpt3.tgz
*******************************************************
*                                                     *
*    PACEdge Backup completed:                        *
*    Mon Oct  9 12:57:47 UTC 2023
*    remove USB boot stick and power cycle now        *
*    see log file for more information                *
*                                                     *
*******************************************************
bash: cannot set terminal process group (-1): Inappropriate ioctl for device
bash: no job control in this shell
bash-5.0# 
```

7. Remove power and USB storage device. Now USB storage device contains a backup image of your IPC 2010.

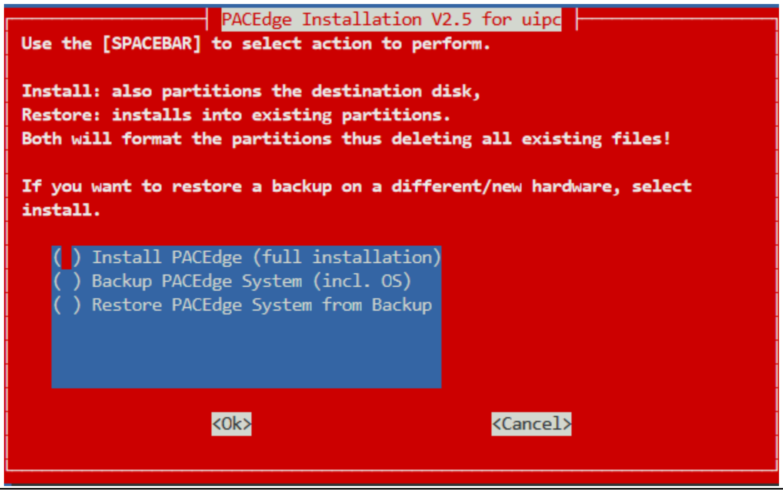
8.9 PACEdge Software Restore/Recovery on IPC 2010

To restore a previously backed up image of IPC 2010 or to recover to Factory Default installation, you will need a serial cable and a USB storage device with software as described in section 8.6, *PACEdge Software* Backup on , the first four steps. If you are performing a Restore, then use same USB storage device you did a Backup on.

Then, perform following steps:

1. In the first screen, using arrow keys and space bar, mark the first entry that says: Install PACEdge. Note, please use this entry for both Restore and Recovery operations

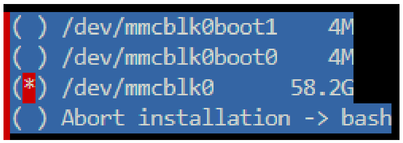
Figure 211: IPC 2010 Restore/Recovery Utility Screen 1



```
PACEdge Installation V2.5 for uipc
Use the [SPACEBAR] to select action to perform.

Install: also partitions the destination disk,
Restore: installs into existing partitions.
Both will format the partitions thus deleting all existing files!

If you want to restore a backup on a different/new hardware, select
install.

    ( ) Install PACEdge (full installation)
    ( ) Backup PACEdge System (incl. OS)
    ( ) Restore PACEdge System from Backup

             <Ok>                    <Cancel>
```

2. Next, in the screen that appears, using arrow keys and space bar mark the entry that correspond to internal disc (typically third entry that is close to 60GB in size). Highlight OK and hit Enter key

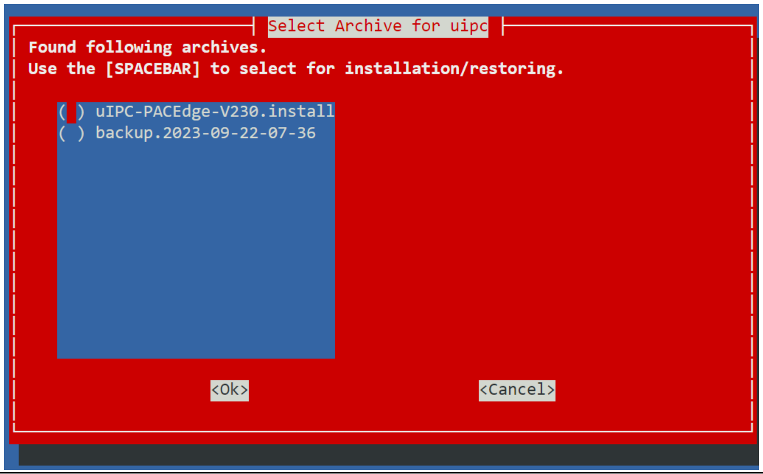
Figure 212: IPC 2010 Disc Partition Selection Screen4



```
( ) /dev/mmcblk0boot1    4M
( ) /dev/mmcblk0boot0    4M
(*) /dev/mmcblk0      58.2G
( ) Abort installation -> bash
```

3. In the screen that appears, mark the image that you want to restore.
   a. In case you want to perform Factory Default recover please select uIPC-PACEdge-V230.install. Note, all your existing data on the IPC 2010, including license file, will be overwritten and the unit will be reset to factory defaults minus a valid license file. If you do not have a backup copy of your license file, please contact Customer Care. For more details, please refer to section: 3.7 PACEdge License File
   b. In case you want to perform Restore from Backup operation, please select one of the images that you have previously backed up. Backup images have word "backup" and time of the backup in the file name.

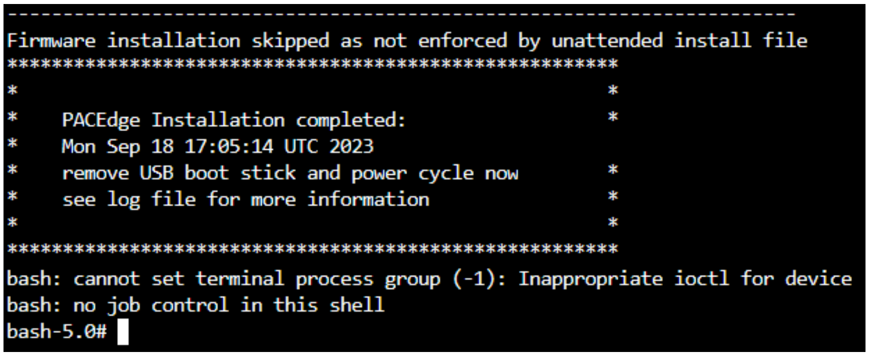
Figure 213: Select Archive



```
Select Archive for uipc
Found following archives.
Use the [SPACEBAR] to select for installation/restoring.

    ( ) uIPC-PACEdge-V230.install
    ( ) backup.2023-09-22-07-36
```

c. Highlight OK option and hit enter key to proceed. Once installation is finished a following screen will be shown. Please remove power and USB storage device, then reapply power to boot new image.

Figure 214: IPC 2010 Restore/Recovery Finished Screen



```
-----------------------------------------------------------------------
Firmware installation skipped as not enforced by unattended install file
*******************************************************
*                                                     *
*    PACEdge Installation completed:                  *
*    Mon Sep 18 17:05:14 UTC 2023
*    remove USB boot stick and power cycle now        *
*    see log file for more information                *
*                                                     *
*******************************************************
bash: cannot set terminal process group (-1): Inappropriate ioctl for device
bash: no job control in this shell
bash-5.0#
```

Section 9: PACEdge Version Update

9.1 Upgrading to PACEdge v2.4.0 (Security Update)

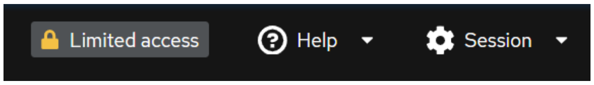
PACEdge v2.4.0 is a security update for version v2.3.0.

NOTE: PACEdge v2.3.0 is a pre-requisite for the PACEdge v2.4.0 update.

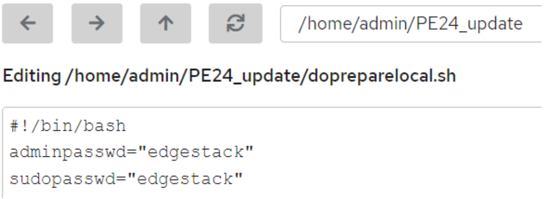
There are two methods for the version v2.4.0 update, one for a standalone system using Navigator and the other for the group update using Group Manager.

9.1.1 Single Device Update via Navigator

1. From the Customer Center, PACEdge, version 2.4 page, download **PACEdge Upgrade to v2.4 via USB Navigator** (actual file name: **PEv240UpdateUSBUtil.zip**)
2. Log into Cockpit on the PACEdge device and make sure you are in the **Limited Access** mode (as opposed to Administrative access):



3. Go to Navigator, navigate to */home/admin/* folder and create new directory called: **PE24_update.**
4. Using Navigator, open the newly created directory and drag and drop **PEv240UpdateUSBUtil.zip** file into it. Refresh the browser and wait until the file is fully copied and is listed in the directory.
   Note: Depending on the hardware, this step will take up to two minutes.
5. Go to the Terminal and issue the following commands:
   a. **cd /home/admin/PE24_update**
   b. **unzip PEv240UpdateUSBUtil.zip**
6. Go to Navigator and open file dopreparelocal.sh. Change both **adminpasswd** and **sudopasswd** to the passwords you have configured for the Linux/Cockpit. Save the file.

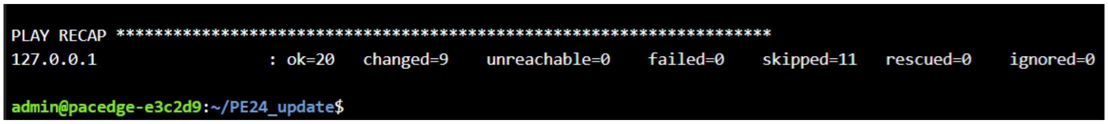


Editing /home/admin/PE24_update/dopreparelocal.sh

```
#!/bin/bash
adminpasswd="edgestack"
sudopasswd="edgestack"
```

7. Go to Terminal and issue following commands:
   a. **cd /home/admin/ PE24_update**
   b. **./localupdate.sh**

8. Wait until the update completes, as indicated by message:



```
PLAY RECAP ********************************************************************
127.0.0.1                  : ok=20   changed=9    unreachable=0    failed=0    skipped=11   rescued=0    ignored=0

admin@pacedge-e3c2d9:~/PE24_update$
```

Note: depending on the hardware, this step might take up to 1h.

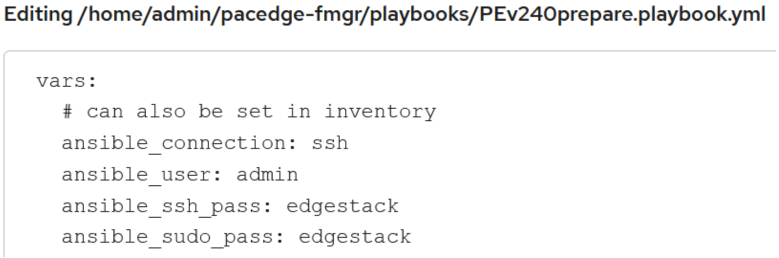
9. Reboot the PACEdge.

9.1.2 Multiple Device Update via Group Manager

To utilize Group Manager to perform updates, make sure that you have a PACEdge device with Group Manager license.

1. From the Customer Center, PACEdge, version 2.4 page, download **PACEdge Upgrade to v2.4 via Group Manager** (actual file name: **PEv240UpdateGMUtil.tgz**)
2. Log into Cockpit on the Group Manager device, open Terminal and issue following commands:
   a. **cd /home/admin/pacedge-fmgr**
   b. **./activate-dropzone.sh**
   Note: The command prompt will not return after the last command, indicating that the dropzone is active.
3. Using Cockpit Navigator, navigate to the **/home/admin/pacedge-fmgr/dropzone** and drag and drop file **PEv240UpdateGMUtil.tgz** from your computer to the Navigator window. Wait until archives are transferred and processed. Depending on the hardware, this step will take up to 1 minute.
   Note: While this step is in process, periodically refresh the Navigator. Wait until all files disappear from the dropzone folder, indicating that processing is complete.
4. Go back to the Terminal in Cockpit and use the **ctrl+c** key combination stop the dropzone agent.

10. Go to Navigator, navigate to folder */home/admin/pacedge-fmgr/playbooks* and open file: **PEv240prepare.playbook.yml**. Change both **adminpasswd** and **sudopasswd** to the passwords you have configured for the Linux/Cockpit. Save the file.

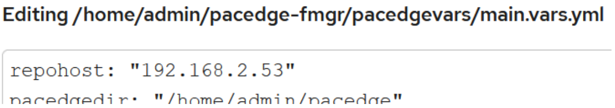


Editing /home/admin/pacedge-fmgr/playbooks/PEv240prepare.playbook.yml

```
vars:
  # can also be set in inventory
  ansible_connection: ssh
  ansible_user: admin
  ansible_ssh_pass: edgestack
  ansible_sudo_pass: edgestack
```

5. Make sure your remote devices are listed in the group file*: **/home/admin/pacedge-fmgr/hosts.ini***, refer to **4.6.1 Configuring Device Groups**
   Note: please make sure that the very first line in the host.ini file is as follows:
   localhost ansible_host=127.0.0.1 ansible_connection=local ansible_user=admin

6. Make sure your Group Manager IP address is listed in *the /home/admin/pacedge-fmgr/pacedgevars/main.vars.yml* file.



Editing /home/admin/pacedge-fmgr/pacedgevars/main.vars.yml

```
repohost: "192.168.2.53"
pacedgedir: "/home/admin/pacedge"
```

7. Go to Terminal and issue following commands:
   a. **cd /home/admin/pacedge-fmgr**
   b. **./playbooks/PEv240UpdateGMUtil.startscript.sh my_group_name**
      Note: replace my_group_name with your group name
8. Once remote devices have been updated, update the Group Manager device itself by issuing command:
   a. **./playbooks/PEv240UpdateGMUtil.startscript.sh localhost**
9. Reboot each device

9.2 Upgrading to PACEdge v2.3.0

Starting with PACEdge v2.3.0, which has a Group Manager functionality, PACEdge updates to the latest version can be performed in one of the following ways:

1. Using PACEdge v2.3.0 device and the Group Manager functionality, multiple PACEdge v2.2.x devices can be remotely upgraded to version v2.3.0
2. Using the Cockpit Navigator, an individual PACEdge v2.2.x can be remotely upgraded to version v2.3.0
3. Using the USB storage device, an individual PACEdge v2.2.x can be upgraded to version v2.3.0
4. Using the USB storage device, an individual PACEdge device (version v2.0 or later) can be upgraded to a fresh (factory default) PACEdge v2.3.0

All updates are self-contained and do not require an Internet connection when installing.

Note that due to disc size limitations, RXi2-LP with 32GB SSD cannot be upgraded using the Cockpit Navigator approach.

If you choose option 4, which is upgrading to the Factory Default image, the existing user data will not be preserved. However, if you choose any of the other options, the Node-RED flows, Grafana dashboards, databases and additional Linux libraries will remain intact. Any modificatiosn you may have made to the configuration files or the docker-compose.yml file will be back up in the path **/home/admin/pacedge.pe22update**, with the same folder structure as the original unit, so you can easily access and transfer your files to the upgraded unit.

The recommended upgrade path is:

1. If the existing PACEdge installation does not have data that needs to be preserved or if that data is backed up as described in Section 7: Saving and Restoring User Data, If upgrading from PACEdge versions older than v2.2.0, it is strongly recommended to use option 4 above and perform the Factory Default restore directly into PACEdge version v2.3.0

2. Otherwise, use option 2 and perform the upgrade using Cockpit Navigator. Please ensure the necessary SSD space is available before beginning this method of update.

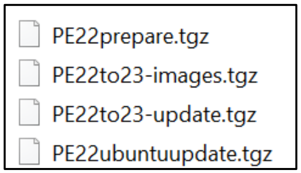
9.2.1 Upgrading via Group Manager

This upgrade procedure is applicable to all hardware platforms, including RXi2-BP, RXi2-LP, CPL410 and CPE400, and IPC 2010. PACEdge devices to be upgraded need to be of version v2.2.x or later. This upgrade process will attempt to preserve user data (Node-RED flows, Grafana dashboards, databases, etc). To perform an upgrade using a Group Manager, at least one PACEdge device version v2.3.0 or later with Group Manager license is required.

From the Emerson Software Downloads site, download **PACEdge Upgrade to v2.3 via Group Manager** (file name: **PEv230UpdateGMUtil.zip**). Then follow the steps below to perform a backup:

1. Unzip the archive locally on your computer, resulting files will be following:

Table 9 File Archives for Update using Group Manager

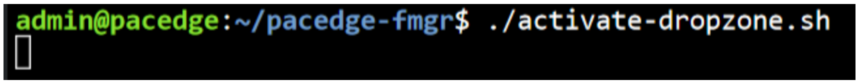


PE22prepare.tgz
PE22to23-images.tgz
PE22to23-update.tgz
PE22ubuntuupdate.tgz

2. Connect to a PACEdge device with Group Manager functionality, go to Cockpit-> Terminal and issue the command:
   a. **cd /home/admin/pacedge-fmgr**
3. Next, start Dropzone utility by issuing the following command:
   a. **./activate-dropzone.sh**

   Note: At this point Terminal will remain busy, ready to printout status information:

Figure 215: Printout Status Information

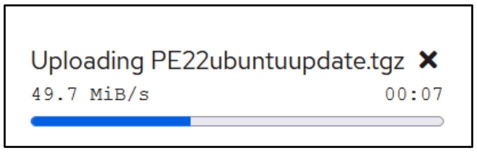


```
admin@pacedge:~/pacedge-fmgr$ ./activate-dropzone.sh
```

4. Go to Cockpit->Navigator and navigate to folder: **/home/admin/pacedge-fmgr/dropzone**
5. Upload the archive files to the Group Manager by performing following steps:
   a. Drag and drop one of the four archive files from your computer to the Navigator window.

c. Wait until the file is uploaded. Note, some files are large and will take few seconds to upload. You can see upload progress at the bottom of the screen:

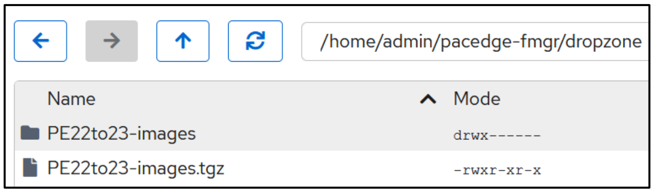
Figure 216: Uploading the file



Uploading PE22ubuntuupdate.tgz ✖
49.7 MiB/s 00:07

d. Once the file has been uploaded, it will be processed and, in the Navigator window, one or more new files will appear. Press the Navigator refresh button every few seconds to get an updated view and wait until these files disappear and folder is empty again.

Figure 217: Updated File View



/home/admin/pacedge-fmgr/dropzone

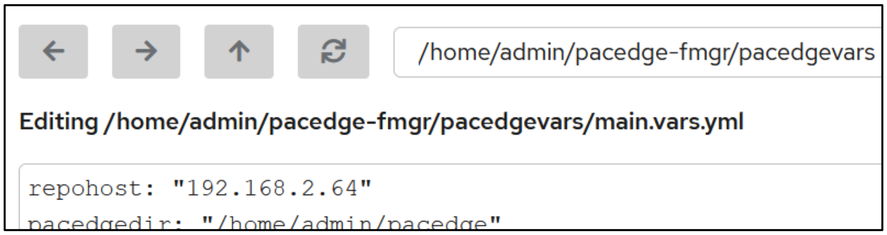
Name	Mode
PE22to23-images	drwx------
PE22to23-images.tgz	-rwxr-xr-x

e. Repeat these steps for each of the four archive files. Once finished, go back to Terminal window and press CTRL+c keys to stop Dropzone utility

6. Configure Group Manager as follows:

   a. Using Cockpit->Navigator, open file: **/home/admin/pacedge-fmgr/pacedgevars/main.vars.yml** and enter Group Manager’s IP address into line “repohost”.
   Save the file after editing.

Figure 218: Enter the Group Manager’s IP Address



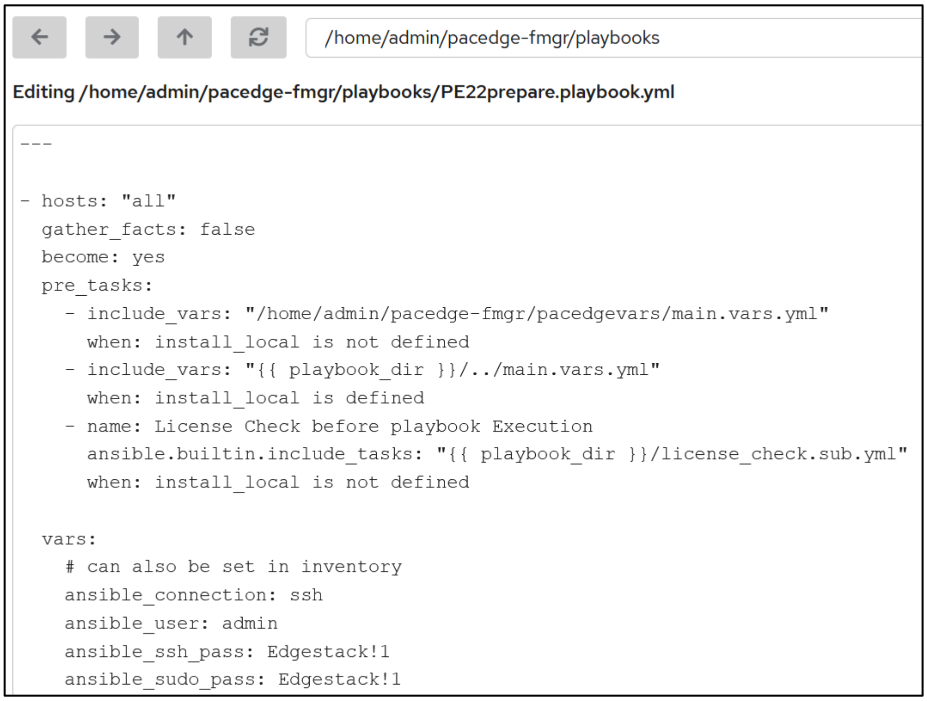
/home/admin/pacedge-fmgr/pacedgevars

Editing /home/admin/pacedge-fmgr/pacedgevars/main.vars.yml

```
repohost: "192.168.2.64"
pacedgedir: "/home/admin/pacedge"
```

b. Using Cockpit->Navigator, open file: **/home/admin/pacedge-fmgr/playbooks/PE22prepare.playbook.yml** and enter administrator password for your target PACEdge device as “**ansible_ssh_pass**” and your local administrator password for Group Manager as “**ansible_sudo_pass**”.
Note, since these passwords are stored in clear text, it is highly recommended to delete them after the device provisioning step.
Save the file after editing.

Figure 219: Enter the Admin Password



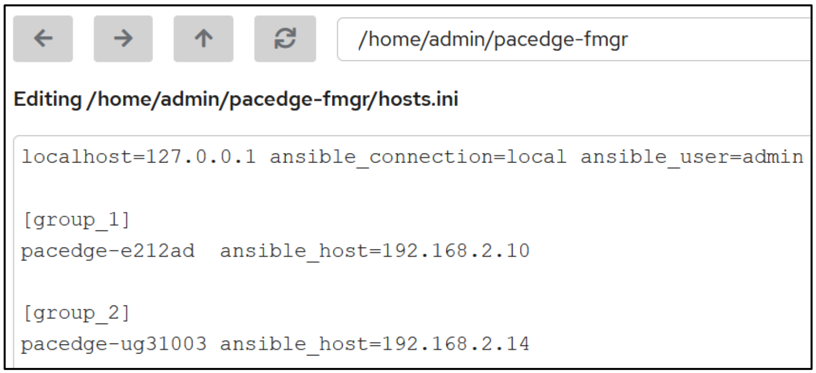
/home/admin/pacedge-fmgr/playbooks

Editing /home/admin/pacedge-fmgr/playbooks/PE22prepare.playbook.yml

```
---

- hosts: "all"
  gather_facts: false
  become: yes
  pre_tasks:
    - include_vars: "/home/admin/pacedge-fmgr/pacedgevars/main.vars.yml"
      when: install_local is not defined
    - include_vars: "{{ playbook_dir }}/../main.vars.yml"
      when: install_local is defined
    - name: License Check before playbook Execution
      ansible.builtin.include_tasks: "{{ playbook_dir }}/license_check.sub.yml"
      when: install_local is not defined

  vars:
    # can also be set in inventory
    ansible_connection: ssh
    ansible_user: admin
    ansible_ssh_pass: Edgestack!1
    ansible_sudo_pass: Edgestack!1
```

c. Setup your group of devices to be upgraded. To do so, using Cockpit->Navigator, open file: **/home/admin/pacedge-fmgr/hosts.ini**. Enter remote devices hostname as well as IP address as shown below. Optionally, you can split devices into groups by providing the group name in the square brackets.
Save the file after editing

Figure 220:Set up Group of Devices



/home/admin/pacedge-fmgr

Editing /home/admin/pacedge-fmgr/hosts.ini

```
localhost=127.0.0.1 ansible_connection=local ansible_user=admin

[group_1]
pacedge-e212ad  ansible_host=192.168.2.10

[group_2]
pacedge-ug31003 ansible_host=192.168.2.14
```

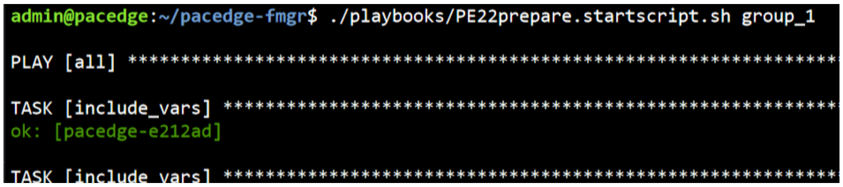
3. Perform PACEdge Upgrade, which consists of four steps:

   a. Prepare target PACEdge devices.
      This step updates older PACEdge devices so that they support Group Management utility. Using Cockpit->Navigator issue commands:

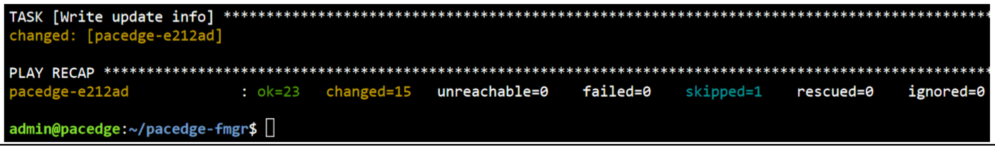
      i. **cd /home/admin/pacedge-fmgr**

      ii. **./playbooks/PE22prepare.startscript.sh group_1**

Figure 221: Prepare Target PACEdge devices



```
admin@pacedge:~/pacedge-fmgr$ ./playbooks/PE22prepare.startscript.sh group_1

PLAY [all] *******************************************************************

TASK [include_vars] **********************************************************
ok: [pacedge-e212ad]

TASK [include_vars] **********************************************************
```



```
TASK [Write update info] ****************************************************************************************
changed: [pacedge-e212ad]

PLAY RECAP ******************************************************************************************************
pacedge-e212ad             : ok=23   changed=15   unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

admin@pacedge:~/pacedge-fmgr$ 
```

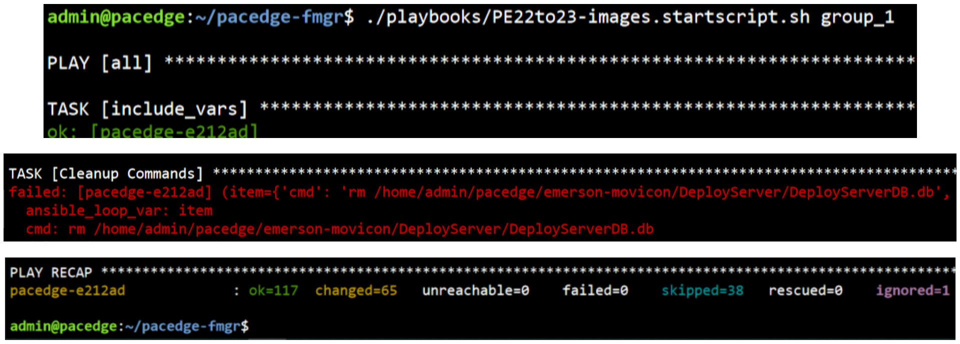
      iii. At the end of this step, when successful, output will show: **failed=0**

   b. Transfer the files.
      This step will transfer new files to target device. Since some files are large, this step will take few minutes to complete.
      Note: in this step some error statements might occur when trying to remove files that do not exist. Please ignore them.
      Using Cockpit->Navigator issue commands:

ii. **cd /home/admin/pacedge-fmgr**

iii. **./playbooks/PE22to23-images.startscript.sh group_1**

Figure 222:Transfer the files



```
admin@pacedge:~/pacedge-fmgr$ ./playbooks/PE22to23-images.startscript.sh group_1

PLAY [all] *******************************************************************

TASK [include_vars] **********************************************************
ok: [pacedge-e212ad]
```

```
TASK [Cleanup Commands] ******************************************************************************
failed: [pacedge-e212ad] (item={'cmd': 'rm /home/admin/pacedge/emerson-movicon/DeployServer/DeployServerDB.db',
  ansible_loop_var: item
  cmd: rm /home/admin/pacedge/emerson-movicon/DeployServer/DeployServerDB.db
```

```
PLAY RECAP *******************************************************************************************
pacedge-e212ad             : ok=117  changed=65   unreachable=0    failed=0    skipped=38   rescued=0    ignored=1

admin@pacedge:~/pacedge-fmgr$
```

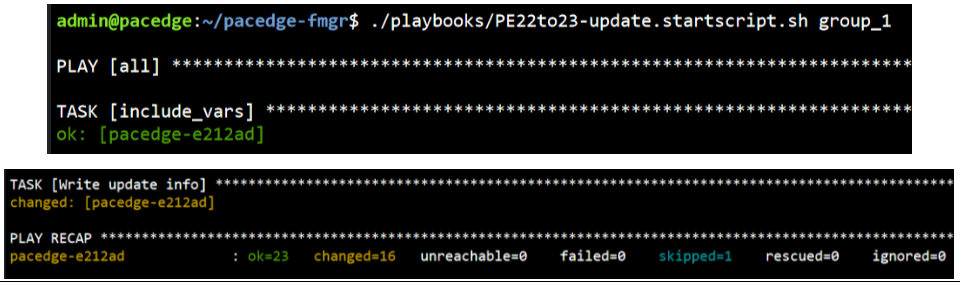
iv. At the end of this step, when successful, output will show: **failed=0**

c. Perform PACEdge upgrade.
This step will upgrade PACEdge components, with exception of host Ubuntu Linux, which will be updated in the following step.
Using Cockpit->Navigator issue commands:

i. **cd /home/admin/pacedge-fmgr**

ii. **./playbooks/PE22to23-update.startscript.sh group_1**

Figure 223: Perform PACEdge upgrade



```
admin@pacedge:~/pacedge-fmgr$ ./playbooks/PE22to23-update.startscript.sh group_1

PLAY [all] *******************************************************************

TASK [include_vars] **********************************************************
ok: [pacedge-e212ad]
```

```
TASK [Write update info] *****************************************************************************
changed: [pacedge-e212ad]

PLAY RECAP *******************************************************************************************
pacedge-e212ad             : ok=23   changed=16   unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

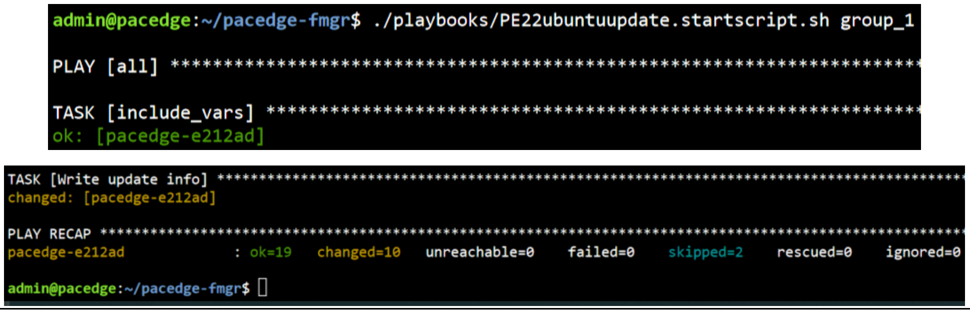
iii. At the end of this step, when successful, output will show: **failed=0**

d. Perform Linux Operating System update.
This step will update the host Linux operating system
Note: depending on the HW performance, this step will take some time (10min or so), especially the Ubuntu update portion.
Using Cockpit->Navigator issue commands:

i. **cd /home/admin/pacedge-fmgr**

ii. **./playbooks/PE22ubuntuupdate.startscript.sh group_1**

Figure 224: Perform Linux OS Update



```
admin@pacedge:~/pacedge-fmgr$ ./playbooks/PE22ubuntuupdate.startscript.sh group_1

PLAY [all] ********************************************************************

TASK [include_vars] ***********************************************************
ok: [pacedge-e212ad]
```

```
TASK [Write update info] ******************************************************************************************
changed: [pacedge-e212ad]

PLAY RECAP ********************************************************************************************************
pacedge-e212ad             : ok=19   changed=10   unreachable=0    failed=0    skipped=2    rescued=0    ignored=0

admin@pacedge:~/pacedge-fmgr$ 
```

iii. At the end of this step, when successful, output will show: **failed=0**

e. At this point PACEdge update to the new version has been completed. Please reboot the target system by logging into it and using Cockpit to reboot. It is expected that Node-RED flows, Grafana dashboards, databases and any additional Linux applications will continue to function as before, but any custom changes to configuration files, such as telegraf.conf or docker-compose.yml will be overwritten. To help backport these changes to the new version, original files are stored in the folder: **/home/admin/pacedge.pe22update**, which preserves the same folder structure as original unit, to help user navigate and find required files.

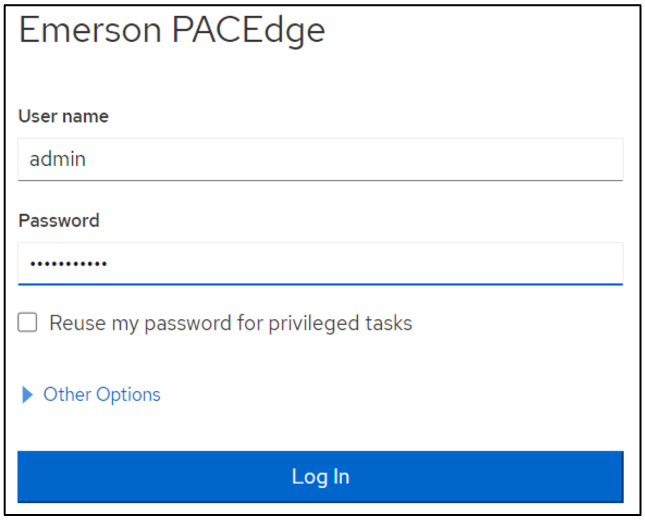
9.2.2 Upgrading via Navigator

Using the Cockpit Navigator an individual PACEdge v2.2.x can be remotely upgraded to version v2.3.0. This upgrade procedure is applicable to hardware platforms that have 60GB SSD or larger. When upgrading RXi2-LP with 32GB SSD please use either Group Manager or USB storage device upgrade path.

From the Emerson Software Downloads site, download **PACEdge Upgrade to v2.3 via USB Navigator** (file name: **PEv230UpdateUSBUtil.zip**). Then follow the steps below to perform a backup:

1. Log into Cockpit as admin user but keep the button “Reuse my password for privileged tasks” unchecked.

Figure 225: Cockpit Login Dialogue



Emerson PACEdge

User name

admin

Password

•••••••••••

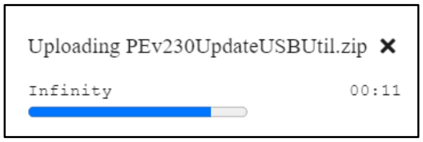
□ Reuse my password for privileged tasks

2. Using Cockpit->Navigator:

   a. Navigate to /home/admin folder, right click with the mouse and create new directory: **update_23**
   Note: check to make sure that **update_23** folder owner is **admin**. If you see owner as a **root**, then you need to log out of Cockpit and log in again as described in step 1 above.

   b. Using Navigator, drag and drop downloaded archive to **/home/admin/update_23** directory. Note, the archive is quite large and will take some time to download. Once the upload indicator at the bottom of the screen is finished, please click on Navigator refresh button and make sure that uploaded archive is listed. You might have to wait and try refreshing again as it takes time

Figure 226: Upload Archive



Uploading PEv230UpdateUSBUtil.zip ✕

Infinity 00:11

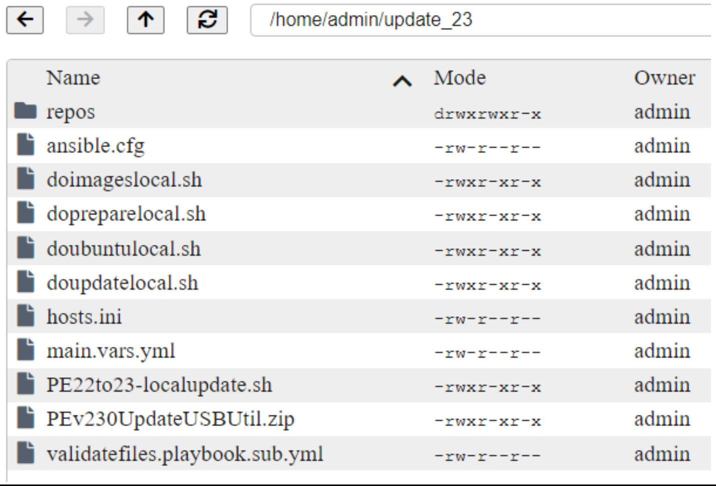
3. Using Cockpit->Terminal, issue following commands:

   a. cd /home/admin/update_23

   b. unzip PEv230UpdateUSBUtil.zip

Using Navigator, go to folder /home/admin/update_23. You should see the file structure as follows:

Figure 227: Locate Update_23

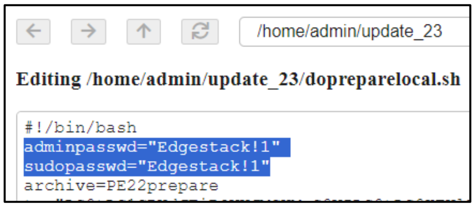


/home/admin/update_23

Name	Mode	Owner
repos	drwxrwxr-x	admin
ansible.cfg	-rw-r--r--	admin
doimageslocal.sh	-rwxr-xr-x	admin
dopreparelocal.sh	-rwxr-xr-x	admin
doubuntulocal.sh	-rwxr-xr-x	admin
doupdatelocal.sh	-rwxr-xr-x	admin
hosts.ini	-rw-r--r--	admin
main.vars.yml	-rw-r--r--	admin
PE22to23-localupdate.sh	-rwxr-xr-x	admin
PEv230UpdateUSBUtil.zip	-rwxr-xr-x	admin
validatefiles.playbook.sub.yml	-rw-r--r--	admin

4. Open file: **dopreparelocal.sh** and enter devices admin password as “adminpasswd” and as “sudopasswd”:

Figure 228: Open dopreparelocal.sh



/home/admin/update_23

Editing /home/admin/update_23/dopreparelocal.sh

```
#!/bin/bash
adminpasswd="Edgestack!1"
sudopasswd="Edgestack!1"
archive=PE22prepare
```

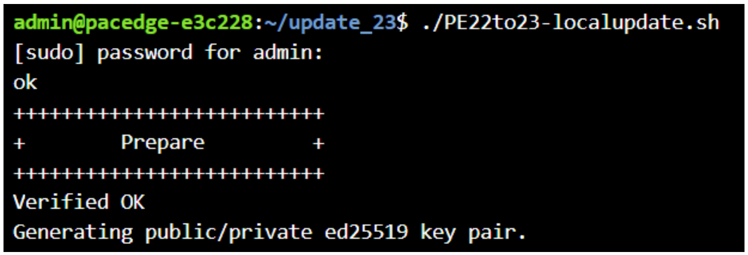
   a. Save the file

5. Using Terminal, issue following commands:

   a. cd /home/admin/update_23

   b. ./PE22to23-localupdate.sh

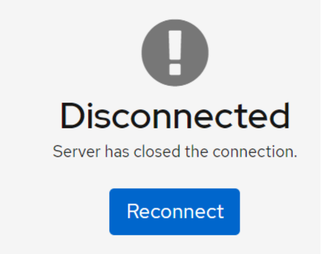
Figure 229: Update_23



```
admin@pacedge-e3c228:~/update_23$ ./PE22to23-localupdate.sh
[sudo] password for admin:
ok
++++++++++++++++++++++++++
+        Prepare         +
++++++++++++++++++++++++++
Verified OK
Generating public/private ed25519 key pair.
```

d. The update will continue until the Cockpit itself has to be updated. At that point connection via web browser will be lost.

Figure 230: Disconnected



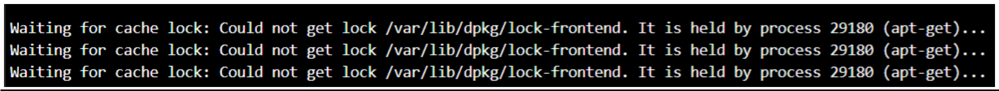
Disconnected

Server has closed the connection.

e. Click on **Reconnect**, log in as administrator. Go to Terminal and again issue the two commands:
cd /home/admin/update_23
./PE22to23-localupdate.sh

f. For some time you will see following messages, indicating that PACEdge is performing upgrade tasks in the background. Just wait (depending on the hardware up to 15 min).

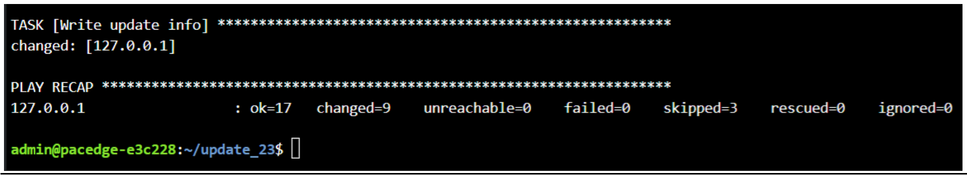
Figure 231: Update Can Take Up to 15 Minutes



```
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is held by process 29180 (apt-get)...
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is held by process 29180 (apt-get)...
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is held by process 29180 (apt-get)...
```

g. Once the update tasks are finished the script will resume and run to completion

Figure 232: Normal View While Script Runs to Completion



```
TASK [Write update info] *******************************************************
changed: [127.0.0.1]

PLAY RECAP *********************************************************************
127.0.0.1                  : ok=17   changed=9    unreachable=0    failed=0    skipped=3    rescued=0    ignored=0

admin@pacedge-e3c228:~/update_23$
```

i. At the end of this step, when successful, output will show: **failed=0**

h. At this point reboot the unit by issuing:

i. **sudo reboot**

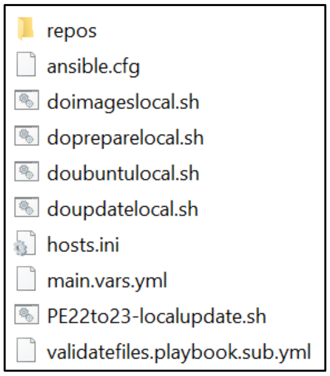
9.3.1 Upgrading via USB Storage Device

Using the USB storage device method an individual PACEdge v2.2.x can be locally upgraded to version v2.3.0. This upgrade procedure is applicable to all hardware platforms, especially one that have limited disk space, such as RXi2-LP with 32GB SSD.

From the Emerson Software Downloads site, download **PACEdge Upgrade to v2.3 via USB Navigator** (file name: **PEv230UpdateUSBUtil.zip**). Then follow the steps below to perform a backup:

1. Unzip the downloaded archive locally on your PC. The expected file structure will look like this:

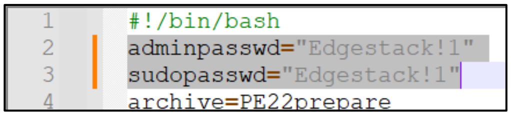
Table 10 Files for Update via USB storage device



```
repos
ansible.cfg
doimageslocal.sh
dopreparelocal.sh
doubuntulocal.sh
doupdatelocal.sh
hosts.ini
main.vars.yml
PE22to23-localupdate.sh
validatefiles.playbook.sub.yml
```

2. Copy all the files onto empty USB storage device, root directory.
3. Using editor on your PC, such as Notepad in Windows, open file: **dopreparelocal.sh** and enter devices admin password as “adminpasswd” and as “sudopasswd”.

Figure 233: Edit Passwords



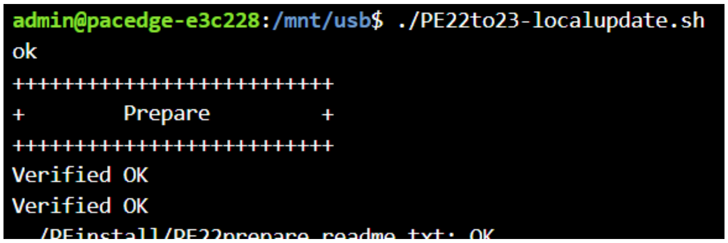
```
#!/bin/bash
adminpasswd="Edgestack!1"
sudopasswd="Edgestack!1"
archive=PE22prepare
```

   a. Save the file.

4. Plug the USB storage device into the PACEdge device, connect via Cockpit and mount USB storage device into path **/mnt/usb** with owner being **admin** as described in section: 10.1 Mounting USB .
   Note:Uuse Custom mounting options to make owner **admin**, as described in the mounting procedure.

6. Using Cockpit->Terminal, issue following commands:
    a. **cd /mnt/usb**
    b. **./PE22to23-localupdate.sh**

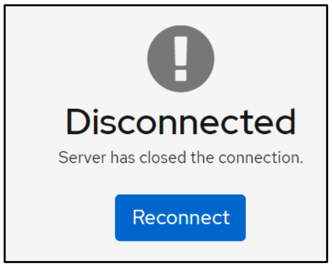
Figure 234: Update



```
admin@pacedge-e3c228:/mnt/usb$ ./PE22to23-localupdate.sh
ok
+++++++++++++++++++++++++
+        Prepare        +
+++++++++++++++++++++++++
Verified OK
Verified OK
  /PEinstall/PE22prepare_readme.txt: OK
```

7. The update will continue until the Cockpit itself has to be updated. At that point connection via web browser will be lost.

Figure 235: Connection to Browser Lost

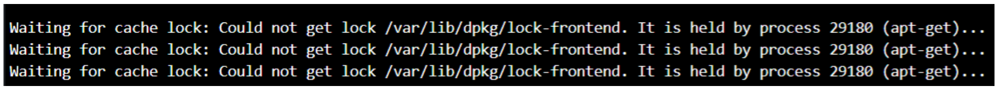


Disconnected

Server has closed the connection.

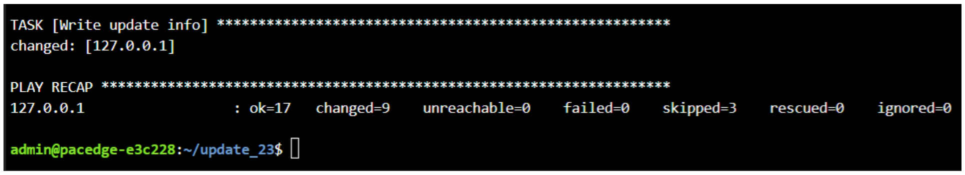
8. Click on **Reconnect**, log in as admin user. Go to Terminal and again issue the two commands:
    a. **cd /mnt/usb**
    b. **./PE22to23-localupdate.sh**

9. For some time you will see following messages, indicating that PACEdge is performing upgrade tasks in the background. Just wait (depending on the hardware up to 15 min).



```
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is held by process 29180 (apt-get)...
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is held by process 29180 (apt-get)...
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is held by process 29180 (apt-get)...
```

10. Once the update tasks are finished the script will resume and run to completion



```
TASK [Write update info] ********************************************************
changed: [127.0.0.1]

PLAY RECAP **********************************************************************
127.0.0.1                  : ok=17   changed=9    unreachable=0    failed=0    skipped=3    rescued=0    ignored=0

admin@pacedge-e3c228:~/update_23$ 
```

    i. At the end of this step, when successful, output will show: **failed=0**

11. At this point reboot the unit by issuing:
    a. **sudo reboot**

9.3.2 Upgrading via USB storage device to the Factory Default State

Using the USB storage device method an individual PACEdge of version v2.0 or newer can be upgraded to the clean Factory Default state in PACEdge version v2.3.0.

Note that this upgrade method will erase all your user data on the unit.

Depending on the hardware being updated:

9.4 For the RXi2-BP, RXi2-LP follow same steps as in section: 8.2

1. PACEdge Software Restore/Recovery on RXi2-BP, RXi2-LP IPCs and select a factory default image to restore to (**PACEdge-V230.install**).

2. For the CPL410, CPE400, from the Emerson Software Downloads site, download **PACEdge 2.3 Install CPL410 CPE400** file (file name: **PEv230InstUtilCPL.zip**). Then follow the steps below to perform an upgrade:

   a. Get an empty, min 32 GB size USB storage device. Make sure it is empty, as conflicts might arise.

Note: if using old USB storage devices, one might experience a problem in which a device may not boot from the USB storage device. This is typically due to the file system changes and boot record configuration on the USB storage device. The workaround, in this case, is either: 1) use a brand new USB storage device, 2) download from the Internet one of Linux installation ISO images, use a Windows utility such as Rufus, to make a bootable USB storage device, then delete all the files from the USB storage device and proceed with next steps.

   b. Unzip the archive locally on the PC and copy all the files to the root directory of the USB storage device.
      NOTE: when copying the files, make sure all the files including the EFI folder are in the root directory of the USB storage device.

   c. Plug the USB storage device into the CPL410 or CPE400

   d. Power up the unit and wait until upgrade completes.

   e. Cycle the power to complete the upgrade process.

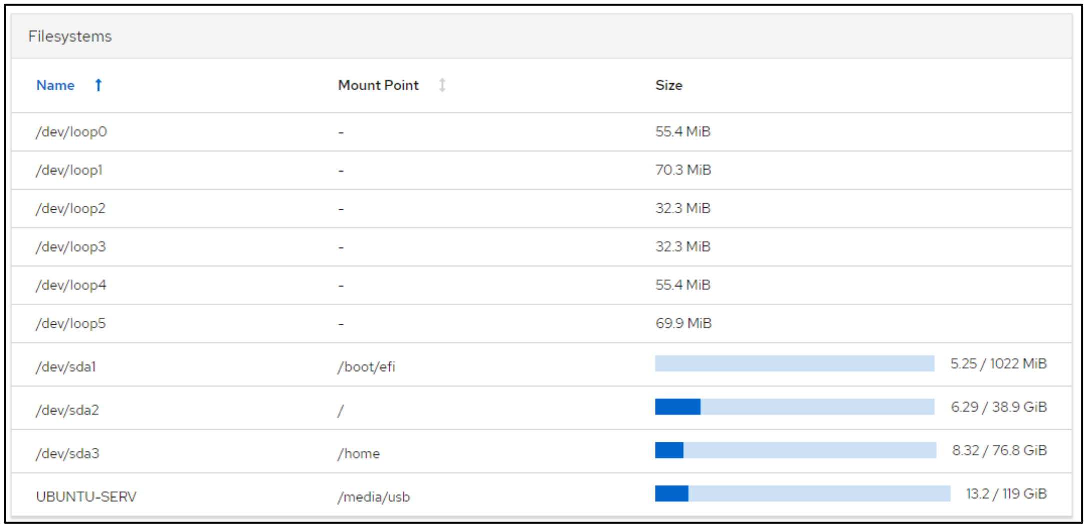
Section 10: Utilities and Troubleshooting

10.1 Mounting USB Storage Device

PACEdge can read and write data from/to a USB storage device. However, the USB storage device must first be properly mounted in Linux OS. The procedure to mount a USB storage device is the same for RXi2-BP, RXi2-LP, CPL410, and CPE400; the only difference between devices will be the disc partitioning scheme that you see in Navigator; ignore inconsistencies with the images shown below.
To mount the USB storage device, please follow the steps below:

1. Insert the USB storage device into any USB port.
2. Login into Cockpit as admin.
3. Go to the Storage tab on the left side of the screen, look for the USB storage device in the list of Filesystems (typically at the very bottom of the list, UBUNTU-SERV in the example below), and click on it.

Figure 236: List of Filesystems (CPL410, CPE400)

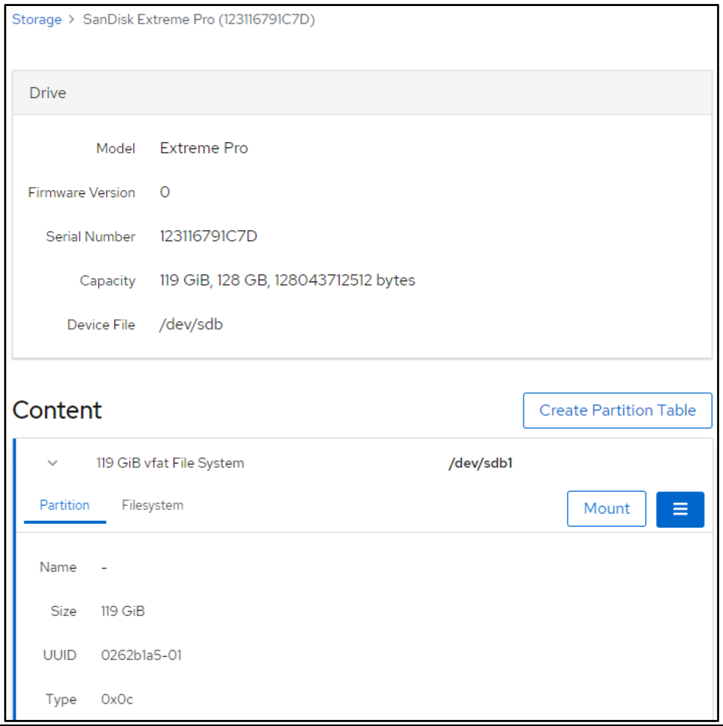


Filesystems

Name ↑	Mount Point ↕	Size
/dev/loop0	-	55.4 MiB
/dev/loop1	-	70.3 MiB
/dev/loop2	-	32.3 MiB
/dev/loop3	-	32.3 MiB
/dev/loop4	-	55.4 MiB
/dev/loop5	-	69.9 MiB
/dev/sda1	/boot/efi	5.25 / 1022 MiB
/dev/sda2	/	6.29 / 38.9 GiB
/dev/sda3	/home	8.32 / 76.8 GiB
UBUNTU-SERV	/media/usb	13.2 / 119 GiB

4. On the next page, click to expand the view of the Content and then click on the **Mount** button on the right side of the screen.

Figure 237: Mount the Device



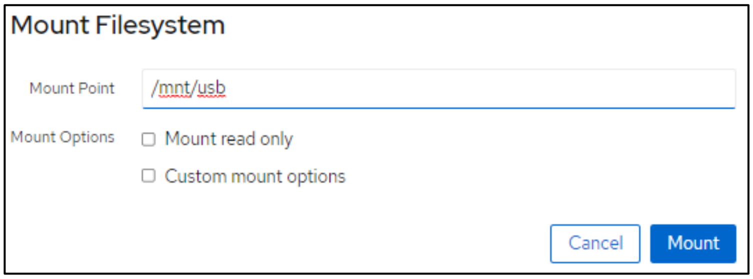
Model: Extreme Pro
Firmware Version: 0
Serial Number: 123116791C7D
Capacity: 119 GiB, 128 GB, 128043712512 bytes
Device File: /dev/sdb

119 GiB vfat File System /dev/sdb1

Name: -
Size: 119 GiB
UUID: 0262b1a5-01
Type: 0x0c

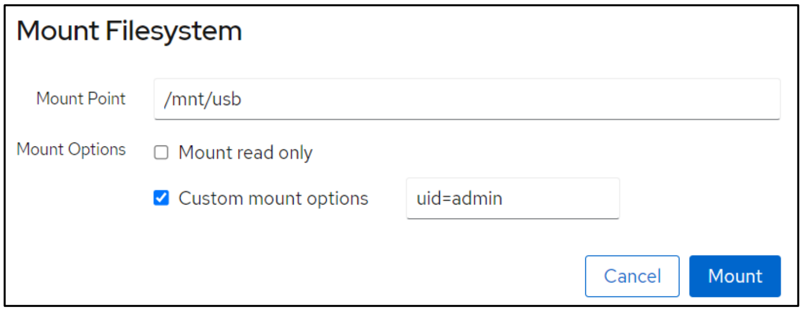
5. In the open dialogue, enter Mounting Point**: /media/usb** or **/mnt/usb** and click on the **Mount** button.
NOTE: By default, the owner of the USB storage device will be root. If specific instruction procedure asks to mount USB storage device with owner admin, please check the **Custom mount options** box and enter uid=admin as shown below.

Figure 238: Mount the USB storage device with owner root



Mount Point: /mnt/usb
Mount Options: □ Mount read only
□ Custom mount options

Figure 239: Mount the USB storage device with owner admin



Mount Filesystem

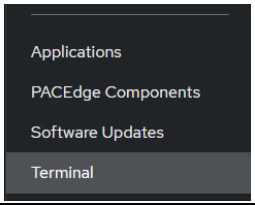
Mount Point: /mnt/usb

Mount Options: □ Mount read only

☑ Custom mount options: uid=admin

6. At this point, the USB storage device is mounted and accessible. To see the files, you can use Navigator or go to the Cockpit->Terminal screen and enter the following commands:

Figure 240: Cockpit's Terminal

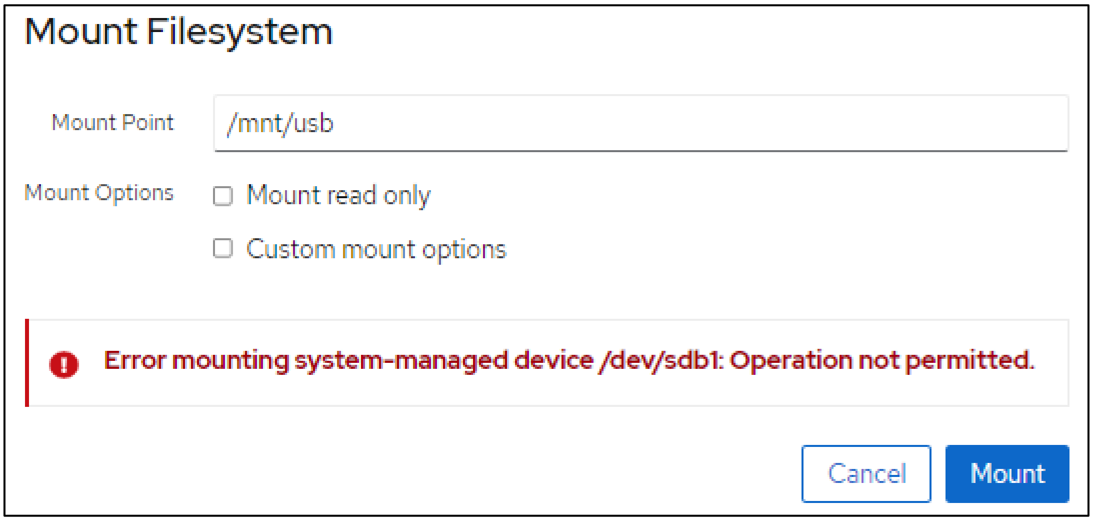


a. **`cd /mnt/usb/`**

b. **ll**

Note: If the USB storage device has not been properly un-mounted in the past, it might lead to an error

Figure 241: Mount File System



Mount Filesystem

Mount Point: /mnt/usb

Mount Options: □ Mount read only

□ Custom mount options

Error mounting system-managed device /dev/sdb1: Operation not permitted.

In such case, entries in the/etc/fstab file need to be cleaned up as follows:

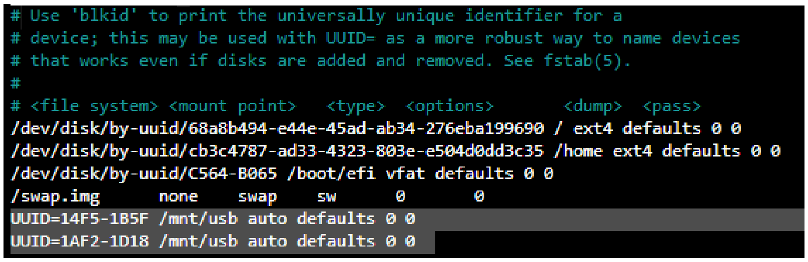
a. Log in as **admin** into Cockpit

b. Go to Terminal

c. Type: **sudo nano /etc/fstab**

d. Using arrow keys move your cursor behind the last character and delete all lines that start with UUID at the end of the file (last two lines in the given an example):

Figure 242: Example



```
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
/dev/disk/by-uuid/68a8b494-e44e-45ad-ab34-276eba199690 / ext4 defaults 0 0
/dev/disk/by-uuid/cb3c4787-ad33-4323-803e-e504d0dd3c35 /home ext4 defaults 0 0
/dev/disk/by-uuid/C564-B065 /boot/efi vfat defaults 0 0
/swap.img      none    swap    sw      0       0
UUID=14F5-1B5F /mnt/usb auto defaults 0 0
UUID=1AF2-1D18 /mnt/usb auto defaults 0 0
```

e. Press **CTRL+X** key combination, then the **y** key to confirm changes

f. Try to mount the USB storage device again.

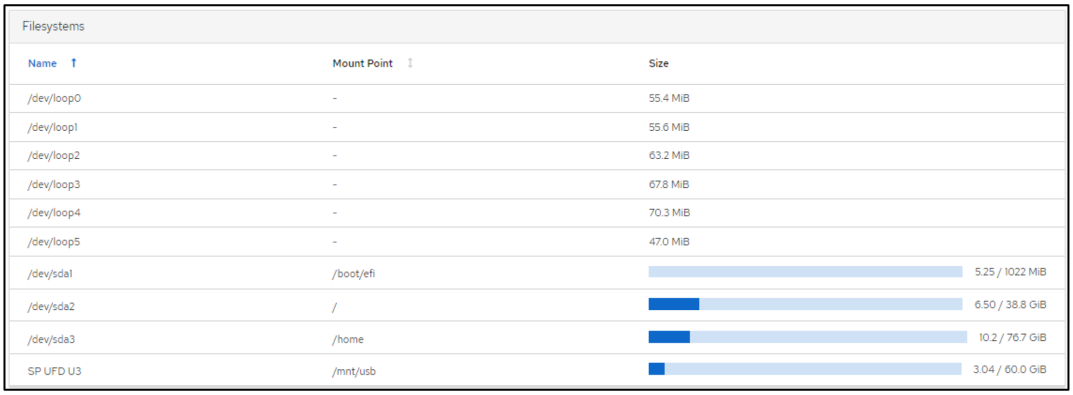
10.2 Un-Mounting USB Storage Device

In Linux operating system, it is important to properly un-mount the USB storage device before its removal.

To un-mount USB storage device:

1. In Cockpit, go to the Storage tab.
2. Click on the line which represents the USB storage device. One way to identify it is by Mount Point being **/mnt/usb** or **/media/usb.**

Figure 243: Select Storage to Unmount

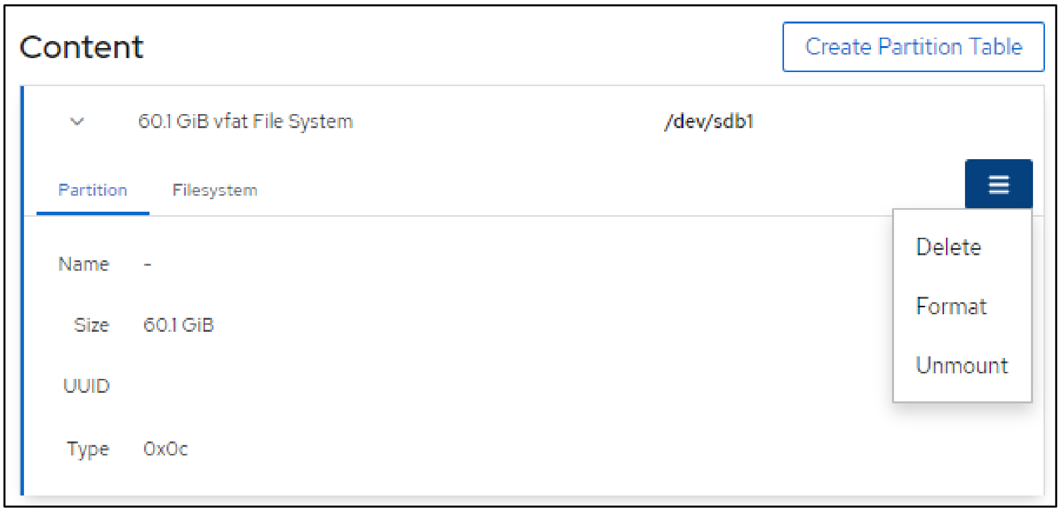


Filesystems

Name ↑	Mount Point ↕	Size
/dev/loop0	-	55.4 MiB
/dev/loop1	-	55.6 MiB
/dev/loop2	-	63.2 MiB
/dev/loop3	-	67.8 MiB
/dev/loop4	-	70.3 MiB
/dev/loop5	-	47.0 MiB
/dev/sda1	/boot/efi	5.25 / 1022 MiB
/dev/sda2	/	6.50 / 38.8 GiB
/dev/sda3	/home	10.2 / 76.7 GiB
SP UFD U3	/mnt/usb	3.04 / 60.0 GiB

3. Next, click on the disc within the Content window to open more details
   Then, click on the three lines to open options and then click on **Unmount.**

Figure 244: Unmount Storage



Content

60.1 GiB vfat File System /dev/sdb1

Name -

Size 60.1 GiB

UUID

Type 0x0c

10.3 Serial RS232 Cable for IPC 2010

On IPC 2010, to gain access to the system, utilize the serial console port by connecting it to the host PC's RS232 interface using a null modem cable. Once connected, the host can establish a connection by running a terminal emulator such as HyperTerminal or minicom. The default serial port parameters are **115200, 8N1**

The connector is RJ-45 style with pinout as defined in EIA/TIA-561, so that off-the-shelf adapters to DB9 or USB can be used.

The pinout for RJ-45 (8P8C) connector is defined as follows:

Table 11 IPC 2010 Serial Cable Pinout

Signal	RJ-45 Pin
GND	4
RxD	5
TxD	6

10.5 Difficulties Accessing PACEdge Components, Erratic Behavior

- It is always a good first step to reboot the hardware.
- If you can log in, go to Cockpit and analyze the Logs. Note that logs contain several warnings and notices about different services as part of normal operation, so search for the messages related to the specific problem.
- If you have a problem accessing a specific PACEdge application, say Node-RED, Grafana, or InfluxDB, either login into Cockpit, go to Docker Containers, or log into Portainer and check the status of the container that hosts the specific application. Clues to look for:
    - Check if the container is not continuously restarting. You can see the last start time in the statement, such as: "Up since Today xxx."
    - In Portainer, look into the log file specific to each container for further clues.

For technical support, it is helpful to provide an exported Syslog file. To do that, please log in via a terminal window and execute **journalctl > log**.

10.6 PACEdge Files

On Emerson IPCs with preinstalled PACEdge, you will find the PACEdge-Files in the Admin's home directory (***/home/admin/pacedge***). Expert users can use these files to adapt and modify the PACEdge Docker environment to their needs and stop and start the PACEdge system via the docker-compose command (see below). Normally there is no need to use these files.

10.6.1 Installation Test

PACEdge comes with a script to test if PACEdge installation is working as expected: tst_inst.sh. To execute this script, please change into folder: /home/admin/pacedge/ and run the following command:

./tst_inst.sh -u admin -p your-admin-password

This script should produce an output similar to the following:

```
admin@pacedge:~/pacedge$ ./tst_inst.sh -u admin -p edgestacka
The Parameter IP address was set to: 127.0.0.1
The parameter user was set to: admin
The Parameter passwd was set to: edgestacka
start: OK

Wait 30 sec to finish the nodered start process

Check if all packages are installed
```

```
apache2-utils : OK


Check if all images are available

        Container                       Repository
Tag           Image Id       Size

----------------------------------------------------------------------
-------------------------------------

Emerson-aes-daemon          iiot.aventics.com/images/aes-daemon
1.0.9-edgestack   eb2946d2bf57   182.7 MB

emerson-chronograf          chronograf
1.8.8             2f07c735db5e   194.8 MB

emerson-grafana             grafana/grafana
7.5.7             bc8c9ea5532e   201.8 MB

emerson-influxdb            influxdb
1.8.6             66f3325c5416   307.8 MB

emerson-mqtt                eclipse-mosquitto
2.0.11            c3a8baae5c3f   9.918 MB

emerson-mqtt-internal-ipc   eclipse-mosquitto
2.0.11            c3a8baae5c3f   9.918 MB

emerson-mysql               mariadb
10.5.11           8bd09b203b08   407.5 MB

emerson-nginx                                       1.21.0
87560fc16fed   536.4 MB

emerson-nodered             emerson-node-red
1.3.5             2066beab39b7   612.5 MB

emerson-php                 php                                   fpm
f8ceec02a25f   406.8 MB

emerson-portainer           portainer/portainer-ce
2.5.1             45be17a5903a   209.1 MB

emerson-progea                                     latest
a16ebaded4b0   3.713 GB

emerson-telegraf            telegraf
1.18.3            34535e6964f5   314 MB

emerson-traefik             traefik
v2.4.8            deaf4b1027ed   91.31 MB


emerson-node-red : OK

portainer/portainer : OK

traefik : OK

grafana/grafana : OK

chronograf : OK

nginx : OK

influxdb : OK

eclipse-mosquitto : OK
```

```
mariadb : OK
emerson-progea : OK

Check if all docker-compose images are available

Check if the containers are running
emerson-nodered : OK
emerson-portainer : OK
emerson-traefik : OK
emerson-grafana : OK
emerson-nginx : OK
emerson-chronograf : OK
emerson-mysql : OK
emerson-mqtt : OK
emerson-mqtt-internal-ipc : OK
emerson-influxdb : OK
emerson-progea : OK

Check if Cockpit is started and running
cockpit.socket : OK

Check if node-red palettes are installed correctly
To be done ...

 Check if WEB-interfaces are accessible
/nodered/ 1880: OK
/grafana/ 3000: OK
/chronograf/ 8888: OK
/portainer/ 9000: OK
:9090/cockpites/ 9090: OK
/emerson/eula.html 9999: OK
Number of errors 0
OK
```

Note: an early version of this script has a bug and checks for the nodered/node-red container image, which is an interim product in building a final emersonnode-red image and is removed later in the build process. Please ignore this error:

emerson-node-red : OK

nodered/node-red : ERROR

portainer/portainer : OK

10.7 Docker Commands

PACEdge is heavily based on Docker and Docker application images. Docker is a kind of lightweight virtualization allowing to group applications together in a protected self-contained environment within the Linux operating system. Setting up and configuring such a set of applications is a complex task whose description is beyond the scope of this document. Fortunately, this arrangement is already mastered by PACEdge and you only need some commands to manage PACEdge.
PACEdge uses “docker -compose” to start and configure the set of PACEdge applications. The configuration for “docker-compose” is stored in a Yaml file: **docker-compose.yml**. It comes with the installation package.

To view docker-compose.yml file and execute other docker commands please first change into directory: **/home/admin/pacedge/**

Following are the most important commands:

10.7.1 docker compose up -d

Note that with PACEdge version 2.2.0, there is no longer dash in between docker and compose words. With this command, the PACEdge environment is started. All the containers configured in the **docker-compose.yml** file are created, configured, started, and connected. Normally, this command is not needed, as the PACEdge environment is automatically started during system start. But it is used whenever the configuration has changed or the following command has shut down PACEdge:

10.7.2 docker compose down

Note that with PACEdge version 2.2.0, there is no longer dash in between docker and compose words. Use this command whenever you want to shut down PACEdge. This command will delete all containers, and data is stored within the containers, but persistent data will remain in external volumes. This command is useful if you want to exclude an application from the PACEdge configuration (see the following command).

10.8 ./createcomposefile.sh

The script createcomposefile.sh, located in the */home/admin/pacedge/*, let you select the PACEdge applications you want to run. From the factory a default set of applications is activated, but if you don´t need some of them or want to add some optional applications, you can make your selections in this semi-graphic dialogue. The dialogue will guide you through bringing down all docker containers, allowing you to add/remove docker containers, and then bringing up containers. createcomposefile.sh generates a new **docker-compose.yml** configuration file (see above).
In order to run this script in an interactive mode, use -i flag:
./ createcomposefile.sh -i

The user executing these commands must be a member of the Linux “docker” group. Preinstalled PACEdge user “admin” is a member of this group and therefore recommended for these actions.

General Contact Information

Home link: http://www.emerson.com/industrial-automation-controls

Knowledge Base: https://www.emerson.com/iac-support

Technical Support

Americas
Phone: 1-888-565-4155
1-434-214-8532 (If toll-free option is unavailable)

Customer Care (Quotes/Orders/Returns): customercare.mas@emerson.com
Technical Support: support.mas@emerson.com

Europe
Phone: +800-4444-8001
+420-225-379-328 (If toll-free option is unavailable)

Customer Care (Quotes/Orders/Returns): customercare.emea.mas@emerson.com
Technical Support: support.mas.emea@emerson.com

Asia
Phone: +86-400-842-8599
+65-3157-9591 (All other Countries)

Customer Care (Quotes/Orders/Returns): customercare.cn.mas@emerson.com
Technical Support: support.mas.apac@emerson.com

Any escalation request should be sent to mas.sfdcescalation@emerson.com

Note: If the product is purchased through an Authorized Channel Partner, please contact the seller directly for support.

Emerson reserves the right to modify or improve the designs or specifications of the products mentioned in this manual at any time without notice. Emerson does not assume responsibility for the selection, use, or maintenance of any product. Responsibility for the proper selection, use, and maintenance of any Emerson product remains solely with the purchaser.